Map Series Production Using ArcGIS (Workstation/Desktop) and ArcSDE

Byron W. Taylor

Abstract

Large scale topographic quadrangle maps (1:24,000/1:63,360) and intermediate scale base series maps (1:126,720) to support the mission of the United States Department of Agriculture (USDA) Forest Service are produced at the Geospatial Service and Technology Center (GSTC). When aging technology put GSTC's legacy map production capability at risk, a project was launched to migrate map production to the Forest Service corporate environment which included ArcGIS technology. The project was accomplished by assembling a multi-disciplinary team that included GSTC developers and product experts, as well as a contractor (T-Kartor Sweden) experienced in automated cartography software development. The project was divided into four general phases: Requirements Analysis, ArcSDE/Oracle Database Design, Software Prototyping and Testing, System Configuration and Implementation. This paper presents the development of GSTC's new mapping system and summarizes the lessons learned.

Section 1 – Introduction

This paper documents the migration of GSTC's legacy map production system (Intergraph) to the ArcGIS environment. The primary objective is to document the migration as a software development project, not to provide software functionality detail. Information in this introductory section is provided as an overview to develop the context for the sections that follow. In some cases, supporting documentation has been placed in the *Appendices.*

Mention of trade names or commercial products does not constitute endorsement or recommendation for use by the U.S. Government.

Project Overview

System Name: Cartographic Production System (CPS)
Project Duration: June 2000 – July 2004
Contracts: The project was accomplished with two contracts. The first (cartographic editing) was awarded to T-Kartor Sweden in May 2001. The second (cartographic output) was awarded to T-Kartor Sweden in June 2002. CPS was built on T-Kartor's commercial-off-the-shelf (COTS) product *Cartographic Production System Next Generation* (CPS NG).

Divisions of Responsibility: T-Kartor was responsible for virtually all of the cartographic production software. They were also responsible for online user help documentation and formal software training. GSTC was responsible for database design, database migration, system configuration, symbology development, data integration, and relief and woodland map layer process development. Digital Visions, a Forest Service enterprise technology services unit, was responsible for developing Secondary Base Series (SBS) generalization tools.

CPS Software

COTS Software	ArcInfo Workstation, ArcMap, ArcSDE, ArcView, Oracle, CPS NG, Harlequin Raster Image Processor (RIP), Alchemy (RIP), Adobe Illustrator
Custom Software	CPS NG/Forest Service, Digital Visions Generalization Tools, utilities created by GSTC developers
Programming Languages	Arc Macro Language (AML), Delphi, Visual Basic, C

CPS Hardware

CPS Server	IBM P620 (AIX), 4 Processors (600 MHZ), 6 GB RAM, 12 X 36 RAID
CPS Clients (32)	IBM PC, 2.8 GHZ processor, 1 GB RAM, 80 GB Hard Drive
Network	Gigabit bandwidth from server to client

CPS Functionality Overview

Two modules contain core CPS functionality:

Editor: A graphical user interface (GUI) written in Delphi that makes the rich command set of ArcEdit easy to use. The Editor GUI conforms to industry standards. It presents and simplifies a long list of basic ArcEdit commands and provides a variety of tools, many that are customized for GSTC map production requirements.

Product Manager: This suite is used to define and administer all cartographic product decisions and to perform cartographic output. It includes a variety of tools to define and enforce product specifications, generate neatline and marginalia map elements, and apply common cartographic information to a user-defined map group.

Other modules provide a wide array of cartographic production applications that support data management, symbology management, historical edit tracking, workflow management, rules definition and management, database editing, and map projection.

For more information on CPS NG, visit http://www.t-kartor.se/ Products> CPSNG.

CPS Map Product Descriptions

Single Edition Quadrangle (SEQ)	
Scale	1:24,000 except for Alaska which is 1:63,360
Map Sheet Extent	7.5 minutes by 7.5 minutes except for Alaska which uses two extents according to latitude: 15 minutes by 20 minutes, and 15 minutes by 22.5 minutes.
Projection	Universal Transverse Mercator (UTM)
Datum	North American Datum (NAD) 83
Coverage	All national forests in the United States (approximately 10,600 quadrangles)
Major Themes	Roads, Drainage, Contours, Land Survey, Ownership Status, Culture, Landforms, Woodland Tint, and Text
Users	Forest Service Personnel, Public, Other Government Agencies
CPS Derived Products	Printed 7.5 minute Topographic Quadrangle (SEQ) maps, Digital Raster Graphics (DRGs), Vector Data, Text Data

SEQ Cartographic Challenges:

- Map production standards are highly detailed and tightly controlled by the Single Edition Agreement with the United States Geological Survey (USGS).
- This map series is designed for a diverse user base and a variety of applications. Cartographic problems cannot be resolved by thematic simplification or by compromising standards or specifications. Symbology conflict resolution is particularly challenging.
- Certain themes become outdated relatively quickly (such as roads and culture).
- Marginalia are complex and highly variable or conditional.
- Neatline coordinate information is complex and conditional.
- Revision data must be integrated from a variety of sources and applications.

