UNIVERSITY
OF TRENTO

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

38050 Povo – Trento (Italy), Via Sommarive 14
http://www.dit.unitn.it

ODRL/L(S): A Language for Service Licensing

G.R. Gangadharan, Vincenzo D'Andrea, Renato Iannella, and Michael Weiss

May 2007

Technical Report # DIT-07-027

ODRL/L(S): A Language for Service Licensing

G.R.Gangadharan[1], Vincenzo D'Andrea[1], Renato Iannella[2], and Michael Weiss[3]

[1] Department of Information and Communication Technology,
University of Trento, Via Sommarive, 14, Trento, 38050 Italy
{gr,dandrea}@dit.unitn.it
[2] National ICT Australia,
Level 19, 300 Adelaide Street, Brisbane, Queensland, 4000 Australia
renato@nicta.com.au
[3] School of Computer Science, Carleton university,
1125 Colonel By Drive, Ottawa, K1S 5B6, Canada
weiss@scs.carleton.ca

Abstract. Service licensing is a significant way to manage the rights between service consumers and service providers. Licensing, being the mechanism of technology transfer, is the common denominator for distribution of services associated with designing business strategies and relationships. The currently available service description standards and languages do not cover the critical clauses of licensing. This paper proposes a language ODRL/L(S), derived from the Open Digital Rights language (ODRL), to represent exclusively the licenses for services, in machine interpretable form.

1 Introduction

A license is an agreement between parties in which one party receives benefits by giving approximately equal value to the other party in exchange. Licensing [7] includes all transactions between the licensor and the licensee, in which the licensor agrees to grant the licensee the right to use and access the asset under predefined terms and conditions.

Though service oriented computing (SOC) continues to proliferate as a technology for connecting applications in a loosely coupled manner across organizational boundaries, the use of services for building successful business models is a different tale altogether. Services today are used within the enterprise level for integration purposes. Inter-enterprise integration is still far from reach. Nowadays, services offer programmatic interfaces. However, many available services are not even considered to provide relevant business value. Standing on the shoulders of giant industries, the future of services from the perspective of business is still unclear.

Beyond the requirement for standardizing infrastructure and industry standard interfaces of SOC, one of the significant reasons for unfulfilling [16] the promise of services is the lack of agreement binding service and business. A service level agreement (SLA) is viewed as an agreement that specifies a range of performance parameters within which the service is provided [19]. Current SLA and policies specifications for services (WSLA [17], WS-Agreement [2], SLANG [25], WSOL [26], WS-Policy [14]) define what to measure/ monitor and describe payments/ penalties. Generally, all the specifications focus on the quality of service issues and the terms and conditions agreed by the

provider and consumer. The majority of attention on SOC has been contemplated on its related technical standards and technology integration. Managerial issues and business strategy for implementing SOC have not been studied intensively.

Though there are examples of service licenses in practical use[4] (by Amazon, Google, Yahoo!), to the best of our knowledge, there appears to be no conceptualization of service licensing in general. The business and legal contractual information are not described at a detailed level by the services research community, either in industry or academia. Though the design of service licenses seems to be an initiative of the software industry, there is no active involvement in this topic by industry. One of the primary causes for this could be fear still faced by industries over the lack of standardization of technologies surrounding SOC. The need for a language defining both the internal business needs of an organization and its requirements on external services, and for a systematic way of linking them to business processes is proposed in [21]. As the mechanism of technology transfer, licensing addresses how a process is related to and affects business requirements and needs, describing the legal requirements. Licenses affect the design of business strategies and relationships, linking the business processes across boundaries.

The trend of software transforming to a service oriented paradigm demands for a new way of licensing for services [9]. Different types of licenses exist for software. As the nature of services differs significantly from traditional software and components, services prevent the direct adoption of software and component licenses. As services are being accessed and consumed in a number of ways, a spectrum of licenses suitable for services with differing license clauses can be definable. In this paper, we present a language, ODRL/$L(S)$, to express a service license so that any services can automatically interpret the licensing clauses.

The paper is organized as follows: In Section 2, we present a formalisation of licensing clauses specific to services for unambiguous definition of a license. Section 3 describes the complete syntax and semantics of the proposed language. In Section 4, we illustrate some examples of service licenses in ODRL/$L(S)$.

