

Bringing a Business Rule Perspective to Resource Discovery (Position Paper)

Carlos N. Cumberbatch, Suzanne M. Embury, and Carole A. Goble

Department of Computer Science
University of Manchester
Oxford Road, Manchester M13 9PL
cumberbc@cs.man.ac.uk

Abstract. The emergence of the Internet has changed the way in which many aspects of business are conducted. One such aspect, that of resource discovery, is being changed by the development of a number of resource discovery systems, which facilitate the discovery of businesses (and their resources) anywhere in the world, through the medium of the Internet. However, these systems lack the power to discover potential trading partners based on business policy criteria, simply because they do not contain this sort of information. In this paper, we give an overview of current business discovery initiatives (focussing in particular on UDDI) and examine the consequences of their lack of support for business-related search criteria. We also present and discuss a potential first step towards a solution.

1 Introduction

Since the inception of the Internet, the way in which business is conducted has been completely revolutionised. Markets have become much more dynamic and highly competitive, with the concept of national borders having becoming, in many respects, a thing of the past. The Internet effectively plays the role of a global marketplace in which companies of any size, category or location can (potentially) conduct business with each other.

Traditionally, most companies preferred to transact their business with partners whom they knew and trusted. Nowadays, however, the increasing range of global communication mechanisms plus more flexible trading practices mean that businesses are more inclined to trade with those companies that can satisfy their objectives with respect to some specific criteria (fastest, cheapest, highest quality, etc). This trend has led to the proposal of several initiatives, such as ebXML [29, 18, 24], UDDI [29, 2] and XML/EDI [29, 5], aimed at allowing companies to advertise their products in a globally accessible repository, so that they may be discovered by potential trading partners.

This discovery process [21, 6], illustrated in Figure 1, normally involves a service requestor/client (who is interested in acquiring a particular service or

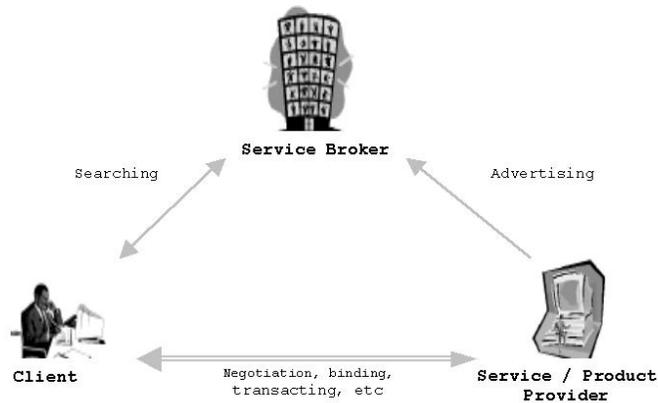


Fig. 1. Overview of the Global Business Discovery Cycle

product) querying a service broker [6] for details of the services or products that match its criteria. The service broker stores this information in a registry, which contains information such as the contact details of the business that provides the service, what the service actually does and how it may be activated. If, on querying the service broker, the client finds an appropriate company to do business with, a series of actions are carried out which ultimately results in the client and the service/product provider transacting with each other.

A major drawback of most the initiatives proposed to facilitate this discovery process has been their focus on the technical aspects of the products and services, rather than the dual focus on both the technical and business aspects that characterises more traditional business practices. This one-sided approach can be seen as a major limitation, since these business aspects are often the most significant factors in choosing an appropriate business partner. If we disregard for the moment technological issues, the process of forming a business relationship typically involves the following sequence of activities:

1. A business entity A realises that it has a need to satisfy (e.g. it needs a currency converter).
2. Business entity A uses some mechanism to discover another business entity B that has a product or service which will satisfy that need (e.g. a business directory, a search engine or teletext).
3. Business entity A establishes contact with B to confirm whether the product/service can satisfy its need (e.g. does the currency converter offered convert all necessary currencies?).

4. Business entity A establishes whether the B's policies in relation to the product or service are compatible with its own (e.g. how much does the service cost? Is it too expensive?)

The fourth activity in this sequence, which checks for compatibility of the provider entity's business policies, is just as important as the third activity when determining whether the service is suitable or not. However, the business discovery initiatives proposed so far provide very little support for this kind of compatibility checking. This means that, in many cases, businesses cannot simply accept the recommendations of the service registry without further manual checking and verification that they are indeed appropriate for the business context in which they will be applied. In addition, as the number of businesses registered with the broker increases, so the need to provide business-oriented search criteria will grow, since there will eventually be too many services that are all but indistinguishable when compared in terms of their technical characteristics alone.

