Context Mediation Demonstration of Counter-Terrorism Intelligence (CTI) Integration

Stuart E. Madnick

Allen Moulton

Michael D. Siegel

Working Paper CISL# 2005-03

May 2005

Composite Information Systems Laboratory (CISL)
Sloan School of Management, Room E53-320
Massachusetts Institute of Technology
Cambridge, MA 02142

Context Mediation Demonstration of
Counter-Terrorism Intelligence (CTI) Integration

Stuart Madnick (smadnick@mit.edu)
Allen Moulton (amoulton@mit.edu)
Michael Siegel (msiegel@mit.edu)

<u>Table of Contents</u>

1. Introduction – Context Mediation Problems in Counter-Terrorism Intelligence

Examination of intelligence failures prior to the 9/11/01 attacks made clear it that lack of effective information exchange among government agencies hindered the capability of identifying potential threats and preventing terrorist actions. A 2002 National Research Council study noted that "*Although there are many private and public databases that contain information potentially relevant to counterterrorism programs, they lack the necessary context definitions (i.e., metadata) and access tools to enable interoperation with other databases and the extraction of meaningful and timely information.*"[14] This report clearly recognized the importance of problems that the semantic data integration research community has been studying.

Counter-terrorism intelligence analysis requires fitting together fragments of information drawn from a variety of heterogeneous sources. In the typical case, the information originates in a written report based on field surveillance, interviews, document analysis, or other observations. These reports are then transformed into structured data using spreadsheet or database software. In the original written report and in the data structuring process, different sources may use different conventions for representing the same information. For example, a British observer may report a subject's weight in stones, while an American might use pounds and a German might use kilograms. In the case of geographic locations, a British observer may report coordinates taken off a commonly available LandRanger map, while an American observer may report coordinates for the same location using latitude and longitude readings from a hand-held GPS device and a German observer may report locations from Universal Transverse Mercator map coordinates. An intelligence analyst may need to conform all location information to a common frame of reference using the Military Grid Reference System.

Identifying significant patterns out of fragments of information is a difficult problem that can involve both art and science. When some data is represented in one way while other data is represented another way, an additional level of complexity is introduced that can be alleviated by converting data into some common form at the point of analysis. Traditional solutions to this data representation conformance problem make use of attempted imposition of data standards and purpose-built data conversion software. Context mediation offers a new means to solving the same problem with greater flexibility and reduced costs for constructing and maintaining conversion software.

Context Mediation technology deals directly with the integration of heterogeneous contexts (i.e. data meaning) in a flexible, scalable and extensible environment. The COntext INterchange (COIN) System [6] makes it easier and more transparent for receivers (e.g., applications, sensors, users) to exploit distributed sources (e.g., databases, web, information repositories, sensors). Receivers are able to specify their desired context so that there will be no uncertainty in the interpretation of the information coming from heterogeneous sources. The approach and associated tools significantly reduce the overhead involved in the integration of multiple sources and simplifies maintenance in an environment of changing source and receiver context.

An overview diagram of the COIN approach is shown in Figure 1. The COIN project provides for a systematic representation and automatic processing of data semantics. Instead requiring prior determination of semantic conflicts, the COIN approach records data semantics declaratively and uses a mediation engine to detect all conflicts, which are reconciled by automatically rewriting user queries to incorporate conversions that can be defined either internally or remotely on the network. This approach provides great extensibility and adaptability.

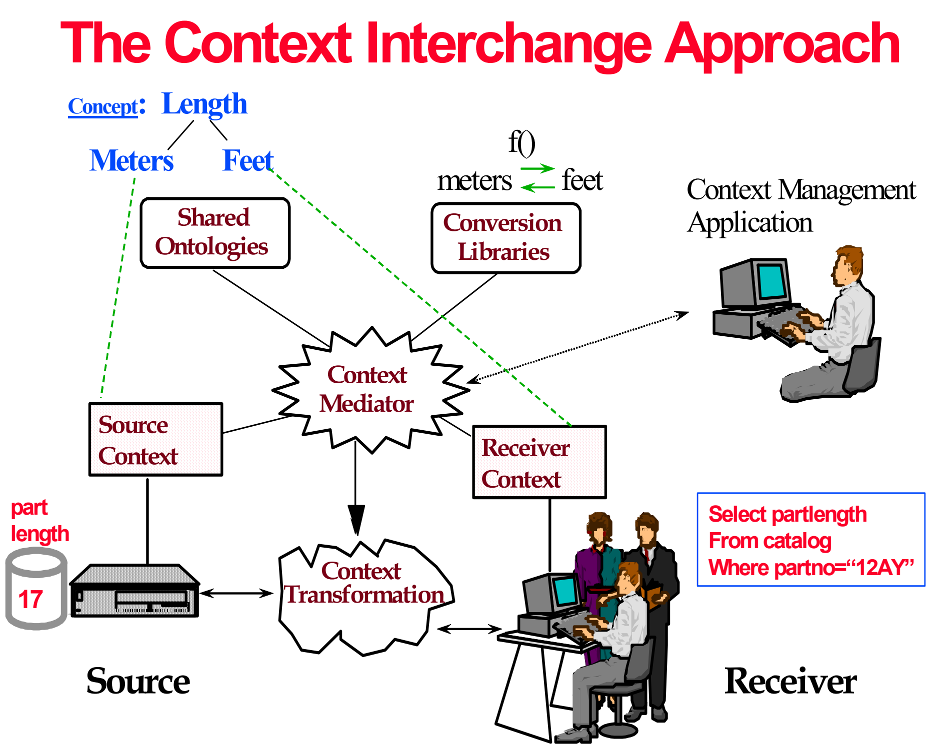

Figure 1. Context Interchange Approach

We refer readers to [2, 3] for a formal description of the COIN approach. The COIN framework is built on a deductive object-oriented data model where semantic data types and their properties are represented in an ontology. A modifier is a kind of property that determines how an instantiated semantic object is interpreted in different contexts. Data semantics are declared with 1) elevation axioms that map data elements to the semantic types in the ontology; and 2) context definitions that specify modifier values. An abductive reasoning engine is used to detect semantic conflicts and rewrite the query into one that resolves the conflicts. COIN also implements tools for authoring ontologies, interfacing with other representations (e.g., RDF, OWL)[7], specifying contexts, merging applications and domains, optimizing and executing queries.

This technology is essential in the counter-terrorism environment in a number of areas including: (1) allowing for receivers (i.e., applications, analysts) to have multiple views of the same data (e.g., different semantic assumptions depending on the needs of each application), (2) allowing for the collection of information into a single data warehouse, and (3) use in a dynamic federated environment where applications may have changing contexts and sources are added and removed from the grid. This approach is essential to the agile integration of information to support counter terrorism intelligence analysis.

2. <u>Scenario Overview</u>

The intelligence function requires integration of information obtained from a variety of autonomous sources. Much of the information originates in the form of written reports from observers in the field. Along the way, salient data from these reports is captured in structured form often using desktop spreadsheet or database software. Other data may be obtained in structured form as a result of queries to government agencies or from records of events and transactions. In each case, the individual or organization that provides the data determines how information is represented, often without any consideration of the needs of the intelligence analyst. The intelligence analyst is faced with the problem of taking data from multiple sources with different semantics and fitting the pieces together into an integrated fabric from which timely conclusions can be drawn.[13]

Context mediation provides the means to formally document the semantics of the data used by each source and to automatically generate the necessary conversion procedure to put the data into any required form.

To experiment with the use of context mediation for counter-terrorism intelligence, we developed a fictional scenario with surrogate data comparable to some types of data that might be used in actual practice. The scenario involves events, people, and places located primarily in Great Britain. The people and events are entirely fictional. The places are real, in the sense that they can all be located on a map, though there is nothing to suggest that the places chosen have any relation to action terrorist activity. Figure 2 shows a high level framework for the information used in the experimental scenario.

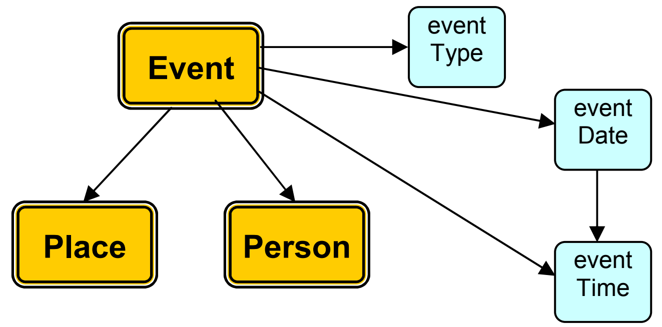

Figure 2. CTI Scenario Framework

In the initial scenario, there are five intelligence sources that report on observations of activities of some fictional individuals who may or may not be involved in a terrorist cell in the London area. Some of the individuals meet at a restaurant and a shopping mall in the Hammersmith section of London and rent a nearby apartment. Others fly into Heathrow Airport from Vienna and Munich, then take the London Underground to Hammersmith Station and are seen at a later meeting at the same restaurant. Information is incomplete and reported using different conventions by different observers. The intelligence reports from the different observers are written and then coded into a database using three different data representation conventions – USA, UK, and NATO. The analyst trying to fit the pieces together has another set of data representation conventions that are suited to matching this new data to other information. The high-level scenario framework of events, places, and persons shown in Figure 2 was elaborated into the conceptual model shown in Figure 3 with additional attributes and structure representative of the sorts of information that might be relevant for counter-terrorism intelligence analysis. The concept of an Event is associated with structural entities Place and Person and with attributes for event type, date, and time.

For this experiment, event type is represented as a text string that is not subject to conversion from one form to another by the mediator. In actual practice, one would expect that classification schemes for events might well vary from one source to another and would be a suitable object of mediation. If realistic event type schemes and conversion methods become available, this additional mediation could be readily added.

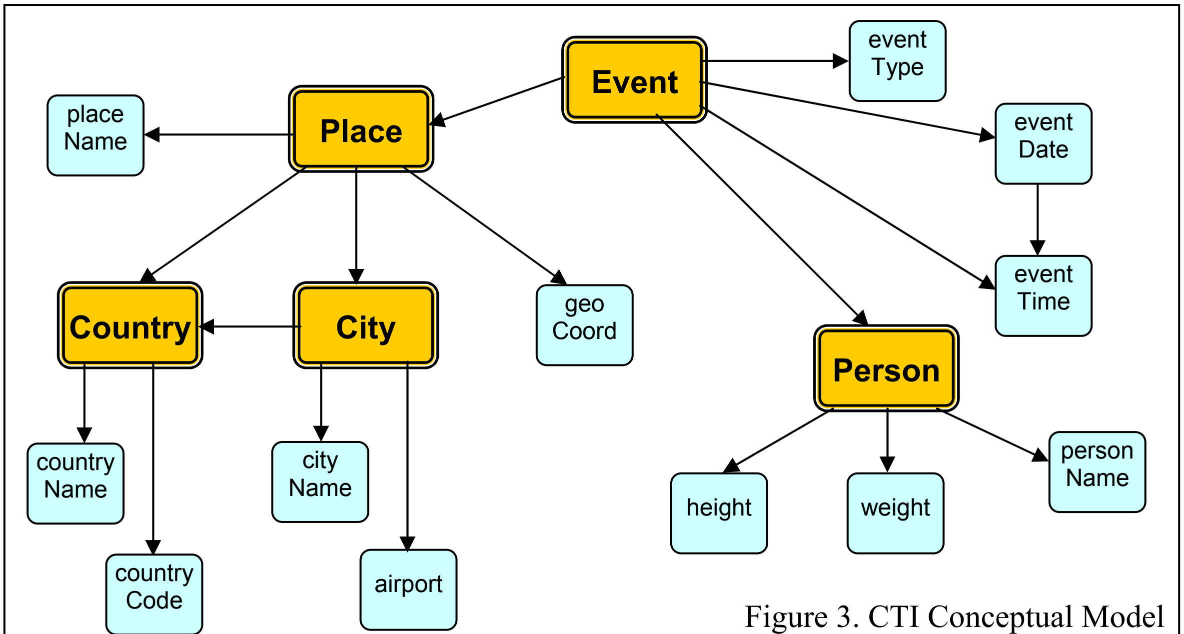

Figure 3. CTI Conceptual Model

2.1 Time

Dates and times are subject to variations in format, e.g. month in front of day in the U.S. vs. day in front of month in Europe. With the new century, additional potential confusion arises when two-digit year numbers are combined with days in the early part of a month, e.g. 03-04-05 could be March 4, 2005, April 3, 2005, May 4, 2003, or April 5, 2003. Time might be in either 12-hour or 24-hour format and both date and time may need to by adjusted for time zone, possibly different time zones.

2.2 Person

The Person entity has attributes for name, height, and weight, commonly used as descriptive characteristics for identifying an individual. Height and weight are subject to difference in units of measure. Context mediation can convert among different units of measure, allowing observers to continue to report information in units they are comfortable with, while at the same time allowing analysts to work with data converted to their individual standard

Person names can have systematic or unsystematic differences in representation. One type of systematic difference that was investigated at some length is Romanization of Arabic and other language names. Romanization is a form of transliteration that maps characters in an original language into the latin alphabet used in English and most Western languages. Different schemes for Romanization, when systematically applied, can result different representations of the same name, leading to potential difficulties in matching two reports about the same individual. While it appears that systematic application of Romanization schemes would be convertible, satisfactory methods for performing these conversions were not available within the time frame of the project. Similarly, while sophisticated conversion methods might well be able to deal with unsystematic name representation conventions, it was not possible to include any such examples in the experimental demo of context mediation.

2.3 Place

The Place entity has attributes for a simple descriptive place name and for geographic coordinates, as well as associated location entities for City and Country. Country has attributes for name and country code and City has attributes for name and a nearby airport.

Country and city names may be reported differently depending on the language used. For example, Germany in English is Deutschland in German and Allemagne in French. Austria is Österreich in German and Autriche in French. United States is Vereinigte Staaten in German and États-Unis in French. Simlarly, the city called Munich in English is called München or Muenchen in German (the latter substituting "ue" for u-umlaut). Vienna in English is Wien in German and Vienne in French.

Other semantic confusions can result from pseudonyms used for countries. For example, the United States is often referred to as America and the UK may be called Great Britain or less precisely England. Country codes are standard sets of symbols intended to eliminate confusions resulting from alternative spellings of country names. Unfortunately, as is often the case with standards, there are a number of widely used standards for country codes. The U.S. Government uses the FIPS code consisting of two letters. The ISO 3166 country code standard has three different forms: a two-letter code, a three-letter code, and a three-digit numeric code. The ISO 3166 two-letter codes and the FIPS codes look similar, but in many cases assign different codes to the same country and the same code to different country. For example, the ISO 3166 two-letter code "BG" refers to Bulgaria, but the same symbol "BG" in FIPS refers to Bangladesh. If the country code system used by each source and receiver are known, context mediation can introduce a conversion table lookup to convert the data and avoid unnecessary confusion.

Standard codes are also used for airports. In this case, there are two widely used standards – IATA, developed by the International Association of Travel Agents, and ICAO, developed by the International Civil Aviation Organization. IATA codes have three characters, while ICAO codes have four. As with country codes, conversion is possible by table lookup or by accessing a web site that performs a table lookup.

Geographic coordinates provide a standard means to more precisely define a location. There are approximately three dozen geographic coordinate reference frame (GCRF) systems in general use, each adapted to particular purposes and situations. GCRF systems include the familiar geodesic coordinates of longitude and latitude, which may be expressed in degrees or radians in a number of different numeric formats and hemispheric designations. For example, the location of the Hammersmith station on the London Underground subway would be 51:29:37N, 0:13:30W in geodesic coordinates expressed in degrees, minutes, and seconds of North latitude and West longitude. Lesser precision is expressed by dropping seconds or minutes. In addition, the final component of the measure can be extended with a decimal fraction and positive/negative values can be used to express north/south or east/west respectively.

