VDW Tutorial for Programmers

Stolen from Roy Pardee

Mangled by Gene Hart

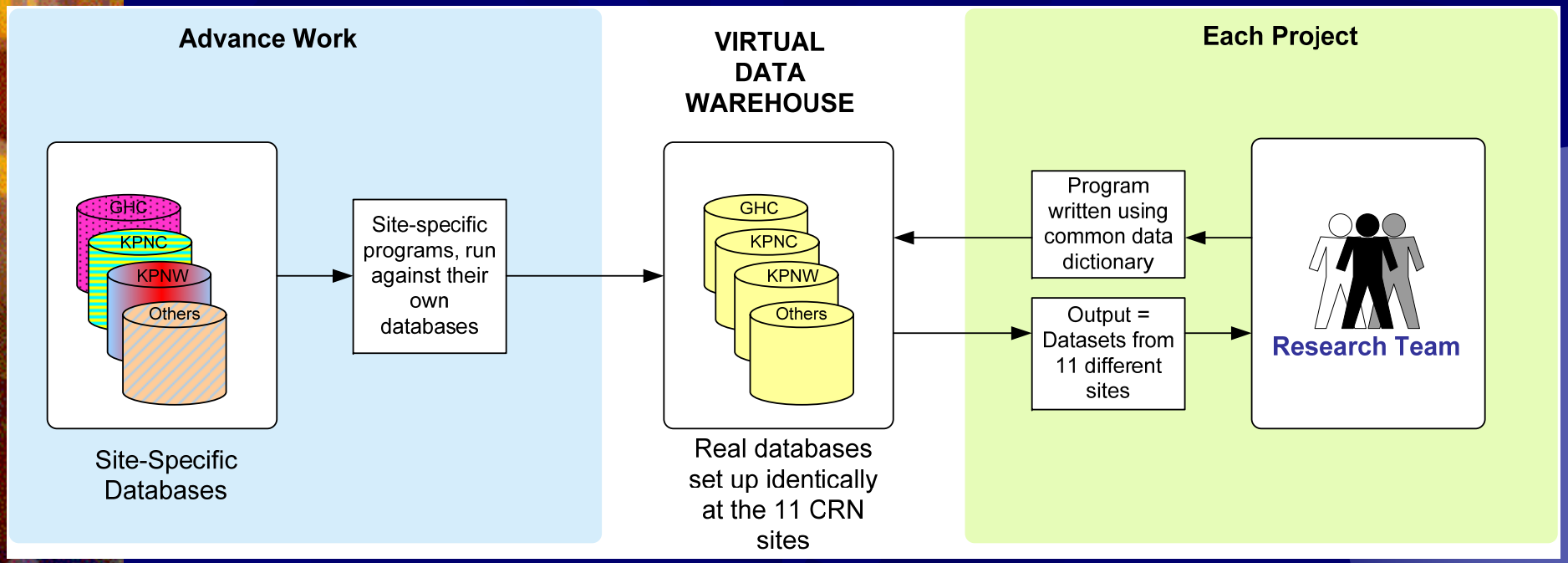
The VDW is...

- ...a series of **dataset standards** and **automated processes** at each of 11 sites,
- that allow a SAS program written at one site to be run against all the others
  - quickly
  - and with a minimum of site-specific customization.

The VDW is *Distributed*

- You could also call it a *Distributed* Data Warehouse
  - the data are real enough
  - but each site maintains its own separate slice of the whole.

The Visual Aid



VDW Data Areas

- Tumor
- Enrollment
- Demographics
- Outpatient Pharmacy
- Procedures and Diagnoses
- Census
- Vital Signs

Compared to CHS SAS DW

- MRN is person identifier
- NDC is drug identifier
- Single dataset per data area – not annualized
- Decimal points in DX codes
- Some data cleaning done
- Some quality checks comparing across HMOs

Where is VDW Documentation

- On CRN website
- Programmer's Guide
- CHS Website

The Standard Macros

- In the course of writing VDW programs, we have come up with a series of SAS macros to use as building blocks for new programs.
- Full Details are in the handout, but here is a quick overview...

Current Macros

%NDCLookup	Searches EverNDC for drugs whose generic/brand names include user-supplied search strings
%GetRxForPeople	Pulls all Rx fills for a user-supplied list of people.
%GetRxForDrugs	Pulls all Rx fills for a user-supplied list of drugs.
%GetRxForPeopleAndDrugs	Pulls all Rx fills for a user-supplied set of NDC codes and people
%GetPxForPeople	Pulls all Procedures (inpt/outpt) for a user-supplied list of people.
%GetDxForPeople	Pulls all Diagnoses (inpt/outpt) for a user-supplied list of people.

