ITHEA

International Journal INFORMATION MODELS & ANALYSES

2017 Volume 6 Number 1

International Journal
INFORMATION MODELS AND ANALYSES
Volume 6 / 2017, Number 1

IJ IMA is official publisher of the scientific papers of the members of the ITHEA® International Scientific Society

IJ IMA rules for preparing the manuscripts are compulsory.
The **rules for the papers** for ITHEA International Journals are given on www.ithea.org .
The camera-ready copy of the paper should be received by ITHEA® Submission system http://ij.ithea.org .
Responsibility for papers published in IJ IMA belongs to authors.

International Journal "INFORMATION MODELS AND ANALYSES" Volume 6, Number 1, 2017

Edited by the **Institute of Information Theories and Applications FOI ITHEA**, Bulgaria, in collaboration with
Institute of Mathematics and Informatics, BAS, Bulgaria,
V.M.Glushkov Institute of Cybernetics of NAS, Ukraine,
Universidad Politechnika de Madrid, Spain,
Hasselt University, Belgium
Institute of Informatics Problems of the RAS, Russia,
St. Petersburg Institute of Informatics, RAS, Russia
Institute for Informatics and Automation Problems, NAS of the Republic of Armenia,

Publisher: **ITHEA®** Sofia, 1000, P.O.B. 775, Bulgaria. www.ithea.org , e-mail: office@ithea.org
Technical editor: **Ina Markova**

Printed in Bulgaria

ISSN 1314-6416 (printed) **ISSN 1314-6432 (Online)**

CLOUD SIMULATORS – AN EVALUATION STUDY

Khaled M. Khalil, M. Abdel-Aziz, Taymour T. Nazmy, Abdel-Badeeh M. Salem

Abstract: *Cloud Computing support better computation through improved resources utilization and reduced infrastructure costs. Number of big providers like Google, Oracle, and Amazon sell services of Cloud Computing which are managed by Cloud Management Systems. It is very difficult to setup research on live cloud environments for individuals and small institutions due to the costs involved in setting up a cloud. Thus, Cloud Simulators are a cost effective way to study cloud components behavior and performance against different situations and work load. Several Cloud Simulators were developed over the years and this paper presents an evaluation study of most of them. We discuss the common architecture of Cloud Simulators. Then we evaluate 33 Cloud Simulators based on different criteria. The results are discussed. We are showing different capabilities, suitability for problems, and extendibility of the studied Cloud Simulators.*

Keywords: *Cloud Computing, Cloud Simulator, Simulation Tools, Modeling Framework.*

ITHEA Keywords: *I.6 SIMULATION AND MODELING, I.6.8 Types of Simulation*

Introduction

Cloud Computing is a trend all over the world and people today are shifting towards it from traditional computing. This is due to the promise of cost reduction, greater flexibility, elasticity, scalability, on-demand access, resource utilization, minimal infrastructure management, and location independence [Lokesh et al., 2015]. Cloud Computing can be defined as the computing concept that involves a large number of computer connected through a real-time communication network such as the internet [Alam et al., 2015]. Cloud Computing applications range from science to engineering, gaming, and social networking. In scientific applications domain, Cloud Computing is applied to High Performance Computing (HPC) [Ekanayake and Fox, 2009], High Throughput Computing (HTC) [Rho et al., 2012], and data intensive applications [Shamsi et al., 2013]. In Healthcare domain, Cloud Computing is supporting doctors to provide more effective diagnostics [Kumar and Chaithra, 2015] and building healthcare monitoring systems [Deepa and Boopathy, 2014]. In Biology, Cloud Computing is helping in prediction of protein structure [Li et al., 2012], designing new drugs for the treatment of diseases [Nony et al., 2014], and supporting Gene Expression Data Analysis for Cancer Diagnosis [Vecchiola et al.,

2009]. While, in geoscience domain, Cloud Computing is supporting in satellite image processing [Golpayegani and Halem, 2009]. At last from the business domain, Cloud Computing is supporting Customer Relationship Management (CRM) [Shaqrah, 2016] and Enterprise Resource Planning (ERP) [Saini et al., 2011] systems.

There are many service providers available for the cloud. Cloud Services are provided to end users using a standard pay-as-per-use model. Organizations have to take a decision which service provider's services are more advantageous to the organization. The cost of purchasing the services from different service providers leads to increased budget and wastage of time. A comprehensive study of advantages and disadvantages of Cloud Computing in the real Internet platform is very expensive and difficult. This requires interaction with several computing and network elements that cannot be controlled or managed by application developers.

In addition, the service providers themselves need to examine the proposed updates to their service before getting them applied. They don't need to apply the changes and then being faced with issues due to related aspects not covered during the design. Before real-time implementation, system administrators, cloud specialists, and researchers need to measure performance and check all security issues. Moreover, Service Providers need to examine the services from time to time [Lokesh et al., 2015]. Real-time evaluation proves to be costly and impractical, so simulation offers an easy way out to handle this evaluation.

To solve these problems, Cloud Simulation tools have been developed to study and evaluated of cloud computing technology. These tools include different algorithms and models, so organizations and service providers can change them to meet the problem resolution before investing in hardware and software. Cloud Simulators allow to change input very easily as when needed, which provide better results. In addition, they do not need the deep knowledge to use for setting up and experimenting a simulated cloud.

But getting the right tool for a given scenario or knowing what features each tool has is a challenging and complex task. It is necessary to perform a comprehensive survey and critical evaluation of them. This paper evaluates some of the Cloud Simulators used for the purpose of simulation and modeling of Cloud Computing. In next we discuss the background and related work. While, the section after presents the common architecture of Cloud Simulators. Then, we provide the evaluation criteria of the selected Cloud Simulators and finally we present conclusion and future work.

Background

Cloud Computing is considered as a virtual pool of computing resources. It provides a whole dynamic computing system for application environment to the users. Cloud Computing environments have specific characteristics such as heterogeneity, dynamicity, and scalability which require particular research tools [Lokesh et al., 2015]. Performing real cloud experiments are challenged with some limitations on software and hardware reconfiguration and rescaling which impede performance of the whole system. Service providers and researchers need to tune their proposed Cloud Computing method under different scenarios and varying number of resources/users to realize its potential before using in a real environment.

Simulation is a science and technique to make a model of process or a real system, it is designed to evaluate and test strategies. Simulation aims to study and understand the behavior of the system or evaluating various strategies [Ingalls, 2008]. Simulator plays the role for designing a model of a real system and conducting experiments with this model [Das et al., 2014]. The model is used later on for the purpose of either of understanding the behavior of the system or of evaluating various strategies for the operation of the system [Buyya et al., 2009].

Salama et al. [Salama et al., 2013] proposed a generic framework for modeling and simulation of Cloud Computing Services. They assume that the Cloud Simulation Framework needs to fulfill the following functional requirements: (1) Support a list of quality metrics, from which the user will select his preferences, is carried out, (2) Support solver for user problems like optimum service select, (3) Dynamically run the simulation, (4) Set preferences for different criteria and use the preferences to run any service related problem, (5) Application should also display a graphical representation for the final calculations, as well as intermediate results, and (6) The system should allow the decision makers to repeat their experience under various parameters.

From the other hand, Oujani and Jain [Oujani and Jain, 2013] indicated that there is two types of Cloud Simulators, Cloud Simulators based on software only and Cloud Simulators based on software and hardware. They compared eight Cloud Simulators and the comparison criteria was underlying platform, programming language, and hardware/software composition. In addition, Guérout et al. [Guérout et al., 2013] provided a survey on energy-aware simulation techniques with DVFS (Dynamic Voltage and Frequency Scaling).

Sakellari and Loukas [Sakellari and Loukas, 2013] provided another way to classify Cloud Simulators into Mathematical Modeling of Cloud Systems, Cloud Simulation Software Tools, and Cloud Testbeds. They evaluated some of the existing mathematical models, Cloud Simulators and testbeds based on the criteria of energy efficiency, Quality of Service (QoS), programming language, and availability as an open source tool.

Based on the above surveys, it is shown that all the review generally provide the information about various available approaches and their features. They do not however provide clear guidelines as to which approach is suitable for a particular situation. Hence, we consider a complementary review of the existing approaches. The following sections describe the common architecture of Cloud Simulators. Then evaluation criteria is proposed and simulators evaluated.

Common Architecture of Cloud Simulators

Cloud is considered as a large pool of resources (CPU, Memory, Network Bandwidth, Disk I/O, Library … etc.) which can be accessed through set of APIs [Alam et al., 2015]. Figure 1.a shows the architecture of Cloud Computing, while Figure 1.b shows the common architecture of Cloud Simulators.

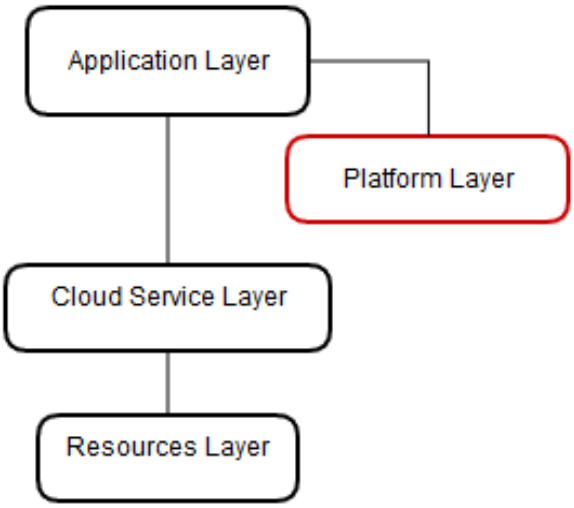


Figure 1.a Cloud Architecture

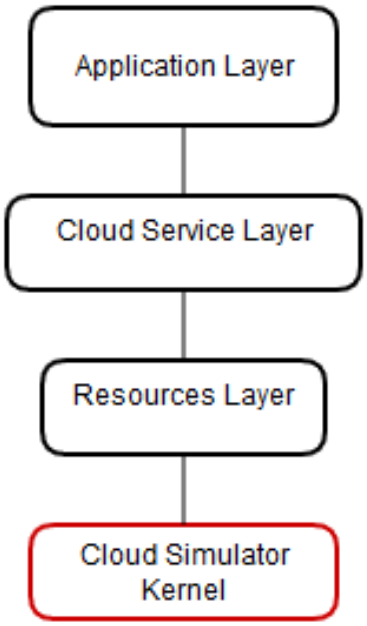


Figure 1.b Cloud Simulator Architecture

Both have four layers. Three out of the four layers are shared between Cloud Computing and Cloud Simulators and they are Resources Layer, Cloud Service Layer, and Application Layer. Resources Layer consists of the hardware devices including the CPU, Memory, Storage and Network Bandwidth which are physically resident on a server farm. Due to limited number of resources, resources needed to be utilized. To utilize the resources layer, virtualization technology is used to provide Infrastructure as a Service (IaaS). Examples of IaaS providers are Amazon Web Services [AWS, 2017] and Rackspace [Rackspace, 2017] which provides a pool of Virtual Machines (VMs) that can be provisioned based on users' requests. The Cloud Services Layer virtualizes available resources in the Cloud Computing System as stream of resources available for provisioning to users' requests. The Application Layer is the layer in which cloud users can submit their applications to the Cloud Services Layer to consume the resources. At the Application Layer service providers can provision ready-to-use software and applications for the business needs of the cloud users. Hence, this layer provides the Software as a Service (SaaS). Examples of SaaS providers are Google Apps [Google App Engine, 2017] and Salesforce [Salesforce, 2017].

Cloud Computing Architecture has a distinct layer called the Platform Layer. The Platform Layer is the Software Development Kit (SDK) which contains the interface the user application can use to communicate with the Cloud Service Layer. It simplifies the development of the Cloud Applications and contains references to call the Cloud Service Layer Application Programming Interface (APIs). This layer provides the Platform as a Service (PaaS). Examples of PaaS providers are Google App Engine [Google App Engine, 2017] and Windows Azure [Windows Azure, 2017].

At the end, Cloud Simulator Architecture has Cloud Simulator Kernel Layer which is the layer that contains the libraries that manage the simulation and its parameters. It also contains the configuration of the cloud research experiments that need to run on the virtual resources.

Evaluation Criteria and Cloud Simulators

In this section, we provide list of evaluation criteria with brief about each criteria then we discuss some available Cloud Simulators and get them evaluated based on the mentioned evaluation criteria.

Evaluation Criteria

We consider the six functional requirements of Salama et al. [Salama et al., 2013] (quality metrics, solver for user problems, dynamically run the simulation, preferences, graphical representation, and repeatability), three requirements from Oujani and Jain [Oujani and Jain, 2013] (underlying platform,

programming language, and hardware/software companions), and three types of services (IaaS, PaaS, SaaS) supported from the Cloud Architecture as our first set of evaluation criteria. Additional to them, we proposed these requirements for evaluation of the Cloud Simulators as we studied the simulators:

1. **Federation Policy** is to allow Cloud Applications to run on heterogeneous clouds in different domains.
2. **Modeling of Public Cloud Providers** is to provide modeling capabilities of public cloud providers like Amazon, Oracle and Google.
3. **Migration Policy** is when, which and where to migrate a virtual machine.
4. **Security** is whether the Simulator support applying security policies on users, resources and access to the modeled system.
5. **Mobile Cloud Computing** is to support integration between Cloud Computing and Mobile Service system to offload mobile data and intensive computation requirements to the Cloud infrastructure.
6. **Desktop Cloud Computing** is to support integration with Desktop Machines to offload the intensive computation requirements.
7. **Cost Modeling** determines the price of the service usage based on a model or system policies.
8. **Communication Modeling** concerns with the costs involved in the data center communication.
9. **Energy Modeling** is to model the energy by the aim to reduce the heat produced in the data center.
10. **Power Saving Modes** are the modes for saving power consumption in data centers.
11. **Physical Modeling** is to model physical layer entities such as cache, allocation policies for memory, file system models … etc.
12. **Application Models** are the models supported by the framework for different application components.
13. **Availability** specifies whether the simulator is commercial or available as in open source.
14. **Simulation Time** determines how long the simulator takes to perform the simulation.
15. **Parallel Experiments** is to run the modeling experiments through several machines.
16. **Simulator Type** is whether the Simulator an event based or packet level. Event Based Simulators model the operations of the system as a discrete sequence of events. Packet Level Simulators analyze packets interaction between different network entities.

Cloud Simulators

1. **CloudSim [Calheiros et al., 2011]** is a complete extendible simulation tool for modeling and simulation of Cloud Computing. It allows extending and defining policies for all system components. It supports both system and behavior modeling like data centers, virtual machines and resource provisioning. It is considered the most popular Cloud Simulation tool.
2. **CloudAnalyst [Wickremasinghe et al., 2010]** is based on CloudSim and SimJava frameworks. It is developed to simulate Cloud applications with the purpose of studying the behavior of such applications under various deployment configurations. It supports configuring any geographically distributed system including information of geographic location of users and data centers.
3. **GreenCloud [Kliazovich et al., 2010]** is a packet level simulator that is specially made for energy-aware environment. It calculates energy consumption of all data center components and communication between the packet levels. It is designed to capture details of the energy consumed by data center components as well as packet-level communication patterns between them.
4. **iCanCloud [Núñez et al., 2012]** is complete simulation framework for cloud infrastructures. It is specially focused on the simulation of Amazon instance types. In addition, it allows design and implementation of a flexible hypervisor module that provides an easy method for integrating both existent and new cloud brokering policies. It can predicts the trade-offs between cost and performance of applications executed in modeled hardware.
5. **MDCSim [Lim et al., 2009]** is developed to measure performance, energy, and infrastructural availability cost for multi-tier data center simulation. It simulates data center components such as servers, switches, and communication links. MDCSim has been validated through a three tier Linux Cluster based data center connected with InfiniBand Architecture (IBA) and 10-gigabit Ethernet (10 GbE) under different conditions and cluster specifications.
6. **NetworkCloudSim [Garg and Buyya, 2011]** is an extension to CloudSim with a scalable network and generalized application model, which provides accurate evaluation scheduling and resource provisioning policies. It supports modeling complex applications with data driven applications and workflow. As it implements network flow model design with low computational overhead.
7. **EMUSIM [Calheiros et al., 2013]** is a simulator and emulator of a cloud environment based on CloudSim and Automated Emulation Framework (AEF). It is doing that by extracting information

from applications through emulation then use this information to generate the corresponding simulation model. Emulation has scalability limitations due to either hardware constraints or difficulty in generating large and realistic workloads.

8. **CloudReports [Sá et al., 2014]** is a graphic tool that simulates distributed computing environments based on the Cloud Computing. CloudReports provide different aspects for researcher to play role of service providers and users. Supported types of extensions in the CloudReports are broker policies, virtual machines allocation policies, power consumption models, virtual machines schedulers, and resource utilization models.
9. **CloudSched [Tian et al., 2013]** is a Cloud resources scheduling emulator. It helps to identify and explore the optimal solutions for different resource scheduling policies and algorithms. Different resource scheduling policies and algorithms can be compared with each other for performance evaluation.
10. **CloudExp [Jararweh et al., 2014]** is a modeling and simulation environment which introduced a specialized mobile cloud computing experimental framework. It conducts various mobility scenarios for mobile devices. It provides user-friendly GUI to enhance the users' experience in building their own infrastructure. In addition, it allows researchers to study the communication cost between users and cloud.
11. **DCSim [Tighe et al., 2011]** is developed to simulate a virtual data center. It is using centralized management and neglects the network topology. DCSim provides extra features of replicated Virtual Machines (VMs) with a multi-tier application model to simulate dependencies between VMs, VM replication as a tool for handling increasing workload.
12. **ICARO [Badii et al., 2016]** is a cloud simulator developed in the ICARO project. The main aim of this simulator is to analyze the changes on workload in a data center when the structure of workload is modified dynamically in real time. All, other simulators study change of the structure of the data center, but ICARO is interested in the impact of workload changes on the data center changes like adding move Virtual Machines.
13. **SPECI [Sriram, 2009]** is a discrete event simulation tool for Elastic Cloud Infrastructure that enables exploration of scaling properties of large data centers. The aim of this project is to simulate the performance and behavior of data centers, given the size and middleware design policy as input. SPECI does not provide any support for VMs, load balancing, security and job scheduling.

14. **GroudSim [Ostermann et al., 2010]** is a simulator for both Cloud and Grid Computing environments. It is developed for scientific workflow applications. The developed simulation framework supports modelling of network resources, job submissions, file transfers, as well as integration of failure, background load, and cost models.
15. **SmartSim [Shiraz et al., 2012]** is developed to simulate applications for Mobile Cloud Computing running in Smart Mobile Devices (SMDs). It simulates the behavior of the mobile devices and resources intensive mobile applications. In addition, SmartSim models the mechanism of runtime partitioning of elastic mobile application and determines resources utilization on SMDs during the execution of the elastic application.
16. **SimIC [Sotiriadis et al., 2013]** is aiming to achieving interoperability, flexibility and service elasticity while at the same time introducing the notion of heterogeneity of multiple cloud configurations. It uses Inter-Cloud Meta Scheduling (ICMS) algorithm for inter-cloud scheduling with several distributed parameters.
17. **DynamicCloudSim [Bux and Leser, 2015]** is an extension of CloudSim to simulate instability caused due to heterogeneity of cloud computing, dynamic changes due to several factors at runtime and failures during task execution. Furthermore, DynamicCloudSim introduces a fine-grained representation of computational resources, thereby enabling the simulation of executing different kinds of applications (CPU-, I/O-, communication-bound) on machines with different performance characteristics.
18. **CloudSimSDN [Son et al., 2015]** is based on CloudSim. It is a scalable simulation environment to analyze the network allocation capacity policies like measuring the network performance and host capacity allocation approaches simultaneously within a data center.
19. **secCloudSim [Rehman et al., 2014]** is an extension of iCanCloud simulator which provides security features like authentication and authorization. However, it does not support advanced security mechanisms like privacy, integrity and encryption of VMs.
20. **CEPSim [Higashino et al., 2016]** (Complex Event Processing Simulator) is an extension to CloudSim that allows to simulate cloud applications based on directed acyclic graphs used to represent continuous CEP queries. It includes simulating queries in heterogeneous cloud environments under different load conditions.
21. **PICS [Kim et al., 2015]** (Public IaaS Cloud Simulator) is a simulator to evaluate the cost and performance of public IaaS cloud along dimensions like VM, storage service, resource elasticity,

job scheduling and diverse workload patterns. It does not support heterogeneous cloud deployment feature nor modeling the communication costs.

22. **TeachCloud [Jararweh et al., 2013]** is an extension to CloudSim that helps students to have hands-on experiment with various components involved in Cloud environment such as processing elements, data centers, networking, Service Level Agreement (SLA) constraints, web-based applications, Service Oriented Architecture (SOA), virtualization, management and automation, and Business Process Management (BPM).
23. **CDOSim [Fittkau et al., 2012]** is a simulation tool providing Cloud Deployment Options (CDOs) for Software Migration Support. It identifies the process of analyzing potential CDOs manually is intractable, costly, and time consuming. CDOSim simulates various properties of CDOs such as response times, SLA violations, and costs. CDOSim is integrated with their own Cloud migration framework CloudMIG.
24. **CloudNetSim++ [Malik et al., 2014]** is a simulator for distributed data centers. CloudNetSim++ supports to analyze energy consumption by varying number of nodes and other parameters. In addition to standard network performance measures like delay and throughput for various topologies. It provides a rich GUI, and communication among different nodes which is achieved through packets.
25. **DartCSim+ [Li et al., 2013]** is an enhancement to CloudSim. It integrates power and network models so making network and scheduling algorithms power-aware. In addition, it has a mechanism for controlling transmission of network links is also added which solve the problem of distortion. It hides simulation details and provides friendly GUI for users to conduct their experiments.
26. **GDCSim [Gupta et al., 2014]** is developed to support and handle compute-aware applications such as computational fluid dynamics (CFD). GDCSim supports online resource management and makes prediction of performance and energy consumption for data centers. Having three components, individual component can be used independently or they can be used all together. The simulator can be used for data center infrastructure management, facility performance analysis, workload scheduling or thermal modeling.
27. **FlexCloud [Xu et al., 2015]** is a flexible and scalable simulator that simplifies the scheduling process and enables cloud data centers initializing, VM requests allocation, and performance evaluation for various scheduling algorithms. It offers infrastructure as a Service. It has

advantage over CloudSim in computing time and memory consumption to support large-scale simulations.

28. **VirtualCloud [Das, 2010]** is developed to get efficient usage of virtualized resources (specifically testing policies on real world) in Cloud environment and it is yet useful for modeling and testing new policies.
29. **Cloud²Sim [Kathiravelu and Veiga, 2014]** is an adaptively scaling middleware platform for concurrent and distributed cloud and MapReduce simulations, by leveraging CloudSim. Cloud²Sim proposes a distributed concurrent architecture to CloudSim simulations by using Hazelcast in-memory data grid. In addition, it adopts an adaptive architecture to elastically scale the resources made available to the simulation.
30. **DesktopCloudSim [Alwabel et al., 2015]** as an extension tool CloudSim that enables the simulation of node failures in the infrastructure of Cloud. It demonstrated that the tool can be used to study the throughput of a Desktop Cloud using NotreDame real traces.
31. **WorkflowSim [Chen and Deelman, 2012]** extends the CloudSim simulation toolkit to support workflow preparation and execution. In addition, it includes implementation of a stack of workflow parser, workflow engine and job scheduler. It supports set of workflow scheduling algorithms (e.g., HEFT, Min-Min, and Max-Min) and task clustering algorithms.
32. **CloudMIG [Frey and Hasselbring, 2011]** facilitates the comparison and planning phases for the migration of software systems to PaaS or IaaS-based Cloud environments. Code models can be extracted from Java-based software to (1) model the current system deployment and augment it with a present workload profile, (2) compare the trade-offs that have to be made for different cloud deployment options, and (3) automatically transform the system model to a CloudSim model to enable integrated simulation of various cloud deployment options regarding future costs, response times, and SLA violations.
33. **EduCloud [Cemim et al., 2012]** is a tool to build testbeds using standard hardware and software. EduCloud executes tasks related to the management of a cloud infrastructure, serving as an option to demonstrate the concepts of cloud computing. It also enables the deployment of a private cloud using heterogeneous resources, composed by common hardware usually found in academic environments.

Tables 1, 2, 3, and 4 show evaluation of some of the Cloud Simulators against the functional requirements mentioned in sub-section 4.1. We grouped the requirements based on cloud services

related requirements, special aspects related requirements, cloud components modeling requirements, and simulation related requirements.

