

SERVICE-ORIENTED ARCHITECTURES MATURITY MODELS

Sandro GERIĆ

Author: Sandro Gerić
Workplace: Faculty of organization and informatics, University of Zagreb
Address: Pavlinska 2, Varaždin, Croatia
E-mail: sandro.geric@foi.hr

Abstract

Service oriented architecture (SOA) represents a new evolutionary spiral in the software-applications development and in the evaluation of the information systems' concept. SOA is based on Web services, so called "granulated distributed program entities" that independently coexist in interaction with other programs and services. They can revile it and be used under special conditions determined by service publisher. Those entities are connected in complete software solution in the case of SOA.

SOA represents an extraordinary challenge at scientific, practical and economical level. That is a reason why SOA represents the topic of many different researches originating from different sources. SOA represents a framework for business processes and information and communication technology integration in form of components (services) that are reusable and interoperable, and that satisfy the needs and demands of dynamical business processes. (Vrček, Gerić, Kermek, IIS, 2006.) [7]

Until present different researches have been done related to SOA, and some technical standards like: Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL), Universal Discovery Description and Integration (UDDI) were developed. Important program's equipment manufacturers (like IBM, Oracle and Microsoft) are working intensively and are investing in SOA and necessary standards development. From published development strategies it is clearly that SOA represents an essence of their future development. So, it is clearly that SOA will be developed in domain of practice, but in the scientific researches as well, because it opens a set of questions that can not be resolved just by technical standardization. Besides the basic infrastructure standards, SOA has a significant influence on business processes, and because of that, standardized and not-standardized specifications, like XML Process Definition Language (XPDL), Web Services Flow Language (WSFL), Business Process Modelling Language (BPML), Business Process Execution Language for Web Service (BPEL4WS), Web Services Choreography Description Language (WS-CDL) are developed in this domain. Unified Modelling Language (UML) gave also a strong motivation for SOA development, and in the last few years necessary UML profiles were developed. Service oriented security aspect is resolved as well – different technological standards and protocols have been developed, e.g Secure Sockets Layer (SSL), Security Assertion Markup Language (SAML), Web Services Security (WSS), eXtensible Access Control Markup Language (XACML), XML signature, XML encryption. Based on these facts, we may conclude that the implementation and usage of service oriented architectures is mostly organizational, and not a technical issue [6].

The use of service oriented architecture from legislative point of view is mostly defined by national laws, protocols and rules. Relationship between the service provider and service user is usually defined by *service level agreements* that are unique for each implementation and usage of service oriented architecture.

The domain that is still not understood completely is the influence that service oriented architectures has on business process reengineering and organizational prerequisites that should be fulfilled as a presumption for successful SOA implementation. We believe that the main reason for this is the fact that SOA is still not understood and observed as the most complicated form of information system.

It is not easy to define a unique set of criteria, or prerequisites that organization has to fulfil in order to successfully implement SOA. The variety of organizations and their business environment is simply too large for identifying a unified set of such criteria. Because of this the ability for organization to adopt service oriented architecture can be estimated by *service oriented architectures maturity models* (SOAMM). This bring us to the goal of this paper – to find out if different SOAMM's, and their maturity levels are compatible and equivalent; how do different SOAMM correlate, and do they define similar or the same levels of SOA implementation?

SERVICE-ORIENTED ARCHITECTURE MATURITY MODEL – AN OVERVIEW

The level of development, implementation and usage of service oriented architectures can be monitored by service oriented architectures maturity models. The use of SOAMM can result with better understanding of the exact level of SOA implementations, changes that still have to be made, the needs for business process reengineering, and better planning of SOA implementation process.

There are several SOAMM, and their common characteristic is that they define the process of SOA implementation using different SOA maturity levels. Each maturity level represents a set of criteria that have to be fulfilled during the process of SOA implementation. Those criteria can be categorized in several groups: *organizational prerequisites*, *technical prerequisites*, *security prerequisites* and *legislative prerequisites*.

