INFORMATIONAL GUIDE TO INDUSTRY

THE SENTINEL INITIATIVE

INDUSTRY DAY | 26 APRIL 2018

CONTENTS

I. Sentinel Common Data Model

Project Title	**Sentinel Common Data Model**
Date Posted	*Wednesday, October 4, 2017*
Status	Complete
Project ID	DA00001
Deliverables	Sentinel Common Data Model v6.0.2
Description	The Sentinel Operations Center (SOC) coordinates the network of Sentinel Data Partners and leads development of the Sentinel Common Data Model (SCDM), a standard data structure that allows Data Partners to quickly execute distributed programs against local data. The SOC Data Core manages creation of the Sentinel Distributed Database (SDD) using the SCDM, and maintains complete documentation of the implementation and characteristics of the SDD. The SDD refers to the data held and maintained by the Data Partners in the SCDM format. The SCDM was developed in accordance with the Mini-Sentinel Common Data Model Guiding Principles and was originally modeled after the Health Care Systems Research Network Virtual Data Warehouse (HCSRN/VDW). The SCDM currently includes 12 tables that represent information for the data elements needed for Sentinel activities. Records are linked across tables by a unique person identifier, PatID. Details of the 12 tables are provided in the SCDM v6.0.2 document. Both major and minor revisions and enhancements to the SCDM are made on a regular basis, including the addition of clinical information, incorporation of other data types and sources, and revisions based on lessons learned from use of the SDD. This may include adopting variables and formats developed by other programs. Changes in this minor release from v6.0.1 to v6.0.2 are listed in the “History of Modifications” tab of this SCDM v6.0.2 document. Major changes from version 5.0.1 to version 6.0 of the SCDM are described in detail in the v6.0 release notes. The SCDM is an Excel spreadsheet. If you do not have Excel, click here to obtain a free version of Microsoft’s Excel Viewer. If you are using a web page screen reader and are unable to access the document, please contact the Sentinel Operations Center for assistance by clicking on the Submit Comments link above or sending an email requesting assistance to info@sentinelsystem.org.

Project Title	**Sentinel Common Data Model**
	For prior versions of the SCDM, contact the Sentinel Operations Center at info@sentinelsystem.org.
Related Links	Mini-Sentinel Common Data Model – Guiding Principles v1.0 Sentinel Common Data Model – Laboratory Result Table Documentation v1.0
Workgroup Leader(s)	Jeffrey Brown PhD and Nicolas Beaulieu MA; Department of Population Medicine, Harvard Pilgrim Health Care Institute and Harvard Medical School, Boston, MA Lesley Curtis PhD; Duke Clinical Research Institute, Durham, NC Marsha A Raebel PharmD, BCPS, FCCP; Institute for Health Research, Kaiser Permanente Colorado, Denver CO, and University of Colorado Skaggs School of Pharmacy and Pharmaceutical Science, Aurora, CO Kevin Haynes PharmD, MSCE; HealthCore, Inc., Wilmington, DE
Workgroup Members	Representatives of FDA and all Sentinel Data Partners Robert Rosofsky; Health Information Systems Consulting LLC, Boston, MA
Data Sources	Sentinel Distributed Database (SDD)

II. DATA QUALITY PROCESSING

SENTINEL COMMON DATA MODEL

DATA QUALITY REVIEW AND CHARACTERIZATION PROCESS AND PROGRAMS

Program Package version: 4.1.0

Prepared by the Sentinel Operations Center
February 2018

The Sentinel System is sponsored by the U.S. Food and Drug Administration (FDA) to proactively monitor the safety of FDA-regulated medical products and complements other existing FDA safety surveillance capabilities. The Sentinel System is one piece of FDA's Sentinel Initiative, a long-term, multi-faceted effort to develop a national electronic system. Sentinel Collaborators include Data and Academic Partners that provide access to healthcare data and ongoing scientific, technical, methodological, and organizational expertise. The Sentinel Coordinating Center is funded by the FDA through the Department of Health and Human Services (HHS) Contract number HHSF223201400030I.

Table of Contents

Modification History

Version	Date	Modification	By
4.1.0	02/22/2018	• Minor bug fixes • Optimization of L1 module	Sentinel Operations Center
4.0.2	11/14/2017	• Minor bug fix	Sentinel Operations Center
4.0.1	09/2/2017	• Minor bug fixes	Sentinel Operations Center
4.0	04/27/2017	• Major updates	Sentinel Operations Center
3.3.4	2/16/2017	• Added/modified diagnosis datasets	Sentinel Operations Center
3.3.3	8/17/2016	• Added/modified core and lab datasets	Sentinel Operations Center
3.3.2	9/30/2015	• Corrected error in PatID exclusion in Labs module	Mini-Sentinel Operations Center
3.3.1	9/17/2015	• Corrected error in DEM module	Mini-Sentinel Operations Center
3.3	5/1/2015	• Redesign of labs module, addition of core checks	Mini-Sentinel Operations Center
3.2.4	9/23/2014	• Updated program for new RequestID compatibility, fixed bug	Mini-Sentinel Operations Center
3.2.3	9/2/2014	• Implemented minor bug fixes	Mini-Sentinel Operations Center
3.2.2	8/26/2014	• Implemented minor bug fixes	Mini-Sentinel Operations Center
3.2.1	7/25/2014	• Implemented minor bug fixes	Mini-Sentinel Operations Center
3.2	5/30/2014	• Integrated Common Components • Added MSCDM v4.0 compliance checks, minor bug fixes	Mini-Sentinel Operations Center
3.1.3	03/03/2014	• Implemented minor bug fixes	Mini-Sentinel Operations Center
3.1.2	09/25/2013	• Implemented minor bug fix	Mini-Sentinel Operations Center
3.1.1	09/16/2013	• Implemented minor bug fix	Mini-Sentinel Operations Center
3.1	09/05/2013	• Added/modified lab and vitals checks, fixed bugs	Mini-Sentinel Operations Center
3.0.1	03/01/2013	• Updated for UNIX compatibility	Mini-Sentinel Operations Center
3.0	11/12/2012	• Added PatID and EncounterID matching, enhanced valid date checks, fixed bugs • Added clinical data (labs and vitals) programs	Mini-Sentinel Operations Center
2.0	10/14/2010	• Added duplicate record checks, modified dataset names, fixed bugs	Mini-Sentinel Operations Center
1.0	09/20/2010	• Initial published version	Mini-Sentinel Operations Center

I. OVERVIEW

This document describes the version 4.1.0 program package used by the Sentinel Operations Center (SOC) for data quality assurance (QA) review and characterization of the Sentinel Distributed Database (SDD). To create the SDD, each Data Partner (DP) transforms local source data into Sentinel Common Data Model (SCDM) format. The SOC uses a set of data quality review and characterization programs to ensure that the SDD meets reasonable standards for data transformation consistency and quality. The version 4.1.0 QA program package queries SCDM version 6.0.2 tables.

II. DISTRIBUTED PROGRAMMING AND DATA QUALITY REVIEW CHECKS

To evaluate data characteristics and quality, SOC developed distributed code to query the content of SCDM-formatted tables. The distributed code generates aggregate output tables that help determine whether the data conform to SCDM specifications, maintain integrity across variables and across tables, and trend as expected over time. Output tables are designed to evaluate one or more data checks, i.e., pre-defined data quality measures or characterizations. Each data check is designated a “level 1,” “level 2,” or “level 3” data quality check depending on the complexity and assigned a “FlagID” for tracking and reporting purposes. A “FlagID” can represent a data characteristic or a data issue (see Section **IV.C.2.** for more information on FlagIDs).

Level 1 data checks review the completeness and content of each variable in each table to ensure that the required variables contain data and conform to the formats specified by the SCDM specifications. For each SCDM variable, level 1 data checks verify that data types, variable lengths, and SAS formats are correct and reported values are acceptable. For example, ensuring that the variable SEX in the *Demographic* table has a value of “F,” “M,” “A,” or “U” is a level 1 data check. Another example is ensuring that the variable MS_RESULT_C in the *Laboratory Result* table is only populated with values of “POSITIVE”, “NEGATIVE”, “BORDERLINE”, “UNDETERMINED”, or a RANGE: start|end unit (*i.e.,* “50|100 MG/ML”) for all laboratory tests.

Level 2 data checks assess the logical relationship and integrity of data values within a variable or between two or more variables within and between tables. For example, the SCDM requires that the variable ADMITTING_SOURCE in the *Encounter* table is populated only for inpatient and institutional encounters (*i.e.*, ENCTYPE= “IP” or “IS”). A level 2 data check would ensure that ADMITTING_SOURCE is populated only when ENCTYPE= “IP” or “IS”.

Level 3 data checks examine data distributions and trends over time, both within a Data Partner’s database (by examining output by year and year/month) and across a Data Partner’s databases (by comparing updated SCDM tables to previous versions of the tables). For example, a level 3 data check would ensure that there are no large, unexpected increases or decreases in diagnosis records over time.

III. NEW FEATURES/ENHANCEMENTS

A. EFFICIENCY

1. The new QA program package reduces the overall number of output files significantly, which reduces space and clarity of output.

B. CONSISTENT FLAG SCHEME

1. A new flag token structure now ensures consistency across all program modules. Each flag token can be linked to a flag description for easier interpretation.

C. FLAG INDICATORS

1. New flag indicators (i.e., "AbortYN", "FlagType") have also been introduced for each individual flag. This will help automate the program and provide clear indication of flag implication:

   – If AbortYN = "Y": the package will abort, errors will be reported in the log and all flags will be output to the "dplocal" folder for review by the Data Partner
   – If AbortYN = "N": the package will not abort, flags will be output to "dplocal" and/or "msoc" folder for review by the Data Partner and SOC

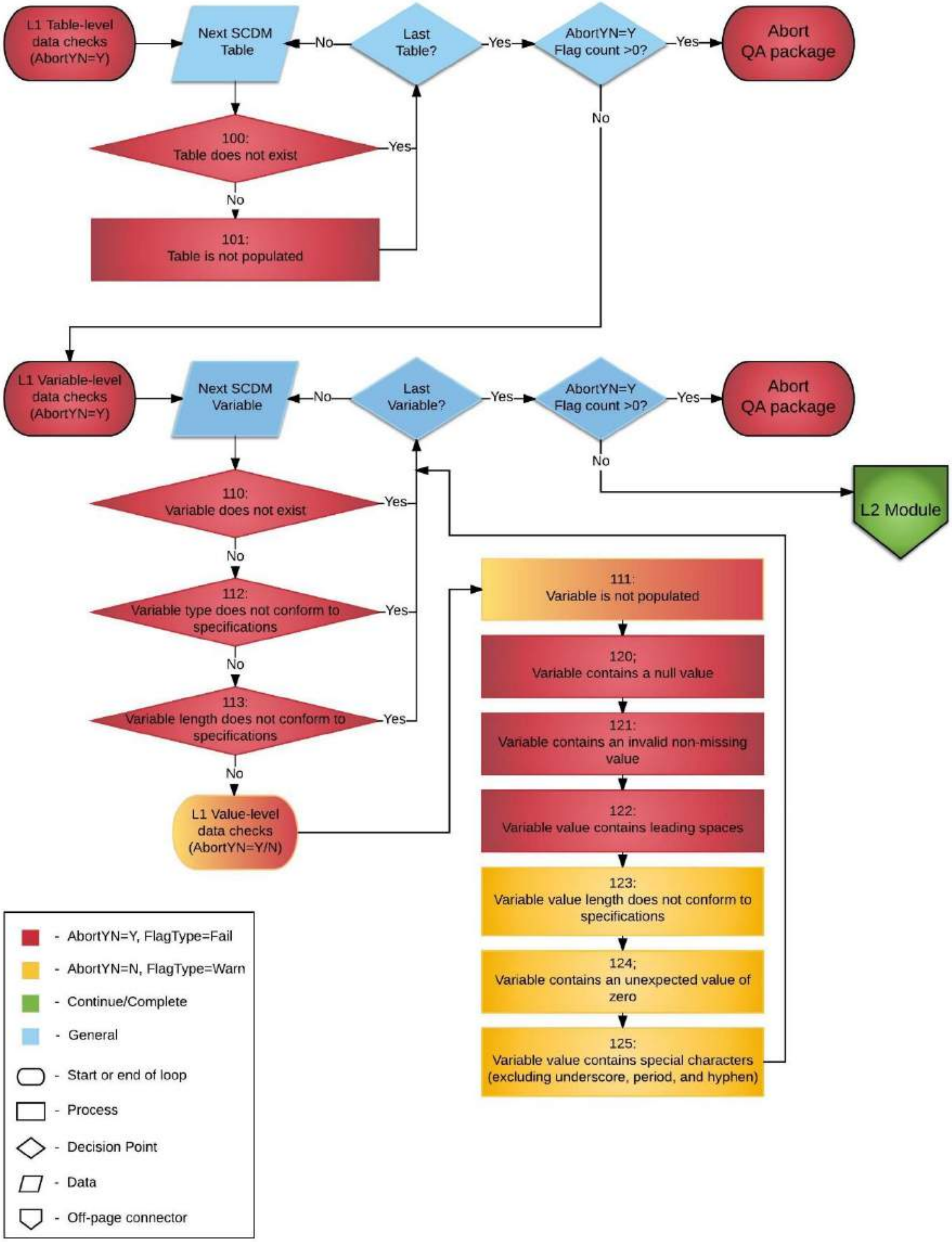
   – If FlagType = "Fail": the ETL cannot pass QA review until error is fixed
   – If FlagType = "Warn": the ETL may pass QA, but explanation or investigation could be warranted (i.e., more information is needed to determine QA pass/fail)

D. UPDATED MODULE FLOW

1. In the new QA program package, the module execution sequence has been reordered and module stop-gaps have been added as described below.
   a. All Level 1 checks for all tables are performed first and, if any major issues with the data are detected, the package will abort (**Figure 1**).
   b. All Level 2 checks are then performed in a logical sequence and abort at each step if "AbortYN" = Y for any flag in that step (**Figure 2**). The logical sequence is as follows:
      i. Perform critical intra-table checks that would cause downstream data integrity issues;
      ii. Perform critical cross-table checks for the same reasoning as above (this is the final place where the package may abort);
      iii. Continue to remaining Level 2 checks and all remaining modules, regardless of the resulting data flags.

