SAND2016-0089R

IDC RE-ENGINEERING REPORT

SAND20XX-XXXX
Unlimited Release
January 2016

IDC Re-Engineering Phase 2 Iteration E1 Data Model

Version 1.0

J. Mark Harris, Chris Young, Shack Burns, Ben Hamlet, Mark Montoya, Rudy Sandoval, James Vickers

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Approved for public release; further dissemination unlimited.

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

SAND20XX-XXXX
Unlimited Release
January 2016

IDC Re-Engineering Phase 2 Iteration E1 Data Model

J. Mark Harris, Chris Young, Shack Burns, Ben Hamlet, Mark Montoya, Rudy Sandoval, James Vickers
Next Generation Monitoring Systems
Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-MS0401

Abstract

This document contains the Data Model generated from the model contained in Rational Software Architect.

REVISIONS

Version	Date	Author/Team	Revision Description	Authorized by
1.0	12/17/2015	SNL IDC Re-Engineering Team	Release for E1	M. Harris

TABLE OF CONTENTS

1 Design Description

The System is fundamentally concerned with acquiring and processing waveform data for the purpose of detecting events of monitoring interest. Thus, to properly design the system it is necessary to model the types of data used by the system and the relationships between those types of data, i.e. to formulate a data model. Data modeling can represent a high, medium, or low level of detail. This document focuses on the high to medium level of detail, establishing the most significant classes and their inter-relationships.

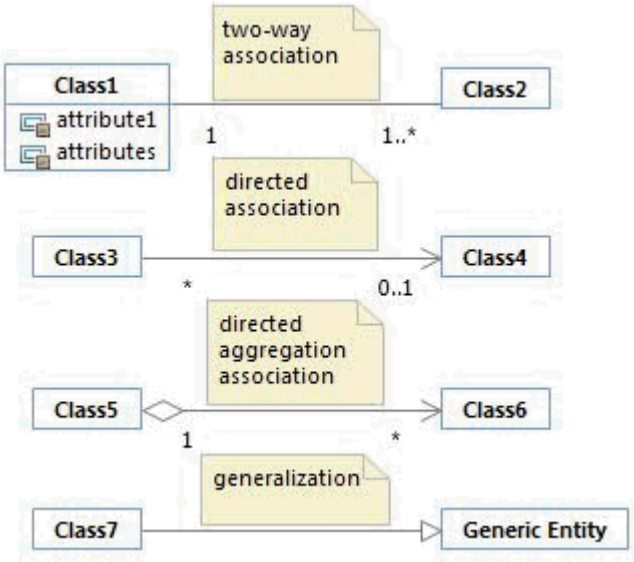
The data model presented here should not be confused with a data format such as CSS3.0 or QuakeML. Regardless of the way the information is represented within the System, i.e. the data model, the system will provide a capability to export data in a variety of standard formats.

Because the overall data model is large and complex, it is useful to break it into several groups of classes related to the flow of data through the system:

1. Stations - SHI sensors are deployed at a set of stations organized into networks.
2. Waveforms - Continuous waveform data are recorded at those stations and forwarded to a data center for processing and analysis.
3. Signal Processing Operations - Waveform data are passed through a series of transforms to enhance signals of interest.
4. Signal Detections - Waveform data from each station are processed to identify signals of interest.
5. Events - Signal detections from multiple stations are combined to build events.
6. Location Solutions - For each event, one or more location solutions are calculated.
7. Magnitude Solutions - For each location solution, one or more magnitude solutions are calculated.

2 Class Diagrams

2.1 Classes - Example



This report contains numerous diagrams showing different parts of the data model. In this section we explain the notation used for these diagrams, providing the diagram shown above as an example.

Each of the boxes (e.g. the box named 'Class1') represents a class. Classes will have zero or more attributes, methods, and associations to other classes.

The small rectangles within each class are attributes. These represent the fields of a given class. These fields can represent: primitive values, collections, and other class objects. In the instance of Class1, its "attribute1" field is a primitive, while its "attributes" field is some collection of primitives. The lists of attributes presented in this document are not intended to be complete, but represent the attributes that best characterize the class.

Relationships between classes are shown as lines and can be modeled in different ways. The line from Class1 to Class2 is a two-way association. This means that a Class1 object and Class2 object both have references to each other. The multiplicity is explained by the numbers on each end. The '1..*' on the right means that Class1 has one or more references to Class2, and the "1" on the left means the Class2 has exactly one reference to Class1.

The arrow from Class3 to Class4 is a directed association. This means that Class3 has a reference to Class4, but that Class4 has no reference back to Class3. The multiplicity rules work the same way as described above: Class3 has a reference to zero or one Class4s, but a Class4 can be referenced by zero or more Class3s.

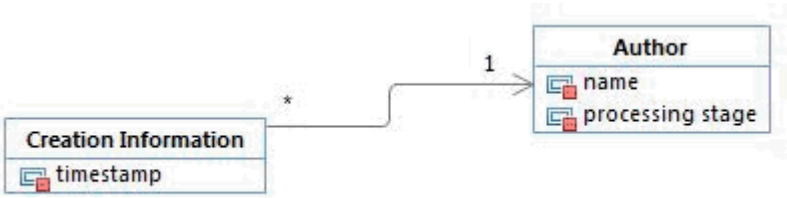
The arrow with a diamond at the tail from Class5 to Class6 is a directed aggregation association. While the previous associations were only references to other classes, a directed aggregation

association defines ownership over the lifecycle of the associated classes. Here, a Class5 aggregates Class6s. This means Class5 controls the lifecycle of all Class6s it aggregates, and that no Class6 can exist without its parent Class5. The multiplicity rules are the same as explained above. Class5 aggregates zero or more Class6s, but a Class6 can be aggregated by exactly one Class5.

IMPORTANT: A directed aggregation association assumes that every child class has a way of accessing its parent class. In this diagram, this means that Class6 would have some 'getParent' method that would return back a reference to the Class5 that aggregates it.

