SAND2016-1273R

IDC Re-Engineering Phase 2

Draft Data Model v1.0
February 2016

Draft Data Model

Version 1.0

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.

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

Distribution authorized to the CTBTO for administrative or operational use..

This page intentionally left blank.

Draft Data Model v1.0
February 2016

Draft Data Model

Version 1.0

Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-MS0414

ABSTRACT

This initial draft document contains formative data model content for select areas of Re-Engineering Phase 2 IDC System. The purpose of this document is to facilitate discussion among the stakeholders. It is not intended as a definitive proposal.

This page intentionally left blank.

REVISIONS

Version	Date	Author/Team	Revision Description	Authorized by
V1.0	11/2015	IDC Re-engineering Team	Initial delivery	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 - Seismic, Hydroacoustic, and Infrasound (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.

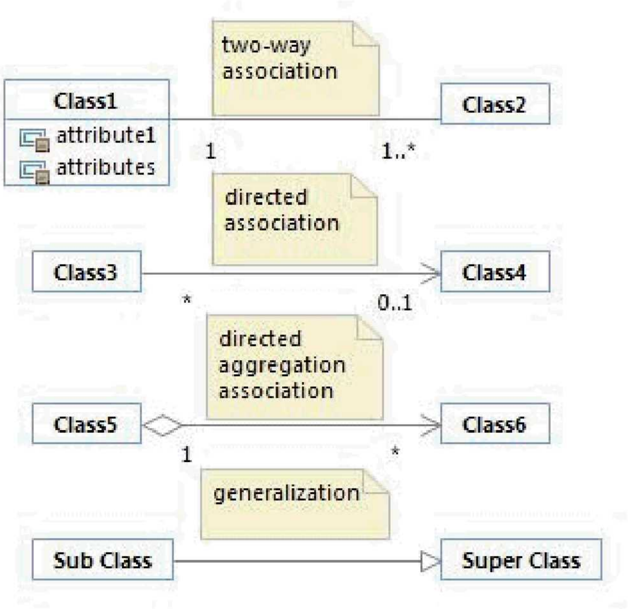
5. Events - Signal detections from multiple stations are combined to build events.

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.

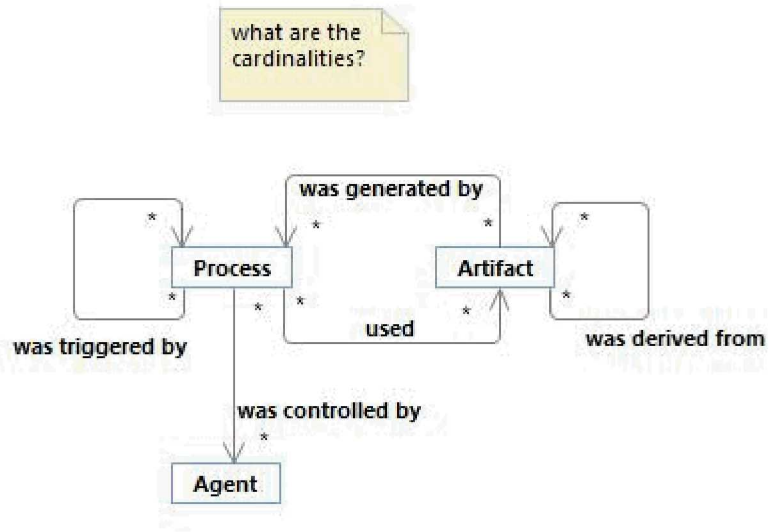
The arrow from Class3 to Class4 is a directed association. This means that Class3 has a reference to Class4, but that Class4 has no reference back to Class3. The multiplicity rules work the same way as described above: Class3 has a reference to zero or one Class4s, but a Class4 can be referenced by zero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

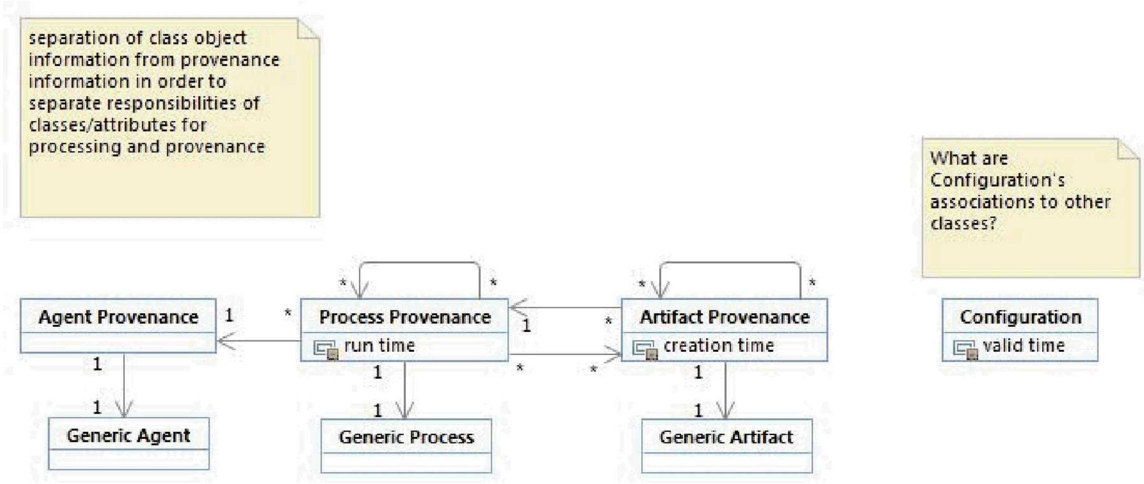
2.2 Classes - Open Provenance Model



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 Open Provenance Model (OPM) is a model for supporting provenance inter-operability with the goal to allow provenance from individual systems to be expressed, connected in a coherent fashion, and queried seamlessly. OPM is the result of a Provenance Challenge series that was initiated in May 2006, at the first International Provenance and Annotation Workshop (IPAW).