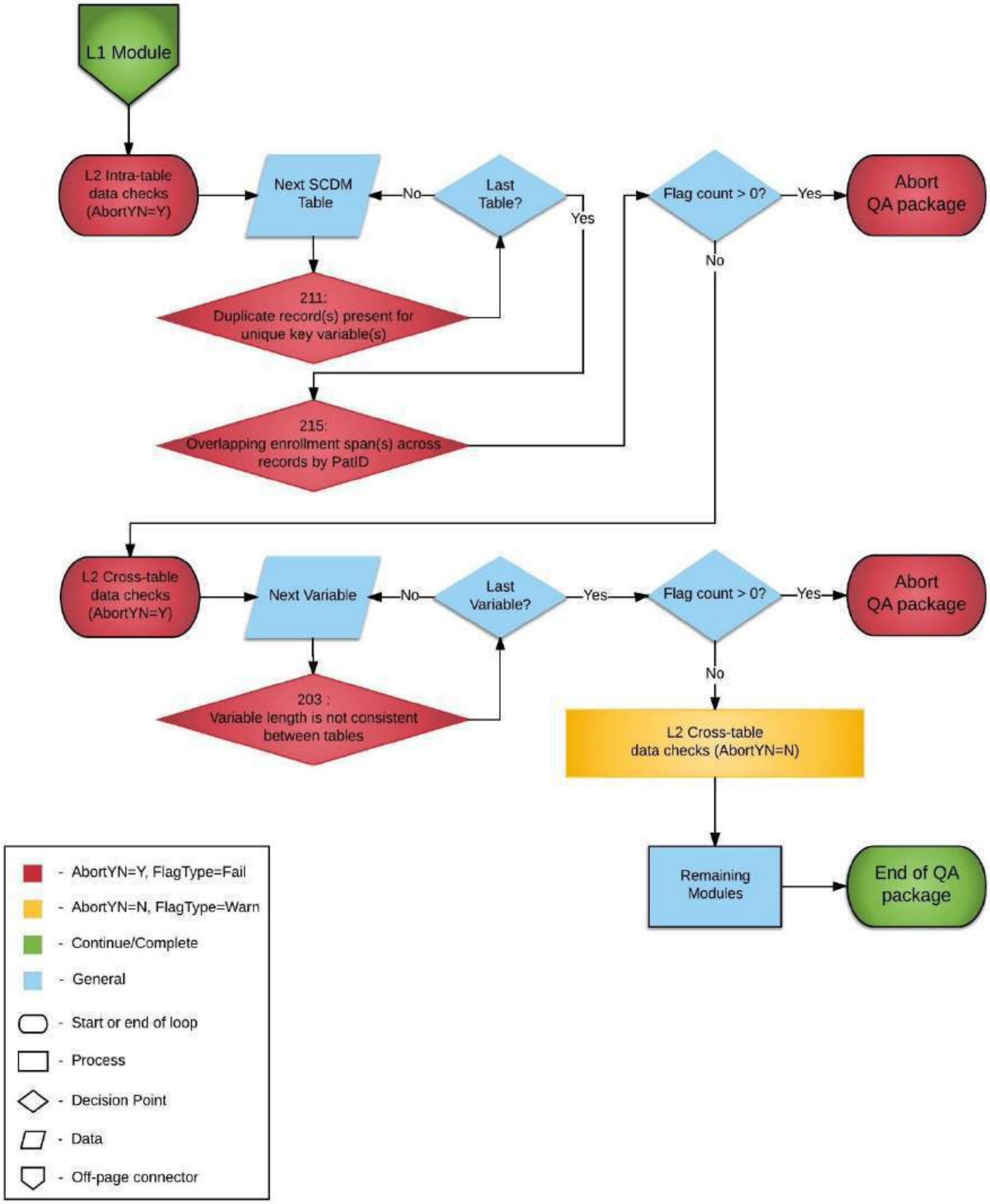
Note: Prior to step 1.b.iii., all resulting datasets are output to the "dplocal" folder. Only log, metadata, and signature files will be in the "msoc" subfolder, until all abort checks have been successfully executed. At that time, all appropriate datasets will be moved to the "msoc" subfolder. See Section **VII** for more information on "dplocal" and "msoc" output files.

Figure 1. Flowchart of Level 1 module abort logic[1]


L1 Table-level data checks (AbortYN=Y)
Next SCDM Table
No
Last Table?
Yes
AbortYN=Y Flag count >0?
Yes
Abort QA package
No
100: Table does not exist
Yes
No
101: Table is not populated
L1 Variable-level data checks (AbortYN=Y)
Next SCDM Variable
No
Last Variable?
Yes
AbortYN=Y Flag count >0?
Yes
Abort QA package
No
L2 Module
110: Variable does not exist
Yes
No
112: Variable type does not conform to specifications
Yes
No
113: Variable length does not conform to specifications
Yes
No
L1 Value-level data checks (AbortYN=Y/N)
111: Variable is not populated
120; Variable contains a null value
121: Variable contains an invalid non-missing value
122: Variable value contains leading spaces
123: Variable value length does not conform to specifications
124; Variable contains an unexpected value of zero
125: Variable value contains special characters (excluding underscore, period, and hyphen)
🟥 - AbortYN=Y, FlagType=Fail
🟨 - AbortYN=N, FlagType=Warn
🟩 - Continue/Complete
🟦 - General
- Start or end of loop
- Process
- Decision Point
- Data
- Off-page connector

[1] The rules are governed by two lookup tables, **lkp_all_l1** and **lkp_all_flags**, which are located in the "inputfiles" directory of the QA program package.

Figure 2. Flowchart of Level 2 module abort logic[2]


L1 Module
L2 Intra-table data checks (AbortYN=Y)
Next SCDM Table
No
Last Table?
Yes
211: Duplicate record(s) present for unique key variable(s)
215: Overlapping enrollment span(s) across records by PatID
Flag count > 0?
Yes
Abort QA package
No
L2 Cross-table data checks (AbortYN=Y)
Next Variable
No
Last Variable?
Yes
203 : Variable length is not consistent between tables
Flag count > 0?
Yes
Abort QA package
No
L2 Cross-table data checks (AbortYN=N)
Remaining Modules
End of QA package
🟥 - AbortYN=Y, FlagType=Fail
🟨 - AbortYN=N, FlagType=Warn
🟩 - Continue/Complete
🟦 - General
- Start or end of loop
- Process
- Decision Point
- Data
- Off-page connector

[2] Please note that flags in figures 1 and 2 may change over time.

IV. PROGRAM PACKAGE

A. FOLDER STRUCTURE

The standard SOC folder structure should be used for creating this package, with the following subfolders:

1. **dplocal**

   The subfolder containing output generated by the request that should remain with the DP (and may be used to facilitate follow-up queries).

2. **inputfiles**

   The subfolder containing all input files and lookup tables needed to execute a request. Input files contain information on what tables should be output and the type of analyses conducted on the variables in each table. Input files are created for each run of the QA program package by the SOC DMQA team.

3. **msoc**

   The subfolder containing output generated by the local SAS program. This will be used by SOC for post-QA processing.

4. **sasprograms**

   The subfolder containing the master SAS program that must be edited and then executed by the SOC analyst.

B. SAS PROGRAM MODULES

Discrete program functions should be contained in separate macros, to facilitate use by multiple programmers simultaneously, and for ease of future modification. The program package includes the following SAS modules:

1. **sasprograms/00.0_mscdm_data_qa_review_master_file.sas**

   This master program requires editing by the Data Partner to identify the location of the SCDM tables in staging, as well as the location of the Common Components program. It is the only program that is edited and executed by the partner.

2. **inputfiles/00.0_mscdm_control_flow.sas**

   This program module selectively and sequentially executes the QA program modules.

3. **inputfiles/00.1_mscdm_standard_macros.sas**

   Contains macros used across QA program modules.

4. **inputfiles/00.2_mscdm _formats.sas**

   Contains standard formats used across QA program modules.

5. **inputfiles/00.3_mscdm_sas_log_checker_directory_cc.sas**

   Checks all program logs and summarizes notes, warnings, and errors in an output PDF file after all modules have completed.

6. **inputfiles/00.4_mscdm_qasignaturerequest.sas**

   Creates a final signature file (msoc.alltable_signature) that summarizes metadata from individual module-level signature files.

7. **inputfiles/01.1_mscdm_data_qa_review-level1.sas**

   Queries applicable SCDM tables to perform level 1 data checks and creates all l1 output datasets. The data checks included in the module check compliance with the current SCDM specifications (*e.g.*, appropriate length, type, and format). Reference **Figure 1** above.

8. **inputfiles/01.2_mscdm_data_qa_review-level2.sas**

   Queries applicable SCDM tables to perform all level 2 abort checks, as well as L2 cross-table checks and output datasets. Reference **Figure 2** above.

9. **inputfiles/02.1_mscdm_data_qa_review-enrollment.sas**

   Queries the *Enrollment* table, and outputs L2 and L3 datasets containing information on medical and drug coverage indicators, enrollment start, end, enrollment duration, and chart availability.

10. **inputfiles/03.1_mscdm_data_qa_review-demographic.sas**

    Queries the *Demographic* table, and outputs L2 and L3 datasets containing information on age, sex, race, and zip code.

11. **inputfiles/04.1_mscdm_data_qa_review-dispensing.sas**

    Queries the number of members and records in the *Dispensing* table, and outputs information on dispensing date, dispensings over time, dispensings per member over time, days supply and dispensed amount.

12. **inputfiles/05.1_mscdm_data_qa_review-encounter.sas**

    Queries the number of members, records, encounters, and providers in the *Encounter* table, and outputs information on admission and discharge date, encounters over time, encounter type, length of stay, facility location, admitting source, discharge status and disposition, DRG and DRG type, and number of encounters per member.

13. **inputfiles/05.2_mscdm_data_qa_review-diagnosis.sas**

    Queries the number of members, records, encounters, and providers in the *Diagnosis* table, and outputs information on encounter type, admission date, diagnosis code and type, principal diagnosis indicators, number of diagnoses per encounter, and number of diagnoses over time.

14. **inputfiles/05.3_mscdm_data_qa_review-procedure.sas**

    Queries the number of members, records, encounters, and providers in the *Procedure* table, and outputs information on encounter type, admission date, procedure code and type, number of procedures per encounter, and number of procedures over time.

15. **inputfiles/06.1_mscdm_data_qa_review-death.sas**

This module is only executed by Data Partners who have death data available. Queries the number of members and records in the *Death* table, and outputs information on the source of and confidence in death information, number of deaths over time, and if the death date has been imputed.

16. **inputfiles/06.2_mscdm_data_qa_review-causeofdeath.sas**

    This module is only executed by Data Partners who have cause of death data available. Queries the number of members and records in the *Cause of Death* table, and outputs information on cause of death codes and cause type, and source of and confidence in cause of death information.

17. **inputfiles/07.1_mscdm_data_qa_review-labs.sas**

    This module is only executed by Data Partners who have laboratory data available. This program queries the number of members and records in the *Laboratory Result* table and outputs information on lab tests included, result values, units, and available dates.

18. **inputfiles/99.1_mscdm_data_qa_review-minmax_dates.sas**

    Queries SCDM tables and outputs minimum and maximum dates per table (as applicable), and calculates DP Min (calculated as the maximum of the Min Dates) and DP Max (calculated as the minimum of the Max dates).

19. **inputfiles/99.2_mscdm_data_qa_review-level3.sas**

    Creates Level 3 cross-table and age-related datasets from intermediate table-level datasets, and performs many “housekeeping” activities, such as moving specific files from “dplocal” to “msoc”, and bulk addition of DPID and SITEID variables to “msoc” datasets.

C. LOOKUP FILES

A set of lookup tables will be included in the ‘/inputfiles’ directory of the QA program package to allow easy modifications of value sets and error codes.

1. **control_flow.sas7bdat**

   Used by control flow module to selectively and sequentially execute QA modules. Used in the L1 and L2 modules to selectively execute data checks by SCDM table.

2. **lkp_all_flags.sas7bdat**

   This lookup table provides a list of DMQA-assigned error codes (*FlagID*) and descriptions (*Flag_Descr*), and the name of the output dataset used to evaluate checks associated with the error code.

   Each error code (FlagID) is comprised of five meaningful tokens:

   1. Table(s) Abbreviation (*Table*): Abbreviation that indicates SCDM table(s) queried for the data check
   2. Check Level (*Level*): Level of data check; i.e., the type of quality assurance check performed (1=basic, single variable model compliance; 2=cross-variable and cross-table compliance checks; 3= temporal trends within and across database versions)
   3. Variable identifier (*VarID*): Unique variable identifier

4. Test Identifier (*TestID*): Unique MS_Test_Name value identifier concatenated with Result_type (e.g. 01-N)
5. Check Identifier (*CheckID*): Unique data check identifier

Each of these tokens is also included as a separate variable in the lookup table.

Data checks for which the value of the variable *FlagYN* = "Y" indicate that the check is performed automatically and output to the flags output dataset. Level 1 and 2 error codes that do not pass automatic evaluation (i.e., have a count of one or more records that meet the error code description) will get written a flags dataset.

AbortYN = "Y" is used to indicate checks that must pass in order to complete the QA module successfully. Data checks for which the value of the variable *FlagType*= "Fail" indicate that the check must pass in order to pass QA review.

3. **lkp_all_l1.sas7bdat**

   This lookup table provides information on all variable attributes by SCDM table in order to perform Level 1 model compliance checks. For each variable in the table, the variable identifier, required type (numeric or character), and required length are specified. An indicator of whether a variable is allowed to have a missing/blank value is also included, as well as a list of all allowable variable values (as applicable).

4. **lkp_all_minmax.sas7bdat**

   This lookup table provides table-level information such as source table and variable names, as well as inclusion status in order to calculate minimum and maximum dates of data completeness.

5. **lkp_dem_zip.sas7bdat**

   Zip code lookup that links valid 5-digit zip codes with the associated state. SAS creates a zip file quarterly and is the source of this data.

6. **lkp_dia_l2.sas7bdat**

   This lookup table provides a comprehensive list of all allowable cross variable value combinations to be used for the *Diagnosis* table module Level 2 data checks.

7. **lkp_enc_l2.sas7bdat**

   This lookup table provides a comprehensive list of all allowable cross variable value combinations to be used for the *Encounter* table module Level 2 data checks.

8. **lkp_lab_l2.sas7bdat**

   This lookup table provides a comprehensive list of all allowable combinations for the following variables in the *Laboratory Result* table: MS_Test_Name, Result_type (numeric or character), MS_Test_Sub_Category, Fast_ind, Specimen_Source, and LOINC. To be used for Level 2 error checks.

9. **lkp_lab_l2_nc.sas7bdat**

   Allowable *Laboratory Result* table lab test result, modifier, and unit combinations. This lookup table maps the acceptable MS_Result_C values to the Modifier values and MS_Result_unit values, as appropriate. Only characterized laboratory tests are included in this lookup table.

10. **lkp_lab_result_ranges.sas7bdat**

    Expected *Laboratory Result* table lab test ranges. This lookup table defines the expected result value ranges for lab tests with numeric results in the *Laboratory Result* table. Only MS_Test_Name values where Result_Type= "N" and lkp_lab_test.Characterized="Y" are included in this lookup table.

11. **lkp_lab_test.sas7bdat**

    Lab Test/Test type relationship. This lookup table provides a comprehensive list of all laboratory tests included in the *Laboratory Result* table. It includes laboratory test IDs and associated MS_Test_Name, Result_type, whether a unit is required, and whether a test has been characterized or not. To be used for Level 2 error checks.

12. **lkp_pro_l2.sas7bdat**

    This lookup table provides a comprehensive list of all allowable cross variable value combinations to be used for the *Procedure* table module Level 2 data checks.

V. PROGRAM EXECUTION

A. MASTER PROGRAM INPUT TO BE COMPLETED BY SOC

Table 1 below defines the variables that must be initialized by the SOC to execute the QA program package (*i.e.*, defined by the SOC before execution of the programs). Please note that these values cannot be left blank. These inputs are unique to each request and/or Data Partner.

Table 1. Master Program Variable Definitions to be completed by SOC

Field Name	Description
ReqETL	Request ETL number
MSProjID	Project ID
MSWPType	Workplan Type
MSWPID	Workplan ID
MSDPID	Unique Data Partner ID
MSVerID	Version ID

B. MASTER PROGRAM INPUT TO BE COMPLETED BY DP

Table 2 below defines the variables that must be initialized by the end user to execute the QA program package (*i.e.*, defined by the Data Partner before execution of the programs). Please note that these values cannot be left blank. Each Data Partner is required to enter user inputs at the beginning of the applicable data quality program. These inputs are unique to each Data Partner.

Table 2. Master Program Variable Definitions to be completed by Data Partner

Field Name	Description
MSCC	Path to directory containing the Common Components file (ms_common_components.sas)
Evaluate_MSCDM	Path to directory containing the SCDM datasets pending QA review

VI. PROGRAM ALGORITHMS/LOGIC

A. DEFINITION OF ENROLLMENT SPAN COMPARISONS

Table 3. Definitions and examples of enrollment date range relationships by PatID

Definition	Illustration	Enr_Start	Enr_End	DMQA Action
Disjoint: No conflict or overlap		**01/01/2015**	**03/31/2015**	Pass
		05/01/2015	**08/31/2015**	
Consecutive: Two date ranges are consecutive		**01/01/2015**	03/31/2015	Warn
		04/01/2015	**08/31/2015**	
Overlap: Two date ranges overlap over a range		**01/01/2015**	03/31/2015	Fail
		02/01/2015	**08/31/2015**	
Duplicate: Two date ranges are identical		**01/01/2015**	**08/31/2015**	Fail
		01/01/2015	08/31/2015	
Subset: One date range is a subset of another		**01/01/2015**	**08/31/2015**	Fail
		02/01/2015	03/31/1015	

B. MINIMUM AND MAXIMUM DATES OF DATA COMPLETENESS

Minimum and Maximum dates (min/maxdate) of data completeness are created by this package for all SDCM tables containing at least one date variable, as defined in the input file *lkp_all_minmax.sas7bdat*.