Table 1. Cloud Simulators Evaluation based on Cloud Services Requirements

Requirement / Simulator	Migration Policy	Modeling of Public Cloud Providers	Federation Policy	Support SaaS	Support PaaS	Support IaaS
CloudSim	√	X	√	X	X	√
CloudAnalyst	X	X	√	X	X	√
GreenCloud	X	X	X	X	X	√
iCanCloud	X	Amazon EC2	X	HPC	X	√
MDCSim	X	X	X	X	X	X
NetworkCloudSim	√	X	√	X	X	√
EMUSIM	X	Amazon EC2	X	X	X	√
CloudReport	X	X	X	√	X	√
CloudSched	X	Amazon EC2	X	X	X	√
CloudExp	X	X	X	√	√	√
DCSim	√	X	X	X	√	√
ICARO	X	X	X	X	√	√
SPECI	X	X	X	X	X	√
GroudSim	X	X	X	X	X	√
SmartSim	X	X	√	X	X	√
SimIC	X	X	√	X	X	X
DynamicCloudSim	X	X	√	X	X	√
CloudSimSDN	X	X	√	X	X	√
SecCloudSim	X	X	X	HPC	X	√
CEPSim	X	X	√	X	X	√
PICS	X	Amazon S3	X	X	X	√
TeachCloud	X	√	√	X	X	√
CDOSim	X	√	√	X	X	√
CloudNetSim++	X	X	√	X	X	√
DartCSim+	X	X	√	X	X	√
GDCSim	X	X	X	X	X	√
FlexCloud	√	Amazon EC2	X	X	X	√
VirtualCloud	√	X	X	X	X	√
Cloud²Sim	X	X	√	X	X	√
DesktopCloudSim	X	X	X	X	X	√
WorkflowSim	X	X	X	X	X	√
CloudMIG	X	X	X	X	√	√
EduCloud	X	X	X	X	X	√

Table 2. Cloud Simulators Evaluation based on Special Related Requirements

Requirement / Simulator	Problems Solver	Quality Metrics	Desktop Cloud	Mobile Cloud	Security
CloudSim	√	√	X	X	X
CloudAnalyst	√	√	X	X	X
GreenCloud	√	√	X	X	X
iCanCloud	√	√	X	X	X
MDCSim	√	√	X	X	X
NetworkCloudSim	√	√	X	X	X
EMUSIM	√	√	X	X	X
CloudReport	√	√	X	X	X
CloudSched	√	√	X	X	X
CloudExp	√	√	X	√	√
DCSim	√	√	X	X	X
ICARO	√	√	X	X	X
SPECI	√	√	X	X	X
GroudSim	√	√	X	X	X
SmartSim	√	√	X	√	X
SimIC	√	√	X	X	X
DynamicCloudSim	√	√	X	X	X
CloudSimSDN	√	√	X	X	X
SecCloudSim	√	√	X	X	√
CEPSim	√	√	X	X	X
PICS	√	√	X	X	X
TeachCloud	√	√	X	X	X
CDOSim	√	√	X	X	X
CloudNetSim++	√	√	X	X	X
DartCSim+	√	√	X	X	X
GDCSim	√	√	X	X	X
FlexCloud	√	√	X	X	X
VirtualCloud	√	√	X	X	X
Cloud²Sim	√	√	X	X	X
DesktopCloudSim	√	√	√	X	X
WorkflowSim	√	√	√	X	X
CloudMIG	√	√	X	X	X
EduCloud	√	√	X	X	X

Table 3. Cloud Simulators Evaluation based on Cloud Components Modeling Requirements

Requirement / Simulator	Application Models	Physical Modeling	Power Saving Modes	Energy Modeling	Communication Modeling	Cost Modeling
CloudSim	Computation & Data Transfer	X	DVFS	√	Limited	√
CloudAnalyst	Computation & Data Transfer	X	DVFS	√	Limited	√
GreenCloud	Computation, Data Transfer, & Execution Deadline	X	DVFS, DNS, & Both	Servers & Network	√	X
iCanCloud	Computation & Data Transfer	√	X	X	√	√
MDCSim	Computation	X	X	Servers	Limited	X
NetworkCloudSim	Computation, Data Transfer, & Execution Deadline	X	√	√	√	√
EMUSIM	Computation & Data Transfer	X	√	√	Limited	√
CloudReport	Computation & Data Transfer	√	X	Servers & Network	√	√
CloudSched	Computation & Data Transfer	√	√	Servers & Network	√	X
CloudExp	Computation & Data Transfer	√	√	Devices & Network	√	√
DCSim	Computation & Data Transfer	X	√	Hosts	X	√
ICARO	Multi-Tier	X	X	X	X	X
SPECI	Computation & Data Transfer	X	X	X	√	X
GroudSim	Computation & Data Transfer	X	√	X	X	√
SmartSim	Computation & Data Transfer	X	√	Devices & Network	Limited	√
SimIC	Computation & Data Transfer	X	X	Servers	Limited	√
DynamicCloudSim	Computation & Data Transfer	X	√	X	√	√
CloudSimSDN	Computation & Data Transfer	X	√	Servers & Network	√	√
SecCloudSim	Computation & Data Transfer	√	X	X	√	√
CEPSim	Computation & Data Transfer	X	√	√	Limited	√
PICS	Computation, Data Transfer, & Execution Deadline	X	X	X	X	√
TeachCloud	Computation & Data Transfer	X	√	√	Limited	√
CDOSim	Computation & Data Transfer	X	√	√	Limited	√
CloudNetSim++	Computation & Data Transfer	X	√	√	Limited	√
DartCSim+	Computation & Data Transfer	X	√	√	Limited	√
GDCSim	Computation & Data Transfer	X	√	√	X	X
FlexCloud	Computation & Data Transfer	√	√	√	Limited	√
VirtualCloud	Computation & Data Transfer	√	X	X	Limited	X
Cloud²Sim	Computation & Data Transfer	X	√	√	Limited	√
DesktopCloudSim	Computation & Data Transfer	X	√	√	Limited	√
WorkflowSim	Computation & Data Transfer	X	√	√	Limited	√
CloudMIG	Computation & Data Transfer	X	√	√	Limited	√
EduCloud	Computation & Data Transfer	X	√	√	Limited	√

DVFS: Dynamic Voltage and Frequency Scaling. DNS: Dynamic Network Shutdown.

Table 4. Cloud Simulators Evaluation based on Simulation Related Requirements

Simulator Type	Parallel Experiments	Simulation Time	Availability	Hardware / Software Companion	Programming Language	Underlying Platform	Experiments Repeatability	Graphical Interface	User Preferences	Dynamic Run	Requirement / Simulator
Event-Based	X	Seconds	√	Software	Java	SimJava	√	X	√	√	CloudSim
Event-Based	X	Seconds	√	Software	Java	CloudSim / SimJava	√	√	√	√	CloudAnalyst
Packet Level	X	Minutes	√	Both	C++ / oTcl	NS2	√	√	X	√	GreenCloud
Event-Based	WiP	Seconds	√	Software	C++	SIMCAN, OMNET, MPI	√	√	√	√	iCanCloud
Event-Based	√	Seconds	X	Software	C++ / Java	CSIM	√	X	√	√	MDCSim
Packet Level	X	Seconds	√	Software	Java	CloudSim	√	X	√	√	NetworkCloudSim
Event-Based	X	Seconds	√	Both	Java	CloudSim & AEF	√	√	√	√	EMUSIM
Event-Based	X	Seconds	X	Software	Java	CloudSim	√	√	√	√	CloudReport
Event-Based	X	Seconds	√	Software	Java	-	√	√	√	√	CloudSched
Event-Based	X	Seconds	X	Software	Java	CloudSim	√	√	√	√	CloudExp
Event-Based	X	Minutes	√	Software	Java	-	√	X	√	√	DCSim
Event-Based	X	Seconds	√	Software	Java	-	√	√	√	√	ICARO
Event-Based	X	Seconds	√	Software	Java	SimKit / Java DES	√	X	√	√	SPECI
Event-Based	X	Seconds	√	Software	Java	-	√	X	√	√	GroudSim
Event-Based	X	Seconds	√	Software	Java	CloudSim	√	X	√	√	SmartSim
Event-Based	X	Seconds	X	Software	Java	SimJava	√	X	√	√	SimIC
Event=Based	X	Seconds	√	Software	Java	CloudSim	√	X	√	√	DynamicCloudSim
Packet Level	X	Seconds	√	Software	Java	CloudSim	√	√	√	√	CloudSimSDN
Event-Based	X	Seconds	X	Software	C++	iCanCloud	√	√	√	√	SecCloudSim
Event-Based	X	Seconds	X	Software	Java	CloudSim	√	√	√	√	CEPSim
Event-Based	X	Minutes	√	Software	Python	-	√	X	√	√	PICS
Event-Based	X	Seconds	√	Software	Java	CloudSim	√	√	√	√	TeachCloud
Event-Based	X	Seconds	X	Software	Java	CloudSim	√	X	√	√	CDOSim
Packet Level	X	Seconds	√	Software	Java	CloudSim	√	√	√	√	CloudNetSim++
Packet Level	X	Seconds	√	Software	Java	CloudSim	√	√	√	√	DartCSim+
Event-Based	X	Seconds	√	Software	C++/XML	Bluetool	√	X	√	√	GDCSim
Event-Based	X	Seconds	√	Software	Java	-	√	√	√	√	FlexCloud
Event-Based	X	Seconds	X	Software	XML	-	√	√	√	√	VirtualCloud
Event-Based	X	Seconds	√	Software	Java	CloudSim	√	X	√	√	Cloud²Sim
Event-Based	X	Seconds	√	Software	Java	CloudSim	√	X	√	√	DesktopCloudSim
Event-Based	X	Seconds	√	Software	Java	CloudSim	√	X	√	√	WorkflowSim
Event-Based	X	Seconds	√	Software	Java	CloudSim	√	X	√	√	CloudMIG
Event-Based	X	Seconds	√	Both	Java	-	√	X	√	√	EduCloud

Discussion

The first step in selecting the Cloud Simulator to model a problem is to decide which of the features required in the Cloud Simulator. Many challenges are facing the Cloud Computing like: security, cost modeling, energy management, and virtual machine migration. Unfortunately, most of the developed simulators were developed to handle one challenge only. That's prevent getting a standard platform for Cloud Simulation. This sub-section details the evaluation discussion based on our study of Cloud Simulators mentioned in Tables 1, 2, 3, and 4. The interesting results are mentioned below with explanations.

From tables 1, 2, 3 and 4 it is observed that:

- All studied simulators support quality metrics (e.g response time, throughput … etc). Each simulator supports a set of quality metrics which may differ from other simulators but they are sharing the implementation of extending the quality metrics with more metrics based on the problem requirements.
- All studied simulators support suggesting optimum configuration of the cloud elements that support the studied environment.
- All studied simulators support dynamic run of the experiments that gives the user the ability to change the elements configuration while the experiment is running.
- All studied simulators support storing user preferences for the experiments parameters so that the user can keep them for future run of the experiment.
- Around 48% of the Cloud Simulators under study have Graphical User Interface (GUI). Researchers prefer simulators with support of the GUI to make it easier to configure and run the experiments.
- All studied simulators support running experiments for several time with the same user preferences so that user can get average of the experiments results.
- Around 52% of the Cloud Simulators are built upon CloudSim simulator. This is give an indication why Java is the most common programming language used for developing Cloud Simulators under study. The second predominant base programming language is C++. Only PICS simulator was developed using Python.
- GreenCloud, EMUSIM, and EduCloud are the only simulators that combines simulation and modeling of the hardware with simulation of the cloud software.

- Around 94% of the simulators support simulating IaaS, while few of them support simulating PaaS and SaaS.
- Around 76% of the Cloud Simulators under study are available under the licenses of open source code for download.
- Around 79% of the Cloud Simulators studied support modeling of the cost. This is important for cloud providers to examine the new pricing plans and strategies.
- Few of the cloud simulators studied don't support getting a communication model in place for interaction of the cloud elements.
- Most of the simulators studied are fast and show the experiments results in tense of seconds.
- Around 63% of the studied simulators support energy modeling by modeling servers, devices, network, or all of connected devices at the same time.
- Around 73% of the studied simulators support power saving modes or event collecting information about devices power consumption.
- Around 39% of the Cloud Simulators studied support federation policy simulator as the need to study multiple clouds connected together.
- Around 21% of the Cloud Simulators studied support modeling of the physical components of the resources like allocation policy of memory, scheduling algorithm of the tasks, latency of I/O… etc.
- Four Cloud Simulators support modeling of Amazon EC2 and one simulator support modeling of Amazon S3.
- iCanCloud is the only Cloud Simulator which has a plan to get the feature of running several independent experiments in parallel to utilize the available resources for the simulator.
- All of the studied simulators support application models through parameterizing computation, data transfer and some of them support execution deadlines.
- All simulators interested in network simulation are based on simulating packets communicated among the cloud elements.
- CloudExp and secCloudSim are the only simulators supporting security features while modeling the interaction between Cloud elements.

- CloudExp and SmartSim are Cloud Simulators supporting modeling of mobile devices connected to the cloud infrastructure to offload data and intensive computations.
- DesktopCloudSim and WorkflowSim are Cloud Simulators supporting modeling of desktop machines connected to the cloud infrastructure to offload data and intensive computations.
- CloudSim, NetworkCloudSim, DCSim, FlexCloud, and VirtualCloud are the only studied simulators studying VM migration policies.

Conclusion and Future Work

Cloud computing is getting computing resources available over a network as a service to end user. It is growing at a much faster rate and faced with many challenges. To carry out a fundamental research in Cloud Computing, Cloud Simulators are considered to be a better option rather than real deployment of cloud. This paper discussed the benefits of cloud simulators along with brief descriptions of 33 Cloud Simulators. The management, load, and test tools are found for Cloud Computing. All the 33 Cloud Simulators have been compared based on 28 evaluation criteria namely: (1) quality metrics, (2) solver for user problems, (3) dynamically run the simulation, (4) preferences, (5) graphical representation, (6) experiments repeatability, (7) underlying platform, (8) programming language, (9) hardware/software companions, (10) support IaaS, (11) support PaaS, (12) support SaaS, (13) federation policy , (14) modeling of public cloud providers , (15) VM migration policy, (16) security, (17) support mobile cloud computing, (18) support desktop cloud computing, (19) cost-modeling, (20) communication modeling, (21) energy modeling, (22) power saving models, (23) physical modeling, (24) application models, (25) availability (open-source), (26) simulation time, (27) parallel experiments, and (28) simulator type (event-based, packet-level). The results and discussion of the evaluation analysis have been presented. Although, there are several Cloud Simulators developed, choosing the best simulator for an experiment or for a proposed resolution of an issue depends up on the type of the problem. None of the introduced tools is the best of all. Each simulator is proposed for a specific application and conditions of cloud environment and the researchers choose the best adapted tool according to their requirements and applications.

Bibliography

[Lokesh et al., 2015] Lokesh V., Jayaraman S., Soni A. N, A., and Guruprasad H. S., A Survey on the capabilities of Cloud Simulators. International Journal of Engineering Research and General Science, Vol.3, Issue 4, 2015. pp. 958-969. ISSN 2091-2730.

[Alam et al., 2015] Alam Md. I., Pandey M., and Rautaray S. S., A Comprehensive Survey on Cloud Computing. International Journal of Information Technology and Computer Science (IJITCS), Vol.7, Issue 2, 2015. pp. 68-79.

[Ekanayake and Fox, 2009] Ekanayake J. and Fox G., High Performance Parallel Computing with Clouds and Cloud Technologies. In: First International Conference on Cloud Computing CloudComp, 2009. Vol. 34. pp. 20-38.

[Rho et al., 2012] Rho S., Kim S., Kim Sa., Kim Se., Kim J.-S., and Hwang S., HTCaaS: a large-scale high-throughput computing by leveraging grids, supercomputers and cloud. High Performance Computing, Networking, Storage and Analysis (SCC), 2012 SC Companion, 2012. pp. 1341-1342.

[Shamsi et al., 2013] Shamsi J., Khojaye M. A., and Qasmi A., Data Intensive Cloud Computing: Requirements, Expectations, Challenges, and Solutions. Journal of Grid Computing, Vol.11, Issue 2, 2013. pp. 281-310.

[Kumar and Chaithra, 2015] Kumar P. S. J. and Chaithra A. S., A Survey on Cloud Computing based Health Care for Diabetes: Analysis and Diagnosis. IOSR Journal of Computer Engineering (IOSR-JCE), Vol.17, Issue 4, 2015. pp. 109-117.

[Deepa and Boopathy, 2014] Deepa R. and Boopathy K., Cloud Care: A Remote Health Monitoring System. Topics in International Journal of Engineering Science and Innovative Technology (IJESIT), Vol.3, Issue 1, 2014. pp. 264-271.

[Li et al., 2012] Li H., Tounkara J.-C., and Liu C., Prediction of Protein-Protein Docking Sites Based on a Cloud-Computing Pipeline. International Journal of Machine Learning and Computing IJMLC, Vol.2, Issue 6, 2012. pp. 798-801.

[Nony et al., 2014] Nony P., Kurbatova P., Bajard A., Malik S., Castellan C., Chabaud S., Volpert V., Eymard N., Kassai B., Cornu C., and the CRESim and Epi-CRESim study groups, A Methodological Framework For Drug Development. Rare Diseases. Orphanet Journal of Rare Diseases, Vol.9, Issue 164, 2014.

[Vecchiola et al., 2009] Vecchiola C., Abedini M., Kirley M., Chu X., and Buyya R., Classification of Gene Expression Data on Public Clouds. Cloud Computing and Distributed Systems (CLOUDS)

Laboratory, Dept. of Computer Science and Software Engineering, The University of Melbourne, Australia 2009.

[Golpayegani and Halem, 2009] Golpayegani N. and Halem M., Cloud Computing for Satellite Data Processing on High End Compute Clusters. In: Proceedings of the 2009 IEEE International Conference on Cloud Computing, 2009. pp. 88-92.

[Shaqrah, 2016] Shaqrah A., Cloud CRM: State-of-the-Art and Security Challenges. (IJACSA) International Journal of Advanced Computer Science and Applications, Vol.7, Issue 4, 2016. pp. 39-44.

[Saini et al. 2011] Saini S. L., Saini D. K., Yousif J. H., and S. V. Khandage, Cloud Computing and Enterprise Resource Planning Systems. Proceedings of the World Congress on Engineering, Vol. 1, 2011. pp. 681-684,

[Ingalls, 2008] Ingalls R. G., Introduction to Simulation. In Proceeding WSC '08 Proceedings of the 40th Conference on Winter Simulation, 2008, pp. 17-26.

[Das et al., 2014] Das P., Kumar A., and Das A., Execution Study of Cloud Simulators Load Balancing Approach. International Journal of Engineering Research & Technology (IJERT), Vol.3, Issue 8, 2014. Paper ID IJERTV3IS081012.

[Buyya et al., 2009] Buyya R., Ranjan R., and Calheiros R. N., Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities. In: Proceedings of the 7th High Performance Computing and Simulation (HPCS 2009) Conference, Leipzig, Germany, 2009. pp. 1-11.

[Salama et al., 2013] Salama M., Shawish A., and Zeid A., A Generic Framework for Modeling and Simulation of Cloud Computing Services. International Journal of Computer Applications, Vol.77, Issue 17, 2013. pp. 33-38.

[Oujani and Jain, 2013] Oujani A. and Jain R., Survey on Cloud Computing Simulations and Cloud Testing, 2013. http://students.cec.wustl.edu/ azinoujani/

[Guérout et al., 2013] Guérout T., Monteil T., Da Costa G., Calheiros R. N., Buyya R., and Alexandru M., Energy-aware simulation with DVFS. Simulation Modelling Practice and Theory, Vol.39, 2013. pp. 76–91.

[Sakellari and Loukas, 2013] Sakellari G. and Loukas G., A survey of mathematical models, simulation approaches and testbeds used for research in cloud computing. Simulation Modelling Practice and Theory, Vol.39, Issue 0, 2013. pp. 92–103.

[AWS, 2017] Amazon Web Services (AWS). https://aws.amazon.com/

[Rackspace, 2017] Rackspace. https://www.rackspace.com/

[Google App Engine, 2017] Google App Engine. https://appengine.google.com/

[Salesforce, 2017] Salesforce. https://www.salesforce.com/

[Windows Azure, 2017] Windows Azure. https://azure.microsoft.com/en-us/

[Calheiros et al., 2011] Calheiros R. N., Ranjan R., Beloglazov A., De Rose C. A. F., and Buyya R., CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. Software - Practice and Experience, Vol.41, Issue 1, 2011. pp. 23–50.

[Wickremasinghe et al., 2010] Wickremasingh, B. Calheiro R. N., and Buyy R., CloudAnalyst: A CloudSim-based Visual Modeller for Analysing Cloud Computing Environments and Applications. In: Proceeding AINA '10 Proceedings of the 2010 24th IEEE International Conference on Advanced Information Networking and Applications, 2010, pp. 446-452.

[Kliazovich et al., 2010] Kliazovich D., Bouvry P., Audzevich Y., and Khan S. U., GreenCloud: a packet-level simulator of energy-aware cloud computing data centers. The Journal of Supercomputing, Vol.62, 2012. pp. 1263-1283.

[Núñez et al., 2012] Núñez A., Vázquez-Poletti J. L., Caminero A. C., Castañé G. G., Carretero J., and Llorente I. M., iCanCloud: A Flexible and Scalable Cloud Infrastructure Simulator. Journal of Grid Computing, Springer, Vol.10, Issue 1, 2012. pp. 185-209.

[Lim et al., 2009] Lim S.-H., Sharma B., Nam G., Kim E. K., and Das C. R., MDCSim: A multi-tier data center simulation, platform. In: IEEE International Conference on Cluster Computing and Workshops CLUSTER '09, 2009, pp. 1-9.

[Garg and Buyya, 2011] Garg S. K. and Buyya R., NetworkCloudSim: Modelling parallel applications in cloud simulations. In: Proceeding of the 4th IEEE International Conference on Utility and Cloud Computing UCC 2011, 2011, pp. 105–113.

[Calheiros et al., 2013] Calheiros R. N., Netto M. A. S., De Rose C. A. F., and Buyya R., EMUSIM: an integrated emulation and simulation environment for modeling, evaluation, and validation of performance of cloud computing applications. Software Practice and Experience, Vol.43, Issue 5, 2013. pp. 595–612.

[Sá et al., 2014] Sá T. T., Calheiros R. N., and Gomes D. G., CloudReports: An Extensible Simulation Tool for Energy-Aware Cloud Computing Environments. In: Cloud Computing: Challenges,

Limitations and R&D Solutions. Ed. Mahmood Z., Cham: Springer International Publishing, 2014. pp. 127-142.

[Tian et al., 2013] Tian W., Zhao Y., Xu M., Zhong Y., and Sun X., A Toolkit for Modeling and Simulation of Real-Time Virtual Machine Allocation in a Cloud Data Center. IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, Vol.12, Issue 1, 2013. pp. 153-161.

[Jararweh et al., 2014] Jararweh Y., Jarrah M., kharbutli M., Alshara Z., Alsaleh M. N., and Al-Ayyoub M., CloudExp: A Comprehensive Cloud Computing Experimental Framework, Journal of Simulation Modelling Practice and Theory, Elsevier, Vol.49, 2014. pp. 180-192.

[Tighe et al., 2011] Tighe M., Keller G., Bauer M., and Lutfiyya H., DCSim: A Data Centre Simulation Tool for Evaluating Dynamic Virtualized Resource Management. In: Network and service management (cnsm) the 8th international conference and workshop on systems virtualization management (svm), 2012, pp. 385–392.

[Badii et al., 2016] Badii C., Bellini P., Bruno I., Cenni D., Mariucci R., and Nesi P., ICARO Cloud Simulator exploiting knowledge base. Journal of Simulation Modelling Practice and Theory, Vol.62, 2016. pp. 1-13.

[Sriram, 2009] Sriram I., SPECI, a simulation tool exploring cloud-scale data centres. In: IEEE International Conference on Cloud Computing CloudCom 2009, Lecture Notes in Computer Science, Vol.5931, 2009. pp. 381-392.

[Ostermann et al., 2010] Ostermann S., Plankensteiner K., Prodan R., and Fahringer T., GroudSim: An event-based simulation framework for computational grids and clouds. In: Proceeding of Euro-Par Conference on Parallel Processing, Vol.6586, 2010. pp. 305–313.

[Shiraz et al., 2012] Shiraz M., Gani A., Khokhar R. H., and Ahmed E., SmartSim: An extendable simulation framework for modeling application processing potentials of smart mobile devices for mobile cloud computing. In: Proceeding of the 10th International Conference on Frontiers of Information Technology, 2012. pp. 331–336.

[Sotiriadis et al., 2013] Sotiriadis S., Bessis N., Antonopoulos N., and Anjum A., SimIC: Designing a new Inter-Cloud Simulation platform for integrating large-scale resource management. In: Proceeding of the 27th International Conference on Advanced Information Networking and Applications, 2013. pp. 90–97.

[Bux and Leser, 2015] Bux M. and Leser U., DynamicCloudSim: Simulating heterogeneity in computational clouds. In: Proceeding SWEET '13 of the 2nd ACM SIGMOD Workshop on Scalable Workflow Execution Engines and Technologies, Vol.46, article no. 1, 2015. pp. 85–99.

[Son et al., 2015] Son J., Dastjerdi A. V., Calheiros R. N., Ji X., Yoon Y., and Buyya R., CloudSimSDN : Modeling and Simulation of Software-Defined Cloud Data Centers. In: Proceeding of the 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), 2015. pp. 475-484.

[Rehman et al., 2014] Rehman U. U., Ali A., and Anwar Z., secCloudSim: Secure Cloud Simulator. In: Proceedings of the 12th International Conference of Frontiers of Information. Technology, 2014. pp. 208–213.

[Higashino et al., 2015] W. A. Higashino, M. A.M. Capretz, and L. F. Bittencourt, CEPSim: Modelling and Simulation of Complex Event Processing Systems in Cloud Environments. Journal of Future Generation Computer Systems, Vol.65, Issue C, 2016. pp. 122-139.

[Kim et al., 2015] Kim I. K., Wang W., and Humphrey M., PICS: A Public IaaS Cloud Simulator. In: Proceeding of the 8th IEEE International Conference on Cloud Computing, 2015. pp. 211-220.

[Jararweh et al., 2013] Jararweh Y., Alshara Z., Jarrah M., Kharbutli M., and Alsaleh M. N., TeachCloud: a cloud computing educational toolkit. International Journal of Cloud Computing, Vol.2, Issue 2/3, 2013. pp. 237-257.

[Fittkau et al., 2012] Fittkau F., Frey S., and Hasselbring W., CDOSim: Simulating cloud deployment options for software migration support. In: Proceeding of the 6th IEEE International Workshop on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems (MESOCA), 2012. pp. 37-46.

[Malik et al., 2014] Malik A., Bilal K., Aziz K., Kliazovich D., Ghani N., Khan S., and Buyya R., Cloudnetsim++: A toolkit for data center simulations in omnet++. In: 11th Annual Conference on High-capacity Optical Networks and Emerging/Enabling Technologies (HONET), 2014. pp. 104-108.