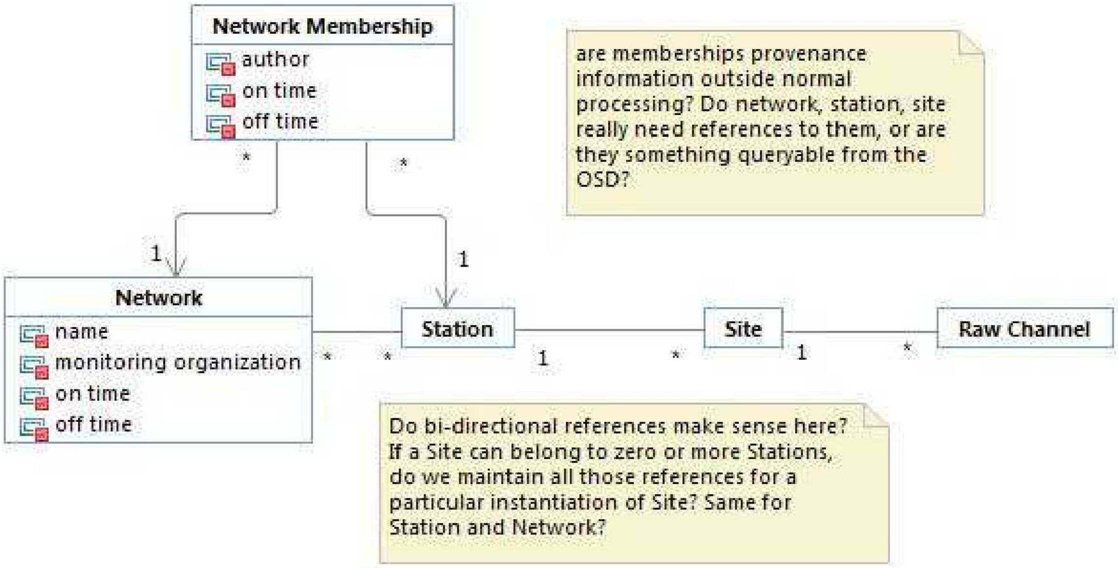
An Agent is a contextual entity acting as a catalyst of a Process, enabling, facilitating, controlling, or affecting its execution. A Process is an action or series of actions performed on or caused by Artifacts, and resulting in new Artifacts. An Artifact is an immutable piece of state, which may have a physical embodiment in a physical object, or a digital representation in a computer system.

2.3 Classes - Provenance (Proposed 1)


separation of class object information from provenance information in order to separate responsibilities of classes/attributes for processing and provenance
What are Configuration's associations to other classes?
Agent Provenance
Process Provenance
run time
Artifact Provenance
creation time
Configuration
valid time
Generic Agent
Generic Process
Generic Artifact
1
*
1
*
*
*
1
1
1
1
1
1
*
*
*
*

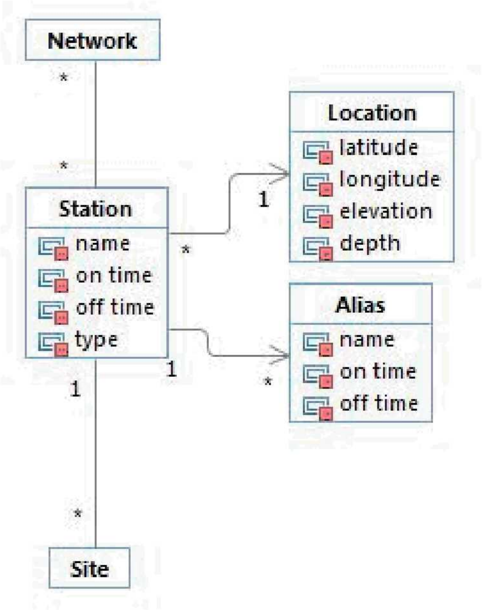
A proposal for using the Open Provenance Model is separation of provenance information from the actual data used in the System.

2.4 Classes - Network


Network Membership
author
on time
off time
are memberships provenance information outside normal processing? Do network, station, site really need references to them, or are they something queryable from the OSD?
Network
name
monitoring organization
on time
off time
Station
Site
Raw Channel
Do bi-directional references make sense here? If a Site can belong to zero or more Stations, do we maintain all those references for a particular instantiation of Site? Same for Station and Network?
*
*
1
1
*
*
1
*
1
*

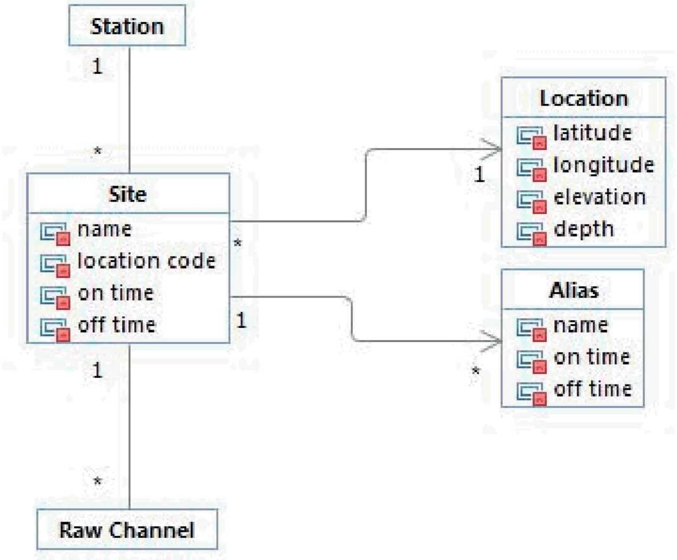
A Network is a collection of affiliated Stations. The relationship is often that these Stations are operated by a particular monitoring agency, but a Network can be arbitrarily defined as well (e.g. a set of Stations that a researcher is using for a project). A Network has a name (e.g. “GDSN”), a start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

