Bo Guo, PE, Ph.D.
Senior Programmer and DBA,
Maricopa County Assessor's Office
301 W Jefferson St. Suite 250
Phoenix, AZ 85003
602-506-0930
boguo@mail.maricopa.gov

Russ Heisinger
GIS Division Manager
Maricopa County Assessor's Office
301 W Jefferson St. Suite 250
Phoenix, AZ 85003
602-506-3406
rheising@mail.maricopa.gov

BUILDING A SPATIOTEMPORAL PARCEL MANAGEMENT SYSTEM ON A RDBMS PLATFORM

Abstract: While spatiotemporal databases (temporal GIS systems) have yet to emerge as commercial products, recent developments in the temporal and spatial database technologies have made it possible to implement temporal GIS systems, which reconcile evolutions of spatial objects as well as non-spatial events that are associated with the spatial objects.

Since 1999, the Maricopa County Assessor's Office has embarked on a GIS-based Parcel Information Management System, or PIMS. The project started by converting existing parcel maps in paper or in various CAD formats to a standard CAD format before converting into an Oracle Spatial layer. The CAD to Oracle Spatial conversion was assisted by an in-house developed conversion tool, which not only conducts extensive quality checks, but also uses a spatiotemporal parcel leveling technique to automatically maintain correct spatiotemporal topology during parcel transactions.

This paper has three parts. The first part presents the background of the PIMS project, including the discussion of the selection of the software platforms on which PIMS is built. The second part introduces the development in spatiotemporal database, in particular the advantages of the third-generation of spatial data technology as the key GIS platform, and the advantages of the object-based database technology in modeling the spatiotemporal features. The third part introduces the spatiotemporal leveling technique, its underlying rules and its applications.

BACKGROUND

Maricopa County, Arizona, is one of the largest and fastest-growing counties in the nation. Four years ago, the County Assessor initiated an effort to bring the Assessor's Office into the 21st century by converting cadastral mapping into GIS. It was envisioned that a GIS-based Parcel Information Management System, or PIMS would be created that would allow various customers to explore current as well as historical parcel information through various GIS-based interfaces on different platforms.

