

Improve your Clinical Data Management With Online Query Management System

Whitepaper

PharmaSUG conference
May 23rd-May 26th 2010 in Orlando,Florida

Romain Miralles, Clinovo, Sunnyvale, CA
Ale Gicqueau, Clinovo, Sunnyvale, CA

ABSTRACT

Many organizations are still performing paper-based trials. As in any other trials, the data from paper-based studies have to be validated and cleaned. To clean the database, data managers create discrepancies, which are sent to the investigator sites. Tracking and managing these discrepancies for large studies can be a painful task without the right tool.

This paper introduces a SAS®-based innovative query-management solution: Online Query System Management (OQMS). This tool has recently been used by one of the top five medical device companies to solve 8,000 queries in less than a month.

OQMS is a web application developed using SAS/IntrNet®, with a SAS dataset back-end and a user interface. With OQMS, data managers connect to the system, enter their username and password, manage and a study's discrepancies using a user-friendly interface. The application allows multiple users to connect simultaneously and securely while working on the study queries.

Since the back-end and user interface are both developed in SAS, additional SAS-based modules can be added to the system. For instance, users can view patient profiles by clicking on the patient number, access the audit trails to see the history of each discrepancy, as well as export the data in Excel format. Furthermore, many trial-specific reports can be easily programmed and added to the interface.

There are several aspects in which this paper will benefit data managers:

- To understand the core functionality of stand-alone query-management systems.
- To identify the business situations in which such systems can be valuable.
- To recognize where web applications are superior to traditional client/server applications.
- To realize how and why SAS/IntrNet is an excellent platform for rapid web custom development in clinical trials.

INTRODUCTION

Clinical trials are conducted to collect and analyze data regarding the safety and efficacy of new drugs and devices. That said, accurate results for any clinical trial are determined by the quality of the collected data. However, data collected from the investigator sites are never completely consistent, requiring sponsor companies to validate the data in order to get a true representation of what took place in the trials.

In general, the data management group or the data managers are responsible for checking discrepancies and getting the data cleaned up. They identify discrepancies using edit checks, which are small programs designed to find inconsistencies in the database. The number of edit checks for a clinical trial generally ranges from ten to hundreds.

Once the discrepancies have been identified by the edit checks, they are stored in a system until they can be resolved. Some of these can be resolved by the clinical team, but many are sent to investigators for resolution. Discrepancies that are sent to investigators are called “queries”.

One of the biggest challenges for data managers is to generate, track and manage these queries.

QUERY MANAGEMENT IN CLINICAL TRIALS

EDC AND QUERY MANAGEMENT

EDC systems include a tool that provides a simple way for generating and resolving study queries. The entire query management process is handled within the EDC system. With this system, users can easily create a query for a patient and send it to the investigator via the EDC interface. The EDC system will then track the query and record any change.

While commercial EDC systems are very effective in handling queries, most are very expensive. They are convenient for large companies with substantial budgets, but small and medium companies, as well as academic groups, often lack the funds to purchase such licenses. Moreover, implementing a complex trial using an EDC system can be very challenging. These reasons explain why many organizations are still running paper-based trials.

QUERY MANAGEMENT FOR PAPER-BASED STUDIES

Data from paper-based trials are cleaned up by sending queries to the investigator. However, unlike in EDC-based studies, the query management process is not handled within a system. In paper-based studies, queries are generated by completing a query form that is sent to the investigator for resolution by email, fax or mail. The answers are then passed on to the sponsor.

Tracking and managing these queries can be difficult without the right tool. While some companies use spreadsheets for managing their queries, and while Microsoft Excel VBA applications may be quite sophisticated, they come with many drawbacks: maintenance issues, lack of authentication and audit trail, single-user, formatting constraints.

WHEN TO USE STAND-ALONE QUERY-MANAGEMENT SYSTEMS

Tracking queries manually with a spreadsheet can quickly become a nightmare. Information can be deleted by mistake, wrong data can be entered, or the document can become corrupted. Additionally, access to spreadsheet files are usually limited to one single user at a time.

Stand-alone query-management systems offer an efficient solution for query managing. Queries are automatically generated by edit checks, all discrepancies are located in a secured system, and multiple users can work in collaboration. It offers users the capability to communicate reviewed queries to the investigator and process the answers.

Stand-alone query-management systems provide great benefits for all studies in which discrepancies are not monitored by the data management software.

