On Analyzing and Specifying Concerns for Data as a Service

Hong-Linh Truong and Schahram Dustdar
Distributed Systems Group, Vienna University of Technology
Email: {truong,dustdar}@infosys.tuwien.ac.at

***Abstract*—Providing data as a service has not only fostered the access to data from anywhere at anytime but also reduced the cost of investment. However, data is often associated with various concerns that must be explicitly described and modeled in order to ensure that the data consumer can find and select relevant data services as well as utilize the data in the right way. In particular, the use of data is bound to various rules imposed by data owners and regulators. Although, technically Web services and database technologies allow us to quickly expose data sources as Web services, until now, research has not been focused on the description of data service concerns, thus hindering the discovery, selection and utilization of data services. In this paper, we analyze major concerns for data as a service, model these concerns, and discuss how they can be used to improve the search and utilization of data services.**

I. Introduction

Taking the advantage of Web services technologies, the software as a service model [1] and cloud computing [2], recently, various research effort have concentrated on the development of the concept of data/information as a service (DaaS) [3]. Whether a service is a DaaS can depend on specific context. For example, a service can simply allow consumers to create, store and manage their own data according to their specific data models (Storage as a Service) or can provide credits and balance sheets of companies for consumers. However, DaaSs have a common property: they mainly provide data capabilities based on common data CRUD (Create, Read, Update, Delete) commands rather than computation on data. DaaSs offer functionalities to allow their consumers to acquire or provide data under the service model, regardless of whether the offerings are free or commercial. Over the last few years, various providers have provided (and claimed) services as DaaS. Yet still from an outlook of a consumer, it is difficult to distinguish a DaaS from other types of services. It is partially due to the fact that currently there is a lack of well-defined and -understood description models that are able to characterize concerns for DaaS. Most of today's DaaSs in the Web just provide WSDL- or REST-based interfaces describing their operations and static Web pages about pricing and usage permission[1]. DaaSs are still described in terms of typical QoS (Quality of Service), but not of specific concerns related to the data a DaaS provides, while data is the main ingredient that makes DaaS different. This problem has hindered the consumer from the selection and utilization of DaaSs due to the lack of knowledge about data.

We argue that a DaaS should be described and published in a way that it is able to highlight distinguishable characteristics of the data it provides, for example, whether a DaaS supports fundamental requirements for data governance, which meta-data is associated with the data a DaaS provides, whether the data can be used freely for commercial purpose, to name just a few. A DaaS has, therefore, to be characterized by not only traditional QoS but also quality of data and other concerns. While data quality (DQ) has been extensively studied in database research, how to associate DQ with DaaS is not defined yet, let alone the combination of QoS and DQ for DaaSs. We further argue that characterizing QoS and DQ is not enough and we also need to address other concerns such as data usage, service context, and data source concerns as well as the license issue associated with DaaSs. Among these issues, only QoS has been extensively studied for services that can be utilized for DaaSs. The issues of DQ, data service licensing and other concerns, and their combination for DaaSs remain open. A systematic approach to the description of DQ/QoS and service/data license for DaaSs is missing. Likewise, current QoS-based service selection and composition methods (e.g., [4], [5]) need to be extended to cover also DQ and service/data license aspects. Addressing these needs is a must to support mashups of data from global services for business and e-science, e.g. shown in [6].

To tackle the above-mentioned issues, we aim at analyzing concerns related to DaaS. We present a detailed study of DaaS concerns and propose a model describing these concerns which helps enhancing the search, comparison, selection and utilization of DaaSs. We also propose that service contract models should combine service licensing and data licensing, DQ, QoS, and other concerns. Our main contribution of this paper is a novel model and approach for describing concerns of DaaS that provides a foundation for future search and composition of DaaSs based on data concerns. In this paper, we also present a study of how existing providers describe their DaaSs.

The rest of this paper is organized as follows: Section II discusses the related work. We elaborate existing issues and our approach in Section III. DaaS concerns and the model describing them are presented in Section IV. We present our empirical studies of DaaS in Section V. Section VI

This paper is published in the proceedings of the 2009 IEEE Asia-Pacific Services Computing Conference (APSCC).

[1]e.g., see StrikeIron Web Services catalog at http://www.strikeiron.com/strikeironservices.aspx

summarizes the paper and discusses our future work.

II. Related Work

Existing SOA techniques can be used to model and design interfaces for accessing and managing data held within a DaaS. QoS models for Web services have been well developed and various techniques, methods and tools support the developers to model QoS for Web services [7], [8]. However, they focus on operational aspects of services like performance, reliability, availability, and security, while the information/data aspects related to the publishing of DaaS are almost ignored. DaaS publishing not only requires an appropriate description of the semantics of the data and the data sources (e.g., a data schema or an ontology), but also a more general specification reflecting DQ, QoS, data governance, data usage, and data service licensing. The past research has focused on quality of data from database perspectives, such as in [9], [10], [11]. A detailed review of DQ metrics and methodologies is given in [12]. While these DQ research works can be utilized for describing DaaS, until now there is a lack of integration between DQ metrics and service information for DaaS. In our work, we utilize many common DQ metrics in [9], [10], [11], [12] to describe DaaSs.

