

Reengineering CTEPESYS: An Ongoing Evolution

Steven Friedman, MHSA¹, George Redmond, M.Sc, MBA¹, Shanda Finnigan, RN, BSN, CCRC¹, Jeff Abrams, MD¹, Mehul Shah, Executive Vice President & COO, CTIS, Inc.

¹Cancer Therapy Evaluation Program, NIH, Rockville MD;

²Capital Technology Information Services, Inc. Rockville, MD

Abstract

The Cancer Therapy Evaluation Program (CTEP) has developed an enterprise system designed to enhance the scientific and administrative aspects of cancer clinical trial development. Faced with a maturing enterprise system, CTEP is migrating to a Service Oriented Architecture which will reduce long term maintenance cost without sacrificing flexibility, and will enhance the continuous improvement of the clinical trial development process by converging with the Cancer Biomedical Informatics GRID (caBIG). This paper will outline the reengineering process CTEP is using to move towards an SOA for its enterprise system.

Introduction

The Cancer Therapy Evaluation Program (CTEP) provides oversight of research conducted by eleven cooperative groups investigating new distinctive and effective anticancer agents, radiation treatments, and surgical methods. Throughout its history, CTEP has continually invested in informatics technology to suite of CTEP informatics applications is called the CTEP Enterprise System or CTEP-ESYS. Indeed, the productivity gains and consequent returns on the investments could serve as a benchmark for most public and commercial enterprises. Nonetheless, new technology brings new capabilities: CTEP is embarking on a fundamentally new program to leverage its existing technology base by exploiting best components developed to date as a base for new development. Analyses to date indicate this approach simultaneously reduces the ongoing cost of operations, frees capital for new development, and reduces the effort, cost and time of new development.

Accomplishments in Improved Productivity

During the period from 2000 to 2005, CTEP tracked the performance indicators shown below (Table 1).

The Office of Management and Budget reported that this advance in performance yielded a 200% to 300% return on

investment – in 2006, the highest ROI of any major IT investment in the Department of Health and Human Services (HHS)².

Performance Indicator	2000	2005	Change
NDA's submitted	4	8	100%
IND's submitted	120	162	35%
Drug agents (cumulative)	781	1,160	55%
Patients Accrued (cumulative)	335001	681483	133%
Drug orders created (in year)	31713	50837	60%
Treatment courses collected (in year)	8699	22,951	160%
Active Sites	4859	6960	43%
Active Investigators	8697	13,412	35%
Processing time: Protocol Revisions	31 days	21 days	-32%
AE's processed per staff FTE	424 (in 2001)	959	126%
Clinical data upload time	45 days	8 days	-45%

Table 1. CTEP Key Performance Indicators.

Each of these reported gains and improvements are quite dramatic. In our general experience and in our work in reengineering at CTEP, such improvements cannot be achieved through technology alone. Transformation of the business processes and the roles and responsibilities of the people involved is also needed. In fact, we have found this three-part combination (people, process and technology) to be essential to substantial transformation of any enterprise.

Although dramatically improved, the indicators show a focus on the volume of work being done and how to do more. Recent work¹ by David Dilts, PhD highlights the complementary advantages of being more selective about what clinical research work to do. Dilts' work suggests that CTEP is now in a position to accomplish an even better job of determining what clinical trials to approve, activate, and execute to completion given the richly detailed data generated from the informatics system now facilitating program administration.

A focus on both (how to do trials and what trials to do) is needed to truly optimize the return from investment in cancer research. As we will describe in later sections, some of the systems for the former task are prerequisite to the latter task.

Since 2005, CTEP has continued to develop tools that will further streamline workflow. The recently developed Docu-MART application is a case in point. This application will make it easier and faster to generate LOIs, concepts and protocols. Docu-MART also handles the workflow and tracking for protocols. CTEP is promoting its use as a means to expedite review and approval of high quality clinical trials without proportional staffing increases.

It is perhaps even more important to make sure that the most promising clinical studies are started and managed through this efficient trial lifecycle. Docu-MART will provide a consistent and comparable representation of LOIs, concepts and protocols, and will help with status and tracking. It provides reviewers with facilities for online real-time review and commentary. The net productivity gains free up assigned reviewers' time to concentrate on scientific analyses and evaluations. Additionally, with data from the workflow facilitation systems, data about the studies, more integrated access to external reference materials, we can do a much better job of supporting the scientific work – one crucial focal area for the next phase of development work.

