7

Sharing statistical information

Summary. In the last chapter, we introduced a number of basic techniques for retrieving and integrating heterogeneous information sources. In this chapter, we report an application of some of these techniques in a project on the integration of European fishery statistics. We identify the special characteristics of statistical information and focus on the use of the Web Ontology Language for representing statistical information and for retrieving information based on a semantic description.

Statistics are indispensable for political decision making. Economic, demographic and environmental statistics are used for monitoring social and physical processes and for measuring policy effectiveness. National governments usually have organized statistical services in order to fulfill their demand for decision support. At supranational level, and even at national level, homogeneous statistics are often not available. So, for supranational economic research and policy evaluation, heterogeneous statistics from a variety of independent sources must be integrated. In integration of statistics, all general problems known from other areas of information integration occur, such as ontological and notational differences and differences in units of measurement and typology. In addition there are some specific problems in the integration of statistics. The first class of problems specific for statistics are differences in the population, e.g. differences in the threshold for inclusion of objects. For example, does a boat with engine power less than 20 hp count as a fishing vessel? A second class of problems are differences in reported statistics, e.g. sum vs. average. Further, there are classification differences, e.g. age classes bounded by 20, 35, 50 and 65 years vs. 15, 35 and 55 year; length vs. gross register tonnage as vessel size indicator; differences in nomenclature. In order to overcome these heterogeneities, we often need background information like the correlation between the membership in different classes. In order to find and compare such statistics with needed information, we need

to be able to formally describe the domain ontology underlying a statistic and the statistical information itself. For these purposes we need an ontology of statistical terms and a framework for describing, comparing and translating the domain ontologies of heterogeneous statistical tables. Fig. 7.1 shows an example of a statistical table that will be used in the remainder of this chapter.

Table 7.1. Summary of the German fleet's catch in 2000

Zone	Grosse Hoch-seefischerei	Ab 20m	10 bis 19,99m	Bis 9,99m	Gesamte kleine Hochsee und Kuestenfischerei	Gesamte Kutter und Hochseefischerei
EG	93.932,0	53.258,2	30.222,0	6.268,3	89.748,5	183.680,5
A	7.966,5	0,2	0,8	0,0	1,0	7.967,5
FAR	0,0	213,4	0,0	0,0	213,4	213,4
NF	2.995,3	0,0	0,0	0,0	0,0	2995,3
NFGD	1.924,5	0,0	0,0	0,0	0,0	1.924,5
GD	5.005,6	0,0	0,0	0,0	0,0	5.005,6
IS	0,0	659,6	0,0	0,0	659,6	659,5
EST				0,0	0,0	
LET				0,0	0,0	
LIT				0,0	0,0	
NN	2.564,1	0,0	0,0	0,0	0,0	2.564,1
NSP	2.206,0	0,0	0,0	0,0	0,0	2.206,0
MAU	0,0	0,0	0,0	0,0	0,0	0,0
Gesamt:	116.594,0	54.131,4	30.222,8	6.268,3	90.622,5	207.216,5

In this chapter, we first discuss the special nature of statistical information that has to be taken into account when trying to integrate and share it and present a core ontology os statistical information. We then introduce a framework for modelling statistical information using OWL for capturing the ontology of statistics and combining it with domain concepts as well as data items to be shared. We explain the different features of the representation using the example of European fishery statistics and show the benefits of this representation with respect to the retrieval of information using conjunctive queries.

7.1 The nature of statistical information

Before we can define a representation for statistical data, we first have to get a better understanding of the nature of the information we have to capture. For this purpose we adopt the abstract model of statistical data described in [Sundgren, 1995]. Following this model, we first have to distinguish statistical microdata and macrodata. The former refers to the actual observations that

have been made about single objects in the World (statistical units) and their properties at a certain point in time (e.g. the salary of a person in a certain month). It can be modelled as a list of quadruples: object, property, value, time point or interval. Each of these quadruples forms an elementary message. A number of messages form a statistical register of observations about some phenomenon of interest. A register is the basis for generating aggregated information about a population of objects, also called macrodata. Note that a register is intended to represent a population, but that it is not identical: the register may be a sample or some other incomplete selection. The generated macrodata are estimates of the actual values of population properties. At large the process of statistics involves the following activities: (1) identify the objects to be included in the register; (2) observe the objects and enter observed values into the register and (3) process register data to obtain estimates for the population or cross classifications. The first two activities result in the production of microdata. The third activity results in macrodata. Models for describing statistical information systems are given by [Catarci et al., 1998] and [De Giacomo and Naggar, 1996].

7.1.1 Statistical metadata

When we talk about statistical tables, we always refer to aggregated information. Therefore, a general model of macrodata is needed as a foundation for modelling these statistics. Because microdata are in most cases not available for end users, from the user's perspective the model should abstractly describe the table contents, in stead of the statistical information system that produced them. Such a general model has four components that will be discussed in the following.

The reference population

Macrodata always refers to the characteristics of a set of objects. This set of objects, called a population, is important in order to draw conclusions about the relevance for a specific question. Statistics are often used in order to compare two different populations without having to compare single objects. Further, correlation between the values of two properties can only be established if the statistics refer to the same population. The population of statistical macro-data is described by a set of criteria that hold for all objects in the population. These criteria include the type of objects under consideration (e.g. employed persons). Often the type criterion is combined with other criteria, in particular geographic constraints (employed persons in central Europe) or combinations of different type constraints (e.g. employed females).