The majority of SOAMM have been developed by ICT industry, e.g.: *Service Integration Maturity Model* (SIMM) (Arsanjani & Holley 2005), *Sonic SOAMM* (Sonic Software Corp. et al., 2006), *IT Service Capability Maturity Model* (Niessink et al., 2005), *ITIL IT Service Maturity Model* (ITIL Implementation Model, 2005), *Web Services Maturity Model* (Sprott, 2006), *SOAMM Mittal* (Mittal, 2006), *Enterprise SOA Maturity Model* (SOA Practitioners Guide: Part 2, 2006), *IBM Global Services Service Integration Maturity Model* (IBM, 2005), as well as maturity models that represents the base for SOAMM development, like *Capability Maturity Model Integration* (CMMI) (CarnegieMellon SEI 2002). SOAMMs are not unique, they are defined in different environment and they are using different criterion to define the SOA maturity level. The question that arises is the question of SOAMM's comparability, their compatibility, and the possibility to develop a new SOAMM for certain domains (e.g. public administration, industry, retail, etc.).

SOA MATURITY MODEL ELEMENTS

Since the service oriented architecture domain is relatively new and in constant development the main challenge is how to define a SOAMM that will simplify the SOA implementation process, and that will not make it even more complicated [3].

Significant number of developed SOAMMs is based on *Capability Maturity Model* (CMM) [16] that was firstly published in 1993., and on its latest version *Capability Maturity Model Integration* (CMMI) [16] [17] published in 2002. Both versions (CMM and CMMI) were developed at the *Software Engineering Institute* at Carnegie Mellon University.

CMMI can be understood as a method for evaluating and measuring the maturity levels of integration and software development processes. CMMI defines five maturity levels. Each maturity level represents an achieved level of ICT development and identifies focal points where certain improvements need to be made [3]. Although SOAMM's are conceptually similar to CMM and CMMI there are important differences between those two groups of maturity models. While CMM and CMMI are used for measuring maturity of ICT processes, SOAMMs are used for measuring the maturity of organizational architectures and their capabilities to implement service oriented architectures [17].

During the development of SOAMM it is necessary to define a set of criteria (e.g. organizational characteristics) that will be used for evaluating organization's capabilities to successfully implement SOA. By this we can identify drawbacks and omissions in organizational architecture that have to be eliminated in order to improve the SOA's implementation processes. Defined criteria are selected and grouped into categories (similar to CMMI model) that represent characteristics and capabilities of certain architecture level. Those characteristics can be used as reference points for defining and monitoring the SOA development and implementation plan. Here it is important that focus is not only on SOA implementation but also on organization's architecture maturity measurement.

The level of understanding SOAs is different in different organizations, and it varies from almost none to excellent and detailed understanding of technical and organizational aspects of these architectures. Because of that SOAMM has to encompass different elements of maturity that usually starts with *vision* or understanding what SOA actually is. Besides understanding what SOA is, existing SOAMMs encompass aspects like *technical prerequisites* (on architecture's lower levels), *data architecture*, *information and process oriented architectures* in integrated SOA. SOAMM has to evaluate the way and the level of details in which organization has developed *service models* that are used to define existing services as well as *business requirements* for new services. A presumption is that on lower maturity levels organization will have only ambiguously defined service models, or will not have them at all. On higher levels of maturity, service models should be precisely defined and elaborated, and will represent services that are already developed or are being developed. Next element that SOAMM should have defined is *scope of SOA's implementation*. On lower maturity levels scope of SOA implementation will be narrow and focused on pilot project or department, and on higher maturity levels it will encompass different department, even different organizations outside of organizations boundaries. The last element of SOAMM is *maturity of SOA implementation* that can vary from theoretical models on lower maturity levels to fully implemented SOA on higher maturity levels [3].

COMPARISON OF EXISTING SOA MATURITY MODELS

In this section basic elements of existing SOAMMs and their differences will be described. The SOAMMs that are included in this analysis are: *Service Integration Maturity Model* (SIMM), *Sonic SOAMM*, *IT Service Capability Maturity Model*, *ITIL IT Service Maturity Model*, *Web Services Maturity Model*, *SOAMM Mittal*, *Enterprise SOA Maturity*, *IBM Global Services Service Integration Maturity Model*.