2 Formal Representation of Service Licenses

A service is represented by an interface part defining the functionality visible to the external world and a realization part implementing the interface [20]. A service could allow/ deny itself to be used/ accessed by other services with or without modification. For drafting machine readable licenses, the clauses of a service license should be unambiguous. We formalize the service licensing clauses for unambiguous definition of a license in the following manner:

2.1 Subject

The subject of the license relates to the definition of the service being licensed. This defines some related information about the service and may include a unique identification code for the service, service name, service location, and other relevant information.

[4] These service licenses are not machine interpretable.

2.2 Scope of Rights

The scope of rights of a service license reflect on what could be done with the service. We represent the set of operations of a service S by $O(S)$. Following are representative of typical scope of rights associated with services:

Adaptation: Adaptation signifies making a new independent service from an existing service interface. A service S' is an adapted service that uses the interface of another service S, if S and S' are independent in execution and $\{\text{O}(S')\} \neq \{\text{O}(S)\}$.

Composition: Service composition [1] is related to the implementation of a service whose internal logic involves the invocation of operations offered by other services. We refer to composition as the federation of a service with other remote services. In other words, the operations of a composite service relies on the availability of services being composed [13]. A service S is said to be composite if $\text{O}(S) \supset \{o_f : o_f \in \{\text{O}(S_i)\}\}$ and $\exists\, S \mid S_i, i = [1, .., n]$. o_f could be a single operation or a set of operations adding value addition by combining all or some of the operations of S_i.

Derivation: Derivation of a service, inspired by Free[5] and Open Source[6] Software (FOSS), is a new aspect of creating a new service from existing service, modifying the WSDL [6] interface and service realization. A service S' is said to be derived from S if $\text{O}(S') \supseteq \text{O}(S)$ on satisfying the following two conditions: (i) To exist S', S should be a Free/Open Service [8] and (ii) S and S' are independent in execution.

Attribution: A service may expect the attribution for its use by the other service in any of the forms. Thus, attribution is ascribing a service to the entity responsible for its creator. If a service S' uses a service S, then the attribution to S could be formally represented as $A(S') \supset A(S)$. The levelled attribution as in BSD[7] styled service licensing is represented by $A(S'') \supset A(S') \supset A(S)$ where the service S'' uses S' and S' uses S.

ShareAlike: A service, S, could expect the service, say S', being composed / derived to reflect the same terms and conditions of the S (Similar to Copyleft of GNU[8] or 'Sharealike' of CreativeCommons [11]). A service S' may expect another service S (which uses S') to have the same terms as of S'. In other words, $L(S) = L(S')$ where S' uses S and $L(S)$ describes the service license.

Non-Commercial Use: A service could allow/deny the services using it either for non-commercial purposes or for commercial purposes. A service S' could deny its use for commercial purposes. $N(S') = 1$ implies that an other service S could use S' if S is not commercial.

The detailed discussions and illustrations regarding the scope of rights for SOC could be found in [12].

2.3 Financial Terms

Service consumers make payments either as royalties or lump sum for the use of services. Generally royalties are based on per unit sales. In case of services, royalties can

[5] http://www.fsf.org/
[6] http://www.opensource.org/
[7] http://www.openbsd.org/policy.html
[8] http://www.gnu.org/copyleft/

be viewed as the amount for each use of the service. For a payment of p per use, a service consumer has to pay $R = n * p$ where n being the number of times of the service has been used. In this case, p can be renewed annually or over the life of license.

Lump sum payments are alternative method to royalties. Sometimes lump sum payments are also used in addition to royalties. In case of services, a lump sum payment can be paid by the service consumer before using the service (prepay) or at the later stage (postpay). For a lump sum payment m, a service consumer obtains the usage rights of a service. Paying the lump sum amount, the licensee obtains the rights to use the service over a period t (irrespective of number of times that services being invoked).

2.4 Warranties, Indemnities, and Limitation of Warranties

A warranty is a promise regarding the description of services and their quality, stated by the service provider. Warranties describe the functional and non-functional properties of services, providing as a way for the attraction and retention of consumers. Warranties are generally similar to the notions given in WSLA [17], defined by a negotiation process between the provider and consumer or in SLAng [25].