The mindate is calculated by determining the earliest year-month (e.g. 2010-01) with a record count within an 80% threshold of the next month (e.g. 2010-02) and then assigning the first day of the month to create a SAS date, formatted as YYYY-MM-DD (e.g. 2010-01-01).

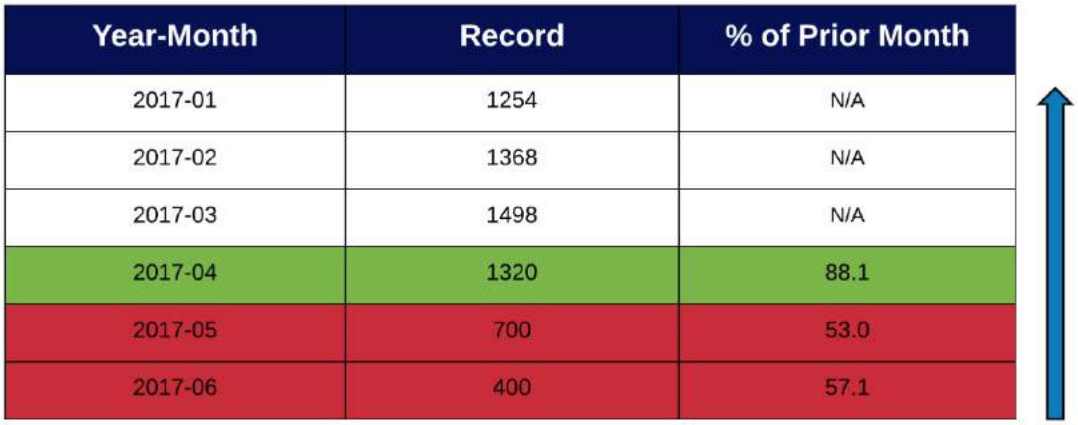
The maxdate is calculated by determining the latest year-month (e.g. 2017-10) with a record count within an 80% threshold of the prior month (e.g. 2017-09) and then assigning the last day of the month to create a SAS date, formatted as YYYY-MM-DD (e.g. 2017-10-31).

Overall min/maxdates are then created, based on the SCDM table min/maxdates, as defined in the input file *lkp_all_minmax.sas7bdat*. The overall mindate is calculated by determining the latest mindate (i.e. the “maximum of the minimum”). The overall maxdate is calculated by determining the earliest maxdate (i.e. the “minimum of the maximum”).

These dates are stored in a SAS dataset (msoc.minmax_dates). The overall min/maxdates associated with the latest production ETL at each DP site will be used by Common-Components (CC) to populate the global macro variables &mindate and &maxdate for distributed request packages.

It should be noted that the min/maxdate algorithm may not work well with all types of date distributions (e.g., a distribution with a large drop proceeded or followed by a long, flat tail of many months).

Figure 3. Example of Maximum date of data completeness algorithm[3]



Year-Month	Record	% of Prior Month
2017-01	1254	N/A
2017-02	1368	N/A
2017-03	1498	N/A
2017-04	1320	88.1
2017-05	700	53.0
2017-06	400	57.1

MaxDate = 2017-04-30

[3] When there are at least two consecutive months at the tail end of the distribution with relatively low counts, the algorithm may sometimes pick a month with incomplete data. For example, if the month 2017-06 had a count of "600" instead of "400", it would meet the 80% threshold and be incorrectly chosen as the max date.

C. AGE CALCULATION

Age in years (age_years) is calculated using the Kreuter method using the date of birth variable (DEM.birth_date) and the new Overall MaxDate macro variable (&dp_maxdate) calculated for the ETL under review.

The following equation (first proposed by William Kreuter) measures age in whole years. It counts the months between the two dates, subtracts one month if the day boundary has not been crossed for the last month, and then converts months to years and reports it as an integer.

```
Age_years= floor((intck('month',birth_date,&DP_MaxDate.)-
           (day(&DP_MaxDate.)<day(birth_date)))/12)
```

D. AGE GROUP CATEGORIES

Age in years will be summarized based on the following categories:

```
"00. Missing"
"00. Negative"
"01. 0-1 yrs"
"02. 2-4 yrs"
"03. 5-9 yrs"
"04. 10-14 yrs"
"05. 15-18 yrs"
"06. 19-21 yrs"
"07. 22-44 yrs"
"08. 45-64 yrs"
"09. 65-74 yrs"
"10. 75+ yrs"
```

E. MODULE-LEVEL EXECUTION SIGNATURE FILES

Individual module-level signature files (msoc.{module}_signature) containing metadata and basic benchmarking statistics are created after each module executes.

The table below describes the contents of the enr_signature file:

Variable	Description	Source/Derivation	Example Values
DPID	2 letter Data Partner ID	Common Components &DPID	MS
SiteID	1-4 letter Site ID	Common Components &SITEID	OC
MSReqID	Request ID	Master program &MSREQID	soc_qar_v4_msoc_b05
MSProjID	Project ID	Master program &MSPROJID	soc
MSWPType	Workplan Type	Master program &MSWPTYPE	qar
MSWPID	Workplan ID	Master program &MSWPID	v4
MSDPID	Unique DP Site Identifier	Master program &MSDPID	msoc
MSVerID	Request Version ID	Master program &MSVERID	b05
QAVer	QA program package version	Master program &QAVER	4.1.0
SCDMVer	Current SCDM version	Master program &SCDMVER	6.0.2
Module	QA program package Module	Control flow &MODULE	enr
OSABBR	Operating System Environment	SAS automatic macro variable &SYSSCP	WIN
OSNAME	Operating System Name	SAS automatic macro variable &SYSSCPL	X64_7PRO
SASVersion	SAS version (short)	SAS automatic macro variable &SYSVER	9.4
SASVersionLong	SAS version (long)	SAS automatic macro variable &SYSVLONG	9.04.01M3P062415
RunType	SAS execution mode*	SAS automatic macro variable &SYSENV	FORE
NCPU	Number of CPU	SAS automatic macro variable &SYSNCPU	4
StartTime	Module start time	Standard macros %TIMESTAMP	12SEP2017:11:32:48.00
StopTime	Module end time	Standard macros %TIMESTAMP	12SEP2017:11:32:58.40
Seconds	Module run time in seconds	Standard macros %TIMEREPORT	10
RunTime	Module run time	Standard macros %SIGNATURE_END	0 h 0 m 10 s

* FORE=Interactive, BACK=Batch

F. REQUEST-LEVEL EXECUTION SIGNATURE FILES

A single signature file (msoc.alltable_signature) contains request-level metadata and basic benchmarking statistics after all modules have completed, and summarizes data from all module-level signature requests.

The table below describes the contents of the alltable_signature file:

Variable	Description	Source/Derivation	Example Values
DPID	2 letter Data Partner ID	Common Components &DPID	MS
SiteID	1-4 letter Site ID	Common Components &SITEID	OC
MSReqID	Request ID	Master program &MSREQID	soc_qar_v4_msoc_b05
MSProjID	Project ID	Master program &MSPROJID	soc
MSWPType	Workplan Type	Master program &MSWPTYPE	qar
MSWPID	Workplan ID	Master program &MSWPID	v4
MSDPID	Unique DP Site Identifier	Master program &MSDPID	msoc
MSVerID	Request Version ID	Master program &MSVERID	b05
QAVer	QA program package version	Master program &QAVER	4.1.0
SCDMVer	Current SCDM version	Master program &SCDMVER	6.0.2
OSABBR	Operating System Environment	SAS automatic macro variable &SYSSCP	WIN
OSNAME	Operating System Name	SAS automatic macro variable &SYSSCPL	X64_7PRO
SASVersion	SAS version (short)	SAS automatic macro variable &SYSVER	9.4
SASVersionLong	SAS version (long)	SAS automatic macro variable &SYSVLONG	9.04.01M3P062415
RunType	SAS execution mode	SAS automatic macro variable &SYSENV	FORE
NCPU	Number of CPU	SAS automatic macro variable &SYSNCPU	4
TotalRequestTime	QA program package run time	Signature request module	0 h 6 m 27 s

VII. APPENDICES

Execution of all modules of the QA program package generates the output files in Appendix A and Appendix B.

A. APPENDIX A: PROGRAM PACKAGE OUTPUT IN “DPLOCAL”

Reference Appendix A for a list of “dplocal” datasets.

B. APPENDIX B: PROGRAM PACKAGE OUTPUT IN “MSOC”

Reference Appendix B for a data dictionary containing “msoc” datasets.

SENTINEL DATA QUALITY ASSURANCE PRACTICES

COMPLIANCE WITH "GUIDANCE FOR INDUSTRY AND FDA STAFF: BEST PRACTICES FOR CONDUCTING AND REPORTING PHARMACOEPIDEMIOLOGIC SAFETY STUDIES USING ELECTRONIC HEALTHCARE DATA"

Prepared by: Sentinel Operations Center

February 27, 2017

The Sentinel System is sponsored by the U.S. Food and Drug Administration (FDA) to proactively monitor the safety of FDA-regulated medical products and complements other existing FDA safety surveillance capabilities. The Sentinel System is one piece of FDA's Sentinel Initiative, a long-term, multi-faceted effort to develop a national electronic system. Sentinel Collaborators include Data and Academic Partners that provide access to healthcare data and ongoing scientific, technical, methodological, and organizational expertise. The Sentinel Coordinating Center is funded by the FDA through the Department of Health and Human Services (HHS) Contract number HHSF22301001T.

Sentinel Data Quality Assurance Practices

Compliance With "Guidance for Industry and FDA Staff: Best Practices for Conducting and Reporting Pharmacoepidemiologic Safety Studies Using Electronic Healthcare Data"

Table of Contents

I. PURPOSE

The Food and Drug Administration (FDA) set forth its current recommendations for data quality assurance (QA) in the following document: "Guidance for Industry and FDA Staff: Best Practices for Conducting and Reporting Pharmacoepidemiologic Safety Studies Using Electronic Healthcare Data" (Guidance), section IV.E "Best Practices – Data Sources: Quality Assurance (QA) and Quality Control (QC)," in May 2013.[1]

This Guidance describes best practices that particularly apply to observational studies designed to assess the risk associated with a drug exposure using electronic healthcare data. While the guidance specifically mentions that it does not address real-time active safety surveillance assessments such as Sentinel on page 3, many of its recommendations regarding data QA apply more broadly to include the practices and standards used by the Sentinel Coordinating Center's (SOC). This document describes the ways in which the SOC upholds the FDA's standards regarding data quality assurance.

II. COMPLIANCE WITH THE BEST PRACTICES FOR DATA SOURCES

As part of the SOC, the Data Management and Quality Assurance (DMQA) Team addresses the following topics recommended by the FDA in section IV.E. The topics are in bolded font and the manner in which they are addressed by the DMQA Team is in italicized font.

- **The general procedures used by the data holders to ensure completeness, consistency, and accuracy of data collection and management**

  While the SOC has no access to the source data of its Data Partners (DPs), each DP transforms local source data into the Sentinel Common Data Model (SCDM) format to be included in the Sentinel Distributed Database (SDD). The purpose of the SOC data QA activities is to assess whether the SDD meets reasonable standards for data transformation consistency and quality, including reviewing data integrity across data tables as well as characterizing data trends and patterns.

- **The frequency and type of any data error corrections or changes in data adjudication policies implemented by the data holders during the relevant period of data collection**

  To evaluate data characteristics and quality, SOC developed distributed code to query the content of SCDM formatted tables. The distributed code generates aggregate output tables that help determine whether the data conform to SCDM specifications, maintain integrity across variables and across tables, and trend as expected over time. Execution of all sections of the data quality review and characterization program package generates up to 244 output files: 164 Core output tables, up to 59 Lab output tables (depending on laboratory tests present in the data), and 21 Vital Signs output tables.

[1] http://www.fda.gov/downloads/drugs/guidances/ucm243537.pdf, accessed on February 14, 2017.

Output tables are designed to evaluate one or more data checks, i.e., pre-defined data quality measures or characterizations. Approximately 1,200 data checks are evaluated during each DP data refresh. Each data check is designated a “level 1,” “level 2,” “level 3,” or “level 4” data quality check depending on the complexity of a data characteristic/issue:

- Level 1 data checks review the completeness and content of each variable in each table to ensure that the required variables contain data and conform to the formats specified by the SCDM specifications (e.g., data types, variable lengths, SAS formats, acceptable values, etc.).
- Level 2 data checks assess the logical relationship and integrity of data values within a variable or between two or more variables within and between tables (e.g., variable ADMITTING_SOURCE in the Encounter table is populated only for inpatient and institutional encounters).
- Level 3 data checks examine data distributions and trends over time, both within a Data Partner’s database (by examining output by year and year/month) and across a Data Partner’s databases (by comparing updated SCDM tables to previous versions of the tables). For example, a level 3 data check would ensure that there are no large, unexpected increases or decreases in records over time.
- Level 4 data checks examine the occurrence and prevalence of nonsensical diagnoses and examine variations in care practices across Data Partners (e.g., the proportion of prostate cancer diagnoses among women). Level 4 checks are designed to provide more targeted data analyses and profiling of Data Partner data, and are not necessarily designed to detect and correct errors.

Once the DMQA team receives the output from the data quality review and characterization programs provided by each DP, the following steps are implemented at the SOC, as part of DMQA standard operating procedures, to achieve uniform performance of the QA processes across all DPs and timeframes:

1. A Data Quality Analyst does a primary review of the output to ensure that data quality acceptance criteria are met.
2. The Data Quality Analyst who performs the primary review prepares a Data Quality Findings Report (hereafter referred to as the Report).
3. Another Data Quality Analyst does a secondary review of the output and the Report to ensure that data quality acceptance criteria are met.
4. The Data Quality Analyst performing the secondary review annotates the Report with additional findings or corrections.
5. The Data Manager reviews the Report and the output, finalizes the Report, and transmits the Report to the Data Partner using the Sentinel Secure Portal or other approved secure mechanism.

If data issues are found in the Report, the DP investigates and provides a written response either explaining the results or proposing corrective action. All decisions and discussions are documented in the Report in order to develop a knowledge repository about each DP’s data. The SOC and its DPs work closely together to resolve QA-related data issues in order to approve the data for use in the FDA’s data requests.

- **A description of any peer-reviewed publications examining data quality and/or validity, including the relationships of the investigators with the data source(s)**

  The SOC data quality assurance is not geared toward any one study type or outcome of interest, thus the DMQA team does not maintain its own list of peer-reviewed publications.

  However, Sentinel as a whole is committed to publishing findings in journals and sharing information at relevant conferences. "Publications and Presentations" section of the Sentinel Initiative website provides summary information about Sentinel activities that have appeared in peer-reviewed journals or conference materials.[2] Additionally, "Health Outcome of Interest Validations and Literature Reviews" section of the Sentinel Initiative website lists literature reviews and validation studies of a number of safety outcomes.[3] Lastly, some published articles specifically focused on building and maintaining a framework and infrastructure for data quality assessments in distributed data networks.[4 ,5]

- **Any updates and changes in coding practices (e.g., ICD codes) across the study period that are relevant to the outcomes of interest**

  The SOC data quality assurance is not geared toward any one study type or outcome of interest. Any such updates and changes are documented by the project teams leading various evaluations using the SDD.

  In general, some of the principles that guide the development and maintenance of the SCDM[6] ensure flexibility designed to capture various drug, diagnosis and procedure code types, including the new and evolving ones:

  - Principle 2: The SCDM is able to incorporate new data types and data elements as needs indicate.
  - Principle 6: The SCDM leverages evolving healthcare coding standards.

  The DMQA team uses licensed databases of diagnosis, procedure and NDC codes to validate a list of codes included in each DP's data, incorporates the results of this medical code verification process into the summary QA review report and communicates the findings to the DPs.

