ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL, 2006, Vol.35, No.4

PERFORMANCE MODELING OF INTEROPERABILITY SYSTEM FOR SME'S

Claudia Guglielmina

TXT e-solutions, Via Frigia, 27, 20126 Milano, Italy

Audrone Janavičiute, Mindaugas Kiauleikis, Valentinas Kiauleikis

Kaunas University of Technology, Computer Engineering Department
Studentų Str. 50, LT-58631 Kaunas, Lithuania

Nerijus Morkevičius

Kaunas University of Technology, Department of Applied Mathematics
Studentu Str. 50, LT-58631 Kaunas, Lithuania

Abstract. Three stages of resource integration can be distinguished in evolution of IT: the stage of creation and implementation of middleware technologies and means (1990-2000), the stage of modeling and creation of enterprise application integration solutions (1995-2005) and the stage of enterprise service bus (ESB) development and technologies creation (2004-…). ESB concept was born after the need for business integration emerged, especially under the influence of SME's needs to have sufficient system functionality for affordable price. Problems of creation and performance issues of one of ESB variants – SME application bus infrastructure for interoperability of business subjects are discussed in this paper. Colored Petri Net model for performance evaluation of such infrastructure is presented.

Introduction

Transition from "pure" accounting systems to need of information mining, from business oriented software to IT solutions offers, from development of local systems to integration of distributed resources in service oriented architectures (SOA) [4, 5] was on the way during last ten years. New combinations of already implemented systems and their components are created in SOA for maximal synergetic effect of new systems with consideration of autonomy of systems under integration.

Middleware technologies were changed by enterprise application integration (EAI) solutions [3] establishing XML standard [1, 6] in this environment. The latter are gradually changed by enterprise service bus concept. These changes are going under active motivation of this process to IT users, mostly businessmen, which are dissatisfied with the situation in IT services market. Small and medium enterprises (SME) are even more affected.

According to Forrester research [2] prices of integration solution licenses start from 100 thousand dollars for one project and average price goes up to 400-750 thousand dollars. Naturally, high price is caused by functionality of solutions needed for enterprises, which can afford this kind of solutions. These solutions are not affordable for SME's. New generation solutions based on ESB integrates formerly developed technologies – message based middleware, internet services, message transformation and routing and claim to become multifunctional but inexpensive technology for usage in solutions with SOA.

Business information interchange infrastructure based on global information transfer network and middleware technologies, supplemented by internet services for business information interchange must be created to implement this technology. The aim of this paper is: 1. to define which additional internet services are needed to create interoperability infrastructure; 2. to create a model of such services and examine computational resource needs for interoperability infrastructure.

1. Interoperability problems

The development of global data transfer network in Lithuania was started in 1990 and now it is practically complete. This network enables business enterprises to connect to Internet and use all its services. Data transfer network lets Lithuanian businessman go out of Lithuania borders freely making and keeping business contacts with foreign

businessmen, but initiative of businessmen is restricted by other barriers. There are four evident barriers. To solve them, technological solutions, which compose basis of business information exchange infrastructure, are sufficient.

1. **Language.** The second language to Lithuanians is Russian. Using this language, businessman can speak and write, create new documents and understand received documents. But countries to the west or north (Latvia, Estonia, Scandinavian countries) use their own languages so business relations are encumbered by translations to other language, explanation of meanings and, of course, misunderstanding. There is no common language in which it is possible to interact (for example, English), and watching tendencies in EU expansion it won't be in future.
2. **Law.** Each country has its own law according which businessmen register and run their activity. Law describes processes and documents of interoperability between businessmen and state institutions. Each must be adjusted when business goes out of country's border.
3. **Forms of documents.** In Lithuania, there are regulated forms of documents that are in strict accountability in state institutions. Documents for inter-organizational accountings have a free form. Only attributes, which are required in some documents and which can differ (for example, because of specifics of business or local requirements), are regulated. While seeking comfort or in other consideration businessmen can use their own created forms with freely positioned required attributes. Any deviation from usual businessman's form is unacceptable, especially when there is no required attribute. This problem emerges not only between local interactions, but also in processes of interoperation with businessmen from other countries.
4. **Business process.** Same as document's forms business process in Lithuanian business enterprises can differ because of order inside enterprise or business specific. So deviations from business process implemented in enterprise are unacceptable no matter interoperation is inside or outside of the country.

