The following is a draft version of a SEDD Implementation Guide.
This document is intended to help data requesters, data generators, and data recipients of SEDD deliverables in understanding SEDD requirements. It should also assist them in setting up criteria for the requesting, generating, and reviewing of SEDD files and ensuring that the major element of the SEDD files are program neutral in order to facilitate data exchange.

This document will be will be discussed at the SEDD workshop at PITTCON 2004.

We appreciate your comments on any aspect including format, use of data elements and nodes and valid values.

Please let us know whether you are a data requestor (Federal, State, Other Government Agency, private firm), Data Generator (Laboratory, LIMS Vendor), or Data Reviewer (Software developer to check EDDs).

Please send e-mail your comments to Anand Mudambi at mudambi.anand@epa.gov

The SEDD Implementation Guide for Stage 2a.

Draft 1.0 February 27, 2004

The Staged Electronic Data Deliverable (SEDD) delivers data as an XML document. Due to the complexity of the SEDD hierarchal structure, various stages have been defined that will deliver this data at various levels of detail as needed by a data requestor. A typical Stage 2a deliverable will only use eight of the twelve parent nodes available within SEDD at four of the six total hierarchal levels within SEDD and will only use 118 of the 363 total available data elements. For a Stage 2a, the data delivered will consist of the reported sample values along with the summary method quality control information. This guide will focus on those data elements required to maintain the correct relationships between the various nodes, preserve data integrity, and provide traceability.

This guide will address specific purposes, uses, and definitions from three different perspectives:

a. The first perspective is that of the data requestor, who will be the ultimate user of the sample data. The data requestor is responsible for defining the specific Data Quality Objectives (DQOs) for their specific project. In addition, the data requestor will usually specify the level of detail required for all reported data based on the ultimate data use. Data may be requested at a Stage 1, Stage 2a, Stage 2b, Stage 3, or Stage 4.

b. The second perspective is that of the data provider who creates the actual SEDD deliverables (files). Typically this would be the laboratory that analyzes the samples, working with their LIMS support vendors.

c. The third perspective is that of the data recipient, who must perform automated reviews of the data to support data assessment/validation against the project specific DQOs. The data recipient may also parse the reviewed data into other databases or software applications.

Please note that it is possible for one party to fill more than one role. The initial data recipient may provide updated SEDD files to additional recipients, and the initial data requestor may be the first or the ultimate data recipient.

Required Data Elements: These data elements must be present (and in some cases must also use the SEDD Valid Values) in order for the data to be understood and exchanged between various parties.

There will be a section explaining the different types of required data elements from the SEDD Specification:

Required
Conditionally Required
Required for Traceability
Required for Traceability of QC
Required for Portability
Required for Portability of QC

Level I: <Header> Node

Purpose: The <Header> node conveys administrative information about the contents of the Electronic Data Deliverable (EDD) itself.

Use: For every set of data generated by a single method (i.e., Volatile Organics by 8260B, Pesticides by 8081A, Metals by 6010C), a single <Header> node is required. The 'set' of data reported could be all the data for a given project or the 'set' could be multiple sets of data reported by a single method using separate unique <Header> nodes for each 'set' reported. Either approach is valid. But, in either case, the processing of a SEDD file continues until the next <Header> node is encountered, then, the processing of the file must begin from scratch. From this point of view, it is generally more efficient to place all of the project data in a single <Header> node for a given method rather than spread it across several <Header> nodes to minimize the information that may need to be repeated.

Data Elements:
1. <EDDID> (Required): A value that specifies the format of an electronic data deliverable.

This data element allows data recipients to distinguish SEDD Deliverables from other XML based deliverables. As such data generators must ensure that this element is present and contain the valid value 'SEDD'. Data recipients must be able to scan the <Header> node for this data element to determine if the file being evaluated is based on SEDD.

Allowed valid value(s):
'SEDD'

2. <EDDVersion> (Required): A value that specifies the version of the format of an electronic data deliverable.

This data element allows data recipients to distinguish the specific version of the SEDD Specification that was used to create this deliverable. As such data generators must ensure that this element is present and contain the valid value of the version of the SEDD Specification that is required. The most recent version of the SEDD Specification is 'Draft 5.0'. This document can be found at the following website: www.epa.gov/superfund/programs/clp/sedd.htm

Note: The version of the SEDD Specification to be followed must be (or is recommended) to be decided by the data requestor in conjunction with the data generator.

Allowed valid value(s):
'Draft 4.1', 'Draft 4.2', 'Draft 5.0', etc.

3. <EDDImplementationID> (Required): A value that identifies the specific implementation of an electronic data deliverable.

This data element allows data generators to specify the exact Document Type Definition (DTD) or Schema that was used when the SEDD file was created. The DTD or Schema specifies what part of the overall SEDD structure and what specific data elements will be used. The data generators and data recipients can also use this information to structurally validate the file against the requirements of the data requestor. The valid value for the SEDD Stage 2a is 'General_2a'.

Allowed valid value(s):
'General_2a'

4. <EDDImplementationVersion> (Required): A value that identifies the version of the specific implementation of an electronic data deliverable.

This data element allows data recipients to distinguish the specific version of the DTD or Schema that was used to create this deliverable. An example valid value for the SEDD Stage 2a might be '1'.

Allowed valid value(s):
'1', '2', '3', etc.

5. <DateFormat>: A value that specifies the format of all date/time values in an electronic data deliverable.

This data element provides the date/time format that was used for all date/time data elements in the deliverable.

Unless otherwise specified by the data generator and/or data recipient, the default SEDD valid value for this data element is 'YYYYMMDD HHMMSS'.

Data generators must ensure that all reported date/time values are consistent with the format specified in this data element. Data generators must also ensure that the specific format used is consistent with the needs of the data requestors and data recipients.