2.5 Classes - Station



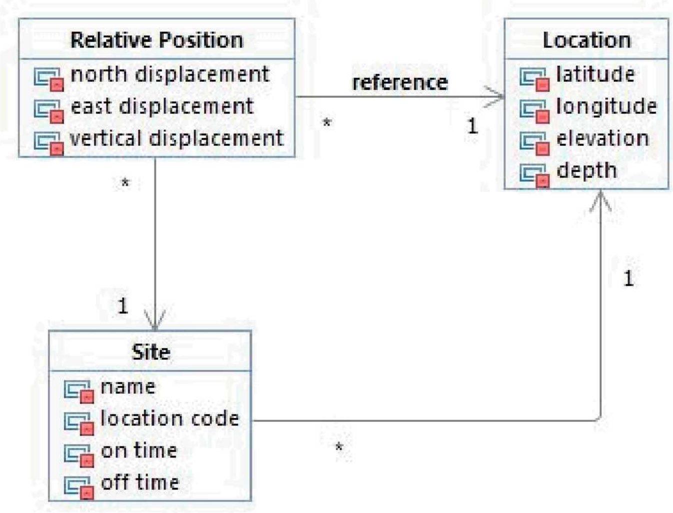
A Station has a Location (latitude, longitude, elevation, depth). This position may be used for indicating the general location of Sites associated with the Station (e.g. on a map). Stations may be known by two or more different names (e.g. for a Station: PS1 vs. ASAR). Because of this, in addition to its primary name, a Station has a set of Aliases. A Station will have a name, an on time and off type indicating the time range the Station was active, and the type of the Station (e.g. seismic three-channel, hydroacoustic array, multi-phenomenology). A Station can have zero or more Sites. Sites may be related by proximity or arranged to operate as a sensor array. A Site can belong to exactly one Station.

2.6 Classes - Site



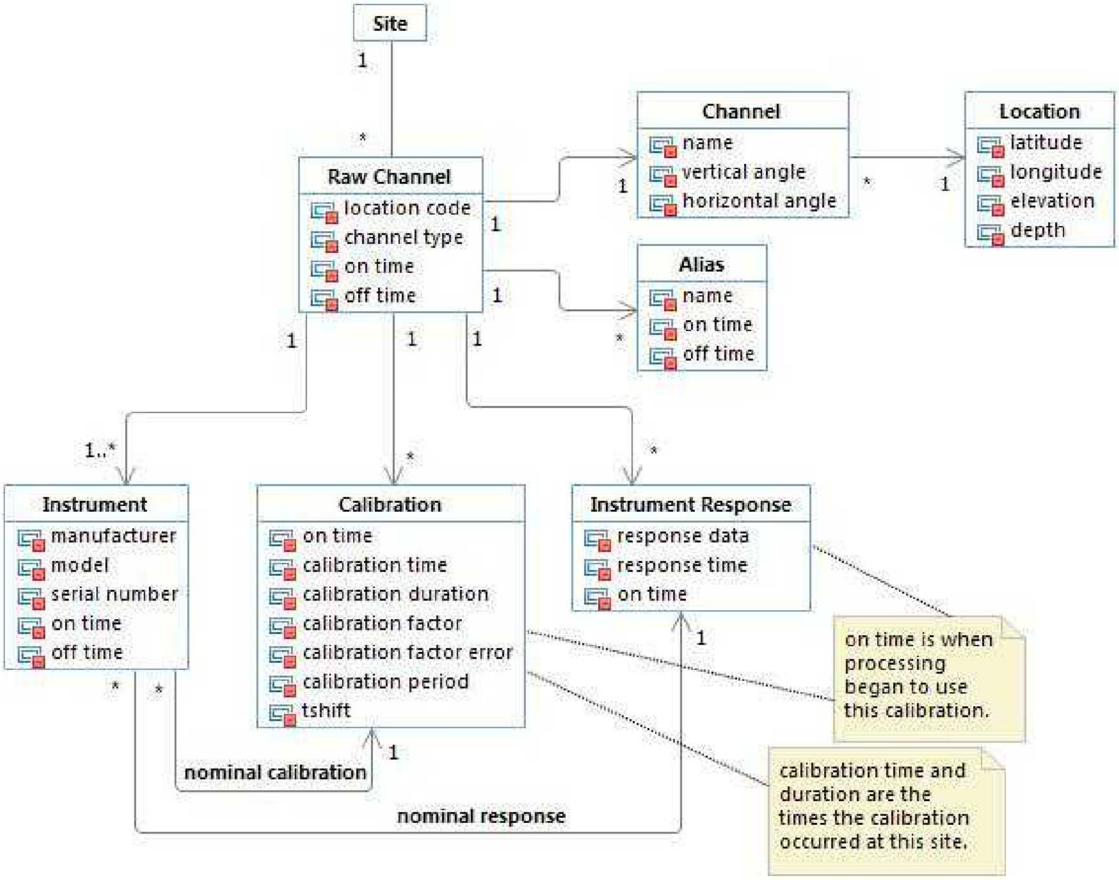
A Site is a physical installation (e.g. a building, an underground vault, or a borehole) containing a collection of Instruments that produce Raw Channels. A Site has a Location(latitude, longitude, elevation and depth), a name (e.g. "MK01"), the time it was activated, the time it was de-activated, and 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.

