![JASSS logo]

Stefan König, Sebastian Hudert and Torsten Eymann (2010)

Socio-Economic Mechanisms to Coordinate the Internet of Services: The Simulation Environment SimIS

Abstract

Visions of 21st century information systems show highly specialized digital services and resources, which interact continuously and with a global reach. Especially with the emergence of technologies, such as the semantic web or software agents, intelligent services within these settings can be implemented, automatically communicating and negotiating over the Internet about digital resources without human intervention. Such environments will eventually realize the vision of an open and global Internet of Services (IoS). In this paper we present an agent-based simulation model and toolkit for the IoS: 'SimIS: Simulating an Internet of Services'. Employing SimIS, distributed management mechanisms and protocols can be investigated in a simulated IoS environment before their actual deployment.

Keywords: Multi-Agent Simulation, Internet, Simulation Tools

Introduction

1.1   Visions of 21st century information systems show highly specialized digital services and resources, which interact continuously and with a global reach. Today's Internet of mainly human interaction evolves to a global, socio-technical information infrastructure, where humans as well as automated software units (software agents acting on their behalf) continuously interact to exchange data and computational resources. Especially emerging technologies such as the semantic web or software agent technology foster the implementation of intelligent services that communicate and negotiate with each other employing the Internet as a communication infrastructure. Such environments will eventually consist of millions of service providers, consumers and a multitude of possible intermediaries like brokers, workflow orchestrators and others, thus forming a global economic environment. Electronic services and resources traded on a global scope will ultimately realize the vision of an open and global Internet of Services (IoS). The IoS Vision as it is developed and deployed now is mainly based on a specific set of technical standards, such as Web Services (WS), the Universal Description, Discovery and Integration (UDDI) standard or SOAP messages. These widely agreed upon conventions enable the technical definition of services and interfaces (WS), give a formalized process of discovering and binding of these services (UDDI) and define message formats for service-based communications (SOAP) (GUDGIN 2003). Intelligent agents, acting as a management layer within such IoS settings, build on these standards as a technical base for communication and negotiation processes. On the other hand socio-economic mechanisms, that proved their efficiency in real-world scenarios, such as negotiations, markets, distributed learning etc., represent a promising concept for handling massively distributed and cross-organisational settings as just described. In this paper we present a Simulator for the Internet of Services (SimIS), a new agent-based model and simulation toolkit for future IoS settings. Building on scalable agent-based simulation models SimIS aims at providing a generic tool for the investigation of the abovementioned socio-economic mechanisms to be applied

in the IoS. In doing so, SimIS incorporates the given technical conditions (technical standards as mentioned above), different possible IoS connection topologies as well as rich possibilities of implementing the services and mechanisms under investigation. In the next section a thorough requirements analysis will be presented. Together with the subsequent presentation of related work in this area, a structured problem definition and motivation for the development of SimIS will be given. In section 3 the actual simulation model and toolkit will be presented. Finally section 4 will present our work done through verificating the SimIS model by employing it in two of our own research efforts, the investigation of electronic negotiations and reputation-based interaction mechanisms.

Problem Definition and Motivation

2.1   The IoS vision represents a large and very complex application domain for the distributed management of digital resources. New scalable and efficient mechanisms must be developed for this vision to come true. Due to the inherent complexity of IoS settings these new mechanisms cannot easily be incorporated into running service systems to be tested. Especially their effects in large scale scenarios that are especially interesting, these being the most challenging settings for the applied algorithms, should be simulated before implemented in real systems. As a simulation method, multi-agent simulations seem to fit the methodological requirements. Therefore, a multi-agent simulation toolkit is needed that allows modelling IoS settings for the sole purpose of testing and evaluating distributed mechanisms in a controlled environment. Employing such a toolkit the effects of the developed mechanisms can be evaluated with minimal risk of system crashes in a running IoS. In the following, we will present a set of requirements posed on such a simulation toolkit to model the IoS reality correctly and thus provide the user of SimIS with valid simulation results. Subsequently related research efforts in this area will be sketched, elaborating the need for a simulation model and toolkit especially tailored to the IoS vision.

