UCGE Reports
Number 20066

Department of Geomatics Engineering

An Object Oriented Approach to Data Quality Reporting Within Geographic Information Systems

(URL: http://www.geomatics.ucalgary.ca/links/GradTheses.html)

by

John Lethaby

September 1994

UNIVERSITY OF CALGARY

THE UNIVERSITY OF CALGARY

An Object Oriented Approach To Data Quality Reporting Within Geographic Information Systems

by

John Lethaby

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF ENGINEERING

DEPARTMENT OF GEOMATICS ENGINEERING

CALGARY, ALBERTA

SEPTEMBER, 1994

ABSTRACT

Most GIS implementations do not include a realistic indication of the quality of their data. Since it is difficult to make a reliable decision without an understanding of data quality, a GIS that does not include data quality reporting can only play a limited decision support role. If GIS technology is to be used as a decision support tool, a data quality reporting function must be implemented as an integral component of the GIS.

Within this thesis, roles and characteristics associated with practical data quality reporting are presented. In addition, the relationships between error, uncertainty, accuracy and quality are reviewed. Based on this, a generic data quality reporting framework is developed that supports reporting of all classifications of data quality at multiple levels of data aggregation.

A prototype application has been developed, based on the data quality reporting framework, using an Object-Oriented application development environment. The prototype has been used to demonstrate how the framework supports practical data quality reporting. This has been accomplished through a series of examples that highlight the capabilities and flexibility of the approach provided by the prototype.

ACKNOWLEDGEMENTS

I would like to thank Dr. M. A. Chapman for providing supervision and advice in the development of this thesis.

The City of Calgary is thanked for providing the opportunity to consider data quality reporting as part of an evolving GIS implementation. Many of my colleagues at The City of Calgary must also be acknowledged for providing a practical sounding board for testing data quality reporting concepts. By having access to their varied opinions, I was able to develop a more comprehensive understanding of this complex issue. In addition, Bill Winspur and Doug Attfield supplied technical support, and Pat Drake provided business perspective.

My wife Carol cannot be thanked enough for her support and encouragement. Her enthusiastic reviews of my progress provided the incentive to complete this thesis.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES

1 INTRODUCTION

1.1 Background

The relationship between data quality and decision making has been identified and discussed extensively within current Geographic Information Systems (GIS) literature [Openshaw, 1989]. The premise is that, without a meaningful indication of the quality of data, a decision maker cannot make a decision with an acceptable level of confidence. Since GIS is ultimately a decision making support system, it follows that a mechanism that will provide decision makers with the information they need to quantify their decisions is needed. Within this thesis, the function that provides decision makers with the information they need is termed Data Quality Reporting (DQR).

Despite the clear requirement for Data Quality Reporting, this functionality does not exist within current GIS products [Burrough, 1986]. Commercial GIS vendors do not see this as a priority, as it is not a high profile feature and has limited immediate payback. As well, it is a complex problem which can not be easily addressed utilizing the technology prevailing in current commercial GISs. GIS users have not pushed the vendors for this capability, mostly because they are in the early stages of GIS development and are more focused on data collection than decision making. Typical GIS users also fail to recognize quality management as a critical part of their operational environment.

Even though GIS vendors have not confronted this problem, others have accepted the challenge. There have been several attempts at addressing DQR-related issues on two main fronts; the theoretical and the practical. The theoretical approaches have focused mainly on error propagation models and algorithms, as well as lineage tracking stratagem [Miller, 1991; Lanter, 1991]. This important research has identified many complex issues and potential approaches but has provided little that can be utilized in a production environment. For this reason, others [Garza and Foresman, 1991] have implemented simplified quality reporting schemes within their existing systems. These, though useful, lack the rigour required by complex analyses.

Most of the DQR-related work to-date has focused on general classifications of the types of data quality information that is needed to support decision making. Less attention has been paid to issues such as sources of quality information, the level at which this information should be stored, and how it should be presented. In order to provide adequate DQR, there is a need for a better understanding of the entire Data Quality Reporting environment.

1.2 The DQR Function

The Data Quality Reporting function should be considered an integral component of future GIS implementations. The roles and characteristics of DQR within a GIS are discussed in this section.

1.2.1 DQR Roles Within GIS

The DQR function has two main roles within GIS. Firstly, it must provide a means to review and evaluate sources of data, prior to actual data usage. This allows potential users of the data to determine if the data is suitable for their specific application before proceeding with spatial analyses. Within this thesis this role is termed *data review for suitability.*

The second main role of DQR is to provide *spatial analysis decision support.* This requires a DQR function that is integrated with the GIS application. It must be capable of providing decision support information based on the results of the spatial query and the quality of the data sources.

1.2.2 Practical DQR Characteristics

In order for a DQR function to be effective, it must provide a practical approach to *data review for suitability* and *spatial analysis decision support*. The characteristics considered essential to practical DQR are reviewed in this section.

Since GISs provide the potential to carry out ad-hoc queries that include data combinations not anticipated during system and data design, all potential sources of information of data quality should be made available within a GIS. If this information is available, analyses of unforeseen data combinations can be supported by DQR. This means that, in addition to generic quality information such as intended use and lineage, data quality reports need access to the status of an organization's quality assurance initiatives.

The content and format of data quality-related reports should be capable of changing over time. This is because the potential users of DQR will be effected by the state of technology as well as their own level of sophistication. Due to the nature of the technological advances associated with the competitive GIS industry and the advent of open systems development, implementors of GIS technology will be exposed to an environment that is constantly providing increased functionality. In addition, it is not anticipated that the information requirements or the sophistication level of DQR users, even within individual organizations, will advance at a uniform rate. For these reasons the DQR function must be flexible enough to allow individual users to adapt the data quality content and output to their needs.

In addition to the contents of a data quality report being flexible, they must also be meaningful to decision makers. This implies that DQR has to be capable of providing information that is understandable, and relevant to the user or an application. Without meaningful quality-related information as a base, the use of GIS as a decision support system will be limited.

The characteristics associated with practical DQR are summarized as follows:

- The DQR function should be *comprehensive* by providing access to all relevant aspects of data quality.

- The DQR format should be *flexible* so that it can support changing needs.

- The contents of the report should be *meaningful* to both data users and GIS applications.

1.3 Objectives

The main objective of this research is to develop a practical framework for Data Quality Reporting that will support evolving technologies and requirements within the emerging GIS realm.

The secondary objective is to test the DQR framework against a simulated real world scenario. This is to be accomplished by developing a prototype application for demonstration purposes. The prototype must be able to support both *data review for suitability* and *spatial analysis decision support*, and be capable of demonstrating the characteristics of practical DQR.

1.4 Approach

In order to meet the objectives defined in Section 1.3, a systematic approach to analysis, design and construction has been adopted. Although there has been an extensive review and incorporation of existing studies, the focus has consciously been on how Data Quality Reporting could be approached in the future, rather than how it is dealt with currently. A *top-down* evaluation of the problem has been employed throughout the thesis to support this position.

Despite the fact that most current commercial GISs are based on relational database technology, Object-Oriented (OO) modelling techniques have been utilized to develop the prototype used in this thesis. The reason for selecting this development environment is that leaders in GIS research [Kemp and Thearle, 1992; Egenhofer and Frank, 1992] agree that OO structures are better suited to spatial analyses than to those being used today in most commercial GISs. The fact that this methodology is rapidly emerging as the preferred choice in computer science disciplines [Weinberg et al, 1992], supports the view that future GISs will adopt objects. The structures and properties associated with OO technology promise to provide the foundation for more powerful GISs in the future.

In Chapter 2 the issues of error, accuracy, uncertainty and quality are examined. Working definitions of these terms are provided and the relationships between them are discussed. The theoretical concepts reviewed in Chapter 2 form the basis for the Chapter 3 discussion of Data Quality Reporting structure.

Chapter 3 provides a generic view of what should be included in a data quality report. This includes a review of existing DQR classification schemes and of the various levels of data aggregation that can be used to report on data quality. By combining generic DQR classification and reporting level considerations, Chapter 3 develops a conceptual DQR framework that is independent of GIS products.

The development of a prototype test application is initiated in Chapter 4. This prototype is used to demonstrate the potential of the concepts described in Chapter 3. As a first step, an analysis of the DQR problem domain is completed in Chapter 4. This includes the introduction of Object-Oriented analysis techniques and the SAIF data model [SAIF, 1991]. A sample business application is introduced to provide the test scenario and data set for the demonstration.

Chapter 5 moves from the analysis stage to system design. Further Object-Oriented concepts are introduced which result in the design phase deliverables. These include a more detailed description of the problem domain and a definition of the human interaction component.

Chapter 6 provides the details of software construction. The Smalltalk Object-Oriented programming language is explained and the techniques used to code the prototype are reviewed.

The prototype application is tested and evaluated against a sample data set in Chapter 7. The application is run under multiple scenarios for comparison purposes. The results are compiled and reviewed.

In Chapter 8 conclusions are presented based on the overall study. Recommendations on further research are made as a result of the conclusions.

2 ERROR, ACCURACY, UNCERTAINTY AND QUALITY

2.1 Introduction

Prior to carrying out a detailed analysis of Data Quality Reporting, it is important to have a clear understanding of the relationship between error, accuracy, uncertainty and data quality. The remainder of this chapter reviews relevant issues associated with these concepts, provides definitions and describes their relationships.

2.2 Error

It is generally acknowledged that within GISs there is error associated with any measure or representation of an entity or its properties [Openshaw, 1989, Veregin, 1989]. Errors are introduced through non-rigorous procedures or actions. These procedures and actions introduce error into their results, which in the case of a GIS is the data. In this section, the sources of error, the types of data error and error propagation are discussed.

2.2.1 Sources of Error

In order to appreciate the impact of error, it is necessary to understand the sources of errors that are likely to affect spatial information. In describing error sources, Hunter [1991] identified the three major error source groupings of *data collection and compilation, data processing* and *data misuse*. Hunter and Beard [1992] provided a detailed list of error sources within each of these major classifications.

Errors introduced by data collection and compilation are the result of imprecision in data capture technology and/or misinterpretation by data collection operators. The magnitude of these errors are normally estimated through redundant measurements or comparisons to independent sources of higher accuracy.

Errors that are the result of data processing are a source of error propagation (along with data collection and compilation errors). They are made up of errors introduced by base data transformations, as well as errors accumulated through strings of data manipulations. All data manipulations from simple data translations to complex spatial analysis are potential sources of these types of errors. The mathematical approach to modelling processing errors involves the development of error propagation models based on the individual data processing algorithms employed in the data manipulation. This is a complex task which is further complicated by the inaccessibility of processing algorithm documentation due to vendor confidentiality considerations. Less rigorous approaches (e.g., Epsilon Band, Fuzzy Logic/Knowledge Based) to estimating errors propagated through data processing have been investigated as substitutes to strict mathematical error models. These approaches are considered acceptable alternatives to this complicated problem [Hunter, 1992].

Data misuse errors occur when a user does not clearly understand the intended purpose of data. An example of this would be attempting to use data digitized from a small scale map source for infrastructure design purposes. It is unlikely that the level of positional accuracy associated with the digitized product would be adequate for precise design purposes. Another example, which is currency related, would be the use of addresses compiled for assessment purposes (on a yearly basis) as an emergency vehicle dispatch data source. The introduction of these errors can be due to a lack of knowledge of the data quality or a deficiency in the user's understanding of GIS concepts. It is likely that data misuse errors will become more common as the GIS user base grows. Comprehensive data quality reports and education programs are the best methods to reduce the impact of these errors.

2.2.2 Types of Error

The two main types of errors existing in GIS data are associated with the data types found in GIS data models. In making this association, data errors are considered one of two general types: spatial data errors and business data errors.

Spatial error considerations include position and spatial relationship properties. Errors associated with an entity's geometry or topology are examples of these types of errors.

Business-related errors are normally associated with the textual attributes of a GIS. These error types are a result of invalid values, misclassification, incompleteness, generalization and poor definition.

2.2.3 Propagation of Error

Error propagation modelling and management are among the most difficult challenges within spatial information systems research and development. The challenge is largely due to the impact of combining complex and heterogeneous data structures and their associated errors [Openshaw, 1989]. These errors can be of various types and magnitudes. The problem is further complicated by the potential for these data being manipulated through adhoc and often unpredictable processes.

Due to their independent data structures and data creation processes, different error types (spatial vs business) are often separable [Goodchild, 1991]. Since these error types have discrete data properties it is difficult and possibly unnecessary to combine them in integrated error propagation algorithms. This suggests that multiple processing streams may be needed, dependant on the nature of a data manipulation and data relevance, that utilize manual intervention to manage the effect of error propagation.

2.3 Accuracy

By definition [Webster, 1984], in order for a data property to be considered accurate it must be free from error. While a truly accurate database would be the ideal within a GIS environment, it is not considered a feasible goal for the foreseeable future. Despite this, the term data accuracy is prevalent in information system literature. In this context, it is used as a measure of closeness of a given value to the true value. Data accuracy can, therefore, be considered a complement measure of data error. This more positive expression of data inconsistency is often useful when attempting to determine or quantify data quality.

2.4 Uncertainty

Uncertainty and error are similar in that they both have a negative connotation [Chrisman 1991] and both propagate as a result of data manipulations. Although they are often used interchangeably in GIS literature, within this thesis they are distinguished, as error is associated with data and uncertainty is considered a perception held by data users. As an example of this distinction, an assessor, interested in area calculations, may be uncertain about the suitability of a property data source that has an approximated indication of error associated with it.

Uncertainty within spatial information systems stems from the question of whether or not available data is suitable for a specific application. Two main types of uncertainty are considered. The first type is error-related uncertainty, which can result from the lack of a reliable indication of data accuracy. This type of uncertainty also occurs when the only source of data available is not considered suitable for an application. A second type of uncertainty includes issues that are not related to error, such as data currency, which affect a potential user's confidence in a specific data type. This type of uncertainty is based on a potential user's requirements and perception.

2.4.1 Sources of Uncertainty

Bedard [1987] identified two sources of error-related uncertainty; limitations of the modelling process, and limitations of the model makers. The models he refers to are used in the process needed to communicate real world properties to a GIS user. This process involves several stages of cognitive, communication and physical models. Each of these models has the potential to introduce uncertainty in the 'real world - data - information - knowledge' chain.

The limitations in the modelling process relate to both the identification and definition of entities as well as measurement of properties associated with the entities. These limitations are due to the approximations used to model real world properties. The approximation processes result in the types of errors discussed in Section 2.2.2.

The limitations related to the model makers are a result of the subjectivity introduced by humans in determining the properties associated with real world entities. Subjectivity is a result of both the observer's background and their perspective. Subjectivity can be introduced at any stage of the GIS data life cycle including data collection, processing or end usage. While subjectivity is difficult to control and measure, it is another source of error-related uncertainty.

An additional source of uncertainty is introduced when data is considered in context with a user's specific application. This type of uncertainty is not error related and can be introduced at various stages of data manipulation. An example of a source of this type of uncertainty is the information used to describe a data supplier. This information may, depending on previous experiences, make a user less comfortable with the information environment. Similarly, information on data currency may determine data to be suitable or unsuitable for an application.

2.4.2 Management of Uncertainty

While it is unrealistic to expect that uncertainty can be completely eliminated within spatial information systems, it is possible to manage the level of uncertainty. Management in this context is defined as establishing a level of uncertainty that users are comfortable with in assessing data suitability. Management of uncertainty is seen as a requirement if GIS is to develop into a production decision making environment.

A primarily technical approach to managing uncertainty is through uncertainty reduction [Bedard, 1987]. This is an error reduction based process which utilizes technical and procedural means to reduce error-related aspects of uncertainty. It utilizes self-checking measurement processes to reduce error, and statistical reporting to provide more rigorous indications of data accuracy. Since it is difficult to eliminate all sources of error, this approach cannot provide a complete solution to uncertainty management.

Uncertainty absorbtion is an alternative approach to managing uncertainty [Bedard, 1987]. This requires either the data supplier or data user to 'absorb' the uncertainty by guaranteeing data (data supplier) or accepting the consequences (data user). Absorbtion can also be shared as proposed by the NCDCDS [1988] *truth in labelling* and *fitness for use* concepts, where a data supplier is expected to provide meaningful indications of data quality so that potential users can determine whether the data is appropriate for their needs. Uncertainty absorbtion approaches that utilize Data Quality Reporting concepts are able to address both error and non error-related considerations. Uncertainty absorbtion has a subjective component and cannot eliminate uncertainty.

Uncertainty cannot be completely managed by either of the main uncertainty management approaches (uncertainty reduction or shared uncertainty absorbtion) on its own. The two approaches are, however, complementary and should be pursued concurrently.

2.5 Quality

The existence of uncertainty with respect to data makes it necessary to determine data quality. If the quality (relative degree of excellence) of available data is known, a data user is able to establish an appropriate level of uncertainty with respect to a given application. In order to facilitate this, a data quality indication should provide information on non error-related uncertainty as well as data accuracy. Since user expectations are considered key to the determination of data uncertainty [Ryan and Masters, 1991], it is not feasible to explicitly assign an all purpose definitive quality measure to data. Instead, data quality reports should be provided that contain the information required by individual users to establish the quality of data relative to their specific applications. Data Quality Reporting considerations are discussed in detail in Chapter 3.

2.6 Definitions

The following definitions are based on the preceding discussion and will be used throughout the remainder of this thesis.

2.6.1 Data Error

Data error is defined as *the difference between an observed or approximately determined value and the true value of a quantity* (after [Webster, 1984]).

2.6.2 Data Accuracy

Data accuracy is defined as *the level of conformity of an observed or approximately determined value to the true value of a quantity* (after [Webster, 1984]).

2.6.3 Uncertainty

Uncertainty is defined as *the lack of confidence in a stated data property due to the knowledge that there is some level of data error or other source of ambiguity associated with that property.*

2.6.4 Data Quality

Data quality is defined as *the level of excellence of an observed or approximately determined value with respect to the true value of a quantity* (after [Webster, 1984])

2.7 Relationships

As a result of the discussion in the previous sections, the relationships shown in Figure 2.7 have been derived for use within this thesis. Brief textual descriptions of these relationships are provided in the remainder of this section.

Data Error is described by a statement of Data Accuracy, and Data Error contributes to overall Uncertainty.

Data Accuracy is used to describe or measure the degree of Data Error, and Data Accuracy contributes to the overall measure of data quality.

Uncertainty is influenced by Data Error. Although not shown in the diagram, there are other influences on Uncertainty which are not error related. Uncertainty can be managed by knowing the level of Data Quality.

Data Quality is influenced by Data Accuracy. Although not shown in the diagram, there are other influences on Data Quality which are not accuracy related (these are discussed in Chapter 3). An indication of Data Quality can be used to manage Uncertainty.

The direct relationships not shown in Figure 2.7 (Data Error to Data Quality and Uncertainty to Data Accuracy) are not considered relevant to the DQR approach developed in this thesis. The DQR concepts address them indirectly through the other relationships.

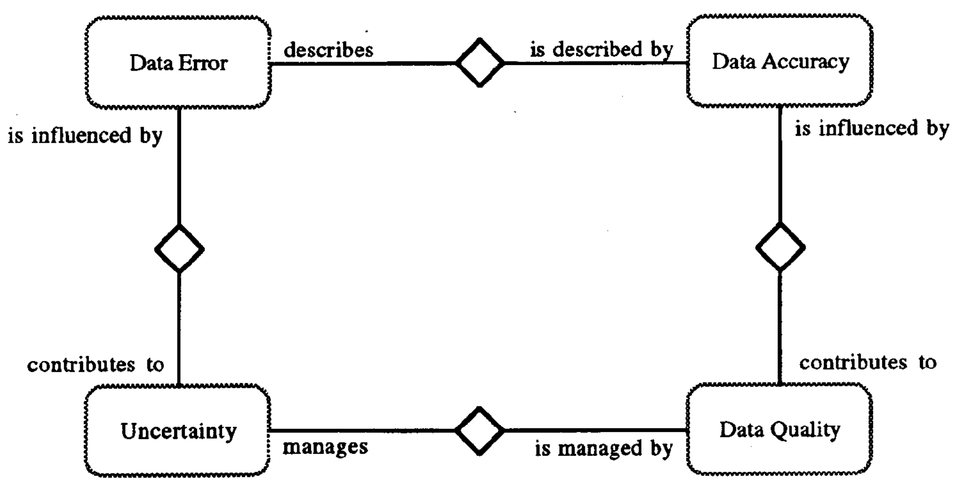

Figure 2.7 Error-Accuracy-Uncertainty-Quality Relationships

2.8 Summary

The concepts of error, accuracy, uncertainty and quality have been reviewed with respect to the spatial information system environment. Definitions for these terms and the relationships between them have been developed for use within this thesis.

Error and accuracy are considered data specific properties. Uncertainty and quality, however, are influenced by both data and application (user) dependant considerations.

The knowledge of the existence of errors in data, combined with other non error-related considerations, result in a potential user being uncertain as to the suitability of data with respect to a specific application. By providing an indication of the quality of data through DQR, this uncertainty can be better managed.

3 DATA QUALITY REPORTING FRAMEWORK

3.1 Introduction

In order to provide the meaningful indication of data quality discussed in Section 2.5, there is a need for a Data Quality Reporting function. This chapter reviews data quality report classification and reporting level considerations and proposes a structure that would support an 'ideal' Data Quality Reporting environment.

3.2 Data Quality Reporting Classifications

The groupings of information for which quality indicators are required are referred to as Data Quality Reporting classifications. By utilizing standard classifications to report on data, reports that will be of use to all potential applications of spatial data will result. Without standard classification schemes, automated error propagation algorithms are complicated and end user evaluations are difficult.

There have been several different classification approaches developed as part of major data modelling and data transfer initiatives. A cursory review of some of these approaches follows as background information. Other projects [SAIF, 1991; SDTS, 1990] employ similar classification schemes.

3.2.1 National (USA) Committee for Digital Cartographic Data Standards (NCDCDS)

The NCDCDS is an American committee formed to develop digital cartographic data standards [NCDCDS, 1988]. A major part of the standard is the Digital Cartographic Data Quality module. In this module, it is recommended that a quality report be comprised of five reporting sections (specific reference must be made to data currency for each section), including:

Lineage

The lineage section includes the description of the source of all data, including methods of derivation, information on datum (control and projection base) and any coordinate transformations utilized.

Positional Accuracy

Reports on positional accuracy are required to include consideration of any transformations utilized in the data derivation.

Attribute Accuracy

An attribute accuracy report is intended to provide an indication of the correctness of specific attributes.

Logical Consistency

The logical consistency report indicates how well a data structure models the real world entity.

Completeness

The completeness report primarily describes the exhaustiveness of a set of features at both the entity and attribute levels.

3.2.2 Alberta Land-Related Information Systems (LRIS)

In order to be able to provide the integration required for Alberta's land-related information, LRIS is developing a Spatial Database (SDB) System. In planning for the SDB, the Province of Alberta initiated a business modelling project to determine the requirements of the SDB. Within the LRIS Spatial Database System Model [LRIS, 1991] data quality classifications are addressed as follows:

Accuracy

An accuracy measure is an indication of the closeness of position values to the true position for a feature. Positional accuracy can be represented as either absolute or relative accuracy.

Coverage/Completeness

A coverage/completeness report provides a measure of how complete a data coverage for a specific entity is.

Age/Currency

This report indicates when data was originally captured and when any subsequent verification or updating took place.

Native Form of Data

The native form of data specifies the datum, coordinate system, projection and storage format of all data available.

Structuring/Integrity

This classification reports on the topological structure and integrity constraints imposed during data creation.

Significance and Precision

Significance and precision reports document the number of digits used to store spatial coordinates and relevance of these values.

Supplier Specific Measures

The supplier specific measure report allows suppliers to include any other supplier-based quality indicators considered relevant.

3.2.3 Proposed Data Quality Classifications

Although there are differences in the approaches to the data quality classifications reviewed above, there is a strong correlation between the types of information considered important to quality reporting. The following Data Quality Reporting classification structure attempts to incorporate all aspects identified above and to group them into the three main classifications: accuracy, source, and currency. The accuracy classification is introduced to deal with the error-related aspects of uncertainty, while source and currency classifications provide information on the aspects of uncertainty that are not error driven. Data Quality Reporting classifications are summarized in Figure 3.2.3 with examples of the types of data quality classification information that would be found in the main classifications.

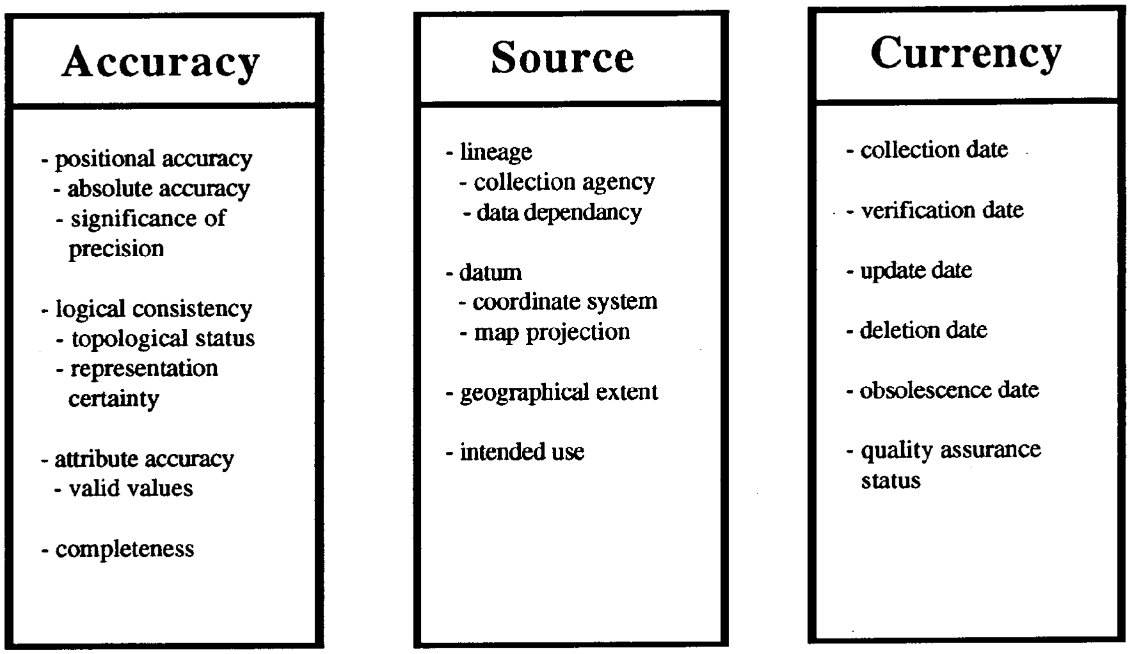

Figure 3.2.3 Data Quality Reporting Classifications

Accuracy

All aspects of accuracy reporting should include a description of the procedures used in the determination of the accuracy statement. This could take the form of a detailed explanation or a reference to a standard procedure documented elsewhere.