2.7 Classes - Site - Relative Location



Each Site may aggregate a Relative Position. The dnorth and deast attributes of a Relative Position correspond respectively to the North and East displacement of a Site's location from a particular latitude/longitude. Historically, the location of a Site may have been measured not in absolute lat/lon terms using GPS but by a surveyed reference distance to some central point, the 'refsta'. As there may be slight discrepancy in these measurements, maintaining this historic information is necessary to produce the same processing results as systems which rely on the dnorth/deast/refsta construct. Note that the Relative Position class is intended as backwards-compatible support for legacy measurements, and future processing should rely on Site displacement values which are calculated from an accurate absolute location measurement of the Site itself.

2.8 Classes - Raw Channel



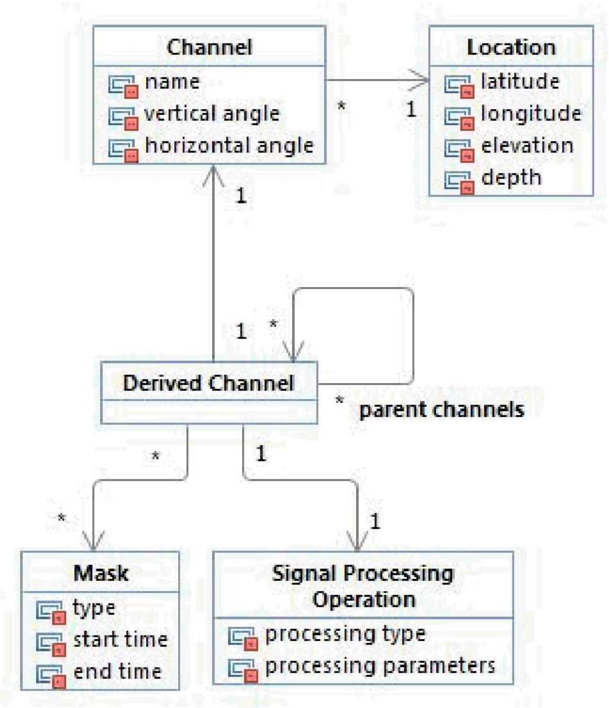
A Raw Channel represents a data source from an Instrument (sensor) that measures a particular aspect of some physical phenomenon (e.g. ground motion or air pressure). A Raw Channel has metadata, such as its on and off time, the Instrument phenomenology (i.e. Seismic, Hydroacoustic, or Infrasonic), and 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. A particular Site may have one or more redundant Raw Channels (e.g. two Raw

Channels with the same name - BHZ - each corresponding to primary and backup sensors of the same type). These Raw Channels can be differentiated by their location code (e.g. 0,1,2,...).

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.

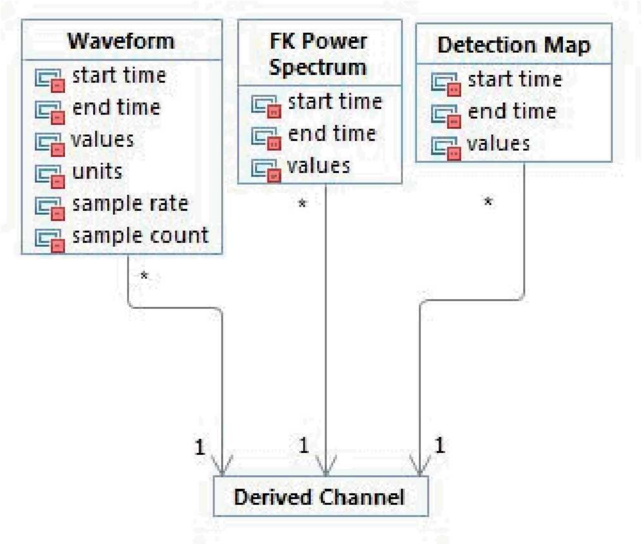
A Calibration stores the results of an Instrument calibration. This information is used to convert the output of the Instrument (e.g. volts, counts) into the phenomenon that the Instrument is measuring (e.g. seismic ground displacement). A Calibration includes information about how and when a Raw Channel's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

2.9 Classes - Derived Channel



A Derived Channel describes the processing history of that Channel's data. It contains a nested collection of Derived Channels and a Signal Processing Operation. The combination of these describes the most recent processing and the input data to that processing.

