

## Seamless Web Data Entry for SAS® Applications

D.J. Penix, Pinnacle Solutions, Indianapolis, IN

### ABSTRACT

For organizations that need to implement a robust data entry solution, options are somewhat limited on the SAS® platform. The SAS/FSP® module provides integrated tools for data entry, computation, query, editing, validation, display, and retrieval; however, the solution falls short for organizations that need to deploy a web-entry system on a thin-client platform (i.e. Internet Explorer, Mozilla Firefox, etc.) Some companies will tackle the problem using other technologies such as HTML, .NET, or JAVA. But each of these approaches requires expertise that is often beyond the typical SAS programmer to build and maintain. Built upon an open-source platform, a simple, yet elegant, data entry solution can be obtained. The application data can then be integrated into any SAS application by connecting it via the SAS/ACCESS® modules.

### TYPICAL WEB DATA ENTRY FEATURES

Although requirements for a web data entry system may vary from company to company, there are several features that are typically requested. Based on informal discussions with various customers, the following items were at the top of the list:

- 100% thin-client data entry using a browser such as Internet Explorer or Mozilla Firefox.
- Ability to enter data on customized data entry screens that simulate business forms.
- Ability to use color and highlighting to enhance the screen display.
- Ability to validate and cross-validate data automatically or define custom error-checking logic.
- Ability to compute new field values--instantly--based on values entered by the user.
- Ability to browse and edit data in both spreadsheet (horizontal) and column (vertical) form.
- Ability to capture audit trail information.
- Ability to view and edit data from SAS data files or external databases such as Oracle, DB2, Sybase, MS SQL, etc.
- Ability to simplify data entry using selection lists or table look-up. When a user enters information in one field, the web data entry form should automatically pull in information in other fields.
- Easy to maintain without SAS expertise.
- Ability to leverage the back-end data with their SAS environment to produce analytics, reporting, or any other use.

### SYSTEM ARCHITECTURE

For some organizations, a data entry solution based on the SAS/FSP<sup>1</sup> software is adequate. This solution does allow organizations to build complex data entry screens that mimic complex business data forms and using the power of SAS' database connectivity features, can access almost any warehouse on the back end. However, the solution is limited in that it is a "client" based system (which requires SAS to be installed on a desktop) and it is not easily scalable for large organizations that may have users spread throughout multiple locations. A simple, scalable, web-based solution would be more desirable.

So what architecture and software options are available for organizations looking to get started building web applications? One such option is to look at a bundled solution called LAMP. LAMP is an acronym for a solution stack of free and/or open source software, originally coined from the first letters of Linux (operating system), Apache (HTTP Server), MySQL (database software), and PHP (scripting language), principal components to build a viable general purpose web server.<sup>2</sup> WAMP is similar, except instead of Linux, it is supported on the Windows operating system. There are other variants as well, such as MAMP (Mac OS), SAMP (Solaris), and OpAMP (OpenBSD).

Although the original authors of these programs did not design them all to work specifically with each other, the development philosophy and tool sets are shared and were developed in close conjunction. The software combination has become popular because it is *free of cost*, mostly *open-source*, and therefore *easily adaptable*.

The stacks are widely used because it offers a great number of advantages for developers. It is easy to code, easy to deploy, you can develop it locally and then push it to the web, and it is available on almost every web host. Even the cheapest web host options

---

<sup>1</sup> SAS/FSP Product Page:

<http://www.sas.com/products/fsp/index.html>

<sup>2</sup> Wikipedia LAMP:

[http://en.wikipedia.org/wiki/LAMP\\_\(software\\_bundle\)](http://en.wikipedia.org/wiki/LAMP_(software_bundle))



value sources, labels, validation rules, etc. When documented in a spreadsheet (or within a web data entry application itself!), this process can help facilitate the requirements gathering process.

The following spreadsheet shows an example of collecting metadata items for a sample healthcare questionnaire.