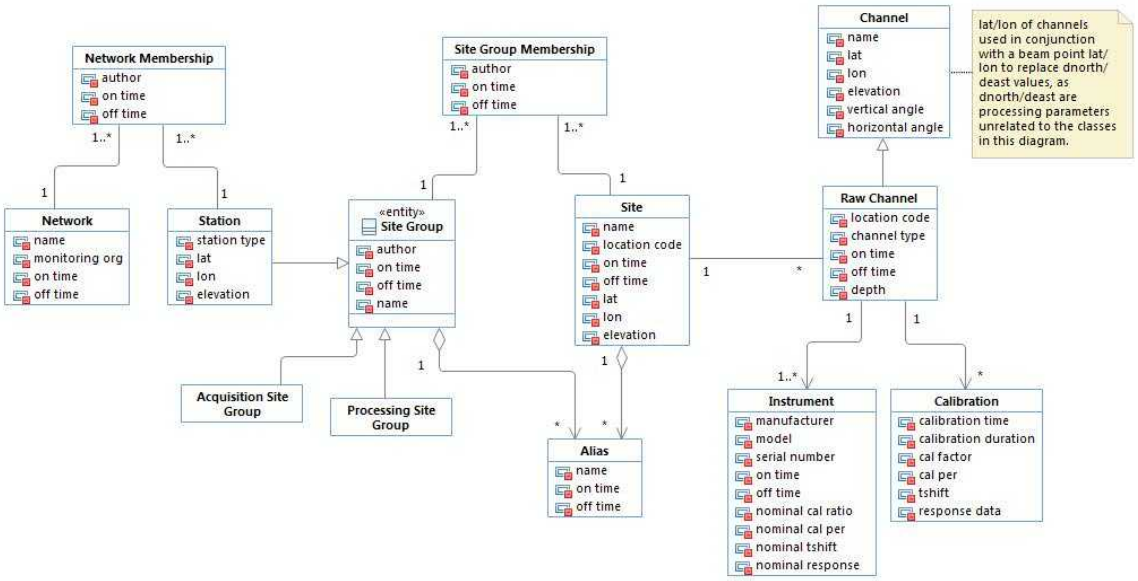
The arrow with a triangular arrowhead from Class7 to Class8 is a generalization. This means that Class8 inherits from Class7, i.e. Class7 is the base class.

2.2 Classes - Creation Information



When objects are created and changed within the System, the history of those objects are just as important to capture as the values they currently contain. The Creation Information class is one step towards this goal of object provenance. It contains a creation timestamp, representing the date and time that this object was created. It also contains a reference to the Author and the Algorithm that created it. The Author stores a name, such as the username for the analyst who created it, or a name indicating the System's automatic processing created it. It also contains the version of the System and the Operating System when the object was created. The Algorithm stores a name, version, and any runtime parameters used in the creation of this object. All of this information is important, as changes in the System between the time of the object's creation and now could cause the creation of objects with different values, differences important to minimize when attempting to investigate or recreate an object.

2.3 Classes - Station



A Network is a collection of affiliated Stations. The relationship is often that these Stations are operated by a particular monitoring agency, but a Network can be arbitrarily defined as well (e.g. a set of stations that a researcher is using for a project). A Network has a name (e.g. "GDSN"), a start and end time, and a collection of Stations it is composed of. The same Station can belong to more than one Network.

A Site Group is a named set of one or more related Sites. Concrete examples of Site Groups include Station, Acquisition Site Group, and Processing Site Group. In a Station, Sites may be simply related by proximity or purposefully arranged to operate as a sensor array. Sites may be grouped by more than one Site Group, which is the reason for the Site Group Membership class. The Site Group Membership class records the time interval for which a Site was a part of a Site Group. A Site Group has a name (e.g. for a Station: "MKAR"), the time it was activated, and the time it was de-activated (a null value if it is currently operating). These activation/deactivation times should span the time intervals from the Site Group Membership class for all of the grouped Sites. A Station has a position (latitude, longitude, elevation). This position may be used for indicating the general location of Sites associated with the Station (e.g. on a map). Site Groups may be known by two or more different names (e.g. for a Station: PS1 vs. ASAR). Because of this, in addition to its primary name, a Site Group has a set of Aliases. Other than Station, Site Groups can be used for administrative purposes (Acquisition Site Group), or processing (Processing Site Group - e.g. processing the N/S components of a Hydro station separately.)

A Site is a physical installation (e.g. a building, an underground vault, or a borehole) containing a collection of Instruments that produce Raw Channels. A Site has a position (latitude, longitude and elevation), a name (e.g. "MK01"), the time it was activated, the time it was de-activated, and a collection of Raw Channels. Each Raw Channel goes with exactly one Site. As a Site may be known by two or more names, a Site has a set of Aliases.

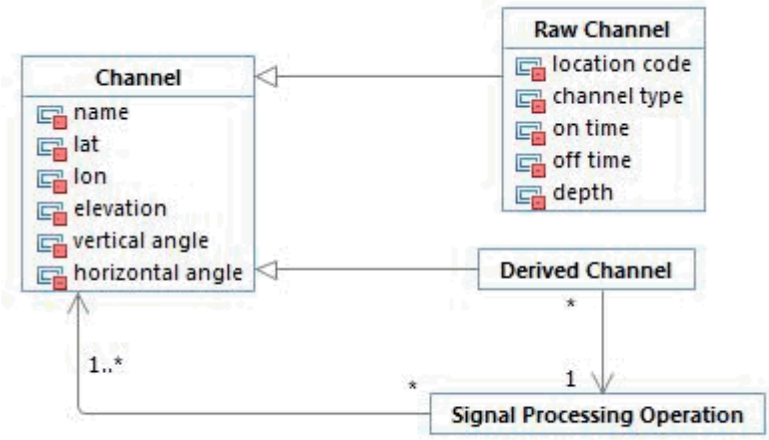
A Raw Channel represents a data source from an Instrument (sensor) that measures a particular

aspect of some physical phenomenon (e.g. ground motion or air pressure). A Raw Channel has metadata such as its on and off time, the Instrument phenomenology (i.e. Seismic, Hydroacoustic, or Infrasonic), and a channel name that encodes the type of data recorded by that channel (e.g. “BHZ” is broadband ground motion in the vertical direction). It also includes information about how the Instrument was placed and oriented at the Site: depth (relative to the elevation of the Site), horizontal angle, and vertical angle. The actual Instrument used may change (e.g. upgrade to a more current model), but the type of information that the channel records will not.

