PhUSE 2007

Paper AD05

Web-enabling the Pharmaceutical Sector

Exploring the SAS Information Delivery Portal, skipping the Double Dutch

Raymond Ebben, OCS Consulting, Rosmalen, the Netherlands
Jules van der Zalm, OCS Consulting, Rosmalen, the Netherlands

ABSTRACT

The paper will discuss how pharmaceutical companies can take advantage of web technology when integrating information from departments that contribute to the clinical trial process.

It is based upon a version of the SAS Information Delivery Portal which has been set up to function as a proof of concept environment to illustrate our vision of how web technology can be used to allow an organisation to improve collaboration between people and across departments.

Supported by the examples within this portal environment, possibilities for integration will be explored thus creating an understanding of how to fully utilise the functionality the portal has to offer.

The topics that will be covered are:

- Objectives of this paper;
- Introduction to portals;
  - Portals;
  - The SAS Information Delivery Portal;
  - Portal environment.
- Our portal environment;
  - Motivation;
  - Users of the portal;
  - Integrating departmental information;
  - Taking advantage of SAS on the web;
  - The value of metadata.
- Maintaining the Information Delivery Portal;
- Validation of portal programs;
- Security of the portal;
- Tips;
- Conclusion.

INTRODUCTION

With a specific interest in web technology and several years of experience using SAS mostly within the pharmaceutical sector we decided to define a project in which we could bundle this experience and knowledge to develop a “proof of concept” version of the SAS Information Delivery Portal.

This paper will walk through multiple ways to integrate processes and departments that contribute to the core activities of a pharmaceutical company. From patient recruitment to data collection and from data cleansing to generating reports, rather than covering all functionalities, the paper will explore how web technology can be of added value.

NOTE: All clinical information used within the paper and demonstration portal is dummy data, created by OCS Consulting for demonstration purposes only.

OBJECTIVES OF THIS PAPER

The objective of this paper is to illustrate how pharmaceutical companies can take advantage of web technology to improve collaboration between people and across departments when integrating information from departments that contribute to the clinical trial process. For this purpose, OCS Consulting has developed a “proof of concept” instance of the SAS Information Delivery Portal presenting a broad range of examples to demonstrate the potential.

INTRODUCTION TO PORTALS

PORTALS IN GENERAL

A portal can be best described as a website that acts as a "doorway" to an Intranet or a portion of the Internet, targeted towards one particular person or group of people.

The main function of a portal is to bundle information in logical or functional areas.

When breaking down a portal into its core elements the main element of the portal is a web site.
To structure the content on this website, the content is categorised in so called portal pages which can be accessed through the navigation menu.

In principle a portal page is a page within the portal that contains related information. For instance a page called “Validation” would contain information related to the topic “Validation”.

The information on such a portal page is most commonly presented in one or more “portlets”. A portlet can be best described as a container for holding related information. This information can consist of a broad range of items. Typically a portlet contains information such as a web page, an image, a hyperlink or a web application. Examples of such web applications are news, weather reports, forums, and email.

Given the above a portal is therefore an environment that allows for a personalised virtual workspace to be created. Because a portal is available through a web browser, this workspace can be made available anytime, anywhere.

A more specific type of portal is an enterprise portal. This is a portal which assists in sharing information and promoting collaboration in workplaces. The most common example of such an enterprise portal would probably be Microsoft SharePoint.

THE SAS® INFORMATION DELIVERY PORTAL

The SAS Information Delivery Portal is an enterprise portal which provides a common front end to the SAS enterprise intelligence platform, providing a personalised virtual workspace which deploys the power of SAS software. The following image is a screenshot of the SAS Information Delivery Portal.

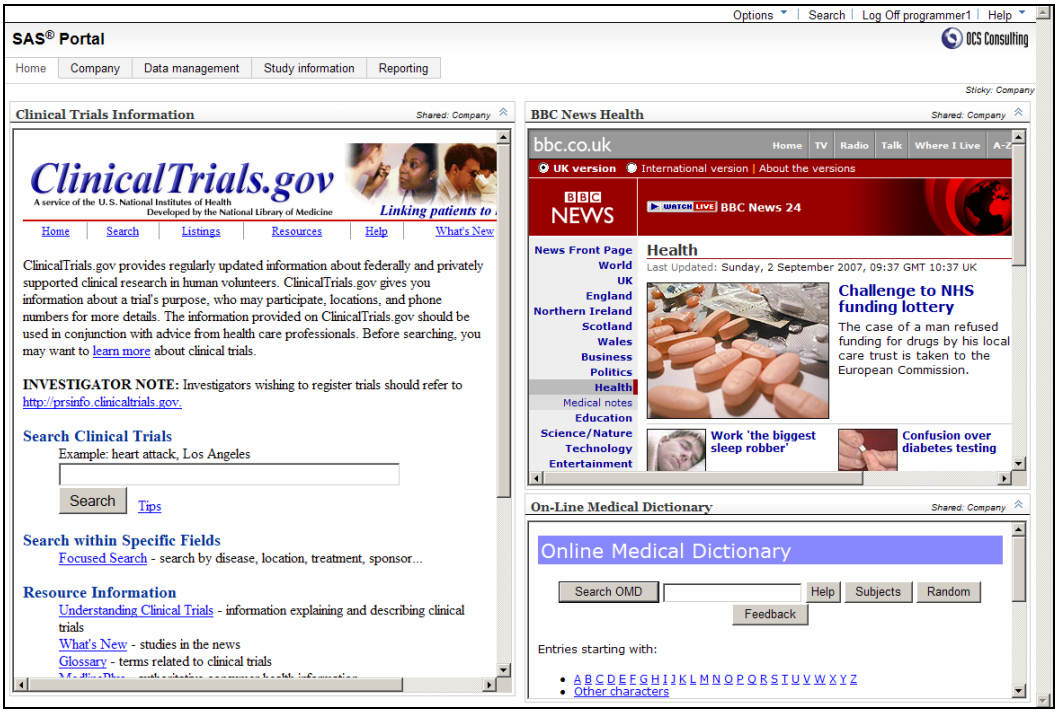


Image 1: The SAS® Information Delivery Portal

The appearance of the portal can be customised to represent the corporate image. To illustrate this, the demonstration portal has been adapted in appearance.