- **Any changes in key data elements during the study time frame and their potential effect on the study**

  Regardless of outcomes of interest and any specific time frames, the DMQA team routinely monitors data trends and patterns of the key data elements (i.e., the SCDM variables needed to make sure that the FDA's data queries can be executed properly). If any data anomalies are

[2] https://www.sentinelinitiative.org/communications/publications, accessed on February 22, 2017.
[3] https://www.sentinelinitiative.org/sentinel/surveillance-tools/validations-lit-review, accessed on February 22, 2017.
[4] http://repository.edm-forum.org/cgi/viewcontent.cgi?article=1052&context=egems, accessed on February 27, 2017.
[5] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4306391/pdf/nihms-491345.pdf, accessed on February 27, 2017.
[6] https://www.sentinelinitiative.org/sites/default/files/data/DistributedDatabase/Mini-Sentinel_CommonDataModel_GuidingPrinciples_v1.0_0.pdf, accessed on February 27, 2017.

found, the SOC works together with the DP to investigate the issues and find suitable solutions in order for the DP's data to be included in the SDD.

- **A report on the extent of missing data over time (i.e., the percentage of data not available for a particular variable of interest) and a discussion on the procedures (e.g., exclusion, imputation) employed to handle this issue. Investigators should also address the implications of the extent of missing data on study findings and the missing data methods used**

  The DMQA team routinely collects information about the missingness of all variables in the SCDM and communicates the findings to each DP. Any procedure to address missingness for a particular variable of interest is done by the project team conducting a study of any specific health outcome of interest using the SDD. All Sentinel studies include a study-specific data quality/fitness-for-use assessment of the fields to be included in that study.

4. How ARIA Analyses Have Been Used by FDA

This page summarizes how select analyses conducted in Sentinel's Active Risk Identification and Analysis (ARIA) system have been used by FDA since Sentinel's official launch in February 2016. ARIA can contribute to FDA's regulatory process in a variety of ways, such as contributing evidence to support a label change, respond to a Citizens Petition, or become part of an Advisory Committee deliberation. Information from ARIA can also provide evidence that alleviates concerns about a particular safety issue and might lead FDA to determine that no regulatory action is necessary based on the available information.

Each ARIA analysis listed below contributed in some material way to inform an important regulatory discussion or action. FDA makes decisions about drug safety issues based upon the totality of evidence. The listing of an ARIA analysis in the table means that Sentinel's ARIA system was one important source of evidence considered.

Drug Name	Outcome Assessed	ARIA Analysis	Regulatory Determination / Use	Date Posted
Keppra (levetiracetam)	Anaphylaxis and angioedema	Level 1	Drug Safety Label Change, Warnings and Precautions • Results • FDA Drug Safety Labeling Changes Page	11/30/2017
Ketoconazole oral tablets	Drug use trends after safety label change and use in labeled indications	Level 1	Citizen Petition Response • Results • Letter from FDA (Docket No. FDA-2015-P-0578)	12/4/2017
Antipsychotic agents (including haloperidol injection)	Ischemic and hemorrhagic stroke	Level 1, Level 2	Sentinel data was used to support decisions around potential labeling changes for antipsychotics and stroke risk. FDA decided that no action is necessary at this time, based on available information. • Level 1 Results • Level 2 Results • Results among SSRI Users • 2017 ICPE Symposium	12/8/2017
Gadolinium-based contrast agents	Exposure in pregnancy	Level 1	Advisory Committee Presentation & FDA Drug Safety Communication	12/19/2017

			• Results • Medical Imaging Drugs Advisory Committee (MIDAC) Slides • FDA Drug Safety Communication	
TNF-alpha inhibitors	Exposure in pregnancy	Level 1	Drug Safety Label Change, Pregnancy and Lactation • Results • FDA Drug Safety Labeling Changes Page Enbrel (etanercept)	12/21/2017
None	Respiratory syncytial virus associated illness (RSV-AI)	Level 1	Epidemiological assessment of RSV-AI and patterns of health care utilization to help inform development of novel RSV therapeutics • Results	1/25/2018
Sinuva (mometasone furoate)	Cataracts and glaucoma	Level 1	Feasibility assessment of ARIA sufficiency to replace a sponsor postmarketing requirement (PMR) safety study • Results • Approval letter	2/5/2018
Continuous or extended cycle oral contraceptives	Venous thromboembolism	Level 1, Level 2	FDA decided that no action is necessary at this time, based on available information. • Level 1 Results • Level 2 Results • 2017 ICPE Symposium	3/5/2018

How Mini-Sentinel Analyses Have Been Used By FDA

Drug Name	Outcome Assessed	Analysis	Regulatory Determination / Use	Date Posted
Pradaxa (dabigatran etexilate)	Intracranial hemorrhage, gastrointestinal bleed	Level 1*	FDA decided that no action is necessary at this time, based on available information. • Results • Drug Safety Communication • Publication	1/24/2018
Olmesartan medoximil	Intestinal sprue	Level 1*	Safety Labeling Change, Warnings and Precautions; Drug Safety Communication • Results • Safety Labeling Change • Drug Safety Communication	1/24/2018
Xarelto (rivaroxaban)	Intracranial hemorrhage, gastrointestinal bleed, and ischemic stroke	Level 3	FDA decided that no action is necessary at this time, based on available information. • Results • Publication	1/24/2018
Second generation antipsychotic agents	Metabolic effects in children (Type 2 diabetes, metabolic syndrome, weight gain)	Protocol-based assessment**	FDA decided that no new action on behalf of pediatrics is necessary at this time, based on available information. • Results • Publication	2/2/2018
Onglyza (saxagliptin) and Januvia (sitagliptin)	Acute myocardial infarction	Protocol-based assessment**	Advisory Committee Presentation • Results • Endocrinologic and Metabolic Drugs Advisory Committee (EMDAC) Slides • Publication	2/2/2018
Onglyza (saxagliptin) and Januvia (sitagliptin)	Hospitalized heart failure	Protocol-based assessment**	Advisory Committee Presentation • Results • Endocrinologic and Metabolic Drugs Advisory Committee (EMDAC) Slides • Publication	2/2/2018
Intravenous iron products	Anaphylaxis	Protocol-based assessment	FDA decided that no action is necessary at this time, based on available information. • Results	2/12/2018

*This query was performed using Mini-Sentinel's Modular Program 3, the precursor to an ARIA L1 analysis.

**Complete results are contained in the associated publications and/or final written reports.

V. HELPFUL RESOURCES

A. Conversion of Medicare Claims Synthetic Public Use Files (SynPUFs) to Sentinel Common Data Model (SCDM) Format

Project Title	**Conversion of Medicare Claims Synthetic Public Use Files (SynPUFs) to Sentinel Common Data Model (SCDM) Format**
Date Posted	*Thursday, January 25, 2018*
Status	In progress
Description	As part of a broader intiative to enhance the accessibility of the Sentinel Common Data Model (SCDM) and related tools, this work will develop and post SCDM formatted files to the Sentinel website for public use. In addition, this work will also develop a sample Routine Analytic Framework (RAF) package so that the public may easily execute Sentinel tools on the available SCDM-formatted files.
Workgroup Leader(s)	Lauren Zichittella MS; Tiffany S. Woodworth MPH; Department of Population Medicine, Harvard Pilgrim Health Care Institute and Harvard Medical School, Boston, MA
Workgroup Members	David Cole BM; Andrew Petrone MPH; Natasha De Marco MPH; Emily Welch MPH; Tancy Zhang MPH; Ella Pestine MPH; Department of Population Medicine, Harvard Pilgrim Health Care Institute and Harvard Medical School, Boston, MA
Data Sources	Sentinel Distributed Database (SDD)

https://www.sentinelinitiative.org/sentinel/methods/conversion-medicare-claims-synthetic-public-use-files-synpufs-sentinel-common-data

B. PATIENT EPISODE PROFILE RETRIEVAL (PEPR)

Mini-Sentinel

MINI-SENTINEL CBER/PRISM SURVEILLANCE

INFRASTRUCTURE FOR EVALUATION OF STATISTICAL ALERTS ARISING FROM VACCINE SAFETY DATA MINING ACTIVITIES IN MINI-SENTINEL

Prepared by: David V. Cole, BM,[1] Martin Kulldorff, PhD,[2] Meghan Baker, MD, ScD,[1] Grace Lee, MD, MPH,[1] Judith C. Maro, PhD, MS,[1] Inna Dashevsky, MS,[1] W. Katherine Yih, PhD, MPH,[1] Carolyn Balsbaugh, MPH,[1] Estelle Russek-Cohen, PhD,[3] David Martin, MD, MPH,[3] Michael Nguyen, MD[3]

Author Affiliations: 1. Department of Population Medicine, Harvard Pilgrim Health Care Institute and Harvard Medical School, Boston, MA; 2. Division of Pharmacoepidemiology & Pharmacoeconomics, Brigham and Women's Hospital and Harvard Medical School, Boston, MA; 3. Center for Biologics Evaluation and Research, Food and Drug Administration, Rockville, MD

July 22, 2016

v0.6

sMini-Sentinel is a pilot project sponsored by the U.S. Food and Drug Administration (FDA) to inform and facilitate development of a fully operational active surveillance system, the Sentinel System, for monitoring the safety of FDA-regulated medical products. Mini-Sentinel is one piece of the Sentinel Initiative, a multi-faceted effort by the FDA to develop a national electronic system that will complement existing methods of safety surveillance. Mini-Sentinel Collaborators include Data and Academic Partners that provide access to health care data and ongoing scientific, technical, methodological, and organizational expertise. The Mini-Sentinel Coordinating Center is funded by the FDA through the Department of Health and Human Services (HHS) Contract number HHSF223200910006I.

Mini-Sentinel CBER/PRISM Surveillance

Infrastructure for Evaluation of Statistical Alerts Arising from Vaccine Safety Data Mining Activities in Mini-Sentinel

Table of Contents

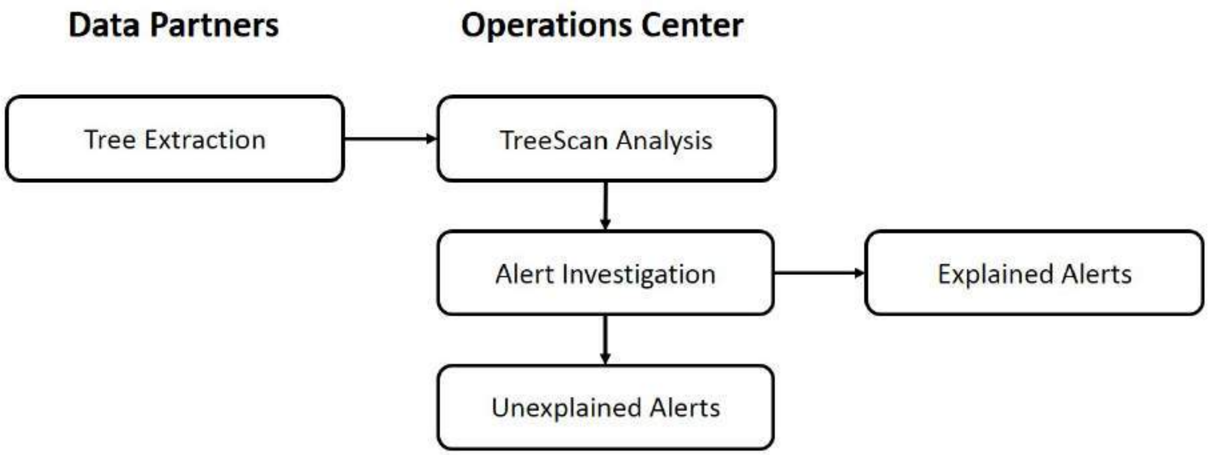
I. INTRODUCTION

Vaccine safety work in Mini-Sentinel to date has been conducted through protocol-based assessments (PBAs) with chart review1-5 and sequential surveillance in near-real time,6 all of which look for an association between an exposure and one or more pre-specified outcomes. At the same time, development work has been conducted on data mining methods using a tree-based scan statistic to look for unanticipated adverse events following a specified vaccine exposure.7-9 Since the purpose of TreeScan is to generate hypotheses that might represent potential vaccine safety concerns, the implementation of TreeScan requires approaches to efficiently investigate alerts. Since chart review is resource- and time-intensive, a more efficient, scalable, and timely approach was needed to obtain some of the important clinical context to help evaluate TreeScan alerts to differentiate between those which are expected or known and alerts that require more study. This project fulfills those goals by creating the infrastructure to freeze, evaluate, and visualize the procedure and diagnostic codes associated with the TreeScan alerts.

Description of TreeScan Operations at Project Start

During the developmental phases of TreeScan, the operational process started with the Tree Extraction workplan, prepared at the Operations Center and run at the participating Data Partner sites. After all sites uploaded their results, a programmer/analyst at the Operations Center combined and summarized the data into the input format required by the TreeScan software. The analysis was run and the results reviewed by the TreeScan workgroup. However, the original data pull was large and not frozen nor formatted specifically to enable follow-up investigations. There were no automated ways for the larger dataset to be reduced in size to focus on alerts, nor were the data captured in a way that enabled further evaluation either in aggregate or at an individual patient level in de-identified manner.

Figure 1: TreeScan Process Flow During Methods Development Stage



This report illustrates how the original TreeScan operational process was modified in 3 key ways to achieve the goal of creating a new alert follow-up approach. It also details how we tested the software enhancements using vaccine safety data from Sentinel.

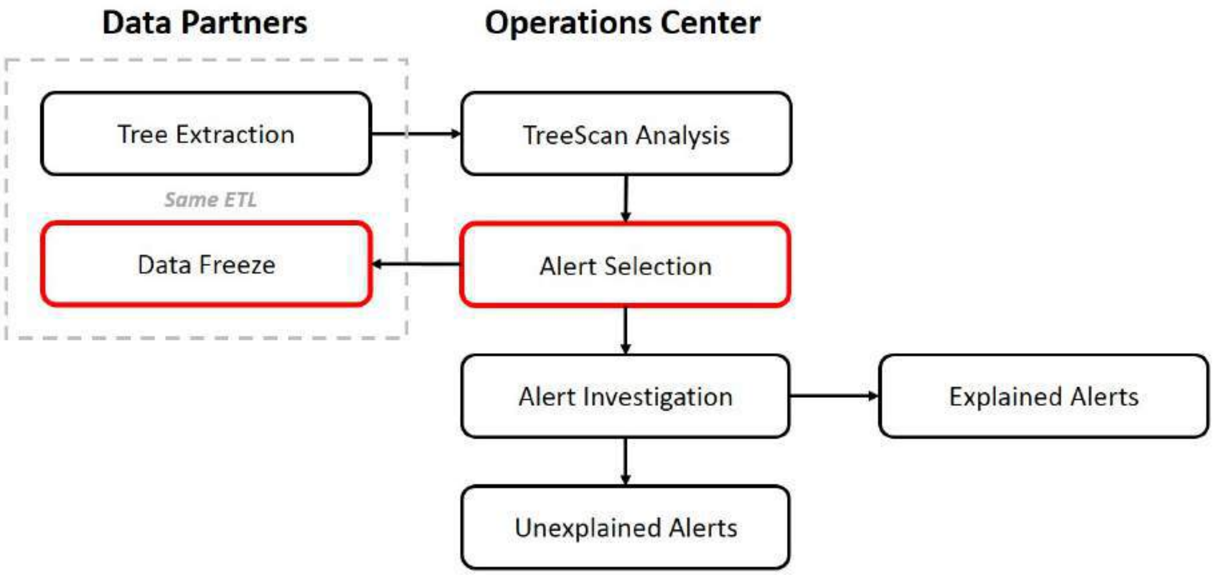
The three aims of this project were:

1. To integrate a data freeze into the Tree Extraction and Analysis process for one TreeScan method
2. To develop a means by which limited, de-identified, patient-level case data associated with selected alerts could be retrieved from the Data Partner sites
3. To create a reporting tool for FDA and Mini-Sentinel clinicians and investigators to use in reviewing the case data

II. DEVELOPMENT

A. AIM 1: DATA FREEZE

Figure 2: TreeScan Process Flow Adding Data Freeze



The first aim addresses the dynamic nature of the Mini-Sentinel Distributed Database (MSDD) (http://mini-sentinel.org/data_activities/distributed_db_and_data/default.aspx), where Data Partners periodically extract, transform, and load (ETL) their data into the Mini-Sentinel Common Data Model (CDM) format on a staggered schedule to add data for the newly-available time period as well as to add, delete, and update data for the previously reported period. Since there is no existing requirement that the patient identifier (PatID) value be consistent from one ETL to the next, a PatID value in one ETL may or may not be associated with the same patient in subsequent ETLs. Even if we kept a list of PatIDs which contributed to an alert, we would have no guarantee these were the exact same *patients* in a more recent ETL. By freezing case data from the same ETL used during Tree Extraction at each site, we can avoid such issues.