Service Integration Maturity Model (SIMM) was firstly published in its simplified form 2003. [2], and current version was published in 2005. SIMM was developed in the scope of IBM's service oriented architectures development projects. Development team lead by A.

Arsanjani and K. Holley has developed a SOA maturity models that consist of 4 maturity levels [2]:

1. Implementing individual Web services - Creating services from tasks contained in new or existing applications;
2. Service-oriented integration of business functions - Integrating services across multiple applications inside and outside the enterprise for a business objective. Organizations transform itself into structured form of enterprise application integration. System that combines legacy software and data integration is being established;
3. Enterprise-wide IT transformation - An architected implementation enabling integration across business functions throughout an enterprise;
4. On Demand Business Transformation - Broad transformation of existing business models or the deployment of new business model;

Comparison between SIMM and CMMI shows significant similarity and overlaying between maturity levels (*incomplete, performed, managed, defined, quantitatively managed* and *optimizing*). Each maturity level consists of specific process areas and generic practice for process area that can be used to improve organization capabilities. According to SIMM following elements are in connections with maturity levels: level of coupling, the use of standards, the level of service identification, business models, business goals and objectives, metrics supported by services, technologies, management, infrastructure, skills and implementation capabilities.

Service oriented architecture maturity model simply called *SOA Maturity model* was published at the end of 2005. by Sonic Software Inc., Bearingpoint, Systinet i AmberPoint. This model was developed as a result of information gathered from more then 2000 software architects and software engineers that participated at the *Architect Forum* organized by Sonic Software Inc., as well as information gathered from SOA implementations reports and best practice experience. This maturity model consists of 5 maturity levels. And for each maturity level basic advantages and criterions are defined.

SOA Maturity Level 1 is called *Initial Services*. Initial Services represent the initial learning and initial project phase of SOA adoption. Projects here are typically done to meet a specific need to implement functionality while trying out specific technologies and an approach to SOA. This maturity level includes R&D activities testing the SOA technologies in a controlled environment. The initial introduction of SOA is driven as part of an application integration project. The key benefits of this initial project are to provide the business functionality required while also learning how to develop and deploy a basic SOA application [8].

SOA Maturity Level 2 is called *Architected Services*. It is at this level that standards are set as to the technical governance of SOA implementation, typically under leadership of the architecture organization. The key business benefit of this level is development and deployment cost reductions through the use of SOA standard infrastructure and components as compared to using older technologies or costs accumulated through multiple unique one-time projects. Based on the learning and feedback from the initial services at *Initial Services* level, architectural standards and standard implementation technologies are defined [8].

SOA Maturity Level 3 is called *Business/ collaborative services*. The transition from *architected services* level to *measured business services* level is possible on two different ways. Business services level is focused on improvement of internal business process (internal SOA), and Collaborative services level is focused on improvement of externally oriented business processes (external SOA). Between those two maturity levels focus should be on Business services because it is necessary to improve internal processes and only after that

organization can improve its externally oriented business processes. This maturity level puts focus on creating strong interconnections between technologies and business processes, in a way that enables fast and painless changes in business processes, integration of business processes between different business units and ICT support to entire system [8]. The collaborative services maturity level is oriented towards development and implementation of services that will act as a link between internal and external business interface [8].

Measured Business Services level focuses on measuring and presenting business processes at the business level so as to provide continuous feedback on the performance and business impact of the processes implemented at maturity level 3 [8].

SOA Maturity Level 5 or *optimized business services* SOA, adds automatic response to the measurements and displays of maturity level 4. In this way, the SOA information system becomes the “enterprise nervous system” and takes action according to events occurring at the business level according to rules optimizing business goals.

IT Service Capability Maturity Model was published in 2005. by F. Niessinka, V. Clerca, T. Tjeldink and H. Van Vliet. This is a maturity model that is focused on establishing and determining the maturity level of services, and that encompass all activities necessary for service establishment in organization. It is not primarily used as maturity model for service oriented architectures, but if we presume that ability to define services at organization level is a necessary presumption for service-oriented architecture implementation. Model defines a minimal set of criteria necessary for achieving certain level of maturity. IT service capability maturity model is measuring the capability of organization to realise ICT services using scale of 5 maturity levels. *IT service process capability* represents current result achieved by IT services processes, and *IT service process maturity* is a measure that shows how a certain process is defined, measured, managed, controlled and effective [4].

