INTRODUCTION
SAS/PH-Clinical 2.0 provides completely new technology for clinical data review for Computer Assisted New Drug Applications (CANDAs) and internal review of clinical trials. As discussed at SUGI19, this new release will be available with Release 6.11 on all PC and Unix desktop environments, including the Macintosh and Power PC architectures. This major new release provides significant new functionality in a Graphical User Interface (GUI) environment. Both the functionality and interface have undergone significant usability testing to ensure ease-of-use by reviewers in the pharmaceutical, biotechnology, and regulatory arenas. This paper provides an update on the software and includes several figures that illustrate the functionality and the interface. The table below summarizes key enhancements of SAS/PH-Clinical 2.0.

POINT AND CLICK INTERFACE
The best way to describe the new graphical user interface for SAS/PH-Clinical 2.0 is to provide examples. Figure 1 displays one way the software can appear on startup. (Note that all figures appear at the end of the paper. Also note that the interface uses color; these figures have been converted to grayscale for inclusion in the paper.) In this window, the active study is available for browsing or for reporting purposes. The hierarchical figure displays the user's structure of folders and sub-folders. The appearance of items in the window is a function of user preferences, which are specified by choices from the Preferences item on the action bar.

This window illustrates a key principle for the GUI: when the software starts, you can just start to work. The new interface is easy to learn and does not surface power unnecessarily. The power and customization are available, but you can begin to work with only a few features.

Note that there may be interface differences between the production release and the figures shown in this paper. The software is currently in the final stages of preparation for an experimental release, and may change based on customer feedback and additional usability testing. The next several sections provide additional detail on the new user interface.

ADHERENCE TO WINDOWING GUIDELINES
By following available interface guidelines, the SAS/PH-Clinical 2.0 interface is easier to learn since it leverages knowledge gained from other activities such as word processing, spreadsheets, or other familiar tools. Examples include providing toolbars, default double-click actions that perform as users expect, and familiar choices on the command bar.

ORGANIZING INFORMATION
The new user interface reflects an increased understanding that each user wants to organize information according to their needs, rather than following a standard paradigm. Each user chooses to display items in a way that makes sense to them. For example, users can place all their items—reports, graphs, queries, cohorts, and so on—in a single folder. Other users can choose to organize information according to study; or according to how the results are used (for safety or efficacy analyses, for example). Users can build folders within folders, and combine different types of items in a single folder.

For example, Figure 1 displays a window where the items are first organized according to study, and organized into safety, efficacy, and general reports. Figure 2 shows the results of double-clicking on the Studies folder to collapse the organization; this is especially useful when you have multiple layers of folders. Figure 3 displays the results of opening one of the folders.
The folders are not only a way to display items together, they are a way to operate on a group of items. You can move, copy, rename or delete an entire folder or individual items. You can easily change the structure of folders and sub-folders with drag-and-drop technology. Using this same feature, you can link to another user's library, access her reports and graphs, and drag them to one of your folders. This establishes a dynamic link that gives all users an easy way to share information with other users.

In addition, you can use drag-and-drop technology to copy individual items or folders. This is available within and across windows. For example, you can reorganize the structure of folders in your own library by dragging and dropping folders to indicate the structure you want. Or, you can drag another users' folder and drop it on one of your folders to enable you to access all the items in the folder.

USING THE BROWSE ENVIRONMENT
Both medical and statistical reviewers at sponsor firms and regulatory agencies are comfortable with viewing data in a spreadsheet format. Often the reviewers are concerned with specific parts of the study. SAS/PH-Clinical 2.0 accommodates these needs, and lets users focus on the variables of interest rather than on datasets.

COHORTS AND VARIABLE GROUPS
An important concept in browsing clinical data is that of a cohort, which is simply a group of patients. Figure 4 shows the results of selecting the cohorts item on the toolbar. This window displays the list of available cohorts and allows you to switch between cohorts. By selecting the Patients folder tab, you can view all the patients in the cohort. (Another way to do this is to choose the Patients item in the toolbar in the main window.) Figure 5 shows the list of patients. From this window, you can access the patient profile or CRF images for any patient or group of patients.

Another important concept in browsing clinical data is that of variable groups. Figure 6 shows the variable groups for a sample study. You can manage this hierarchy of variable groups in the same way you organize folders and items within them. You can add, delete, or rename groups, or change the structure of the groups. Figure 7 shows the results of collapsing the hierarchy. Notice that some of the groups are editable (indicated by the edit icon) and some can simply be browsed.