An Instrument measures the physical phenomenon that a Raw Channel represents (i.e. the instrument produces the data for that Raw Channel). The Instrument producing a Raw Channel can change over time (e.g. it can be upgraded to a better model), but at any given time there is only one Instrument for that Raw Channel. An Instrument has design and metadata information for the sensor corresponding to a given Raw Channel. It includes manufacturer and model, and other properties such as nominal instrument response.

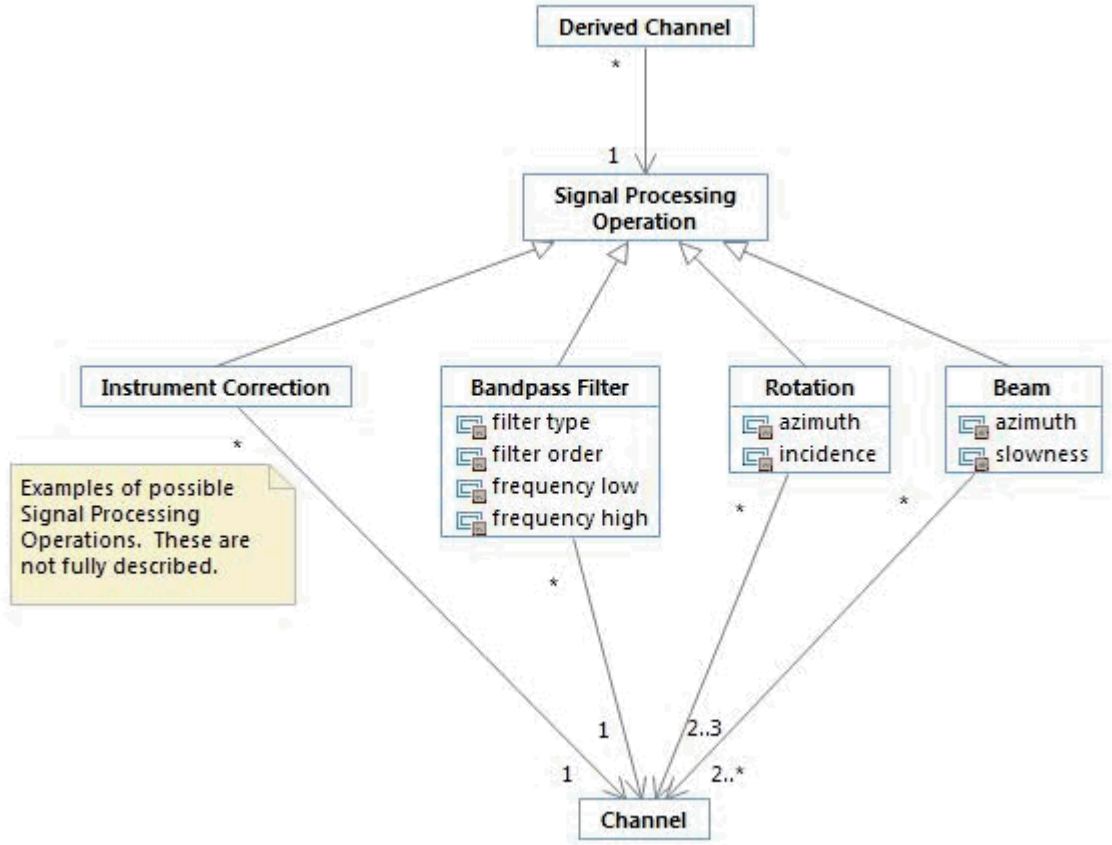
A Calibration stores the results of an Instrument calibration. This information is used to convert the output of the Instrument (e.g. volts, counts) into the phenomenon that the Instrument is measuring (e.g. seismic ground displacement). A Calibration includes information about how and when a Raw Channel's Instrument was last calibrated. A Calibration defines calibration properties (i.e. instrument response) relating to a specific time range for a Raw Channel.

2.4 Classes - Channel



A Derived Channel is a nested collection of Channels and Signal Processing Operations describing the processing history of that Channel's data, all the way back to the Raw Channel(s) (See 'Classes - Signal Processing Operation'). Every Waveform has a start time, end time, sample rate, number of samples, sample values and their respective units.

2.5 Classes - Signal Processing Operation



To enhance signals of interest, or for measuring some useful property of the signal (e.g., amplitude in a particular frequency band), Raw Channels are typically processed to produce Derived Channels. The history of this processing is stored in the Signal Processing Operation of the referenced Derived Channel.

A Signal Processing Operation is an operation of a particular type, along with any metadata about the parameters used to perform that operation, including the Channels the operation was performed on.