Each barrier contributes to preventing e-business expansion not only in international scale but also between Lithuanian businessmen. Information technologies should help eliminate theses barriers but discussion and solutions must come not only from IT companies, which offer expensive products that doesn't solve necessary problems, but also from businessmen.

These four barriers can be minimized only by development of existing interoperability infrastructure (data transfer network and e-services) using means of business information exchange. One of means is EU funded project ABILITIES, which is described in the next section.

2. ABILITIES Project

The EU funded project ABILITIES has been launched with the aim to address SME interoperability technologies and solutions. Interoperability identifies the ability of new generation Enterprise Applications (EA) to simplify, speed-up and rationalize the implementation of EAI projects involving them. The analysis of research about enterprise interoperability led to the identification of the two main research priorities: adaptation of UBL documents [9] to the requirements of SMEs in Enlarged Europe; and federated architecture for interoperability with intelligent, adaptive business documents; reconciliation and integration of state-of-the-art languages and standards for Business Process Management and Service Orchestration with new research results in the field of Business Documents and Messages.

The solution proposed for supporting both interoperability levels is a blended architecture which could join the peculiar advantages of message-based Service Oriented Architectures (efficiency, maintainability, modularity, scalability, portability, security and privacy assurance) in the business documents format context and intelligent systems for mutual understanding and agreement on document contents, in a context where business documents become intelligent, interactive multi-media objects. The overall goal is not to change the business, processes or existing systems, but to adopt them.

The heart of the ABILITIES Architecture is the ABILITIES Interoperability Bus (AIB), based on an open source Enterprise Service Bus1. All core components that implement the interoperability support for the collaborating enterprises are connected to this ESB, as shown in Figure. The architecture integrates and supplies a number of functions that allow independence and autonomy of each network member, while supporting interoperation and fruition of transacted content.

- **Collaboration Configuration** – the communication between SMEs purchase order details and defining specifications of the goods or services is essential. The habits and structure of the company need to be taken into account. A standard collaboration configuration would not be sufficient. With this configuration it can be defined in what cases a collaboration session between two companies should be started and what channel should be used, to handle open issues, in case of not convergence of transactions.
- **Negotiation Rules Handling** – each company can define common rules it follows and even

[1] Apache ServiceMix – Open Source ESB, http://servicemix.org/site/home.html

special rules for certain partner companies. These rules will be used to avoid or at least to shorten bargin by given rules from each company. It is important that the system serves as an independent trust worth mediator following the given rules.

- **Process Design** – processes of companies differ depending on size, domain, location, etc. For the test cases we explore each test case will have a defined Global Process designed, capturing the peculiarities of the location and domain. However, there will be differences according to the processes of different companies. Each company therefore needs to define Private Processes as sub blocks of the global one. This gives flexibility and provides a guideline for companies that run an intuitive business and did not yet think about their internal business processes.
- **Roles Management** – the employees of the company need to be adjusted to roles and/or tasks defined during the process design, to enable the system to transmit it via the appropriate channel to the correct person.

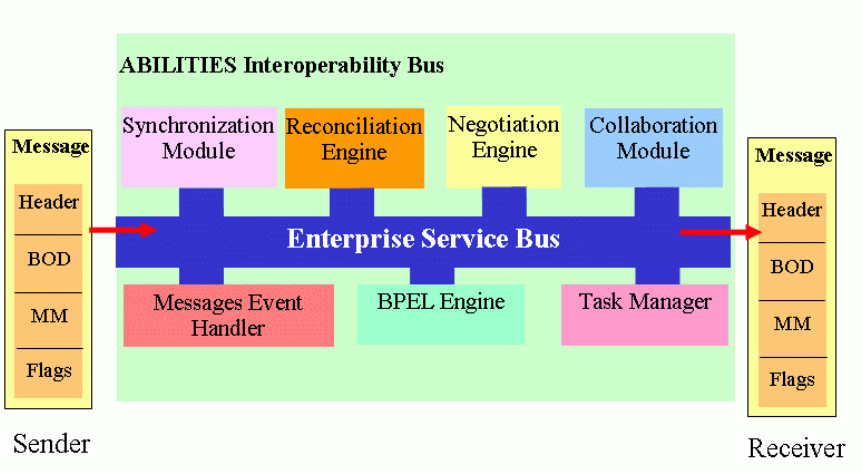

Figure 1. Internal view of the ABILITIES Interoperability Bus (AIB) showing the core components