Current Macros (Cont.)

%GetPxForPx	Pulls all of a user-specified list of procedures.
%GetDxForDx	Pulls all instances of a user-supplied list of diagnoses.
%GetDxForPeopleAndDX	Pulls all diagnoses for a user-supplied list of DX and people.
%BreastCancer-Definition01	Pulls the set of incident breast tumors, based on the Early Screening criteria.
%PullContinuous2	Reduces an input dataset of people to those continuously enrolled over a user-specified period.
%DeIDDset	Replaces true identifiers with an arbitrary StudyID.

Detailed Macro Example

```
%GetRxForDrugs
(DrugLst /* The name of a dataset containing the NDCs of the
            drugs whose fills  you want. */
 , StartDt   /* The date on which you want to start collecting
                fills. */
 , EndDt     /* The date on which you want to stop collecting
                fills. */
 , Outset    /* The name of the output dataset containing the
                fills. */) ;
```

How to get VDW macros

```
filename crn_macs FTP
   "CRN_VDW_MACROS.sas" HOST =
   "centerforhealthstudies.org" CD =
   "/CRNSAS" PASS = "$blue33volcano#"
   USER = "CRNReader" ;
%include crn_macs ;
```

Standard Variables

%include “\\ctrhs-sas\warehouse\Sasdata\CRN_VDW\lib\StdVars.sas;”

Contains lines such as:

```
%let _TumorLib = \\ctrhs-
    sas\warehouse\sasdata\crn_vdw ;
%let _TumorData              = tumor ;
```

Look at VDW Guide

- Data Structures (Appendix 3)
- Walk through sample program

How to develop a VDW program

1. Familiarize yourself w/the data structures
2. Formulate the question
3. Choose sites
4. Decide on desired outputs
5. Get IRB approval
6. Write your program
7. Beta test at selected sites
8. Distribute to remaining sites.
9. Use Secure File Transfer to collect your data

1: Familiarize yourself with the data structures

- The most current versions will be on the CRN Website:

  https://www.kpchr.org/crn2/

- They are also reproduced in Appendix 3 of the VDW Developer's Guide (handout).

2: Formulate The Question

- Always important
- Probably more important for a VDW project—the potential time waste is multiplied by the number of people involved.

3: Choose your Sites

	GHC	HPHC	HPRF	HFHS	Fallon	KPNC	KPNW	KPSC	KPCo	KPH	KPG
Tumor	1974	N/A	N/A	1972	N/A	1973	1960	1988	1987	1960	1995
Enroll-ment	1988	1969	1990	1980	N/A	1970†	1982	1988	1992	1958	1995
RX	1977	1988	1990	1990	N/A	1993†	1986	*1992 (Year end)*	1993	1987	1995
DX & PX	1993	*1996 (Oct 18)*	1994	1988	1988*	*1993*†	*1995*	*1998*	1993	1992	1995

* Not yet available.

† Data available on a 10% sample

4: Decide on outputs

- What do you want back from the sites?
  - SAS datasets (recommended!)
  - Log files (recommended!)
  - Listing files
  - ODS Output (HTML, RTF, etc.)

5: Confirm IRB approval

- While this is the Investigators' ultimate responsibility, don't forget to make sure that your project has IRB approval to touch data at all your chosen sites.
- Allow for delays in herding the various IRB committees into a consensus.

6: Develop and Debug your program

- Program defensively
- Make things easy on the HMOs
- Use the variables defined in StdVars.sas to make code portable.
- Make things easy on yourself
  - Use already written macros and code snippets

7: Send ‘beta’ program to your test HMOs

- Expect additional bugs to appear as each additional site runs your program.
- Do one HMO at a time until program runs without errors.
- The ‘least virtual’ sites (e.g., those with standing datasets) usually give the best turnaround.

8: Send tested program to remaining sites

- Here again, expect this to be an iterative process.
- *Hopefully* the iterations are only caused by yet-undetected programming errors.
- Consider mocking up multi-site versions of your output, to cut down on the “Oh. That’s not what I meant.” factor.

9: Transfer data back to lead site using Secure File Transfer

- As part of the CRN Web Infrastructure, KPNW has put together an excellent Web-based application for moving files from one site to another: https://secure2.kpchr.org/crnfiletransfer
- Send mail to Gary Ansell for access: Gary.Ansell@kpchr.org

Getting Help: Sources

- Your Local Site Data Manager.
- The VDW-USERS listserv:
  - http://lists.centerforhealthstudies.org/read
- Gene and Roy:
  - hart.je@ghc.org
  - pardee.r@ghc.org