Requirements

2.2   A simulation toolkit to be used in the investigation of socio-economic mechanisms for the IoS has to meet certain requirements:

1. Capability of modelling socio-economic mechanisms for distributed coordination and management: As we envision such mechanisms as crucial in the future IoS, the developed simulation toolkit must support their implementation. The toolkit has to provide flexibility to the user in combining different socio-economic mechanisms, like negotiation or trust and reputation mechanisms. Further it has to provide and API to extend these mechanisms or event to test completely new paradigms.
2. Distinction between active nodes and passive arcs: In order to represent valid IoS settings the simulation toolkit must be able to represent services and platforms (nodes) as well as logical connections between those (arcs).
3. Structural integrity of node interfaces with real-world Web Service (WS) interfaces: The implemented services must adhere to the structural constraints of real world WS. This way the simulated services can seamlessly be mapped onto actual WS, as would be present in the IoS.
4. Structural integrity of node communication with real-world SOAP message conversations: Accordingly the simulated conversations must employ data objects which correspond to real-world SOAP messages as applied in WS communications.
5. Possibility of simple scenario generation for each simulation run (in terms of nodes and arcs): In order to simulate different IoS settings an easy way of defining scenarios, in terms of nodes and arcs, must be offered by the simulation toolkit.
6. Scalability and stability of the underlying platform: The simulation toolkit must scale with a potentially high number of simulated services and connections.
7. Easy-to-use graphical and file output gathering data from simulation runs: For the thorough examination of simulation results the developed toolkit must offer various possibilities to display relevant data during simulation and to store it in respective data files.
8. Access to probability functions and scheduled events for dynamic behaviour during simulation: In order to allow the simulation of real-world system dynamics, probability-based as well as scheduled events must be possible within such a toolkit.

Related Work

2.3   Related research efforts can be structured in two levels of abstraction: On the one hand several researchers

propose agents and intelligent services in conjunction with socio-economic mechanisms as crucial for the future IoS, thus approving the need for a respective simulation toolkit. On the other hand a variety of simulation tools have been developed, however not covering all requirements as stated above; thus the development of SimIS in particular is motivated. Leading research groups and international research projects envision the future IoS to become a system of intelligent services that communicate and negotiate about computational and data resources. Buyya et al. (2008) or Barros et al. (2005) for example, argue in that direction. The intuitive choice for implementing such intelligent services is software agents, which in turn act within socio-economic mechanisms such as negotiations or reputation concepts (see for example (Braun 2006; Chhetri 2006; Neumann 2008). Based on this trend the need for an agent-based simulation toolkit for the IoS is clearly visible. Then again a set of simulation toolkits for distributed systems have been designed. One of the most promising ones, regarding the abovementioned requirements, is the OptorSim toolkit (Bell 2003). However this system is not further developed as of 2006 and therefore lacks user support and adoption to future network settings. GridSim (Buyya 2002; Sulistio 2007) presents a quite comprehensive simulation framework; however it focuses only on Grid Computing applications, thus stressing technical details such as scheduling or generation of virtual organizations and advance reservation of resources. Another approach already building on agent-based technology, and thus much more relating to the vision we have, is the AgentGrid Project. This simulation tool however also is no longer supported and further development is stopped.

2.4 As we are going to use socio-economical mechanisms to coordinate the IoS, it might also be sufficient to use pure agent based simulation toolkits that are widely used in social simulation. As the field of simulation environments is quite vast here, we just refer to some surveys of simulation tools, like Gilbert et al. (2002) or Railsback et al. (2006).

2.5 Due to the sophisticated network usage and the time-discrete nature of the simulation of Repast, we decided to use this environment for agent-based simulation and add some functionality to be able to model Internet-like network structures and to test socio-economical mechanisms to coordinate such networks. Our approach equals the one proposed by Dibble et al. (2004). We will present SimIS, which represents the result from these efforts, during the next subsections in brief.

SimIS Toolkit

3.1 In this section the SimIS toolkit will be presented in more detail. A theoretical basis for this framework is the structural resemblance between electronic services, as present in the IoS, and software agents, being the fundamental concept of most simulation toolkits. Software agents represent autonomous pieces of software (basically of varying granularity) that act and react within a given environment. These software components try to reach individual or collective goals by interacting with each other and/or their environment. The abstraction level represented by such autonomous agents can be employed to implement largely distributed software systems (for example supporting logistics) or to investigate decentralised algorithms for managing such distributed systems. However, on a technical level agents serve similar purposes services do: each concept stands for a software component, capable of receiving (invocation) messages, execute the triggered tasks and potentially send further (answer) messages to other components. Due to this analogy we deem software agents an appropriate abstraction for modelling the intelligent services of the IoS and therefore propose an agent-based architecture for the SimIS toolkit. In the next subsections we will shortly sketch the technological base for SimIS and afterwards present how it was extended within the SimIS design process.

