SOA GOVERNANCE DYNAMICS IN A MULTIPARADIGM APPROACH

CLAYNOR MAZZAROLO, MsC
Universidade de Brasilia
Brasilia - Brazil
claynor@me.com
ANDRE AMARO TOFFANELLO, MsC
Universidade de Brasilia
Brasilia - Brazil
andre.toffanello@ibti.net.br
RICARDO STA CIARINI PUTTINI, PhD
Universidade de Brasília (UnB)
Brasilia - Brazil
puttini@unb.br
RICARDO MATOS CHAIM, PhD
Universidade de Brasília (UnB)
Brasilia - Brazil
rmchaim@gmail.com

Abstract

SOA refers to the coordination of people, processes and technologies within an organization that has the aim of achieving certain strategic benefits, both by increasing the return on investment (ROI) and organizational agility and by reducing the information technology governance (IT) burden. The ability of SOA to interoperate many different information systems helps to accelerate the integration of core systems. From a combination of established methods that include system thinking, Godet's strategic prospective, discrete event simulation, agent based modeling and System Dynamics comes an adaptation of Forrester's Market Growth model to formulate a decision-making governance flight simulation model to represent the interactions of intended rational policies in the study of dynamics that can arise from the complex combinations of SOA Governance factors, principles, and structures.

KEYWORDS: **System Dynamics, Competitive Intelligence, Strategic Planning Product Project; Relevant information, SOA governance, cloud computing, multi-paradigm**

I. Introduction

Service Oriented Architecture (SOA) and cloud computing provide the IT infrastructure, design patterns and other artifacts to an IT company get interoperability between software programs. A SOA Governance problem refers to the coordination of people, processes and technologies involved in each area within an organization.

SOA involves coordination between people, processes and technologies and provides some strategic benefits as it allows reducing costs, increasing organizational agility and reducing the effort in governing information technology. For enabling the interoperability of various information systems via

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

the service bus, SOA helps in the integration between legacy systems and creates the conditions for new applications and future information systems so that they can achieve greater productivity and standardization since its development will advocate the consumption of services of SOA bus.

It mean planning under uncertainty and considering risks once it aims to make processes more efficient, reduce redundancy and optimize them in order to improve supervision and giving better decision-making processes.

Software engineering methodologies are derived from the principles, best practices and standards advocated by the software industry [1][2][3][4][5]. In this paper we present a governance flight simulator that aims at service-oriented software and leverage software factory [6][7][8] practices in order to establish clear Information Technology (IT) policies that are connected to IT infrastructure. Our approach is derived from research and field experience collected during the structuring of Service-Oriented Architecture (SOA) adoption initiatives in three major governmental organizations in Brazil (Ministry of Health, Ministry of Defense/Army, and Federal Sanitary Surveillance Agency) [9][10][11]. These organizations have different maturity levels and leverage on outsourced working force for performing software development tasks.

By evolving the policies that guide decision making, there are emergencies, feedback control mechanisms and many events that must be considered in order to better comprehend the complexities in the real world and the factors that could influence a problem that must be faced. Agent based modeling, system dynamics modeling and event based modeling are discussed and exemplified as three methods that could be used as a way to give to managers and technical staff flight simulators that could aid them to cope with governance complexity and comprehend how a policy change will affect their SOA Governance system.

SOA Governance (SG) problem refers to the coordination of people, processes and technologies involved in each area within an organization to estimate problems that comes from subjective variables like credibility over time. It means planning under uncertainty and considering operational and even image risks once its goal is to make processes more efficient, to reduce redundancy and to optimize them in order to improve supervision for the creation and evolution of IT assets and resources.

To govern a SOA environment, the industry standards referenced by the SOA Reference Architecture are:

- Standards for Web protocols (W3C): HTTP, SOAP.
- Standards related to HTML technology: HTML, CSS, Javascript, JSON, AJAX.
- Standards related to XML metaformat technology (W3C): XML, XML Schema (XSD), XSLT, XQuery, XPath.
- Standards related to Web services technology (W3C, OASIS, WS-I): SOAP, UDDI, WS-*, REST.
- Standards related to business processes technology (OASIS, OMG): BPEL, BPMN.

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

- Standards related to Portlets technology (W3C, OASIS): Portlet, WSRP.
- Standards related to Java technology: JMS, JCA, RMI, JDBC.

The best practices are continuously incorporated in software development processes. A Reference Architecture provides a set of specifications derived from best practices of software project under the SOA architectural paradigm.

In addition to the recommended practices directly in the reference architecture, there are explicit references to:

- Best Practices for SOA development process applied to the development methodology in SOA.
- Design Patterns: referenced in SOA-Reference Architecture at SOA Development Methodology and other catalogs of design patterns such as SOA Design Patterns Catalogue.

In addition to the strategic goals associated with SOA, service-oriented solutions are designed to deliver value to business, particularly in the form of:

- Automated solutions for executing business processes and information processing;
- Interoperability of systems and solutions, both internally and externally;
- Performance monitoring of business processes;
- Extraction and analytical exploration of process information to obtain systems of decision support;
- Specific business objectives, associated with the corresponding service-oriented solution, are usually described in the form of requirements specification, which normally includes a description of the business processes that are subject to automation and monitoring as well as non-functional requirements (such as performance, security, reliability, usability, etc.).

Technical requirements for the development of services and technological solutions under the paradigm of service orientation:

- Interoperability and modularity: services are designed to be intrinsically interoperable, from the systematic adoption of a set of industry and design standards, in which are defined by the SOA Reference Architecture.
- Scalability and performance: services, SOA applications and the SOA Platform are designed to possess horizontal scalability, enabling to accommodate different load conditions and their variations.
- Reliability: services, the SOA applications and the SOA Platform are designed to perform accurately or provide conditions for the identification and treatment of exceptions and runtime errors.
- Resiliency and Disaster Recovery: Services, SOA applications and the SOA Platform are designed to perform in environments with clustered configuration (in the same datacenter) and replicated implementations in computational environments located in different places (in different datacenters).
- Maintainability and Extensibility: Services and SOA applications are designed to have capacity for maintenance, development and

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

extension without such activities degrading its architecture and the SOA enterprise.

Beyond the technical requirements mentioned above, specific service oriented solutions are related to the uncertainty and operational risks involved on a SOA governance initiative.

Planning under uncertainty and considering operational risks inherent of IT enterprises requires reliable tools to do better analysis and to manage IT assets in order to set policies that assure good performance and credibility to such organization.

Thus, combining SD method to agent-based modeling and event based modeling (Fig.1), a three decision making simulation methods, applies dynamic business simulation to cope some aspects of the social-economic and political environment under an IT organizational perspective via computer simulations before managers interfere in the reality. The primary gain is a better planning process that benefits from simulation before making a decision and thus moving energy and resources to achieve the goals.

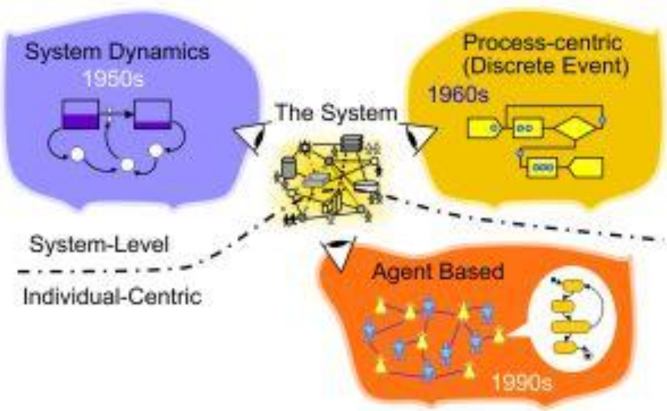


Figure 1: Multimethod simulation approach [1]

To place the issues into perspective, this paper has four sections. First, it observes a little literature review of the three simulation methods. Next, discusses ways to model SOA governance by the combination of three methods. Follows considerations on modeling SOA governance in IT companies. Finally, in the conclusion, there are considerations regarding the use of a multi-paradigm approach to address SOA Governance typical problems.

