DEVELOPING OPEN TRAVEL
ALLIANCE-BASED ONTOLOGY OF GOLF

Agnieszka Cieslik

Dep. of Mathematics and Information Technology, Warsaw University of Technology, Warsaw, Poland

Maria Ganzha, Marcin Paprzycki

Systems Research Institute Polish Academy of Science, Warsaw, Poland

Keywords: OTA, ontology, golf, agent-based system, travel support system.

Abstract: For usage of ontologies to become more prevalent, not only new ontologies have to be created to represent the world, but also ontological support for existing domain-specific real-world standards has to be provided. One of such standards that gains popularity is the Open Travel Alliance (OTA) messaging system that defines, among others, the way that entities should communicate about golf as a travel-related entity. The aim of this paper is to outline our efforts leading toward creating an ontology of golf that would match the OTA messaging specification.

1 INTRODUCTION

Our current work is focused on developing an agent-based system that provides comprehensive support for needs of a traveler. To this effect we have been involved in two related projects. First project is the design and implementation of a model agent-based e-commerce system (see (Bădică et al., 2007b; Bădică et al., 2007a), and references to our earlier work contained there). This project is focused on utilization of price negotiations in a comprehensive e-commerce scenario, and was extended to facilitate possibility of airline ticket auctioning (Vukmirovic et al., 2006c; Vukmirovic et al., 2006a; Vukmirovic et al., 2006b; Szymczak et al., ; Vukmirovic et al., 2007). Second, we work on creation of an agent based *Travel Support System (TSS)* (Salam and Stevens, 2006; Gawinecki et al., 2005b; Gordon and Paprzycki, 2005). In the *TSS*, travelers are going to find, among others, personalized information about restaurants, hotels, historical points of interest, local weather etc. The main idea of the *TSS* is to utilize a central repository of semantically demarcated travel data, and use it to facilitate personalized information provisioning (Gawinecki et al., 2005b; Gawinecki et al., 2005c).

These two projects (airline ticket auctioning system and Travel Support System) were developed separately. It was only in (Vukmirovic et al., 2006c) where we have looked into issues involved in their possible merger. The key consideration was related to the fact that ontologically demarcated data is the central component of the *TSS* (Salam and Stevens, 2006; Gawinecki et al., 2005b; Gordon and Paprzycki, 2005). Therefore, the two projects were conceptually merged through development of a common travel ontology. First, still within the *TSS*, we have developed and then merged ontologies of hotels and restaurants. These two ontologies were developed utilizing a very pragmatic approach to ontology building and were based on the *concept of a hotel* as represented in travel-related WWW sites and *concept of a restaurant* as proposed in the ChefMoz project, respectively (Gawinecki et al., 2005a; Gordon et al., 2005).

In the airline ticket auctioning system we have pursued an even more reality-grounded approach. Here, of particular importance is the fact that all air-travel related activities are regulated by IATA, the global air-travel governing organization. This fact is not often recognized by non-practitioners and as a result, as shown in (Vukmirovic et al., 2006c; Vukmirovic et al., 2006a), most of existing air travel ontologies, being typically developed as an academic exercise, lack specific features that are required by IATA as well as a number of additional features that are needed in day-to-day operation of existing air-

Table 1: Summary of OTA golf messages.

Message type	List of fields
OTA_GolfCourseSearchRQ—request for course information; used to find golf courses that satisfy a given set of criteria	Architect, ADAChallenged, Slope, Metal Spikes, Caddies available, Yardage, Personal Carts Permitted, Grass Type, Singles Confirmed
OTA_GolfCourseSearchRS—list of courses that meet the requested criteria; if attribute is specified as Required (set to *Yes*) then only courses that meet criteria will be returned; if Required attribute is *No* a course that does not meet a given criteria may be included in the list	Golf Course ID, Golf Course address, Contact information—telephone number, List of requested criteria
OTA_GolfCourseAvailRQ—requests information about availability of a specific golf course	Golf Course ID, Tee Time—start and end date, Number of golfers, Number of holes, Maximum price for one person
OTA_GolfCourseAvailRS—provides information about field availability	Golf Course ID, Tee Time, Number of golfers, Number of holes, Maximum price for one person, List of fees. Fee has name, information about amount, currency and taxes
OTA_GolfCourseResRQ—requests a reservation of a given golf course	Information about person who makes reservation (first and last name, address, date of birth, telephone number), Mean of payment, Date of game, Number of golfers, Number of carts, List of fees
OTA_GolfCourseResRS—confirms (or denies) reservation of a given golf course	Reservation ID, Information about person who makes reservation (first and last name, address, date of birth, telephone number), Mean of payment (credit cart information), Date of game, Number of golfers, Number of carts, List of fees, Information concerning cancellation penalties and date and time by which a cancellation must be made