The portal consists of the following main screen components:

- Portal menu;
- Information header;
- Navigation menu;
- Portal page(s);
- Portlets.

PORTAL MENU

The default portal menu consists of the items: options, search, log off and help. The options menu allows the user to customise the portal, its content, the layout etcetera. Search allows to user to search for specific information within the portal. Log off will log off the current user. Help will open up the portal's help menu.

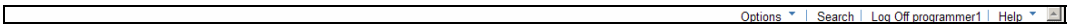


Image 2: Portal menu

INFORMATION HEADER

The information header contains information on what is currently displayed in the main portal screen. If, for instance, the result of a SAS program is displayed in the main portal window, the information header will present the name of the executed program.

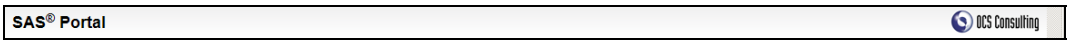


Image 3: Information header

NAVIGATION

The navigation menu allows the user to navigate the portal content. Via the tabs the available portal pages can be accessed.

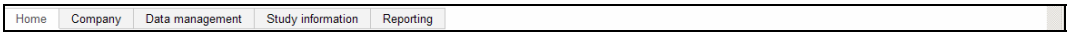


Image 4: Navigation

PORTAL PAGES

By means of the navigation menu the user can access the available portal pages. A portal page is a page that can contain one or multiple related portlets. The image below illustrates a part of the first portal page as presented in Image 1, with portlets, each showing an Internet site.

Image 5: Portal page

PORTLETS

Portlets are visible containers, used to bundle related information. Within these portlets different types of content can be stored. The most common types are links, web pages and stored processes. Links can be links to other web sites, documents, images etc. Web pages can be a website as shown in image 5, but could just as well be a web application such as web mail or a discussion forum. Stored processes are SAS programs that can be executed through the portal and have the ability to accept parameters, creating a possibility to develop dynamic programs. For instance, when a SAS program generates a listing on all subjects, it would be possible for the user to select the subject ID(s) to be present in the listing.

PORTAL ENVIRONMENT

This section will create a high level view of the different components of the portal environment.

The SAS Information Delivery Portal is a portal that runs on a web server. From the portal itself the visible elements are controlled. With the SAS Web administrator account, portal pages can be defined, as well as portlets and assigning content to the portlets.

These visible elements can be assigned to user groups which will determine the appearance of the portal for a user (group). Information such as user groups are derived from a central SAS server called the SAS metadata server. This server stores the metadata of the company: Libraries, Data, business rules and much more. Access to this metadata is controlled via access rights linked to users or user groups.

The content of the SAS metadata can be maintained via the SAS management console, which is a graphical shell allowing the user with the right credentials to add, remove and modify metadata.

The SAS metadata server also contains settings to connect to the stored process server, a SAS server dedicated to execute stored processes. Therefore the SAS metadata server is the main control for all SAS interfaces such as the Information Delivery Portal, Enterprise Guide and the MS Office Add-in. Stored processes which are not designed specifically for usage via a web browser can be used from all these interfaces.

The following image illustrates a high level overview of the above

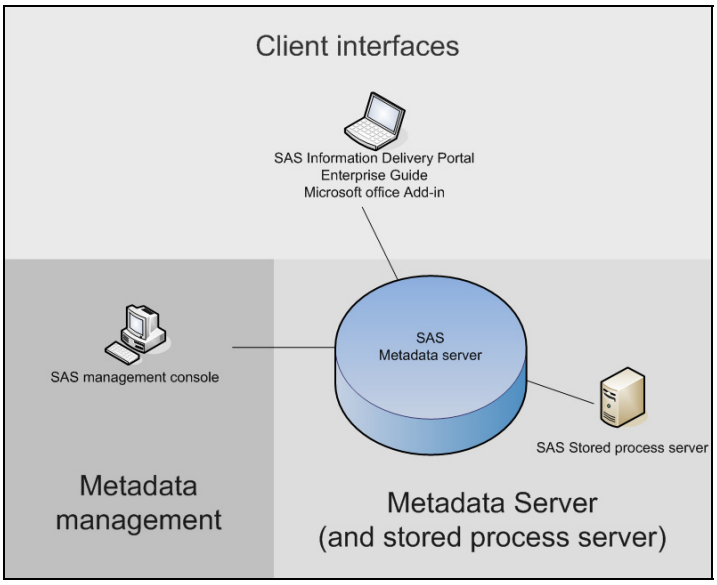


Image 6: High level overview

OUR PORTAL ENVIRONMENT

MOTIVATION

After developing a demo on how the Information Delivery Portal can be used in the pharmaceutical sector, we undertook an assignment with one of our clients to customise the portal to their company style, and to develop specific SAS programs for use in their Information delivery Portal. With the existing demo, newly gained knowledge and deepened insights from our client case we decided to take our existing demo a step further, not just showing how the information delivery portal can be used within the pharmaceutical sector, but also show how web technology can be used to allow an organisation to improve collaboration between people and across departments.

USERS OF THE PORTAL

For our portal environment we have selected the type of information consumers to be:

- Managers;
- Data managers;
- Statistical programmers;
- Statisticians.

These users are purely illustrative and the general concept of the examples given could just as well be applied to other types of information consumers in the pharmaceutical sector.