II Literature review

Modeling is based on abstraction, simplification, quantification, and analysis [15]. The modeling process is an iterative learning process that encompass many steps like getting information about the real world or the problem articulation, structure a dynamic hypothesis, formulation or transformation of the dynamic hypothesis in many diagrams, testing and the policy formulation and evaluation [26] to assure that “the model’s structure is sound and that it is capable of reproducing the dynamic symptons of the original problem” [25]

IT organizations are complex systems so interactions among many components may cause relevant differences in system’s performance. When a

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

study domain is quite complex, approaches based on equations or on other analytical techniques are impracticable or even impossible to be applied. [19]

System Dynamics (SD) method was proposed by [21] that observes that once a decision is based on the observed state of the system, there are a structure of interacting feedback loops and it implies a circularity of cause and effect, where the system produces the decision which produces the action which produces change in the system. Within the feedback loop there are level variables that mean accumulations within the system and mathematically are integrations. Rates variables that are system condition at any point of time represent the system activity and are the policy statements in, the system which defines how the existing conditions of the system produce a decision stream controlling action. Rates are not instantaneously observable once they depend only on the values of the level. Rate equation defining a rate variable is a statement of system policy that describes how and why decisions are made. “A policy statement incorporates four components - the goal of the decision point, the observed conditions as a basis for decision, the discrepancy between goal and observed conditions, and the desired action based on the discrepancy” [21].To know more details about the methodology, see [21].

Agent based modeling (ABM) method try to relate the heterogeneous behavior of the agents (different information, different decision rules, and different situations) with the macro behavior of the system [23], [18]. The agents have several interaction rules and, by simulation, it is possible to explore the emergent behavior along the time and the space [16], [17]. This modeling technique does not assume a unique component that takes decisions for the system as a whole. Agents are independent entities that establish their own goals and have rules for the decision making process and for the interactions with other agents. The agents’ rules can be sufficiently simple, but the behavior of the system can become extremely complex .To use agent based modeling, the first stage is the definition of the rules to model agents’ behavior. The criteria that can be used to the rules delimitation is based on the variables used in the dynamic model and the agent-based model.

The event based modeling (EBM) is a kind of discrete event modeling that considers that “processes we observe in the world consist of continuous changes” and that the technique “approximate continuous real-world processes with non-continuous events that you define” [15]

> The term Discrete Event is however mainly used in the narrower sense to denote "Process-Centric" modeling that suggests representing the system being analyzed as a sequence of operations being performed on entities (transactions) of certain types such as customers, documents, parts, data packets, vehicles, or phone calls. The entities are passive, but can have attributes that affect the way they are handled or may change as the entity flows through the process. Process-centric modeling is a medium-low abstraction level modeling approach. Although each object is modeled individually as an entity, typically the modeler ignores many “physical level” details, such as exact geometry, accelerations, and decelerations. Process-centric modeling is used widely in the manufacturing, logistics, and healthcare fields [15].

In order to cope with the complexities and peculiarities of IT organizations and to get a better SOA governance, the authors considered the use of these three methods combined. Agent based modeling is better when there are individual data available, SD when you have information about global dependencies, and EBM if the system can be easily described as a process[15].

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

The authors believe that these techniques are most useful particularly for organizations that are certified at level four or five of the Capability Maturity Model Integration (CMMi) Model.

III SOA Governance modeling

There are seven strategic goals of service oriented computing for the long term benefit of an IT company[20]: increase intrinsic interoperability, increase federation, increase business and technology alignment, increase vendor diversification options, increase return over investment (ROI), increase organizational agility and reduce IT Burden.

The movement to cloud computing is the disruptive change that IT departments will soon face as service oriented architecture (SOA) and cloud computing begin to have an effect on the modern enterprise [24].

A SOA Governance system places constraints on decisions; determines who has responsibility and authority to make decisions; establishes constraints and parameters that control, guide, or influence decisions; and, prescribes consequences for non-compliance. In discussing the differences between management and governance states that "management is a system and resources that are responsible for day-to-day operations" while "a governance system establishes rules and constraints" or, at a final conclusion, it could mean the ability of doing applied policies[20]. To the author, "an organization establishes governance to mitigate risk and to help advance its strategy, goals, and priorities".

SOA governance is about complexity and the process of conceptualize, develop and apply policies, to establish practices, manage risks and consider uncertainties. Sterman (2006) remember that "most people define complexity in terms of the number of components or possible states in a system" [28]. Thus, the more complex the phenomenon, the more difficult is to conduct controlled, reliable and replicable experiments and discriminate among rival hypotheses because of the large number of "physical, biological, ecological, technical, economic, social, political, and other relationships". Replication is difficult or even impossible and "decisions taken in one part of the system ripple out across geographic and disciplinary boundaries" [28].

The authors concerns about policy structure and remember that "those with power and authority routinely manipulate the policy process for ideological, political, or pecuniary purposes " and about systems principles as long time delays that "mean we never experience the full consequences of our actions and follow-up studies must be carried out over decades or lifetimes, while at the same time changing conditions may render the results irrelevant and then complexity hinders the generation of evidence"[28]. So, "effective interventions require changes in the beliefs and behaviors of a large majority of actors involved, supported by complementary changes in education, incentives, and institutions" [28].

Policies come with resistance from the people and "most cases of policy resistance arise from dynamic complexity—the often counterintuitive behavior of complex systems that arises from the interactions of the agents over time". Among the elements of dynamic complexity people find most problematic are

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

feedback, time delays, and stocks and flows, because systems are constantly changing, are tightly coupled, governed by feedback, nonlinear once “effect is rarely proportional to cause” and “also arises as multiple factors interact in decision making” [28].

Modeling is based on abstraction, simplification, quantification, and analysis [15]. The modeling process is an iterative learning process that encompass many steps like getting information about the real world or the problem articulation, structure a dynamic hypothesis, formulation or transformation of the dynamic hypothesis in many diagrams, testing and the policy formulation and evaluation [26] to assure that “the model’s structure is sound and that it is capable of reproducing the dynamic symptons of the original problem” [25]

SOA Governance is a multifactor approach and SD combined to ABM and EBM gives the capability to better manage risks factors to help managers to better know the productivity, the comprehension of business growth (revenues), the results of marketing campaigns (credibility) and the wealth of the company so they can identify and analyze trends. A stochastic programming model for a SOA Governance is dynamic since the information on the actual value of uncertain parameters is revealed in stages.

IV Modeling SOA Governance on IT Companies

The preceding concepts will be exemplified in order to show their utility and how it can be used by a SOA governance research. As it is about interactions within the company, within the market, and between the two, in this paper, a model with no influences of the outside can be made by system dynamics modeling, market dynamics can be understood by an agent based model, processes analysis can be done by event based modeling in order to represent their behavior, the social-economic and political environment to provide deeper insights by simulation experiments and the flows of goods, services, money and information.