Secondary Base Series (SBS)	
Scale	1:126,720 (other scales on an occasional basis)
Map Sheet Extent	Entire forest or part of a forest
Projection	Albers
Datum	North American Datum (NAD) 83
Coverage	All national forests in the United States
Major Themes	Roads, Drainage, Contours or Shaded Relief, Land Survey, Culture, Landforms, Ownership Status, Recreation, Wilderness, Special Areas, and Text
Users	Public, Forest Service Personnel, Other Government Agencies
CPS Derived Products	Printed Forest Visitor Maps, SBS Raster Products, Vector Data, Text Data

SBS Cartographic Challenges:

- Like SEQ maps, SBS maps are multi-themed with a wide variety of users and applications. The same complexities apply; however, the enhanced display of recreational and land-use themes introduces additional challenges.

- Most SBS data sources are collected at a scale larger than 1:126,720. Substantial production time is needed for data aggregation, feature displacement, thinning, and other generalization tasks.
- Since the SBS area of interest (AOI) is substantially larger than the SEQ AOI, other data sources (such as TIGER, Digital Line Graphs, state and forest GIS) must be acquired for integration with the generalized SEQ data.
- Conflict decision is increased by the smaller scale and additional feature content not found in the SEQ.
- Even though SBS is defined as a map series, certain characteristics limit typical map series production efficiencies:
    - Map sheet size, extent, marginalia, and feature content are highly variable.
    - Map scales can vary.
    - Feature display varies across the country and between Forest Service Regional Offices where agreements have been drafted independently.
    - Numerous standards and symbol sets have been created and perpetuated because there is no system for nationally integrated and applied symbology.
    - A cohesive intermediate scale national data set containing all SBS themes does not exist.

The Project Team

GSTC Team members were selected according to their experience and success in their respective disciplines. Team make-up changed over time according to evolving project phases. Eventually it became an "aggregate" team that included T-Kartor as well as Digital Visions. Henceforth, "Team" will be used to identify the aggregate project team. The following areas of subject matter expertise were represented on the Team: general cartographic production, SEQ production and management, SBS production and management, CPS NG production and management, cartographic/geospatial software development, Oracle/ArcSDE database administration, GIS, and project management.

Section 2 – Requirements Analysis

The value of effective requirements analysis to the success of any software development project cannot be overstated. The GSTC Team addressed requirements at a high level during *upstream* project work, when alternative solutions were explored and design changes were relatively cheap. Low-level requirements were analyzed later during the *downstream* project phase when high-level requirements were firm and the architecture was fairly well-defined. A *checkpoint review* was held at the end of upstream requirements analysis so that resulting conclusions could be reviewed and evaluated by all project stakeholders before moving on to downstream work (for a more complete discussion of upstream, downstream, and checkpoint review concepts, see: Steve McConnell, *Project Software Survival Guide*, Microsoft Press, 1998, pages 28-40).

Upstream Requirements Analysis

Requirements Framework

The GSTC Team was directed to work within a project framework described by a scope definition and associated constraints and assumptions. As requirements analysis progressed, this project framework was referenced repeatedly by the Team as our project "constitution".

Technology Evaluation

After reviewing the project framework and conducting preliminary requirements discussions, the GSTC Team recommended a *Technology Evaluation* as the project's first phase. It was designed to support technology research to increase the GSTC Team's knowledge base for improved decision making and planning. The GSTC Team advised GSTC management that to endeavor on a development path without adequate upstream research and requirements analysis would introduce an unacceptably high level of risk. GSTC management agreed and the Technology Evaluation was conducted *(for Technology Evaluation and requirements framework details, see Appendix A – Requirements Analysis).*

Project Checkpoint Review

The GSTC Team also recommended that a *checkpoint review* be conducted at the conclusion of the Technology Evaluation. The checkpoint review provided an opportunity for all project stakeholders to review the results of the Technology Evaluation and share in final decisions prior to downstream development.

