System Framework for Generating Open Development Platforms for Web-Service Enabled Applications Using Semantic Web Technologies, Distributed Decision Support Units and Multi-Agent-Systems – INFRAWEBS II

H.-JOACHIM NERN [1], G. AGRE [2], T. ATANANSOVA [2], J. SAARELA [3]

[1]Aspasia Knowledge Systems, D-40210 Duesseldorf,

[2]Institute of Information Technologies, BAS, BG-1113 Sofia,

[3]Profium SAE, FIN-02600 Espoo,

Abstract: - The paper introduces a system framework, which enables software and Service providers to generate and establish open and extensible development platforms for Web Service enabled applications. Within the system design several non-functional aspects are addressed and covered by applying a set of powerful (AI-) methodologies: interoperability and adaptability are achieved by exchanging semantically enriched Services among platform partners within P2P environments. The machine processability of ontology based structures in conjunction with "closed loop" approaches in the Service operation, maintenance and feedback cycle (QoS, full life cycle) ensures implicitly a high degree of stable adaptability and scalability, as well as the establishment of extensible platform structures (self-organizing structures). Service composability is guaranteed by applying open standards. To achieve a customizable modular development environment "experience" feedback patterns in distributed rule and case bases (CBR) are archived, evaluated and utilized within distributed decision making processes.

Key-Words: Semantic Web, Web-Services, Ontologies, Distributed Decision Making, Development platforms, Multi-Agent Systems

1 Introduction

The European research and RTD area related to "Software Architectures & Components" resp. the area of "Engineering of Service Functionality" is characterized by several European projects running or being completed since 1999/2000. In the following an overview is given with respect to the overall research and development objective: Open Development Platforms for Software and Services. Furthermore it will be shown that the formerly formulated vision of the Semantic Web (SW) [1], [2] is becoming reality, especially considering the aspect of merging Semantic Web technology and Web Service approaches [3], [4], [5].

The ACE-GIS project (Adaptable and Composable E-commerce and Geographic Information Services) aims at developing of distributed WEB-Services with special emphasis to geographical information systems. First approaches are done toward building new as well as integrating and composing existing Web Services [6].

Methodologies and tools for constructing and using very large and distributed ontologies are the subject of the ON-TO-KNOWLEDGE project (Tools for content-driven knowledge management through evolving ontologies). In the frame of this project (completed in 2000) some tools and methods for intelligent access to semi-structured and textual data sources have been developed. The ON-TO-KNOWLEDGE project and the thematically related projects: SWWS (Semantic Web Enabled Web Services), DIP (Data, Information and Process Integration with Semantic Web Services), KnowledgeWeb [6], [7], can be determined as the first application oriented projects within the domain of the Semantic Web and Semantic Web enabled Web Services.

An analysis of the state-of-the-art shows the lack of systems for effective collaboration in reconfigurable networks for the design, deployment and maintenance of Web Service enabled applications. The existing software tools and systems are mostly inco-

herent and static. In the majority of cases it is due to proprietary aspects or features preventing an effective adjustment and balancing of information pools.

The approach proposed in this paper tries to advance the state of the art by developing generation tools and systems for establishing and using development platforms where Services will be run according to open standards such as BPEL4WS, DAML, OWL, RDF(S) [8], [9] etc. This will imply the step from creating just "static" to creating and maintaining "dynamic" Web Service applications due to the machine-readability feature of semantically structured information sources and domains.

2 INFRAWEBS Vision & Objectives

The vision of the presented approach is to provide a system framework endowed with tools for building open development and run-time environments for software and Services. This system framework should provide the next generation of methodologies and middleware for supporting developers through all phases of the software and Service life-cycle (requirements analysis, design, deployment, maintenance) [6].

The INFRAWEBS II system design is to provide an intelligent framework**,** which enables software and Service providers to generate and establish open, extensible and reconfigurable development platforms for Web-Service applications. Established in such a way the open platforms consist of coupled and linked INFRAWEBS II units, whereby each unit provides tools and adaptable system components to analyze, design, conjointly compose, and maintain Services (SW Services) within their complete life cycle. Due to the versatile and modular structure, the designed framework enables the handling of a broad class of Services. In this sense the objectives are to develop a software framework to: provide a comfortable and “easy to use” knowledge brokering unit, which is used within a collaborative environment for analysis, design and composition of SW Services.

For achieving these objectives the following main software and system components are designed:

1 An ontology and similarity based organizational memory (as the entity specific information pool) and tools for extracting and retrieval of SW Service specific knowledge and data, and

2 A P2P, Multi-Agent and SW Service based interoperability environment (Distributed SW Service repositories) for the CBR based design and composition of SW Services (experience based CASE-tool), and

3 A run-time supervision (Executor, QoS-Broker) and maintenance module enriched with distributed decision making features.