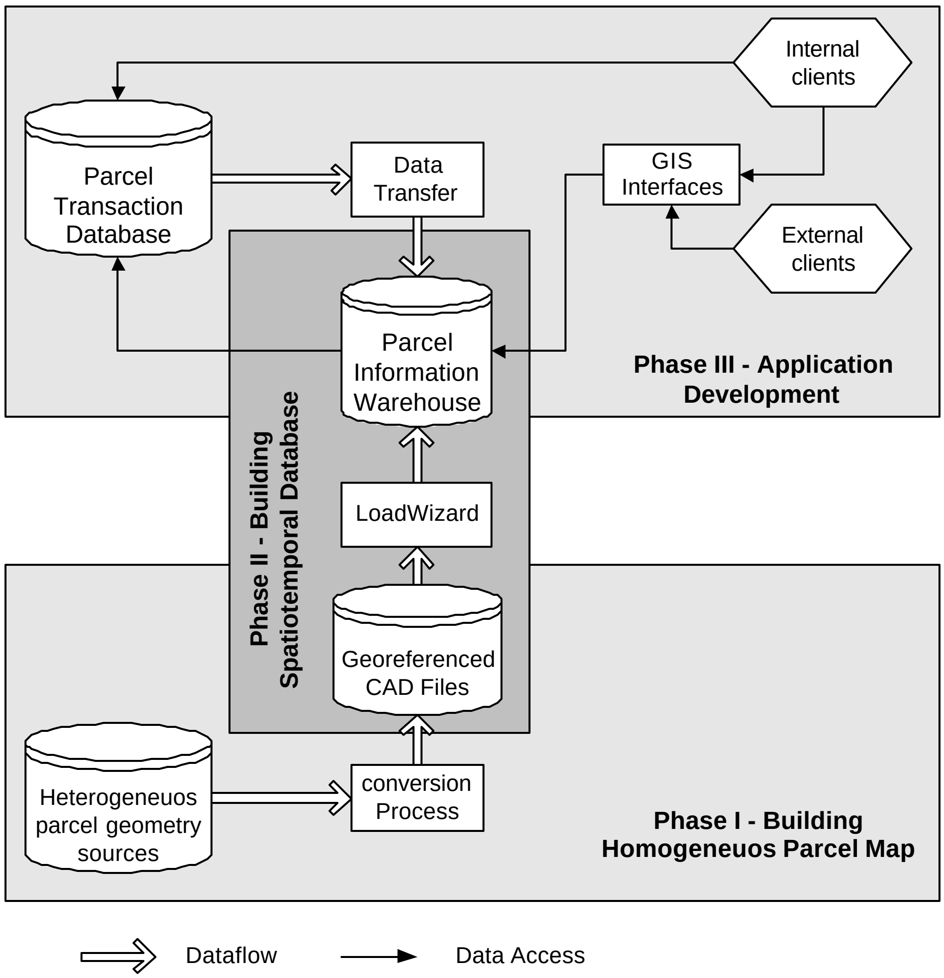


FIGURE 1 PIMS PROJECT OVERVIEW

The PIMS project was staged in three phases as shown in Figure 1. Phase I of the PIMS project called for converting the heterogeneous sources of parcel maps into homogeneous, geo-referenced standard CAD files. In April 2002, after three years of effort approximately 21,000 paper maps (of which 3000 were in digital formats) representing about 1.2 million parcels had been converted into nearly 2300 standardized and geo-referenced CAD maps. This phase cost $1.7 million in total.

Currently, PIMS is well into Phase II that converts geo-referenced CAD maps into a spatiotemporal database layer, which serves as the foundation for applications development. To successfully execute Phase II, it is critical to select suitable software environments, to have a solid spatiotemporal database design, and to be able to automate the maintenance of spatiotemporal topology changed as a result of parcel-related transactions.

SOFTWARE ENVIRONMENTS

The Assessor's IT operations have been on quite diverse platforms. Parcel transaction and feature information are stored in an Oracle database. Parcel geometry data is stored in various CAD formats with MicroStation being the prevailing format. ArcView and ArcInfo are used for GIS analysis while AutoDesk's MapGuide is used for web development.

The existing environments have proved to be inefficient as far as GIS operation is concerned. Since the parcel geometry updates are performed in CAD, frequent conversion from CAD to ArcView and to MapGuide format is necessary to maintain GIS data "up-to-date." It takes an engineer well versed in GIS and CAD about 8 hours to finish one-pass of the conversion. Still, the resulting data has limited capability to link to the transactional data stored in Oracle. Moreover, historical parcel geometry is lost in the process. The existing GIS environment has been deemed incapable of supporting PIMS.

Spatial Database Platform

The most crucial software decision is perhaps the selection of a spatial database platform. Such a database system serves as the central repository of spatiotemporal data that will be converted from CAD maps and eventually serves various client applications.

Spatial database technologies have gone through several generations (Adler, 2001). The first generation technology was represented by AM/FM/GIS packages, where the application interface (API) communicates with proprietary data storage formats. The second-generation products access and process spatial data stored in BLOB of RDBMS databases through proprietary middle ware, such as ESRI's SDE, and MapInfo's SpatialWare. The latest development, represented by Oracle Spatial, IBM DB2 Spatial Extender, and IBM Informix Spatial DataBlade, has seen RDBMS as both storage and processing servers to spatial data. There are many advantages to this latest development.

1. Spatial data becomes part of the "normal" enterprise database system. There is no division between spatial data and non-spatial data any more. Spatial data is stored in tables in a database just as non-spatial data.
2. Spatial data enjoys RDBMS benefits. Database administration and development tools can maintain security, and integrity of spatial data. Database server architecture promises better performance and scalability, and platform-independence.
3. Spatial data is now open more than ever. Spatial data can be accessed and processed directly from SQL as well as by third party GIS applications/tools. The SQL access simplifies client application development.
4. Spatial data enjoys strong industry support. RDBMS vendors that develop the technologies tend to have established market and industry support. Their active

participation in OpenGIS Consortium (OGC[1]) further encourages industry support.
5. These products are built on object-relational technology in which object topology is not stored but computed in the database. The relaxation of the topological constrains makes the space-time composite model a viable method in modeling spatiotemporal events. Discussion on the issue is found in the next section of the paper.

