Transitioning to SAS®
Enterprise Guide®

Michael Todd
Nth Analytics

September 15, 2008
Introduction

- Enterprise Guide is a very powerful programming environment
- However, it was designed as a tool for power users to run interactively
- This complicates using it to replace traditional SAS coding
- So we have the question: replacement, or adjunct?
Enterprise Guide Overview

- Enterprise Guide (EG) is a visually oriented SAS programming environment
- Enables programmers to create diagrams linking objects that EG turns into SAS code
  - PROC SQL instead of data step
  - Reporting PROCs are PROC TABULATE and PROC PRINT
Process Flow

- The main interface is the process flow.
- Initially, it is a blank screen on which to arrange the objects.
- The programmer sets properties for objects, and the objects generate SAS code “behind the scenes”.
- “Code objects” can also be included, containing user-written custom code
  - These can be linked to the built-in objects in a process flow.
Enterprise Guide Demo

- Start with a blank process flow
- Load the data from one of the libraries
Enterprise Guide Demo

- Build the query
Enterprise Guide Demo

- Build the SELECT statement by picking values from a list

<table>
<thead>
<tr>
<th>Value</th>
<th>Formatted Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABDOMINAL ABSCESS</td>
<td>ABDOMINAL ABSCESS</td>
</tr>
<tr>
<td>ABDOMINAL DISCOMFORT</td>
<td>ABDOMINAL DISCOMFORT</td>
</tr>
<tr>
<td>ABDOMINAL DISTENSION</td>
<td>ABDOMINAL DISTENSION</td>
</tr>
<tr>
<td>ABDOMINAL PAIN</td>
<td>ABDOMINAL PAIN</td>
</tr>
<tr>
<td>ABDOMINAL PAIN LOWER</td>
<td>ABDOMINAL PAIN LOWER</td>
</tr>
<tr>
<td>ABDOMINAL PAIN UPPER</td>
<td>ABDOMINAL PAIN UPPER</td>
</tr>
<tr>
<td>ABDOMINAL TENDERNESS</td>
<td>ABDOMINAL TENDERNESS</td>
</tr>
<tr>
<td>ABNORMAL BEHAVIOUR</td>
<td>ABNORMAL BEHAVIOUR</td>
</tr>
<tr>
<td>ABNORMAL DREAMS</td>
<td>ABNORMAL DREAMS</td>
</tr>
<tr>
<td>ABNORMAL FAECES</td>
<td>ABNORMAL FAECES</td>
</tr>
<tr>
<td>ABORTION MISSED</td>
<td>ABORTION MISSED</td>
</tr>
<tr>
<td>ABORTION SPONTANEOUS</td>
<td>ABORTION SPONTANEOUS</td>
</tr>
<tr>
<td>ABSCESS</td>
<td>ABSCESS</td>
</tr>
<tr>
<td>ACCIDENT</td>
<td>ACCIDENT</td>
</tr>
<tr>
<td>ACCOMMODATION DISORDER</td>
<td>ACCOMMODATION DISORDER</td>
</tr>
<tr>
<td>ACIDOSIS HYPERCHLORAEMIC</td>
<td>ACIDOSIS HYPERCHLORAEMIC</td>
</tr>
<tr>
<td>ACNE</td>
<td>ACNE</td>
</tr>
<tr>
<td>ACRODERMATITIS</td>
<td>ACRODERMATITIS</td>
</tr>
<tr>
<td>ACTIVATED PARTIAL THROMBOPLASTIN TIME PROLONG...</td>
<td>ACTIVATED PARTIAL THROMBOPLASTIN TIME PROLONG...</td>
</tr>
<tr>
<td>ACTIVITIES OF DAILY LIVING IMPAIRED</td>
<td>ACTIVITIES OF DAILY LIVING IMPAIRED</td>
</tr>
</tbody>
</table>
● EG generates the SELECT statement for you
Enterprise Guide Demo