Time pressure is also somewhat mitigated, since frozen data may be retrieved and reviewed at any time without regard to refresh schedules. The main time pressure is thus limited to the requirement to complete Tree Extraction, TreeScan analysis, and data freeze on the same ETL for all Data Partners. Further, by freezing data for patients associated only with selected alerts rather than for the entire cohort, we minimize the amount of data that must be frozen and stored at the Data Partner sites, and we also gain efficiency in subsequent program runs on these much smaller datasets.

1. Tree Extraction Crosswalk

The first step in the Data Freeze process is to identify patients who contributed exposure-event episodes to each alert. Since the Tree Extraction program already identifies those episodes, we have added code to the Tree Extraction program to create and save at the local Data Partner sites a crosswalk dataset containing one record per incident event. For self-controlled analyses, the incident event is recorded as a combination of vaccine exposure and a subsequent event occurring within a pre-specified time window following exposure.9

At minimum, the following variables are required in the crosswalk dataset: patient ID, admission date of exposure, admission date of event, diagnosis code, diagnosis code type, and the analysis group to which the patient episode applies. (See Appendix A for the data dictionary.) The analysis group is used to distinguish between analysis looks at more than one vaccine (for example, MMR and MMRV) and/or at more than one age group (for example, 1 to 2 years and 3 to 5 years). Using the preceding examples, we would have four analysis groups, one for each unique combination of vaccine and age group.

2. Alert Selection

Once Tree Extraction has been completed by all Data Partners, the output data are aggregated at the Operations Center and run through the TreeScan analysis software (http://www.treescan.org), which in turn outputs a results file. To automate the steps between analysis and running the data freeze, five SAS macro programs were developed:

a. Convert the Tree Temporal Scan Analysis Results Flat File Into a SAS Dataset

The first macro converts the Tree Temporal Scan analysis results from a delimited flat file into a SAS dataset. Two versions of the macro were developed: one to be used if the relative risk and excess case count are already included in the file, and another to be used if those variables are not included. This alternative macro calculates the relative risk and excess case count using the other variables in the file, using the following formulas:[12]

$$\text{relative risk} = \frac{\frac{\text{observed}}{\text{expected}}}{\frac{\text{node cases} - \text{observed}}{\text{node cases} - \text{expected}}}$$

$$\text{excess cases} = \text{observed} - \left(\text{expected} \times \frac{\text{node cases} - \text{observed}}{\text{node cases} - \text{expected}}\right)$$

b. Convert the Horizontal Diagnosis Tree Dataset Into a Vertical Child-Parent[1] Structure

The second macro converts the diagnosis tree used by Tree Extraction in horizontal form, i.e. one record per diagnosis code and additional variables for each higher-level node of the tree, to a vertical form with one record per child-parent relationship which will be useful when identifying the next-higher node for any node which produces an alert. To illustrate, take a simple example from the current form of the tree based on the Multi-level Clinical Classifications Software (MLCCS), a product of the Agency for Healthcare Research and Quality's Healthcare Cost and Utilization Project (http://www.hcupus.ahrq.gov/toolssoftware/ccs/ccs.jsp). The ICD-9 diagnosis code for febrile seizures is 780.31, and the record in the diagnosis tree lookup dataset looks like this:

dx_codetype	Dx	mlccs5	mlccs4	mlccs3	mlccs2	mlccs1
09	780.31	780.31	06.04.02.00	06.04.02	06.04	06

When this horizontal record is converted into the vertical child-parent form, we have four records:

child	parent
780.31	06.04.02.00
06.04.02.00	06.04.02
06.04.02	06.04
06.04	06

Since future versions of the diagnosis tree could conceivably have a different number of levels than the current five, the conversion macro includes code to automatically determine the number of node variables in the horizontal tree and extract the nodes one by one into the vertical child-parent tree.

c. Convert the Child-Parent Tree Dataset Into dx-node Structure

In order to construct the alerts SAS lookup file to be used with the Data Freeze program, the third macro converts the child-parent tree into a different vertical form with one record per unique combination of diagnosis code and node. Taking the above example, we have the following five records:

dx	node
780.31	780.31
780.31	06.04.02.00
780.31	06.04.02
780.31	06.04
780.31	06

d. Automatically Identify Statistical Alerts Using Criteria Agreed Upon by the TreeScan Workgroup Prior to Analysis

The fourth macro selects the alerts for data freeze based on criteria that should be established by the FDA and TreeScan workgroup prior to analysis. First, primary alerts are selected from nodes meeting both of the following criteria: 1) p-value less than or equal to a maximum value; and 2) relative risk greater than or equal to a minimum value. These primary alerts are then compared to the child-parent

[1] The term "child-parent" here refers to tree structure terminology and not to a familial relationship between people.

tree, and if the parent for any primary alert has a p-value less than or equal to a maximum value, the node is added to the set of alerts to be frozen. All of the p-value and relative risk inputs are represented by macro parameters to allow investigators the flexibility to adjust the criteria.

Note that this data freeze is meant as a defensive measure to preserve data related to alerts that *may* require further investigation, and to do so in a timely fashion, we select a broader set of alerts than will actually require review. In reality, most if not all of the alerts will cover outcomes that already have an established association with vaccines in general or with the specific vaccine of interest. Further investigation will be limited to those in the "unexplained" category, as illustrated in **Figure 2**, which represents a small subset of the alerts that are selected for data freeze.

e. Create a Lookup Dataset for Use with the Data Freeze Program

The fifth and final macro compares all selected nodes to the dx-node tree to create a lookup table with the diagnosis codes associated with each node along with an arbitrarily assigned alert ID to be used to distinguish between frozen datasets if further investigation is required. For this step in particular, it is essential that the exact tree from Tree Extraction has been used. The tree may be pruned differently depending on the vaccine and age group under evaluation, and new versions of the tree will be developed to account for the transition to ICD-10, meaning the nodes of different versions of the tree may have different sets of underlying diagnosis codes. Finally, an additional table is created to display the selected alerts for the FDA and PRISM investigators to review and approve as the final step before distributing the Data Freeze program. At this point, additional nodes that did not meet the primary or secondary alert selection criteria may be added manually to the set of alerts for data freeze.

The Alert Selection program package was developed to automate processes, minimize opportunities for human error, and, most importantly, to shorten the time needed between analysis and data freeze. The timing of the data freeze is critical, as it is most desirable to freeze from the same ETL used to generate the alerts. Data Partners refresh as often as every three months on differing schedules, so the data freeze must be done as quickly as possible to avoid issues with CDM data at any site being overwritten with a new ETL. Before running Tree Extraction, care should also be taken to choose the optimal time window to complete all steps from Tree Extraction through Data Freeze on the same ETL at all Data Partner sites. To maintain efficiency in the Alert Selection phase, any alterations to the TreeScan analysis output file structure should be communicated in advance so the programmer can ensure compatibility with the programs.

3. Data Freeze Program

The Data Freeze workplan package includes the PRISM TreeScan Data Freeze macro, required utility macros (including the standard ms_freezedata macro), the alert lookup dataset with AlertID and associated Dx values, and a master SAS program.

The Data Freeze macro uses the standard ms_freezedate macro to create and save a snapshot of each available CDM table, populated with data for only the patients associated with one or more alerts. A preliminary step in the macro obtains a list of the PatIDs associated with each alert, and then creates a list of unique values across all alerts. Then a single set of CDM tables is saved to avoid duplication of data that must be stored at the Data Partner sites. The macro also creates and saves two subsets of the Crosswalk dataset for each alert. The first contains the exact exposure-event records which contributed to the alert. The second contains all other records in the Crosswalk dataset for the patients associated with the alert; that is, it contains records for incident events that are unrelated to the alert in question.

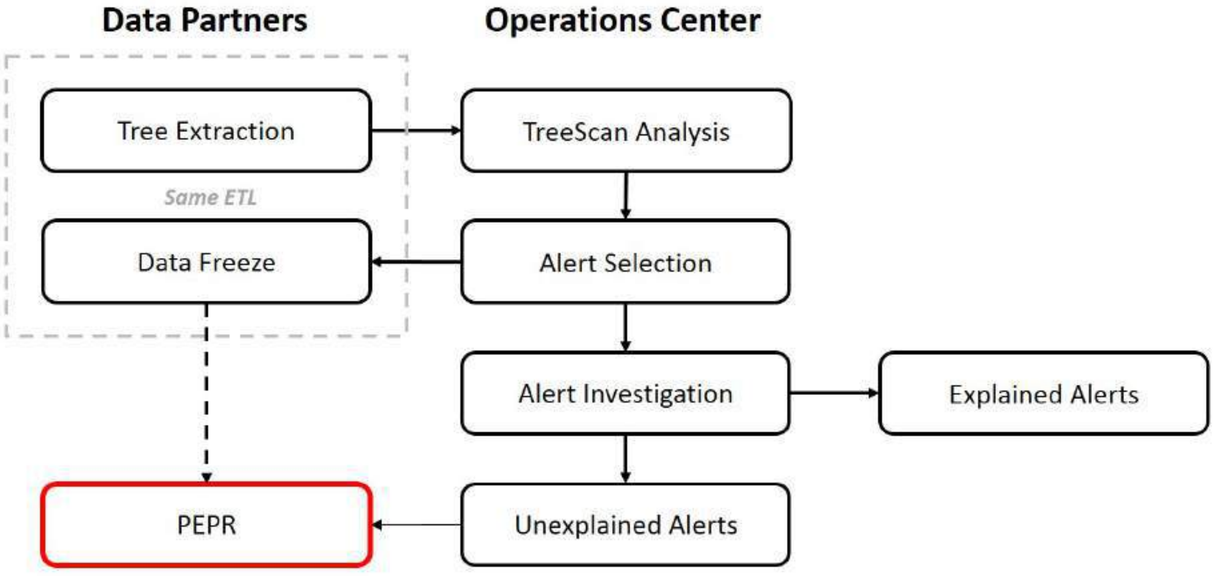
The Data Freeze package does not create any output datasets to be returned for review to the Operations Center. Only the SAS log and signature data set are returned for review by the programmer to confirm that the program ran without error.

Once the data are frozen, time may be taken to consider what, if any, follow-up investigation is necessary. Since Data Freeze is the time-critical step, and then follow-up analysis can be done at any point with the frozen datasets, care should be taken to select the widest range of alerts that may require follow-up while still being thoughtful in minimizing the size of files to be stored at the Data Partner sites. In general, far more alerts will be frozen than are of concern to the FDA, but due diligence requires the data be available in case a review becomes necessary. Additionally, alerts on the larger branches of the tree will likely have smaller branches that also produce alerts, which implies many of the saved alerts will be related and involve the same patients. Take a simple example of febrile seizures, which feeds into a higher node for convulsions. If we save the patients related to the convulsions node, then we would automatically capture those related to the febrile seizures node.

The frozen datasets are subject to standard Mini-Sentinel data retention policies (www.mini-sentinel.org/work_products/About_Us/Mini-Sentinel-Principles-and-Policies.pdf).

B. AIM 2: PATIENT EPISODE PROFILE RETRIEVAL (PEPR)

Figure 3: TreeScan Process Flow Adding PEPR. Note That PEPR Can Be Run at Any Time After Data Freeze, Even if the Original ETL is no Longer Available.



Many alert investigation tools are already available, including (but not limited to): checking programs and data for possible errors; conducting literature review for the exposure/outcome pair and coding practices for the outcome; reviewing descriptive statistics from the Tree Extraction data and the Mini-Sentinel modular programs or summary tables; and when using methods other than Tree Temporal Scan, looking for clusters in time from exposure to event.[10,11] PRISM and FDA also have experience with PBAs using chart review to validate outcomes and investigate associations with vaccine exposures,1′2′3 but PBAs are time-consuming and resource-intensive, making them cost-prohibitive as a routine tool for investigation of statistical alerts arising from data mining.

The second aim of this activity fills the gap between broader investigation tools and detailed PBAs by creating re-usable programs to extract and retrieve patient-level case data for review by FDA and PRISM investigators. This type of review is not intended to validate the outcome or determine the validity of an alert but rather to determine whether further investigation is warranted. It is also not intended as an automatic first option but instead should be used only on a small subset of alerts, if at all, and then only after careful consideration of the circumstances by the surveillance team.

Interest in similar capabilities from outside the TreeScan workgroup led us to develop the Patient Episode Profile Retrieval (PEPR) as a self-contained macro requiring a single input dataset stored at the Data Partner site to identify patient episodes of interest. At minimum, the input dataset must contain a PatID variable and at least one date variable. Since the PEPR macro itself does not determine the patient episodes to be included, other methods must first be used to identify patient episodes of interest and complete any required sampling or sub-setting.

1. Security

The PEPR output datasets are based on the CDM tables with certain modifications made to protect patient privacy. In order to strike a balance between the need for robust patient-level case data and minimum-necessary data requirements that are fundamental to the distributed network model, we implemented both mandatory and optional security measures.

a. Mandatory

Pseudo-identifiers are automatically assigned to replace four identifiers – PatID, EncounterID, Provider, and Facility_code – in the output files, and crosswalks are saved at the local sites to allow translation to the original. Each pseudo-identifier is assigned using sequential numbering to assure uniqueness, but the original values are first randomly sorted to further mask the identifier. A random seed parameter is included to assure the randomization process can be reproduced, if necessary.
Additional care is taken in assignment of the PatID pseudo-identifier to account for situations in which a single patient may contribute more than one episode to an analysis. To distinguish between separate episodes for the same patient in the output datasets, particularly when using relative dates (described in the next section), the PatID pseudo-identifier contains two parts: the first identifies the patient, and the second identifies the episode. For example, the current Tree Extraction program defines incidence at the third level of the tree. If an alert occurred at a higher level, a patient could contribute more than one incident event to the alert, and if the events occurred on different days with the 1 – 56 days after exposure, we would not be able to distinguish between those events to assign appropriate relative dates and preserve data integrity unless we assign the pseudo-identifier to the patient-episode combination rather than to the patient alone.

Future TreeScan methods development will add multi-dose analysis, i.e. inclusion of doses beyond the first observed per patient. Once again, depending on the incidence definition and the spacing of doses, a patient could conceivably contribute the same or a similar incident event following each dose. Further, with an eye toward extending use of PEPR beyond TreeScan, we considered PBAs such as the PRISM evaluation of febrile seizures following influenza vaccination in young children.3 Incidence was defined as first observed in 42 days. Since small children are recommended to have two doses of the vaccine, a single patient could have been identified as a case once for each dose if a seizure followed each dose.

That evaluation also looked at PCV and DTaP-containing vaccines, and those additional exposures could have resulted in identification of separate adverse event episodes if the vaccines were administered on different days.

b. Optional

Date variables may be masked by calibrating values to a meaningful relative index variable specified by macro parameter. The relative index date value is subtracted from each CDM date variable so that the index date now has a value of 0 (SAS date value = Jan. 1, 1960), and then any other date now represents the number of days before or after that index date. For Tree Temporal Scan using a self-controlled analysis, the natural choice for relative index is the exposure date, which is then set to 0, and then all other dates represent how many days before or after exposure the encounter, drug dispensing, enrollment start, or enrollment end occurred. The format of the date variables is preserved but the identifiable information removed. Thus, the reviewer will not know the actual calendar date of an event, only the number of days before or after the relative index.