lines. Separately, the Open Travel Alliance (OTA) (OTA, a) has proposed a set of messages that organizations involved in travel-related activities can to use to meaningfully communicate about travel entities such as flights or golf course reservations. Interestingly, one can infer that OTA messages may be currently on the way to become an industry-wide standard. For instance, a number of travel-related businesses (e.g. some of US-based airlines) have accepted OTA messages in their business practices. Let us also note that, from the technical perspective, OTA messaging is an open standard that consists of a set of XML-demarcated messages. Obviously, by the very nature of their design and function, these messages represent a certain conceptualization of the "world of travel." However, they *do not* explicitly define an ontology. In (Vukmirovic et al., 2007) we have proposed how OTA air-travel-related messages can be used as a basis for development of an ontology of air-travel. Furthermore, we have utilized IATA manuals, and practical knowledge of a member of our team, in an attempt to assure compliance of the proposed ontology with air-travel regulations and practices. Finally, the proposed air-travel ontology was successfully merged with the ontology of restaurant and hotel. The results (complete ontology of restaurants, hotels and air travel) can be found at (tss,). The aim of this paper is to describe how, in an attempt at extending the *TSS* ontology and functionality, the OTA golf messaging has been turned into an *OTA ontology of golf*.

To this effect we proceed as follows. In the next section we describe the OTA golf messages. We follow with the specification of concepts that have to be represented in the *OTA golf ontology*. Next we analyze which concepts should be re-used from the *TSS* ontology (in order to allow later merging of these ontologies). Finally we present the most important parts of the proposed *OTA ontology of golf*.

2 OTA GOLF MESSAGES

Let us now briefly describe golf-related OTA messages (description is based on (OTA, b)). As in the case of all OTA messages, they come in pairs. There is alway a request (RQ) message (a query) and, corresponding to it, a response (RS) message. For the golf-course-related communications the OTA standard identifies three pairs of messages summarized in Table 1. In short, these messages provide the following functionalities: (1) ability to find a golf course with specific characteristics, (2) check if a course of interest is available at a specific time and under a specific set of conditions (e.g. below a certain maximum price), and (3) make an actual reservation.

To illustrate the specific form that OTA messages take, in Figure 1 we present an example on an *OTA_GolfCourseAvailRQ* message (based on (OTA, b)). In this message two friends specify that they would like to play golf on October 31st, and the re-

quested tee-off time is to be between 1:00 and 2:30. They are interested in playing at a specific golf course with the identifier FL3421. The maximum price that they are willing to pay for 18 holes is $80.00 per person. The aim of this message is to find if the FL3421 course is available at a given time and if the price condition is satisfied.

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA\_GolfCourseAvailRQ xmlns=
   ''http://www.opentravel.org/OTA/2003/05''
xmlns:xsi=
''http://www.w3.org/2001/XMLSchema-instance''
xsi:schemaLocation=
''http://www.opentravel.org/OTA/2003/
        05 OTA_GolfCourseAvailRQ.xsd''
  EchoToken="12345"
  TimeStamp="1003-05-31T13:20:00-05:00"
  Target="Production" Version="1.001"
  SequenceNmbr="123456">
<GolfCourseTeeTimes CourseID="FL3421">
<GolfCourseTeeTime Start="2003-10-31T13:00:00"
    End="2003-10-31T14:30:00"
    NumberOfGolfers="2"
    NumberOfHoles="18"
    NumberOfTimes="1"
    MaxPrice="80.00"
    CurrencyCode="USD">
</GolfCourseTeeTime>
</GolfCourseTeeTimes>
</OTA\_GolfCourseAvailRQ>
```

Figure 1: Example of an OTA golf course availability query message.

In response to the *OTA_GolfCourseAvailRQ* message depicted in Figure 1, the *OTA_GolfCourseAvailRS* message presented in Figure 2 could have been received. This response indicates that the requested FL3421 course is available on October 31st, with the tee-time between 1:00 and 2:30. Furthermore, the green fee is $70.00 per person, while the cart fee is $10.00 per person.

As a result, a *GolfCourseResRQ* message could be send, requesting a reservation at a specific time. This message would then have to be followed by a *GolfCourseResRS* message that would confirm the reservation.

3 DESIGNING THE ONTOLOGY—PRELIMINARY CONSIDERATIONS

Now, we can discuss how OTA golf-course-related messages can be used as a basis for the development of an *OTA golf course ontology*. Analysis of OTA