2.10 Classes - Processing Results



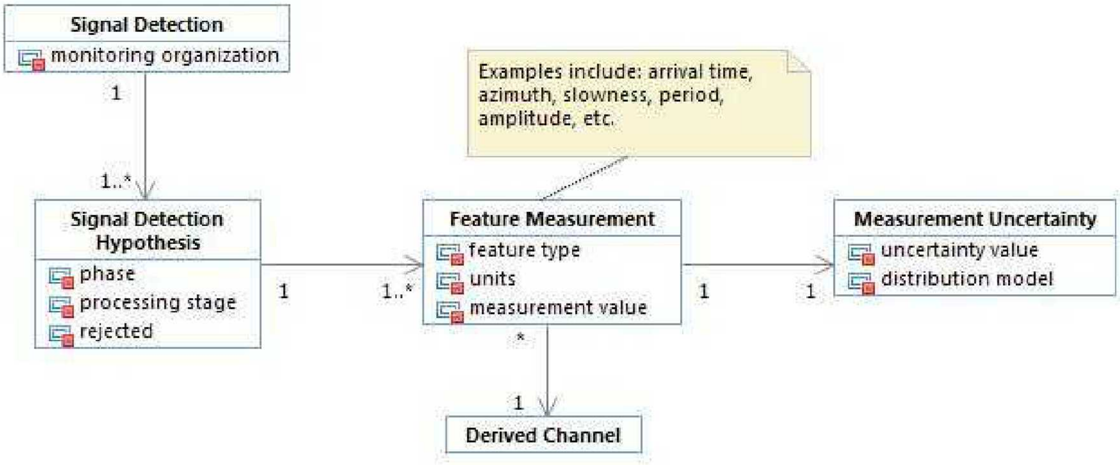
Data from Channels can produce a variety of object types. All of these object types will be calculated across a particular start time and end time, but the data values they hold can be vastly different.

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 multi-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.

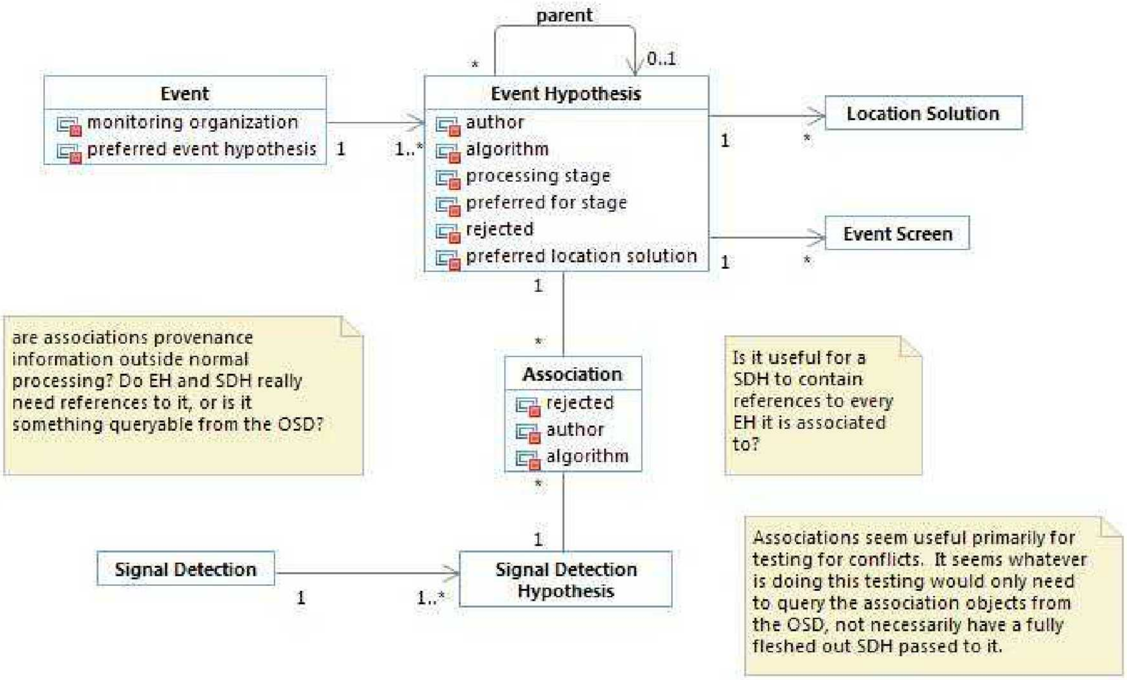
2.11 Classes - Signal Detection



A Signal Detection represents the recording of the arrival of energy at a Station. 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 (e.g. time, azimuth, slowness), 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.