As discussed the functionalities offered by the portal are managed by the SAS Metadata server. Access rights to separate items such as libraries are also controlled by the metadata server and can be granted to a user or a user group. In our environment we have decided to manage security ONLY via user groups because this improves maintainability. The user account is used to grant access to the portal, and if requested to perform some logging on the processes started by this user. The user itself has no rights within the portal. All rights the user has are inherited from the different user groups the user is a member of.

For user groups we distinguish' three different types of groups: Company, roles and projects.
The first group type is a physical group called company. All users are a member of this group and will be allowed to perform everything within the portal that is not restricted.

The second group type consists of all the groups that specify a specific role the user has within the portal. These roles define what the portal will look like, and what functionalities are available to the user. In our portal environment we have defined role groups that define the different roles our users have in the portal, which are:

- Manager Data Management;
- Manager Statistical programming;
- Manager Statistics;
- Data manager;
- Statistical programmer;
- Statistician.

In an example we will take a manager who is responsible for Data Management as well as Statistics. This manager would be a member of the group company, Manager Data Management and Manager Statistics. If this manager would be responsible for Statistics only the manager would be member of the groups company and Manager Statistics. In the first situation, this manager would have access to the management information for Data Management and Statistics (1), as in the latter this manager would only be able to view the management information regarding Statistics(2). It is with these groups that the appearance of the portal is defined.

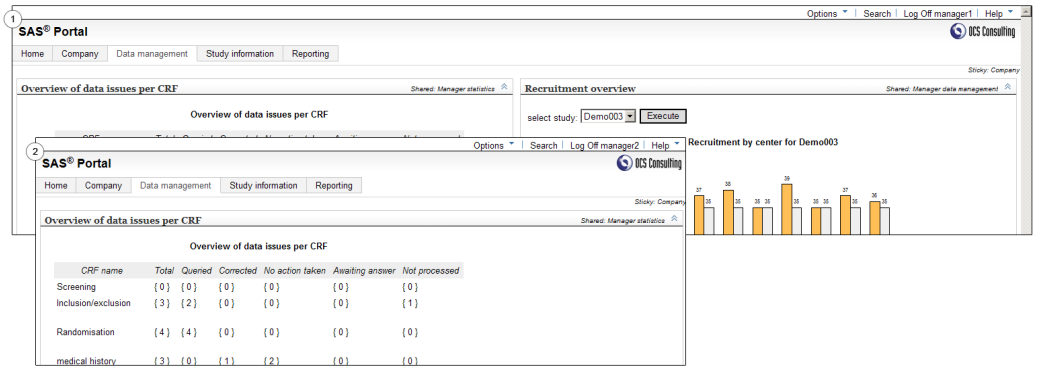


Image 7: Appearance defined by group membership Manager1 is allowed to see the recruitment overview

The third type of group contains all the groups that are created per project. This means that for each project or study, a group will be created. If a user is working on a particular project, the user will be a member of this project group. These groups will not be used to control portal content, but will come into play when we start working with the project data.

To summarise: All users will be member of company and can view all non restricted contents of the portal. The role groups define the behaviour, appearance and functionality of the portal, and the projects groups will be used when data is requested through stored processes. This is summarised in the following image.

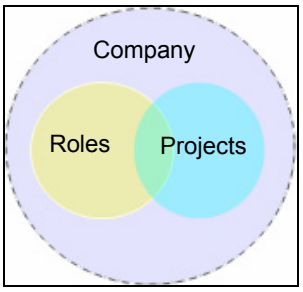


Image 8: user groups

INTEGRATING DEPARTMENTAL INFORMATION

Within each department of a pharmaceutical company there is valuable information present. Some of this information could be of added value to other departments, be it direct or as supportive or background information.
When the need for this information arises, most likely someone will pick up the phone and call the department that has the required information. This information will then be provided (often manually) to the requesting department.

Next to the fact that there is a threshold because you have to pick up the phone and explain what you want; the person providing the information has to stop their day to day routine to fulfil the request. More often than not the reason that this information is not available is that the requested information resides in a system that is not accessible to the other department.

To illustrate how the Information Delivery Portal can assist to improve collaboration between people and across departments we will present an example situation where there is data available within a department which is of value to another department and how this requested information can be presented trough the portal.

The example situation is related to data quality. For the example we assume that the company uses a data management tool like Oracle Clinical. A data manager is able to provide you with the number of data quality checks within a specific study, if requested even per panel. Also the number of data issues found in the study as to date can be provided. This is because this information is available within Oracle Clinical and the data managers has access to Oracle clinical. A statistical programmer has no access to Oracle clinical, but this information can be of use. The most trivial example is that a statistical programmer finds a possible data issue and want to ensure that the data issue is tracked by data management.

As a possible approach we have created SAS programs that extract data from the Oracle clinical database and present this information to the specified information consumer. For our example we would present listings to the statistical programmer. These programs allows the statistical programmer to specify the type of information to be presented: A list of data quality checks in the study which can be filtered per panel and a list of known data issues which can be filtered per panel or per subject.

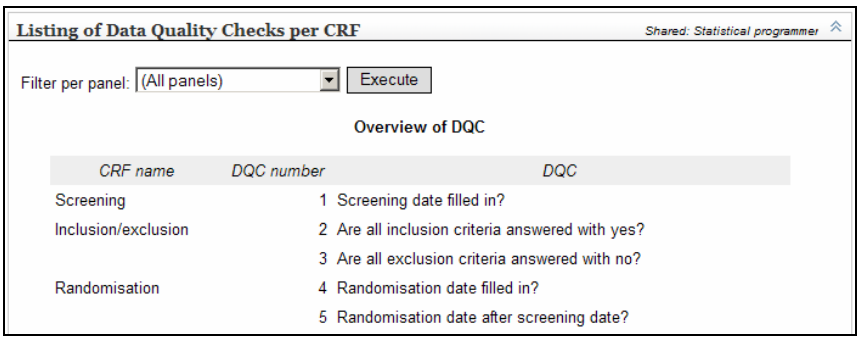


Image 9: List of data quality checks in the study which can be filtered per panel

The statistical programmer is capable of retrieving this information because we granted the role of statistical programmer to request this type of data from oracle clinical through our SAS program. This is also where the project groups kick in. The statistical programmer is only allowed to retrieve this information for the projects / studies of which the programmer is a member.