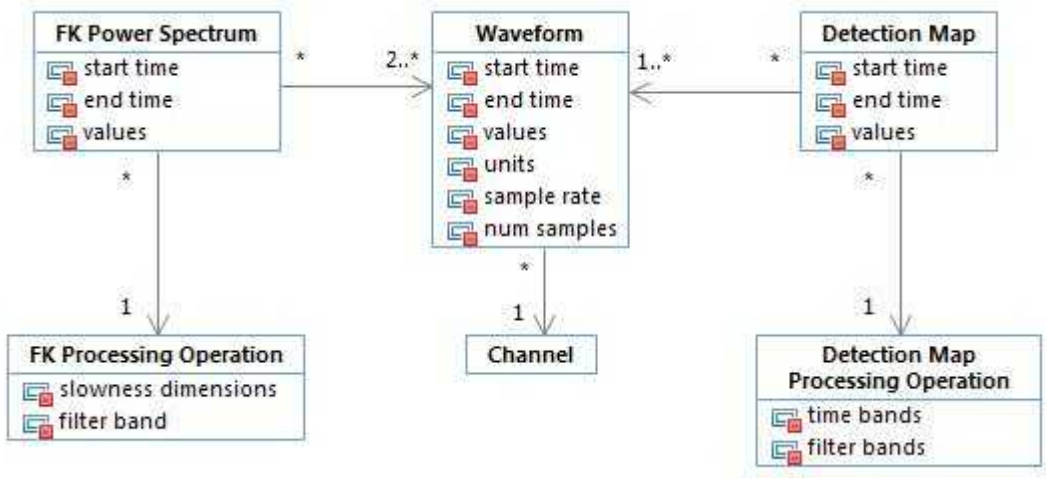
A Derived Channel can be further processed to produce another Derived Channel. An example of this is the filtering of a beam from an array station:

-The Derived Channel would reference a Filter Signal Processing Operation. This Signal Processing Operation would contain all the filtering parameters and the Channel the filtering was applied to.
-That Channel would be a Derived Channel referencing the Beam Signal Processing Operation, which would include the beaming parameters, and all the Channels used in the beaming operation.
-Those Channels would all be Raw Channels from which the data originated from.

This example shows how a user could trace back the processing history of a set of filtered and

beamed Waveforms that were all processed in the same way from the same set of Raw Channels.

2.6 Classes - Processing Results



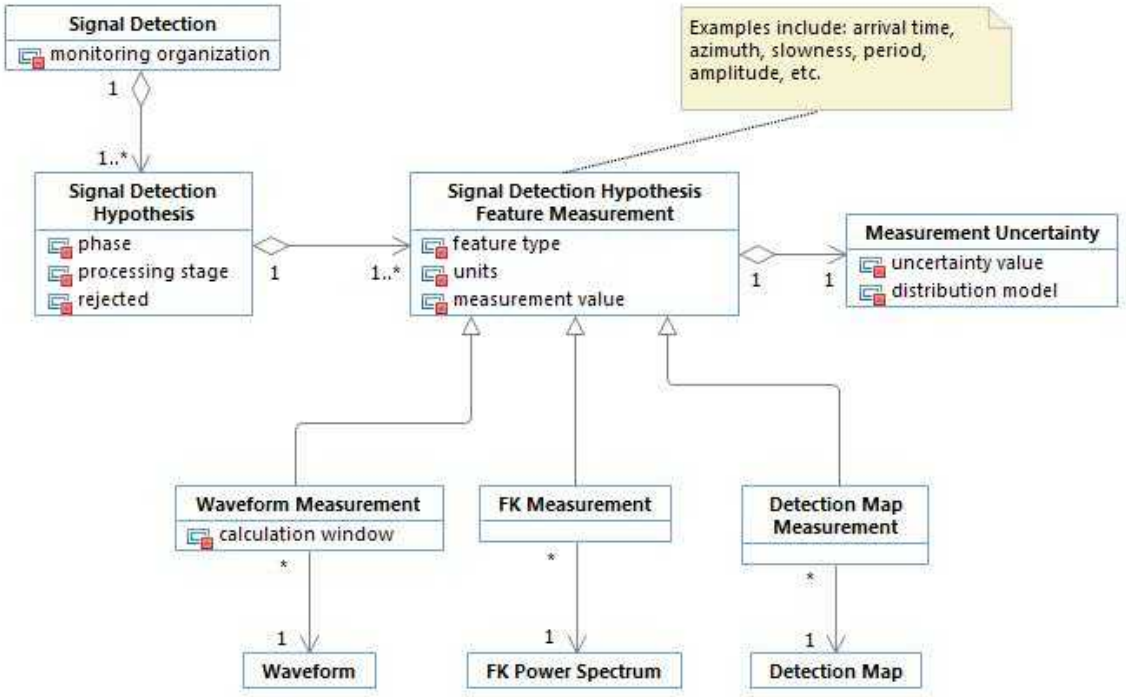
Data from Channels can produce a variety of object types. All of these object types will be calculated across a particular start time and end time, but the data values they hold can be vastly different.

A Waveform is a collection of time-sampled data from a referenced Channel that produced it, either a Raw Channel or a Derived Channel. A Waveform from a Raw Channel is a continuously sampled time series of some phenomenon, whether related to passage of a wave (e.g. ground motion, pressure), or to auxiliary information (e.g. temperature, wind speed). The raw Waveforms are typically stored in digital units (e.g. counts) that are related to the data acquisition system. Waveform data can be converted back to direct measurements of the underlying phenomenon by using the Calibration information referenced by the Raw Channel.

An FK Power Spectrum is a two-dimensional collection of power values derived from a collection of channels over a given range of slowness values and filtered over a given frequency band. It is the primary way a seismic array Station can accurately measure the azimuth and slowness of an arrival of energy.

A Detection Map is a collection of FK Power Spectrums across a particular range of divided time and frequency bands. It is the primary way infrasound stations detect arrivals and measure their respective azimuths and slownesses.