- **Business Documents Definition and Adjustment** – each SME relies on its specificities, adherence to local or national regulations and norms, as well as internal norms. Starting from a common UBL base of Business Documents, ABILITIES supports derivations and adjustment of the standard UBL messages – called UBLtc. This approach ensures domain or regional or custom localizations while at the same time ensuring interoperation at the level of business documents. In case of existing legacy systems or peculiarities of the company itself further adjustments might be useful.
- **Multimedia Content Management** – in the process of transmitting information and ensure mutual understanding of the transmitted content, a picture can say more than a hundred words. Since the goal is to reduce communication and bargin between companies, multimedia, like pictures, videos or voice records are supported by the system during the trading process. There are various ways to handle multimedia files like attachment, link to a repository, etc.

The core functionality of the system is to exchange the messages between the two trading parties. While passing the message from one party to another the message will be transformed and might be modified based on the defined negotiation rules. The transformation follows the unified approach translating the sender message according to the ontology adjustments performed by the sending and receiving party (reconciliation engine). Additionally, the federated approach is used to negotiate the message business content (price, quantity, warranty duration, delivery date, quality). This is the task of the negotiation engine that in case of existing multi-media content interacts with the collaboration module.

3. Model of Interoperability System

An abstract model of interoperability system for business collaboration is described in this section. This model is based on “real world” ABILITIES system implementation so the main components of the model are similar to AIB.

The AIB consists of two types of modules: design time modules and run time modules. The design time modules are: Collaboration Configuration Manager, Process Designer, Ontology Delta Editor, Negotiation Designer, Multimedia Content Manager, and User Manager. All these modules are used by end user of ABILITIES portal to prepare data and rules for normal ABILITIES system work at runtime. These modules use the following repositories: Negotiation Rules, Reconciliation Rules, Multimedia and Users repositories.

The runtime modules of ABILITIES system receive, handle, convert, process (according to data prepared at design time) and send messages to destination points. The runtime modules include: Message

Event Handler, Synchronization module, Reconciliation Engine, Collaboration module. Some repositories are used at run time too: Negotiation Rules, Reconciliation Rules and Messages repositories.

It is assumed that the performance of interoperability system (IS) at design time is not the option for optimization. On the other hand, the runtime performance is time critical. In the proposed IS model, only runtime modules are presented. Collaboration module (CM) needs end users' interactions so it is assumed that message leaves IS when it reaches CM. The runtime modules use some repositories. In the first version of the model, only two of them are included – Reconciliation rules repository and Negotiation rules repository. Other repositories are used only at design time (User repository) or are accessed directly by the clients (Multimedia repository), so they do not add direct impact on the message processing time. Message flow through such system is shown in Figure 2.

