

Delivering Clinical Study Data Using SAS/IntrNet

Robert Davis, GlaxoSmithKline, Harlow, UK

Barrie Nelson, GlaxoSmithKline, Harlow, UK

[based on a paper given at PharmaSUG 2000 (Seattle) by

Damian Ciccariello, Phil Giangliulo and Gregory Weber all of GlaxoSmithKline, Collegeville, PA, US]

ABSTRACT

Newly adopted FDA guidelines require clinical study data be delivered as SAS® data files in all electronic submissions. The speed and depth at which Clinical Scientists need to assimilate this data is ever increasing. SmithKline Beecham, utilizing SAS/IntrNet technology, has developed a Patient Data Browser to address this need.

This presentation will describe the Patient Data Browser and how SAS/IntrNet was employed. Topics will include the following:

- Patient Data Browser Overview
- Customer Requirements
- Regulatory Requirements
- Technical Requirements
- Application Security
- Overcoming SAS® 6.12 and SAS/IntrNet 1.2 Limitations
- Functional Overview

INTRODUCTION

This paper will address the following aspects of the Patient Data Browser (PDB) application project: The Business Requirements phase, The Development Process, Technical & Design Considerations, a Functional Overview, and Futures.

BUSINESS REQUIREMENTS

Background

The clinical reporting environment at SmithKline Beecham utilises SAS® software on a SUN® UNIX hardware platform and data is stored on a Clintrial™ / Oracle® database.

Following FDA data submission guidelines SmithKline Beecham have implemented a standard reporting data mart structure [clearly defined and documented SAS® datasets]. The data mart contains both collected and derived data and facilitates querying, reporting and data submission to the FDA. Having standard data structures in place enables tools and applications to be developed, maintained and deployed across the department much more easily.

What follows is the set of customer and technical data browser requirements used to justify the Patient Data Browser initiative.

Customer Requirements (near term)

- Clinical customers required a mechanism to assimilate patient data content, to get a better feel for what to expect from a study, prior to receiving formal SAS® output reports from the Biometrics department.
- Clinical customers required quick and easy access to electronic patient data to identify trends or patterns in data that surfaced from their review of SAS® output reports already provided by the Biometrics department.
- Clinical customers required a more facile and autonomous method of answering follow-up data questions, to reduce their dependency on ad-hoc requests to the Biometrics department.
- Clinical customers required that all browser interfaces into our electronic patient data be very fast and easy to learn and use.

Customer Requirements (longer term)

- An improved definition of up-front formal report requests by Clinical personnel to the Biometrics department, based on a better understanding of the patient data gained from Clinician's use of data browsing tools.
- A reduced number of reports to be designed, produced, and quality checked initially by Biometrics avoiding the tendency of asking for everything up-front, building a paper based ad-hoc query tool, just in case additional information might be needed later on.
- The timeline in meeting submission quality targets to be improved as the increase in speed of response to queries allows more questions to be investigated and factored into the submission.
- Ultimately, some tables and listing to be produced directly by Clinical via the data visualization and reporting tool.

Regulatory Requirements

- FDA Guidance: Computerized Systems Used in Clinical Trials. This guidance addresses how elements of data quality might be satisfied where computerized systems are being used to create, modify, archive, retrieve, or transmit clinical data.
- The data browser must be a validated application.

Technical Requirements

- Low administration maintenance. Functions such as adding drugs, studies, and users must be low-overhead activities.
- Acceptable performance when accessing remote server data over the SB WAN. Clinical customers in the US and UK frequently require access to remote study data.
- Reduce volume of printed output, and avoid tendency to use boxes and boxes of paper output as a database.
- The data browser to fully leverage our newly deployed SAS® processing environment, reporting database, and reporting toolkit.

Easy to use and deploy Thin-Client (WEB Browser) interfaces to the SAS® system had been envisioned as a very good method for widening the audience and use of SAS®, and consequently our SAS® return on investment. However, in the past, this access method could not easily be developed and supported since only very limited enabling software was available. When the SAS/IntrNet common gateway interface, publishing macros, and graphing applets were announced by the Institute the potential to meet previously unsatisfied business requirements appeared to be achievable.

DEVELOPMENT PROCESS

The Patient Data Browser was developed using the Rapid Application Development (RAD) methodology. A prototype and pilot of the application were performed prior to the development of the production version.

