Towards Agent-Based Rational Service Composition – RACING Approach

Vadim Ermolayev, Natalya Keberle, Sergey Plaksin

Dept. of Mathematical Modeling and IT, Zaporozhye State Univ.,
66, Zhukovskogo st., 69063, Zaporozhye, Ukraine
{eva, kenga, psl}@zsu.zp.ua

Abstract: Presented is the vision of the authors on how diverse web services may be composed, mediated by dynamic task coalitions of agents performing tasks for service requestors. The focus and the contribution of the paper is the proposal of the layered web service mediation architecture. Middle Agent Layer is introduced to conduct service request to task transformation, agent-enabled cooperative task decomposition and performance. Presented are the formal means to arrange agents' negotiation, to represent the semantic structure of task-activity-service hierarchy and to assess fellow-agents' capabilities and credibility factors. Finally, it is argued that the presented formal technique is applicable to various application domains. Presented is the ongoing work on building agent-based layered architecture for intelligent rational information and document retrieval mediation in frame of the RACING[1] project.

Introduction

Web services are the emerging technology promising to become one of the future key enablers of the Semantic Web. There are strong prerequisites that, being self-described and self-contained modular active components, web services will appear to be the key elements in assembling intelligent infrastructures for e-Business in the near future.

There is the emerging consensus that the ultimate challenge is to make web services automatically tradable and usable by artificial agents in their rational, pro-active interoperation on the next generation of the Web. It may be solved by creating effective frameworks, standards and software for automatic web service discovery, execution, composition, interoperation and monitoring [1]. Personal opinion of the authors is that the list should be extended by the means for making services the subject of automated negotiation and trade. It is also important for future service enabled web infrastructures to cope with business rules[2], notions and mechanisms of

---

[1] RACING: Rational Agent Coalitions for Intelligent Mediation of Information Retrieval on the Net. Project funded by Ukrainian Ministry of Education and Science. http://www.zsu.zp.ua/racing/

[2] International Workshop on Rule Markup Languages for Business Rules on the Semantic Web, 14 June 2002, Sardinia (Italy) http://tmitwww.tm.tue.nl/staff/gwagner/RuleML-BR-SW.html

reputation and trust with respect to services and service providing agents, dynamic character, flexibility, reconfigurability of partial plans [2], workflows, modeled business processes.

Current industry landscape provides only initial and very partial solutions of the ultimate problem. Existing de-facto standards for web service description (WSDL [3]), publication, registration and discovery (UDDI [4]), binding, invocation, communication (SOAP [5]) provide merely syntactical capabilities and unfortunately do not really cope with service semantics. Known industrial implementations such as HP E-speak [6] base on these standards and do not completely solve the challenge of semantic service interoperability. It should be mentioned that major industrial players realize the necessity of further targeted joint research and development in the field [7].

More recent research and standardization activities of DARPA DAML community resulted in offering semantic service markup language DAML-S [8] based on RDF platform. The constellation of XML based languages/ontologies for business process, logistics description is also expanding: WSFL, ebXML, BPML, RuleML, …

The goal of the paper is to determine what should be still done on the top of recent research accomplishments in order to achieve the ultimate goal: to make web services automatically tradable and usable by artificial agents in their rational, pro-active interoperation on the next generation of the Web. Conceptual frames for this development are under intensive discussion and some proposals already appear (e.g., WSMF [9]).

The paper offers a new understanding of a service as an agent capability implemented as a self-contained software component. From the other hand, provided that agents negotiate and trade exchanging services in the process of their cooperative activities in open organizations, a service may be considered (as, say, in E-speak) a kind of a generalized resource. This approach evidently implies the appearance of the rational service providing agent demanding certain incentive and aiming to increase its utility. If, for example, a service requested from a travel agency is 'BookRoundtrip('Kiev', 'Erfurt', 22/09/2003, 25/09/2003, …)', the price paid by the requestor will comprise the prices of consumable [10] resources (air fare, hotel room, …) plus the incentive paid to the service holder for 'BookRoundtrip' service component usage. This remark seems to be rational as far we are paying either salary to a secretary or a fee to a travel agent, who makes travel arrangements for us in human-business environment. Moreover, it is not in the eye of the service requestor, but the agent performing 'BookRoundtrip' service will realize according to the service markup (or the Partial Local Plan (PLP) in our terminology [11]) that the requested process [10] (or the task in our terminology [11]) is composite and will require cooperation with at least Air Companies' service providing agents and hotel booking service providing agents. These independent actors will evidently also intend to increase their own utilities by requesting fees for service usage.

Detailed discussion of this popular travel planning scenario in Section 2 helps to claim that full-scale web service exploitation in e-Business environment needs solutions beyond the facilities of today's semantic service markup. The paper focuses on one of the major open problems – dynamic composition of a desired complex service by a coalition of rational cooperative independent agents.

Diffuse: Guide to Web Services http://www.diffuse.org/WebServices.html

The authors consider that it is a reasonable architectural solution to introduce an Agent Middle Layer (e.g., [12]) between services and service consumers. Negotiation on web service allocation based on the authors' approach [2] is proposed as the mechanism for dynamic composite service formation. DAML-S[10], our Task and Negotiation ontologies [11] are used for service dynamic composition and to facilitate to inter-agent-operability.

Finally, it is described how the approach to dynamic agent-based service composition is applied to intelligent rational information retrieval from distributed autonomous resources – our RACING project.