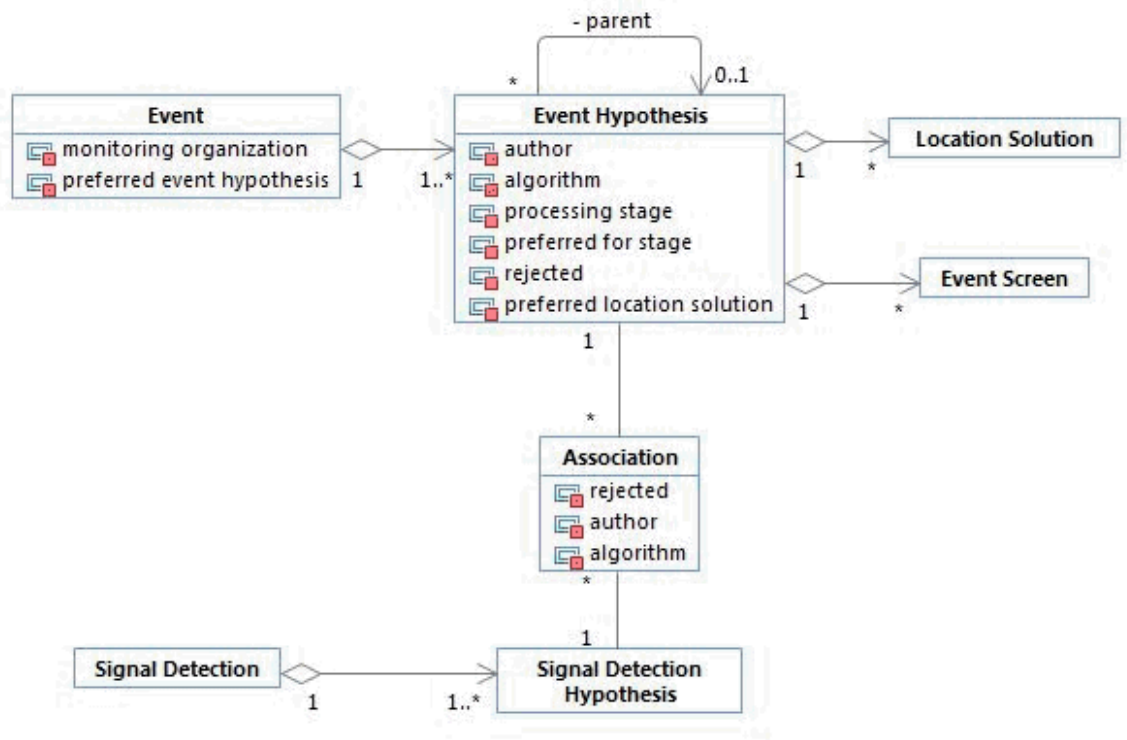
2.7 Classes - Signal Detection



A Signal Detection represents the recording of the arrival of energy at a Site. Since determining the information about a Signal Detection (e.g. arrival time) is an iterative process, we introduce the concept of a Signal Detection Hypothesis. A Signal Detection Hypothesis represents a proposed explanation for a Signal Detection. A Signal Detection can have multiple Signal Detection Hypotheses, e.g. a computer algorithm might make the original detection, while a human reviewing the results might choose to adjust the time. To keep track of the sequence of Signal Detection Hypotheses, we include its processing stage as an attribute. We also include a "rejected" attribute to keep track of Signal Detection Hypotheses that were rejected during a particular processing stage in order to prevent their recreation in subsequent processing stages. A Signal Detection Hypothesis typically will have many measurements associated with it known as Signal Detection Feature Measurements: time, dominant frequency, various types of amplitudes, etc.

A Signal Detection Feature Measurement always has a feature type, measurement value, calculation window, uncertainty, algorithm, and author. A Signal Detection Feature Measurement always references the Waveform it was calculated on, whether raw or derived. Certain Signal Detection Feature Measurements are valid for all Signal Detection Hypotheses (e.g. arrival time, signal-to-noise ratio, amplitude, period), while other Signal Detection Feature Measurements are dependent on the type of Station (e.g. rectilinearity for 3C seismic stations, f-stat for arrays of any kind). The number of potential Signal Detection Feature Measurements is large, and expected to grow, so the data model should be extensible to accommodate this.

2.8 Classes - Event



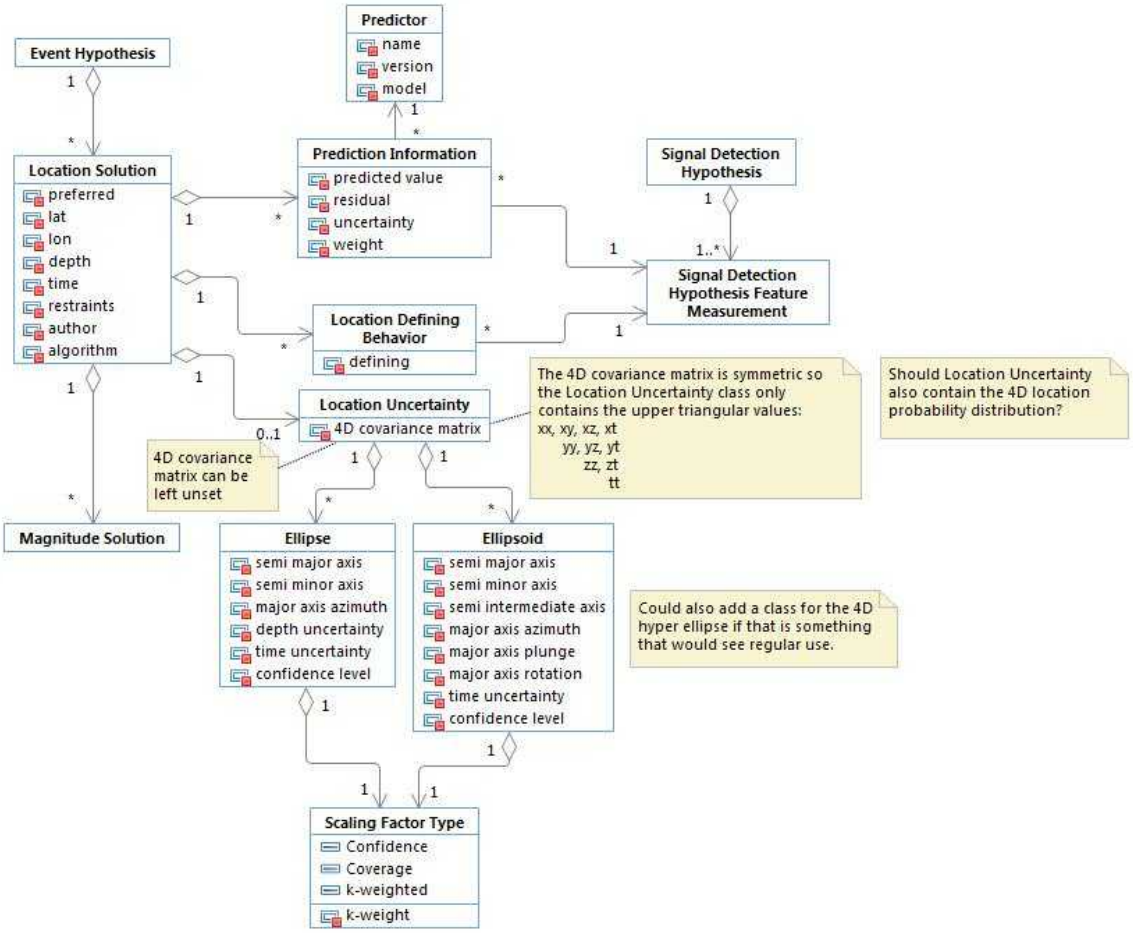
An Event marks the occurrence of some transient source of energy in the ground, oceans, or atmosphere. An Event is a single physical occurrence, e.g. the 2013 North Korean nuclear test, the 2011 Tohoku earthquake. Because the System ingests events from other systems (e.g. the ISC), an Event has a monitoring organization that created it. Note that we have not chosen to link Events from different monitoring organizations as we believe that it is important to keep system results separated from external results.