ID	NAME	SOURCE	TYPE	FORMAT	LENGTH	COMMENTS
1	PATIENT ID	Healthcare system database	TEXT		10	
2	DATE OF BIRTH	Healthcare system database	DATE		8	
3	SEX	Healthcare system database	TEXT		1	
4	DOB	Healthcare system database	DATE		8	
5	HEIGHT	Healthcare system database	TEXT		5	
6	WEIGHT	Healthcare system database	TEXT		5	
7	HAIR COLOR	Healthcare system database	TEXT		10	
8	EYES	Healthcare system database	TEXT		10	
9	SKIN	Healthcare system database	TEXT		10	
10	HAIR	Healthcare system database	TEXT		10	
11	HAIR COLOR	Healthcare system database	TEXT		10	
12	HAIR COLOR	Healthcare system database	TEXT		10	
13	HAIR COLOR	Healthcare system database	TEXT		10	
14	HAIR COLOR	Healthcare system database	TEXT		10	
15	HAIR COLOR	Healthcare system database	TEXT		10	
16	HAIR COLOR	Healthcare system database	TEXT		10	
17	HAIR COLOR	Healthcare system database	TEXT		10	
18	HAIR COLOR	Healthcare system database	TEXT		10	
19	HAIR COLOR	Healthcare system database	TEXT		10	
20	HAIR COLOR	Healthcare system database	TEXT		10	

Figure 4 - Setting up the Metadata

Items such as variable names, labels, formats, types, and property, can all be defined and used as a road map to develop the PHP code.

The metadata can also be coded in SAS to automate the generation of the physical data sets. Standard (i.e. DATE9.) or custom (i.e. SDNAS.) SAS formats can also be specified and utilized in both the PHP code, as well as created and stored for the SAS applications.

ID	NAME	SOURCE	TYPE	FORMAT	LENGTH	COMMENTS
1	PATIENT ID	Healthcare system database	TEXT		10	
2	DATE OF BIRTH	Healthcare system database	DATE		8	
3	SEX	Healthcare system database	TEXT		1	
4	DOB	Healthcare system database	DATE		8	
5	HEIGHT	Healthcare system database	TEXT		5	
6	WEIGHT	Healthcare system database	TEXT		5	
7	HAIR COLOR	Healthcare system database	TEXT		10	
8	EYES	Healthcare system database	TEXT		10	
9	SKIN	Healthcare system database	TEXT		10	
10	HAIR	Healthcare system database	TEXT		10	
11	HAIR COLOR	Healthcare system database	TEXT		10	
12	HAIR COLOR	Healthcare system database	TEXT		10	
13	HAIR COLOR	Healthcare system database	TEXT		10	
14	HAIR COLOR	Healthcare system database	TEXT		10	
15	HAIR COLOR	Healthcare system database	TEXT		10	
16	HAIR COLOR	Healthcare system database	TEXT		10	
17	HAIR COLOR	Healthcare system database	TEXT		10	
18	HAIR COLOR	Healthcare system database	TEXT		10	
19	HAIR COLOR	Healthcare system database	TEXT		10	
20	HAIR COLOR	Healthcare system database	TEXT		10	

Figure 5 - Metadata Custom Formats

Although Pinnacle Solutions has developed utilities to read the example metadata and automatically generate corresponding PHP code, one could also replicate this process either manually, or automatically using web application development software such as Adobe® Dreamweaver® CS5 or Microsoft® Expression® (formerly Microsoft® FrontPage®). But regardless of how this information is converted over into the final web entry application, defining the specific user requirements is always recommended.

Once you have developed your PHP pages, you can simply launch a web browser to test your application. The remainder of the paper will show various examples of what can be done using our in-house developed product called PINNACLE COMPASS™. The application was building using the WAMP stack.

## WEB DATA ENTRY COMPONENTS

The first component of a data entry system that needs to be defined is security. The PINNACLE COMPASS solution utilizes its own security tables that contain fields such as username, password (encrypted), active/deactivated, group membership, and email. These security tables are simply maintained within the MySQL database; however, it could also be maintained in a separate LDAP database. The COMPASS solution can even be connected to Microsoft Active Directory to allow corporate network credentials to be used. Although not necessarily required within all organizations, the COMPASS solution even supports email reminders for users that forget passwords, as well as provide a built-in mechanism for first-time users to register a request for system access. With this feature, an automated email is sent to an administrator, and then access can be granted or denied by management.