Allowed valid value(s):
See Comprehensive list at end of this document.

Level II: <SamplePlusMethod> Node

Purpose: The <SamplePlusMethod> node conveys information about a given reportable sample (regular or QC) and provides the information that associates the reportable sample with its client or lab assigned original identifier along with it's matrix to the primary analytical method that was used to analyze the sample.

Use: For each <SamplePlusMethod> node used there must be a single set of final results reported. All subsequent dilutions, reextractions or reinjections would be noted under this node, however, only a single set of final results would still be reported. All reported results could come from a single analysis or the reported results could come from various analyses that were conducted. If a particular program might require the individual reporting for all separate analyses conducted, then multiple <SamplePlusMethod> nodes would be required, one for each separate analysis performed. Specific instructions must be received from the data requestors as to how these types of results should be reported. The data recipients may also have to be consulted to ensure that the data can be read by the specific software programs that may be used.

Data Elements:

1. <ClientMethodID> (Required): A client defined code for the method used by the lab to analyze the sample. The complete code for each method may be a composite of the method source along with the method number.

This data element is defined by Data Requesters in conjunction with Data Recipients.

This data element allows data recipients and data requestors to determine what primary analytical method was used by the laboratory to analyze this sample. To ensure ease of information interchange between data requestors, the default naming convention to be followed is SourceofMethod_MethodNumberandVersion. The data requestors must specify the specific valid values for this data element since this is the code the client uses specify and recognize a given method.

Allowed SEDD valid value(s):
See comprehensive list at the end of this document. Examples: 'SW846_8270C', 'SW846_8260B', 'SW846_8081A', 'SW846_6010C', etc.

Note: Some of this information may be moved to another data element.

2. <ClientSampleID> (Required): A client defined identifier for a sample.

This data element is the basis on which the data requestor identifies the sample. The data requestor may or may not specify these identifiers for laboratory generated QC samples, such as Method Blanks. If the data requestor does not provide the guidance for the naming of these types of QC samples, then the data generator must provide the unique identifier.

Allowed valid value(s):
Not applicable.

3. <LabID> (Required): Identifier for the lab doing this analysis.

This data element is often defined by the data requestor.

4. <LabReportingBatch>: A lab defined identifier for a batch of samples reported as a group by the lab. In addition, this batch can be used to link certain QC samples to regular ones.

This data element allows the data generators to associate samples that are to be reported together. The data requestor must specify specific instructions as to how this data element will or will not be used.

Allowed valid value(s):
Not applicable.

Are examples needed to explain this concept?

5. <LabSampleID> (Required for Traceability): A lab defined identifier for a sample that uniquely identifies a single sample that is subjected to an analysis.

This data element forms the primary link into the lab's recording keeping system. It does not have to have a one-on-one relationship with the <ClientSampleID>. For example, there may be just one <ClientSampleID> for a given field sample which may separate into two phases during shipment. The lab may give these two phases separate <LabSampleID>s as they may be analyzed separately but both would always have the same <ClientSampleID>.

Allowed valid value(s):
Not applicable.

6. <MatrixID> (Required): A client defined code for the sample matrix or media (e.g., soil, water).

This data element defines the fundamental matrix of the sample. If additional information is needed to further define the specific matrix of the sample, then the <MatrixName> data element could also be used. (An alternative approach could also be used whereby the specific matrix type would be appended to generic matrix type as follows: Matrix_Type)

Allowed valid value(s):
See comprehensive list at the end of this document. Examples: 'Water', 'Soil', 'Sediment', 'Air', 'Filter', 'Wipe', 'Ash', etc.

7. <MethodBatch>: A lab defined identifier for a batch of samples analyzed by one method and treated as a group for QC purposes. A method batch should group samples with similar matrices and potential interferences.

This data element allows the data generators to associate a group of samples together for specific QC reasons at the sample level. This particular association is often used for matrix spikes or duplicates when the data requestor may require that one sample in twenty be spiked or duplicated for QC reasons.

Note: This association is being made at the sample level, NOT at the preparation/digestion or analysis level.

IMPORTANT: The data requestor must specify specific instructions as to how this data element will or will not be used. Data recipients must be fully aware of how this particular data element is being used.

Allowed valid value(s):
Not applicable.

8. <OriginalClientSampleID> (Conditionally Required): The client sample ID number of the original regular sample from which the QC sample was derived.

This data element allows a QC sample which might be derived in the lab to be linked to the original field sample that was used to derive it. As an example, the lab may select a given client's sample to perform a matrix spike. The <OriginalClientSampleID> would allow this lab derived QC sample to be linked back to the client's original sample.

Allowed valid value(s):
Not applicable.

9. <QCCategory> (Required for Portability of QC): A generic code that specifies the basic properties or category of a method QC sample.

This data element provides a general classification of QC sample types.

Allowed valid value(s):
'Blank', 'Blank_Spike', 'Duplicate', 'Blank_Spike_Duplicate', 'Serial_Dilution', 'Spike', 'Spike_Duplicate', 'Non-Client_Sample', 'PT_Sample'.

10. <QCLinkage> (Required for Portability of QC): For a QC sample, specifies which batch is the basis for the association between the QC sample and the regular samples.

This data element

Allowed valid value(s):
'SamplingBatch', 'EquipmentBatch', 'ShippingBatch', LabReportingBatch', 'MethodBatch', 'HandlingBatch', 'PreparationBatch', 'CleanupBatch', 'StorageBatch', 'AnalysisBatch', 'RunBatch'

11. <QCType> (Conditionally Required): The client defined code for the specific type of QC sample used.

This data element

Allowed valid value(s):
See comprehensive list at the end of this document. Examples; 'Method_Blank'