Indemnification clauses [5] specify the provision of defense by the licensor for the licensee if a third party sues the licensee, alleging that the licensee's use of the licensed software infringes or violates the third party's intellectual property rights. Limitation of liability clauses limit the liability of each of the parties under the license agreement.

2.5 Evolution

Service evolution is an implication of modifications by the service provider in functional and/or non-functional specifications of a service, represented by new releases or new versions [4, 3]. Service provider can offer a new version of a service, which could behave differently than its previous version. For a service consumer who consumes the previous service,the new version could cause significant unexpected behavior. In a similar way, the changes by the service provider in the physical infrastructures of the service could also cause the unexpected behaviors in the non-functional properties [9] and quality of service[10]. The changes in the service interface (service operations and parameters) and/or service realization could directly impact the composition and normal use of the service at the consumer's end. Further, a service changing it's service license could be referred as an evolved service. Thus, it becomes critical to consider the concept of service evolution also from the perspective of licensing.

Generally a technology license addresses the rights for a licensee over its future releases or versions [30]. The most of the commercial software and component licenses deny the rights of future versions to the consumers and also avoid agreements to agree

[9] Non-functional properties [27] of a service are the properties that a service offers (for provided services) and expects (for required services) other than the functional ones.

[10] Quality of services [22] is referred as a set of non-functional attributes that may impact the result of the service related to run-time, transaction support, configuration management, and cost and security. In a broad manner, quality of services can also imply both functional and non-functional aspects.

in the future. However, the releases of new versions of traditional software and components do not affect the behavior of functioning of their existing systems as these systems are independent from the new versions. But in case of services, the evolution affects the consumption and the composition of the existing systems. Hence, a service license is expected not to affect the consumption as well as the composition for the duration of agreement with the consumer. Further, the license of an evolved service could affect the compatibility with the existing service licenses.

3 Expression of Service Licenses

The Open Digital Rights Language (ODRL) [23] has following three core entities:

- *Assets:* a resource being licensed (identified uniquely).
- *Rights:* rules concerning permissions (the actual usages or activities allowed over the assets), constraints (limits to these permissions), requirements (the obligations needed to exercise the permission), and conditions (the specifications of exceptions that, if become true, expire the permissions and re-negotiation may be required).
- *Parties:* information regarding the service provider, consumer, broker etc.,

With these three entities, ODRL expresses offers (proposals from rights holders for specific rights over their assets) and agreements (contracts or deals between the parties, with specific offers). These core entities together allow for a wide and flexible range of ODRL expressions to be declared.

Our intentions for ODRL as an appropriate rights expression language for describing machine readable licensing agreements for services are as follows:

- ODRL is an open standard language, for expressing rights information.
- Being defined in XML, ODRL provides syntactic and semantic interoperability.
- ODRL is extensible and capable of incorporating specific clauses related to service licenses.
- Several business scenarios across various domains are expressable in ODRL.
- Being published in the World Wide Web Consortium (W3C), ODRL has a wide acceptance.
- ODRL is supported by several industries and consortia like the Dublin Core Metadata Initiative (DCMI)[11] and the Open Mobile Alliance (OMA)[12].

We extend (see Figure 1) the ODRL to express the clauses of service licensing, named as ODRL/$L(S)$, making a service license compatible with all the existing service standards. The core models and semantics of the language are as follows.

3.1 Subject

The Subject model of a service license could directly adopt the ODRL Context Model [23]. We directly reuse these semantics for the Subject.