The innovative feature of the present INFRAWEBS II approach is the integration and "closed loop" characteristic, which persists in the semantic based coupling of Service processing issues to knowledge brokering structures: Service handling actions and knowledge management actions are integrated. (Integration of the tripartite SOA - Service requester / provider / registry - approach with information brokering modules).

Applying comprehensive (AI-) methodologies yields in the fulfillment of several non-functional aspects: *interoperability* and *adaptability* are achieved by exchanging semantically enriched Services among open platform partners within P2P environments. A high degree of *scalability* and *extensibility* is attained (self-organizing structures) by exploiting the machine readability feature of ontology based structures. The application of "*closed loop*" approaches within the Service operation and feedback cycle (QoS) ensures implicitly system stability and a high "*quality of Service*" handling, deployment and maintenance. Service *composability* is guaranteed by applying open standards like XML, RDF(S), BPEL4WS, WSFL, WSIL, DAML(s), OWL etc. Distributed decision making processes based on archiving experience feedback patterns in distributed rule and case bases (CBR) ensure the generation of *customizable modular* development environments.

3 INFRAWEBS II Methodology and System Scheme

The design of the presented system framework is based on the comprehensive adoption of AI methods and techniques, especially decision making (CBR), machine learning, classification & clustering and closed loop feedback cycles embedded in interoperable middleware structures. The main approach consists in separating the system functions according to there determination and finally in integration and expansion of the SOA structure with respect to Service handling (SWU – Semantic Web Service Unit) and knowledge brokering (OM/SIR – Organizational Memory / Semantic Information Router). As a precondition the system segments are realized in modular blocks connected to each other.

The system framework consists of two main components: a knowledge brokering component and a client & provider component, for coupling the Ser-

vice processing issues and the knowledge management actions as one integrated approach.

The knowledge broker is realized as an organizational memory (OM) with decision support and recommender facilities. This broker is designated to endow the specific entity (business enterprise, governmental institution etc.) with comprehensive facilities to maintain and process the entity specific knowledge (knowledge artifacts) with respect to the SW Service design. Via this broker the user (developer) accesses the entity specific knowledge and extracts decision supported (via system recommendations) the SW Service relevant knowledge artifacts (see layer 1 in Fig. 1). These artifacts are transferred to the SWU, where the developer has the capabilities (designer, composer, executor) to "build" (design and compose) the knowledge artifacts to running SW Services.

The client & provider component (SWU) is designed as a semantic and ontology based module handling and maintaining Semantic Web Services – the Semantic Web Service Unit (SWU) as a workflow and knowledge sharing component - as a type of SW Service based collaboration platform and interoperable middleware. It acts in two directions: it provides Web Services to the platform and collaboration partner (layer 2) and secondly to the customer of the entity (layer 3). The main B2C / B2B process is conducted via this channel.

Implemented in different entities (software and/or Service providers) (see illustration in Fig. 1), the net of coupled INFRAWEBS II units build a comprehensive development platform for Web Service enabled applications. Coupling the units to a network means generating and establishing an open development and collaboration (workflow) platform.

The networking and platform partners provide SW Services, which are incorporated in their own Services product range (composition). On the one hand, it is a conceptional expansion of "own" (entity specific) Services, on the other hand, the creation of new products and Services is facilitated to develop value added products and Services. This reflects an innovative type of semantic based platform oriented networking (semantic based interoperable middleware). The platform partners are endowed with facilities to optimize their partnership relations; furthermore, they are enabled to build faster and more effective partnerships with "new" business partners.