Another commonly used GCRF system is the Universal Transverse Mercator (UTM) which defines a class of projections from the round globe of the Earth to flat maps. UTM measures latitude as northing in meters based on the assumption of a distance of 10 million meters from equator to each pole, with nothing running from south to north to 9,999,999 meters just south of the equator and then beginning again at zero for the northern hemisphere. Longitude is also expressed as easting in meters, but here to reduce distortion, the Earth's surface is divided into sixty zones, each six degrees wide from pole to pole and numbered from 180 degrees longitude (the international date line on the opposite side of the globe from the Greenwich meridian). Since the width of each zone is no larger than 667 kilometers, assigning a value of 500 kilometers to the central meridian of each zone allows the easting of any point of the globe to be designated with a positive number measured on the flat map projection. Hammersmith station UTM coordinates would be 30N 692630, 5708370, which is read easting of 692630 meters in zone 30, northing of 5708370 in the Northern hemisphere. When less precision is needed, digits are dropped from the end of each number and rounded. Hammersmith station to kilometer precision would be located at 30N 693, 5708

Another GCRF system for map coordinates is the Military Grid Reference System (MGRS), developed by NATO to systematize military maps and then adopted for civilian use as well. MGRS begins with the same sixty longitudinal slices as UTM, but then subdivides each slice twenty-four latitude zones, eight degrees high except near the poles and designated by a letter. Each zone is then further divided into 100 kilometer square MGRS grids designated by two letter digraphs. Easting and northing are then measured relative to the MGRS zone, thereby requiring no more than five digits for one-meter precision and three digits for commonly used 100 meter precision. The one-meter precision MGRS coordinates for Hammersmith station would be 30UXC9263008370, read as zone 30U, grid XC, easting 92630 meters, northing 08370 meters. To kilometer precision, the MGRS coordinates would be 30UXC9308. If zone can be inferred from context (as in military operations in a limited region), the MGRS coordinates at 100 kilometer precision would be XC926084. Note that the digits for easting and northing are run together, with the first half of the digits assigned to easting and the second half to northing.

Another GCRF system in use in the U.K. is the British National Grid System (BNG) developed by the British Ordnance Survey and used for widely available Landranger for maps of Great Britain. BNG also uses a two-letter digraph to designate a 100 kilometer square grid overlaid on the territory of Great Britain. The digraphs and grids predate and do not correspond to the similarly named and structured MGRS grids. Within a grid square, coordinates for easting and northing designate a location in the same fashion as MGRS. Hammersmith Station in BNG is TQ2332278678, read as grid square TQ, easting 23322 meters, northing 78678 meters. For lesser precision, digits can be dropped from the end of both easting and northing and the rounded. To 100 meter precision, the same location would be TQ233787.

While the geo-coordinate reference frame describes the system in which coordinates are written, to fully understand the meaning of a set of coordinates additional information may be needed. For coordinates along the surface of the Earth, parameters describing the implicit model of the shape of the Earth are needed. For map projections, additional information about the projection system and its parameters are also needed. The combination of geo-coordinate reference frame parameters, including datum and ellipsoid, will be subsumed into a single reference term `datum` in documenting data semantics.

3. Experimental Sample of Intelligence Reporting Data

For the experiments with context mediation, an Oracle table was constructed with a small sample of events, places, and people. The attributes of this CTI_Reports table are shown in Table 1 along with a brief description of each attribute's meaning and semantic variations.

Table 1. CTI_Reports – Description of Table Attributes

recID	A unique number used for convenience to identify a report record.
srcID	An identifier for the source of the report (including the original observer, the author of the written report, and the coder of the data in the database.)
countryName	Name of the country where the event takes place. May vary with language, though the sample data here use english for all country names.
countryCode	A standard code symbol for the country where the event takes place. Country codes are compact and eliminate problems with spelling and translation of country name. There are four different frequently used standards included in the example – FIPS 2-character alpha codes, ISO3166 2-character alpha codes, ISO3166 3-character alpha codes, and ISO3166 3-digit numeric codes. Other standards are also used in other circumstances, but are not included in this example.
cityName	Name of the city where the event take place. May vary with language, though the sample data here use English for all city names.
airport	Symbol to identify an airport where the event takes place, or near the city if the vent is not at an airport. Two different airport symbol systems are in common use: IATA and ICAO.
placeName	a brief description of the specific place where the event take place.
geoCoord	Geographic coordinates for the location where the event take place. Four geographic coordinate reference systems are included in the example – geodetic latitude and longitude, Military Grid Reference System (MGRS), British National Grid System (BNG) and Universal Transverse Mercator (UTM). Values for geographic coordinates also depend on the datum and ellipsoid used to represent the shape of the Earth and to provide other model parameters.
personName	Name of a person observed at the event. While there are variations in how a person's name may appear, based on ordering of parts of the name, language translation, Romanization or other transliterization system, in this example, these variations are not mediated and names are treated as invariant text.
height	Height of the person, often used as a significant identifying characteristic. May be represented in different units of measure: inches, feet, centimeter, meters.
weight	Weight of the person, often used as a significant identifying characteristic. May be represented in different units of measure: pounds, stones, kilograms.
eventType	A brief description of the type of event, e.g., meeting, observed, plane departs.
eventDate	Date of the event. All dates used are Gregorian with several different arrangements of year, month, and day and different punctuation. All dates in the example are assumed to be reported consistent with time at zero offset from GMT.
eventTime	Time of the event. All times in this example use 24-hour notation at zero offset from GMT.

Twenty rows of sample data are shown in Tables 2 and 3, with the person and event attributes in Table 2 and the place attributes in Table 3. Within the sample, data comes from five sources using three different conventions for representing information (contexts). The relationship between sources and contexts is explained in Section 5. In Table 2 height, weight, and event date are reported differently in different contexts.

Table 2. CTI_Reports – Sample Data – 5 sources in 3 contexts
Person and Event Attributes

recID	srcID	PersonName	Height	Weight	EventType	EventDate	EventTime
101	src1	Ahmet Khatib	70	170	meeting	12/01/2004	0900
102	src1	Ahmet Khatib	70	170	rents apt	12/01/2004	1200
103	src5	Ahmet Khatib	5.75	11.43	meeting	02/12/2004	1700
104	src3	Akka Mohammed	5.83	11.43	observed	03/12/2004	1430
105	src3	Al Pavlakkih	5.75	12.14	observed	04/12/2004	2100
106	src2	Aleph Faruk	173	77	plane departs	04.12.2004	1100
107	src3	Aleph Faruk	5.67	12.14	plane arrives	04/12/2004	1245
108	src3	Aleph Faruk	5.67	12.14	observed	04/12/2004	1330
109	src4	Akka Mohammed	178	73	plane departs	03.12.2004	1415
110	src4	Al Pavlakkih	175	77	plane departs	03.12.2004	1415
111	src3	Akka Mohammed	5.83	11.43	plane arrives	04/12/2004	1630
112	src3	Al Pavlakkih	5.75	12.14	plane arrives	04/12/2004	1630
113	src3	Akka Mohammed	5.83	11.43	observed	04/12/2004	1710
114	src3	Al Pavlakkih	5.75	12.14	observed	04/12/2004	1710
115	src5	Al Pavlakkih	5.75	12.14	observed	04/12/2004	1800
116	src4	Ali Abdullah	168	73	plane departs	04.12.2004	1000
117	src3	Aleph Faruk	5.67	12.14	plane arrives	04/12/2004	1245
118	src5	Ali Hakem	5.83	15.71	observed	04/12/2004	2100
119	src1	Ahmet Khatib	70	170	meeting	12/04/2004	2100
120	src1	Aleph Faruk	68	170	meeting	12/04/2004	2100

In Table 3, country code, airport symbol, and geographic coordinates vary with context.

Table 3. CTI_Reports – Sample Data – 5 sources in 3 contexts
Place Attributes with Event Type

recID	srcID	Country Name	Country Code	City Name	Airport	PlaceName	GeoCoord	EventType
101	src1	United Kingdom	UK	London	LHR	Fez Restaurant	51:29:25N, 0:13:26W	meeting
102	src1	United Kingdom	UK	London	LHR	Apt on Mall Road	51:29:25N, 0:13:50W	rents apt
103	src5	United Kingdom	GB	London	EGLL	King's Mall Shopping Ctr	TQ2309278610	meeting
104	src3	United Kingdom	GB	London	EGLL	Heathrow Airport	TQ0776675326	observed
105	src3	United Kingdom	GB	London	EGLL	Heathrow Tube Station	TQ0758175848	observed
106	src2	Austria	040	Vienna	LOWW	Schwechat Airport	33N 599043, 5341242	plane departs
107	src3	United Kingdom	GB	London	EGLL	Heathrow Airport	TQ0776675326	plane arrives
108	src3	United Kingdom	GB	London	EGLL	Heathrow Tube Station	TQ0758175848	observed
109	src4	Germany	276	Munich	EDDM	Franz Josef Strauss/ Riem/Munich Intl Airport	32N 706207, 5358946	plane departs
110	src4	Germany	276	Munich	EDDM	Franz Josef Strauss/ Riem/Munich Intl Airport	32N 706207, 5358946	plane departs
111	src3	United Kingdom	GB	London	EGLL	Heathrow Airport	TQ0776675326	plane arrives
112	src3	United Kingdom	GB	London	EGLL	Heathrow Airport	TQ0776675326	plane arrives
113	src3	United Kingdom	GB	London	EGLL	Heathrow Tube Station	TQ0758175848	observed
114	src3	United Kingdom	GB	London	EGLL	Heathrow Tube Station	TQ0758175848	observed
115	src5	United Kingdom	GB	London	EGLL	Hammersmith Tube Station	51:29:37N, 0:13:30W	observed
116	src4	Austria	040	Vienna	LOWW	Schwechat Airport	33N 599043, 5341242	plane departs
117	src3	United Kingdom	GB	London	EGLL	Heathrow Airport	TQ0776675326	plane arrives
118	src5	United Kingdom	GB	London	EGLL	Hammersmith Tube Station	TQ2332278678	observed
119	src1	United Kingdom	UK	London	LHR	Fez Restaurant	51:29:25N, 0:13:26W	meeting
120	src1	United Kingdom	UK	London	LHR	Fez Restaurant	51:29:25N, 0:13:26W	meeting

4. <u>Domain Model</u>

The next step in applying the COIN context mediation technology is to build a Domain Model that captures the semantics used across all sources and receivers. Figure 4 shows a Domain Model for the semantics of the data in the CTI scenario. We begin with the conceptual model from Figure 3 and assign a COIN *semantic type* to each attribute. Where several attributes share the same semantics, a common semantic type can be used. In this example, each attribute has different semantics and a separate semantic type. For convenience, the semantic types have been given the same names as the attributes.

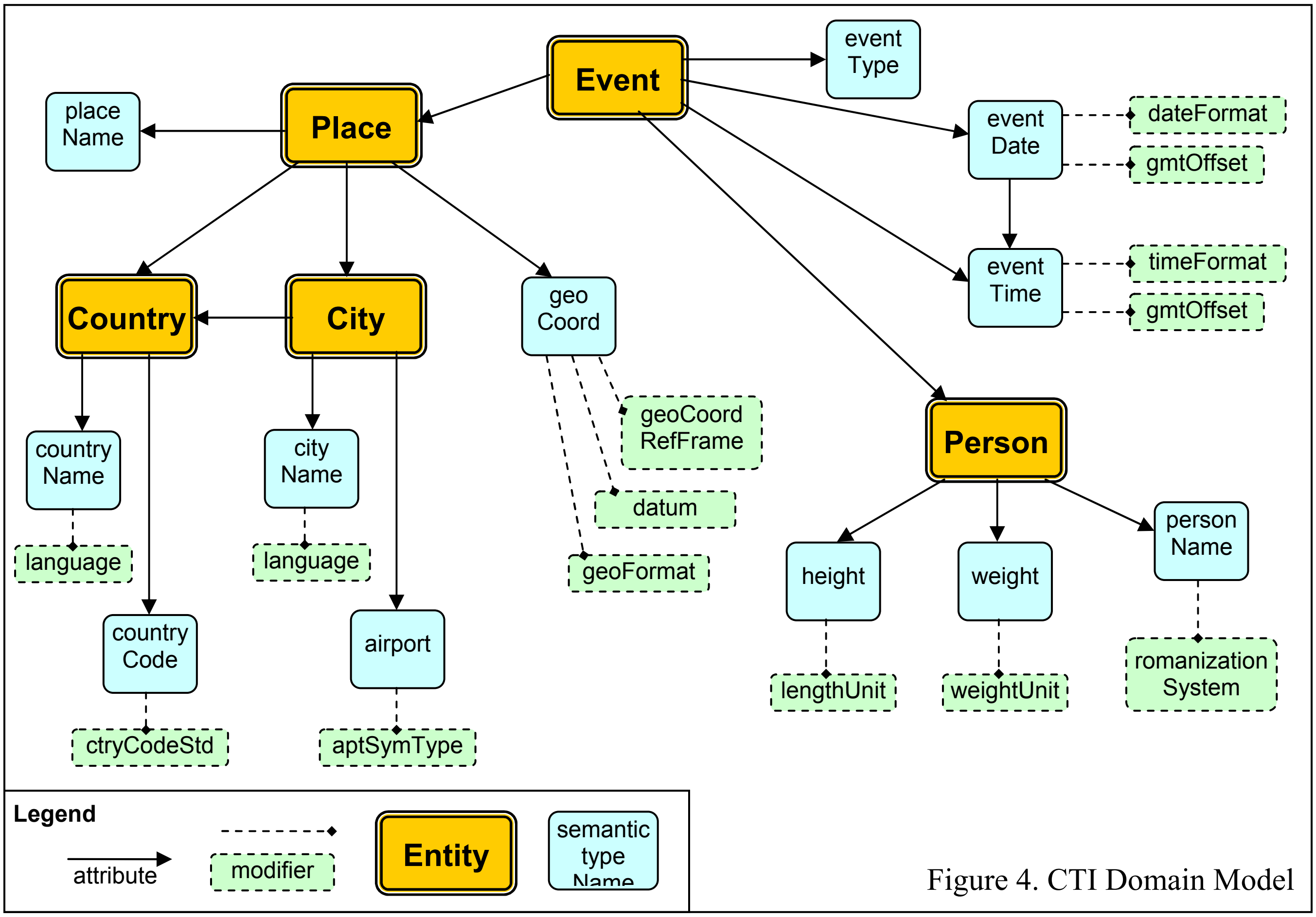

Figure 4. CTI Domain Model

One or more modifiers are associated with each semantic type to capture the differences in representation of data provided by sources and needed by receivers. For example, the semantic type height has a modifier lengthUnit that describes the units of measure used by a source or receiver. Modifiers allow the context mediator to identify semantic conflicts and find an appropriate conversion function to transform from the form coming from a source into the form needed by a receiver (see section 6 for more details).

All semantic types derive from a root semantic type called basic with no modifiers. For the eventType and placeName semantic types, no modifiers have been identified. In this example, data attributes assigned to these semantic types will be passed from source to receiver without any conversion. In the case of eventType, in particular, additional analysis or new requirements many find semantic differences for which a conversion needs to be added. When that arises, the semantic type can be changed to add a modifier and conversion rules added to allow the mediator to plan the necessary conversions.