[11] http://dublincore.org/
[12] http://www.openmobilealliance.org/

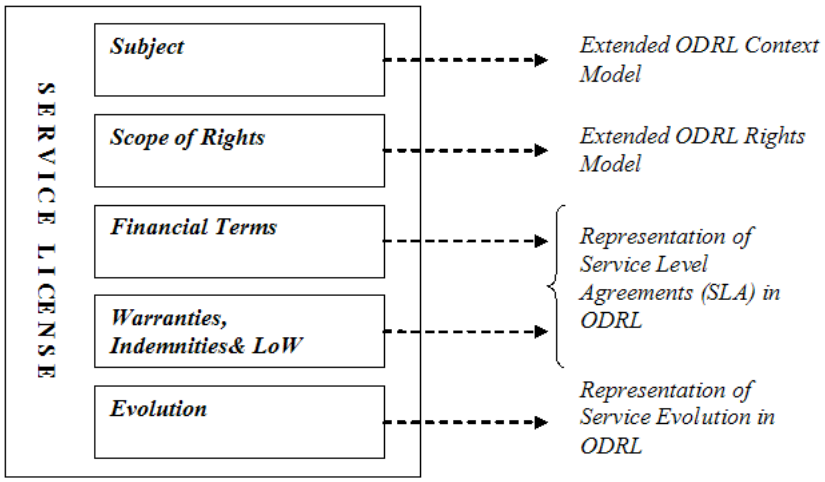

Fig. 1. Conceptual mapping of service licensing clauses in ODRL

3.2 Scope of Rights

The Scope of Rights is the extended from ODRL Permission, Requirement, and Constraint Models.

From the perspective of rights expression, composition is the right of execution with the right of interface modification. Derivation is the rights of allowing modifications to the interface as well as the implementation of a service. Further, derivation requires independent execution of the service where composition is dependent on the execution of services being composed. Adaptation refers to the right of allowing the use of interface only without any modifications.

ODRL/$L(S)$ reuses the concept of sharealike and non-commercial use of the creative commons profile [24]. Further, attribution to services is facilitated by the ODRL.

3.3 Financial Terms

A service license could also leverage the way of making revenue opportunities and generating new markets. Some of the feasible licensing models in line with the software models [10] for services are shown in Table 1.

Table 1. Service License Marketing Models and Description

License Models	**Description**
Utility Based	Allows charging based on service usage (pay-per-use)
Subscription Based	Subscribed for the accessing of services (for certain time period)
Evaluation	Access of trial versions of services

We adopt ODRL payment model for services to represent the Financial Terms model in ODRL/$L(S)$. However, Free/Open services could be represented without payment/cost models.

3.4 WIL (Warranties, Indemnities, and Limitation of Warranties)

The WIL model [13] (see Figure 2) defines warranties, indemnities and limitation of liabilities associated with services.