With respect to the service model, DaaS has some analogy to SaaS (Software as a Service). Therefore, even though SaaS offers on-demand software application capability and DaaS offers on-demand data, there exists a question of whether the SaaS model with some specific QoS metrics is enough for publishing DaaS concerns. We believe that DaaS concerns are much more complex than just some specific QoS metrics and they also include domain-specific knowledge. Nevertheless, how to extend current QoS models to describe data quality and licensing for data is an open research question.

When DaaSs are commodity and accessible from the Web, they can be utilized by different types of consumers (e.g., humans or software) for different purposes, e.g., via data mashup techniques [6]. As a result, data service licensing also becomes increasingly important. DaaS licensing is a complex issue because the data may be diverse and is space and time dependent, thus requiring complex license models that are able to describe rights of using data. Unlike service licensing [13] and existing service level agreements (SLAs), see e.g, [14], which are defined mainly based on the above-mentioned QoS models, the DaaS license heavily depends on DQ and data aspects of QoS. Moreover, new methods of data service selection and data service combination are needed for DaaSs as existing service selection and combination techniques are built around the QoS and the semantics of service operations [4], [5] without paying attention to DQ, data aspects of QoS and service licenses. Some data licensing models exist but are not standard and formalized, see e.g., [15]. Service licensing [13] is one key element in the concept of SaaS but only few aspects of service licensing have been addressed. The ODRL-S model [16] proposed by [13] is only for service-related licensing terms. We utilize permissions in this model to describe permissions associated with DaaS from service perspectives and also extend them for describing permissions associated with data. Furthermore, the warranty, indemnity and liability specifications in ODRL-S are also reused in our DaaS concerns model.

Recently, the role of licenses for open data has been stressed in many places, such as in [17]. Current service licensing research addresses the use of services but not the result produced by services. With DaaS, the data provided by DaaS is also strongly bound to specific data licenses. Until now, we are not aware of any data service licensing models for DaaS. There are some initiatives working on licenses and their guidelines for open data, such as Talis Community License[2], the Open Knowledge Foundation Wiki[3] and Open Database License[4]. However, these initiatives do not address licenses for DaaS. Nevertheless, they provide a good model for data licenses, thus we can utilize various terminologies and concepts in describing DaaS concerns.

III. Issues and Approaches

The use of a DaaS is bound to various concerns. Some concerns are technical specific to the data and the service, for example, the quality of data and service. However, there are also many other concerns related to business, regulatory and compliance aspects, such as pricing, copyright, and law enforcement. All of these concerns are critical for the search, comparison and selection of DaaSs. To date, there is a lack of models specifying these concerns. In the following, we outline the importance of having these concerns explicitly specified when publishing a DaaS.

Quality of data concerns: the core of a DaaS is the data it provides. Therefore, DQ concerns are what the consumer would like to utilize in selecting DaaSs. In particular, many similar DaaSs may use the same source of raw data but support different updating, cleaning and enhancement techniques, resulting in different DQ metrics which are critical in many businesses. For example, checking a company credit is required in many transactions, thus if a DaaS offering this function does not provide an up-to-date data, then it may provide the credit information about some companies which are out of business.

Data source concerns: typically, the service consumer wants to know information about data sources (providers and quality) that a DaaS relies on. This type of information contributes to the trust the service consumer has on a DaaS when the data source is reputable.

Usage concerns: these concerns are related to both data and service aspects. While many DaaSs provide detailed information about business models (e.g., price model), these models are not given in well-structured documents to be processed by tools. Furthermore, current DaaSs lack well-defined documents about usage permission, intellectual property rights and legal issues. In particular, usage permission and intellectual property rights are associated not only with DaaS APIs (how

[2]http://www.talis.com/tdn/tcl

[3]http://www.okfn.org/wiki/OpenDataLicensing

[4]http://www.opencontentlawyer.com/open-data/open-database-licence/

to use the service) but also with the data the DaaS produces. For example, many DaaSs can be used freely, but their data is copyrighted. All of these issues prevent the consumer to utilize DaaSs because of the unclear permissions, in particular, associated with the data.

Data governance concerns: the use of a data source is typically followed the cycle of data governance[5]. Depending on different types of businesses, before deciding which DaaS to be used, a consumer may analyze the impact of local law to the data (e.g., the data has to be encrypted), the support of data quality assurance, security and privacy compliance, data classification, information lifecycle, and auditing features that a DaaS can support.

Quality of services concerns: as usual in the service environment, a DaaS is a service thus QoS information is necessary. In particular, the issues of availability and security are critical for accessing data in DaaSs. QoS has been well studied in the past and many existing works can be utilized. However, QoS concerns have not been linked with other concerns to enhance the selection of DaaSs.

Service context concerns: context associated with service, such as location and classification, is also important. The location of data is an important issue as various rules require the data to be processed and provided for particular consumers based on location. For example, the Canadian government policy on using service[6] forces public agencies to use storage data services only in Canada.

In tackling the above-mentioned issues, we need to focus on publishing information characterizing DaaSs. Currently, there is no well-understood publishing model for DaaSs. Existing works tend to consider DaaSs as normal services whose publishing information is based on service interfaces (described by WSDL and REST API/WADL) and QoS only. This neglects the data aspect which is the core of DaaSs. To this end, we need to understand, for example, DQ, data security, lifecycle, business, and service context concerns for DaaSs, describe them together with QoS and WSDL, and to provide service contracts, thus allowing consumers to search and select DaaSs based on these concerns.