Aggregation criteria

(Cross-classifications) In most cases statistics do not consider a population as a whole, but define additional aggregation criteria that split the underlying

population into a number of disjoint, exhaustive subgroups. The values for each of the subgroups are determined independently and can be compared in order to make assertions about the specific group. Aggregation criteria again can be very different, the only restriction is that they cross-classify the population. We find aggregation criteria related to the type of objects (male vs. female employees, age groups), the geographic location (inhabitants of different federal states) or time (months of a year). The aggregation criteria are especially important when the statistic is intended to be used to answer a particular question (e.g. are female employees discriminated with respect to their salary ?).

Aggregation operator

The next important aspect is the method used in order to aggregate the values of the observed property in the different subgroups. Its function is to abstract from the properties of individual objects. It serves as a means for normalizing and abstracting the observations contained in the microdata. This method can range from a simple count of the objects in a subgroup to complex aggregation functions. The concrete function depends on the nature of the observed property. Often, the values of a considered property is a numerical value. In this case the aggregation function can be defined by any mathematical formulas mapping a set of numbers onto a single one. Typical examples of aggregation functions beside the count are the sum, the average and the median of a set of values.

The time frame

Properties of objects often change over time. Therefore, it is important to consider the time frame in which the microdata a statistic is based on has been acquired. It is also relevant for comparing the properties of different populations on the basis of the same time frame or the same population in different periods of time. There are two different aspects in the definition of the time frame. The first is the beginning and the end of the observation period and the second is the frequency and the time points for which data has been acquired (once, monthly, yearly,...). Both aspects are relevant when trying to compare two statistics. In the case of different frequencies, the results of the statistic that is based on more frequent observations can still be aggregated to match the other given that the other aspects are the same.

7.1.2 A basic ontology of statistics

The general data model of statistical data is the data matrix, the rows representing objects of interest, the columns representing attributes (properties) of the objects. For microdata, the rows represent statistical units and the columns represent observed variables. For macrodata, the rows represent classes of statistical units and the columns represent estimators of population

properties. Statistical methods are generic. They map data matrices to data matrices. The semantics of the data matrices is in the meaning we give to the rows and the columns, and in the definition of the represented population. For reasoning about statistics, we need an ontology of statistical terms – referring to the generic properties of data matrices and statistical operators – and an ontology of the domain described by the statistics. The statistical ontology should provide the framework for relating statistical knowledge to the domain ontology by giving definitions of reference populations, their properties and cross-classifications (compare [Grossmann, 2002]). Statistical metadata literature emphasizes three main properties that describe statistical tables: the population represented by the table, the population characteristics represented by the data content and the variables used to cross-classify the population. Some models have an explicit notion of time ([Sundgren, 1995], [Grossmann, 2002]), while others ([Catarci et al., 1998]) rely on explicit modelling of time as a cross-classifying variable. Where temporal awareness is included in the model, it has two roles: (1) as a validity label of metadata definitions and (2) as a time-coverage label for the data. In some models there is a more or less formal definition of the classes used for cross-classifying the data, but the population is taken as primitive in most models: there is a slot for specifying a textual definition, but no formal definition of population constraints. For integration purposes a formal specification is necessary because we need to reason about populations and differences between them, in contrast with the statistical production process where the population is given. Denk and Froeschl [Denk and Froeschl, 2000] treat temporal as well as geographic coverage as a special variable category. They define a request template for a table to be mediated from heterogeneous macrodata sources, with clauses for specification of: the mediated source table, the estimator to report, geographical constraint, temporal constraint, cross classification, and additional constraints. The template does not explicitly specify the population or the type of statistical units. The definition of the population to report about is hidden in the constraints part of the request specification and is implicitly bounded by the available sources.

While the general aspects are assumed to be the same for any source of statistical information, the domain-specific aspects may be different. This corresponds to the basic distinction between ontology (a shared conceptualization of a specific subject matter) and context (a subjective view of a domain). In this section, we concentrate on those aspects of statistical information that are the same across different domains and define a basic ontology of statistical information. This ontology will provide the backbone for modelling statistical information in different contexts.

Statistical Units and Attributes

The basis of statistical information is the notion of a "statistical unit" which refers to an individual object in the domain of discourse. These objects

have certain "attributes" that provide input to the generation of aggregated information. The value of a specific attribute of a statistical unit is referred to as an "observation". Observations are further defined by the unit and the scale they are measured in. Both, unit and scale are defined in the particular context the statistic has to be interpreted in.

We can further distinguish between different types of attributes that demand a different treatment due to their conceptual nature. A basic distinction is between qualitative and quantitative attributes. Quantitative attributes often contain the information that is presented in an aggregated way by the statistics. Qualitative attributes are often used as a grouping criterion for statistical units. Specific types of qualitative attributes are classifications and spatial attributes further defined in the context of the statistics.

Classes and estimates

A fundamental property of statistics is that they do not provide information about individual objects, but abstracted information about groups of objects sharing some common property. In our basic ontology of statistics, such groups of objects are referred to as "classes". We distinguish interval classes and nominal classes. Mutually exclusive lists of classes, used for discriminating and grouping of statistical units, are called "classifications".