HIGH-LEVEL USER REQUIREMENTS

OQMS goals are multifold:

- Save time to data managers when tracking and processing queries, via an intuitive user interface
- Deliver a reliable application that will demonstrate strong performance and can handle many users without data conflict or corruption
- Capture any data update in the system for compliance and security reasons using an audit trail
- Provide an extensive framework to make it easy to include additional SAS modules (e.g. reporting)

OQMS: A USER CASE

OQMS was developed with SAS and SAS/IntrNet for the needs of a clinical trial sponsored by a medical device company.

The study was outsourced to an academic group that was in charge of the data cleaning. However, the sponsor company became aware that there were still a lot of issues in the database. After investigating the issues further, they found that the academic group had not performed the data cleaning to the level required by the industry regulations.

Because they needed to improve the quality of the data for the analysis, they decided to perform additional in-house data cleaning. They extracted SAS data sets from the clinical database and developed around 100 edit checks through SAS with the initial idea to output any discrepancy to an Excel spreadsheet. However, the edit checks found over 8,000 discrepancies, quickly leading them to realize that using an Excel spreadsheet to track and manage queries was not reasonable, and that they needed to find another solution.

They had a few other requirements:

- The query management system had to be tailored to the study since the trial had been running for several years, the CRF had been designed, and the data had already been collected.
- The development had to be fast because of the study's tight deadlines.
- OQMS had to enable users to access the application simultaneously (10 or more) so that a team of data managers could work together to process over 8,000 discrepancies under very short deadlines.

SAS/IntrNet was the ideal candidate for the development of the required application for different reasons:

- It allows the rapid development of complex and well-designed web applications.
- Because the clinical staff had good SAS programming skills, it was easy to maintain on the long-run.
- The edit checks were already programmed in SAS and the code could be used to implement the back-end of the system.

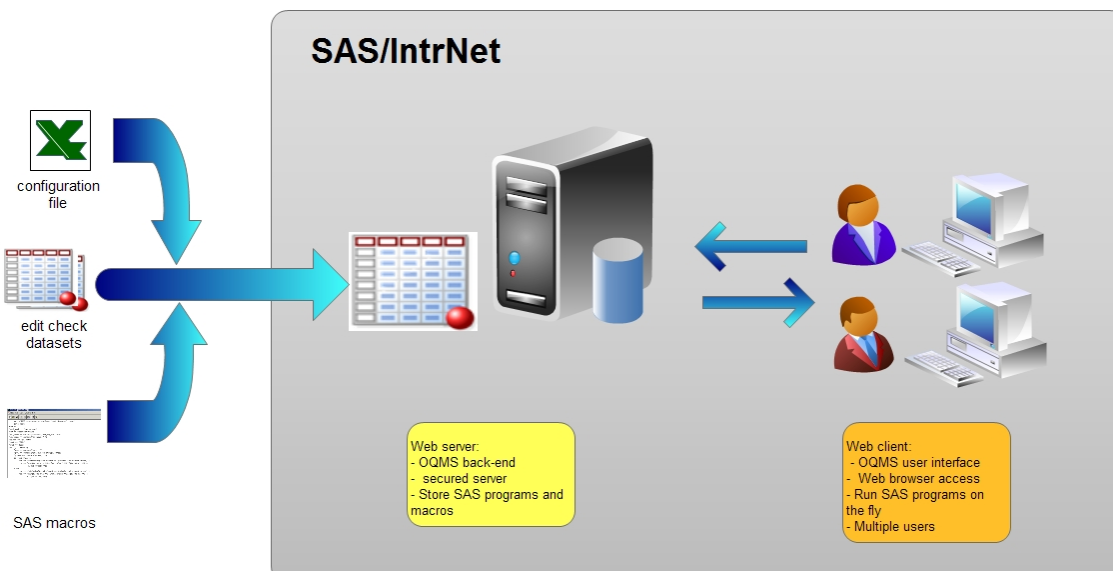


Figure 1. OQMS diagram.

The SAS dataset back-end is generated using the configuration file, edit check datasets, and SAS macros. Users access the application through a web page.

THE SAS DATASET BACK-END

The back-end of the OQMS application is a SAS dataset containing all the discrepancies found in the database. This SAS dataset is stored on a secure SAS/IntrNet server and is generated using the edit check programs, SAS macros and Excel according to the process described below.

Edit checks

The edit checks defined by the clinical team and programmed with SAS are used to build the back end. Each edit check is a SAS program designed to identify data-related issues. They are programmed in a way that each edit check can output a SAS dataset listing all the inconsistencies identified by the program.

