Managing Information Quality in e-Science: A Case Study in Proteomics

Paolo Missier*, Alun Preece†, Suzanne Embury*, Binling Jin†, Mark Greenwood*, David Stead‡, and Al Brown‡

*University of Manchester, School of Computer Science, Manchester, UK
†University of Aberdeen, Computing Science, Aberdeen, UK
‡University of Aberdeen, Molecular and Cell Biology, Aberdeen, UK
info@qurator.org, http://www.qurator.org

Abstract. We describe a new approach to managing information quality (IQ) in an e-Science context, by allowing scientists to define the quality characteristics that are of importance in their particular domain. These preferences are specified and classified in relation to a formal IQ ontology, intended to support the discovery and reuse of scientists' quality descriptors and metrics. In this paper, we present a motivating scenario from the biological sub-domain of proteomics, and use it to illustrate how the generic quality model we have developed can be expanded incrementally without making unreasonable demands on the domain expert who maintains it.

1 Introduction

A key element of e-Science is the development of a stable environment for the conduct of information-intensive forms of science. Increasingly, scientists expect to make use of information produced by other labs and projects in validating and interpreting their own data, while funding bodies expect the results generated by their projects to have greater longevity and wider usefulness. In this context, information does not merely document the state of the art in a domain, it also becomes a fundamental resource in the discovery of new knowledge. Hence the increasingly stringent requirements by funding bodies and publishers that scientists place their experimental data in the public domain in forms that are amenable to analysis by software tools as well as by humans.

At present, a variety of obstacles prevent the full realisation of this e-Science vision, not least of which are those caused by the inevitable variations in the quality of the information being shared [3]. It is tempting to view this as a problem for data producers, and to concentrate on defining standards and procedures for data capture (such as those defined by the MGED consortium for the capture and recording of information about microarray experiments [4]). While such standards are important and worthwhile, they cannot provide a complete solution. They can do little to address the quality of the volumes of legacy data that have so far been amassed. Moreover, like other forms of quality, information quality (IQ) is typically a function of the requirements of the information consumer rather than its producer. A scientist searching for information relating to a drug that is about to be used in patient trials will have more stringent requirements than one searching for examples to be used in a textbook, for example. Similarly,

one scientist might think of “accuracy” in terms of some calculated experimental error, while another might define it as a function of the equipment that captured the data.

What is required, therefore, is some means by which we can determine the quality of a specific data set relative to the needs of a specific user. For example, data sets that are incomplete or inaccurate can still be used to good effect by those who are aware of these deficiencies and can work around them. The viability of this approach depends on the ability to elicit and manage detailed specifications of the IQ requirements of individual users (or, at best, communities of like-minded individuals). The task of specifying new forms of quality preference should not be too onerous on users or those managing the information environment. IQ preferences should ideally be expressed in a formal language so that the definitions are machine-manipulable, both to allow (semi-)automatic determination of the quality of a data set, and to facilitate browsing and searching of the quality model.

The Qurator project[1] aims to provide the software infrastructure needed to support this form of domain-specific IQ management, focussing specifically on two domains of post-genomic biology: proteomics and transcriptomics [5]. We envision an e-Science environment in which a new user (scientist) can use IQ tools to discover potentially-useful IQ preferences for adaptation and reuse, and which allows new customised preferences to be defined without involving an expensive knowledge capture exercise.

The existing IQ literature offers useful starting points to meet these goals, by providing a common terminology for describing quality properties, or *dimensions* [11, 9]. However, it falls short of providing principled solutions to the problem of expressing quality requirements in a formal way, let alone to the problem of expressing complex quality-oriented views of data. In this paper, we describe a knowledge-intensive approach to modelling both the quality and application domains, which may serve a foundation to address these problems. We present the ontological model of IQ that forms the heart of our approach (Section 3), and show how the ability to reason over the model allows it to be self-managing under the addition of new quality preferences (Section 4).

The ontology is implemented in OWL and makes use of OWL-DL reasoning features[2]. Although we do not add any new theoretical elements to the Semantic Web framework, its application to this problem is, to the best of our knowledge, novel. This project is still in its early stages, and validation of the ideas presented here is in progress with the collaboration of the Aberdeen Proteomics Facility, at the University of Aberdeen, UK. Tool support for exploiting the ontology is in the planning stage.