5. Defining Sources and Contexts

The CTI_reports table described in Section 3 above includes a mixture of intelligence reports from a number of sources. The scenario assumes that each report source, identified by the srcID attribute, uses a consistent pattern of data representation. In COIN terminology, that means that all the data from a particular source has the same context. For the data in the experimental scenario, there are three source contexts – USA, UK, and NATO – used by the five sources. The context used by each source is documented in a separate called CTI_src_context.

srcID	Context
src1	USA
src2	NATO
src3	UK
src4	NATO
src5	UK

Using this source context assignment table, Oracle views were created to select all rows from the underlying CTI_reports table with sources in each of the three contexts: USA, UK, and NATO. These views then act as virtual sources with all data in each view in a single context.

An additional context, called Analyst, is not used by any source. This represents a possible receiver of information with its own semantic requirements. The values for the modifiers of each semantic type in the Domain Model for each context is shown in Table 4.

Table 4. Semantic Type Modifier Values for each Context

Semantic Type	Modifier	USA context	UK context	NATO context	Analyst context
eventDate	dateFormat	MM/DD/YYYY	DD/MM/YYYY	DD.MM.YYYY	DD-MM-YYYY
	offsetGMT	0000	0000	0000	0000
eventTime	timeFormat	24HHMM	24HHMM	24HHMM	24HHMM
	offsetGMT	0000	0000	0000	0000
countryCode	ctryCodeStd	FIPS	ISO3166 2-alpha	ISO3166 3-digit	ISO3166 3-alpha
countryName	language	English	English	English	English
cityName	language	English	English	English	English
airport	aptSymType	IATA	ICAO	ICAO	IATA
geoCoord	gcrf	Geodetic	BNG	UTM	MGRS
	datum	WGS84	OGB7	WGS84	WGS84
	format	X (GeoTran)	X (GeoTran)	X (GeoTran)	X (GeoTran)
height	lengthUnit	inches	feet	cm	m
weight	weightUnit	pounds	stones	kg	kg
personName	romanSystem	none	none	none	none

6. Conversion Functions

Context mediation identifies differences between the semantics of data provided by sources and data needed by a receiver. Having identified semantic conflicts by discovering modifier value differences on semantic types, the mediator then looks for a conversion procedure that can be inserted into the query plan to convert source data into the form needed by the receiver.

Some conversions can be done using built-in operators that the mediator knows are available in the query execution engine (e.g. adding or multiplying variables). Other conversions make use of external conversion resources that are defined to the mediator in much the same manner as data sources. In this case, the COIN context mediator needs to know the input-output functionality of a conversion source, but the actual implementation is entirely autonomous. Conversion sources can be web sites wrapped with a Cameleon wrapper [8], Java or other code encapsulated in an HTML or XML interface, relational database tables, XML sources, and web services.

Whatever type of conversion function is used, the reasoning about the applicability of a function to the problem at hand and the insertion of the function into the mediated query plan takes place in the context mediator. The execution of the plan, setting up the call to the conversion source, and integrating the returned data into the query execution stream take place in the COIN execution engine.

6.1 Converting units of measure

To see how conversions functions are defined and used, we will look first at converting units of measure for height and weight. The problem is simple. If source and receiver use different units of measure, the source data value must be multiplied by an appropriate conversion factor.

The Domain Model (discussed in Section 4) defines semantic types for *height* and *weight*. The *height* semantic type has a modifier called *lengthUnit*, which is used to specify the unit of measure for heights used in each context. The *weight* semantic type has a modifier called *weightUnit*, which us used to specify the unit of measure for weights in each context. The attributes height and weight are assigned semantic type of the same name in the elevation axioms for each source. The values of the modifiers are assigned by the context definitions.

When the context mediator detects that the lengthUnit modifier value is different in the source and receiver context, it looks for conversion rules that can resolve the semantic conflict. The conversion rule height is shown below[1]:

```
rule(cvt(commutative, height, _O, lengthUnit, Ctxt, Ms, Vs, Mt, Vt),
       (unit_conv_p(FromUnits, ToUnits, UnitFactor),
       value(FromUnits, Ctxt, Ms),
       value(ToUnits, Ctxt, Mt),
       value(UnitFactor, Ctxt, Ufv),
       multiply(Vs, Ufv, Vt))).
```

The mediator applies the rule binding variables in the rule head (the first line) to context (*Ctxt*), modifier value for the source context (*Ms*), attribute value in the source (*Vs*), modifier value for the target (receiver) context (*Mt*), and attribute value in the target (*Vt*). The fist line of the body

[1] There is a user-friendly interface available for creating conversion rules, as well as the domain model and context definition. We are showing the internal representation in this document.

of the height conversion rule says that the *unit_conv_p* relation will supply a *UnitFactor* given *FromUnits* and *ToUnits*. The next two lines assure that the modifier values (*Ms* and *Mt*) are adjusted to the semantic requirements of the inputs to the *unit_conv_p* relation (*FromUnits* and *ToUnits*). The final *value* clause adjusts the *UnitFactor* to the semantics required for the final step, which multiplies the value in the source (*Vs*) by the adjusted unit factor (*Ufv*) to obtain the value in the target (*Vt*). It should be noted that the mediator does not actually perform the unit factor retrieval or the multiplication itself. Rather, it uses the specifications in the conversion rule to build a plan (i.e., rewrite the query) that is then passed to the Execution Engine for execution.

The *unit_conv_p* relation used in the height conversion rule is specified just as any other source, in this case as a simple elevation of the *unit_conv* external relation that assigns each of the three attributes the root semantic type *basic*.

```
rule('unit_conv_p'(
skolem(basic,_FrmUnit,Ctxt,1,'unit_conv'(_FrmUnit,_ToUnit,_UnitFactor)),
skolem(basic,_ToUnit,Ctxt,2,'unit_conv'(_FrmUnit,_ToUnit,_UnitFactor)),
skolem(basic,_UnitFactor,Ctxt,3,'unit_conv'(_FrmUnit,_ToUnit,_UnitFactor))),
('unit_conv'(_FrmUnit,_ToUnit,_UnitFactor))).
```

The external source relation *unit_conv* is specified twice, once as a relation rule in Prolog for the mediator and again as a schema definition in XML for use by the Execution Engine. The capabilities (*cap*) clause in the Prolog rule and the *BOUND* element in the Prolog specify that the first two attributes (*FrmUnit* and *ToUnit*) must be bound to constant, source attributes, or values known to be previously derived for the relation to be usable.

```
rule(relation(oracle,'unit_conv',ie,
  [['FrmUnit',string],
  ['ToUnit',string],
  ['UnitFactor',string]],
    cap([[1,1,0]],[])),(true)).
```

```
  <RELATION name="unit_conv">
    <SOURCENAME>oracle</SOURCENAME>
    <ATTRIBUTE type="string">FrmUnit</ATTRIBUTE>
    <ATTRIBUTE type="string">ToUnit</ATTRIBUTE>
    <ATTRIBUTE type="NUMBER">UnitFactor</ATTRIBUTE>
    <CAPRECORD>
      <BOUND>[b(1),b(1),f]</BOUND>
      <UNOPS />
    </CAPRECORD>
  </RELATION>
```

In the case of the self-contained version of the CTI Demo, conversion factors were obtained from a web source and stored into an Oracle table. The definitions of the *unit_conv* source just shown use that Oracle table. For the web-based version of the same demo, the same web source is accessed directly by the Execution Engine using a Cameleon wrapper. The web source wrapped was a general units conversion web site located in Germany:

http://www.chemie.fu-berlin.de/chemistry/general/units.html.

Since the structure and functional behavior of the *unit_conv* Oracle table is identical to the structure and functional behavior of the *unit_conv* Cameleon web wrapped source, the only change necessary in the conversion source definitions is the substitution of "cameleon" for "oracle" in each case.

```
rule(relation(cameleon,'unit_conv',ie,
  [['FrmUnit',string],
  ['ToUnit',string],
  ['UnitFactor',string]],
    cap([[1,1,0]],[])),(true)).
```

```
  <RELATION name="unit_conv">
    <SOURCENAME>cameleon</SOURCENAME>
    <ATTRIBUTE type="string">FrmUnit</ATTRIBUTE>
    <ATTRIBUTE type="string">ToUnit</ATTRIBUTE>
    <ATTRIBUTE type="NUMBER">UnitFactor</ATTRIBUTE>
    <CAPRECORD>
      <BOUND>[b(1),b(1),f]</BOUND>
      <UNOPS />
    </CAPRECORD>
  </RELATION>
```

The weight conversion rule is defined similarly to height, in fact, using the same conversion factor source relation.

```
rule(cvt(commutative, weight, _O, weightUnit, Ctxt, Ms, Vs, Mt, Vt),
       (unit_conv_p(Fu, Tu, Uf),
       value(Fu, Ctxt, Ms),
       value(Tu, Ctxt, Mt),
       value(Uf, Ctxt, Ufv),
       multiply(Vs, Ufv, Vt))).
```

6.2 <u>Converting date formats</u>

The conversion of event dates illustrates another type of conversion component supported by context mediation – a purpose-built Java servlet designed to perform the necessary conversions. The DateXForm servlet supports two orders of the elements of a Gregorian date – "American" with month before day and year, and "European" with day before month and year – as well as a choice of punctuation character separating the three parts of the date.

To the mediator, the conversion rule for the eventDate semantic type looks little different from that for converting units of measure. In this case, the whole conversion is performed by the *datexform* relation and, since the *datexform* servlet is designed to work with modifier values directly, there is no need for the value conversion clauses.

```
rule(cvt(commutative, eventDate, _O, dateFmt, Ctxt, Ms, Vs, Mt, Vt),
       (datexform(Vs, Ms, Vt, Mt))).
```

The mediator relation definition rule is also quite similar. The four attributes are specified and the capabilities clause that *Format1* and *Format2* must always be bound along with either *Date1*

```
rule(relation('DateXform', datexform, ie,
       [['Date1', string],
       ['Format1', string],
       ['Date2', string],
       ['Format2', string]],
             cap([[1, 1, 0, 1], [0, 1, 1, 1]],
             [<, >, <>, =<, >=])), (true)).
```

or *Date2*.

Although the mediator sees *datexform* as just another relation, the Execution Engine knows from the separate schema definition that it must invoke a servlet during execution of the plan. The schema definition in XML looks quite different from either the Oracle or Cameleon version of unit conversion.

By locating the datexform servlet locally, it can be used in both the web-based and self-contained version of the demo. Support for additional date conversion functionality or for time conversion can be added by modifying the Java code and building a new servlets.

```
<SOURCE name="datexform" type="SERVLET">
  <URL>http://localhost:8080/gcms_v6/servlet/edu.mit.gcms.datexform.datexform</URL>
  <MAXCONNECTION>10</MAXCONNECTION>
</SOURCE>
```

6.3 Converting country codes

Country codes are converted using an Oracle table loaded with data extracted from an Appendix to the online CIA Factbook. The *ctrycode* table has five columns: row ID, FIPS code and ISO 3166 code in 2-letter alpha, 3-letter alpha, and 3-digit numeric form. By using the commutative property, six conversion rules specify all twelve possible combinations of code conversions. Each rule selects an appropriate pair of columns as source and target. The commutative property allows the mediator to reverse source and target.

```
rule(cvt(commutative, countryCode, _O, ctryCodeStd, Ctxt,
  "ISO3166A2", Vs, "FIPS", Vt),
      ('ctrycode_p'(Nm,Fs,A2,A3,N3),
      value(A2, Ctxt, Vs), value(Fs, Ctxt, Vt))).

rule(cvt(commutative, countryCode, _O, ctryCodeStd, Ctxt,
  "ISO3166A2", Vs, "ISO3166A3", Vt),
      ('ctrycode_p'(Nm,Fs,A2,A3,N3),
      value(A2, Ctxt, Vs), value(A3, Ctxt, Vt))).

rule(cvt(commutative, countryCode, _O, ctryCodeStd, Ctxt,
  "ISO3166A2", Vs, "ISO3166N3", Vt),
      ('ctrycode_p'(Nm,Fs,A2,A3,N3),
      value(A2, Ctxt, Vs), value(N3, Ctxt, Vt))).

rule(cvt(commutative, countryCode, _O, ctryCodeStd, Ctxt,
  "ISO3166A3", Vs, "FIPS", Vt),
      ('ctrycode_p'(Nm,Fs,A2,A3,N3),
      value(A3, Ctxt, Vs), value(Fs, Ctxt, Vt))).

rule(cvt(commutative, countryCode, _O, ctryCodeStd, Ctxt,
  "ISO3166A3", Vs, "ISO3166N3", Vt),
      ('ctrycode_p'(Nm,Fs,A2,A3,N3),
      value(A3, Ctxt, Vs), value(N3, Ctxt, Vt))).

rule(cvt(commutative, countryCode, _O, ctryCodeStd, Ctxt,
  "ISO3166N3", Vs, "FIPS", Vt),
      ('ctrycode_p'(Nm,Fs,A2,A3,N3),
      value(N3, Ctxt, Vs), value(Fs, Ctxt, Vt))).
```

6.4 Converting airport codes

The web-based version of the CTI Demo uses two Cameleon wrappers, *airportiata* and *airporticao*, to access a web site that can convert IATA codes to ICAO or vice versa:

http://www.airlinecodes.co.uk/aptcodesearch.asp

The self-contained version of the Demo emulates the Cameleon wrappers using views on an Oracle table which was loaded with approximately 1000 rows of airport code data.

```
rule(cvt(commutative, airportCode, _O, aptSymType, Ctxt,
  "IATA", Vs, "ICAO", Vt),
      ('airportiata_p'(Tc,Fc,Nm),
      value(Fc, Ctxt, Vs), value(Tc, Ctxt, Vt))).

rule(cvt(commutative, airportCode, _O, aptSymType, Ctxt,
  "ICAO", Vs, "IATA", Vt),
      ('airporticao_p'(Fc,Tc,Nm),
      value(Fc, Ctxt, Vs), value(Tc, Ctxt, Vt))).
```

The conversion rules are simple and similar to country codes, except that the commutative property cannot override the source capability definitions which specify that each relation can only be used in one direction. Capabilities are specified in the *cap* clause of the relation rule for each source.

```
rule(relation(oracle,'airportiata',ie,
  [['icao',string],
  ['iata',string],
  ['aptname',string]],
    cap([[0,1,0]],[])),(true)).

rule(relation(oracle,'airporticao',ie,
  [['icao',string],
  ['iata',string],
  ['aptname',string]],
    cap([[1,0,0]],[])),(true)).
```

6.5 Converting geographic coordinates

The National Geospatial Intelligence Agency provides the GeoTrans software package that can convert among a wide variety of geographic coordinate representations. The ultimate design for the CTI Demo is to access the GeoTrans software through a web service. Until the web service is completed, the demo uses an Oracle table to emulate geographic coordinate conversions. The table is loaded with each form of geographic coordinate needed and conversion rules similar to those used for country codes tell the mediator to use appropriate columns from the Oracle table to accomplish the conversion. The GeoTrans software is available at:

http://earth-info.nga.mil/GandG/geotrans/index.html

6.6 Summary: Conversion Functions

It is important to review that the role of Context Mediation is to automatically discover conflicts, locate and/or build the appropriate conversion functions, and rewrite the original query so that it return the correct (syntactically and semantically) correct results to the receiver. The conversion functions play the role of resolving the semantic conflicts, the use of these is automatic and in most cases transparent to the application.