To conduct the checkpoint review the GSTC Team consolidated their Technology Evaluation findings into a report that included recommendations for project development. The report was distributed to project stakeholders that included GSTC's management team and the GSTC steering committee. The stakeholders agreed with the recommendations of the GSTC Team:

- The legacy system should not be rebuilt.
- A contract should be awarded to a commercial vendor to help build CPS.
- CPS should be built in the Forest Service corporate environment, however, the decision to use ArcInfo Workstation or ArcInfo Desktop should be left to the offerors to propose in their responses to the RFP.

Downstream Requirements Analysis

Contract Award

After the checkpoint review, the GSTC Team wrote a Request for Proposal (RFP) to start the contract procurement process. The RFP represented a well-defined threshold leading from upstream to downstream work because it demanded that requirements be defined to a lower level of detail. However, the RFP emphasized the need for ongoing requirements analysis after contract award. The contract was awarded to T-Kartor Sweden.

Prototype Zero
T-Kartor's proposal called for three formal prototype events: Prototype Zero, Prototype One, and Prototype Two (Prototypes One and Two will be explained in *Section 4 - Prototyping and Testing*). Prototype Zero served as the contract kickoff where low-level requirements analysis began in earnest. The two-week agenda was driven by the Prototype Zero Report that had been prepared by T-Kartor and contained a comprehensive analysis of RFP requirements in the context of CPS NG COTS software. Prototype Zero activities:

Software Development Diagnostic (SDD) Analysis: The SDD was created by T-Kartor to ensure that all project requirements were analyzed systematically. It categorized requirements into three groups: New (to be developed); Base (already exists in CPS NG COTS); and Uncertain. The first version of the SDD was included in T-Kartor's contract proposal and provided a preliminary analysis of RFP requirements. During Prototype Zero, the Team collaborated to refine and update the SDD, working to move all requirements listed as Uncertain, to either the Base or the New categories.

Concept of Operations (CONOPS) Presentation: The CONOPS was presented as both a written and oral report. It served as a vehicle to initiate discussion that helped T-Kartor and GSTC reach common understanding and agreement on the design of new functionality.

Software Verification Demonstrations: By observing COTS CPS NG functionality, the GSTC Team was able to determine what CPS NG COTS components met their production requirements. The SDD was updated accordingly.

System Design Presentation: T-Kartor integrated their original CONOPS with design elements mutually defined during Prototype Zero and presented the design to the GSTC Team. By the end of Prototype Zero, the Team had agreed on the system design and development strategy.

Post Prototype Zero Requirements Analysis
Requirements analysis continued throughout the project on two levels: 1) Formal – driven by scheduled prototype deliveries that were accompanied by on-site requirements analysis following the pattern established in Prototype Zero; 2) Informal – driven by patch software updates, software tests, e-mail, and conference calls.

Conclusion
Once this project was launched requirements analysis became the most demanding activity, at least from the perspective of the GSTC Team. We have become proponents of the upstream/downstream approach to software development and feel strongly that this project succeeded because of the upstream requirements analysis performed during the Technology Evaluation. We have confirmed that software development success is tied directly to thorough and painstaking requirements analysis.

Section 3 – ArcSDE/Oracle Database Design

Database design for CPS started during the Technology Evaluation. At that time a decision had not been made whether the ArcGIS geodatabase would be used in the CPS database. To assist in making that decision some members of the GSTC Team attended ArcGIS geodatabase training. Also, a GIS consultant experienced with the geodatabase model was hired to help the GSTC Team perform a thorough analysis of geodatabase implementation considerations.

Database Design Step 1 – Design a Geodatabase for CPS

Even though database model decisions had not been made, we determined that the logical structure of a CPS geodatabase would translate into any other database model. The database design consultant confirmed this and directed the GSTC Team as we designed a first-cut CPS geodatabase.

Database Design Step 2 – Adapt the Geodatabase Design for CPS

When the contract was awarded to T-Kartor in May 2001, the Team concluded that CPS would not be built with a geodatabase. The geodatabase design created in the previous step was adapted into a first-cut CPS production database design. The most important design work performed during this step focused on the structuring of the logical thematic spatial layers that would most effectively support database-driven, map series production.

Database Design Step 3 – Refinement of the CPS Database Design

This stage of database design lasted for a significant portion of the project and required extensive collaboration with T-Kartor. The database design matured on a parallel path with requirements analysis and software prototyping, focusing on the refinement of the logical database structure and associated attribute definition.

During this step the database design was enhanced to improve map series production relative to non-spatial margin information that is common to a group of adjacent maps. For example, each SEQ map in a group of adjacent map sheets has a unique geographic extent. However, much of the margin information on each of those same map sheets is identical. An SEQ database table was designed so that more than forty marginalia items can be populated in a single process on a map group size of the user's choosing.