Since determining the parameters for an Event is an iterative process, we introduce the concept of an Event Hypothesis. An Event Hypothesis represents a proposed explanation for an Event such that the set of Event Hypotheses grouped by an Event represents the history of that Event (e.g. automatic computer processing might generate an initial Event Hypothesis, while subsequent refinement by an analyst might result in a different Event Hypothesis). A Processing Stage can have multiple Event Hypotheses, but only one Event Hypothesis can be designated as the preferred Event Hypothesis for each Processing Stage. The "rejected" attribute of an Event Hypothesis for a given processing stage is used to ensure that any rejected Event Hypothesis will not be rebuilt in subsequent processing stages. Only one Event Hypothesis can be designated as the overall preferred Event Hypothesis for the Event, across all Processing Stages.

An Event Hypothesis is based on a set of associated Signal Detection Hypotheses. Choosing which set of Signal Detection Hypotheses are associated with an Event Hypothesis and what phases each of those represent is what is known as “building” an Event, and can be done either automatically by an event building algorithm or manually by a human analyst. An Association

represents this association between Event Hypotheses and Signal Detection Hypotheses. The "rejected" attribute of an Association is used to ensure that any rejected Associations will not be reformed in subsequent processing stages. During analysis, a Signal Detection Hypothesis can be associated to multiple Event Hypotheses, but must be reduced to a single Event Hypothesis before the processing stage is completed.

2.9 Classes - Location Solution

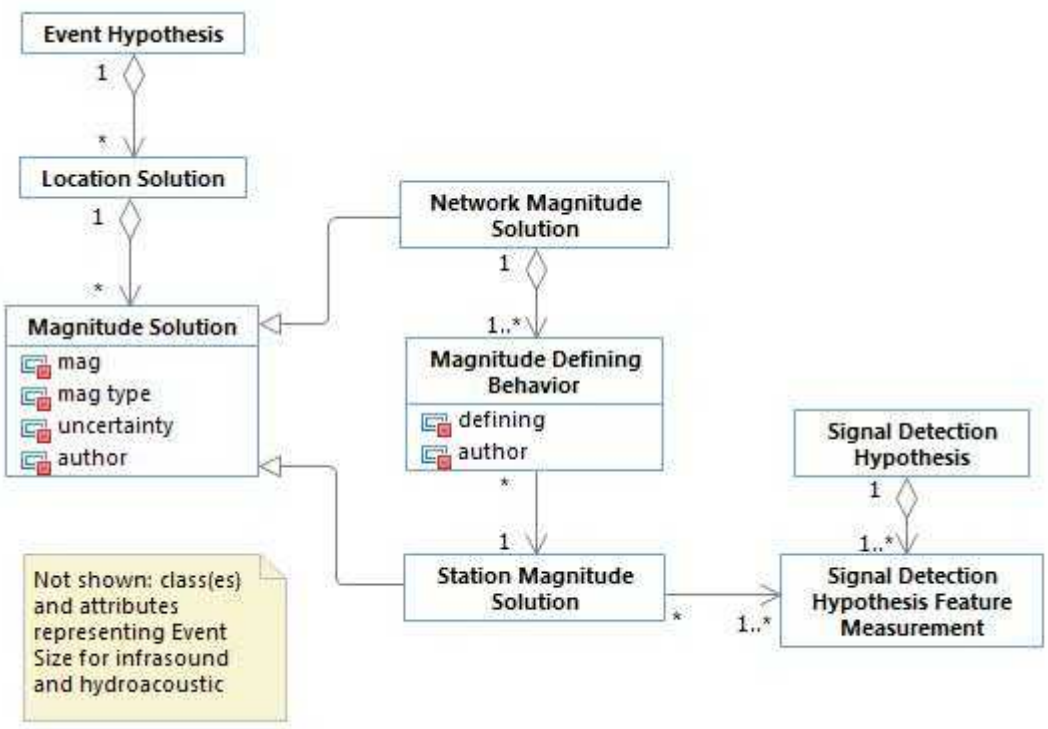


An important feature of an Event Hypothesis is the location (latitude, longitude, depth, and time), which is determined by a location algorithm that minimizes the difference between observed and modeled Signal Detection Hypothesis Feature Measurements (usually travel time, azimuth, slowness). This type of information for an Event Hypothesis is captured in a Location Solution. For the same Event Hypothesis (i.e. the same set of associated Signal Detection Hypotheses), multiple Location Solutions can be formed by running a location algorithm with different constraints such as depth (unconstrained, fixed to surface, fixed to a depth below the surface), location, or time. The rationale is that comparing how well the Signal Detection Hypothesis Feature Measurements fit with an unconstrained Location Solution versus a constrained Location Solution is a reliable method to assess how reasonable certain constraints are. Events with well-resolved depths appreciably below the surface of the Earth can be screened out as non-nuclear. Multiple Location Solutions for the same Event Hypothesis can also be created by using different locator algorithms on the same Event Hypothesis. Note that while the same set of Signal

Detection Hypothesis Feature Measurements are available for use by each Location Solution, which ones are defining can vary for a given Location Solution, and must be tracked separately.