Technological Base: Repast Toolkit

3.2 The SimIS toolkit was implemented as an extension to the Recursive Porous Agent Simulation Toolkit (Repast 2009), developed at the Argonne National Lab, Chicago. Repast is a free and open source agent-based modelling toolkit (North 2006). This foundation was chosen due to its comprehensive API, the very generic and easy to use set of data gathering and analysis functions as well as the support for network modelling (including respective programming libraries). Hence, Repast corresponds directly with the envisioned IoS structure. Technically, the current version of SimIS is based on Repast Symphony that represents the current version, completely implemented in the Java programming language.

SimIS Architecture

3.3 In order to map the abstract IoS architecture to our simulation model we propose a two-tiered architecture for SimIS. The overall system is thus divided into an Application Layer and an Infrastructure Layer. An overview of

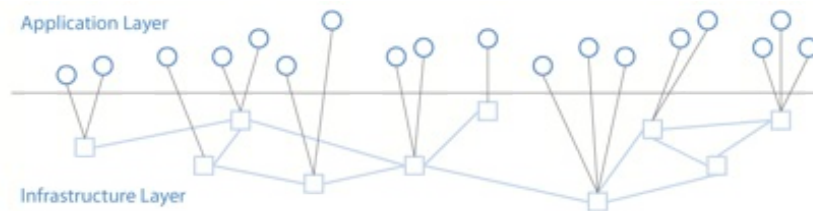the overall architecture is illustrated in Figure 1.

Figure 1. The SimIS Agent Architecture

3.4   The Infrastructure Layer models topological settings of the IoS. The basic idea is that all Application Layer Agents/Services are linked to a single Infrastructure Agent each, which is representing their server platform. This platform is therefore responsible for a) sending messages to other Application Layer Agents (including routing and communication patterns, such as broad- or multicast) and b) receiving messages from other Infrastructure Agents and passing them on to either other Infrastructure Agents (in case the agent represents only the next step on the message's route) or to one or more Application Layer Agents associated with it (in case these are the recipients).

3.5   The functionality offered by Infrastructure Agents is divided into two distinct interfaces, the ServiceAccessPointInterface describing all methods offered to the Application Layer and the Infrastructure Interface containing all functionality needed for message passing among Infrastructure Agents. As the Infrastructure Agents' only purpose is the delivery of messages, the ServiceAccessPointInterface thus offers methods for uni-, multi- and broadcasting message objects. While unicast messages are delivered to only one destination, a broadcast message does not have a distinct recipient. For such messages only the maximum hop number is required, restricting how far the message object is forwarded into the network. In the next step the Infrastructure Agents calculate the shortest route through the network to the message's recipient and send the message over this route respectively. For this purpose two basic constructs of the Repast API were used: JungNetWorks, for representing the network topology and the Dijkstra algorithm (Dijkstra 1995) for routing in these networks.

3.6   Analogously the InfrastructureInterface comprises the functionality for relaying messages from one Infrastructure Agent to another, thus forming the basis for message driven communications as present in the IoS. Message Formats In this section a short overview on the structure of SimIS message objects is given. In order to stay consistent with real-world SOAP messages, the message objects in SimIS exhibit their content data according to XML conventions, as SOAP messages are basically XML documents restricted to a given structure. SOAP messages must have two high-level elements, the header and the body element both of which can have attributes and either child element(s) recursively or typed content. For this purpose we implemented a set of data classes (e. g. for a String or Integer content) respectively used for typed attributes or element contents. Employing the aforementioned concepts we could design a Message class fulfilling the constraints posed by the SOAP standard (requirement 5). By strictly following this standard for our message structures we can assure that all mechanisms designed and tested within SimIS, building on the aforementioned message classes, can easily be ported to a real-world infrastructure, if based on WS and SOAP technology such as the majority of current distributed Information Systems.