What is required, therefore, is some means of expressing these business policies in a form that can be queried easily and efficiently. One possible solution is to allow businesses to describe their policies as collections of business rules [14]. There has recently been a rise in interest in the possibility of sharing business rules over the web, and a number of mark-up languages for rules have been proposed [8, 7, 9]. In addition, many rule engine products are now available which can facilitate compatibility checking and inferences over business policy descriptions.

In the remainder of this position paper, we will discuss the potential for using business rules to enhance current proposals for business discovery facilities, and will outline the challenges that must be overcome before such a solution can become a reality. We begin by describing the general business discovery problem, and present one of the most popular initiatives (UDDI [2]) in more detail, as an illustrative example. We will then discuss the implications of, and challenges involved in, applying a business perspective to registries such as UDDI. In the final sections of the paper, we discuss how business rules can actually be applied to UDDI to enhance its business discovery capabilities, and conclude.

2 The Business Discovery Dilemma

The difficulty of locating potential trading partners is a problem that has dogged the e-business community ever since its inception. Nowadays, there are various search mechanisms available to the casual web user but we are still far from an ideal solution for discovering businesses over the Internet. Such a solution must possess the following ambitious list of characteristics:

Scope and Relevance of Coverage The value of a business discovery system would be limited if its coverage was restricted to businesses registered in a particular country or those of a particular size. Since the Internet has created a trading environment in which there are no international borders, there

is now the possibility (and, because of this, the need) to be able to locate potential trading partners regardless of their location, size or industry sector. From the client's point of view, it is also necessary for the business discovery system to store details of a sufficient number of business entities, so that the probability of a successful outcome from each query is sufficiently high.

In addition to the scope of the coverage, it is also important that the business discovery system should not contain too much information that is unlikely to be of use to its clients. The storage of a high proportion of irrelevant information is likely to result in many irrelevant query results, unless the search criteria and matching engine are both extremely advanced.

Power of Business Queries The ability to respond efficiently to a wide range of business query expressions is clearly a significant factor in the success of any business discovery system. The more coarse-grained the search criteria, the greater the number of irrelevant businesses that will be returned in the result set, and therefore the greater the amount of manual filtering that the client will have to perform (e.g. by telephone). Since time is money, inaccurate searching capabilities can significantly add to the costs of using a business discovery service. A related issue is that of how much information can be stored about each registered business. With current technologies, only very simple descriptions of the policies by which a business operates (largely in the form of natural language) are stored, and therefore their ability to locate businesses through queries is severely limited. This is another cause of irrelevant results, and therefore of added costs of using the business discovery service.

Simplicity This is also a major factor for the success of a business discovery system, since if a system is not easy to use then it will not become popular, no matter how useful the functionality it offers. This point is related to the issue of querying power just mentioned, in that in general more complex search criteria imply the need for a more complicated user interface. In current systems, the user interface is simplified at the expense of fixing many of the search criteria (see, for example, the interface of the Where2Go system [27], which is illustrated in Figure 2). Finding a broad set of fixed criteria that is relevant to a wide range of industries is an extremely difficult task. Ease of use is also an important factor for the companies who wish to register with business discovery systems, as well as for clients who wish to search for companies. If it is awkward to register accurate information with the discovery service, then companies may be reluctant to invest the time and effort required. It is also important that the registering company feels that it retains ownership of the information about itself, that its entry can be kept up-to-date easily and that it is not expected to reveal information that it would rather keep confidential.

Robustness Although the downtime of systems across the Internet is not a major issue today, if such business discovery systems become successful and are well used, then their users may change their business practices so that they are reliant on having day-to-day access to the system. If users are

Search Field	Operator	Parameter(s)
Keyword Search	Contains	
and Business Name	Begins With	
and URL	Contains	
and Category	Begins With	
and Brand Name	Contains	
and City	Begins With	

Sales Type

Retail Wholesale Service Professional

Payment Options

Master Card Visa Discover American Express
 Check Certified Check Electronic Payment Financing
 Leasing

Delivery Options

Overnight International UPS U.S. Mail

Sales Support

Telephone Fax Internet Toll Free Phone
 EMail In Person

Sell Via

Internet Mail Order Store Front Catalog
 Telephone Toll Free Telephone

GO

Fig. 2. Power Search GUI of Where2go.com

frequently unable to access the business discovery system, then its reputation will be tarnished.