Database Design Step 4 – Database Migration

Eventually, the CPS database design reached a level of maturity where we determined that populating the CPS database with legacy data carried an acceptably low level of risk. Initial population of the CPS database took place in two steps: 1) Legacy SEQ data were populated into CPS SEQ cartographic layers; 2) Legacy SEQ data were populated into CPS SBS cartographic layers.

Database Design Step 5 – CPS Database Maintenance

This step is more accurately described as a phase that persists for the life of the system. During this phase the design is mature, requiring minor and infrequent modifications. The CPS database design will likely remain relatively stable until compelling business reasons emerge that make migration to the geodatabase necessary.

Conclusion

Much of the risk inherent to software development, including database design, can be mitigated by iterative and incremental staging. The "coarse" nature of the first-cut geodatabase design is what made it readily adaptable to the eventual CPS database design. Each stage of the CPS database design provided new information that helped in the refinement of the following stage.

Section 4 – Prototyping and Testing

Formal Prototyping and Testing

CPS software was delivered in two stages. The primary functionality set of Stage I dealt with cartographic editing. Stage II focused on map finishing and map product output. T-Kartor's software development plan called for three scheduled prototype deliveries (in each of the two project stages) before the final delivery. T-Kartor visited GSTC for all formal prototype deliveries. Prototype Zero was explained in *Section 2 - Requirements Analysis*. Prototypes One and Two each lasted one week and provided intermediate prototype deliverables. The most important objective of prototyping and testing activities was to document the status of software maturity as the testers compared delivered prototype functionality with defined requirements.

Prototype One

The first formal customized build of CPS functionality in each stage was delivered to GSTC during Prototype One.

Prototype One activities:

- Demonstration of prototype functionality
- Instruction on new functionality for the GSTC Team
- First level software testing and defect documentation
- Definition of ongoing software tests to be conducted by the GSTC Team
- Presentation of the Concept of Operations for the next formal prototype
- Discussion of database design and system configuration impacts

Prototype Two

The main difference between Prototype One and Prototype Two was the maturity level of the software, otherwise they both followed the same general format.

Final Delivery

This could have been called Prototype Three. The format was similar to Prototypes One and Two and it represented a fairly significant functionality upgrade like the previous prototype deliveries. It differed in that closeout discussions at the conclusion of Final Delivery focused on Test & Acceptance procedures rather than the next prototype.

Informal Prototyping and Testing

Since the formal prototype deliveries were scheduled months apart, it was necessary to supplement them with smaller scale software updates. The informal prototype/test cycle complemented the formal cycle:

- Software testing and defect reporting that had not been completed during the formal prototype sessions continued into the informal cycle.
- Software updates were retrieved via FTP.
- Testing results were forwarded to T-Kartor and pertinent communication followed using e-mail, phone calls, document sharing, and T-Kartor's technical support system.
- Adjustments were made to database design and system configuration as required by maturing prototype deliveries.

Conclusion

T-Kartor's prototyping methodology was logical and well-designed. GSTC's greatest challenge relative to testing centered on resource allocation. Most of the GSTC Team carried regular production responsibilities in addition to their project assignments. CPS testing suffered because of production pressures. CPS maturity would have undoubtedly been accelerated with a more fully dedicated testing cadre. The absence of timely, exhaustive testing undermines all other development efforts.

Section 5 – System Configuration & Implementation

While system configuration and system implementation could be addressed independently, in this project they were sufficiently interdependent to warrant simultaneous treatment here. Logically, configuration design must, of course, precede implementation; however, we discovered that like other overlapping areas of project development, they required a significant level of parallel activity.

Risks Posed by Map Series and Database-Driven Production

In addition to defined functionality, the GSTC Team identified system performance as a critical requirement. T-Kartor was responsible to deliver functionality that embodied acceptable programmatic performance. GSTC was responsible to provide an Information Technology (IT) infrastructure that provided acceptable operational performance. The planned CPS client-server architecture posed the greatest risk to the implementation and operation of database-driven map series production:

- The CPS production model required that standards and rules be distributed and enforced from a single server-based platform where they are defined once and read many times by many clients.
- CPS was designed to exploit the economies of project-based production where the common characteristics of a logical group of map sheets could be applied with batch

processes or simplified interactive processes. This design is most effectively implemented where the production data reside on the server and processes are initiated at the client.

- Production efficiency is improved when project-sized datasets (multiple map sheets) can reside on a single server-based platform, available for edit by multiple clients.
- Software configuration management and maintenance is simplified when the applications and production components are deployed from a single server.
- The cartographic database must reside as a single instance on the server. There are no reasonable client-based alternatives.