Classes can have a special role in statistical datasets, namely as a "reference population", the set of all statistical units that are described by the statistics. In a register a population is normally represented by a subset of statistical units – e.g. a random sample – whose attribute values have actually been observed. A register may contain special attributes for identification of the statistical units, that will never be included in statistical tables.

The actual numbers contained in a statistical table represent the result of applying a certain statistical "operator" to the values of one or more particular attribute of all members of the population. The particular attribute that is observed for a complete population or subclass of it is called a statistical indicator. The result of aggregating the observations is called an estimate. The connection between an estimate and a particular context is established via the definition of the classes involved and via the statistical indicator that is based on attributes of objects in the domain.

The ontology

Based on the terminology used in the statistical domain explained above, we formalized a basic ontology of statistics that is shown in Fig. 7.1. We start from the basic idea of a data source as a data matrix. Correspondingly, we describe information sources by the three elements of a data matrix: the statistical attributes it describes (the columns), the classification used

to aggregate information (the rows) and the observation it contains (the actual entries of the matrix). The corresponding classes of the ontology are connected to the classes of information sources using the relations *contains* for the observations, *based-on* for the classification and *describes* for the statistical attribute. Further, each information source refers to a class of objects that act as a population.

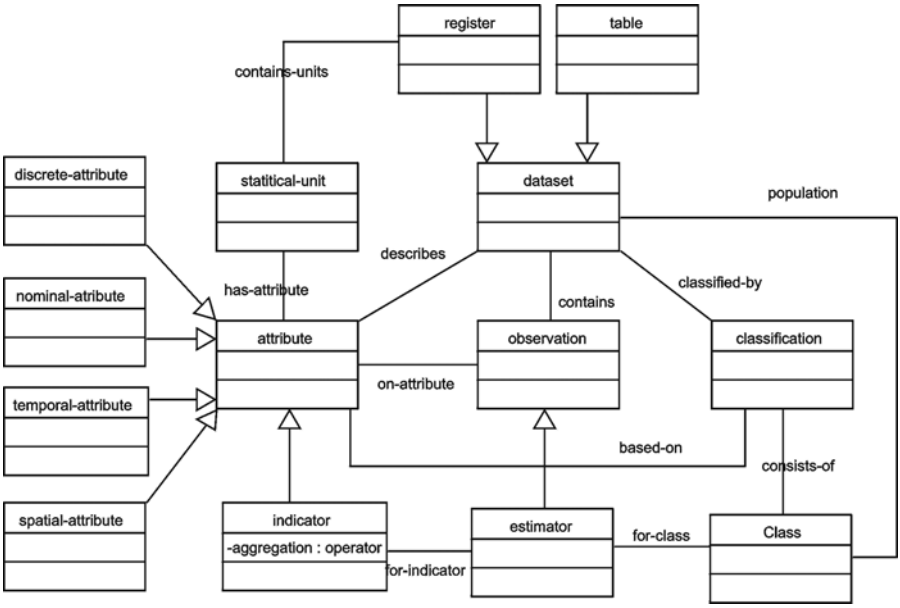

Fig. 7.1. A basic ontology of statistics

We distinguish registers and tables as special types of data sources. While registers contain information about individual objects (the classification consists of one class per object), tables contain aggregated information for classes of objects. Entries containing this aggregated information are special observations called estimators. They refer to a class of objects and describe a statistical indicator rather than a statistical attribute. Indicators are special kinds of attributes that include the notion of an aggregation operator (e.g. total income or average age).

This ontology is not meant to provide a complete conceptual model of all phenomena in statistics. It is rather a core model with the specific purpose to support the process of sharing statistical information. In the following section, we will show how the ontology can be used to provide a general structure for modelling and finding statistical data.

7.2 Modelling Statistics

The basic statistics ontology described above provides us with a domain-independent vocabulary for describing information in statistical tables. In particular, the different elements of a table correspond to the terms introduced. The columns of a table correspond to statistical indicators, the rows to classes and the actual numbers in the table are estimators of a certain indicator with respect to a class of statistical units. The union of all classes in a table is assumed to cover the underlying population. Further, the classes in the rows of a table are defined by a common observation of a certain attribute. A question that remains open is an appropriate structure to combine these elements into a description of a statistical table.

7.2.1 Statistics as views

A promising approach is to interpret the estimator in a statistical table as the answer to a query to a virtual database of observations about statistical units (compare [De Giacomo and Naggar, 1996]). The main problem we face in the integration of these query answers is that we do not have access to the underlying virtual database. Nevertheless, research in database systems has shown that under certain circumstances it is sufficient to compare queries in order to make assertions about the relation of two result sets [Calvanese et al., 1998a]. The ontology described in the last section provides us with a vocabulary for defining such queries. Using the terms defined in the ontology the most general description of an entry in a statistical table can be formulated as follows using an SQL-like syntax:

```
 SELECT
    indicator
FROM
    population
WHERE
    class = ...
```

An example of how this pattern describes different values would be: select the total catch of the German fishing fleet in 2000 where the size class is 20 to 50 meters and the fishing are is the Irish Sea. In this example, total catch is the indicator that is estimated. The population consists of all fishing vessels of German nationality that had been registered in the year 2000. The size class and the fishing area define classes that have been used to aggregate objects and estimate the value for the indicator.