3.7   Within the *Application Layer* the actual services of the IoS vision are modelled. Basically the underlying Infrastructure Layer provides researchers with a high-enough flexibility for implementing any service logic in terms of Application Layer Agents communicating via the offered message objects and routing functionality. Each service, i.e. Application Layer Agent, is to be implemented as a plain Java class and can therefore exploit the full potential this programming language offers in addition to the libraries present within the SimIS toolkit. As we anticipate future IoS settings to adhere to market structures analogous to real-world economies we implemented such a setting within the Application Layer of SimIS. Based on this basic setting further investigations can be made regarding different negotiation protocols, reputation mechanisms etc. In our scenarios so called basic services are sold on a service market. These basic services on the other hand need resources that have to be bought on a resource market, using a common negotiation protocol. For our research efforts we take the abovementioned two-tiered market structure as a given. Figure 2 depicts this abstract scenario. Although we implemented this basic model as a foundation for all our following simulations, even this market structure can be easily extended or altered.
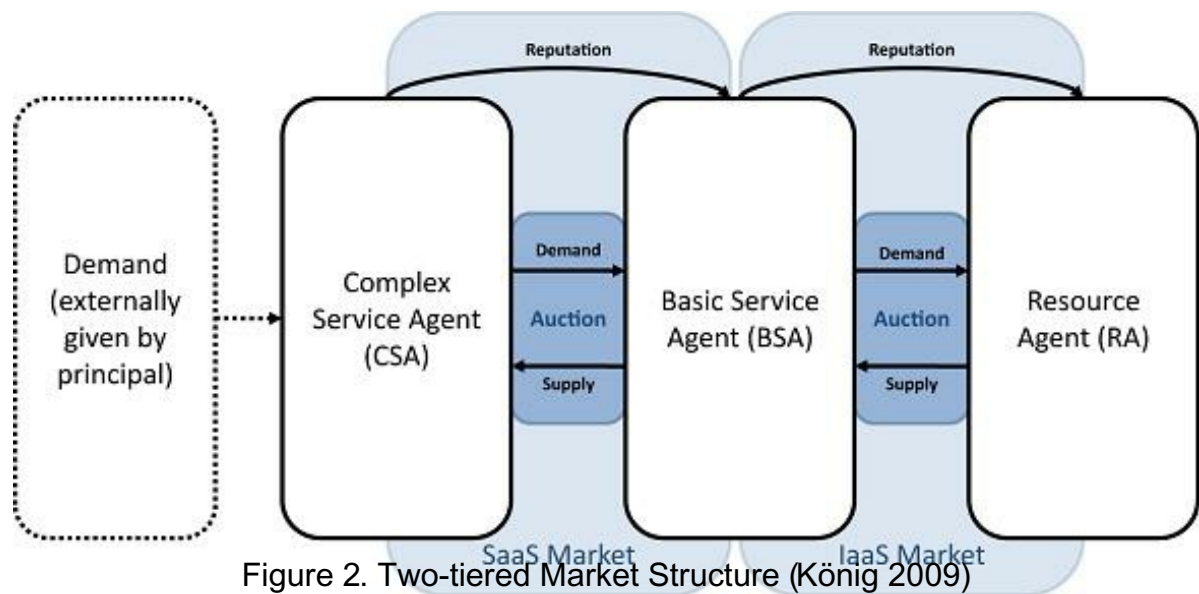
Figure 2. Two-tiered Market Structure (König 2009)

3.8   Using a two-tiered market to model an Internet of Service is quite useful, because the corresponding business models are twofold and can be summarized with the two terms Software as a Service (SaaS) and Infrastructure as a Service (IaaS). SaaS describes the paradigm that providers offer their software products in an Internet environment that can be accessed at any time from any computer by the buyer of the service. The service sold thereby is an end-user-application that is restricted to what the application is and what it can do. Hence, the buyers neither know nor control details of the underlying technology but only use the service as such. IaaS on the other hand is the hardware counterpart to SaaS. Thus in the IaaS business model case customers do not pay for services, but pay to use a shared infrastructure. In our view, SaaS and IaaS can build on top of each other, resulting in a scenario in which providers can play more than one role. These business models, called Cloud Computing, are currently hot topics in information systems science and can also be subsumed under the IoS paradigm.

Verificating SimIS - A Use Case driven Approach

4.1   This section will provide two different use cases, which illustrate the application and some preliminary simulation outcomes when using SimIS. The first scenario uses negotiation as a social mechanism to allocate resources and services on the two-tiered market. The latter one uses reputation in addition to simple negotiations in order to achieve equilibrium of well-behaved agents in the IoS.

Use Case 1: Negotiating Agents in the Internet of Services

4.2   In detail the scenario for trading on the IoS markets works as follows: If a Complex Service Agent (CSA) is given a demand for a service externally by its principal, it will try to satisfy this demand by breaking the complex service down into several basic services and buy the basic services required on the SaaS market from the Basic Service Agents (BSA). In order to buy basic services, the CSA is being given an initial budget that it can spend. After a deal on the SaaS market is closed, the CSA will pay the BSA the money for the services. In order to be able to "produce" basic services, the BSA needs resources which it can buy on the IaaS market from the Resource Agents (RA). It thereby can invest the money it got from the CSA for the transaction on the SaaS market (and optionally other budget it may have saved from earlier successful transactions). Again, if a deal is closed (on the IaaS market) the BSA pays the RA in advance. In case the RA delivers the promised resources, the BSA can and will deliver the basic services to the CSA.