For the same data quality information we take another example; a statistician. Unlike the statistical programmer the statistician might be interested in an overview of the number of data issues encountered per panel, and the action taken on these data issues. The main difference with the previous example is that the statistician requires a different level of the same information. For the rest the same principles apply. The role determines that the statistician will be presented with the higher level view, and the project groups determine of which projects the information can be viewed.

Overview of data issues per CRF

CRF name	*Total*	*Queried*	*Corrected*	*No action taken*	*Awaiting answer*	*Not processed*
Screening	{0}	{0}	{0}	{0}	{0}	{0}
Inclusion/exclusion	{3}	{2}	{0}	{0}	{0}	{1}
Randomisation	{4}	{4}	{0}	{0}	{0}	{0}
medical history	{3}	{0}	{1}	{2}	{0}	{0}
Demographics	{0}	{0}	{0}	{0}	{0}	{0}
Medication	{0}	{0}	{0}	{0}	{0}	{0}
Adverse Events	{0}	{0}	{0}	{0}	{0}	{0}
Serious Adverse Events	{0}	{0}	{0}	{0}	{0}	{0}
End of trial	{1}	{0}	{0}	{0}	{1}	{0}

Image 10: Overview of data issues found per panel

TAKING ADVANTAGE OF SAS ON THE WEB

The SAS Information delivery Portal provides access to the SAS enterprise intelligence platform. This platform allows applications such as the Information Delivery Portal to access SAS software in a regulated manner. Most of these "rules" are defined in the SAS metadata server and therefore apply to all applications that execute a stored process through the SAS enterprise intelligence platform.

As an example we will take a SAS stored process which results in an image of a simple line plot. Within the SAS metadata server it is defined which user (groups) have the right to execute this stored process. We can therefore choose to execute this process through an application as Microsoft PowerPoint. After we provided the correct credentials to the login we are presented with the result: an image in a PowerPoint presentation.

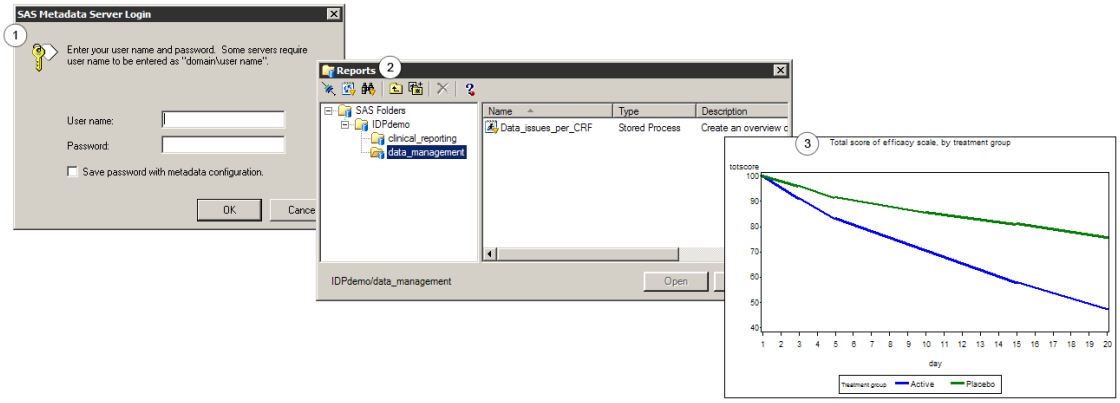
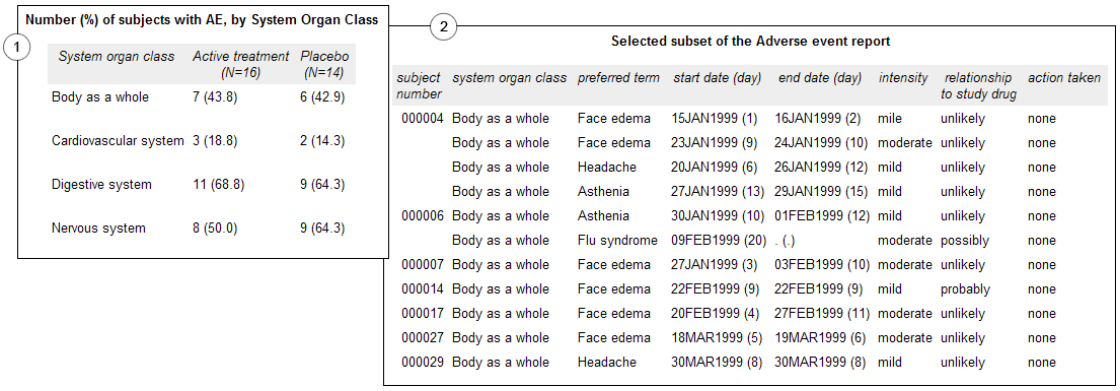


Image 11: Logon (1), select a stored process (2) and view the output in PowerPoint (3)+

If we would execute the same stored process through the SAS Information Delivery Portal we would end up with the exact same result, but then presented in a web browser, rather than in PowerPoint.

When looking only at stored processes, the biggest advantage the portal has to offer is that the stored processes are executed through a web browser and the result is presented in a web browser, allowing you to exploit this functionality.