- Run the process flow
PROC SQL;
  CREATE TABLE WORK.Query_for_AE AS SELECT AE.STUDYID FORMAT=$15.,
    AE.USUBJID,
    AE.SEX,
    AE.AGE FORMAT=BEST22.,
    AE.SAFETY,
    AE.ARM,
    AE.AEBODSYS,
    AE.AEDECOD,
    AE.AESEV,
    AE.AESER,
    AE.AEACN,
    AE.AEOUT,
    AE.PHASE
  FROM NTH.AE AS AE
    WHERE AE.SAFETY = "YES" AND AE.AEDECOD IN ("AXILLARY PAIN", "BACK INJURY", "BACK PAIN"
    "BENIGN PROSTATIC HYPERPLASIA", "BEREAVEMENT REACTION", "BLADDER DISORDER", "BLADDER SP"
    "BLOOD BILIRUBIN INCREASED", "BLOOD CALCIUM INCREASED", "BLOOD CHLORIDE DECREASED", "BL"
    "BLOOD LACTATE DEHYDROGENASE INCREASED", "BLOOD OSMOLARY DECREASED", "BLOOD PHOSPHORU"
    "BLOOD TRIGLICERIDES ABNORMAL", "BLOOD TRIGLUCERIDES INCREASED", "BLOOD UREA INCREASED"
    "BOWEL SOUNDS ABNORMAL", "BRADYCARDIA", "BRADYKINESIA", "BRADYPHRENIA", "BRAIN NEOPLASM"
    "BREATHE ODOR", "BREATHE SOUNDS ABNORMAL", "BRONCHIOLITIS", "BRONCHITIS", "BRONCHITIS AC"
    "BURSA DISORDER", "BURSITIS", "BUTTOCK PAIN", "CALCINOSIS", "CALCULUS URETERIC", "CALCU"
    "CARTILAGE INJURY", "CAT SCRATCH DISEASE", "CATARACT", "CATARACT OPERATION", "CATHETER"
    "CERVICOBRACHIAL SYNDROME", "CHALAZION", "CHANGE OF BOWEL HABIT", "CHEST DISCOMFORT");
QUIT;
EG as a Programming Environment

- Advantages:
  - Enterprise Guide generates better and more reliable code than programmers produce manually
  - Enterprise Guide enforces a standard programming style
  - SAS programmers can be developers while end users run the code.
  - Using stored processes, the source code can be secured and hidden from the end users.
  - Enterprise Guide can be deployed worldwide, allowing users to connect to a central SAS repository and run SAS programs without having SAS installed on their systems.
EG as a Programming Environment

- However, there are many pitfalls to go along with the advantages.
  - Introduction of a new programming environment requires substantial IT support.
    - Installation, Validation, FDA 21 CFR Part 11 compliance.
  - Move away from stable programming environment
    - Can’t assume easy upgrade path
    - Requires archive copies be maintained
Considerations

- This is a completely new environment
- Not like installing a new version of SAS
- Will require full IQ/PQ/OQ
- What do to with old programs
  - Is it too big a transition?
  - Start fresh with a new project?
  - Incremental?
  - Same problem as with installing any new reporting system
- Steep learning curve
More considerations

- EG is designed as a tool for power users to run interactively.
- Therefore, the easiest transition is to provide a new environment for statisticians to run their programs interactively.
- EG can replace the traditional production batch environment, but this takes some doing
  - You are asking EG to do something it was not designed to do
  - But there are many advantages to doing so
Advantages: PROCs

PROC Tabulate and PROC SQL
PROC TABULATE

- One of the built-in tasks generates PROC TABULATE code
  - Summary Tables
- One could say Enterprise Guide rescued PROC TABULATE from oblivion
- Can anyone actually write PROC TABULATE code?
  - Bizarre syntax makes it a daunting task
- EG makes it possible to use it easily and effectively
PROC SQL

- PROC SQL has been around since 1990, but many programmers still cannot use it.
- Lack of widespread knowledge makes it difficult to deploy this tool.
- By generating the code with an easy-to-use front end, EG makes it feasible to deploy PROC SQL with a minimum of pain.
Uses of EG

● Validation
  – Statisticians
  – Throw-away code

● Ad-hoc analyses
  – Statisticians
  – One-shot
  – May be preserved if part of submission

● Production tables and listings
  – Programmers
  – Full production/validation/documentation requirements
Statisticians: Ad-hoc Analyses
Ad-hoc Analyses

- Usually, programmers do standard tables for a study, and statisticians may be expected to handle non-standard analyses
- Non-standard analyses can evolve into big projects
- Need some heavy-duty tools
Ad-hoc Analyses

- While statisticians understand the analyses, they may lack the technical skills to produce maintainable code.
- Advantages of EG come into play:
  - Easier to get other statisticians involved
  - Transfer to programming group
  - Use EG to produce tables and graphs
Ad-hoc Analyses
proc sql;
create table work.adult_not_children as
select distinct aebodsys,
       aedecod
from work.all_adult_adr
except
select distinct aebodsys,
       aedecod
from work.all_children_adr;
quit;
## Results