4.3   Within the simulations we will concentrate on the English Auction protocol as negotiation mechanism, both on the IaaS and the SaaS market, even if our simulation toolkit provides several alternatives, like a Dutch auction protocol, several Reverse Auction protocols or even the FIPA Alternate Offers protocol. The assumed market structures and the fact that services and resources are sold and not requested determine, in our point of view, the English Auction protocol. Each selling agent (that is BSA on the SaaS market and RA on the IaaS market) sells its own service that means the agent fulfils also the role of an auctioneer. Using SimIS' time discrete simulation environment, at each time tick exactly one agent is able to decide whether to start an auction or not. As a consequence, the buyers need to decide, whether they buy from the agents offering services/resources at a specific point of time, or whether they wait some more time until the next seller offers its product. However, the

more time passes by, the higher the risk of not getting any of the needed services/resources. Hence, first the seller (and auctioneer) proposes an auction and all agents, which are interested, register for participating. The call for bids messages are then sent to all participants, which can place their bids subsequently. During the auction the auctioneer increases the price in each simulation tick. The increasing price results in an out dropping of bidders of the auction. In the end, the last remaining buyer wins the auction and has to pay the second-highest bid.

4.4 After the auction has finished, the seller sends a message with the winning bid and the winner to all participating agents. They can use the information for comparison with their own bidding, and thus learn towards a better strategy for the next auctions. In this simulation the learning strategy is simplified, such that agents are just increasing their reservation price when loosing an auction and decreasing it otherwise. The reservation price represents the maximum price the agent is going to bid.
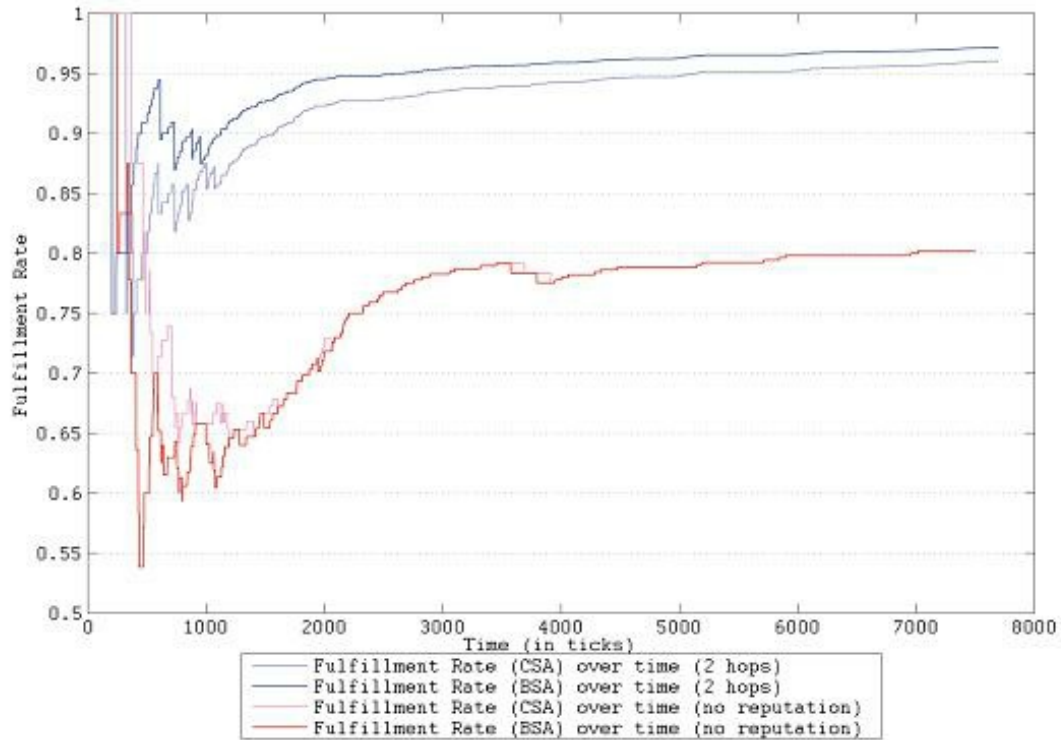

Figure 3. Simulation with and without the Usage of Reputation (König 2009)

4.5 To validate the use case, we use a simulation environment based on our SimIS toolkit. The underlying network consists of 100 nodes, connected in an Internet-like way without any clusters or heavy-tailed elements. Further, 160 CSA, 200 BSA and 40 RA have been initialized in the system. Additionally a 20 percent failure rate concerning the RAs was set. In Figure 3, the red lines note the fulfilment rate of interactions in a typically IoS setting. With the current setting we expect a fulfilment rate of about 80 percent. This is exactly what the graph denoted in the figure shows after the initial settling phase.