A simple example of this is to create a summary table of for instance Adverse Events, and adding a hyperlink to each result. When clicking on the hyperlink another stored process is started retrieving the detailed information on which the summary is based. In the example below we click on the cell with value 6 (37.5) representing the SOC Body as a whole for the active treatment and as a result a listing is produced presenting the subset on which the figure is based.



1

Number (%) of subjects with AE, by System Organ Class

System organ class	*Active treatment (N=16)*	*Placebo (N=14)*
Body as a whole	7 (43.8)	6 (42.9)
Cardiovascular system	3 (18.8)	2 (14.3)
Digestive system	11 (68.8)	9 (64.3)
Nervous system	8 (50.0)	9 (64.3)

2

Selected subset of the Adverse event report

subject number	*system organ class*	*preferred term*	*start date (day)*	*end date (day)*	*intensity*	*relationship to study drug*	*action taken*
000004	Body as a whole	Face edema	15JAN1999 (1)	16JAN1999 (2)	mile	unlikely	none
	Body as a whole	Face edema	23JAN1999 (9)	24JAN1999 (10)	moderate	unlikely	none
	Body as a whole	Headache	20JAN1999 (6)	26JAN1999 (12)	mild	unlikely	none
	Body as a whole	Asthenia	27JAN1999 (13)	29JAN1999 (15)	mild	unlikely	none
000006	Body as a whole	Asthenia	30JAN1999 (10)	01FEB1999 (12)	mild	unlikely	none
	Body as a whole	Flu syndrome	09FEB1999 (20)	. (.)	moderate	possibly	none
000007	Body as a whole	Face edema	27JAN1999 (3)	03FEB1999 (10)	moderate	unlikely	none
000014	Body as a whole	Face edema	22FEB1999 (9)	22FEB1999 (9)	mild	probably	none
000017	Body as a whole	Face edema	20FEB1999 (4)	27FEB1999 (11)	moderate	unlikely	none
000027	Body as a whole	Face edema	18MAR1999 (5)	19MAR1999 (6)	moderate	unlikely	none
000029	Body as a whole	Headache	30MAR1999 (8)	30MAR1999 (8)	mild	unlikely	none

Image 12: Summary table with drilldown functionality (1), and the detailed (drilled) information (2)

Another example is a graphical presentation of a patient profile. This table is created as an HTML table and can therefore benefit from both graphic and dynamic functionalities. It contains the domains Adverse Events, Drug accountability and Vital Signs. To assist the reviewing process, the user has the ability to expand and collapse the different domains allowing two domains to be displayed directly underneath each other.
Of course we could add functionality equal to the one above allowing us to view all subjects with the “syncope” adverse events when clicking on this name.

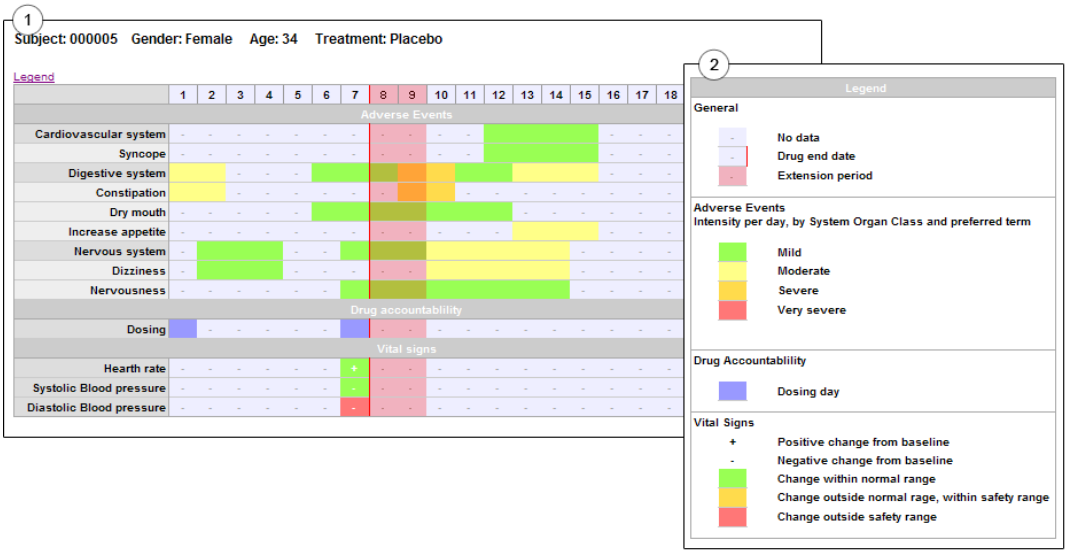


Image 13: Graphical patient profile (1), and the legend (2)

Both examples demonstrate how to add functionality to the output side of the process.
We can also take advantage of controlling the input side. The simplest example of achieving this is to create a stored process that reads all the members of a library and creates a selection form from the result, allowing the user to select a dataset to review and optionally set a where clause.

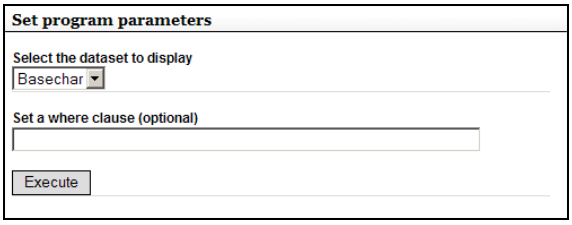


Image 14 controlling the input side

As a final example we complete the circle. This example allows the user to dynamically create a table by dragging variables into the column or row dimension of a table from the list of available variables (classes). Inside the table are the calculation variables from which the user can select which ones to include in the report. When this stored process is executed the table specified will be generated. Above the table the same table selection screen will be presented, so the user can easily modify and re-execute the request. Some pre defined variables will initially be summarised (in this case position). If the user clicks on this variable the stored procedure is re-executed with the information to expand (drill) the selected variable.