Although relative accuracy indictors are useful, the positional accuracy component in a GIS is best expressed as an absolute accuracy indicator. This allows for direct comparisons between non related data sources. In making positional accuracy statements attention should be given to the significance of precision.

The logical consistency component proposed by NCDCDS can also be considered as part of an accuracy report. This should include a statement of the entity's topological status and indicate any uncertainty associated with non rigorous approaches to feature representation.

Attribute accuracy reporting should reflect the quality of any data interpretation or classification procedures utilized. Valid value testing results should also be included.

Completeness statements are applicable to all aspects of an accuracy report. This includes statements on the extent and success of collection campaigns as well as the sample size of any quality assurance testing.

Source

A source report should provide a lineage statement. In a generalized form it would contain information on the agency that collected the data and a description of the technology used. A detailed lineage statement would include explicit data dependency descriptions.

A datum definition statement should also be included. This must clearly define the reference ellipsoid and map projection utilized in establishing the spatial component. Any non-standard parameters or transformations must be identified.

An explanation of the geographical extent of the data collection effort is sometimes necessary. This could be in the form of a polygon definition referenced to the standard map projection or the legal survey fabric.

Statements of intended use should be added to reduce the potential for data misuse-related errors (see Section 2.2.1).

Currency

Currency reporting should clearly identify the date and time of collection, verification, update, deletion and obsolescence. In addition, it should provide a status indication of on-going quality assurance testing.

3.3 Levels of Data Quality Reporting

While it would be ideal to have all aspects of data quality reported on at the lowest level of data aggregation (entity/attribute), this is not considered a practical approach. The heavy storage and computation overhead associated with such a strategy would significantly constrain processing efficiency. In addition, it is not feasible to collect data quality information at that level of detail. If instead, a multi-levelled reporting structure was implemented, it is likely that a compromise between rigour and efficiency could be reached.

The Data Quality Reporting structure identified in the LRIS Spatial Database System [LRIS, 1991] includes reporting at the supplier, entity type, entity occurrence and attribute levels. While this addresses the majority of entity based reporting requirements, it does not make provision for quality reporting of spatial groupings of entities. This type of reporting is considered necessary in order to indicate data quality parameters that are common to all entities within a specific geographical extent. The Geographic Data Files standard [GDF, 1992], which addresses this requirement at the data set level (all entity types combined), can be extended to include a spatially partitioned entity type report.

Based on the discussion above, there have been five quality reporting levels identified as necessary for effective data quality reporting. These methods are described below and summarized in Figure 3.3.

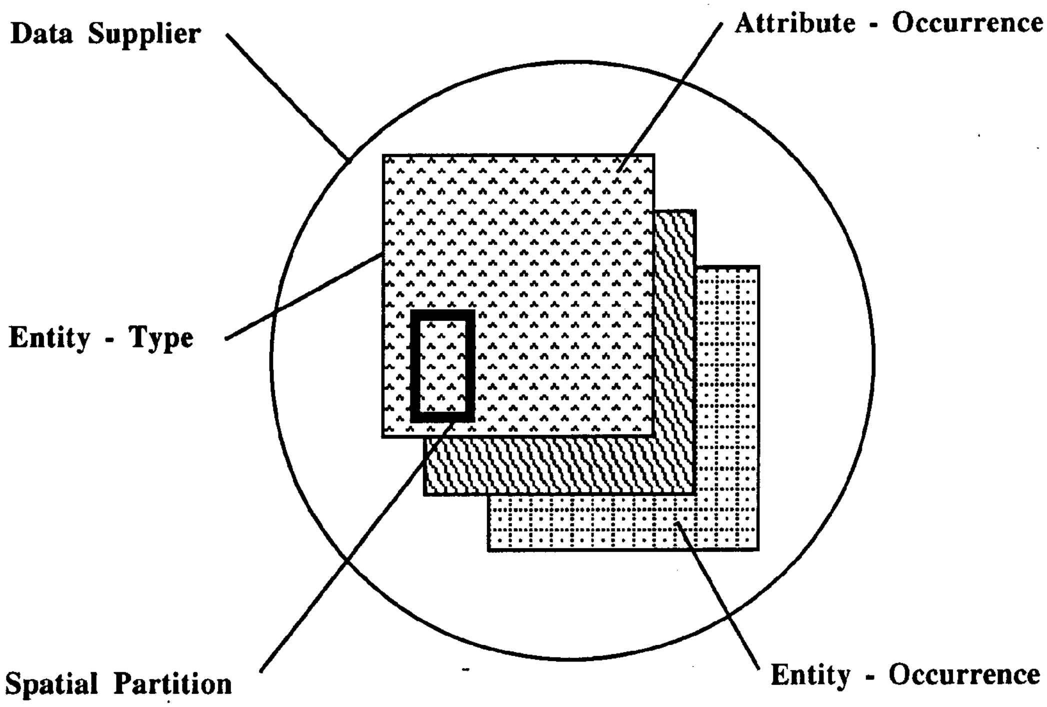

Figure 3.3 Data Quality Reporting Levels

Data Supplier Reporting

Data Supplier Reporting should be in the form of a report provided by a data supplier (or data convertor) indicating the quality of data provided. As a minimum, it should be a generic statement outlining the procedures used in creating the data set and a detailed explanation of storage structures and geographical datum used. This level of reporting provides a means to report on multiple entity types.

Entity - Type Reporting

Entity Type Reporting provides information on the quality of a data type in the form of a generic data quality statement for all features of that type. This form of reporting can be used to assign data quality specification values or 'default' quality indicators.

Spatial Partition Reporting

Spatial Partition Reporting provides information on the quality of data in the form of a generic data quality statement for all elements within a specified geographical area. This is similar to Data Supplier Reporting; the difference being that the members of the Spatial Partition are defined as all instances contained within the extent of a user defined geometric shape. Spatial Partition Reporting can be restricted to a single entity type or include all entity types within the partition. This approach supports the statement made by NCDCDS [NCDCDS, 1988] that where spatial variation in quality is known, it must be recorded.

Entity - Occurrence Reporting

Entity Occurrence Reporting provides information on the quality of data on an individual feature basis. A quality statement at this level refers to the entity as a whole.

Attribute - Occurrence Reporting

Attribute reporting provides information on the quality of data on an individual attribute basis. This represents the most detailed level of Data Quality Reporting. Due to the high costs associated with collecting information at this level of detail, and the resulting storage and computational overhead, it is expected that attribute reporting will only be implemented in special circumstances.

3.4 Combined DQR Framework

In the preceding parts of this chapter, generic approaches to Data Quality Reporting classifications (Section 3.2) and levels of Data Quality Reporting (Section 3.3) have been developed independently. By combining these two DQR components, a Data Quality Reporting framework that would support the 'ideal' data quality environment described by Goodchild [1991] is achievable.

The proposed framework allows the reporting of any data quality classification at any reporting level. In addition, all classifications and levels are optional. For these reasons the framework possesses the characteristics (comprehensive, flexible and meaningful) associated with practical DQR (discussed in Section 1.2.2). For example, an organization could make a generic statement of Positional Accuracy at the Spatial Partition level and supplement (override) it with Positional Accuracy statements for some of the entities at the Entity Occurrence level, as this information became available.

Figure 3.4 illustrates the combined classification and level reporting framework. The types of information that might be reported on at the various levels are included in this figure. This is provided as an example for illustration purposes and not intended to be an exhaustive list of quality reporting classification-level combinations.

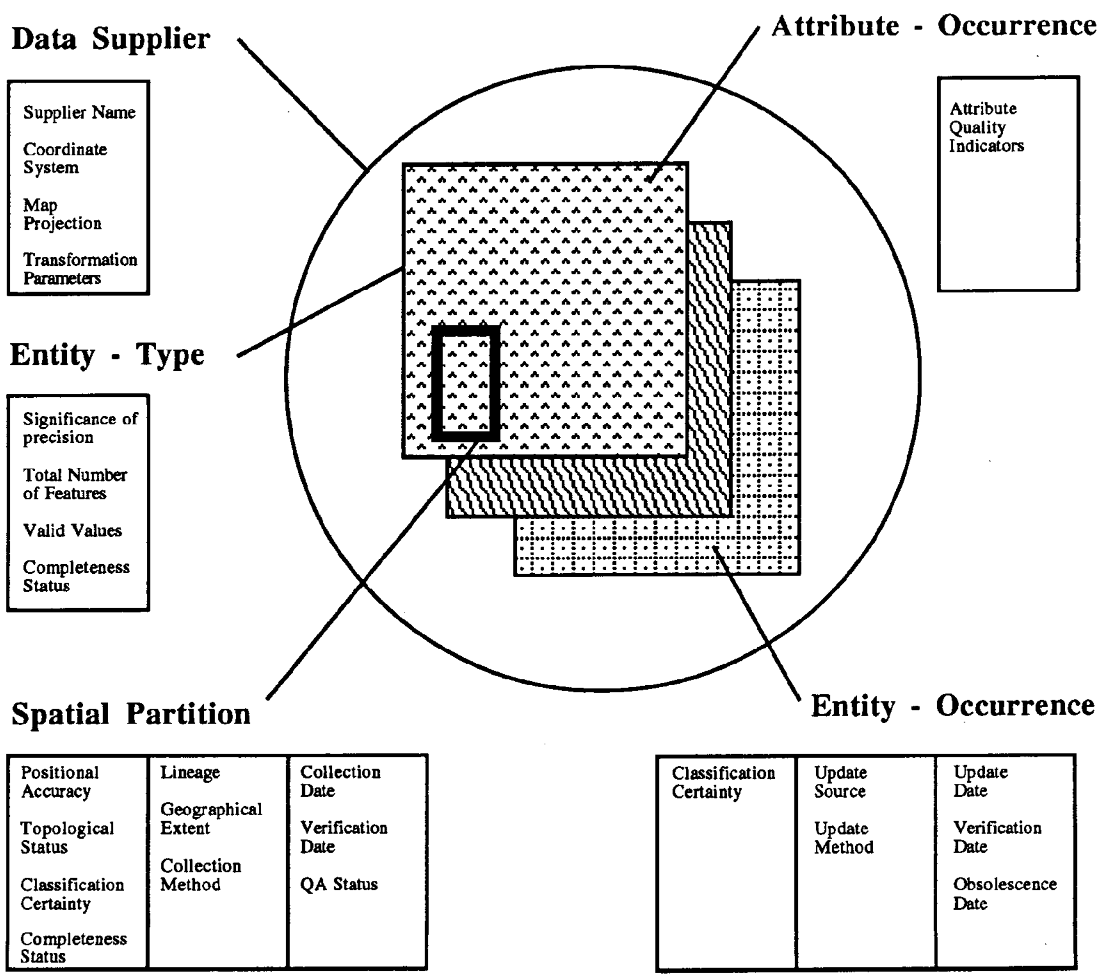

Figure 3.4 Data Quality Reporting Framework

3.5 Summary

Generic Data Quality Reporting classifications have been developed in this chapter. These are based on emerging industry standards and the uncertainty considerations discussed in Chapter 2. Various levels of Data Quality Reporting have also been reviewed.

In combining the classification and level considerations, a general DQR framework has been proposed. This technology independent model provides a reference for the development of practical Data Quality Reporting structures that are capable of addressing future DQR requirements.

4 DATA QUALITY REPORTING ANALYSIS

4.1 Introduction

In order to investigate some of the concepts introduced in the preceding chapters, a prototype system has been developed in the remainder of this thesis. This system is not intended to be a prototype of a production system. It has been created to demonstrate the potential of Data Quality Reporting within GISs. This type of approach could be adopted in future production GIS systems and implementations. While the prototype system utilizes a real data set, the data is simulated and limited in scope. In addition, the GIS functionality is restricted to a single hypothetical application.

In this chapter, an industry standard data model and two system development tools are introduced. These have been utilized to provide a sound starting point and to improve the quality of the resulting prototype system. The SAIF data model has been used as a reference in the spatial data model development portion of the analysis. The Coad/Yourdon methodology, which SAIF has incorporated, provides a structured set of techniques and conventions that support the entire system development life cycle. MacA&D is a CASE tool that provides an automated environment to carry out the analysis documentation and supports the Coad/Yourdon methodology [MacAnalyst, 1993]. The remainder of this section explains the prototype approach, provides background information on the SAIF model and explains the documentation techniques and conventions used in the data model development.

4.1.1 Prototype Approach

The purpose of the DQR demonstration application is to test the feasibility of the DQR concepts and framework presented in this thesis. Since this approach to DQR utilizes new and unproven concepts, there are advantages to evaluating it within a prototype development environment. A prototype can be used to test feasibility while limiting the time and cost (and therefore commitment) associated with production-oriented application development in commercial GIS development environments.

A prototype is often positioned as a 'throw-away' in that a decision is made prior to starting development that any subsequent production-oriented versions of it will constitute a complete redesign in another development environment [Martin, 1990]. In doing this, the development effort can be more focussed on testing concepts than on providing rigorous and end user-oriented solutions. This is the approach to the DQR prototype development utilized in this thesis.

4.1.2 SAIF

On June 26, 1991, the original SAIF (Spatial Archive and Interchange Format) standard [SAIF, 1991] was accepted, by the Canadian General Standards Board's Committee on Geomatics, as a Canadian standard for the exchange of geomatics data. Subsequent work on this standard has resulted in the release of a draft version of the future Canadian Geomatics Interchange Standard [CGIS, 1992]. Although it will eventually be renamed CGIS, this standard will be referred to as SAIF for the remainder of this thesis.

The SAIF data model (and SAIF schema) provide a structure to support the sharing of any information which can be referenced to the earth [CGIS, 1992]. Although SAIF has been designed to facilitate data exchange, its extensive data structure definition provides a standardized framework for spatial data modelling and GIS systems development [Strand, 1994].

SAIF is based on the Object-Oriented paradigm and supports both Whole-Part and Generalization-Specialization data structures. In order to simplify the model, a single inheritance approach has been adopted, where each class can have only one parent class. SAIF does not include methods to access objects. The methods are considered implementation specific and, therefore, beyond the scope of a data exchange standard.

4.1.3 Analysis Techniques and Conventions

The use of a structured systems development methodology, supported by a Computer Aided Systems Engineering (CASE) tool, is seen as a significant advantage in the development of quality computer systems [Yourdon, 1989]. The Coad/Yourdon methodology has been used in the development of the prototype system. The MacA&D CASE tool has been used to capture the analysis documentation and to verify its contents.

The Coad/Yourdon approach to Object-Oriented analysis employs a methodology which is based on four main identification and definition tasks. These tasks include identifying classes (or objects), identifying their structures, defining the object attributes, and defining the methods (or services) that access the objects. The procedures utilized in carrying out these tasks are described in detail by Coad and Yourdon [1991a].

Within the text component of this thesis, class names begin with capital letters and are italicized (e.g., *City*). Attribute and method names start with capital letters but are not italicized (e.g., Population). The class names, which are normally singular nouns, have been pluralized within the text on several occasions to produce more readable text. Class instance (object) names are contained in quotation marks (e.g., 'Calgary').

The Coad/Yourdon Object-Oriented modelling components and their corresponding diagramming conventions are briefly described in the remainder of this section.

Classes are groups of objects that share common data and processing characteristics. Objects represent instances of classes. An example of a class is *City*, and an example of a *City* object is 'Calgary'. Classes are represented by boxes with rounded corners. Class names are indicted in the upper section of the class boxes. In Figure 4.1.3 *School* and *Student* are examples of classes.

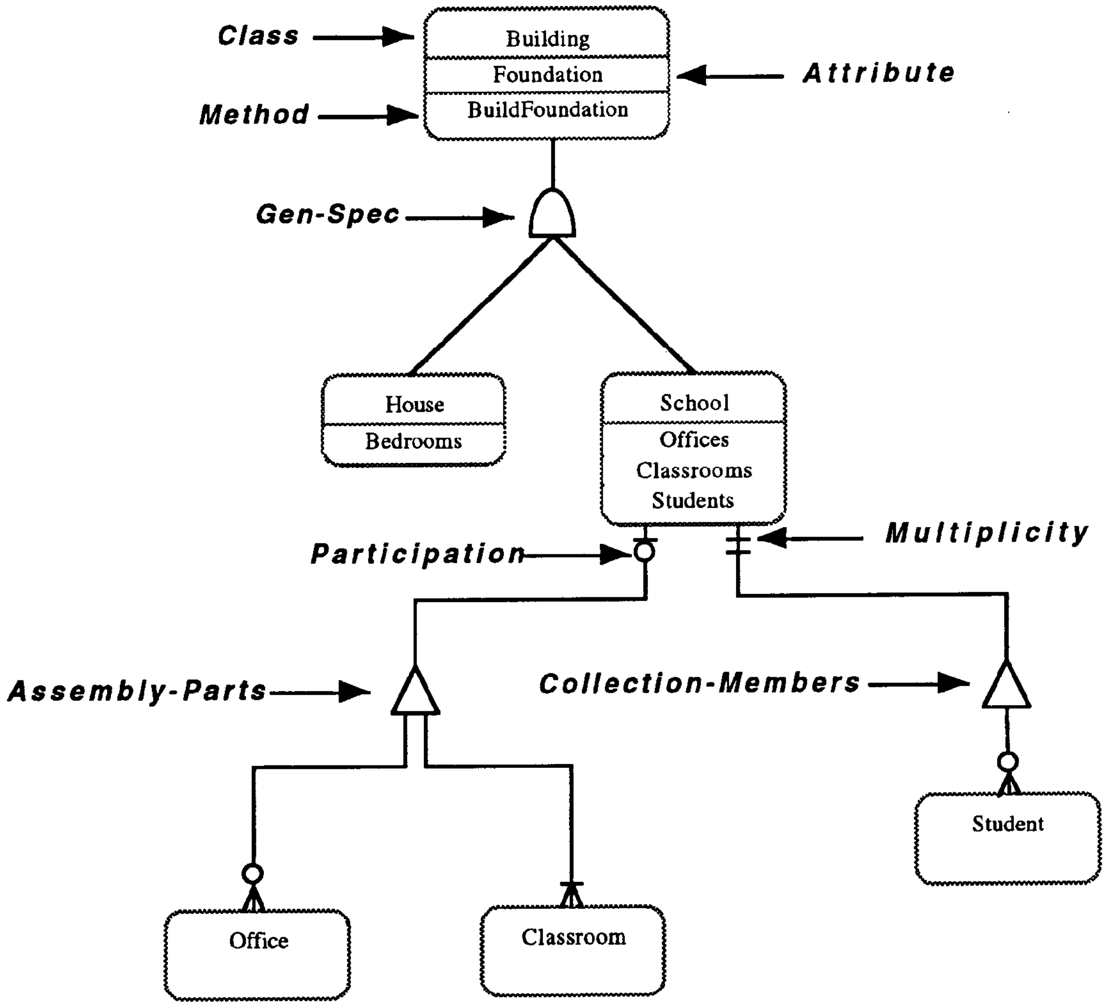

Figure 4.1.3 Coad/Yourdon Modelling Conventions

In order to organize the complexity existing in problem space, two types of class structure are used; Generalization-Specialization structure and Whole-Part structure. The Generalization-Specialization (Gen-Spec) structure models the subclass relationship between classes. This structure enables the inheritance capabilities associated with Object-Oriented development. Inheritance is accomplished through parent-child associations between classes, where a child class (subclass) inherits characteristics from its parent class. An example of a Gen-Spec structure is a generalized *Building* class being specialized into two child classes: *House* and *School*. Both *House* and *School* inherit the Foundation attribute from Building. In addition to this generalized attribute, each subclass has its own attributes; a *House* has Bedrooms and a *School* has Classrooms. Neither *House* or *School* requires the other's attributes. The Gen-Spec structure relationships are drawn as half-circles with connections between related classes shown as class connection lines. Gen-Spec structure is illustrated in Figure 4.1.3, where *House* and *School* are specializations of *Building*.

Whole-Part structure can either represent Assembly-Parts or Collection-Members relationships. An example of an Assembly-Parts structure is a *School* class aggregating *Classroom* and *Office* classes, as shown in Figure 4.1.3. A Collection-Member structure would be used to model the relationship between *School* and *Student*. Often, 'Part' classes are associated to their 'Whole' classes as attributes of the 'Whole' class. The *School* - *Student* Collection-Members structure shown in Figure 4.1.3 illustrates this. Whole-Part structure relationships are depicted as triangles with connections between related classes shown as class connection lines.

Within Whole-Part structures class connection lines symbolize the existence of a relationship and the level of multiplicity and participation. For each direction between related classes, the class connection lines are marked with a single bar (one to one) or a "crow's foot" (one to many) to indicate multiplicity relationships. These symbols are placed close to the class symbols (box with rounded corners). Similarly, participation is shown on class connection lines. A single bar indicates that participation in the relationship is mandatory, and an 'o' represents an optional connection. Participation symbols are placed furthest away from the class symbols (i.e., multiplicity symbols are placed between class symbols and participation symbols). The Whole-Part structure in Figure 4.1.3 indicates that a *Student* instance must be associated (participation) with only one (multiplicity) *School* , and that *School* instances may be associated with (participation) at least one (multiplicity) *Student* object. *Office* and *Classroom* instances may be associated with only one *School*. *Schools* may be associated with at least one *Office*, and must have at least one *Classroom*. The relationships in this example are used to illustrate the Coad/Yourdon conventions, and do not necessarily reflect real world relationships.

Attributes are data used to describe individual objects. Attributes are inherited by subclasses within Gen-Spec structures and are listed in the centre section of class symbols. In Figure

4.1.3, Bedrooms is an attribute of *House*. *House* also inherits the Foundation attribute from *Building*.

Methods represent the processing that is carried out by an object when it receives a message. Although Coad/Yourdon refer to these as 'services' in the analysis and design stages, they are normally referred to as 'methods' during the implementation stages. To eliminate any confusion this may introduce, the term 'method' will be used in this thesis through all stages of development. Methods are also inherited by subclasses within Gen-Spec structures and are shown in the lower section of class symbols. In Figure 4.1.3, BuildFoundation is a *Building* method. The BuildFoundation method is inherited by *House* and *School*.

4.2 The Analysis

The analysis carried out in this chapter provides a high level definition of the DQR prototype system developed in this thesis. This conceptual view of the system requirements is normally the communication link between the system analyst and the client. In a top-down approach to systems development, only issues that are problem space specific are examined at the conceptual level, while the technical implementation concerns are addressed in subsequent stages of the development.

Although a top-down approach has been utilized in this thesis, the conceptual model presented in this chapter has been modified to reflect some of the design decisions made in the next chapter. This has been done for three main reasons. First, it provides a better quality model since the problem is better understood as a result of the work carried out in the more detailed design stage. Secondly, it ensures that the model is consistent throughout the system development stages. Lastly, by eliminating classes that were found to be unnecessary in the design stage, the conceptual model is simplified and, therefore, easier to understand.

The model development presented in this chapter is primarily focused on the spatial component of the problem space. This is consistent with developing a generic GIS Data Quality Reporting function. There is, however, a need to model a typical business application to facilitate meaningful demonstrations within the DQR prototype. The building permit application review problem space has been modelled in this analysis to satisfy this requirement. This example is based on a real world municipal government business function, but is not intended to represent production requirements. The resulting prototype Building Permit Application Review Support System (BPARSS) developed in this thesis, has introduced many assumptions and simplifications in order to manage the scope of the test case. Despite this, the example provides sufficient content to illustrate the DQR concepts discussed in this thesis. It also portrays a situation where an erroneous decision, due to a lack of understanding of data quality, could result in significant legal and financial ramifications.

The analysis includes a Statement of Purpose and Functional Requirements for both the generic DQR function and the example BPARSS. The remainder of the analysis is documented through eight views of the model, presented as diagrams with detailed explanations. The details provided in this chapter are only the highlights of the analysis process. Further model documentation is available in MacA&D format.

4.2.1 Statement of Purpose

The purpose of a Data Quality Reporting function is to communicate the information (data and metadata) required to support data suitability review and decision making within a GIS environment.

The purpose of a Building Permit Application Review Support System is to assist in the determination of whether a building permit should be granted for a specific building permit application.

4.2.2 Functional Requirements

In order to support the Statement of Purpose, the functional requirements of a DQR function are as follows:

- To determine data quality.
- To document data quality.
- To determine the effect of data manipulations on data quality.
- To present a meaningful indication of data quality to end users.

In order to support the Statement of Purpose, the functional requirements of a BPARSS are as follows:

- To review a building permit application.
- To determine the relationship between relevant business entities.
- To establish compliance with building regulations.
- To recommend a decision on whether or not to grant a building permit.

4.2.3 Information

When reviewing Data Quality Reporting it is useful to examine the relationship between data and information. Information has been defined [Boddie, 1993; Walsh, 1993] as being data put into context. Data context is normally documented as metadata (data about data). The data definitions and relationships within a data model are one example of providing data context. Within a production GIS, an understanding of the quality of data can be considered the context required to transform data to information. This distinction between data and information is used in this thesis and serves as a starting point in the analysis.

This relationship between *Information*, *Data* and *Context* is illustrated in the upper part of Figure 4.2.3. This mandatory Assembly-Parts structure illustrates that *Information* can only exist if *Data* is put into the appropriate *Context*. *Information*, *Data* and *Context* are all considered high level abstract classes and are discussed here only to position the DQR function within Information System Theory.

An abstract class is a class that is not intended to have instances. It defines functionality that is common to all its subclasses, which do have instances. By using abstract superclasses, common functionality is developed and maintained in one place [Smalltalk, 1993].