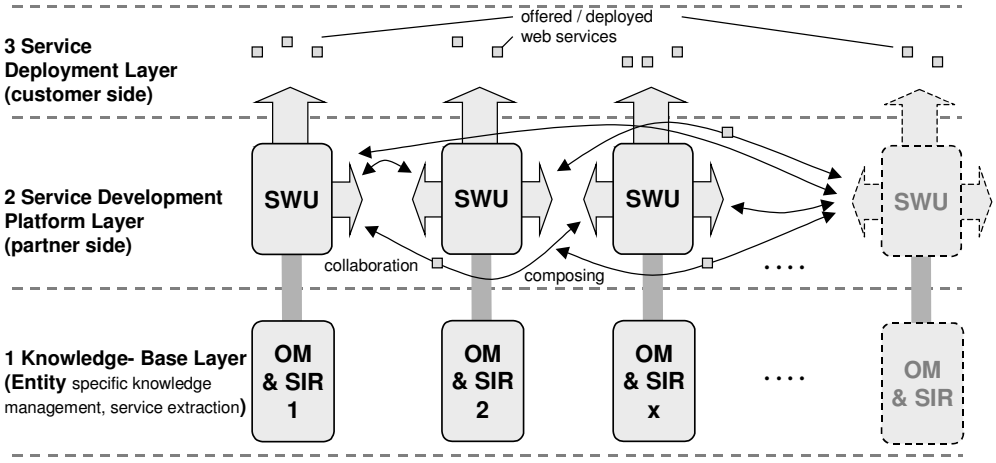

Fig. 1. The coupling of the INFRAWEBS II units enables the establishing of reconfigurable self-organizing development platforms for the design, deployment and maintenance of Web Service enabled applications.

4 INFRAWEBS II Architecture & Modules

The presented system architecture is an expansion and enhancement of the formerly framework and system design of INFRAWEBS I [10]: "Intelligent Framework for Networked Businesses and Governments Using Semantic Web Services and Multi-Agent-Systems". The enhancement consists in applying new modules, like the distributed decision support module, a run-time Service executor coupled with a QoS-Broker and Service specific knowledge extraction and recommendation as well as in modifying existing modules.

4.1 INFRAWEBS II Architecture

As illustrated in **Fig. 2**, the system architecture of the INFRAWEBS II is basically designed using a two component approach – a knowledge brokering component (OM/SIR - Broker) and a dynamic SW

Service Unit (SWU – Client & Provider). The broker component consists of the similarity and Ontology based organizational memory of the entity (i.e. business enterprise). A semantic information router (SIR) interconnects the broker component and the client & provider component by accessing the Distributed SW Service Repository module of the SWU. Coupled with this repository are mainly four modules for 1) designing static Web Services (no inferencing facilities) and composing dynamic SW Services (with inferencing capabilities), 2) for distributed decision making and execution and 3) for agent based discovery and mediation of SW Service. The designing, composing and execution modules are supplemented by a multi-agent module. Security and privacy aspects are considered in a further module connected to the execution and agent related modules.

4.2 INFRAWEBS II Modules

The system design follows strictly the demand for concise modularity, whereby each module is independently realizable. In the following the modules are described with respect to their mode of operation and their interaction resp. interrelation capabilities.

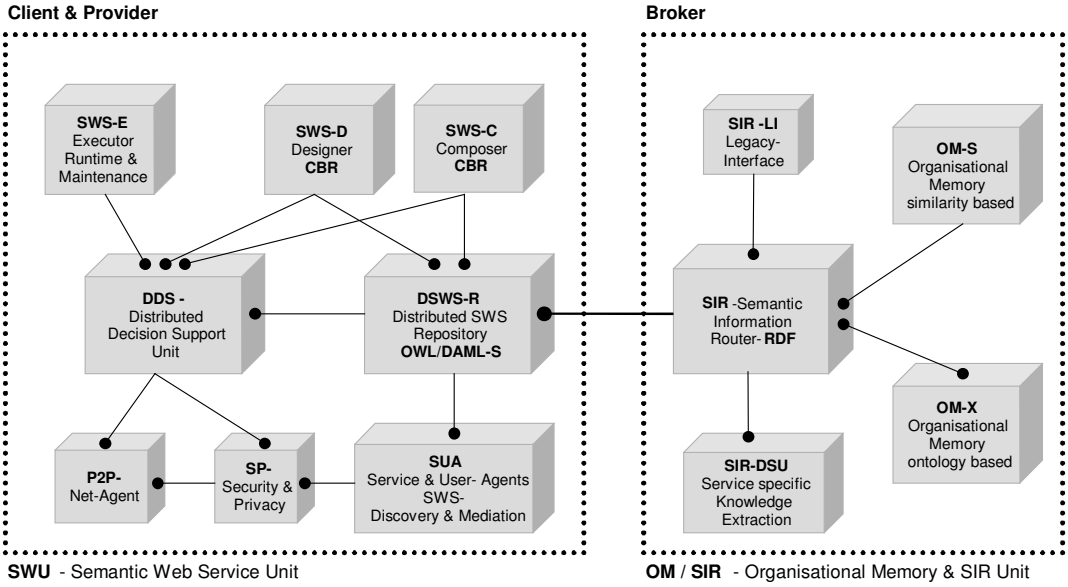

Fig. 2. The INFRAWEBS II architecture designed as a modular two component framework