Model defines following maturity levels [4]:

1. *Initial level* – is characterized by *ad-hoc* processes of service implementation. Small number of defined services. Success strongly depends on individual efforts and enthusiasm.
2. *Repeatable level* – basic processes for services management have been established. There is a certain discipline that is characterized by repeatable use of successful services implementation.
3. *Defined level* – ICT service development process is well defined, standardized and integrated into standard business processes. All services are developed and implemented using standardized procedures.
4. *Managed level* – measurement and evaluations of development and implementation processes are conducted. Service development processes and implemented services are described and evaluated.
5. *Optimizing level* – is characterized by continuous improvement of developed services and development and implementation processes.

ITIL uses the same philosophy. It observes users (*lifeware*), processes and supporting technologies as three main success and sustainability factors for quality of services [11]. It defines 5 maturity levels [11]:

1. *initial*;
2. *repeatable*;
3. *standardised*;
4. *predictable*;
5. *optimising*.

Besides generic maturity levels that are defined, model defines specific indicators for each maturity level through 5 focal areas:

- *governance* – vision, strategy, management, client-service relationship.
- *service culture* – key element for service use.
- *people* – roles definition, skills and training.
- *processes* – key processes and the way there are managed.
- *technology* – the way how selection and integration of technology components into SOA is conducted.

Although *Web Services Maturity Model* [5] is not directly linked to service oriented architecture we have included it into this analysis because majority of service oriented architecture is based on web services as main technology component.

Web services maturity model is based on models for business and technology maturity [5]. Based on maturity levels defined by these models, a new model was defined, a model that consists of 4 maturity levels for web services technology.

Based on this model structure there are 4 phases that each organization should evolve through when it implement SOA [5]:

1. early learning phase;
2. integration phase;
3. reengineering phase;
4. maturity phase.

SOAMM “Mittal” maturity model defines a scale of 5 maturity levels, which are: initial, repeatable, defined, managed, and optimized [12]. The names of maturity levels are identical to those in CMM, but their interpretation is slightly different.

Organizations at the Initial SOA maturity model level are typically those in which no formal processes for architecture exist. There is no separation of architecture from projects. Typically, these organizations do not have an EA team; each project team, generally broken down by line of business, works in a silo. The focus is on delivering an individual project. The result of this level is unpredictable project schedules, budget overruns, and poor quality code that is typically not reusable and hard to maintain. Projects repeat the same tasks, thus increasing the cost of delivery and maintenance. At this level of maturity, or rather immaturity, IT tends to dictate business agility instead of the other way around [12].

At repeatable level, some elements of architecture efforts emerge. Project teams tend to define a reusable architecture that they use from one project to another. Informal paths of communication are established across project teams. The result is some level of reuse of architectural components. Ad hoc processes and chaotic lines of communication lead to some level of repeatability in architectural solutions, thus lowering the cost of delivery and the maintenance costs of software. The greatest advantage of this level of maturity is the realization of benefits that a structured process can lead too. Project teams are realizing the potential benefits of a more synergetic approach to software development. They realize that they can prevent the huge cost overruns, create a predictable software development schedule and improve the overall quality of the software.

At maturity level 3 SOA team is in place that is tasked with standardizing architecture elements. This team is responsible for creating reference architecture, educating project teams on this architecture, and defining a governance and enforcement policy. Typically it starts with creating a set of technical components and frameworks with standardizing the use of these frameworks across project teams. The biggest cost that comes with this maturity level is the ongoing cost of architecture maintenance. This level is a first step in recognizing the need for an SOA. Enterprise application development (EA) efforts seem to fail, because they are not capable to satisfy needs across different line of business [12].