These datasets contained the patient number and other variables of interest but no description of the issues. The investigator couldn't resolve a discrepancy if he did not receive an explanation about the problem.

Configuration file

A discrepancy must provide enough information to identify the questionable value completely, describe the problem and point to its source.

An Excel file provides the query message, which is then sent to the investigators explaining the issue found in the database. The Excel file has three columns with the edit check name, the corresponding query message and an option to activate or deactivate the edit check. The advantage of this configuration file is that the clinical team is able to:

- Change the query message or deactivate an edit check whenever they want without modifying the SAS code.
- Customize the query message to display a SAS variable value (any text between two '\$' is interpreted as a SAS variable).

SAS macros

A SAS macro combines the information provided in the configuration file with the SAS datasets, created by the edit checks, to generate the back-end of OQMS. This powerful macro can merge heterogeneous edit checks into a single SAS dataset. This dataset contains all the inconsistencies found in the database and the variable of interest for OQMS application.

Then, another macro generates a unique identifier to track the query and ensure that no duplicates are created for the same discrepancy. This unique identifier is used as a reference by all the SAS programs running in OQMS and by the users for all their internal communications.

This unique identifier is calculated by a SAS macro, using the information available in the database to make it unique. It combines the patient number and edit check name for non-repeated forms. For the repeated forms, the macro combines the patient number, edit check name and repeated form number.

The final result of these SAS programs was the creation of a SAS dataset used as a database for the OQMS application. All the information displayed in the interface is stored in this database and each record has a unique identifier.

FRIENDLY USER INTERFACE

One of the biggest challenges of the project was to develop a user-friendly interface. The application had to be delivered in a short length of time, had to provide a secured connection and allow multiple users to connect simultaneously.

SAS/IntrNet was the ideal solution for several reasons:

- OQMS could easily access the discrepancies generated by the edit checks without any additional conversion, and display them in a web view with minimum effort.
- Well-designed applications can be developed with a combination of SAS programs, HTML and AJAX that can be learned quickly by a good SAS programmer.
- Web-based applications are accessible through web browsers and therefore do not require any software installation by the end-users.
- Few macros are necessary to program the SAS/IntrNet interface, and the learning curve is small for a SAS programmer.

SAS/IntrNet applications are accessible through a web browser, and consequently, developed in HTML. HTML is a simple hypertext language to publish web pages on the internet. Nowadays, web applications offer even more capability than traditional desktop applications through the use of innovative web technologies like AJAX, flash and other technologies. Moreover, developing in HTML is easier than programming in complex languages, such as C++ or JAVA.

The user interface of OQMS was built with a combination of SAS, HTML and AJAX. SAS/IntrNet allows the web application server to communicate with the SAS dataset back-end. HTML is generated by SAS programs to render the web page on the client (web browser), and AJAX retrieves the data from the server asynchronously in the background, without interfering with the display and behavior of the existing page.

After entering a login and password, the users arrive on the main page, which displays the list of discrepancies. This list is generated from the SAS dataset that is stored on the server using a SAS macro and displayed in a spreadsheet format, by rows and columns. Each row represents a discrepancy and each column represents information of interest for the users. Additionally, columns may be sorted and searched for easy browsing.

From this page, users can easily navigate through the list of discrepancies and perform their duties. Most of the fields are not editable and are only displayed for convenience. The user's primary activity is to manage the queries by updating the status. The user uses a drop-down list to define the discrepancy status. The user can also rephrase the query message or enter a comment if it is necessary. Any query update is saved in the back-end. As the user interacts with the application to process the queries, the back-end dataset is programmatically updated. Moreover, the interface was designed with a front-end validation to minimize data-entry errors.