```
<?xml version="1.0" encoding="UTF-8"?>
<OTA\_GolfCourseAvailRS
  xmlns=''http://www.opentravel.org/OTA/2003/05''
  xmlns:xsi=''http://www.w3.org/2001/
           XMLSchema-instance''
  xsi:schemaLocation=
  ''http://www.opentravel.org/OTA/2003/05
           OTA_GolfCourseAvailRS.xsd''
    EchoToken="12345"
    TimeStamp="2003-05-31T13:20:00-05:00"
    Target="Production"
    Version="1.001"
    SequenceNmbr="123456">
<Success/>
<UniqueID Type="4" ID="FL3421">
</UniqueID>
<GolfCourseTeeTimes>
<GolfCourseTeeTime
    Start="2003-10-31T13:00:00"
    End="2003-10-31T14:30:00"
    NumberOfGolfers="2"
    NumberOfHoles="18"
    NumberOfTimes="1"
    MaxPrice="80.00"
    CurrencyCode="USD">
<Fees>
   <Fee TaxInclusive="true"
   Amount="70.00" CurrencyCode="USD">
   <Description Name="GreensFee">
   <Text Formatted="false" Language="EN"/>
   </Description>
</Fee>
<Fee TaxInclusive="true"
   Amount="10.00" CurrencyCode="USD">
  <Description Name="CartFee">
   <Text Formatted="false" Language="EN"/>
  </Description>
</Fee>
</Fees>
</GolfCourseTeeTime>
</GolfCourseTeeTimes>
</OTA\_GolfCourseAvailRS>
```

Figure 2: Example of an OTA golf course availability response message.

golf related messages showed us a need for defining of two core concepts: *Golf Course* and *Golf Course Tee Time*. The first one is going to uniquely identity a given golf course and its features. This concept is related directly to the first of OTA message pairs, where a golf course with a specific set of features is sought. This concept is similar to the restaurant and hotel concepts from the *TSS* ontology of travel. All three concepts define a *static* object and a set of its features. The second concept will define all information that is necessary for completing a reservation of a golf course. Thus, the *Golf Course Tee Time* concept defines *dynamic* characteristics of a static object specified in the *Golf Course* object. The *Golf Course*

concept and its features are represented in Table 2.

Table 2: Golf Course concept and its features.

Class GolfCourse	
Course ID	ID originates from the OTA_GolfCourseSearchRS message; can be used for getting information about golf course availability and for making reservations
Address	Address of golf course
Contact	Contact information (e.g. telephone number)
Features	List of golf course features

In Table 3 we list features that constitute the necessary information to define the *Golf Course Tee Time* concept. Since the "names of features" are self-explanatory, we do not define them further. It is of value to compare the list of features defining the *Golf Course Tee Time* concept with the content of the golf course availability querying (*Golf-CourseAvailRQ*) message presented above.

Table 3: Golf Course Tee Time concept and its features.

Class GolfCourseTeeTime
Course ID
Start date and time
End date and time
Price
Max price for one person
Number of holes
Number of golfers
Number of games
List if fees

After identifying two concepts that are going to constitute the core of the *OTA golf ontology*, we have to address the following question: how does this ontology relate to the existing *TSS* ontology. In other words, we have to establish which already existing / defined concepts can (and should) be immediately re-used in the new ontology.

3.1 Common Concepts with the *TSS* Ontology

Ontology re-use is one of important concepts in ontological engineering (Fensel, 2003). Therefore we should re-use as much as possible of the existing ontologies (and we have done so within the *TSS* ontology). Furthermore, the *OTA golf course ontology*, should be integrable with the *TSS* ontology and thus utilize immediately as many of its concepts as possible. Thus we have identified existing concepts that could be re-used; let us list them here.

Outdoor Location—geographical location is one of common features of objects populating the *TSS*

(restaurant, hotel, airport). Obviously, it is also a concept that is directly related to the golf course. The *OutdoorLocation* concept (class) describes geographical location through a set of geographical properties, such as: street address, country, city/town, region, zip code, reference points or location description (see the *TSS* ontology available at (tss,) for a complete listing). In the *TSS* ontology, the *Hotel*, the *Restaurant* and the *Airport* classes are sub-classes of the *OutdoorLocation* class. Therefore, the proposed class *GolfCourse* should also become a subclass of *OutdoorLocation*. This is a natural decision as the *Golf Course* should be an object of the same "stature" as the other objects mentioned here.

Discounts—in general, is the class that contains the following information:

- code of the particular discount,
- amount of reduction of the base-price,
- short description of the discount policy.

However, when dealing with air travel support we have realized that IATA defined specific air travel discount codes. Therefore, the question has arisen: how to integrate these with hotel and restaurant discount codes (including both OTA-specific and general discounts—these omitted in the OTA specification). For the purpose of integration of ontologies, specific discounts codes were distinguished and defined as subclasses of the general *DiscountTypes* class. Specifically, in the *TSS* there exist three classes defining possible discounts:

- *OTADiscountTypes*—discount types originating from the OTA specification
- *IATADiscountTypes*—discount types originating from the IATA specification
- *DiscountTypes*—general class; all discount types

Obviously, classes *OTADiscountTypes* and *IATADiscountTypes* are subclasses of *DiscountTypes*. Note that since the proposed ontology of golf is based on OTA messages, the discount concepts used here belong to the *OTADiscountTypes* class.

The remaining common parts between the *TSS* and the *OTA golf course* ontologies are:

- *MeanOfPayment*—concept defining possible mean of payment (e.g cash, credit card, check, etc.)
- *AdressRecord*—class that in the *TSS* ontology describes the address
- *Currency*—defines what is the currency that the fees are in
- *FareTax*—information about taxes

- *Contacts*—possible ways of contacting an entity (e.g. telephone number)
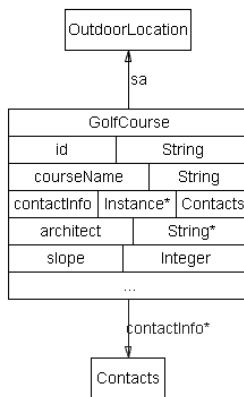
4 THE *OTA GOLF ONTOLOGY*

Figure 3: Golf Course concept; graphical representation.

Based on the above considerations we can now briefly discuss the most important aspects of the proposed *OTA golf ontology*. Let us start from definitions of the two basic classes, the main class defining the concept of the *Golf Course* is presented in Figure 4 (in the form of the standard RDF notation) and in Figure 3 (in the visual representation).

As can be easily seen in the graphical representation, the *GolfCourse* class is a subclass of *Outdoor-Location* and utilizes the *Contacts* concept (from the *TSS* ontology). In its definition we use strings for: *id*, *courseName*, *architect*; and an integer for the *slope*.

The second concept that belongs to the core of the *OTA golf course ontology is Golf Course Tee Time* is presented in Figure 5 (in the RDF notation) and in Figure 6 in its graphical representation.

Observe that while the *GolfCourseTeeTime* class is relatively simple itself (it consists of strings for: *startDate*, *endDate* and *golfCourseID*; float for *max-Price*; and an integer for *numberOfTimes*), it utilizes also a fairly extensive concept of a *Fee*. This points out to the fact that in addition to the two basic concepts (classes *GolfCourse* and *GolfCourseTeeTime*) we had to define the following additional concepts / classes:

- *Price*—concept of price (includes: amount, taxes, currency, etc.)
- *Fee*—concept of fee (e.g. green fee, cart fee)
- *Description*—additional descriptions that are needed for the traveler to be able to effectively utilize the information provided by the system