The editable groups have been created by the user; browse-only groups were created when the study was defined. Since variables can belong in multiple groups, users can create their own groups that, for example, combine variables from several of the browse-only variable groups. To do so, users select the variables in the right panel and after completing selections, drag the variables to the variable group in the left panel. Depending on choices from the action bar, the right panel can display all variables, variables in the current group, or variables not in the current group. Two key points are that variables can belong to multiple groups, and that a variable group can contain variables from many datasets.

In this window, you can double-click on a variable to obtain summary statistics without opening a spreadsheet. You can also plot or graph the variable by selecting it and then selecting a toolbar item (or a choice from the action bar). In addition, you can access available CRF images.

SPREADSHEETS, GRAPHS, AND STATISTICS
After choosing variable groups—or just specific variables—of interest, you can click on the Tables icon in the toolbar to open a spreadsheet. Figure 8 displays a spreadsheet for selected variables. Notice that the variable labels appear at the top of each column; this is the result of preferences set with the Preferences menu bar choice.

Browsing data typically generates a series of questions that need quick answers. Looking at a column of numbers often begs for a graph for quick visual interpretation—or for a set of summary statistics. For example, double-click on a column to generate summary statistics for the column. Figure 9 shows the results for the Height variable. Notice the Summary and Image buttons; these illustrate an important principle for the software. From anywhere in the Browse environment, you can access all the data for patients. Suppose you want to drilldown on the patient with a height of -5. You can either select the -5 in the frequency table and then select the Summmary button; or double-click in the frequency table. This displays the patient profile, shown in Figure 10. Using the drop-down lists at the top of the window, you can easily view any information for the patient. In this example, the spreadsheet contains demographic information, and the user has switched to reviewing the lab information in the patient profile. From the summary window, you can also view any available CRF images in a similar manner.

From the spreadsheet, you can also select columns and then select a graph—either from the toolbar or from the pull-down menu choices. Clicking on the Scatterplot toolbar item and then selecting variables of interest generates the plot shown in Figure 11. You can select points on the plot and drilldown to the summary or CRF images, or exclude the point from the plot. Figure 12 shows the results of excluding the outlier, and adding a class variable to indicate the patients' surgery code. In addition, by-processing is available to enable you to generate multiple plots for the different levels of the by-variables. Figure 13 shows one of the plots generated by requesting the Protocol and Investigator as by-variables. You can scroll through the plots using the up- and -down arrows, or scroll to a particular plot using the choices from the Values drop-down list. You can also change plots simply by selecting a different plot on the toolbar. Figure 14 shows a horizontal bar chart for the patient height variable.

Returning to the spreadsheet, you can also select rows or cells and view the associated patient profile and available CRF images. Figure 15 shows the results of selecting the row for patient 14 and then viewing the images; this example displays the first image, which contains demographic information. (Notice that the spreadsheet in the background window has been sorted by height, and the lowest height value (-5) selected.) Using the drop-down lists to the right of the image, you can scroll between the various images. Had you selected multiple patients, you could scroll through the patients using the first drop-down list.
viewing. Available features include zooming, mirroring, rotating, and reverse video. In all cases, the original image is not modified; you are simply changing the view of the accessed image.

**ADDITIONAL FEATURES**

Some of the additional features not discussed in detail include the ability to:

- add new variables; for example, sums across columns, differences between columns, means, and more.
- add new variables based on a key variable; for example, a new variable that contains the average blood pressure across visits.
- obtain summary statistics based on a key variable; for example, to obtain the summary statistics for lab variables separately for each visit.
- dynamically rotate the spreadsheet. For example, if a spreadsheet contains one row for each patient’s lab test at each visit, rotation produces a table with one row for each visit, where the columns identify the lab tests and the cells contain the results for the test.
- create simple and complex queries to further subset the patient cohort. Features include the ability to build multiple clauses, join the clauses with combinations of AND and OR, and to use either a value list or simply type in the values.
- appearance enhancements, including the ability to control the fonts and colors in the windows.

This section has given an overview of the functionality and interface for the Browse environment in SAS/PH-Clinical 2.0. There are, however, additional features under development.

**REPORTING ENVIRONMENT**

SAS/PH-Clinical 2.0 combines standard and ad-hoc review tasks by providing a library of reports (including many graphical reports) with the software. This library is a reflection of increased communication between sponsors, regulatory agencies, and the Institute about the types of review tasks performed by medical and statistical reviewers. However, no predefined library can encompass all the analyses performed on clinical data. Thus, in addition to the Institute-provided reports, the powerful new PH-Templates environment has been developed. This accomplishes the need for customization and integration of existing systems, and does so while requiring no knowledge of SCL or FRAME. This section first briefly discusses sample reports and then discusses the PH-Templates environment in more detail.