7. Examples of the CTI Demo in Operation

The examples in this section illustrate how context mediation works on increasingly complex queries involving several different semantic differences and required conversions. For illustrative purposes, the execution of the context mediation process is shown in detail, step by step from query interpretation through to execution results. In actual use, the receiving application (or user) presents a query and receives mediated results in much the same fashion as any other database query. The COIN context mediation process rewrites a receiver query into a mediated query plan instantly and then passes that plan directly along for execution, returning data in the correct semantics needed by the receiver.

The first example in Section 7.1 is a query from an Analyst for the height and weight of a particular individual across sources with different semantic conventions or contexts. The context mediator examines the semantic differences between each source and the Analyst receiver, prepares a plan that introduces necessary conversions, and then executes the plan to return the data in the form the receiver expects. This query demonstrates the simplest form of context mediation involving units of measure conflicts.

The second example in Section 7.2 uses the same query, but shifts the receiver to USA context, illustrating how the context mediator can rapidly reevaluate the semantic differences and generate a new plan meeting the changed requirements.

The third example in Section 7.3 illustrates context mediation on a query for place information to be delivered in the Analyst context. This query demonstrates a number of more complex context conversions. Conversions, when necessary, are automatically inserted for country codes, airport codes, and geographic coordinates.

The fourth example in Section 7.4 repeats the same query for place information with a receiver in UK context, illustrating again the rapid regeneration of the plan and also a limitation of mediation where no conversion of some data is possible (geographic coordinates for places outside Great Britain cannot be converted to BNG coordinates).

A fifth example in Section 7.5 illustrates a query including all CTI reports attributes from all sources. This example requires the simultaneous evaluation of semantic conflicts in six dimensions and thus demonstrates the capabilities of context mediation to rapidly identify and resolve complex semantic conflicts over multiple heterogeneous sources.

In the examples that follow, we use an interface that allows us to control the mediation process and see the operation of the mediation at various stages. Figure 7.1 shows the demo user interface. The Context box is used to select the receiver context, in this case they are: none, c_USA, c_UK, c_NATO, or c_Analyst. Setting the receiver context affects all of the context-dependent actions of the mediator. A context of "none" requests that no mediation be performed.

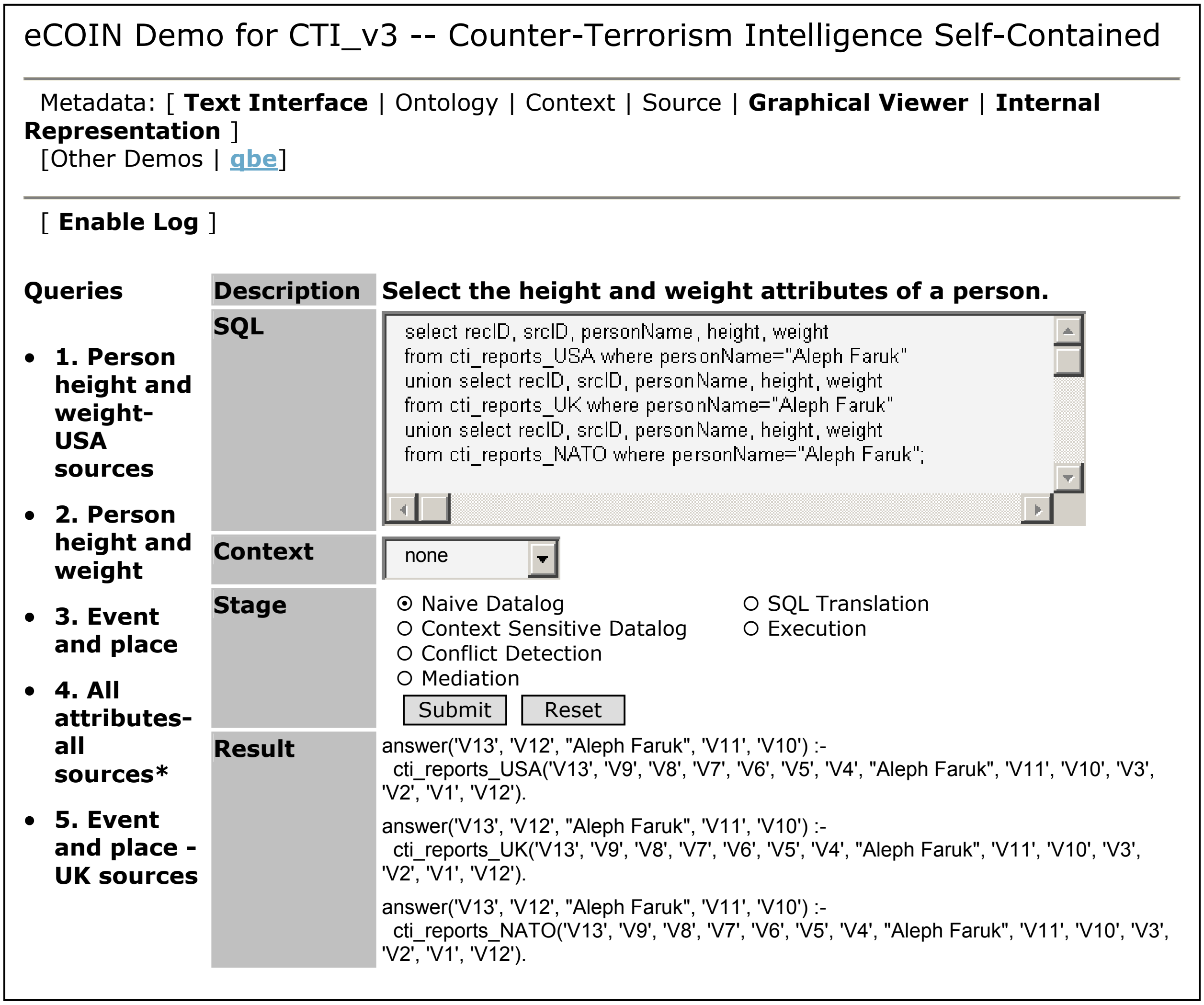

Figure 7.1 CTI Demo User Interface

Below the Context box are radio buttons to select the mediation Stage to be displayed in the Result area below. These stage outputs are provided to provide access to the inner working of the mediator. The first stage is the Naïve Datalog translation of the SQL query, which displays the translated query ignoring all context. The second stage is Context Sensitive Datalog, in which the receiver context is added to the Datalog representation of the query. The third stage is Conflict Detection, which tabulates the semantic conflicts found by the mediator. The fourth stage is Mediation, which displays the mediated Datalog query. The fifth stage is SQL Translation, which shows the mediated query translated back into SQL. The last stage is Execution, which displays the data retrieved by executing the mediated query against the sources referenced in the query (or unmediated in the case of a context of "none").

In the examples in Sections 7.1 through 7.5, these stages will be illustrated in specific cases.

7.1 Example 1. Mediating Person Height and Weight into Analyst Context

The table below shows the height and weight of Aleph Faruk as reported by three different observers: src1 using USA conventions of height in inches and weight in pounds, src2 using NATO conventions of height in centimeters and weight in kilograms, while src3 reports in UK conventions of height in feet and weight in stones. Without mediation it is hard to recognize that the three reports describe the same individual.

recID	srcID	PersonName	Height	Weight	Context
120	src1	Aleph Faruk	68	170	USA
106	src2	Aleph Faruk	173	77	NATO
107	src3	Aleph Faruk	5.67	12.14	UK

The next table shows how the same data would appear when converted to a common Analyst context with height in meters and weight in kilograms:

recID	srcID	PersonName	Height	Weight	Context
120	src1	Aleph Faruk	1.73	77	Analyst
106	src2	Aleph Faruk	1.73	77	Analyst
107	src3	Aleph Faruk	1.73	77	Analyst

Using declarative information about source and receiver data and semantics, the COIN mediator can take a query for height and weight for a person named “Aleph Faruk” from all data sources combined and devise a plan for retrieving the information with appropriate conversions to meet the semantic requirements of the receiver.

An SQL query combining all sources for height and weight of “Aleph Faruk” would like like this:

```
select recID, srcID, personName, height, weight
from cti_reports_USA where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_UK where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_NATO where personName="Aleph Faruk";
```

COIN translates the SQL query into a Naïve Datalog ignoring context differences.

SQL

```
select recID, srcID, personName, height, weight
from cti_reports_USA where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_UK where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_NATO where personName="Aleph Faruk";
```

Context none

Stage ◉ Naive Datalog ○ Context Sensitive Datalog ○ Conflict Detection ○ Mediation ○ SQL Translation ○ Execution

Result

```
answer('V13', 'V12', "Aleph Faruk", 'V11', 'V10') :-
  cti_reports_USA('V13', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', "Aleph Faruk", 'V11', 'V10', 'V3', 'V2', 'V1', 'V12').

answer('V13', 'V12', "Aleph Faruk", 'V11', 'V10') :-
  cti_reports_UK('V13', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', "Aleph Faruk", 'V11', 'V10', 'V3', 'V2', 'V1', 'V12').

answer('V13', 'V12', "Aleph Faruk", 'V11', 'V10') :-
  cti_reports_NATO('V13', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', "Aleph Faruk", 'V11', 'V10', 'V3', 'V2', 'V1', 'V12')
```

Setting the context to "none" and executing the query returns the raw data as provided by the sources

SQL

```
select recID, srcID, personName, height, weight
from cti_reports_USA where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_UK where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_NATO where personName="Aleph Faruk";
```

Context none

Stage ○ Naive Datalog ○ Context Sensitive Datalog ○ Conflict Detection ○ Mediation ○ SQL Translation ◉ Execution

Result

recID	srcID	personName	height	weight
106	src2	Aleph Faruk	173	77
107	src3	Aleph Faruk	5.67	12.14
108	src3	Aleph Faruk	5.67	12.14
117	src3	Aleph Faruk	5.67	12.14
120	src1	Aleph Faruk	68	170

We see that the same individual as reported by three different sources appears to have a height of 68 from src1, 173 from src2, and 5.67 from src3. Similarly his weight is reported as 170 by src1, 77 by src2, and 12.14 by src3. We know in this case that all these reports refer to the same Aleph Faruk at approximately the same point in time. For the analyst to recognize that these separate reports refer to the same individual, we need account for differences in units of measurement used by each source. We have previously discussed how each source is associated with a context (USA for src1, NATO for src2, and UK for src3) and each context defines the units of measure for heights and weights.

We now ask the COIN application to resolve context differences between each of the sources and the context of the Analyst by setting the requested context to “c_Analyst” in the context box. The first step in the reasoning process is to restate the query in Context Sensitive Datalog

```
answer('V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_USA_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V7', c_Analyst, 'V17'),
        'V17' = "Aleph Faruk",
        value('V14', c_Analyst, 'V19'),
        value('V1', c_Analyst, 'V18'),
        value('V6', c_Analyst, 'V16'),
        value('V5', c_Analyst, 'V15').

answer('V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_UK_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V7', c_Analyst, 'V17'),
        'V17' = "Aleph Faruk",
        value('V14', c_Analyst, 'V19'),
        value('V1', c_Analyst, 'V18'),
        value('V6', c_Analyst, 'V16'),
        value('V5', c_Analyst, 'V15').

answer('V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_NATO_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V7', c_Analyst, 'V17'),
        'V17' = "Aleph Faruk",
        value('V14', c_Analyst, 'V19'),
        value('V1', c_Analyst, 'V18'),
        value('V6', c_Analyst, 'V16'),
        value('V5', c_Analyst, 'V15').
```

In the Context Sensitive Datalog, each variable in the answer set is requested as a value converted to the “c_Analyst” context. The COIN mediator then proceeds to examine the context definitions for each source compared to the required output context and detects semantic conflicts that must be resolved.

The COIN mediator performs Semantic conflict detection for each of the disjunctive subqueries in the Context Sensitive Datalog query. For USA context sources, conflict are detected for both height and weight. The COIN mediator also determines that the differences in units of measure can be resolved by applying the unit_conv conversion function (from "lb" to "kg" for weight and from "in" to "m" for height) and multiplying the source value by the factor returned.

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
weight	_weight	cti_reports_USA(Name, _countryName, _countryCode, _cityName, _airport, _placeName, _geoCoord, Aleph Faruk, _height, _weight, _eventType, _eventDate, _eventTime, Name)	weightUnit	c_USA : lb	c_Analyst : kg	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)
height	_height	cti_reports_USA(Name, _countryName, _countryCode, _cityName, _airport, _placeName, _geoCoord, Aleph Faruk, _height, _weight, _eventType, _eventDate, _eventTime, Name)	lengthUnit	c_USA : in	c_Analyst : m	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)

For the UK sources, the COIN mediator performs a similar analysis and detects semantic conflicts again on both height and weight, which can also be resolved by applying the unit_conv conversion function (from "stone" to "kg" for weight and from "ft" to "m" for height) and multiplying the source value by the factor returned.

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
weight	_weight	cti_reports_UK(Name, _countryName, _countryCode, _cityName, _airport, _placeName, _geoCoord, Aleph Faruk, _height, _weight, _eventType, _eventDate, _eventTime, Name)	weightUnit	c_UK : stone	c_Analyst : kg	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)
height	_height	cti_reports_UK(Name, _countryName, _countryCode, _cityName, _airport, _placeName, _geoCoord, Aleph Faruk, _height, _weight, _eventType, _eventDate, _eventTime, Name)	lengthUnit	c_UK : ft	c_Analyst : m	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)

Finally, the COIN mediator examines the subquery for NATO sources, finding only one semantic conflict for height, since weight is represented in kilograms in both NATO sources and the Analyst context. The mediator determines that same unit_conv conversion function can resolve the differences for height by converting from "cm" to "m" and multiplying the source value by the factor returned.

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
height	_height	cti_reports_NATO(Name, _countryName, _countryCode, _cityName, _airport, _placeName, _geoCoord, Aleph Faruk, _height, _weight, _eventType, _eventDate, _eventTime, Name)	lengthUnit	c_NATO : cm	c_Analyst : m	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)

Having detected the semantic conflicts and identified that the conflicts can be resolved by appropriate conversion functions, the COIN mediator rewrites the query as a Mediated Datalog Query.

```
answer('V17', 'V16', "Aleph Faruk", 'V15', 'V14'):-
        cti_reports_USA('V17', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', "Aleph Faruk", 'V7', 'V6', 'V5', 'V4', 'V3', 'V16'),
        unit_conv("in", "m", 'V2'),
        'V15' is 'V7' * 'V2',
        unit_conv("lb", "kg", 'V1'),
        'V14' is 'V6' * 'V1'.

answer('V17', 'V16', "Aleph Faruk", 'V15', 'V14'):-
        cti_reports_UK('V17', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', "Aleph Faruk", 'V7', 'V6', 'V5', 'V4', 'V3', 'V16'),
        unit_conv("ft", "m", 'V2'),
        'V15' is 'V7' * 'V2',
        unit_conv("stone", "kg", 'V1'),
        'V14' is 'V6' * 'V1'.

answer('V15', 'V14', "Aleph Faruk", 'V13', 'V12'):-
        unit_conv("cm", "m", 'V11'),
        'V13' is 'V10' * 'V11',
        cti_reports_NATO('V15', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', "Aleph Faruk", 'V10', 'V12', 'V3', 'V2', 'V1', 'V14').
```

In the first subquery for USA sources in the Mediated Datalog, unit_conv is used to find a conversion factor (V2) for inches to meters. The conversion factor returned in V2 is then multiplied by the height source value, V7, to obtain the required height value, V15, in the receiver c_Analist context. The same unit_conv function is used again to obtain the conversion factor for pounds to kilograms as variable V1, which is then multiplied by the weight source value, V6, to calculate weight in the c_Analyst context, V14. The calculated values, V14 and V15, are then included in the answer returned to the user. Similar calculations are written for each of the other conversions that the mediator has determined are necessary.