2 Background on protein identification

To motivate the ontology presented in this paper, we present a scenario from the area of proteomics that illustrates the kinds of quality preference which arise from the domain-specific approach we are investigating. Proteomics is the study of the set of proteins that are expressed under particular conditions within organisms, tissues or cells. One experimental approach that is widely used to gain information about the large-scale expression of proteins involves extracting the proteins from a biological sample, then sep-

[1] Qurator is funded by the EPSRC Fundamental Computer Science for e-Science Programme.

[2] http::://www.w3.org/TR/owl-guide/

arating them by a technique known as 2-dimensional gel electrophoresis (2DE). With this technique, the proteins are separated into a 2D matrix, where they are distinguished by net charge and molecular size. These two separating factors are typically enough to differentiate each protein in the sample, so that each spot on the gel contains just one kind of protein. The spots can be examined individually and the amount of protein in each can be estimated after staining and densitometric scanning.

In a typical proteomic experiment, several different samples are subjected to the procedure outlined above and the resulting 2DE maps are compared. This allows the biologist to compare the expression rates of various proteins under contrasting conditions, for example to examine the different expression rates between a healthy tissue sample and a diseased one. By comparing the gel images that are produced from the samples, the biologist can hypothesise that the changes in protein expression thus highlighted may be a significant cause or result of the biological phenomenon under study.

Before such a hypothesis can be fully stated, it is necessary to identify the proteins that are present in the spots that indicate varied expression levels. This task is routinely performed using the technique of peptide mass fingerprinting (PMF). In PMF, the protein within the gel spot is first digested with an enzyme that cleaves the chain of the protein at certain predictable sites. The fragments of protein that result (called *peptides*) are extracted and their masses are measured using mass spectrometry. The list of peptide masses is then compared against theoretical peptide mass lists, derived by simulating the process of digestion on protein sequences extracted from a protein database (e.g. NCBInr[3]). Since, for various reasons, it is unlikely that an exact match will be found, the protein identification search engines typically return a list of potential protein matches, ranked in order of search score. Different search engines calculate these scores in different ways, so their results are not directly comparable. Furthermore, although some search engines (e.g. Mascot[4]) attempt to estimate the probability that a match is valid, others (e.g. MS-Fit[5]) do not, and it may be difficult for the experimenter to decide whether a particular protein identification is acceptable or not.

It would be useful for biologists seeking to interpret the results of proteomic experiments to be able to assess the credibility of a protein identification result by comparing readily accessible metrics for a list of protein matches. There are three metrics that can be used for this purpose, and which are independent of the search engine used:

- Hit ratio: the number of peptide masses matched, divided by the total number of peptide masses submitted to the search. Ideally, most of the masses should be accounted for by the protein identified, but because of additional peaks in the mass spectrum (originating from the presence of other proteins, for example) the hit ratio is unlikely to reach unity.
- Excess of limit-digested peptides: calculated by subtracting the number of matched peptides containing a missed cleavage site from the number of peptides with no missed cleavages. Ideally, a complete (limit) digest will have been achieved during PMF, in which case the number of missed cleavage sites would be zero. However, in practice a small number of missed cleavages are to be expected.

[3] http://www.ncbi.nih.gov/BLAST/
[4] http://www.matrixscience.com/
[5] http://prospector.ucsf.edu/

– Sequence coverage: the number of amino acids contained within the set of matched peptides, expressed as a percentage of the total number of amino acids recorded for the protein in the database. The higher the coverage, the greater the confidence in the match, but limitations of the experimental technique mean that full coverage is never achieved. It is also necessary to consider the size of the protein. A lower coverage of, say, 15% may be satisfactory for a large protein where many other peptides have been successfully matched. A similar coverage for a smaller protein, on the other hand, would be indicative of a poor match. Therefore, care must be exercised when interpreting the value of this metric.

These three metrics can be combined in a logical expression that allows us to classify protein matches as being either acceptable or unacceptable. A software tool can then be envisaged that allows the user to set acceptance criteria for each metric independently and to see the effect in real time of altering any or all of the threshold values on the acceptability of the data set.

While we use this as a simple example, a more general approach for the creation of quality preferences is described in the next section. The choice of these metrics also represents a simplification. In protemics, many more variables can be used to formulate statements regarding the quality of experimental results. For in-depth reviews of the field and of the variables involved, please see [6, 7, 1].

