Abstract

In providing optimal service to customers, Information Technology departments are faced with many challenges. One of the biggest is making the transition to a distributed applications environment with minimum impact on current structure and plans. With support from key areas, North Carolina State University has taken an innovative approach to moving into the Client/Server arena using Rapid Application Development methodology and a dedicated work team. We were able to build and maintain momentum installing Client/Server applications, while continuing to maintain legacy systems and respond to existing customer requests. This paper will demonstrate a proven, successful path to the understanding and implementation of developing technology.
**Introduction**

One of the greatest challenges facing IT professionals today is how to implement Client/Server applications and maintain momentum in the Client/Server arena without negatively impacting the application development staff’s ability to stay on top of an overwhelming backlog of work. Failure to take advantage of new technology is an unacceptable alternative.

With the support of the University Administration and a few key customers, Administrative Computing Services at North Carolina State University was able to take an innovative approach to solving the problem. By creating a new application development team, operating as a self-directed work team, and employing major components of Rapid Application Development (RAD) methodology, we have quickly and successfully moved into the Client/Server arena. We also have been able to maintain momentum in this environment and to explore and implement new technology as it becomes available.

Our work team completed one cycle which resulted in the production installation of legacy data extract databases available to Campus for ad-hoc use and three client/server applications serving a varying number of customers. In addition, we charged a Continuous Quality Improvement (CQI) team to explore the implementation of Object Oriented technology and methods. We have now begun the second cycle with the new development team. The new charge shifts focus from medium/long-term projects to short-term projects with a high impact on improving processing efficiency within Campus departments. This shift in focus was, in part, brought about by budget cuts and staff shortages. Long range plans will include the development of Executive Information Systems using Object Oriented technology and methods. We are committed to continued advancement.

**Developing the Plan**

North Carolina State University had a backlog of requests from the customer community and a small IS staff to respond to those requests. Our customers were becoming used to working in a Windows environment and expected their new applications to use GUI standards and “leading-edge” tools. Our customers demanded easier, less restrictive access to their data. Faced with the restraints of the existing backlog, the maintenance of legacy systems, and previously committed resources, we sought innovation solutions. To address these changes, we needed to assess the feasibility of changing the way we have traditionally operated without losing sight of the need to stay responsive in the existing environment.

Based on customer demands for high-quality applications, more user-friendly interfaces, and quicker turnaround, as well as a desire to follow new industry trends, we determined that Client/Server applications were the answer to our needs. A pilot team composed of application developers and systems support personnel was formed to evaluate hardware and software needed to operate in the Client/Server environment. Over the period of a year, the pilot team, working with a small group of friendly, progressive customers, developed several small applications using a variety of tools. At the end of this evaluation period, the decision was made to use Sybase as a database engine and PowerBuilder as the primary development tool. We began researching alternatives to allow us to maintain the status quo while we moved forward into the Client/Server arena. The plan was to:

* Identify the developers to do the work
* Choose an efficient development methodology
* Expand the existing toolset
* Identify projects that were candidates for this environment
* Sell the plan to the customers
Implementing The Plan
We soon recognized that we could not respond as quickly as necessary working within the framework of the existing application development teams. A result of the work done by the pilot team was the implementation of three Client/Server applications in three different customer areas. These applications were used to sell University Administrators and Campus customers on the benefits of moving into the Client/Server arena. Management became convinced Client/Server development was the direction we needed to go and committed to support the effort. Five new positions were created and assigned to the existing application development units. The new positions allowed us to establish a new applications development team. The team, called the Distributed Applications Resource Team (DART), developed applications and provided other customer resources using emerging technology. Each of the existing development units assigned one staff member to the DART team. The team members were rotated annually, providing a means for all staff members to participate and be trained in the new environment. The first group included one senior project leader, two senior programmer analysts, and two junior programmer analysts.

As we read about and discussed the experiences of other institutions and organizations and evaluated development methodologies, we concluded Rapid Application Development Methodology (RAD) had evolved into the methodology of choice for Client/Server development. RAD offered the flexibility needed to develop applications of varied size and complexity and was conducive to mixing with more traditional methods. RAD also promoted increased customer involvement in the process and demonstrated to the customer, first hand, the commitments they would need to make to move successfully into the new environment.