Next, we will review the application suite and technologies in use at CTEP and some of the history of its construction. This will position further elaboration of CTEP's new application architecture, one that makes optimal use of the best technology developed to date and exploits the most recent technical developments in information systems.

The Current Application Suite and Architecture

CTEP's current suite consists of twenty-two applications providing a broad variety of functions spanning the full lifecycle of a clinical trial including: planning, activation, conduct, and reporting. These applications are used by CTEP staff and collaborators in the working groups and clinical research sites. Figure 1 below shows CTEP's applications distributed by usage across these four phases.

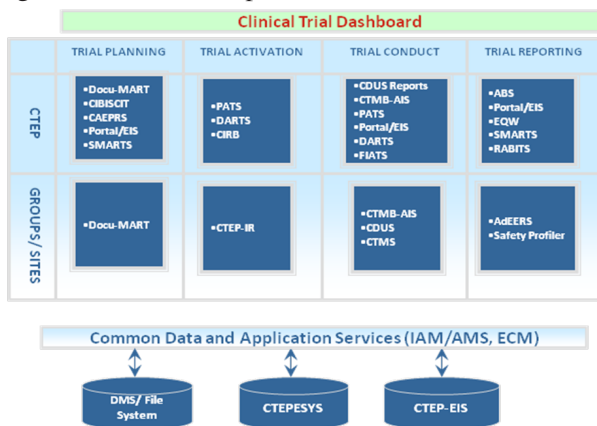


Figure 1. CTEP's applications mapped to clinical trial lifecycle

Each application averages roughly 105K lines of code in the application tier and over 250K lines of code for application logic. In total, there are nearly five million lines of code in the suite. Figure 2 below shows that roughly two million lines of code comprise the data tier with over half of that code concentrated in CTMBAISⁱ. The other segments comprise various application specific data access functions.

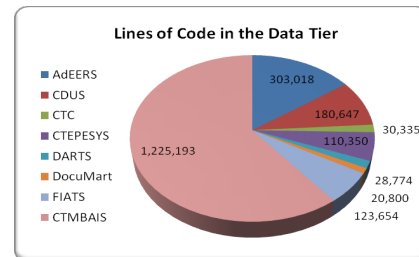


Figure 2. Code distribution in the data tier

The second largest of these is AdEERSⁱⁱ. The AdEERS data layer supports the reporting and analyses of adverse events. This data interface has recently been expanded to support the new CBIITⁱⁱⁱ developed application for adverse event reporting – caAERS^{iv}. It is being expanded to support cancer-imaging data through a set of web services allowing for future support of cancer imaging data exchange from caAERS.

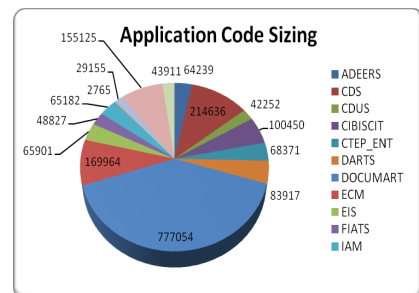


Figure 3. Code distribution in the application tier

Figure 3 above shows the distribution of code supporting application logic for some of the larger applications.

The applications were developed over the past decade. Like many organizations, CTEP employed the best available technology at the time each new set of applications was commissioned.

ⁱ CTMBAIS is the Clinical Trials Monitoring Branch Audit Information System

ⁱⁱ AdEERS is the Adverse Event Expedited Reporting System

caAERS is the cancer Adverse Event Reporting System

ⁱⁱⁱ CBIIT is the Center for Biomedical Informatics and Information Technology
^{iv} The Cancer Adverse Event Reporting System (caAERS) is an open source software tool that is used to collect, process and report adverse events.

^v These have been designed to be consistent with CBIIT's caBIG architecture.

The suite employs five distinct architectures reflecting this development heritage.