Managed level of maturity is achieved when process of defining a path to SOA is started. This level of maturity comes with a lot of risk as well as a lot of benefit. In particular, it is important to realize that Level 4 has little to no short-term cost benefit. Getting to and executing Level 4 is a very costly effort for any organization. If done right, however, it leads organizations to Level 5 in the SOA maturity model. If done poorly, an organization will most likely drop to a Level 2, because the SOA team will be disbanded or have little support from the business [12].

At the optimizing level architectural processes and policies are institutionalized, and there is a clear recognition of the value of services. A framework is in place for each team to expose and consume services. At this level, organizations can truly explore the value of SOA. They start figuring out how to exchange services with their business partners, suppliers, and customers.

Business service level re-use -- not just technology component re-use -- is core to the architecture in order to achieve maximum business agility. At this level, organizations see the cost and time benefits of having a more nimble IT organization that can quickly respond to business needs. The key goal of this level is to define an end-point for your architecture initiatives. High standards and goals need to be clearly defined to achieve them.

This maturity model was developed in industry by BEA Systems [14]. It defines 4 levels of maturity:

1. *traditional development and integration,*
2. *develop web applications,*
3. *develop composite applications,*
4. *automate business processes.*

At the first maturity level (traditional development and integration) there are no activities focused on SOA development or implementation.

At second maturity level (develop web applications) the focus is on providing client and browser-based business solutions to both internal and external users. An example of applications developed and used at this stage is e.g. web-enabled CRM, ERP, or custom applications that support connected and occasionally disconnected operations. IT organizations typically deploy enterprise-based solutions and services such as content management, search, instant messaging, white boards, etc. [14]

Develop composite applications level means that composite applications aggregate and provide information and data from a variety of sources and channels, and make them available to internal and external users as appropriate [14]. The business requirement is for IT to adapt to changing business needs. Several business units may approach IT to develop custom applications, enhance their own branding, increase productivity, or provide additional information to their customers, partners, or employees [14].

At highest maturity level (automate business processes) the applications, data, and infrastructure should enable users to perform their roles effectively by providing the right information at the right time [14] At this stage, the enterprise can start achieving higher profit by consolidating multiple business systems into a single system. Business organizations should now be ready to abandon their conservative and “*islands-of-automation*” type solutions and transition to the target state of end-to-end business process management [14].

The last maturity model explained in this article is *Global Services Service Integration Maturity Model* that describes SOA maturity using 7 maturity levels. This model is an upgrade on *Service Integration Maturity* model

The maturity levels defined by this model are [1.]:

1. *silo / data integration* – integration processes are characterized by ad-hoc integration of elements with similar properties, and often, non-systematically changes in information system architecture;
2. *integrated / application integration* – organization is implementing a more structured form of *EAI – enterprise application integration*, and point of integration have been established. The integration approach is based on use of legacy software and data integration;
3. *componentized / functional integration* – organization defines critical parts of IS in form of components and modules. Transformation, renovation and improvement processes are being used to adapt the system to modern platforms. Components and modules boundaries are clearly defined, and the system functionality is based on module functionality. Components are interconnected via interfaces;
4. *simple services / process integration* – is characterized by first version of SOA in organization. Services are defined, developed and made available for internal and external use;
5. *composite services / supply-chain integration* – organization is spreading its influence through supply chain and service eco-system. Services are becoming the basis of contract relationships between elements of supply chain. They are being used and incorporated into individual information systems environment;
6. *virtualized services / virtual infrastructure* – virtual infrastructure for applications have been prepared. This maturity level is achieved after the strict interconnections between applications, services, components and data flows have been removed;
7. *dynamically reconfigurable services / eco-system integration* – organization is characterised by dynamical architecture that can include, exclude developed services [1].