IV. Modeling Concerns of Data as a Service

In our view, to a service consumer DaaSs can be categorized into (1) Read-only DaaS which only provides data based on existing data sources, such as StrikeIron Address Validation and XigniteRealtime, and (2) CRUD DaaS which allows the consumer to create, retrieve, update and delete data. In the second category, DaaSs can be infrastructure-based in which services typically just provide a storage capability (Storage as a Service) and it is up to consumers to define their own data schema and/or to publish their data. One example of this type of services is the Infochipms[7].

[5]see the IBM Data Governance Maturity Model at http://www-01.ibm.com/software/tivoli/governance/servicemanagement/data-table.html

[6]http://news.bbc.co.uk/2/hi/technology/7421099.stm

[7]http://infochimps.org

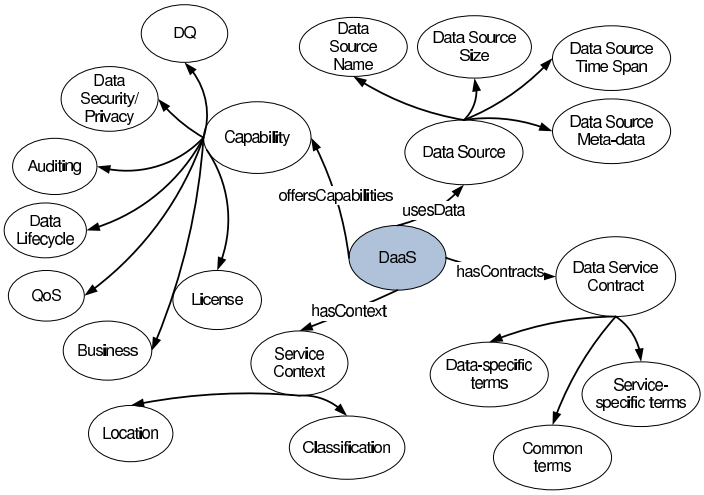


Fig. 1. DaaS's concerns and contracts

Currently, most DaaSs are Read-only (based on our study in Section V). CRUD DaaSs are usually used by consumers to publish data which are then being offered to other consumers. Depending on the type of DaaSs, different concerns have different impact. For example, the service consumer is more concerned on data security and privacy for infrastructure-based DaaSs than for Read-only DaaSs.

In our work, DaaS will be published and utilized based on various concerns. Figure 1 depicts main concerns associated with DaaS that we divide into four main categories: capability, service context, data source, and data service license. The first three concerns represent DaaS properties and the last concern, built upon the first three concerns, represents conditions established on properties under different circumstances.

A. DaaS Capability, Data Source and Service Context

Capability: to allow DaaS being utilized by different types of consumers for different purposes, capability concerns of DaaS need to be defined and published. Capability concerns of DaaS are classified into DQ, Data Security/Privacy, Auditing, Data Lifecycle, and QoS. Table I describes main concerns in the capability category.

DQ capabilities are defined based on a long, well-established research on data quality [9], [10], [11], [12]. We categorize DQ into domain-independent and domain-specific metrics. For domain-independent DQ metrics, several definitions are summarized in [12] which can be used to define the representation of DQ metrics.

Data Security/Privacy capabilities of DaaS describe how a service can ensure secured data. Here we distinguish between the data protection of data transfer (in the communication between the services and the consumers) with the data protection internally in DaaS. This type of capabilities deals with internal data protection, including sharing data between the service and involved third parties (e.g., backup service). The data protection in communication is usually classified in QoS.

Auditing capabilities describe how DaaS supports auditing activities, such as logging, reporting and warning.