Use Case 2: Electronic Reputation to Overcome Strategic Uncertainty

4.6 In the second use case the involved agents try to tackle this uncertainty, concerning a successful service delivery, by employing a distributed digital reputation mechanisms. This uncertainty can either originate in technical conditions but also in strategic behaviour of the selling agents, resulting in moral hazard (Eymann 2008). Such situations can be present any time two parties enter an agreement with one another. Each party in a contract may have the opportunity to gain from acting contrary to the principles laid out by the agreement. For example on eBay, the buyer typically sends the money to the seller before receiving the goods. The seller then is tempted to keep the money and not ship the goods, or to ship goods that are inferior to those advertised. The buyer thus has to take the risk of being cheated, because if it does not, no deal will take place. The seller's payoffs for the single transaction will be higher than in the cooperation case if it does not have to fear any future financial penalties as no record of his misbehaving is being kept. A possible solution to this problem is the usage

of trust and/or reputation, as with the help of reputation mechanisms the independent transactions can be linked.

4.7 To start, we will define the term reputation as we understand it and relate it to the term image that will be of importance in the further course of this use case: Image is a global or averaged evaluation of a given target on the part of an individual. It consists of a set of evaluative beliefs (Miceli 2000) about the characteristics of a target. These evaluative beliefs concern the ability or possibility for the target to fulfil one or more of the evaluator's goals, e.g. to behave responsibly in an economic transaction. An image, basically, tells whether the target is "good" or "bad", or "not so bad" etc. with respect to a norm, a standard, a skill etc. In contrast reputation is the process and the effect of transmission of a target image. The evaluation circulating as social reputation may concern a subset of the target's characteristics, e.g. its willingness to comply with socially accepted norms and customs. More precisely, we define reputation to consist of three distinct but interrelated objects: (1) a cognitive representation, or more precisely a believed evaluation (any number of agent in the group may have this belief as their own); (2) a population-level dynamic, i.e., a propagating believed evaluation; and (3) an objective emergent property at the agent level, i.e., what the agent is believed to be as a result of the circulation of the evaluation (Conte 2002). Putting it simple, an image is the picture an individual has gained about someone else (the target) based on his own previous interaction with that target. If using reputation, the individual expands the information source about the target beyond its own scope and includes the information of others about the target as well by asking other agents regarding their image on the target agent. But how do image and reputation affect uncertainty and trust in the IoS? In order to arrive at hypotheses to answer this questions we will brief look at the general effects of reputation on transactions discussed in literature and abstract from these in a second step.

4.8 Reputation can fulfil several functions: First of all it works as a signalling device to distinguish between trustworthy and untrustworthy transaction partners. Furthermore, it changes the long term utility functions of the markets participant (by introducing potential losses in profit if being identified as cheater) and thereby encourages the transaction partners to cooperate. Due to the closeness of the IoS to the scenarios used in the papers mentioned above, we expect similar effects in IoS markets (like Eymann et al. 2008) and arrive at the following hypothesis: Reputation reduces the uncertainty in the IoS by conveying cooperation. To be able to simulate a scenario to test this hypothesis, we substantiate this hypothesis to "reputation will decrease the number of frauds and thus increase the overall fulfilment rate". During simulation (in each time-discrete tick) every agent logs its accumulated successful transactions and its failed transactions. This metric denotes the mean value over all agents, which had at least one interaction. A transaction is counted as successful if and only if a fulfilment message arrives within the timeout at the buyer agent. If the timeout expires the agent will record a failed interaction. Note: For mathematical reasons the ratio is more volatile at the beginning of the simulation. To test our hypotheses that reputation reduces the uncertainty in the IoS by conveying cooperation we compare two typical simulation runs. During the first simulation run the trust and reputation model from Abduhl-Rahman and Hailes (1997) is used to spread agents' images. In figure 3 the blue lines represent the fulfilment rate over time with shared image usage. The agents are able to ask for opinions for a given target. That means all agents on the same node and all agents on neighbour nodes are asked for their opinion about a target agent.

4.9 Comparing the two simulation outcomes of use case 1 and case 2, we can see that in the first case the overall fulfilment rate settles at about 80 percent, as anticipated with 20 percent failure rate. On the other hand in use case two this metric can be improved steadily to significantly higher values (around 97 percent after 7000 simulation ticks) when using reputation information (König 2009).