Accuracy of Results Clearly, the success of the business discovery system will be affected to a large degree by its ability to return accurate results to queries. While accuracy is dependent in part upon the expressiveness of the query language and the richness of the information stored about businesses, it also depends on the capabilities of the discovery system’s matching engine.

Thus far, we have only begun to address these goals, although there are now several technologies that can be considered as business discovery systems. For example, the two most popular methods for locating businesses via the Internet at the moment are:

Market Places: These are registries of businesses where the contributing members all use some agreed common technology [3]. While they can be very useful in certain circumstances, market places have their limitations. For example, their scope is typically restricted to businesses within a particular sector. This means that it is usually necessary for several market places to be searched whenever a complex combination of services is required. Secondly, the fact that each market place may choose to operate with different technologies means that it can be very costly for a business to register with more than one or two. This further increases the fragmentation of the information about the global market place, ultimately resulting in a reduction in the usefulness of each individual market place. Examples of market places in-

clude MetalSite (www.metalsite.com), which is a market place for the metal industry, and Digital exchange (www.digitalmarket.com) which is associated with the semiconductors industry.

Search Engines: Search engines are probably the most popular means of locating other businesses and web services. Searches usually involve keyword based queries over semi-structured or unstructured data. However, search engines can be a very inefficient means of locating potential trading partners, because of the large number of irrelevant or out of date results that are usually returned.

The above-mentioned solutions have so far proved popular largely because they have excelled at one or more of the desired characteristics previously mentioned. For example, many of the more well known search engines have something approaching a global coverage, but they have a very coarse-grained searching mechanism. Market places, on the other hand, have much more restricted coverage. However, since companies choose to register with them for the specific purposes of being "discovered" by potential customers, they will tend to return a much higher proportion of relevant results than would be expected from a search engine.

3 The Universal, Description, Discovery and Integration Approach (UDDI)

One of the more recent business discovery initiatives is UDDI [29], which was proposed jointly by Microsoft, IBM and Ariba Inc. This initiative has so far created considerable enthusiasm within the e-business community and, in time, is expected to be a standard. However, UDDI is designed as a means of locating businesses and web services according to their technical compatibility with the client's requirements. The designers of UDDI acknowledge this limitation, and propose that the use of UDDI be combined with other means of locating business partners (such as market places and search engines) that can take the additional business focus into account [2]. In this paper, however, we will explore the idea that UDDI (or a similar business discovery system) could usefully be extended by allowing companies to use business rules to describe the more business-oriented aspects of the services that they offer. In order to provide the necessary background for this discussion, we will begin by giving a brief overview of UDDI. In particular, we will outline the basic functionality provided by UDDI, and discuss how far it succeeds in its aim of improving the business discovery process in respect of the key characteristics mentioned in the previous section.

UDDI provides a registry of products and services that allows business entities to discover, and begin to transact with, suitable business partners. Companies can use the UDDI facilities to register any web services that they offer using a standard description format, which can then be searched by interested parties. Currently, UDDI registries are hosted at three so-called "operator sites" (namely, Microsoft, IBM and Sub Microsystems). All registries contain the same

information, and any changes made at one operator site will also automatically be reflected at the other sites. Despite the additional complications involved in synchronising the three registry copies, there are clear advantages to maintaining them (most notably, improved reliability of the global UDDI service and reduced traffic at individual operator sites). Plans for the hosting of instances of the UDDI registry at other sites are under consideration (e.g. Hewlett-Packard has signed an agreement to operate another registry instance [1]).

3.1 How UDDI Works

The UDDI registries are based on a hierarchical information model, defined using XML schema [17, 23]. These hierarchies categorise the information (i.e. the information provided by the businesses to describe themselves and their services) into four main data types: business information, service information, binding information and specification information (illustrated in Figure 3 [1]).

The **businessEntity** element is the root element of the structure. It contains details about the business which serve to uniquely identify it. Collectively, the **businessEntity** entries are usually referred to as the 'White Pages', and they form the basic means by which each service provider is advertised in the registry. The **businessEntity** elements also include general information about the type of business, its location and the kind of products it offers. This information is selected from several preset taxonomies, which can be browsed or searched to locate businesses according to product, industry or geographic location.