Category	Properties	Description
DQ	Timeline	describes the lifetime of the data. For example, the AERS (Advanced E-Commerce Research Systems) API service provides analysis data based on 90 days of history [18].
	Uptodate	indicates the lag time of the data up to the current time. For example, eBay provides data which is two days behind the current time [19].
	Objectivity [11]	describes whether the data is biased.
	Free-of-error [11]	describes to which degree the data is reliable.
	Cleaning	describes to which degree the data is cleaning. For example, the Ebay Data License mentions that the provided data can be raw, partially cleaning and fully cleaning.
	Consistency [11]	describes to which degree the consistency of data is supported.
	Completeness	describes whether the data has missing values. Here we should note that completeness is context-dependent [20]. For the sake of simplicity, we distinguish the completeness of an individual data element - whether a data element misses some data fields - and the completeness of the data set - whether the data set misses some data elements.
	Granularity	describes the degree of data granularity.
	Domain-specific metrics	describes domain-specific DQ based on external models.
Data Security/Privacy	Privacy	describes privacy practices according to The Platform for Privacy Preferences (P3P) project [21].
	Encryption	describes whether the information is encrypted or not. Note that it is applied to the data, not the network security which is defined under the QoS capability.
Auditing	Logging	describes whether all data transactions are logged or not.
	Reporting	describes whether reports are made, for example, in a daily, weekly, and monthly manner.
	Warning	describes whether warnings can be sent in certain situations, for example, when the data source is changed.
Data Lifecycle	Backup/Recovery	describes whether and how the data will be backed up, and to which degree and how long the data can be recovered if the data was lost.
	Distribution	describes whether the data will be distributed externally. For example, data in a service can be stored in an external organization. The distribution information also includes geographical locations.
	Disposition	describes whether the data will be relocated or retained according to consumer-defined or lawful policies
QoS	Performance	includes several metrics describing the performance of DaaSs, such as start/end time, response time, latency, and service throughput [7], [22].
	Dependability	includes several metrics describing the dependability degree of DaaSs, such as availability, accessibility, reliability, and security [7], [22].
Business	Price model	describes possible pricing models, such as flat rate or pay-per-use/pay-as-you-go (subscription) with/without conditional transactions numbers, or free-per-use, and whether the pricing models offer different packages. For example, StrikeIron offers pay-as-you-go (subscription) with conditional transactions for different models.
	Price	describes the price in detail, such as proposed in [23].
	Service Credit	describes whether the customer can get some service credits as a reward or compensation.
License	Usage Permission	describe how a DaaS can be used. This will include both data and service aspects. For example, service-related permissions can be based on the ORDL-S model [16], such as adaptation, composition, and derivation. Data-related permissions can be distribution, transfer, personal use, commercial product, etc. The permission may include which software can use the data. For example, the Free Price Research API eBay [24] stated that it permits a shopping widget or a portal to use the service.
	Copyright	describes how the service and the data it provides is protected.
	Liability	describes the liability associated with the use of the service and its data.
	Law Enforcement	describes the law which is used to deal with the legal of the data and service. For example, the use of many DaaSs is followed the US law.
	Domain-specific IPR	describes specific intellectual property rights for the service and its data.

TABLE I
EXAMPLES OF CAPABILITY CONCERNS FOR DAASS

Data Lifecycle capabilities deal with steps in information lifecycle management[25]. They specify data backup and recovery, data distribution, and data disposition.

QoS capabilities describe well-known QoS associated with services. It includes very common and popular metrics, such as availability, reliability, and security. For this category, we utilize as much as possible existing QoS metrics, such as in [22], [26], [4], [5].

Business capabilities describe the pricing, reward, and compensation capabilities of DaaSs.

License capabilities describe possible usage, IPR (Intellectual Property Rights) and legal concerns for DaaSs.

The capabilities have different impact on the selection of different types of DaaS. For example, DQ concerns are more critical for the consumer to select a Read-only DaaS while auditing and data management lifecycle are more important for infrastructure-based DaaS.

Data Source (DS): this type of concerns provide further information about the source of data. Depending on types of DaaS. DS concerns categorize into domain-specific and domain-independent. Table II presents main concerns associated with data sources.

Service Context: the service context concerns describe the context in which the service can be used. It includes location, service classification and data classification for DaaS. For service classification, we propose to use the UNSPSC Code Classification Services[8]. Table III describes main concerns in

[8]http://www.unspsc.org

this category.

DS Properties	Description
Name	describes where the data is obtained. For example, many DaaSs utilize the ddfFlus[27] or DataFlux [28].
Size	describes the volume of the data
Timespan	describes the time duration in which the data is collected. For example, eBay Data License mentioned that the data has been accumulated in more than 4 years [18].
Update Frequency	describes how often the data is updated. It is especially critical in financial-related applications
Meta-data	describes domain-specific standards that the data follows, data schema, etc.

TABLE II
EXAMPLES OF DATA SOURCE CONCERNS

Properties	Description
Location	describe where a DaaS is hosted. It is particular important for consumers who have constraints on the place where the DaaS operates.
Service Type	describe whether the service is based on SOAP or REST.
Level of Service	describes whether the service is best effort or guaranteed [8]
Service Classification	describe the class, e.g., the financial domain, the service belongs to.
Data Classification	describe the taxonomy characterizing the data provided by the service.

TABLE III
SOME SERVICE CONTEXT CONCERNS

Category	Terms & Description
Generic	Business terms based on business capabilities. License terms based on IPR & Legal capabilities. Location terms based on the service location
Data-specific	DQ terms based on DQ capabilities. Data Security/Privacy terms based on security and privacy capabilities. Auditing terms based on auditing capabilities. Data Lifecycle terms based on data lifecycle capabilities.
Service-specific	QoS terms based on QoS metrics.

TABLE IV
MAIN DATA SERVICE CONTRACT TERMS

B. Data Service Contract

Based on DaaS capability, data source, and service context concerns, data service contracts (DSCs) can specify possible constraints established on the basis of concerns to specify agreements in utilizing DaaS. Unlike contracts for separated services or data sources, contracts for DaaS will reflect the general conditions that the consumers should agree when using services and data: they cover service- and data-relevant aspects. Table IV describes main elements of a data service contract for DaaSs. Generic terms will be built based on business, license and location capabilities of DaaSs, service-specific terms are built based on QoS, and data-specific terms are mostly relied on DQ, auditing, security and privacy, and lifecycle management capabilities.