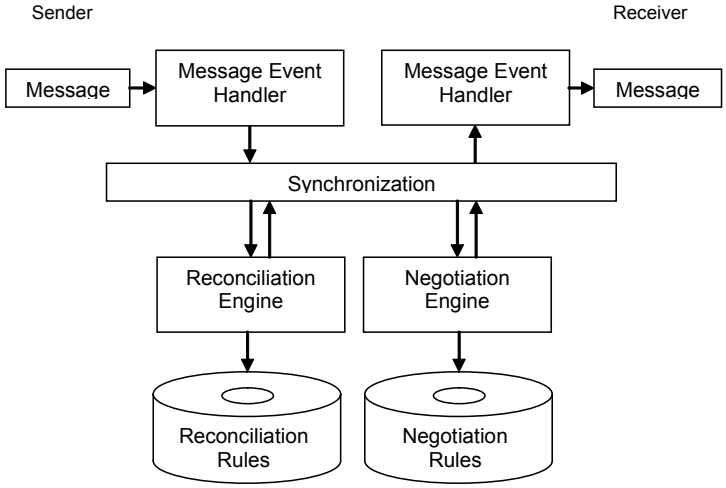

Figure 2. Model and message flow in Interoperability System

A native message enters the IS. Message Event Handler handles the message and delivers it to Synchronization module. Synchronization module decides which message transformations are needed. Then it sends message to the Reconciliation Engine (if needed). The Reconciliation Engine performs any designated operations on the message (for example performs semantic reconciliation, finds and loads semantic reconciliation rules, translates UBL business documents, etc). Synchronization Module then delivers message to the Negotiation Engine (if needed). Negotiation Engine receives the message, applies stored negotiation rules and transforms it. Finally, Synchronization Engine delivers message to sending endpoint. Message Event Handler converts message into the appropriate native format and sends it to destination party.

The target of modeling is to find out the performance of IS system at run time with different load of messages. Petri model presented in the next section simulates the behavior of IS at run time from the moment when message enters the IS until it is processed and is sent to receiver. This model assumes that all repositories are stored on the same system as IS.

4. Petri Net Model of Interoperability System

The basic scheme of IS model created using colored Petri nets [7], [8] is shown in Figure 3. The model consists of four main modules: Message Event Handler (MEH), Synchronization module (SM), Reconciliation engine (RE), Negotiation engine (NE). Each of these modules has its own queue (*MEH_Q*, *SM_Q*, *NE_Q* and *RE_Q,* respectively) in which messages wait until they are processed. Negotiation and Reconciliation engine modules retrieve needed data from two repositories: Reconciliation repository (RR) and Negotiation repository (NE). Both repositories use the same database (*RDBMS*) with queue (*RDBMS_Q*) to store required data.

Each message enters system at *IN* place. The time between two message arrivals has a mean of *IncInt* and exponential distribution. Each message has two parameters (r and n) which may have values 1 or 0. The first one determines the need to process message in Reconciliation engine and the second the need to process message in Negotiation engine. These numbers are generated at the message arrival time. They are random, independent and have probability to be "1" *Rp* and *Np*, respectively. Message Event Handler handles message and sends it to Synchronization Module. According to parameters of message, SM sends it to RE and NE (if needed). At the end, SM sends message to MEH which sends it to receiver (*OUT* place). Delays which are needed to handle messages in modules have Poisson distributions with means *Mehd*, *Smd*, *Red*, *Ned*, *Rrd*, *Nrd*, *Rdbmsd* for MEH, SM, RE, NE, RR, NR, RDBMS respectively. The processed message leaves system at OUT place.

