# On Digital Economy Issues Looking From the Information Systems Viewpoint

Manfred Sneps-Sneppe
Ventspils University College
Ventspils, Latvia
manfreds.sneps@gmail.com

Dmitry Namiot
Lomonosov Moscow State University
Moscow, Russia
dnamiot@gmail.com

Maris Alberts
Institute of Mathematics and Computer Science
Riga, Latvia
alberts@latnet.lv

***Abstract*—This paper is devoted to the digital transformations and digital economy programs. In particular, we are considering programs for the transition to a digital economy in the Russian Federation. It is obvious that telecommunications represent the basis for digital transformations. The paper discusses the software development and mathematical modeling issues relating to the program of Digital Economy, in particular, "Digital Economy of the Russian Federation". The main focus of our review is an information infrastructure. As examples of digital transformation, we are considering the largest information systems in the US. It seems to us that the lessons of their implementation are applicable to Russian problems. We discuss the movement from circuit switching to packet switching and some challenges of transformation. We consider information network interfaces (control points) and discuss Federal Enterprise Architecture, namely, e-Government. Particular attention is paid to issues of cybersecurity and to the issues of system modeling, which, in our opinion, are greatly underestimated.**

## I. Introduction

The national program "Digital Economy of the Russian Federation" [1] defines the goals and objectives within the eight directions of the development of the digital economy for the period up to 2025:

1) State regulation;

2) Information infrastructure;

3) Research and development;

4) Personnel and education;

5) Information security;

6) Public administration;

7) Smart city;

8) Digital health care.

We will focus on issues that fall within the competence of the Faculty of Computational Mathematics and Cybernetics of Lomonosov Moscow State University, more precisely, on software development and mathematical modeling discussed earlier in [20, 21]. To some extent, these issues relate to the all mentioned above eight areas of the digital economy, but especially the "Information Infrastructure" section.

It is obvious, that the connectivity (and telecom, in general) is a critical issue for the infrastructure [2]. The use and analysis of infrastructure at the national level have, of course, its own characteristics. These are the requirements for security, the possibility of dual use of infrastructure, the absence of a critical dependence on imports.

For such an infrastructure, the choice of architecture is very important. The issue of migration and backwards compatibility is important (support for existing services) too. Of course, very important issues of proper design, as well as descriptions (specification) of systems. The latter becomes critical one because of the complexity and size of the projects.

As examples of digital transformation, we are considering the largest information systems in the US [3-8]. We consider information network interfaces (control points) and discuss Federal Enterprise Architecture, namely, e-Government [9], consider the Lifecycle Modeling Language (LML) [12] as an open-standard modeling language designed for systems engineering.

Therefore, we will focus on the move from circuit switching to packet switching (Section 2) and on some challenges relating to the transition to packet switching (Section 3). Section 4 considers information network interfaces (control points) in the US Army Common Operating Environment. In Section 5, we discuss Federal Enterprise Architecture, namely, e-Government. In Section 6, we discuss Lifecycle Modeling Language and its possible deployment for digital economy projects, in Section 7.

## II. The Move from Circuit Switching to Packet Switching

American communications technology for the needs of the military passed three generations of transformation: from signaling SS7 and intelligent networks (Joint Vision 2010) to IP protocol (Joint Vision 2020) and, finally, to the extremely ambitious plans of cybersecurity of networks (GIG-3) [2]. In 2007, Pentagon published a fundamental program [3], in which we find three main points:

1) to build a single Global Information Grid (GIG),

2) focused on network-centric war concept,

3) and the most important, to use IP protocol as the only means of communication between the transport layer and applications.

In our opinion, this is one of the world's largest examples of digital transformation. In addition, this transition obviously affected the critical elements of state systems. Accordingly, the

lessons of this transition can serve as a learning material in the discussion (planning) of the transformation of systems.

Another reason for considering this particular problem is that here and in related systems, perhaps for the first time outside the academic environment, software tools for system modeling and system description have become widely used. These tools, in our opinion, should be an obligatory element of the program of transition to the digital economy.