2.12 Classes - Event

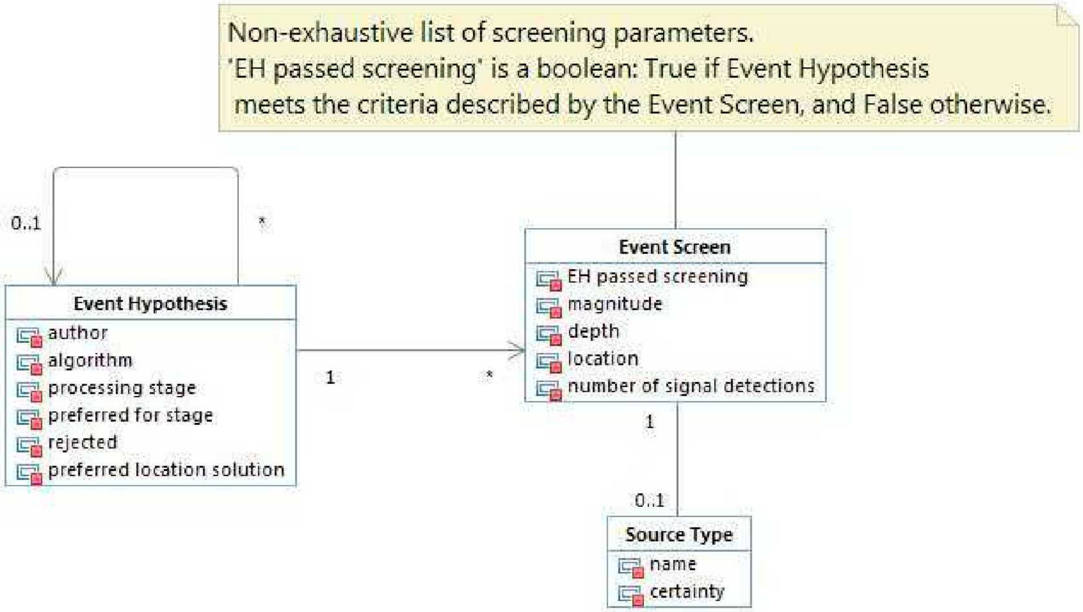

parent
*
0..1
Event
monitoring organization
preferred event hypothesis
1
1..*
Event Hypothesis
author
algorithm
processing stage
preferred for stage
rejected
preferred location solution
1
*
Location Solution
1
*
Event Screen
1
*
Association
rejected
author
algorithm
*
1
Signal Detection
1
1..*
Signal Detection Hypothesis
are associations provenance information outside normal processing? Do EH and SDH really need references to it, or is it something queryable from the OSD?
Is it useful for a SDH to contain references to every EH it is associated to?
Associations seem useful primarily for testing for conflicts. It seems whatever is doing this testing would only need to query the association objects from the OSD, not necessarily have a fully fleshed out SDH passed to it.

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.

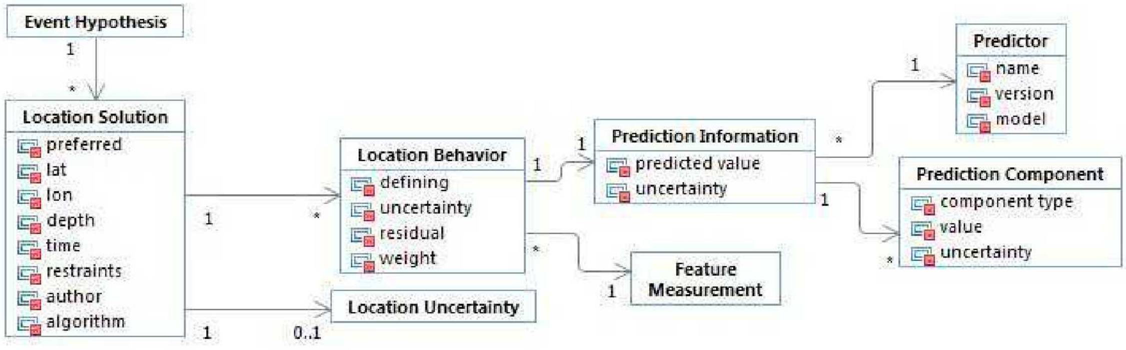
Since determining the parameters for an Event is an iterative process, we introduce the concept of an Event Hypothesis. An Event Hypothesis represents a proposed explanation for an Event such that the set of Event Hypotheses grouped by an Event represents the history of that Event (e.g. automatic computer 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.

2.13 Classes - Event Screening


Non-exhaustive list of screening parameters.
'EH passed screening' is a boolean: True if Event Hypothesis meets the criteria described by the Event Screen, and False otherwise.
0..1
*
Event Hypothesis
author
algorithm
processing stage
preferred for stage
rejected
preferred location solution
1
*
Event Screen
EH passed screening
magnitude
depth
location
number of signal detections
1
0..1
Source Type
name
certainty