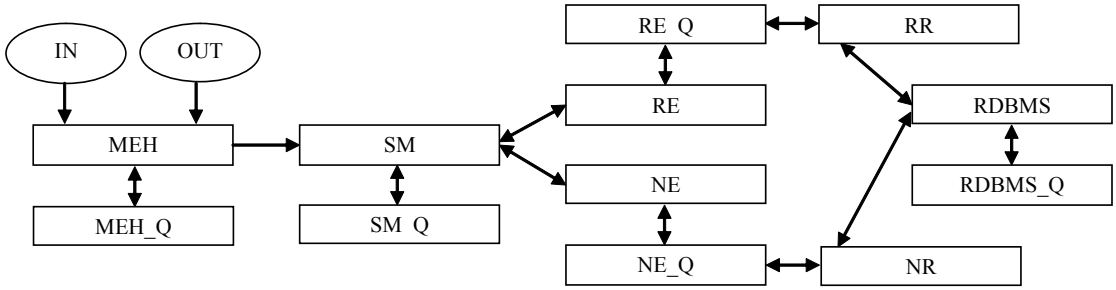

Figure 3. Scheme of Petri net model of interoperability system

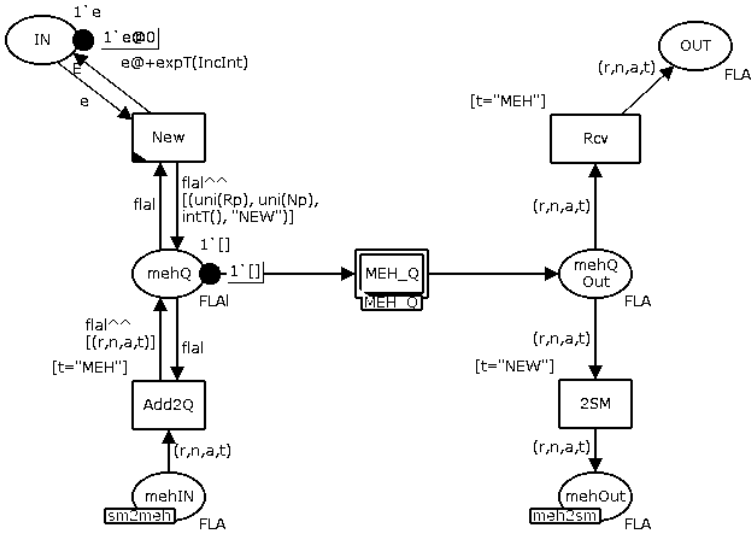

Figure 4. Message event handler page

Petri net is created using CPN tools software for colored Petri net modeling. Petri net consists of several pages for each modeled module. Pages are connected using fusion places and hierarchical structure. Each token in Petri net represents one message (document) in IS system. Message event handler Petri net page is shown in Figure 4. This page also models incoming message flow (*IN* place and *New* transition). Incoming messages are added to MEH queue (in subpage *MEH_Q*) and sent to Synchronization module at fusion place *mehOut*. Messages processed in other parts of Petri net are placed in *mehIn* place, enter the same queue as incoming new messages and leave the system at place *OUT*. Message processing time and other statistics are logged at transition *Rcv*.

Synchronization module part sends messages to other modules (place *smOut*) according to message parameters (*r* and *n*) which are generated at message arrival time (Figure 5). Each message in place *smOut* is handled in Negotiation engine (parameter *t*="NE") and/or Reconciliation engine (parameter *t*="RE"). Messages leave this page at place *smOut2*.

Other parts of IS model are not presented. They are simple Petri nets with few places and some logic. All queues used are FIFO queues with unlimited length and monitoring facilities. Reconciliation and Negotiation engines wait for responses from repositories, so messages usually wait in their queues rather than waiting in database queue.

The created Petri net allows monitoring of maximal and average queue lengths for all modules. Also average, minimal and maximal message processing time may be measured. The model can be used to find out which modules cause biggest delays in whole message flow. According to this information IS architecture can be changed to suit new expectations.

5. Discussion

Region, which aims to implement interoperability infrastructure, could be multilingual, have different legislation, and use different forms of documents and algorithms of workflows when business process is executed. Created model based on ABILITIES project lets evaluate the performance demand of infrastructure components under specific conditions. The conditions should be defined using regional statistics: potential number of business subjects, intensity of message flow, specifics of message flow – time distribution of messages, time parameters of message handling in each of IS modules, needs for message language, semantics, document forms translation and conversion, demand for business process translation and harmonization of resources. The model can show potential abilities of the system, guidelines for further

system's development or architecture perfection when technical characteristics of ABILITIES and regional statistics is known.

For testing purposes initial data shown in Table 1 were used. All times are presented in relative time units (for example milliseconds).