The **businessService** element is used to represent groups of related web services offered by a particular business entity. For example, one **businessService** might include all the web services that are connected with a specific business process. These elements are also described using terms from the UDDI taxonomies, giving details of the industry sector, the products involved and the geographic location of the business entity.

The **bindingTemplate** element contains the URL of an advertised web service (using the **accessPoint** attribute) and basic information about how to invoke that web service. More detailed descriptions of how the web service operates and the preconditions for its use must be provided in the form of technical specifications, which are referenced by the **tModel** elements associated with each **bindingTemplate**. The **tModels** do not themselves contain the specification information, but instead they provide details of how to access it when required (giving, for example, the name of the specification, the organisation which publishes it and the URL for accessing the specification).

To illustrate how this hierarchical information model is used in the process of business discovery, we will present a simple example. Suppose that a bookshop wishes to accept orders over the Internet. To achieve this, it is necessary that a credit card validation service be integrated with the bookshop's existing order handling system. As a first step, a member of the bookshop's IT staff deploys some software that queries the API of the UDDI registry, and asks for details of credit card validation services. The staff member can then browse the **businessEntity** and **businessService** elements that are returned

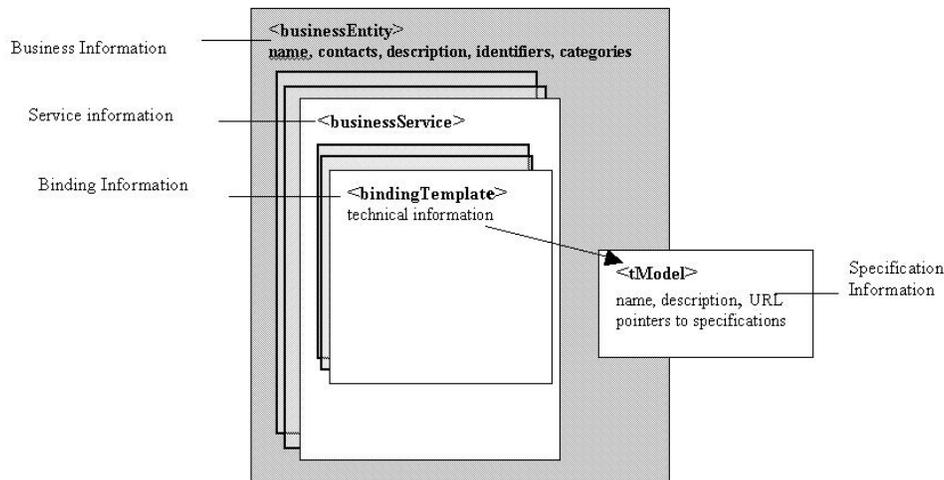


Fig. 3. Template of Classification of Descriptions within the UDDI Model

from the query, to find some basic details about the companies who offer such a service. For a more technical view, the staff member might then examine the `bindingTemplate` element related to a specific service, to discover the URL of that particular creditcard validation web service. In order to ensure that the service is suitable for integration with the bookshop's own system, the staff member might look at the `tModels` associated with the service, to confirm for example that the service conforms to a specific security protocol. Finally, if there is a `tModel` reference to a specification (expressed, for example, in WSDL [10]), which describes the interface of the service in terms of its functions, parameters and return values, then special tools can be used to generate the glueware components necessary for the integration from that specification.

3.2 Conformance to the Desired Characteristics

The scenario described above illustrates a new approach to discovering web services across the Internet. Not only can it dramatically reduce the amount of time needed to determine technical compatibility of candidate services, but it also reduces the amount of time required to code the necessary software components. In fact, UDDI goes some way towards addressing most of the key characteristics that we described as being necessary for a successful business discovery system. The initiative addresses the issue of scope of coverage by admitting and encouraging registrations from companies based anywhere in the world. With major corporations such as Microsoft, IBM, Sun Microsystems, and Hewlett-Packard backing UDDI, it is likely that many other large companies will also be interested in registering. The issue of relevance of scope is catered for to some extent by the

fact that businesses must choose to register, and must invest a certain amount of time and effort in maintaining their registry entry. This self-advertisement approach should mean that much of the irrelevancy encountered using search engines is avoided, although it is probably unrealistic to expect a registry that has broad global scope to provide a highly relevant coverage for all requests.