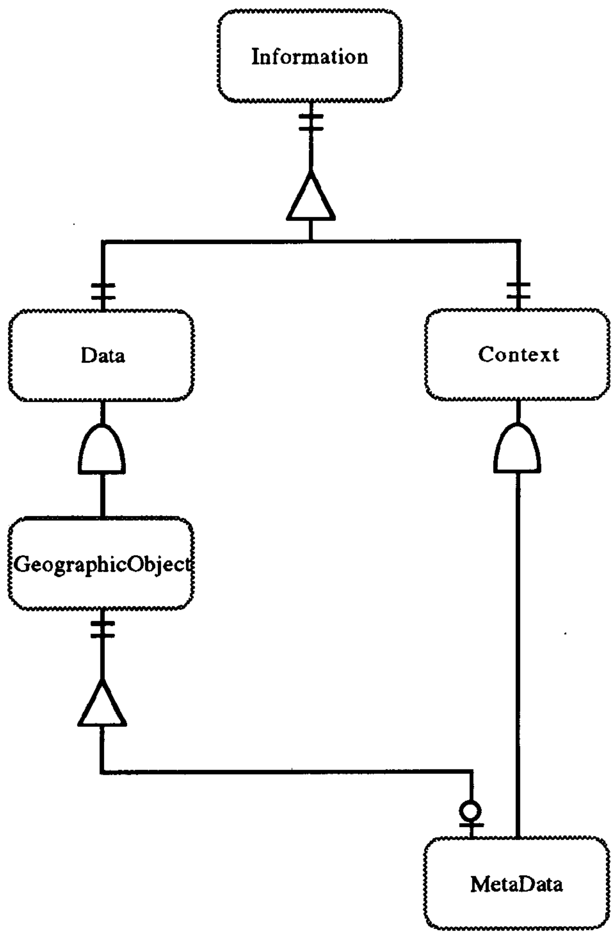

Figure 4.2.3 Information Model

Two other abstract classes, considered key components of the DQR function, are also introduced in this figure. These are *GeographicObject* and *MetaData*. *GeographicObject* is a specialization of *Data* that can be used to describe all spatial data. These objects characterize any real or artificially defined entity above, on or within the earth [CGIS, 1992], and represent both the spatial and business components of spatial data. Other business related subclasses of *Data*, such as *BuildingRegulations*, are also needed to support the DQR function within GISs.

The *MetaData* class, which has an optional Assembly-Parts structure relationship with *GeographicObject*, provides the context required to establish information status. *MetaData* records the quality parameters of *GeographicObjects* and is considered a specialization of the *Context* class.

4.2.4 Geographic Object

The structure shown in Figure 4.2.4 represents the foundation of the SAIF model. The *GeographicObject* class is an abstract class that is the aggregation of *GeometricObject*, *Relationship* and *MetaData*. Each of these subclasses are optional. While a *GeographicObject* can have multiple *Relationships*, it can have only one *GeometricObject* and *MetaData* instance associated with it. Each *Relationship* must have two or more *GeographicObjects* associated with it. Each *GeometricObject* must depict at least one *GeographicObject*.

Although the SAIF model allows a *MetaData* instance to be associated with multiple *GeographicObjects*, this multiplicity rule has been modified within this thesis to restrict each *MetaData* object to representing only one *GeographicObject*. Assignments of multiple *GeographicObjects* to individual *MetaData* instances are addressed by defining *GeographicComposites* (see Section 4.2.5).

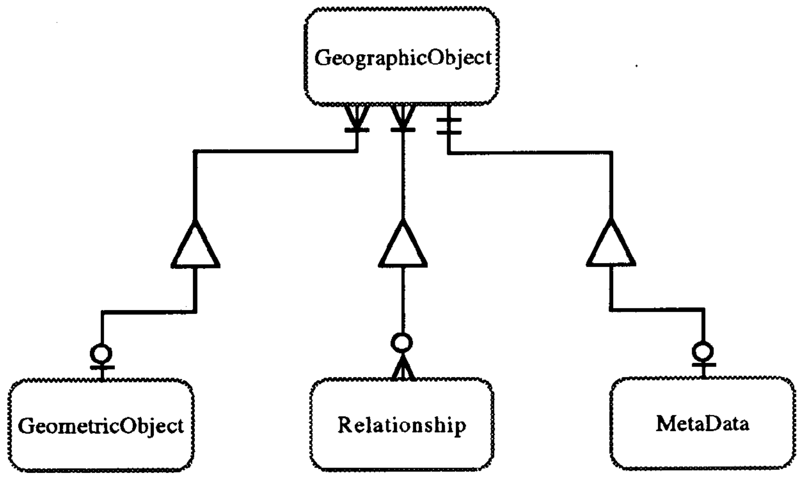

Figure 4.2.4 Geographic Object

4.2.5 Geographic Composite

A specialization of *GeographicObject* that is essential to effective DQR is the *GeographicComposite* class. This class provides an aggregation capability that allows groups of *GeographicObjects* to be consolidated regardless of their types. The ability to assign *MetaData* to *GeographicComposites* supports the requirement, discussed in Section 3.3, to report on data quality at multiple levels. This is the primary use of *GeographicComposite* within this thesis. Figure 4.2.5 shows that a *GeographicComposite* is a specialization of *GeographicObject* and that one to many *GeographicObjects* may be assembled to make a *GeographicComposite.* The *GeographicObject* class is shown twice in this diagram to clarify the Whole-Parts and Gen-Spec relationships between the two classes.

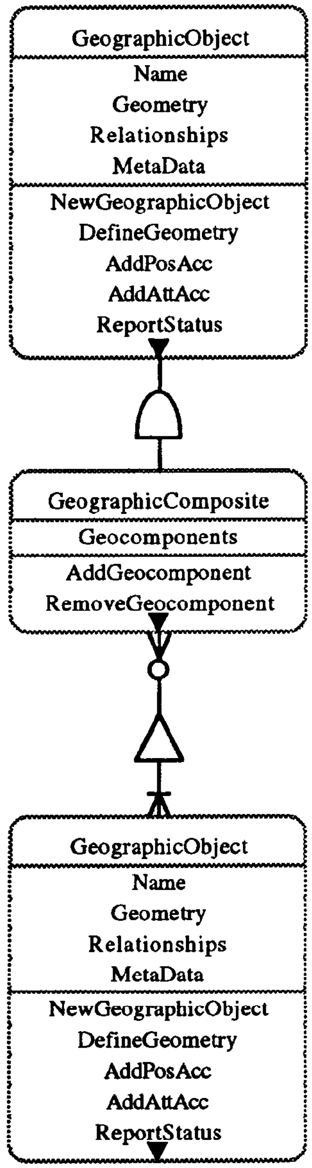

Figure 4.2.5 Geographic Composite

This diagram is the first to show the attributes and methods associated with objects. The *GeographicObject* class has the optional attributes of Geometry, Relationships and MetaData, described in Section 4.2.4, as well as an attribute (Name) used to uniquely identify an instance. The first method (NewGeographicObject) is used to initiate the creation of a new *GeographicObject* instance. The DefineGeometry method defines the association between the *GeographicObject* and a *GeometricObject*. The AddPosAcc and AddAttAcc methods provide the means to add data quality information. The last method (ReportStatus) is used to provide a status report on all known information associated with the *GeographicObject*. The triangle at the bottom of the method list indicates that there are other methods documented in the CASE tool.

Only one high level attribute (Geocomponents) is provided for the *GeographicComposite* class. Geocomponents is a collection of *GeographicObjects*. Recursion can occur within this class if any of the Geocomponents is a *GeographicComposite*. This is the only class within the SAIF model for which this is true [CGIS, 1992]. The two methods (AddGeographicComponent and RemoveGeographicComponent) are used to update the contents of a *GeographicComposite*.

4.2.6 Geometric Object

The *GeometricObject* is an abstract superclass which provides the basis for the spatial representation of all *GeographicObjects*. The *GeometricObject* subclassing definition is extensive within the SAIF model and includes both raster and vector primitives. The choice of classes used in the prototype development is a result of design decisions, as discussed in Section 4.2. Since Smalltalk (the implementation environment) defines polygons through a list of associated vertices, the normal point-line-polygon GIS topology structure has not been used. For this reason only the base vector primitives shown in Figure 4.2.6 have been considered in this thesis.

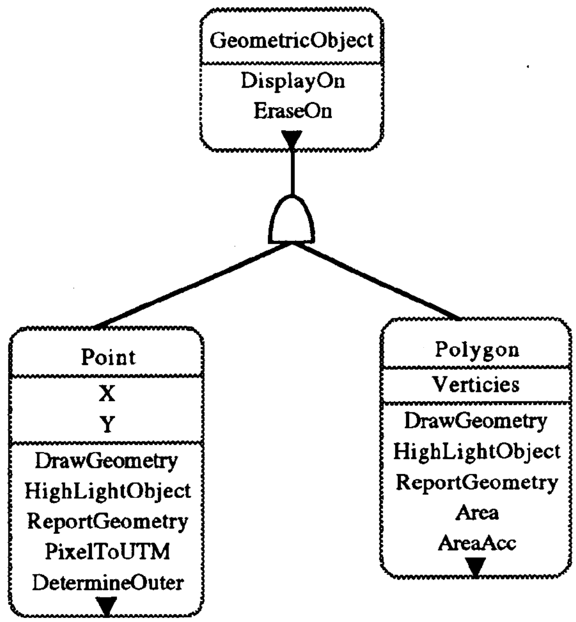

Figure 4.2.6 Geometric Object

GeometricObject is an abstract class that provides base display methods for all of its subclasses. The *Point* class has two attributes (X and Y). The X and Y attributes define a *Point*'s location in the form of a coordinate pair. The *Polygon* class associates a set of *Points* with its instances through its Vertices attribute. The *Point* and *Polygon* classes have their own specific methods (PixelToUTM, DetermineOuter, Area and AreaAcc) as well as methods that are named the same in both classes (DrawGeometry, HighLightObject and ReportGeometry).

The methods with common names provide the base needed to support the polymorphic properties associated with Object-Oriented applications. They employ different internal procedures to provide the same general results. For example, an object requesting that a *GeometricObject* be highlighted (HighLightObject) does not need to know whether it is addressing a *Point* or a *Polygon.* It is up to the *GeometricObject* subclasses to know how to respond to such a request. Polymorphism is discussed in more detail in Section 7.6.4.

4.2.7 Relationship

Similar to *GeometricObject*, the SAIF *Relationship* class is rich with subclasses. These include multiple spatial and temporal relationships between *GeographicObjects*, which can be used to support both geometric constructions and spatial analysis. In the prototype system development, the 'spatial building block' type of relationship between *Points* and *Polygons* is maintained by the *GeographicObject's* Geometry through a set of Vertices (see Section 4.2.6).

The only relationship between *GeographicObjects* that is considered in this thesis is represented by the *DerivedFrom* class, as shown in Figure 4.2.7. This relationship is used to accommodate relative positioning examples. Related *GeographicObjects* are identified by the RelatedObjects attribute (in the *Relationship* class), while the type of relationship (directly, indirectly, etc.) is recorded in the *DerivedFrom* Name attribute. The methods needed to access *DerivedFrom* objects are provided by the *DerivedFrom* class.

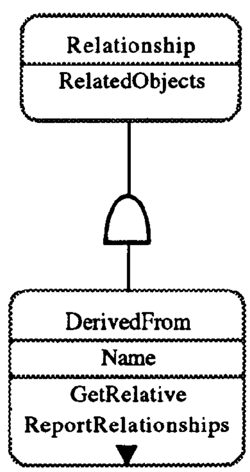

Figure 4.2.7 Relationship

4.2.8 Metadata

Three of the seven *MetaData* associations defined by SAIF are used in this thesis. They are shown in Figure 4.2.8 as an Assembly-Parts structure. These optional components are assembled into a *MetaData* instance. The *MetaData* class provides the methods needed to assemble the *MetaData* instances.

The SAIF usage of the term 'quality' is different from the 'quality' definition provided in this thesis. The SAIF *Quality* class is used to document 'accuracy' parameters (as defined in Section 3.2.3).

The SAIF *Quality* class utilizes another Assembly-Parts structure to record information on the 'accuracy' classifications. This association is made with the *PositionalAccuracy* and *AttributeAccuracy* classes in this model. The *PositionalAccuracy* class shown in Figure 4.2.8 is a modification of the class included in the pure SAIF model. This change has been made to simplify the prototype and is discussed in detail in Section 5.2.9. The *AttributeAccuracy* class is subclassed to address the specialized case of *RegionalizedAccuracy* (discussed in detail in Section 7.2.1). The *RegionalizedAccuracy* class represents an extension of the SAIF model based on an approach developed by Leung [1988], and provides the means to model the accuracy of attributes that vary over space in a continuous manner.

The *UpdateOperation* class provides the means to address the 'currency' considerations presented in Section 3.2.3. *History* is used to document the 'source' (as defined in Section 3.2.3) information associated with DQR.

The attributes for the *MetaData* components (assembly components), as shown in Figure 4.2.8, are described in detail within the SAIF documentation [CGIS, 1992] and in the MacA&D model. The methods provided by these classes support the creation of instances and the reporting of their status.

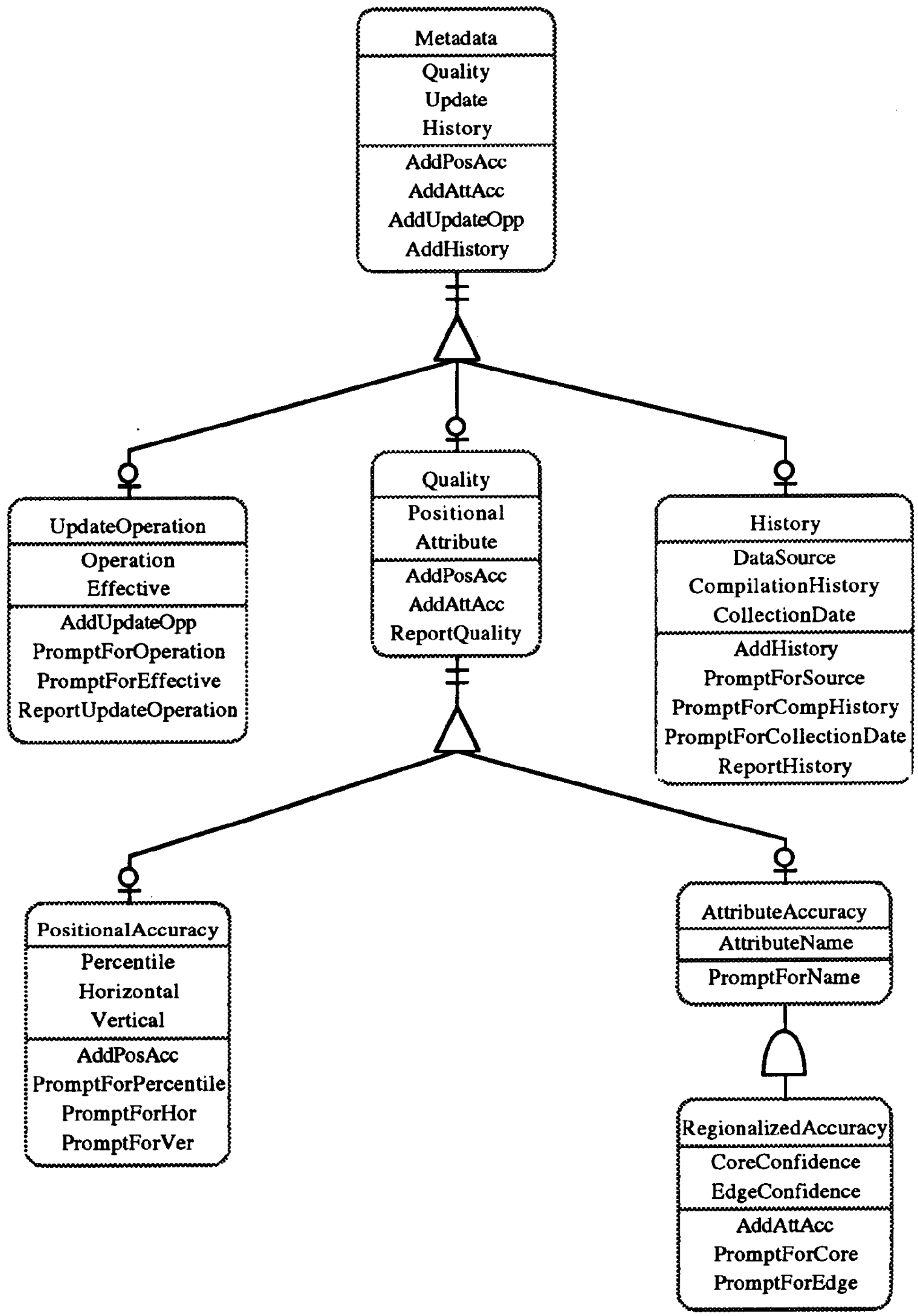

Figure 4.2.8 Metadata

4.2.9 Geographic Object Types

The view provided in Figure 4.2.9 is an extension of the SAIF model. It shows the subclasses of *GeographicObject* that are required by the example developed in this thesis. These are classes that may influence a decision in the building permit application review process, as a result of any development constraints that they may impose.

Although there would normally be several business-oriented attributes and methods associated with the classes shown in Figure 4.2.9, only those that arc significant to the demonstration of the DQR function within the BPARSS application are considered here. The subclasses also inherit the attributes and methods related to *GeographicObject*, which have been described in Section 4.2.5.

The *SpotHeight, ConstructionPoint* and *FireHydrant* classes' Geometry are represented by *Points*, while *Buildings*, *AirportVicinityAreas*, *FloodPlains* and *Properties* have *Polygon* class Geometry characteristics. The *GeographicComposite* class is a collection of *GeographicObjects*, used to address Data Supplier Reporting requirements (see Section 3.3), and does not have a Geometry component.

ConstructionPoints represent field survey points that are used to define the boundaries of polygonal features. Their AssociatedAggregate attribute is used to keep track of what other *GeographicObjects* they have an association with.

SpotHeights are points on the earth's surface that are described by their three dimensional coordinates. The horizontal coordinate values are captured by the coordinate pair within the X and Y attributes of the *Point* class, while the Elevation attribute is used to document the vertical position.

Within the BPARSS prototype, *Buildings* have been associated with the *BuildingPermitApplication* instances (see Section 4.2.10). The PermitApplication attribute is used to record this association.

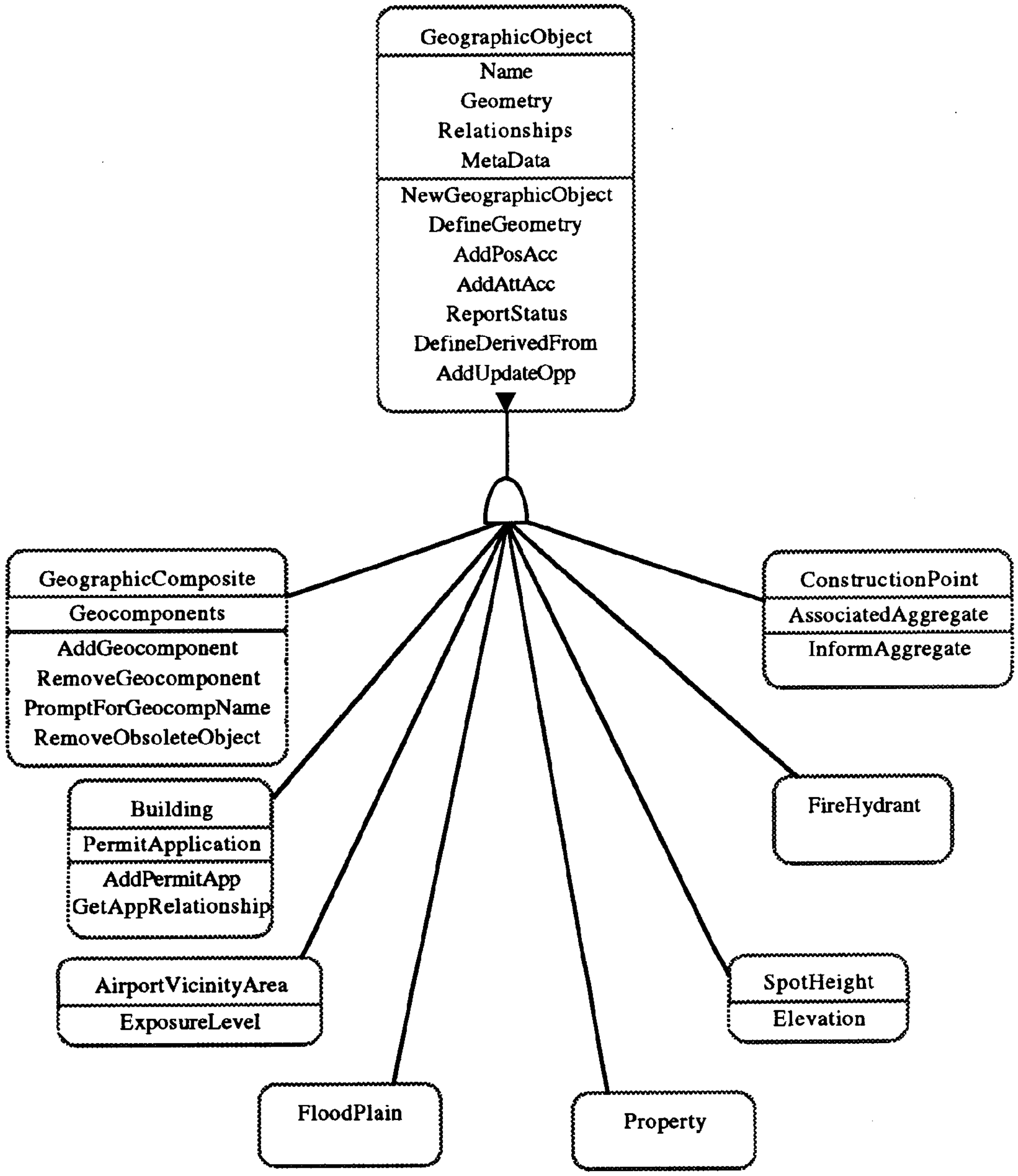

Figure 4.2.9 Geographic Object Types

AirportVicinityArea instances are polygonal features that are subject to high levels of noise due to their proximity to air traffic routes. These have varying degrees of ExposureLevel associated with them.

The *Property*, *FloodPlain* and *FireHydrant* classes do not possess business attributes that are considered important to the BPARSS prototype development. Their existence and location are what is significant to this study.

4.2.10 Building Permit Application

The business classes used by BPARSS that are not *GeographicObjects* are shown in Figure 4.2.10. The spatial analysis example provided in BPARSS is designed around the relationships between *BuildingRegulations*, *BuildingPermitApplications* and *GeographicObjects*. The central class in this relationship is *BuildingPermitApplication*.

The *Building* class provides the link between the *GeographicObjects* and the non-spatial classes. *Buildings* maintain their association with *BuildingPermitApplications* through the PermitApplication attribute. The relationships between *Buildings* and other *GeographicObjects* is based on spatial proximity.

A *BuildingPermitApplication* would typically include information on the application as well as details on the proposed building structure. In this example, the Date of the application, the proposed LandUse and the proposed AcousticStatus of the planned *Building* are recorded as attributes. These parameters, combined with the spatial considerations associated with the *Building*, are used to evaluate the *BuildingPermitApplication* against *BuildingRegulation* instances.

The *BuildingPermitApplication* instance has a mandatory many-to-many relationship with a *BuildingRegulation*; a *BuildingRegulation* must apply to one-to-many *BuildingPermitApplications* and each *BuildingPermitApplication* must be affected by one-to-many *BuildingRegulations*. The instance connection between *BuildingPermitApplication* and *BuildingRegulation* in Figure 4.2.10, represents a mapping between instances and is, therefore, not modelled by explicit class structures [Coad and Yourdon, 1991a]. For example, a *BuildingPermitApplication* does not need to record what *BuildingRegulations* apply to it; this is derivable information and can be established when necessary. The application must be designed to determine at run-time where these relationships exist.

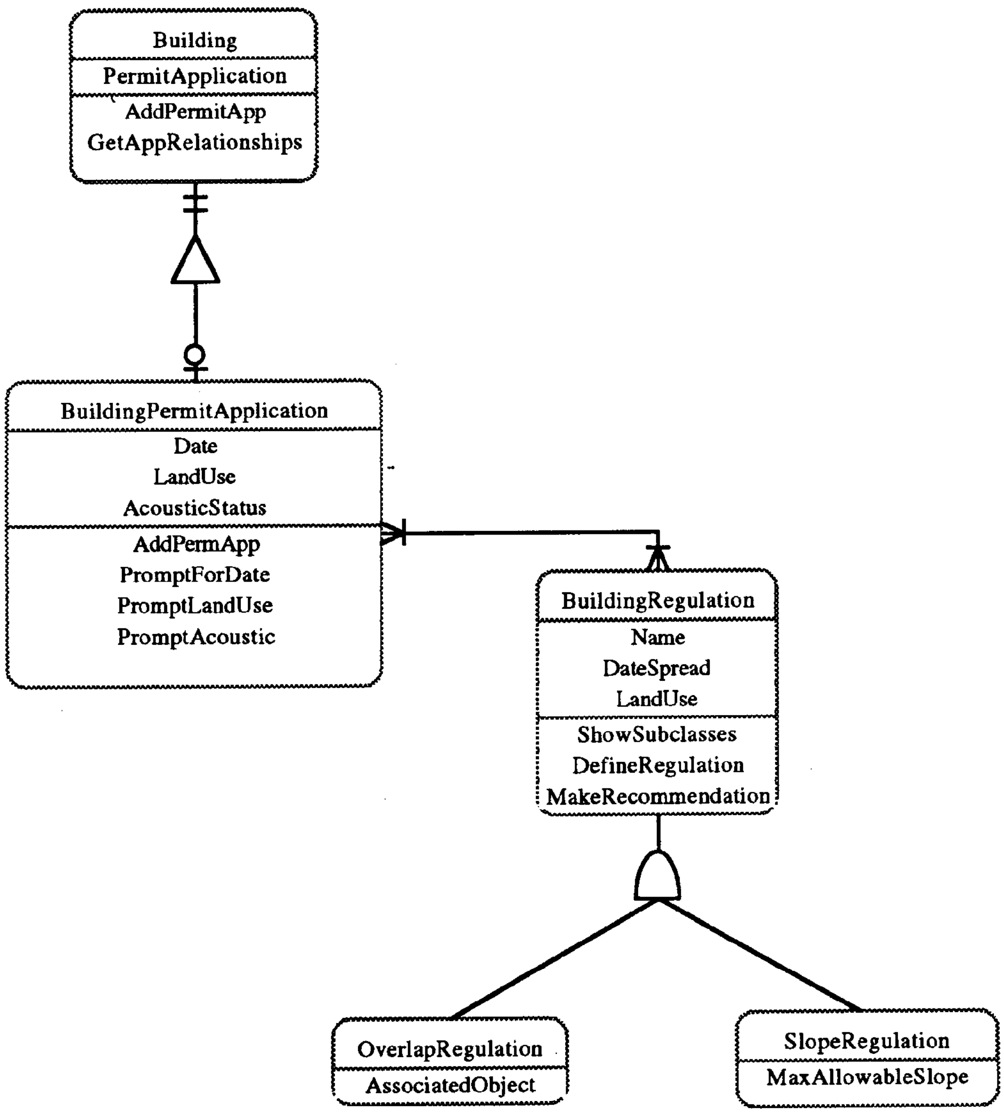

Figure 4.2.10 Building Permit Application

The *BuildingRegulation* class is an abstract class that has *OverlapRegulation* and *SlopeRegulation* subclasses associated with it. *BuildingRegulations* are Named, they are associated with a LandUse, and they have an allowable DateSpread between the *BuildingPermitApplication* Date and the date pertinent information on relevant *GeographicObjects* was collected.