4.10 Last but not least, these two use cases illustrate the high potential that socio-economic mechanisms have in the Internet of Services to overcome allocation problems and strategic behaviour.

SimIS: Current Status and Future Steps

5.1 In this paper we presented SimIS, an agent-based simulation model and toolkit for IoS settings. SimIS has been published under GPL license and is available under http://simis.sourceforge.net. Employing SimIS, especially socio-economical mechanisms, as deemed promising for such settings in the future, can be deployed, tested and, if necessary, adjusted within a simulated IoS environment. On a technical level future steps will include the incorporation of a sophisticated fault model and bandwidth parameters. Concerning the socio-economic mechanisms, we could only present two single settings. For both use cases we made use of an English Auction protocol and an optional reputation mechanism. Beside different auction protocols, SimIS provides the possibility to switch to Reverse Auctions or to bargaining protocols without changing the agents' implementation and

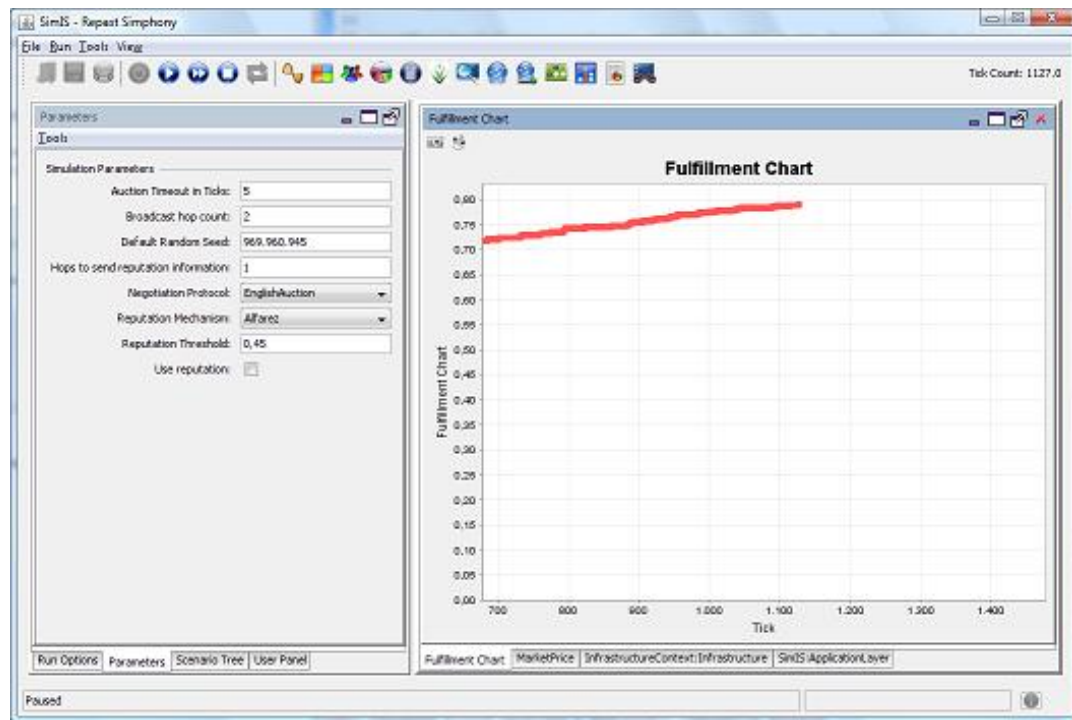without any programming experience by the user (see figure 4) through choosing a protocol in a dropdown list.

Figure 4. Simulating with SimIS

5.2   Besides analyzing possibilities after the simulation runs have succeeded, RePast and thus SimIS also provide nice displays to follow the simulation during runtime. One example is also shown in figure 4.

5.3   Future work will be the addition of market protocols, like Continuous Double Auction protocols, on the one side, and the provision of a more sophisticated service discovery on the other side. In the medium-term future we aim to provide alternative socio-economical mechanisms like insurances.

🌐 References

ABDUL-RAHMAN, A. and Hailes, S (1997) Using recommendations for managing trust in distributed systems. In IEEE Malaysia International Conference on Communication, 1997, 48-60.