[Li et al., 2013] Li X., Jiang X., Ye K., and Huang P., DartCSim+: Enhanced CloudSim with the power and network models integrated. In: Proceeding of the IEEE International Conference on Cloud Computing CLOUD, 2013. pp. 644–651.

[Gupta et al., 2014] Gupta S. K. S., Banerjee A., Abbasi Z., Varsamopoulos G., Jonas M., Ferguson J., Gilbert R. R., and Mukherjee T., GDCSim: A simulator for green data center design and analysis. Journal ACM Transactions on Modeling and Computer Simulation (TOMACS) - Special issue on simulation in complex service systems, Vol.24, Issue 1, 2014. Article 1.

[Xu et al., 2015] Xu M., Li G., Yang W., and Tian W., FlexCloud: A Flexible and Extendible Simulator for Performance Evaluation of Virtual Machine Allocation. In: Proceeding of the IEEE International Conference on Smart City/SocialCom/SustainCom (SmartCity), 2015. pp. 649-655.

[Das, 2010] Das K., VirtualCloud - A Cloud Environment Simulator. Department of Computer Science and Engineering, Indian Institute of Technology, Bombay, India, 2010.

[Kathiravelu and Veiga, 2014] Kathiravelu P. and Veiga L., An Adaptive Distributed Simulator for Cloud and MapReduce Algorithms and Architectures. In: Proceedings of the UCC '14 the IEEE/ACM 7th International Conference on Utility and Cloud Computing, 2014. pp. 79-88.

[Alwabel et al., 2015] Alwabel A., Walters R., and Wills G., DesktopCloudSim: simulation of node failures in the cloud. In: proceedings of the Sixth International Conference on Cloud Coputing, GRIDs, and Virtualization (CLOUD COMPUTING 2015), 2015. pp. 14-19.

[Chen and Deelman, 2012] Chen W. and Deelman E., WorkflowSim: A toolkit for simulating scientific workflows in distributed environments. In: Proceedings of the E-SCIENCE '12 Proceedings of the 2012 IEEE 8th International Conference on E-Science (e-Science), 2012. pp. 1-8.

[Frey and Hasselbring, 2011] Frey S. and Hasselbring W., The CloudMIG Approach: Model-Based Migration of Software Systems to Cloud-Optimized Applications. International Journal on Advances in Software, Vol.4, Issue. 3/4, 2011. pp. 342-353.

[Cemim et al., 2012] Cemim P., Jersak L. C., Lopes G. A., Junior J. D. M., Ferreto T., EduCloud: A private cloud tool for academic environment. In: Proceedings of the 2012 IEEE Latin America Conference on Cloud Computing and Communications (LATINCLOUD), pp. 66-71, 2012.

Authors' Information

Khaled M. Khalil, e-mail: kmkmohamed@gmail.com,

M. Abdel-Aziz, e-mail: mhaziz@cis.asu.edu.eg,

Taymour T. Nazmy, e-mail: timor.mohammad@cis.asu.edu.eg,

Abdel-Badeeh M. Salem, e-mail: absalem@cis.asu.edu.eg

Faculty of Computer and Information Science Ain shams University Cairo, Egypt

APPROACH FOR CHOOSING OF ARCHITECTURE MODELS FOR DISTRIBUTED APPLICATIONS DESIGNING

Elena Chebanyuk, Mykhailo Kostiuk

Abstract: *Distributed system - a number of independent computers linked by a network [Oxford, 2016]. Distributed applications allow different devices to collaborate and simultaneously proceed data. Centrally, designing of such application require some skills and additional knowledge about all levels of architecture patterns.*

In this article terminology used for distributed application designing is systematized. Then approach for matching different types of architectures according to customer requirements is proposed. Schemas and stacks of technologies for different architectural solutions are represented. Also advantages and disadvantages for different architectural solution are represented. Analyzing them software architect can refine architectural solution and compose recommendations for further development.

Comparative cost analysis for implementing different architectural styles is represented. This analysis allows to estimate stakeholders efforts and cost of distribution software on client side.

In conclusion recommendations for implementing proposed approach are represented.

Keywords: *software architecture, distributed application, Web-Service, component, event processor, entity framework, Managed Extensibility Framework (MEF), PushSharp API Development, Amazon, Amazon DynamoDB, ElastiCache Redis, Amazon Simple Queue Service(SQS), Amazon Web Services, cloud computing.*

ITHEA Keywords*: D2 Software Engineering, D 2.0 Tools, D.2.10 – Design Topic, D.2.11 - Software Architectures.*

Introduction

Increasing role of information exchange today is a precondition of appearing many applications designed of client-server and n-tier architectural styles Application that are based on client-server architectural style are widely meet today. Growing amount of different types of devices are precondition for appearing application that need to execute part of task in one device and continue execution by means of another devices.

Cloud computing, necessity to create software, simultaneously proceeding data from different resources, is a precondition for creating distributed applications. From the other side growing amount of mobile devices and tablet PCs causes to designing of applications, aimed to information exchange between different data storages.

For example you create request on mobile phone application and perform it on server. Such companies as IBM and Microsoft [Microsoft, 2003] propose stack of technologies for designing and development of distributed applications. But peculiarities of customer tasks need to precise architectural solutions.

Main definitions and standards for distributed technologies designing

There are several OMG standards that define procedures and algorithms, performed in distributed applications. They are Distributed Simulation Systems Specification, Data Distribution Service (DDS). But definition of several terms is not presented in OMG standards and is systematized from practical usage of researches and stakeholders.

Table 1. Controlled vocabulary of problem domain "Definitions and standards for distributed technologies designing"

Term	Definition
Distributed system	A number of independent computers linked by a network [Oxford, 2016]. Collection of independent computers that appears to its users as a single coherent system (or) as a single system [Kangasharju, 2008].
Distributed applications	A **distributed application** utilizes the resources of multiple machines or at least multiple process spaces, by separating the application functionality into more manageable groups of tasks that can be deployed in a wide variety of configurations [Microsoft, 2012].
Components	1. an entity with discrete structure, such as an assembly or software module, within a system considered at a particular level of analysis. ISO/IEC 15026:1998, Information technology — System and software integrity levels.

Term	Definition
	3.1. 2. one of the parts that make up a system. IEEE Std 829-2008 IEEE Standard for Software and System Test Documentation. 3.1.6. 3. set of functional services in the software, which, when implemented, represents a well-defined set of functions and is distinguishable by a unique name. ISO/IEC 29881:2008, Information technology — Software and systems engineering — FiSMA Note 1 to entry: A component may be hardware or software and may be subdivided into other components. The terms "module," "component," and "unit" are often used interchangeably or defined to be subelements of one another in different ways depending upon the context. The relationship of these terms is not yet standardized. A component may or may not be independently managed from the end-user or administrator's point of view. [ISO/IEC/IEEE 24765:2010, 2010] A specific, named collection of features that can be described by component definition. [Unified Component Model for Distributed, RealTime and Embedded Systems, 2013]
Component standard	A standard that describes the characteristics of data or program components subdivided into other components. [Unified Component Model for Distributed, Real-Time And Embedded System, 2013].
Data Distribution Service (DDS) standard	Standard describes internal structures of classes and modules, that provide interaction between components. Also there are recommendations of organizing and performing such processes as exchange information between components and collaboration of objects by means of subscribing and publishing. Data, prepared for exchanging, are published. Other services that need such data subscribed to procedures of common data updating. After subscribing component or service listens port of other connection to obtain data for further processing. [Data Distribution Service (DDS), 2015]

Term	Definition
Service	Service is an application function packaged as a reusable component for use in a business process. It either provides information or facilitates a change to business data from one valid and consistent state to another. The process used to implement a particular service does not matter, as long as it responds to your commands and offers the quality of service you require. Through defined communication protocols, services can be invoked that stress interoperability and location transparency. A service has the appearance of a software component, in that it looks like a self-contained function from the service requester's perspective. However, the service implementation may actually involve many steps executed on different computers within one enterprise or on computers owned by a number of business partners. A service might or might not be a component in the sense of encapsulated software. Like a class object, the requester application is capable of treating the service as one. Web services are based on invocation using SOAP messages which are described using WSDL over a standard protocol such as HTTP. Use of web services is a best practice when communicating with external business partners [IBM, 2017].
Data-centric publish subscribe (DCPS) mechanism	Data-centric publish subscribe (DCPS) mechanism defines the functionality used by an application to publish and subscribe to the values of data objects. It allows: • Publishing applications to identify the data objects they intend to publish, and then provide values for these objects. • Subscribing applications to identify which data objects they are interested in, and then access their data values. • Applications to define topics, to attach type information to the topics, to create publisher and subscriber entities, to attach QoS policies to all these entities and, in summary, to make all these entities operate. Listeners and conditions (in conjunction with wait-sets) are two alternative mechanisms that allow the application to be made aware of changes in the DCPS communication status [IBM c), 2015].

Term	Definition
Architectural style	A family of systems in terms of a pattern of structural organization. Definition two: a characterization of a family of systems that are related by sharing structural and semantic properties [ISO/IEC/IEEE 24765:2010, 2010].
Architecture	Fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution. According to standard ISO/IEC 15288:2008 (IEEE Std 15288- 2008), Systems and software engineering architecture is the organizational structure of a system or component. Syn: architectural structure [ISO/IEC/IEEE 24765:2010, 2010]
Unified Component Model for Distributed, Real-Time And Embedded Systems	Standard Unified Component Model for Distributed, Real-Time And Embedded Systems defines necessary procedures for communication between components. Structure of distributed software system is represented by means of packages. Data exchanging are performed by special contracts. [Data Distribution Service (DDS), 2015] The contract package holds the definitions of contracts for UCM applications. Contracts mainly cover the definitions of interfaces and data types and package that defines a meta-model for standard data types. The contract package gathers several classes. A set of standard data types is defined; it is also possible to create metamodel extensions in order to define additional data types. Constants define specific values for a declared data type. Interfaces define consistent sets of methods related to a given service. The contract package also provides mechanisms to support the characterization of business and platform elements, using annotations and configuration parameters. Among those declarations, only data types and interfaces are considered as types and can be used to specify interactions between components. Constants and

Term	Definition
	exceptions are used to enrich the domain application specifications but do not directly contribute as the definition of contracts of interaction between components. Annotations and configuration parameters are used to decorate declarations. [Data Distribution Service (DDS), 2015]
Client	the code or process that invokes an operation on an object. ISO/IEC 19500-2:2003, Information technology — Open Distributed Processing — Part 2: General Inter-ORB Protocol (GIOP)/Internet Inter-ORB Protocol (IIOP). a node, cluster or capsule, which: a) contains a basic engineering object corresponding to a computational client object; and b) contains, or is potentially capable of containing, stub, binder and protocol objects in a channel supporting operations involving the client object. (ISO/IEC 14752:2000, Information technology) [Data Distribution Service (DDS), 2015]
Event	Occurrence of a particular set of circumstances. According to ISO/IEC 16085:2006 (IEEE Std 16085-2006), Systems and software engineering event is an external or internal stimulus used for synchronization purposes [ISO/IEC/IEEE 24765:2010, 2010].
Event synchronization	Control of task activation by means of signals. NOTE Three types of event synchronization are possible: external interrupts, timer expiration, and internal signals from other tasks. [ISO/IEC/IEEE 24765:2010, 2010]
Event-sequencing logic	a description of how a task responds to each of its message or event inputs. A communication sent from one object to another. IEEE Std 1320.2-1998 (R2004) IEEE Standard for Conceptual Modeling Language Syntax and Semantics for IDEF1X97 (IDEFobject) [ISO/IEC/IEEE 24765:2010, 2010].
Message	a communication sent from one object to another. IEEE Std 1320.2-1998 (R2004)

Term	Definition
	IEEE Standard for Conceptual Modeling Language Syntax and Semantics for IDEF1X97 (IDEFobject). [ISO/IEC/IEEE 24765:2010, 2010]
Metamodel	a logical information model that specifies the modeling elements used within another (or the same) modeling notation. IEEE Std 1175.1-2002 (R2007) IEEE Guide for CASE Tool Interconnections — Classification and Description. a metamodel Vm for a subset of IDEFobject is a view of the constructs in the subset that is expressed using those constructs such that there exists a valid instance of Vm that is a description of Vm itself. (IEEE Std 1320.2-1998 (R2004) IEEE Standard for Conceptual Modeling Language Syntax and Semantics for IDEF1X97 (IDEFobject)). a model containing detailed definitions of the meta-entities, meta-relationships and meta-attributes whose instances appear in the model section of a CDIF transfer. ISO/IEC 15474-1:2002, Information technology specification of the concepts, relationships and rules that are used to define a methodology. ISO/IEC 24744:2007, Software Engineering — Metamodel for Development Methodologies.3.4. Syn: meta-model [ISO/IEC/IEEE 24765:2010, 2010]

Related papers

Paper [Sharmaa, 2015] describes main architectural styles, that are used for designing of distributed applications. It but lacks from recommendations in which situations which architectural style will be applied.

Paper [Ciuffoletti, 2015] proposes a foundation for choosing of architectural styles for distributed systems.

Usually for designing distributed applications combination of architectural styles is used.

Layered architectural style should be used in application which can be divided into several layers on the basis of services or functionalities in such a way that lower layer forms the basis for upper layer.

Division of system into layers leads to easy debugging and flexible to update and maintain in future. Cloud computing architecture is for cloud based applications like Gmail drive, Google App Engine. It has limited functionality since till now it has not covered all the requirements of information technology. Higher scalability is the main feature of this architecture [Sharmaa, 2015].

Use of Internet is increasing day by day results in heavily used of client server architectural styles in the web based applications. Some examples of web based software are: AceProject, Ganttic, Celoxis etc. For a server based application supporting many clients which is going to be used by a web browser uses client server architecture model. It is also applicable in application area for centralized resource system (like management functions, storage) which is going to be used by many clients. Heavily used and famous example for client server architecture application area is email, World Wide Web, FTP software, e-commerce applications. The complexity of such applications is not as high as blackboard based applications. Client server based applications keeps high functionality and efficiency. 3-tier client server applications have good reliability because of middle tier which perform all security related task [Sharmaa, 2015].

Multi-tier architectural style – is used to distribute application functions on different physical components.

In more simple variant client-server architecture is used. It is used when it is enough to collaborate data exchanging mechanism. Often client-server architecture is a part of n-tire architectural style.

Service oriented architecture – is used to simplify code reuse procedure and when it is necessary to exchange data thought central component, namely service bus.

Paper [Ciuffoletti, 2015] proposes two different approaches to virtualization:

• a container-based approach evolves is grounded on using of separate resources for performing some operation. Also during execution services use isolated memory and computation resources. Other approach, namely hypervisor technique provides performing software functions by means of memory and resources of operating system. Authors make a conclusion that the realization of complex but agile distributed architectures, composed of small and specialized services: the microservice approach is a promising design paradigm that is tightly bound to (or merging with) the container technology.

• another uses the lower layers of the running operating system to implement one or more containers, that are isolated environments.

Task

To design an approach for distributed application architecture choosing. In order to do this it is necessary to do the following:

- systematize both functional and non-functional requirements for designing distributed application;.
- propose architectural solutions of notification engine considering different user requests;
- define a stack of technologies for supporting every architectural solution;
- estimate cost of various architectural solutions implementing;
- propose an approach for architecture of distributed applications choosing.

Summary of functional and non-functional features of designed system

Functional features

The system should provide high level stability during high load on the servers, provide scalability for the solution (based on the number of records in the consumer table of the GMB market DB, number of active markets, and number of active events per market), message durability and ability to re-execute the job even if the job failed.

The system should provide the following main functionality:

- ability to add new triggers for events;
- ability to start/stop triggers for events;
- ability to remove triggers;
- ability to update trigger's configuration;
- ability to generate messages for job execution;
- ability to write messages to queue (send request);
- ability read messages from queue (receive request);
- ability to execute job according to the message data;
- ability to send e-mail notifications;
- ability to send push notifications;
- ability to generate DB notifications.

The main notification engine execution flow should consist of next steps:

- scheduler should initiate trigger work;
- trigger process its internal logic;
- after processing, trigger should generate necessary message for handler;
- generated messages should be pushed to the dedicated queue (name of the dedicated queue should be configured);
- the queue stores the message until it will be read and processed by the handler;
- handler each time monitors the queue for new available messages and reads them;
- after message was read, handler process message according to the message data;
- after successful processing, handler should remove the message.

Non-functional features

In addition to the main functional requirements, the system should provide additional non-functional requirements that contains:

- as the whole GMB system deployed on the Amazon Web Services (AWS), NE should be also deployed on this environment;
- the system should be possible to be moved to another hosting environment, without providing any changes for the existing functionality;
- NE should be independent part of the GMB;
- NE should not influence business logic of the main part of the GMB (backend services).

Distributed system design

At first need to be noted, that NE will be designed as a distributed application. So the system will consist of two parts. The first part will be called Event Processor (EP), and the second one will be called Job Processor (JP). According to the provided functional and non-functional requirements, four different application architectures were designed.

The first architectural solution

Based on the data about the system in general and architectural patterns NE will consist of two separated parts: Event Processor (EP) and Job Processor (JP).

Main definition of Events, Event processing architecture, Event source and event comsumers are represented at [IBM c), 2015].

Principles of organizing job processor are represented on [IBM d) 2015]. It is proposed to design EP is designed as a simple service. JP is designed as a web service, and will have specific URL. This URL will be the same for all markets.

For each market and for each application in the market, which will use NE, event processor will load application plugin. Plugins will be connected to the EP using Managed Extensibility Framework. Each plugin consists of a number of triggers, which will execute specific events (jobs) every 15 seconds (time will be configured in the plugin application configuration file and could be changed for a specific request from the client side).

Each trigger could access to the DB through a separate library called Framework Module. Framework Module – is a custom library, designed especially for code reusing in different modules of the GMB system. It encapsulates DB models (for both market and administrative DBs), generated by Entity Framework (EF), specific constants, general helper methods, methods to send HTTP requests etc. All the data will be stored in the MS SQL Server DBMS (because GMB already has all data in this DBMS).

After successful execution of the trigger, if it is necessary, trigger will send result to EP in order to initiate EP to send HTTP request with necessary information to JP, for further data processing (send push or e-mail notification, or generate DB notification). JP processes request from the EP and sends push notifications through Push Sharp API, sends e-mail notifications through simple mail client, and stores DB notifications in the

The design of the notification engine architecture is shown on the Figure 1.Architectural features

In order to provide updates of the trigger configuration, NE should be restarted at all. Such realization does not support “on flight” reconfiguration of the triggers. All of them are independent from each other, but if the EP fails, all triggers will not work at all. In order to start them working again, the restart of the service is needed.

Figure 2. shows EP functional requirements, and Figure 3 shows JP functional requirements shows covering of the functional requirements.

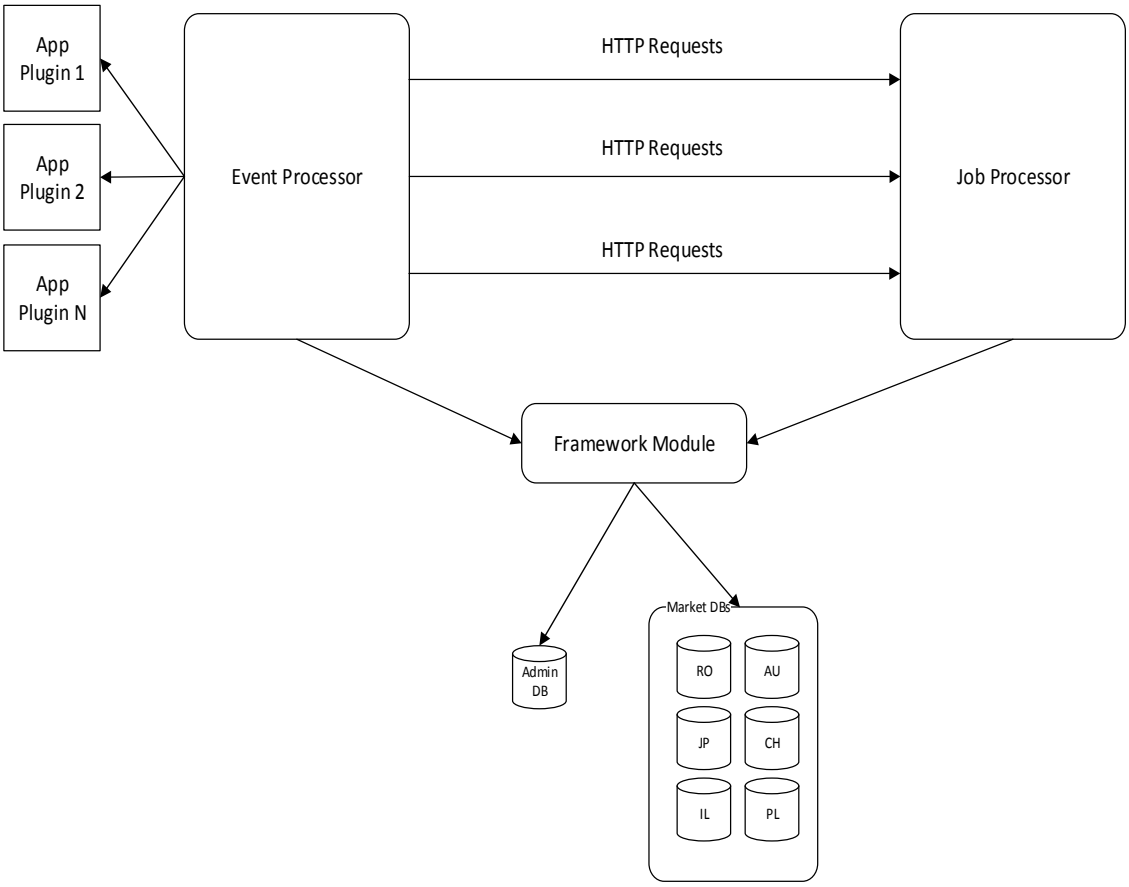


Fig. 1. The first architectural solution of notification engine

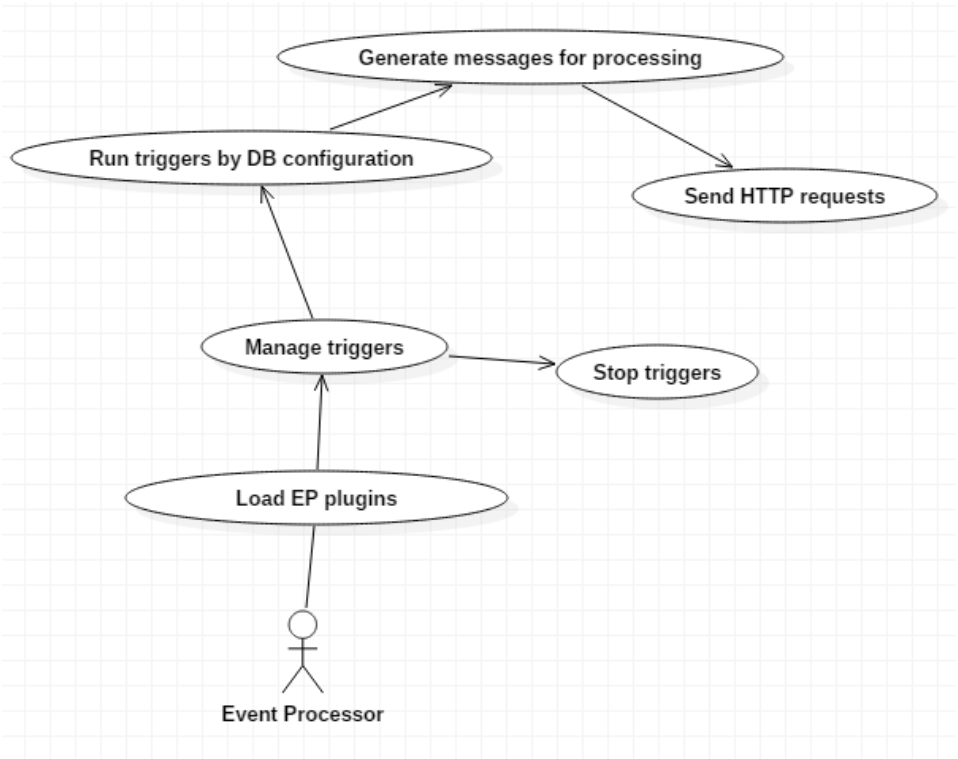


Fig. 2. Use case diagram of event processor

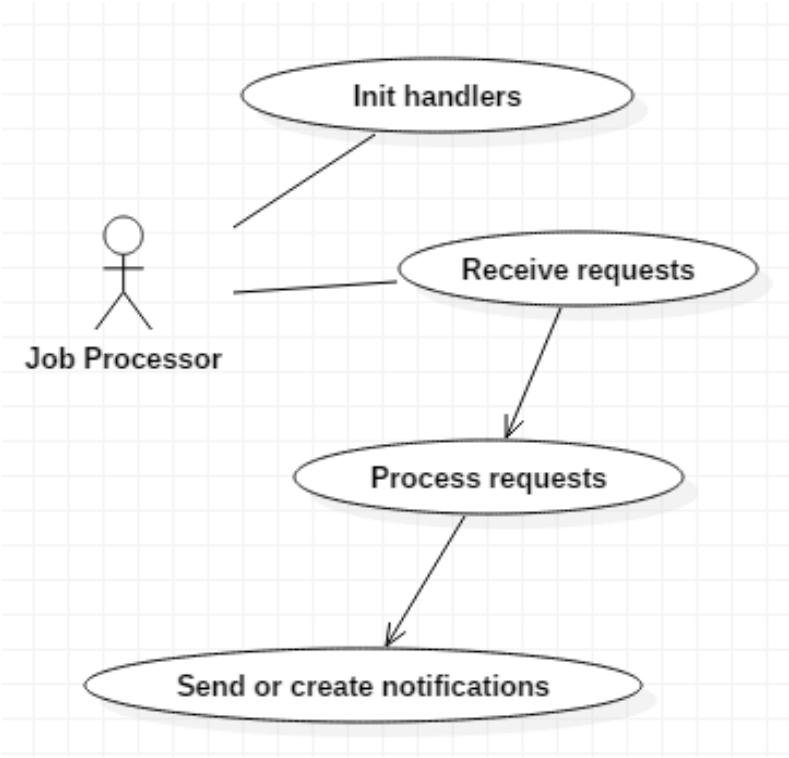


Fig. 3. Use case diagram of job processor

Stack of technologies for the first architectural solution

1. Windows service, used as a core principle of the EP creation.
2. Web service, used as a core principle of the JP creation.
3. Managed Extensibility Framework (MEF), used by EP to load plugins.
4. Entity Framework (EF), used by framework module, to provide access to the market and administrative DBs.
5. Hypertext Transfer Protocol (HTTP), used to send requests from EP to JP, for further processing.
6. PushSharp API used to send push notifications.