OverlapRegulations represent regulations that are imposed due to spatial overlap between proposed *Buildings* and other *GeographicObjects*. The AssociatedObject attribute is used to establish what type of *GeographicObject* the regulation applies to.

SlopeRegulations are concerned with the degree of slope associated with the ground on which a proposed *Building* is to be built. The MaxAllowableSlope attribute records the upper limit of allowable terrain slope.

4.3 Summary

In this chapter, a prototype Building Permit Application Review Support System has been introduced. This test system will take advantage of the capabilities provided by the Object-Oriented approach to systems development to demonstrate the feasibility of the Data Quality Reporting concepts discussed in the previous chapters.

This chapter has provided the analysis stage of the system development life cycle. The SAIF data model has been used as a starting point and has been modified and extended to reflect the requirements of the prototype system. The Coad/Yourdon methodology has provided the techniques and structure needed to carry out the analysis. Generic Object-Oriented terminology and Coad/Yourdon specific conventions have been explained. The MacA&D CASE tool has been used as the primary documentation tool.

The purpose of developing the prototype system is to provide a means to demonstrate a generic approach to Data Quality Reporting within a GIS. For this reason, the spatial data prospective has been modelled in detail. In addition, a typical business GIS application has been included to provide a demonstration data set to support the prototype system.

A statement of purpose and list of functional requirements for both the DQR function and the business application have been included as the first steps in the analysis. The analysis is completed with detailed descriptions of eight views of the prototype system analysis model.

5 BPARSS DESIGN

5.1 Introduction

The analysis carried out in the preceding chapter has focused on defining the problem domain component of the project. The subject of this chapter is the application design, which can be considered the stage that defines the problem's solution; in this case the BPARSS system [MacDesigner, 1993]. Whereas the analysis stage is considered to be technology independent, a design (though still language independent) represents a commitment to a specific technology environment. The BPARSS design is committed to Object-Oriented technology.

As discussed in Section 4.2, many of the decisions made as a result of this design stage have been folded back into the structures presented in Chapter 4. In fact, insights realized as part of the construction stage (the subject of the next chapter) have also been incorporated into the analysis and design representations. This iterative approach to building applications is in alignment with the concurrent development principles used in the implementation of prototype systems [Coad and Nicola, 1993].

This chapter augments the problem domain structures discussed in Chapter 4, by describing the resulting application design. The techniques and conventions used to communicate the design additions are presented in the remainder of this section.

5.1.1 Design Techniques and Conventions

To supplement the four main identification and definition tasks discussed in Section 4.1.3, four components and their associated activities are normally employed in the Coad/Yourdon design methodology. The components include Problem Domain Component, Human Interaction Component, Task Management Component and Data Management Component [Coad and Yourdon, 1991b].

The Problem Domain Component (PDC), within the design phase, consists of improvements and additions to the results of the analysis. These changes are a result of an increased knowledge of the problem, as well as the influence of technology based decisions.

The Human Interaction Component (HIC) is responsible for providing the communication link between the user of an application and the problem domain. It is within the HIC that command hierarchy and other interface issues are considered. HIC classes are also defined and their relationships to the other design components are developed.

The Task Management Component (TMC) is used to model and design the task structure within multitasking applications. By considering the tasks separately, the overall design is simplified within these systems. Since the BPARSS application is a single task system, the TMC will not be considered further in this thesis.

The Data Management Component (DMC) provides the means to access objects from a data management system. The separation of this component from the others simplifies the modelling process.

In addition to the component definitions, object communication models have been included in the design. These diagrams are used to describe the communication flows between objects. Objects are represented by rounded boxes with methods shown as rectangles protruding from the object symbols. Messages sent between methods are shown by arrow symbols connecting them; direction of the message is indicted by the arrow head. Parameter passage is depicted as a coupling symbol, as illustrated by the "ObjectList" parameter in Figure 5.2.4.

5.2 The Design

The BPARSS design described in the remainder of this chapter provides a high level overview of the detailed design available in the CASE tool. The interface is described and several sample object communications are outlined to provide the reader with a sense for the overall prototype design.

Although the design is presented, for the most part, as a language independent product, considerations introduced by the Smalltalk environment used for BPARSS construction (the subject of the next chapter) have been incorporated into the design structures.

The design documentation firstly provides descriptions of the Problem Domain, Human Interaction and Data Management components. It then outlines eight key user invoked events through narrative explanations and object communication diagrams. The object communication diagrams model the behavioral aspects of objects. Object communication diagrams are available for other methods in the MacA&D model.

5.2.1 Problem Domain Component

As discussed above, the Problem Domain Component enhancements introduced during design have been incorporated into the models reviewed in Chapter 4. The key interactions between the PDC and the HIC are included in the object communication diagrams reviewed starting in Section 5.2.4.

5.2.2 Human Interaction Component

The BPARSS HIC structure is shown in Figure 5.2.2. The *BPARSS* class has been modelled as a subclass of the generic *Application* class (not shown). BPARSS has four main attributes: ProtoWorld, ProtoLand, ObjectList and SelectedObject. The ProtoWorld attribute is used to link the HIC to the DMC, as the prototype data set is stored in a global variable of the *ProtoWorld* class (see Section 5.2.3). The ProtoWorld variable can accommodate multiple data sets. This has been done to allow for more than one test data set (or *ProtoLand*) to be available to BPARSS. The chosen data set is determined by an assignment to the ProtoLand attribute. ObjectList is used to store the contents of a set (or list) of objects, while SelectedObject is used to indicate which instance in a set is of interest at a particular time.

The methods that can be invoked by the user are displayed as part of the *BPARSS* class in Figure 5.2.2. There are three categories of methods shown; listing methods, data editing methods and action invoking methods. These categories are discussed in detail later in this section. The first five methods represent listing methods, the next fourteen are data editors and the remaining seven are action invoking methods. The triangle at the bottom of the *BPARSS* symbol indicates that the CASE tool includes other methods (not invoked directly by the user).

The *Window* class is a generic OO class that has been used in the prototype to provide interaction with the BPARSS user. The methods and attributes associated with *Window* are not shown in this diagram. There is a single *Window* instance associated with any BPARSS session.

There are three panes used to assemble a BPARSS window. Each one of these panes must be part of a BPARSS session and is used to display information as a result of an event initiated by the user. The first pane type is a *GraphPane*, which is used to display graphical representations of information. The *ReportPane's* function is to provide textual descriptions of instructions and reports. The *ListPane* provides lists of objects that can be selected and then manipulated. The key communication methods for these three classes are shown in Figure 5.2.2.

There are three *Menus* associated with a BPARSS *Window*. These are used to access the three categories of user invoked methods discussed above. Menus are composed of Labels and Actions, which are the key attributes of the *Menu* class.

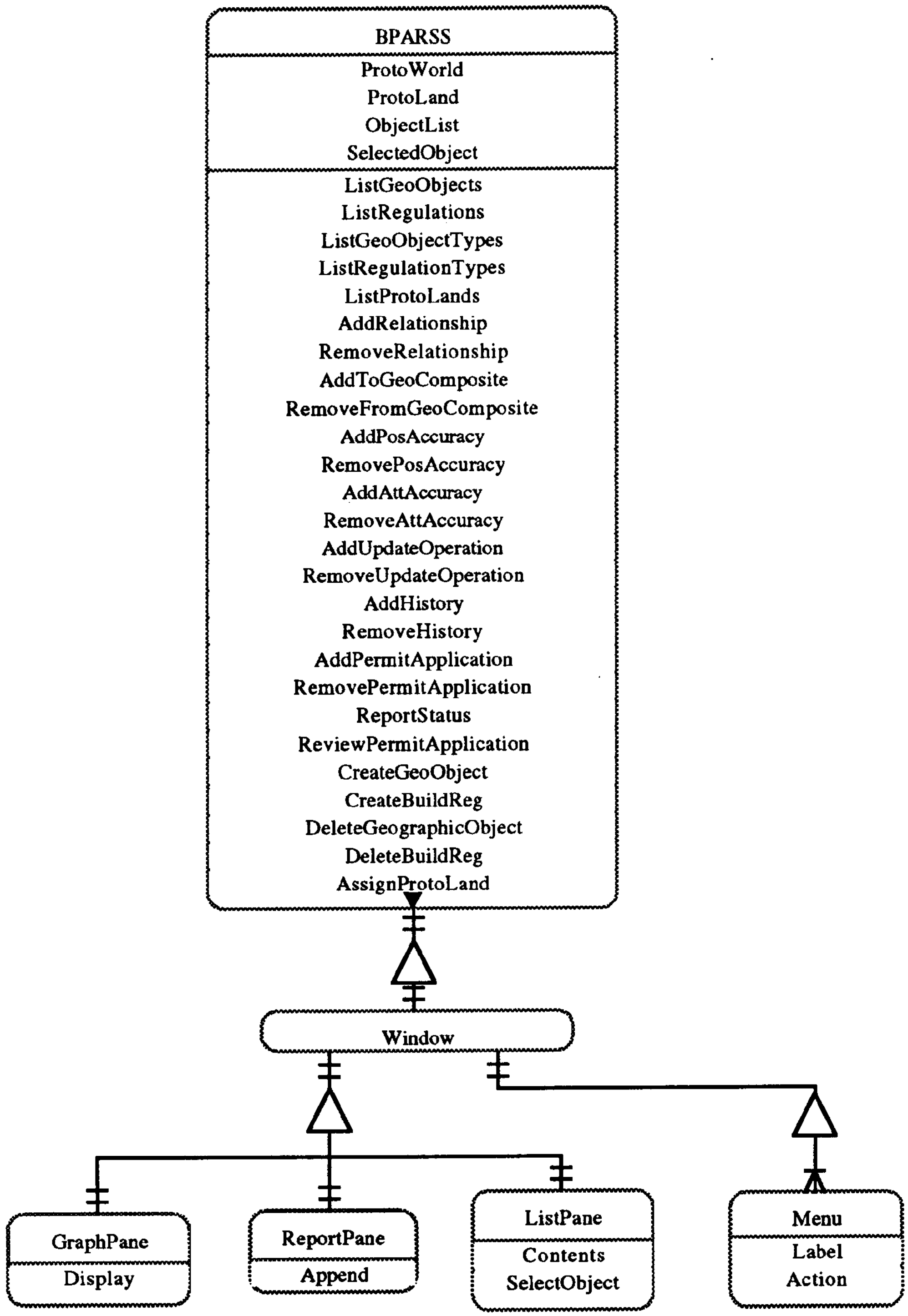

Figure 5.2.2 Human Interaction Component Structure

Much of the HIC design in the Coad/Yourdon methodology is focussed on designing the interface to accommodate the application user. Although this is a sound approach and critical to production system development, there is a limited user community (i.e., the author) for the BPARSS prototype. For this reason, little emphasis has been given to sophisticated interface design.

The interface itself is based on a list-select-execute event series. To initiate this protocol, a user lists a set of objects (to the ListPane), selects the object of interest (from the ListPane) and executes an Edit Data or Action method. All list-select-execute methods are user invoked, and are displayed in Table 5.2.2. Table 5.2.2 portrays the menu structure discussed above.

List	Edit Data	Action
ListGeoObjects	AddRelationship	ReportStatus
ListRegulations	RemoveRelationship	ReviewPermitApplication
ListGeoObjectTypes	AddToGeoComposite	CreateGeoObject
ListRegulationTypes	RemoveFromGeoComposite	CreateBuildReg
ListProtoLands	AddPosAccuracy	DeleteGeographicObject
	RemovePosAccuracy	DeleteBuildReg
	AddAttAccuracy	AssignProtoLand
	RemoveAttAccuracy	
	AddUpdateOperation	
	RemoveUpdateOperation	
	AddHistory	
	RemoveHistory	
	AddPermitApplication	
	RemovePermitApplication	

Table 5.2.2 User Invoked Methods

A 'list' command is initiated by choosing one of the methods available on the List menu. The List methods are responsible for assembling and displaying the information necessary for a user to make a selection. This list of objects is temporarily stored in the BPARSS ObjectList attribute. BPARSS lists are limited to the five object types shown in Table 5.2.2.

A 'select' operation is invoked through the Smalltalk interface. This is simply a point and click procedure on the *ListPane*. The selected object is temporarily stored in the BPARSS SelectedObject attribute.

Edit Data and Action methods constitute the 'execute' events and are object type dependant, in that they are not all applicable to all object types. For example, the only method that is possible to execute against a *ProtoLand* selection is AssignProtoLand (*ProtoLands* are created and deleted outside BPARSS). The ReportStatus method is the only method that is applicable to all listable classes (with the exception of *ProtoLand*). All other Edit Data and Action methods apply only to certain object types. Their applicability can be determined from their names.

Edit Data methods are used to access the information stored as object attributes. The Action methods serve three basic purposes. First, the AssignProtoLand method is a BPARSS initiation operation which links the application to the data. This is a mandatory first step in a BPARSS session. The ReportStatus and ReviewPermitApplication methods are used to query the data, and result in reports. The remaining Action methods serve the third main purpose. They are used to create and delete the two main object types: *GeographicObjects* and *BuildingRegulations*.

5.2.3 Data Management Component

Only a limited data set is needed to demonstrate the DQR functionality in BPARSS. This has made it possible to simplify the BPARSS development and increase the response time by using a simplified Data Management Component. Instead of incorporating a sophisticated Data Base Management System, all data is stored in RAM by a Smalltalk global variable. This is discussed in more detail in the next chapter.

5.2.4 List ProtoLand

The first step in any BPARSS session is to assign the data set, which is stored in an instance of the *ProtoLand* class. This is accomplished through the normal three step (list-select-execute) series of requests. The generic three step process is described in detail in this section and the next one.

First, all possible *ProtoLands* are listed through a ListProtoLands selection. This results in BPARSS requesting the *ProtoWorld* instance to return the set of available *ProtoLands* which are then stored in the ObjectList attribute. BPARSS then instructs the *ListPane* to display this set in the form of a selectable list. The *ReportPane* is told to display a set of user instructions. The ListProtoLands methods are diagrammed in Figure 5.2.4.

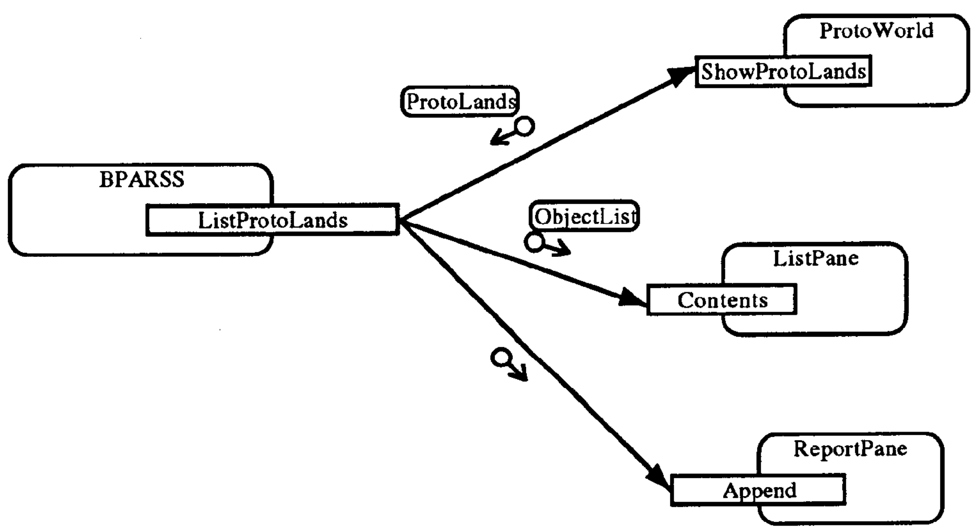

Figure 5.2.4 List ProtoLands

5.2.5 Assign ProtoLand

After the *ProtoLands* have been listed, one is selected by 'pointing and clicking'. on the preferred *ProtoLand* in the *ListPane*. This 'select' step results in the selected instance being stored in the BPARSS SelectedObject attribute.

The final (execute) step in the three step interface protocol, in this example, is initiated by an AssignProtoLand request. This is accomplished by clicking on the AssignProtoLand method in the Action Menu list. At this point, the selected *ProtoLand* instance is assigned to the BPARSS ProtoLand attribute and a message indicating this is sent to the *ReportPane*, as illustrated in Figure 5.2.5. After this has been completed, all other operations are possible.

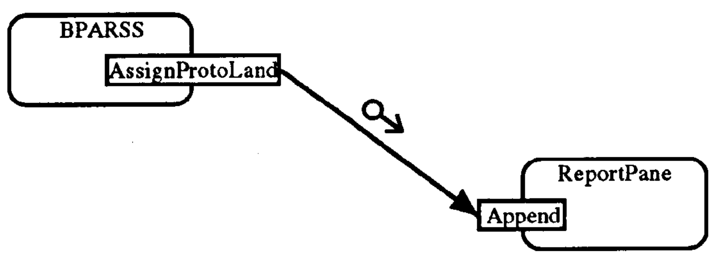

Figure 5.2.5 Assign ProtoLand

5.2.6 List Geographic Objects

A more complex example of a list method is provided in Figure 5.2.6. Here, in addition to providing a list of all available *GeographicObject* instances on the *ListPane*, the ListGeoObjects method asks the individual objects to provide a graphical representation of themselves on the *GraphPane*.

The ListGeoObjects method asks the assigned *ProtoLand* to return a set of available *GeographicObjects*, which BPARSS stores in the ObjectList attribute and instructs the

ListPane to display. This results in the *ListPane* listing the names of all the *GeographicObject* instances. BPARSS then instructs the *ReportPane* to notify the user to *"Please select the GeographicObject you would like to access from the ListPane. Then choose the appropriate action from the Action or Edit Data menu."*. In addition to returning the set of *GeographicObjects* to BPARSS, the *ProtoLand* is instructed to ask each *GeographicObject* to display itself on the BPARSS *GraphPane*. In order to do this each *GeographicObject* asks its respective *GeometricObject* instance to draw itself directly on the *GraphPane*. This results in the *GraphPane* displaying a map of the *GeographicObjects* included in the *ProtoLand*.

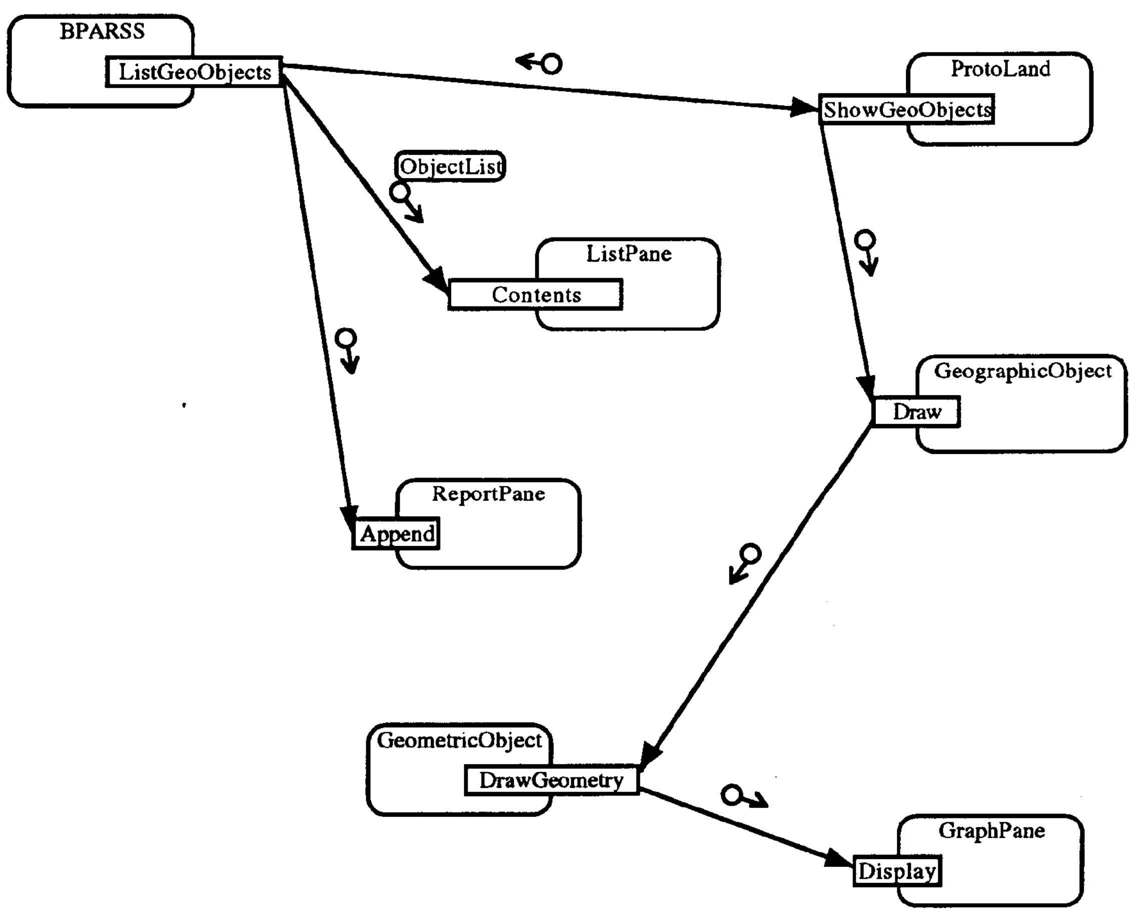

Figure 5.2.6 List Geographic Objects

5.2.7 List Regulation Types

A final list-based method is portrayed in Figure 5.2.7. This example differs from the previous two in that it does not communicate with an instance of a class. Instead the ListRegulationTypes method communicates directly with the *BuildingRegulation* class. It triggers a class method named ShowSubclasses, which returns a set of all valid subclass names. Again, BPARSS stores this set in the ObjectList and sends the appropriate Contents and Append messages. There is, however, no communication with the *GraphPane*, as a class does not have Geometry associated with it and cannot produce a graphical display. A similar approach is used for the ListGeoObjectTypes method.

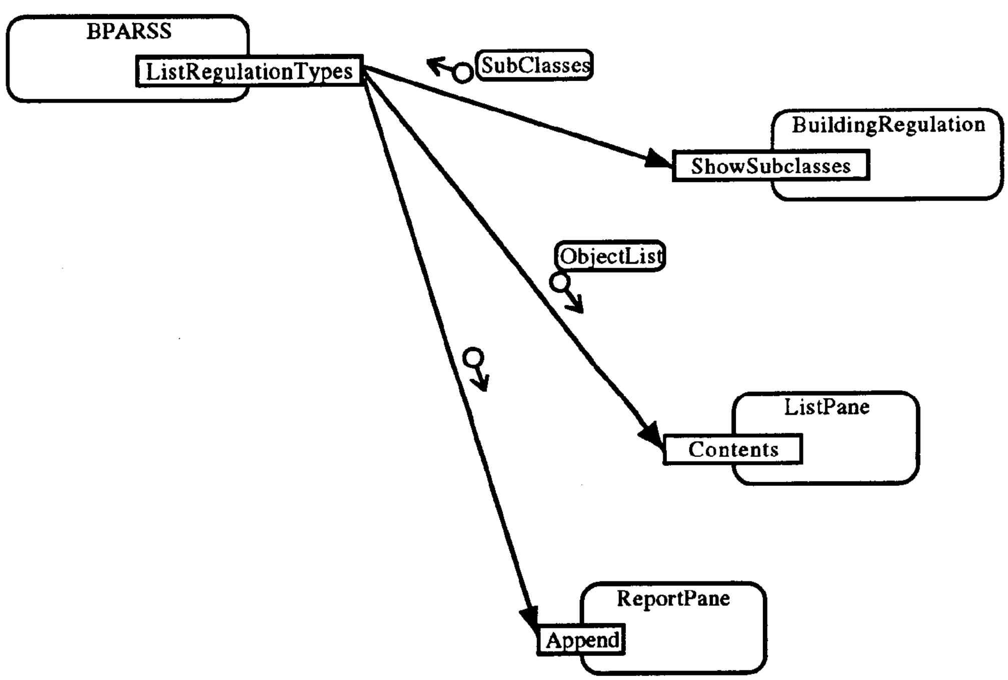

Figure 5.2.7 List Regulation Types

5.2.8 Create Geographic Object

The CreateGeoObject method provides an example of how instances of key object types are generated in BPARSS. As shown in Figure 5.2.8, this method asks the *ProtoLand* to add a new instance of the selected *GeographicObject* subclass to its GeographicObject set (attribute). Note that the type of *GeographicObject* must be selected from a list generated through a procedure similar to the one discussed in Section 5.2.7. The *ProtoLand* sends a message to the appropriate *GeographicObject* subclass asking it to create a new instance. As part of the object initialization, its Geometry and business attributes are created. The Geometry and attributes, if any exist, are dependant on the object's class. Confirmation of the object creation is provided through a message to the *ReportPane*.

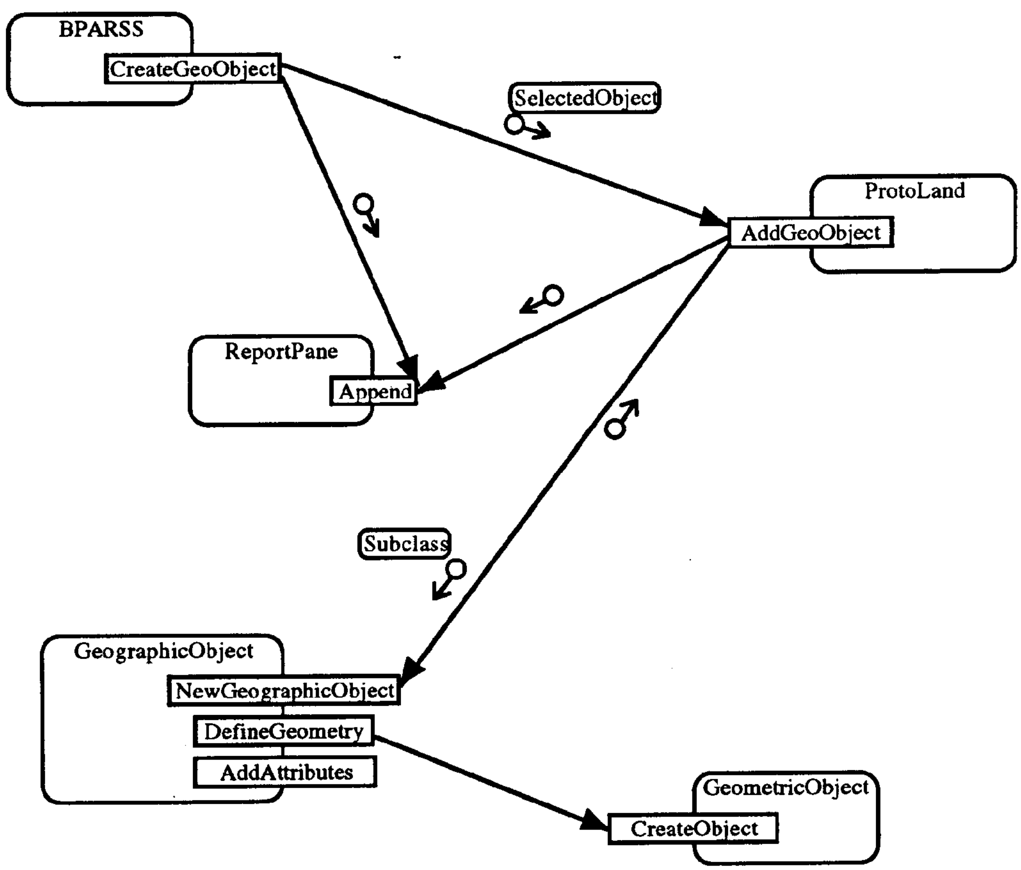

Figure 5.2.8 Create Geographic Object

5.2.9 Add Positional Accuracy

All quality parameters are added and deleted through Edit Data operations in BPARSS. The AddPosAccuracy method illustrated in Figure 5.2.9, is an example of a positional accuracy statement addition.