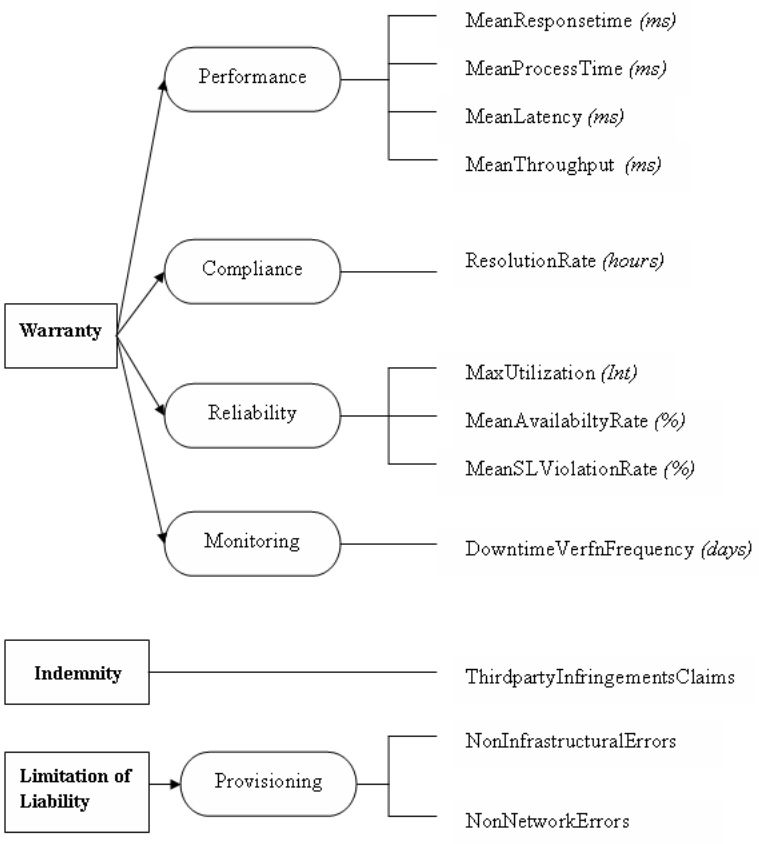

Fig. 2. ODRL/$L(S)$ WIL Model

[13] As warranties could describe several terms on quality of issues and performance parameters, we have attempted to capture certain important criteria. The captured terms are having the similar meanings as used commonly in Service Oriented Computing and conform to general standards of SOC. Generally, the legal terms of indemnities and limitation of liabilities are difficult to interpret in machine readable forms. However, these terms represented here are an endeavour for representing machine readable legal terms.

Following are the service warranties (based on the general definitions from [25, 15, 18])supported in ODRL/$L(S)$:

- *Mean Response Time*: The average time taken by the service for handling user requests.
- *Mean Process Time*: The average time taken by the service from the moment of invocation to the time of completion.
- *Mean Latency*: The time taken for data from the sending application to the receiving application including both transit time over the network, and processing time at both the source and destination.
- *Mean Throughput*: The rate at which service is delivered to the customer.
- *Resolution Rate*: The time period between the problem detection and problem settlement
- *Maximum Utilization*: The maximum service utilization allowed at which a service will perform within guaranteed response times and throughput.
- *Mean Availability Rate*: The average rate of availability of the service to be accessed.
- *Mean SL Violation Rate*: The average rate of violation of a service license due to the infringements over the agreed warranty levels
- *Downtime Verification Frequency*: The rate at which downtime is detected and verified by the provider.

The clause of indemnity $< thirdpartyinfringementsclaims >$ represents the statement provided by the licensor to the licensee to protect against the claims of a third party if any infringements over the intellectual property rights arise.

The limitation of liabilities are described in ODRL/$L(S)$ as follows:

- *NonNetworkErrors*: The license will not be liable for problems raising from the network.
- *NonInfrastructuralErrors*: infrastructure (e.g. Hosting environments)

3.5 Evolution

We define the Evolution model (see Figure 3) to specify the details of modifications by future releases or versions.

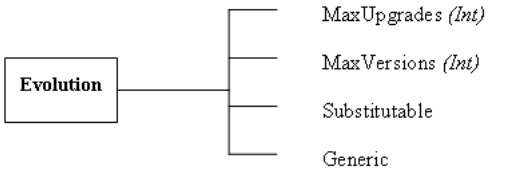

Fig. 3. ODRL/$L(S)$ Evolution Model

The evolution model elements are described as follows:

– *MaxUpgrades*: specifies the validity of service for the consumer allowed updations of services, till to which limit the license remains valid.
– *Maxversions*: specifies the allowed versions of services, till to which limit the license remains valid.
– *Substitutable*: indicates the service could be substituted by other similar service to which the licensee is allowed to use.
– *Generic*: indicates the service could be replaced by a service with broader scope and the licensee retains the rights over the generic service.

We have extended certain data dictionary semantics and defined certain new models to specifically handle the licensing clauses of a service in $ODRL/L(S)$ (see Tables 2 and 3).

Table 2. ODRL/$L(S)$ Data Dictionary Semantics and Schema for *Scope of Rights* Elements

Name	**Identifier**	**Description**
Derivation	derivation	The service may be derived.
`<xsd:element name="derivation" type="o-ex:permissionType" substitutionGroup="o-ex:permissionElement"/>`		
Adaptation	adaptation	The service interface (without modifications) can be used.
`<xsd:element name="adaptation" type="o-ex:permissionType" substitutionGroup="o-ex:permissionElement"/>`		
Composition	composition	The service may be composed.
`<xsd:element name="composition" type="o-ex:permissionType" substitutionGroup="o-ex:permissionElement"/>`		

4 Examples of Service Licenses

Consider a service having the following clauses of licensing:

1. The service can be composed with other services.
2. The service requires attribution.
3. The service provider describes the following warranties for the service:
   (a) Mean response time : 10 milliseconds
   (b) Mean process time : 7 milliseconds
   (c) Mean latency : 0.1 milliseconds
   (d) Mean SL violation rate : 0.1%
   (e) Mean Availability rate: 99.9%
   (f) Resolution rate : 1 hour
4. The service provider is not liable for any kinds of functional or non-functional errors in the infrastructure and networks of the service.
5. The service provider will defend the consumer from any action based on a claim that the use of the service in accordance with the given service license infringes any intellectual property rights of any third party.