```
@prefix rdf:
<http://www.w3.org/1999/02/22-rdf-syntax-ns\#>.
@prefix rdfs:
  <http://www.w3.org/2000/01/rdf-schema\#>.
@prefix xsd:
  <http://www.w3.org/2001/XMLSchema\#>.
@prefix base: <Golf/GolfCourse\#>.
@prefix loc: <Location/OutdoorLocation\#>.
@prefix pmt: <Payment/MeanOfPayment\#>.
@prefix phc: <Contacts/Contacts\#>.
@prefix rsv:
  <HotelInfrastructureCodes/ReservationCodes\#>.
base:GolfCourse a rdfs:Class ;
          rdfs:subClassOf loc:OutdoorLocation ;
rdfs:comment ''Used for city and
        geographical location description''.
base:id a rdf:Property ;
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:string .
base:name a rdf:Property ;
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:string .
base:contactInfo a rdf:Property ;
     rdfs:comment ''Contact information.'';
     rdfs:domain base:GolfCourse ;
     rdfs:range phc:Contacts .
base:architect a rdf:Property ;
rdfs:comment ''Information about architect,
    who design this golf course'';
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:string .
base:slope a     rdf:Property ;
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:integer .
base:availCaddy a rdf:Property ;
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:boolean .
base:permCart a rdf:Property ;
rdfs:comment ''Information if personal
    carts are permitted''
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:boolean .
base:yardage a rdf:Property ;
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:float .
base:singlesConfirmed a rdf:Property ;
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:boolean .
base:metalSpikes a rdf:Property ;
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:boolean .
base:grass a     rdf:Property ;
    rdfs:domain base:GolfCourse ;
    rdfs:range xsd:string ;
```

Figure 4: Golf Course concept; proposed *GolfCourse* class.

Note that the concept of the *Price* is similar to that used in the *TSS* ontology, however in the case of a golf course it is much less complicated than in the case of air travel. Therefore we have decided, for the time

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns\#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema\#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema\#>.
@prefix fee: <file:///D:/ontologies/domain/Golf/Fee\#>.
@prefix base: <file:///D:/ontologies/domain/Golf/GolfCourseTeeTime\#>.
base:GolfCourseTeeTime a rdfs:Class;
base:golfCourseID a rdf:Property;
    rdfs:domain base:GolfCourseTeeTime;
        rdfs:range xsd:string.
base:amount a rdf:Property;
    rdfs:domain base:GolfCourseTeeTime;
    rdfs:range xsd:float.
base:currencyCode a rdf:Property;
    rdfs:domain base:GolfCourseTeeTime;
    rdfs:range xsd:string.
base:startDate a rdf:Property;
        rdfs:comment ``Information about date and time in format yyyy:MM:dd'T'HH:mm:ss'';
    rdfs:domain base:GolfCourseTeeTime;
    rdfs:range xsd:string.
base:endDate a rdf:Property;
    rdfs:comment ``Information about date and time in format yyyy:MM:dd'T'HH:mm:ss'';
    rdfs:domain base:GolfCourseTeeTime;
    rdfs:range xsd:string.
base:maxPrice a rdf:Property;
    rdfs:domain base:GolfCourseTeeTime;
    rdfs:range xsd:float.
base:numberOfHoles a rdf:Property;
    rdfs:domain base:GolfCourseTeeTime;
    rdfs:range xsd:integer.
base:numberOfTimes a rdf:Property;
    rdfs:domain base:GolfCourseTeeTime;
    rdfs:range xsd:integer.
base:fee a rdf:Property;
    rdfs:domain base:GolfCourseTeeTime;
    rdfs:range fee:Fee.
```