2 Travel Planning Scenario

Let's consider the mentioned travel planning scenario having in mind that our intentions have become true and web services are available at the desired level of semantic interoperation. The authors have played the following exercise assuming themselves as "intelligent software agents" participating in cooperative execution of a conference trip planning task (Fig. 1.). Each agent possessed his/her beliefs about the environment and capabilities in performing one or another activity related to the overall high-level goal achievement – 'BookRoundtrip("Kiev, Ukraine", "Erfurt, Germany", 22/09/2003, 25/09/2003, "ICWS'03-Europe", …)'. Agents' capabilities were: their knowledge of relevant websites providing human-oriented services and their ability to operate these services via web interfaces. Agent roles were:

– AUTHOR (A) – an agent representing one of the paper authors intending to attend ICWS'03-Europe and requesting 'BookRoundtrip' service

– TRAVEL AGENT (T) – an agent actually providing 'BookRoundtrip' service by generating and conducting corresponding task execution

– FARE AGENT (F) – agents providing various air fare information and booking services

– ICWS INFO (I) – an agent providing information services on ICWS'03-Europe local arrangements, infrastructure, accommodation, etc in Erfurt

– HOTEL AGENT (H) – agents providing hotel room reservation services

– BUSINESS PARTNER (P) – an agent representing A's business partner in Austria with whom A intends to meet in Germany in time of the conference to discuss a joint proposal

As usual in travel planning an A is capable just to invoke a T with 'BookRoundtrip' task, to formulate his constraints, preferences and needs for special arrangements, to approve solutions proposed by the T. According to 'BookRoundtrip' description in terms of *Task Ontology* [11] known both to A and T (but with different granularity) service inputs are[3]:

```
Starting_Point= "Kiev, Ukraine"
```

[3] Service inputs are given semi-formally in order to avoid unnecessary details and save the paper space.

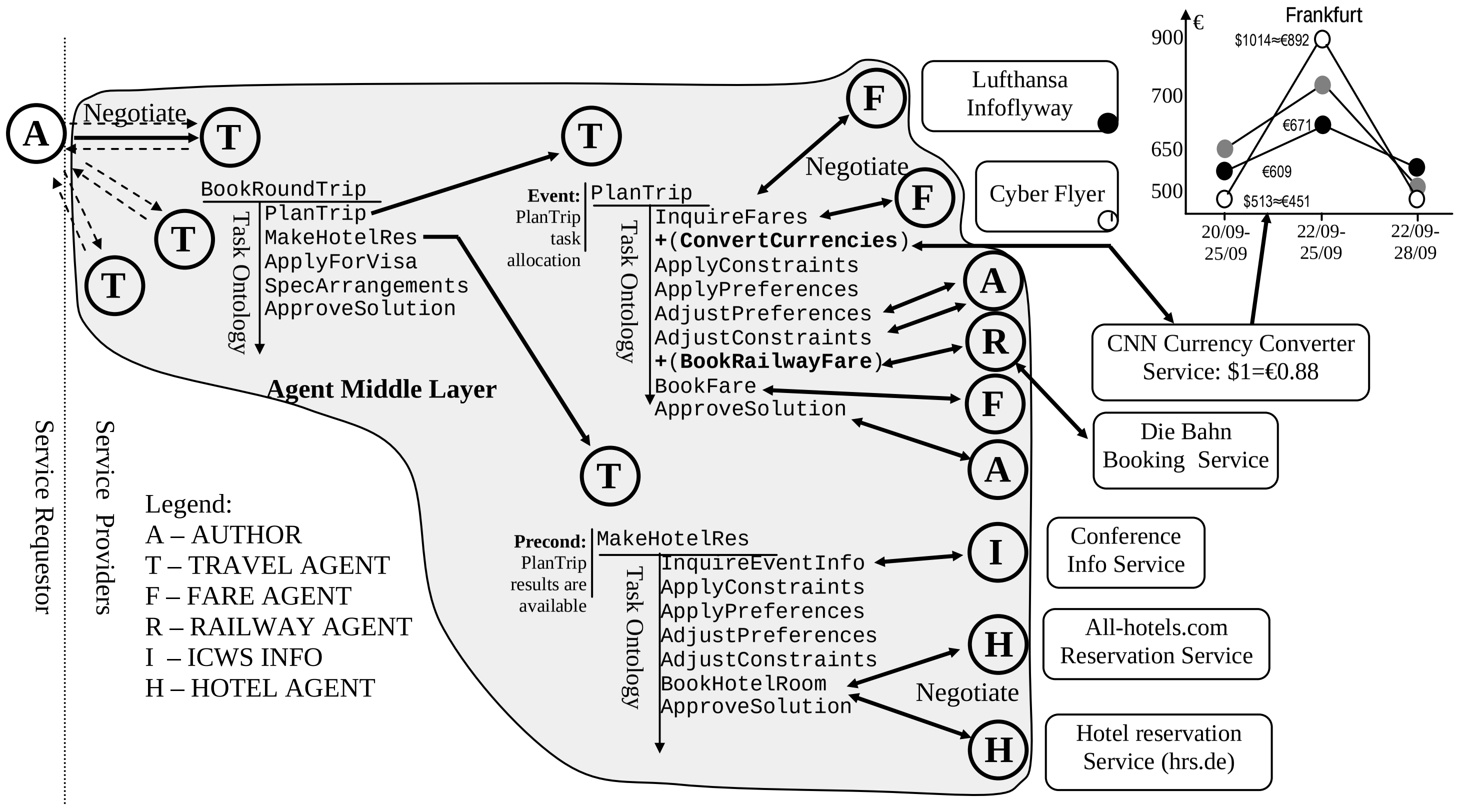


Fig. 1. “`BookRoundTrip`” task execution and service composition.

```
Destination="Erfurt, Germany"
Beg_Date =22/09/2003
End_Date=25/09/2003
Event="ICWS'03-Europe"
Preferences=("low fare, fast connections", "4-star
hotel, continental breakfast, conference discounts")
Constraints=(Budget = €1500,Payment=(VISA, USD),
Hotel >= 3-star, Room-per-night <= €110,
Hotel_Location="in Max 20 min walk from the Conference
venue")
Special_Arrangements=((Event="business dinner",
Agent = ("Prof. Heinrich C. Mayr", http://www.ifi.uni-
klu.ac.at/IWAS/HM/Staff/Heinrich.Mayr/),
Date=(23/09/2003-24/09/2003), Location=(Erfurt,
Munich)),…)
```