Table 3. ODRL/$L(S)$ Data Dictionary Semantics and Schema for *WIL* and *Evolution* Elements

ODRL Element	**Identifier**	**Description**
Warranty	warranty	An implication for the quality of service as provided by the licensor.
`<xsd:element name="warranty" type="o-ex:requirementType" substitutionGroup="o-ex:requirementElement"/>`		
Indemnification	indemnity	A defense for the licensee against third party's allegations.
`<xsd:element name="indemnity" type="o-ex:requirementType" substitutionGroup="o-ex:requirementElement"/>`		
Limitation of Liability	limitationofliability	A limitation of liability of the service terms.
`<xsd:element name="limitationofliability" type="o-ex:requirementType" substitutionGroup="o-ex:requirementElement"/>`		
Evolution	evolution	Use of service over time with changes in physical environment and functions.
`<xsd:element name="evolution" type="o-ex:permissionType" substitutionGroup="o-ex:permissionElement"/>`		

6. The service provider offers the right to use the service upto three versions.
7. The service could be replaced by another service with broader scope in functionality (without affecting the existing functionality) by the service provider to which the licensee is allowed to use.

The above licensing clauses could be represented in ODRL/$L(S)$ as follows:

```
1   <?xml version=1.0 encoding=UTF-8?>
2   <o-ex:rights xmlns:o-ex=http://odrl.net/1.1/ODRL-EX
           xmlns:o-dd=http://odrl.net/1.1/ODRL-DD>
3    <o-ex:offer>
4       <o-ex:asset>
5          <o-ex:context>
6             <o-dd:uid>http://www.gr.com/../listcurrdaily()</o-dd:uid>
7           </o-ex:context>
9       </o-ex:asset>
7     <o-ex:permission>
8        <ls:composition/>
9     </o-ex:permission>
10    <o-ex:requirement>
11       <o-dd:attribution/>
12    </o-ex:requirement>
27    <o-ex:requirement>
28       <o-dd:peruse>
29          <o-dd:payment>
30             <o-dd:amount o-dd:currency=''EUR''>9.99</o-dd:amount>
31          </o-dd:payment>
32       </o-dd:peruse>
```

```
33    </o-ex:requirement>
34    <o-ex:requirement>
35       <ls:warranty>
36          <ls:performance>
37             <ls:meanresponsetime> 10 </ls:meanresponsetime>
38             <ls:meanprocesstime> 7 </ls:meanprocesstime>
39             <ls:meanlatency> 0.1 </ls:meanlatency>
40          </ls:performance>
41          <ls:reliability>
42             <ls:meanslviolationrate> 0.1 </ls:meanslviolationrate>
43             <ls:meanavailabilityrate> 99.9 </ls:meanavailabilityrate>
44          </ls:reliability>
45          <ls:compliance>
46             <ls:resolutionrate > 1 </ls:resolutionrate>
47          </ls:compliance>
48       </ls:warranty>
49       <ls:indemnity>
50          <ls:thirdpartyinfringementsclaims/>
51       </ls:indemnity>
52       <ls:limitationofliability>
53          <ls:provisioning>
54             <ls:nonnetworkerrors/>
55             <ls:noninfrastructuralerrors/>
56          </ls:provisioning>
57      </ls:limitationofliability>
58    </o-ex:requirement>
59    <o-ex:permission>
60       <ls:evolution>
61          <ls:maxversions> 3 </ls:maxversions>
62          <ls:generic/>
63       </ls:evolution>
64    </o-ex:permission>
65  </o-ex:offer>
```

4.1 Feature Level Service License

Generally a service license is associated with a service. However, a service could have several operations and these operations may differ in their usage terms. It is very common in the field of service oriented computing that a service operation or a set of operations could be offered free in terms of cost and another operation requires the payment for usage. Thus, it becomes mandatory for a service license to describe these operational level terms.