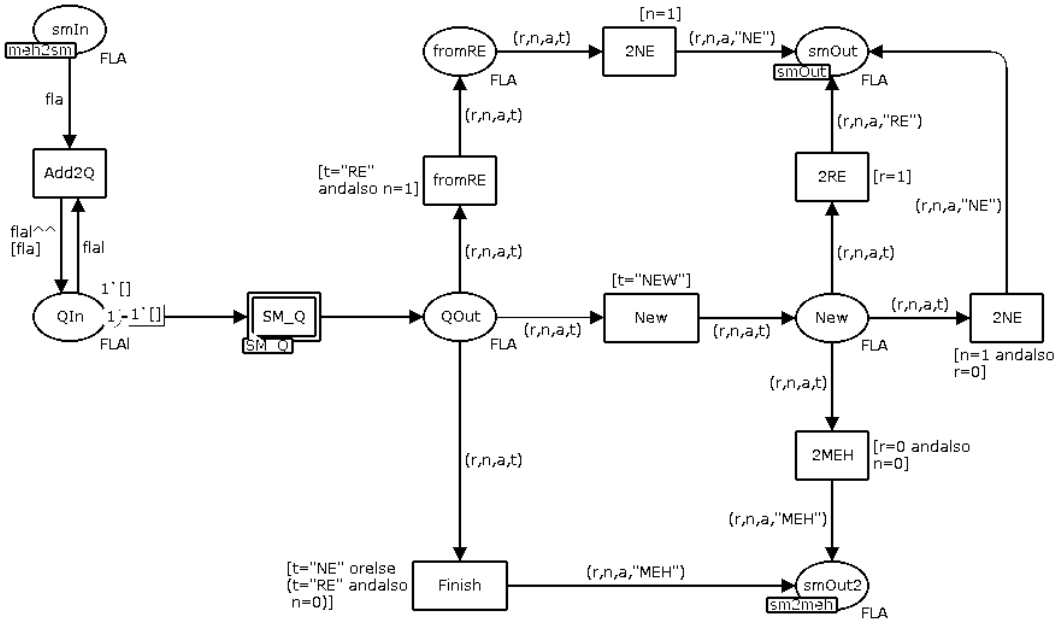

Figure 5. Synchronization module page

Table 1. Parameters for test modeling

Rp	*Np*	*Mehd*	*Smd*	*Red*	*Ned*	*Rrd*	*Nrd*	*Rdbmsd*
0.9	0.3	30	50	200	200	50	50	500

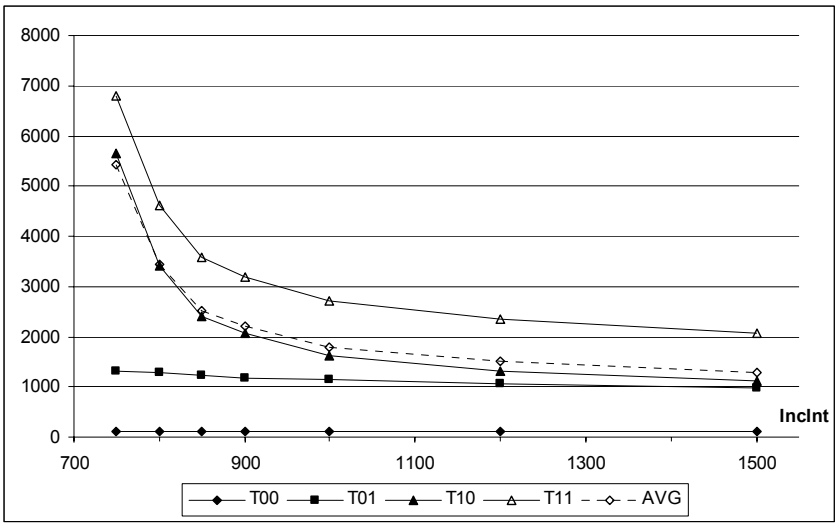

Figure 6. Dependences of message processing time on inter-message arrival time

The dependences of the average message processing time on inter-message arrival time are presented in Figure 6. AVG is an average message processing time in system. T00, T10, T01 and T11 are average processing times of four different message types: T00 represents messages which are not handled at all, they are simply sent to receiver; T01 represents messages which are handled only in Negotiation engine; and so on.

The dependences of maximal count of messages in queues on inter-message arrival time are presented in Figure 7. Here ReMax is the maximal length of the queue in Reconciliation engine, NeMax – the maximal length of the queue in Negotiation engine.

From the first chart (Figure 6), we can see that the maximal affordable inter-message arrival time is about 750 – 800 time units. If this time is smaller, then the average message processing time starts to grow

exponentially and IS becomes unusable. The second chart (Figure 7) shows that the main problem in our IS is performance of Reconciliation Engine. When inter-message arrival time is less than 700, then the maximal length of the queue in Reconciliation engine starts to grow exponentially. If such a situation arises in a real world system, then implementation or even architecture of RE must be revised.