Prototype

A prototype was created as a "Proof of Concept" to determine if the SAS/IntrNet product would adequately address our business needs. It was designed based on the requirements gathered in the business analysis. Once developed, the prototype was demonstrated and enthusiastically received by our customers. Key stakeholders from the Business and IT groups agreed that an application utilizing SAS/IntrNet technology could be used to effectively browse clinical data. A pilot of the Patient Data Browser was approved.

Pilot

The objectives for the pilot were to:

- Allow customers to gain experience in browsing clinical data via the web
- Compile customers and developer feedback
- Solidify user requirements and functional specifications
- Enhance understanding of SAS/IntrNet capabilities and limitations.

A significant portion of prototype software was leveraged in the development of the pilot application. Functional enhancements including additional output capabilities were added. Pilot participants were selected from a specific customer group and clinical reporting effort. The participants were trained in the use of the Patient Data Browser and the pilot was initiated.

Overall feedback from the pilot was quite positive, and based on that feedback the decision was made to develop the pilot into a production application. Functional recommendations from the pilot were compiled, prioritized, and included as user requirements for the production version of the Patient Data Browser.

Production

To develop the Patient Data Browser into a fully supported software product, the following steps were taken:

- Functional and technical requirements were formalized
- Formal technical and detail design specifications were compiled
- Pilot software was updated and standardized
- Software for additional functionality was developed
- Integration and user testing was performed and documented
- Training program was developed
- Application was certified as validated

In conjunction with the Patient Data Browser deployment, a change management process was initiated. The process was designed to ensure that the application maintains its validated status. The process addresses issues such as patches (bug fixes), enhancements (new releases), and upgrades to underlying or base applications.

TECHNICAL / DESIGN CONSIDERATIONS

Several technical and design principles were applied throughout the design and development of the Patient Data Browser. The following factors guided the Patient Data Browser's development.

Conceptual Model

Conceptually, the application is based on a hierarchical structure of a drug or compound that consists of a collection of studies. Each study contains a collection of domains or data sets. The Drug / Study / Data Set structure was a key driver in the design of the user interface.

The Patient Data Browser's HTML frames were developed utilizing this hierarchy.

Application performance

The SAS/IntrNet web based, thin client architecture has yielded considerable performance improvements over our previous Server only applications. The new architecture provides acceptable performance over our wide area network.

User Interface 'Look & Feel'

Key concepts employed in the design of the user interface were as follows:

- Ensure context is preserved (Make sure user has Drug / Study / Data Set information available at all times)
- Allow for easy transition from function to function
- Isolate content from application interface

The applications Status, Display, and Tool Bar windows are developed based on these concepts.

Security and Administration

The Patient Data Browser resides on an internal web or Intranet. The application employs SSL based security system – with every click the id and password of the user is verified.

A Drug is made available to the application via a drug registration file. Within a registered drug, a study, including all associated data sets, can be registered to application. The default for a registered drug is to include all studies. Descriptive names for both drugs and studies can be included in the registration data. Users are granted access to individual drugs via a user registration file.

The Patient Data Browser offers a variety of generic and domain specific reports via the DataCeutics®, CR Toolkit™. This same software is used to produce standard safety/population reports in the clinical reporting environment. The ability to enable reports to be run against specific data sets is controlled via a report registration file.

Overcoming SAS® 6.12 and SAS/IntrNet 1.2 Limitations

One challenge we encountered in utilizing SAS/IntrNet to create a dynamic web application was preserving the state of the application. This is not unique to SAS/IntrNet, rather a function of how web technology works. Each click is its own process and when the process is complete the state is lost. Basically, we had to explicitly preserve the state by passing macro variables (via the broker of course) from one SAS invocation to the next. Besides needing to manage the state, there were times when the number of macro variables we needed to pass to the next program was enough that we started running in to the 200 character limit.

The next biggest challenge was working within the maximum variable length of 200 characters. This became an issue in a couple of areas. As stated above, we had to take this into account when using macro variables to preserve the state of the application. Also, in order to create a link from within an HTML table using the DS2HTML macro, the URL created could not exceed 200 characters. Since we needed to preserve the state by passing macro variables, the URLs could easily get quite long.