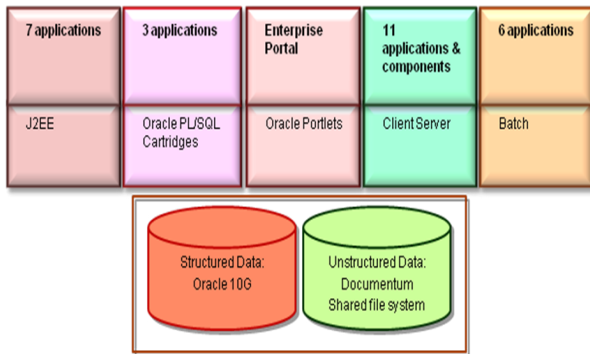


Figure 4. CTEP's application architectures

As the business benefits described in the preceding section show, this proved to be a tremendously effective strategy allowing rapid deployment of new function and rapid realization of improved efficiencies and throughput.

Having achieved all the benefits from automating the basic tasks, CTEP can now focus on integrating the components to improve data exchanges and workflows across trial phases and support better collaboration within phases. During the time this suite was developed, integration of the various application components would have required custom, one-off development. As we will see in the next section, new technologies have been developed in the IT industry that dramatically reduce the integration effort and makes it feasible, even economically advantageous, to preserve substantial portions of the existing code as a basis for the future. The value of this new approach is quite significant. We estimate the cost of rebuilding this code base from scratch to be over \$50M including the development time, defect correction, and deployment. The effort would take years as did the original development effort, and the result would be functionally equivalent albeit on a common platform. We believe that is far more cost effective and far less risky to extract the common components and concentrate on new function and integration. Our initial planning estimates show that we can deliver new function faster this way too. A services oriented architecture makes this possible. We discuss that next.

Services Oriented Architecture for CTEP

A services oriented architecture is a type of software architecture that is based on discrete software provided services. The services are registered in a directory and all aspects of using the service (inputs, what functions the service performs, and the outputs) are described in a standard format. An example is the caAERS-AdEERS/ABS web service. The AdEERS application consists of two components: a data-gathering database for reporting, and analyses and tracking called ABS.

The newly created AdEER/ABS service allows the caAERS application to send adverse event (AE) information to the AdEERS/ABS database. The service receives AE information and processes it according to a set of business rules. If the information conforms to the defined rules, AdEERS/ABS accepts the AE into database. Otherwise, it notifies the service requester of an error.

Note that the actual code doing the work is the same code that has supported the AdEERS data gathering application. Only the interface has been extended to allow for caAERS to use this code in addition to the AdEERS reporting application.

In Services Oriented Architectures (SOA), the service interfaces are standardized, fully described, and registered in a directory. Even the location of the service (similar to a web address) is included in the registration. This greatly eases the burden on a programmer wishing to reuse existing application logic. He or she generally can do so just by finding the service needed in the registry and using the registry information to locate and make use of it in his or her new application.

Notice that the user and author of the service do not need to work directly together. This makes it possible for the service author to simply extend existing application code with a SOA style standardized interface – a task generally far simpler than rewriting the code from scratch. Moreover, the existing code such as the AdEERS/ABS which has been used in production for many years is generally far more reliable than brand new software would be.

While the services are quite useful, there are additional elements of a Services Oriented Architecture that make it really powerful. A very prominent feature of any SOA is the enterprise service bus (ESB). The ESB provides a means for services to convey input and outputs to each other. It also facilitates and monitors service invocations. This allows it to keep track of all of the calls and responses among services, recording statistics on the number and timing of such events. Typically, this information can be used to track the key performance indicators (work getting done, how fast, by whom, etc) of the system and these in turn usually provide some strong indication of performance of the enterprise and its utilization. The ESB would also facilitate the integration of components from CBIIT's Clinical Trial suite. This is another feature of a SOA – easy adoption of alternative technologies. Last, the ESB usually provides a framework for describing and executing workflows, which in turn invoke the system's services. By using the ESB for workflow, an enterprise can extend or modify its workflows and processes more easily than would be possible if the workflow was embedded within its applications.

Fully utilizing SOA in CTEP

This new technology presents an important and valuable opportunity to CTEP.