For each Event Hypothesis with multiple Location Solutions, one Location Solution must be designated as the preferred Location Solution.

2.10 Classes - Magnitude Solutions



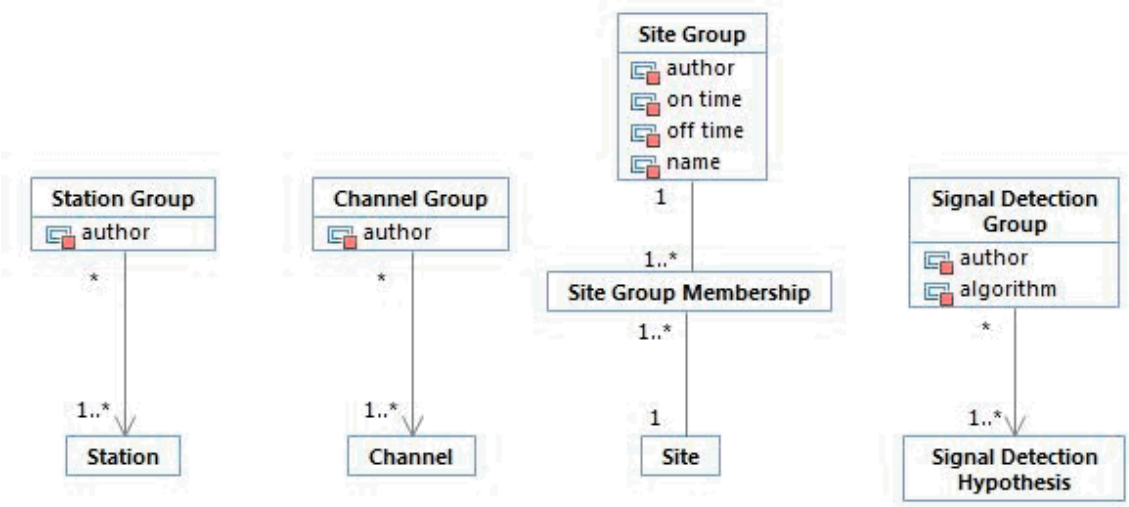
Magnitude is a measure of the size of an Event. Each Event Hypothesis can have multiple magnitude estimates, but all of these require a Location Solution to be calculated. Hence, a Location Solution aggregates all Magnitude Solutions calculated from it. This includes Station Magnitude Solutions that directly use the Location Solution in its calculation, and Network Magnitude Solutions that are calculated from those Station Magnitude Solutions.

Magnitude Solutions are similar to Location Solutions in that there can be many for the same Event Hypothesis (e.g. Mb and MS), and the Signal Detection Hypothesis Feature Measurements can be different for each of these. A subtle but important point is that magnitudes are dependent on event location, thus a Magnitude Solution must be linked to a particular Location Solution, not an Event Hypothesis.

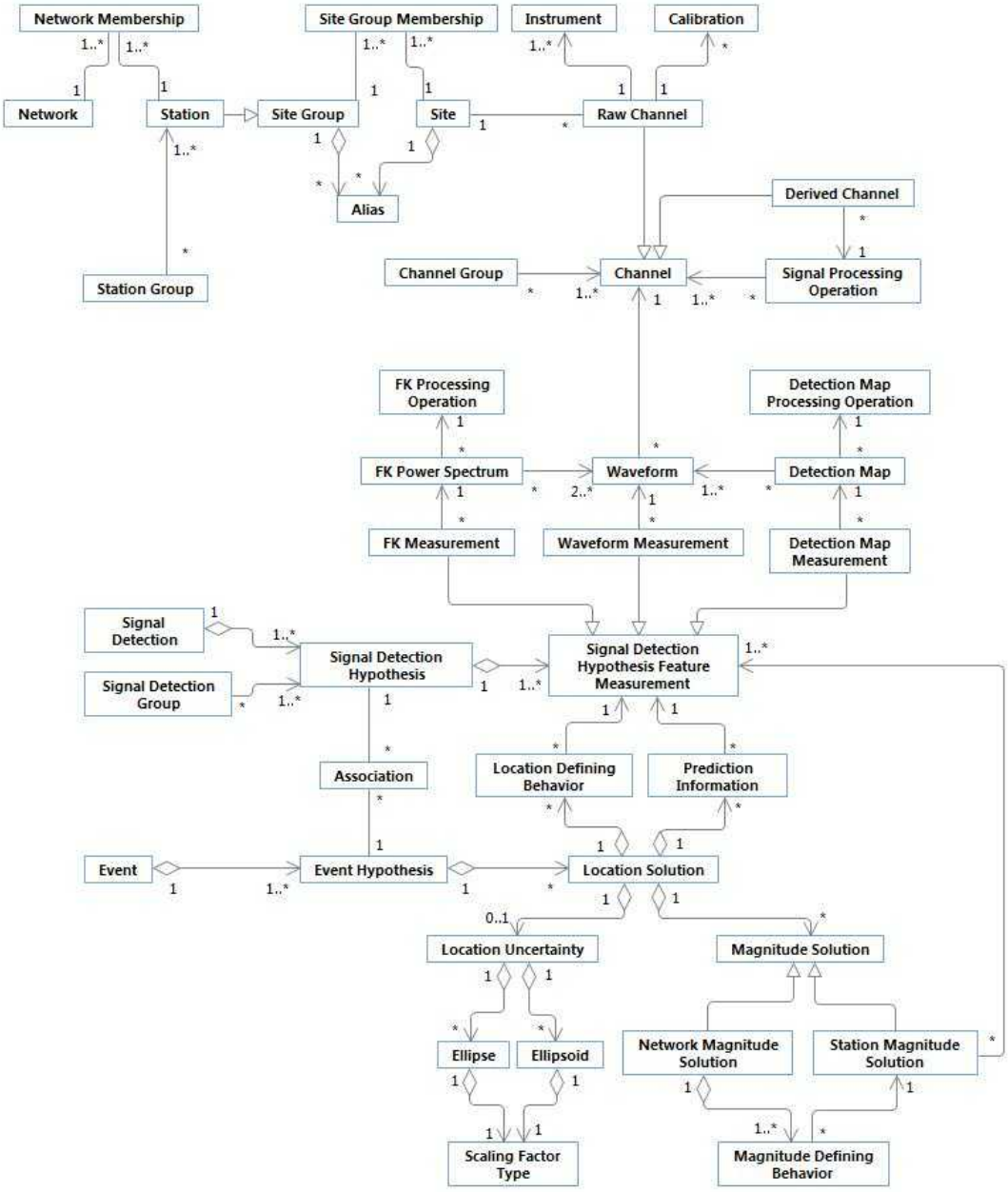
A Station Magnitude Solution is dependent on a particular Signal Detection Hypothesis Feature Measurement (typically amplitude), and a Location Solution. A Network Magnitude Solution is dependent on a collection of defining Station Magnitude Solutions, each of which must be made for the same Location Solution.