O seguinte modelo de referência de governança SOA, adaptado de IBM (2012),

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

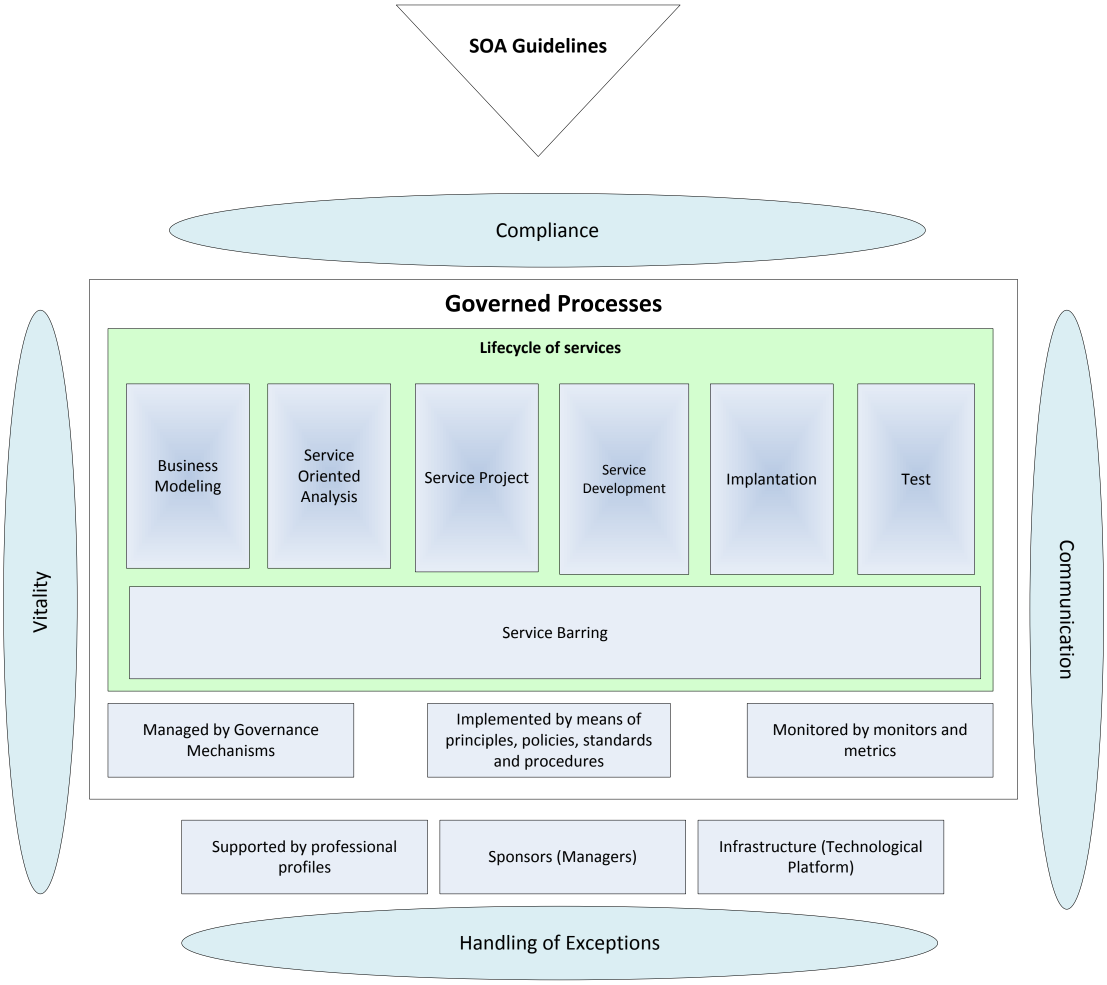

SOA Guidelines
Compliance
Governed Processes
Lifecycle of services
Business Modeling
Service Oriented Analysis
Service Project
Service Development
Implantation
Test
Service Barring
Managed by Governance Mechanisms
Implemented by means of principles, policies, standards and procedures
Monitored by monitors and metrics
Supported by professional profiles
Sponsors (Managers)
Infrastructure (Technological Platform)
Handling of Exceptions
Vitality
Communication

Figure 2: Reference model for SOA Governance
Source: Adapted from IBM (2012)

A. SD Approach

System Dynamics modeling gives a way to address SOA governance problems for IT organizations and to explain phenomena and the structure of the system via stock and flow representation and the causation between factors. Fig.2 shows the rich picture of a SOA governance problema and try to relate its causal loops with the previous reference model presented on figure 2:

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

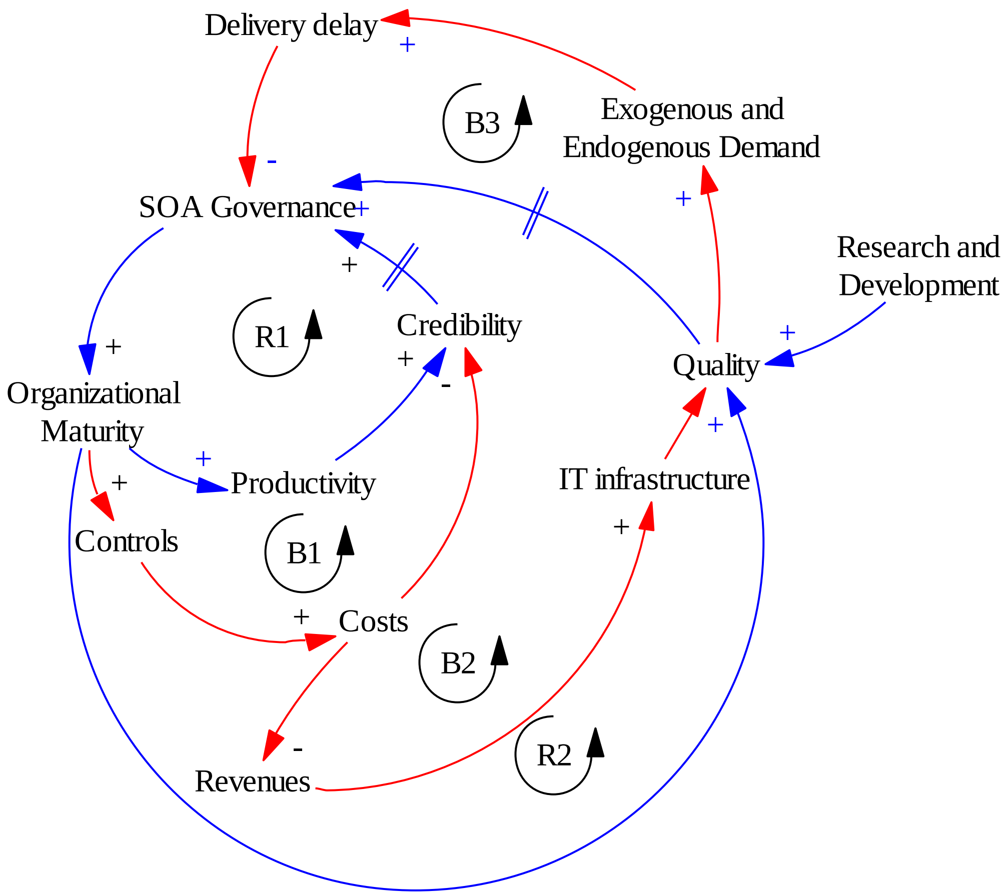


Figure 3: Cause and effects on factors for a SOA Governance Model

Follows the description of the reinforcing and balancing loops governing its dynamics:

R1: SOA governance, organizational maturity, productivity, credibility. This loop focuses on productivity as a consequence of the adoption of an organizational maturity framework and the consequent enhance on productivity.

In relation to the reference model for SOA governance, they represent SOA guidelines, namely the vision of what is to be achieved with the adoption of SOA for the planning and development of technological solutions. The Adoption Program is the reference document that provides, among other things, a set of business objectives from which will develop a collection of Services and Composition Application to be ruled over time.

R2: SOA governance, organizational maturity, quality. This loop is about the quality obtained as a consequence of the adoption of an organizational maturity framework. In the reference model for SOA governance model, governed processes and product life cycle and established services mean:

Governed Processes - Represent the boundaries of SOA Governance to implement the mechanisms and processes of governance, monitoring and evaluation metrics.

Lifecycle Services - Defined by SOA Development Methodology [6], they

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

provide methods to develop the Business Services, specify and describe the stages of development of a service from the Business Process Analysis until the provision of services on Buses Services and use.