Query ID	Fax	Dataset	Site	Patient number	Location	Owner	Status	Code	DDE	Query message	Comment
000016	missdatehisstat	missdatehisstat	002	002-801	R01B 05 R01B 06	MariAnn	Closed				
000016	missdatehisstat	missdatehisstat	002	002-801	R01B 05 R01B 06	MariAnn	Resolved	DU			
000017	missdatehisstat	missdatehisstat	002	002-801	R01B 05 R01B 06	MariAnn	Resolved	DU			
000032	EPD1	EPD1	002	002-803	R05B 023 R05B 024	MariAnn	Resolved	DU			
000033	TargetStenosis_FinalStenosis	TargetStenosis_FinalStenosis	002	002-803	R05B 021 R05B 027	MariAnn	Closed	NA			
000048	ent20schk2	ent20schk2	002	002-804	R05B 017 R05B 027 R09B 041	MariAnn	Closed	NA			
000092	TargetStenosis_FinalStenosis	TargetStenosis_FinalStenosis	002	002-805	R05B 021 R05B 027	MariAnn	Closed	MT			
000093	incomplete_vars	incomplete_vars	002	002-805	R02LVENTHYPER	MariAnn	Closed	MT			
000093	incomplete_vars	incomplete_vars	002	002-805	R05AS+DS	MariAnn	Auto-Closed				
000093	incomplete_vars	incomplete_vars	002	002-806	R05ASADS	MariAnn	Auto-Closed				
000109	incomplete_vars	incomplete_vars	004	004-001	R05CLOP	Ryan	Closed	NA			
000117	aehocomissstat	aehocomissstat	004	004-002	R08B 039	Ryan	Resolved	DU			
000123	incomplete_vars	incomplete_vars	004	004-002	R02LVENTHYPER	Ryan	Closed	MT			
000142	aehocomissstat	aehocomissstat	004	004-004	R08B 039	Ryan	Resolved	DU			
000154	incomplete_vars	incomplete_vars	004	004-005	R05ASADS	Ryan	Auto-Closed				
000155	incomplete_vars	incomplete_vars	004	004-005	R05CLOP	Ryan	Resolved	DU			

Figure 2. OQMS Interface. Web page displaying the data from OQMS back-end.

1. Unique identifier for each query
2. Site fax review feature
3. Rollover menu
4. Query status field

USING A SAS DATASET AS THE BACK-END DATABASE

OQMS uses a SAS dataset as a database. The interface displays the data stored in the back-end and users update the data through the interface. The dataset contains more than 8000 records with a size greater than 80MB. Updating a record through a data step takes a few seconds. It would not be convenient to freeze the application several seconds every time users update a piece of data.

In order to avoid this annoyance, OQMS uses AJAX for asynchronous updates. Users make changes on the data through the interface while database updates are made in the background. The browser page does not reload, but the data is sent to the server for processing.

SECURE CONNECTION

Only authorized users are permitted to access to OQMS. Users log in the application using a username and password. The password is encrypted at the time of the user's login. A secret question mechanism allows the users to easily retrieve their passwords if they forgot them. User accounts can be created or removed by system administrators. In addition, the application records the frequency of login for each individual user.

Below is the macro to authenticate users when they log on the application.

```
%MACRO doAuthenticate();
%GLOBAL save_uid save_pwd save_email save_firstname save_lastname;
%GLOBAL save_userfullname save_department save_init_pw_chg save_auth_level;
```

```
data _null_;
set profile.users;
if (lowercase("&_uid") eq userid and "&decpass" eq password) then do;
call symput('save_uid',trim(userid));
call symput('save_pwd', "&_pwd");
call symput('save_email',trim(email));
call symput('save_firstname',trim(firstname));
call symput('save_lastname',trim(lastname));
call symput('save_userfullname',trim(fullname));
call symput('save_department',trim(department));
call symput('save_init_pw_chg',trim(init_pw_chg));
call symput('save_auth_level',trim(auth_level));
call symput('authenticated', 'true');
end;
run;
```

"SAVE_" macro variables exist as long as the session exists. They are available anywhere and any time in the application.

```
%IF "&authenticated" eq "true" %THEN %DO;
```

```
/* if first login,the application asks the user to change the generic password */
```

```
%IF "&save_init_pw_chg" eq "0" %THEN %DO;
```

```
%LET saspgm = oqms.w_chgpwd.sas%NRBQUOTE(&)firstlogin=true ; %END;
```

```
%ELSE %DO;
```

```
%LET saspgm = oqms.w_mainview.sas ;
```

```
%LET saspgm = &saspgm%NRBQUOTE(&)dspaction=none%NRBQUOTE(&)dpsize=&save_nrowsperpage ;
```

```
%LET saspgm = &saspgm%NRBQUOTE(&)s_queryid=null%NRBQUOTE(&)s_dataset=null%NRBQUOTE(&)s_fax=null ;
```

```
%LET saspgm = &saspgm%NRBQUOTE(&)s_siteid=null%NRBQUOTE(&)s_patientid=null%NRBQUOTE(&)s_owner=null
```

```
;
```

```
%LET saspgm = &saspgm%NRBQUOTE(&)s_crfname=null%NRBQUOTE(&)s_status=null%NRBQUOTE(&)s_code=null ;
```

```

%END;
%doUserLog(action=login);
%RefreshDatasetList(_dataset=&querycrestrando);
%RefreshConfigDataset();
%END;
%ELSE %DO;
  %LET saspgm = oqms.w_loginfailed.sas ;
%END;
%MEND doAuthenticate;