Advantages and disadvantages of the first architecture

Advantages of the designed architecture:

– The system could be migrated from one hosting environment to another without any changes in the source code of the NE.

- As JP is a Web service, it could be deployed using auto scaling technologies, just routing service need to be added (it also does not need any changes in the source code).
- The system could be deployed in different hosting environments (for example: EP on the Microsoft Azure, and JP on the Amazon AWS).

Disadvantages of the designed architecture:

- JP is a Web service, so it needs additional payment for domain address and external IP address.
- HTTP requests is not good solution for communication between EP and JP because in some moments of high load EP could generate so many requests, so that JP (IIS service on the Windows OS) will break down due to overload (the number of maximum parallel threads and connections is limited by the IIS). This case does not provide message durability.
- As this architecture supposes main data processing on the EP side, and JP just executes sending of the notifications leads to that point that all main functionality locates on the EP and its plugins. As all triggers process data in asynchronous way, during the high load on the system, triggers will have long timeouts for connections to the DB or external services. It caused because the number of threads is limited in the machine, on which the service is deployed.
- In order to update configuration of the triggers, NE must be restarted. Reload is a very bad case for NE, because EP need to reload all plugins for all markets and this process is rather long. The restart time for a Windows service is approximately from 3 to 10 minutes (based on the number of plugins).

The second architectural solution

The second version of the architecture of the NE is more complex, than the first one, and it has its own peculiarities.

EP is designed as a number of independent event generators (lambda functions). They are designed using AWS Lambda. Each trigger in the EP will be presented as a Lambda function. JP is designed as a windows service with its plugins.

Each AWS Lambda function starts by scheduler. Each scheduler has its own configuration (configuration stored in the AWS profile for each lambda function), which could be changed from the AWS management console.

The communication between AWS Lambda and JP will be provided through the message bus. The message bus will be durable – if any issues arise, messages will be retained and processed when services are back online. The AWS Simple Queue Service (SQS) will be used as message-bus service. Each Lambda function will send messages to the dedicated queue (names of queues can be configured from AWS management console).

JP will listen all the available queues in the SQS and read messages from them. For each market and for each application in the market, which will use NE, job processor will load application plugin. Plugins will be connected to the JP using MEF. Each plugin consists of a number of handlers, which will execute specific jobs according to the data, provided in the message.

The JP has one job plugin for each application and one default plugin. The design pattern command will be used to execute dedicated functionality for each job. In order to provide command pattern functionality a special Dispatcher class was designed in the JP. Dispatcher task is to subscribe to a dedicated queue, read message from the queue, and pass this messages to a plugin for further execution. Per one queue will exist only one dispatcher instance. Each dispatcher works in its own thread.

DB storage will be moved from MS SQL Server DBMS to the AWS Dynamo DB. Dynamo DB will significantly increase the speed for the read and write requests, because it is NoSQL, object oriented database. Administrative DB will be removed from the system at all, instead of it, a new table with available markets will be created independently in the Dynamo DB storage.

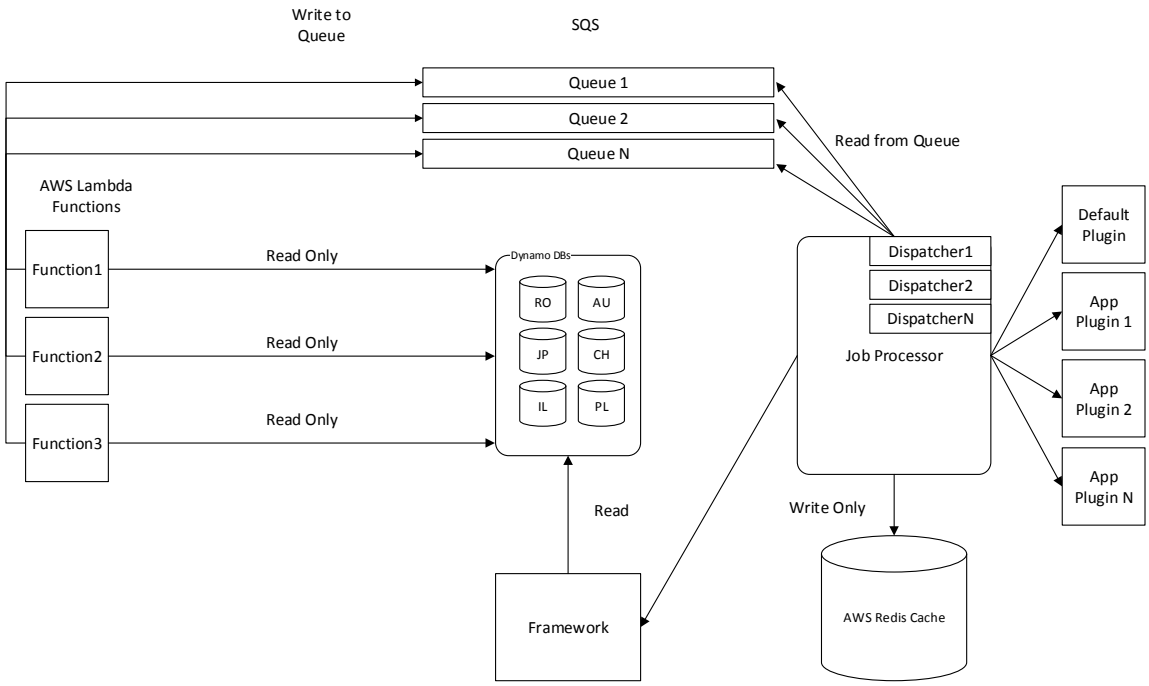


Fig. 4. The second architectural solution of notification engine

AWS Lambda supports direct access from the function code to the AWS SQS and Dynamo DB, so any new framework does not need, as a result, the creation of the new lambda function will not be a problem.

Each handler in JP could access to the DB through a separate library called Framework Module, but only for the read operation. Framework module need to be changed in order to support access to a new DB instance. In order to provide access to the AWS DB and SQS services, AWS Toolkit need to be installed. In addition, direct access to the Dynamo DB from the framework module will be provided through Linq2DynamoDB framework (analog of EF). After successful execution of the handler in JP plugin, if it is necessary, handler sends push or e-mail notification, or generate DB notification, or just stops. Push notifications will be sent through Push Sharp API, e-mail notifications will be sent through simple mail client, and DB notifications will be saved in the ElastiCache Redis, which provided by AWS. As DB notifications need to be accessible very fast (request timeout no more than 100ms for a good 100 Mbit Internet connection), was suggested to use the fastest object oriented data storage – Redis Cache. The design of the notification engine architecture is shown on the Figure 4.

In order to provide updates of the trigger configuration, administrator of the system should just change event data for each Lambda function. All necessary things for updating this data are available in the AWS Management Console. Such realization supports “on flight” reconfiguration of the triggers. All of them are independent from each other.

JP handlers are always loaded into the system, and they don’t need to be reloaded. Handlers task is to execute job according to the message data, they don’t depend on any configuration. If JP service fails, all the data which was not processed will stay in queues, and in order to restore the work of the service, it should be restarted at all.

The next use-case diagrams. Figure 5 shows EP functional requirements, and Figure 6 shows JP functional requirements) show covering of the functional requirements.

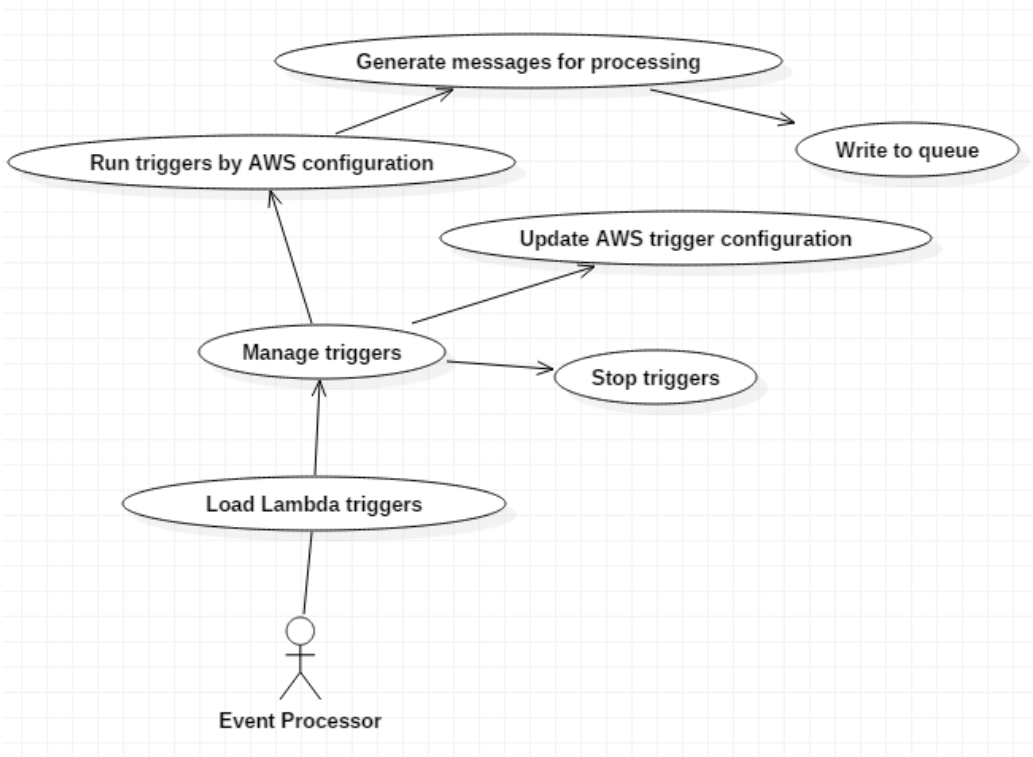


Fig. 5. Event Processor use case diagram

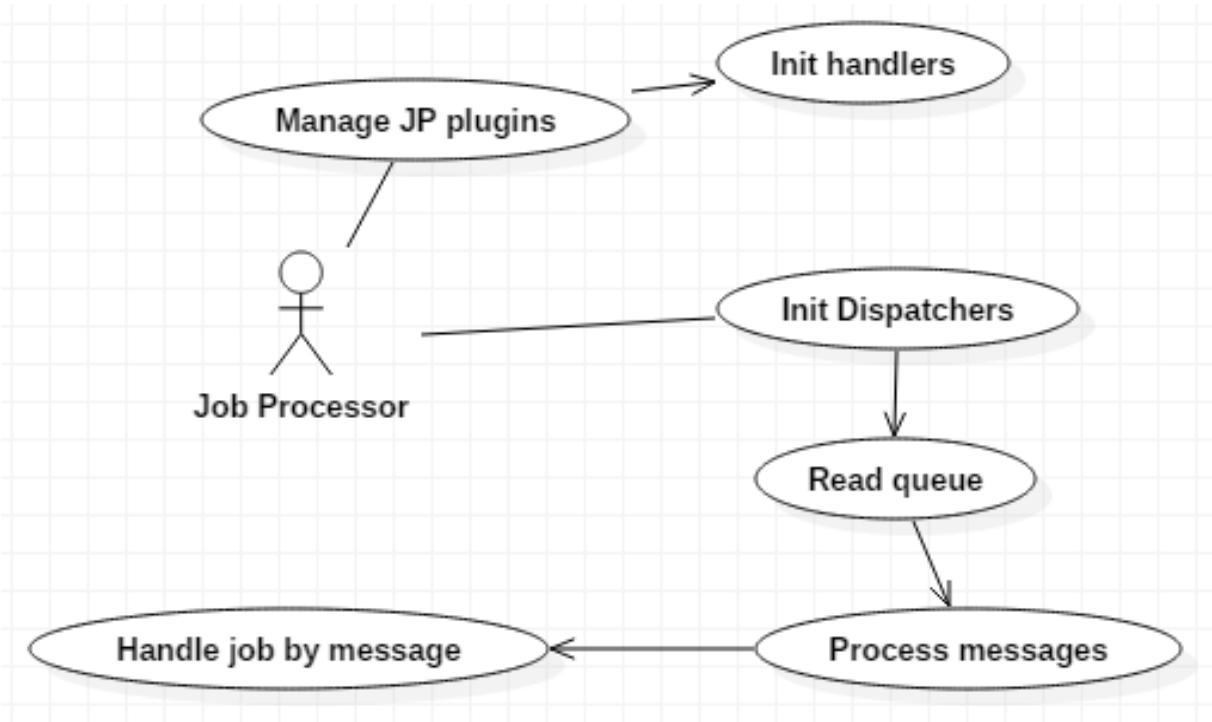


Fig. 6. Job Processor use case diagram

Stack of technologies for realization of the second architectural solution

Stack of technologies, which will be used for this architecture:

- AWS Lambda is used as EP triggers.
- Windows service is used as a core principle of the JP creation.
- Managed Extensibility Framework (MEF), is used by EP to load plugins.
- AWS SQS is used as a message bus between EP and JP.
- AWS Dynamo DB is used as main data storage.
- Linq2DynamoDB (analog of EF), is used by framework module, to provide access to the market DBs.
- Command pattern is used for JP to provide queue reading.
- PushSharp API is used to send push notifications.
- AWS ElastiCache Redis is used to store DB notifications.

Advantages and disadvantages of the second architectural solution

Advantages of the designed architecture:

- EP part is designed as independent functions on the AWS side. It means that AWS will scale up or scale down these functions according to the load, provided on them. It is fully independent part of the NE system.
- The message bus is supported by using SQS. SQS is also independent part of the NE, and it also stores on the AWS side. SQS guarantees, that a message will be delivered at least one time, so it provides durability of the communication. Also AWS will scale up or scale down SQS based on the load provided on it.
- Usage of Dynamo DB instead of SQL Server DB is also a great advantage. NoSQL data storage will provide speed up inserting and key-search mechanisms. As a result, access to the data and data saving will be faster, so EP and JP will have lower timeouts for DB connections.
- Usage of the Redis cache will speed up GMB access to DB notifications, because Redis always stores all data in the RAM. Also inserting process also will be faster, because application does not need to wait while the record will be stored on the file system.
- Updates for trigger's configuration will apply without any restarts of the services. This is possible because configuration is needed only for triggers on the EP side, and AWS Lambda supports real-time reconfiguring.

– In order to add new trigger, administrator should just create new lambda function in the AWS management console and provide correct configuration for this function. This operation does not need any restart for other lambda functions or JP service.

Disadvantages of the designed architecture:

– The biggest disadvantage is that using AWS Lambda, SQS, Dynamo DB, and Redis will limit NE on deployment environments. It means that, if NE migrates to another hosting place (for example Microsoft Azure) a lot of functionality will not work at all. But system could be divided by different hosting places (for example, EP with Lambda, SQS, Dynamo DB, and Redis will still work on AWS, and JP will work on Azure). In this case the problem could be in the timeouts during connections between system parts, and it is also bad case for CRM system. As a result, time of execution of the particular function could be increased in 10-15 times.
– AWS Lambda has limitations for time of execution and for the number of executions. It means that customer should additionally pay for the number of executions and if the execution time of the function became longer than available, function will fail and execution will not provide any result.
– Usage of a number of external (in our case AWS resources) could lead to the problems of management of the system at all. Because configuration of each lambda function is stored in different places, configuration of the SQS is stored in another place, etc.
– In order to use Dynamo DB, firstly, for each environment old data need to be migrated from MS SQL Server to Dynamo DB. This process needs additional time for realization.
– Dynamo DB is a very fast data storage, but it has limitations for provision throughputs (number of units for read and write per second). If these limitations will be exceeded, the application, which sends requests to the DB will receive error: provision throughput exceeded. As a result, data processing will stop for a period of time. Throughput can be changed, but it is rather expensive.

The third architectural solution

The third version of the architecture of the NE is absolutely based on the AWS components, and differs from the first architectures by absence of custom services. Such kind of architecture has its own peculiarities.

EP is designed as a number of independent event generators (lambda functions). They are designed using AWS Lambda. Each trigger in the EP will be presented as a Lambda function.

JP is also designed as a number of independent lambda functions (handlers). Each handler in the EP will be presented as a Lambda function.

Each AWS Lambda function starts by scheduler. Each scheduler has its own configuration (configuration stored in the AWS profile for each lambda function), which could be changed from the AWS management console.

The communication between AWS Lambda and JP will be provided through the message bus. The message bus will be durable – if any issues arise, messages will be retained and processed when services are back online. The AWS Simple Queue Service (SQS) will be used as message-bus service. Each Lambda function will send messages to the dedicated queue (names of queues can be configured from AWS management console).

Each JP handler will listen dedicated queue in the SQS and read messages from it. For each market and for each application in the market, which will use NE, JP will have lambda handlers. Each handler will execute specific jobs according to the data, provided in the message.

DB storage will be moved from MS SQL Server DBMS to the AWS Dynamo DB. Dynamo DB will significantly increase the speed for the read and write requests, because it is NoSQL, object oriented database. Administrative DB will be removed from the system at all, instead of it, a new table with available markets will be created independently in the Dynamo DB storage.

AWS Lambda supports direct access from the function code to the AWS SQS and Dynamo DB, so any new framework does not need, as a result, the creation of the new lambda function will not be a problem.

Also EP will have special lambda function(s) which will have access to AWS Simple Notification Service (SNS) in order to provide direct access for sending push and e-mail notifications. DB notifications will be sent by EP trigger to the SQS as a message, where it will be read by handler later. Handler will process this message and DB notifications will be saved in the ElastiCache Redis, which provided by AWS.

As DB notifications need to be accessible very fast (request timeout no more than 100ms for a good 100 Mbit Internet connection), was suggested to use the fastest object oriented data storage – Redis Cache.

The design of the notification engine architecture is shown on the Figure 7.

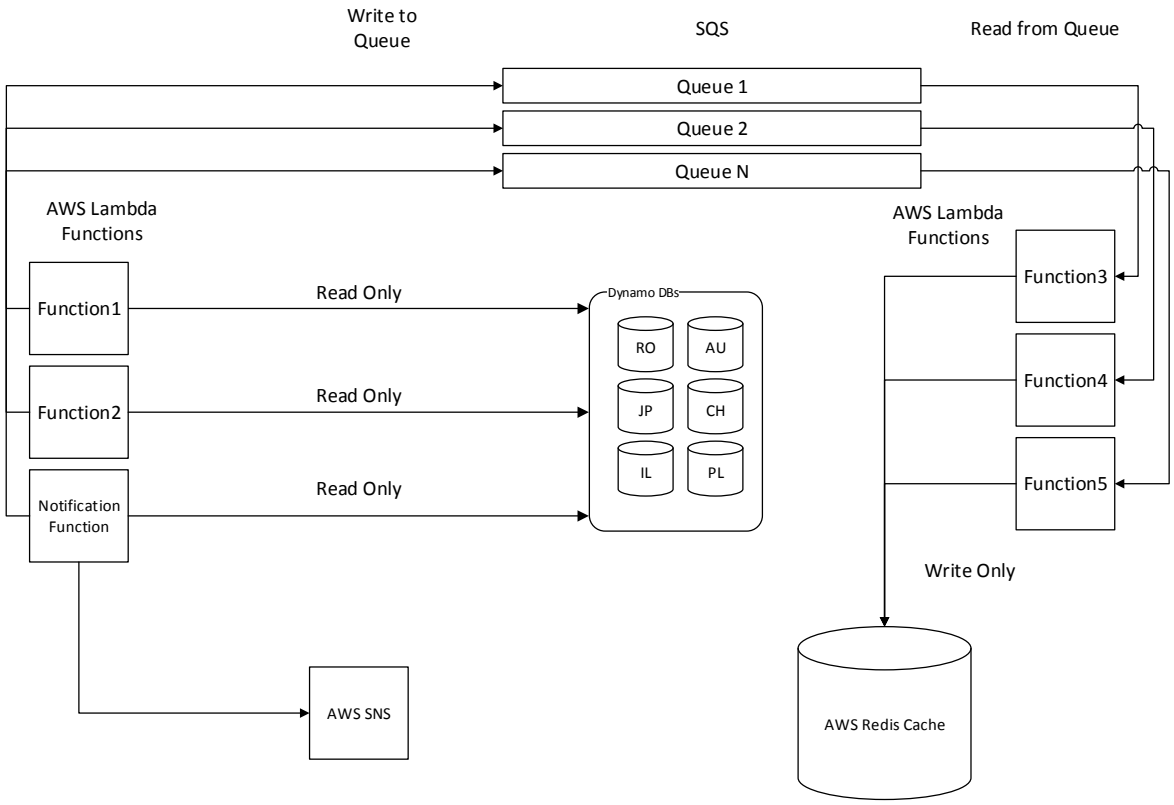


Fig. 7 The third architectural solution of notification engine

Architectural features

Stack of technologies

Stack of technologies, which will be used for this architecture:

1. AWS Lambda is used as EP triggers and JP handlers.
2. AWS SQS is used as a message bus between EP and JP.
3. AWS Dynamo DB is used as main data storage.
4. AWS SNS is used to send push and e-mail notifications.
5. AWS ElastiCache Redis is used to store DB notifications.

Advantages and disadvantages of the architecture

Advantages of the designed architecture:

- The greatest advantage of such architecture is that all parts of the NE are designed in such way, that they don't depend on physical deployment environments. All parts of the system (EP, JP, SQS, DynamoDB, Redis cache) are auto scalable and supported by AWS. This architecture provides the best solution for highly loaded CRM systems.

- EP an JP part is designed as independent functions on the AWS side. It means that AWS will scale up or scale down these functions according to the load, provided on them. It is fully independent part of the NE system.

- The message bus is supported by using SQS. SQS is also independent part of the NE, and it also stores on the AWS side. SQS guarantees, that a message will be delivered at least one time, so it provides durability of the communication. Also AWS will scale up or scale down SQS based on the load provided on it.

- Usage of Dynamo DB instead of SQL Server DB is also a great advantage. NoSQL data storage will provide speed up inserting and key-search mechanisms. As a result, access to the data and data saving will be faster, so EP and JP will have lower timeouts for DB connections.
- Usage of the Redis cache will speed up GMB access to DB notifications, because Redis always stores all data in the RAM. Also inserting process also will be faster, because application does not need to wait while the record will be stored on the file system.

- Updates for trigger's configuration will apply without any restarts of the services. This is possible because configuration is needed only for triggers on the EP side, and AWS Lambda supports real-time reconfiguring.

- In order to add new trigger or handler, administrator should just create new lambda function in the AWS management console and provide correct configuration for this function. This operation does not need any restart for other lambda functions or JP service.

Disadvantages of the designed architecture:

– The biggest disadvantage is that this architecture absolutely depends on the resources of the Amazon services. If the GMB migrates to another hosting, NE system will not work with GMB at all. Of course if GMB migrate to another big vendor of cloud computing (such as Microsoft Azure), the NE also should be modified in order to support the same functionality, but on the side of Azure. For example, Azure has its own cloud functions (analog of AWS Lambda), Azure table storage (analog of AWS Dynamo DB) etc.

– Usage of a number of external (in our case AWS resources) could lead to the problems of management of the system at all. Because configuration of each lambda function is stored in different places, configuration of the SQS is stored in another place, etc.

– In order to use Dynamo DB, firstly, for each environment old data need to be migrated from MS SQL Server to Dynamo DB. This process needs additional time for realization.

– Dynamo DB is a very fast data storage, but it has limitations for provision throughputs (number of units for read and write per second). If these limitations will be exceeded, the application, which sends requests to the DB will receive error: provision throughput exceeded. As a result, data processing will stop for a period of time. Throughput can be changed, but it is rather expensive.

Estimation of distributed application development

In order to estimate the cost of implementing of proposed architectural solutions refer to PNN Soft company practices of counting of economical justifications for different architectural solutions realization. The data for the costs of the deployment environment where provided by Amazon AWS, for EU west-1 data center for Windows OS.[]

The first architectural solution

The table below shows necessary resources, needed to develop and deploy the notification engine:

Table 2. Economical justification for the first architectural solution

№	Name	Number of needed resources/prices	Costs	Total
1	Developers	3 developers	25 ($/h)	75($/h)
2	QA Engineers	2 QAs	20 ($/h)	40 ($/h)
3	Development time	120 hours	75($/h)	9000$
4	Time for testing	40 hours	40 ($/h)	1600$
5	AWS dev environment	160 hours, 2 machines	0.034 ($/h)	10.88$
6	AWS staging environment	160 hours, 2 machines	0.034 ($/h)	10.88$
7	AWS QA environment	40 hours, 2 machines	0.258 ($/h)	20.64$
8	AWS prod environment	20 hours, 2 machines	0.517 ($/h)	20.68$
9	Domain address	160 hours, 4 items	13 ($/year)	3.5$

Total number of team members for NE is: 5 people (3 developers, 2 QA engineers).

Total time for the development and testing is: 160 hours.