B1: SOA governance, organizational maturity, controls, costs, credibility. This loops considers the enhance on costs because the growth of controls because the adoption of the organizational maturity framework. In relation to the reference model for SOA governance, it has to do with the processes of governance mechanisms, principles, policies, standards and metrics so that they can be achieved as well as maintaining levels of credibility and desirable organizational maturity for the operation of the business:

Governance Mechanisms - Provide the software infrastructure of support necessary to deploy and operate the SOA Governance.
Principles, Policy and Standards - The underlying rules established for the development and deployment of technological solutions in SOA and define how projects and technological solutions in SOA will be developed. These principles, policies and standards are available in CDS' SOA Reference Architecture.
Metrics – they conceptualize metrics and are developed to measure the efficiency of the SOA initiative, having an important role in governance. The key metrics of the governance process are artifacts resulting from the development of services coming from SOA Development Methodology.

B2: SOA governance, organizational maturity, controls, costs, revenues, IT infrastructure, quality. This loop considers that IT infrastructure investments cause the growth on quality. In relation to the reference model for SOA governance has to do with the technological infrastructure to support SOA initiatives:
Infrastructure (technology platform) - The infrastructure has a significant number of hardware and software elements that enable the implementation of technological solutions from the SOA projects, such as governance tools, middleware, application servers, monitoring and management, security applications and others that basically provide adequate conditions of reliability and availability for its operationalization. The settings on the architecture of the infrastructure can be defined in a document of Technological Corporate Architecture.

B3: SOA governance, organizational maturity, controls, costs, revenues, IT infrastructure, quality, Exogenous and endogenous demand, delivery delay. In relation to the reference model for SOA governance, particularly in organizational maturity and quality perspectives, this causal loop evidences professional profiles, managers and sponsors:
Professional Profiles - The professional profiles represent the expertise necessary for the human capital implement service development and SOA initiatives governance. The professional profiles are defined in CDS' SOA Development Methodology.
Sponsors and Managers - Decision makers in tactical or strategic level responsible for adapting the management model and defining actions

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

associated with management and development of information systems and deploying SOA Technological Solutions.

In Fig.2, the performance of a SOA governance (SG) program increases the organizational maturity, which increases productivity, which increases credibility which causes better results in terms of SG. Organizational maturity also increases controls that could decrease credibility and thus reducing SG initiatives. Less revenues means less investments in IT infrastructure that could affect credibility of the company.

Productivity, costs and “IT infrastructure” are factors to be managed in order to get a good credibility. They can influence the establishment of good practices for better SOA governance and managers must decide what to do in order to maintain it in adequate levels and thus get more credibility. SD, ABM and EBM simulation can help them with tools that implement management flight simulators [12]. Also is useful to develop an organizational maturity model and a dynamic scorecard model in order to know the actual stage of the organization in terms of their SOA program and to get better management strategies.

Thus, Fig.3 shows a simplified SOA governance model that puts together the capacity dynamics, the revenue process and their interaction to infrastructure all of them accumulating a backlog of agnostic and non-agnostic services that requires the capacity to adjust to the services consumption. More than the model presented in Fig.3, the full model contemplates the delivery delay, production dynamics and capacity adjustment that could not be shown in this paper.

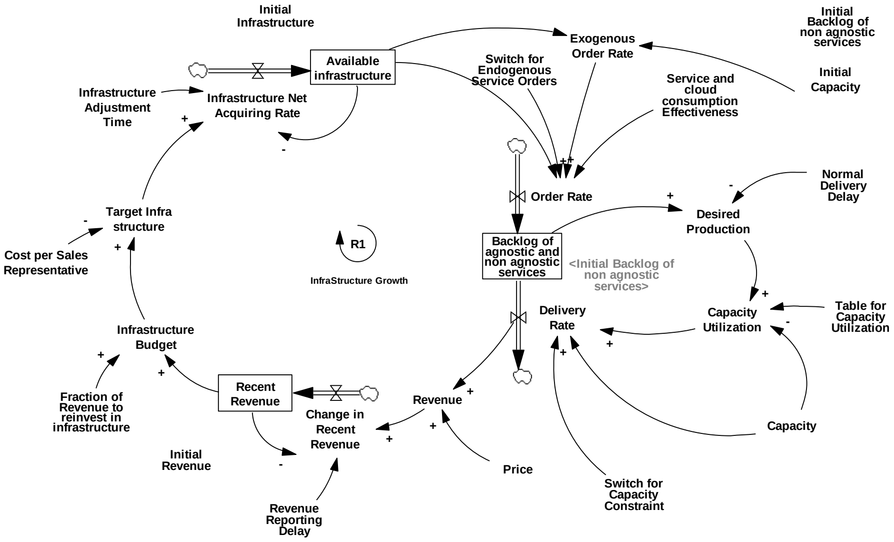

Initial Infrastructure
Available infrastructure
Infrastructure Adjustment Time
Infrastructure Net Acquiring Rate
+
-
Target Infra structure
-
+
Cost per Sales Representative
R1
InfraStructure Growth
Infrastructure Budget
+
+
Fraction of Revenue to reinvest in infrastructure
Recent Revenue
Change in Recent Revenue
Initial Revenue
-
+
Revenue Reporting Delay
Revenue
+
+
Price
Exogenous Order Rate
Switch for Endogenous Service Orders
Service and cloud consumption Effectiveness
Initial Backlog of non agnostic services
Initial Capacity
++
Order Rate
+
Desired Production
-
Normal Delivery Delay
Backlog of agnostic and non agnostic services
<Initial Backlog of non agnostic services>
Delivery Rate
+
Capacity Utilization
-
Table for Capacity Utilization
+
+
Capacity
Switch for Capacity Constraint

Figure 4: A SOA governance model

Fig.3 diagram explain in more details how policies can be quantitatively observed even though some variables are qualitative by nature. Based on the dynamics of a particular IT company under research, some examples of policies were implemented in the model as:

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

- A one quarter-year delay is assumed in the budgeting process;
- Twenty percent of the organization's budget is allocated to the infrastructure;
- The fully loaded cost per infrastructure representative is $8000/month;
- An average delay of 18 months is required to adjust the infrastructure to target levels;
- Capacity utilization saturates 25% above normal, implying normal utilization yields 80% of maximum output (Fig.4).

After a simulation experiment graphic in Fig.4 was produced showing that the system that was in equilibrium alters its structure after some policies were changed, giving another behavior.