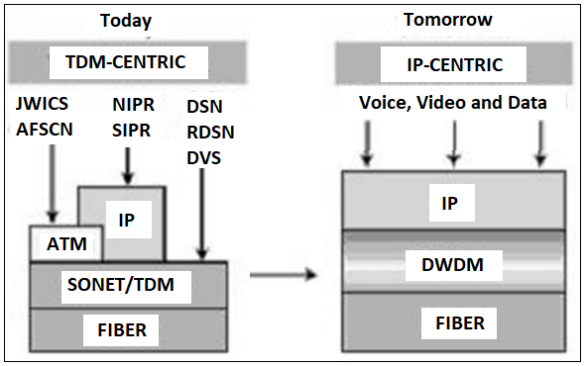


Fig. 1. The current DISN challenge: how to move from a Time Division Multiplexing (TDM) network to an IP network

Up to now, the main military communications networks of the Pentagon (Defense Information System Network, DISN) are circuit-switched networks:

1) DSN - Defense Switched Network,

2) governmental DRSN (Defense Red Switched Network),

3) DVS - video conferencing network (DISN VIDEO).

In addition, Fig. 1 shows three classified networks:

4) JWICS (Joint Worldwide Intelligence Communications System),

5) AFSCN (Air Force Satellite Control Network),

6) SIPRNet (Secret Internet Protocol Router Network) - to transmit sensitive information over TCP/IP protocols.

7) NIPRNet (Non-classified Internet Protocol Router Network) is a network used to exchange unclassified but important service information between "internal" users, and

The target architecture of the DISN network contains two levels: Tier 0 and Tier 1 (Fig. 2).

The Tier 0 cluster is responsible for the invulnerability of the entire DISN network. It contains three Tier 0 soft-switches connected by the ICCS (Intra-Cluster Communication Signaling) protocol, which automatically updates their databases. A cluster is essentially one distributed softswitch. It is required that the delay in the exchange of database contents does not exceed 40 ms. Since the signal transmission takes 6 microseconds per 1 km, the distance between cluster softswiches can not exceed 6,600 km.

At the lower, second level of the DISN network, Tier 1, there are two types of local networks: a secure ASLAN using the AS-SIP (Assured Security SIP) protocol and a traditional LAN using the H.323 protocol. Thus, the secure hybrid network DISN provides voice and video over IP.

## III. THE CHALLENGES OF TRANSITION TO PACKET SWITCHING

**Governmental DRSN as a "birthmark" in the AS-SIP environment**. The DRSN (Defense Red Switch Network) network is a dedicated telephone network that provides control of the US Armed Forces (Fig. 3). "Red Phone" (Secure Terminal Equipment, STE) connects to the network via ISDN line and operates at a speed of 128 kbps (Fig. 4). For data transfer and facsimile, an RS-232 port is built-in. All cryptographic information is stored on the crypto card. "Red phones" communicate via the SCIP (Secure Communications Interoperability Protocol) protocol. Note the slot at the bottom right - for a crypto card and four buttons at the top - to select the priority of the conversation.

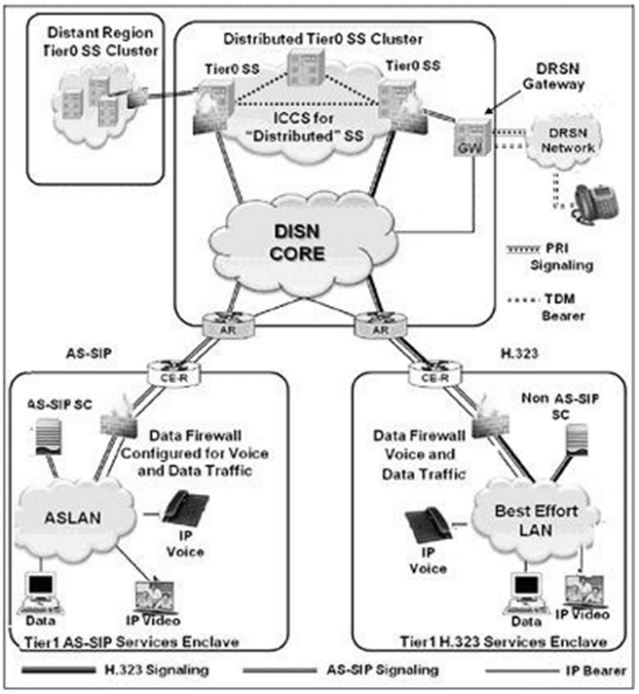


Fig. 2. The target architecture of DISN [4]