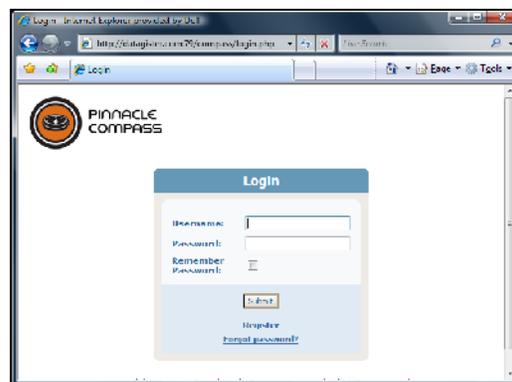


Figure 6 - User Logon Screen

Once logged into the system, the user is redirected to the primary data entry page.

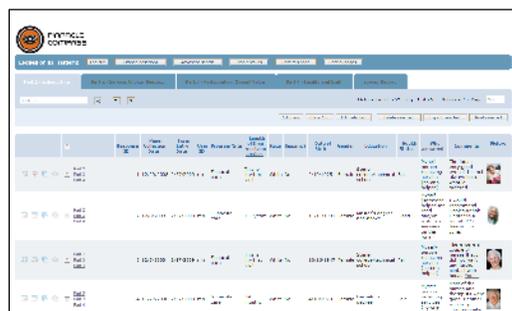


Figure 7 - Sample Data Entry Form

There are various components of the page that will allow the user to quickly navigate and work with the data. At the top row of the screen, the following buttons are available:



Some of these high-level actions will be discussed later in this paper, however starting from left to right,

the user can log out, change the password, conduct an advanced search, export the results, print the current page, or print all pages.

Immediately below those buttons are tabs that allow the user to view different components of the example survey or even view custom HTML pages. In this example, there are actually 4 parts to the survey and a custom report at the end.



The default entry screen is displayed in horizontal format (many records in a row-by-row fashion) however the same form can also be displayed in vertical format (single record on one screen).

Each record in the horizontal layout has several quick click icons.



The edit icon allows the user to edit the field, as displayed in the following screen shot.

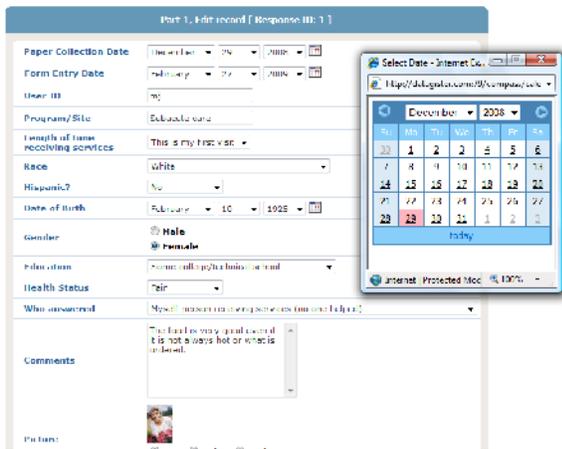


Figure 8 - EDIT Screen with Date Picker

The in-line edit icon allows the user to do an in-line edit of one or more records. The following screenshot shows an example of clicking the in-line edit of the first record.

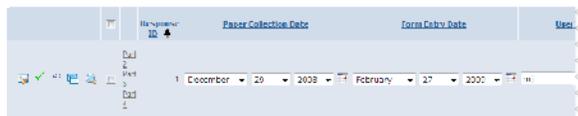


Figure 9 - In-Line Edit Record

After the in-line edit icon is clicked, the icon changes to display two new icons: a checkmark and a circular arrow. These icons allow the user to either save (check mark button) or cancel (undo button) the changes to the data. For that record, the data itself turns into an editable value.

The copy icon allows the user to copy an existing record. For this example, "Response ID" is the key field and is set up as auto-number in the database. It will be automatically assigned in MySQL to ensure uniqueness, however all other fields will automatically be populated with duplicate values.

The browse icon allows the user to simply browse a record without putting it into edit mode. The default layout is in vertical fashion.