C. Populating Data Concerns

Given the list of concerns and their representations, various stakeholders can gather data about concerns and populate the data for the discovery and selection of DaaSs. Methodologies for populating the data are dependent on implementation detail. However, similar to current techniques for publishing and utilizing non-functional parameters associated with services, there are three possibilities to populate the data about concerns: (i) DaaS providers can specify concerns, and publish and manage these concerns, (ii) DaaS consumers can specify consumer-specific data concerns and select DaaS based on these concerns and can play the role of a third-party to provide useful information about data concerns to other consumers, and (iii) third-parties can provide useful feedback and monitoring data about published data concerns.

D. Implementation and Management of DaaS Concerns Description

To support DaaS publishing and selection, DaaS concerns have to be modeled in a form that can be parsed and analyzed by software. Currently, existing providers provide descriptions of only some concerns in a form of static Web pages. Therefore, the search and selection of DaaSs has been mostly carried out by humans in a manual way (see Section V).

We are currently implementing our prototype for collecting, publishing and managing DaaS concerns[9]. There are two issues for the implementation: which model is suitable for describing capability, service context and data source concerns, and which one is for DSC terms. Since the number of capability, service context and data source concerns is large and these concerns are domain-dependent and domain-specific, we need an extensible mechanism to model concerns. Especially, new concerns and different domain-specific models should be seamlessly integrated. One can select different ways to describe concerns associated with his/her DaaSs, such as annotating WSDL or utilizing SAWSDL[10]. Our approach is that for describing DaaS concerns we develop XML/RDF schemas which also support the association of external models. Using URIs, external models of concerns, e.g., domain-specific models like the Darwin Core[11] for biodiversity, can be linked to our model. In this way, concerns can also be described in different languages, such as RDF and OWL.

For describing data service contracts, one can select different techniques, such as WSLA, WS-Policy, WS-Agreement and ontology-based approaches. In our work, we have utilized the WS-Agreement standard and the PCM [29] to define different types of contract properties. Providing models for describing concerns is not enough. We also need to develop a DQ/QoS and DSC management framework that supports the provisioning, management and search of DQ/QoS and DSC information. This framework will allow us to associate different DQ/QoS and DSC models for specific data services

[9]The implementation is reported at http://www.infosys.tuwien.ac.at/prototyp/SOD1/dataconcerns/

[10]http://www.w3.org/2002/ws/sawsdl/

[11]http://wiki.tdwg.org/DarwinCore/

```
<title>CorteraCreditPulse Service</title>
<entry>
   <title>Interface</title>
   <summary>WSDL Interface </summary>
   <category label="Web Service Description" scheme="http://www.dmoz.org/Computers/
     Programming/Internet/Service-Oriented_Architecture/Web_Services/WSDL"
       term="Interface"/>
   <content type="application/wsdl+xml" src="http://ws.strikeiron.com/
     CorteraCreditPulse2?WSDL"/>
</entry>
<entry>
   <title>DaaS Concerns</title>
   <summary>Data Concerns</summary>
   <category label="Data Concerns"  term="DaaSConcern"/>
   <content type="application/xml" src="http://www.infosys.tuwien.ac.at/prototyp/SOD1/
      dataconcerns/samples/CorteraCreditPulseConcerns.xml"/>
</entry>
```

Listing 1. Example of managing DaaS concerns together with DaaS interface using XML.

in order to meet different requirements from different types of consumers. Currently, we are extending the Web services information model in [30] to cover also DQ/QoS and DSC and develop techniques for managing diverse types of DQ/QoS and DSC models. For example, the interface and concerns associated with the `CorteraCreditPulse` service are managed in two separate entries shown in Listing 1, where some data about DaaS concerns are visualized in Figure 2. Such data can be visualized in data composition tools to support the end-user to select DaaSs. Similar to XML-based data, RDF-based data is also developed.