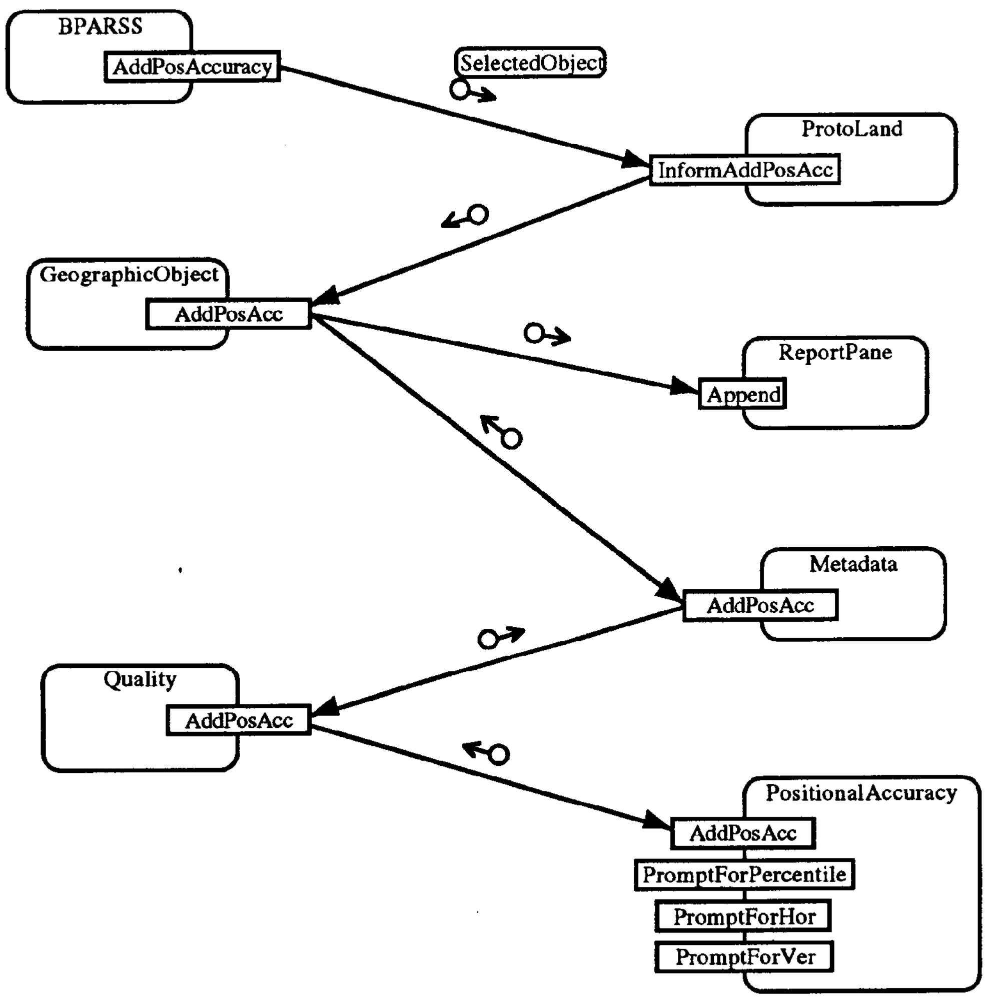

Figure 5.2.9 Add Positional Accuracy

Figure 5.2.9 shows that the prompting and assignment of values is carried out by a *PositionalAccuracy* instance. This example provides an indication of the depth of the object structures associated with the SAIF model. A *PositionalAccuracy* object is defined as an attribute of a *Quality* instance, which is an attribute of a *Metadata* instance, which is an attribute of a *GeographicObject* (see Figures 4.2.5 and 4.2.8). All of these parameters are actual objects.

This design is in fact a simplification of the SAIF model. In SAIF, a *PositionalAccuracy* object does not directly contain coordinate accuracy information. Instead, it has a Relative attribute which is assigned an instance from the *CoordinateAccuracy* class, which contains the coordinate accuracy values. The BPARSS design has incorporated this modification as it is seen as a means to reduce the complexity of the prototype. If a production system was developed with this design, and at a latter date it was decided that the pure SAIF model was required, a simple modification to the data structures and methods would provide the necessary functionality. This is a prime example of how Assembly-Parts structures can be used to support application extensibility. Assembly-parts extensibility is discussed further in Section 7.6.3.

There is a related BPARSS design simplification with respect to absolute accuracy. In SAIF, *PositionalAccuracy* also has an Absolute attribute. This is assigned an instance from the *AbsoluteAccuracy* class which has two attributes; HigherOrderSource and CoordinateAccuracy. HigherOrderSource is a textual description of what 'absolute' is referenced to (i.e., 'control'). The CoordinateAccuracy attribute requires an assignment from the *CoordinateAccuracy* class, discussed above.

Within SAIF the Absolute and Relative attributes are optional; one or the other is expected. The BPARSS design has incorporated a default absolute accuracy, relative to a higher order 'control' network. Instances where positional accuracy is not absolute are indicated by the existence of a *DerivedFrom* (a subclass of *Relationship*) relationship for a *GeographicObject*. Examples of this are provided for the *ConstructionPoints* used to define *Building* instances (see Section 7.3).

5.2.10 Report Status

As discussed in Section 5.2.2, the ReportStatus method applies to most listable objects. The example provided in Figure 5.2.10 shows how this is implemented for a *GeographicObject* instance.

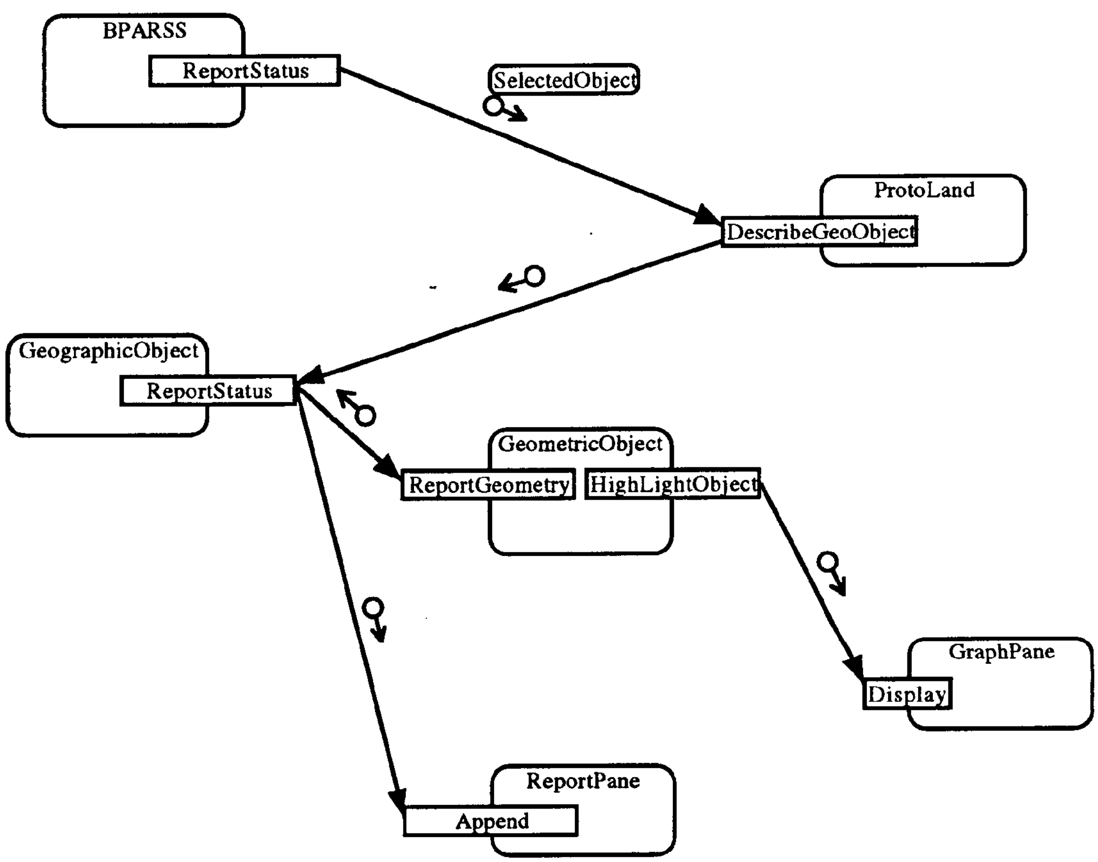

Figure 5.2.10 Report Status

Figure 5.2.10 shows at a very high level the two reporting forums for a *GeographicObject*: the *ReportPane* and the *GraphPane*.

A textual description is generated by the *GeographicObject's* ReportStatus method. This is a multi-step process which is described in the CASE tool documentation with the following description: "*This method returns all information that is available on the status of a GeographicObject. This method calls ReportName, ReportUpdateOperation, ReportHistory, ReportAttributes, ReportAttAcc, ReportGeometry, ReportRelationships and ReportPosAcc.*". The ReportStatus method compiles the textual descriptions returned from all the methods it calls and provides a report to the *ReportPane.*

The *GeometricObject* ReportGeometry method provides the textual response required by the *GeographicObject*'s ReportGeometry method. In addition, it triggers the HighLightObject method which determines how the *GeometricObject* should be displayed and provides a representation directly to the *GraphPane.*

5.2.11 Review Permit Application

The object communication associated with a ReviewPermitApplication request is diagrammed in Figure 5.2.11. The first step is to determine what the LandUse is for the selected *BuildingPermitApplication.* This is established by searching the Assembly-Parts structure to access the LandUse attribute of the *BuildingPermitApplication* instance. When this is returned, the DetermineRegulations method asks the *ProtoWorld* object to determine and return descriptions of all *BuildingRegulation* objects that have a LandUse designation equal to the value assigned to the *BuildingPermitApplication.*

The next step is to evaluate each relevant regulation with respect to the spatial and temporal relationships between a proposed *Building* and other *GeographicObjects*. This is accomplished by invoking the GetAppRelationships methods, which return descriptions of any existing relationships. Spatial relationships are indicated on the *GraphPane.* The relationship determination and graphical display object communications are not shown in Figure 5.2.11.

Descriptions of the relationships are sent to the *BuildingRegulation* subclasses for review by the *BuildingRegulation* subclass instances. These instances carry allowable tolerances as attributes, and the classes have methods which review the relationship descriptions with respect to the allowable tolerances. Based on these reviews, the *BuildingRegulation* instances make recommendations which are returned to BPARSS for reporting.

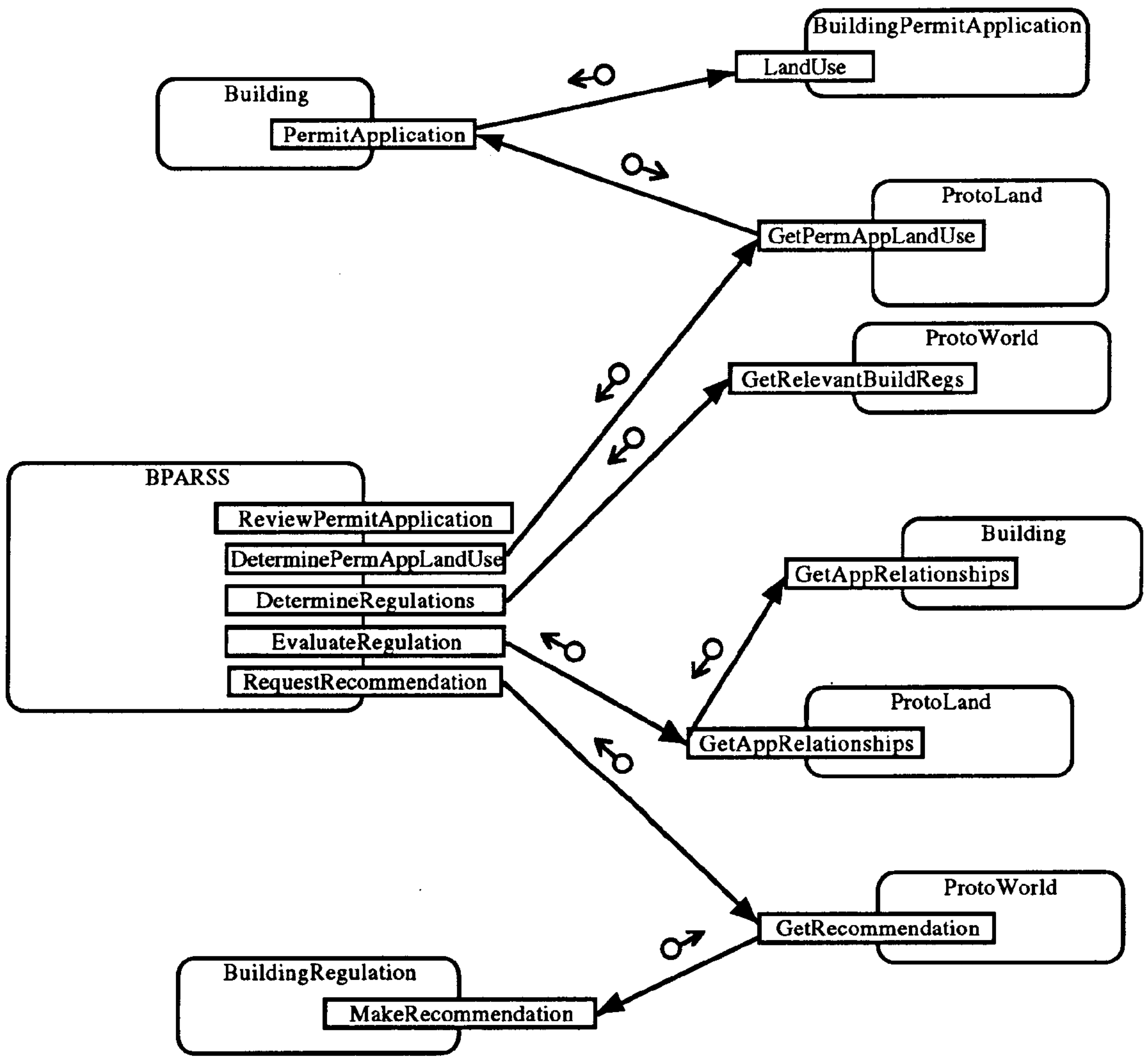

Figure 5.2.11 Review Permit Application

The structure described, positions the decision support 'rules' within the *BuildingRegulation* classes. This facilitates potential modifications to the *BuildingRegulation* class structure in the form of enhancements to the 'rules base' as well as extension to the types of *BuildingRegulations* available.

This component of the BPARSS design provides the means to determine the relationship between a *BuildingPermitApplication* and a *BuildingRegulations*, as discussed in Section 4.2.10.

5.3 Summary

This chapter has extended the definition of the BPARSS prototype through the design specifications presented in this chapter. Although the design stage does not define the implementation environment, the decision to adopt Object-Oriented technology has been made at this level.

The design has been described in terms of definitions of the Problem Domain Component (extension of Chapter 4 definition), the Human Interaction Component and the Data Management Component. Within the HIC, the BPARSS structure, the interface protocol and the menu design have been explained.

The interaction between the PDC and the HIC has been revealed by describing eight sample object communication diagrams. These diagrams and their accompanying textual references are used to plan the object flow prior to construction, and to document the actual implementation. While they do not record every individual object communication, they provide the frame of reference needed to understand the application's overall logic. The diagrams provided in this chapter represent a sample of the design documentation available in the CASE tool.

6 BPARSS CONSTRUCTION

6.1 Introduction

The final stage in the system development life cycle is the construction stage. The construction activity, which is the subject of this chapter, transforms the analysis and design results from system specifications to a functioning application.

The BPARSS prototype application has been constructed utilizing the Smalltalk Object-Oriented development environment. This chapter explains the Smalltalk related considerations that are relevant to the BPARSS development. This final stage of development is described in terms of the interface, the data population and the algorithms employed.

6.1.1 Smalltalk

Smalltalk is a pure Object-Oriented development environment. It provides a complete suite of features including an editor, a compiler, a debugger, a windowing system and a source code manager [LaLonde and Pugh, 1990]. In addition, it incorporates a library of hundreds of existing classes. This extensive library can be extended by a user to develop interactive graphical applications based on Object-Oriented principles.

Although Smalltalk is not a GIS, it has been possible to simulate basic GIS functionality within BPARSS. This has been accomplished using the existing Smalltalk class/method library, as well as BPARSS-specific structures and object communications.

In order to reduce complexity and increase performance, BPARSS has stored its entire test data set in RAM, within a Smalltalk global variable. A global variable is a variable that can be accessed by any object in the Smalltalk environment. These variables exist until they are explicitly deleted. In the BPARSS case, an instance of the *ProtoWorld* class has been defined as a global variable. This approach has been possible due to the limited size of the BPARSS test data set.

BPARSS has been developed using the Smalltalk/V dialect of Smalltalk on a Macintosh computer [Smalltalk, 1993]. Version 2.0 of Smalltalk/V has been used, which is the first release of a significant product upgrade.

6.2 BPARSS Interface

Within a Smalltalk session, BPARSS is accessed by executing the "BPARSS openOn: AWorld" message. This message tells the BPARSS class to create an instance of itself and associate it with the 'AWorld' global variable. 'AWorld' is the instance of *ProtoWorld* which contains the data BPARSS uses.

6.2.1 BPARSS Window

Figure 6.2.1 is a representation of the blank BPARSS window that is displayed when BPARSS is initiated. The entire BPARSS interaction is carried out through this main window and a series of Smalltalk *Prompter* windows. Figure 6.2.1 shows the three panes; the *ReportPane* in the top left corner, the *ListPane* in the top right and the *GraphPane* in the bottom section of the window. The *ReportPane* and *ListPane* are scrollable to facilitate lengthy textual displays. The *GraphPane* is used for display purposes only; this is a standard Smalltalk class and cannot be used to access the data or interface with BPARSS.

The pulldown menu titles are also shown. The menu titled 'List' contains the listing methods, the 'Edit Data' menu provides data editing methods and the 'Action' title controls the action invoking methods. The methods that can be triggered through these menus have been described in Section 5.2.2. The other menu titles shown in Figure 6.2.1 are generic Smalltalk and Macintosh menus.

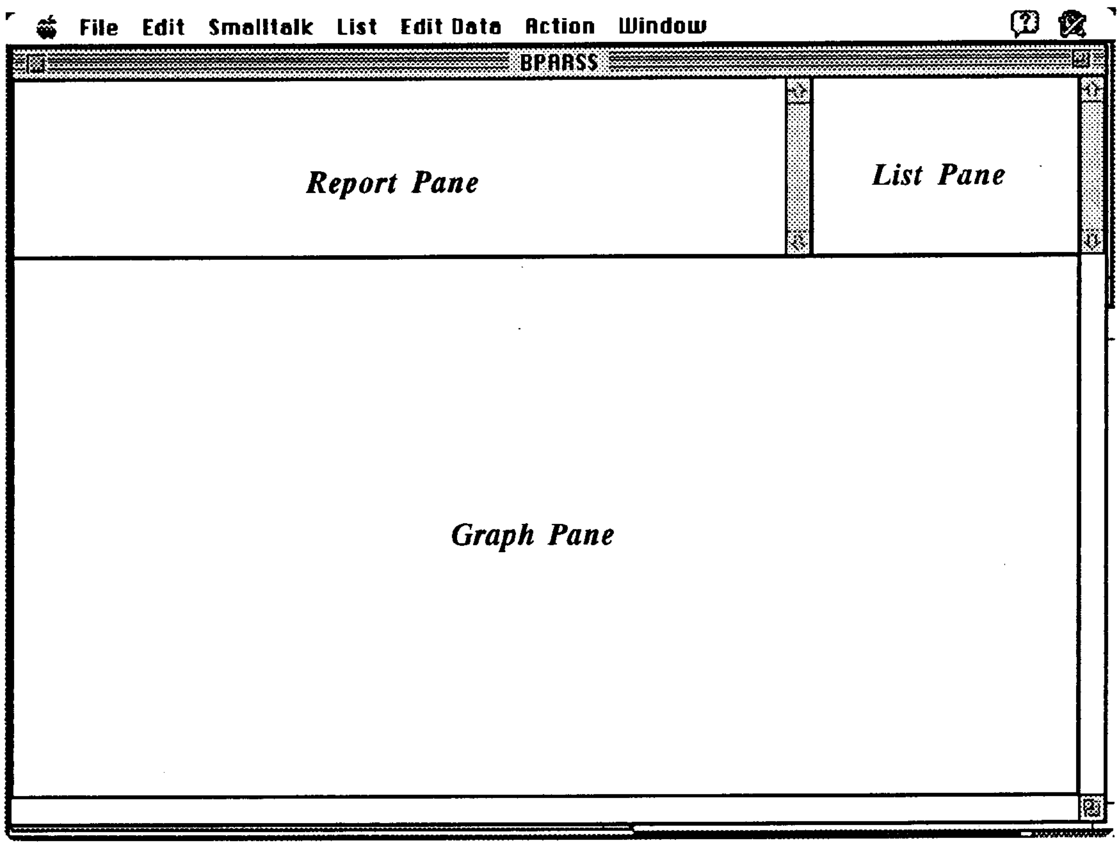

Figure 6.2.1 BPARSS Window

6.2.2 Data Editing

BPARSS is a prototype application and its data editing capabilities have been designed with this in mind. The data editing interface is not sophisticated, but it does allow for rapid data entry. It is not possible to update data; all BPARSS data editing is achieved through additions and deletions of instances. This does not introduce a maintenance concern, as the data set is small and the data structures are modularized (i.e., each data quality class can be accessed directly).

There are no explicit valid values tests imposed by the BPARSS application. There are, however, implicit valid value tests inherent in the use of the existing Smalltalk class library and the BPARSS class structures. For example, by using *Date* instances to store information associated with dates, entries that are not in a *Date* format are not acceptable. The same applies to *Number* and *String* assignments. In addition, the list-select-execute protocol ensures that an attempt to create a new instance cannot be for an invalid object class (e.g., the only way to create an instance of a *GeographicObject* is to select the class from a list of valid *GeographicObject* subclasses).

Data integrity issues have been considered, but only in certain cases to demonstrate how this can be achieved. An example of this is the restriction imposed on attempting to delete any *GeographicObject* instance that has a dependency relationship with another *GeographicObject*.

Within BPARSS, data is added and deleted using the list-select-execute protocol described in Section 5.2.2. When data is added, the appropriate parameters are asked for through *Prompter* windows. Suggested values are normally supplied within the *Prompter*. Requests for data deletion require confirmation to ensure data is not accidently deleted.

6.3 Data Population

As discussed in Section 6.2, all BPARSS data is stored in the global variable named 'AWorld'. Assigned to 'AWorld' are sets of *ProtoLand* and *BuildingRegulation* instances. This allows multiple *BuildingRegulation* instances and geographic data sets to be available to a BPARSS session. The data used in the test data set has been created for example purposes only. The data content represents simplified simulations of real world scenarios.

6.3.1 Building Regulations

There are six *BuildingRegulations* assigned, which define regulations based on land use and relationships to *GeographicObjects*. Both commercial and residential land use are accommodated. Proximity to *AirportVicinityAreas* and *FloodPlains*, as well as the influence of the slope of the terrain are considered. A complete list of all available *BuildingRegulations* is provided in Table 6.3.1, along with their spatial and temporal constraints.

Name	Land Use	Associated GeographicObject	Spatial Constraint	Temporal Constraint
AirCom	commercial	*AirportVicinityArea*	overlap	12 months
AirRes	residential	*AirportVicinityArea*	overlap	12 months
FloodCom	commercial	*FloodPlain*	overlap	12 months
FloodRes	residential	*FloodPlain*	overlap	12 months
SlopeCom	commercial	*SpotHeight*	slope	24 months
SlopeRes	residential	*SpotHeight*	slope	24 months

Table 6.3.1 Building Regulation Data Population

6.3.2 Geographic Objects

A single *ProtoLand* named 'NewLand' has been assigned to 'AWorld' as a demonstration data set for this thesis. 'NewLand' has 54 named *GeographicObjects* assigned to it. Table 6.3.2 shows the number of instances associated with each *GeographicObject* subclass. The *GeographicObjects* have multiple *Geometry, Relationship* and *MetaData* instances assigned to them.

GeographicObject Subclasses	# of Instances
AirportVicinityArea	1
Building	4
ConstructionPoint	35
FireHydrant	1
FloodPlain	1
GeographicComposite	4
Property	4
SpotHeight	4

Table 6.3.2 Geographic Object Data Population

6.4 BPARSS Approaches and Algorithms

In developing a prototype, demonstrability should be the focus rather than rigour. The BPARSS application has used a number of simplified techniques that provide the appropriate level of functionality for a prototype system. These include utilizing existing Smalltalk classes and methods, which are not designed to support GIS concepts, as much as possible. In addition, where new classes and methods have been introduced, approximate algorithms have often been used for development simplification or performance reasons. This section describes several approaches and algorithms used in BPARSS. The topics discussed are concepts that are relevant to DQR. The more generic application development considerations associated with BPARSS are not reviewed in this thesis.