Exposure date	Original ADate value	New ADate value	Unformatted numeric value
10/01/2011	10/01/2011	01/01/1960	0
10/01/2011	10/10/2011	01/10/1960	10
04/15/2006	04/15/2006	01/01/1960	0
04/15/2006	03/15/2006	12/01/1959	-31

If this option is not selected, the original calendar date values are included in the output datasets. An activity involving chart review, for example, would need the original calendar dates in order to match chart data with electronic data.

The only date variable not included in this relative date option is the birth date, since application of the rule would result in representation of the patient's exact age in days on any given relative date. Instead, another option allows for the birth date value to be set to missing for all patients. This rule is optional since the birth date value is necessary for activities involving chart review.

As a compromise, an additional option allows the programmer to specify age groupings to more broadly categorize each patient's age at a selected index date variable.

The final option concerns two variables in the CDM that contain geographic information represented by the ZIP code: Zip in the Demographic table, representing the last known patient ZIP code, and Facility location in the Encounter table, valued with the first three digits of the facility ZIP. An optional macro parameter allows the values to be set to missing, converted to standard postal state abbreviation, or to retain their original value. If geographic clustering is suspected as a confounder, the postal state value may be used while still transmitting a lower level of specificity than the actual five- or three-digit ZIP values. The actual ZIP values would rarely – if ever – be required for anything less than full chart review and should only be used with extreme caution.

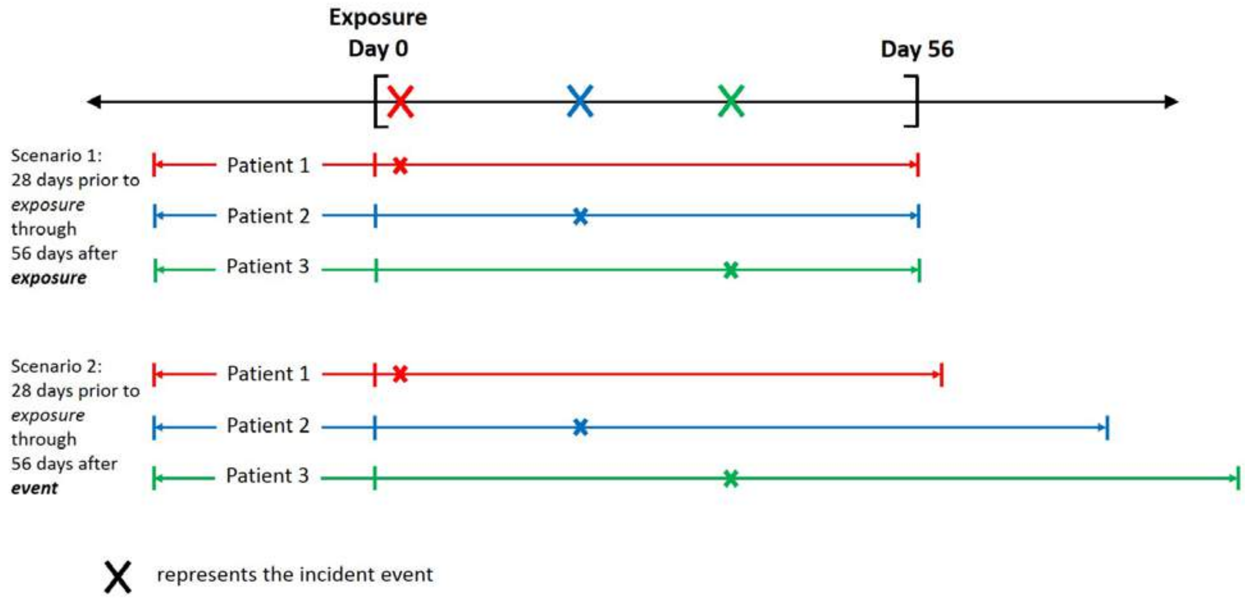
Any altered variables follow the same data type and format as the original CDM variables, but the variable name is changed by adding an underscore character "_" to the beginning of the original variable name. For example, PatID becomes _PatID, and ADate becomes _ADate. This convention serves as a reminder to anyone working with the datasets that those variables have been altered for security purposes. In order to run programs that were written to run on the CDM tables, the programmer only needs to change the variable names back to the original. (See Appendix B for a listing of altered variables.)

2. PEPR Window

In order to limit the amount of patient-level data returned to the Operations Center, the PEPR program selects a limited window of data for each patient episode included in the PEPR datasets as determined by two additional index date variables. Macro parameters are used to define these index dates, which means they can be changed as appropriate for each run of the PEPR macro. The PEPR window, then, is defined as follows: "pre" index date – "pre" index days prior through "post" index date + "post" index days after.

The simplest case is to use the same date variable for both indices, as we might do for a TreeScan analysis by indexing the entire window to the exposure date. That is, we could define the window as 28 days prior to exposure through 56 days after exposure, giving us equal history and follow-up time for all patients, relative to exposure. If instead we wish to have equal follow-up time relative to the event, we would define the window as 28 days prior to exposure through 56 days after event.

Figure 4: Effect of Using Different Index Date Variables to Determine PEPR Window.



Scenario 1 ensures equal follow-up time after exposure and equal calendar time for all episodes but results in varied follow-up time after event. Scenario 2 ensures equal follow-up time after even but results in varying total calendar time per episode

This use of macro parameters to specify the index dates extends the utility of the PEPR macro to a wider variety of evaluations. For example, in a pregnancy-related study, investigators may use an algorithm to estimate the pregnancy start date. Similarly, they may also use an algorithm to estimate the delivery date in an effort to be more precise than simply using the admission date of the encounter containing codes that relate to delivery. The estimated pregnancy start date could serve as the "pre" index date and the estimated delivery date as the "post" index date simply by entering the names of these two variables in the appropriate macro parameters. Another pregnancy-related evaluation may wish to look at mother-baby pairs, where records in the input dataset include information on both mom and baby.

PEPR could be run twice within the same workplan, once to extract the mother's data as described above and a second time to extract the baby's data.[4,5]

3. Output Datasets

The PEPR output datasets include the modified CDM tables, as described above, and others that are not in CDM format, including the Core dataset and prevalence datasets for diagnosis, procedure, and dispensing. The Core dataset contains information pertaining to the index dates for each patient episode, most importantly the age grouping. All date values in the Core dataset are subject to the same security options as the other PEPR datasets. The prevalence datasets contain one record per unique combination of admission (or dispensing) date and clinical code, along with the number of days since the previous appearance of the same code in the patient's history. For first-observed, the prior days variable will have a missing value. The patient's entire history is used to determine the prior days value, but only those records occurring within the retrieval window will be included in the output datasets, which allows accurate categorization of diagnosis, procedure, and drug codes as first-observed vs. prevalent while adhering to the minimum-necessary data standard. Future development could improve the determination of prevalence by comparing codes across all tables, particularly since NDCs may appear in both the Dispensing and Procedure tables.

By retaining the CDM format of the main PEPR datasets, we allow for maximum flexibility and general use across any number of different types of evaluation. The familiar structure of the CDM tables can also make additional downstream programming easier and more efficient. A workgroup conducting a PBA with chart review could use these datasets in the chart selection process to determine which encounters the Data Partners should pursue and then use them as the basis for the chart review database. A different workgroup that finds a safety signal while conducting sequential analysis may also need to go to chart review, where PEPR could be used in the same manner. The general approach to both input and output allows for a wide range of applications, and the whole system benefits from access to a reliable, familiar tool while also saving resources associated with custom program development.

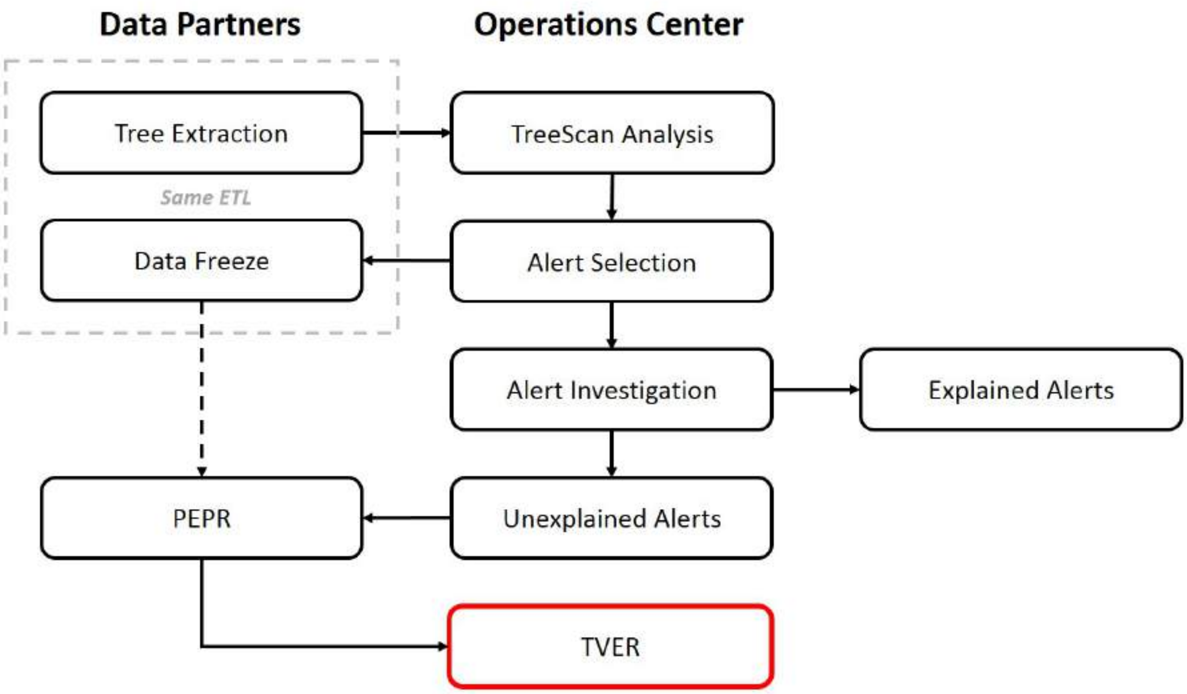
4. PEPR Workplan

The PEPR workplan requires a master program, the PEPR macro, various utility macros, and input macro parameters for each run of PEPR within the workplan. The PEPR macro itself may be run more than once within the same master program if multiple alerts are under investigation, saving the time and expense of multiple program runs by the Data Partners.

Auxiliary macros can be included in the workplan to prepare the input dataset, apply sampling or filters to the output datasets, or to retrieve additional information for output beyond the standard PEPR datasets. For TreeScan, we developed a macro to create and output two additional datasets for each alert. The first contains records from Crosswalk dataset with only the exact episodes which led to inclusion in the node, with the data de-identified in similar manner to the PEPR datasets. An optional variable may be added to this dataset with the calendar month of exposure to provide information on seasonality. The second output dataset is a copy of the Crosswalk dataset containing records of all incident events, regardless of whether they were related to the alert under investigation. This information will used in the report to help the clinician distinguish between events that were identified as incident by the Tree Extraction program, those that were otherwise first-observed, or those that were prevalent, i.e. neither incident nor first-observed.

5. Aim 3: TreeScan Vaccine Episode Report (TVER)

Figure 5: TreeScan Process Flow Adding TVER



The TVER answers the third aim of this activity to transform the PEPR output datasets into a format which facilitates review of vaccine-related TreeScan alerts. The report format is based in part on PRISM experiences with chart review studies, where chart selection reports are provided to clinician reviewers to prioritize encounters for which the Data Partners will pursue medical records. These reports have thus far been highly customized to each specific study population and exposure-outcome pair, but a more generic approach is necessary for TreeScan.

The report contains two sections: the Header with information pertaining to the overall vaccine episode; and the Detail with medical encounter and prescription drug dispensing information. The following sections describe the TVER after the format was modified following initial review during the MMR/MMRV beta-test.

6. TVER Data Preparation

A master dataset for each PEPR table is created by combining datasets from all Data Partners, and a new variable is added to distinguish between records from the different Data Partners. The unique values of this DataPartner variable are assigned a random numeric value and saved to a translation table, and then the programmer has the option to use the actual DataPartner value or the masked value in the report. Finally, a case identifier (CaseID) variable is added to provide a unique value to each patient-episode across all Data Partners.

7. TVER Header

The Header for each case consists of a single record containing the high-level information that relates to the patient-episode as a whole. Unique patient-episodes are identified by CaseID and DataPartner, masked or unmasked. The patient's demographics are represented by sex and age grouping at exposure. The calendar month of exposure and the number of days from exposure to event illustrate the timing of exposure and event, while the node, node description, and risk window of the alert are included to display overall alert information on every report screen.

The PEPR window gives the range of days captured in the PEPR datasets for each patient-episode, relative to exposure, showing the reviewer the range of potential days on which medical encounters and drug dispensings may be observed. If the exposure date has been used as both "pre" and "post" index, the PEPR window values will be the same for every patient-episode. If the event date is used as "post" index in order to ensure equal follow-up time after event for every episode, the PEPR window values will vary based on how many days after exposure the event occurred.

Example scenario 1: PEPR window selected as 28 days before exposure through 56 days after ***exposure***

Days between exposure and event	PEPR window start	PEPR window end
7	-28	56
28	-28	56
55	-28	56

Example scenario 2: PEPR window selected as 28 days before exposure through 56 days after ***event***

Days between exposure and event	PEPR window start	PEPR window end
7	-28	63
28	-28	84
55	-28	111

The coverage window variables serve the similar purpose of informing the reviewer of the potential range of days on which data may be seen in the Detail section, providing some assurance that data are not missing simply because the patient didn't have relevant coverage on certain days.

8. TVER Detail

The Detail section contains records from medical encounters and drug dispensings that occurred on any day within the PEPR window, with one record per unique combination of date (displayed as the number of days before or after exposure), clinical code (i.e. diagnosis, procedure, NDC), and encounter setting (blank for dispensing records). In addition to this basic information, we also include the length of stay (LOS) for inpatient encounters; primary diagnosis indication for inpatient encounters; incidence (incident as defined by the Tree Extraction program, first-observed in the patient's entire history, or prevalent); a node indicator to denote any diagnosis code which applies to the node under review; a main exposure indicator to denote any code which applies to the vaccine under review; an "any vaccine" indicator for any codes relating to vaccine administration; Rx days supply and Rx amount, relevant only to dispensing records; and coverage start and end dates (displayed as number of days relative to exposure) for the medical enrollment segment containing the admission date for medical encounter records or, for dispensing records, the drug coverage segment containing the dispensing date.

Since medical encounter and drug dispensing records are both included in the Detail section, the easiest way to recognize drug dispensing records is to look to the Rx days supply and amount variables, since these will only be populated for dispensing records. The setting (or encounter type) variable will also be missing for dispensing records.

In order to remove clutter in this potentially dense report, a blank is used as default value in several variables. For example, the incidence variable only shows a value for incident or first-observed records. If the code is prevalent, the variable is left blank. The same is true for the node, vaccine, and exposure indicators, where a value of 1 indicates yes and a blank indicates no.

9. Discussion

The current report is meant more as a prototype rather than a permanent solution. A SAS program is used to generate the Header and Detail datasets, but then the data are exported to Excel for presentation to the reviewer(s). This solution is not ideal, since it isn't flexible or scalable and can become quite cumbersome when more than a handful of patients are under review. Excel does have advantages in that it allows reviewers at least some minimal interactive capabilities to sort data, add custom flags, make electronic notes, etc. A more ideal solution would be web-based to enable real-time editing and sharing of custom flags, views, and notes between reviewers.

III. MMR/MMRV BETA TEST

In order to beta-test both local and distributed SAS programs, as well as to test the overall system using actual data to populate the TVER, we chose to re-enact an assessment of MMR and MMRV vaccines used in development of the Tree Temporal Scan statistic methods. We used three of the four original Data Partners (Harvard Pilgrim, HealthCore, and Humana) and the same time period that each contributed to the previous analysis on the assumption that no new alerts would be detected, since the power should be less than with the original four Data Partners. This allowed us to focus on the beta-test by using a previously studied vaccine and established set of statistical alerts. If the results were notably different, we could first suspect an issue with the SAS programs and review for bugs before proceeding.