We model the operational level licensing by the concept of features in services [28, 29]. Each feature is an operation or a bundle of closely related operations that can be invoked as described in the Figure. The concept of features of a service are implemented in ODRL/$L(S)$ by adopting the Context model of ODRL.

Consider a bank service B that facilitates buying and selling of foreign currencies. B offers a free service operation, say B_1, for the consumers to quickly and easily calculate foreign exchange (forex) conversion based on daily rates offered by B. B also keeps a historical data records on forex, accessible via an other service operation, say B_2.

The service B offers B_1 under the following terms and conditions for its usage:

1. The service could be composable for daily forex rates.
2. The service does not require any fees for the usage.
3. The service requires attribution for the usage by other services.

As the value of the service lies in the data provided, B provides B_2 for two euros per use of the service feature. Also, B_2 allows composition and requires attribution for the usage.

The above clauses could be represented in ODRL/$L(S)$ as follows:

```
<?xml version=1.0 encoding=UTF-8?>
<o-ex:rights xmlns:o-ex=http://odrl.net/1.1/ODRL-EX
       xmlns:o-dd=http://odrl.net/1.1/ODRL-DD>
 <o-ex:offer>
    <o-ex:asset>
       <o-ex:context>
          <o-dd:uid>http://www.gr.com/../listcurrdaily()</o-dd:uid>
          <o-dd:name>OPERATON LIST CURRENCY DAILY RATES</o-dd:name>
       </o-ex:context>
    </o-ex:asset>
    <o-ex:permission>
       <ls:composition/>
    </o-ex:permission>
    <o-ex:requirement>
       <o-dd:attribution/>
    </o-ex:requirement>
 </o-ex:offer>
 <o-ex:offer>
    <o-ex:asset>
       <o-ex:context>
           <o-dd:uid>http://www.gr.com/../listcurrdata()</o-dd:uid>
           <o-dd:name>OPERATON LIST CURRENCY DATA</o-dd:name>
        </o-ex:context>
    </o-ex:asset>
    <o-ex:requirement>
       <o-dd:attribution/>
    </o-ex:requirement>
    <o-ex:permission>
       <ls:composition>
    </o-ex:permission>
    <o-ex:requirement>
       <o-dd:peruse>
          <o-dd:payment>
             <o-dd:amount o-dd:currency=''EUR''>2.00</o-dd:amount>
          </o-dd:payment>
        </o-dd:peruse>
    </o-ex:requirement>
 </o-ex:offer>
</o-ex:rights>
```

5 Concluding Remarks

In a dynamic market environment, the usage of services is governed by bilateral agreements that specify the terms and conditions of using and provisioning the services. A service license describes the terms and conditions for the use and access of the service, that services could be able to interpret. In this paper, We have proposed a language ODRL/$L(S)$ for expressing service license clauses in machine readable way.