The process starts with the arrangement [2] A undertakes to hire one of T-s as the contractor for the job. The flow of round trip booking, T performs for A, is presented on Fig. 1. At first T accepts the task from A by means of agents' communication interface. This interface may be built upon ACL [13] for FIPA[4]-compliant agents (Appendix A-1[5]). T than uses its beliefs on how to 'BookRoundtrip'(Appendix A-2), formalized according to the *Task Ontology* (Appendix A-6), to derive that the accepted task is complex and involves at least 'PlanTrip', 'MakeHotelRes', 'ApplyForVisa', 'SpecArrangements' and 'ApproveSolution' activities. 'PlanTrip' activity is chosen (PLP of Task Ontology [11]) the first to be performed and appears to be also a complex task: 'InquireFares', 'ApplyConstraints', …, 'BookFare', 'ApproveSolution'. Before allocating Fare Inquiry to F-s T 'notices' that a slight change in the starting or ending date of the trip may result in a substantial decrease in the airfare expenses because of the Sunday Rule discounts[6] commonly offered by Air Companies.

For our example this means to T that the dates 20/09-25/09 and 22/09-28/09 should be also rationally considered for the trip. T negotiates these input changes with A asking A to provide desirability values for these dates (Fig. 2 – gray dots) indicating max price A is ready to pay for the fare within the specified dates. Requirements, T specifies for 'InquireFares' service, are thus slightly changed by introducing the list of date pairs for which the service should be performed. Contract Net negotiation is than initiated by T having F-s as participants.

F-s propositions,[7] resulting from 'InquireFares' service execution, are also given

[4] Foundation for Intelligent Physical Agents, http://www.fipa.org/, last accessed on Apr. 24, 2003.
[5] Appendixes A-1 – A-7 may be downloaded from http://eva.zsu.zp.ua/services/app.htm.
[6] "One of the most common low fare restrictions is the requirement for your stay to incorporate at least one Sunday. For example, for a round-trip New York to Miami a passenger flying Tuesday to Thursday might pay £328, but a passenger whose stay includes a Sunday would pay much less - £188." – http://www.flightcatchers.com/helpmenu/Howtofindcheapestfare.htm last accessed on Apr. 24, 2003.
[7] Lufthansa Infoflyway Booking Service http://lufthansa.com/ (last accessed on Jul. 15, 2003) and Cyber Flyer Booking Service http://cyberflyer.galileo.com/ (last accessed on Jul. 15, 2003) were

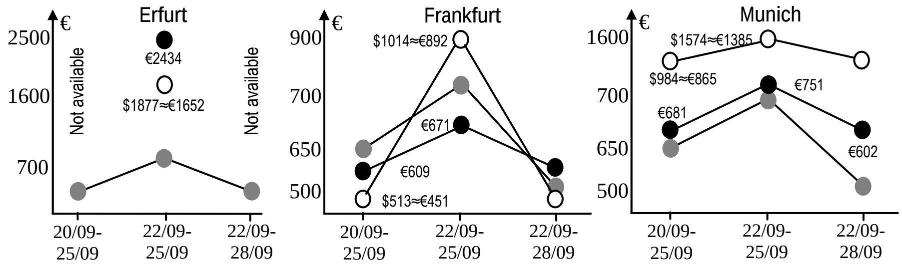


Fig. 2. Fare desirability function and service propositions:
- for how much (max) A desires the fare, - the propositions of F - Service Providers

on Fig. 2. These results cause the necessity to use one more service, which was not initially planned by T's PLP for the task. As far as the offers are provided in different currencies T needs to change the task and require the service for currency conversion[8] (+('ConvertCurrencies', Appendix A-3), Fig. 1). Conversion results are presented on Fig. 2. It is now easy for T to derive that the acceptable proposition is still for the dates 22/09-25/09, but with the destination at Frankfurt (not at Erfurt), which were not initial 'BookRoundtrip' task inputs from A. However, this result comply with A's preferences as far as there are non-stop flights available from Kiev to Frankfurt (but not to Erfurt and Munich). This implies the necessity for T to 'AdjustPreferences' by inquiring A's service. The mechanism may be similar to inputs negotiation discussed above and the outcomes may cause the invocation of some new activities, e.g., change to a train at Frankfurt-Main Airport – inquire the 'BookRailwayFare' service from Die Bahn[9] Agent. Discussion of these emerging task branches is omitted, as far it is conceptually similar to that already given before. It is however important to notice that activities which were not initially planned often emerge and appear to be critical to the overall goal achievement not only in the discussed scenario.

It is not informative to discuss subsequent activities of T. Hotel booking and visa application services are performed merely in the same manner and agents use similar mechanisms of *task (de-)composition* and *negotiation* for that. Special arrangements list is also considered as the list of trip planning tasks. However, it should be mentioned that the execution of these activities should be properly *coordinated:* note for instance that hotel reservation requires that the fare has been already booked as pre-condition (check-in and check-out dates, money left) and German Consular Service may require that the fare and the hotel room have been booked before issuing the visa.

Other important aspects, not mentioned before, are the ones of *credibility, trust* and *meaning negotiation* among agents participating in cooperative task performance and service composition. Recall Special Arrangements input for the illustration. T will negotiate with P on various aspects while arranging the Business Dinner. The dilemma for P in this environment is if to trust T (as the contractor of A which is the

used in the described exercise to obtain the offers from F-s.

[8] CNN Currency Converter: http://qs.money.cnn.com/tq/currconv/, last accessed on Jul. 16, 2003.