3 Ontology-based modelling of information quality

Although we are focussing on two specific application domains within the Qurator project (i.e. proteomics and transcriptomics), our ultimate goal is to produce a model of IQ that can be instantiated to produce domain-specific IQ models for a wide variety of application areas. In order to achieve this, it was necessary to find some over-arching organisational structure that would give meaning to the domain-specific terms and allow for comparison and analysis of quality preferences provided by multiple users. For this purpose, we have adapted generic IQ concepts that have been in use within the IQ community for a long time, and which are grounded in the wealth of existing literature on this topic [3, 8, 10, 9]. These concepts, such as accuracy, completeness and currency, give useful placeholders for common IQ concerns but they are not sufficiently well defined to be directly applicable to real applications. Instead, the user (or group of users) will wish to talk about specific properties relating to the domain of interest. Rather than speaking of accuracy, she will talk of equipment tolerances or scores resulting from error models, for example.

The Qurator quality ontology must therefore bridge the gap between these generic quality concepts (i.e. quality properties) and concepts from the users' own domain. Figure 1 shows a fragment of the OWL ontology we have created for the proteomics scenario described in Section 2, and which we will use to illustrate how this bridging is achieved[6]. Here, specific domain knowledge coming from the biologists and bioinformaticians involved in the project has been imported into the Qurator ontology from the

[6] No standard notation currently exists for expressing the logical features of the OWL language. We use a graphical notation in which ovals represent classes, rectangles represent individuals and lines represent object properties [2]. We show user-defined individuals and properties us-

*my*Grid project data ontology [12], which describes basic biological concepts as well as a number of biological databases and data analysis tools. In Figure 1, the `ApplicationDomain` classes represent domain data concepts that have either been lifted from *my*Grid, or added to it. We have then extended the model with terms of specific interest to proteomics specialists.

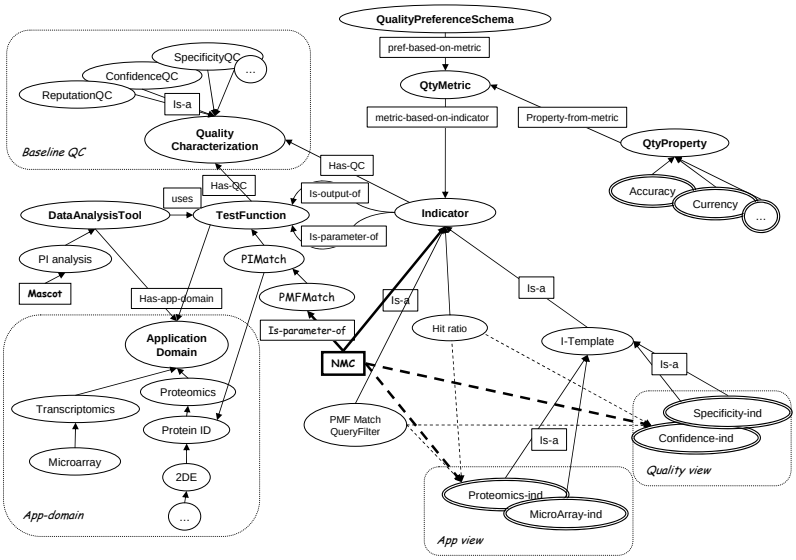

Fig. 1. Fragment of the IQ ontology.

We will use the following simple but realistic proteomics scenario to explain how the ontology supports the definition of new domain-specific quality preferences with the help of the generic quality terms. Suppose that a biologist wishes to rank a set of protein identification experiments performed using 2DE technology. The ranking is based on the scores obtained by matching PMFs against the NCBInr database, using the Mascot analysis tool. This score (the `QualityMetric`) is itself based on some function of the hit ratio and the number of possible missed cleavages (`NMC`) found during the match.

The user's first task is to create a `QualityPreferenceSchema`, which will be used to rank the experimental results. Although the ontology fragment shows only a single generic concept for preference schemas, in practice one would expect a range of more specific schemas to be defined (to partition experiments into classes, for example, or to filter them based on a threshold for the associated quality metric).

The next step is to define the `QualityMetric` itself, which must be based on one or more `Indicators`. An `Indicator` is some value that can be provided by the environment, either directly by retrieval from some persistent store or metadata

ing thick lines, and represent subsumption properties or individuals' classifications obtained through reasoning using dotted lines.

repository, or by computation. In our example, the two indicators are `HitRatio` and `NMC`. We will assume that the former is already present in the domain-specific part of the ontology, but that the user must add the `NMC` indicator to the model. In practice, it is common in e-Science for indicators to be associated with particular data analysis tools, as modelled by the `TestFunction` class. For example, `HitRatio` is part of the output produced by an analysis program which performs protein identification matches against a protein database. There may be many such programs, and the class `PIMatch` represents their general form in the ontology.