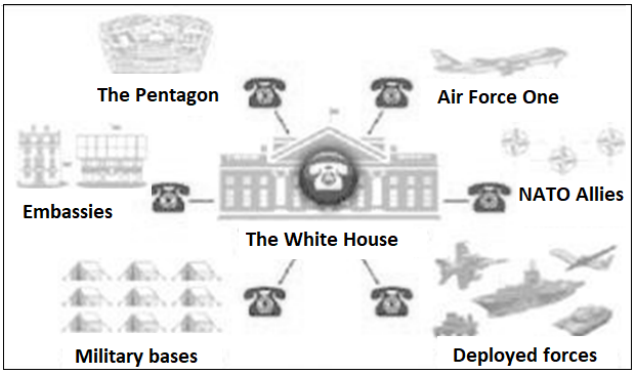


Fig. 3. Scheme of the government network DRSN

**The Pentagon has no power to support the global AIN.** In accordance with the program "Joint Vision 2010", the global defense network DISN is built on the basis of Advanced Intelligent Network (AIN), developed by Bell Labs in the early 1980s. Now after 30+ years, there were extraordinary

difficulties with maintaining the network AIN, which is the core of the global DISN network.

Fig. 4. "Red" phone

From the very beginning of the Joint Vision 2010 program, Lockheed Martin is responsible for the AIN. The emergence of new military equipment and new services requires the continuous improvement of AIN. This is evidenced by the invitation to work in Lockheed Martin. In the long list of vacancies, the first place takes the search for analysts of multifunctional information systems for DISA. From the applicants are required skills to develop new services for AIN and docking the AIN network with equipment from CISCO, Juniper, etc. Veterans with 28 years experience are invited also. Young professionals who grew up in a web programming environment seem unable to support and develop existing AIN networks built on circuit switching technology. It is a typical example of the difficulties in the way of digital transformation.

**The failure of the Lockheed Martin cybersecurity project**. In June 2012, Lockheed Martin won the largest tender for managing the GIG network (Global Services Management-Operations, GSM-O). The essence of the GSM-O contract is the modernization of the GIG network management system for cybersecurity requirements. The cost of work is a huge amount - 4.6 billion dollars for 7 years.

In 2013, the GSM-O team began to study the status of the four GIG network management centers that are responsible for the maintenance and uninterrupted operation of all Pentagon computer networks: 8,100 computer systems in more than 460 locations in the world, which, in turn, connected by 46,000 cables. The first deal was to consolidate the operating centers - from four to two. The GIG network management centers are expanding at the air bases Scott (Illinois) and Hickam in Hawaii, but the centers in Bahrain and Germany closed.

Cybersecurity targets are the Pentagon's top priority, but the lack of necessary standards hampers the implementation of the entire GSM-O program. In 2015, the world of telecommunications was shocked by the news: Lockheed Martin is not coping with the upgrade of the DISN network management, that is, with the implementation of a multi-billion dollar GSM-O contract, and sells its division “LM Information and Global Solutions” to the competing firm Leidos. The failure of the work was most likely due to the inability to recruit developers capable of combining the "old" circuit switching equipment with the latest packet switching systems as well as taking into account the new requirements of cybersecurity.

Therefore, the crucial question arises about the ubiquitous DISN transition to IP technology. It is the question about the very transition to IP technology in the world at all.

## IV. On Information Network Interfaces

**Interfaces of the information network in a combat situation**. In 2010, the US Department of Defense published the important document on the interfaces of the GIG 2.0 network [5]. Seamless integration must exist across the Army Enterprise Network and between computing environments. Control points facilitate the integration of mission environments (Fig. 5), and serve as intermediaries between mission environments and the corresponding computing environments.