The Mediated Datalog Query can then be translated back into SQL.

```
select cti_reports_usa.recID, cti_reports_usa.srcID, 'Aleph Faruk', (cti_reports_usa.height*unit_conv.UnitFactor),
(cti_reports_usa.weight*unit_conv2.UnitFactor)
from   (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, 'Aleph Faruk', height,
weight, eventType, eventDate, eventTime, srcID
        from   cti_reports_usa
        where  personName='Aleph Faruk') cti_reports_usa,
       (select 'in', 'm', UnitFactor
        from   unit_conv
        where  FrmUnit='in'
        and    ToUnit='m') unit_conv,
       (select 'lb', 'kg', UnitFactor
        from   unit_conv
        where  FrmUnit='lb'
        and    ToUnit='kg') unit_conv2
union
select cti_reports_uk.recID, cti_reports_uk.srcID, 'Aleph Faruk', (cti_reports_uk.height*unit_conv3.UnitFactor),
(cti_reports_uk.weight*unit_conv4.UnitFactor)
from   (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, 'Aleph Faruk', height,
weight, eventType, eventDate, eventTime, srcID
        from   cti_reports_uk
        where  personName='Aleph Faruk') cti_reports_uk,
       (select 'ft', 'm', UnitFactor
        from   unit_conv
        where  FrmUnit='ft'
        and    ToUnit='m') unit_conv3,
       (select 'stone', 'kg', UnitFactor
        from   unit_conv
        where  FrmUnit='stone'
        and    ToUnit='kg') unit_conv4
union
select cti_reports_nato.recID, cti_reports_nato.srcID, 'Aleph Faruk',
(cti_reports_nato.height*unit_conv5.UnitFactor), cti_reports_nato.weight
  from (select 'cm', 'm', UnitFactor from unit_conv where FrmUnit='cm' and ToUnit='m') unit_conv5,
       (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, 'Aleph Faruk',
height, weight, eventType, eventDate, eventTime, srcID
        from cti_reports_nato where  personName='Aleph Faruk') cti_reports_nato
```

Executing the mediated query gives the following results

SQL

```
select recID, srcID, personName, height, weight
from cti_reports_USA where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_UK where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_NATO where personName="Aleph Faruk";
```

Context c_Analyst

Stage

- ○ Naive Datalog
- ○ Context Sensitive Datalog
- ○ Conflict Detection
- ○ Mediation
- ○ SQL Translation
- ⊙ Execution

Result

recID	srcID	personName	height	weight
106	src2	Aleph Faruk	1.73	77
107	src3	Aleph Faruk	1.728216	77.0925592052
108	src3	Aleph Faruk	1.728216	77.0925592052
117	src3	Aleph Faruk	1.728216	77.0925592052
120	src1	Aleph Faruk	1.7272	77.1107029

By resolving the semantic differences and returning the data in the Analyst's own context, we can see that all the reports refer to an individual of approximately the same height (1.73 meters) and weight (77 kilograms). Exact matches could be obtained by introducing additional numeric semantic type modifiers and a conversion function for rounding values to a given number of decimal places of precision.

7.2 Example 2. Mediating Person Height and Weight with USA Context Receiver

Suppose another analyst needs to access the same height and weight information used in the preceding example but needs the information in the semantic conventions of the USA context. By changing the receiver context, the COIN context mediation system can rapidly reevaluate the semantic conflicts and rewrite a mediated query plan invoking appropriate conversions where needed.

To see how this works, we change the receiver context in the context box to "c_USA" and ask to see the Context Sensitive Datalog query:

SQL

```
select recID, srcID, personName, height, weight
from cti_reports_USA where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_UK where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_NATO where personName="Aleph Faruk";
```

Context c_USA

Stage
- ○ Naive Datalog
- ⊙ Context Sensitive Datalog
- ○ Conflict Detection
- ○ Mediation
- ○ SQL Translation
- ○ Execution

Result

```
answer('V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_USA_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2',
'V1'),
        value('V7', c_USA, 'V17'),
        'V17' = "Aleph Faruk",
        value('V14', c_USA, 'V19'),
        value('V1', c_USA, 'V18'),
        value('V6', c_USA, 'V16'),
        value('V5', c_USA, 'V15').

answer('V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_UK_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V7', c_USA, 'V17'),
        'V17' = "Aleph Faruk",
        value('V14', c_USA, 'V19'),
        value('V1', c_USA, 'V18'),
        value('V6', c_USA, 'V16'),
        value('V5', c_USA, 'V15').

answer('V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_NATO_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2',
'V1'),
        value('V7', c_USA, 'V17'),
        'V17' = "Aleph Faruk",
        value('V14', c_USA, 'V19'),
        value('V1', c_USA, 'V18'),
        value('V6', c_USA, 'V16').
```

In the Context Sensitive Datalog, each variable in the answer set is requested as a value converted to the "c_USA" context. The COIN mediator then proceeds to examine the context definitions for each source compared to the required output context and detects semantic conflicts that must be resolved.

The COIN mediator performs Semantic conflict detection for each of the disjunctive subqueries in the Context Sensitive Datalog query. For USA context sources, the context is the same in both source and receiver and no conflicts are detected.

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
No conflicts.						

For the UK sources, the COIN mediator performs a similar analysis and detects semantic conflicts again on both height and weight, which can be resolved by applying the unit_conv conversion function (from "stone" to "lb" for weight and from "ft" to "in" for height) and multiplying the source value by the factor returned.

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
weight	_weight	cti_reports_UK(Name, _countryName, _countryCode, _cityName, _airport, _placeName, _geoCoord, Aleph Faruk, _height, _weight, _eventType, _eventDate, _eventTime, Name)	weightUnit	c_UK : stone	c_USA : lb	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)
height	_height	cti_reports_UK(Name, _countryName, _countryCode, _cityName, _airport, _placeName, _geoCoord, Aleph Faruk, _height, _weight, _eventType, _eventDate, _eventTime, Name)	lengthUnit	c_UK : ft	c_USA : in	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)

Finally, the COIN mediator examines the subquery for NATO sources, now finding semantic conflict for both weight and height. The mediator determines that same unit_conv conversion function can resolve the differences for weight by converting from "kg" to "lb" and for height by converting from "cm" to "in" and then multiplying the source value by the factor returned.

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
weight	_weight	cti_reports_NATO(Name, _countryName, _countryCode, _cityName, _airport, _placeName, _geoCoord, Aleph Faruk, _height, _weight, _eventType, _eventDate, _eventTime, Name)	weightUnit	c_NATO : kg	c_USA : lb	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)
height	_height	cti_reports_NATO(Name, _countryName, _countryCode, _cityName, _airport, _placeName, _geoCoord, Aleph Faruk, _height, _weight, _eventType, _eventDate, _eventTime, Name)	lengthUnit	c_NATO : cm	c_USA : in	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)

Having detected the semantic conflicts and identified that the conflicts can be resolved by appropriate conversion functions, the COIN mediator rewrites the query as a Mediated Datalog Query.

```
answer('V13', 'V12', "Aleph Faruk", 'V11', 'V10'):-
        cti_reports_USA('V13', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', "Aleph Faruk", 'V11', 'V10', 'V3', 'V2', 'V1', 'V12').

answer('V17', 'V16', "Aleph Faruk", 'V15', 'V14'):-
        cti_reports_UK('V17', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', "Aleph Faruk", 'V7', 'V6', 'V5', 'V4', 'V3', 'V16'),
        unit_conv("ft", "in", 'V2'),
        'V15' is 'V7' * 'V2',
        unit_conv("stone", "lb", 'V1'),
        'V14' is 'V6' * 'V1'.

answer('V17', 'V16', "Aleph Faruk", 'V15', 'V14'):-
        cti_reports_NATO('V17', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', "Aleph Faruk", 'V7', 'V6', 'V5', 'V4', 'V3',
'V16'),
        unit_conv("cm", "in", 'V2'),
        'V15' is 'V7' * 'V2',
        unit_conv("kg", "lb", 'V1'),
```

In the first subquery for USA sources in the Mediated Datalog, data is simply retrieved and passed straight through. For UK sources, unit_conv is used to find a conversion factor (V2) from feet to inches. The conversion factor returned in V2 is then multiplied by the height source value, V7, to obtain the required height value, V15, in the receiver c_USA context. The same unit_conv function is used again to obtain the conversion factor for stones to pounds as variable V1, which is then multiplied by the weight source value, V6, to calculate weight in the c_USA context in variable V14. The calculated values, V14 and V15, are then included in the answer returned to the user. Similar calculations are written for each of the other conversions that the mediator has determined are necessary for NATO source data.

The Mediated Datalog Query can then be translated back into SQL.

```
select cti_reports_usa.recID, cti_reports_usa.srcID, 'Aleph Faruk', cti_reports_usa.height, cti_reports_usa.weight
from   (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, 'Aleph Faruk', height,
weight, eventType, eventDate, eventTime, srcID
        from   cti_reports_usa
        where  personName='Aleph Faruk') cti_reports_usa
union
select cti_reports_uk.recID, cti_reports_uk.srcID, 'Aleph Faruk', (cti_reports_uk.height*unit_conv.UnitFactor),
(cti_reports_uk.weight*unit_conv2.UnitFactor)
from   (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, 'Aleph Faruk', height,
weight, eventType, eventDate, eventTime, srcID
        from   cti_reports_uk
        where  personName='Aleph Faruk') cti_reports_uk,
       (select 'ft', 'in', UnitFactor
        from   unit_conv
        where  FrmUnit='ft'
        and    ToUnit='in') unit_conv,
       (select 'stone', 'lb', UnitFactor
        from   unit_conv
        where  FrmUnit='stone'
        and    ToUnit='lb') unit_conv2
union
select cti_reports_nato.recID, cti_reports_nato.srcID, 'Aleph Faruk',
(cti_reports_nato.height*unit_conv3.UnitFactor), (cti_reports_nato.weight*unit_conv4.UnitFactor)
from   (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, 'Aleph Faruk', height,
weight, eventType, eventDate, eventTime, srcID
        from   cti_reports_nato
        where  personName='Aleph Faruk') cti_reports_nato,
       (select 'cm', 'in', UnitFactor
        from   unit_conv
        where  FrmUnit='cm'
        and    ToUnit='in') unit_conv3,
       (select 'kg', 'lb', UnitFactor
        from   unit_conv
        where  FrmUnit='kg'
        and    ToUnit='lb') unit_conv4
```

Executing the mediated query gives the following results

SQL

```
select recID, srcID, personName, height, weight
from cti_reports_USA where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_UK where personName="Aleph Faruk"
union select recID, srcID, personName, height, weight
from cti_reports_NATO where personName="Aleph Faruk";
```

Context c_USA

Stage

- ○ Naive Datalog
- ○ Context Sensitive Datalog
- ○ Conflict Detection
- ○ Mediation
- ○ SQL Translation
- ⊙ Execution

Result

recID	srcID	personName	height	weight
106	src2	Aleph Faruk	68.11023667	169.75594174
107	src3	Aleph Faruk	68.04	169.96
108	src3	Aleph Faruk	68.04	169.96
117	src3	Aleph Faruk	68.04	169.96
120	src1	Aleph Faruk	68	170

By resolving the semantic differences and returning the data in the US analyst's USA context, we can again see that all the reports refer to an individual of approximately the same height (68 inches) and weight (170 pounds). Exact matches could be obtained by introducing additional numeric semantic type modifiers and a conversion function for rounding values to a given number of decimal places of precision.

The COIN context mediation system was able to rewrite the query to serve the needs of the USA context user rapidly without needing any new information to programming.

7.3 Example 3. Place Query Mediated into Analyst Context

Query

```
select recID, srcID, geoCoord, cityName, countryName, airport,
   countryCode, placeName from cti_reports_USA
union select recID, srcID, geoCoord, cityName, countryName, airport,
   countryCode, placeName from cti_reports_UK
union select recID, srcID, geoCoord, cityName, countryName, airport,
   countryCode, placeName from cti_reports_NATO ;
```

Naive Datalog

SQL

```
select recID, srcID, geoCoord, cityName, countryName, airport,
   countryCode, placeName from cti_reports_USA
union select recID, srcID, geoCoord, cityName, countryName, airport,
   countryCode, placeName from cti_reports_UK
union select recID, srcID, geoCoord, cityName, countryName, airport,
   countryCode, placeName from cti_reports_NATO ;
```

Context none

Stage
- ⊙ Naive Datalog
- ○ Context Sensitive Datalog
- ○ Conflict Detection
- ○ Mediation
- ○ SQL Translation
- ○ Execution

Result

```
answer('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7'):-
        cti_reports_USA('V14', 'V10', 'V8', 'V11', 'V9', 'V7', 'V12', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1', 'V13').

answer('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7'):-
        cti_reports_UK('V14', 'V10', 'V8', 'V11', 'V9', 'V7', 'V12', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1', 'V13').

answer('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7'):-
        cti_reports_NATO('V14', 'V10', 'V8', 'V11', 'V9', 'V7', 'V12', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1', 'V13').
```

Unmediated Data

SQL

```
select recID, srcID, geoCoord, cityName, countryName, airport,
  countryCode, placeName from cti_reports_USA
union select recID, srcID, geoCoord, cityName, countryName, airport,
  countryCode, placeName from cti_reports_UK
union select recID, srcID, geoCoord, cityName, countryName, airport,
  countryCode, placeName from cti_reports_NATO ;
```

Context none

Stage

- ○ Naive Datalog
- ○ Context Sensitive Datalog
- ○ Conflict Detection
- ○ Mediation
- ○ SQL Translation
- ⦿ Execution

Result

recID	srcID	geoCoord	cityName	countryName	airport	country Code	placeName
101	src1	51:29:25N, 0:13:26W	London	United Kingdom	LHR	UK	Fez Restaurant
102	src1	51:29:25N, 0:13:50W	London	United Kingdom	LHR	UK	Apt on Mall Road
103	src5	TQ2309278610	London	United Kingdom	EGLL	GB	King's Mall Shopping Ctr
104	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
105	src3	TQ0758175848	London	United Kingdom	EGLL	GB	Heathrow Tube Station
106	src2	33N 599043, 5341242	Vienna	Austria	LOWW	040	Schwechat Airport
107	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
108	src3	TQ0758175848	London	United Kingdom	EGLL	GB	Heathrow Tube Station
109	src4	32N 706207, 5358946	Munich	Germany	EDDM	276	Franz Josef Strauss/Riem/ Munich Intl Airport
110	src4	32N 706207, 5358946	Munich	Germany	EDDM	276	Franz Josef Strauss/Riem/ Munich Intl Airport
111	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
112	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
113	src3	TQ0758175848	London	United Kingdom	EGLL	GB	Heathrow Tube Station
114	src3	TQ0758175848	London	United Kingdom	EGLL	GB	Heathrow Tube Station
115	src1	51:29:37N, 0:13:30W	London	United Kingdom	LHR	UK	Hammersmith Tube Station
116	src2	33N 599043, 5341242	Vienna	Austria	LOWW	040	Schwechat Airport
117	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
118	src5	TQ2332278678	London	United Kingdom	EGLL	GB	Hammersmith Tube Station
119	src1	51:29:25N, 0:13:26W	London	United Kingdom	LHR	UK	Fez Restaurant
120	src1	51:29:25N, 0:13:26W	London	United Kingdom	LHR	UK	Fez Restaurant