Next, we associate the indicators with the generic quality concepts. In the ontology, the quality domain is divided into two levels, which together support self-management of the ontology. The lower level is represented by the root class `QualityCharacterization`, or "QC" for short, which is extended with a small and fairly stable collection of key concepts describing information quality, such as `ReputationQC`, `ConfidenceQC` and `CurrencyQC`. These classes are used to characterize the quality of domain concepts that are part of the ontology. In particular, an `Indicator` is defined as any data entity that can be quality-characterized (through the `has-QC` property). Some indicators are domain-independent. For example, time stamps on data are commonly used to assess currency of information, and would therefore be associated to `CurrencyQC`. Others, such as `HitRatio`, are completely domain-specific.

The default associations shown in the model reflect the intended meaning of the quality terms, and have been introduced based on domain experts' recommendations. For instance, a `ConfidenceQC` indicator provides users with a level of confidence in the outcome of an experiment (as is the case for `HitRatio`, for instance) while a `ReputationQC` can be associated to indicators that scientists use to assess the overall reputability of an experiment outcome – these may include the laboratory that performed the experiment, and the standing of the associated journal publications. As the QC concepts form a primitive collection of ontology terms, they also act as the axiomatic base for the quality domain. Qurator makes an effort to ensure that domain experts use these terms consistently when introducing new domain-specific concepts.

The second method of encoding quality into the Qurator ontology is a higher level model of the more traditional data quality terminology found in the literature. These concepts are rooted at the `QtyProperty` class. The mapping between the two levels of quality concepts represents additional knowledge regarding quality, and provides for added flexibility in the specification of the semantics of the terms. For instance, we might define *Accuracy* (i.e. the property that describes how closely a data entity reflects the actual state of the real world entity that it stands for) in terms of *confidence* and *specificity*. The way to read this association is as follows:

> "a quality metric that is based on confidence or on specificity indicators, expresses the intention of the expert to capture accuracy properties of the underlying data."

Of course, there are many who would disagree with this as a definition of accuracy. The key point here is not the exact form of the definition, but the fact that there is a principled way to establish the logical associations between the users' operational definitions of quality, implemented using indicators and metrics, and a shared conceptualization of data quality.

4 A self-managing quality model through DL reasoning

The combination of domain knowledge and generic quality concepts has proved to be surprisingly powerful when combined with the kinds of reasoning facility offered by standard description logic ontologies. As the ontology is implemented in OWL-DL, we have benefited from its ability to provide consistency checks and entailment in supporting additions to and modifications of the ontology. In particular, the inferencing capabilities provided by description logics allow newly introduced concepts, such as indicators, to be classified against the quality model automatically, and therefore to become available to other scientists for reuse.

The principal mechanism that underlies this self-classifying ability is the set of QCs. We have already discussed how indicators can be classified relative to the current set of QCs. Test functions may also be treated in this way, and QCs can be automatically propagated from them to the indicators that act as their parameters or results. For instance, we can formalize the (sufficient) condition that an indicator is a `Confidence-ind` if it is a parameter of any `TestFunction` whose QC includes `ConfidenceQC`. In OWL DL, this can be written as:

(1) $\texttt{Confidence-ind} \equiv \exists\ \texttt{is-parameter-of}\ .\ (\exists\ \texttt{has-QC}\ \texttt{ConfidenceQC})$

Note that the `has-QC` property is many-to-many, so the existential quantifier indicates that *at least* one of the indicator's QCs must be a `ConfidenceQC`.

To see this in action, suppose that when the new user-defined indicator `NMC` is introduced, the only information the user can provide about it is that it is a parameter to a matching algorithm for PMFs. This algorithm is already known to the ontology as the concept called `PMFMatch`, which has a `ConfidenceQC`:

(2) $\texttt{NMC} \in \texttt{Indicator} \sqcap (\exists\ \texttt{is-parameter-of}\ .\ \texttt{PMFMatch})$

(3) $\texttt{PMFMatch} \sqsubseteq \exists\ \texttt{has-QC}\ .\ \texttt{ConfidenceQC}$

Assertion (2), here, defines `NMC` as an individual whose class is the domain of the `is-parameter-of` property, with a range of the class `PMFMatch` (this is called an *anonymous class*). Assertion (3) states that `PMFMatch` is a sub-class of any anonymous class that is a domain of the `has-QC` property, with range `ConfidenceQC` (a necessary condition). By applying standard DL reasoning [7] to these three assertions, we infer that `NMC` is a member of the `Confidence-ind` class (shown as a thick dashed line in Figure 1):

(4) $\texttt{NMC} \in \texttt{Confidence-ind}$.