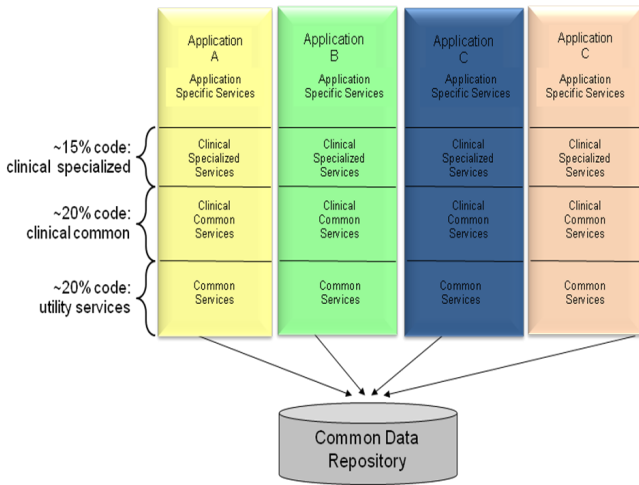


Figure 4. Finding CTEP's best technology

We have completed an assessment of the existing code base. We have found that roughly 20% of each application comprises utility services common among the applications and found many application suites of this scale and complexity. A second subset, similar in size comprises function more oriented to clinical research but nonetheless common across the suite. These sets of nearly redundant services can be replaced by a single set of service components made of the best of the existing software parts. That is, we will keep the best technology, provide it with a proper service interface, and reuse it with the unique parts of each application. Figure 5 shows the results in principle.

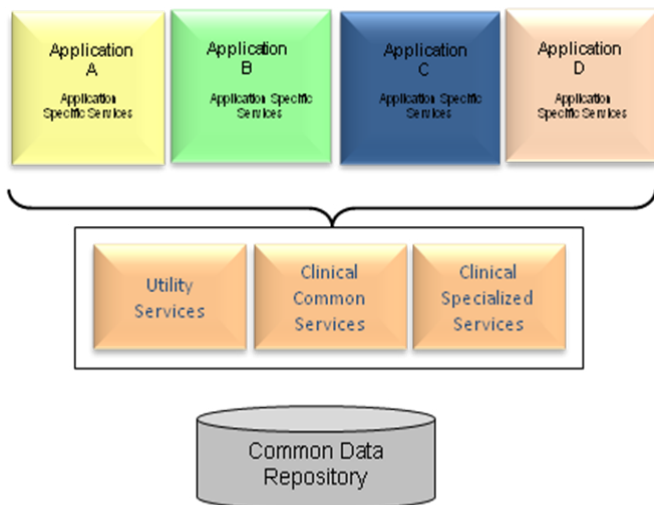


Figure 5. Using CTEP's best technology

Our work shows three advantages from leveraging the best technology across the 11 most important applications in CTEP's suite:

- **Reduces overall code base by over 40%:**

This allows greater focus on the stability and reliability of the common components.

- **Allows construction of new applications faster:**

We will only build truly unique software from scratch and reuse the existing whenever possible. Our initial studies and shows that new applications will take approximately half the time to build.

- **Reduces maintenance costs by over 2X:**

Two causes are active in reducing costs of maintenance. First there is less code to maintain. Second correcting a defect in the common components corrects that defect for all applications that use that common component.

Clearly, these advantages are compelling. Our work to date shows that these benefits are quite feasible and achievable with a consequent effect that more resources can be devoted to new projects. Our 2008 development plan has taken advantage of this reallocation of resource enabling the development of more application more quickly with fewer resources. In the next section, we discuss what new types of new projects would provide the most impact.

Future Development

In general, CTEP has taken a practical approach to selecting new informatics applications. That approach has established an informatics foundation piece by piece – each providing a focused set of benefits to the enterprise. The next phase of development can exploit this data rich foundation and add new data sources to facilitate faster and better decision-making. We can also employ readily available SOA technologies to more effectively integrate data sources and to facilitate collaboration and workflow.

Decision Making

As shown in Figure 1, the suite includes a clinical trial dashboard. This application is implemented as a web portal and provides a set of common reports along with a facility for users to compose additional queries and reports. As we have considered the suggestions for reports and decision models, it is clear that CTEP already has most if not all of the data it needs. However, the decision makers would be able to make better decisions if the data were presented more effectively. For example, borrowing a technique often used in portfolio analyses we can summarize the attributes of a proposed or in progress protocol under a set of (nearly) orthogonal measures. Using these summarized measures, we can then distribute protocols within 2-axis grid. Dilts has suggested using the scientific merit of projects as one axis and the operational complexity as another.