References

1. P. A. Alvarez, J. A. Baares, and M. J. Ezpeleta. Approaching Web Service Coordination and Composition by means of Petri Nets. In *Proceedings of the 3rd International Conference on Service Oriented Computing*, pages 185–197, 2005.
2. A. Andrieux, K. Czajkowski, A. Dan, K. Keahey, H. Ludwig, T. Nakata, J. Pruyne, J. Rofrano, S. Tuecke, and M. Xu. Web Services Agreement Specification (WS-Agreement) Version 2005/09. www.gridforum.org, 2005.
3. L. Aversano, M. Bruno, M. D. Penta, A. Falanga, and R. Scognamiglio. Visualizing the evolution of Web Services using Formal Concept Analysis. In *Proceedings of the International Workshop on Principles of Software Evolution (IWPSE'05)*, 2005.
4. G. Canfora. Software Evolution in the Era of Software Services. In *Proceedings of the 7th International Workshop on Principles of Software Evolution (IWPSE'04)*, pages 9–18, 2004.
5. A. Chavez, C. Tornabene, and G. Wiederhold. Software Component Licensing: A Primer. *IEEE Software*, 15(5):47–53, 1998.
6. E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana. Web Services Description Language (WSDL) 1.1. http://www.w3.org/TR/wsdl, 2001.
7. W. Classen. Fundamentals of Software Licensing. *IDEA: The Journal of Law and Technology*, 37(1), 1996.
8. V. D'Andrea and G. R. Gangadharan. Licensing Services: An "Open" Perspective. In *Open Source Systems (IFIP Working Group 2.13 Foundation Conference on Open Source Software), Vol. 203, Springer Verlag*, pages 143–154, 2006.
9. V. D'Andrea and G. R. Gangadharan. Licensing Services: The Rising. In *Proceedings of the IEEE Web Services Based Systems and Applications (ICIW'06), Guadeloupe, French Caribbean*, pages 142–147, 2006.
10. D. Ferrante. Software Licensing Models: What's Out There? *IEEE IT Professional*, 8(6):24–29, 2006.
11. B. Fitzgerald and I. Oi. Free Culture: Cultivating the Creative Commons. *Media and Arts Law Review*, 2004.
12. G. R. Gangadharan and V. D'Andrea. Licensing Services: Formal Analysis and Implementation. In *Proceedings of the Fourth International Conference on Service Oriented Computing (ICSOC'06), Chicago, USA*, pages 365–377, 2006.
13. R. Hamadi and B. Benatallah. A Petri Net-based Model for Web Services Composition. In *Proceedings of the Fourteenth Australasian Database Conference on Database Technologies*, pages 191–200, 2003.
14. Jeffrey Schlimmer (Ed.). Web Services Policy Framework (WS-Policy). http://www-128.ibm.com/developerworks/webservices/library/ specification/ws-polfram/, 2004.
15. A. Keller and H. Ludwig. The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services. *Journal of Network and Systems Management*, 11(1), 2003.
16. H. Kreger. Fulfilling the Web Service Promise. *Communications of the ACM*, pages 29–34, June 2003.

17. H. Ludwig, A. Keller, A. Dan, R. King, and R. Franck. Web Service Level Agreement (WSLA) Language Specification. IBM Coporation, 2003.
18. A. Mani and A. Nagarajan. Understanding Quality of Service for Web Services. http://www-128.ibm.com/developerworks/library/ws-quality.html?n-ws-1172, 2002.
19. N. Muller. Managing Service Level Agreements. *International Journal of Network Management*, 9:155–166, 1999.
20. M. Papazoglou and D. Georgakopoulos. Service Oriented Computing. *Communications of the ACM*, 46(10):25–28, 2003.
21. M. Papazoglou, P. Traverso, S. Dustdar, F. Leymann, and B. Kramer. Service Oriented Computing Research Roadmap. In *Dagstuhl Seminar Proceedings 05462 (SOC)*, 2006.
22. S. Ran. A Model for Web Services Discovery with QoS. *ACM SIGecom Exchanges*, 4(1):1–10, 2003.
23. Renato Iannella (Ed.). Open Digital Rights Language (ODRL) Version 1.1. http://odrl.net/1.1/ODRL-11.pdf, 2002.
24. Renato Iannella (Ed.). ODRL Creative Commons Profile. http://odrl.net/Profiles/CC/SPEC.html, 2005.
25. J. Skene, D. Lamanna, and W. Emmerich. Precise Service Level Agreements. In *Proc. of 26th Intl. Conference on Software Engineering (ICSE)*, 2004.
26. V. Tosic, B. Pagurek, K. Patel, B. Esfandiari, and W. Ma. Management Applications of the Web Service Offerings Language. In *Proc. of the 15th CAiSE*, 2003.
27. G. Wang and A. MacLean. Software Components in Contexts and Service Negotiations. In *CBSE Workshop*, 1999.
28. M. Weiss. Feature Interactions in Web Services. In *Proceedings of the Seventh International Workshop on Feature Interactions in Telecommunication and Software Systems (FIW'03)*, pages 149–156, 2003.
29. M. Weiss and B. Esfandiari. On Feature Interactions Among Web Services. *International Journal of Web Services Research*, 2(4):21–45, 2005.
30. World Intellectual Property Organization. Successful Technology Licensing. WIPO Publishers, Geneva, Switzerland, 2004.