We can immediately see that the initial format for describing estimators needs to be refined. In particular, the description of the population and the classes can be refined as they are defined using restrictions on the observation of a certain statistical attribute such as the nationality. The actual description would therefore rather look as follows:

```
SELECT total-catch
FROM
    nationality = German
    year = 2000
WHERE
    size-class = [10m, 20m]
    fishing-area = is
```

Another thing we notice is that the SELECT and the FROM part of the view are the same for an information source in most cases. In the unlikely case that a table contains more than one indicator, we can easily see it as being two information sources with the same population and cross-classification. In order to reduce the modelling effort necessary to describe a set of information sources, we also model the complete data source and explicitly connect the description of single estimators to the description of the table they are contained in. We further include information about the classification in the description of the source. A corresponding description has the following format:

```
Source1:
   SELECT indicator
   FROM
       population
   GROUP-BY
       class_1, class_2, ...

Estimator1:
    SELECT *
    FROM
        Source1
    Where
        class = class_n
```

This way of modelling assumes a number of constraints that must hold amongst the descriptions of information sources and their content. The classes named in the descriptions of the information sources are assumed to completely cross-classify the population; therefore, all classes must describe strict subsets of the population. Further, the estimator is indirectly typed by the select statement of the source description. Finally, the classes mentioned in the description of the estimators have to correspond with the classes mentioned in the grouping, and their descriptions have to be consistent with the cross-classification constraint mentioned above. We will come back to these constraints when describing how to formalize and reason about descriptions in the next section.

7.2.2 Connection with the domain

As mentioned above, a complete description of statistical information has to combine statistical and domain-specific terminology and background

knowledge. As the statistical part of the terminology has already been covered, we now turn our attention to domain-modelling aspects and their combination with the notions introduced above.

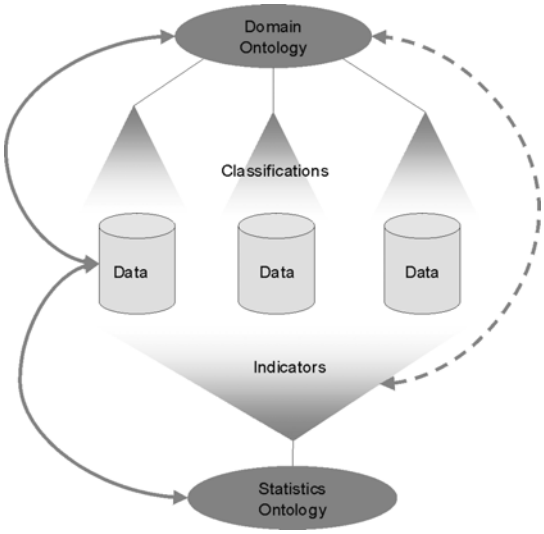

Fig. 7.2. Combined use of statistical and domain ontology

The general strategy for connecting the statistics ontology with the domain is by means of the view definitions above. In particular, the statistics ontology provides the general schema for describing data; the domain ontology is used to describe the concrete definitions of the population and classifications. Another point of connection is the definition of indicators as they mix domain vocabulary (e.g. catch) with general statistical terms (e.g. average or total) thereby connecting the ontologies. Fig. 7.2 sketches the combined use of statistical and domain ontologies in modelling statistics.

As indicated in Fig. 7.2, the domain ontology mainly provides the definitions of classes used in the different tables. Here we assume that a general domain ontology provides a shared vocabulary and the different classifications use terms from this general domain ontology. This enables us to use the techniques described in Chap. 6 to translate between different classification thus guaranteeing interoperability of data sources. Elements in the different data sources are linked to the domain specific classifications. At the same time they are linked to the general ontology of statistics (bold arrows). The link to the domain is mainly established through the notion of a statistical attribute which normally refers to a property of domain objects specified in

the domain ontology. In our model this connection is made by the definition of a hierarchy of indicators linking domain relations to concepts in the statistical model. In order to clarify the connection between the models we use the ontology of the fishery displayed in Fig. 7.3.

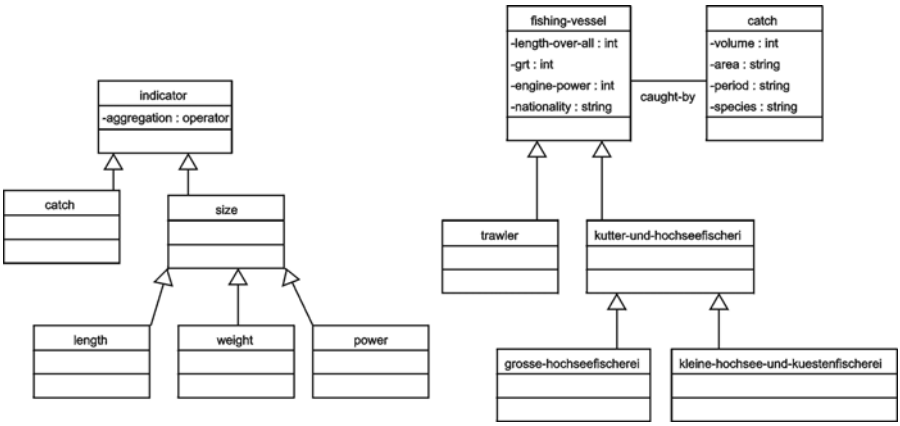

Fig. 7.3. Domain ontology of the fishery domain

In the ontology we see that the two central concepts we are concerned with in the fishery domain are fishing vessels and catch. The two concepts are connected by the *caught-by* relation. Further, each of the concepts has a number of attributes that describe the individual objects of the domain. In principle, each of these attributes can also act as a statistical variable and can therefore be the basis for defining classes and for indicators and corresponding estimates. In the example used to introduce the modelling notation for estimates, for example, the attribute length is used to define a class of vessels while the attribute volume is the basis for the statistical variable total catch.