ANALYSIS RESULTS

When we compared the previously described maturity models, their maturity levels, characteristics and how are they defined, we have concluded that there are some overlapping and joint characteristics. The Table 2. shows levels of all previously described maturity models, and their relationships. It is not possible to define exact (e.g. 1 to 1) relationships between certain maturity levels of different maturity models. The reasons for this are differences in understanding the maturity in different maturity models. But it is clearly that maturity levels are very similar, criteria they are using are also similar, especially if we observe them as an abstract of three maturity areas: lower maturity, medium maturity and higher maturity. Under *lower maturity* we define the early phases of SOA implementation, which are characterized by initial learning and initial project phase of SOA adoption. At this level projects are typically done to meet a specific need to implement functionality while trying out specific technologies and an approach to SOA. This maturity level includes R&D activities testing the SOA technologies in a controlled environment. The initial introduction of SOA is driven as part of an application integration project. The basic characteristics of this maturity are, e.g. new functionality, initial learning phase, pilot projects, ESB, service registry, service level management service, legacy integration, developers learn service development skills, apply SOA technology to immediate organisational needs, define initial ROI measurements for SOA projects, apply ROI measurements to initial projects, create service definitions, integrate SOA into project development methodology, quantify costs, time and business benefits of pilot projects, use of technology standards like: XML, XSLT, WSDL, SOAP, Java, .Net. [1],[3],[4],[5],[8],[11],[12],[13]. After the organization establishes necessary prerequisites from lower maturity area, the next step is so called medium maturity.

At this level that standards are set as to the technical governance of SOA implementation, typically under leadership of the architecture organization. The key business benefit of this level is development and deployment cost reductions through the use of SOA standard infrastructure and components as compared to using older technologies or costs accumulated through multiple unique one-time projects, focus on creating strong interconnections between technologies and business processes, in a way that enables fast and painless changes in business processes, integration of business processes between different business units and ICT support to entire system. The basic characteristics of this maturity can be defined for several aspects, e.g. **integration** (e.g. multiple integrated applications, service and policy repository, transformation service of ESB used, support for heterogeneity and distributed systems reliable messaging, mediation, ease of deployment database integration, versioning, internal security, use of incremental integration, reuse, ease of modification, availability, composite applications, orchestration service to manage long running processes, etc.), **governance and SOA management** (e.g. governance using policies and service definition, exception management service, IT cost reduction and control, performance management, business process rules, event-driven processes, performance measurement in real-time, business activity monitoring (BAM), define and meet business oriented performance metrics, etc.), **security** (e.g. *single sign-on*, implement internal and cross enterprise security, etc.), **organizational support** (e.g. institutional use of SOA, put in place architecture leadership for SOA, proven return from use of standardised technology, integrated SOA into organisation-wide development process, provide organisation-wide SOA training and competency center, specify policies for creation or modification of business processes, extend business processes to external organisations, etc.), **use of standards** (UDDI, reliable-messaging, WS-policy, WS-Addressing, xQuery, WS-security, SAML, WS-BPEL, etc.). The characteristics from lower maturity levels are also included. [1],[3],[4],[5],[8],[11],[12],[13]. Organizations that achieved the highest maturity levels are characterized with SOA information system that has become the “enterprise nervous system” and that takes action according to events occurring at the business level according to rules optimizing business goals. At this level, organizations can truly explore the value of SOA. The basic characteristics of this maturity are, e.g. provides automation in business processes, business can be optimised by reacting and responding automatically, event-driven automation, self organised enterprise, enterprise nervous system, reacting to actions according to rules to optimise business goals, provide enterprise-wide leadership for business and SOA governance, prove return from SOA supported continuous improvement, implement self-correcting business processes, event-driven automation for optimization, eco system integration, automatically reconfigurable architecture, compose services or applications at runtime, based on policy descriptions services. The characteristics from lower and medium maturity levels are also included.[1],[3],[4],[5],[8],[11],[12],[13].

CONCLUSION

Service-oriented architectures represent a new innovative form of information systems. This new architecture is specific because it builds an information system from different components (web services) that are provided by different service providers, developed under different technologies and are functioning under different environments. Those components are interconnected into information system architecture that has the same, or even larger, level of functionality than "classical" IS. Gartner reports are estimating that in the 2008. almost 80% of all ICT development projects will be based on service-oriented architectures (Gartner Inc., http://www.gartner.com/resources/125800/125868/gartners_positi.pdf, 05.09.2006).