Matters of robustness are partially addressed by the replication of the UDDI registry, although adequacy of this scheme is yet to be demonstrated in practice. As for the user interface, UDDI currently offers a relatively simple UI, which is easy for first-time users to learn but which is rather restricted in the kinds of queries that can be formulated with it. There is also an API, which businesses can access through their own custom software and which offers much the same querying capabilities as are available through the GUI.

Clearly, therefore, a business discovery system such as UDDI has many advantages over its nearest rivals (market places and search engines). However, it is possible that its concentration on issues of technical compatibility (although sensible as a first step towards a solution to the more general problem) might prove to be a significant limitation on its usefulness. We have already pointed out the potential longer term problem that, if UDDI is successful and many services are registered, technical criteria alone may be insufficient to distinguish the candidate services. In addition, in the short term, registrations with UDDI might be low due to the fact that only a small proportion of businesses have services to offer that are suitable for packaging as web services. The focus on technical compatibility means that UDDI will not be an attractive repository for information about the vast majority of business capabilities, which are better defined in business rather than technical terms. Examples of such capabilities include manufacture and sale of tangible goods, and business services such as tax-auditing. In its current form, UDDI has little to offer such organisations.

4 Incorporating Business Policies within UDDI

Having established the importance of incorporating business policy aspects into business discovery systems, we now turn our attention to the question of how this might be achieved. Business policies are ambiguous statements that describe the business logic used within an organisation [14]. Even when these policies are formally defined, they may still require the interpretation of a human to translate these policies into meaningful statements about how the business should operate [28]. For instance, a toy store might have a policy which states that it only sells safe toys, but what exactly does 'safe' mean? This term would probably be interpreted by the person responsible for purchasing toys as a statement that all purchased products must satisfy some subjective criteria (e.g. toys cannot be swallowed). The process of capturing and understanding business policies usually involves the breaking down of high level statements of these policies into smaller statements until they cannot be broken down further without losing important information (i.e. they are atomic) and are (as far as is practically possible)

unambiguous. Such atomic statements are generally called business rules (BRs). They can be defined as:

”atomic, declarative statements that describe, constrain and control the structure, operations and strategy of a business.” [14]

The potential roles of BRs in the specification, development and maintenance of information systems have been studied for many years [13]. Because they describe business concepts in a way that is precise enough to be implemented in software, BRs can act as a valuable bridge between the business and the technological aspects of organisations. They provide us with a representation of business policies that can also be manipulated by computer programs.

At first sight, therefore, BRs would seem to provide a natural and straightforward means of incorporating business policy aspects into business discovery systems such as UDDI. However, on closer examination, a number of complications emerge.

4.1 Capturing the Business Rules

In order for business policies to be advertised within a services directory, it is first necessary for these policies to be identified and documented. Unfortunately, this in itself is a non-trivial problem that has been studied by both the business and I.T communities for at least the last 10 years [12]. In all but the smallest and simplest of organisations, the set of BRs that are being enforced at any one time is a result of long term evolution, rather than any single coordinated design effort. Typically, over time, organisations can lose track of the BRs that they have in place (for example, when key employees leave or move to a different position within the company). Even where policies have been documented, the information is typically fragmented and is distributed throughout the company. There is also the additional problem of distinguishing documentation for BRs that are now out of date and no longer enforced from that describing current BRs.

However, BRs are not a new phenomenon, and the questions of how best to elicit, represent, store and manage BRs have been discussed and considered by both industry and academia for well over a decade. A strong foundation of supporting methods and technologies have been developed, such as business rule design methodologies [15], rule engines providing inferencing capabilities over BRs [19] and repositories for storing and managing rule sets [16]. More recently, researchers have even begun to develop tools and techniques for recovering BRs from legacy software systems, since such systems can often be the only reliable source of information about the currently enforced business rules [4, 25].

Because of the increasing support for (and perceived usefulness of) BRs, many organisations have already begun to identify and document their key business rules, as part of an attempt to make their businesses more transparent and better able to respond rapidly to changes in the market place. Such companies would be in a strong position to advertise subsets of their rule sets in an extended UDDI registry.