In order to link domain relations to statistical indicators, we introduce a hierarchy of indicators rooted at the statistical concept *has-indicator*. Certain objects in the domain can be linked to certain indicators using the has-indicator relation. In the fishery domain, fishing vessels are the domain objects that are linked to indicators. As vessels are often aggregated based on size classes, size indicators are of central interest here. We can also define special size indicators such as length, power and weight. These specific indicators can now directly be linked to domain relations. We do this using mapping rules from domain relations to indicators. For our example, these mappings look as follows:

$$has - indicator(x, length) \leftarrow length - over - all(x, y)$$
$$has - indicator(x, power) \leftarrow engine - power(x, y)$$
$$has - indicator(x, weight) \leftarrow grt(x, y)$$

In order to be able to make use of these mapping rules, we have to design the description of data sources in such a way that we actually derive the existence of indicators in a source. In particular, this means that the rule bodies have to be derivable from the descriptions of tables and observations. At the same time, we have to make sure that the descriptions are expressive enough to capture the domain semantics implicitly contained in the information.

It turns out that for describing the data sources from a domain point of view, we can stay inside the metaphor of statistics as views by using conjunctive queries for describing object classes. More specifically, we describe the population of a data source as well as the classes used for aggregating information in terms of a query over the domain ontology that would return all members of the population or the class if we had access to a database with all objects in the domain. The corresponding definitions for the example table above are the following:

$$\begin{aligned} population(source1, X) \leftarrow\ & nationality(X, german), \\ & caught - by(Y, X), period(Y, 2000) \\ for - class(estimator1, X) \leftarrow\ & length - over - all(X, Y), Y > 10, Y < 20, \\ & caught - by(Z, X), area(Z, ire) \end{aligned}$$

This way of describing has several advantages. First of all conjunctive queries are a natural formalism for defining queries, as it is the underlying model for languages like SQL. Therefore, it fits naturally in our modelling syntax. The corresponding description of our example would look as follows:

```
Source1:
    SELECT total-catch of X
    FROM
        nationality(X,german),
        caught-by(Y,X), period(Y,2000)

Estimator1:
    SELECT X
    FROM
        Source1
    Where
        length-over-all,
        Y > 10, Y < 20,
        caught-by(Z,X),
        area(Z,is)
```

By restricting the predicates allowed in the queries to a domain ontology, we can provide guidance for modelling populations and classes. The corresponding ontology can also provide additional background knowledge about the intended meaning of classes and hidden dependencies like the one between domain relations and indicators described above. Finally, as we have seen in Sect. 6.3.1, we can translate conjunctive queries over ontologies into concept expressions and use existing description-logic reasoners to retrieve answers. In the following section we will describe how the description of complete tables can be translated into OWL. Based on this translation we can provide a number of reasoning services for information integration and retrieval that will be described afterwards.

7.3 Translation to Semantic Web languages

There are at least two reasons for translating the semantic descriptions of statistical data sources into Semantic Web languages. The first reason is the ability to publish these semantic descriptions on the Web. This enables other people to locate them and decide whether the information contained in a source is relevant for them. This does not only save the overhead of downloading and checking large amounts of data, it also supports the commercial exploitation of statistical data. Companies whose business is to sell statistical data can make all relevant information available without actually publishing data they want to sell. Potential customers of such companies get the possibility to better check whether an information source meets their information needs without having to buy it. The second reason is the availability of reasoning services for Semantic Web languages that we can use for retrieving and integrating statistical information based on their semantic description.

While an actual online version of the semantic description would be in the RDF-based version of the OWL syntax, we use the abstract syntax defined in [Patel-Schneider et al., 2002b] to illustrate the way our modelling framework can be encoded in OWL. This encoding basically consists of two parts. The first is the representation of the underlying ontologies. It can be done in a straightforward way as OWL is intended to capture this kind of knowledge. The second part is the representation of the statistical information itself. Here we use complex typing axioms that are rather untypical for Web ontologies in order to capture the underlying domain constructs. Both parts of the description will be explained in the following.

7.3.1 Ontologies

As mentioned before, the ontological knowledge used to model statistical information consists of two parts. The first one is the generic ontology of statistics

described in Sect. 7.1.2. The other part is an ontology of the domain that is used to give the information contained in a table a domain-related semantics.