[9] http://www.bahn.de/, last accessed on Jul. 16, 2003.

trusted one because of the long record of partnership) and allow him to make the arrangements for P, or to reason that A may be not really experienced in arranging business dinners in Germany and to decide to better rely on his credible (Sect. 3.4) partners from Germany. In the latter case P will inform T that it will better arrange the event on its own. This in turn may effect in the necessity of the approval from A.

3 Cooperative Dynamic Service Composition

Let's enumerate the features needed to rationally provide composite flexible services for the automation of the scenarios like that of travel planning in e-business environment. Intelligent service provider needs to be capable of:

- Understanding the semantics of the activity it is supposed to perform, reasoning on if the activity is atomic or complex, decomposing complex activities according to its knowledge and the experience of the environment
- Adjusting activity inputs, requestor preferences and constraints in order to proactively reach the high level goal
- Negotiating with the requestor, the other service providers in a rational way on optimal service performance, allocation in order to increase its own utility or to obtain common meaning of the service inputs, outputs, pre-conditions and effects
- Monitoring and assessing credibility and trustworthiness of other service providers to minimize risks
- Coordinating services performance flow according to the inputs and pre-conditions

It seems to be obvious that service providing distributed open software systems possessing these capabilities may be most naturally designed and assembled of software agents. Agent platforms and agent-based systems are already used for service brokerage [1], matchmaking [12], coordination [14]. The reminder of this section will shortly present the formal approach to dynamic task decomposition and performance by coalitions of rational agents [2,11].

3.1 Middle Agents for Service Composition

Conceptual idea of service mediation is not originally new and has been argued by many authors. Strong mediation has been for instance claimed as one of the basic principles for WSMF [9]. What seems to be not really explicitly worked out before is the framework for intelligent dynamic service composition and decomposition according to the changes in the environment affected by the service execution flow.

The proposal of the Mediation Framework for Agent-enabled Service Provision targeted to dynamic service composition is presented on Fig. 3. Control flows are labeled with legends in *italic*, data flows are marked by **bold** legends. The principles around which the proposal is centered are:

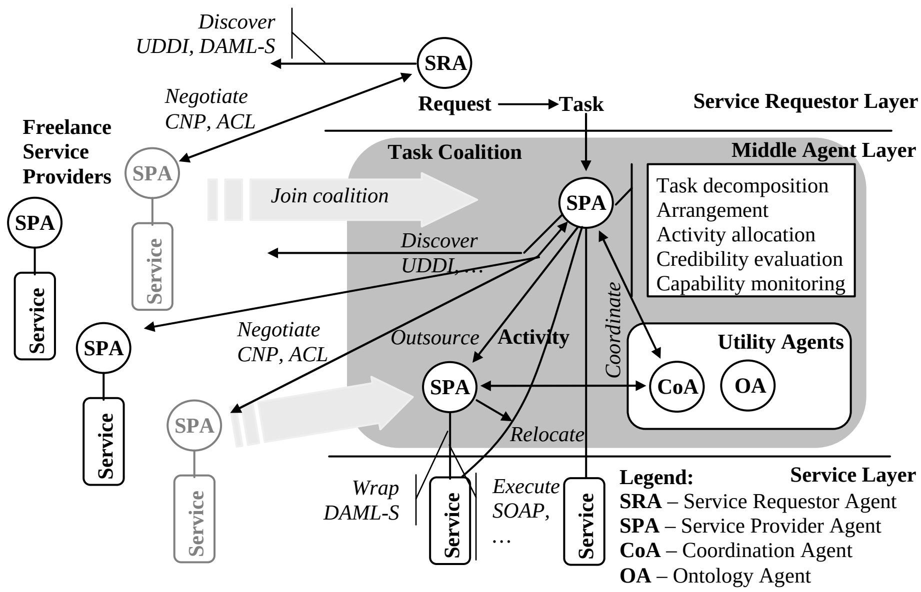


Fig. 3. The proposal of Agent-Based Service Provision Mediation Architecture.

– Agent-based Middle Layer is required for scalable, intelligent, dynamic service composition
– Composite services are interpreted as tasks comprising activities of varying granularity by the Agent Middle Layer
– Service Mediator is formed dynamically as the coalition of service providing agents (SPAs) participating in the task execution
– SPAs join task coalitions only for the time their service is required for the respective task
– SPAs are economically rational [15], autonomous and independent in taking their decisions – the only fact one SPA believes about the behavior of another SPA is: it will try to increase its utility.
– SPAs are capable of: incoming task decomposition according to its local knowledge (Task Ontology, PLP); making arrangements for activity outsourcing to another SPAs based on Contract Net negotiation; activity outsourcing to the chosen contractor SPA; adjusting their beliefs on other SPAs' capabilities and evaluating SPAs' credibility through monitoring cooperative activities
– Services are self-contained modular loosely coupled program components wrapped by SPAs; an SPA may allow another SPA to use its service by providing service context relocation
– Specialization of an SPA is defined by the set of services it wraps

If the proposal is examined from the point of implementability with existing service markup solutions the state of affairs may look like given on Fig. 3 from the authors' point of view. Yet unsolved or partially unsolved problems of service mediation are:

– Lack of common semantic ground and commonly accepted mechanism for activity outsourcing, activity parameters adjustment and meaning negotiation – negotiation ontologies family

– Insufficient representation of task/activity/service dynamic structure and granularity – task/process ontologies family

– Lack of common specifications/criteria for capability monitoring, credibility and trustworthiness assessment

The proposed architectural layering is likely to remain valid for request-task-activity-service ontology hierarchy: a service request is translated to the task at the requestor layer; these tasks are decomposed into activities at the middle layer; activity descriptions actually wrap service markups. The reminder of the section provides some outlines to approach the solutions of the open issues.