These measures would summarize a set of attributes of the various protocols for which CTEP already has the data. The resulting graph might look like Figure 6 below.

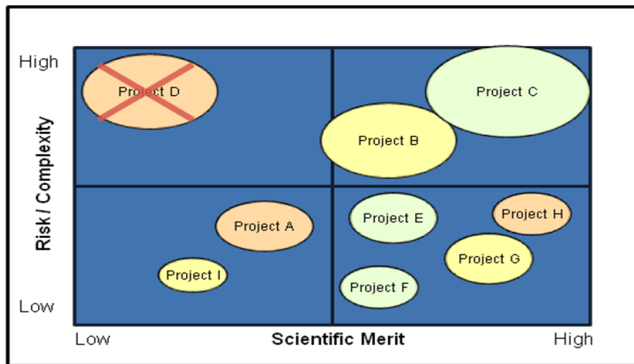


Figure 6. A sample portfolio

In this example, we are representing each project as an ellipse. The merit and complexity of each project is illustrated by its position in the grid. The actual accrual vs. planned by the color (red, yellow, green), and the relative cost by the size of the ellipse. This type of visualization provides a lot of information in a compact form. One can easily identify the high risk, high cost, low accrual, and low merit projects (upper left– Project D) that should potentially be cancelled.

Presenting the characteristics of protocols in this way would help decision makers maintain a balanced clinical research portfolio. For example, a balanced investment portfolio would typically include a subset of projects that are more costly and difficult but have a potentially greater scientific significance. To balance the risk inherent in these projects, the portfolio might also include a subset expected to generate more certain albeit less significant scientific outcomes and at lower cost.

Commercial software and increasingly even open source software can be used to create this example and many other kinds of data visualizations.

Workflow and Collaboration

Referring to Figure 1, we note that the applications used in each phase of the clinical trial are discrete and distinct from one another. Also, the groups/ sites and CTEP use different applications. The applications are engineered to provide effective interchange of data. In some cases the applications are built expressly to gather and submit data to CTEP or to present submitted and aggregated data back to the groups and sites. The data exchanges among applications are in part responsible for the business returns described earlier. Nonetheless, by adopting a Service Oriented Architecture and most notably an enterprise service bus, we will be able to compose more effective intra and inter phase workflows and better intra-phase col-

laborations between CTEP and the cooperative groups/sites.

Data Integration

Last, we note that there is a large number of data exchange interfaces among the branches of CTEP, with other parts of NCI and with the cooperative groups. In general, we find data service interfaces to be a more effective means of data integration than data exchange. Here’s why: data exchanges always introduce time lags. Whenever a source of changing data is copied, the copy must be refreshed on some periodic basis to remain current – it is usually at least a little out of date. In contrast, a data service always accesses the data at its source – the data user gets data that is never out of date. Service interfaces tend to cost less to operate too. This is because storage costs are lower and there is no need to provide administrative services or backups for the copied data. Occasionally, performance and network latency considerations might indicate the need for exchanges but CTEP is investigating the opportunity to replace at least some of the data exchanges with data service interfaces.

Conclusions

CTEP-ESYS has provided a tremendous return on investment to CTEP and to the extended clinical research enterprise helping the enterprise dramatically improve the efficiency of clinical research. Through careful analyses and selection of reusable components and careful reengineering and adoption of SOA technologies, we are further exploiting the CTEP -ESYS foundation to deliver the next wave of benefit – providing greater insights into what research to do.

References

1. David Dilt’s work for CTEP is still pending release. Dr. Dilts has produced a representative prior work for CALGB available at: <http://jco.ascopubs.org/cgi/content/full/24/28/4553#otherarticles>
2. Described in internal letter from Jeffrey H. Weiner, Acting Deputy CIO, NCI to Dr. Michaele Christian Associate Director, CTEP, NCI entitled “NCI CTEP-ESYS Exhibit 300 FY2008 Budget Submission”