<table>
<thead>
<tr>
<th>AEBODSYS</th>
<th>AEDECOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. BLOOD AND LYMPHATIC SYSTEM DISORDER</td>
<td>EOSINOPHILIA</td>
</tr>
<tr>
<td>2. EAR AND LABYRINTH DISORDERS</td>
<td>VERTIGO</td>
</tr>
<tr>
<td>3. GASTROINTESTINAL DISORDERS</td>
<td>VOMITING</td>
</tr>
<tr>
<td>4. GENERAL DISORDERS AND ADMINISTRATION</td>
<td>HYPERTHERMIA</td>
</tr>
<tr>
<td>5. GENERAL DISORDERS AND ADMINISTRATION</td>
<td>PYREXIA</td>
</tr>
<tr>
<td>6. NERVOUS SYSTEM DISORDERS</td>
<td>PSYCHOMOTOR HYPERACTIVITY</td>
</tr>
<tr>
<td>7. SOCIAL CIRCUMSTANCES</td>
<td>LEARNING DISABILITY</td>
</tr>
</tbody>
</table>
Email Results
Production Tables
Project Flow

September 15, 2008
Denominator
Custom Code

Expression text:
```
TRIM(SUBGROUP_SELECT.RXGRPX) || "\n" || compress( '{N=|| put(CALCULATED denom , 5.) || }' )
```

Available variables:
- SUBGROUP_SELECT
  - PROTOCOL
  - RANID
  - POP
  - POPX
  - SEX
  - SEXX
  - RACEH
  - RACEHX
  - ETHCAT
  - AGE
  - AGECAT
  - RXGRP
  - DURROSA
  - PREV_ROSA
  - SKINDRY
  - SUBGROUP
  - RXGRPX
- Current Query
  - DENOM
  - COLHEAD

Variable values:
- **Variable Names**
  - PROTOCOL
  - RANID
  - POP
  - POPX
  - SEX
  - SEXX
  - RACEH
  - RACEHX
  - ETHCAT
  - AGE
  - AGECAT
  - RXGRP
  - DURROSA
  - PREV_ROSA
  - SKINDRY
  - SUBGROUP
- **Variable Labels**
  - Protocol (BER342)
  - Randomization #
  - Population
  - Decoded Population Flag
  - Gender
  - Decoded Gender
  - Ethnic Group
  - Decoded Ethnic Group
  - Ethnic Group Category
  - Age in years
  - Age Category
  - Treatment Group
  - Previous Duration of Rosacea (months)
  - Pre-TMT Facial Signs/Symptoms: Skindry
  - Disease Quartile
Expression text:

```plaintext
TRIM(SUBGROUP_SELECT.RXGRFX) || "\^R\"\line \" || compress('N=' || put(CALCULATED denom, 5.)) || '"
```
 Proc Tabulate

Summary Tables

Available variables:
- ALL (universal ...
- COLHEAD
- ITEM11_recode
- ITEM12_recode
- RXGRP
- THERA_recode

Preview:

<table>
<thead>
<tr>
<th>Box Area</th>
<th>COLHEAD</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
</tr>
<tr>
<td></td>
<td>ColPctN</td>
</tr>
<tr>
<td></td>
<td>N</td>
</tr>
<tr>
<td></td>
<td>ColPctN</td>
</tr>
</tbody>
</table>

Available statistics:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ColP...</td>
<td>Column ...</td>
</tr>
<tr>
<td>ColP...</td>
<td>Column ...</td>
</tr>
<tr>
<td>CSS</td>
<td>Context</td>
</tr>
</tbody>
</table>
<double-click to insert code>

PROC SQL;
%_SASTASK_DROPDS(WORK.SORTTempTableSorted);
QUIT;

/
*/
Run the tabulate procedure

options nodate nonumber orientation=portrait nobyline;
ods rtf style=NthStyles.Nth_RTF_Table notoc_data BODYTITLE
    file="&dir\tables\%trim(&file).rtf";
TITLE1 J=L "XXX Laboratories, Inc."
TITLE2 J=L "Clinical Study Report No. &protocol"
TITLE3 J=C "R\par" Table 1.1.2 Measures of Patient Satisfaction at End of Study (Per-Protocol Population)
TITLE4 J=C "#BYVAL1"

PROC TABULATE
DATA=WORK.TABLE

; by subgroup;