Figure 10 - Browse Mode

The select icon allows the user to select one or more records. Alternatively, by clicking on the select box at the top of the screen, the user can select all records at once. The multiple select record function is useful for performing in-line edits on more than one record at a time. The row of buttons immediately above the data will allow the user to Edit, Delete, Export, or Print the selected records.



The following screenshot shows the result when the 1st and 3rd records were selected, and then the "Edit selected" button was clicked.

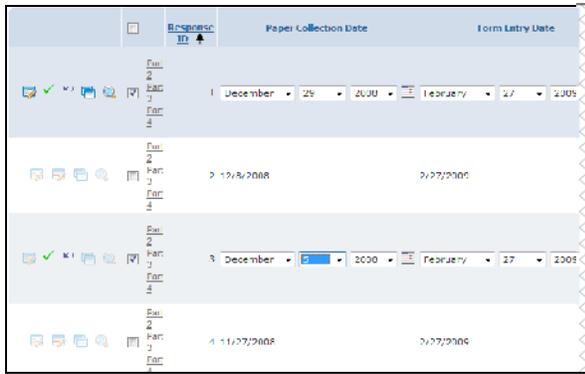


Figure 11 - Multiple Selection with In-Line Edit

The survey links [Part 2](#), [Part 3](#), and [Part 4](#) follow the selection boxes. The following screenshot shows the results of clicking on the "Part 2" link for an individual record.



Figure 12 - Viewing Multiple Parts

This function allows the user to see multiple parts of the survey simultaneously, which can be useful for understanding the context of the different data pieces. The user can also immediately proceed to Part 2 of the survey, or by clicking on the red X, can cancel the view and return to normal viewing mode.

Other features of the application include the ability to export the data, either for specifically selected records, or the entire database. Export formats include Excel, Word, CSV, or XML.

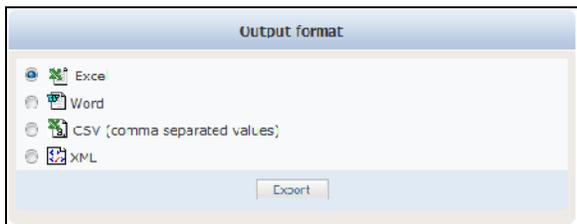


Figure 13 - Export Formats

The user can also print selected records. Although any complex reporting needs would be better served

in SAS, it is a handy feature for doing a quick print for a handful of records.

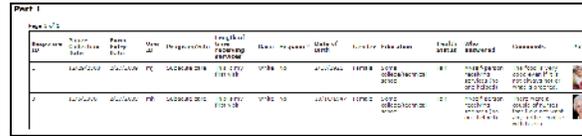


Figure 14 - Printing Selected Records

The COMPASS solution also has some AJAX<sup>7</sup> based functionality implemented within the PHP making available some useful features such as the ability to search information more easily with the Google-like auto-suggest feature. The web pages with AJAX-driven dependent dropdown boxes are also loaded faster.

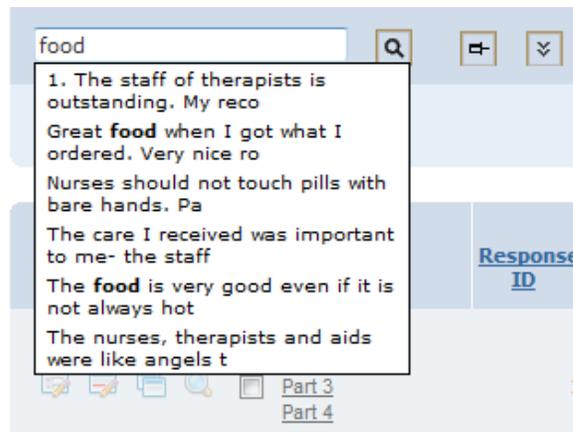


Figure 15 - Ajax Based Searching

There is also an advanced search window that permits users to perform more complex queries.