Statistical ontology

The core notions of the statistical ontology can be described by a set of concepts representing datasets and their content (compare Fig. 7.1). We model these concepts as OWL classes:

```
Class(DataSet)
Class(Observation}
Class(StatisticalAttribute)
Class(Classification)
Class(Class)
```

The basic relations between these classes that link for example a dataset to its population are modelled as properties that link dataset objects to population objects. The latter fact is captured by restrictions on the range and domain of the properties. Further, we capture the fact that the population of a dataset is unique by declaring the corresponding property to be functional.

```
ObjectProperty(population
               domain(DataSet)
               range(Class)
               Functional)
```

The same is done for the other basic relation in Fig. 7.1. The fact that there are special cases of the general notion of datasources, observations and statistical attributes can be captured by the SubClassOf relation. The corresponding subclass relations shown in Fig. 7.1 are represented as follows.

```
SubClassOf(Table DataSet) SubClassOf(Register DataSet)
SubClassOf(DiscreteAttribute StatisticalAttribute)
SubClassOf(NominalAttribute StatisticalAttribute)
SubClassOf(StatisticalIndicator StatisticalAttribute)
SubClassOf(Estimator Observation)
```

In the concrete modelling of statistics, we are often interested in these subclasses as they represent the concrete cases we find in the data. The same holds for relations defined between the more concrete classes. In particular, the following two relations are used because they establish a connection to the domain ontology by relating estimators to indicators and classes of objects.

```
ObjectProperty(forIndicator
               domain(Estimator)
               range(StatisticalIndicator)
               Functional)

ObjectProperty(forClass
```

```
                domain(Estimator)
                range(Class)
                Functional)
```

Before showing how the connection is made, we first introduce the representation of the fishery domain ontology used in our example.

The fishery domain

The basic objects we talk about in the fishery domain are fishing vessels and their properties. In order to be able to do so we introduce the class of fishing vessels and datatype properties for capturing relevant properties of vessels such as length, engine power and gross registry tonnage, etc.

```
Class(FishingVessel)

ObjectProperty(nationality
                domain(FishingVessel)
                range(Country))

DatatypeProperty(lengthOverAll
                  domain(FishingVessel))

DatatypeProperty(enginePower
                  domain(FishingVessel))

DatatypeProperty(grt domain(FishingVessel))
```

The second central part is the information about the amount of fish caught by fishing vessels. This information cannot be represented in a single number (it depends for example on a period of time and the fishing area). We therefore introduce catch as a class which enables us to talk about catch object related to vessels and having certain properties, the volume of catch being amongst them.

```
ObjectProperty(caughtIn
                domain(Catch)
                range(fishingArea))

ObjectProperty(caughtBy
                domain(Catch)
                range(fishingVessel)
                InverseOf(caught))

DatatypeProperty(volume
                  domain(Catch))
```

Vessel classes

While classes are atomic objects from the point of view of the statistical ontology, they actually have a deeper meaning in terms of domain objects and their properties. The use of OWL enables us to make this meaning explicit in terms of class definitions. These definitions can also be used for semantic integration and filtering as described in Sects. 6.1.3 and 6.2.

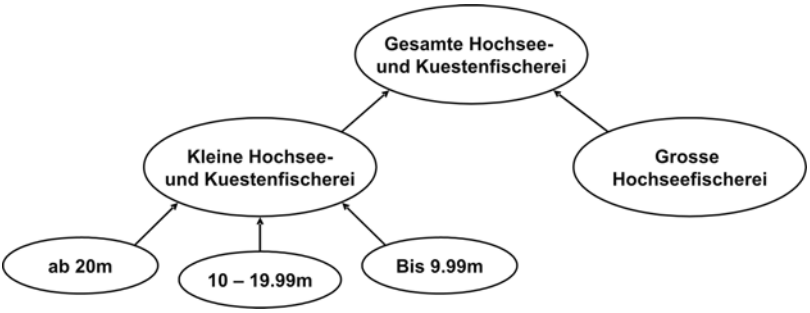

Fig. 7.4. Classification hierarchy of German fishing vessels

The German fishery statistics is a good example for the existence of domain-related semantics of object classes. German vessels are organized in a hierarchy of classes shown in Fig. 7.4. The names of the classes at the bottom of the hierarchy already indicate that the classification of vessels is based on the length. Using the *lengthOverAll* property of vessels defined in the domain we can formally express the intended meaning of the classes using property restrictions on the length property:

```
Datatype(<10)
Datatype(10-20)
Datatype(20-50)
Datatype(>50)

EquivalentClasses(grosseHochseeFischerei
                  restriction(lengthOverAll
                              someValuesFrom(>50)))

SubClassOf(restriction(lengthOverAll
                       someValuesFrom(20-50))
           kleineHochseeUndKuestenfischerei)

SubClassOf(restriction(lengthOverAll
                       someValuesFrom(10-20))
           kleineHochseeUndKuestenfischerei)
```

```
SubClassOf(restriction(lengthOverAll
                       someValuesFrom(<10))
           kleineHochseeUndKuestenfischerei)
```

Together with a straightforward encoding of the hierarchy from Fig. 7.4 in terms of subclass statements, we get a formal model of the classification of German fishing vessels.

Indicator classes

As described in the last section, domain relations such as the length of vessels are also used to link domain objects to statistical indicators. For this purpose, we encode the hierarchy of indicators shown in Fig. 7.3 in OWL and some concrete indicators as instances of the indicator classes in the hierarchy. Total catch for example would be an instance of the indicator class catch. Further, we can encode the mapping rules from domain properties to indicators using subclass axioms between general class expressions in OWL. The mapping rules mentioned above can be encoded as follows:

```
SubClassOf(restriction(grt someValuesFrom(integer))
           restriction(hasIndicator someValuesFrom Weight))
SubClassOf(restriction(lengthOverAll someValuesFrom(integer))
           restriction(hasIndicator someValuesFrom Length))
SubClassOf(restriction(enginePower someValuesFrom(integer))
           restriction(hasIndicator someValuesFrom Power))
```

Here, each property restriction represents a predicate in the mappings. The implication is simulated by the subclass statement itself. In the case of more complex rules, the OWL operators intersectionOf, unionOf and disjointFrom can be used to model conjunction, disjunction and negation in the rules.