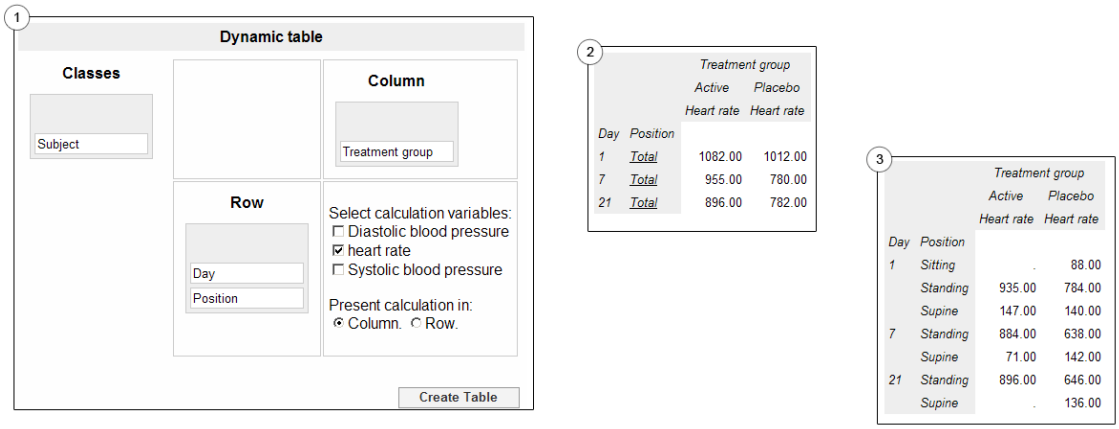


Image 15: Dynamic table selection (1), Result table (2) and the drilled result (3)

THE VALUE OF METADATA

Another powerful feature of the SAS enterprise intelligence platform is the use of metadata. According to the dictionary “Meta“ means “referring to itself” therefore Metadata would mean data referring to itself, and in fact it is!

To start of with an example of metadata in the pharmaceutical sector, let’s take the variable in a dataset called BMI. On its own we can guess what this variable contains. But we cannot be sure. If we look at the metadata description of this variable we find that this variable contains the Body Mass Index and in the extended attributes we find that the derivation method for BMI is weight(kg)/(height(m)*height(M)).

This kind of information is very valuable to a company and is often underestimated. An example of how to make use of this metadata is when insight is required in the derivation method of a variable. Let’s take our BMI example and assume that for some peculiar reason the FDA decides to modify the calculation rule, and 5 years later I need to know which studies were reported containing the BMI variable, categorised by calculation method.
Luckily we have created metadata. By searching on all variables within all studies for the description “Body Mass Index” and retaining the derivation method information, we can easily create a table or listing from the result.

This is a powerful example, but metadata can be more. Within the portal we have several examples where we exploit the value of metadata. A simple example is to display a listing of the team members, which is retrieved from the members of the project group defined in the metadata server.

Another example of using metadata in combination with the authorisation rights is to set authorisation on columns of a dataset. At first this might not sound useful, but when for a blinded study we remove the authorisation of the column containing the unblinding information this gets interesting. In effect this means that a statistical programmer can work with the dataset without even knowing the unblinding information exists, as for his group profile it does not exist.

1

Display of dataset: vitalsig

Subject	Treatment group	Day	Position	Heart rate	Systolic blood pressure	Dyastolic blood pressure
000001	Placebo	1	Standing	78	140	90
000001	Placebo	7	Standing	76	134	85
000001	Placebo	21	Standing	74	133	86
000007	Active	1	Standing	66	155	84
000007	Active	7	Standing	72	154	80
000007	Active	21	Standing	70	149	79
000012	Active	1	Standing	58	145	78
000012	Active	7	Standing	66	148	66
000012	Active	21	Standing	69	152	77

2

Display of dataset: vitalsig

Subject	Day	Position	Heart rate	Systolic blood pressure	Dyastolic blood pressure
000001	1	Standing	78	140	90
000001	7	Standing	76	134	85
000001	21	Standing	74	133	86
000007	1	Standing	66	155	84
000007	7	Standing	72	154	80
000007	21	Standing	70	149	79
000012	1	Standing	58	145	78
000012	7	Standing	66	148	66
000012	21	Standing	69	152	77

Image 16: Showing unblinded (1, highlighted column) versus blinded (2) datasets

As a final example we take the dynamic table creation program described in the section taking advantage of SAS on the web. You might have noticed that the variables used in the table selection screen are of different types. The different types of variables are:

- Variables that are presented as is;
- Variables that are summarised and can be drilled;
- Calculation variables.

This differentiation is not hard coded in the selection screen, or in the procedure. This information is gathered from the Metadata server. In the metadata server it is possible to specify the summary role of a variable (Class, ID, Statistic, Default and (None)). For the example above the “normal” variables will have the “_DEFAULT_” summary role assigned. The calculation variables are defined with a “STATISTIC” summary role, and the drillable variable are defined by a “CLASS” summary role. This information is retrieved by the dynamic table program and used to generate the dynamic selection screen. Below is a screen shot of the assignment of the summary role in the metadata server.

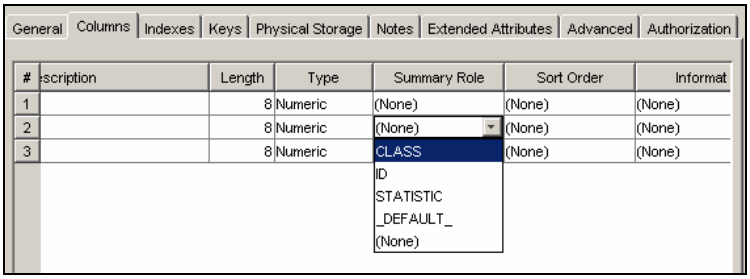


Image 17: Assigning summary roles to the variables

MAINTAINING THE INFORMATION DELIVERY PORTAL

This section will provide a high level overview of the maintenance of the Information Delivery Portal. As briefly explained in the section “Portal environment”, maintenance of the portal can be performed from both the portal as well as the SAS management console.