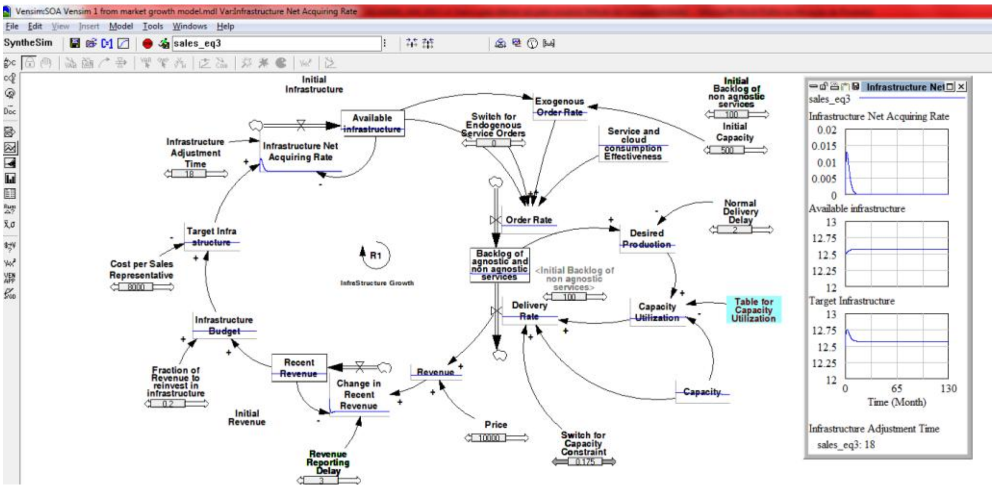

VensimSOA Vensim 1 from market growth model.mdl Var:Infrastructure Net Acquiring Rate
File Edit View Insert Model Tools Windows Help
SyntheSim
sales_eq3
Initial Infrastructure
Available infrastructure
Infrastructure Net Acquiring Rate
Infrastructure Adjustment Time
18
Target Infra structure
Cost per Sales Representative
8000
R1
InfraStructure Growth
Infrastructure Budget
Fraction of Revenue to reinvest in infrastructure
0.2
Recent Revenue
Change in Recent Revenue
Initial Revenue
Revenue Reporting Delay
3
Revenue
Price
10000
Exogenous Order Rate
Switch for Endogenous Service Orders
0
Service and cloud consumption Effectiveness
Initial Backlog of non agnostic services
100
Initial Capacity
500
Order Rate
Backlog of agnostic and non agnostic services
<Initial Backlog of non agnostic services>
100
Delivery Rate
Desired Production
Normal Delivery Delay
2
Capacity Utilization
Table for Capacity Utilization
Capacity
Switch for Capacity Constraint
0.175
Infrastructure Net
sales_eq3
Infrastructure Net Acquiring Rate
0.02
0.015
0.01
0.005
0
Available infrastructure
13
12.75
12.5
12.25
12
Target Infrastructure
13
12.75
12.5
12.25
12
0
65
130
Time (Month)
Infrastructure Adjustment Time
sales_eq3: 18

Figure 5: A SD simulation experiment

Risk management in an IT organization is a decision problem. As in economics an external factor like interest rates can give insights over the decisions and to comprehend the behavior of the system over a fixed value or by a probability distribution that could explain it. SD was useful once the model was built up of probability distributions and rely to some degree on obeying important probability and statistical rules.

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

B. Agent based simulation and Event based simulation

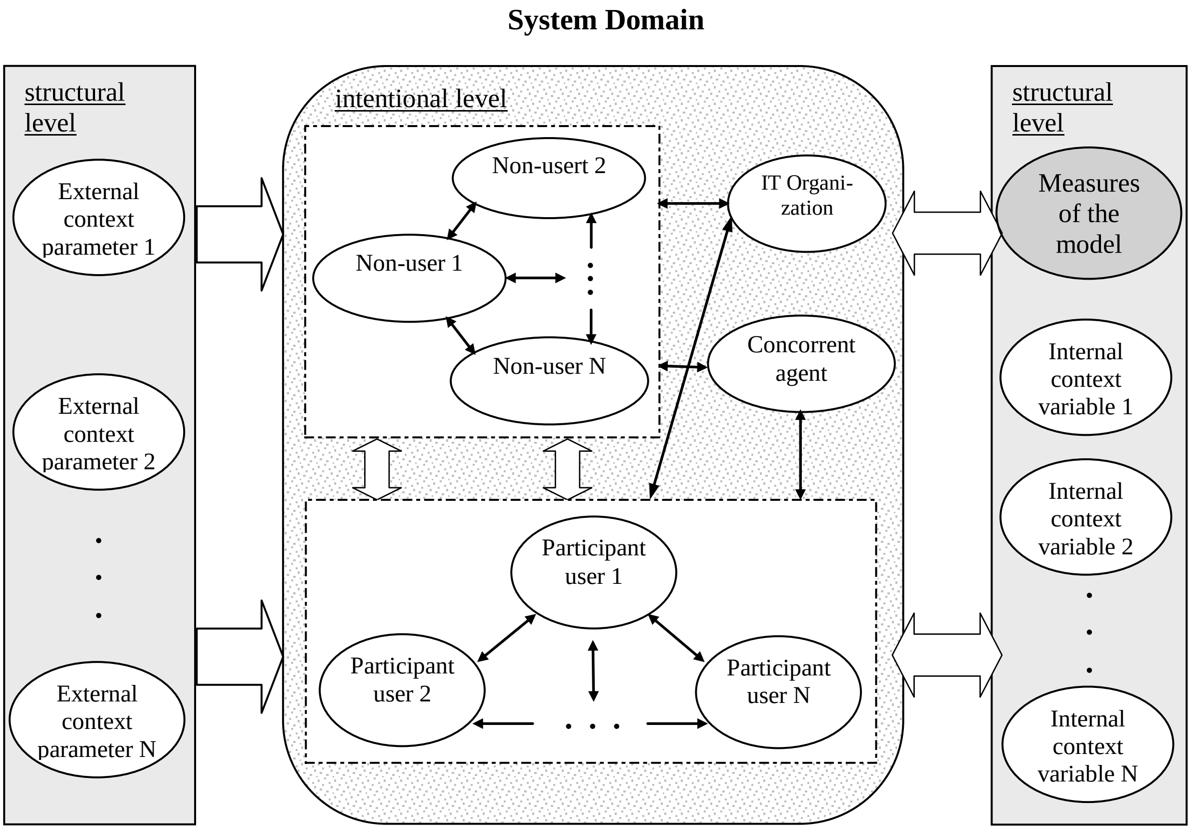
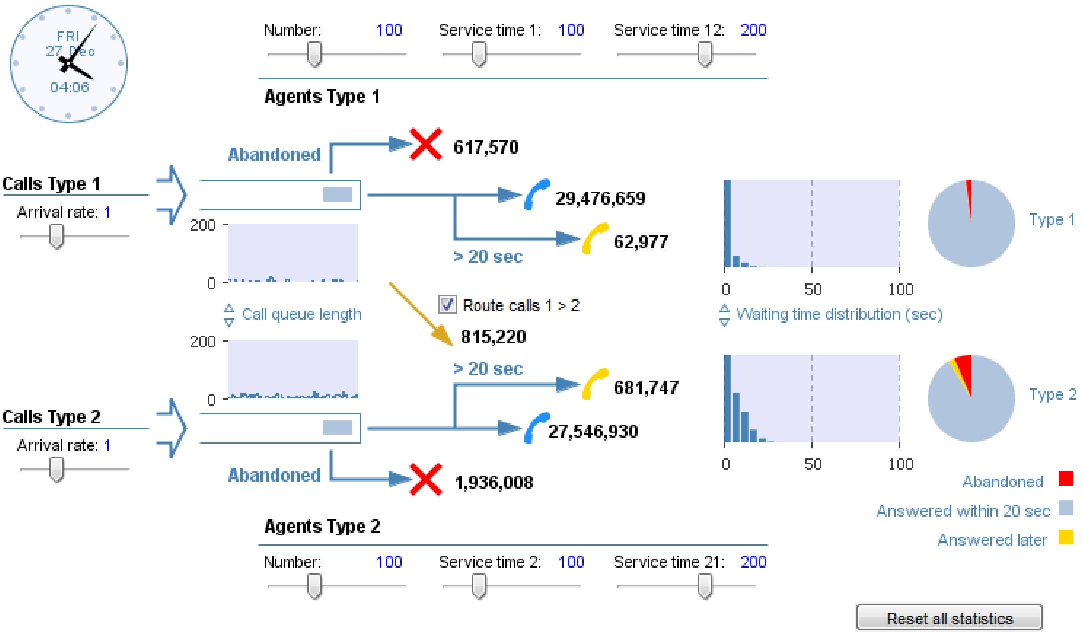


Figure 6 – Generic conceptual model to study IT organizations governance by an Agent based modeling method [27]

This model was originally developed for regulatory governance analysis of sectors under regulation [27]. The conceptual model is generic and, consequently, it is useful to structure different IT organizational scenarios. The intentional level (action level), where the interactions among the agents occur, is differentiated from the structural or contextual level. The structural level indicates the contexts where the interactions happen, e.g., the circumstances that limit, amplify and determine the interactions among the agents and with the environment. Moreover, structural level is the level where the emergent phenomenon takes place. It is a higher level comparing to the intentional level where the agents interact. The basic principle that guides the model is that all interactions have an intention or a set of intentions[27].