Context Sensitive Datalog

```
answer('V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_USA_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V14', c_Analyst, 'V22'),
        value('V1', c_Analyst, 'V21'),
        value('V8', c_Analyst, 'V20'),
        value('V11', c_Analyst, 'V19'),
        value('V13', c_Analyst, 'V18'),
        value('V10', c_Analyst, 'V17'),
        value('V12', c_Analyst, 'V16'),
        value('V9', c_Analyst, 'V15').

answer('V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_UK_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V14', c_Analyst, 'V22'),
        value('V1', c_Analyst, 'V21'),
        value('V8', c_Analyst, 'V20'),
        value('V11', c_Analyst, 'V19'),
        value('V13', c_Analyst, 'V18'),
        value('V10', c_Analyst, 'V17'),
        value('V12', c_Analyst, 'V16'),
        value('V9', c_Analyst, 'V15').

answer('V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_NATO_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V14', c_Analyst, 'V22'),
        value('V1', c_Analyst, 'V21'),
        value('V8', c_Analyst, 'V20'),
        value('V11', c_Analyst, 'V19'),
        value('V13', c_Analyst, 'V18'),
        value('V10', c_Analyst, 'V17'),
        value('V12', c_Analyst, 'V16'),
        value('V9', c_Analyst, 'V15').
```

Conflict Detection

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
countryCode	_countryCode	cti_reports_USA(Name, Name, _countryCode, Name, Name, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	ctryCodeStd	c_USA : FIPS	c_Analyst : ISO3166A3	ctrycode_p(V8, V7, V6, V5, V4), value(V6, V3, V2), value(V7, V3, V1)
geoCoord	_geoCoord	cti_reports_USA(Name, Name, _countryCode, Name, Name, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	geoCoordCode	c_USA : geodetic - WGS84-X	c_Analyst : MGRS-WGS84-X	cti_geoTran_convert2_p(V9, V8, V7, V6), value(V9, V5, V4), value(V7, V5, V3), value(V8, V5, V2), value(V6, V5, V1)

countryCode	_countryCode	cti_reports_UK(Name, Name, _countryCode, Name, _airport, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	ctryCodeStd	c_UK : ISO3166A2	c_Analyst : ISO3166A3	ctrycode_p(V8, V7, V6, V5, V4), value(V6, V3, V2), value(V7, V3, V1)
airportCode	_airport	cti_reports_UK(Name, Name, _countryCode, Name, _airport, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	aptSymType	c_UK : ICAO	c_Analyst : IATA	airportiata_p(V6, V5, V4), value(V5, V3, V2), value(V6, V3, V1)
geoCoord	_geoCoord	cti_reports_UK(Name, Name, _countryCode, Name, _airport, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	geoCoordCode	c_UK : BNG-OGB7-X	c_Analyst : MGRS-WGS84-X	cti_geoTran_convert2_p(V9, V8, V7, V6), value(V9, V5, V4), value(V7, V5, V3), value(V8, V5, V2), value(V6, V5, V1)

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
countryCode	_countryCode	cti_reports_NATO(Name, Name, _countryCode, Name, _airport, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	ctryCodeStd	c_NATO : ISO3166N3	c_Analyst : ISO3166A3	ctrycode_p(V8, V7, V6, V5, V4), value(V6, V3, V2), value(V7, V3, V1)
airportCode	_airport	cti_reports_NATO(Name, Name, _countryCode, Name, _airport, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	aptSymType	c_NATO : ICAO	c_Analyst : IATA	airportiata_p(V6, V5, V4), value(V5, V3, V2), value(V6, V3, V1)
geoCoord	_geoCoord	cti_reports_NATO(Name, Name, _countryCode, Name, _airport, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	geoCoordCode	c_NATO : UTM-WGS84-X	c_Analyst : MGRS-WGS84-X	cti_geoTran_convert2_p(V9, V8, V7, V6), value(V9, V5, V4), value(V7, V5, V3), value(V8, V5, V2), value(V6, V5, V1)

Mediated Datalog

```
answer('V19', 'V18', 'V17', 'V16', 'V15', 'V14', 'V13', 'V12'):-
        cti_geoTran_convert2("geodetic-WGS84-X", 'V11', "MGRS-WGS84-X", 'V17'),
        cti_ctrycode('V10', 'V9', 'V8', 'V13', 'V7'),
        cti_reports_USA('V19', 'V15', 'V9', 'V16', 'V14', 'V12', 'V11', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1', 'V18').

answer('V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15', 'V14'):-
        cti_geoTran_convert2("BNG-OGB7-X", 'V13', "MGRS-WGS84-X", 'V19'),
        airporticao('V12', 'V16', 'V11'),
        cti_ctrycode('V10', 'V9', 'V8', 'V15', 'V7'),
        cti_reports_UK('V21', 'V17', 'V8', 'V18', 'V12', 'V14', 'V13', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1', 'V20').

answer('V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15', 'V14'):-
        cti_geoTran_convert2("UTM-WGS84-X", 'V13', "MGRS-WGS84-X", 'V19'),
        airporticao('V12', 'V16', 'V11'),
        cti_ctrycode('V10', 'V9', 'V8', 'V15', 'V7'),
        cti_reports_NATO('V21', 'V17', 'V7', 'V18', 'V12', 'V14', 'V13', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1', 'V20').
```

Mediated SQL Query

```
select cti_reports_usa.recID, cti_reports_usa.srcID, cti_geotran_convert2.outValue, cti_reports_usa.cityName,
cti_reports_usa.countryName, cti_reports_usa.airport, cti_ctrycode.ISO3166A3, cti_reports_usa.placeName
from   (select 'geodetic-WGS84-X', inValue, 'MGRS-WGS84-X', outValue
        from   cti_geotran_convert2
        where  inCode='geodetic-WGS84-X'
        and    outCode='MGRS-WGS84-X') cti_geotran_convert2,
       (select name, FIPS, ISO3166A2, ISO3166A3, ISO3166N3
        from   cti_ctrycode) cti_ctrycode,
       (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, personName, height, weight,
eventType, eventDate, eventTime, srcID
        from   cti_reports_usa) cti_reports_usa
where  cti_ctrycode.FIPS = cti_reports_usa.countryCode
and    cti_geotran_convert2.inValue = cti_reports_usa.geoCoord
union
select cti_reports_uk.recID, cti_reports_uk.srcID, cti_geotran_convert22.outValue, cti_reports_uk.cityName,
cti_reports_uk.countryName, airporticao.iata, cti_ctrycode2.ISO3166A3, cti_reports_uk.placeName
from   (select 'BNG-OGB7-X', inValue, 'MGRS-WGS84-X', outValue
        from   cti_geotran_convert2
        where  inCode='BNG-OGB7-X'
        and    outCode='MGRS-WGS84-X') cti_geotran_convert22,
       (select icao, iata, aptname
        from   airporticao) airporticao,
       (select name, FIPS, ISO3166A2, ISO3166A3, ISO3166N3
        from   cti_ctrycode) cti_ctrycode2,
       (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, personName, height, weight,
eventType, eventDate, eventTime, srcID
        from   cti_reports_uk) cti_reports_uk
where  cti_ctrycode2.ISO3166A2 = cti_reports_uk.countryCode
and    cti_geotran_convert22.inValue = cti_reports_uk.geoCoord
and    airporticao.icao = cti_reports_uk.airport
union
select cti_reports_nato.recID, cti_reports_nato.srcID, cti_geotran_convert23.outValue, cti_reports_nato.cityName,
cti_reports_nato.countryName, airporticao2.iata, cti_ctrycode3.ISO3166A3, cti_reports_nato.placeName
from   (select 'UTM-WGS84-X', inValue, 'MGRS-WGS84-X', outValue
        from   cti_geotran_convert2
        where  inCode='UTM-WGS84-X'
        and    outCode='MGRS-WGS84-X') cti_geotran_convert23,
       (select icao, iata, aptname
        from   airporticao) airporticao2,
       (select name, FIPS, ISO3166A2, ISO3166A3, ISO3166N3
        from   cti_ctrycode) cti_ctrycode3,
       (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, personName, height, weight,
eventType, eventDate, eventTime, srcID
        from   cti_reports_nato) cti_reports_nato
where  cti_ctrycode3.ISO3166N3 = cti_reports_nato.countryCode
and    cti_geotran_convert23.inValue = cti_reports_nato.geoCoord
and    airporticao2.icao = cti_reports_nato.airport
```

Mediated Data Execution

SQL

```
select recID, srcID, geoCoord, cityName, countryName, airport,
  countryCode, placeName from cti_reports_USA
union select recID, srcID, geoCoord, cityName, countryName, airport,
  countryCode, placeName from cti_reports_UK
union select recID, srcID, geoCoord, cityName, countryName, airport,
  countryCode, placeName from cti_reports_NATO ;
```

Context c_Analyst

Stage

- ○ Naive Datalog
- ○ Context Sensitive Datalog
- ○ Conflict Detection
- ○ Mediation
- ○ SQL Translation
- ⦿ Execution

Result

recID	srcID	geoCoord	cityName	countryName	airport	country Code	placeName
101	src1	30UXC9272208003	London	United Kingdom	LHR	GBR	Fez Restaurant
102	src1	30UXC9225907985	London	United Kingdom	LHR	GBR	Apt on Mall Road
103	src5	30UXC9240108300	London	United Kingdom	LHR	GBR	King's Mall Shopping Ctr
104	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	Heathrow Airport
105	src3	30UXC7692705325	London	United Kingdom	LHR	GBR	Heathrow Tube Station
106	src2	33UWP9904341242	Vienna	Austria	VIE	AUT	Schwechat Airport
107	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	Heathrow Airport
108	src3	30UXC7692705325	London	United Kingdom	LHR	GBR	Heathrow Tube Station
109	src4	32UQU0620758946	Munich	Germany	MUC	DEU	Franz Josef Strauss/Riem/Munich Intl Airport
110	src4	32UQU0620758946	Munich	Germany	MUC	DEU	Franz Josef Strauss/Riem/Munich Intl Airport
111	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	Heathrow Airport
112	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	Heathrow Airport
113	src3	30UXC7692705325	London	United Kingdom	LHR	GBR	Heathrow Tube Station
114	src3	30UXC7692705325	London	United Kingdom	LHR	GBR	Heathrow Tube Station
115	src1	30UXC9263008370	London	United Kingdom	LHR	GBR	Hammersmith Tube Station
116	src2	33UWP9904341242	Vienna	Austria	VIE	AUT	Schwechat Airport
117	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	Heathrow Airport
118	src5	30UXC9263008370	London	United Kingdom	LHR	GBR	Hammersmith Tube Station
119	src1	30UXC9272208003	London	United Kingdom	LHR	GBR	Fez Restaurant
120	src1	30UXC9272208003	London	United Kingdom	LHR	GBR	Fez Restaurant

7.4 Example 4. Place Query Mediated into UK Context

Context Sensitive Datalog

SQL

```
select recID, srcID, geoCoord, cityName, countryName, airport,
   countryCode, placeName from cti_reports_USA
union select recID, srcID, geoCoord, cityName, countryName, airport,
   countryCode, placeName from cti_reports_UK
union select recID, srcID, geoCoord, cityName, countryName, airport,
   countryCode, placeName from cti_reports_NATO ;
```

Context c_UK

Stage

- ○ Naive Datalog
- ⊙ Context Sensitive Datalog
- ○ Conflict Detection
- ○ Mediation
- ○ SQL Translation
- ○ Execution

Result

```
answer('V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_USA_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V14', c_UK, 'V22'),
        value('V1', c_UK, 'V21'),
        value('V8', c_UK, 'V20'),
        value('V11', c_UK, 'V19'),
        value('V13', c_UK, 'V18'),
        value('V10', c_UK, 'V17'),
        value('V12', c_UK, 'V16'),
        value('V9', c_UK, 'V15').

answer('V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_UK_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V14', c_UK, 'V22'),
        value('V1', c_UK, 'V21'),
        value('V8', c_UK, 'V20'),
        value('V11', c_UK, 'V19'),
        value('V13', c_UK, 'V18'),
        value('V10', c_UK, 'V17'),
        value('V12', c_UK, 'V16'),
        value('V9', c_UK, 'V15').

answer('V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_NATO_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V14', c_UK, 'V22'),
        value('V1', c_UK, 'V21'),
        value('V8', c_UK, 'V20'),
        value('V11', c_UK, 'V19'),
        value('V13', c_UK, 'V18'),
        value('V10', c_UK, 'V17'),
        value('V12', c_UK, 'V16'),
        value('V9', c_UK, 'V15')
```

Conflict Detection – USA Sources

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
countryCode	_countryCode	cti_reports_USA(Name, Name, _countryCode, Name, _airport, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	ctryCodeStd	c_USA : FIPS	c_UK : ISO3166A2	ctrycode_p(V8, V7, V6, V5, V4), value(V6, V3, V2), value(V7, V3, V1)
airportCode	_airport	cti_reports_USA(Name, Name, _countryCode, Name, _airport, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	aptSymType	c_USA : IATA	c_UK : ICAO	airportiata_p(V6, V5, V4), value(V5, V3, V2), value(V6, V3, V1)
geoCoord	_geoCoord	cti_reports_USA(Name, Name, _countryCode, Name, _airport, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	geoCoordCode	c_USA : geodetic-WGS84-X	c_UK : BNG-OGB7-X	cti_geoTran_convert2_p(V9, V8, V7, V6), value(V9, V5, V4), value(V7, V5, V3), value(V8, V5, V2), value(V6, V5, V1)

Conflict Detection – UK Sources

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
No conflicts.						