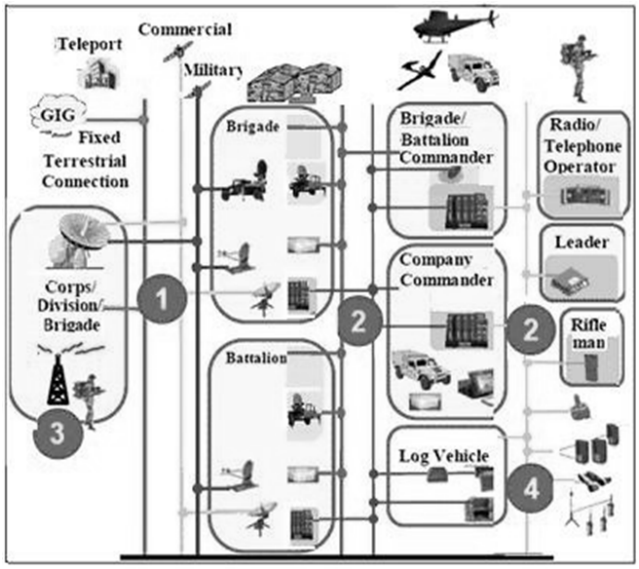


Fig. 5. Tactical Network and Control Points

Control points are placed to enforce the following requirements:

1) Interoperability of Structured Data (e.g., databases, geospatial data, spreadsheets) and Unstructured Data – Data that have no defined data model (e.g., documents, presentations, pictures, audio, video);
2) Security;
3) Gateways - to evaluate the request according to its filtering rules (e.g., by IP address or protocol).

Let us consider in more detail Control Point 2 (Enterprise/Command Post to Platform/Soldier/Sensor). This provides the interface to/from the enterprise standard/protocol by the following means.

- Interoperability: authentication via PKI, LDAP or Active Directory; the messaging – VMF; geospatial data standard is VMF/MIL-STD 2525C.
- Security: encryption – NSA/NIST-certified solutions; key management – EKMS/KMI-compliant solutions; end-point protection – Host-Based Security System (HBSS); enterprise service management –

Remedy/ITSM, IP Management/SPECTRUM (configured to roll up data at control points); and patch management – manual.

- Gateways: the enterprise/command post server is responsible for the translation of XML/SOAP to/from VMF.

The work of control points is regulated by a long list of open and closed standards – the full size is 20 pages in [5]. Note, that end-to-end service may span multiple service portfolios and/or organizational boundaries and therefore may require both internal and external Service Level Agreements (SLAs) to clarify the cooperation and governance for configuration control.

The model architecture organizes components based on overall system functional partitioning. It delegates lower level responsibilities to subordinate components. These models can be simulated to understand the system's behavior and performance. Also, they can generate an executable code. The generated code can be used as a base implementation for services and applications. It is important to note also, that the models include test cases, which will be used to validate that requirements are being satisfied. It is a mandatory part of the model.

This architecture is considered here because it is, in fact, the world's largest project to create information systems. Therefore, his lessons are extremely important for large infrastructure projects.

**On Common Operating Environment**. We look through the description of the architecture of the army information network of the US Army [6]. Its main goal is to ensure the army's combat capability in the current conditions of the network-centric war. In the previous section, we considered the architecture of a Common Operating Environment (COE) in one particular case. Document [5] also contains instructions to developers regarding the selection and use of approved computing technologies and standards to ensure the integration and compatibility of a common army information network.

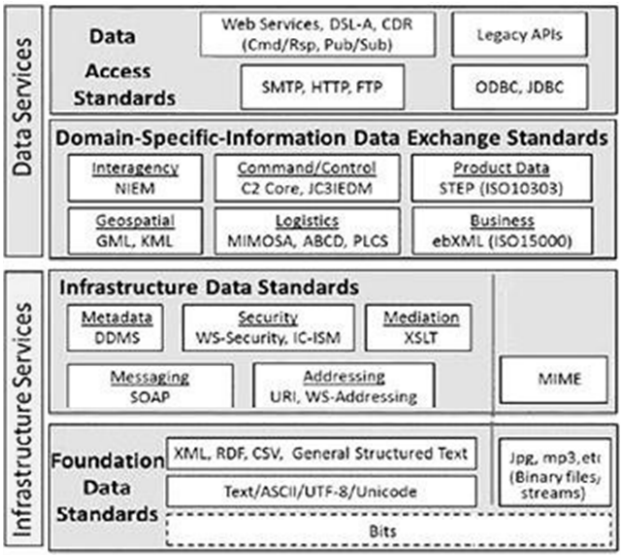


Fig. 6. Data Standard Classifications

Data standards fall into four broad categories (Fig. 6):