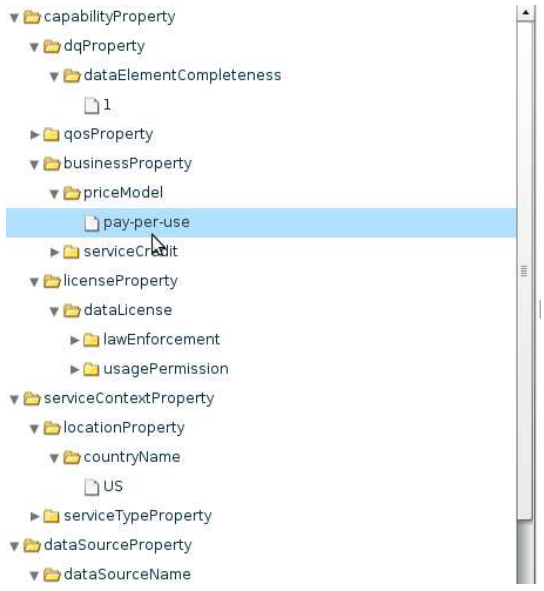


Fig. 2. Snapshot of exemplified data about DaaS concerns

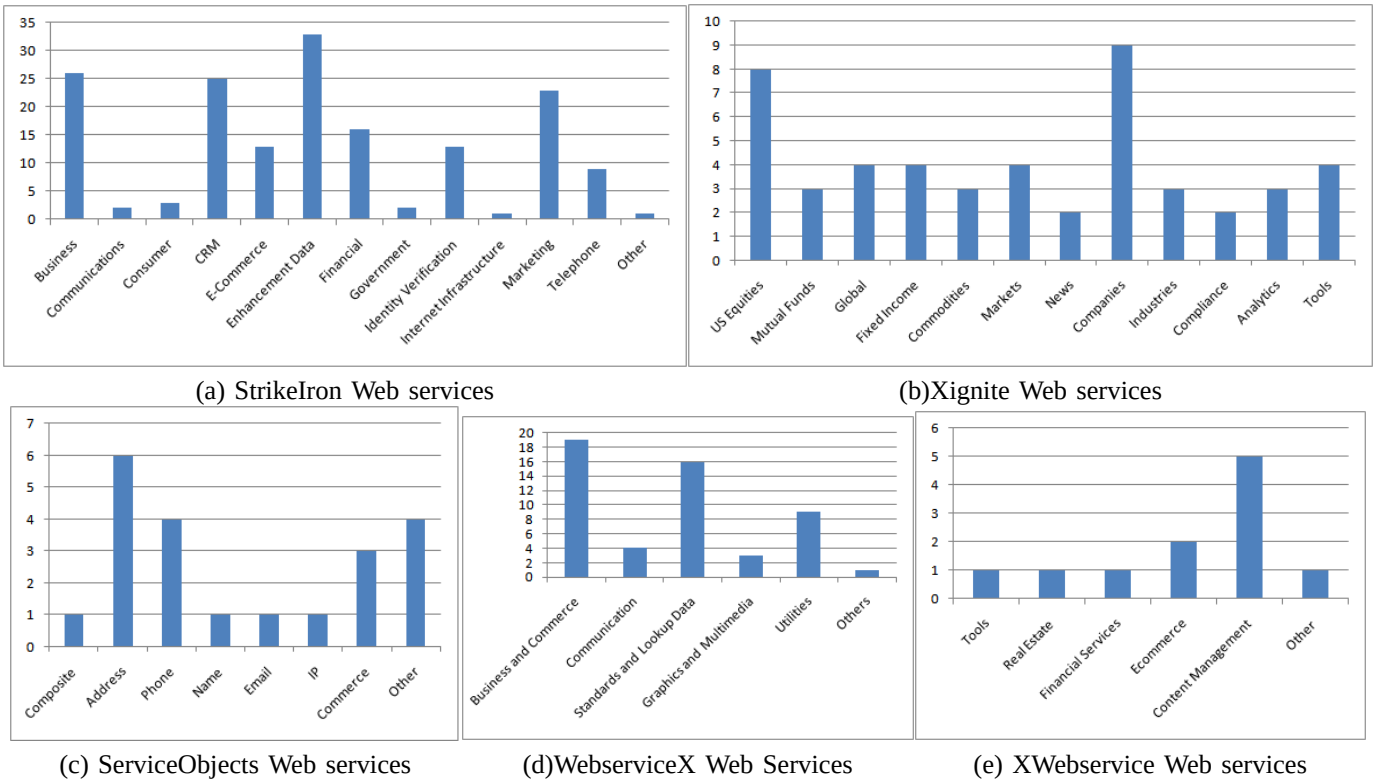
V. Evaluation of Current DaaS Publishing

As mentioned before, we are not aware of any providers that publish DaaS's concerns in a well-defined, standard form. To evaluate how service providers support DaaS concerns, we focused on DaaSs in the enterprise computing domain by utilizing information described in Web services categories of StrikeIron[12], Xignite[13], serviceobjects.NET[14], WebserviceX[15], and XWebServices[16]. These service providers arrange their services into Web directories in which services are described in HTML text. Thus, we have read DaaS descriptions and manually mapped their information to our models.

First, we played the role of an end-consumer to analyze how DaaSs are classified in DaaS providers. To focus on Read-only DaaSs, we selected these providers as they offer typical Read-only DaaSs, not infrastructure-based DaaSs. Figure 3 presents DaaS's classifications by analyzing Web descriptions provided by the five above-mentioned providers. It is obvious that different providers classify their services differently. For example, the validation of US addresses is provided by two services in StrikeIron and XWebServices, but they are in different categories. This difference in service classification prevents the automatic comparison and selection.

Second, we examined how existing DaaS providers support DaaS concerns. Besides the above-mentioned five providers, we also manually gathered information from AERS[17] and Amazon. Figure 4 presents how concerns associated with DaaSs are mentioned for 29 services from 7 providers. Overall, price models are well described, however, auditing, data lifecycle, usage permission, IPR, and legal enforcement are not

[12]http://www.strikeiron.com/strikeironservices.aspx
[13]http://www.xignite.com/Products/ProductDirectory.aspx
[14]http://www.serviceobjects.com/products/directory_of_web_services.asp
[15]http://www.webservicex.net/WCF/webServices.aspx
[16]http://www.xwebservices.com/Web_Services/
[17]http://www.researchadvanced.com



(a) StrikeIron Web services

(b)Xignite Web services

(c) ServiceObjects Web services

(d)WebserviceX Web Services

(e) XWebservice Web services

Fig. 3. DaaSs provided by StrikeIron, Xignite, ServiceObjects, WebServicesX and XWebService.

clearly mentioned. It is probably because that most services studied are Read-only DaaSs, for which auditing and data lifecycle concerns are not important to service consumers, and are for enterprise computing, for which price models are an important concern. However, this figure, resulted from the analysis of service description given by providers, shows that current DaaS providers focus too much on traditional service concerns, such as business and QoS metrics. They lack information on data quality, licensing and legal concerns which are critical in service-oriented data composition.