3.2 Negotiation Patterns, Ontology and Social Norms

In frame of the reported research some work in specifying and designing negotiation patterns for dynamic activity composition and performance has been done already. Negotiation ontology [11] and negotiation mechanism [2] for dynamic task coalition formation were designed to facilitate inter-agent cooperation in open organizations like B2B mediation e-marketplaces [11] or virtual organizations [16]. Mechanism for activity allocation negotiations is based on the metaphor of parametric feedbacks [2] provided as Contract Net participants' proposals in response to activity results' desirability function advertised by the negotiation initiator.

Let's recall 'InquireFares' service negotiation mentioned in Section 2 for illustration and discuss it in more details. First step T needs to perform is to choose contractors providing required service according to its desirability function derived from given budget and given deadline for service provision[10]. Advertised desirability and two-point parametric responses [2] of negotiation participants are given on Fig. 4a. After the contractors (Lufthansa Infoflyway Agent and Cyber Flyer Agent) are chosen, T allocates 'InquireFares' activity execution to both of them by providing activity description containing inputs and patterns for the results (Appendix A-4). Contractors' feedbacks after applying currency conversion service are shown on Fig. 4b. Negotiations are, thus, used by T to: optimally choose the contractors; get optimal fare information from the chosen contractors.

Contracting negotiation takes place in frame of the Arrangement Phase [2,11] each time there is the need to allocate an activity to SPAs. A kind of FIPA Contract Net Protocol is used for these arrangements. *Negotiation ontology* [11] is used as the namespace and the formal semantic frame for the contents of the messages agents communicate with while negotiating on activity allocation. It is considered that the contractors join the Task Coalition for the time necessary to play their part. Task Coalitions are considered to be a kind of social structures. Coalition members are thus bounded with coalition commitments and convention regulating their ratios of self-interest and benevolence [11].

---

[10] It is a bit artificially supposed here that F-agents provide their services for an incentive. It is, well, not very realistic for the case, because respective web sites provide their services for free at the moment – i.e. are paid by their holders in some ways.

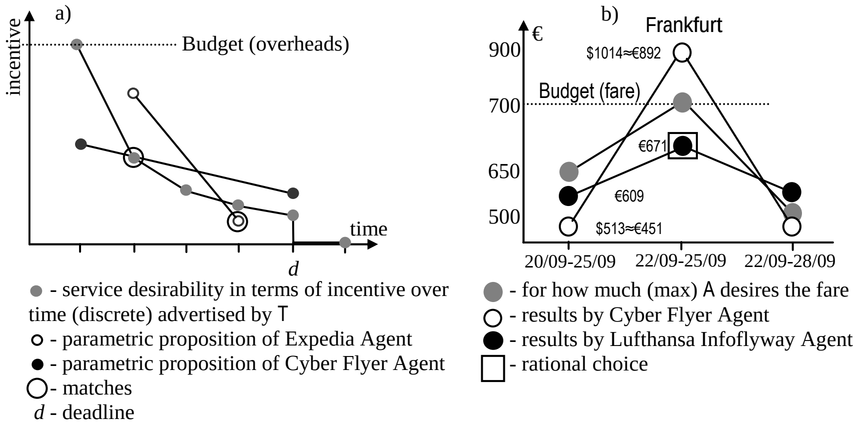


Fig. 4. 'InquireFares' negotiations: a) on service allocation – chosen are the agents proposing the conditions of service provision which match to T desires; b) on service results – see also Fig. 2.

3.3 Request-Task-Activity-Service Hierarchy

As it was mentioned before service request-task-activity-service semantic hierarchy reflects the principles of the proposed architectural layering. A request belongs to the sphere of Service Requestor Layer and is specified in terms of Task Ontology [11]. The function of the SPA chosen as the contractor for the specified request is to determine if the incoming task is the atomic activity according to its local specifications (Task Ontology). In case the task is complex and should be decomposed into atomic activities at the local level of granularity the next round of activities allocation negotiations is initiated. Only the activities the given SPA is not capable to perform on its own are negotiated with another SPAs, while the ones corresponding to initiator's capabilities are rooted to self-performance. Only an activity, for which it is true that: a) it is atomic and b) SPA is able to perform it on its own, is in relationship with the corresponding service or service loop. Atomic activity execution is performed by SPA by invoking its capability macro-model [2]: activity context is translated into DAML-S markup corresponding to Service Profile; the service is than invoked via the interface specified by its binding (or grounding in terms of DAML-S) description. Service invocation loop may actually result in one or several service runs depending on the wrapping activity inputs. For example, 'InquireFares' service will be performed three times as far as 3 different date intervals are to be processed (Fig. 4).

Semantic facet of request-task-activity-service layering is presented on Fig. 5. Specifications for 'InquireFares' activity and service are given in Appendix A-5.

3.4 Capability and Credibility Assessment

SRA and SPAs are to be able to determine which of the SPAs are capable to perform the task to be allocated. Possible mechanism to define the perspective contractors is capability matchmaking (e.g., based on LARKS [17]), or service discovery technique based on UDDI, or another service matching facilities (e.g., semantic matching based on DAML-S profiles [18]). However, in case there is some capability beliefs record maintained autonomously by an SPA in the course of cooperative task execution, the use of this knowledge may substantially facilitate to lowering computation costs by eliminating unnecessary directory/matching service usage. Evidently, if A believes that B, C and D are capable of performing desired activity because they did it before, it will rather proceed to contracting negotiation with B, C and D directly instead of trying to find some other SPAs[11] with matching capabilities.

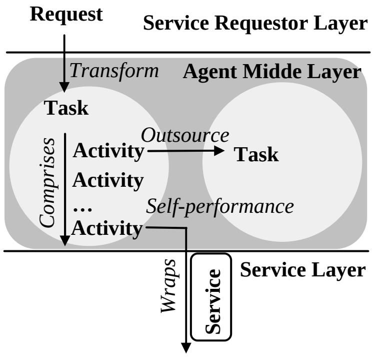


Fig. 5. Semantic layering.