Oracle Spatial was selected among the third generation products. This was mainly because Oracle had been the County Assessor's primary database platform for over twelve years, and we have extensive in-house expertise as well as existing data and applications in Oracle. Meanwhile, Oracle Spatial had solid industry support. Products by ESRI, Intergraph, MapInfo and AutoCAD were supporting Oracle Spatial to various extents. This was a very important advantage as selection of the best or most suitable GIS and CAD software was made possible.

GIS Software

Once Oracle Spatial was selected as the spatial database platform, four desktop GIS packages and two map-control products that supported Oracle Spatial were evaluated. The main selection criteria were Oracle Spatial access type, Oracle Spatial format fidelity, ability to handling of topology, ability to validate and fix geometry errors, presentation and analysis capabilities, and license and pricing schedules. GeoMedia Pro 4.0 was selected to automate CAD to Oracle Spatial conversion for its strong CAD and Oracle Spatial support and robust automation programming tools. ArcView and ArcIMS won the favor for their capability of direct connections to Oracle Spatial and their presentation and analysis features.

CAD Packages

With approximately 80,000 parcel updates a year, CAD was deemed the most efficient and economical tool to manage this workload. The Assessor's Office has been a MicroStation shop for over eight years with about 20 technicians and engineers using MicroStation tools on a daily basis to keep up with the transactions that involve parcel geometry changes. The successful conversion from MicroStation CAD format to Oracle Spatial consolidates MicroStation as the main CAD platform. It is conceivable that PIMS parcel geometry maintenance process can be simplified once MicroStation future releases fully support Oracle Spatial as its storage format.

The All-star Software Team

The software evaluation concluded the following all-star software team to address PIMS needs:

1. Oracle and Oracle Spatial to store and process geographic as well as non-geographic data;
2. MicroStation and GeoGraphics to capture and maintain parcel geometry and conduct initial quality control;
3. ArcGIS8.1 and ArcIMS to distribute and analyze parcel data; and

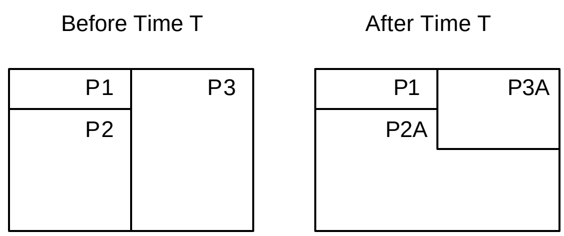
[1] In OGC, GIS software vendors, earth-imaging vendors, database software vendors, integrators, computer vendors and other technology providers work together to reach agreement on the technical details of open interfaces that allow these systems to work together.

4. GeoMedia Pro 4.0 for quality control and for loading parcels in MicroStation format to an Oracle Spatial Layer.

SPATIOTEMPORAL DATABASE DESIGN

A database is a tool for modeling real world phenomena that exist in time and space. Research in spatiotemporal database started in the late 80's; however, RDBMS' recent use of object or object-relational technology, and the third generation of spatial data technology (discussed in previous section) have made it possible for end-users to develop spatiotemporal applications of their own. This section briefly discusses a few spatiotemporal data models, and in particular, the benefit of using object-based database technology in modeling spatiotemporal changes of parcels. Since the field of temporal or spatiotemporal database is vast and dynamic, interested readers are encouraged to read relevant research studies (Clifford, Tansel, 1985, Jensen, Snodgrass, 1993, Langran, 1992, Snodgrass, 1987, Tansel, 1993.) for important concepts such as the time domain, levels of versioning, types of temporal databases and temporal data representations etc.