Figure 5: Golf Course Tee Time concept; proposed *GolfCourseTeeTime* class.

being, to leave this concept golf-specific and return to this issue when the *OTA golf ontology* is going to be integrated with the *TSS* ontology. At the same time, the *Fee* concept, which is the superclass of the *Price*, is golf-specific. Finally, in Figure 6, we can see how the *DiscoutTypes* concept (from the *TSS* ontology) is utilized. To complete the description of the proposed *OTA golf ontology*, in Figure 7 we depict the *Price* class (in the RDF notation), in Figure 8 we present the *Fee* class (in the RDF notation), while in Figure 9 we introduce the *Description* class (also in the RDF notation).

5 CONCLUDING REMARKS

In this paper we have discussed how an *OTA golf ontology* can be reverse engineered from the OTA golf messaging system. In the proposed approach we have paid attention to re-use of concepts that have been defined earlier within the *TSS* ontologies of restaurant, hotel and air-travel. In this way the proposed *OTA golf ontology* can be easily integrated with the *TSS ontology*. The only issue that has to be solved is the utilization of the *Price* concept across ontologies of multiple travel entities (as some of them have more complicated price concept than others). This issue is being resolved currently as the *OTA golf ontology* is being merged with the *TSS* ontology and the results will shortly be available at (tss,). Finally, let us note that we have already created a translator that allows us to deal with the following scenario. Let us assume that information about golf courses is stored in a repository (e.g. a Jena repository (jen,)) as instances of the above described *OTA golf ontology*. Such a repository has to be queried using one of languages designed for this purpose (e.g. SPARQL (SPA,)).