Conflict Detection – NATO Sources

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
countryCode	_countryCode	cti_reports_NATO(Name, Name, _countryCode, Name, Name, _placeName, _geoCoord, _personName, _height, _weight, _eventType,	ctryCodeStd	c_NATO : ISO3166N3	c_UK : ISO3166A2	ctrycode_p(V8, V7, V6, V5, V4), value(V6, V3, V2), value(V7, V3, V1)

		_eventDate, _eventTime, Name)				
geoCoord	_geoCoord	cti_reports_NATO(Name, Name, _countryCode, Name, Name, _placeName, _geoCoord, _personName, _height, _weight, _eventType, _eventDate, _eventTime, Name)	geoCoordCode	c_NATO : UTM-WGS84-X	c_UK : BNG-OGB7-X	cti_geoTran_convert2_p(V9, V8, V7, V6), value(V9, V5, V4), value(V7, V5, V3), value(V8, V5, V2), value(V6, V5, V1)

Mediated Datalog

```
answer('V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15', 'V14'):-
        cti_geoTran_convert2("geodetic-WGS84-X", 'V13', "BNG-OGB7-X", 'V19'),
        airportiata('V16', 'V12', 'V11'),
        cti_ctrycode('V10', 'V9', 'V15', 'V8', 'V7'),
        cti_reports_USA('V21', 'V17', 'V9', 'V18', 'V12', 'V14', 'V13', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1', 'V20').

answer('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7'):-
        cti_reports_UK('V14', 'V10', 'V8', 'V11', 'V9', 'V7', 'V12', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1', 'V13').

answer('V19', 'V18', 'V17', 'V16', 'V15', 'V14', 'V13', 'V12'):-
        cti_geoTran_convert2("UTM-WGS84-X", 'V11', "BNG-OGB7-X", 'V17'),
        cti_ctrycode('V10', 'V9', 'V13', 'V8', 'V7'),
        cti_reports_NATO('V19', 'V15', 'V7', 'V16', 'V14', 'V12', 'V11', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1', 'V18').
```

Mediated SQL Query

```
select cti_reports_usa.recID, cti_reports_usa.srcID, cti_geotran_convert2.outValue, cti_reports_usa.cityName,
cti_reports_usa.countryName, airportiata.icao, cti_ctrycode.ISO3166A2, cti_reports_usa.placeName
from   (select 'geodetic-WGS84-X', inValue, 'BNG-OGB7-X', outValue
        from   cti_geotran_convert2
        where  inCode='geodetic-WGS84-X'
        and    outCode='BNG-OGB7-X') cti_geotran_convert2,
       (select icao, iata, aptname
        from   airportiata) airportiata,
       (select name, FIPS, ISO3166A2, ISO3166A3, ISO3166N3
        from   cti_ctrycode) cti_ctrycode,
       (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, personName, height, weight,
eventType, eventDate, eventTime, srcID
        from   cti_reports_usa) cti_reports_usa
where  cti_ctrycode.FIPS = cti_reports_usa.countryCode
and    cti_geotran_convert2.inValue = cti_reports_usa.geoCoord
and    airportiata.iata = cti_reports_usa.airport
union
select cti_reports_uk.recID, cti_reports_uk.srcID, cti_reports_uk.geoCoord, cti_reports_uk.cityName,
cti_reports_uk.countryName, cti_reports_uk.airport, cti_reports_uk.countryCode, cti_reports_uk.placeName
from   (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, personName, height,
weight, eventType, eventDate, eventTime, srcID
        from   cti_reports_uk) cti_reports_uk
union
select cti_reports_nato.recID, cti_reports_nato.srcID, cti_geotran_convert22.outValue, cti_reports_nato.cityName,
cti_reports_nato.countryName, cti_reports_nato.airport, cti_ctrycode2.ISO3166A2, cti_reports_nato.placeName
from   (select 'UTM-WGS84-X', inValue, 'BNG-OGB7-X', outValue
        from   cti_geotran_convert2
        where  inCode='UTM-WGS84-X'
        and    outCode='BNG-OGB7-X') cti_geotran_convert22,
       (select name, FIPS, ISO3166A2, ISO3166A3, ISO3166N3
        from   cti_ctrycode) cti_ctrycode2,
       (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, personName, height, weight,
eventType, eventDate, eventTime, srcID
        from   cti_reports_nato) cti_reports_nato
where  cti_ctrycode2.ISO3166N3 = cti_reports_nato.countryCode
and    cti_geotran_convert22.inValue = cti_reports_nato.geoCoord
```

Mediated Data Execution

SQL

```
select recID, srcID, geoCoord, cityName, countryName, airport,
  countryCode, placeName from cti_reports_USA
union select recID, srcID, geoCoord, cityName, countryName, airport,
  countryCode, placeName from cti_reports_UK
union select recID, srcID, geoCoord, cityName, countryName, airport,
  countryCode, placeName from cti_reports_NATO ;
```

Context c_UK

Stage

- ○ Naive Datalog
- ○ Context Sensitive Datalog
- ○ Conflict Detection
- ○ Mediation
- ○ SQL Translation
- ⦿ Execution

Result

recID	srcID	geoCoord	cityName	countryName	airport	countryCode	placeName
101	src1	TQ2340878309	London	United Kingdom	EGLL	GB	Fez Restaurant
102	src1	TQ2294578298	London	United Kingdom	EGLL	GB	Apt on Mall Road
103	src5	TQ2309278610	London	United Kingdom	EGLL	GB	King's Mall Shopping Ctr
104	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
105	src3	TQ0758175848	London	United Kingdom	EGLL	GB	Heathrow Tube Station
106	src2	N/A	Vienna	Austria	LOWW	AT	Schwechat Airport
107	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
108	src3	TQ0758175848	London	United Kingdom	EGLL	GB	Heathrow Tube Station
109	src4	N/A	Munich	Germany	EDDM	DE	Franz Josef Strauss/Riem/ Munich Intl Airport
110	src4	N/A	Munich	Germany	EDDM	DE	Franz Josef Strauss/Riem/ Munich Intl Airport
111	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
112	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
113	src3	TQ0758175848	London	United Kingdom	EGLL	GB	Heathrow Tube Station
114	src3	TQ0758175848	London	United Kingdom	EGLL	GB	Heathrow Tube Station
115	src1	TQ2332278678	London	United Kingdom	EGLL	GB	Hammersmith Tube Station
116	src2	N/A	Vienna	Austria	LOWW	AT	Schwechat Airport
117	src3	TQ0776675326	London	United Kingdom	EGLL	GB	Heathrow Airport
118	src5	TQ2332278678	London	United Kingdom	EGLL	GB	Hammersmith Tube Station
119	src1	TQ2340878309	London	United Kingdom	EGLL	GB	Fez Restaurant
120	src1	TQ2340878309	London	United Kingdom	EGLL	GB	Fez Restaurant

7.5 Example 5. Mediating All Attributes from All Sources into Analyst Context

Query

select recID, srcID, geoCoord, cityName, countryName, airport, countryCode, eventDate, eventTime, eventType, placeName, personName, height, weight from cti_reports_USA
union select recID, srcID, geoCoord, cityName, countryName, airport, countryCode, eventDate, eventTime, eventType, placeName, personName, height, weight from cti_reports_UK
union select recID, srcID, geoCoord, cityName, countryName, airport, countryCode, eventDate, eventTime, eventType, placeName, personName, height, weight from cti_reports_NATO ;

Naïve Datalog

```
answer('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'):-
        cti_reports_USA('V14', 'V10', 'V8', 'V11', 'V9', 'V4', 'V12', 'V3', 'V2', 'V1', 'V5', 'V7', 'V6', 'V13').

answer('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'):-
        cti_reports_UK('V14', 'V10', 'V8', 'V11', 'V9', 'V4', 'V12', 'V3', 'V2', 'V1', 'V5', 'V7', 'V6', 'V13').

answer('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'):-
        cti_reports_NATO('V14', 'V10', 'V8', 'V11', 'V9', 'V4', 'V12', 'V3', 'V2', 'V1', 'V5', 'V7', 'V6', 'V13').
```

Execute in context none

recID	srcID	geoCoord	city Name	country Name	air port	country Code	event Date	event Time	event Type	placeName	personName	height	weight
101	src1	51:29:25N, 0:13:26W	London	United Kingdom	LHR	UK	12/01/2004	0900	meeting	Fez Restaurant	Ahmet Khatib	70	170
102	src1	51:29:25N, 0:13:50W	London	United Kingdom	LHR	UK	12/01/2004	1200	rents apt	Apt on Mall Road	Ahmet Khatib	70	170
103	src5	TQ2309278610	London	United Kingdom	EGLL	GB	02/12/2004	1700	meeting	King's Mall Shopping Ctr	Ahmet Khatib	5.75	11.43
104	src3	TQ0776675326	London	United Kingdom	EGLL	GB	03/12/2004	1430	observed	Heathrow Airport	Akka Mohammed	5.83	11.43
105	src3	TQ0758175848	London	United Kingdom	EGLL	GB	04/12/2004	2100	observed	Heathrow Tube Station	Al Pavlakkih	5.75	12.14
106	src2	33N 599043, 5341242	Vienna	Austria	LOW W	040	04.12.2004	1100	plane departs	Schwechat Airport	Aleph Faruk	173	77
107	src3	TQ0776675326	London	United Kingdom	EGLL	GB	04/12/2004	1245	plane arrives	Heathrow Airport	Aleph Faruk	5.67	12.14
108	src3	TQ0758175848	London	United Kingdom	EGLL	GB	04/12/2004	1330	observed	Heathrow Tube Station	Aleph Faruk	5.67	12.14
109	src4	32N 706207, 5358946	Munich	Germany	EDDM	276	03.12.2004	1415	plane departs	Franz Josef Strauss/ Riem/Munich Intl Airport	Akka Mohammed	178	73
110	src4	32N 706207, 5358946	Munich	Germany	EDDM	276	03.12.2004	1415	plane departs	Franz Josef Strauss/ Riem/Munich Intl Airport	Al Pavlakkih	175	77
111	src3	TQ0776675326	London	United Kingdom	EGLL	GB	04/12/2004	1630	plane arrives	Heathrow Airport	Akka Mohammed	5.83	11.43
112	src3	TQ0776675326	London	United Kingdom	EGLL	GB	04/12/2004	1630	plane arrives	Heathrow Airport	Al Pavlakkih	5.75	12.14
113	src3	TQ0758175848	London	United Kingdom	EGLL	GB	04/12/2004	1710	observed	Heathrow Tube Station	Akka Mohammed	5.83	11.43
114	src3	TQ0758175848	London	United Kingdom	EGLL	GB	04/12/2004	1710	observed	Heathrow Tube Station	Al Pavlakkih	5.75	12.14
115	src1	51:29:37N, 0:13:30W	London	United Kingdom	LHR	UK	12/04/2004	1800	observed	Hammersmith Tube Station	Al Pavlakkih	69	170
116	src2	33N 599043, 5341242	Vienna	Austria	LOW W	040	04.12.2004	1000	plane departs	Schwechat Airport	Ali Abdullah	168	73
117	src3	TQ0776675326	London	United Kingdom	EGLL	GB	04/12/2004	1245	plane arrives	Heathrow Airport	Aleph Faruk	5.67	12.14
118	src5	TQ2332278678	London	United Kingdom	EGLL	GB	04/12/2004	2100	observed	Hammersmith Tube Station	Ali Hakem	5.83	15.71
119	src1	51:29:25N, 0:13:26W	London	United Kingdom	LHR	UK	12/04/2004	2100	meeting	Fez Restaurant	Ahmet Khatib	70	170
120	src1	51:29:25N, 0:13:26W	London	United Kingdom	LHR	UK	12/04/2004	2100	meeting	Fez Restaurant	Aleph Faruk	68	170

Context Sensitive Datalog

```
answer('V28', 'V27', 'V26', 'V25', 'V24', 'V23', 'V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_USA_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V14', c_Analyst, 'V28'),
        value('V1', c_Analyst, 'V27'),
        value('V8', c_Analyst, 'V26'),
        value('V11', c_Analyst, 'V25'),
        value('V13', c_Analyst, 'V24'),
        value('V10', c_Analyst, 'V23'),
        value('V12', c_Analyst, 'V22'),
        value('V3', c_Analyst, 'V21'),
        value('V2', c_Analyst, 'V20'),
        value('V4', c_Analyst, 'V19'),
        value('V9', c_Analyst, 'V18'),
        value('V7', c_Analyst, 'V17'),
        value('V6', c_Analyst, 'V16'),
        value('V5', c_Analyst, 'V15').

answer('V28', 'V27', 'V26', 'V25', 'V24', 'V23', 'V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_UK_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V14', c_Analyst, 'V28'),
        value('V1', c_Analyst, 'V27'),
        value('V8', c_Analyst, 'V26'),
        value('V11', c_Analyst, 'V25'),
        value('V13', c_Analyst, 'V24'),
        value('V10', c_Analyst, 'V23'),
        value('V12', c_Analyst, 'V22'),
        value('V3', c_Analyst, 'V21'),
        value('V2', c_Analyst, 'V20'),
        value('V4', c_Analyst, 'V19'),
        value('V9', c_Analyst, 'V18'),
        value('V7', c_Analyst, 'V17'),
        value('V6', c_Analyst, 'V16'),
        value('V5', c_Analyst, 'V15').

answer('V28', 'V27', 'V26', 'V25', 'V24', 'V23', 'V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15'):-
        cti_reports_NATO_p('V14', 'V13', 'V12', 'V11', 'V10', 'V9', 'V8', 'V7', 'V6', 'V5', 'V4', 'V3', 'V2', 'V1'),
        value('V14', c_Analyst, 'V28'),
        value('V1', c_Analyst, 'V27'),
        value('V8', c_Analyst, 'V26'),
        value('V11', c_Analyst, 'V25'),
        value('V13', c_Analyst, 'V24'),
        value('V10', c_Analyst, 'V23'),
        value('V12', c_Analyst, 'V22'),
        value('V3', c_Analyst, 'V21'),
        value('V2', c_Analyst, 'V20'),
        value('V4', c_Analyst, 'V19'),
        value('V9', c_Analyst, 'V18'),
        value('V7', c_Analyst, 'V17'),
        value('V6', c_Analyst, 'V16'),
        value('V5', c_Analyst, 'V15').
```

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
weight	_weight	cti_reports_USA(Name, Name, _countryCode, Name, Name, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	weightUnit	c_USA : lb	c_Analyst : kg	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)
height	_height	cti_reports_USA(Name, Name, _countryCode, Name, Name, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	lengthUnit	c_USA : in	c_Analyst : m	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)
eventDate	_eventDate	cti_reports_USA(Name, Name, _countryCode, Name, Name, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	dateFmt	c_USA : American Style /	c_Analyst : American Style -	datexform(V4, V3, V2, V1)
countryCode	_countryCode	cti_reports_USA(Name, Name, _countryCode, Name, Name, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	ctryCodeStd	c_USA : FIPS	c_Analyst : ISO3166A3	ctrycode_p(V8, V7, V6, V5, V4), value(V6, V3, V2), value(V7, V3, V1)
geoCoord	_geoCoord	cti_reports_USA(Name, Name, _countryCode, Name, Name, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	geoCoordCode	c_USA : geodetic-WGS84-X	c_Analyst : MGRS-WGS84-X	cti_geoTran_convert2_p(V9, V8, V7, V6), value(V9, V5, V4), value(V7, V5, V3), value(V8, V5, V2), value(V6, V5, V1)

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
weight	_weight	cti_reports_UK(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	weightUnit	c_UK : stone	c_Analyst : kg	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)
height	_height	cti_reports_UK(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	lengthUnit	c_UK : ft	c_Analyst : m	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)
eventDate	_eventDate	cti_reports_UK(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	dateFmt	c_UK : European Style /	c_Analyst : American Style -	datexform(V4, V3, V2, V1)
countryCode	_countryCode	cti_reports_UK(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	ctryCodeStd	c_UK : ISO3166A2	c_Analyst : ISO3166A3	ctrycode_p(V8, V7, V6, V5, V4), value(V6, V3, V2), value(V7, V3, V1)
airportCode	_airport	cti_reports_UK(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	aptSymType	c_UK : ICAO	c_Analyst : IATA	airportiata_p(V6, V5, V4), value(V5, V3, V2), value(V6, V3, V1)
geoCoord	_geoCoord	cti_reports_UK(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	geoCoordCode	c_UK : BNG-OGB7-X	c_Analyst : MGRS-WGS84-X	cti_geoTran_convert2_p(V9, V8, V7, V6), value(V9, V5, V4), value(V7, V5, V3), value(V8, V5, V2), value(V6, V5, V1)