Total price of NE in current approach is: 10 666.58$.

Table 2. Economical justification for the second architectural solution

№	Name	Number of needed resources/Prices	Costs	Total
1	2	3	4	5
1	Developers	3 developers	25 ($/h)	75($/h)
2	QA Engineers	2 QAs	20 ($/h)	40 ($/h)
1	2	3	4	5
3	Development time	130 hours	75($/h)	9750$
4	Time for testing	40 hours	40 ($/h)	1600$
5	AWS dev environment	170 hours, 1 machine	0.034 ($/h)	5.44$
6	AWS staging environment	170 hours, 1 machine	0.034 ($/h)	5.44$
7	AWS QA environment	30 hours, 1 machine	0.258 ($/h)	10.32$
8	AWS prod environment	20 hours, 1 machine	0.517 ($/h)	10.32$
9	AWS SQS	3000000 requests, 4 queues	0.00000040 ($/request)	14.4$
10	ElastiCache Redis	170 hours, 4 instances	$0.638	433.84$
11	DynamoDB tables	170 hours, 20 tables (for 1 environment), 4 environments	0.0145 ($/h) 0.0029 ($/h)	236.64$
12	Lambda functions	10 functions (for 1 environment), 4 environments	$9.13 ($/month)	365.2$

Total number of team members for NE is: 5 people (3 developers, 2 QA engineers).

Total time for the development and testing is: 170 hours.

Total price of NE in current approach is: 12431.6$.

The data for the development and testing costs were provided by software development company “PNN Soft”.

The data for the costs of the deployment environment where provided by Amazon AWS, for EU west-1 data center for Windows OS.

Table 4. Economical justification for the third architectural solution

№	Name	Number of needed resources/Prices	Costs	Total
1	2	3	4	5
1	Developers	2 developers	25 ($/h)	50($/h)
2	QA Engineers	2 QAs	20 ($/h)	40 ($/h)
1	2	3	4	5
3	Development time	90 hours	50($/h)	4500$
4	Time for testing	30 hours	40 ($/h)	1200$
5	AWS SQS	3000000 requests, 4 queues	0.00000040 ($/request)	14.4$
6	ElastiCache Redis	120 hours, 4 instances	$0.638	306.24$
7	DynamoDB tables	120 hours, 20 tables (for 1 environment), 4 environments	0.0145 ($/h) 0.0029 ($/h)	167.04$
8	Lambda functions	20 functions (for 1 environment), 4 environments	$9.13 ($/month)	730.4$

Total number of team members for NE is: 4 people (2 developers, 2 QA engineers).

Total time for the development and testing is: 120 hours.

Total price of NE in current approach is: 6 918.08$.

The data for the development and testing costs were provided by software development company “PNN Soft”. The data for the costs of the deployment environment where provided by Amazon AWS, for EU west-1 data center for Windows OS.

Conclusion

Approach for choosing architectural solution for distributed application designing is proposed in this paper. Systematization of terminology allows to organize collaboration between software architectures and researches in sphere of facilitation software components and services reuse. Also typical requirements for distributed application tasks are systematized. Advantages and disadvantages of different architectural solutions let estimate limitations of and effectiveness of future software. Functional and non-functional requirements for distributed applications allow to refine software requirement specification of future software. Then according to project budget it is defined if proper architectural solution for future project. Proposed advantages and disadvantages for every architectural solution let estimate cost of supporting and deployment processes.

Further works

Design a formal method for defining constrains from text of requirement specification. Then gather collections of features identifying every architectural solution. Prepare profile for problem domain “distributed application architectures”. Estimate this profile by following SOLID design principles using the approach, proposed in[].

Using this profile and constrains propose a method for requirement specification elicitation.

Bibliography

[Sharmaa, Kumarb, Agarwal, 2015] Anubha Sharmaa, Manoj Kumarb ,Sonali Agarwal A Complete Survey on Software Architectural Styles and Patterns Proceedings of the 4th International Conference on Eco-friendly Computing and Communication Systems. Published by Elsevier Volume 70, 2015, pp. 16–28 doi: 10.1016/j.procs.2015.10.019

http://ac.els-cdn.com/S187705091503183X/1-s2.0-S187705091503183X-main.pdf?_tid=9be93464-dc31-11e6-aded-00000aacb35f&acdnat=1484601810_4a7b6416ce69969da4b27cf755c2dfa3

[Chebanyuk and Markov, 2016] Elena Chebanyuk, Krassimir Markov An approach to class diagram verification according to SOLID design principles. In: MODELSWARD 2016, Proceedings of the 4th International Conference on Model-Driven Engineering and Software Development. Edited by S.

Hammoudi, L.F. Pires, B. Selic and P. Desfray. SCITEPRESS – Science and Technology Publications, Lda. Portugal, 2016. ISBN: 978-989-758-168-7. pp. 435-441.

DOI: 10.5220/0005830104350441

[Ciuffoletti, 2015] Augusto Ciuffoletti Automated deployment of a microservice-based monitoring infrastructure rs. Procedia Computer Science. Published by Elsiver. Volume 68 (2015) pp. 163 – 172 doi: 10.1016/j.procs.2015.09.232

http://ac.els-cdn.com/S187705091503077X/1-s2.0-S187705091503077X-main.pdf?_tid=937c55e2-dc2f-11e6-8f44-00000aab0f01&acdnat=1484600937_7d6e20f713eac3e85c067f9964396752

[Data Distribution Service (DDS), 2015] Data Distribution Service (DDS)

http://www.omg.org/spec/DDS/1.4/

[IBM a), 2015] IBM Knowledge Center

http://www.ibm.com/support/knowledgecenter/SSGMCP_4.2.0/com.ibm.cics.ts.eventprocessing.doc/concepts/dfhep_definition.html

[IBM b), 2015] IBM Knowledge Center

http://www.ibm.com/support/knowledgecenter/en/SSETD4_9.1.3/lsf_admin/job_processor_binding.html

[IBM c), 2015] IBM Knowledge Center

http://www.ibm.com/support/knowledgecenter/SSGMCP_4.2.0/com.ibm.cics.ts.eventprocessing.doc/concepts/dfhep_definition.html

[IBM d), 2015] IBM Knowledge Center

http://www.ibm.com/support/knowledgecenter/en/SSETD4_9.1.3/lsf_admin/job_processor_binding.html

[IBM, 2017] Service-Oriented Architecture expands the vision of web services, Part 1

http://www.ibm.com/developerworks/webservices/library/ws-soaintro/ws-soaintro-pdf.pdf

[IEEE, 1998] IEEE Standard for Conceptual Modeling Language - Syntax and Semantics for IDEF1X97

https://standards.ieee.org/findstds/standard/1320.2-1998.html

[ISO/IEC 16085:2006, 2006] ISO/IEC 16085:2006 Systems and software engineering -- Life cycle processes -- Risk management http://www.iso.org/iso/catalogue_detail.htm?csnumber=40723

[ISO/IEC/IEEE 24765:2010, 2010] ISO/IEC/IEEE 24765:2010

https://webstore.ansi.org/RecordDetail.aspx?sku=ISO%2FIEC%2FIEEE%2024765:2010&source=google&adgroup=iso-iec-ieee&gclid=Cj0KEQiA_HDBRD2lomhoufc1JkBEiQA0TVMmr0KV9TkTcCQRCmhms_mfaLD2fTktvSgfjE7uBxcdBwaAmdA8P8HAQ

[Kangasharju, 2008] Jussi Kangasharju Distributed Systems: What is a distributed system? University of Helsinki https://www.cs.helsinki.fi/u/jakangas/Teaching/DistSys/DistSys-08f-1.pdf

[Microsoft, 2003] A Guide to Building Enterprise Applications on the .NET Framework https://msdn.microsoft.com/en-us/library/ms954601.aspx

[Microsoft, 2012] Components and Web Application Architecture https://msdn.microsoft.com/en-us/library/bb727121.aspx#mainSection

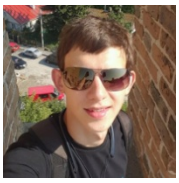
[Oxford, 2016] Oxford Living Dictionaries. Oxford University Press, 2016. https://en.oxforddictionaries.com/definition/us/distributed_system

[SAD, 2016] Software Architecture & Design Tutorial https://www.tutorialspoint.com/software_architecture_design/pdf/distributed_architecture.pdf

[Unified Component Model for Distributed, RealTime and Embedded Systems, 2013] Unified Component Model for Distributed, Real-Time and Embedded Systems RFP http://www.omg.org/cgi-bin/doc?mars/2013-09-10

Authors' Information

Elena Chebanyuk *– Software Engineering Department, National Aviation University, Kyiv, Ukraine,*
Major Fields of Scientific Research: *Model-Driven Architecture, Model-Driven Development, Software architecture, Mobile development, Software development,*
e-mail: chebanyuk.elena@ithea.org



Mykhailo Kostiuk - .Net software developer at PPN Soft, *Kyiv, Ukraine*
Major Fields of Scientific Research: Software development, Software architecture, Test-Driven Development
e-mail: hell.gunshe@gmail.com

FROM IDEA TO MARKET: MATERIALIZING INNOVATION IN INFORMATION AND COMMUNICATION TECHNOLOGIES IN ACADEMIA THROUGH ENTREPRENEURSHIP AND COMMERCIALIZATION

Martin P. Mintchev, Orly Yadid-Pecht

Abstract*: The path of an idea from the first moment it has been conceived to successful entry to the information and communication technology market has been veiled in mystery for years. Although bits and pieces of that path, along with the amazing stories of successful academic entrepreneurs, have been widely publicized, a coherent, comprehensive and organized description of how this can happen in the complex modern world of today is lacking. The purpose of this paper is to partially fill this void in a succinct, yet thorough manner.*

Keywords*: Information and Communication Technology, Entrepreneurship, Patents, Intellectual Property, Academia, Technology Transfer, Lean Start-up, Commercialization.*

ITHEA Keywords*: Information and Communication Technology*

Introduction

The complex world of materializing innovative ideas in information and communication technology has been the subject of numerous studies [Santangelo, 2002], [Etzkowitz, 2000], [Osterwalder, 2005]. Several books have been written to help aspiring entrepreneurs in the area succeed [Berker 2005], [Davenport 2013], [Unwin, 2009]. However, we have not seen a comprehensive and yet succinct guide into the kitchen of this process in academia. The aim of the present paper is to fill this void, offering a complete walk from the very inception of a possibly innovative idea, to its protection, its dissemination, its prototyping, the testing of the developed prototypes, actual product development, exit strategies etc. Our goal is to send an optimistic message to academics pondering commercialization pathways, that entrepreneurial success in this rapidly developing area is very possible, and to help them structure their own path to such success.

Conceiving the Idea

Generating a really innovative idea in the area of information and communication technologies requires first and foremost, intimate, and tirelessly pursued up-to-date knowledge of the cutting edge of such ideas generated by all other players in the narrow subfield to which such family of ideas would belong. One favorite analogy of ridiculously obvious idea that can convert even a respected scientist into the laughing stock of his or her colleagues is the discovery of the “wheel”, or the “hot water”. The context of the joke is clear – one can live in an oblivion that these have been discovered centuries ago, and seriously hope that he or she has actually invented them. We are not going to dwell here on the well-defined differences between discoveries and inventions [Myers 1995], but rather will discuss in the necessary details what we consider adequate work with the existing scientific and patent literature, which would prevent an inventor from the embarrassment of being padded on the shoulder with the condescending remark that in fact, they have invented the “hot water”. From our experience, we suggest the following approach in testing what you perceive to be a valuable innovative idea against the four pillars of successful modern patent protection: proper subject, novelty, inventive step, and industrial applicability.

Proper Subject

United States patent law has been considered a pivotal basis for defining the proper subject of a patent, first because it is the first to discuss it, second, because it is very well established and tested in numerous real court cases, and third, because successful patenting in the largest world economy is the most convincing argument of the seriousness of the invention. The United States Patent and Trademark Office (USPTO, www.uspto.gov) defines a proper subject if it belongs to 4 distinct categories: process, machine, manufacture, or composition of matter.

Process is defined as an act, or a series of acts or steps performed upon the subject-matter to be transformed and reduced to a different state or thing.

Machine is a concrete thing, consisting of parts, or of certain devices and combination of devices. This includes every mechanical device or combination of mechanical powers and devices to perform some function and produce a certain effect or result.

Manufacture (or manufacturing, or even manufacturability. i.e. being able to be manufactured) includes any article produced from raw or prepared materials by giving to these materials new forms, qualities, properties, or combinations, whether by hand labor or by machinery.

Composition of matter includes all compositions of two or more substances and all composite articles, whether they be the results of chemical union, or of mechanical mixture, or whether they be gases, fluids, powders or solids.

It has been argued successfully, that some specific software is in fact a form of a software process, i.e. a prescribed procedural way of making a certain process materialize. In these circumstances, software based patents have been recently allowed [Geiger, 2005]. However, because of the many ways to achieve a certain programming results by using a huge variety of different processes, the strength of such patents in real court proceedings has been questionable. Therefore, an argument persists that some software processes could bring more commercial benefits if they are copyrighted, rather than patented. Future legal and court practices will undoubtedly demonstrate which approach will be gaining or losing value.

Novelty

In a brilliant graduation thesis entitled “Novelty in Patent Law” defended as early as 1896 [Ryan, 1896], Michael L. Ryan characterizes the concept of novelty in the context of U.S. patent law not only as consisting in the substantial variation of the invention in question from all inventions, which, in contemplation of law, are already open to the public, but also, as the subject of various practical court-defined tests. Since then, these tests have substantially evolved, and the matter has become so legalistically complex, that our suggestion is not to sink into the deep waters of the legal evolution of this subject, but rather, to concentrate on a single, simple test, which we would define as follows:

Can you formulate a major claim for your invention that will contain a detail or details making your proposal functionally better compared to the closest similar inventions that have been already published (as disseminated scientific articles, internet pages, issued patents of published patent applications), or of their combinations? And, vice versa, if the closest inventions to yours that have been already published, or their combinations, lack this detail or details that are pivotal to your invention, will they be functionally inferior?

Of course, this test is very broad and rough, but it can be considered a good starting point. How well this initial test will serve you depends on two important factors: (1) How well (meaning, how mercilessly) will you search the abundant world of any possible publications, product brochures, issued patents, and published patent applications to discover as extensive as possible list of already published inventions that are close to yours; and (2) will your invention be replicated if you simply combine two or several of the already published inventions that you discovered. The most prevalent problem in defining novelty, in

our opinion, is the poor preliminary work with the existing published literature, resulting in the inability to define a comprehensive enough list of prior art related to your invention. This creates an aura of false hope for some imaginary value in your invention, which, if not properly controlled can result in an almost religious belief that you have come up with something really “big”. Several years into prototype development and testing, and deep into the patenting process, a diligent and professional patent examiner will send you the devastating message, that in fact, you have been one of the many “mad inventors”, pretending that they have discovered the “wheel”. By then, you would have probably invested lots of time and money in developing your “invention”. More importantly, you could have even misled not only yourself, but external partners, collaborators, investors, and even relatives to support you in your quest to fame. Second, the lack of mercilessly combining the set of previously published inventions in order to make sure that such combinations do not result in replicating your invention, can be equally embarrassing. Moreover, the later into the invention and patent developing process the failure of this simple novelty test is discovered, the greater will be the potential damages to your professional reputation, your reliability, and your finances. It is harder to recover from such damage the later you are in the process. Therefore, our recommendation can be only one: have no mercy in your data mining, and in combining the already published inventions close to yours in the quest to determine whether they would not replicate, alone or in combinations, what you claim as novel.

Inventive Step

Inventive Step, or non-obviousness of an invention is defined in the US and international patent law as an invention, which, if regarded with respect to the prior art, is not, at the prescribed relevant date, obvious to a person skilled in the art. There are three tricky points here:

First, the prior art includes public domain publications (journal and conference papers, public presentations, Internet articles and publications, and even a conversation between two or more persons. In addition, issued patents and published patent applications with earlier priority dates compared to yours. If a party that contests your invention can demonstrate that any of the priority dates preceded your provisional patent application priority date (or regular patent application priority date, should you decide not to file a provisional patent application), your invention, even if the patent has been issued, can be invalidated.

Second, the prior art includes any possible combinations of single prior art items in any possible order, i.e. public domain publication(s) combined with other public domain publication(s), public domain

publication(s) combined with patent application(s), patent publication(s) combined with other patent application(s), etc.

Third, the definition of an invention becoming obvious to a person skilled in the art familiar with the cited prior art can be truly based on the subjective assessment of the patent examiner, who often plays the role of such person himself or herself. In cases of dispute expert witnesses are requested to define, for the relevant patent in question what defines “a person skilled in the art” for that domain.

Interestingly, even an issued patent can be later invalidated by the courts if upon judicial examination any of the requirements for Inventive Step are not met. Thus, vigorous and maximally comprehensive and broad self-testing before actual patent filing (provisional or not) is a must. In this process, of extreme importance become three factors – (a) the flexible and broad selection of combinations of searching keywords; (b) the correct identification of the numerical patent category and subcategory to which your invention belongs; and (c) the rigorousness of your data mining in both the patent and patent application databases, as well as in all possible corners of the public domain. Practically, your work on these three factors should exceed the rigorousness of the same work performed by the patent examiner who would be determining the merit of your patent application at one point of the process. It is not only embarrassing to recognize late that your critical analysis of the state-of-the–art closely related to your invention had not been adequate. It will cost you. It will cost you, first of all, money. Very often – lots of it. As we are going to discuss later, one of the best approaches in patenting is to have patent protection as early as possible, but procrastinate the associated patent costs as much as possible. However long this procrastination might be, the associated costs will be piling. Perhaps you will not notice them, because they might be distributed in time. However, at one point, the moment of truth will come, and you will face the assessment of your patent application by a qualified and very well trained patent examiner. One of the most devastating ways to defeat a patent application and throw it into oblivion is to find it already replicated in prior state-of-the-art. Then you can really feel that your investment, both in terms of time and money, and your hopes go down the drain. “From the great to the ridiculous there is only but a step”, said once Napoleon Bonaparte, and this immortal wisdom applies in full swing to the patenting process.

Finally, we would like to mention, that the US patent law allows you to patent a previously published work of yours in the form of a journal article, conference article, or public presentation, as long as the patent application filing is done within one year after such publication. However, many other patent jurisdictions across the globe are not that generous.

Industrial Applicability

The onus is on you to demonstrate that your invention, for which you seek from the government exclusive patent protection, has some direct utility in the industry as the latter is defined about the priority filing date of your patent application. Otherwise, the invention remains an abstract intellectual contribution without any practical value, and although it could very well be a major discovery, it cannot be protected as an invention. Thus, for example, Albert Einstein could not have patented any process, machine, manufacturing, or composition of matter, based on his Theory of Relativity [Einstein, 1922], because this theory belongs to the realm of the major discoveries in the area of physics, rather than being possibly associated with something that can be directly and immediately utilized by any industry while he (Einstein) as alive and productive. Of course, a machine that can achieve the speed of light, for example, the design of which could be based on this theory, could be patentable, if its practical implementability could be convincingly demonstrated to a patent examiner at the time of such eventual patent application filing and examination.

Worthy of note is that a patent lifetime is 17 years. The inventors receive the protection of the USPTO for this time frame, while after it lapses the patent is available for public use. This means that time is of essence and commercial benefits from your successful invention would be limited in time as well.

Writing and Filing a Patent Application

It is widely believed that to write and file a competitive and serious patent application, you need the help of a lawyer. This, of course, poses a major obstacle for individual inventors and small company startups, particularly if located outside the USA. In our opinion, engagement of a lawyer is not necessary, particularly in early stages of patent application writing and even filing. Patent writing is not significantly different in its logistics than writing a scientific journal or conference paper. However, it does have a specific logistic that is quite unique and different.

The best start is to define how broad the invention is. Can it involve all three major components of a technological innovation: apparatus, system and method (not necessarily in this order)? If so, you need to protect all these major components, and learn how to write such a patent application. The most appropriate approach is to examine a successfully issued patent, preferably in the area of your invention, which very clearly and distinctly describes these three aspects and the claims of which are very clearly separated into them. For example, pretend that your invention is in the area of information management. A rigorous search of the state-of-the-art in the area satisfies you that you have something

really innovative and unique. However, it should also acquaint you with many already issued patents that describe innovations in this area. One of them is the US Patent No. 7,599,983 issued on Oct 6, 2009 by Harper et al, entitled “Method, apparatus and system for management of information content for enhanced accessibility over wireless communication networks” [Harper, 2009]. This prior art is a great example of patent writing covering all three aspects of a broad patent, it is professionally written, and the assignee is a really existing US corporation, indicating its seriousness. Structurally, it has everything that you need to learn from it how to write your application. Moreover, its claims are divided into all three major components (method, system and apparatus). If your invention is not that broad, and involves only an apparatus, a system, or a method, or any combination of two rather than all three of these components, your writing should be in fact easier, since you would need to cover only a subset of the three components. If you really have the motivation to be successful as an entrepreneur in information and communication technologies, you can start by writing your own patent application draft, and performing your thorough state-of-the-art data mining to convince yourself that your idea is really a patentable invention. If this is done relatively well, a professional and experienced patent lawyer’s review can be done with reasonable funds and hence be cost effective.

Patent Application Is Written and Ready. So, Now What?

As we mentioned earlier, in our opinion the best approach in patent pursuit is to have protection as early as possible, while procrastinating the patenting costs as long as possible. Thus, it seems that filing the ready patent application as a provisional patent application with the USPTO is the cheapest possible approach to take. At the time of writing of this article, the cost for such application is only 130USD for a small entity, and the necessary form is available online. An even cheaper option is to claim a micro entity status, which, however, has some more stringent restrictions that should be carefully observed. The filing of a US provisional patent application, particularly if it is done online, gives you an immediate protection date for a period of 1 year, without the provisional application being examined at all, i.e. this protection is unconditional. As mentioned earlier, a patent lawyer can be consulted in relation to this filing, with initial costs not beyond $2500. Within this one full year of protection, you should be able to do many important things in order to test the value of your invention.

First, you can design and test a working prototype of the invention. The closer your working prototype would be to an actual commercial product, the better, of course.

Second, you can contact the top executives of various companies that might be interested in your invention. Very often, top executives of such companies would not communicate with individual inventors, because they have been overexposed to sufficient number of “mad inventors” trying to convince them that their invention is “the next big thing” in the area. Therefore, and particularly if you have a working prototype of your invention, it might be prudent to consider opening a start-up company and assigning the filed provisional patent application to it. Thus, contacting the CEO even of a large corporation could become a CEO-CEO communication, rather than a “mad inventor” bother. In addition, if your invention truly has some merit, negotiations between corporations, including licensing and/or acquisition, could be far better handled.

Third, you can start writing scientific conference papers and present the invention at major scientific forums in the area, and in particular the ones that are integrated with large technological exhibit. In fact, you can have both scientific presentation related to your invention, and rent a booth to present your working prototype at the associated technological exhibit. In some situations, large technological exhibits are separate from any scientific conference forums, but renting a booth on them might be financially prohibitive. In such situations, preference given to scientific conferences in order to spread the word about your invention is an obvious choice.

Fourth, you can publish your research associated with the invention in a major journal in the area that is normally read by the industry. You will have to assess the impact factor of the journal, and probably look for evidence of some industrial involvement in it.

In addition, in this time span, you need to do some market research to understand if your invention solves a real “pain”. We will point this out later when we discuss exit strategies and going all the way on your own by creating a start-up company.

If within the first 9 months after the filing of your provisional patent application you have successfully walked all of the above four avenues to disseminate the news about your invention, and you have not heard any positive feedback as a result, and on the other hand you cannot point out where your invention solves a “real pain”, the time has come for you to reassess its merit and decide what to do next. More often than not, cutting your losses by abandoning the provisional application and re-directing or perfecting your efforts by re-focusing would be a wise choice.

One way or the other, if you decide to continue with your application, in our opinion your best choice is to convert the provisional patent application into an international patent application according to the so-called Patent Cooperation Treaty (PCT) administered under the auspices of the World International Property Organization (WIPO, www.wipo.int). The signatories of the PCT are all major economies of the

world, and such an application can later be converted into a national patent application in any, several or all countries that has signed the treaty. This will give you not only a time extension to perfect your work in the above four invention dissemination avenues, but also a non-binding professional examination by a patent examiner working for the patent office of the country in which the international patent application is filed. This preliminary examination process by the WIPO is approximately 18 months. After the International Search Authority issues its professional opinion about your invention, you will have another chance to cut your losses and call it quits. This time, however, the international PCT application filing and the process of preliminary non-binding examination will cost you over a thousand dollars (see e.g. http://www.wipo.int/export/sites/www/pct/en/fees.pdf). If, therefore, after 30 months (2 ½ years) of continuous patent protection, and in the presence of genuine dissemination attempts and search for real "pain" your invention solves, no external indications have been received of the merit of the invention, the only asset that can be of potential future value would be a very positive opinion by the International Search Authority in the three major aspects of any invention (novelty, inventive step and utility). If you feel pushed back by the lack of any positive feedback about your invention resulting from your dissemination efforts and market research, would you be prepared for yet another round of such efforts based on this positive opinion of the International Search Authority? At this point, the decision is critical, but it is yours, and yours alone. If you decide to continue with the patenting process, you will need to consider converting the PCT application into a "national phase". Very often, and considering the fact that the US economy is the biggest in the world, and to reduce costs, a conversion to a full-blown US patent application only is the preferred choice. A second and broader choice would be to convert it also to the national phases of English-speaking countries or the ones accepting and processing patent applications in English (e.g., the UK, Ireland, Canada, Australia, Switzerland, the Netherlands, Sweden, Norway, Finland, or a subset of these). Practice shows that patenting in Latin America, China, Japan, in the rest Asia and in Africa do not add much value, as the level of governmental protection to foreign inventors could be questionable, if not because of other issues, due to language barriers and the associated costs. At the point of conversion of your PCT patent application to a full-blown US patent application we recommend the engagement of patent agents in order for the logistics of the filing and the patent processing and prosecution to be handled properly. Indeed, by the time of the PCT conversion, the inventor should be already very clear whether a more substantial financial investment in the invention is warranted.