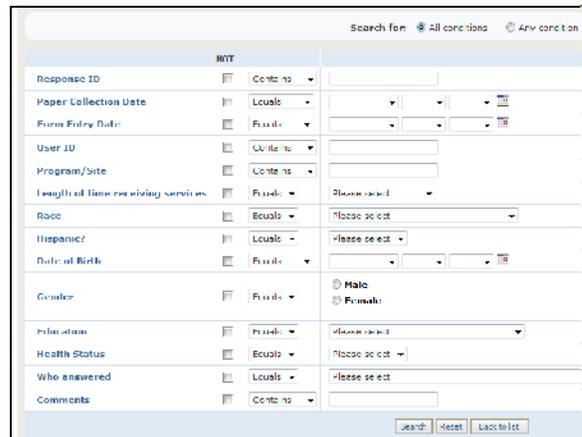


Figure 16 - Advanced Search

<sup>7</sup> Wikipedia Ajax: [http://en.wikipedia.org/wiki/Ajax\\_\(programming\)](http://en.wikipedia.org/wiki/Ajax_(programming))

And, the user can even create their custom search criteria in a floating window.

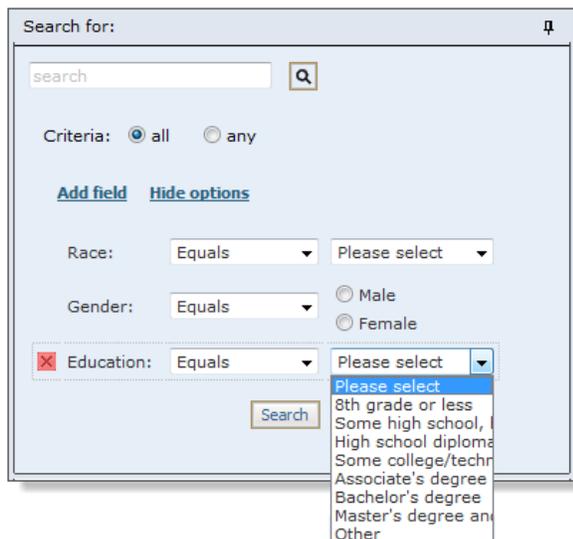


Figure 17 - Undocked Customized Floating Window

Other features of the application include audit-trail logging, sorting the data, and automatic record locking to prevent multiple users from editing the same record simultaneously.



Figure 18 - Record Locking Feature

The MySQL database can even support BLOB<sup>8</sup> (Binary Large Object) data types, which are ideal for scanned images such as photographs, case report forms, x-rays, or other relevant documents.



Figure 19 - BLOB Object for Binary File Types

<sup>8</sup> Wikipedia BLOB: [http://en.wikipedia.org/wiki/Binary\\_large\\_object](http://en.wikipedia.org/wiki/Binary_large_object)

## INTEGRATION WITH SAS

As you now see, a robust web data entry system can be built independently of SAS. By using widely available free software stack such as Apache web server, MySQL, and PHP, you can build complex data entry screens for the web. Once you've collected the data, you can fully take advantage of the reporting and analytics features of SAS.

To surface the data into SAS, you simply need to have the corresponding SAS/ACCESS<sup>9</sup> technology licensed. In this example, one would only need to have SAS/ACCESS Interface to MySQL in order to get access to the data. The screenshot below shows the data being browsed in Enterprise Guide or any other SAS application.

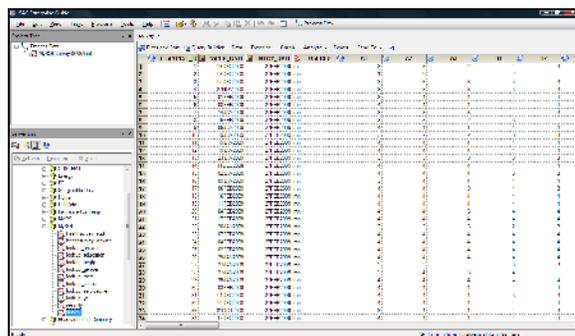


Figure 20 - Accessing MySQL Data in SAS

Notice that the raw data and variable names correspond to the variable names defined in the metadata worksheet. The underlying data is all stored as numeric values, even though the users see drop down lists with the business label descriptions that are more intuitive.

	A1	A2	A3	B1
	3	3	2	4
	1	2	.	2
	3	3	3	3
	3	3	3	3
	3	4	4	4

Figure 21 - Raw Data Unformatted

A SAS view can easily be created that applies the labels and formatted values. The following is a screenshot of a SAS view within Enterprise Guide<sup>®</sup>.