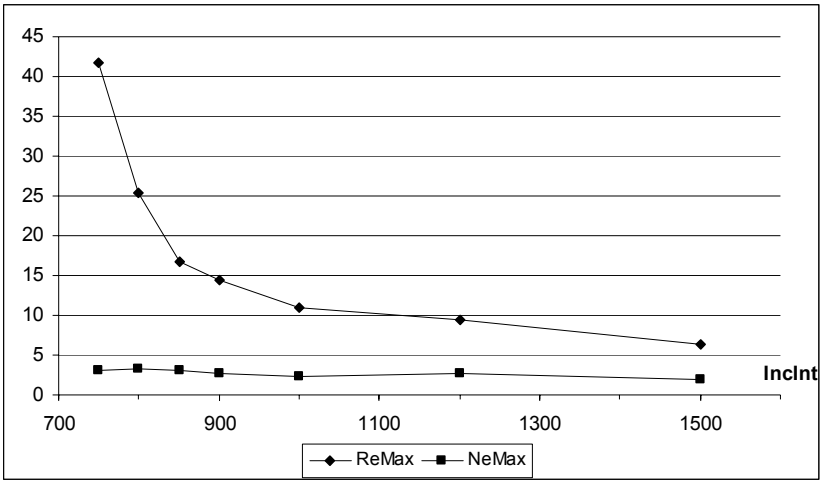

Figure 7. Dependences of maximal count of messages in queues on inter-message arrival time

Summarizing it can be stated that ABILITIES project is perspective for creation of interoperability infrastructure.

1. This infrastructure should solve businessmen interoperability problems by introducing needed functionality and extending web services. ABILITIES technical parameters are unavailable, so testing of the created model can't be finished until all time characteristics of message handling in various modules are available. It is likely that the performance of modern IT shouldn't cause additional problems to minimize message delays inside the IS.

2. ABILITIES provide ability to create interoperability infrastructure which enables everyone to get web services without big additional expenses for purchase and maintenance of software. This is the main ABILITIES advantage to cheapen, but at the same time make more effective IT implementation for SME's.

3. The created model can be used for wide (international) region interoperability infrastructure modeling. It can help to answer questions about architecture of the net, component arrangement and throughput of net elements.

References

[1] **P. Aiken, D. Allen.** XML for Data Management. *Morgan Kaufman Publishers*, 2004, 398.

[2] **D.A. Chappell.** Enterprise Service Bus. *O'Reilly Media*, 2004, 247.

[3] **N.Erasala, D.C. Yen, T.M. Rajkumar.** Enterprise Application Integration in the electronic commerce world. *Computer Standards & Interfaces, Vol*.25(2), 2003, 69 - 82.

[4] **R. Jardim-Goncalvesa, A. Grilob, A. Steiger-Garcaoa.** Challenging the interoperability between computers in industry with MDA and SOA. *Computers in Industry, Vol*. 57(8-9), 2006, 679 - 689.

[5] **P. Patrick.** Impact of SOA on enterprise information architectures. *International Conference on Management of Data archive. Proceedings of the* 2005 *ACM SIGMOD international conference on Management of data*, 2005, 844 - 848.

[6] **Yu-Hui Tao, Tzung-Pei Hongand, Sheng-I Sun.** An XML implementation process model for enterprise applications. *Computers in Industry. Vol*.5 (2) 2004, 181 - 196.

[7] **K. Jensen.** An Introduction to the Practical Use of Coloured Petri Nets. *Reisig, W.; Rozenberg, G.: Lecture Notes in Computer Science, Vol*.1492: *Lectures on Petri Nets II: Applications. Springer-Verlag*, 1998, 237 - 292

[8] **C. Girault, R. Valk.** Petri Nets for Systems Engineering. *Springer-Verlag, Berlin*, 2003, 608.

[9] **M. Gertner, E. Gutentag, A. Gregory**. Guidelines For The Customization of UBL v1.0 Schemas. *OASIS Working Draft, April* 2004.

Received August 2006.