Dissemination of the Protected Invention Was Successful. Now What?

Well, enough with the possible negative scenarios! Let us assume that you followed the 4 dissemination avenues described above and your invention generated the long anticipated industrial interest or that you have found real "pain" that your invention solves, to an extent that you firmly believe it is worth your investment in a start-up. From this point on, and depending on the seriousness of this interest, you have exactly 4 options in front of you.

1. Licensing Agreement.

Although engaging a contract specialist (a lawyer or not) is recommended, do not forget that licensing agreements are made by people like you and for people like you. You need to understand very clearly what is offered, is it exclusive or non-exclusive, and never forget one simple principle in business negotiations: whoever talks numbers first during the (often informal) start of these negotiations, usually demonstrates eagerness and loses. So, patience is a virtue, but in the information and communication technology asking for more than 300-400K USD down payment and 4% royalty on sales can often mean overestimating your invention. However, who knows – may be your invention is really a major innovative step in the area! So, our recommendation is – stay cool and wait for the other party to talk numbers first.

2. Right-of-First-Refusal Agreement (ROFR).

This is a possible avenue for collaboration with an industrial partner who is willing to fund you to perfect your prototype, make it closer to a commercial product, test mass production, test the market at arm's length from this partner, etc. The onus will be on you to present a good business model as to how will you propose to collaborate with the partner. At the end of the duration of the collaboration, the funding partner will have the exclusive right to license the invention, and you will resort to Option 1 above. Keep in mind that if at this point the partner refuses to exercise its right to license, this could be the "kiss of death" for you, since in the due diligence process in any future licensing or acquisition negotiations with other potential partners, you need to disclose what has happened with this ROFR.

3. Acquisition.

This, of course, could be the sweetest possible outcome of your inventive effort, resulting in you selling your start-up company to a bigger fish and putting a period to your inventive effort on this file altogether. However, keep in mind that history has seen many bitter inventors that have done just that, and for what they believed to be very good money at the time, only to observe their invention making it tremendously

big, to an extent that the money they received for it now start looking to them like small cramps from the big feast that their invention has become. On this feast, however, the guests are now others, not the original inventor.

4. Going All the Way on Your Own.

This is the most difficult, but also, the most rewarding path to entrepreneurial success. It, however, requires an entirely new discussion, ranging from your ability to attract governmental support for your first steps, to attracting investors, and further to manufacture, market and sell the product based on the invention. We believe in the "Lean Start-up" methodology as presented by Steve Blank [Blank, 2013]. The idea is that a start-up _looks_ for a business model, so it needs to come up with the minimal viable product as soon as possible. What we mean is that we encourage young entrepreneurs to do a basic prototype of their invention, go out and test it and improve it with feedback gotten from the potential customers. In essence, we extend this methodology to the patenting realm as well, and see a provisional and PCT filing as a "Lean patenting" way available to academics. Following our proposal of doing the patenting in such a way, we embrace the idea of prototype buildup and checking with potential customers their real needs (for example in forums, or in web based product reviews) to have assurance that this invention answers a real need or solves a real problem that is worthy of pursuit. For instance, while researching compact food allergen detection devices in our lab, we have noticed that many customers currently complain in forums on a newly marketed product for gluten detection that fails to detect the gluten in beer, soy sauce or other fermented foods. Hence, we understood that solving the invariability to different pH levels in our product in development is addressing a real unresolved "pain" and we focus our efforts on solving this. As outlined by Blank, lean start-ups don't begin with a business plan; they begin with the search for a business model. Only after quick rounds of experimentation and feedback reveal a model that works do lean founders focus on execution.

Conclusion

In this article we tried to outline possible avenues to add an entrepreneurial twist to routine academic endeavors in the area of information and communication technologies, with the message that this is not tremendously difficult, nor it is tremendously expensive, as long as there are inventive ideas to be protected and developed. We basically offer an extension to the "Lean Start-up" methodology to include patent filing as part of the minimum viable product and to be done in such a way to be cost effective.

Bibliography

[Santangelo, 2002] Santangelo, Grazia D. The regional geography of corporate patenting in information and communications technology (ICT): domestic and foreign dimensions. Regional Studies 36.5: 495-514, 2002.

[Etzkowitz, 2000] Etzkowitz, Henry, and Loet Leydesdorff. The dynamics of innovation: from National Systems and “Mode 2” to a Triple Helix of university–industry–government relations. Research policy 29.2: 109-123, 2000.

[Osterwalder, 2005] Osterwalder, Alexander, Yves Pigneur, and Christopher L. Tucci. Clarifying business models: Origins, present, and future of the concept. Communications of the association for Information Systems 16.1: 1, 2006

[Berker 2005] Berker, Thomas, Maren Hartmann, and Yves Punie. Domestication of media and technology. McGraw-Hill Education (UK), 2005

[Davenport 2013] Davenport, Thomas H. Process innovation: reengineering work through information technology. Harvard Business Press, 2013.

[Unwin 2009] Unwin, P. T. H., ed. ICT4D: Information and communication technology for development. Cambridge University Press, 2009.

[Myers 1995] Myers, Greg. From discovery to invention: The writing and rewriting of two patents. Social Studies of Science 25.1: 57-105, 1995.

[Geiger 2005] Geiger, Christophe, and Reto M. Hilty. Patenting Software?-A Judicial and Socio-Economic Analysis. IIC-international review of intellectual property and competition law 36.6: 615-647, 2005.

[Ryan, 1896] Ryan, Michael L. Novelty in patent law, 1896

[Einstein, 1922] Einstein, Albert. The general theory of relativity. The Meaning of Relativity. Springer Netherlands, 1922. 54-75.

[Harper 2009] Harper, David Walker, Jason James Sabella, and William Henry Munch. Method, apparatus and system for management of information content for enhanced accessibility over wireless communication networks. U.S. Patent No. 7,599,983. 6 Oct. 2009.

[Blank 2013] Blank Steve. Why the Lean Start-Up Changes Everything, Harvard Business Review,2013.

Authors' Information

Martin P. Mintchev, Ph.D., P.Eng., FAIMBE *– Professor in the Department of Electrical and Computer Engineering, Schulich School of Engineering, University of Calgary, Calgary, Alberta, Canada T2N1N4; e-mail: mintchev@ucalgary.ca*

Major Fields of Scientific Research: Electronic systems, biomedical and oilfield instrumentation, information and communication systems.

Orly Yadid-Pecht, D.Sc., FIEEE, FSPIE, FAIMBE *– Professor and AITF Strategic Chair in the Department of Electrical and Computer Engineering, Schulich School of Engineering, University of Calgary, Calgary, Alberta, Canada T2N1N4; e-mail: Orly.Yadid-Pecht@ucalgary.ca*

Major Fields of Scientific Research: Integrated Sensors, Very Large Scale Integration, biomedical sensors and systems, information and communication systems.

FORMATION OF PRIORITIES OF NATIONAL MEZOEKONOMICAL POLITICS UNDER THE CONDITIONS OF IMPLEMENTATION ON OF THE PARIS AGREEMENTS

O.Voloshyn, V.Kudin, A.Onyshchenko, L. Khrushch

Abstract. *In the article the modified balanced ecologic-economical model is offered as "input-output" with the Paris agreement set limits on greenhouse gas emissions. The mathematical definition of the device changes the volume of gross output of main and auxiliary production in case of change of branch structure.*

Keywords: *sustainable development, the Paris Agreement, an ecological and economic system, Leontief "input-output" model, Leontief-Ford "input-output" model, simulation modeling.*

ACM Classification Keywords*: I. Computing Methodologies – I.6. Simulation and modeling – I.6.5. Model Development – Modeling Methodologies.*

Introduction

A number of issues related to the country's participation in the Paris Agreement [1] determines necessary, first of all, estimation potential future market volume of ecological services, determination of possible partners, development of economic strategy that would identify priorities for each economic mechanism, proportions of their application in order to attract the maximum amount of ecological investments.

A special role in solving the fundamental problems of nature using, studying values of environmental expenditures considering socio-economic impact and their distribution in a territorial-branch cut [2, 3], belongs to balance ecologic-economical models of "input-output" as well as regional and sectorial models [4,5,6].

Therefore, there is need to build a balance ecologic-economical model that would include the cost of implementing commitments under the Paris Agreement. Thus, inalienability on the maintenance of economic and ecological indicators requires research ofquestion of the productivity of balance model productivity. The last is related to properties of technological matrix model. Changing the sectorial (brunch) structure of ecological and economic system is reflected in the data matrix coefficients influences on production volumes and requires development of algorithms of determination of decision without the decision of model equalizations.

Objectives

Implementation of the set obligations in relation to unexceeding of quotas on the extras of greenhouse gases imposes certain limits on the basic economic indicators of productive activity. As there is dependence between the volumes of gross output and volumes of emissions of CO2, first of all, it concerns the gross output of products, the volume of investments, the final product, their optimal distribution in the system of national wealth. Following the general principle of division of economic researches into micro, meso and macro levels, we will consider the production functioning in the cut of existent industries, that conditioned by complication and manyfactoriness of tasks of reduction of greenhouse gas emissions in the national economy.

As the content of the Paris agreement has ecological and economic character, the realization of its positions requires the use of a multidisciplinary approach. As an effective tool of research in this case the balance ecological and economic model "input-output" [2] can be examined. It has a special role in solving the fundamental problems of the perspective planning taking into account nature using, namely the value of study costs of environmental protection taking into account the socio-economic effect and their distribution in a territorial-branch cut.

The first balance model that covers the relationship of the economy and the environment, was proposed by V.Leontyev D.Ford [3]. It generalizes the classical scheme of inter-branch balance and includes two groups of industries: basic production (material production industry) and auxiliary production (industry to eliminate contaminants). The basic conditions of the model are expressed by the system of equalizations:

$$\begin{aligned} x_1 &= A_{11}x_1 + A_{12}x_2 + y_1, \\ x_2 &= A_{21}x_1 + A_{22}x_2 - y_2. \end{aligned} \tag{1}$$

In the system (1) $x_1 = \left(x_1^1, x_2^1, \ldots, x_n^1\right)^T$ – vector-column of production volumes;

$x_2 = \left(x_1^2, x_2^2, \ldots, x_m^2\right)^T$ – a vector-column of destroyed pollutants volumes;

$y_1 = \left(y_1^1, y_2^1, \ldots, y_n^1\right)^T$ – a vector-column of final production volumes;

$y_2 = \left(y_1^2, y_2^2, \ldots, y_m^2\right)^T$ – a vector-column of undestroyed pollution volumes;

$A_{11} = \left(a_{ij}^{11}\right)_1^n$ – a square matrix coefficients of direct costs of production i per unit of production j;

$A_{12} = \left(a_{ig}^{12}\right)_{i,g=1}^{n,m}$ – a rectangular matrix of production costs i per unit of destroying pollutants g;

$A_{21} = \left(a_{kj}^{21}\right)_{k,j=1}^{m,n}$ – a rectangular matrix of pollutants producing k per unit of output j;

$A_{22} = \left(a_{kg}^{22}\right)_{1}^{m}$ – a square matrix of pollutants producing k per unit of destroying pollutants g.

It is unobviously assumed in the system (1), that coefficients $a_{ij}^{11} \geq 0$, $a_{ig}^{12} \geq 0$, $a_{kj}^{21} \geq 0$, $a_{kg}^{22} \geq 0$ distribute on all types of productiveactivity (material production and elimination of pollutants) the hypotheses of basic model of inter-branch balance: the amount of technological methods equals the amount of types of products and in every technological method produces only one product type. In the future we will consider matrixes A_{11}, A_{12}, A_{21}, A_{22} inalienable: $A_{11} \geq 0$, $A_{12} \geq 0$, $A_{21} \geq 0$, $A_{22} \geq 0$. The economic content of the Leontiev-Ford model requires that all its variables were inalienable, that is, $x_i^1 \geq 0$, $x_k^2 \geq 0$, $y_i^1 \geq 0$, $y_k^2 \geq 0$.

Research results

Let's put the problem on the basis of the above balance circuit "input-output" considering the costs of complying with restrictions on the Paris agreement. Solving this problem involves solving of the whole complex of the modern science fundamental problems. It includes, for example, the development of reliable methods for predicting environmental parameters and criteria of quality that can provide a quantitative measuring of the degree of the human needs satisfaction in a cleanness and natural variety; creation of scientifically reasonable methodology of determination of economic loss is from the pollution of environment; construction of the system of models of cooperation of different components of natural complexes taking into account natural and anthropogenic factors and conditions.

It is suggested to take into account the costs on implementation of emission limitations of greenhouse gases in the structure of industries of basic production in the form:

$$\begin{cases} x_1 = A_{11}x_1 + A_{12}x_2 + Cy_2 + y_1, \\ x_2 = A_{21}x_1 + A_{22}x_2 - y_2, \end{cases} \tag{2}$$

here Cy_2 – the costs of greenhouse gas emissions (that is maintenance costs of greenhouse gas emissions, including a payment for allowances);

$C = \left(c_{ig}^{12}\right)_{i,g=1}^{n,m}$ – rectangular matrix of production costs i per unit of pollutant emissions g;

In vector-matrix form model (2) can be represented as:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} + \begin{pmatrix} E_1 & C \\ 0 & -E_2 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix},$$

here E_1 and E_2 – are corresponding single diagonal matrixes.

The first equalization of the offered model represents the economic balance – the distribution of the branch grossproducing of products on the productive consumption of basic and auxiliary productions, final consumption ofbasic production and charges related to fulfiling commitment on the Paris agreement. The second equalization represents physical balance ofgreenhouse gases, as a sum of the emissions caused byactivity of basic and auxiliary productions, and their undestroyed volumes.

The economic content of the variables of the model (2) requires the consideration of their inalienable values. The latter is closely connected with the question of balance model production to talk about the actual functioning of the production system that can provide intermediate consumption, positive volume of the final product and implementation of the set limitations from the extras of greenhouse gases.

In of order to study the issue of solutions express inseparable x_2 from the second equalization and substitute it in the first place:

$$x_1 = (E_1 - A_1)^{-1}\left(y_1 + Cy_2 - A_{12}(E_2 - A_{22})^{-1} y_2\right),$$

here $A_1 = A_{11} + A_{12}(E_2 - A_{22})^{-1} A_{21}$ – a square matrix of n - order.

Let's also express x_1 from the first equalization and substitute it into the second equalization:

$$x_2 = (E_2 - A_2)^{-1}\left(A_{21}(E_1 - A_{11})^{-1} y_1 + A_{21}(E_1 - A_{11})^{-1} Cy_2 - y_2\right),$$

here $A_2 = A_{22} + A_{21}(E_1 - A_{11})^{-1} A_{12}$ – a square matrix of m - order.

Thus, the formal solution of system (2) can be written as:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} (E_1 - A_1)^{-1} & (E_1 - A_1)^{-1}\left(A_{12}(E_2 - A_{22})^{-1} - C\right) \\ (E_2 - A_2)^{-1} A_{21}(E_1 - A_{11})^{-1} & (E_2 - A_2)^{-1}\left(E_2 - A_{21}(E_1 - A_{11})^{-1} C\right) \end{pmatrix} \begin{pmatrix} y_1 \\ -y_2 \end{pmatrix}.$$

According to the methodology proposed in [3] we can generalize the concept of productivity in the case of block matrix with the inalienable elements:

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \geq 0. \tag{3}$$

Let's consider an inalienable block matrix productive, if productive are matrixes A_{11}, A_{12}, A_1 and A_2. The productivity of matrixes A_1 and A_2 means profitability of basic and auxiliary productions after the

complete cycle of production of goods and after the complete cycle of greenhouse gases elimination. If matrixes A_{11}, A_{12}, A_1 та A_2 – are productive, then matrixes

$$(E_1 - A_{11})^{-1} \geq 0,\ (E_2 - A_{22})^{-1} \geq 0,\ (E_1 - A_1)^{-1} \geq 0,\ (E_2 - A_2)^{-1} \geq 0$$

exist and have inalienable elements.

The productivity of the block matrix (3) does not guarantee inalienability of solutions of the system (2). Let's analyze the got expressions for x_1 and x_2. From the system (2) we get

$$x_1 = (E_1 - A_{11})^{-1}(A_{12}x_2 + Cy_2 + y_1).$$

From this equalization it appears, that when $x_2 \geq 0$, $y_1 \geq 0$, $y_2 \geq 0$ the condition $x_1 \geq 0$ is completed

Thus, a necessary and sufficient condition of inalienability of the model solutions (2) when the block matrix (3) is productive and when $y_1 \geq 0$, $y_2 \geq 0$ the condition $x_2 \geq 0$, will be completed, that is

$$(E_2 - A_2)^{-1}\left(A_{21}(E_1 - A_{11})^{-1} y_1 + A_{21}(E_1 - A_{11})^{-1} Cy_2 - y_2\right) \geq 0.$$

From the last inequality we get the sufficient condition of existence of inalienable solutions:

$$A_{21}(E_1 - A_{11})^{-1}(y_1 + Cy_2) \geq y_2,$$

that can be replaced yet by more hard sufficient condition:

$$A_{21}(y_1 + Cy_2) \geq y_2.$$

The last inequality means that a sufficient operating condition of basic and auxiliary productions is unexceeding of volume of unutilized of greenhouse gas emissions above complete emissions of greenhouse gases that arise up at the production of final product and costs sent to the service obligations according to the Paris Agreement.

Another actual scientific problem is that the basis of the proposed model (2) is a schematic diagram of inter-branch balance, the first quadrant of which are inter-branch streams that are built to the functional and structural industry connections. A branch plays a crucial role in providing progressive structural changes in the economic system, accelerating development and improving the technical level of production. It creates socio-economic background of deep transformations in the field of labour, service and householding. The interbranch economy relations are multilateral, the scale of their development determine the volume of trade and its structure. Features and change of the branch complexes operating conditions determine the necessity of their taking into account during the planning and forecasting calculations of gross branch production, of the basic and auxiliary productions, of the final

product, of the volume of greenhouse gas emission reductions, of the interbranch ecological and economic connections.

The structural change of the indicated indexes is determined by the elements of technological matrixes of model (3) that stipulates the necessity of development of the estimation algorithms of the change influence in matrix structure system on the solution of the system of equalizations.

The model (2) can be also represented as:

$$Au = C. \tag{4}$$

Where

$$A = \begin{pmatrix} E_1 - A_{11} & -A_{12} \\ -A_{21} & E_2 - A_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad u = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - (n+m)\text{-dimensional vector,}$$

$$C = \begin{pmatrix} E_1 & C \\ 0 & -E_2 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}, \; E_1, \; E_2 \text{ – corresponding dimension block unit matrixes,} \tag{5}$$

0 – block zero matrix.

Let us also consider the system perturbations (in matrixes A_{11}, A_{12}, A_{21}, A_{22}, C elements) to linear algebraic equations system (5):

$$\bar{A}u = \bar{C}. \tag{6}$$

Where A, C - are corresponding perturbed matrix. Let us suppose that for the system (4) the basic solution and inverse matrix were found. Then there is the following theorem [Kudin, 2007].

Theorem 1. There are the following ratio for vectors normal restrictions expansion coefficients on matrix basic lines, inverse matrices elements, basic solutions and restriction residuals in two related basic solutions:

$$\bar{\alpha}_{rk} = \frac{\alpha_{rk}}{\alpha_{lk}}, \; \bar{\alpha}_{ri} = \alpha_{ri} - \frac{\alpha_{rk}}{\alpha_{lk}}\alpha_{li}, \; r = \overline{1, n+m}, \; i = \overline{1, n+m}, \; i \neq k. \tag{7}$$

$$\bar{e}_{rk} = \frac{e_{rk}}{\alpha_{lk}}, \; \bar{e}_{ri} = e_{ri} - \frac{e_{rk}}{\alpha_{lk}}\alpha_{li}, \; r = \overline{1, n+m}, \; i = \overline{1, n+m}, \; i \neq k. \tag{8}$$

$$\bar{u}_{0j} = u_{0j} - \frac{e_{jk}}{\alpha_{lk}}\Delta_l, \; j = \overline{1, n+m}. \tag{9}$$

$$\overline{\Delta}_k = -\frac{\Delta_l}{\alpha_{lk}},\ \overline{\Delta}_r = \Delta_r - \frac{\alpha_{rk}}{\alpha_{lk}}\Delta_l,\ r = \overline{1, n+m},\ r \neq k. \tag{10}$$

The matrix condition for being basis when entering the normal vector a_l restrictions $a_l u \leq c_l$ for k-y basic matrix position A is the inequality fulfillment: $\alpha_{lk} \neq 0$.

Based on the reduced ratio we can build algorithmic scheme of study (6) (when the model has changes). The algorithm will be based on simplex method ideology [Voloshyn, Kudin, 2015], involving some iterative process features. In particular, the transition from the system (4) to the system (6) will be carried consecutively by relevant perturbed rows ($i, i+1, i+2, \ldots, i+i_0$) replacement.

This means that the normal vectors hyperplanes that form the basis matrix lines and the corresponding inverse matrix will be replaced by appropriate "perturbed" normal vectors. Following basic solutions and inverse matrixes will be recalculated based on simplex relations (7)-(10). While maintaining the basic properties on replacement iterations, system (6) solution would be found by i_0 iterations. The result is a new base solution and the inverse matrix.

During the realization of iterations to clarify the elements the models can have some changes and clarifications not onlyseparate elements and lines of matrix of limitations, but also columns. The above formulas do not clearly show the influence of such changes on solutions (6). Let's study the impact of changes k column of the matrix restrictions A as $\overline{A}_k = A_k + A_k^{'}$ to the solution u_0, where $A_k = (a_{1k}, a_{2k}, \ldots, a_{mk})^T$, $A_k^{'} = (a_{1k}^{'}, a_{2k}^{'}, \ldots, a_{mk}^{'})^T$, that is in such a form A is replaced by $\overline{A}$.

On basis of (6) let us build the next auxiliary linear system:

$$\max C^T u, \tag{11}$$

$$Au = C \tag{12}$$

$$u \geq 0 \tag{13}$$

$$\min C^T x, \tag{14}$$

$$A^T x > C \tag{15}$$

By the structure it is the dual pair of linear programming with some peculiarities of the vector coinciding - normals of objective function and vector type, the matrix of restrictions tasks A, A^T - are square. Let's consider that for (12) $u_0, A = A_b, A_b^{-1}$ - are known - the basic solution, basic line and inverse matrixes,

and a column A_k is a subject of indignation in the next kind $\overline{A}_k = A_k + A_k^{'}$ (accordingly $\overline{A}_k^T = A_k^T + A_k^{'T}$ - is an indignant row k of the matrix A^T). It should be noted that for (15) $A^T = A_b^T,\ \left(A_b^T\right)^{-1}$ - accordingly basic and inverse basic matrixes in the context of formulas (11) -(13). Let us draw a connection according to formulas (11) - (13) elements basic matrixes method in two contiguous basic matrixes the research values of component vector normal C^T decomposition of the objective function (14) by the substitution in the basic matrix A^T line k by a line $\overline{A}_k^T = A_k^T + A_k^{'T}$ and the conditions of nondegenerate preservation of matrix limits. Let's also establish their links with the problem (11) - (13). Where as, $A_b \times A_b^{-1} = E,\ \left(A_b^T\right)^{-1} \times A_b^T = E$, for the expansion component of the vector-line C^T we can write down

$$C^T = L_0 \times A_b^T,\ L_0 = C^T \times \left(A_b^T\right)^{-1} = A_b^{-1} \times C\ , \tag{16}$$

here $L_0 = (L_{01}, L_{02}, ..., L_{0m})$ - is a vector of separation of the vector C^T according to the lines of the basic matrix A_b^T. It is not hard to persuade, that $A_b^{-1} \times C$ coinsides with u_0, that is $u_0 = A_b^{-1} \times C$ for (12) (or (6)). Similarly, while considering the indignant matrix $\overline{A}_b^T$ ($\overline{A}^T = \overline{A}_b^T,\ \left(\overline{A}_b^T\right)^{-1}$), that is constructed by the substitution of the line k of matrix A^T by the line $\overline{A}_k^T = A_k^T + A_k^{'T}$ the separation of vector C^T will look like

$$C^T = \overline{L}_0 \times \overline{A}_b^T,\ \overline{L}_0 = C^T \times \left(\overline{A}_b^T\right)^{-1} = \overline{A}_b^{-1} \times C\ , \tag{17}$$

where $\overline{L}_0 = (\overline{L}_{01}, \overline{L}_{02}, ..., \overline{L}_{0m})$ - is a vector of separation of the vector C^T according to the row of the basic matrix $\overline{A}_b^T$.