Note that similar entailments are performed on the built-in indicators (thin dashed lines in the figure). Following this approach, it is possible to classify indicators with respect to both the quality and the application domains; the application domain hierarchy is

[7] Our implementation makes use of the RACER DL reasoner (`http://www.racer-systems.com/`).

rooted at `ApplicationDomain` in Figure 1. The role of the classes under the `I-template` class is to provide both a *quality view* and an *application view* of the indicators. Once they have been populated through reasoning, such views provide a basis for a variety of user queries regarding available indicators, including the user-defined ones. For example, they make it possible to query the ontology to discover what indicators exist that are suitable for a specific quality purpose (e.g. "give me all confidence indicators") so that users can browse the currently available resources before they go to the trouble of creating their own from scratch.

4.1 A detailed scenario

We can now illustrate how the ontology supports the user scenario introduced earlier. The following assertions, stated informally, capture the user's intuitions regarding quality preferences for his data collection:

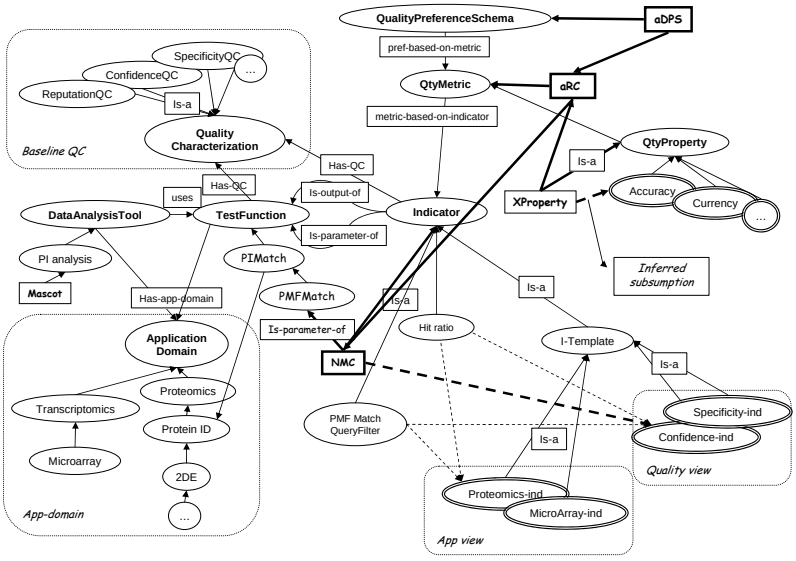

Fig. 2. Entailments for the user scenario (dotted lines)

1. a new indicator, `NMC`, must be introduced;
2. `NMC` is used as a parameter of any PMF matching algorithm (i.e. it does not need to be associated with a specific algorithm instance at this stage);
3. a ranking criterion `aRC` is introduced, as a function of `NMC` and the pre-existing indicator `HitRatio`;
4. a particular quality preference schema, `aDPS`, is defined;
5. it is stated that `aDPS` uses the `aRC` metric;
6. it is stated that `aDPS` applies to any proteomic data collection.

The effects of these assertions are shown in Figure 2 by rectangles (individuals) and thick solid lines (properties).

In addition to this domain-specific information supplied by the user, we also assume the existence of assertions within the ontology stating that a `Confidence-ind` is either an indicator whose set of QCs includes a `ConfidenceQC`, or that it is either the parameter or the output of a test function whose QC includes `ConfidenceQC`. Note that this definition generalizes assertion (1) to capture the case in which a direct assertion regarding the QC properties of an indicator is made. Finally, `Accuracy` is assumed to be defined in terms of quality metrics and QC-indicators as follows:

> Accuracy is the quality property for which there exists a metric qm that is based on a set of indicators, at least one of which is either a `Confidence-ind` or a `Specificity-ind`.

The DL expressions corresponding to these informal assertions are presented in the Appendix. Based on these expressions, Qurator introduces an additional individual, `XProperty` $\in$ `QtyProperty`, and establishes an association between the user annotations and the shared quality terminology, by asserting that `XProperty` is related to metric `aRC`.

With these definitions, the reasoner infers that (a) `NMC` $\in$ `Confidence-ind`, and (b) `XProperty` $\in$ `Accuracy`. In practice, we have used the reasoning capabilities associated with OWL DL to classify elements from the user input, in a way that is consistent with a predefined ontology. The resulting quality annotations are consistent with the model and can be stored for future querying.