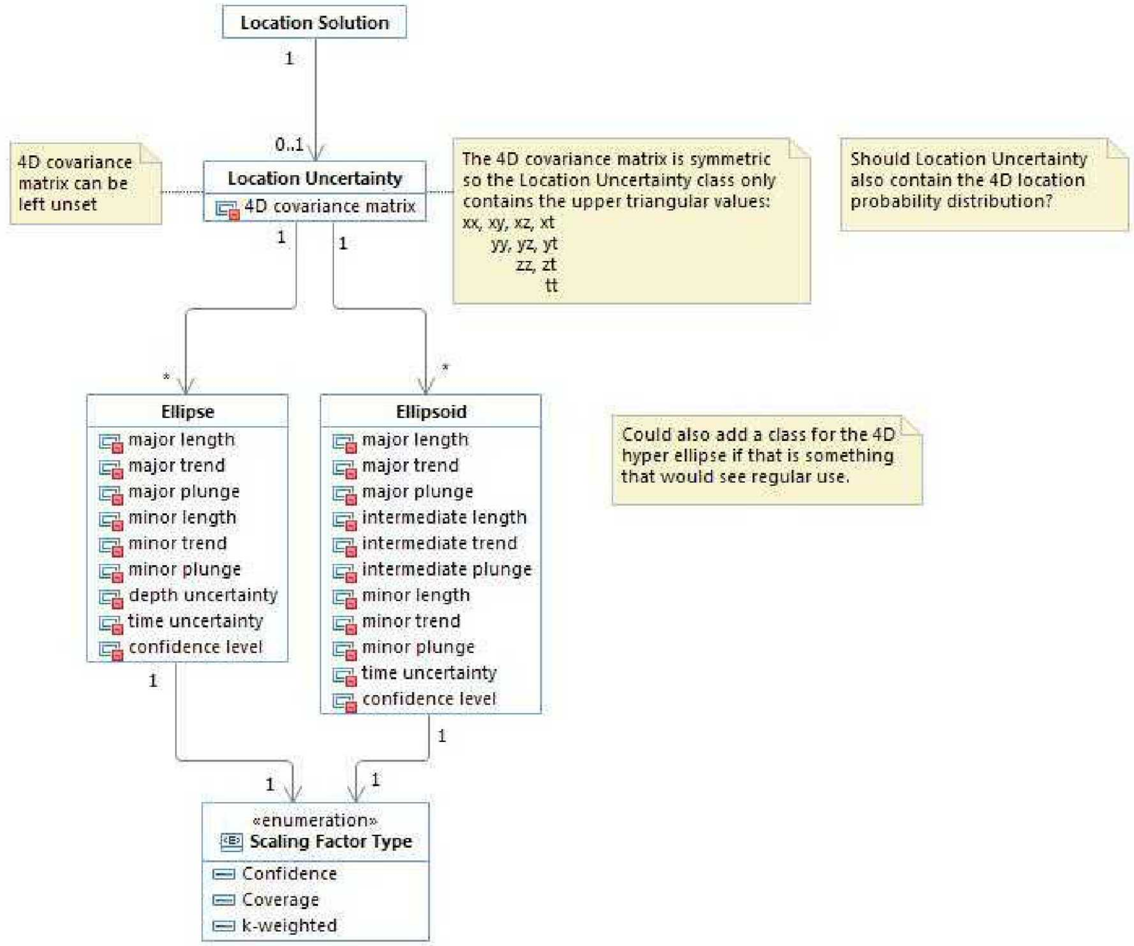
2.14 Classes - Location Solution



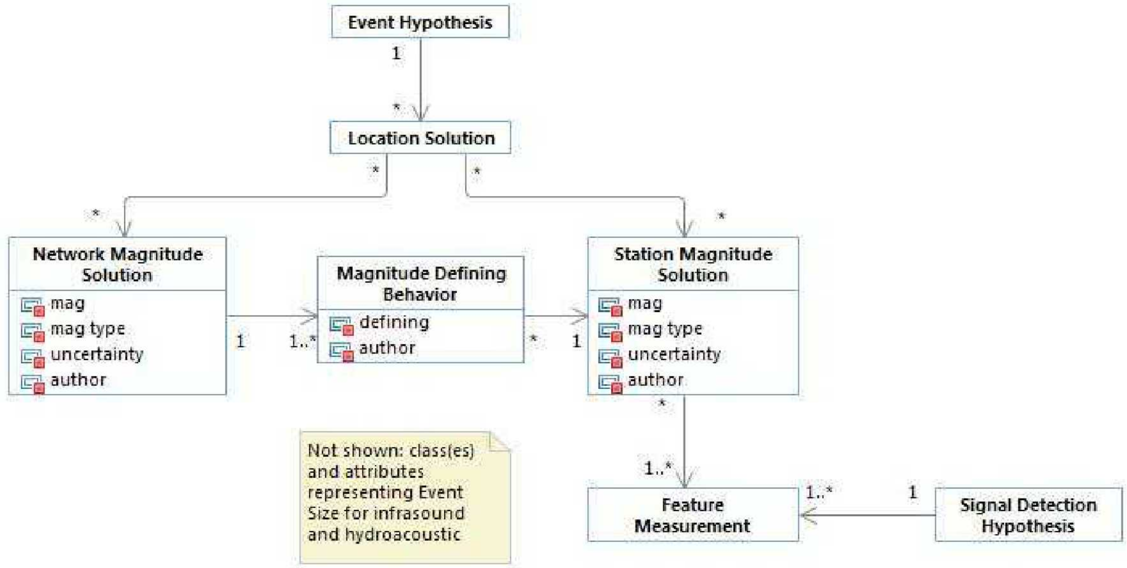
An important feature of an Event Hypothesis is the location (latitude, longitude, depth, and time), which is determined by a location algorithm that minimizes the difference between observed and modeled Signal Detection Hypothesis Feature Measurements (usually arrival time, azimuth, slowness). This type 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.

2.15 Classes - Location Uncertainty



2.16 Classes - Magnitude Solutions



Event Hypothesis
1
*
Location Solution
*
*
*
*
Network Magnitude Solution
mag
mag type
uncertainty
author
1
1..*
Magnitude Defining Behavior
defining
author
*
1
Station Magnitude Solution
mag
mag type
uncertainty
author
*
1..*
Not shown: class(es) and attributes representing Event Size for infrasound and hydroacoustic
Feature Measurement
1..*
1
Signal Detection Hypothesis

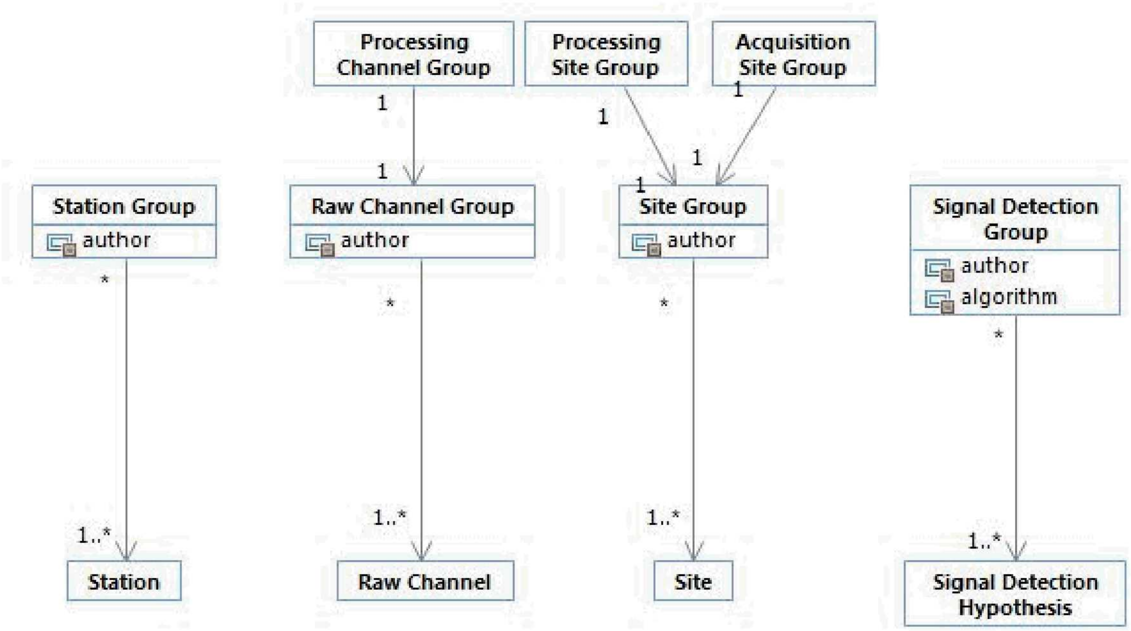
Magnitude is a measure of the size of an Event. Each Event Hypothesis can have multiple magnitude estimates, but all of these require a Location Solution to be calculated. Hence, a Location Solution aggregates all Magnitude Solutions calculated from it. This includes Station Magnitude Solutions that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