A model and a mechanism of agents' capability assessment based on SPA beliefs representation in the form of Fellows' Capability Expectations Matrix (FCEM) has been elaborated in frame of the reported research [2]. SPAs accumulate and adjust their local beliefs on the capabilities of their collaborators in the course of cooperative performance. New portions of this knowledge appear each time an activity is allocated to the chosen contractor SPA. Subjective beliefs of the SPA requesting the activity on the probability of its fellows' capability to perform the given activity are thus updated. FCEM for capability beliefs representation is maintained in the following form:

$$\mathbf{C} = \begin{array}{c} \\ \mathrm{SPA}_1 \\ \\ \dots \\ \\ \mathrm{SPA}_n \end{array} \begin{array}{c} \begin{array}{ccccc} a^1 & \dots & a^j & \dots & a^m \end{array} \\ \begin{bmatrix} c_1^1 & & c_1^j & & c_1^k \\ & & \dots & & \\ & \dots & c_i^j = (q_i^j, p_i^j) & \dots & \\ & & \dots & & \\ c_n^1 & & c_n^j & & c_n^m \end{bmatrix} \end{array}, \tag{1}$$

where dimensions m and n change reflecting the appearance of new incoming activities and newly discovered or perishing activity providers.

Capability estimations c_i^j change each time an agent negotiates with its fellows to allocate an activity. Element q_i^j in tuple c_i^j stands for the quantity of recorded negotiations with fellow agent SPA_i concerning activity a^j. Element p_i^j stands for the capability expectation. The rule for c_i^j updates is as follows:

$$\begin{aligned} &1. \quad p_i^j \leftarrow p_i^j + \frac{r}{q_i^j}, \\ &2. \quad q_i^j \leftarrow q_i^j + 1 \end{aligned} \tag{2}$$

[11] Applying to a capability registry may still appear to be necessary in case B, C and D fail to provide constructive proposals.

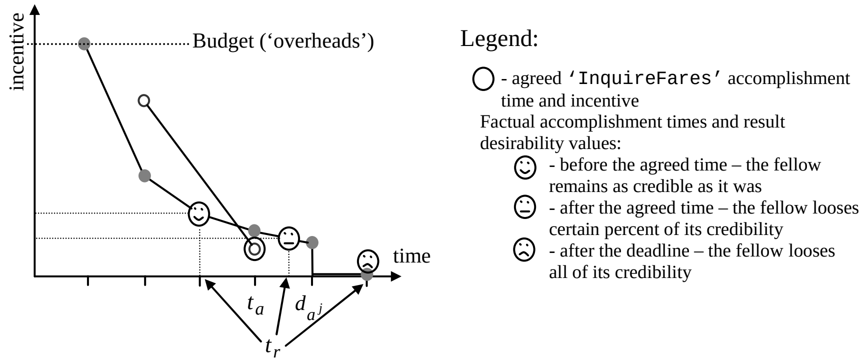


Fig. 6. 'InquireFares' accomplishment times and corresponding credibility changes.

where *r* is equal to: 0 – if the fellow rejected the activity, 0.5 – if the fellow replied that it can accept the activity and 1 – if the activity was finally allocated to the fellow.

One more aspect providing influence on a task requestor's decision to allocate an activity to one or another negotiation participant is its assessment of the participant's *credibility*. A self-interested SPA, due to the appearance of the new highly attractive activity offers in the competitive environment or due to the peculiarity of its behavior, may lower previously declared capacity [2,11] it is spending for the bulk of the activities under execution. This will lead to the increase of the performance duration and may seriously decrease the requestor's desirability of these results and, thus, lower the credibility value for the SPA selling its' fellows short.

The mechanism of accounting fellows' credibility values is merely the same as that of adjusting the beliefs on changing fellow capabilities (1-2). Credibility assessment values change over time as the requestor agent adjusts its subjective beliefs by comparing the desirability values (Fig. 6) derived from:

1-st – activity duration the executive committed to within the activity allocation arrangement negotiation and

2-nd – actual results delivery time. Corresponding credibility matrix elements are than recomputed due to the following:

$$Cr_{i,j} := Cr_{i,j} \times \begin{cases} 1, t_r \le t_a \\ p_{a^j}(t_a / t_r), t_a < t_r \le d_{a^j} \\ 0, t_r > d_{a^j} \end{cases}, \quad (3)$$

where t_a is the time the parties have agreed to accomplish the activity a^j, t_r is the actual time of a^j results delivery, d_{a^j} is the deadline and p_{a^j} is the weight coefficient characterizing the current priority of a^j for the activity requestor agent.

Credibility threshold values associated with respective activities and stored in agents' PLPs are used by task requesting agents to assess possible risks and alter their strategies.