In modeling vector-based spatiotemporal data, there are snapshot model, amendment model and space-time composite model (Langran, 1992, pp. 40-48). The snapshot model records a sequence of time slice images, each of which represents an independent spatial dataset at a particular time. The snapshot model lacks temporal structure to capture temporal topology and carries significant data redundancy. Most GIS dataset we see today belong to this category. The amendment model removes redundancy in the snapshot model by recording only the differences between two consecutive editions. However, shortcomings of the snapshot model persist in the amendment model. It requires more than simple merging or overlay of amendments and base to obtain the cohesive image for a given time.



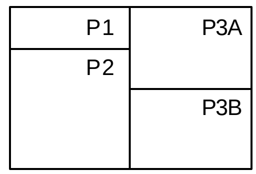
P_ID	Owner
P1	Tom
P2	Joe
P3	John

P_ID	Ower
P1	Tom
P2A	Joe
P3A	John

FIGURE 2 A PARCEL TRANSACTION AT TIME T

The space-time composite model keeps time-stamped spatial objects in a single consistent spatial dataset thereby eliminating redundancy while maintaining temporal topology. According to Langran [Langran, 1992, pp. 46], the space-time composite model suffers "unfortunate side effects: the representation decomposes into progressively smaller objects, and the identifiers of changed objects must be altered retroactively." These drawbacks, however, have been largely eliminated when an object-based instead of topology-based spatial data storage scheme is assumed.

To illustrate the differences in the implementation of the space-time model in two storage schemes, a simple parcel transaction is shown in Figure 2. Before the transaction occurs at time T, Tom, Joe and John owned parcels P1, P2, and P3 respectively. John, the owner of P3, sold portion of his parcel P3 to Joe, the owner of P2, resulting in new parcel and ownership configurations represented by P2A and P3A.



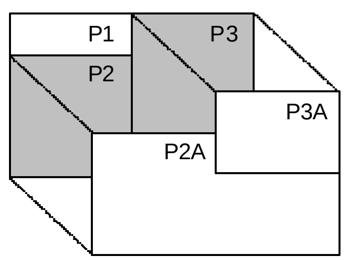
P_ID	Owner Before T	Owner After T
P1	Tom	Tom
P2	Joe	Joe
P3A	John	John
P3B	John	Joe

P3A & P3B merges to represent what John owned before T; and the combined state of P2 and P3B represents what Joe owns after T.

FIGURE 3 TIME-SPACE COMPOSITE MODEL IMPLEMENTED IN TOPOLOGICAL STORAGE SCHEME.

If the space-time composite model is used to depict the transaction in a topological storage scheme context, P3 is decomposed into P3A and P3B in order to maintain the correct topology. Parcel P3B represents the portion of P3 that changed hands at time T (Figure 3). Due to the decomposition, P3B needs to be coalesced into either P3A or P2A in order to represent a complete ownership state at any given time. Accumulation of transactions over time amongst parcels will lead to an increased number of parcel fragments and increased complexity in relationships between parcel ownership and parcel fragments.

In the object storage scheme spatial topology is not stored, providing the freedom to store geometric objects in a database without any topological pre-conditions. As a result, the geometry of the parcels P2 and P3 changed by the transaction can be kept intact as the before-image, while the new geometry representing the after-image of the transaction (P2A & P3A) can be stored in the database "on top of" the before-image. Semantically, the before image is retired and replaced in its entirety by the transaction that has created the after-image (Figure 4). Object or Object-relational databases, such as Oracle Spatial, lend itself very well to implementation of the space-time composite model.



P_ID	Owner Before T	Owner After T
P1	Tom	Tom
P2	Joe	-
P3	John	-
P2A	-	Joe
P3A	-	John

P2 and P3 were retired and replaced by P2A and P3A respectively at time T.

FIGURE 4 TIME-SPACE COMPOSITE MODEL IMPLEMENTED IN OBJECT STORAGE SCHEME

However, the freedom provided by the object technology comes with a price: users are responsible for maintaining topology to safeguard their spatial or spatiotemporal database. Maintaining proper topology is one of the key components of maintaining integrity of spatial or spatiotemporal databases. The next section will introduce a technique that maintains parcel topology to protect spatiotemporal parcel databases.