1) the most fundamental standards dealing with bit/byte level patterns for representing primitive information and structure;

2) infrastructure data standards that pertain to technology and its use, applicable throughout the COE and are not tied to or unique to any specific domain;

3) the exchange format for usage-domain-specific information;

4) the widely known data access standards.

**Control points in COE**. The document [7] describes the organization of the development of an army information environment between six divisions of the ministry (Fig. 7), corresponding respectively to the development of six types of Computing Environments (CE):

1) Data Center, DC: consists of 65 primary systems

2) Command Post, CP: consists of 26 primary systems

3) Mounted, MC: Operating and run-time systems, native and common applications and services. Consists of 6 primary systems

4) Mobile/Hand Held, MHH: consists of 10 primary systems

5) Sensors: for specialized, human-controlled or unattended sensors. Consists of 38 primary systems

6) RT/Safety Critical/Embedded, RTSCE: Consists of 44 primary systems. Coordination of work between these six units requires a very rigid, clearly standardized discipline. It is required to meet the requirements for fifteen control points (!) of the common operating environment for 189 (!) interfaces totally.

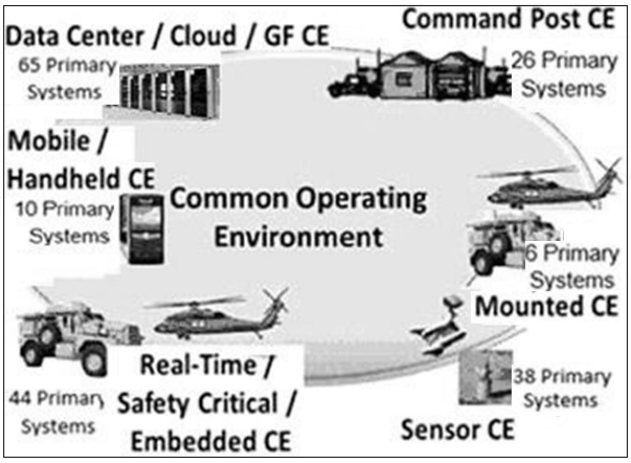


Fig. 7. COE focuses on 6 Computing Environments (CE) having 15 control points and 189 primary systems

In general, it is a typical System of Systems (SoS). As per Systems Engineering Guide for Systems of Systems, it is defined as a "set or arrangement of systems that results when independent and useful systems are integrated into a larger system that delivers unique capabilities"

SoS's can be

- Virtual (that is no central management and purpose)

- Collaborative (there is a voluntary interaction)
- Acknowledged (systems are independent, with higher level coordination)
- Directed (integrated)

There is a common denominator in all types of SoS: systems are dependent on other systems.

The report [8] of 2015 discusses the difficulties of developing and implementing COE. It is noted that: (1) until 2010, the development of COE was gone in two parallel but different directions of investment, which was unacceptable; (2) the proper coordination of work between the six COE development units was not ensured.

## V. Federal Enterprise Architecture

In 1987 there was an article by J.A. Zachman "Structure of the architecture of information systems" and for the first time the concept "enterprise architecture" was introduced [8]. John Zachman proposed an idea that is comparable to the periodic table of the Mendeleev for the IT industry.

Based on the Zachman model, the NIST has developed an e-government model for the US federal government - FEAF (Federal Enterprise Architecture Framework) [9]. The architecture of the federal organization is an attempt to bring countless agencies (ministries) of the US federal government to a single and universally used architecture. Fig. 8 shows a diagram of segments of the federal government: many segments (vertical columns) are used in many agencies and all or almost all of these segments can be reused.