4.2.1 SIR – Semantic Information Routing

The SIR module extracts the relevant knowledge artifacts out of the OM (Organizational Memory) and processes a conversion procedure to transform and adapt the provided knowledge objects to a SW Service specific structure using RDF/XML and further meta-data description schemes. The knowledge objects are semantically preprocessed for delivery to the SW repository. It acts on the one hand as a conversion system (i.e. similarity based organization to semantic based structures). On the other hand it provides an interface for the Web Service generation tools for accessing the OM and extracting Web Service specific knowledge artifacts. A recommender system (SIR-DSU) enables the developer to extract Web Service specific knowledge artifacts. Due to the fact that in several organizations different legacy systems exist a legacy interface (SIR-LI) provides capabilities to access different types of OMs.

4.2.2 SWS-D / SWS-C Semantic Web Service Designer & Composer

These two modules access the DSWS-R (repository) for single SW Service descriptions. The designer and the composer are realized as CBR based SW Services development tools and are endowed with decision-making capabilities. The SWS-Designer processes a designing procedure for single static Web Services advertised to the external (customer) as well as in the internal (networking) area.

The SWS-C – the composer – is responsible for the dynamic composing (orchestration) of available (existing) SW Services. This SW Services might be located and designed within the actual INFRAWEBS II unit or loaded and accessed via the collaboration network and incorporated from other SWUs – a composing process by combining existing "own" and "foreign" SW Services, provided by the collaboration and platform partners (partner INFRAWEBS II unit). The DSWS registry is re-

quested for SW Service delivering information. It also uses the ontology approach to represent the Service capabilities.

The SUA is requested to process agent based discovery and mediation actions to provide adequate SW Services for composition purposes. The designing and composing process is comprehensively decision supported (CBR – case based reasoning approach). Accordingly, a self-organizing case base is implemented for storage and retrieval of designing/composing "experience" patterns.

The tool supports the following main functionalities:

- Designing static (single) web Services
- Dynamic composing (orchestration) of SW Services
- Re-use of SWU Service composition experience

The tool consists of the following sub modules:

- Case-Based SW Services Composition
- Self-organizing Case-Memory of SWU Web Services
- Static Web Services Designer
- Dynamic Web Services Composer

The architectural design of the Web Service modeling module is based on case and rule based reasoning, whereby the Web Service designing procedure is related to the INRECA methodology. The designing and composing process is a comprehensively decision supported, especially under the aspect of "learning from the past".

A self-organizing case-based memory provides capabilities for effective storing and retrieval of SW Service composition patterns. It is an adaptation of the Dynamic Episodic Memory method with respect to the composition and orchestration process of SW Services. A description vocabulary of client goals, preferences and constraints provides ontology information for the domain-specific Services of the SWUs. The selected schemes for case representation and the case indexing vocabulary are based on the analyses of SW Service descriptions provided by the DSWS-R.

4.2.3 DDS / SWS-E Distributed Decision Support Unit & Service Executor / QoS-Broker

The issues of collaborative decision making (Distributed Decision Support), Quality of Services monitoring and Service execution is considered in the DDS and the SWS-E modules. The SWS-E provides a run-time environment for running SW Services and monitoring the Service execution process considering QoS metrics. Extracted QoS specific data patterns are fed back to the designing and composing modules and guarantees in this way a closed loop optimization cycle (Fig. 3).

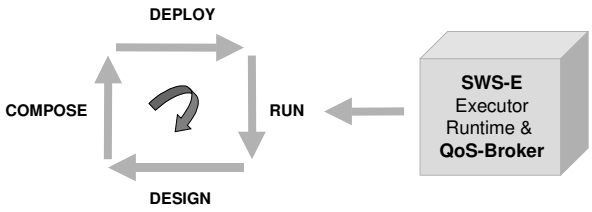

Fig. 3 The QoS-Broker within a closed loop process

The behavior patterns (profiles) of the Service designing and composing process (see self organizing case base in the Composer / Designer) are formalized and archived as design / composing rules and stored in local as well as global rule bases (Distributed Rule Bases for Distributed Decision Support). An agent based replication and adjustment procedure ensures a "global" consistency of the rule bases implemented in the different platform environments of the platform partners.

The function of the modules are:

· Monitoring the SW Service execution process oriented on QoS metrics

· Extract QoS related statistical data patterns out of the monitoring process

· Feedback the QoS patterns to the designing and composition process

· Extraction of design / composing patterns of each coupled platform partner

· Formalization, transformation and archiving of adequate rules

· Analyzing the distributed local rule bases for adjustment of the global rule base (global rule base replication)

This central unit is responsible for the distributed decision support structures. It is connected to the distributed Web Service repository (DSWS-R), the designing and composing modules (SWS-D, SWS-C) and via a Service Net agent (p2p net Agent) to the local rule bases of the platform partner modules. It summarizes and investigates the decision relevant information tags for forming and formalizing general influences from the platform partners.