SemanticType	Column	Source	Modifier	Modifier value in source context	Modifier value in target context	Conversion Function
height	_height	cti_reports_NATO(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	lengthUnit	c_NATO : cm	c_Analyst : m	unit_conv_p(V9, V8, V7), value(V9, V6, V5), value(V8, V6, V4), value(V7, V6, V3), multiply(V2, V3, V1)
eventDate	_eventDate	cti_reports_NATO(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	dateFmt	c_NATO : European Style .	c_Analyst : American Style -	datexform(V4, V3, V2, V1)
countryCode	_countryCode	cti_reports_NATO(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	ctryCodeStd	c_NATO : ISO3166N3	c_Analyst : ISO3166A3	ctrycode_p(V8, V7, V6, V5, V4), value(V6, V3, V2), value(V7, V3, V1)
airportCode	_airport	cti_reports_NATO(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	aptSymType	c_NATO : ICAO	c_Analyst : IATA	airportiata_p(V6, V5, V4), value(V5, V3, V2), value(V6, V3, V1)
geoCoord	_geoCoord	cti_reports_NATO(Name, Name, _countryCode, Name, _airport, Name, _geoCoord, Name, _height, _weight, Name, _eventDate, Name, Name)	geoCoordCode	c_NATO : UTM-WGS84-X	c_Analyst : MGRS-WGS84-X	cti_geoTran_convert2_p(V9, V8, V7, V6), value(V9, V5, V4), value(V7, V5, V3), value(V8, V5, V2), value(V6, V5, V1)

```
answer('V24', 'V23', 'V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15', 'V14', 'V13', 'V12', 'V11'):-
        cti_geoTran_convert2("geodetic-WGS84-X", 'V10', "MGRS-WGS84-X", 'V22'),
        cti_ctrycode('V9', 'V8', 'V7', 'V18', 'V6'),
        datexform('V5', "American Style /", 'V17', "American Style -"),
        cti_reports_USA('V24', 'V20', 'V8', 'V21', 'V19', 'V14', 'V10', 'V13', 'V4', 'V3', 'V15', 'V5', 'V16', 'V23'),
        unit_conv("in", "m", 'V2'),
        'V12' is 'V4' * 'V2',
        unit_conv("lb", "kg", 'V1'),
        'V11' is 'V3' * 'V1'.

answer('V26', 'V25', 'V24', 'V23', 'V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15', 'V14', 'V13'):-
        cti_geoTran_convert2("BNG-OGB7-X", 'V12', "MGRS-WGS84-X", 'V24'),
        airporticao('V11', 'V21', 'V10'),
        cti_ctrycode('V9', 'V8', 'V7', 'V20', 'V6'),
        datexform('V5', "European Style /", 'V19', "American Style -"),
        cti_reports_UK('V26', 'V22', 'V7', 'V23', 'V11', 'V16', 'V12', 'V15', 'V4', 'V3', 'V17', 'V5', 'V18', 'V25'),
        unit_conv("ft", "m", 'V2'),
        'V14' is 'V4' * 'V2',
        unit_conv("stone", "kg", 'V1'),
        'V13' is 'V3' * 'V1'.

answer('V24', 'V23', 'V22', 'V21', 'V20', 'V19', 'V18', 'V17', 'V16', 'V15', 'V14', 'V13', 'V12', 'V11'):-
        cti_geoTran_convert2("UTM-WGS84-X", 'V10', "MGRS-WGS84-X", 'V22'),
        airporticao('V9', 'V19', 'V8'),
        cti_ctrycode('V7', 'V6', 'V5', 'V18', 'V4'),
        datexform('V3', "European Style .", 'V17', "American Style -"),
        unit_conv("cm", "m", 'V2'),
        'V12' is 'V1' * 'V2',
        cti_reports_NATO('V24', 'V20', 'V4', 'V21', 'V9', 'V14', 'V10', 'V13', 'V1', 'V11', 'V15', 'V3', 'V16', 'V23').
```

```
select cti_reports_usa.recID, cti_reports_usa.srcID, cti_geotran_convert2.outValue,
cti_reports_usa.cityName, cti_reports_usa.countryName, cti_reports_usa.airport, cti_ctrycode.ISO3166A3,
datexform.date2, cti_reports_usa.eventTime, cti_reports_usa.eventType, cti_reports_usa.placeName,
cti_reports_usa.personName, (cti_reports_usa.height*unit_conv.UnitFactor),
(cti_reports_usa.weight*unit_conv2.UnitFactor)
from   (select 'geodetic-WGS84-X', inValue, 'MGRS-WGS84-X', outValue
        from   cti_geotran_convert2
        where  inCode='geodetic-WGS84-X'
        and    outCode='MGRS-WGS84-X') cti_geotran_convert2,
       (select name, FIPS, ISO3166A2, ISO3166A3, ISO3166N3
        from   cti_ctrycode) cti_ctrycode,
       (select date1, 'American Style /', date2, 'American Style -'
        from   datexform
        where  format1='American Style /'
        and    format2='American Style -') datexform,
       (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, personName,
height, weight, eventType, eventDate, eventTime, srcID
        from   cti_reports_usa) cti_reports_usa,
       (select 'in', 'm', UnitFactor
        from   unit_conv
        where  FrmUnit='in'
        and    ToUnit='m') unit_conv,
       (select 'lb', 'kg', UnitFactor
        from   unit_conv
        where  FrmUnit='lb'
        and    ToUnit='kg') unit_conv2
where  cti_ctrycode.FIPS = cti_reports_usa.countryCode
and    datexform.date1 = cti_reports_usa.eventDate
and    cti_geotran_convert2.inValue = cti_reports_usa.geoCoord
```

```
union
select cti_reports_uk.recID, cti_reports_uk.srcID, cti_geotran_convert22.outValue, cti_reports_uk.cityName,
cti_reports_uk.countryName, airporticao.iata, cti_ctrycode2.ISO3166A3, datexform2.date2,
cti_reports_uk.eventTime, cti_reports_uk.eventType, cti_reports_uk.placeName,
cti_reports_uk.personName, (cti_reports_uk.height*unit_conv3.UnitFactor),
(cti_reports_uk.weight*unit_conv4.UnitFactor)
from   (select 'BNG-OGB7-X', inValue, 'MGRS-WGS84-X', outValue
        from   cti_geotran_convert2
        where  inCode='BNG-OGB7-X'
        and    outCode='MGRS-WGS84-X') cti_geotran_convert22,
       (select icao, iata, aptname
        from   airporticao) airporticao,
       (select name, FIPS, ISO3166A2, ISO3166A3, ISO3166N3
        from   cti_ctrycode) cti_ctrycode2,
       (select date1, 'European Style /', date2, 'American Style -'
        from   datexform
        where  format1='European Style /'
        and    format2='American Style -') datexform2,
       (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, personName,
height, weight, eventType, eventDate, eventTime, srcID
        from   cti_reports_uk) cti_reports_uk,
       (select 'ft', 'm', UnitFactor
        from   unit_conv
        where  FrmUnit='ft'
        and    ToUnit='m') unit_conv3,
       (select 'stone', 'kg', UnitFactor
        from   unit_conv
        where  FrmUnit='stone'
        and    ToUnit='kg') unit_conv4
where  cti_ctrycode2.ISO3166A2 = cti_reports_uk.countryCode
and    datexform2.date1 = cti_reports_uk.eventDate
and    cti_geotran_convert22.inValue = cti_reports_uk.geoCoord
and    airporticao.icao = cti_reports_uk.airport
```

```
union
select cti_reports_nato.recID, cti_reports_nato.srcID, cti_geotran_convert23.outValue,
cti_reports_nato.cityName, cti_reports_nato.countryName, airporticao2.iata, cti_ctrycode3.ISO3166A3,
datexform3.date2, cti_reports_nato.eventTime, cti_reports_nato.eventType, cti_reports_nato.placeName,
cti_reports_nato.personName, (cti_reports_nato.height*unit_conv5.UnitFactor), cti_reports_nato.weight
from   (select 'UTM-WGS84-X', inValue, 'MGRS-WGS84-X', outValue
        from   cti_geotran_convert2
        where  inCode='UTM-WGS84-X'
        and    outCode='MGRS-WGS84-X') cti_geotran_convert23,
       (select icao, iata, aptname
        from   airporticao) airporticao2,
       (select name, FIPS, ISO3166A2, ISO3166A3, ISO3166N3
        from   cti_ctrycode) cti_ctrycode3,
       (select date1, 'European Style .', date2, 'American Style -'
        from   datexform
        where  format1='European Style .'
        and    format2='American Style -') datexform3,
       (select 'cm', 'm', UnitFactor
        from   unit_conv
        where  FrmUnit='cm'
        and    ToUnit='m') unit_conv5,
       (select recID, countryName, countryCode, cityName, airport, placeName, geoCoord, personName,
height, weight, eventType, eventDate, eventTime, srcID
        from   cti_reports_nato) cti_reports_nato
where  airporticao2.icao = cti_reports_nato.airport
and    cti_ctrycode3.ISO3166N3 = cti_reports_nato.countryCode
and    datexform3.date1 = cti_reports_nato.eventDate
and    cti_geotran_convert23.inValue = cti_reports_nato.geoCoord
```

rec ID	src ID	geoCoord	city Name	country Name	air port	country Code	event Date	event Time	event Type	place Name	person Name	height	weight
101	src1	30UXC9272208003	London	United Kingdom	LHR	GBR	12-01-2004	0900	meeting	Fez Restaurant	Ahmet Khatib	1.778	77.1107029
102	src1	30UXC9225907985	London	United Kingdom	LHR	GBR	12-01-2004	1200	rents apt	Apt on Mall Road	Ahmet Khatib	1.778	77.1107029
103	src5	30UXC9240108300	London	United Kingdom	LHR	GBR	12-02-2004	1700	meeting	King's Mall Shopping Ctr	Ahmet Khatib	1.7526	72.5838510474
104	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	12-03-2004	1430	observed	Heathrow Airport	Akka Mohammed	1.776984	72.5838510474
105	src3	30UXC7692705325	London	United Kingdom	LHR	GBR	12-04-2004	2100	observed	Heathrow Tube Station	Al Pavlakkih	1.7526	77.0925592052
106	src2	33UWP9904341242	Vienna	Austria	VIE	AUT	12-04-2004	1100	plane departs	Schwechat Airport	Aleph Faruk	1.73	77
107	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	12-04-2004	1245	plane arrives	Heathrow Airport	Aleph Faruk	1.728216	77.0925592052
108	src3	30UXC7692705325	London	United Kingdom	LHR	GBR	12-04-2004	1330	observed	Heathrow Tube Station	Aleph Faruk	1.728216	77.0925592052
109	src4	32UQU0620758946	Munich	Germany	MUC	DEU	12-03-2004	1415	plane departs	Franz Josef Strauss/ Riem/Munich Intl Airport	Akka Mohammed	1.78	73
110	src4	32UQU0620758946	Munich	Germany	MUC	DEU	12-03-2004	1415	plane departs	Franz Josef Strauss/ Riem/Munich Intl Airport	Al Pavlakkih	1.75	77
111	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	12-04-2004	1630	plane arrives	Heathrow Airport	Akka Mohammed	1.776984	72.5838510474
112	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	12-04-2004	1630	plane arrives	Heathrow Airport	Al Pavlakkih	1.7526	77.0925592052
113	src3	30UXC7692705325	London	United Kingdom	LHR	GBR	12-04-2004	1710	observed	Heathrow Tube Station	Akka Mohammed	1.776984	72.5838510474
114	src3	30UXC7692705325	London	United Kingdom	LHR	GBR	12-04-2004	1710	observed	Heathrow Tube Station	Al Pavlakkih	1.7526	77.0925592052
115	src1	30UXC9263008370	London	United Kingdom	LHR	GBR	12-04-2004	1800	observed	Hammersmith Tube Station	Al Pavlakkih	1.7526	77.1107029
116	src2	33UWP9904341242	Vienna	Austria	VIE	AUT	12-04-2004	1000	plane departs	Schwechat Airport	Ali Abdullah	1.68	73
117	src3	30UXC7711904806	London	United Kingdom	LHR	GBR	12-04-2004	1245	plane arrives	Heathrow Airport	Aleph Faruk	1.728216	77.0925592052
118	src5	30UXC9263008370	London	United Kingdom	LHR	GBR	12-04-2004	2100	observed	Hammersmith Tube Station	Ali Hakem	1.776984	99.7631058578
119	src1	30UXC9272208003	London	United Kingdom	LHR	GBR	12-04-2004	2100	meeting	Fez Restaurant	Ahmet Khatib	1.778	77.1107029
120	Src1	30UXC9272208003	London	United Kingdom	LHR	GBR	12-04-2004	2100	meeting	Fez Restaurant	Aleph Faruk	1.7272	77.1107029

8. Conclusion

Context Mediation technology addresses the important problem of data interpretation and deals directly with the integration of heterogeneous contexts (i.e. data meaning) in a flexible, scalable and extensible environment. This approach makes it easier and more transparent for receivers (e.g., applications, sensors, users) to exploit distributed sources (e.g., databases, web, information repositories, sensors). The approach and associated tools significantly reduce the overhead and possible errors involved in the integration of multiple sources and simplifies maintenance in an environment of changing source and receiver context. Context mediation addressees a range of contextual issues including complex temporal, equational, scale, and units conflicts.

In the Context Mediation approach, a user query is expressed as if all sources were in the user's context. The abduction engine identifies semantic conflicts through the comparison of modifier values (i.e. declarative context). Resolution of conflicts takes place automatically. The mediation engine then generates a mediated query that reconciles semantic differences, if any, between all sources involved and the user. The query optimizer and executioner implements a capability aware and cost based distributed query optimization algorithm that takes advantage of parallel execution of sub-queries in multiple sources.

In this report, we have demonstrated the applicability and value of the context mediation approach in facilitating the effective and correct use of counter-terrorism intelligence information coming from diverse heterogeneous sources.

References

[1] Alatovic, T. (2002) "Capabilities Aware Planner/ Optimizer/Executioner for Context Interchange Project", MIT EECS Master's Thesis.

[2] Goh, C.H. (1997) Representing and Reasoning about Semantic Conflicts in Heterogeneous Information Systems, PhD Thesis, MIT Sloan School of Management

[3] Goh, C.H., Bressan, S., Madnick, S., Siegel, S. (1999) "Context Interchange: New Features and Formalisms for the Intelligent Integration of Information", ACM Transactions on Information Systems, 17(3), 270-293

[4] Lee, P.W. (2003) Metadata Representation and Management for Context Mediation, MIT EECS Master's Thesis.

[5] Liburd, S. (2003) Conceptual Mapping and Structural Conversions between COIN and RDF, technical report, MIT Sloan School of Management.

[6] Madnick, S.E. (1999) "Metadata Jones and the Tower of Babel: The Challenge of Large-Scale Semantic Heterogeneity", Proceedings of the 1999 IEEE Meta-Data Conference.

[7] Manola, F. Miller, E. (2003) "RDF Primer", W3C..

[8] Firat, A., Madnick, S., Siegel, M. (2000) "The Cameleon Web Wrapper Engine", Proceedings of the Workshop on Technologies for E-Services, September 14-15, Cairo, Egypt.

[9] Firat, A., Madnick, S., Grosof, B. (2002) Financial Information Integration in the Presence of Equational Ontological Conflicts, 12th WITS, December 14-15, Barce-lona, Spain.

[10] Moulton A., Siegel, M.D., Madnick, S.E. (2003) Potential for Cost Savings from Application of Context Mediation Technology to Problems of Standards Interoperability in the Financial Services Industry, internal re-port, MIT Sloan School of Management.

[11] Xian, X. (2003) Corporate Householding Knowledge Engineering and Processing us-ing Extended COIN, MIT EECS Master's Thesis.

[12] Zhu, H. Madnick, S., Siegel, M. (2002) "Global Comparison Aggregation Services", 1st Workshop on eBusiness, December 14-15, Barcelona, Spain.

[13] Chourci, N., Madnick, S. E., Moulton, A., Siegel, M. D., Zhu, H. (2004) "Information Integration for Counter Terrorism Activities: The Requirement for Context Mediation," IEEE Aerospace Conf 2004.

[14] National Research Council. (2002) Making the Nation Safer: The Role of Science and Technology in Countering Terrorism. http://www.nap.edu/html/stct/index.html

[15] National Geo-Spatial Intelligence Agency. http://earth-info.nga.mil/GandG/pubs.html

[16] Zhu, H. and Madnick, S.E. (2004) "Context Interchange as a Scalable Solution to Interoperating Amongst Heterogeneous Dynamic Services" Third Workshop on eBusiness, Washington, D.C.