4.2 Representing the Business Rules

Given that the business policies of the company wishing to register with the service broker have already been identified and broken down into business rules, how are the resulting rules to be represented so that they can be manipulated by the business discovery service? Several options can be considered, ranging from natural language (i.e. free text descriptions of BRs) to formal rule notations (e.g. Petri Nets [20], Ross' Method [13]) and database triggers [28]. Clearly, the choice of notation will depend upon how we would expect the rules to be manipulated as part of the business discovery process. We foresee a need for (at least) the following forms of rule manipulation:

- The most obvious form of manipulation that will be required is the process of checking for compatibility between the BRs that describe the client's requirements and the BRs that describe the policies associated with a particular service. This amounts to a check for satisfiability of the conjunction of both these sets of rules, and therefore the expressiveness of the rule language chosen will have a profound effect on the ease and efficiency with which this check can be performed (since certain forms of the satisfiability problem are known to be NP-complete [11]).
- A naive approach to locating services that meet the user's requirements is to match the BRs that describe those requirements against the BRs associated with every service in the registry. Given the comments just made on the likely inefficiency of a full compatibility check, any practical rule matching system must have some quick and dirty means of locating a small set of candidate services that are likely to match. Therefore, the rule language must be amenable to this kind of approximate matching. For example, one solution might be to pre-process the BRs, to extract some kind of indexing information from them that can indicate potential matches quickly.
- Regardless of any ability to match business rule descriptions automatically, it is inevitable that the BRs advertised by service providers will also be browsed by human clients of the registry. We therefore require a rule language that can be converted into a more readable form, for presentation to human users of the business discovery system.
- Since some of the business rules advertised by service providers may be highly dynamic, the representation format should be able to support rapid modification of individual rules within rule sets. Ideally, these modifications would be made by business users, rather than programmers, but this is very difficult to achieve when the rule language is highly expressive.

As this list indicates, the key issues here are expressiveness and precision (i.e. formality) of the rule representation language, rather than the exact form of notation used. For example, a free text format would be easily understood and modified by human clients (especially if they are business users), but it is impossible to perform exact matches on rule sets of this kind. At the other end of the spectrum, a formal mathematical language such as first order predicate calculus can be manipulated by software, but is not suitable for use by business users. In

addition, such a language can express a wide range of rule semantics, but by the same token it is very difficult to find efficient algorithms for performing complex analyses over such an expressive language.

A number of researchers have already begun to develop notations for expressing BRs, in a form that is suitable for communicating and sharing such rules between software components over the Internet. One example of such a notation is the Rule Markup Language (RuleML) [8]. This is an XML based, semiformal language that permits web-base storage, interchange, retrieval and firing (activation) of rules. Other examples are the Business Rule Markup Language (BRML) [7] and the Simple Rule Markup Language (SRML) [9]. Any of these languages (or their successors) might be candidates for representing BRs for business service discovery.

4.3 Disclosing the Business Rules

Even assuming that sufficient BRs can be captured and represented, there is a further potential barrier to their registration with a public service broker. Some of the BRs enforced by a company effectively describe its strategy for staying ahead of the competition. The company may therefore be unwilling to disclose such rules publicly. However, it is also possible to identify several reasons why it may be worthwhile to submit to publication of certain carefully selected subsets of the BRs. For example, as stated earlier, publication of the business policies associated with a service could be the key factor which differentiates that service from the crowd of other similar services. It could therefore be in a company's interest to disclose some of its BRs, if this leads to a greater chance of being discovered and distinguished by potential customers.

Of course, one would not expect a company to want to advertise all its identified rules which each service. Instead, the company would need to select a (hopefully small) number of rules that characterises the important features of its policy relating to each specific service, and include only these in the registry entry for that service.

4.4 Storing Business Rules

The issue of where the advertised BRs will be stored is an important one, as it is a significant factor in determining the efficiency of the resulting business discovery service, and the kinds of problems that can arise. A spectrum of storage options are possible, ranging between the following two extreme positions:

Storage of All Advertised Business Rules within the UDDI Registry

In this case, the BRs of all the companies registered within UDDI are stored within some central repository, against which all matching queries are applied. The obvious advantages of this system include robustness, in that if the registry is accessible then all rules are also accessible, and speed of matching, since all rules are available locally for querying purposes. However, one disadvantage of

this approach is that BR specifications may be bulky, and the storage of large rule sets for large numbers of companies may require a prohibitive amount of disk space. In addition, companies may feel that they have lost ownership of their rules by transferring them to the central registry, where they may become old and stale. A further disadvantage is the extra loading placed on the UDDI registry if requests for rule matches is high.