6.4.1 Graphical Display

One of the ways BPARSS addresses the 'meaningful' component associated with practical DQR, is through the graphical display. The *GraphPane* provides spatial context within any query by indicating where a specified object is in relation to the entire *GeographicObject* data set. In addition, spatial data quality parameters are indicated using methods such as those discussed in Section 6.4.3.

The BPARSS *GraphPane* is refreshed as part of any 'list' or 'execute' request that involves *GeographicObject* instances. In order to provide the spatial context discussed above, all *GeographicObjects* are displayed as part of any graphics refresh, with the exception of *ConstructionPoints*. *ConstructionPoints* are not displayed in a graphics refresh because they clutter the display and affect BPARSS performance. Individual *ConstructionPoints* are, however, displayed when there is a request to report their status.

6.4.2 Coordinate Transformation

The BPARSS *GraphPane* is restricted to approximately 470x320 pixels, due to hardware constraints imposed by the computer used for the application development. In order to be able to graphically display the test data to the resolution required to demonstrate DQR principles, a coordinate transformation has been utilized that equates a BPARSS pixel to decimetre accuracy. The transformation is applied by the *BPoint* PixelToUTM method.

By being able to display the data at decimetre resolution, it has been possible to indicate *GeographicObject* locations and positional accuracy indicators to the same scale. This is considered important for providing meaningful results.

The cost of providing this level of resolution is that the spatial extent available for displaying test data is restricted to an area of about 60x30 metres. This restriction has not posed a problem in demonstrating the DQR concepts described in this thesis.

6.4.3 Error Corridor Determination

The approach to providing a positional accuracy indication for polygonal features has been to define 'buffer zones' for the polygon sides, which represent error corridors. This is an extension of the geodetic concept of error ellipses [Bomford, 1980] and similar to the Epsilon Band approach [Blakemore, 1983]. The width of the error corridor is based on the standard deviations of the end points of the sides of the polygon of interest (actual polygon). This has been determined by calculating an 'inner polygon' and an 'outer polygon', and defining the error corridor as the area between the two new polygons. Figure 6.4.3 illustrates this approach.

The points that define the inner and outer polygons are calculated based on the position and positional accuracy of the points that define the actual polygon. The error corridor is the area in which the true boundary is expected to be located, at a stated confidence level. In this case, the confidence level is assumed to be equivalent to that associated with the standard deviations of the points used to determine the boundaries.

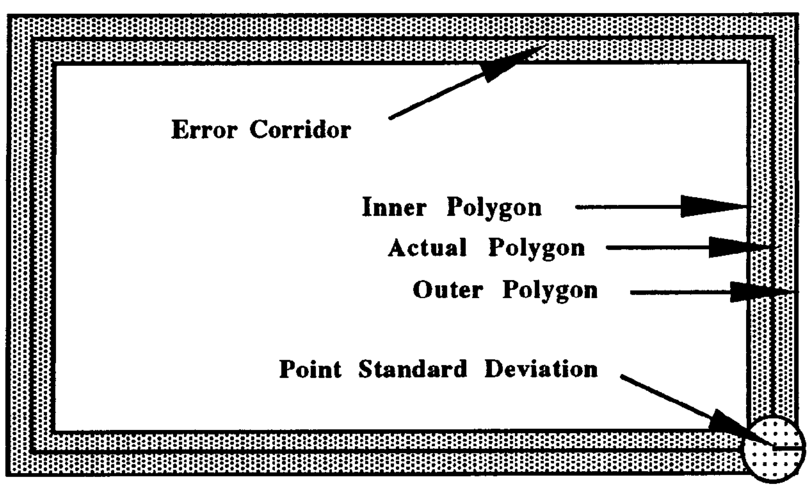

Figure 6.4.3 Error Corridor

6.4.4 DQR Level Determination

Within BPARSS, a hierarchy of reporting levels has been incorporated for the Entity Type - Data Supplier - Entity Occurrence approaches to DQR discussed in Section 3.3. This approach allows an object to 'acquire' data quality information from the higher levels in the hierarchy, if necessary. BPARSS utilizes an algorithm that retrieves data quality information at the lowest possible level in the hierarchy, as illustrated in Figure 6.4.4. Its searching routine first checks the Entity Occurrence level to see if the required information is available. If it exists at that level, the appropriate DQR response is given; if the Entity Occurrence does not contain the information, BPARSS checks to see if the instance is part of a Data Supplier report. If so, a response is given; if not, BPARSS lastly searches the Entity Type for default data quality information.

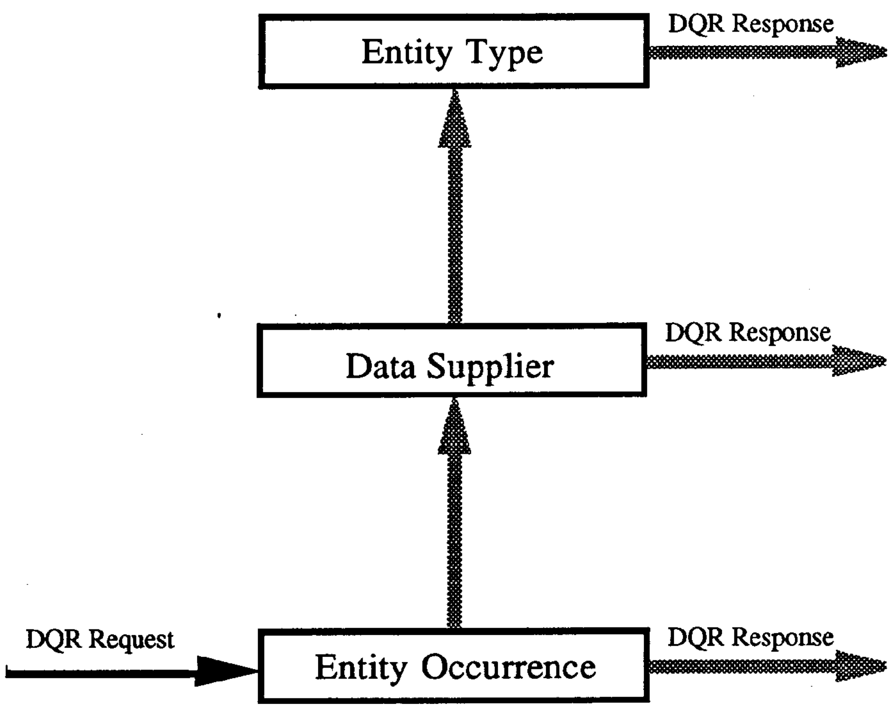

Figure 6.4.4 DQR Level Hierarchy

Within BPARSS, Entity Occurrence level information is implemented by object instances. Data Supplier reports have been assigned to *GeographicComposites*. Entity Type reporting is invoked by Smalltalk classes. Detailed examples of how this hierarchy operates are given in Section 7.2.2.

Spatial Partition reporting has not been implemented in BPARSS. The limitations associated with the *GraphPane* do not support the graphics based definition of spatial extent required by this approach. Spatial Partition reporting would be positioned between Entity Occurrence and Data Supplier reporting in the hierarchy shown in Figure 6.4.4.

6.4.5 Topology Integrity

The point to polygon topology structure used in BPARSS is based primarily on Smalltalk structures and *ConstructionPoints'* AssociatedAggregate attribute. These structures are described in Sections 4.2.6 and 4.2.9. In order to ensure the integrity of these topological relationships, methods have been built into BPARSS to assure that a *ConstructionPoint* is not deleted if it has an association with a polygonal *GeographicObject*. In addition, if a polygonal *GeographicObject* is deleted, all associations to *ConstructionPoints* are eliminated as part of the process. These integrity safeguards have been provided primarily to maintain the integrity of the test data set, but they also demonstrate an Object-Oriented approach to providing topology integrity .

6.5 Summary

The construction stage of the BPARSS development has been reviewed in this chapter. This review includes a brief overview of the Smalltalk Object-Oriented development environment, which has been used to implement BPARSS.

BPARSS interface and test data set considerations that are specific to the Smalltalk implementation have been described. This provides the background necessary for the BPARSS functional review provided in the next chapter.

Several approaches and algorithms that are specific to the DQR framework developed in this thesis have been explained in this chapter. These procedures have, for the most part, been adopted to take advantage of application development features available in the Smalltalk environment.

7 BPARSS FUNCTIONAL REVIEW

7.1 Introduction

This chapter reviews the BPARSS prototype with respect to its ability to support the DQR principles developed in the first three chapters of this thesis. Throughout the review, BPARSS query and design examples are used to demonstrate DQR functionality.

As discussed in Chapter 1, the DQR function must be capable of serving two key roles in a GIS; *data review for suitability* and *spatial analysis decision support*. Both these roles have been provided in the BPARSS prototype. The *review data for suitability* capacity is invoked by choosing the Report Status option, whereas *spatial analysis decision support* is initiated by the Review Building Permit Application selection. *Data review for suitability* considerations are reviewed as part of Section 7.2 (Data Quality Reporting Framework), and *spatial analysis decision support* is the subject of Section 7.4.

The effect of propagation of errors on data quality statements is the subject of Section 7.3. Two examples are provided to illustrate positional accuracy based DQR considerations.

An example of the effect of not knowing the quality of spatial data is provided in Section 7.5. This shows that a lack of knowledge could lead to erroneous decisions.

Section 7.6 discusses the Object-Oriented properties associated with BPARSS. It indicates how this development environment facilitates practical DQR.

7.2 Data Quality Reporting Framework

In this section the Data Quality Reporting framework developed in Chapter 3 is demonstrated using BPARSS examples.

7.2.1 DQR Classifications

All three of the major data quality classifications discussed in Section 3.2.3 have been incorporated into the development of the BPARSS prototype. These include Accuracy, Source and Currency. This capability is demonstrated through the Report Status request for a *ConstructionPoint* instance.

Figure 7.2.1A shows the BPARSS response to a Report Status request for a *ConstructionPoint* instance. The location of the selected *ConstructionPoint* is indicated on the *GraphPane* by a red filled circle whose radius equals the point's standard deviation (PP2 in the bottom right side of the *GraphPane*). The filled circle is used to both highlight the point for identification purposes and to illustrate the uncertainty of the location of the point. This concept of point uncertainty is used in the derivation of error corridors for polygon boundaries, discussed later in this section. The position uncertainty indication is to the same scale as the base drawing in Figure 7.2.1A (and all subsequent *GraphPane* representations). The *ReportPane* provides a textual description of the status of the *ConstructionPoint*. Only the beginning of this report is visible in Figure 7.2.1A. Figure 7.2.1B shows the entire textual report.

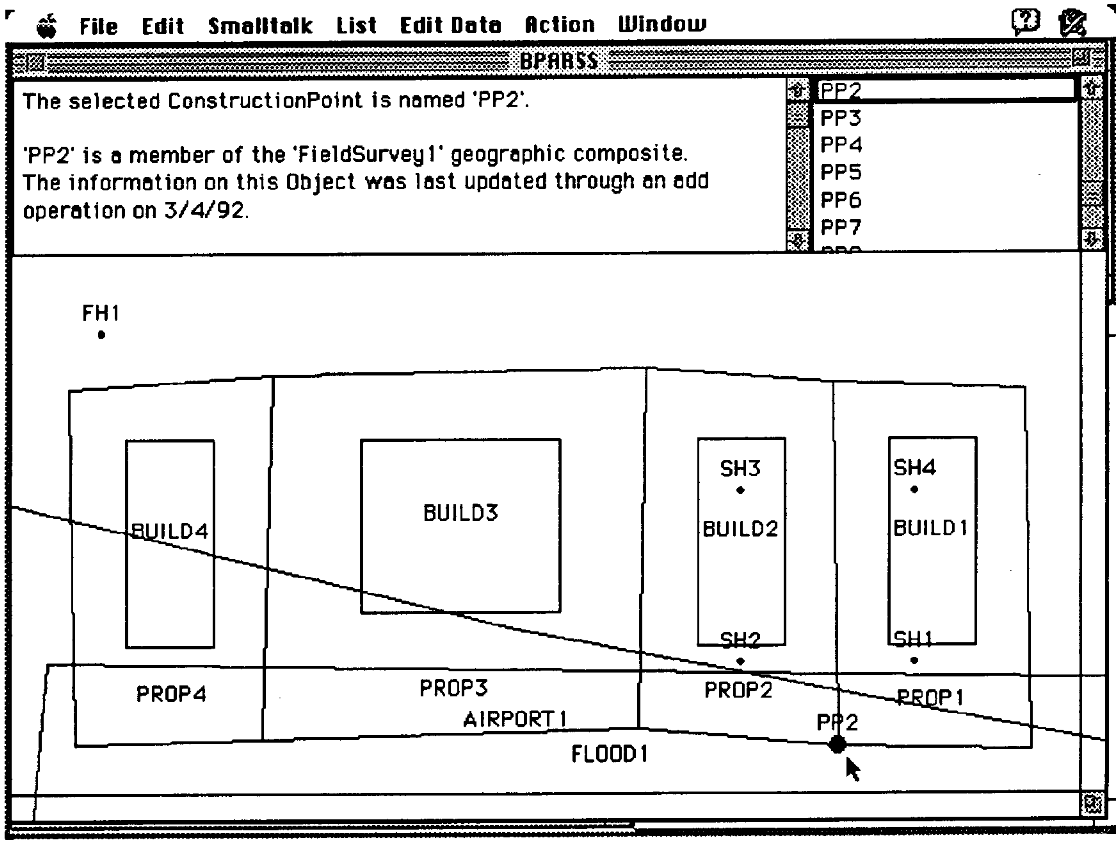

Figure 7.2.1A Data Quality Reporting Classification - Example 1

The *ConstructionPoint's* business and geometry attributes are shown in the first, fourth and fifth paragraphs (name, associated aggregates and position). The coordinates provided in Figure 7.2.1B (and all subsequent textual reports) are based on the Universal Transverse Mercator (UTM) map projection. Data quality aspects are also included in the report provided in Figure 7.2.1B. Positional accuracy information is available in the last paragraph where positional standard deviation and confidence level are indicated. Source information is contained in the third paragraph. This includes the name of the collection source and the compilation history. The collection date and the update date (in the second paragraph) provide currency information.

The selected ConstructionPoint is named 'PP2'.

'PP2' is a member of the 'FieldSurvey1' geographic composite.
The information on this Object was last updated through an add operation on 3/4/92.

'PP2' is a member of the 'FieldSurvey1' geographic composite.
This data was supplied by Cutting Surveys. Operations carried out as part of the data compilation include: total station survey - least squares adjustment - COGO data entry.
This data was collected on 3/4/92.

'PP2' is used in the construction of the following Polygons: (PROP1 PROP2).

Its Northing Coordinate = 5660004.2 metres and its Easting Coordinate = 710047.6 metres.

'PP2' is a member of the 'FieldSurvey1' geographic composite.
Its horizontal coordinate standard deviation is 0.5 metres.
All positional accuracy values have a 95.0% confidence level associated with them.

Figure 7.2.1B Data Quality Reporting Classification - Example 1 Text

A second Report Status example, shown in Figure 7.2.1C, further demonstrates how BPARSS supports DQR classification. The north boundary of the selected *AirportVicinityArea* (the *AirportVicinityArea's* area extends beyond the limits of the *GraphPane*) is denoted in response to the request. Again the location of the selected *AirportVicinityArea* is indicated on the *GraphPane*, this time by a highlighted error corridor indicating its positional accuracy. This indication of the uncertainty of the polygon's boundary is based on the standard deviation of the *ConstructionPoints* used in the polygon definition.

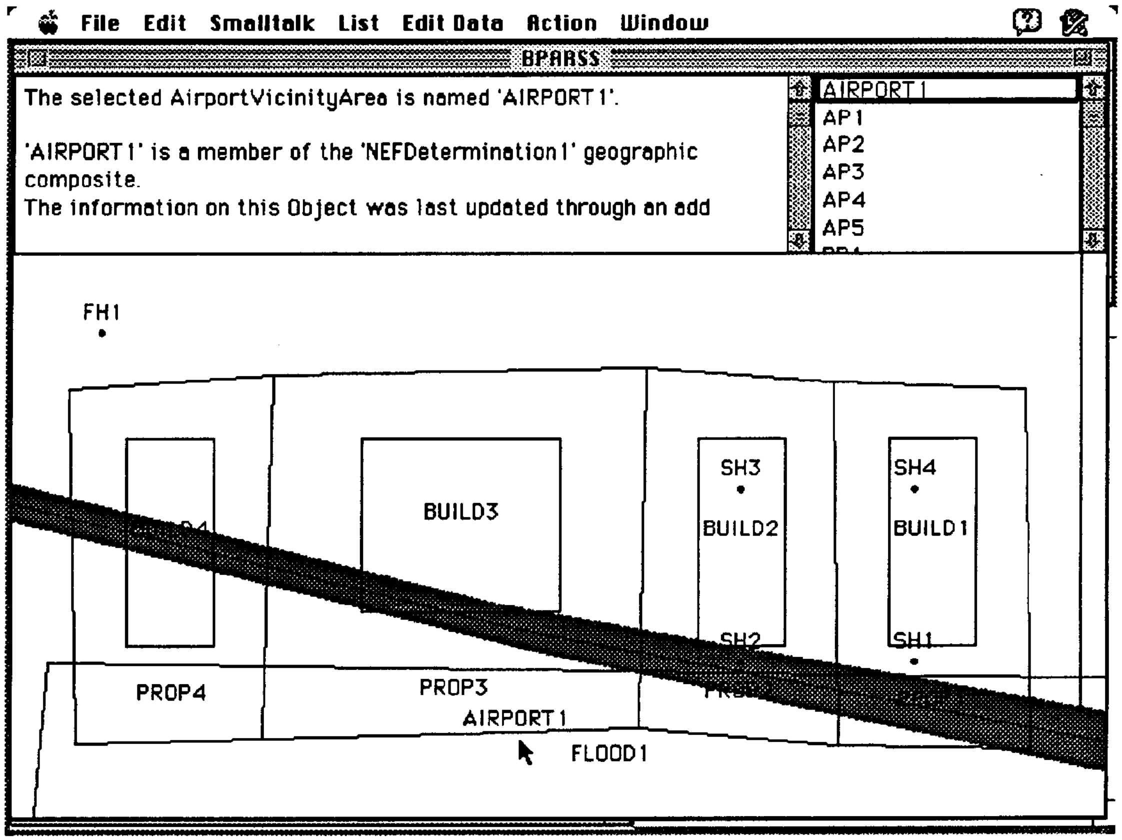

Figure 7.2.1C Data Quality Reporting Classification - Example 2

Figure 7.2.1D shows the complete textual report for the *AirportVicinityArea* example. In addition to the types of information discussed in the first example, an indication of attribute accuracy is provided for the ExposureLevel attribute. This is provided in terms of a confidence level statement for both the region core and the region edge of the polygon. This represents a simplified approach to the regionalization procedures introduced by Leung [1988].

The *ConstructionPoints* that define the *AirportVicinityArea* polygon are listed at the end of the report. The accuracy of these points determines the positional accuracy of the polygon.

This example illustrates that BPARSS allows data accuracy to be represented by positional accuracy statements, attribute accuracy indications, or a combination of the two. An approach to determining accuracy based on both spatial and attribute characteristics is desirable where the two are interdependent [Lam, 1992].

The selected AirportVicinityArea is named 'AIRPORT1'.

'AIRPORT1' is a member of the 'NEFDetermination1' geographic composite.
The information on this Object was last updated through an add operation on 8/16/93.

'AIRPORT1' is a member of the 'NEFDetermination1' geographic composite.
This data was supplied by Canadian Environment Department. Operations carried out as part of the data compilation include: decibel level sampling - noise exposure modelling - automated contouring.
This data was collected on 6/5/93.

The noise exposure level range for 'AIRPORT1' is '25-30'.

The confidence level associated with the 'exposureLevel' attribute at the regions core is 95.0 %.
The confidence level associated with the 'exposureLevel' attribute at the regions edge is 50.0 %.

Its location is defined by the following ConstructionPoints: ('AP1' 'AP2' 'AP3' 'AP4' 'AP5' 'AP1').
'AIRPORT1' is a polygonal feature. Its positional accuracy is determined by the positional accuracy of its Construction Points.

Figure 7.2.1D Data Quality Reporting Classification - Example 2 Text

7.2.2 Levels of DQR

Four of the levels of DQR, discussed in Section 3.3, have been demonstrated in the BPARSS prototype. The following examples are used to illustrate this. These examples incorporate the DQR level hierarchy algorithm discussed in Section 6.4.4.

Entity Type Level

The highest level in the DQR level hierarchy is the Entity Type. Within BPARSS, DQR has been established at this level by allowing DQR assignments at the class level. This means that all instances of a class acquire the data quality parameters assigned to the class, subject to the hierarchy considerations discussed in Section 6.4.4. This provides a means to assign default values. Figure 7.2.2A is the result of a Report Status request for the *ConstructionPoint* class.

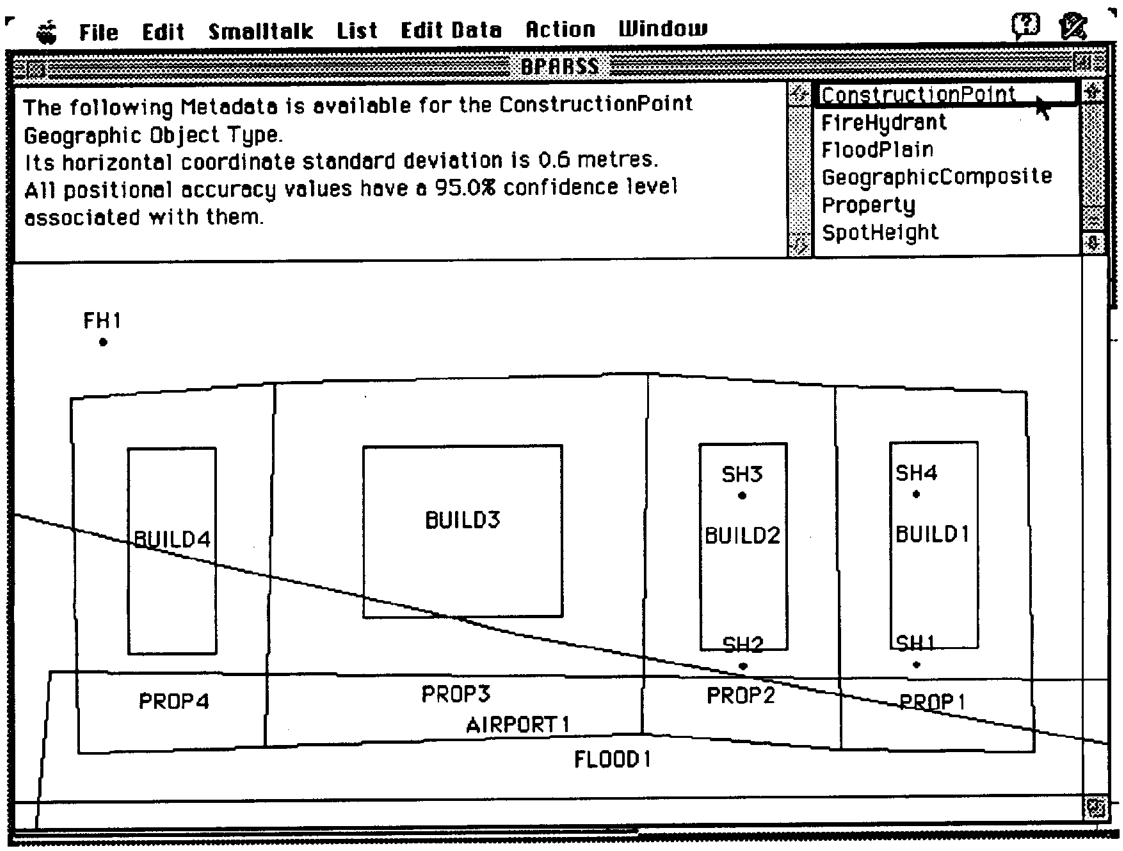

Figure 7.2.2A DQR Levels - Entity Type Level

The entire class level report is visible within the *ReportPane* in Figure 7.2.2A. The *GraphPane* display is not relevant to this report. The only data quality information available at this level for *ConstructionPoints* is horizontal positional accuracy.

An example of data quality information being acquired from the Entity Type level is provided in Section 7.3. In Figure 7.3B the positional accuracy information described in Figure 7.2.2A has been acquired by point 'BP11' as indicated following the "'BP11' is a member of the ConstructionPoint class." statement in Figure 7.3B.

Data Supplier Level

As discussed in Section 6.4.4, Data Supplier reports are implemented as *GeographicComposites* in BPARSS. Figure 7.2.2B shows the contents of the *ReportPane* after invoking a Report Status request for the 'FieldSurvey1' *GeographicComposite*. In addition to listing all the *GeographicObjects* that are considered part of this *GeographicComposite*, data quality information is described for all DQR classifications (accuracy, source and currency) assigned to 'FieldSurvey1'.

By referring back to Figure 7.2.1B it can be seen that 'PP2' has acquired the data quality information of 'FieldSurvey1'. In Figure 7.2.1B each section of the report that is preceded by the "'PP2' is a member of the 'FieldSurvey1' geographic composite." statement has acquired that section's information from the 'FieldSurvey1' *GeographicComposite*.

The selected GeographicComposite is named 'FieldSurvey1'.

The information on this Object was last updated through an add operation on 3/4/92.

This data was supplied by Cutting Surveys. Operations carried out as part of the data compilation include: total station survey - least squares adjustment - COGO data entry.
This data was collected on 3/4/92.

The GeographicObjects included as part of 'FieldSurvey1' are: (PP1 PP2 PP3 PP4 PP5 PP6 PP7 PP8 PP9 PP10 PROP1 PROP2 PROP3 PROP4).

'FieldSurvey1' is a GeographicComposite Object. Its positional accuracy information only applies to its Point Object members.

Its horizontal coordinate standard deviation is 0.5 metres.
All positional accuracy values have a 95.0% confidence level associated with them.

Figure 7.2.2B DQR Levels - Spatial Partition Level

Entity Occurrence

The case shown in Figure 7.2.2C provides an example of DQR at the Entity Occurrence level. In this example the point 'PP2' has been assigned its own individual positional accuracy information. It has been given a standard deviation of 1 metre. The magnitude of the standard deviation is indicated by the size of its highlighting circle. This can be compared to Figure 7.2.1A where its standard deviation was 0.5 metres (acquired from 'FieldSurvey1').