<sup>9</sup> SAS/ACCESS Product Page: <http://www.sas.com/technologies/dw/etl/access/index.html>

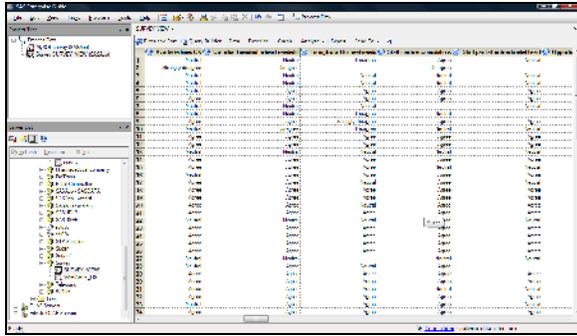


Figure 22 - SAS View of MySQL Table

And here is a close up view of the formatted values and column labels.



Figure 23 - Raw Data Formatted

But what if you don't have any SAS/ACCESS engines licensed at your organization? The COMPASS solution can also connect to any ODBC connection source. You can download the SAS ODBC<sup>10</sup> driver from the SAS support website and configure the ODBC with your SAS installation. Some features such as the BLOB data format will not be available. However, other features such as record locking are still available because the COMPASS solution maintains its own record locking table.

SAS can also access the audit trail table to enable reporting for corporate compliance or regulatory agencies. The following screenshot shows an example audit record.

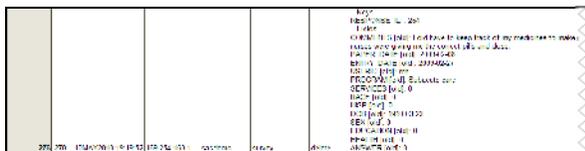


Figure 24 - Audit Record

Other procedures such as PROC UNIVARIATE can immediately produce descriptive statistics to provide insight into the data.

<sup>10</sup> SAS ODBC Driver Download: [SAS Support Website](http://www.sas.com/techsup/notes/n1111022.html)

PINNACLE COMPASS, and all other Pinnacle Solutions, Inc. product or service names are trademarks of Pinnacle Solutions, Inc. in the USA and other countries.

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.

Distribution analysis of: Staff made accommodations  
The UNIVARIATE Procedure  
Variable: H1 (Staff made accommodations)

Basic Statistical Measures			
Location		Variability	
Mean	3.476596	Std Deviation	0.64258
Median	4.000000	Variance	0.41291
Mode	4.000000	Range	3.00000
		Interquartile Range	1.00000

Basic Confidence Limits Assuming Normality			
Parameter	Estimate	95% Confidence Limits	
Mean	3.47660	3.39101	3.55918
Std Deviation	0.64258	0.50926	0.70659
Variance	0.41291	0.34729	0.49727

Tests for Location: Mu=0			
Test	Statistic	Pr >  T	p Value
Student's t	82.93917	Pr >  T	<.0001
Sign	M	117.5	Pr =  M  <.0001
Signed Rank	S	13885	Pr =  S  <.0001

Missing Values			
Missing Value	Count	All Obs	Percent Of Missing Obs
.	17	6.75	100.00

Figure 25 - PROC UNIVARIATE Results

## CONCLUSION

Although there are many approaches available for building data entry forms on the web, they usually involve a high level of programming expertise or lack the robust functionality of a mature system. Using a readily available and free software stack of tools such as Apache, MySQL, and PHP, one can quickly and easily develop sophisticated thin-client data entry solutions. Once the back-end data has been established, SAS can get access to the data using one of the SAS/ACCESS technologies. Alternatively, one could build a data entry solution using the SAS ODBC driver and input the data directly into SAS datasets if a 3rd party database was not preferred.

## CONTACT INFORMATION

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

Name: Donald L. Penix, Jr. (D.J.)  
 Enterprise: Pinnacle Solutions, Inc.  
 Address: 120 East Market Street  
 Suite 900  
 Indianapolis, IN 46204  
 Work Phone: 317-423-9143 x801  
 Fax: 317-348-1633  
 E-mail: dj.penix@psiconsultants.com  
 Web: http://www.psiconsultants.com