Another related issue is the accessing temporary data sets between invocations of SAS® programs. Since there is no persistence of the SAS® work environment, temporary data sets must either be recreated or made permanent and managed separately for reference by subsequent programs.

We were able to overcome these obstacles by various techniques, but see these going away when moving to SAS/IntrNet Version 8 which will have the capability of referring back to the state of a previous process (e.g. macro variables, data sets) as well as support for longer character values.

We are currently using launch services as we found this to be a more stable solution in our environment, but will be reviewing this when looking at the Version 8 enhancements.

FUNCTIONAL OVERVIEW

This section describes Patient Data Browser (PDB) application interface, navigation and functions. Those attending the conference presentation will see a live demo.

The main application interface consists of six frames (Drug, Study, Data set, Toolbar, Output, Status Bar). Figure 1 below shows the state of the PDB immediately after the user has logged in. At this point, only the Drug and Output frames are populated. The user's ID determines a list of available drugs. The output frame contains a welcome screen.

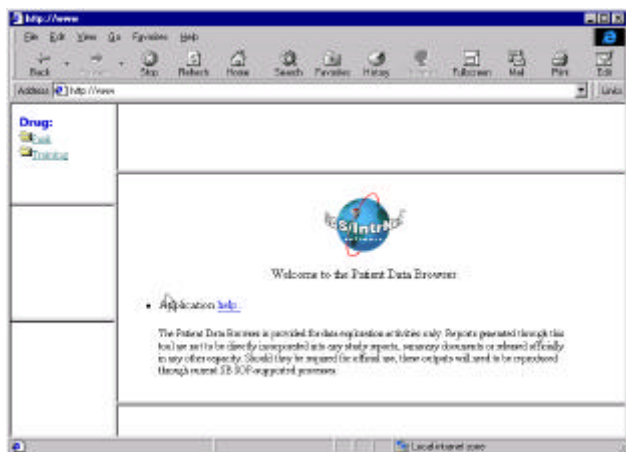


Figure 1 – Application interface as presented to user upon login.

The user selects a Drug project by clicking on the link. The list of available studies appears in the Study frame. The Status Bar (bottom right frame) now indicates the selected drug and the Output frame (largest frame) is cleared (see Figure 2 below).

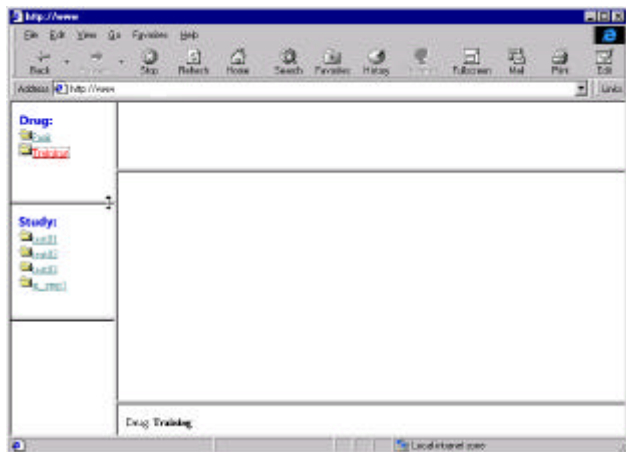


Figure 2 – After choosing Drug Project.

The user then selects a study. The list of Data available for that study appears in the Data set frame (bottom left frame). The Status Bar is updated and now shows both the currently selected Drug and Study (see Figure 3 below).

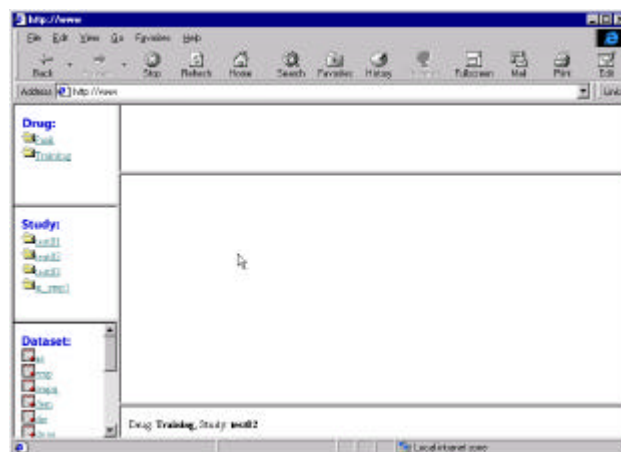


Figure 3 – After choosing a Drug and Study.