Many organizational process are internal and there are few that involves managing the relationship with clients of an organization. In order to illustrate an agent based simulation combined to an event based simulation, a very simplified call center process will be considered. The authors used an anylogic [15] simulation model to study a contact center process behavior considering that there were two “types” of calls arriving at a contact center.

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)



Number: 100 Service time 1: 100 Service time 12: 200

Arrival rate: 1

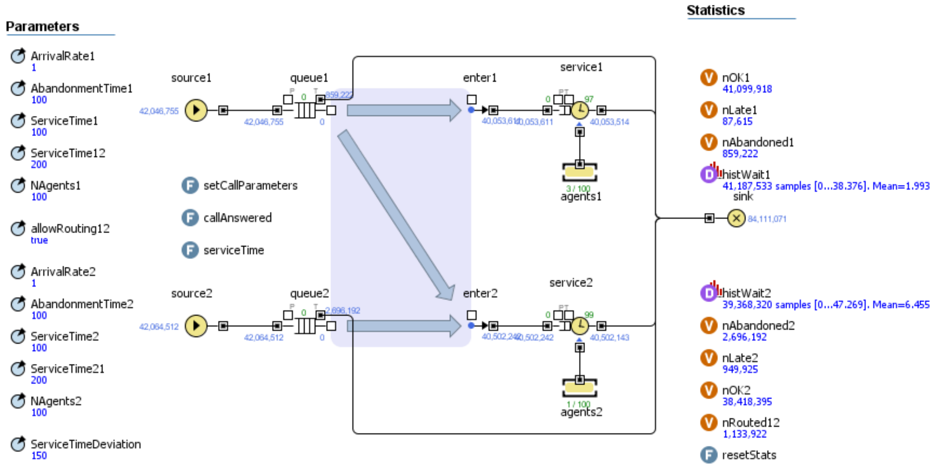
☑ Route calls 1 > 2

Arrival rate: 1

Number: 100 Service time 2: 100 Service time 21: 200

Figure 7: An event based simulation combined to and agent based simulation [15]

As stablished by anylogic [15], the model use a poisson distribution with two known arrival rates and there is a queue for each call type and some callers abandon from the queue after a certain time according to an exponential distribution with mean calculated from real experiments by using statistics methods. There are 2 agent groups, one trained to handle calls of type 1 with a known mean service time and group 2 is trained to handle calls of type 2 with a knowable mean. However the agents are also cross-trained such that group 1 agents can handle type 2 calls with a mean service obtained from the reality Typically the cross-trained agents will have lower performance with their non-favorite skill. The logic for routing the calls can take various forms. In this implementation when the call is processed it is routed to the "native" agent, if there is one available, otherwise, the call is routed to the "alien" agent, again if the latter is idle.The output metrics in this model are the queue lengths and "service levels" for both call types. By service level we mean the percent of incoming calls answered by an agent within 20 seconds.After simulating for a while, results on Fig.7 is considering the logic of the business process and the factors.



Parameters

ArrivalRate1 1
AbandonmentTime1 100
ServiceTime1 100
ServiceTime12 200
NAgents1 100
allowRouting12 true
ArrivalRate2 1
AbandonmentTime2 100
ServiceTime2 100
ServiceTime21 200
NAgents2 100
ServiceTimeDeviation 150

Statistics

nOK1 41,099,918
nLate1 87,615
nAbandoned1 859,222
histWait1 41,187,533 samples [0...38.376]. Mean=1.993
histWait2 39,368,320 samples [0...47.269]. Mean=6.455
nAbandoned2 2,696,192
nLate2 949,925
nOK2 38,418,395
nRouted12 1,133,922

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

Figure 8: A simulation experiment of ABM and EBS [15]

Additionally, in relation to their expectation, a particular analysis of SOA Governance of its stage vision in the adoption of the architecture is required.

Architecture and Information Technology

This domain is focused on the technological architecture composed of topology, technical interoperability, use of design patterns, a subject to the model of organizational reference architecture, the compliance with the rules and policies, the level of adoption of services, the experience in the use of SOA and its compliance criteria as well as the quality of the typical elaborated artifacts.

It gives focus on the structure of the modeled information, on the method of access to enterprise data, the abstraction of data access from the functional aspects, the characteristics of data, the data processing capabilities, handling of identifiers, security credentials, knowledge management, business information model and content management. Thus involves a clear and correct definition of data and their semantic (master data models), canonical models, schema definition, the management of application owners and their organization, and metadata support processes; all facing the definition of a model based on primary sources of data services with proper integration of business information in support of decision-making. Another important factor is the proper use of interoperable models such as XML as a standard for messaging.

It is through this domain that the quality evaluation of current services and the different services layers in SOA are carried out, that is to say, the evaluation of stages of SOA adoption, such as the efficient decomposition and coupling of business logic as a facilitator of the integration of organizational applications. The integration processes and its direct support of business processes through the service orchestration in an assisted interaction to the user.

As an application, this domain also addresses the proper use of the fundamental principles that govern the proper use and according to SOA in the CDS. Worth highlighting: the standardization of service contracts, the reduction of dependencies between services and consumers at low coupling, the use of abstraction principles and autonomy in forming capabilities or methods of service contracts, the constant concern with the analysis and application of the reuse principle by implementing or decomposition of agnostic logic of services into reusable components, and the dealings of independent principles of the state aimed at improving the service performance by the services available at all levels. Finally, to support the application of these principles adequately, it is expected a greater capacity for interpretation and discovery with the(s) inventory(s) of existing services, which must be properly managed and governed.

Infrastructure and operations

This domain examines the quantity and quality of existing IT infrastructure in the aspects of availability and performance, with proper and effective management, with a focus on communication through a common layer of high availability, reliable and secure for all services, so that they can be

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

extended by additional components layers to other layers (for example, monitoring or enforcement of security activities).

It deals primarily with the analysis of the management characteristics through greater complexity controls to the control infrastructure based on traditional models. To do so, it contains related capacities to infrastructure and the set of services and tools listed in the field of Technological Architecture and Information.

The management of IT assets is made through records, controls, history or versions, besides the use of practices for configuration and deployment. A detailed description of the integration processes, the infrastructure involved, continuous monitoring by analyzing operational metrics, records and management of incidents and events and issues relating to security are topics related to that domain.

It also contains the capabilities related to the aspects of maintenance of solutions based on service-oriented architecture.

Maturity Levels

As well as CMMI is a natural way of market to understand and assess the maturity of IT in general, several studies have established mappings from level of specific maturity for service-oriented architecture, such as SOMM, OSSIM, GARTNER, Oracle's SOA Assessment, IBM SIMM and ISOAMM, NSOAMM). In these works, the main challenge was to be more than just a categorization according to some similarities from CMMI, justifying why SOA - as a technique - can, for example, be instantiated as defined, repeatable and optimized. "It is often easy to do because of the open and interpretable nature of the descriptions of each maturity level CMMI " (IBM, 2009).