4 RACING Functionalities, Agents and Services

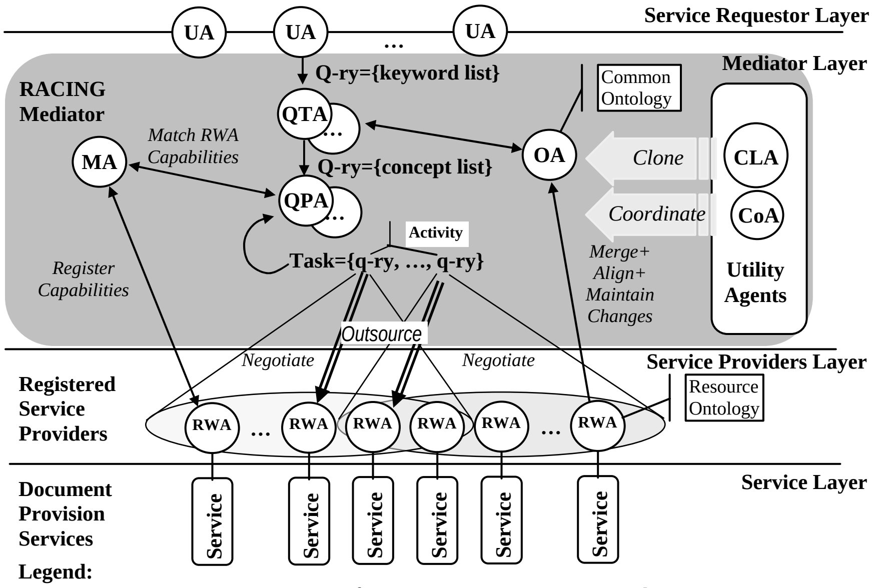
A reader might argue that, fairly, travel planning is not the task that really requires sophisticated agent-enabled automation technique: negotiations, coalitions, service wrapping and composition – at least from the customer's side. Travel planning is not that time consuming to make its performance impossible without automation. Moreover, a human will sometimes still be better in arranging loosely formalized things that require intuition and context dependent understanding with complexity beyond the capacity of, say, the first order logic based languages. However, the presented technique is applicable not only in case you plan your conference trip [11, 16].

Let's project the above discussion to distributed information and document retrieval domain. In the terms of document retrieval a service request is commonly formulated as a search phrase – a first order logic expression over the list of keywords or phrases. Documents (web pages, scientific papers, magazines, books) are stored at disparately structured distributed autonomously maintained databases or text collections in a digital form, are marked-up according to different standards and *often cost money*. A task for document retrieval may thus be presented as the set of interrelated activities distributed over the document providers. These activities wrap the (partial) queries derived from the initial user's request.

The goal of our RACING project is to provide mediation facilities for user query processing by the means of the query semantic decomposition, the rational distribution among independent, autonomous, rational document retrieval service providers wrapping respective document resources, and the fusion of the obtained results (Fig. 7.). User agents acting on behalf of the human users or real organizations (e.g., libraries) and service providing agents are considered as business representatives or business models in frame of the project. RACING mediation may thus be classified as B2B mediation. It is evident that such a kind of intelligent activities really needs sophisticated automation to be scalable and gracefully downgradable.

User query processing, resource wrappers registration by the capability matchmaker and common ontology maintenance are the basic functionalities of the RACING mediator (Fig. 7.). Though only query processing may be considered as a real business process involving third-party service providers for money, the other two ones are also performed as tasks and require various types of negotiation and semantic interoperation.

For example, the outline for the User Query Processing scenario is as follows. The process starts at UA with the formulation of the query in terms of the key phrases familiar to the given user. UAs are cloned by CLA utility agent each time a new user comes to the mediator and perish when the user leaves. User profiles (mappings of their most frequently used key words or phrases to the Mediator Common Ontology (MCO) concepts) is incrementally collected, stored at OA [19] in the form of the reference ontology and is used by QTAs. UA actually generates and conducts the task of query processing and acts as the proxy between the user and mediator. Query processing task generated by UA contains 'CloneQTA', 'TransformQry', 'CloneQPA', 'ExecuteQry' activities. The cloning activities are outsourced to CLA which clones QTA and QPA for query processing. 'TransformQry' activity is outsourced to QTA



Legend:
UA – User Agent, **QTA** – Query Transformation Agent, **QPA** – Query Planning Agent, **RWA** – Resource Wrapper Agent, **OA** – Ontology Agent, **MA** – Matchmaking Agent, **CLA** – Cloning Agent, **CoA** – Coordination Agent

Fig. 7. RACING reference architecture.

which performs the transformation of the query in terms of keywords to semantically matching query in terms of the concepts of the MCO. The last activity is outsourced to QPA which generates the following set of activities for 'ExecuteQry' task: 'DecomposeQry', 'PerformQryset'. Query decomposition is performed by QPA in order to extract the parts of the incoming query, which may require different capabilities from document service providers. This extraction is guided by topic classification of the MCO. Resulting set of partial queries is performed by QPA as the following activity sequence: 'MatchRWA', 'PerformQry'. Matching activity is allocated to MA for a certain incentive over accomplishment time. MA returns[12] the list of RWAs capable to perform document providing services relevant to the partial query. 'PerformQry' activity allocation is negotiated with pre-selected RWAs in terms of service 'overheads' over time and document price and the contractor is chosen for query performance (Section 3.2). Contractor RWA receives the partial query in terms of MCO. It therefore needs to transform the query into the terms of its Resource Ontology. This transformation activity is outsourced to OA which actually holds the necessary mappings. RWA than invokes document service it wraps with the transformed query and provides documents relevant to the query to QPA.

[12] As QPAs in RACING have limited life time, RWAs' credibility and capability assessment (Section 3.4.) is performed by MA for registered resource wrappers. QPAs supply MA with necessary data obtained from cooperation with RWAs.

5 Concluding Remarks

The paper presented the vision of the authors on how diverse web services may be composed, mediated by means of middle agents and their coalitions performing tasks for service requestors. It is also claimed that such a mediation facility may substantially enhance today's solutions available in web service provision. This vision is grounded on the results obtained in agent-enabled business process modeling and management.