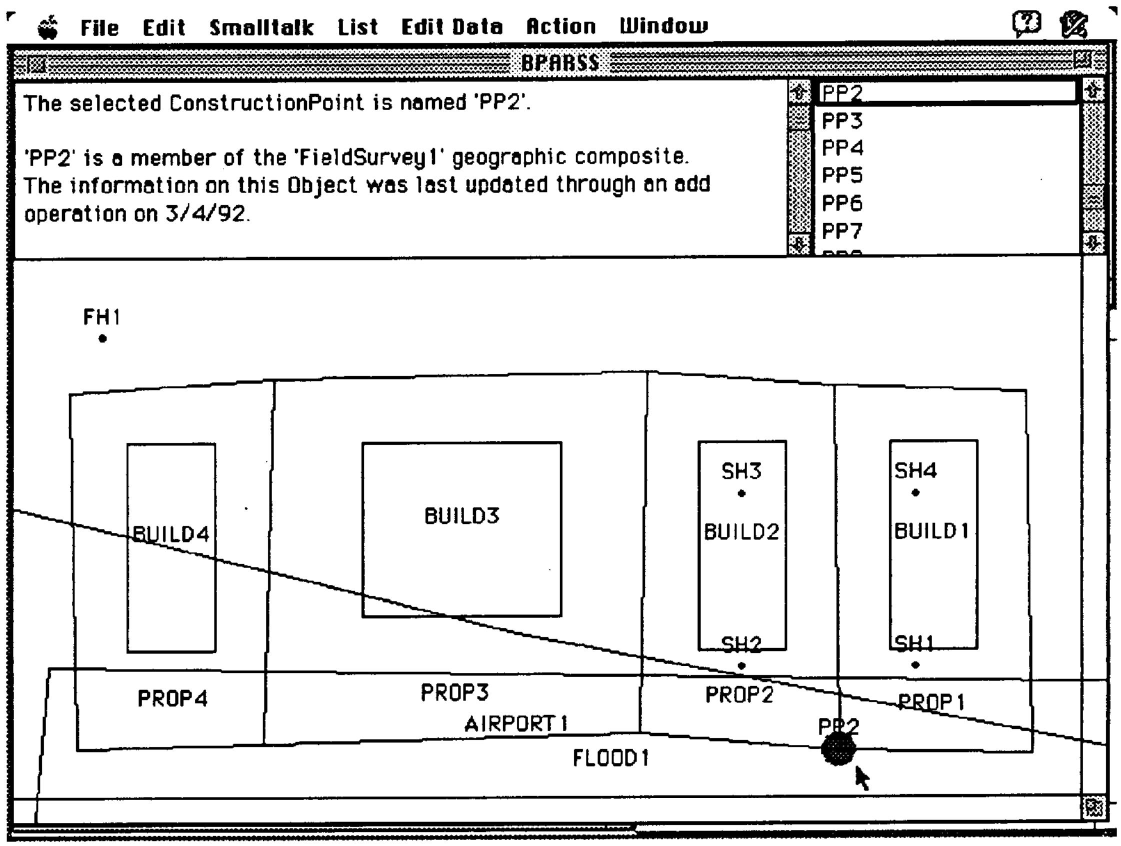

Figure 7.2.2C DQR Levels - Entity Occurrence Level

The *ReportPane* for this case, as shown in Figure 7.2.2D, is the same as that displayed in Figure 7.2.1B with the exception of the positional accuracy report. The 'FieldSurvey1' positional accuracy assignment has been superseded by a 'PP2' positional accuracy designation at the Entity Occurrence level. Note that the source and currency metadata of 'FieldSurvey1' still are relevant to 'PP2'.

The selected ConstructionPoint is named 'PP2'.
'PP2' is a member of the 'FieldSurvey1' geographic composite. The information on this Object was last updated through an add operation on 3/4/92.
'PP2' is a member of the 'FieldSurvey1' geographic composite. This data was supplied by Cutting Surveys. Operations carried out as part of the data compilation include: total station survey - least squares adjustment - COGO data entry. This data was collected on 3/4/92.
'PP2' is used in the construction of the following Polygons: (PROP1 PROP2).
Its Northing Coordinate = 5660004.2 metres and its Easting Coordinate = 710047.6 metres.
Its horizontal coordinate standard deviation is 1.0 metres. All positional accuracy values have a 95.0% confidence level associated with them.

Figure 7.2.2D DQR Levels - Entity Occurrence Level Text

Attribute Occurrence

An example of reporting at the attribute level is provided in the *AirportVicinityArea* example given above. The attribute level accuracy indicators shown in Figure 7.2.1D are discussed above (Section 7.2.1) as part of the attribute accuracy classification discussion.

7.3 Accuracy Propagation

The accuracy propagation characteristics provided in BPARSS are used to show how a Data Quality Reporting function can include DQR information that is derivable from other quality statement input. The two examples provided in this section are positional accuracy based.

Positional Accuracy

In the first example, a Report Status request for a *ConstructionPoint* that has been used to define a *Building* has resulted in the report shown in Figure 7.3A. In this figure, 'BP11' (a *ConstructionPoint* used to define the 'BUILD3' *Building*) has been identified and an indication of the standard deviation of its position has been given.

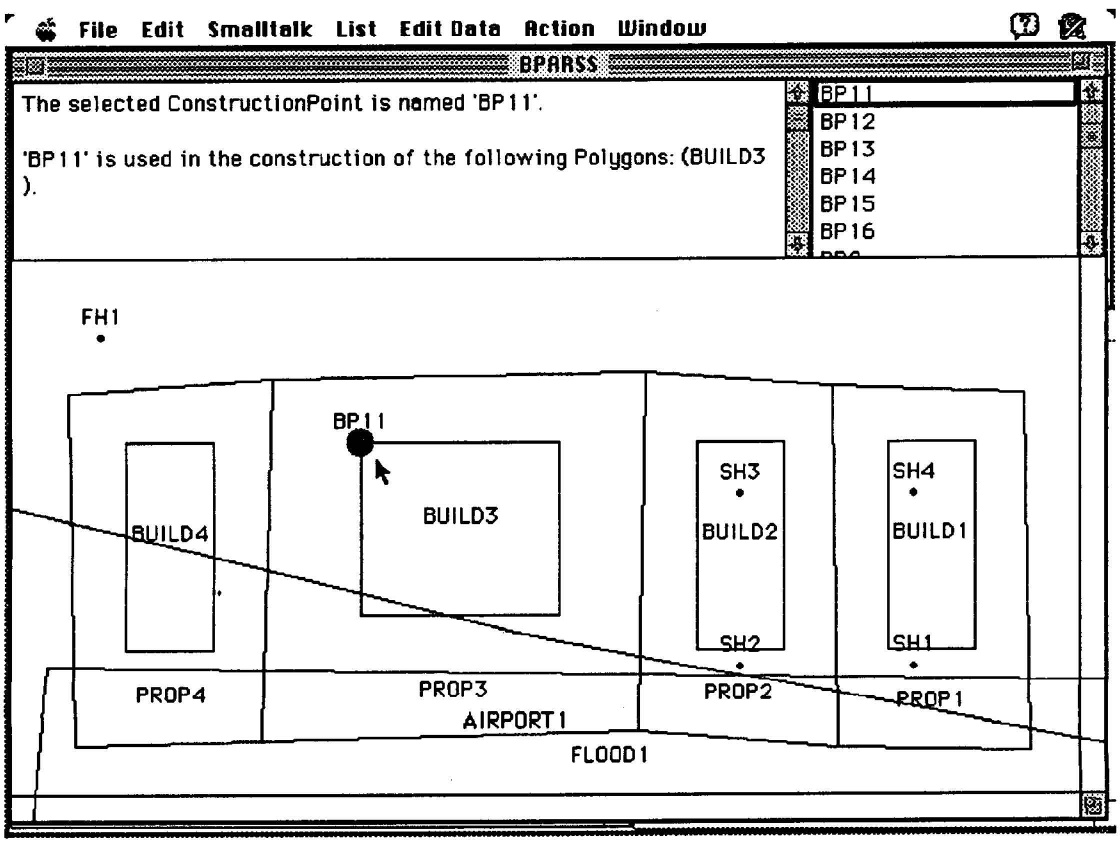

Figure 7.3A Positional Accuracy Propagation

The positional accuracy of 'BP11' has been determined based on its spatial relationship to another *GeographicObject* and their individual standard deviations. The *ReportPane* text, shown in Figure 7.3B, indicates that 'BP11' has its positional accuracy defined relative to the 'PP8' *ConstructionPoint.* This means that the standard deviation of 0.6 metres (acquired from the *ConstructionPoint* class, see Section 7.2.2) is relative to 'PP8', which has an absolute standard deviation of 0.5 metres (not shown here, 'PP8' is a member of the 'FieldSurvey1' *GeographicComposite*). By using a variance propagation approach, the individual standard deviations are combined to result in an absolute positional accuracy of 0.78 metres for 'BP11'. This value is reflected in the radius of the highlighted circle displayed in Figure 7.3A.

The selected ConstructionPoint is named 'BP11'.

'BP11' is used in the construction of the following Polygons: (BUILD3).

Its Northing Coordinate = 5660022.0 metres and its Easting Coordinate = 710020.0 metres.

The following relationships exist for 'BP11':
'BP11' is derived directly from 'PP8'. Its positional accuracy is relative to the positional accuracy of 'PP8'.

'BP11' is a member of the ConstructionPoint class.
Its horizontal coordinate standard deviation is 0.6 metres.
All positional accuracy values have a 95.0% confidence level associated with them.
The derived absolute standard deviation for 'BP11', based on its relative spatial relationship, is 0.78 metres.

Figure 7.3B Positional Accuracy Propagation - Text

Area Accuracy

The second example focuses on the 'area' attribute of polygonal *GeographicObjects.* Area is a derivable attribute whose accuracy is dependant on the positional accuracy of a *GeographicObject.* In Figure 7.3C the positional accuracy of the 'PROP2' *GeographicObject* is indicated by a corridor around its boundary. This corridor is based on the standard deviations of its *ConstructionPoints.*

Note that the south east corner of 'PROP2' is not as accurate as the other corners. This is because the south east corner ('PP2') has had its standard deviation increased to 1.0 metre while the other corners are members of 'FieldSurvey1' and have 0.5 metre standard deviations.

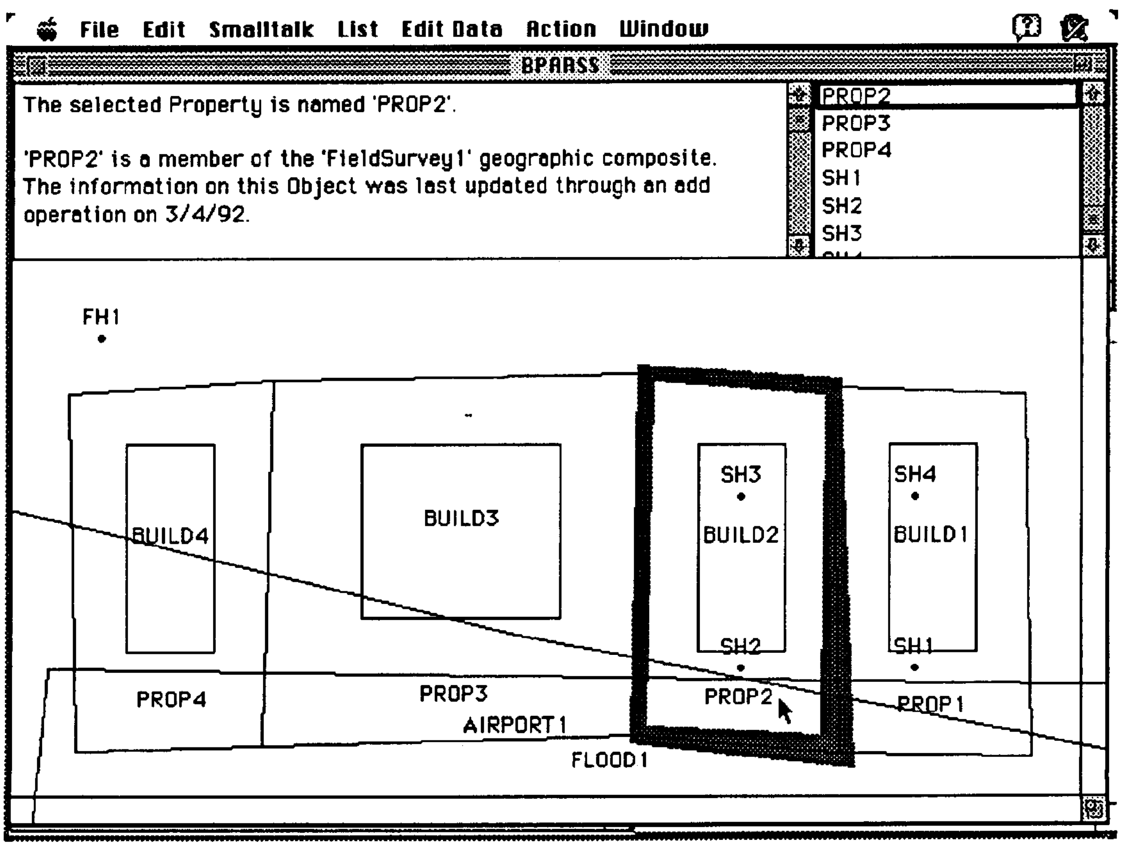

Figure 7.3C Area Accuracy Propagation

The area shown in Figure 7.3D has been calculated based on the location of 'PROP2's boundary. The area accuracy is given as percentage values based on the difference between the area derived from the 'PROP2' boundary and the area calculated at the error corridor limits.

Note that this example considers only the propagation of error within the spatial component of the object. This is generic to all polygonal *GeographicObjects* and can be addressed independent of business-related data, as discussed in Section 2.2.3.

> The selected Property is named 'PROP2'.
>
> 'PROP2' is a member of the 'FieldSurvey1' geographic composite.
> The information on this Object was last updated through an add operation on 3/4/92.
>
> 'PROP2' is a member of the 'FieldSurvey1' geographic composite.
> This data was supplied by Cutting Surveys. Operations carried out as part of the data compilation include: total station survey - least squares adjustment - COGO data entry.
> This data was collected on 3/4/92.
>
> The calculated area of 'PROP2' is 234.2 square metres.
> Based on the corridor defined by the inner polygon associated with 'PROP2', the actual area is within 16.4 % of this value.
> Based on the corridor defined by the outer polygon associated with 'PROP2', the actual area is within 17.8 % of this value.
>
> Its location is defined by the following ConstructionPoints: ('PP2' 'PP5' 'PP6' 'PP3' 'PP2').
> 'PROP2' is a polygonal feature. Its positional accuracy is determined by the positional accuracy of its Construction Points.

Figure 7.3D Area Accuracy Propagation - Text

7.4 Spatial Analysis Decision Support

In order to contribute to decision support, a DQR function should be integrated with spatial queries to provide recommendations for decision options. The following two examples of Review Permit Application requests are used to demonstrate this capability in BPARSS.

As described in Section 5.2.11, each *BuildingPermitApplication* has a LandUse attribute associated with it. When a Review Permit Application request is made, all relevant (based on LandUse) *BuildingRegulations* are reviewed to determine their influence on the application request acceptance or rejection.

Both available *BuildingRegulation* subclasses inherit the currency considerations associated with the *BuildingRegulation* DateSpread attribute (see Section 4.2.10). *SlopeRegulations* apply to the degree of terrain slope (determined from *SpotHeights)* within the proposed *Building* extent. *OverlapRegulations* are concerned with horizontal spatial coincidence between the proposed *Building* extent and other *GeographicObjects*.

The *GraphPane* shown in Figure 7.4A illustrates the spatial relationships between the *GeographicObjects* relevant to a review of the *BuildingPermitApplication* associated with the proposed 'BUILD4'. Three types of *GeographicObjects* (*SpotHeight, FloodPlain* and *AirportVicinityArea)* are of interest and correspond to three separate *BuildingRegulation* instances.

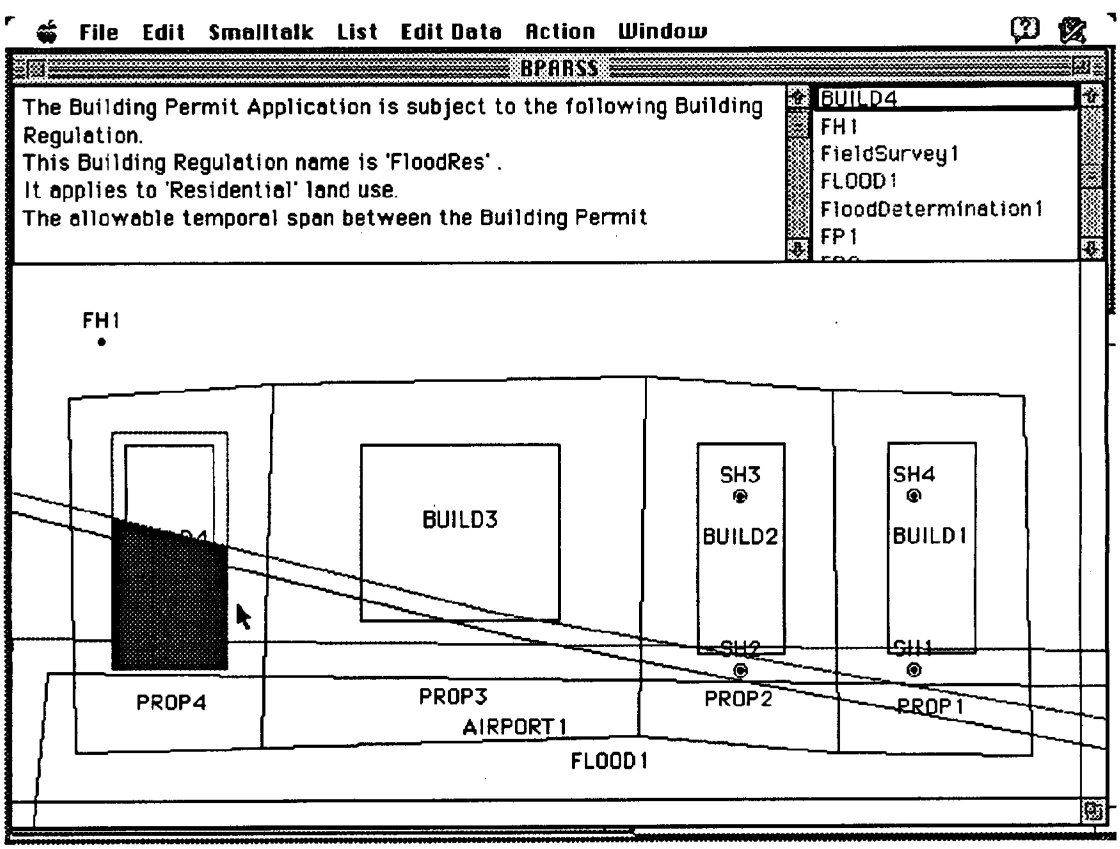

Figure 7.4A Decision Support - Example 1

The 'outer boundary' of the spatial extent of each of the relevant *GeographicObject* instances is utilized in determination of spatial overlap. In Figure 7.4B one of the *ConstructionPoints* associated with the 'FloodDetermination1' *GeographicComposite*, which is used to define the *FloodPlain* instance, is described. The 2.0 metre positional accuracy statement is used to determine the position of the 'outer boundary' of 'FloodDetermination1'. The 'outer boundaries' of the other relevant *GeographicObjects* are also based on their positional accuracy. Within the BPARSS prototype, all instances are tested without considering their spatial proximity to the *Building* of interest. The colour filled area indicates overlap between the largest expected extent of 'BUILD4' and the other 'outer boundaries'.

The selected ConstructionPoint is named 'FP1'.

'FP1' is a member of the 'FloodDetermination1' geographic composite.
The information on this Object was last updated through an add operation on 3/12/90.

'FP1' is a member of the 'FloodDetermination1' geographic composite.
This data was supplied by Alberta Environmental Services. Operations carried out as part of the data compilation include: elevation model creation - digital terrain analysis - flood plain extent determination.
This data was collected on 2/14/90.

'FP1' is used in the construction of the following Polygons: (FLOOD1).

Its Northing Coordinate = 5659998.0 metres and its Easting Coordinate = 710064.0 metres.

'FP1' is a member of the 'FloodDetermination1' geographic composite.
Its horizontal coordinate standard deviation is 2.0 metres.
All positional accuracy values have a 95.0% confidence level associated with them.

Figure 7.4B 'FP1' Status - 1

Figure 7.4C provides the contents of the *ReportPane* for the first example. Each paragraph in this report represents the review of an instance of a *BuildingRegulation* subclass. The first five statements in each paragraph are descriptions of the *BuildingRegulation* itself. The remaining statements are indications of the relationships between the *GeographicObject* instances in question, and recommendations based on these relationships. The recommendations, which are given in upper case bold text, are dependant on decision support attributes such as DateSpread and MaxAllowableSlope. In a production system this level of detail may not be necessary; it is provided within BPARSS for demonstration purposes only.

The first paragraph in Figure 7.4C describes the review of an *OverlapRegulation* that is associated with *FloodPlain* instances. This review indicates that there is a 50 month temporal span between the date of the permit application and the date the *FloodPlain* data was collected. This exceeds the allowable elapsed time. Based on this, a 'do not approve' recommendation is provided. In addition, there is spatial overlap between the two objects in question. This also results in a recommendation not to approve the application.

The second paragraph is similar in that it represents an *OverlapRegulation* associated with *AirportVicinityAreas*. In this case the data is considered current enough, but again there is overlap which results in a negative recommendation.

The *SlopeRegulation* review provided in the last paragraph indicates that slopes of up to 5% are permitted for residential buildings. The report states that all available *SpotHeights* are of acceptable currency. There is, however, no *SpotHeight* coverage for the area of interest. This lack of coverage is evident in Figure 7.4A.

The Building Permit Application is subject to the following Building Regulation.
This Building Regulation name is 'FloodRes' .
It applies to 'Residential' land use.
The allowable temporal span between the Building Permit application and the date data was collected on related Geographic Object is 12 months.
The related Geographic Object Type associated with this Building Regulation is 'FloodPlain' .
The Building Permit Application date is 4/1/94; the data collection date for the 'FLOOD1' Geographic Object is 2/14/90.
50 months have elapsed between these two dates.
BASED ON THIS TEMPORAL SPAN, THIS APPLICATION SHOULD NOT BE APPROVED UNTIL MORE CURRENT 'FLOOD1' INFORMATION IS AVAILABLE.
The available information indicates that there is spatial overlap between the proposed building and 'FLOOD1'.
BASED ON THIS SPATIAL RELATIONSHIP, THIS APPLICATION SHOULD NOT BE APPROVED.

The Building Permit Application is subject to the following Building Regulation.
This Building Regulation name is 'AirRes' .
It applies to 'Residential' land use.
The allowable temporal span between the Building Permit application and the date data was collected on related Geographic Object is 12 months.
The related Geographic Object Type associated with this Building Regulation is 'AirportVicinityArea' .
The Building Permit Application date is 4/1/94; the data collection date for the 'AIRPORT1' Geographic Object is 6/5/93.
10 months have elapsed between these two dates.
The available information indicates that there is spatial overlap between the proposed building and 'AIRPORT1'.
BASED ON THIS SPATIAL RELATIONSHIP, THIS APPLICATION SHOULD NOT BE APPROVED.

The Building Permit Application is subject to the following Building Regulation.
This Building Regulation name is 'SlopeRes' .
It applies to 'Residential' land use.
The allowable temporal span between the Building Permit application and the date data was collected on related Geographic Object is 24 months.
A building of this type can be built on a slope not exceeding 5 %.
The Building Permit Application date is 4/1/94; the data collection date for the 'SH2' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The Building Permit Application date is 4/1/94; the data collection date for the 'SH3' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The Building Permit Application date is 4/1/94; the data collection date for the 'SH1' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The Building Permit Application date is 4/1/94; the data collection date for the 'SH4' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.

Figure 7.4C Decision Support - Example 1 - Text

The second example demonstrates the review of the *BuildingPermitApplication* associated with the proposed 'BUILD1'. 'BUILD1' is again planned for residential LandUse, so the same *BuildingRegulations* apply. The *GraphPane* shown in Figure 7.4D indicates the pertinent spatial relationships.

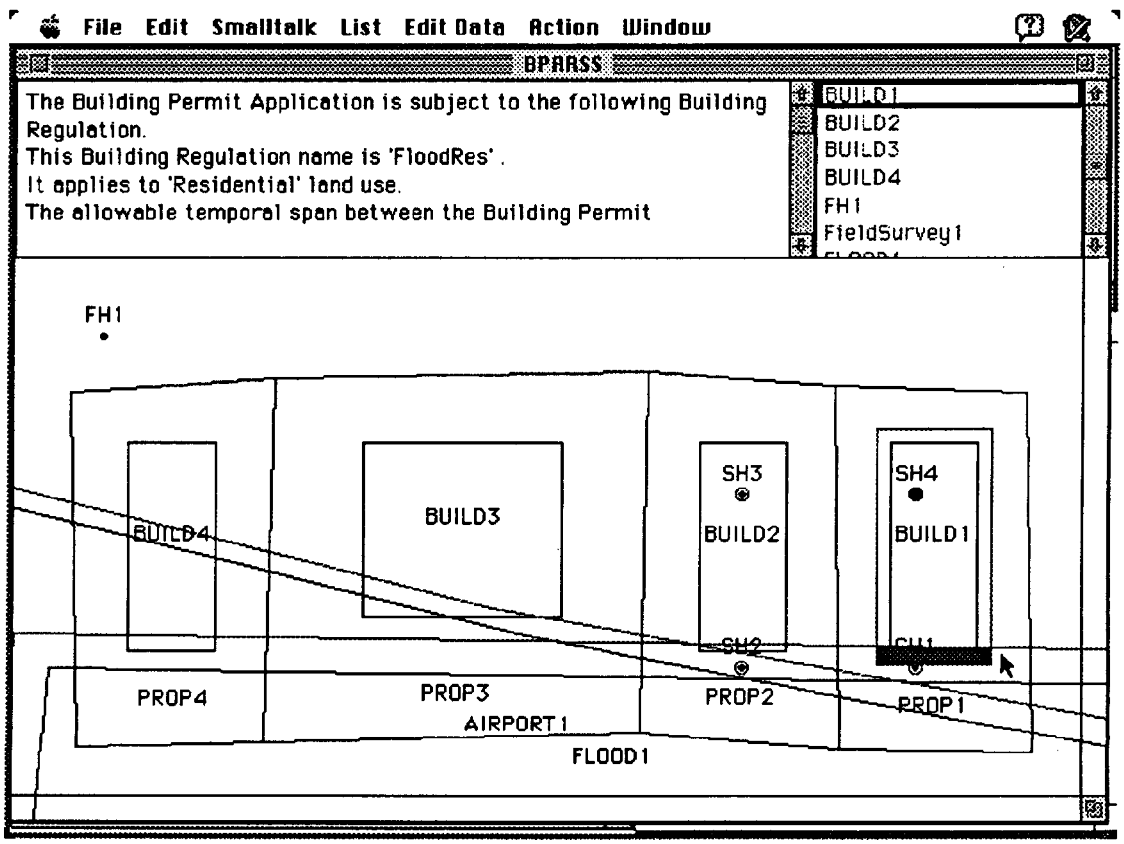

Figure 7.4D Decision Support - Example 2

In the *ReportPane* shown in Figure 7.4E, recommendations are made based on the currency of data and the overlap associated with the 'FLOOD1' instance. In this example there is no overlap between 'AIRPORT1' and the proposed 'BUILD1', and therefore no recommendations are provided. 'SH1' and 'SH4' are located within the proposed extent of 'BUILD1' but result in a slope determination within allowable slope tolerances.