7.3.2 Description of information

The descriptions introduced so far represent background knowledge that helps to interpret statistical information. An OWL-based representation of the actual description of statistical data sources in terms of views as introduced in the last section will be discussed in this section. In short, we model a statistical data source as a set of objects that belong to classes in the statistical as well as the domain ontology. The distinction between these two ontologies is necessary due to the dual nature of classes as atomic objects and as complex definitions. In the following, we first describe how data sources and their content are translated into objects. Afterwards, we discuss how these objects are linked to definitions in the ontologies.

Instance information

The most straightforward way of giving a semantic description of statistical information is in terms of instances of the statistics ontology presented above. We can derive the type of an object with respect to the statistical ontology from its position in the view definition that describes a table or an observation. The FROM clause of a view definition contains information about population objects. We directly encode them as instances of the Class concept, which is the range of the population property. We define three example classes representing fishing vessels of the German fleet in 1998 and 2000 as well as of the Danish fleet in 2000.

```
Instance(germanFleet2000 type(Class))
Instance(germanFleet1998 type(Class))
Instance(danishFleet2000 type(Class))
```

In the same way, we introduce instances for the other parts of the description of an information source and represent concrete sources as an instance that is related to these objects by the corresponding relations. In particular, we look for definitions of observations contained in a table, introduce objects for each observation found and link them to the names of the class of objects they describe. The following definition corresponds to the example view definition given on page 150.

```
Instance(germanCatch2000 type(Table)
         value(population germanFleet2000)
         value(describes TotalCatch)
         value(contains sumG1))

Instance(sumG1 value(for-class '10-20m')
               value(for-class 'is')
```

This basic way of encoding view definitions in OWL already helps to share statistical information, because the reference to the statistical ontology provides a common vocabulary for heterogeneous information. This shared vocabulary can be used to query certain kinds of information across information sources. We could for example ask for the populations of all registers available.

Typing information

We have argued above that an important part of the semantics of information is encoded in the definition of the population and the classification of a source. We chose to capture these definitions by conjunctive queries over the domain ontology. One of the rationales for choosing this kind of representation was that they can be translated into OWL class definitions [Horrocks and Tessaris, 2000]. When translating view definitions to OWL, the corresponding class objects are modelled as instances of the resulting class expression. This, however, is not done directly but by means of declaring the

corresponding data-source objects to be members of the class of things that have a population of a certain type, where this type is defined by the translation of the conjunctive query defining it. The following definition corresponds to the refined view definition shown on page 154.

```
Instance(germanCatch2000 type(
         intersectionOf(
         restriction(population allValuesFrom(
                restriction(nationality
                    value(Germany))))
         restriction(caught allValuesFrom(
                restriction(period
                    value(2000)))))))
```

The indirect description allows us to explicitly state that the observation is about objects in the intersection of the two classes:

```
Instance(sumG1 type(
         intersectionOf(
         restriction(for-class allValuesFrom(
                restriction(lengthOverAll
                    someValuesFrom(10-20))))
         restriction(for-class allValuesFrom(
                restriction(caught allValuesFrom(
                    restriction(area
                        value('is'
                        )))))))))
```

The best argument for an indirect description of classes is the possibility to generalize from the description of individual observations in the table. This is necessary for very large data sources. Typical examples in the fishery domain are fleet registers that contain thousands of entries each containing the same information about different objects in the domain. Instead of introducing an observation object for each of these entries, we can use the indirect description of the population represented by the register to provide information about the aspects represented in the data. A corresponding description of the register containing data about the German fleet in the year 2000 is shown below:

```
Instance(germanRegister1998 type(
        intersectionOf(
        restriction(population allValuesFrom(
            restriction(nationality value(Germany))))
        restriction(contains someValuesFrom(
            restriction(for-class someValuesFrom(
                intersectionOf(
                    restriction(lengthOverAll
                                someValuesFrom(integer))
                    restriction(enginePower
                                someValuesFrom(integer))
```

```
                    restriction(grt
                                someValuesFrom(integer))
))))))))
```

The definition states that the corresponding data-source object belongs to the class of objects that have a population of a certain type (the same as in the example above) and that it contains some information about the length, the power, and the weight of the objects it represents. Note the compactness of the representation as compared to an explicit modelling of thousands of register entries. Depending on the requirements on the retrieval of information, this kind of indirect description of the content of an information source can be used for all data sources if there is no need to retrieve individual entries in a table.

7.4 Retrieving statistical information

The logical interpretation of view definitions allows us to reason about available information on a conceptual level. In particular, we can use the logical model to check whether a piece of information matches our information needs and to retrieve all available information that matches our requirements. In principle, the encoding above allows us to answer any conjunctive query that uses the vocabulary defined by the statistical and the domain ontologies. In the following we discuss some typical kinds of queries users often want to ask about information.