It is stated that though the concept of service mediation is not totally new there is still some work to be done before it becomes real engineering technology. For example, from the authors' point of view what seem to be not really explicitly worked out before is the framework for intelligent dynamic service composition and decomposition according to the changes in the environment affected by the service execution flow. The rationale to cope with such kind of dynamic composite service execution representation is argued by the discussion of a popular travel planning scenario. The main focus and the contribution of the paper is the proposal of the layered service mediation architecture. Agent Middle Layer is introduced to conduct service request to task transformation, agent-enabled cooperative task decomposition and performance. Outlined are the formal means to arrange agents' negotiation, to represent the semantic structure of task-activity-service hierarchy and to assess fellow-agents' capabilities and credibility factors. Other important aspects of cooperative agent-enabled service mediation, just mentioned in the paper because of space limits, are the questions of *meaning negotiation* and *activity coordination* among agents participating in cooperative task performance and service composition. Finally, it is argued that the presented formal technique is applicable not only to the tasks like travel planning. Presented is the reference architecture of the rational multi-agent mediator for intelligent information and document retrieval. Further development and deployment of the mediator is in progress in frame of the RACING project.

Though thorough standardization and harmonization work should be performed before the presented approach becomes an engine for web service provision, the authors are certain, that agent-enabled rational web service composition and mediation may provide a substantial contribution bringing closer the day, when the brave new world of machine-processable automated web services comes true at least in e-business domain.

References

1. McIlraith, S. A., Son, T. C. and Zeng, H.: Semantic Web Services, IEEE Intelligent Systems. Sp. Issue on The Semantic Web 16 (2002) 46–53
2. Ermolayev, V. A., Plaksin, S. L.: Cooperation Layers in Agent-Enabled Business Process Management. Problems of Programming 1-2 (2002) 354-368
3. Web Services Description Language (WSDL) 1.1. W3C Note 15 March 2001 http://www.w3.org/TR/2001/NOTE-wsdl-20010315/ Last accessed June 3, 2003
4. UDDI Technical White Paper http://uddi.org/pubs/Iru_UDDI_Technical_White_Paper.PDF Last accessed June 3, 2003

5. Simple Object Access Protocol (SOAP) 1.1. W3C Note 08 May 2000 http://www.w3.org/TR/2000/NOTE-SOAP-20000508/ Last accessed June 3, 2003
6. Karp, A.: E-speak E-xplained HPL-2000-101 20000807 External Technical Report http://www.hpl.hp.com/techreports/2000/HPL-2000-101.pdf Last accessed June 3, 2003
7. Layman, A.: Web Services Framework. In: Proc of W3C Workshop on Web Services 11-12 April 2001, San Jose, CA USA http://www.w3.org/2001/03/wsws-program/ Last accessed June 3, 2003
8. Ankolekar, A., Burstein, M, Hobbs, J. R., Lassila, O., Martin, D., McDermott, D., McIlraith, S. A., Narayanan, S., Paolucci, M., Payne, T., Sycara, K.: DAML-S: Web Service Description for the Semantic Web In: Proc,. of Int. Semantic Web Conference, June 9-12th, 2002 Sardinia, Italy
9. Fensel, D., Bussler, C.: The Web Service Modeling Framework WSMF. White Paper http://www.cs.vu.nl/~dieter/wese/wsmf.paper.pdf. Last accessed June 3, 2002
10. The DAML Services Coalition. DAML-S: Semantic Markup for Web Services. http://www.daml.org/services/daml-s/2001/10/daml-s.pdf Last accessed June 3, 2003
11. Ermolayev, V. Keberle, N., Tolok, V.: OIL Ontologies for Collaborative Task Performance in Coalitions of Self-Interested Actors. In: H. Arisawa, Y. Kambayashi, V. Kumar, H.C. Mayr, I. Hunt (eds.): Conceptual Modeling for New Information Systems Technologies ER 2001 Workshops, HUMACS, DASWIS, ECOMO, and DAMA, Yokohama Japan, November 27-30, 2001. Revised Papers, LNCS, Vol. 2465. Springer-Verlag, Berlin Heidelberg new York (2001) 390-402
12. Sycara, K. P., Klusch, M., Widoff, S., Lu, J.: Dynamic Service Matchmaking Among Agents in Open Information Environments. SIGMOD Record 28(1) (1999) 47-53
13. FIPA Communicative Act Library Specification. Doc. No: XC00037H, Doc. Status: Experimental, http://www.fipa.org/specs/fipa00037/XC00037H.pdf last accessed June 14, 2003.
14. Papadopoulos, G. A.: (2001) Models and Technologies for the Coordination of Internet Agents: A Survey. In: A.Omicini, F.Zambonelli, M.Klusch, R.Tolksdorf (eds.): Coordination for Internet Agents - Models, Technologies, and Applications, Springer-Verlag, Berlin Heidelberg New York (2001)
15. Nwana, H. S.: Software Agents: an Overview. Knowledge Engineering Review, 3(11) (1996) 205–244
16. Ermolayev, V. A., Tolok, V. A.: Modelling Distant Learning Activities by Agent Task Coalitions. In: Q. Jin, J. Li, J. Cheng, C. Yu and S. Noguchi (eds.): Enabling Society with Information Technology, Springer-Verlag, Tokyo (2002)
17. Sycara, K., Widoff, S., Klusch, M., Lu, J.: LARKS: Dynamic Matchmaking Among Heterogeneous Software Agents in Cyberspace. Autonomous Agents and Multi-Agent Systems 5 (2002) 173–203
18. Paolucci, M., Kawamura, T., Payne, T., Sycara, K.: Semantic Matching of Web Services Capabilities In: Proc. of Int. Semantic Web Conference (ISWC'2002), June 9-12, 2002, Sardinia, Italy
19. Ermolayev, V., Keberle, N., Plaksin, S., Vladimirov, V.: Capturing Semantics from Search Phrases: Incremental User Personification and Ontology-Driven Query Transformation. In: Godlevsky, M., Liddle, S., Mayr, H. (eds.): Information Systems Technology and its Applications. Proc. of the 2-nd Int. Conf. ISTA'2003 Lecture Notes in Informatics (LNI) - Proceedings, Vol. P-30, GI-Edition, Bonn (2003) 9–20.