PARCEL TRANSACTION AUTOMATION: PARCEL LEVELING TECHNIQUE

Common parcel-related transactions are Parcel Split, Parcel Merge, Parcel Merge-split, Parcel Creation and Parcel Retirement. ROW-related transactions are ROW (right-of-way) Acquisition and ROW Abandonment. These transactions can result in complicated spatiotemporal changes in the parcels involved. Temporally, a parcel transaction often leads to the creation of is-parcel(s) and to the retirement of was-parcel(s) on the date of transaction. On the spatial side, the nature of the transaction dictates what spatial/topological relationships should exist between the is-parcel(s) and the was-parcel(s). In the transaction illustrated in Figure 5, the combined geometry of is-parcels equals to the geometry of the was-parcel, but such is not the case in transactions where both parcels and ROW have changed (Figure 6).

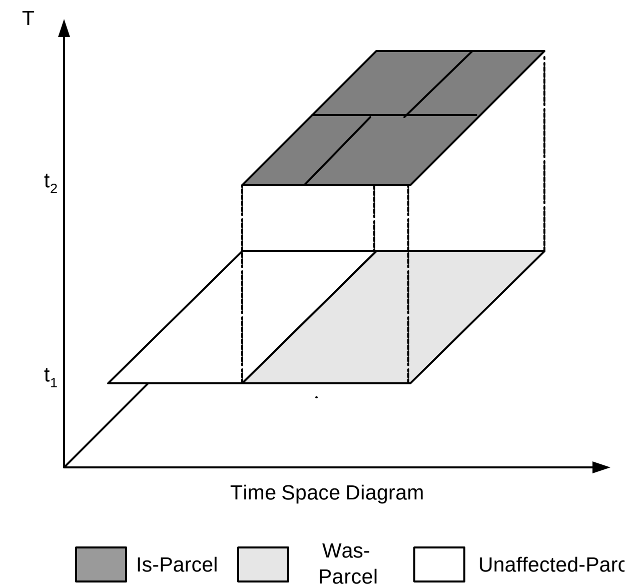


FIGURE 5 SIMPLE PARCEL SPLIT TRANSACTION

Figures 5 & 6 illustrate spatiotemporal topological handling for two transaction types each with specific parcel configurations. Various types of transactions coupled with various spatial parcel configurations can result in quite complicate spatiotemporal topologies. Therefore, it is highly desirable to

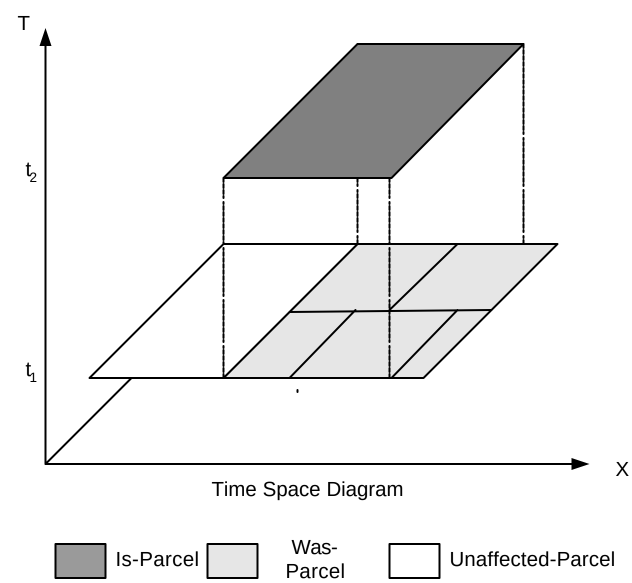


FIGURE 6 PARCEL MERGE WITH ROW ACQUISITION

develop a general "parcel leveling" technique, as part of a transaction process, to ensure the spatiotemporal data integrity.

In a typical parcel transaction process, parcel leveling first qualifies that is-parcels and/or was-parcels are legitimate parcels for the given transaction based on their spatiotemporal features. If they do, the was-parcel(s) is (are) retired and the is-parcel(s) is (are) created on the date of the transaction. The key, therefore, is to develop a set of rules that can positively qualify the parcels involved before executing transactions. In most transactions, this means identifying the spatiotemporal lineage between was-parcel(s) and is-parcel(s).