Besides the technical standards and solutions that are necessary for SOA implementation, an organization has to fullfield a set of different criteria in order for SOA implementation to

be successful. It is not easy to define a unique set of criterion, or prerequisites that organization has to full field in order to successfully implement SOA. The variety of organizations and their business environment is simply too large for identifying a unified set of criterion. Because of this the ability for organization to adopt service oriented architecture can be estimated by *service oriented architectures maturity models* (SOAMM). Different service-oriented architecture maturity models exists, and analysis that we conducted in this paper has shown that they define very similar maturity levels, and a very similar set of prerequisites that organization has to achieve in order to increase its maturity level in the context of SOA implementation. Based on that we can conclude that is possible to define a basic set of criteria, as a necessary set of prerequisites that an organization has to fullfield if it wants to establish successful SOA implementation. Which criteria should be used in this set, and what are the differences and additional prerequisites for some specific domains (e.g. public administration, manufacturing, retail, financial institutions, etc.) are the elements of further research in this area.

References

- [1] ARSANJANI, A., HOLEY, K., *Increase flexibility with the Service Integration Maturity Model (SIMM)*, Maturity, adoption, and transformation to SOA, <http://www-28.ibm.com/developerworks/webservices/library/ws-soa-simm/>, <21.06.2006.>
- [2] ARSANJANI, A., *Toward a pattern language for Service-Oriented Architecture and Integration*, Part 1: Build a service eco-system,
- [3] <http://www-128.ibm.com/developerworks/webservices/library/ws-soa-soi/>, <13.11.2006.>
- [4] BLOOMBERG, J., *What to look for in SOA maturity model*, http://www.zapthink.com/report.html?id_ZAPFLASH-20051031, <20.10.2006.>
- [5] NIESSINK, F. Et al, *IT Service Capability Maturity Model*, 2005, <http://www.itservicecmm.org/>, <12.10.2006.>
- [6] SPROTT, D., *Web Service Maturity Model*, <http://roadmap.cbdiforum.com/reports/maturity/index.php>, <10.10.2006.>
- [7] VRČEK, N., GERIĆ, S., KERMEK, D., *Stanje razvoja i primjenjivost servisno orijentiranih arhitektura*, CASE 18, Zbornik radova, Opatija 2006., pp. 87-94.
- [8] VRČEK, N., GERIĆ, S., KERMEK, D., *The State of Development and Applicability of Service Oriented Architectures*, IIS 2006, Proceedings of the 17th international conference "Information and intelligent systems", Varaždin, Croatia, september 2006, pp. 77. – 85
- [9] ***: *A New Service-Oriented Architecture (SOA) Maturity Model*, http://www.sonicsoftware.com/solutions/service_oriented_architecture/soa_maturity_model/index.ssp, <02.05.2006.>
- [10] ***: *Capability Maturity Model Integration (CMMI SM),Version 1.1, CMMI SM for Software Engineering*, Carnegie Mellon University, Software Engineering Institute, 2002., str. 1., <http://www.sei.cmu.edu/cmmi/models/>, <20.10.2006.>
- [11] ***: *IBM's SOA Foundation: An Architectural Introduction and Overview*, <http://download.boulder.ibm.com/ibmdl/pub/software/dw/webservices/ws-soa-whitepaper.pdf>, <15.05.2006.>
- [12] ***: *ITIL Implementation Project, Introduction to the IT Service Maturity Model, v0.4, ITIL Implementation Manager*, University of Melbourne, <http://www.infodiv.unimelb.edu.au/itil/framework/smf.html>, <12.09.2006.>
- [13] ***: *Service Oriented Architecture Maturity Model*, <http://www.kunalmittal.com/html/soamm.shtml>, <11.09.2006.>

- [14] ***: SOA Practitioners' Guide Part 2: SOA Reference Architecture, dev2dev.bea.com/2006/09/SOAPGPart2.pdf, <15.09.2006.>
- [15] ***: <http://dev2dev.bea.com/>
- [16] ***: <http://www.rosettanet.org>
- [17] ***: <http://www.sei.cmu.edu/cmm/> <21.11.2006>
- [18] ***: <http://www.sei.cmu.edu/cmml/cmml.html> <21.11.2006.>