The executor - realized as a run-time and executor module - is responsible for executing and managing the constructed Service compositions. Through evaluation and interpretation of performance specific metrics QoS specific patterns will be extracted and brokered.

The QoS Broker operates via the central unit (global rule bases) and influences the Service designing and composing process (feedback) in the sense of a closed loop optimization procedure. The broker evaluates the running SW Services according defined major requirements (availability, accessibility, integrity, performance, reliability, regulatory, etc).

5 Conclusion

The system framework described in this paper provides tools and system components for generating and establishing open and extensible development platforms for Web Service enabled applications. Several non-functional aspects are fulfilled by applying effective AI-methodologies: Interoperability and adaptability are reached by interchanging semantically enriched Services among platform and collaboration partners. Implementing "closed loop" approaches within the Service operation, maintenance and feedback cycle (QoS, full life cycle) ensures stable adaptability, as well as the establishment of scalable, extensible platform structures.

Experience feedback patterns in distributed rule and case bases are archived, evaluated and utilized within distributed decision making processes. For user friendly handling of the collaboration and composing structures, several supporting tools are designed to assist the developer in execution of his problem, such as Web Service designers, composers and executors. A multi-agent-system provides user interface agents as well as agents to perform Web Service discovery and mediation.

Distributed decision support systems and tools open complex and impervious structures to the user and the customer. This aspect is especially related to the business sector, where the amount of materials, objects, coherences and relations is in permanent growth. This permanent growth provides - implicitly - a massive loss in straightforwardness – intelligent platforms and tools like the presented one could help to bring back what has been lost. Being directly connected to business partners and having direct access to their Services and products, advances the stimulation of innovation "through positive effects on the knowledge value chain".

The system immanent property, consisting in the fact that the platform partners have the ability to change their "action role" within the development platform - to be client, broker and Service provider at the same time - ensures a highly dynamic and efficient Service production process and workflow.

The coupling and integration of knowledge management units (Broker) and dynamic SW Service production and execution units (SWU) invokes on the user side a comprehensive stimulation of the creativity in Service generation processes. For example a similarity based knowledge organization system enables that yet unknown knowledge artifacts and could be discovered and formed to successfully implemented and running Services. Based on features for Service specific knowledge scanning yields in stimulation with respect to create and generate new products along the B2B added value chain.

References

[1] Fensel, D.,: The Semantic Web and its Languages, IEEE Intelligent Systems, Trends and Controversies, pp.1, November/December, 2000.

[2] Berners-Lee, T., Hendler, J., Lassila, O.: The Semantic Web, Scientific American, 2001

[3] Bussler, C., Maedche, A., Fensel, D.: Web Services: Quo Vadis? In: IEEE Intelligent Systems, Trends and Controversies: Web Services: Been There, Done That? Vol. 18, No. 1, January/February 2003

[4] Bussler, C., Fensel, D., Maedche, A.: A Conceptual Architecture for Semantic Web Enabled Web Services. In: SIGMOD Record, Vol. 31, No. 4, December 2002

[5] Fensel, D., Bussler, C.: The Web Service Modeling Framework WSMF. In: Electronic Commerce Research and Applications, Vol. 1, Issue 2, Elsevier Science B.V., Summer 2002

[6] Lacroix, M. Bus, J. :Technologies & Engineering for Software, Systems & Services, IST IV.3, http://www.cordis.lu/ist/ka4/tesss/projects.htm.

[7] Fensel, D. , Bussler, C.: Digital Enterprise Research Institute, http://deri.ie/projects.html , http://knowledgeweb.semanticweb.org/

[8] Hendler, J.: Agents on the Web", IEEE Intelligent Systems, Nov/Dec 2001

[9] Mcllraith, S. Son, T.C. , Zeng, H.: Mobilizing the Semantic Web with DAML-Enabled Web Services", Proc. of the 2nd Int. Workshop on the Semantic Web – SemWeb'2001, pp.82-87, May 1, 2001, Hongkong, China.

[10] Atanasova, T., Nern, Hans-Joachim, Agre, G., Saarela, J.: INFRAWEBS - Intelligent Framework for Networked Businesses and Governments Using Semantic Web Services and Multi-Agent Systems, 7th WSEAS International Conference on COMPUTERS, WSEAS, Corfu, Greece, 2003