Risk Mitigation Measures

The identified areas of risk were mitigated:

- The project team met with the system manager and network administrator early in the project to analyze risks, to devise and execute performance tests, and to create contingency plans. GSTC's IT staff worked closely with the GSTC Team throughout the project.
- The system manager set up a test production cell that was independent of GSTC's enterprise cell for early configuration testing.
- A server was configured, optimized, and fully dedicated to CPS production requirements.
- The network administrator periodically analyzed network performance and reported results to the project team.
- The gradual implementation of CPS over time enabled incremental or "baby step" problem resolution.
- CPS clients (Windows 2000 PCs) were optimized for production.
- T-Kartor collaborated aggressively with GSTC to resolve IT infrastructure-related problems even though they didn't "own" the problems.
- Known weaknesses in GSTC's LAN were corrected.

Implementation Challenges and Successes

In reality, CPS implementation has been more difficult than anticipated. Implementation problems that undermined general system stability earlier in the project have been resolved by aggressive collaboration between T-Kartor, the GSTC Team, and GSTC's IT staff. Random problems still occur, but they can usually be resolved fairly easily. The GSTC Team will continue to work towards improved system stability. In spite of the challenges, CPS implementation has been strengthened by the following:

- Once the major architecture issues were resolved and Stage I software was formally accepted, CPS map production began even though the project was not complete. Staged software delivery proved to be very effective in accelerating system maturity as the user base and production level increased, providing the opportunity to exercise the system more rigorously and realistically.
- CPS was designed to be flexible. It provides a "toolbox" of utilities so that users are not constrained to a rigid, pre-determined production path that might undermine creative problem-solving. Staged implementation provided the opportunity to test and define edit-related workflows prior to full-scale production.

Conclusion

The CPS configuration clearly benefited from aggressive risk mitigation measures that were strengthened by a good relationship with GSTC's IT staff. Success was achieved because problems were anticipated and mitigated. In retrospect, we should not have been surprised by software implementation challenges with a project of this complexity. It is likely that implementation problems would have been more successfully contained by "preemptive" measures similar to those applied to system configuration.

For additional detail concerning risks and problems resolved, and a CPS configuration diagram, see *Appendix B - Configuration and Implementation.*

Section 6 – Lessons Learned

This section was inspired by a Forest Service software development conference held a few years ago that featured a *Lessons Learned* session. Many of the lessons had universal application regardless of the nature of the IT development project. The GSTC Team learned many lessons during the course of this project.

Flexibility: A successful project requires a well-defined scope and a solid project plan. However, don't be surprised when unexpected events, conditions, or challenges appear. It is critical that the project plan embodies a certain degree of flexibility to allow for the unforeseeable. This project presented its share of "twists". Fortunately, the project plan was appropriately flexible.

Requirements Analysis: Early in the project, we assumed that requirements analysis would dominate the front end of the project and code writing would dominate the back end. While this was generally true, we learned that requirements analysis persists heavily throughout most of the project. In one sense this was a lesson in the reality of software development. This does not suggest, however, that efforts should not be made to contain requirements analysis (see *Communication* and *Formal Project Management Training* below).

Implementation: Implementation proved to be more challenging than expected. In general, it relates to the fact that CPS was developed at one site and implemented at another. Problems experienced during the project would have been mitigated by more formal treatment of system implementation in the RFP.

Communication: It is difficult to overstate the importance of communication to the success of a project. We would do some things differently if we started this project over:

A project communication plan would be defined as a distinct element in the RFP. The contractor would be asked to propose a communication plan designed specifically for the project. The proposed communication plan would become one of the elements for RFP evaluation. GSTC would collaborate with the contractor to integrate GSTC ideas into the communication plan. Much of what we learned here is basic to principles of project management (see *Formal Project Management Training* below). The plan would include:

- A mechanism to track documents and datasets shared between the contractor and GSTC
- A formal method of adding, removing, changing, and tracking requirements
- A reliable method of measuring and reporting project status in relation to established project milestones
- A formal plan for technology transfer that parallels staged software delivery
- A formal software defect reporting process to minimize the reporting of false defects to the contractor
- A formal plan to support bilateral understanding of emerging tasks and events

User Interface Prototype: This is a method of presenting the “look and feel” of the software application to the user before any of the underlying code is written (McConnell, page 117). When the user is satisfied with the functionality as represented by the interface mockup, coding can begin. This technique was used in this project to redesign the transaction management graphical user interface (GUI) and it greatly accelerated common understanding and redesign delivery. This method should be applied wherever practical in the software development process.