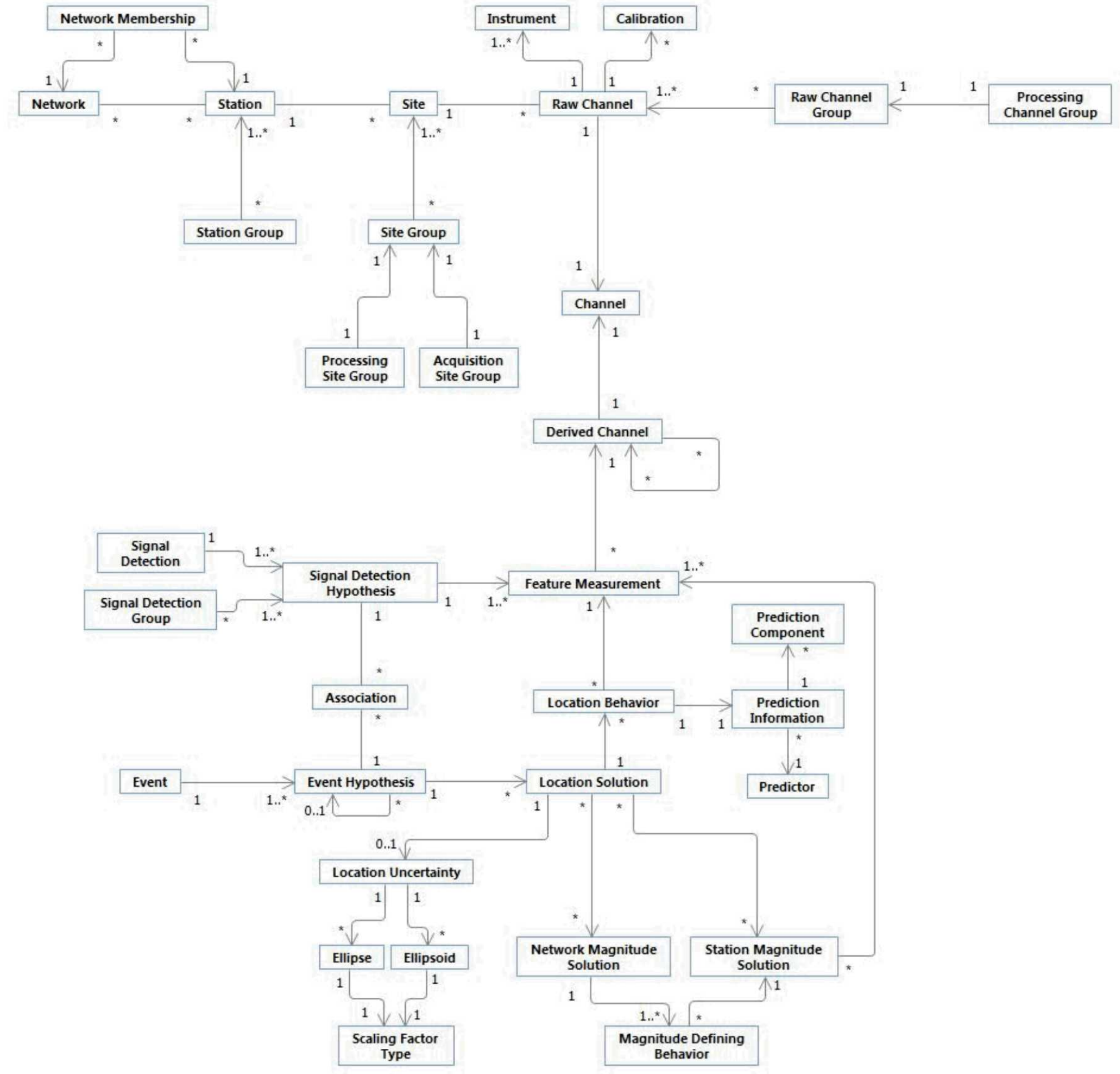
Note that once calculated, all Magnitude Solutions are aggregated by the Location Solution used to calculate them.

2.17 Classes - Class Groupings



A Site Group is a named set of one or more related Sites. Concrete examples of Site Groups include Acquisition Site Group and Processing Site Group. 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.)

2.18 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 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.

4 OPEN ISSUES

1. 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.

f. 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.

2. Event model updates

a. Determine how to model hydroacoustic event size and infrasound event size. The current magnitude classes are only appropriate for seismic events.

3. 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).

This is the last page of the document.