The study period ranged from January 2004 through November 2011 for one Data Partner, June 2007 through October 2011 for another, and January 2000 through December 2011 for the third. MMR vaccine exposure were identified using ICD-9 procedure code 99.48, ICD-9 diagnosis code v06.4, and CPT4 code 90707. MMRV vaccine exposure was identified using CPT4 code 90170. Two age cohorts were selected, with group 1 representing 330 days to 760 days of age at exposure, roughly 11 to 25 months, and group 2 representing 1430 to 2220 days, roughly 47 to 73 months. The two vaccine exposures and two age groupings provided four analytic groups. The Tree Temporal Scan method was used for analysis, and the Tree Extraction program was set to identify and collect incident adverse events in the 1 to 56 days following the first-observed exposure in each analytic group.

After Tree Extraction datasets were returned from all Data Partner sites, the data were aggregated and run through the Tree Temporal Scan analysis. The Alert Selection package was then run on the results, with a total of 30 primary alerts selected according to pre-specified criteria of p-value less than or equal to .05 and relative risk greater than or equal to 1.2. We also applied a rule to select secondary alerts if the next-higher node on the tree for any primary alert had a p-value less than or equal to .20, but no additional nodes were selected with this criterion.

The selected alerts were mostly related to skin conditions (rash), convulsions (febrile), allergic reactions, and fluid and electrolyte disorders (nausea and vomiting). None of these were surprising, and in particular, the cluster of febrile seizures in 7 to 10 days following vaccination is consistent with findings in the literature.[13] The workgroup decided to manually add immune thrombocytopenic purpura (ITP) to the data freeze as a precaution, since excess cases were detected in the younger age group following MMR. However, the p-value was too high to meet the selection threshold.

From this set of alerts, two were chosen for the beta-test of the PEPR program, with both occurring in age group 1 (330 to 760 days). The first, alert 25 for Nausea and Vomiting, represented ICD-9 diagnosis codes 787.0, 787.01, 787.02, 787.03, and 787.04. There were 14 cases in the risk window of 5 to 9 days after MMRV exposure and 39 cases overall. The TreeScan workgroup decided to retrieve data from four weeks, or 28 days, prior to exposure through eight weeks, or 56 days, after the exposure to ensure capture of the entire 1 to 56 day exposure window and to gather uniform calendar time for all patients. PEPR was run at all three Data Partners for this alert.

Table 1: Nodes related to Alert 25 (Convulsions in 7 – 10 days after MMR vaccine exposure)

Alert ID	Vaccine	Age group	Node	Node description	Total node cases	Risk window	Observed	Expected	Relative risk	Excess Cases	P-value
26	MMRV	1	17.01.06	Nausea and vomiting	39	5 - 9	14	3.48	5.71	11.55	0.01856
25	*MMRV*	*1*	*17.01.06.00*	*Nausea and vomiting*	*39*	*5 - 9*	*14*	*3.48*	*5.71*	*11.55*	*0.01856*
24	MMRV	1	787.03	Vomiting alone	34	5 - 9	13	3.04	6.31	10.94	0.01779

The second, alert 4 for Convulsions, represented ICD-9 diagnosis codes 780.3, 780.31, 780.32, 780.33, and 780.39. There were 80 cases in the risk window of 7 to 10 days after MMR exposure and 290 cases overall. The TreeScan workgroup decided to retrieve data from eight weeks (56 days) prior to exposure through twelve weeks (84 days) after the event to ensure uniform follow-up to event for all patients. Because of the relatively high number of cases, we chose to send run PEPR for this alert at only two Data Partners, which limited the number of cases for review to 13.

Table 2: Nodes related to Alert 4 (Convulsions in 7 – 10 days after MMR vaccine exposure)

Alert ID	Vaccine	Age group	Node	Node description	Total node cases	Risk window	Observed	Expected	Relative risk	Excess Cases	P-value
2	MMR	1	06	Diseases of the nervous system and sense organs	678	7 - 11	156	60.54	3.05	104.8	0.00001
1	MMR	1	06.04	Epilepsy; convulsions	317	7 - 10	88	22.64	5	70.38	0.00001
4	*MMR*	*1*	*06.04.02*	*Convulsions*	*290*	*7 - 10*	*80*	*20.71*	*4.95*	*63.85*	*0.00001*
3	MMR	1	06.04.02.00	Convulsions	290	7 - 10	80	20.71	4.95	63.85	0.00001
10	MMR	1	780.31	Febrile convulsions simple unspec	178	7 - 10	50	12.71	5.08	40.15	0.00001
11	MMR	1	780.39	Other convulsions	95	7 - 9	25	5.09	6.31	21.04	0.00001

For both alerts, the following options were selected:

1) Set the patient birth date to missing.
2) Set the patient and facility ZIP codes to missing.
3) Categorize age at exposure in three-month bands, i.e. 0M-12M 12M-14M 15M-17M 18M-20M 21M-23M 24M+, where the first and last groupings represent the tails of the overall age cohort.
4) Set all calendar dates relative to the exposure date.
5) Include the calendar month of exposure in the PEPR Core output table.
6) Output the Demographic, Enrollment, Encounter, Diagnosis, Procedure, and Dispensing tables.
7) Output the PEPR Core table.
8) Run the Alert Crosswalk Retrieval auxiliary macro to output de-identified copies of the two alert crosswalk datasets described in the PEPR section above.

The TVER was produced for both alerts, and since neither outcome was surprising, the group did not conduct a serious review of the clinical data but instead reviewed the reports to tune format and content as well as to gain familiarity with actual data. This initial review led to modifications of the TVER format to collapse separate tables for medical encounters and drug dispensing detail into a single Detail section, add the PEPR window information to the Header section, and set all default values in the Detail section to blank to remove clutter.

The group also used this review as an opportunity to evaluate whether these data could be useful for the stated purpose of determining the likelihood of an event being caused by vaccine, mostly as a "likely rule-out" tool. As an example, we have included below in Tables 3 and 4 the TVER for a composite case from the nausea and vomiting alert.

In the header we can see from the "Days from expos to event" variable that the nausea and vomiting event occurred 6 days after exposure to MMRV vaccine, placing this case within the 5 to 9 day risk window identified by Tree Temporal Scan analysis. In the detail section, we see that the patient received the MMRV vaccination as part of a routine visit and that PCV7 vaccine was also given on the same day.

Four days after exposure, the patient had another ambulatory visit encounter, where we see a first-observed diagnosis code of 009.0 (infectious colitis, enteritis, and gastroenteritis), a code that was pruned from the tree because the cause is specifically noted as infection.

Three days later, the patient was apparently taken to the emergency department and then admitted to the hospital with the primary diagnosis of 276.51 (dehydration) as well as secondary diagnosis of 787.03 (vomiting alone), the incident code which led to inclusion in this particular node. The patient was administered fluids through IV, and multiple tests were ordered. The patient stayed overnight, as indicated by the value of 1 in the LOS (length of stay) variable. Note that all four diagnosis codes listed for this inpatient encounter, including 786.2 (cough) and 535.50 (unspecified gastritis and gastroduodenitis without mention of hemorrhage), were identified as incident by the Tree Extraction program, but only the code for vomiting belongs to the node for this alert.

Two days after discharge, the patient had one more ambulatory visit encounter where the only diagnosis given was again 009.0, which may have been based on confirmation from test results or simply as an indication of history from the previous ambulatory visit on day 4 after exposure.

Although the patient had medical and drug coverage far beyond the end of the PEPR window (56 days after exposure), no further follow-up visits were observed during that time.

Based on these data, we would probably conclude the vomiting episode was more likely due to gastro-intestinal infection than the exposure to MMRV vaccine.

Table 3: TVER example header and detail for composite patient

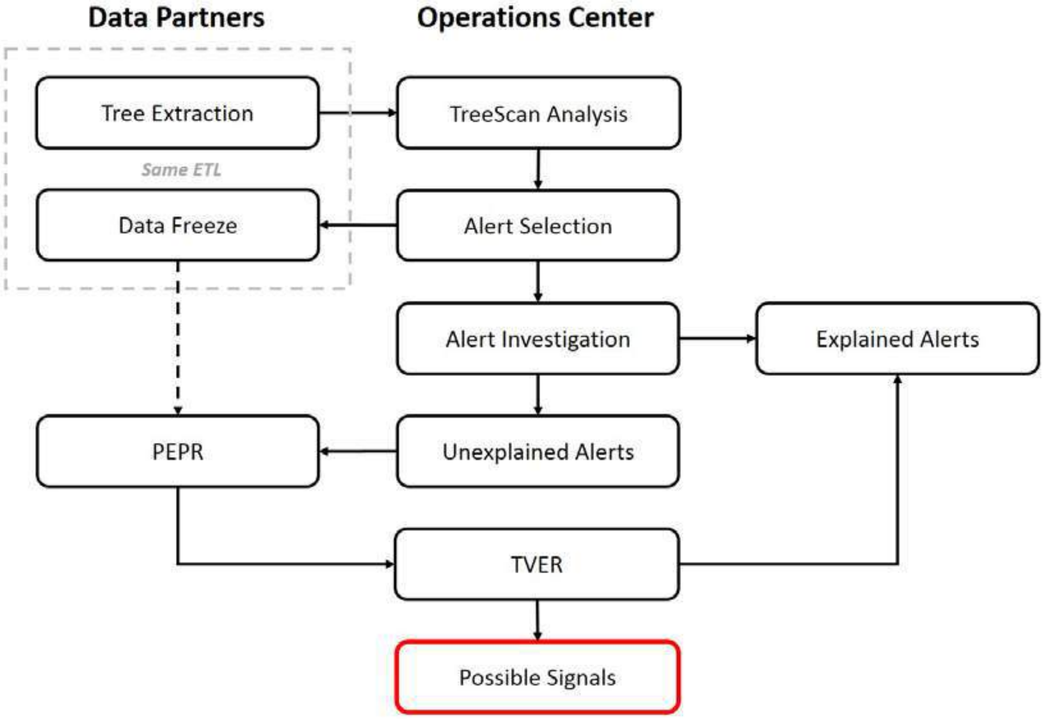
Episode Header		~ Coverage refers to the enrollment segment containing the admission date of the encounter													
Node	Node desc	Data Partner	Case ID	Sex	Age band at expos	Month of expos	Days from expos to event	Risk win start	Risk win end	PEPR win start	PEPR win end	Med cov start~	Med cov end~	Drug cov start~	Drug cov end~
17.01.06.00	Nausea and vomiting	1	40	F	12M-14M	JAN	7	5	9	-28	56	-386	1260	-386	1260

Table 4: TVER example header and detail for composite patient

Episode Detail						^ Incidence: F = first observed; I = incident; blank = prevalent # Primary Dx: P = primary; S = secondary; X = N/A ~ Med enroll segment containing the admission date of the encounter or the drug enroll segment containing the dispensing date									
Days from expos	Enc type	LOS	Clinical code			Code description	Incidence^	P Dx#	Node (Y/N)	Main expos (Y/N)	Any vacc (Y/N)	Rx days supp	Rx amt	Cov start~	Cov end~
			Cat	Type	Code										
0	AV		DX	09	V0382	Need Proph Vacc Agnst Strep Pne					1			-386	1260
0	AV		DX	09	V068	Need Proph Vacc Against Oth Comb Dz	F				1			-386	1260
0	AV		DX	09	V202	Routine Infant/Child Health Check								-386	1260
0	AV		PX	C4	90471	Immunization Admin	F				1			-386	1260
0	AV		PX	C4	90472	Immunization Admin Each Add	F				1			-386	1260
0	AV		PX	C4	90669	PCV7 Vaccine Im					1			-386	1260
0	AV		PX	C4	90710	MMRV Vaccine Sc	F			1	1			-386	1260
0	AV		PX	C4	99392	Prev Visit Est Age 1-4	F							-386	1260
4	AV		DX	09	0090	Inf Colitis Enterit & Gastroenterit	F							-386	1260
4	AV		PX	C4	99213	Office/Outpatient Visit Est	F							-386	1260
7	IP	1	DX	09	27651	Dehydration	I	P						-386	1260
7	IP	1	DX	09	53550	Uns Gastrit & Gastroduodit No Hemorr	I	X						-386	1260
7	IP	1	DX	09	7862	Cough	I	X						-386	1260
7	IP	1	DX	09	78703	Vomiting Alone	I	S	1					-386	1260
7	IP	1	PX	C4	71020	Chest X-Ray 2Vw Frontal & Latl	F							-386	1260
7	IP	1	PX	C4	74000	X-Ray Exam Of Abdomen	F							-386	1260

IV. CONCLUSIONS AND RECOMMENDATIONS

Figure 6: TreeScan Process Flow with Full Programming and Data Management Infrastructure



The data captured by PEPR and displayed in TVER can fill the gap in our investigative toolset between broader preliminary investigation and full-scale PBAs with chart review by:

1) Providing clinical context
    a. on key co-morbidities, pre-exposure events, and risk factors that occur before the exposure of interest;
    b. on the temporal sequence of events as coded in administrative data;
    c. on key medical procedures and fuller details of the clinical evaluation on the actual date of the event, to potentially increase or decrease the index of suspicion about the accuracy of the code;
    d. on key follow-up healthcare encounters and their evaluation that might increase the precision of a health outcome of interest (i.e., if the patient subsequently is treated with anti-coagulants, this increases the likelihood that the original VTE event was accurate)
2) Providing stakeholders an additional opportunity to increase the precision when alerts contain ambiguous diagnostic codes or codes that contain numerous actual diagnoses
3) Displaying potential for use as a follow-up tool for statistical signals emerging from use of the Routine Querying System (http://www.mini-sentinel.org/data_activities/modular_programs/details.aspx?ID=166) along with standard analytical tools such as Prospective Routine Observational Monitoring Program Tools (PROMPT) (http://www.mini-sentinel.org/methods/methods_development/details.aspx?ID=1045)

4) Maintaining consistency with Congressional FDAAA mandate to maximize the use of electronic healthcare data for active risk identification and risk analysis (ARIA)
5) Allowing FDA to prioritize finite resources for chart review by using these tools to refine and triage alerts

This last point is particularly important, as triage of statistical alerts by stakeholders will likely **decrease** the need to obtain more granular, individual-level health information that would be required as part of medical record review.

For wider applications, the TVER format could be modified for use with other types of exposures beyond vaccines or other types of evaluations beyond data mining. For drug exposures, dispensing records could be grouped into dispensing episodes with or without allowances for stockpiling, and variables could be added to describe drug therapeutic classes and other higher-level groupings of NDCs that would be of interest to reviewers. Such development could include a report for use with DrugScan (http://www.mini-sentinel.org/methods/methods_development/details.aspx?ID=1061), an outcome-based version of TreeScan that chooses a single adverse event and looks for unanticipated exposures that may be associated with the adverse event. Functionality could be added to the existing Routine Querying System to enable use of PEPR to gather data for initial review or for chart review of statistical alerts or signals resulting from standard analytical tools such as PROMPT.

Alerts involving a large number of patients present problems that should be addressed through further development. The potential need to gather patient-level data, even if de-identified, on hundreds of patients to investigate a data mining alert presents a challenge to the principle of minimum-necessary data disclosure, and TVER in its current state has limited functionality, making high-volume review difficult, particularly for more complex outcomes.

These issues could be addressed by:

1) developing standard aggregate analyses, such as rank order listing of most frequent diagnosis, procedure, and drug codes, to be run on the frozen datasets behind the firewall;
2) adding an auxiliary macro for use with PEPR to take a random sample from each Data Partner;
3) adding an auxiliary macro for use with PEPR to subset the cases according to specific criteria, such as limiting to cases in the risk window (Tree Temporal Scan) or limiting to exposed cases (Poisson);
4) enhancing TVER and adding more sophisticated reporting tools at the Operations Center to facilitate review of larger numbers of cases and of more complex outcomes;
5) working with the Data Partners to adjudicate some or all cases at the sites.