5 Conclusion

By bridging the gap between the quality domain and application domains, the Qurator ontology allows scientists and bioinformaticians to describe their personal perceptions of data quality in e-science in a natural yet formal way, while relationships with a shared quality model are automatically established. This allows us to provide a controlled environment for managing user extensions to the ontology, which in turn facilitates incremental development of domain-specific quality models. As our scenario illustrates, users are only expected to provide information about their own tools and indicators. Qurator then searches for relationships between the new domain-specific concepts and the existing quality concepts. This combination of extensibility and reusability has the potential to produce rich, community-supported bases of shared knowledge that eases the navigation and exploitation of the information resources provided by e-Science. It is expected to have particular value for scientists who are not experts in the domain of the data (e.g. proteomics scientists wishing to make use of transcriptomics data).

In addition to the entailment patterns described in this paper, a number of other patterns can potentially be supported by the Qurator model. We are currently exploring these, as well as expanding our understanding of the uses of information quality in our two application areas. We need to learn more about the ways in which quality annotations of the kind provided by our model can be used in practical applications, and how the model can be better embedded within upcoming e-Science tools.

A DL assertions for the user scenario

1. $\texttt{aDPS} \in \texttt{QualityPreferenceSchema}$
2. $\texttt{aRC} \in \texttt{QualityMetric}$
3. $\texttt{aDPS pref-based-on-metric aRC}$
4. $\texttt{metric-based-on-indicator HitRatio}$
5. $\texttt{aRC metric-based-on-indicator NMC}$
6. $\texttt{NMC} \in \texttt{Indicator} \sqcap (\exists\ \texttt{is-parameter-of}\ .\ \texttt{PMFMatch})$
7. $\texttt{PMFMatch} \sqsubseteq \exists\ \texttt{has-QC}\ .\ \texttt{ConfidenceQC}$
8. $\texttt{Confidence-ind} \equiv (\exists \texttt{is-parameter-of}\ (\exists \texttt{has-QC ConfidenceQC})) \sqcup (\exists \texttt{is-output-of}(\exists \texttt{has-QC ConfidenceQC})) \sqcup (\exists \texttt{has-QC ConfidenceQC})$
9. $\texttt{Accuracy} \equiv \exists\ \texttt{QtyProperty-from-metric}\ (\exists\ \texttt{metric-based-on-indicator}\ (\texttt{Specificity-ind} \sqcup \texttt{Confidence-ind}))$
10. $\texttt{XProperty} \in \texttt{QualityProperty}$
11. $\texttt{XProperty property-from-metric aRC}$

References

1. R. Aebersold and M. Mann. Mass spectrometry-based proteomics. *Nature*, 422:198–207, March 2003.
2. F. Baader, I. Horrocks, and U. Sattler. Description Logics. In S. Staab and R. Studer, editors, *Handbook on Ontologies*, pages 3–28. Springer-Verlag, 2004.
3. L. English. *Improving Data Warehouse and Business Information Quality*. Wiley, 1999.
4. A. Brazma *et al.* Minimum Information about a Microarray Experiment (MIAME) — Toward Standards for Microarray Data. *Nature Genetics*, 29:365–371, December 2001.
5. P. Missier, S. Embury, M. Greenwood, A. Preece, and B. Jin. An ontology-based approach to handling information quality in e-science. In *Proc 4th e-Science All Hands Meeting*, 2005.
6. A. Pandey and M. Mann. Proteomics to study genes and genomes. *Nature*, 405:837–846, June 2000.
7. S. D. Patterson and R. H. Aebersold. Proteomics: the first decade and beyond. *Nature Genetics*, 33(Supplement):311–323, March 2003.
8. T.C. Redman. *Data Quality for the Information Age*. Artech House, 1996.
9. R.Y.Wang, M.Ziad, and Y.W.Lee. *Data Quality*. Advances in Database Systems. Kluwer Academic Publishers, 2001.
10. Y. Wand and R.Wang. Anchoring data quality dimensions in ontological foundations. *Communications of the ACM*, 39(11), 1996.
11. R. Y. Wang and D. M. Strong. Beyond accuracy: What data quality means to data consumers. *Journal of Management Information System*, 12(4), 1996.
12. C. Wroe, R. Stevens, C. Goble, A. Roberts, and M. Greenwood. A suite of DAML+OIL ontologies to describe bioinformatics web services and data. *International Journal of Cooperative Information Systems*, 12(2):197–224, 2003.