Adhering to the understanding of the CMMI market, our proposed model fits into five levels of maturity:

1. Initial: Characterized by no SOA planning, incidental SOA, use of silo technology and SOA based potential. The processes are usually ad hoc and chaotic. Generally does not provide a stable environment. The success in these organizations depends on the competence and the heroism of people, and not on the use of proven processes. Also on this level are often produced products and services of work. However, they frequently exceed the budget and schedule projects. Other characteristics of this level are: tendency to abandon processes in times of crisis and not be able to repeat past successes.
2. Managed: Characterized by the establishment of the initial requirements for SOA, such as data sharing, the initial learning of the principles and the application of SOA in short-term projects to demonstrate value. Furthermore, it focuses on communication and interoperability between systems, as well as the use of technologies with low compliance with requirements for long-term use. Processes are planned, executed, measured and controlled. Once requirements are managed, projects are implemented and managed in accordance with documented plans. Requirements, commitments, processes, work products, and services are managed and reviewed by stakeholders, as well as the status of service delivery.
3. Defined: Characterized by the expansion of services and collaboration beyond the exhaustive use of SOA in daily projects to build a service

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

portfolio. These services are focused on reuse and orchestration. Thus, at this level is set a management of the proliferation of service and the tactical encapsulation of the requirements in unique functional contexts. Processes are well characterized, understood, and are described in standards, procedures, tools and methods. In previous level patterns, descriptions of processes and procedures may vary according to each project. At level 3, these measures are tailored to a set of guidelines of the organization, generating processes as a result of greater consistency. Still at level 3, processes are also often written in more detail, being managed more proactively using an understanding of the interrelationships of activities and detailed measures, products and services.

4. Quantified: Characterized by measuring and evaluating the visible impact of SOA to the business and by extending services for process automation, with the breakdown of silos organization. As the organizational model gets more benefits, business is enabled with a considerable lower risk. The sub-processes are evaluated and controlled by using statistical and other quantitative techniques. The quantitative objectives for quality and performance are established and used as criteria in the management of sub-processes. Quantitative objectives are based on the need of the customers, the end users, organization, and process implementers; where quality and performance are managed throughout the life-cycle processes. With the support of metric measurement is then possible to analyze the performance/quality of a particular process, being possible even to predict performance.
5. Optimized: It is characterized by strong realization of discoverability. Thus, SOA is proliferated. The focus on measurement is extended to how it is used, that is, to conduct a feedback for improvement and the inclusion of measures, not only business, but also for the measured use of services and reuse. That done, the next step is to summarize indicators of how SOA is being adopted and what benefits are being achieved. The processes are continuously improved based on a quantitative understanding of the common causes of variation inherent in them. The focus of this level also gives a performance boost by improving and technological innovation. The effects of deployed process improvements are measured, evaluated and compared, with the objective of improving processes in cycles. In the previous level there is a concern with the treatment of the common causes of variation, in order to analyze alternatives for improving performance.

To understand the heuristics in order to apply the evaluation, these two representations are important because they establish the essential characteristics in the inferences made - and they were passed on to the system: the situation of Evolution or Stabilization.

In this sense, our framework involves these two representations by Domain and differ them according to an inherent aspect of continuous improvement process, namely:

- In a specific domain, the CDS may be in Evolution (positioned according to the levels of continuous representation), that is, under achievements aiming at stabilization at a recognized level of maturity.

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

- In a specific Domain, the CDS may be in Stabilization (due to the recent achievement of representation levels on stage), that is, in the adjustment phase within a recognized maturity level.

The result of the Risk Assessment of SOA Initiatives aim to support the development of both the strategic goal planning and the actions prioritization of SOA adoption by CDS.

The base of the Risk Assessment is the result of Maturity Assessment, whose first among five preset in the model was achieved by the evidence evaluated in this analysis. It was recommended that the strategic goals were based on definition considering the goal of achieving higher levels of maturity related to deadlines and actions.

Finally, the level of maturity combined with risk analysis can be considered a starting point for planning actions for SOA initiatives. Also, is considered of key importance for Governance, which is considered one of the Governance Guidelines indicated by the Model.

After presenting the results of risk and maturity, it comes the time to indicate the functional model of SOA governance.

V Conclusions

The research is multidisciplinary and interdisciplinary by nature and the article presents part of the literature review and methodological strategy to develop the research. The multi-paradigm approach is suitable to model subjective factors to simulate the complexity of IT organizations systems considering their risks and uncertainties in a SOA Governance program.

On a SOA governance management problem, SD combined to an agent based modeling (ABM) method can help to relate the heterogeneous behavior of the agents (different information, different decision rules, and different situations) with the macro behavior of the system [23], [18].

Como se pode verificar no modelo de referência SOA da figura 8,

The agents have several interaction rules and, by simulation, it is possible to explore the emergent behavior along the time and the space [16], [17]. This modeling technique does not assume a unique component that takes decisions for the system as a whole. Agents are independent entities that establish their own goals and have rules for the decision making process and for the interactions with other agents.

The agents’ rules can be sufficiently simple, but the behavior of the system can become extremely complex .To use agent based modeling the first stage is the definition of the rules to model agents’ behavior. The criteria that can be used to the rules delimitation is based on the variables used in the dynamic model and the agent-based model.

The modeling process of an agent-based model defines its individual components, as a bottom-up approach. The definition of the agents’ behaviors is extremely important for a good representation of a SOA-Governance model. Besides, there must be a very good equivalence between the system under analysis and the conceptual model to guarantee great consistency to the agent-based model and reliability from the simulation results[22].

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

Once credibility is being modeled, population dynamics and client satisfaction studies focus on the population dynamics of an IT organizations that has, among others, rates of complaint, satisfaction and efficiency that must be considered in assessing credibility and estimating opportunities to optimize business process in order to forecast productivity enhancement opportunities quantitatively. This way, event based modeling,
a kind of discrete event modeling that considers that “processes we observe in the world consist of continuous changes” and that the technique“ approximate continuous real-world processes with non-continuous events that you define” [15].

In order to cope with the complexities and peculiarities of IT organizations and to get a better SOA governance, the authors are considering the use of these three methods combined. To address SOA governance problem, agent based modeling is better when there are individual data available, SD when you have information about global dependencies, and EBM if the system can be easily described as a process[15]. The authors believe that these techniques are most useful particularly for organizations that are certified at level four or five of the Capability Maturity Model Integration (CMMi) Model.

This way, the research is being conducted by the authors and it combines methods and techniques to study IT organizations models and the influence of subjective factors over it. It is projected to combine structural model and internal model to better mimic the real system.

The research is in progress and Fig.8 shows the software planned to be developed in order to combine these methods and to get a multi-paradigm approach on modeling credibility, including a fuzzy logic engine in order to better manage risks and uncertainties over the mapping of probabilities and creating different combinations of logic that can be applied to the model.