At the same time, other entities "want to" communicate with our system using OTA messaging. This being the case OTA request messages have to be translated into SPARQL queries, while responses to SPARQL queries have to be translated back to OTA

Figure 6: Golf Course Tee Time concept; graphical representation.
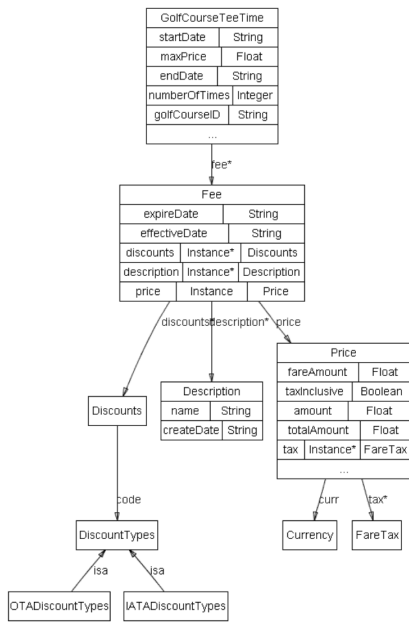
```
@prefix rdf :
<http://www.w3.org/1999/02/22-rdf-syntax-ns\#>.
@prefix rdfs:
<http://www.w3.org/2000/01/rdf-schema\#>.
@prefix xsd :
<http://www.w3.org/2001/XMLSchema\#>.
@prefix tax : <Payment/FareTax\#>. @prefix cur :
<http://protege.stanford.edu/CurrencyOntology\#>.
 base:Price a  rdfs:Class;
   rdfs:comment ''Description of a price''.
 base:amount a rdf:Property;
   rdfs:domain base:Price;
   rdfs:range xsd:float.
 base:taxInclusive a rdf:Property;
  rdfs:domain base:Price;
  rdfs:range xsd:boolean.
base:totalAmount a rdf:Property;
  rdfs:domain base:Price;
  rdfs:range xsd:double;
  rdfs:comment ''Total amount (taxes incl.)''
base:fareAmount a rdf:Property;
  rdfs:domain base:Price;
  rdfs:range xsd:double;
  rdfs:comment ''Fare amount (taxes excluded)''
base:tax a rdf:Property;
  rdfs:domain base:Price;
  rdfs:range tax:FareTax;
  rdfs:comment ''Information about taxes''
base:curr a rdf:Property;
  rdfs:domain base:Price;
  rdfs:range cur:Currency;
  rdfs:comment ''The currency information.''
```

Figure 7: Price concept; proposed *Price* class.

```
@prefix rdf:
<http://www.w3.org/1999/02/22-rdf-syntax-ns\#>.
@prefix rdfs:
<http://www.w3.org/2000/01/rdf-schema\#>.
@prefix xsd:
<http://www.w3.org/2001/XMLSchema\#>.
@prefix desc:
<file:///D:/ontologies/domain/Golf/Description\#>.
@prefix price:
<file:///D:/ontologies/domain/Golf/Price\#>.
@prefix disc:
<file:///D:/ontologies/domain/Payment/Discount\#>.
@prefix base:
<file:///D:/ontologies/domain/Golf/Fee\#>.
base:Fee a rdfs:Class;
    rdfs:comment ''Description of a fee''.
base:description a rdf:Property;
        rdfs:domain base:Fee;
        rdfs:range desc:Description.
base:price a rdf:Property;
        rdfs:domain base:Fee;
        rdfs:range price:Price.
base:effectiveDate a rdf:Property;
        rdfs:domain base:Fee;
        rdfs:range xsd:date.
base:expireDate a rdf:Property;
        rdfs:domain base:Fee;
        rdfs:range xsd:date.
base:discounts a rdf:Property;
        rdfs:domain base:Fee;
        rdfs:range disc:Discounts.
```