The Building Permit Application is subject to the following Building Regulation.
This Building Regulation name is 'FloodRes' .
It applies to 'Residential' land use.
The allowable temporal span between the Building Permit application and the date data was collected on related Geographic Object is 12 months.
The related Geographic Object Type associated with this Building Regulation is 'FloodPlain' .
The Building Permit Application date is 4/4/94; the data collection date for the 'FLOOD1' Geographic Object is 2/14/90.
50 months have elapsed between these two dates.
BASED ON THIS TEMPORAL SPAN, THIS APPLICATION SHOULD NOT BE APPROVED UNTIL MORE CURRENT 'FLOOD1' INFORMATION IS AVAILABLE.
The available information indicates that there is spatial overlap between the proposed building and 'FLOOD1'.
BASED ON THIS SPATIAL RELATIONSHIP, THIS APPLICATION SHOULD NOT BE APPROVED.

The Building Permit Application is subject to the following Building Regulation.
This Building Regulation name is 'AirRes' .
It applies to 'Residential' land use.
The allowable temporal span between the Building Permit application and the date data was collected on related Geographic Object is 12 months.
The related Geographic Object Type associated with this Building Regulation is 'AirportVicinityArea' .
The Building Permit Application date is 4/4/94; the data collection date for the 'AIRPORT1' Geographic Object is 6/5/93.
10 months have elapsed between these two dates.
The available information indicates that there is no spatial overlap between the proposed building and 'AIRPORT1'.

The Building Permit Application is subject to the following Building Regulation.
This Building Regulation name is 'SlopeRes' .
It applies to 'Residential' land use.
The allowable temporal span between the Building Permit application and the date data was collected on related Geographic Object is 24 months.
A building of this type can be built on a slope not exceeding 5 %.
The Building Permit Application date is 4/4/94; the data collection date for the 'SH2' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The Building Permit Application date is 4/4/94; the data collection date for the 'SH3' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The Building Permit Application date is 4/4/94; the data collection date for the 'SH1' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The Building Permit Application date is 4/4/94; the data collection date for the 'SH4' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The elevation difference between 'SH1' and 'SH4' is -0.2 metres. This represents a -2.0 % slope.

Figure 7.4E Decision Support - Example 2 - Text

7.5 The Importance Of Data Quality Information

In order to demonstrate the effect that a lack of knowledge of the quality of data can have on spatial analyses, the second decision support example (Section 7.4) is repeated with the positional accuracy of the *FloodPlain* and *AirportVicinityArea* set to a standard deviation of zero. The resulting response to the ReviewPermitApplication request is what would occur if the positional accuracy was known perfectly or, as in this case, not known at all.

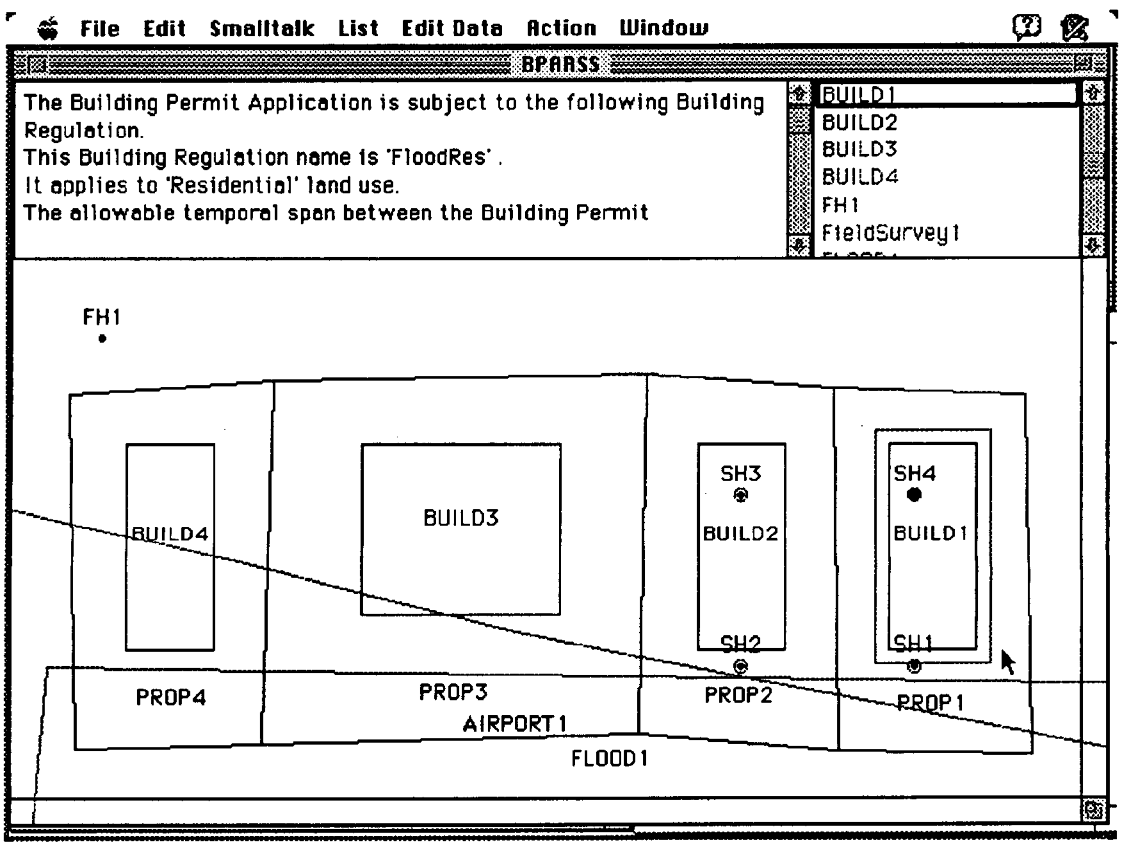

Figure 7.5A Data Quality Importance

The graphic response to the request, shown in Figure 7.5A, illustrates (by the lack of an outer boundary) that there is either perfect positional accuracy associated with the two changed objects (an unlikely situation) or, no positional quality information available. The Report Status request shown in Figure 7.5B, is for 'FP1', which is one of the *ConstructionPoints* used to determine the position of the 'FloodDetermination1' *GeographicComposite*. It shows that the confidence level (Percentile) associated with the *PositionalAccuracy* is 0% (compare to Figure 7.4B), indicating that there is no knowledge of the level of *PostionalAccuracy*. Figure 7.5A also shows that there is now no spatial overlap between these objects and the proposed *Building*.

The selected ConstructionPoint is named 'FP1'.

'FP1' is a member of the 'FloodDetermination1' geographic composite.
The information on this Object was last updated through an add operation on 3/12/90.

'FP1' is a member of the 'FloodDetermination1' geographic composite.
This data was supplied by Alberta Environmental Services. Operations carried out as part of the data compilation include: elevation model creation - digital terrain analysis - flood plain extent determination.
This data was collected on 2/14/90.

'FP1' is used in the construction of the following Polygons: (FLOOD1).

Its Northing Coordinate = 5659998.0 metres and its Easting Coordinate = 710064.0 metres.

Its horizontal coordinate standard deviation is 0.0 metres.
All positional accuracy values have a 0.0% confidence level associated with them.

Figure 7.5B 'FP1' Status - 2

The textual report shown in Figure 7.5C also indicates that there is no spatial overlap with the *FloodPlain* or *AirportVicinityArea*. In addition, it indicates that there is no positional accuracy information available for these *GeographicObjects* and provides appropriate warnings.

This example illustrates the risk of making important decisions, based on spatial analysis, without a knowledge of data quality.

The Building Permit Application is subject to the following Building Regulation.
This Building Regulation name is 'FloodRes' .
It applies to 'Residential' land use.
The allowable temporal span between the Building Permit application and the date data was collected on related Geographic Object is 12 months.
The related Geographic Object Type associated with this Building Regulation is 'FloodPlain' .
The Building Permit Application date is 4/4/94; the data collection date for the 'FLOOD1' Geographic Object is 2/14/90.
50 months have elapsed between these two dates.
BASED ON THIS TEMPORAL SPAN, THIS APPLICATION SHOULD NOT BE APPROVED UNTIL MORE CURRENT 'FLOOD1' INFORMATION IS AVAILABLE.
The available information indicates that there is no spatial overlap between the proposed building and 'FLOOD1'.
WARNING - THERE IS NO POSITIONAL ACCURACY INFORMATION AVAILABLE FOR THIS GEOGRAPHIC OBJECT.

The Building Permit Application is subject to the following Building Regulation.
This Building Regulation name is 'AirRes' .
It applies to 'Residential' land use.
The allowable temporal span between the Building Permit application and the date data was collected on related Geographic Object is 12 months.
The related Geographic Object Type associated with this Building Regulation is 'AirportVicinityArea' .
The Building Permit Application date is 4/4/94; the data collection date for the 'AIRPORT1' Geographic Object is 6/5/93.
10 months have elapsed between these two dates.
The available information indicates that there is no spatial overlap between the proposed building and 'AIRPORT1'.
WARNING - THERE IS NO POSITIONAL ACCURACY INFORMATION AVAILABLE FOR THIS GEOGRAPHIC OBJECT.

The Building Permit Application is subject to the following Building Regulation.
This Building Regulation name is 'SlopeRes' .
It applies to 'Residential' land use.
The allowable temporal span between the Building Permit application and the date data was collected on related Geographic Object is 24 months.
A building of this type can be built on a slope not exceeding 5 %.
The Building Permit Application date is 4/4/94; the data collection date for the 'SH2' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The Building Permit Application date is 4/4/94; the data collection date for the 'SH3' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The Building Permit Application date is 4/4/94; the data collection date for the 'SH1' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The Building Permit Application date is 4/4/94; the data collection date for the 'SH4' Geographic Object is 3/17/93.
13 months have elapsed between these two dates.
The elevation difference between 'SH1' and 'SH4' is -0.2 metres. This represents a -2.0 % slope.

Figure 7.5C Data Quality Importance - Text

7.6 Object Oriented Environment

The BPARSS prototype has been developed (through all stages of system development) within an Object-Oriented environment. In this section the OO structures and properties that are relevant to BPARSS are reviewed.

7.6.1 Classification

One of the main abstraction mechanisms used within the Object-Oriented paradigm is Classification [Egenhofer and Frank, 1992], which is the grouping of similar objects into classes. This allows all instances of a particular class to share data structures and behavioral patterns. The instances are differentiated by their individual properties or attribute values.

Within Smalltalk, the classes themselves can be assigned variables and methods. The values associated with these class variables [Smalltalk, 1993] apply to the class and consequently every instance of the class. Within BPARSS, class instance variables have been used to enable the Entity-Type reporting discussed in Section 7.2.2. Class instance variables names are inherited by subclasses, but the values of these variables are not. This property has proven most useful in allowing each subclass of *GeographicObject* to be assigned its own Entity-Type level metadata.

7.6.2 Gen-Spec Structure

The Gen-Spec structure has been used throughout the BPARSS application. It is within these structures that the reusability of the OO environment prevails. Useful properties inherent to Gen-Spec structures that have been utilized in BPARSS include class abstraction and inheritance.

Abstract classes are classes that do not have instances. They are created to model characteristics that are common to a number of other classes. These other classes become subclasses of the abstract class and share attributes and methods that are defined within the abstract class. Within BPARSS there are several abstract classes including *GeographicObject, GeometricObject, BuildingRegulation* and others (see Chapter 4).

Inheritance is one of the most powerful principles of Object-Orientation. It allows classes within Gen-Spec structures to share attributes and methods (including abstract classes). This facilitates rapid and rigorous expansion of OO models and applications. For example, the addition of *GeographicObject* subclasses was relatively straightforward since most of the functionality and attribution was already available within the *GeographicObject* superclass, including the Geometry and Metadata considerations.

7.6.3 Whole-Part Structure

Two types of Whole-Part structures have been employed within BPARSS: Assembly-Parts and Collection-Members.

Assembly-Parts structures have been discussed in detail in Section 4.1.3 and have been implemented throughout BPARSS. They provide the means to easily expand the depth of data structures. The example given in Section 5.2.9 illustrates this; the BPARSS coordinate accuracy information is associated directly with the *PositionalAccuracy* class, whereas the SAIF model assigns coordinate accuracy data to the *CoordinateAccuracy* class (lower in the Assembly-Parts structure). The modified BPARSS structure has been adopted to simplify the prototype development. If the pure SAIF structure was required in the future, only a minor update would be necessary.

The best example of a Collection-Members structure in BPARSS is the relationship between a *GeographicComposite* and its Geocomponents. This relationship has been used to implement Data Supplier level reporting as described in Section 7.2.2.

7.6.4 Polymorphism

Another key abstraction within the Object-Oriented framework is polymorphism. Simply stated, polymorphism allows different objects to respond to the same message in different but appropriate ways. In other words, for a message to be polymorphic it must be capable of being sent to different types of objects, invoking different methods within those object types to accomplish the same operational purpose [Martin and Odell, 1992]. This means that the sender of the message does not need to know what type of object it is communicating with. If the receiving object has been implemented to understand the message, it responds in an appropriate manner.

Many of the polymorphic strengths inherent to Smalltalk have been utilized within BPARSS. An example of this is Smalltalk's ability to request graphical displays of different *GeometricObjects* (*Points*, *Polygons*, *Rectangles*, etc) using the same message string. In addition, the classes created for BPARSS have utilized polymorphism, particularly in the report-based methods. Many examples of this exist within the *GeographicObject* and *BuildingRegulation* subclass structures.

7.6.5 Encapsulation

By combining their structural and behavioral characteristics, objects are able to protect their data through encapsulation. Encapsulation (often referred to as *information hiding*) prevents unwanted access to the data by only allowing access through the object's own methods.

In addition to protecting the integrity of the data, encapsulation facilitates the upgrading of processing algorithms without having to be concerned about the effect of a change on other objects or applications. For example, the procedure used to calculate the area accuracy (discussed in Section 7.3) could be changed within the *Polygon* class without making any changes to the objects that access the *Polygons*. A user of the object does not need to know what internal processes are triggered when they make a request of an object, only that it will provide a suitable response.

7.7 Summary

This chapter has briefly reviewed the BPARSS prototype functionality with respect to its ability to support the Data Quality Reporting framework introduced in Chapter 3. Several examples have been discussed to demonstrate that this framework facilitates practical DQR.

The Object-Oriented techniques used in the development of BPARSS have also been discussed. Brief examples of the OO structures and properties considered most effective have been described. These help to provide an environment that is easily extended to satisfy emerging DQR requirements.

8 CONCLUSIONS AND RECOMMENDATIONS

8.1 Conclusions

This section presents a series of high level conclusions based on the observations made throughout this thesis. These conclusions are used as the basis for the recommendations presented in the next section. The conclusions are grouped under DQR Concepts, BPARSS Prototype and Object-Oriented Approach headings.

DQR Concepts

The BPARSS prototype developed in this thesis has demonstrated how a DQR function is capable of supporting the *data review for suitability* and *spatial analysis decision support* roles in a GIS. These roles are considered key to the concept of Data Quality Reporting.

A set of characteristics (comprehensive, flexible and meaningful) that should exist in a practical DQR function has been proposed in this thesis. The DQR framework presented, and demonstrated in BPARSS, incorporates these characteristics.

Although the framework supports meaningful DQR, an application's interface plays a major role in communicating results to users. The BPARSS prototype demonstrates how an interface can contribute by combining textual and graphical reports. A more detailed examination of this issue is considered outside the scope of this thesis.

The framework is based on three broad data quality classifications (accuracy, source and currency) and a multi-level reporting hierarchy. An implementation utilizing this framework can begin by initially providing basic DQR functionality, and be incrementally expanded to support more rigorous DQR.

BPARSS Prototype

The purpose of BPARSS is to test and demonstrate the DQR framework and characteristics presented in this thesis. The results reviewed in the previous chapter have shown how this goal has been realized.

BPARSS has been developed as a prototype with the intent that it be used only within the context of this thesis and not be extended for other purposes. For this reason, it has been possible to utilize simplified data management structures and processing algorithms, and to develop it outside commercial GIS products. This has resulted in an application that is a true prototype, in that it is not sophisticated but is capable of demonstrating DQR concepts.

The example query used in BPARSS is based on a typical municipal GIS application, but represents a simplified approach to the problem. The author has not carried out any business analysis and has presented a solution that is based on assumptions that may not be valid. Again, this approach is seen as appropriate for demonstration purposes within a prototype system.

A form of the Coad/Yourdon systems analysis and design methodology has been used in the development of BPARSS. Although there are more rigorous Object-Oriented development methodologies available, the Coad/Yourdon approach provides the structure and methodology required to understand a systems development problem. The analysis and design documentation is easily understood and provides the basis for the application construction process.

The SAIF geomatics data exchange standard has been used as a basis for the development of the BPARSS data structures. This has been done to align the DQR framework with an emerging industry standard. The SAIF model has proven to be both comprehensive and expandable. For the most part, BPARSS has implemented the SAIF model rigorously, but on a few occasions the structures have been simplified. As discussed in Section 5.2.9, this demonstrates the flexibility associated with using the SAIF standard.

A CASE tool has been used to document the analysis and design stages of the BPARSS development. The CASE tool facilitates the definition of the data structures and object communications. Although all details of the development have not been documented, the key concepts are available. The use of the CASE tool has been essential to understanding and ensuring the integrity of the object structures and communications during BPARSS development. The CASE documentation also provides a forum to explain the BPARSS application and DQR concepts.

Object-Oriented Approach

An Object-Oriented environment has been used to develop the BPARSS prototype. This environment was selected because it is expected that future GISs will utilize OO technology. In addition, Object-Oriented concepts were seen to be well suited to the development of a DQR framework.

Although the DQR principles developed in this thesis could be implemented in most Data Base Management Systems (DBMS) and application development environments, there are advantages to an Object-Oriented DQR approach. These include simplified representations of object relationships, the ability to accommodate multi-valued attributes [Wiegand and Adams, 1994], and the benefits offered by the OO properties discussed in Section 7.6. The Object-Oriented environment supports the DQR framework by allowing data quality parameters to be assigned directly to classes and *GeographicComposites* .

The main disadvantage of the OO approach to DQR is the lack of existing commercial Object-Oriented GIS products. This problem is expected to be reduced as Object-Oriented GISs are developed. Currently, the SmallWorld GIS product offers a commercial OO product [Newell, 1992] and other leading vendors, such as Intergraph, are developing OO based GISs [Albaredes, 1994].

8.2 Recommendations

The following recommendations are offered as suggestions as to where future DQR-based research should be carried out.

The BPARSS prototype has been developed outside of commercial GIS products. Future work should include investigation into implementing the DQR framework within production GIS tools. This should include both Object-Oriented and Relational DBMS environments.

The work presented in this thesis has provided limited attention to the requirement that data quality reports be meaningful to users. This is seen as something that is primarily addressed through user interface design and should be reviewed as a separate research project. A commercial GIS which includes sophisticated interfacing tools should be utilized in this review.

The data structures that have been captured within the CASE model as part of this thesis, represent a generic approach to GIS data modelling that is based on an emerging industry standard. The model within this CASE tool should be refined by referencing it to all GIS related development (at The University of Calgary), and modifying it as necessary. This will promote a broader understanding of GIS concepts.

REFERENCES

Albaredes, G., (1994), "GIS and Beyond", Computers, Environment and Urban Systems, March/April, Vol. 18, No. 2, pp. 147-151.

Bedard, Y., (1987), "Uncertainties in Land Information Systems Databases", Proceedings of Auto-Carto 8, Baltimore, March/April, pp. 175-184.

Blakemore, M., (1983), "Generalisation and Error in Spatial Data Bases, Proceedings of Auto-Carto 6, Ottawa, October, pp. 313-322.

Boddie, J., (1993), The Information Asset, Prentice-Hall, New Jersey.

Bomford, G., (1980), Geodesy, Oxford, New York.

Burrough, P., (1986), Principles of Geographic Information Systems for Land Resources Assessment, Clarendon Press, Oxford.

CGIS (Canadian Geomatics Interchange Standard), (1992), Canadian Geomatics Interchange Standard: Formal Definition, Canadian General Standards Board, Draft 0.2.

Chrisman, N.R., (1991), "The Error Component in Spatial Data", Geographical Information Systems Principles and Applications (Maguire, Goodchild and Rhind, editors), John Wiley & Sons, Inc., New York, pp. 165-174.

Coad, P. and Nicola, J., (1993), Object-Oriented Programming, Prentice-Hall, Inc., New Jersey.

Coad, P. and Yourdon, E., (1991a), Object-Oriented Analysis, Prentice-Hall, Inc., New Jersey.

Coad, P. and Yourdon, E., (1991b), Object-Oriented Design, Prentice-Hall, Inc., New Jersey.

Egenhofer, M.J. and Frank, A.U., (1992), "Object-Oriented Modeling for GIS", Journal of the Urban and Regional Information Systems Association, Fall, Vol. 4, No. 2, pp. 3-19.

Garza, R.J. and Foresman, T.W., (1991), "Embedding Quality Into County-Wide Data Conversion", GIS/LIS '91 Proceedings, Atlanta, pp. 130-136.

GDF, (1992), "Geographic Data Files", Geographic Data Files Release 2.0, Task Force European Digital Road Map (Project Ref: V1021), January.

Goodchild, M.F., (1991), "Keynote Address", Proceedings of the Symposium on Spatial Database Accuracy, Melbourne, Australia, June, pp. 1-16.

Hunter, G.J., (1991), "Processing Error in Spatial Databases: The Unknown Quantity", Proceedings from the Symposium on Spatial Database Accuracy, Melbourne, pp. 203-214.

Hunter, G.J., (1992), "Techniques for Assessing the Effect of Processing Errors in Spatial Databases", Proceedings from the URISA 1992 Conference, Washington, D.C., Vol. 1, pp. 24-33.

Hunter, G.J. and Beard, K., (1992), "Understanding Error in Spatial Databases", The Australian Surveyor, Vol. 37, No. 2, pp. 108-119.

Kemp, Z., and Thearle, R., (1992), "Modelling Relationships in Spatial Databases", Proceedings from the 5th International Symposium on Spatial Data Handling, Charleston, South Carolina, Vol. 1, pp. 313-322.

LaLonde, W.R., and Pugh J.R., (1990), Inside Smalltalk, Volume 1, Prentice-Hall Inc., New Jersey.

Lam, S.S.M., (1992), "Uncertainty Management With Fuzzy Sets In A Commercial Vector GIS", M.Sc,Eng. Thesis, The University of Calgary Geomatics Engineering Publication, Calgary.

Lanter, D.P., (1991), "Design of a Lineage-Based Meta-Data Base for GIS", Cartography and Geographic Information Systems, Vol. 18, No. 4, pp. 255-261.

Leung, Y., (1988), Spatial Analysis and Planning Under Imprecision, Elsevier Science Publishers B.V., North-Holland.

LRIS Project, (1991), "Alberta LRIS Network Spatial Database System, Business Area Analysis Stage Report", Alberta Forestry, Lands and Wildlife, Edmonton.

MacAnalyst, (1993), MacAnalyst Software User's Manual, Version 4.0, Excel Software, Marshalltown, Iowa.

MacDesigner, (1993), MacDesigner Software User's Manual, Version 4.0, Excel Software, Marshalltown, Iowa.

Martin, J., (1990), Information Engineering, Book III: Design & Construction, Prentice-Hall Inc., New Jersey.

Martin, J. and Odell, J.J., (1992), Object-Oriented Analysis & Design, Prentice-Hall Inc., New Jersey.

Miller, R., (1991), "An Investigation into Modelling and Managing Uncertainty in Geographic Information Systems", M.Sc. Thesis, The University of Calgary, Surveying Engineering Publication, Calgary, Canada.

NCDCDS (National Committee for Digital Cartographic Data Standards, (1988), The Proposed Standard for Digital Cartographic Data, The American Cartographer, Vol. 15, No. 1, pp. 9-140.

Newell, R., G., (1992), "Practical Experiences of Using Object-Orientation to Implement a GIS", GIS/LIS '92 Proceedings, Vol. 2, pp. 624-629.

Openshaw, S., (1989), "Learning to live with errors in spatial databases", The Accuracy of Spatial Databases, (Goodchild and Gopal, editors), Taylor and Francis, London, pp. 263-276.

Ryan, T. and Masters, E., (1991), "Spatial Accuracy as a Quality Indicator", Proceedings of the Symposium on Spatial Database Accuracy, pp. 175-185, Melbourne, Australia, June.

Smalltalk, (1993), Smalltalk V Programming Reference, Digitalk Incorporated, Los Angeles, California, July.

SAIF (Spatial Archive and Interchange Format), (1991), "SAIF Interchange Specification", Draft National Standard of Canada, Volume 2, Draft 1.0.

SDTS (Spatial Data Transfer Standard), (1990), "Logical Specifications", U.S. Geological Survey, Version 12/90.

Strand, E.J., (1994), "ISO Extends Structured Query Language the 'SAIF' Way", GIS World, Vol. 7, No. 1, pp. 24-25.

Veregin, H., (1989), "Error modeling for the map overlay operation", The Accuracy of Spatial Databases, (Goodchild and Gopal, editors), Taylor and Francis, London, pp. 3-17.

Walsh, P., (1993), "Competitive Advantage in Geomatics", M.Sc,Eng. Thesis, The University of Calgary Geomatics Engineering Publication, Calgary.

Webster, (1984), New Webster's Dictionary of the English Language, Delair Publishing Company, Inc.

Weinberg, R., Guimaraes, T. and Heath, R., (1992), "A Data Base Manager's Guide to Object-Oriented Systems Development", Data Base Management, Auerbach Publications, November/December.

Wiegand, N. and Adams, T., M., (1994), "Using Object-Oriented Database Management for Feature-Based Geographic Information Systems", Journal of the Urban and Regional Information System Association, Spring, Vol. 6, No. 1, pp. 21-36.

Yourdon, E., (1989), Modern Structured Analysis, Prentice-Hall Inc., New Jersey.