MAINTENANCE IN THE PORTAL

From within the portal, portal pages and portlets are created. To perform this task SAS has a special user called SAS Web Administrator. This account has the privileges to create the required portal pages, portlets and assign content. When creating these elements they can be assigned to a user group. When assigning a portal page or portlet to a specific group every member of the group is allowed to view the element, and of course everybody who is not a member will not see the element. This was illustrated in image 7. The following image displays the creation of a portlet.

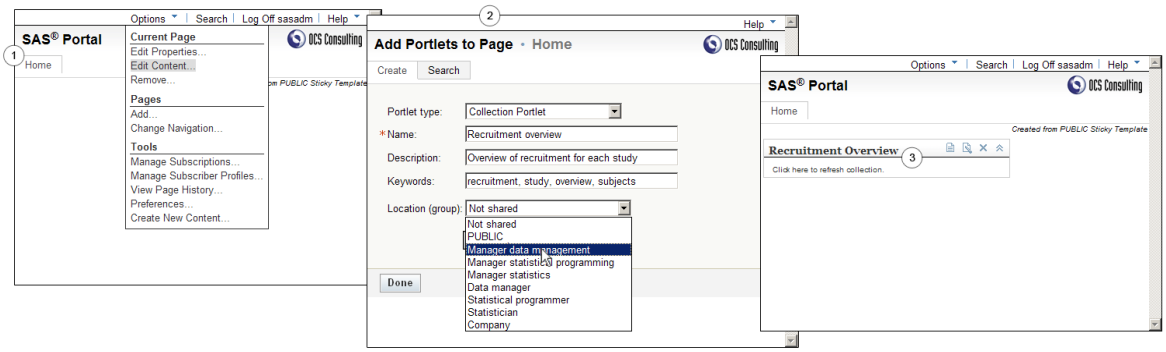


Image 18: Creating a new portlet

The elements created in the portal are merely containers. The main function of a portal is to bundle information in logical or functional areas. This information can be information like other web pages, websites, documents or images. This content is assigned to a portlet and is maintained via the Portal itself. The user groups and SAS related information such as stored processes are regulated via the SAS management console.

MAINTENANCE IN THE SAS MANAGEMENT CONSOLE

The SAS management console is a graphical user interface from which users with the appropriate access rights can add, remove and modify the metadata. The information that can be stored within the metadata server is too divers to discuss in this paper. Therefore we will present an example of how to use the SAS management console to give an impression.

To define a user, the user manager is selected. Right click on the right pane or on the user manager node will present a menu from which we can select New -> user or New -> group. When selecting New-> User, we are presented with a screen to fill out the information required for this user. The same applies when we select New-Group. To assign a user to a group we double click on the user from the groups tab and move the desired groups from the “Available groups” pane to the “Member of” pane.

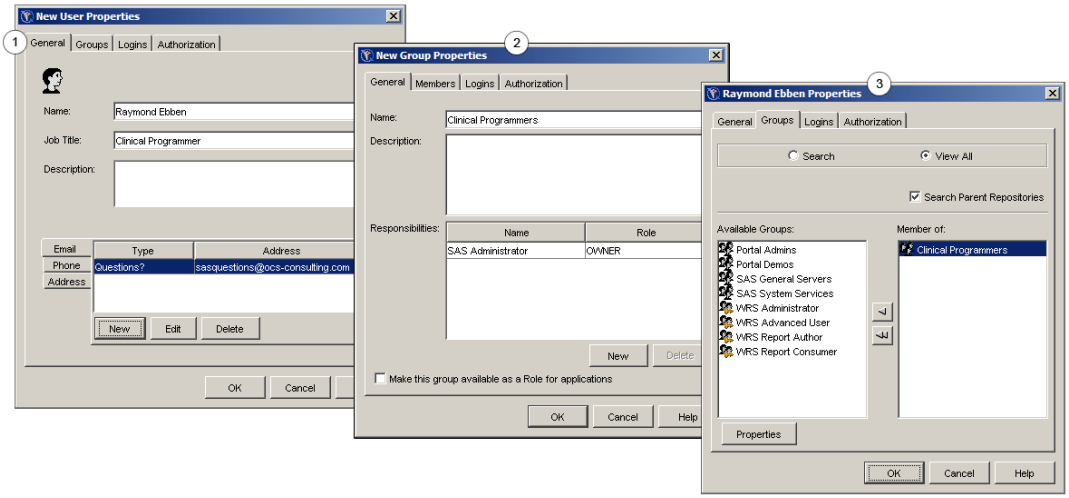


Image 19: Create a user (1), a user group (2) and assigning membership (3)

VALIDATION

An important topic in the pharmaceutical industry is validation. Therefore we want to create a clear understanding of the main difference between a regular SAS program and a stored process. From this we can analyse the impact on validation.