Our internal management decided that the DART team would operate as a self-directed work team. The team was staffed with developers with varied skill levels and experience who were participating as volunteers. By functioning without a supervisor, all DART team members would share equally in the success or failure of the project. It was necessary for team members to evaluate new concepts and work toward meeting team, rather than individual goals. Main criteria for choosing projects for the team were:

* The task could be completed in a relatively short time
* Would have high visibility when implemented
* Would alleviate some of the stress on the existing development units.

Executing The Plan
All members of the development units within the organization had received training in Sybase, SQL, and PowerBuilder. All members of the initial DART team had participated on one of the pilot teams and had some experience in relational databases and Client/Server development. To allow the DART team to become comfortable working in a new environment, the team’s first assignment was to continue developing the legacy system extract databases that were begun by the pilot team. This allowed the DART team to get to know each other and to acclimate themselves to the self-directed team concept. This process also assisted us in meeting the goals of the organization by providing easy access to legacy data for the customers which, in turn, alleviated some of the workload from the existing development units. Once extract databases were established for financial, purchasing, student, and vendor data and the Campus was provided with a user-friendly inquiry tool, the customers were no longer dependent on IS support to access their data. This reduced the number of requests to the existing development teams. The extract databases were also highly visible in most all areas of the campus community.
A division-wide application to track staff training was selected as the first Client/Server software package to be developed by the DART team using RAD methodology. The team began the process working with requirements from the primary customer. From the initial requirements document, we laid out the first draft of the database design and developed a high-level prototype for the main functions of the application. From this point through production implementation of the application, the package was developed using RAD methodology. Joint Application Development (JAD) sessions were held to allow the developers to work through the prototype and to interact with customer representatives. JAD sessions allowed developers to assess the customer’s reaction to application elements and to change the prototype, as needed. Important knowledge about their preferences for the look and feel of the model was gained. The customers were able to see what the application was going to look like and how it would work before a single line of code was written.

After the first two JAD sessions, components of the package were assigned within the DART team and coding began. As programming of the major functions of each module was completed, the module was made available to the customer for testing. The customers and developers worked in a mode of iterative project development, one module at a time, and each module was installed into production following completion of testing by the customer. The application had been broken into logical pieces and each piece was delivered upon completion. This process let us incrementally install the Application Modules and let the customer work with sections of the package as they came available. A key benefit of the process was that the customer could initially populate the database through the application which provided part of the training necessary for their staff. The end results of following this process were:

* The customer began using the application much earlier in the development process than was normal with traditional methodologies
* The customer was actively involved in the development throughout the entire process
* The deliverables required much less change than normal
* The application was what the customer needed and wanted
* Delivery was made within a shorter time frame than we felt would have been the case using a traditional approach

This initial experience with RAD methodology reinforced our decision that RAD was appropriate for developing Client/Server applications. The process took longer than we had expected, but we felt the quality of the resulting product was significantly better than using traditional methods and that time was saved by not having to revisit code and enhance the product to what the customer really wanted.

Iterative prototyping was very helpful in giving the team a true feel for what the customer liked and for identifying needs and expectations. We felt that the process provided the customer with a “feel” for the new technology, and it exposed them to new terminology. The process enabled us to make changes easily, encouraged brainstorming, and uncovered elements of the application that might have been missed. The customer was much more involved in the design process, reactions to the application were easier to gauge, and pitfalls of the design were exposed very early in the process.

The JAD sessions were invaluable. They required a commitment from the customer to make available the resources necessary for the project to be successful. Consequently the DART team did not have to wait for customer availability to move forward with the project. To use JAD successfully, the right people must be involved. Decision makers, as well as people who understand and use the process the application is accommodating were required. We found the sessions to be much more successful when attainable goals were set before each one. Our experience showed it to be important for the developers to know the requirements and have a prototype before the first session. It was important that roles be assigned to each developer for each session so that everything that occurred was documented and nothing was missed. We learned to listen to everything that was said by each person involved.