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

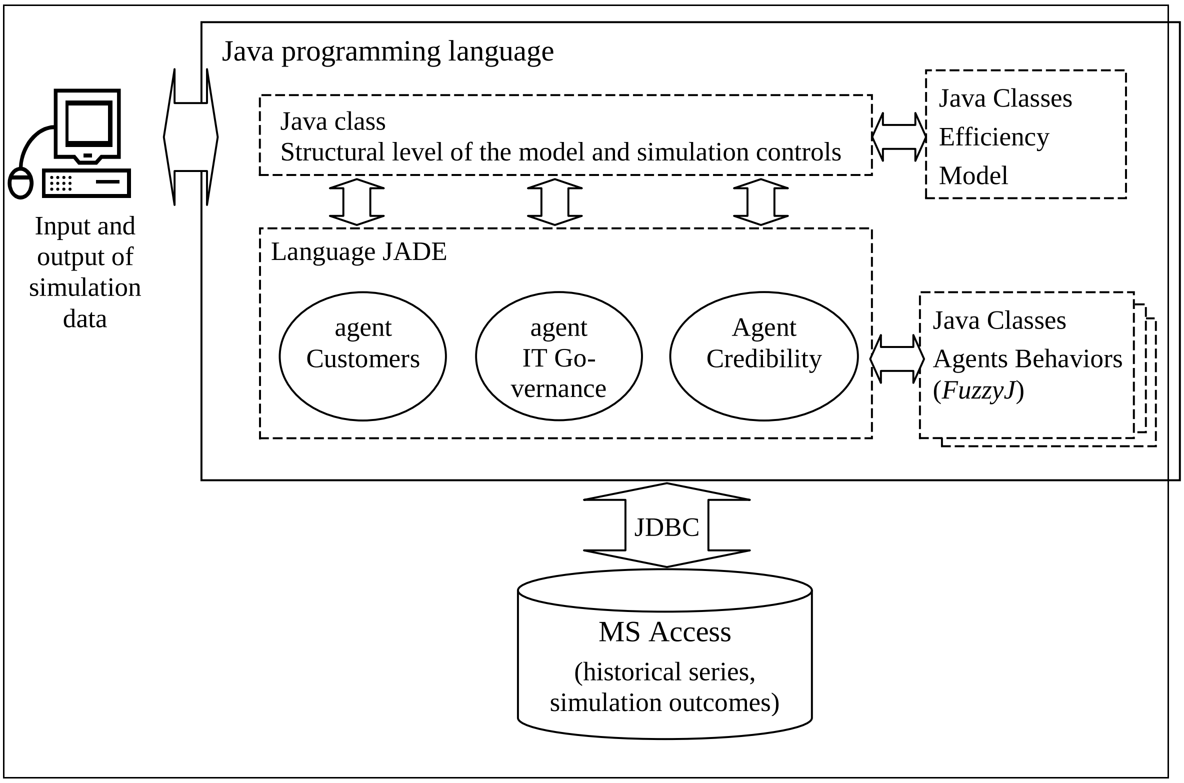

Java programming language
Java class
Structural level of the model and simulation controls
Java Classes
Efficiency
Model
Input and output of simulation data
Language JADE
agent Customers
agent IT Go-vernance
Agent Credibility
Java Classes
Agents Behaviors
(FuzzyJ)
JDBC
MS Access
(historical series, simulation outcomes)

Figure 9: The combination of many approaches in order to develop a SOA-Governance Information System

The authors identified the main actors and the methodology to proceed the modeling recommendations identified on the literature review. The software to be produced will consider efficiency, effectiveness, credibility and productivity based on operational and image risks and compliance to help Brazilian IT companies on anticipating problems, better produce policies and determine the growth and stability of the enterprise by comprehending how a policy change will affect the total system.

It’s also possible to say that, before starting a software engineering development project by defining functional requirements and structuring use cases, these techniques are useful to model the behavior or processes and comprehend how distinct functions can interact with each other. Thus, the non-functional requirements could be better elicited once it engineers the system as a whole using systems principles.

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

REFERENCES

1. ANVISA, Service-oriented Methodology (April 2009) (in Brazilian Portuguese)

2. ANYLOGIC. Available at: < http://www.anylogic.com/>. Acessed in: January, 08th, 2013.

3. Brown, A., Lopez, A., Reyes, L.: Practical Experiences with Software Factory Approaches in Enterprise Software Delivery. In: The Sixth International Conference on Software Engineering Advances, ICSEA 2011, Barcelona, Spain, October 23 (2011)

4. CDS/Brazilian Army, SOA Development Methodology (preliminary) (December 2012) (in Brazilian Portuguese)

5. CHAIM, Ricardo Matos. Combining ALM and System Dynamics in IT organizations. In: 24th International Conference of System Dynamics Society, 2006a. Proceedings of the 24th International Conference The Netherlands: Wiley Inter Science, 2006. Available at: <http://www.systemdynamics.org/conferences/2006/ proceed/papers/ CHAIM315.pdf>. Acessed in: 30 de setembro de 2006.

6. CHAIM, Ricardo Matos. 2007. Gestão das informações sobre riscos de ativos e passivos previdenciários em fundos de pensão: associação entre a Dinâmica de Sistemas e o Asset and Liability Management (ALM). PhD dissertation, Information Science School, University of Brasilia, Brazil.

7. CHAIM, Ricardo Matos; STREIT, Rosalvo Ermes. IT organizations governance: combining SD, Agent based Modelling and fuzzy logic to adress Dynamic Asset and Liability Management. In: 26th International Conference of the System Dynamics Society, 2008, Athens, Greece. Proceedings of the 26th International Conference of the System Dynamics Society. Hoboken, NJ, Estados Unidos : Wiley Blackwell, 2008. p. 42-42.

8. Clements, P., Northrop, L.: Software Product Lines: Patterns and Practices, 3rd edn. Addison Wesley (2001)

9. DATASUS/Ministry of Health, SOA Development Methodology (March 2012) (in Brazilian Portuguese)

10. EDMONDS, Bruce. Simulation and complexity: How they can relate. Centre for Policy Modelling Discussion Papers, CPM Report No.: CPM-03-118, 2003.

11. Erl, T.: SOA Principles of Service Design. Prentice Hall (2009)

12. Erl, T.: Service-Oriented Architecture – SOA Concepts, Technology, and Design. Prentice Hall (2008)

13. ERL, Thomas; Bennett, Stephen G; Carlyle, Benjamin ;Gee, Clive ; Laird, Robert ; Manes, Anne Thomas; Moores, Robert; Schneider, Robert; ,Shuster,

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

Leo; Tost, Andre ;Venable, Chris; Santas, Filippos. SOA Governance: Governing Shared Services On-Premise and in the Cloud. Boston, Pearson Education,Inc. 2011.

14. FORRESTER, Jay W. Market growth as influenced by capital investment. Boston, Industrial Managament Review, MIT Sloan School, vol. 9, no 2, 1968.

15. GILBERT, Nigel. Simulation: An emergent perspective. Draft Paper, Department of Sociology, University of Surrey, 1995.

16. Gu, Q., Lago, P.: Guiding the selection of service-oriented software engineering Methodologies. Service Oriented Computing and Applications, SOCA 5, 203–223 (2011)

17. Hotle, M., Landry, S.: Application Delivery and Support Organizational Archetypes: The Software Factory, Gartner Research Report G00167531 (May 2009)

18. IBM – “SOA Readiness Assessments”, 2012. Disponível em: https://www14.software.ibm.com/iwm/web/cc/flash/soa/en/us/Tool.htm

19. Jeston, J., Nelis, J.: Business Process Management, 2nd edn. Practical Guidelines to Successful Implementations. Butterworth-Heinemann (2008)

20. LEMPERT, Robert. Agent-based modeling as organizational and public policy simulators. Proceedings of the National Academy of Sciences of the United States of America (PNAS), vol. 99, suppl. 3, p. 7195 – 7196, 2002.

21. LINTHICUM, David S. Cloud computing and SOA convergence in you enterprise. Boston, Pearson Education, 2011.

22. MacKenzie, C., Laskey, K., McCabe, F., Brown, P., Metz, R. (eds.): OASIS Reference Model for Service Oriented Architecture Version 1.0. OASIS Standard (October 12, 2006)

23. MORECROFT, John. Strategic Modelling and Business Dynamics. NJ-USA, Wiley,2007.

24. Project Management Institute, A Guide to the Project Management Body of Knowledge: PMBOK Guide, 4th edn. Project Management Institute (2008)

25. Ramollari, E., Dranidis, D., Simons, A.: A survey of service oriented development methodologies. In: Proceeding of Second European Young Researchers Workshop on Service Oriented Computing, pp. 75–80. University of Leicester, Leicester (2007)

26. STERMAN, John D. Business Dynamics: Systems Thinking and Modeling for a Complex World. Boston, Irwin McGraw-Hill, 2000.

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)

27. STERMAN, John D. Learning from Evidence in a Complex World. March 2006, Vol 96, No. 3. American Journal of Public Health.

28. STREIT, Rosalvo E. Um modelo baseado em agentes para a análise da governança regulamentar do sistema financeiro. PhD dissertation, Management School, Federal University of Rio Grande do Sul, Brazil, 2006.

29. The Open Group - The Open Group Architecture Framework – TOGAF, version 9.1. The Open Group Standard (2011)

Work supported by Fundação de Apoio à Pesquisa do DF (FAPDF) and Instituto Brasília de Tecnologia e Inovação (IBTI)