BRAUN, P. And Brzostowski, J. and Kersten, G. and Kim, J.B. and Kowalczyk, R. and Strecker, S. And Vahidov, R. (2006) E-negotiation systems and software agents: Methods, models, and applications. In Gupta, J. and Forgionne, G. and Mora, M. (eds.), *Intelligent Decision-Making Support Systems: Foundation, Applications and Challenges*, Decision Engineering Series. Springer, Heidelberg, 271-300. [doi:10.1007/1-84628-231-4_15]

BELL, W.H. and Cameron, D.G. and Millar, A.P. and Capozza, L. and Stockinger, K. and Zini, F. (2003) Optorsim: A grid simulator for studying dynamic data replication strategies. *International Journal of High Performance Computing Applications*, 403-416. [doi:10.1177/10943420030174005]

BARROS, A. and Dumas, M. and Bruza, P. (2008) The move to web service ecosystems. *BPTrends Newsletter*, 3(3).

BUYYA, R. and Murshed, M. (2002) Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing. *Concurrency and Computation: Practice and Experience* (CCPE), 1175-1220.

BUYYA, R. and Yeo, C.S. and Venugopa, S. (2008) Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities. In *Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications* (HPCC-08), Los Alamitos, CA, USA. [doi:10.1109/hpcc.2008.172]

CHHETRI, M.B. and Lin, J. and Goh, S. and Zhang, J.Y. and Kowalczyk, R. And Yan, J. (2006) A coordinated architecture for the agent-based service level agreement negotiation of web service composition. In *Proceedings*

of the Australian Software Engineering Conference (ASWEC'06), pages 90-99, Washington, DC, USA. IEEE Computer Society. [doi:10.1109/ASWEC.2006.1]

CONTE, R. and Paolucci, M. (2002) *Reputation in Artificial Societies: Social Beliefs for Social Order*. Kluwer Academic Publishers: Dordrecht. [doi:10.1007/978-1-4615-1159-5]

DIBBLE, C. and Feldman P.G. (2004) The GeoGraph 3D Computational Laboratory: Network and Terrain Landscapes for RePast. *Journal of Artificial Societies and Social Simulation* 7 (1) 7 http://jasss.soc.surrey.ac.uk/7/1/7.html

DIJKSTRA, E.W. (1995) A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1), 269-271. [doi:10.1007/BF01386390]

EYMANN, T. and König, S. and Matros, R. (2008) A Framework for Trust and Reputation in Grid Environments. *Journal of Grid Computing*; Special Issue on Grid Economics, 6(3) 225-237. [doi:10.1007/s10723-008-9101-4]

GILBERT, N., and Bankes, S. (2002) Platforms and methods for agentbased modeling. *Proceedings of the National Academy of Sciences*, vol. 99, suppl. 3, 7197-7198. [doi:10.1073/pnas.072079499]

GUDGIN, M. and Hadley, M. and Mendelsohn, N. and Moreau, J. and Nielsen, H.F. (2003) Soap version 1.2 part 1: Messaging framework. W3C. http://www.w3.org/TR/2007/REC-soap12-part1-20070427/

KÖNIG, S. and Balke, T. and Quattrociocchi, W. and Paolucci, M. and Eymann, T. (2009) On the effects of reputation in the internet of services. In *Proceedings of the International Conference on Reputation (ICORE)*, 200-214.

MICELI, M. and Castelfranchi, C. (2000) The role of evaluation in cognition and social interaction. In K. Dautenhahn, editor, *Human cognition and social agent technology*. Benjamins, Amsterdam. [doi:10.1075/aicr.19.12mic]

NORTH, M.J. and Collier, N.T. and Vos, J.R. (2006) Experiences creating three implementations of the repast agent modeling toolkit. *ACM Trans. Model. Comput. Simul*., 16(1), 1-25. [doi:10.1145/1122012.1122013]

NEUMANN, D. and Stoesser, J. and Weinhardt, C. and Nimis, J. (2008) A framework for commercial grids - economic and technical challenges. *Journal of Grid Computing*, 6(3), 325-347. [doi:10.1007/s10723-008-9105-0]

OCKENFELS, A. (2003) Reputationsmechanismen auf Internet-Marktplattformen: Theorie und Empirie. *Zeitschrift für Betriebswirtschaft,* 73(3), 295-315.

RAILSBACK, S. and Lytinen, S.L. and Jackson, S.K. (2006) Agent-based Simulation Platforms: Review and Development Recommendations. *Simulation*, vol. 82, 609-623. [doi:10.1177/0037549706073695]

REPAST Developers Group (2009) Repast Home Page. Web page. Available at http://repast.sourceforge.net

SULISTIO, A. and Poduval, G. and Buyya, R. and Tham, C. (2007) On Incorporating differentiated Levels of Network Service into Gridsim. *Future Generation Computer Systems*, 23, 606-615. [doi:10.1016/j.future.2006.10.006]