It is not hard to persuade, that $\overline{A}_b^{-1} \times C$ coinsides with $\overline{u}_0$, that is $\overline{u}_0 = \overline{A}_b^{-1} \times C$ for (12) (or (6)). Here the matrix $\overline{A}_b^{-1}$ is inversed to $\overline{A}_b = \overline{A}$, that is inversed to the matrix A, in which the substitution of k -column in matrix limits A_k by the column $\overline{A}_k = A_k + A_k^{'}$ is made. According to the positions of the basic matrix method (to the problem (14) - (15)) the connection is a component of the separation vector of normal vector C^T (L_0 and $\overline{L}_0$) at two contiguous basic matrixes (different by the k - row) is given in the co-relation

$$\overline{L}_{0i} = L_{0i} - \frac{L_{0k}}{\overline{L}_{kk}} \times \overline{L}_{ki},\ i \neq k \tag{18}$$

$$\overline{L}_{0k} = \frac{L_{0k}}{\overline{L}_{kk}},\ \ i = k \tag{19}$$

where for (14)-(15)

$$\overline{L}_{ki} = \overline{A}_k^T \times (A_b^T)_i^{-1}$$
$$\overline{L}_{ki} = (A_k^T + A_k^{"T}) \times (A_b^T)_i^{-1} = A_k^T \times (A_b^T)_i^{-1} + A_k^{"T} \times (A_b^T)_i^{-1}$$
$$\overline{L}_{ki} = L_{ki} + L_{ki}^{'} = 0 + L_{ki}^{'}$$

$$\overline{L}_{kk} = \overline{A}_k^T \times (A_b^T)_k^{-1} = \overline{A}_k^T \times (A_b^T)_k^{-1}$$
$$\overline{L}_{kk} = (A_k^T + A_k^{"T}) \times (A_b^T)_k^{-1} = A_k^T \times (A_b^T)_k^{-1} + A_k^{"T} \times (A_b^T)_k^{-1}$$
$$\overline{L}_{kk} = L_{kk} + L_{kk}^{'} = 1 + L_{kk}^{'}$$

accordingly, for (11)-(13)

$$\overline{L}_{ki} = \overline{A}_k^T \times (A_b^T)_i^{-1} = (A_b)_i^{-1} \times \overline{A}_k$$
$$\overline{L}_{ki} = (A_k^T + A_k^{"T}) \times (A_b^T)_i^{-1} = (A_b)_i^{-1} \times (A_k + A_k^{"})$$
$$\overline{L}_{ki} = (A_b)_i^{-1} \times (A_k + A_k^{"}) = (A_b)_i^{-1} \times A_k + (A_b)_i^{-1} \times A_k^{"} =$$
$$\overline{L}_{ki} = L_{ki} + L_{ki}^{'} = 0 + L_{ki}^{'}$$

$$\overline{L}_{kk} = \overline{A}_k^T \times (A_b^T)_k^{-1} = (A_b)_k^{-1} \times \overline{A}_k$$
$$\overline{L}_{kk} = (A_k^T + A_k^{"T}) \times (A_b^T)_k^{-1} = (A_b)_k^{-1} \times (A_k + A_k^{"})$$
$$\overline{L}_{kk} = (A_b)_k^{-1} \times (A_k + A_k^{"}) = (A_b)_k^{-1} \times A_k + (A_b)_k^{-1} \times A_k^{"} =$$
$$\overline{L}_{kk} = L_{kk} + L_{ki}^{'} = 1 + L_{ki}^{'}$$

here $(A_b^T)_i^{-1}$ - a column i of the matrix $(A_b^T)^{-1}$, $(A_b)_k^{-1}$ - a row k of the matrix $(A_b)^{-1}$ coinside with the coincide with the components of the basic solutions u_0 and $\overline{u}_0$ problems (11)-(13) with the substitution of k -column in matrix limits A_k by the column $\overline{A}_k = A_k + A_k^{'}$.

Statement 1. The evolution of solutions (6) at the indignation of the matrix columns in $\overline{A}_k = A_k + A_k^{'}$ coincides with the coefficients changes of the normal vector C^T separation(14) by changing the row k of the matrix A_b^T in $\overline{A}_k^T = A_k^T + A_k^{'T}$ according to the scheme of the basic matrixes method.

Statement 2. Each component connection of solutions vectors (4) u_0 and $\overline{u}_0$ (6) during the substitution of k -column in matrix limits A_k by the column $\overline{A}_k = A_k + A_k^{'}$ is described in the co-relation

$$\overline{u}_{0k} = \frac{u_{0k}}{1 + (A_b^{-1})_k \times A_k^{'}}, \ i = k \tag{20}$$

$$\overline{u}_{0i} = u_{0i} - \frac{u_{0k}}{1 + (A_b^{-1})_k \times A_k^{'}} \times \left[(A_b^{-1})_i \times A_k^{'} \right], \ i \neq k \tag{21}$$

thus the condition of preserving nondegenerate of solution is implementation of condition $1 + (A_b^{-1})_k \times A_k^{'} \neq 0$.

Consequence 1. During realization of calculations it ispossible to apply more general formula of calculation of the components of the vector solution (6) as a result of indignation of vector of column of matrix limits (the condition of the statement 2) is the next

$$\bar{u}_0 = u_0 - \frac{u_{0k}}{1+(A_b^{-1})_k \times A_k^{'}} \times \left[(A_b^{-1}) \times A_k \right], \tag{23}$$

or

$$\bar{u}_0 = u_0 - \frac{u_{0k}}{\bar{L}_{kk}} \times L_k^{'} ,$$

here

$$\bar{L}_{kk} = 1+(A_b^{-1})_k \times A_k^{'}, \quad L_k^{'} = A_b^{-1} \times A_k^{'}$$

In the statement 2 it is established, that $\bar{u}_{0i} = u_{0i} - \frac{u_{0k}}{1+(A_b^{-1})_k \times A_k^{'}} \times \left[(A_b^{-1})_i \times A_k \right], \ i \neq k$.

Let's make transformation of co-relation $\bar{u}_{0k} = \frac{u_{0k}}{1+(A_b^{-1})_k \times A_k^{'}}, \ i = k$ to the view that is close to the previous

$$\bar{u}_{0k} = \frac{u_{0k}}{1+(A_b^{-1})_k \times A_k^{'}} = u_{0k} - u_{0k} + \frac{u_{0k}}{1+(A_b^{-1})_k \times A_k^{'}} = .$$

$$= u_{0k} - \left[u_{0k} - \frac{u_{0k}}{1+(A_b^{-1})_k \times A_k^{'}} \right] =$$

$$= u_{0k} - \left[\frac{u_{0k} + u_{0k} \times (A_b^{-1})_k \times A_k^{'} - u_{0k}}{1+(A_b^{-1})_k \times A_k^{'}} \right] =$$

$$= u_{0k} - \frac{u_{0k} \times (A_b^{-1})_k \times A_k^{'}}{1+(A_b^{-1})_k \times A_k^{'}} = u_{0k} - \frac{u_{0k}}{1+(A_b^{-1})_k \times A_k^{'}} \times (A_b^{-1})_k \times A_k^{'} .$$

From these we have got the next

$$\bar{u}_0 = u_0 - \frac{u_{0k}}{1+(A_b^{-1})_k \times A_k^{'}} \times \left[(A_b^{-1}) \times A_k \right] \quad .$$

The statement is brought out from the previously established correlations, that connect the problems (11)-(13) and (14)-(15), inparticular, $u_0 = L_0$, $\bar{u}_0 = \bar{L}_0$, formulas (7), connections of lines, inverse and transpose matrixes in the scheme of basic matrixes method.

The Remark. The structure of formula $\bar{u}_0 = u_0 - \frac{u_{0k}}{\bar{L}_{kk}} \times L_k^{'}$ indicates that this presentation of the line equalization in parametric form, where u_0 - is an initial vector, $L_k^{'}$ - is a normal vector, $-\frac{u_{0k}}{\bar{L}_{kk}}$ - is the meaning of parameter of displacement along the vector $L_k^{'}$ (from u_0). The value of vector component $L_k^{'}$ and $-\frac{u_{0k}}{\bar{L}_{kk}}$ is influenced by the size of indignation in the column with the number k.

Here are the basic steps of the algorithm of indignant solution component (6) - as a result of changes of the column system (4) A_k by the column $\bar{A}_k = A_k + A_k^{'}$

1. We have got a known vector $u_0 = (u_{01}, u_{02}, ..., u_{0m})^T$, A_b, A_b^{-1} - line and inverse basic matrix (4).

2. Let conduct the substitution of the k - column in the matrix limits A_k by the column $\bar{A}_k = A_k + A_k^{'}$.

We find the vector

$$L_k^{'} = (L_{k1}^{'}, L_{k2}^{'}, ..., L_{km}^{'}) = A_b^{-1} \times A_k^{'},\ L_{kk}^{'} = \left(A_b^{-1}\right)_k \times A_k^{'},$$

$$\bar{L}_{kk} = 1 + L_{kk}^{'} = 1 + \left(A_b^{-1}\right)_k \times A_k^{'},$$

here $\left(A_b^{-1}\right)_k$ - k -row matrix A_b^{-1}.

3. Let us make the new solutions $\bar{u}_0 = u_0 - \frac{u_{0k}}{\bar{L}_{kk}} \times L_k^{'}$ according to the formula (23).

The steps of the algorithm of indignant solution component (6) - as a result of changes of the column system (4) A_k by the column $\overline{A}_k$

1. We have got a known vector $u_0 = (u_{01}, u_{02}, ..., u_{0m})^T$, A_b, A_b^{-1} - line and inverse basic matrix (4).

2. Let conduct the substitution of the k - column in the matrix limits A_k by the column $\overline{A}_k$.

We find the vector $\overline{L}_k = (L_{k1}, L'_{k2}, ..., L_{km}) = A_b^{-1} \times \overline{A}_k$.

3. We make a new solution based on (20) (21)

$$\overline{u}_{0k} = \frac{u_{0k}}{1 + (A_b^{-1})_k \times A'_k} = \frac{u_{0k}}{\overline{L}_{kk}}, \ i = k$$

$$\overline{u}_{0i} = u_{0i} - \frac{u_{0k}}{1 + (A_b^{-1})_k \times A'_k} \times \left[(A_b^{-1})_i \times A'_k \right] = u_{0i} - \frac{u_{0k}}{\overline{L}_{kk}} \times \overline{L}_{ki}, \ i \neq k$$

.

Let's illustrate the offered algorithm of the determination volumes of gross branch output in case of technological inter-branch changes on conditional data. Let the coefficients of technological matrixes of ecological-and economic model (3) have the following meanings:

$$A_{11} = \begin{pmatrix} 0.2 & 0.1 \\ 0.3 & 0.2 \end{pmatrix}, \ A_{12} = \begin{pmatrix} 0.1 & 0.2 \\ 0.1 & 0.2 \end{pmatrix}, \ A_{21} = \begin{pmatrix} 0.1 & 0.3 \\ 0.2 & 0.3 \end{pmatrix}, \ A_{22} = \begin{pmatrix} 0.2 & 0.3 \\ 0.3 & 0.1 \end{pmatrix},$$

the matrix of charges on maintenance of greenhouse gases emissions and vectors of the branch final producing and limitation after the extrass of greenhousegases accordingly:

$$C = \begin{pmatrix} 0.3 & 0.2 \\ 0.1 & 0.5 \end{pmatrix}, \ y_1 = \begin{pmatrix} 12 \\ 23 \end{pmatrix}, \ y_2 = \begin{pmatrix} 5 \\ 8 \end{pmatrix}.$$

Let's check implementation of the productivity condition for ecological and economic system in the case of selected numerical data. The block matrix A

$$A = \begin{pmatrix} 0.2 & 0.1 & 0.1 & 0.2 \\ 0.3 & 0.2 & 0.1 & 0.2 \\ 0.1 & 0.3 & 0.2 & 0.3 \\ 0.2 & 0.3 & 0.3 & 0.1 \end{pmatrix}$$

obviously is productive after the sufficient condition of the productivity of technological matrixes of balance models of Leontiev type. Except that the considered higher sufficient condition of the productivity model(3) – inequality $A_{21}(y_1 + Cy_2) \geq y_2$:

$$\begin{pmatrix} 9.76 \\ 11.27 \end{pmatrix} \geq \begin{pmatrix} 5 \\ 8 \end{pmatrix}.$$

Let us pass to the step by step algorithm realization

1. Find the solution of the original system (4) and the block inverse matrix technology:

$$u_0 = \begin{pmatrix} 38.17 \\ 60.43 \\ 32.67 \\ 30.62 \end{pmatrix}, A = \begin{pmatrix} 0.2 & 0.1 & 0.1 & 0.2 \\ 0.3 & 0.2 & 0.1 & 0.2 \\ 0.1 & 0.3 & 0.2 & 0.3 \\ 0.2 & 0.3 & 0.3 & 0.1 \end{pmatrix},$$

$$A^{-1} = A_b^{-1} = \begin{pmatrix} 1.25 & 3.125 & -3.125 & 0.625 \\ -11.25 & 6.875 & 3.125 & -0.625 \\ 8.75 & -8.125 & -1.875 & 4.375 \\ 5.0 & -2.5 & 2.5 & -2.5 \end{pmatrix}.$$

2. We assume that the that indignation in a model(6) tests the third column ($k = 3$): let's make the substitution of k-column in the matrix limit $A_3 = \begin{pmatrix} 0.1 \\ 0.1 \\ 0.2 \\ 0.3 \end{pmatrix}$ by the column $\overline{A}_3 = \begin{pmatrix} 0.2 \\ 0.2 \\ 0.1 \\ 0.1 \end{pmatrix}$. Find the vector $\overline{L}_k = (L_{k1}, L'_{k2}, \ldots, L_{km}) = A_b^{-1} \times \overline{A}_k$:

$$\overline{L}_3 = \begin{pmatrix} 1.25 & 3.125 & -3.125 & 0.625 \\ -11.25 & 6.875 & 3.125 & -0.625 \\ 8.75 & -8.125 & -1.875 & 4.375 \\ 5.0 & -2.5 & 2.5 & -2.5 \end{pmatrix} \cdot \begin{pmatrix} 0.2 \\ 0.2 \\ 0.1 \\ 0.1 \end{pmatrix} = \begin{pmatrix} 0.625 & -0.625 & 0.375 & 0.5 \end{pmatrix}$$

3. Let's find the solutions:

$$\overline{u}_1 = 38.17 - \frac{32.67}{0.375} \cdot 0.625 = -16.28$$

$$\overline{u}_2 = 60.43 - \frac{32.67}{0.375} \cdot (-0.625) = 114.88$$

$$\overline{u}_3 = \frac{32.67}{0.375} = 87.12$$

$$\overline{u}_4 = 30.62 - \frac{32.67}{0.375} \cdot 0.5 = -12.94$$

The got solutions coincide with the solutions got by the direct calculations what are easily seen and specify on a substantial change in functioning of auxiliary production, in particular, negative indexes require a change in the structure of technological matrixes of both basic and auxiliary spectrum of industries.

Conclusion

The necessity of taking into account of ecological factor in the modern system of further development of civilization causes the actuality of consideration of productive activity of society within a single socio-ecological-economic system. Thus, the important requirement of her existence is a necessity of the balanced interests each of the indicated subsystems. An effective instrument is a balance method and corresponding models worked out on its basis serve as for this purpose, model of taking into account of charges offered in particular in the article on realization of projects of reduction of greenhouse gases emissions. With the aim of its effective use the terms of the productivity are set and the algorithm of determination of volumes of the gross branch producing is offered in case ofchange of technological branch structure. Further researches should be performed towards the including of additional economic and ecological limitations, and change of classic original assumptions in relation to the technological structure of the offered model.

Bibliography

[KYOTO] KYOTO PROTOCOL TO THE UNITED NATIONS FRAMEWORK CONVENTION ON CLIMATE CHANGE http://unfccc.int/kyoto_protocol/items/2830.php

[Leontief, 1986] Leontief, Wassily W. Input-Output Economics. 2nd ed., New York: Oxford University Press, 1986.

[Lyashenko, 1999] Lyashenko I.M. Ekonomiko–matematychni metody ta modeli stalogo rozvytku. – K.: Vyshha shkola, 1999. – 236 s.(in Ukr.)

[Khrushch, 2007] Khrushch L. Z. Ekoloho-ekonomichni vyrobnychi funktsii (pobudova ta zastosuvannia u prognozuvanni) // Modeluvannia ta informaciini systemy v ekonomici. Zbirnyk naukovykh prac. – Vypusk 76. – Kiev, 2007. – S. 115-165. (In Ukr)

[Lyashenko, Khrushch, 2008] Lyashenko I.M., Khrushch L. Z. Uzahalnennia modeli “Vytraty-vypusk” Leontieva-Forda na mnozhynu tekhnolohichnykh sposobiv // Ekonomika: problemy teorii ta praktyky/ Zbirnyk naukovykh prac. – Vypusk 239. Tom 3. – Dnipropetrovsk: DNU, 2008. – № 4. – S. 240-244. (In Ukr)

[Khrushch, Korzhevska, 2012] Khrushch L. Z. , Korzhevska O.P. Rozshyrennia mizhhaluzevoi ekoloho-ekonomichnoi modeli Leontieva-Forda // Biznes inform. Naukovyi zhurnal. – № 3, 2012. – S. 75-78. (In Ukr)

[Kudin, 2007] Kudin V.I., Lyashko S.I., Hritonenko N.V., Yatsenko Yu.P. Analiz svoystv lineynoy sistemyi metodom psevdobazisnyih matrits // Kibernetika i sistemnyiy analiz. – 2007. – N 4. –.S. 119–127.

[Lyashenko, 2009] Lyashenko I.M., Onyshhenko A.M. Pryami ta dvoyisti balansovi modeli „vytraty-vypusk” // Ekonomichna kibernetyka. – 2009. – #1-2. – S. 14–18. (in Ukr.)

[Mechanisms] The Mechanisms under the Kyoto Protocol: Clean development mechanism, joint implementation and emissions trading http://unfccc.int/kyoto_protocol/mechanisms/items/1673.php

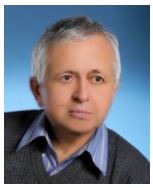
[Onyshhenko, 2011] Onyshhenko A. M., Lyashenko I.M. Metodologiya matematychnogo modelyuvannya ekonomiko–ekologichnoyi vzayemodiyi v umovax realizaciyi Kiotskogo protokolu // Ekonomichna kibernetyka. – 2011. – #4–6(70-72) – S. 17-26. (in Ukr.)

[Voloshin, 2010] Voloshyn A., Maschenko S. Individually optimal principles of distribution of greenhouse gas emission quatas // “Information Models of Knowledge”, ITHEA, Kiev-Sofia, 2010, 470 p. - P. 209-214.

[Voloshin, Kudin, 2013] Voloshin O.F., Gnatienko G.M., Kudin V.I. Poslidovniy analiz variantiv: Technologiy ta zastosuvannya // Monografiya.-K.: Stilos.-2013,-304s (in Ukr.)

[Voloshin, Kudin, 2015] Voloshin O.F., Kudin V.I. Poslidovniy analiz variantov v zadachach isledovaniya i proektirovaniya system // Monografiya.-K.: Izdatelsko-poligraficheskiy centr "Kievskiy universitet" - Naukova dumka, - 2015. - 351s. (in Rus.)

Authors Information



Oleksii Voloshyn *- Dr. Sc., Professor, Kyiv National Taras Shevchenko University, Faculty of Computer Science and Cybernetics ; e-mail: olvoloshyn@ukr.net*

Research interests*: decision-making theory, decision support systems, information technology, expert systems, mathematical economics, fuzzy analysis, e-learning*

Vladimir Kudin *- Dr. Sc., Professor, Kyiv National Taras Shevchenko University; Faculty of Information Technology; e-mail: V_I_Kudin@mail.ru*

Research interests*: optimization techniques, systems analysis, computational method, decision-making, mathematical economics*

Andrii Onyshchenko *– Dr. Sc., Professor, Kyiv National Taras Shevchenko University; Faculty of Information Technology; e-mail: onyshchenko@yandex.ru*

Research interests*: mathematical economics, eco-economic modeling, optimal control, data maining*

Lesya Khrushch *- **Dr.**, Associate Professor of Information Technology Department, Vasyl Stefanyk Precarpathian National University, IvanoFrankivs'k,Ukraine; Faculty of Mathematics and Informatics; Lessja2501@gmail.com, lesia.khrushch@pu.if.ua*

Research interests*: mathematical economics, optimization techniques, information technology,* web programming

BENCHMARKING AS A RESEARCH METHOD FOR OBTAINING SYSTEM KNOWLEDGE OF THE STATE INSTITUTIONAL DEVELOPMENT

Givi Bedianashvili

Abstract: *The article looks at current central issues of modernization of the country's socio-economic system institutional structure within the context of the EU, Georgia Association Agreement, and conceptual and methodology benchmarking issues related with the same. For monitoring of the institutional development, possibilities are shown for the use of current international indicators including the economic freedom index.*

Key words: *Globalization, EU, benchmarking, modernization of institutional structure*

ITHEA Keywords*: I.2 Artificial Intelligence; D.2.2 Design Tools and Techniques*

Introduction

European orientation of Georgia stresses the urgency of a systematic and effective consistent integration of our country in the EU and Euro-Atlantic structures. At the same time, there is certainly the expectation in the Georgian society that the new European potential will be used for the country's socio-economic development in the modern and global conditions.

By reference to the characteristics of the EU, as a socio-economic system, and due to Georgia's current and future multilateral correlation positions in the framework of the Association Agreement [Georgia-EU Association Agenda, 2014], first of all, it is important for our country to identify challenges and to develop and implement relevant social and economic policy based on the same.

In our opinion, proper understanding of the issues and their consistent solution is possible only with the institutional approach. It is transformation of the institutions which is a really important foundation for Georgia to benefit from the EU and its business environment potential [Bedianashvil, 2014]. We believe that the urgency of the institutional approach casts no doubt as world-known researches have proved that predominantly institutional factors were involved in the formation of different levels of development between countries during long historical periods.

Literature review

The most important issue is that the process of creating institutions (institutionalization) must be systemic and synchronous in the dynamics (time). **It should be noted that institutionalization ultimately means changing of spontaneous and experimental behavior to regulated, expected and predictable behavior.** In addition, due to the positions of effective functioning of institutions in the process includes (theoretically and methodologically must be separated) several interrelated stages, such as: **Development (identification) of demand** which requires organized joint **(not spontaneous)** activities from humans; **Development of standards and regulations in the process of natural social and economic interaction; application of sanctions imposed for** the **promotion** of such **regulations**; Determination of **respective status and roles** for each member of the institute; **Establishing formal and informal organizations,** which will **regulate the institute activities.** Thus, it can be said that in case of institutionalization, institutions (primarily economic institutions, but not only) are “cultivated” locally and not imported from outside i.e. "transplanted" (see [Полтерович, 2001] for economic transplantation of institutions). As the transformation practice of post-communist period shows, import of institutions to other countries, when an imported institute evolved for many years and functioned efficiently in a qualitatively different environment other than the "importer" country, in most cases, tends to yield no desired results, which is due to a number of reasons (the complex analysis of the same is provided in the paper [Papava, 2014], and some important aspects of institutional analysis of economic transformation is explored in the paper [Papava and Khaduri, 1997], in this regard the interesting papers are as well [Asatiani, 2014] and [Tokmazishvili, 1997]).

The key to the issue of institutionalization in our view is an adequate measurement of the development level of the institutional structure (the main structural component which ensures effective functioning and complexity of dynamics of the country's socio-economic system) [Bedianashvili, 1995]) to provide effective monitoring of development and functioning of institutions, as well as to determine a specific country’s state with regard to other countries (benchmarking).

It logically raises the question about the EU's business environment, also the compatibility of institutional structures of the socio-economic systems of the EU countries and Georgia. Therefore, it is of great importance, if necessary, to determine certain directions for institutional changes, and their practical and systemic implementation.

As mentioned above, the mechanical copying of institutions is not effective for the country. Based on the logic of systemic institutionalization, the first phase should include identification of needs and relevant problems, which could be solved with an institutional approach (arrangement). In this respect, we think it is recommended to consider the areas where the country is experciencing a significant lag, which essentially prevents the country's socio-economic development and probably business operations.

For Georgia's institutional benchmarking purposes, we will choose Switzerland (a European country with effective institutional arrangement, although not a EU member) and Estonia. The efficiency of the institutional structure, we think, can be indirectly measured by the country's sustainability level in extreme (crisis) periods (generally, with the economic theory instrumentals for measurement of institutions, previously argued by R. Matthews [Matthews, 1986]). Indeed, Switzerland shows quite good dynamics during the 2008 crisis and the following periods, and Estonia also has the favourable characteristics upon the post crisis stage (see, for example, the statistical data [The World Bank, 2017]).

Here, it is interesting to bring up a question: is economic development level impotant for the quality of its economic dynamics sustainability? In general, of course, it is, but not decisive. The effectiveness of the institutional structure is more significant, we believe. This is clearly evidenced in the Euro Zone and some countries, for example, USA, China, Turkey, Japan (the statistical data applied [The World Bank, 2017]).

Data and methodology/Analyses

As for the institutional structure, as the country's socio-economic development sustainability ensuring factor, as noted above, in our opinion, the key is identification and monitoring with, for example, the Index of Economic Freedom, which has been developed (calculated) annually by **The Wall Street Journal** and Research Center **Heritage Foundation** since 1995 in the light of most countries of the world. [Index of Economic Freedom, 2017]. Indices are published once a year and as of 2017 data cover 186 countries.

Economic Freedom Index is designed to reflect important institutional characteristics required for any country's economic growth. The indicator includes dozens of indicators, which are united in the following 12 groups (quantitative and qualitative factors): 1.**Property Rights;** 2.**Government Integrity;** 3.**Judicial Effectiveness;** 4.**Government Spending;** 5.**Tax Burden;** 6.**Fiscal Health;** 7.**Business Freedom;** 8.**Labor Freedom;** 9.**Monetary Freedom;** 10.**Trade Freedom;** 11.**Investment Freedom;** 12.**Financial Freedom**. The indicators are placed in 4 groups: Law and Order (1; 2; 3), Government Size (4; 5; 6), Regulatory Efficiency (7; 8; 9), Markets Openness (10; 11; 12). Each of the above score is measured from 0 to 100. For this purpose 100 corresponds to maximum freedom, and 0 - minimum. Each factor in the above ten is included with the equal weight into the integral indicator (economic freedom index), which is calculated as arithmetic average.

According to the quantitative values of integral indicators, the world countries are divided into five respective groups with their Economic Freedom levels: **Free**, a range of 80-100; M**ostly Free,** a range of 70-79.9; **Moderately Free,** a range of 60-69.9; **Mostly Unfree**, a range of 50-59.9 and **Repressed**, a range of 0-49.9.