```

MULTI USER APPLICATION

Multiple users can login concurrently to perform tasks in OQMS. The back-end is a SAS dataset, and like any SAS dataset, it is impossible for two users to have write access at the same time. This problem was resolved with the macro `w_trylock` (see code below). Every time a field is updated in the interface, the application calls this macro before making the change in the back-end. The macro tries to lock the dataset. If no other user is accessing the dataset, the macro locks it, and then the update is performed before unlocking it. However, if someone is updating the dataset, the macro `w_trylock` will not execute and will retry until it can get access to it.

```

/* Macro to lock a dataset*/
macro w_trylock(member=,timeout=5000,retry=500);

  %local starttime;
  %let starttime = %sysfunc(datetime());
  %do %until(&syslckrc <= 0 or %sysvalf(%sysfunc(datetime())) > (&starttime + &timeout));
  %put trying open ...;
  data _null_;
    dsid = 0;
    do until (dsid > 0 or datetime() > (&starttime + &timeout));
      dsid = open("&member");
      if (dsid = 0) then rc = sleep(&retry);
    end;
    if (dsid > 0) then rc = close(dsid);
  run;
  %put trying lock ...;
  lock &member;
  %put syslckrc=&syslckrc;
  %end;
%mend w_trylock;

%w_trylock(member=&querycrestrando);

data &querycrestrando;
set &querycrestrando;
if status in("ready to send") then do;
  status="sent";
  if comment ne "" then comment=trim(left(comment))||" | |--||&today - OQMS: Query sent";
  else if comment eq "" then comment="&today - OQMS: Query sent";
end;

```

```
run;
lock &querycrestrando clear;
```

PRODUCT FEATURES

QUERY PROCESS WORKFLOW

The users navigate through the list of discrepancies, review them and take action by updating the query status. They can, for example, close a query if they think the inconsistency is not important or they can decide to send a query to the investigator.

A workflow is designed so users can change a query's status to properly transition a query throughout its complete life cycle. The process workflow is described in picture 3 below. The process workflow was completely defined by the study's clinical team and can be customized for any study. A query always starts with a DM review status and ends with auto-closed, closed or resolved as the end status. The workflow is enforced by OQMS.