In search for the rules, a set of parcel/ROW operations that are fundamental to the various transactions was examined. These basic operations are classified in terms of the nature of geometric transformations such as creation, retirement, split, merge, etc. In terms of geometric equivalency before and after the transactions, the basic operations are further classified as symmetrical or asymmetrical. An operation is asymmetrical when combined is-parcel geometry is not equal to the combined was-parcel geometry. Typical asymmetrical operations are Creation and Retirement operations. Other operations such as Split or Merge that involve ROW transactions are also asymmetrical (Figure 7).

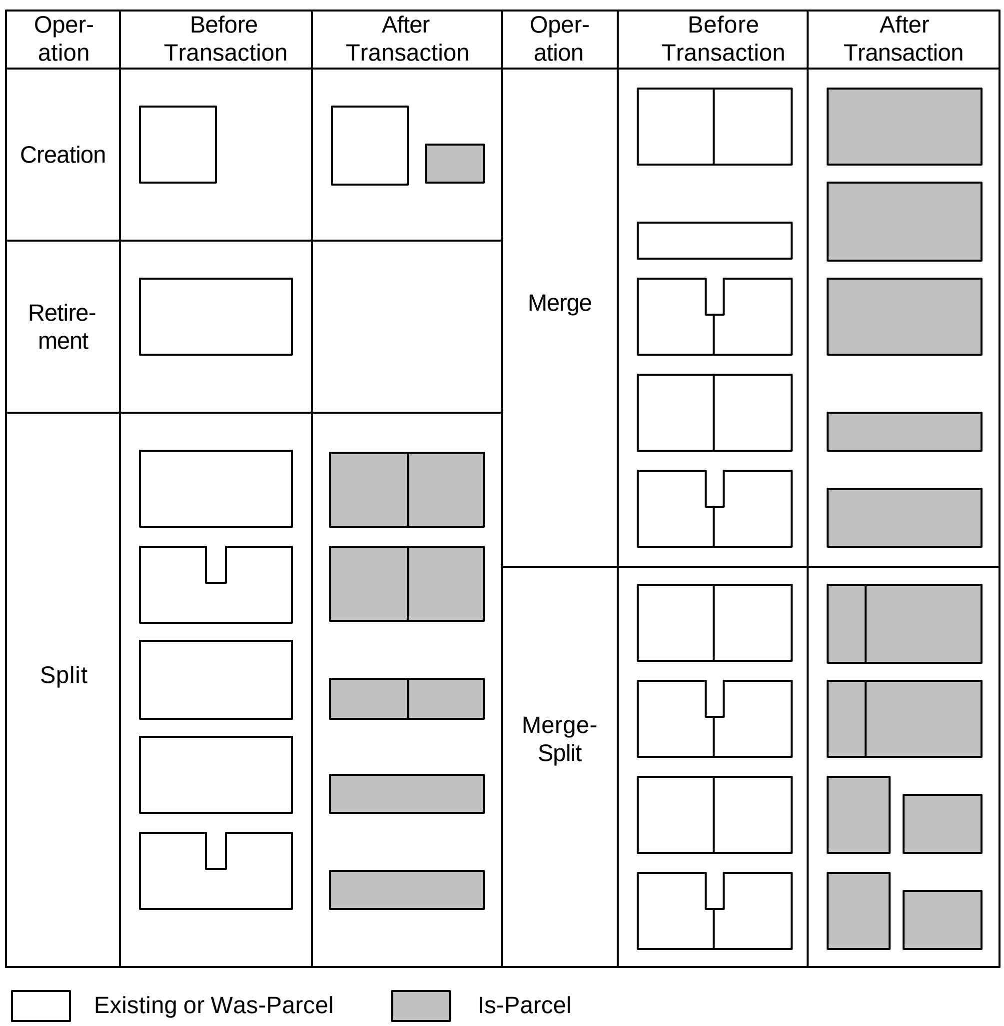


FIGURE 7 BASIC PARCEL OPERATIONS

The study of the basic operations has led to the three rules that can qualify the was-is parcel lineage. They are the Non-Encroachment Rule, the Containment Rule, and the Identity Rule.