We also examined DaaSs in e-science by studying scientific DaaSs published through GBIF networks[18]. For data sources published, metadata about DaaS based on DiGIR protocols[19] can be obtained together with domain-specific metadata about data sources. However, information about DQ, IPR and QoS is missing. The information about usage permission and licensing is missing or is described in an unstructured format.

With this evaluation we examined possible concerns associated with DaaSs. The result of this evaluation actually guides our work in the design of the DaaS concerns model to include most relevant DaaS concerns which have not been modeled and published to support the (automatic) discovery of DaaSs and the on-demand utilization of DaaSs. Our approach can enhance this situation because it provides extensible models for describing and managing various types of DaaS concerns that are currently missing.

[18]http://www.gbif.net

[19]http://www.digir.net/schema/protocol/2003/1.0/digir.xsd

VI. Conclusions and Future Work

In this paper, we have analyzed concerns for data as a service (DaaS). We have found that various important concerns for DaaS have not been well described. Concerns associated with DQ, auditing, business, IPR and legal, and service location are important information that should be well-specified and publishable so that DaaS can be searched, evaluated and selected. Until now, research effort has focused on system perspective to make the data available through the service but not the concerns associated with the data provided by the service. Therefore, we have proposed and implemented a model for specifying and managing concerns of DaaS.

Our work is just the initial step in tackling issues related to the selection and utilization of DaaSs. Various future research activities have to be performed in order to consolidate the concepts of DQ, QoS, and data service contracts together for DaaS. Our future work will be focused on the study of concerns associated with CRUD DaaS and the discovery and selection of DaaSs based on studied concerns.

Acknowledgment

This work was supported by the Vienna Science and Technology Fund (WWTF), project ICT08-032. We thank G.R.Gangadharan, Andrea Maurino, Reinhard Pichler, and Vadim Savenkov for valuable discussions and suggestions.

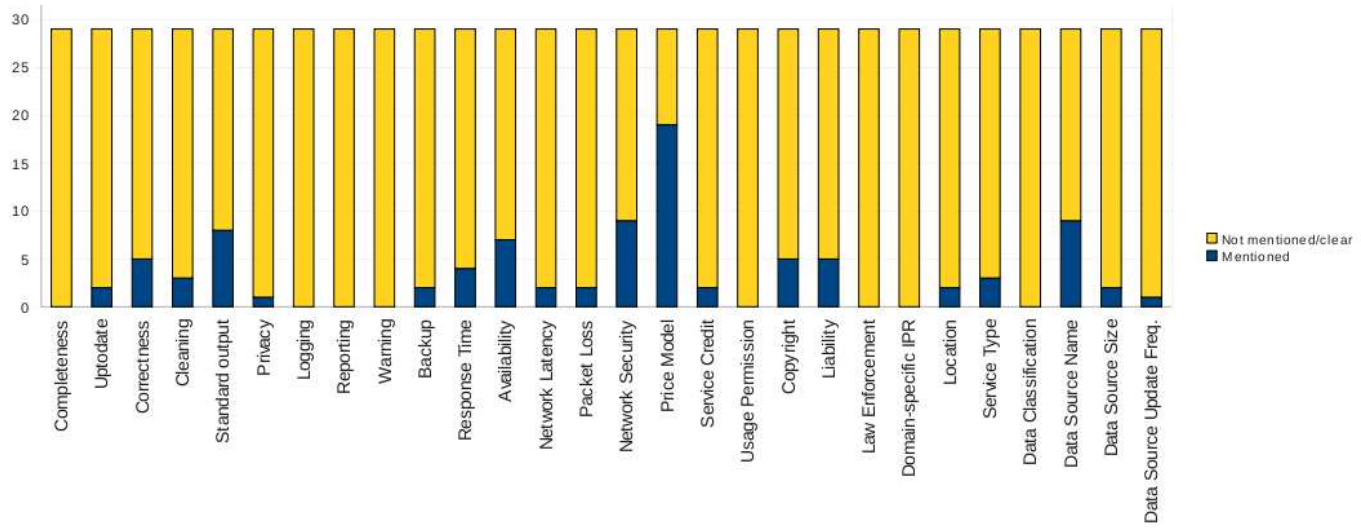


Fig. 4. Examples of concerns supported in DaaSs studied

REFERENCES

[1] K. Bennett, P. Layzell, D. Budgen, P. Brereton, L. Macaulay, and M. Munro, "Service-based software: the future for flexible software," *Asia-Pacific Software Engineering Conference*, vol. 0, p. 214, 2000.

[2] R. Buyya, C. S. Yeo, and S. Venugopal, "Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities," *High Performance Computing and Communications, 10th IEEE International Conference on*, vol. 0, pp. 5–13, 2008.

[3] A. Dan, R. Johnson, and A. Arsanjani, "Information as a service: Modeling and realization," *Systems Development in SOA Environments, 2007. SDSOA '07: ICSE Workshops 2007. International Workshop on*, pp. 2–2, May 2007.

[4] S. Ran, "A model for web services discovery with qos," *SIGecom Exch.*, vol. 4, no. 1, pp. 1–10, 2003.