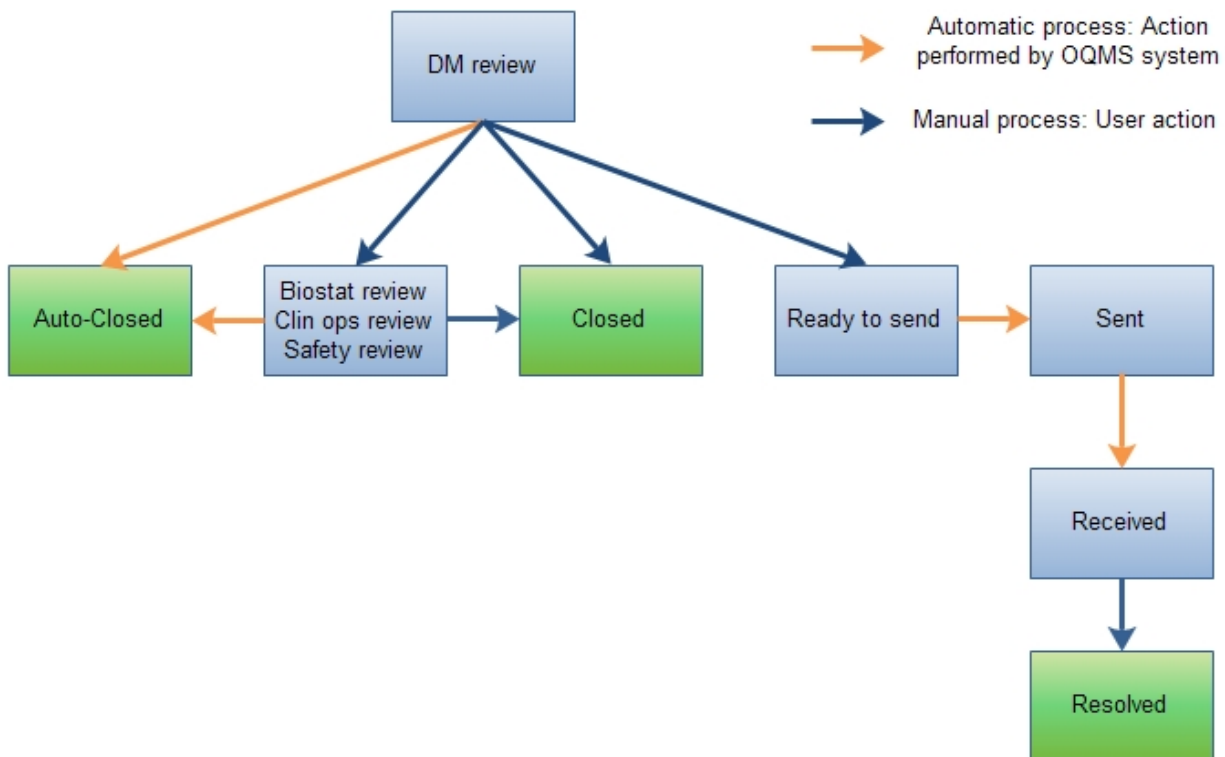


Figure 3. OQMS query workflow. It is enforced by the application. Some status changes can be made by OQMS only (orange arrows).

REPORTS

Using SAS/Internet for application development provides a unique advantage when generating listings and reports. As the leader in business intelligence, SAS is one of the most powerful and flexible tools available to create custom reports. Any SAS program, from simple listings to very complex reports, can be integrated into the application. SAS is much more flexible and powerful but it requires knowledge of SAS.

SAS has the advantage to prepare custom reports and query statistics for top management. The reports provide very valuable metrics to update the top management about the progress of data management activities. These reports were programmed in SAS and run interactively by the users directly from the web.

MANUAL QUERIES

Edit check programs catch a lot of the discrepancies. However many issues are also found by manual reviews. To resolve these discrepancies, data managers issue a manual query to the site.

OQMS provides a tool to write manual queries. Users create a query by filling a form. A SAS program ensures that the entered information is correct and updates the back-end by adding a row for the new query. This new query is then tracked and managed in the application.

PATIENT PROFILES

Patient profiles can be directly accessed from OQMS. Patient profiles were already programmed in SAS before starting OQMS project. However, these files were not convenient for access. The clinical team had to give a list of subjects to the programmer who then ran the program which generated patient profiles as requested by the team. Then, the programmer had to send the files to the team. This was a time consuming and inefficient process. Web applications are superior as they offer direct interactivity with the user.

With OQMS, users can run the patient profile program without the help of programmers by selecting the patient number and application and calling the macro to generate the patient profile. Users can review the patient profile directly in a new window.

```
put "function ShowPatientProfile(patientid) {"/
    "tab = patientid.split('-);"/
    "patnum = tab[0] + tab[1];"/
    "theUrl='&MainUrl.%NRBQUOTE(&)_program=oqms.w_showpatientprofile.sas%NRBQUOTE(&)patnum=' +
    patnum;"/    "popupwin=window.open(theUrl, 'popup_pp');"/    "};"
```

Patient profile program is called from OQMS interface. This program takes the patient number as parameter and displays the result in a new window.

AUDIT TRAIL

The application includes an audit trail system. Every change is recorded in a SAS dataset, which includes the username, a time stamp of the change, the name of the field that was changed, the old and the new values. Users can access the audit trail from the interface. A click on the query identifier opens a new window which displays the history of the changes.

Recording all the changes in one dataset would have created a very large file and slowed down the application. For that reason, the audit trail is made of multiple SAS datasets and one dataset per investigator site is created. Any update related to the investigator's patients is recorded in the corresponding table.

SITES FAX REVIEW

For this study, the queries were sent to the clinical research coordinators (CRC) by email. OQMS generated one PDF file per site with the list of discrepancies to resolve. Four elements were shown on the query form:

- A unique identifier
- The location of the issue on the CRF
- A query description
- An empty cell for the investigator answer

The CRCs receive the PDF file by email, print it, answer the queries and fax back the document to a central fax number. The fax document is automatically converted into a PDF file that is saved into a structured directory on the server. The data processor then links them to the corresponding query using the previously defined unique identifier.

Each file is named following a specific mechanism. The data processor enters the PDF file name and the corresponding query identifier into a special form designed for this task. After the data processor submits the data, the information is stored in a SAS dataset. This SAS table contains two columns, one for the PDF file name and another one for the query identifier. The application uses this table to retrieve a fax associated with a query.