Iterative project development and incremental installation of the application provided many plusses to the process. One of the biggest benefits was that this technique allowed for early discovery of problems. The DART team had the luxury of being able to deal with problems at a more leisurely pace than if many problems surfaced at once during system testing of an entire application. We were able to make logical adjustments to the design instead of applying stop-gap measures. This development method
kept the customer actively involved throughout the process and provided working modules of the application as the project progressed. This involvement boosted staff morale within the customer environment and assisted in gaining customer confidence early in the process. It helped to eliminate surprises at final installation, provided a built-in method to expose the customer to the client/server environment, and resulted in a higher quality application.

The increased involvement of the customer in the development process when using RAD methodology was also valuable from the developer’s standpoint. Even reluctant customers were led into a hands-on involvement in the project from start to finish. Iterative development techniques provide them with a venue for discovering design problems, omissions, and mistakes very early in the cycle. They have more of a feeling of being in the driver’s seat and have a sense of accomplishment and ownership of the application, take pride in the final product and feel a need to share “their” application with their contemporaries.

**The Learning Experience**

Following this process was a real learning experience for the developers and the customers. It gave us a good insight for developing with RAD, operating with a self-directed work team, and keeping pace within the framework of developing technology.

RAD does not allow for skimping in the design process. Detailed attention to the database design process is essential. The database will be the foundation from which a good application can be developed. It is critical to have a prototype available for the first JAD session. The prototype is the tool that gets the customer’s attention and motivates their involvement. The development team should set reasonable goals for each JAD session so that the team and the customers leave the sessions with a feeling of success and accomplishment. Roles should be assigned to each development team member for each session so that nothing is missed. Examples are:

* One person to facilitate the session
* One to track requirement and database changes
* One to operate and change the prototype
* One to keep minutes

The DART team used laptops during sessions to perform their assignments. Roles should be rotated at each session. It is important that the right application and customer for the first project. The customer needs to understand the commitment they have to make for the process to be successful. You must take care not to create unrealistic expectations of the time frame of the project.

One of the most important factors in ensuring the success of a self-directed work team is to choose the right people to serve on the team. Members must be open to new ideas, be willing to cooperate in the self-directed environment, and be willing to commit to the concept. One of the members should be appointed to the role of integrator. This role can be rotated to alleviate the feeling that the integrator is a supervisor. The integrator’s role is to:

* Ensure adherence to standards
* Coordinate the pulling together of application modules
* Oversee system testing
* Serve as the customer contact
* Direct the flow of information within the project

Client/Server technology is evolving continually. It is important to use CASE tools when they are available to facilitate and help to document the development of the application. The project team should continue to explore new tools for design, development, creating on-line help, testing, and documentation. You cannot be afraid to experiment and you should never tell a customer that a request cannot be fulfilled without first exploring for a way to accomplish what they need.

**Continuing The Process**

The process we at North Carolina State University have followed to move into the Client/Server arena has been very successful for us. One of the key benefits has been that management has had the flexibility to change direction when necessary. We can adapt to shifts in focus brought about by
emerging technology or customer desires without negatively impacting progress. The second generation of the DART team started out focusing on learning about data warehousing and moving towards developing a data warehouse for the University. In response to the impact of legislative budget cuts and mandates to downsize throughout the campus and to the desire of the University Financial Officers to improve processing efficiency within departments, the focus of the team shifted to taking on short term projects to achieve more efficiency. An emphasis has been placed on publicizing campus wide applications as well as departmental applications that may be sharable. Campus departments are helping by funding both contract and student staff on a temporary basis to supplement the development team’s productivity. At this point, the DART team is a dynamic, flexible group with no long term commitments. In addition, there is a group operating as a Continuing Quality Improvement (CQI) team to explore and evaluate new software, hardware, and ideas. The CQI team operates under the direction of the Administrative Computing Services Research and Development unit. Members consist of staff from all areas of Business and Administrative Systems Division. They are currently evaluating Object Libraries, Object Oriented development tools, and will be recommending a future direction for applications development at North Carolina State University.

We feel that we are on the right track for providing our customers with “Leading Edge” solutions to their problems. We have been able to assist other institutions in moving into the Client/Server arena and consider ourselves to be progressive in this area. The process we have followed to move in a new direction in applications development is one that has been highly successful for us and one that we are proud to share with our peers in hopes that we can help others to be equally as successful.