Formal Test & Acceptance: The project timeline called for an aggressive Test & Acceptance schedule that allowed only for the testing of system components in “laboratory” conditions rather than a true production environment. We understand now that this is unrealistic and unreasonable with something as complex as this project. Test & Acceptance should be defined to parallel the complete production cycle of a real product.

Software Testing: The importance of testing to successful software development is intuitive. Allocating adequate resources is another matter. When fighting resource allocation battles, place testing resources high on the list.

Project Resources: There will never be enough. Accept this reality and manage accordingly. Be vigilant in reporting resource risks to management. Understand the interdependency of the three key resource categories: time, personnel, money. Saying "Keep the project the same, but give me a shorter schedule" makes about as much sense as trying to make a basketball smaller just by squeezing it harder (McConnell, page 162).

Database Design: We thought that database design would “solidify” earlier in the project than it did. In retrospect, we should not have been surprised that as long as requirements analysis was active, database design was susceptible to modification.

Project Team Diversity: GSTC worked to assure that all appropriate disciplines were represented on the team but one position was overlooked until late in the project. When that position was filled, there were immediate benefits. Do not underestimate the importance of carefully selected subject matter experts.

Formal Project Management Training: No one on the GSTC Team was formally trained in project management principles. While selected books were very helpful, as well as the collective experience of the team, in the risky world of software development such training should be a requirement. This risk was mitigated by awarding the contract to a vendor who had a welldocumented track record of cartographic development success, drastically reducing the project management responsibilities of the GSTC Team.

Lessons Confirmed

This section was added to document lessons that were not necessarily new, but were reinforced as important to project success:

Contract: During the Technology Evaluation the GSTC Team concluded that for this project to be successful, it would be necessary to award a contract. While this was our firm belief at the end of the Technology Evaluation, three years later we are absolutely certain of that recommendation.

Trust: "Successful projects are built on people who are able to collaborate successfully and have trust with each other." — Edward Hoffman, director of NASA's Academy of Program and Project Leadership. The GSTC Team agrees overwhelmingly with Mr. Hoffman. The trust between GSTC and T-Kartor was critical to project success.

Management Support: This is essential and was manifest at various stages throughout the project. This does not mean that this project was the number one priority for the duration. It does mean that management leveled with the GSTC Team and let us know when it was not number one. This gave us the opportunity to document the risks of sacrificing resources to other priorities. Straight forward communication between GSTC management and the GSTC Team created a good project development atmosphere.

Project Duration: That software development will take longer than planned has become axiomatic. While the basic “project duration factor” of two (estimate project duration and multiply by two) did not strictly apply to this project, it took longer than anticipated. Make every attempt to establish project dates according to empirical analysis rather than wishful thinking.

Relationships: While it is obvious that a sound relationship with management is important, don’t forget other groups that are critical to project success. Work to foster strong relationships with IT, Contracting, and other support staffs. The same applies for external resources such as vendors and contractors.

Users: Software users simply have to be involved for the duration of the project; not as occasional “consultants”, but as full-time members of the project team.

Passion: Hand pick your project team; beyond knowledge and experience, look for passion. It will be required to carry the project through the discouraging and painful moments that are a certainty in a project of this nature.

Morale: If it’s all pain all the time, something is wrong. Work to have fun. At the beginning of this project, the GSTC Team “commandeered” GSTC’s Juniper conference room as our “war room”. We met for countless hours in that room. Not surprisingly, there were periods of stress and frustration. There were also abundant opportunities for laughter.

Conclusion

This *Lessons Learned* section demonstrates that this was not the perfect project. The GSTC Team learned a great deal that will undoubtedly be of value in the future. But perhaps the most important lesson learned is that when project stakeholders at all levels are committed to success, project imperfections can be overcome.

Section 7 – Appendices

Appendix A – Requirements Analysis

Project Definition

Project Scope

To migrate legacy map production processes (Single Edition Quad and Secondary Base Series) from the legacy environment to current, stable technology.

Project Constraints

- The new system shall operate within the USDA Forest Service corporate Information Technology (IT) infrastructure (ArcGIS, ArcSDE, Oracle).
- Custom software development shall be minimized through the use commercial-off-the-shelf (COTS) and government-off-the-shelf (GOTS) software.
- The project shall be funded by GSTC's baseline budget.

Project Assumptions

- The new system shall be constructed to exploit efficiencies available through database-driven, map series production.
- Production processes shall be automated to the extent practical.
- Finished map output shall support legacy filmwriting at GSTC as well as current industry standards preferred by service bureaus and other map producers.