Classes of objects

When asking about statistical information, the user always has a class of objects in mind that is described by the information. In the fishery domain, these are almost always classes of fishing vessels that fulfill certain requirements that act as the population of an information source. As we encoded populations explicitly as instances of the statistical ontology, we can retrieve populations present in the information by asking for vessels with certain properties. We could for example ask for object classes that describe German vessels using the following query:

$$Q(X) \leftarrow FishingVessel(X), nationality(X, Germany) \tag{7.1}$$

This query will return a set of objects representing different populations that underly information sources known to the system. The result will be a list of populations that consist of German fishing vessels in different years. We assume that the user is interested in information from the year 2000. As the names of objects returned do not necessarily provide some information about the year, we explicitly have to ask for classes of vessels that are relevant to the catch in the year 2000. This can be done by the following query:

$$Q(X) \leftarrow FishingVessel(X), caught(X, Y), period(Y, 2000) \tag{7.2}$$

For this query, the result will be classes of fishing vessels from different countries that are related to the catch in the year 2000. If we combine these two queries, we get the German fleet in the year 2000 as an example.

Data sources

Once we have retrieved a class of objects, we can use the name of this class in queries in order to find out more about information related to that class of objects. We can for example ask for registers that contain information about the members of this class using the following query:

$$Q(X) \leftarrow register(X), population(X, german - fleet - 2000) \tag{7.3}$$

Normally, a user is not interested in any kind of information about a population, but in a specific aspect of that population in terms of a statistical indicator. This requirement can easily be formulated by asking for data sources that contain observations for a specific indicator using the query below:

$$Q(X) \leftarrow contains(X, Y), for - indicator(Y, total - catch) \tag{7.4}$$

Directly referring to a specific indicator like the total catch might sometimes be too restrictive, because we can also derive that information from the average catch if we know the number of vessels. Using the indicator hierarchy, we can ask for data sources that contain information about certain types of indicators. We might for example be interested in some indicator for the capacity of vessels. Specific instances of this general aspect are length, engine power or gross registry tons. The following query will return all those data sources that contain information on one of these aspects:

$$\begin{aligned} Q(X) \leftarrow\ & contains(X, Y), for - unit(Y, Z), \\ & has - indicator(Y, Z), capacity - indicator(Z) \end{aligned} \tag{7.5}$$

The possibility to ask for a wider range of aspects leaves space for an interaction with a human expert knowing about ways to combine and process information in order to get the required result.

Observations

Depending on the level of detail we chose in modelling the information, we can even ask more specific questions concerning individual values in tables.

Retrieving specific entries in a table can be done based on explicit relations of entries to other objects in the semantic model or based on the class of objects it describes. An example for retrieving information based on explicit relations is the following:

$$Q(X,Y) \leftarrow contains(german-catch-2000, X), for-class(X,Y) \quad (7.6)$$

The query returns pairs containing observations found in the table german-catch-2000 and the vessel class the observation is assigned to. In this way, we can get more detailed information about the content of a data source. The real benefit of the semantic description, however, only becomes clear when asking for specific information about a certain class of objects based on an abstract description of that class. The following query is an example of a simple case of looking for statistics based on a description of a set of objects.

$$Q(X) \leftarrow for-class(X,Y), kleineHochseeUndKuestenFischerei(Y) \quad (7.7)$$

The query asks for all observations made about the vessel class "kleine Hochsee- und Kuestenfischerei" (compare Fig. 7.4). The answer to this query will not only contain the observations that are directly made about this class, but also observations about subclasses of this class, in our example all vessels with a length of less than 50 meters, independent of the name of the class they are explicitly assigned to.

7.5 Conclusions

Literature study reveals that the results of intelligent information integration do not cover the specific problems of statistical information integration. An exception is [Klinkert et al., 2000], in which an overall model was proposed that is dedicated to the statistical integration process used to support the European Common Fisheries Policy. That model does not use either a generic ontology of statistics or generic models of statistical methods. Furthermore, the problem of possible classification differences was solved in an ad hoc manner for specific data sources.

Statistical techniques that are an obvious source of inspiration are not generally applicable. This is caused by the inaccessibility of data and by lack of domain specific statistical models. Formalization of human expert knowledge did not solve these problems. However, the acquired heuristic knowledge did enable the formalization and implementation of a model that can be used to explicitly represent the domain-related semantics of statistical information.

The model supports selection of datasets based on an abstract description of their expected content and has been found useful for selecting primary sources, weight matrices and registers. The structure of the model is set up in such a way that it allows easy extension with other methods such as the following that are not supported in the current system:

- An explicit model of space: we have to able to define the geographic region in which the population and the classes of objects used for aggregation are located. Further, we need to be able to analyze and reason about the relation between the locations of objects in two different statistics.
- An explicit model of time: we have to able to define the time frame in which information about a population has been acquired. We need to be able to analyze and reason about the relation between the time frames of two data sources.
- An explicit model of statistical operators: we need the possibility to describe a variety of statistical operators that might occur in a statistical table. We need the possibility to identify and define possible transformations between values that are the result of these operators and the background information needed for the transformation.

While the last point is currently mainly unexplored, work on extending semantic descriptions of information with explicit notions of space and time exist. We address spatial aspects of information sharing in the following chapter.

Further reading

Sundgren [Sundgren, 1995] proposed a unifying model for modelling statistical metadata that is based on the different components considered here. The use of description logics for formalizing descriptions of statistical information is described in [Catarci et al., 1998] and [De Giacomo and Naggar, 1996], who also introduce the use of views for modelling statistics. Earlier work on a knowledge-based approach to integrating fishery statistics is reported in [Klinkert et al., 2000] and [Jonker and Verwaart, 2003].