**SAMPLE REPORTS**

Figures 16-22 show some of the reports provided with the software. Each set of figures shows a window where the user has provided details, and then shows the resulting output. A key aspect of the Reporting environment is that the program, log, and output are all automatically linked; ensuring that when you save the output from a report, you automatically save the associated SAS program and log.

The figures provide only a small sample of the available reports. For information on the complete set of reports provided with the software, a Vertical Products White Paper "SAS/PH-Clinical 2.0 Reporting Functionality" (unpublished) is available from SAS Institute.

**USING THE PH-TEMPLATES ENVIRONMENT**

As a brief technical summary, the PH-Templates environment provides you with the ability to create generic reports that can run against any study without making any changes to the underlying SAS programs. It incorporates a program editor with a process that allows all dataset and variable names to be coded as substitution fields rather than requiring you to supply the actual dataset and variable names. When the program runs, these fields are replaced with the correct dataset and variable names for the current study, resulting in a correct program for the current set of clinical data.

Further, if your program requires several variables to be in the same dataset and this is not the case, the software dynamically builds the dataset containing the required variables. This is possible because of the underlying study dictionary.

The PH-Templates environment also enables you to associate a user interface with each program. This is especially useful when you want to add new ad-hoc reviewing tasks to the software. You can define user windows that display when users select the report from the desktop. In the windows, users select variables and supply other details for the ad-hoc review task. For SAS programmers, using the PH-Templates environment to design a report or graph consists of supplying the SAS procedure code and designing the user window. However, this design process does not require you to know SCL. Note that user windows are not required; you can build reports that simply run when a user double-clicks on the icon in the desktop.

A final, but vital, aspect of the resulting report is the integral linking of the program, log, and output. When users create a report and save the results, they are also automatically saving the SAS program used to generate the report and the SAS log created as the report was produced. This ensures that, at any time, users can regenerate the report using the same SAS code that was originally used, can check the SAS log for notes and other information, and can thus easily validate the results.

**EXAMPLE**

Consider a simple printing task. Suppose the program for printing adverse event information from a given study is as follows:

```sas
proc print data=mylib.adverse noobs;
  id patnum;
  var trt aecode aedesc severity related recur hosp death;
  format aecode $whocode. trt $TRT.;
  title 'Adverse Events Summary';
run;
```

This program may work for one or more studies, but perhaps not for all studies, given the inconsistencies in variable names, dataset structures, and the like. Or, you may want to add other variables to the report, such as additional demographics or related concomitant medications. If you were to revise the program above to use macro variables, it might look as shown below.
This is exactly the same code that converts the program to a PH-Template. The familiar &-variables are not macro variables, but instead indicate substitution fields, whose values are supplied from the underlying dictionary.

Incorporating existing programs using the PH-Templates environment is an easy process: copy them into the program editor and change a few of the hard-coded dataset and variable names to substitution fields, and then create a user window if you want one. If the programs are written as macros, conversion is even easier because the substitution fields resemble macro variables.

**DICTIONARY TECHNOLOGY**

Data from a clinical trial are typically stored in several datasets. Each dataset has a number of keys, such as the patient identifier, visit number, laboratory test, and so on. These keys are used to identify records in the datasets and to enable datasets to be joined. The software is designed to access virtually any dataset and key structure. With equal ease, you can define parallel-group and crossover studies, single-center and multi-center trials, and trials with simple or composite keys. Normalized datasets and their associated keys are the preferred data structures. Datasets can be defined as datasets with multiple keys does not adversely affect performance.

An important aspect of both the product and study dictionaries is that they are dynamic. Whenever users create a new variable (perhaps by adding a new column to the browse spreadsheet and saving the results), the dictionary retains new information just as if it had been supplied during the study definition process. Thus, medical reviewers can essentially enhance the clinical datasets without modifying the original underlying data structures. For example, users can add a new variable that reclassifies patients' evaluability status—by first copying the existing evaluability variable and then modifying the status for individual patients. Both evaluability variables are defined in the study dictionary, and can be used in subsequent review tasks. As another example, users can transpose a spreadsheet view of the data and save the transposed view. Both the original and transposed spreadsheets are now available for subsequent review tasks.

Perhaps most importantly, dictionary management is done without unnecessarily surfacing the underlying technical details on the structure of the clinical datasets to users. The study dictionary stores enough information to ensure that data manipulations required to complete user requests are performed intelligently and efficiently. Also, the dictionary stores commonly-used information about the data. This information might include: a list of patients, summary statistics for selected columns, adverse event counts for each patient, and so on. Access to this information can be performed very quickly because it is stored in the dictionary and does not need to be calculated each time.