Fig. 8. Segment map of the federal government

Unfortunately, the results of the FEA program development were not encouraging. In the official report of the Government Accountability Office for the US Congress on the status of the FEA program in 2002, it was concluded that "in general, the FEA system is not sufficiently developed to make informed investment decisions in the IT field" [10]. In addition, the FEA program was extremely expensive. For example, "by the end of 2010, the federal government spent more than a billion dollars on corporate architecture, and much, if not most, of it was wasted" [11].

The main source of problems is the complexity. So, one of the following projects is the language LML, which directly targets the complexity of the development process. We discuss it in Section 6 below.

## VI. Lifecycle Modeling Language

The Lifecycle Modeling Language (LML), according to its specification [12], is an open-standard modeling language designed for systems engineering. LML targets the complexity of formal languages as the first goal. The main idea was to create a language that can be understood by most system stakeholders, not just Systems Engineers. It supports the full lifecycle: requirements, design, acquisition, verification, operation, support, and disposal.

Right from the beginning, it was declared that the predecessor languages are UML and SysML. And the goal of LML is to replace them, because they are overcomplicating the systems engineering process. A complexity has been identified by many as a critical problem facing system engineers. Nowadays, larger and more complex systems (including systems of systems) development creates a demand for a clear and logically consistent semantics, for a clear and concise way to express the system design. Modern development is performed in larger distributed teams, which need new tools to enable collaboration across the entire lifecycle.

Originally, the system modeling tools have been created with the perception that the main problem is the software, an object-oriented approach for software development is the main goal, etc. It is what SysML and UML are for. E.g., the main declared goal for SysML is how to improve communications between systems engineers and software developers. As per modern vision, software is not really the main problem. Right now the problems are relating to bad requirements analysis, to verification and validation (to the process of checking that a software system meets specifications and that it fulfills its intended purpose) planning in the design phase, and to the monitoring and estimation throughout the lifecycle.

LML uses common language to define its modeling elements such as entity, attribute, schedule, cost, and relationship. Taxonomy has 12 primary element classes. The simplified model includes such elements as Action, Artifact, Asset, Resource, Characteristic, Connection, Cost, Decision, Input/Output, Location, Risk, Statement, and Time, shown in Fig. 9.

LML is simpler than other System Engineering languages like SysML in terms of ontology and visual expressions. For every class, LML defines a set of very explainable diagrams (only three of them are mandatory): Action, Asset, Spider, Interface Diagrams, Hierarchy Diagrams, Time Diagrams, etc.

At this moment, declared program for Digital Economy in Russian Federation [1] does not provide (does not recommend and does not require) system modeling or system description tools. In the digital economy with the digital twins [14], the main program for that economy does not provide digital model themselves.

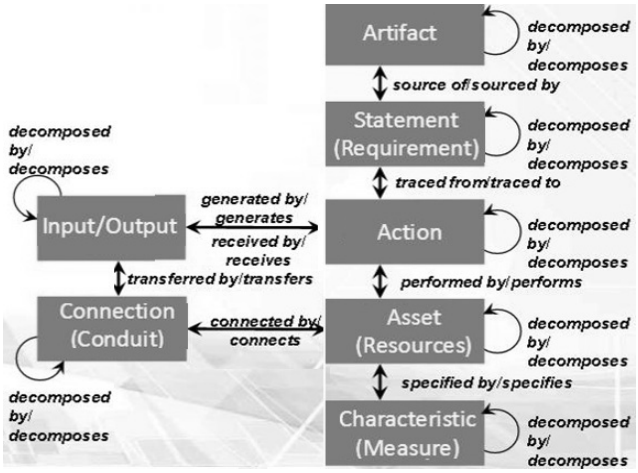


Fig. 9. LML [13]

The current LML standard 1.1 includes all the key features of the Systems Modeling Language, and thus can be used by system engineers to generate the complete SysML diagram set. LML supports both the functional and object-oriented approaches within the same design. One of the major problems with SysML is the lack of an ontology. And LML, potentially, could be used as an ontology for SysML [15].

In general, the LML language is the response of Systems Engineering Community to need to evolve the traditional document-based approach to a model-based approach. So, the models can be easily tailored to changing conditions and needs, re-used, and could be executable to test the static architecture in a dynamic environment. This turned into a result in so-called Model-Based Systems Engineering, defined as a set of the formalized application of modeling to support systems design and analysis, throughout all phases of the system lifecycle, through the collection of related processes, tools, methods, and languages used to support the systems engineering in a model-based (model-driven) context.