Storage of Business Rules by Individual Companies At the other extreme of the spectrum, we have the case in which BRs are stored by companies themselves (for example, as web pages), in a form that allowed them to be referenced from the UDDI registry. This approach has some advantages in that it places fewer resourcing demands on the UDDI server. Additionally, companies would have direct access to their own BRs, and would be in a position to maintain them more easily. However, the process of matching with business rules is clearly less efficient in this case, since they must be fetched from the company before they can be checked for compatibility. This solution is also less robust, since if a company's system goes off-line then its advertised BRs will not be available for use by the business discovery service.

In reality, the optimal approach is likely to be some hybrid of these two, in which the full business rule sets are stored locally by the advertising company, but some summary information is stored centrally in the service registry for quick and reliable access. Clearly, some experimentation with the possible options will be required in order to determine which solution will work best in practice.

5 Incorporating Business Rules into the Querying Mechanism

In addition to the benefits we have already discussed, the use of BRs for describing the policies associated with advertised services also has the advantage that it is consistent with the way in which people think when searching for products to purchase. For example, the following queries combine constraints on location and product type with additional constraints on service policy:

- Find all toy stores in Manchester, UK, that accept VISA credit cards
- Find me a free weather-forecasting web service
- Find a bookstore which offers discounts on book purchases over \$100
- Find a furniture store in Florida, USA, that allows hire purchase of tables at 10% or less

Clearly, formulation of such queries will require a more sophisticated user interface than the simple form-based interfaces typically offered by service brokers and market places at the moment. This is particularly important if queries are to be posed by business users, rather than by programmers acting on the instructions of business users. Ideally, the user would enter the query as free text, which would then be translated into a machine-processable query (possible through a

dialogue with the user as to the exact meaning of certain parts of the query). However, in practice, it will probably be necessary to provide interfaces which can formulate only certain restricted classes of query, but which are relatively easy to use, while the full power of the query language is only available through the API.

Of course, it will also be necessary to assume some common agreed set of terms and concepts with which to formulate BRs, whether they are to be used in queries or in service advertisements. But, within a global registry such as UDDI, absolute and complete agreement is not a practical possibility. However, this is not a problem that is limited to the specification of BRs alone - this is a general problem for all information that is shared between different systems over the Internet. The current best-of-breed solution to the more general problem is to make use of ontologies [22]. An ontology is a shared formal conceptualisation of a particular domain and provides a common understanding of concepts and terms that can be communicated between people and application systems. An ontology contains a hierarchy of concepts within a domain and describes each concept's essential properties through an attribute-value mechanism [26]. Ontologies also have underlying rules (axioms) which are associated with the hierarchy and which form the basis on which inferences regarding the concepts in the ontology can be made.

Ontologies are helpful in this context as they provide a means of establishing equivalence (or, at least, overlap) of meaning between different terms. Their deductive capabilities can also be used to infer useful relationships between concepts that are not explicitly present in the ontology. For example, if when processing the last of the example queries given above we find a company offering hire purchase on tables at a rate of 2%, it is possible to deduce that this service meets the requirements imposed by the query, even though they are not directly equivalent.

6 Conclusions

We have discussed the negative effects that the focus on technical compatibility aspects has on current business discovery initiatives, such as UDDI. Perhaps the most significant of these for the long term is the fact that, when a type of service is provided by many different providers, the really useful distinguishing features of those providers will be the business context of the service provision, rather than its technical context. Unless business discovery systems begin to include aspects of business policies, both in how services are described and in how search criteria are formulated, then they run the risk that as they become more popular so the results of searches will become less relevant and therefore less useful.

We have proposed the use of business rules as a means of describing the business context for web services to service brokers, and also for specifying business-oriented constraints in searches. Business rules have been a topic of interest in both the academic and industrial communities for many years now, and products for designing, storing and managing rules are now widely available. These

products, in conjunction with novel information sharing technologies, such as ontologies and rule markup languages, provide a good technological foundation for incorporating business rule based advertising and searching facilities into existing business discovery initiatives. However, a number of open questions remain, and in this paper we have outlined some of the research questions that must be answered before such facilities can become a reality.