For details on the SAS/PH-Clinical 2.0 dictionary technology, see the paper by Martin Michael (1995) in the SUGI20 proceedings.

**CLIENT-SERVER SUPPORT**

SAS/PH-Clinical 2.0 software requires read access to the clinical data for the data reviewer. The Browse and Reporting environments both generate joins of the clinical data to address each request to these components. These joins, called Subsets, are generated using standard DATA and PROC (including PROC SQL) statements which will execute correctly in all releases of SAS from 6.07 onwards. Further, reports built with the PH-Templates environment generate SAS language code which can also be submitted remotely. In this case however, it is possible that the code generated by a particular report may require more recent releases of SAS software.

You can define data remotely to the client session. In this case, you can choose whether a subset is created locally, by accessing the clinical data library through Remote Library Services (RLS); or remotely on the host. In the latter case, you can also choose to run reports against the remotely generated subsets, or to download the subset to the client for local processing. Note that the Browse environment requires the subset, and any CRF images, to be available locally to provide reasonable performance.

This architecture enables SAS/PH-Clinical 2.0 to support the majority of client-server configurations. SAS Release 6.06 or later must be installed on the server, SAS Release 6.11 and SAS/PH-Clinical 2.0 must be installed on each client (if an earlier SAS release is installed on the server), and the respective SAS sessions on the client and server communicate using SAS/CONNECT software.

It is advisable to consider performance of the software in the client-server model. Many factors impact on performance. Optimum performance is most likely in a structure where the clinical data and the software are on a local machine with large processing power, small load, and large disk storage. When SAS/PH-Clinical 2.0 is installed on the server (where the clinical data resides), the option of using the client machines as dumb terminals to the server can be considered. The table below summarizes performance issues.

<table>
<thead>
<tr>
<th>Performance Issues for Client-Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processing power, load and disk storage on the server</td>
</tr>
<tr>
<td>Processing power, disk I/O, and disk storage on the client</td>
</tr>
<tr>
<td>Volume and rate of data transfer between the client and the server</td>
</tr>
<tr>
<td>Volume of clinical data</td>
</tr>
</tbody>
</table>

719
SUMMARY

SAS/IPH-Clinical 2.0 provides a major advancement in clinical data review technology. Key features include a friendly interface; multi-functional browser; dynamic queries; fast data access; a rich set of reports, analyses, and graphs; and automated study definition. Further, you can incorporate additional functionality and can customize the interface according to your preferences. Key aspects of the software functionality and their relationship to the goal of achieving a single process for CANDA use and internal clinical data review are:

- The dictionary technology facilitates using a single data structure for all review tasks, rather than duplicating datasets to generate different reports and analyses.
- The user interface provides a common tool that meets the diverse interface needs of medical reviewers, statisticians, statistical programmers, data analysts, and regulatory reviewers; rather than requiring different tools for different reviewers.
- The Browse environment includes spreadsheet functionality, drilldown, integrated links to CRF images, and a visual query facility, which eliminates the need to anticipate all different views of the data that any reviewer might want to investigate.
- Browse provides the ability to dynamically transpose the spreadsheet, add new columns to the spreadsheet, switch to a graphical view of columns in the spreadsheet, and obtain summary statistics on columns in the spreadsheet. Since restructuring is dynamic, eliminates the need to anticipate the varying structures that help different reviewers attain a comfort level with the clinical data.
- The dictionary provides increased robustness in handling more types of dataset structures, including those with multiple levels of keys, which facilitates the ability to use dataset structures more typical of a database management system.
- The Reporting environment provides a rich set of reports, analyses, and graphs, including all functionality from the first release of the software.
- The PH-Templates environment provides a way for sponsors to leverage their own investment in custom reporting programs. By enabling sponsors to integrate their own custom reporting programs into SAS/IPH-Clinical 2.0, the software can be fully customized to the needs of specific reviewers at the sponsor firm or the FDA.

Authors Addresses
SAS Institute Inc.; SAS Campus Drive; Cary, NC 27513

Acknowledgments
The authors would like to thank other members of the SAS/IPH-Clinical 2.0 development team for providing input and reviewing drafts of this paper.

SAS and SAS/IPH-Clinical are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. © indicates USA registration. Other brand and product names are registered trademarks or trademarks of their respective companies.
Figure 3

Figure 4

Figure 5

Figure 6