[5] X. Wang, T. Vitvar, M. Kerrigan, and I. Toma, "A qos-aware selection model for semantic web services," in *ICSOC*, ser. Lecture Notes in Computer Science, A. Dan and W. Lamersdorf, Eds., vol. 4294. Springer, 2006, pp. 390–401.

[6] D. Butler, "Mashups mix data into global service," *Nature*, vol. 439, no. 7072, pp. 6–7, January 2006. [Online]. Available: http://dx.doi.org/10.1038/439006a

[7] "QoS for Web Services: Requirements and Possible Approaches, http://www.w3c.or.kr/kr-office/tr/2003/ws-qos/."

[8] B. Sabata, S. Chatterjee, M. Davis, J. J. Sydir, and T. F. Lawrence, "Taxomomy of qos specifications," in *WORDS '97: Proceedings of the 3rd Workshop on Object-Oriented Real-Time Dependable Systems - (WORDS '97)*. Washington, DC, USA: IEEE Computer Society, 1997, p. 100.

[9] W. E. Moen, E. L. Stewart, and C. R. McClure, "Assessing metadata quality: Findings and methodological considerations from an evaluation of the u.s. government information locator service (gils)," *Advances in Digital Libraries Conference, IEEE*, vol. 0, p. 246, 1998.

[10] D. M. Nichols, C.-H. Chan, D. Bainbridge, D. McKay, and M. B. Twidale, "A lightweight metadata quality tool," in *JCDL '08: Proceedings of the 8th ACM/IEEE-CS joint conference on Digital libraries*. New York, NY, USA: ACM, 2008, pp. 385–388.

[11] L. L. Pipino, Y. W. Lee, and R. Y. Wang, "Data quality assessment," *Communications of the ACM*, vol. 45, pp. 211–218, 2002.

[12] C. Batini, C. Cappiello, C. Francalanci, and A. Maurino, "Methodologies for data quality assessment and improvement," *ACM Comput. Surv.*, vol. 41, no. 3, 2009.

[13] G. R. Gangadharan and V. D'Andrea, "Licensing services: Formal analysis and implementation," in *ICSOC*, ser. Lecture Notes in Computer Science, A. Dan and W. Lamersdorf, Eds., vol. 4294. Springer, 2006, pp. 365–377.

[14] A. Keller and H. Ludwig, "The wsla framework: Specifying and monitoring service level agreements for web services," *J. Netw. Syst. Manage.*, vol. 11, no. 1, pp. 57–81, 2003.

[15] Committee on Licensing Geographic Data and Services, National Research Council, Ed., *Licensing Geographic Data and Services*. The National Academies Press, 2004.

[16] "ODRL-S," http://odrl.net/Profiles/Services/WD.html, Last access: 6 July 2009.

[17] P. Miller, R. Styles, and T. Heath, "Open data commons, a license for open data," 2008. [Online]. Available: http://CEUR-WS.org/Vol-369/paper08.pdf

[18] "The AERS API," http://www.researchadvanced.com/products/aers-api/., Last access: 8 October 2009.

[19] "Ebay data licenses," http://dataunison.com/ebay_data/licenses.php, Last access: 6 July 2009.

[20] G. Shankaranarayanan and Y. Cai, "Supporting data quality management in decision-making," *Decis. Support Syst.*, vol. 42, no. 1, pp. 302–317, 2006.

[21] "The platform for privacy preferences," http://www.w3.org/P3P, Last access: 6 July 2009.

[22] H.-L. Truong, R. Samborski, and T. Fahringer, "Towards a framework for monitoring and analyzing qos metrics of grid services," in *e-Science 2006*. IEEE Computer Society, 2006.

[23] O. Kelkar, J. Leukel, and V. Schmitz, "Price modeling in standards for electronic product catalogs based on xml," in *WWW '02: Proceedings of the 11th international conference on World Wide Web*. New York, NY, USA: ACM, 2002, pp. 366–375.

[24] "Ebay research apis," http://dataunison.com/ebay_research/api_uses.php, Last access: 6 July 2009.

[25] "Information lifecycle management," http://en.wikipedia.org/wiki/Information_Lifecycle_Management, Last access: 6 July 2009.

[26] "Understanding quality of service for Web services, http://www-128.ibm.com/developerworks/library/ws-quality.html," Jan 2002.

[27] "The ddfflus," http://www.ddfflus.com, Last access: 6 July 2009.

[28] "The dataflux," http://www.dataflux.com, Last access: 6 July 2009.

[29] F. D. Paoli, M. Palmonari, M. Comerio, and A. Maurino, "A meta-model for non-functional property descriptions of web services," in *ICWS*. IEEE Computer Society, 2008, pp. 393–400.

[30] M. Treiber, H.-L. Truong, and S. Dustdar, "SEMF - Service Evolution Management Framework," in *34th EUROMICRO Conference on Software Engineering andAdvanced Applications (SEAA), Special session on Quality and Service-Oriented Applications*. IEEE Computer Society, 2008.

[31] A. Dan and W. Lamersdorf, Eds., *Service-Oriented Computing - ICSOC 2006, 4th International Conference, Chicago, IL, USA, December 4-7, 2006, Proceedings*, ser. Lecture Notes in Computer Science, vol. 4294. Springer, 2006.