References

1. Microsoft Ariba, IBM. Uddi frequently asked questions (faq). Nov 2001. available online at <http://www.uddi.org>.
2. Microsoft Ariba, IBM. Uddi technical white paper. Sep 2001. available online at <http://www.uddi.org>.
3. Judith Gebauer Arie Segev and Frank Farber. Internet-based electronic markets. *International Journal of Electronic Markets*, 1999.
4. Janet Wall Barbara Von Halle, Art More. Excavating the business from its legacy electronic assets.
5. Bruce Peat Benoit Marchal, Norbert H Mikula and David RR Webber Version 0.05. Guidelines for using xml for electroinc data exchange. Jan 1998. available online at <http://www.geocities.com/wallstreet/floor/5815/guide.htm>.
6. Steve Burbeck. The toa of e-business services- the evolution of web applications into service oriented components with web services. Oct 2000. available online at <http://www.-4.ibm.com/software/developer/library/ws-toa/index.html>.
7. Robin Cover. Business rule markup language (brml). May 2001. available online at <http://www.oasis-open.org/cover/brml.htm>.
8. Robin Cover. Rule markup language (ruleml). Oct 2001. available online at <http://www.oasis-open.org/cover/ruleML.htm>.
9. Robin Cover. Simple rule markup language (srml). May 2001. available online at <http://www.oasis-open.org/cover/srml.htm>.
10. Greg Meredith Erik Christensen, Francisco Cubera and Sanjiva Weerawarana. Web services description language (wsdl) 1.1. Marc 2001. available online at <http://www.w3.org/tr/2001/note-wsdl-20010315>.
11. Michael R. Garey and David S. Johnson. Computers ad intractability: a guide to the theory of np-completeness. *W.H. Freeman and Company*, 1999.
12. Ellen Gottesdiener. Business rules - show power, promise. *Application Development Trends*, 4(3), Mar 1997.
13. Ronald G.Ross. The business rule book: Classifying, defining and modelling rules. *Boston Massachusetts:Database Research Group,Inc*, 1994.
14. Business Rule Group. Defining business rules what are they really? 2001.
15. Barbara Von Halle. Power by rules. *Business Rule Journal*, 2(10), Oct 2001. available online at <http://www.BRCcommunity.com/a2001/b088.html>.
16. David C. Hay. A repostiotry model- business rules - part 1 (structural assertions and derivations. *The Data Administration Newsletter*. available online at <http://www.tdan.com/i019ht01.htm>.
17. H.S.Thompson. Xml schema part 1: Structures. *W3c working-in-progress*, Apr 2000. available online at <http://www.w3c.org/tr/2000/WD-xmlschema-1-20000407/>.
18. John Ibbotson. ebxml trading-partners specification. May 2001. available online at <http://www.gca.org/papers/xml europe2001/papers/html/s09-2.html>.

19. ILOG. Business rules: Powering business and e-business. May 2001.
20. K Jensen. *Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use*, volume 2. Monographs in Theoretical Computer Science, 1997.
21. Cape Clear SOftware Limited. Capeconnect three- web services and capeconnect. Nov 2001. available online at <http://www.capeclear.com>.
22. Deborah L.McGuinness. Ontologies and online commerce. *Computer.org*, Jan 2001. available online at <http://ww.computer.org/intelligent>.
23. P.V.Biron and A.Malhotra. Xml schema part 2: Structures. *W3c working-in-progress*, Apr 2000. available online at <http://www.w3c.org/tr/2000/WD-xmlschema-2-20000407/>.
24. Karsten Riemer. ebxml business process. May 2001. available online at <http://www.gca.org/papers/xml europe2001/papers/html/s18-1.html>.
25. J Shoa and C J Pound. Extracting business rules from information systems. *BT Technical Journal*, 17(4), 1999.
26. Frank Van Harmelen Dieter Fensel Michel Klein Jeen Groekstra Michael Erdmann Stefan Decker, Serge Melnki and Ian Horrocks. The semantic web: Roles of xml and rdf. *IEEE Internet Computing*, Sep/Oct 2000.
27. HLI Systems. Where2go.com launches free web contest engine. Jun 2000. available online at <http://www.where2go.com/pr/contests.html>.
28. Brokat Technologies. Broakat advisor (from blaze) white paper. Marc 2001.
29. David Webber and Anthony Dutton. Understanding ebxml,uddi,xml\edi. *XML.org*, Oct 2000. available online at http://www.xml.org/feature_articles/20001_1107_miller.shtml.