Figure 8: Fee concept; proposed *Fee* class.

```
@prefix rdf:
<http://www.w3.org/1999/02/22-rdf-syntax-ns\#>.
@prefix rdfs:
<http://www.w3.org/2000/01/rdf-schema\#>.
@prefix xsd:
<http://www.w3.org/2001/XMLSchema\#>.
@prefix base:
<file:///D:/ontologies/domain/Golf/Description\#>.
base:Description a rdfs:Class;
    rdfs:comment "Description".
base:name a rdf:Property;
        rdfs:domain base:Description;
        rdfs:range xsd:string.
base:createDate a rdf:Property;
        rdfs:domain base:Description;
        rdfs:range xsd:date.
```

Figure 9: Description concept; proposed *Description* class.

response messages. We have implemented the needed translator and will discuss its design in a separate report.

REFERENCES

Jena—rdf persistency engine. http://jena.sourceforge.net/

Open travel alliance. http://www.opentravel.org/

Ota golf messaging manual. OTA_ MessageUserGuide2006V1.0

Sparql—rdf query language. http://www.w3.org/TR/rdf-sparql-query/

Travel support system, software repositories. http://www.e-travel.sourceforge

Bădică, C., Bădiță, A., Ganzha, M., and Paprzycki, M. (2007a). *E-Service Intelligence—Methodologies, Technologies and Applications*, chapter Developing a Model Agent-based E-commerce System, pages 555–578. Springer, Berlin.

Bădică, C., Ganzha, M., and Paprzycki, M. (2007b). *Joutnal of Universal Computer Science*, volume 13, chapter Implementing Rule-Based Automated Price Negotiation in an Agent System, pages 244–266. Springer, Berlin.

Fensel, D. (2003). *Ontologies: A Silver Bullet for Knowledge Management and Electronic Commerce*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Gawinecki, M., Gordon, M., Nguyen, N. T., Paprzycki, M., and Szymczak, M. (2005a). Rdf demarcated resources in an agent based travel support system. In et. al., M. G., editor, *Informatics and Effectiveness of Systems*, pages 303–310, Katowice. PTI Press.

Gawinecki, M., Gordon, M., Nguyen, N. T., Paprzycki, M., and Vetulani, Z. (2005b). chapter Ontologically Demarcated Resources in an Agent Based Travel Support System, pages 219–240. Advanced Knowledge International, Adelaide, Australia.

Gawinecki, M., Kruszyk, M., and Paprzycki, M. (2005c). Ontology-based stereotyping in a travel support system. In *Proc. of the XXI Fall Meeting of Polish Information Processing Society*, pages 73–85. PTI Press.

Gordon, M., Kowalski, A., Paprzycki, M., Pełech, T., Szymczak, M., and Wasowicz, T. (2005). *Internet 2005*, chapter Ontologies in a Travel Support System, pages 285–300. Technical University of Wroclaw Press.

Gordon, M. and Paprzycki, M. (2005). Designing agent based travel support system. In *ISPDC'2005:Proc. of the ISPDC 2005 Conference*, pages 207–214, Los Alamitos, CA. IEEE Computer Society Press.

Salam, A. F. and Stevens, J., editors (2006). chapter Utilizing Semantic Web and Software Agents in a Travel Support System, pages 325–359. Idea Publishing Group, Hershey, USA.

Szymczak, M., Gawinecki, M., Vukmirovic, M., and Paprzycki, M. *Ontological Reusability in State-of-the-art Semantic Languages*, pages 129–142. PTI Press.

Vukmirovic, M., Ganzha, M., and Paprzycki, M. (2006a). *Developing a Model Agent-based Airline Ticket Auctioning System*, pages 297–306. Springer, Berlin.

Vukmirovic, M., Paprzycki, M., and Szymczak, M. (2006b). Designing ontology for the open travel alliance airline messaging specification. In et. al., M. B., editor, *Proceedings of the 2006 Information Society Multiconference*, pages 101–105. Josef Stefan Institute Press.

Vukmirovic, M., Szymczak, M., Ganzha, M., and Paprzycki, M. (2006c). Utilizing ontologies in an agent-based airline ticket auctioning system. In et. al., V. L., editor, *Proceedings of the 28th ITI Conference*, pages 385–390, Piscatawy, NJ. IEEE.

Vukmirovic, M., Szymczak, M., Gawinecki, M., Ganzha, M., and Paprzycki, M. (2007). *Designing New Ways for Selling Airline Tickets*, volume 31(3), pages 93–104.