The user selects a data set and the Toolbar and Status bar are updated. With the Drug, Study and Data set selected the context is set for this PDB session and tools are now available via the Toolbar (see Figure 4 below). At any time, the user can go back and change the selected Drug, Study and Data set.

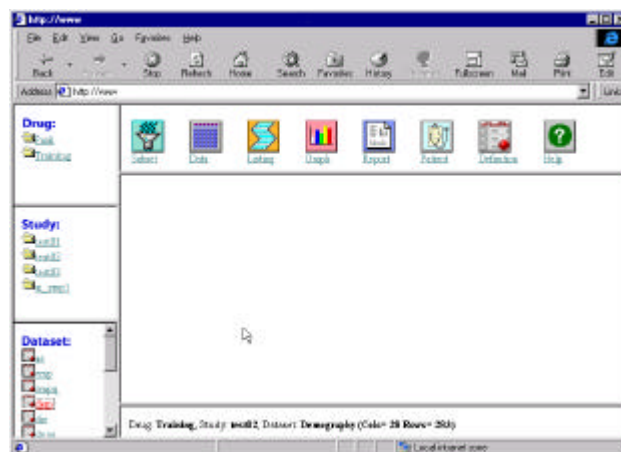


Figure 4 – After choosing a Drug, Study and Data set.

With the application context determined, the Toolbar is populated with different functions (Figure 5 below)



Figure 5 – Close-up of the toolbar.



The Subset button allows the user to subset the currently active data set. For instance, the user may only want to see data on specific patients or for certain adverse experiences.



The data button allows the user to select columns and view the data as either an HTML table or download to MS Excel.



The Listing Button is used to create Data Listings in report format and has options for Column Selection and Sorting and Titles/Footnotes. These reports can be viewed as HTML or PDF



The Graph Subset button provides a front-end for the SAS® Java™ Graph Applet. From a graph, the user can use the drill-down functionality to examine the underlying data.



A set of standard reports is available. The list of reports available is determined by the currently active data set.



The Patient Button launches a Patient Review interface. Through this interface the user is able to browse all data for the list of patients in the currently active data set.



The Definition button displays the Meta data associated with the active data set. Tools are also available to compare Meta data across multiple studies or against a pre-defined specification.



A help facility is available containing FAQ's and HOWTOs that cover use, setup and administration of the Patient Data Browser.

FUTURE

The current version of the PDB was developed with SAS® 6.12 and SAS/IntrNet 1.2. The development team recently upgraded the PDB to SAS® Version 8.1 and SAS/IntrNet (8.1) with relative ease. The ability to refer back to the process ID to recall the state will simplify development. Version 8 support for longer character data also eliminates problems/hurdles associated with the 200 char limit. The ODS system will add flexibility and capability for producing HTML (XML and PDF too, hopefully). Another initiative under way at GlaxoSmithKline is an Automated Authoring Project and as part of that project making links directly into the PDB from another document is being looked into. In this way the PDB will become an integral part of the reporting process.

CONCLUSION

Patient Data Browser has been a successful application in large part due to ease of use. Most users were very familiar with Internet Browsers and quickly learned to navigate the application. The thin-client architecture allowed for rapid development as well as rapid deployment.

SAS/IntrNet, although still maturing as a product, has become a valuable tool for deploying WEB solutions that leverage the SAS® System are easy to use, maintain and administer.

® indicates registration.

ACKNOWLEDGMENTS

SAS and SAS/IntrNet are registered trademarks or trademarks of the SAS Institute Inc. in the USA and other countries.

CR Toolkit is a trademark of DataCeutics Inc.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Robert Davis
GlaxoSmithKline Pharmaceuticals,
New Frontiers Science Park (South),
Third Avenue, Harlow, Essex,
CM19 5AW. UK
Phone: 01279-646813
Fax: 01279-644430
Email: robert_2_davis@sbphrd.com

Barrie Nelson
GlaxoSmithKline Pharmaceuticals,
New Frontiers Science Park (South),
Third Avenue, Harlow, Essex,
CM19 5AW. UK
Phone: 01279-646423
Fax: 01279-644430
Email: barrie_nelson-1@sbphrd.com