Discussion of results

According to the 2017 data, Georgia belongs to the group of countries with mostly free rating as it moved from the last year's 23rd place to 13th in the world, while among 44 European countries it already takes the 5th place. Pursuant to the **Economic Freedom Score** (76), it stands alongside with the countries such as Taiwan (76.5), the United Kingdom (76.4), Luxembourg (75.9) and the Netherlands (75.8). For example, the United States is rated 17, while the corresponding score is 75.1. As for the top five countries, they are: Hong Kong (89.8), Singapore (88.6), New Zealand (83.7), Switzerland (81.5), Australia (81.0). It should be noted that the world average score is 60.9 (which corresponds to the level of moderately free score), and the average regional (European) is 68.0 (also moderately free country level with freedom level higher than moderate world average score). Thus, Georgia according to its Economic Freedom integral indicator takes higher position in the region (in Europe) and the world.

For years, unfavorable lower level of systemic institutionalization has been observed in Georgia, although some subsystems of institutions were evolving quite rapidly. It should be noted that according to 2017 data (Fig.1), Georgia's state has significantly improved compared to previous years, but still it lags behind the benchmark and leading European countries in terms of development of market openness (investment freedom and financial freedom), also Law and Order (property rights, government integrity and judicial effectiveness) institutions.

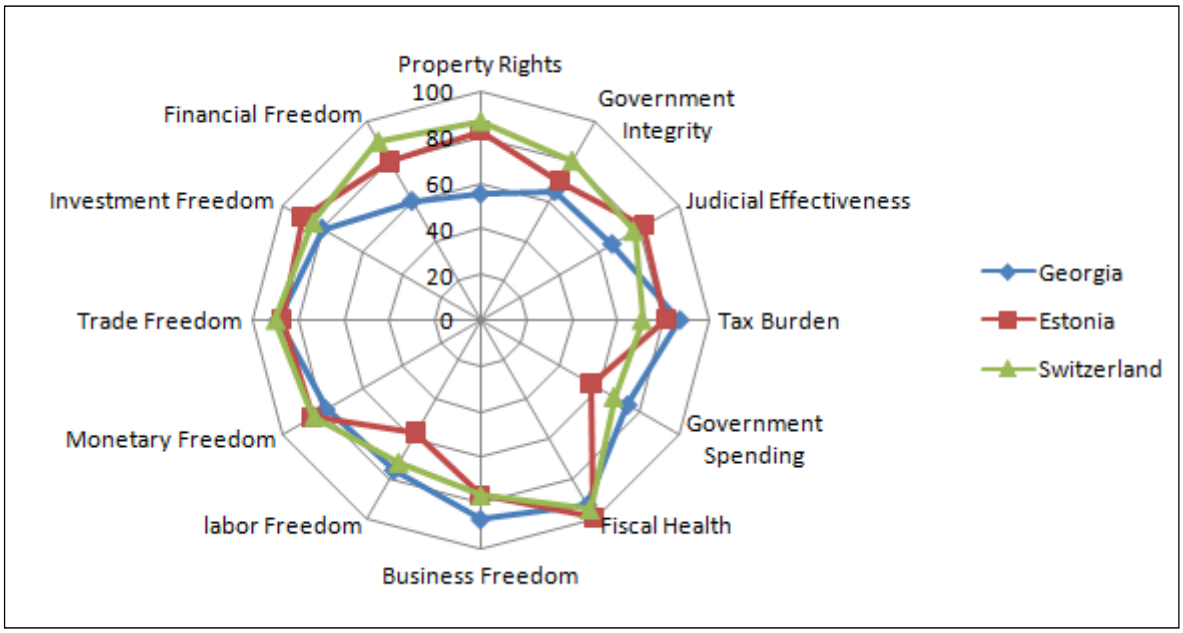


Figure 1. The institutional development complexity level of Switzerland, Estonia and Georgia

As noted in the 2017 research, the "open" market policy in Georgia, accompanied by a relatively low tax rates and a higher level of regulation, has promoted the development of trade and investment growth. Monetary stability and "healthy" fiscal policy has contributed to the country's macroeconomic stability. Despite the positive changes, profound and rapid institutional reforms are still urgent, especially in terms of strengthening the judicial independence. This in turn ensures the acceleration of country's development.

Conclusions

We believe that a gradual reduction of the above mentioned lag should be one of the main directions of systemic institutionalization of our country in the coming years. This will allow to fully use modern European institutional structure and the EU business potential for our country's socio-economic development. The implementation of a complex of activities has already been included in Georgia-EU Association agenda [Georgia-EU Association Agenda, 2014], as well as in the socio-economic development strategy of Georgia (2020) [The socio-economic development strategy of Georgia, 2014].

One can confidently say that Georgia is facing fundamentally new challenges, and in our opinion, the key for our country, which is a part of Europe, is to manage and establish closer ties with the European countries within globalization processes taking place in the world in parallel with mutually beneficial and pragmatic cooperation with other countries. In addition, the focus must be placed on strengthening of objectively positive aspects of our own state social and economic system, and transforming of negative and weak elements of our society with a consistent and clear, acceptable and painless for the society methods and forms. At the same time, we should take into account Georgia's own national and world civilization centuries-old experience of the past and best samples and models of modern periods.

Bibliography

[Georgia-EU Association Agenda, 2014] http://www.mfa.gov.ge .

[Bedianashvili, 2014] Bedianashvili G. The country's socio-economic development, institutions and systemic institutionalization. International scientific practical conference "Actual problems of economy and economic development of science." Ivane Javakhishvili Tbilisi State University, P. Gugushvili Institute of Economics. Tbilisi, 2014.- pg. 158-161.

[Полтерович В.,2001] Полтерович В. М. Трансплантация экономических институтов. «Экономическая наука современной России», 2001, No. 3.С.24 -50.

[Papava V., Conventional Economix. P. Gugushvili Institute of Economics. Tbilisi, 2011.

[Papava and Khaduri, 1997]. Papava V., Khaduri N. On the Political Economy of the Post-Communist Transformation. An Institutional Analysis Problems of Economic Transition. Vol. 40. No 6. October 1997, pp. 15-34

[Asatiani, 2014]. Asatiani R. Where is Georgia heading for? Conceptual analysis of socio-economic development. Tbilisi: Siakhle, 2014. p. 78-112.

[Tokmazishvili, 1997]. Tokmazishvili M. Socio-economic reforms: Short macro economic and institutional analysis. Economy and Economic Policy Research Center of Caucasus, Tbilisi, 1997.

[Bedianashvili, 1995]. Bedianashvili G. State, Power Stucture and Socio-Economoc Reforming of Society. P. Gugushvili Institute of Economics of the Georgian Academy of Sciences. Tbilisi: Metsniereba, 1995. .P. 5.

[Matthews, 1986]. Matthews R The Economics of Institutions and Sources of Growth. Economic Joutnal. 1986. Vol. 96. P.903-910.

[The World Bank, 2017] The World Bank . http://data.worldbank.org/indicator/NY.GDP.MKTP.KD.ZG/countries/EU-EE?display=graph

[Index of Economic Freedom,2017] Index of Economic Freedom. http://www.heritage.org/index/pdf/2017/book/index_2017.pdf

[The socio-economic development strategy of Georgia, 2014] The socio-economic development strategy of Georgia, Georgia 2020. http://www.mof.ge

Authors' Information

Givi Bedianashvili *– European University, Director of the Institute for the Research of Economic and Social problems of Globalization. Editor in chief of journal "Globalization and Business". Theories and Applications; ITHEA Editor in chief. 76 D. Guramishvili ave,Tbilisi, 0141, Georgia; e-mail: g.bedianashvili@esu.edu.ge*

Major Fields of Scientific Research*: System analysis, Institution economics, Econometric, methods of socio-economic policy, global business*

SEARCH IDEAS FOR BUSINESS ALUMNI

Kuzomin O.Ya. Tkachenko B., Tkachenko M.

***Abstract**: The article considers some methods and models for searching ideas for startups and business projects for the realization of the goals of the project TEMPUS544521Tempus-1-2013-1-De-Tempus-Shmes "EANET". An algorithm for the search of new technical graduates for a business organization is presented.*

***Keywords**: Data Base (DB), BackUp Keeper, Archive, Restore, Full/Differential/TRN-Log backup, Emergency Management Australia (EMA), Garland of associations and metaphors*

Introduction

There are two trends that can characterize society development: first – continuous increase in human needs for new technologies, new services and products, tools, etc., and needs are doubled in less than seven years; second – a significant lag the pace of development of new technologies, products and services.

Today we need not any products and services, but only high-quality and competitive products and services, which have rates of 20-25% higher than existing products and services.

Scientific and engineering community is constantly in search mode of something new that can satisfy the particular needs of humanity. At present there is the need for new knowledge as they also grow old together with the corresponding already produced products and processes, and therefore obsolete and qualified specialists.

The pace of aging knowledge necessitates regular and systematic updating of knowledge, i.e. competence of professionals and workers. Assuming that the doubling of knowledge necessary for any profession occurs within 12 years, then people that started to work immediately after graduating from the university at 23 years will have only half of required knowledge at 35, 47 - a quarter, and 59 - one-eighth part of the knowledge. There is the need for regular, systematic training of specialists at all levels and in all branches of science and technology.

Relevant is the development of creative and innovative start from experts in particular business alumni (university graduates who can grow their business and to participate in market-society).

New ideas, products, technical solutions, services etc. for business alumni are encouraged to develop a modified method of garlands associations.

Main part

The task of forming a list of possible solutions to the problem in the decision-making process has traditionally solved by non-directional search (trial and error). Its solution in terms of lack of information, inability to use logic and the need to find new solutions to the original requires the creation and application of methods of activating the creative process or automation and modernization of methods previously considered non-algorithmic and formal; - methods of psychological thinking activation [2].

Despite the fact that in general the process of searching for new ideas for methods of psychological thinking activation difficult lack of ability to manage this search, support tools to find new ideas are widespread [4]. Among these tools can be called "matrix of ideas" by Lebedev Studio [3], «Ideas2.0», «Idea Generator» TDB Spesials Project [7], «Creativity Machine» of Steven Tyler [5], «IdeaFinder+» that is developed on the basis of linguistic processor of A. Baryshnikov and the others. To activate the creative process is also used audiovisual stimulation device (called Mind-machine), leading recipients in a state of deep relaxation, concentration, or an altered state of consciousness [6].

For methods garlands associations and focal objects conditional complexity of finding solutions to the increasing complexity of the problem to be solved is lower than for the method of trial and error, and methods of psychological activation as a whole, but higher than for the methods of directed search [4] (TRIZ [1], functional and physical method). In addition, methods for focal objects and garlands associations may be made of different control actions to increase the efficiency of the search for new solutions. In addition to automating the process of building a simple associative chains, implementation of algorithms of these methods will enhance the uniqueness found associations to solve inventive problems compared with the methods of random search.

To study the method of garlands associations to generate solutions to for solving inventive problems are encouraged to use a modified version of the method.

Methods for solving the problems of searching technical ideas for business alumni depend on psychodiagnostics in psychosomatics, the features of the subject area in which the task is being solved, and the requirements that the user sets for solving the search problem. The characteristics of the domain in terms of methods of solution can be characterized by the following parameters:

-the size that determines the amount of space in which the solution is to be sought;

-the changeability of the region, characterizes the degree of variability of the region in time and space (here we will allocate static and dynamic regions);

-the completeness of the model describing the region characterizes the adequacy of the model used to describe the given region. Usually, if the model is not complete, several models are used to describe the region, complementing each other by reflecting different properties of the domain;

The certainty of the data about the problem being solved characterizes the degree of accuracy (inaccuracy) and completeness (incompleteness) of the data. Accuracy (error) is an indicator that the subject area is described from the point of view of the tasks being solved by accurate or inaccurate data; Under the completeness (incompleteness) of data is meant the sufficiency (insufficiency) of input data for unambiguous solution of the problem.

The user's requirements for the result of the task solved by searching can be characterized by the number of decisions and the properties of the result and (or) the method of obtaining it. The parameter "number of decisions" can take the following basic values: one solution, several solutions, all solutions. The "properties" parameter specifies the restrictions that the received result or method of obtaining it must satisfy. So, for example, for a system issuing recommendations for the treatment of patients, the user can specify the requirement not to use some medicine (due to his absence or because it is contraindicated to this patient). The parameter "properties" can also determine such features as the decision time ("not more than", "time range", etc.), the amount of memory used to obtain the result, the indication of the mandatory (impossible) use of any knowledge (Data), and the like. Thus, the complexity of the problem, determined by the above set of parameters, varies from simple problems of small dimension with unchangeable definite data and the absence of constraints on the result and the method of obtaining it to complex large-dimensional problems with variable, erroneous and incomplete data and arbitrary limitations on the result and the method for obtaining it. From general considerations it is clear that no one method can solve all problems. Usually some methods surpass others only in some of the listed parameters.

The methods discussed below can work in static and dynamic problem environments. In order for them to work in dynamic conditions, it is necessary to take into account the lifetime of the values of the variables, the data source for the variables, and also to provide the possibility of storing the history of the values of the variables, modeling the external environment, and operating time categories in the rules.

The existing methods of solving problems used in expert systems can be classified as follows:

- methods of searching in one space - methods intended for use under the following conditions: small-dimensional areas, model completeness, accurate and complete data;
- search methods in hierarchical spaces - methods designed to work in areas of large dimension;
- search methods for inaccurate and incomplete data;

- search methods using several models designed to work with areas for which adequate description of one model is not enough.

It is assumed that the listed methods, if necessary, should be combined in order to allow solving problems whose complexity increases simultaneously by several parameters.

Currently, there are many methods for finding ideas. With a certain degree of conventionality, they can be divided into three groups:

1. Methods of psychological activation of thinking.
2. Methods of systematic search.
3. Methods of directed search.

Each of the methods is designed to facilitate the search for a solution to a creative problem in comparison with the so-called "trial and error" method, which is usually used by a person. The expediency of applying a method belonging to a particular group, in particular, depends on the complexity of the problem being solved.

To solve relatively simple problems, it is expedient to use methods relating to the first two groups. Directional search methods were specially created to solve complex problems and, in spite of the complexity of these methods, their application in this case is justified. The use of direct search methods for simple tasks may not be appropriate because the complexity of the methods themselves will be higher than the complexity of the problem being solved, or because these methods are not suitable for solving such problems.

Proposed solution

First of all, to find ideas it is recommended to build garlands of associations. The most acceptable form of representation of the generalized domain model of the solution can be given in the form of a triple

$$\langle A,P,G \rangle ,$$

where A – the final alphabet of the system; P – he final set of rules for which garlands of associations are drawn up; G – a finite set of associations' garlands.

At the first stage of building a garland of associations for the object chosen for analysis, a set of words characterizing its properties is determined (these can be, for example, adjectives and verbs). The second stage defines a set of random objects that form the knowledge base for building associations' garlands. For these objects, sets of characteristics characterizing them are also determined. The third stage is the compilation of a chain of associations. A property is selected for the analyzed object, and an object with the same property as the original object is selected from the list of random objects

according to a particular rule (the frequency of occurrence of the property, the uniform random distribution or another). After that, the following element is attached to the chain by the same rule. The stage ends when the specified length of the chain is reached, or it is impossible to add a new element to the garland. The fourth stage is the generation of new ideas. Elements of a garland of associations and their signs join the analyzed object. At the fifth stage, the most rational ideas are selected for further use.

So, the algorithm of the method of garlands of randomness and association has such stages of implementation:

1. Definition of synonyms for the object.
2. Select random objects.
3. Drawing up combinations of elements of a garland of synonyms for the object and

elements of a garland of random objects. Combinations are made up of two elements by attempting to merge each a synonym of the object under consideration with each random object.

4. Drawing up a list of signs of random objects. The signs of randomly selected objects with the greatest possible number of symptoms for a limited time (minutes). The success of the search into a large extent depends on the latitude of coverage of the characteristics of random objects, therefore it is advisable to list both basic and secondary signs. For convenience, a table of characteristics is drawn up, one column of which is indicated by random objects, and in the other (on the contrary) - the signs of these random objects.
5. Generating ideas by alternately joining the technical object and its synonyms for signs of randomly selected objects.
6. Generating garlands of associations.

Alternately, from the signs of random objects identified in the fourth step, generate garlands of free associations. For each of the individual characteristics they can be virtually unlimited length, so the generation should be limited in time or number of elements of the garland.

Note: If the generation of associations' garlands is collectively conducted, then each member of the collective is engaged in this independently.

7. Generating new ideas.

Elements of garlands of technical object synonyms are trying to attach elements of associations' garlands.

8. Choosing an alternative.

At this step, the question is solved - to continue generating garlands of associations or they are already enough to select useful ideas. Note: if, according to preliminary estimates, there are few such ideas, one can continue creation of garlands of associations, starting with some new element of garlands, created in the sixth step and acting in a similar way.

9. Evaluation and selection of rational ideas.

Among the many irrational, trivial and even ridiculous ideas, as a rule, always are original and rational. If for a short time you can find several dozens of solutions, it is quite satisfactory a situation in which at least several variants of variants seem useful.

10. Choose an option

Note: Often talk about "optimal" options, but forget to indicate, relative to whom or what they are optimal. Therefore, it is advisable to use the term "rational solution"

Mathematical model of the design object

Recognition when designing complex systems is a multi-level process. It is characterized by the successive stages provided by the procedural model, at each of which the projected system receives a description of the structure of the elements of the object and the parameters. In this connection, the description of the design object can be called stratified, which evolves from a "short" stage of the procedural model to "unfolded" in the lower levels.

To use a computer, the description of the design object should be the nature of mathematical models.

For any decision situation, they are a set of relationships that connect actions (replaceable, the value of which is selected by the person who makes the decision LPR) that is controlled, and the parameters of the given task with the original removable (changeable, dependent on the type of actions that control).

In a meaningful sense, the description of the design object in the form of a mathematical model should contain the following components and rules:

1. $\mathbb{Z}$ – purpose of functioning.
2. $\{\mathrm{Elem}\}$ – set of elements making up the system.
3. $\{T(t_\tau)\}$ – set of time elements.
4. $\{P(P_i)\}$ – a set of characteristics that characterize the system as a whole.
5. $\{P_{el}\}$ – a set of characteristics that characterize the elements of the system.

6. $\{Sost\}$ – set of states of elements in a given time interval.

7. $H(\mathbb{Z})$ – rule for ordering the change of states in the course of achieving the goal.

8. $\{Q\}$ – rule of relations between all elements of the system;

9. $F=\{P_{el},P(P_i)\}$ – mathematical relations schemes that describe, features

Elements and featured systems.

10. $\{P_{sred}\}$ – a set of characteristics that determine the interaction of the system with the environment.

The system will be defined if all of the above sets and rules 7 and 9 are defined.

A lot of goals, attributes and elements are best represented in the form of graphs.

A plurality of states includes a defined set of characteristic values of a system, subsystem, or elements at a time point t_τ.

Individual elements or the entire system for a given time $t_0,\ldots,t_\tau,\ldots,t_k$ a certain number of times passes from one state to another. A single transition is an elementary operation:

$$Op_0 = Sost^\tau \succ Sost^{\tau+1},$$

where $Sost^\tau$ – position; Op_0 – elementary operation; $\succ$ - order-of-charge sign.

It assumes that the operation is defined if the specified state, the final state, the order of state changes of the system, which can be described by the differential equations, finite automata, probability automata, Markov chains, Boolean functions, predicate functions, are defined for it.

The interaction of elements is determined by the relationships

The design process as a transition from one description to another can be expressed by a relationship:

$$\Pi_0 = \widetilde{\Pi}_1 \to \widetilde{\Pi}_2 \to \ldots \to \widetilde{\Pi}_i,$$

where Π_0 – means the design process, $\widetilde{\Pi}_1 \to \widetilde{\Pi}_2 \to \ldots \to \widetilde{\Pi}_i$ – description of the design object at different stages of design at different stages of its development.

Description $\widetilde{\Pi}_1$, of the determinants achieved with its creation and use, is called the target: $\widetilde{\Pi}_1 = \mathbb{Z}_0$.

Description $\widetilde{O\Pi}_2$, that gives an idea of the idea of its technical solution, is called conceptual: $\widetilde{\Pi}_2 = (\mathbb{Z}_0, P)$.

Description $\widetilde{\Pi}_3$ hat gives a representation in the function of an object, let's call it functional:

$$\widetilde{\Pi}_3 = \left(P_{sred}, H, Sost\right).$$

Mathematical models that relate to the structural description of a system include many elements that represent the system $\{Elem\}$; set of characteristics that characterize the elements $\{P_{el}\}$; set of connections between all elements of the system $\{Q\}$:

$$\widetilde{\Pi}_4 = \left(Elem, P_{el}, Q\right)$$

The dynamic description includes mathematical models based on a set of characteristics that determine the interactions of the system with the environment $\{P_{sred}\}$, at a multitude of times $\{T\}$ and mathematical schemes that describe the relationship between the characteristics of elements and the characteristics of the system: $\widetilde{\Pi}_5 = \left(P_{sred}, T, F\right)$.

A description that defines parameters for an object is called parametric. It includes many parameters: $\widetilde{\Pi}_6 = \left(P_1, P_2, \ldots, P_i\right)$.

In this section, we consider the general representation of the description and the mathematical model of the design object. In the future, it will be specified with respect to individual procedures and operations.

The method of garlands of associations and metaphors - is a heuristic method of technical creativity, which is the development of the method of focal objects.

The method of garlands of associations and metaphors includes the following procedures:

1. Definition of synonyms of the object, as a result of which a garland of synonyms is formed (for example desk-bureau-desk -...);

2. Choice of random nouns, with the help of which a garland of random nouns is generated (for example, pencil-chair -...);

3. Combining all the elements of a garland of synonyms with each element of a garland of random nouns. Some of the combinations represent ideas for solving a problem (for example, a table like a pencil-table in the form of a chair -...);

4. Compilation of a list of signs in the form of adjectives for each element of a garland of random nouns (clause 2). These lists are garlands of signs (for example, pencil: wooden-automatic -...; chair:..);

5. Combining elements of a garland of synonyms with elements of garlands of features, as a result of which ideas may appear to solve the problem (for example a table - wooden (in the form of a tree), automatic (automatic increase);

6. Generation of garlands of free associations.

The starting point is each element of a garland of signs. Number of garlands of free associations is equal to the number of all elements of garlands of signs. Garlands of free associations are formed by repeatedly stating the question "What does the word ...?" Resemble. The answer to the question obtained on the basis of the association is a new element of the garland, which is the starting point for re-stating the question (for example: "What does the word" green "refer to? -On the grass," What ... " Grass "? - About" field ";" About what ... "field"? - About "cold", etc.

A garland of associations contains: grass - field - cold ...);

7. Combining elements of a garland of synonyms with elements of garlands of free associations, resulting in new ideas for solving the problem;

8. Evaluation of the need to continue associations, based on an analysis of all received in paragraphs. 1-7 ideas and determining their sufficiency. In the latter case, the transition to paragraph 9 is carried out, otherwise with the initial beginning of the elements of free associations, secondary garlands are generated (through free associations), elements of which are combined with elements of a garland of synonyms, resulting in new ideas;

9. Evaluation and selection of rational ideas. It is recommended to conduct by classifying all ideas into irrational (unfit, bad), semi-rational (attractive), rational (good). Irrational ideas are discarded; Rational form the core for choosing the optimal variant, and the semi-rational (which are attractive but have obvious drawbacks) are analyzed again, and then included in the list of irrational or rational ideas;

10. Choosing the best option. A stage performed by some optimization method, for example, expert assessments.

The method of garlands of accidents and associations is the development of the MFO method. Its author is Henry Bush. Through associations, this method allows finding a large number of hints for the researcher. From the method of focal objects, it differs in that it gives a large number of combinations of a focal object with random ones. The expansion of combinations of concepts is achieved using the synonyms of the object.

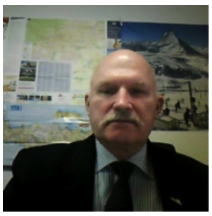
Acknowledgement

The paper is published with financial support by the project ITHEA XXI of the Institute of Information Theories and Applications FOI ITHEA (www.ithea.org) and the Association of Developers and Users of Intelligent Systems ADUIS Ukraine (www.aduis.com.ua).

Bibliography

1. Altshuller G. S. Creativity as an Exact Science / G.S. Altshuller. — 2 edition – Petrozavodsk: publishing "Scandinavia", 2004 – 208 pages.
2. Dvoryankin A. M., Polovinkin A. I. Automation exploratory design: artificial intelligence in computer-aided design / A. M. Dvoryankin and others under redaction of A. I. Polovinkin. – Moscow: Radio and Communications, 1981 – 344 pages
3. Matrix of ideas [electronic resource], – [2009]. – Access mode: http://www.artlebedev.ru/tools/matrix/.
4. Search methods and ideas of innovation [electronic resource], - [2004], – Access mode: http://www.inventech.ru/pub/methods/.
5. Program Creativity Machine dreams up, invents and makes openings [electronic resource] – [2004]. – Access mode: http://www.membrana.ru/articles/inventions/2004/01/26/212000.html.
6. Light and sound stimulation [electronic resource], – [2008]. – Access mode: http://www.medsport.spb.ru/.
7. The Director Bureau Special Projects [electronic resource], – [2009]. – Access mode: http://www.tdbspecialprojects.com/.

Authors' Information



Prof. Dr.-hab. Oleksandr Kuzomin *– Informatics chair of Kharkiv National University of Radioel ectronics; Kharkiv, Ukraine;*

e-mail: kuzy@daad-alumni.de

tel.: +38(057)7021515

Major Fields of Scientific Research*: General theoretical information research, Decision Making, Emergency Prevention, Data Mining, Business Informatics.*

TABLE OF CONTENTS