September 15, 2008
/* End of custom user code. */

PROC TABULATE
   DATA=WORK.TABLE

/* Start of custom user code. */
; by subgroup;
/* End of custom user code. */
;
   CLASS RXGRP / ORDER=UNFORMATTED DESCENDING MISSING;
   CLASS COLHEAD / ORDER=UNFORMATTED MISSING;
   CLASS ITEM11_recode / ORDER=UNFORMATTED MISSING;
   CLASS ITEM12_recode / ORDER=UNFORMATTED MISSING;
   CLASS THERA_recode / ORDER=UNFORMATTED MISSING;
   TABLE /* Row Dimension */
ITEM11_recode
ITEM12_recode
THERA_recode,
/* Column Dimension */
COLHEAD={LABEL="" STYLE={NOBREAKSPACE=ON}} (*)
   N*F=3.0
ColPctN={LABEL="%"} *F=PCTFMT. ;
;
### Table 1.1.2 Measures of Patient Satisfaction at End of Study (Per-Protocol Population)

**Duration of Disease Quartile 1 (≤15 months)**

<table>
<thead>
<tr>
<th></th>
<th>Active (N=27)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>N</td>
</tr>
<tr>
<td><strong>Cosmetic acceptance of the topical preparation</strong></td>
<td></td>
</tr>
<tr>
<td>(1) Very good</td>
<td>11</td>
</tr>
<tr>
<td>(2) Good</td>
<td>11</td>
</tr>
<tr>
<td>(3) Satisfactory</td>
<td>4</td>
</tr>
<tr>
<td>(4) Poor</td>
<td>1</td>
</tr>
<tr>
<td><strong>Patient tolerability in general</strong></td>
<td></td>
</tr>
<tr>
<td>(1) Good</td>
<td>16</td>
</tr>
<tr>
<td>(2) Acceptable despite minor irritation</td>
<td>9</td>
</tr>
<tr>
<td>(3) Less acceptable due to continuous irritation</td>
<td>1</td>
</tr>
<tr>
<td>(5) No opinion</td>
<td>1</td>
</tr>
<tr>
<td><strong>Patient overall rating of improvement</strong></td>
<td></td>
</tr>
<tr>
<td>(1) Excellent improvement</td>
<td>8</td>
</tr>
<tr>
<td>(2) Good improvement</td>
<td>9</td>
</tr>
<tr>
<td>(3) Moderate improvement</td>
<td>7</td>
</tr>
<tr>
<td>(4) No improvement</td>
<td>2</td>
</tr>
<tr>
<td>(5) Aggravation</td>
<td>1</td>
</tr>
</tbody>
</table>
Issues

- As SAS users, we are used to a stable programming environment.
- While the rest of the computing world has undergone several revolutions, SAS code for tables, listings, and graphs circa 1990, or earlier, can still be used.
  - Change of other hardware/software from 15 years ago still working is remote.
SAS has made a tremendous commitment to backwards compatibility for SAS programs across versions.

For example, there has been no need to archive the operating system, database, and SAS version along with the code.

SAS programs from 15 years ago still run, with at most minimal modification, and the legacy datasets can easily be updated to the current version.

In short, even with the tremendous changes in the computing environment since the 1990s, the migration path for SAS programs and datasets has been relatively smooth.
One of the main reasons is that environment was text-based, consisting of ASCII text files that can be opened with any basic text editor.

With Enterprise Guide, we are clearly moving away from this paradigm.

Instead, we have an environment that, while it offers much more power, will change from version to version without backwards compatibility.
It is not possible to convert an EG 4 project to EG 3, and it will not be possible to back-convert an Enterprise Guide 7 project to Enterprise Guide 4 in the future.

- SAS products will become like Microsoft products in that sense.
- We will migrate forwards, and the forward migration will not always be smooth.
- Because they are binary files, you cannot automate the conversion of Enterprise Guide projects by running the projects through a batch processor and applying global changes, like you can with text-based SAS programs.
Archive Copy as a Safety Net

- Here is where the concept of an archive copy becomes critical.
- An archive copy in the Enterprise Guide environment is a SAS program that is generated from the code generated by the Enterprise Guide tasks.
- Regardless of what problems arise from version compatibility issues in the programming environment, the archive copy can always be run as traditional SAS programs.
- It is critical that the archive copies be generated from an automated, validated process.
Conclusion

- Despite some limitations, Enterprise Guide offers the evolution in SAS programming, and is an exciting replacement for the traditional Display Manager programming environment.
- However, the programming environment itself, not just the SAS code, will change from version to version.
- This is a major change in the SAS programming world, and has significant implications for regulatory compliance in the pharmaceutical industry.
- Using Enterprise Guide to automatically generate archive copies (i.e., traditional SAS code) of process flows mitigates these risks.
SAS Trademarks

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 registered trademarks or trademarks of their respective companies.
My Contact Info

- Michael Todd
- info@nthanalytics.com
- www.nthanalytics.com