Ultimately, the infrastructure developed through this activity and described in this report can only remain effective if ongoing maintenance and enhancements are included as part of standard operating procedures to ensure continued compatibility with all current and future TreeScan activities. Since this activity only had resources to address one analysis method, the Tree Temporal Scan, all pieces of the infrastructure should be assessed, adapted, and implemented for other existing methods, including Poisson and self-controlled. Any modifications or enhancements to Tree Extraction, the diagnosis tree (including implementation of an ICD-10 tree or a combined ICD-9/ICD-10 tree), or the TreeScan analysis results file should be communicated in advance to the responsible programmer to ensure compatibility with existing infrastructure.

V. APPENDIX A: TREE EXTRACTION OUTCOME CROSSWALK DATA DICTIONARY

Variable	Data Type	Format	Label	Valid Values	Source/Comments
PatID	C(varies by Site)	$##.	Patient ID	Unique alpha-numeric identifier	Pseudoidentifier assigned to a unique patient by the data partner and which can be used to link across tables in the MSCDM
Dx	C(18)	$18.	Diagnosis code	Valid diagnosis code	Code used to identify incident outcome event
Dx_codetype	C(2)	$2.	Diagnosis code type	09 = ICD-9-CM 10 = ICD-10-CM 11 = ICD-11-CM SM = SNOMED CT OT = Other	Includes all code types for current and possible future use.
DX_ADate	N(4)	MMDDYY 10.	Diagnosis admission date	Valid calendar date within parameters of the study period	ADate of the encounter from which the incident diagnosis was identified.
Exp_ADate	N(4)	MMDDYY 10.	Exposure admission date	Valid calendar date within parameters of the study period. Will be blank if the outcome did not occur during the risk or control window.	ADate of the encounter from which the exposure was identified. For AV and ED, this generally corresponds to the date of service. For IP, this is the admission date but not necessarily the date of service.
Group	C(30)	$30.	Group	alpha-numeric	Describes analysis groupings, usually a combination of vaccine exposure and age

VI. APPENDIX B: PEPR ALTERATIONS TO CDM TABLE VARIABLES

CDM Table	Variable	Transformation
All	_PatID	Always masked; crosswalk retained in local folder
Demographic	Birth_date	If prefixed with underscore, set to missing. Otherwise, original value
Demographic	Zip	If prefixed with underscore, set to missing <u>or</u> converted to postal state abbreviation, depending on macro parameter selection. Otherwise, original value
Demographic	Zip_Date	If prefixed with underscore, calculated as Zip_Date – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
Enrollment	Enr_start	If prefixed with underscore, calculated as Enr_start – relative index date value, i.e. days +/- the relative index date. Otherwise, original value

CDM Table	Variable	Transformation
Enrollment	Enr_end	If prefixed with underscore, calculated as Enr_end – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
Encounter, Diagnosis, Procedure	_EncounterID	Always masked; crosswalk retained in local folder
Encounter, Diagnosis, Procedure	ADate	If prefixed with underscore, calculated as ADate – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
Encounter	DDate	If prefixed with underscore, calculated as DDate – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
Encounter, Diagnosis, Procedure, State Vaccine	_Provider	Always masked; crosswalk retained in local folder
Encounter	Facility_location	If prefixed with underscore, set to missing or_converted to postal state abbreviation, depending on macro parameter selection. Otherwise, original value
Encounter, Lab Results	_Facility_code	Always masked; crosswalk retained in local folder
Dispensing	RxDate	If prefixed with underscore, calculated as RxDate – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
State Vaccine	SIIS	If prefixed with underscore, set to missing Otherwise, original value
State Vaccine	VaxDate	If prefixed with underscore, calculated as VaxDate – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
Lab Results	Order_dt	If prefixed with underscore, calculated as Order_dt – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
Lab Results	Lab_dt	If prefixed with underscore, calculated as Lab_dt – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
Lab Results	Result_dt	If prefixed with underscore, calculated as Result_dt – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
Vital Signs	Measure_date	If prefixed with underscore, calculated as Measure_date – relative index date value, i.e. days +/- the relative index date. Otherwise, original value
Death	DeathDate	If prefixed with underscore, calculated as DeathDate – relative index date value, i.e. days +/- the relative index date. Otherwise, original value

VII. ACKNOWLEDGEMENTS

We gratefully acknowledge the contributions of the following organizations and individuals:

Harvard Pilgrim Health Care Institute: Jeffrey Brown, Claudia Colonel-Moreno, Alison Kawai, Matt Lakoma, Jen Popovic, Lisa Ortendahl, Megan Reidy, Schyler Versteeg, Tiffany Woodworth, Diana Santiago, Robert Rosofsky

HealthCore: Bolarinwa Ekezue, Chunfu Liu, Fang Tian

Humana: Mano Selvan, Yunping Zhou

VIII. REFERENCES

1. Yih WK, Lieu TA, Kulldorff M, Martin D, McMahill-Walraven CN, Platt R, Selvam N, Selvan M, Lee GM, Nguyen M. Intussusception Risk after Rotavirus Vaccination in U.S. Infants. N Engl J Med, 2014, 370:503-12.
2. Yih WK, Zichittella LJ, Greene SK, et al. Evaluation of the Risk of Venous Thromboembolism After Gardisil Vaccination. Report to FDA; April 23, 2015. (Accessed at http://mini-sentinel.org/work_products/Assessments/Mini-Sentinel_PRISM_Gardasil-and-Venous-Thromboembolism-Report.pdf)
3. Kawai AT, Martin D, Kulldorff M, Li L, Cole DV, McMahill-Walraven CN, Selvam N, Selvan MS, Lee GM. Febrile Seizures After 2010–2011 Trivalent Inactivated Influenza Vaccine. Pediatrics, 2015-0635.
4. Kawai AT, Li L, Kulldorff M, et al. Mini-Sentinel CBER/PRISM Surveillance: Influenza Vaccines and Pregnancy Outcomes. Protocol for FDA; March 28, 2014. (Accessed at http://www.mini-sentinel.org/work_products/PRISM/Mini-Sentinel_PRISM_Influenza-Vaccines-and-Pregnancy-Outcomes-Protocol.pdf)
5. Kawai AT, Li L, Andrade SE, et al. Mini-Sentinel/CBER Protocol: Influenza Vaccines and Birth Outcomes. Protocol for FDA; December 5, 2014. (Accessed at http://www.mini-sentinel.org/work_products/PRISM/Mini-Sentinel_PRISM_Influenza-Vaccines-and-Birth-Outcomes-Protocol.pdf)
6. Yih WK, Zichittella LJ, Sandhu SK, et al. Accessing the Freshest Feasible Data for Conducting Active Influenza Vaccine Safety Surveillance. Report to FDA; April 8, 2015. (Accessed at http://www.mini-sentinel.org/work_products/PRISM/Mini-Sentinel_PRISM_Active-Influenza-Vaccine-Safety-Surveillance-Report.pdf)
7. TreeScan Extension (PRISM). Methods project for FDA; in progress. (Accessed at http://www.mini-sentinel.org/methods/methods_development/details.aspx?ID=1056)
8. TreeScan Power. Methods project for FDA; in progress. (Accessed at http://mini-sentinel.org/methods/methods_development/details.aspx?ID=1055)
9. Yih WK, Nguyen M, Maro JM, et al. Mini-Sentinel CBER/PRISM Surveillance: Pilot of Self-Controlled Tree-Temporal Scan Analysis for Gardasil Vaccine. Protocol for FDA; March 30, 2015. (Accessed at http://mini-sentinel.org/work_products/PRISM/Mini-Sentinel_PRISM_Pilot-Self-Controlled-Tree-Temporal-Scan-Analysis-Gardasil-Vaccine-Protocol.pdf)
10. McClure DL, Raebel MA, Yih WK, Shoaibi A, Mullersman JE, Anderson-Smits C, Glanz JM Mini-Sentinel methods: framework for assessment of positive results from signal refinement. Pharmacoepidemiology and Drug Safety. 2014;23:3-8.
11. Yih WK, Kulldorff M, Fireman BH, et al. Active surveillance for adverse events: the experience of the Vaccine Safety Datalink Project. Pediatrics. 2011;127:S54–64.
12. Kulldorff M. TreeScan User Guide, v1.1. November, 2014; 37-38. (Accessed at http://www.treescan.org/cgi-bin/treescan/register.pl/treescanv1.1-userguide.pdf?todo=process_userguide_download)
13. Klein NP, Fireman B, Yih WK, Lewis E, Kulldorff M, Ray P, Baxter R, Hambidge S, Nordin J, Naleway A, Belongia EA, Lieu T, Baggs J, Weintraub E, for the Vaccine Safety Datalink. Measles-mumps-rubella-varicella combination vaccine and the risk of febrile seizures. Pediatrics, 2010, 126, e1-8.

C. Sentinel Training #1

Scan the QR code included to access the weblinks, videos and slide deck for the Sentinel Training Day 1 Event.

Note: FDA makes no promise on the longevity of the information link/ QR code as provided. Kindly key in the weblink provided below, if needed, as an additional means to access information for Sentinel Training #1

https://www.sentinelinitiative.org/communications/sentinel-initiative-events/public-sentinel-training-fda

D. Sentinel Training #2

Scan the QR code included to access the weblinks, videos and slide deck for the Sentinel Training Day 2 Event held at FDA on 08 FEB 2018.

Note: FDA makes no promise on the longevity of the information link/ QR code as provide. Kindly key in the weblink provided below, if needed, as an additional means to access information for Sentinel Training #2

https://www.sentinelinitiative.org/communications/sentinel-initiative-events/sentinel-initiative-public-workshop-tenth-annual-day-2

E. KEY PRESENTATIONS, SYMPOSIA AND WORKSHOPS FROM ICPE 2017

i. https://www.sentinelinitiative.org/communications/publications/2017-icpe-plenary-medical-product-and-performance-evaluation-programs

ii. https://www.sentinelinitiative.org/communications/publications/2017-icpe-symposium-integrating-sentinel-routine-regulatory-drug-review

iii. https://www.sentinelinitiative.org/communications/publications/2017-icpe-presentation-promises-and-challenges-screening-adverse-events

iv. https://www.sentinelinitiative.org/communications/publications/icpe-2017-workshop-treescan-novel-data-mining-tool-medical-product

Note: FDA makes no promise on the longevity of the information links/ QR codes as provided.
Kindly key in the weblinks shared above, if needed, as an additional means to access information for the respective ICPE 2017 presentations, symposia and workshops as aforementioned.

Mini-Sentinel Common Data Model
Guiding Principles
November 2010
Version 1.0

Introduction

The primary goal of the Mini-Sentinel pilot is to build and operate a national public health surveillance system to improve the safety of FDA-regulated medical products, including drugs, biologics, and devices. Mini-Sentinel is a major element of the Sentinel Initiative, the FDA's response to a Congressional mandate to create an active surveillance system using electronic health data for 25 million people by 2010 and 100 million people by 2012.

The Mini-Sentinel pilot will undertake three major types of activities: (1) prospective evaluation of accumulating experience about specific medical products and specific suspected safety problems; (2) evaluation of the impact of FDA actions (e.g., labeling changes) on medical practice and health outcomes; and (3) rapid assessment of past experience in response to FDA questions about specific medical product exposures and health outcomes.

A wide range of Collaborating Institutions will provide access to data environments and other resources, including expertise, as needed to meet the epidemiologic requirements of Mini-Sentinel. In addition, representatives of the Collaborating Institutions will provide ongoing scientific, technical, and methodological expertise by participating in Mini-Sentinel in various capacities, including as members of the Planning Board, the Safety Science Committee, the three Mini-Sentinel Coordinating Center Cores (Data, Methods, and Protocol), and various Mini-Sentinel workgroups.

Mini-Sentinel uses a distributed data model [1-3] that gives Data Partners complete autonomy over access to and use of data in their possession. The distributed model requires development and implementation of a common data model to allow a single analytic program to be distributed and run identically in each data environment.

The Mini-Sentinel Coordinating Center (MSCC) Data Core coordinates the network of Data Partners and leads development and utilization of the Mini-Sentinel Common Data Model (MSCDM), a standard data structure that allows Data Partners to quickly execute programs against their local data. In addition, the MSCC Data Core facilitates creation of the individual Mini-Sentinel Distributed Databases (MSDD) at Data Partner sites using the MSCDM. The Data Core also works closely with the MSCC Methods and Protocol Cores. The MSDD refers to the data held and maintained by the Data Partners in the MSCDM format.

This document describes the Guiding Principles of the MSCC Data Core as well as the initial priorities and approach to the MSCDM.

Guiding Principles

The MSCC Data Core coordinates the network of data partners who actively participate in the creation, implementation, updating, maintenance, enhancement, and use of the MSCDM and their MSDDs. The design and implementation of the MSCDM strives for a high level of cross-institutional and longitudinal consistency and requires that data comparable in format and meaning are stored at all sites.

The following principles guide the development and maintenance of the MSCDM:

1. The MSCDM accommodates all requirements of Mini-Sentinel activities and may change to meet FDA objectives.

2. The MSCDM is able to incorporate new data types and data elements as needs indicate.

3. Development of the initial MSCDM and all enhancements requires input and acceptance from the Mini-Sentinel Data Partners.

4. Documentation of Data Partner specific issues and qualifiers that may impact use and interpretation of the data is crucial for the effective operation of Mini-Sentinel activities.

5. The MSCDM design is transparent, intuitive, well-documented, and easily understood by analysts, investigators, and stakeholders. It is easy for experienced analysts and investigators to use; special skills or knowledge beyond those commonly found among pharmacoepidemiologists and professional analytic staff is not necessary.

6. The MSCDM leverages evolving healthcare coding standards.

7. The MSCDM captures values found in the source data. When necessary, mapping to standard vocabularies is transparent. Validated mappings should be used whenever available.

8. Calculated variables should not be included in the MSCDM.

9. Distributed programs should be executed with minimal to no site-specific modification.

10. Data Partners have the best understanding of their data and its uses; valid use and interpretation of findings requires input from the Data Partners.

11. Only the minimum necessary information should be used and shared with authorized staff of the MSCC.

12. Data Partners may include “site-specific” information in their implementation of the MSCDM.

Initial Priorities and Approach to the Mini-Sentinel Common Data Model (v1.0)

The overall goal of Version 1.0 of the MSCDM is to build the foundation for Mini-Sentinel to begin active surveillance activities and to have the capability to quickly generate information in response to urgent public health needs. Initial functionality will rely on claims and administrative data with additional functionality to be added in subsequent years.

In order to achieve this goal, Version 1.0 of the MSCDM will be implemented by Data Partners representing at least 25 million lives. It was agreed that Version 1.0 should:

i. Reflect the guiding principles
ii. Focus on claims and administrative data elements
iii. Leverage the cumulative experience of the data partners
iv. Rely on existing and standardized coding schemas (e.g., ICD-9-CM, HCPCS/CPT, and NDC)
v. Be compatible with claims-based components of existing CDMs (e.g., Observational Medical Outcomes Partnership, HMO Research Network Virtual Data Warehouse)
vi. Include all of the data elements necessary to achieve the goals for Year 1 of the Mini-Sentinel pilot

Revisions and enhancements to the MSCDM are expected in subsequent years, including the addition of clinical information, incorporation of other data types and sources, and revisions based on lessons learned from use of the MSDD and other programs' CDMs. This may include adopting variables and formats developed by other programs.

References

1. Maro JC, Platt R, Holmes JH, *et al*. Design of a National Distributed Health Data Network. Ann Intern Med 2009;151:341-344.
2. Brown JS, Lane K, Moore K, *et al*. Defining and Evaluating Possible Database Models to Implement the FDA Sentinel Initiative; U.S. Food and Drug Administration: FDA-2009-N-0192-0005.2009.
3. Velentgas P, Bohn R, Brown JS, *et al*. A distributed research network model for post-marketing safety studies: the Meningococcal Vaccine Study. Pharmacoepidemiology and Drug Safety 2008;17:1226-1234.