The following lines of code are a snapshot of a regular SAS program generating a Vital Signs listing.

```
%*****************************************************************************;
%* Output the listing                                                        *;
%*****************************************************************************;
options ls=126;
title "Vital Signs";

proc report data=work.vitlis headskip split = '~' spacing=4 nowindows;
  column sid vitstr hrtxt bp;
         define sid     / order   'Subject no.'        Spacing=0        width=7 ;
         define vitstr  / display 'Start date (day)'                    width=16;
         define hrtxt   / display 'Heart rate'         left             width=5 ;
         define bp      / display 'Blood pressure~(systolic/diastolic)' width=25;

  break after sid / skip;
run;
```

The changes to be made to this program, so that it can function as a stored process, are minimal.
Before the program starts outputting (proc report), the SAS reserved macro call “%STPBEGIN;”, which comes with the SAS System version 9 architecture, must be inserted in the program. When the programs stops outputting (run of proc report), the SAS reserved macro call “%STPEND;” must be inserted in the code. This tells the SAS System that the output is the output of a stored process, and works similar to the BASE SAS procedure “proc printto”. The snapshot of the code when changed to a stored process is listed below.

```
%*****************************************************************************;
%* Output the listing                                                        *;
%*****************************************************************************;
options ls=126;
title "Vital Signs";

%stpbegin; /* Re-direct the output */

proc report data=work.vitlis headskip split = '~' spacing=4 nowindows;
  column sid vitstr hrtxt bp;
         define sid     / order   'Subject no.'        Spacing=0        width=7 ;
         define vitstr  / display 'Start date (day)'                    width=16;
         define hrtxt   / display 'Heart rate'         left             width=5 ;
         define bp      / display 'Blood pressure~(systolic/diastolic)' width=25;

  break after sid / skip;
run;
%stpend; /* Close re-direction*/
```

Given the example above, the difference in validating a “regular” SAS program and a stored process is minimal. In general you could state that all output created with a SAS program producing static output, will require minimal changes to be transformed into a stored process, and therefore the impact on validation is minimal. When desired the portal is capable of creating a SAS log of the executed process.

Of course the portal offers a multitude of possibilities for creating dynamic output, some of which are described in this paper. This would require an alternative approach to receive the status of validated program. However, since these programs are used to gain insight into the data and assist with reviewing rather than for reporting to authorities, validation is less of an issue. Hence it is important to differentiate between programs used to create the actual study report and programs used to support this process.

SECURITY

Due to the confidential nature of most clinical data, the topic of security is also important. For this paper two types of security will be divided in to two types of security: access to the portal and access within the portal. Because the portal is available through a web browser, it could be made accessible anytime, anywhere. How this can be secured has more to do with accessing secure sites (for example https) than with the portal itself, therefore this will not be discussed within this paper.

ACCESS WITHIN THE PORTAL

Access to the portal itself is granted via a logon procedure. The user has to specify username and password which are verified by the same procedure as when the user would log on to the company network. The portal makes use of Single Sign On, this means that once logged on, no credentials are requested until the session ends when the user logs off, or by means of a session time-out.

One of the advantages of the SAS Information Delivery Portal is that the security within the portal is regulated though the SAS metadata architecture. This makes maintenance easy, and assists in solving the issue of access to data and securing the execution of SAS programs.

The security is based on users and user groups via the metadata server. This security can be applied to all portal elements such as: portal pages, portlets and portlet content. If a user has no access rights to a specific item, the item is not visible for this user. This means that the content of the same portal page can differ between users.
But this security can also apply to SAS libraries, SAS data and even columns in SAS datasets.
For our portal we have created several groups and users. An overview can be found in the section “users of the portal”.

TIPS

When implementing the portal environment the following tips could be of assistance:

- With regard to the basic structure of the portal; the content and security could be derived from existing Standard Operation Procedures as the portal is just a means to execute existing procedures;
- Demonstrating a “proof of concept” portal before the actual development assists future users in creating a better understanding of the possibilities which usually results in much more concise user requirements;
- When security is based on user groups instead of single users the maintenance becomes easier;
- When generating stored processes, make a clear distinction between programs used to create the actual study report and programs that create dynamic output for reviewing purposes only. This will decrease the validation effort required;
- We have found it is beneficial to ensure your portal development team is used to work in a regulated environment;
- When applying the use of metadata such as to store derivation rules, it is important that there are standards in place; preferably a metadata librarian is appointed;
- When developing stored processes, define the purpose upfront. When taking advantage of the web, the stored process cannot be executes through other SAS clients;
- When developing stored processes for the web and for SAS clients, differentiate these processes by assigning a descriptive subfolder name like “web-processes” and “general-processes’ within the SAS metadata server.

CONCLUSION

When integrating departmental information the SAS® Information Delivery Portal can deliver added value to the daily process of the pharmaceutical sector by making information available to a user that would otherwise not, or not easily be available, thereby increasing efficiency and delivering even more quality to the process. As illustrated in this paper even the smallest examples such as the metadata on the derivation rules can deliver great value add.

ACKNOWLEDGMENTS

We would like to take the opportunity to thank the following people for contributing to our paper:

- Egbert Biesheuvel, Organon NV;
- Bas van Bakel, OCS Consulting;
- Yves Poriau, OCS Consulting.

RECOMMENDED READING

Detailed information on the examples mentioned in the topic “The value of metadata” is described in: Phuse 2007 paper TS08 “The value of metadata” – Bas van Bakel, OCS Consulting

Detailed information on the examples mentioned in the topic “Taking advantage of SAS on the web” is described in: Phuse 2007 paper CC03 “Web techniques for SAS ODS output and more” – Jules van der Zalm, OCS Consulting

SAS® 9.1.3 Management Console, User’s Guide
SAS® 9.1.3 Open Metadata Interface, User’s Guide
SAS® 9.1.3 Integration Technologies, Technical Overview
SAS® 9.1.3 Integration Technologies, Administrator's Guide, Fourth Edition

CONTACT INFORMATION

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

Author Name	:	Raymond Ebben Jules van der Zalm
Company	:	OCS Consulting
Address	:	PO BOX 490 5240 AL Rosmalen the Netherlands
Work Phone	:	+31 (0)73 523 6000
Fax	:	+31 (0)73 523 6600
Email	:	sasquestions@ocs-consulting.com
Web	:	www.ocs-consulting.nl