Note that once calculated, all Magnitude Solutions are aggregated by the Location Solution used to calculate them.

2.11 Classes - Class Groupings



2.12 Classes - Data Model Interconnections



The Data Model includes both definitions for the individual classes and the relationships between all these classes. This diagram is an overarching look at the relationships between all the main classes.

3 Class Descriptions

None

4 Sequence Diagrams

None

5 State Machine Diagrams

None

6 Notes

-There is a distinct difference between parameters/information stored for reference/administrative purposes, and the parameters/information used in processing. The data model needs to be able to handle both.

-Since Waveform Correlation is an algorithmic means of generating Signal Detections, Events, and Location Solutions, it is captured as provenance information.

7 Open Issues

1. Station model updates
a. (DONE: Linked Site to Alias as well. other classes are TBD) add Determine if the data model should support Aliases for classes other than Station (e.g. Network, Site, Channel) and update model if necessary.
b. (DONE) Change multiplicity on the association from Site to Raw Channel to (1 -> *) (the model currently uses (1 -> 1..*)) to allow a Site to exist even if it is not related to any Raw Channels.
c. Determine if the model should support additional station or site groupings beyond grouping stations into networks and grouping sites into stations. Examples include Administrative groups (e.g. group all IMS stations operated by a particular country, group all certified IMS stations, etc.), Transmittal/Authentication groups (e.g. group sites which transmit data to the IDC over the same communication link or multiplexed in the same packets), Processing Groups, etc. Update model if necessary.
d. Determine how to indicate a location (e.g. a Site location) was determined based on a relative measurement from another location instead of being determined directly using something like GPS. Update model if necessary.
e. Update model to support primary and backup sensors (i.e. redundant hardware installations). One option is to use a location code. If the model already supports this or it can be addressed by processing configuration then add a note describing how it is addressed in the model.

2. Signal Detection model updates
a. Update model to support generating Detection Maps and FKs. Also model relationships between the new classes and feature measurements, signal detections, etc.
b. Update the measurement uncertainty class. The current approach of using a single uncertainty value is too limiting since it assumes uncertainty follows a Gaussian distribution.

c. Determine how to model missed or expected information (e.g. non-detections, expected detections, etc.). Update model. Also consider how to model missed or expected events.
d. Determine if a separate class is needed to model the signal feature predictor and earth model used to make a signal feature prediction. The new classes would replace Prediction Information's predictor field. Is there predictor information that needs to be captured and modeled beyond the provenance information that will be captured for all types of calculated values?

3. Fundamental Concepts
a. Determine if the model needs to be updated to support different signal detection phase assignments for each of an event hypothesis' location solutions. As an example scenario, consider what would happen if a location algorithm is allowed to change phase labels – the locator may select different phase labels for a location restrained to the surface than it selects for a free depth solution.
b. Determine if the model should include a generalized class that is an extension point for event information. The model currently includes classes for event location and event magnitude, will eventually include event screening classes, and should be extensible to support future types of information.
c. Determine if any data model changes are required to support reprocessing historic data with 1) the original parameters used when the data was processed and 2) a new set of parameters. Update the data model if any changes are necessary or add a note explaining how both types of reprocessing are support by the current data model and system architecture.
d. Determine how to describe data model constraints (e.g. a hydro only feature measurement cannot be made on seismic data). The current approach is to use notes. Another approach is to use a constraint language such as OCL.
e. Ensure consistency between terms defined in the Glossary and their use in Data Model descriptions.

4. Event model updates
a. Replace the Event Hypothesis "source type" attribute with classes capturing event screening information.
b. Determine if there are any changes required to support events, event hypotheses, location solutions, etc. created by waveform correlation techniques. Update the model if necessary.
c. Determine how to model hydroacoustic event size and infrasound event size. The current magnitude classes are only appropriate for seismic events.
d. Add class diagram to elaborate on source type and screening criteria.
e. How to reference an event/template used in waveform correlation for the creation of an Event Hypothesis? Is this a generic or specific to waveform correlation?

5. Model additional concepts
a. Model classes for additional provenance information (e.g. algorithm version, input parameters, etc.)

6. Mapping to external formats
a. Develop a mapping from the Data Model classes to the International Federation of Digital Sesimograph Networks (FDSN) StationXML format (http://www.fdsn.org/xml/station/).

b. Develop a mapping from the Data Model classes to the QuakeML format (https://quake.ethz.ch/quakeml).