This feature allows users to refer to the faxes from the application. If a fax is received for a query, a new icon appears next to the query identifier. A click on this icon opens the fax document.

In addition, users can browse faxes using the fax module available in the interface menu. The faxes are stored on the server and organized by site and by patient. Users can navigate in the repository and select a fax that can be reviewed in the same window.

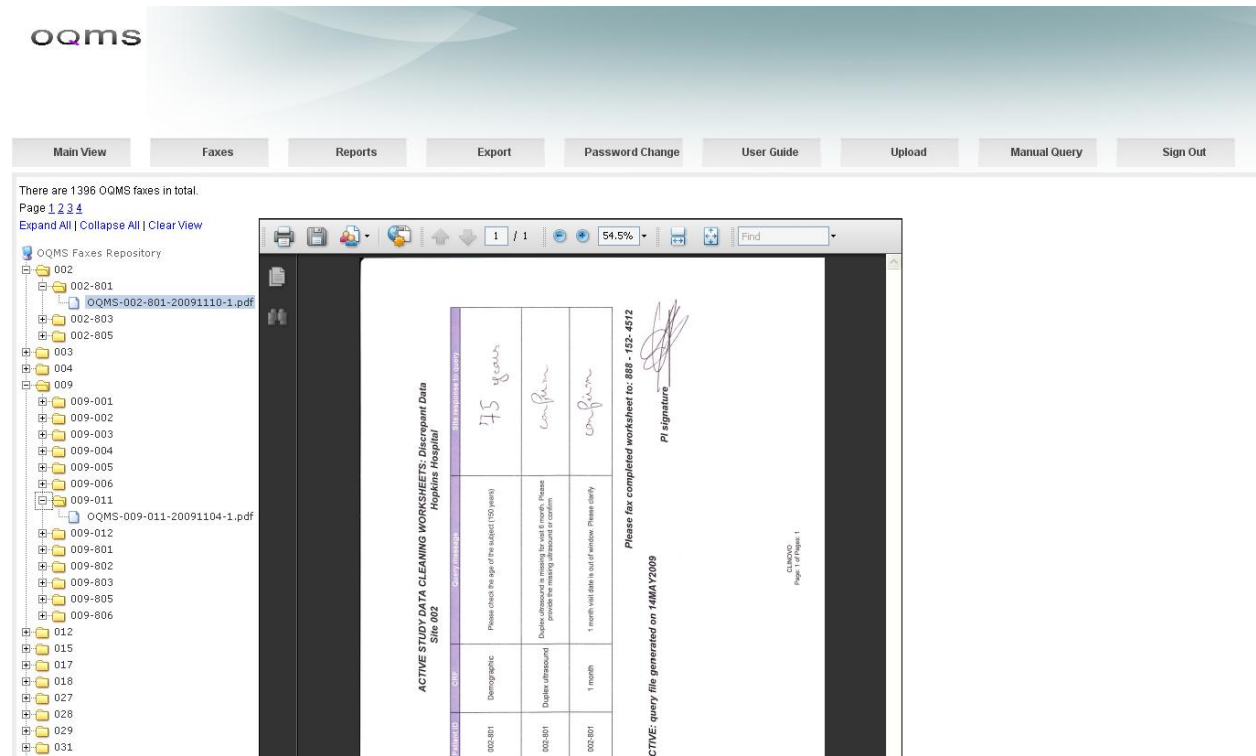


Figure 4. Site fax review module. Users can navigate in the repository and view the received faxes.

SAS FOR THE DEVELOPMENT OF CUSTOMIZED APPLICATION

OQMS: A STAND-ALONE SOLUTION FOR QUERY MANAGEMENT

OQMS is a new stand-alone solution for query management developed with SAS and SAS/IntrNet. It was successfully used for a strategic and critical clinical trial with a very tight deadline. It helped the data managers to clean the data and meet their deadlines. SAS/IntrNet offers higher flexibility in the design of applications. It can be quickly implemented and easily customized to meet users' requirements. This application can efficiently replace spreadsheets to manage and track discrepancies.

LESSONS LEARNED

Over 15 different users used OQMS. Each user found the application user-friendly and enjoyed working with the application. The majority had no experiences with SAS, and they did not know it was possible to develop an entire application in SAS with a point and click interface.

They appreciated the flexibility of the system. OQMS was customized for and by the users. Seven versions of the application were released during the project and each new version provided new features and bug fixes. The users also appreciated that everything was integrated in the system, from reports to sites fax review, including patient profiles.