Technology Evaluation Tasks

- Update existing map production use-case documents.
- Design a first-cut ArcGIS geodatabase.
- Participate in the ArcGIS 8.1 Beta Test.
- Test and evaluate ArcInfo Desktop and Workstation COTS as well as government-off-the-shelf (GOTS) functionality against SEQ and SBS mapping requirements.
- Obtain training for appropriate ArcGIS technologies.
- Learn more about Maplex.
- Interview potential project contractors.
- Interview ArcGIS 8.x users.
- Perform cost/benefit/risk analyses involving logical development scenarios.

Technology Evaluation Conclusions

Of the Technology Evaluation documentation generated, the following table is the single most useful piece of information that helped guide development recommendations. The GSTC Team compiled a comprehensive list of functionalities required to produce SEQ and SBS maps. We grouped the items on the list into eight logical categories (left-hand column on the Summary Table that follows) and assigned developers to each of them for requirements evaluation. We also identified four separate applications to evaluate against the eight categories (top row on the Summary Table). The information in the table summarizes the level of customization required beyond the COTS or GOTS version of the software.

In the *Summary Table* the terms *Limited*, *Moderate*, and *Extensive* indicate the level of custom development required to achieve the functionality category at left in relation to the software listed at the top. *Unavailable* means that no functionality is available beyond the underlying COTS software. *Untested* indicates that a test dataset was not available.

Summary Table

	ArcInfo Desktop 8.1 Beta	**ArcInfo Workstation**	**COTS Map Application**	**GOTS Map Application**
Text (new)	Limited	Extensive	Untested	Moderate
Text (conversion)	Extensive	Extensive	Unavailable	Unavailable
Data Validation	Moderate	Limited	Untested	Unavailable
Data Conversion	Moderate	Extensive	Unavailable	Unavailable
Editing	Moderate	Extensive	Moderate	Extensive
Marginalia	Extensive	Extensive	Moderate	Extensive
Symbol Sets	Extensive	Extensive	Extensive	Extensive
Map Finishing	Extensive	Extensive	Moderate	Extensive

General Conclusions and Observations:

- Any development path would require substantial customization. COTS GIS software is developed for users spanning numerous disciplines and environments. COTS functionality tends towards general solutions, stopping short of specific solutions as required by standards-driven products such as SBS and SEQ maps and GSTC's requirement for high-volume, database-driven production.
- Functionality such as data validation, data conversion, and editing is highly dependent on the product data model.
- Marginalia, fonts, and symbology are highly dependent upon specific product standards. The problem is complicated further by GSTC's need to build an integrated enterprise mapping system supporting two different products through all production phases.
- While there are problems that will require substantial effort to solve, it does not appear that there is any single problem that cannot be solved.
- GOTS and COTS tools offer improved functionality in some areas such as editing, text placement, and marginalia

Appendix B – Configuration and Implementation

Other Configuration Risks

In addition to client-server design risks, the GSTC Team identified infrastructure characteristics that might introduce additional risks:

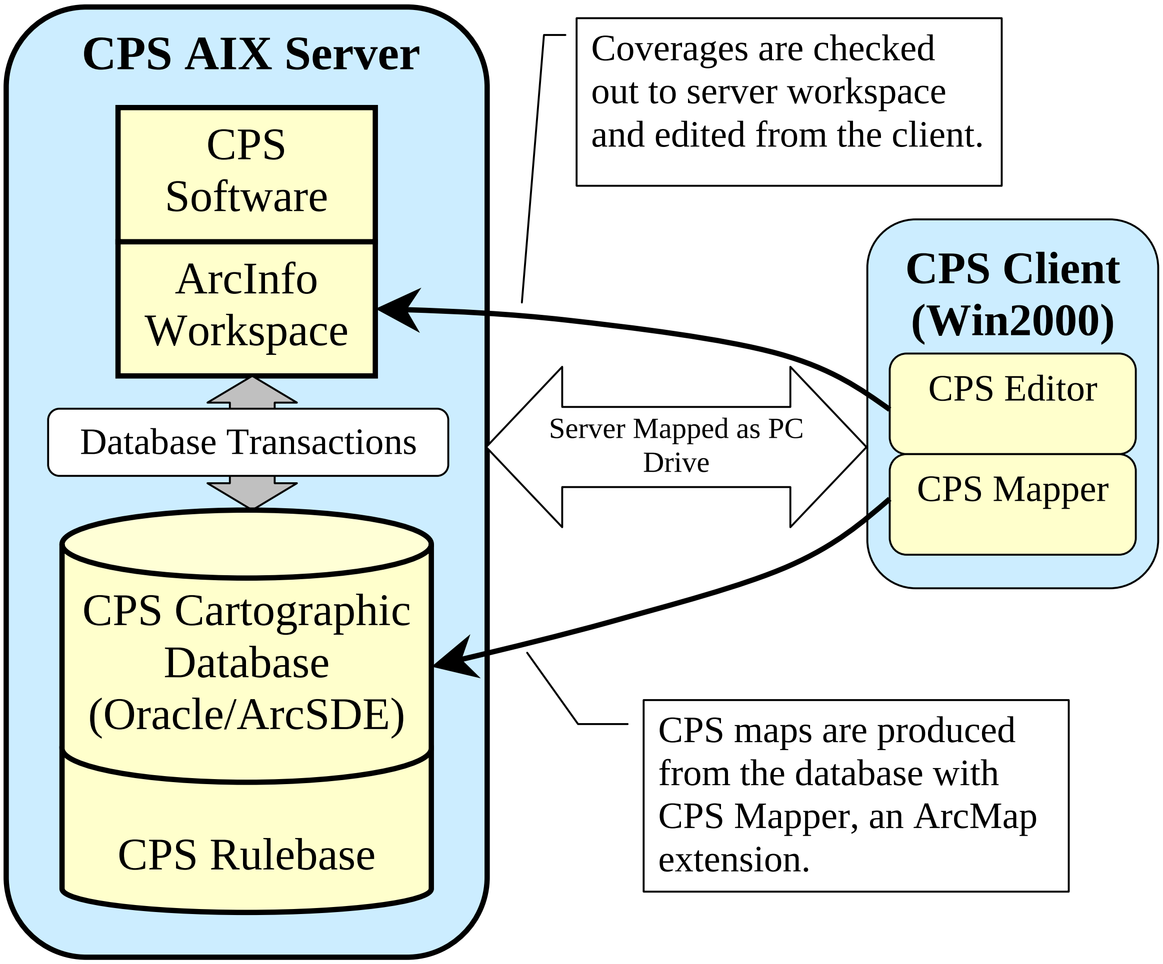
- The project required that a new, relatively complex production system be integrated into GSTC's established corporate environment, without disrupting enterprise users.
- Corporate system administration protocols departed from industry standards and had created configuration problems in the past.
- Forest Service applications (including ArcGIS) are integrated into the corporate image and are not exactly the same as their true COTS counterparts. The ArcGIS software used in development at T-Kartor would be different from the ArcGIS software at GSTC.
- While T-Kartor attempted to simulate GSTC's production environment, it was impossible to match it exactly.
- GSTC's network had known performance deficiencies.
- Other unforeseeable problems were likely.

Problems Encountered and Resolved

Our concerns with the GSTC IT infrastructure proved to be well-founded. Following is a list of problems that were actually encountered and how they were resolved. It is reasonable to believe that aggressive risk mitigation measures prevented performance problems that we didn't have to deal with directly.

- CPS testers working in the SBS production area experienced performance problems that were consistent with the known limits of reliable data transfer with the network hardware in place at the time. After consulting with the GSTC Team, the network administrator reprioritized the network upgrade plan so that the SBS production area was updated sooner. The performance problems were resolved when the network was upgraded.
- A variety of permission problems hampered testing efforts relentlessly in the early development stages. The project team worked closely with the IT staff and over time, most problems were resolved. These types of problems still occur randomly but can generally be resolved with simple work-arounds. Continued collaboration with the IT staff will be necessary.
- A key element of the client-server configuration was implemented by mapping the CPS server as a local drive on the client. After a period of persistent problems experienced at the client, the disk mapping technology became suspect. The system manager replaced the corporate disk mapping software with another and the problems ended immediately.
- In early tests, the CPS Rulebase, a Microsoft Access database used for rules management, was staged on a Windows 2000 platform. This configuration had performed well at the contractor site but for unknown reasons was highly unstable at GSTC. T-Kartor restaged the CPS Rulebase on GSTC's AIX server and stability was achieved.

CPS Configuration



High-Level Workflow

1. Cartographic layers are checked out from the CPS database to server workspace.
2. Cartographic layers are updated with revision edits.
3. Fully revised layers are checked into the CPS database.
4. Map products are generated from the CPS database by client users.

Note: The client connection to the CPS/ArcSDE database is through TCP/IP. All other client-server communications are supported by mapping the server as a client drive.

Author Information

Byron W. Taylor
Development Unit Leader
USDA Forest Service
Geospatial Service & Technology Center
Salt Lake City, Utah
Phone: 801-975-3447
E-Mail: btaylor01@fs.fed.us