The Non-Encroachment Rule specifies that is-parcel(s) should have touch or disjoint relationships with existing parcels other than was-parcel(s). In other words, only was-parcel(s) should be affected by transaction that has resulted in is-parcel(s). The Containment Rule states that is-parcel(s) should form contain or cover relationships with was-parcel(s). Or, the combined geometry of is-parcels should either contain or be contained by the combined geometry of was-parcels. The last rule, the Identity Rule, applies only the symmetrical operations, where the combined geometry of is-parcels should be identical to the combined geometry of was-parcels.

A symmetrical Split transaction shown in Figure 5 is ideal to illustrate the application of all three rules in order to qualify the lineage between is- and was- parcels. First of all, the is-parcels should not encroach upon any unaffected parcels (or existing parcels that are not was-parcels) by forming either touch or disjoint relationships with the unaffected parcels. Secondly, the is-parcels should be fully contained by the was-parcel. Lastly, the combined geometry of the is-parcels should be identical to the was-parcel. Only when all three rules are met before the lineage can be positively identified and the transaction be completed. In asymmetrical transactions, the identity rules are relaxed. In Parcel Creation transactions where there are only is-parcels, only the non-encroachment rule applies. Parcel Retirement transactions, which involve only was-parcels, do not need to meet any of the rules.

The parcel leveling algorithm is process and data intensive, therefore it is more efficient to be implemented on database servers. The back-end or the server approach also enjoys the benefit of supporting different clients regardless of their GIS and/or programming platforms. When the back-end approach is assumed, leveling logic can be written through triggers or through stored procedures. The trigger implementation can provide further protection of spatiotemporal database as it rejects uncontrolled database updates that violate the rules. The drawbacks are the additional complexities resulting from the restrictions of trigger programming. The stored procedure implementation is more flexible, but additional security measures should be taken to protect the spatiotemporal database from any unauthorized update attempts.

CONCLUSIONS

In Phase II of Maricopa County’s PIMS project, a CAD to Oracle Load Wizard was developed using Visual Basic, GeoMedia 4.0 automation and Oracle Spatial. The load wizard has been successful in not only loading qualified parcels in CAD files into a seamless spatiotemporal parcel layer, but properly maintaining the spatiotemporal integrity as well. Maricopa County’s experience has shown that the recent database technological advancements have provided IT/GIS professionals with opportunities to develop spatiotemporal parcel management systems. In particular, the object relational databases that support spatial data are solid platforms upon which to develop spatiotemporal applications. However, the success of such applications lies on careful selection of software environments, sound design of spatiotemporal databases and implementation of the parcel leveling technique.

REFERENCES

Adler, D., 2001. “IBM DB@ Spatial Extender – Spatial Data Within the RDBMS”, Proceedings of the 27th VLDB conference, Rome, Italy.

Clifford, J., Tansel, A., 1985. “On an Algebra for Historical relational Databases: Two Views,” Proceedings of the ACM SIGMOD International conference on Management of Data, May 1985, pp. 247-265

Jensen, C.S., Snodgrass, R., 1993. “Proposal for a Data Model for the Temporal Structured Query language,” TempIS technical Report No. 37, Department of Computer Science, University of Arizona, Tucson, AZ.

Langran, G., 1992. Time in Geographic Information Systems, Taylor & Francis.

Snodgrass, R., 1987. “The Temporal Query Language Tquel,” ACM Transaction on Database Systems, Vol. 12, No. 2, pp. 247-298.

Tansel, A.U., 1993. “A generalized Relational Framework for Modeling Temporal Data,” Temporal Databases: theory, Design, and Implementation, Benjamin/Cummings Publishing Company, pp. 183-201

ACKNOWLEDGEMENTS

The authors like to acknowledge the support from many staff members in the Maricopa County Assessor’s Office, in particular, the PIMS project team members: Frank Harrison, Glenn Emmanuel, David Minton, Steve Bell and Robert Gubser for their contributions and inspirations. The authors also like to thank Michael Agne and Jian Fei who provided valuable feedback and constructive comments.