The most challenging aspect of the study was to develop the application in a very short time period. It only took around three weeks for two programmers to develop the first alpha version. This first version did not include a lot of the final features but was functional enough for users to start processing the queries.

BENEFITS OF WEB APPLICATION

Many software applications could be improved by migrating them to a web-based application, which offers the following benefits:

- They are compatible across platforms and operating systems with a web browser, whereas a desktop application has to be developed for a specific OS.
- Web-based applications are deployed directly on the server, which greatly facilitates the application maintenance. New features and new releases can then be rolled-out on the server without the need for client update.
- Users can access the application from anywhere because it only requires a browser and an internet connection.
- With a simplified architecture, maintenance is reduced and there are fewer requirements on the end user's system. A SAS developer in a small team can then easily take over the maintenance without the need of external specialized resources.
- The system is automatically upgraded with new features that are added to the application on the server and become immediately available.

For all these reasons, migrating SAS programs to a web-based application is a very interesting opportunity.

SAS AND SAS/INTRNET: AN EXCELLENT PLATFORM FOR RAPID WEB CUSTOM DEVELOPMENT

An application that is developed with SAS and SAS/IntrNet offers many benefits.

- It combines the advantages of web application for a point and click interface with the power of SAS for data exploration, analysis, and reporting.

- SAS programmers with HTML knowledge can quickly and cost-effectively develop applications.
- Many clinical studies use derived or analyzed SAS data sets. With SAS/IntrNet, programmers can quickly create customized applications to fit any user need at a minimal cost.
- Using SAS/IntrNet to directly manipulate SAS data is faster than calling or retrieving SAS from another programming language and then parsing the SAS output.
- Many programs could use the benefits of SAS/IntrNet to develop dynamic application like OQMS or to make SAS program outputs and reports available to all the users in an organization.

CONCLUSION

This paper introduces OQMS, a new stand alone query management system. Though the application has been developed for a specific study, its flexibility and multiple features make it a possible solution for query management. It can be used with any clinical trial simply by transferring the data to SAS data sets.

OQMS demonstrates the power of SAS and SAS/IntrNet for developing customized applications. SAS/IntrNet is a good tool that makes tasks easier and allows users to access information from dynamic web pages.

This application is a good example of the capabilities of SAS and SAS/IntrNet. With these tools, many more applications are imaginable and possible.

REFERENCES

Play Nice with Others: Waiting for a Lock on SAS Table, John Leveille, Jimmy Hunnings, pharماسug 2005. <http://www.lexjansen.com/pharماسug/2005/posters/po33.pdf>

ACKNOWLEDGEMENTS

We would like to thank all the people involved in the development of OQMS: Ryan, Hendra, Dmitry, Jian, Miki and Bing, as well as the users for their valuable feedbacks.

ABOUT CLINOVO

Clinovo is the leading Clinical Data Solution Provider delivering expert, tailor-made biometrics services to accelerate clinical trials and ensure successful FDA submission. Clinovo delivers the fastest clinical trial development implementation in the industry, leveraging a suite of internal clinical applications as well as best-of-breed new technologies.

Clinovo's comprehensive suite of services includes EDC, data management, biostatistics, and CDISC. Clinovo is an expert in open source clinical trial technologies such as OpenClinica, allowing it to deliver some of the industry's most cost-effective solutions.

Sponsors choose Clinovo for its unmatched domain technology expertise, attention to details, problem solving skills, and quick execution.

Founded in 2003, Clinovo is headquartered in Sunnyvale, California, USA.

Additional information about us and our services is available at www.clinovo.com

A copy of this white paper and the OQMS business brochure are also available at www.clinovo.com/oqms

CONTACT INFORMATION

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

Romain Miralles
SAS specialist at Clinovo
1208 E. Arques Avenue, suite 114
Sunnyvale, CA 94085
E-mail : mrom34@gmail.com

Ale Gicqueau
President and CEO at Clinovo
1208 E. Arques Avenue, suite 114
Sunnyvale, CA 94085
E-mail : ale@clinovo.com
Phone : +1 800 987 6007

You can also contact the Marketing and Communication Coordinator for more information about Clinovo and its solutions.

Sophie McCallum
Marketing & Communication Coordinator at Clinovo
1208 E. Arques Avenue, suite 114
Sunnyvale, CA 94085
E-mail : sophie@clinovo.com
Phone : +1 408 462 5448

OQMS is available today through our clinical data services

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.



Online Query Management System
An innovative SAS/IntrNet Solution for
Clinical Data Management