## VII. Discussion

Concluding the review of the state of information models of the digital economy, we note the following.

There are many serious organizations behind this huge work. Firstly, we point to the Open Group association, whose members published the first version of the TOGAF (The Open Group Architecture Framework) in 1995, which formed the basis for DODAF, MODAF, NAF, and DNDAF. In 2013, a modernized version of the TOGAF 9.1 language appeared.

Based on the OMG UML core, the Lifecycle Modeling Language (LML) was developed, which, according to its creators, is understandable not only to system engineers, but even ordinary shareholders who do not have a mathematical education. Thus, a lot has been done, however, the problems still remain. One of the biggest problems for LML is a very low level of adoption. Simply, it remains unknown for the majority of engineers. In the current Russian practice, due to the efforts of software vendors (for example, SAP, IBM), the modeling tools such as BMPM and UML have become known and are used to varying degrees. For system modeling, LML has got great advantages over them. In our opinion, LML should be adopted for digital economy projects in Russia.

The most important direction in the development of language tools is the development of ODM (Ontology Definition MetaModel). All terms in it are given through a description of a particular application to avoid synonymous mixing of concepts. On the basis of ODM, the UPDM methodology was developed (as well as the newest versions of DoDAF, MODAF, and NAF).

For the success of the national program "Digital Economy of the Russian Federation" [1], a lot of work should be done. For a survey, see our paper [16]. Some early ideas were presented in our work [17]. We have focused on software development and mathematical modeling. To some extent, these issues relate to all areas of the digital economy, but especially to the "Information Infrastructure" section.

At the moment, for the program of transition to the Digital Economy in Russia, there is no such element as architecture. We cannot name anything that would, for example, be an analogue of the above-mentioned federal architecture. At the same time, the published documents for unknown reasons do not pay any attention even to what is available for European countries (such as Latvia, for example) within the Single Digital Market approach. This seems to us the most serious shortcoming of the published program.

In our opinion, not only the architecture but the formal methods of specifications too should be standardized within the project office of the digital economy. We also see here great intersections with information modeling systems (BIM), which, in fact, are systems of formal specifications. One of the main tasks of information modeling is precisely the management of an object throughout the life cycle, when the operational phase is much more expensive and more complex than a one-time design. As a supporting argument we can cite the fact, for example, that one of the well-known LML implementations - 3DExperience platforms [18] is provided by Dassault Systems, which also produces tools for BIM. Also, LML could be translated to UML and enables translation to SysML, DoDAF 2.0, and other languages [19]. In general, this direction – Model-based System Engineering, in our opinion, should be a mandatory part of digital transformation programs. LML here is a first element in the chain of industry standards (together with SysML, and IDEF0), providing the modern end-to-end design, modeling, and traceability capabilities for systems engineers.

## VIII. Summary

This paper is devoted to the digital transformations and digital economy programs. The paper discusses the software development and mathematical modeling issues relating to the program of Russian Digital Economy, especially an information infrastructure. As examples of digital transformation, we are considering the largest information systems in the US. It seems to us that the lessons of their implementation are applicable to Russian problems. We discuss the movement from circuit switching to packet switching and some challenges of transformation. We consider information network interfaces (control points) and discuss

Federal Enterprise Architecture, namely, e-Government. We consider the Lifecycle Modeling Language (LML) as an open-standard modeling language designed for systems engineering. Particular attention is paid to issues of cybersecurity and to the issues of system modeling, which, in our opinion, are greatly underestimated. We focus on issues that fall within the competence of the Faculty of Computational Mathematics and Cybernetics of Lomonosov Moscow State University, more precisely, on software development and mathematical modeling. We guess the same type problems are typical for universities.

## ACKNOWLEDGMENT

We would like to thank the people from Open Information Technology Lab in Lomonosov Moscow State University and professor V. Sukhomlin for the valuable discussions. Some ideas outlined here were tested at the annual conference at Lomonosov Moscow State University. We are also grateful to the reviewers for their critical remarks. We have tried to make the asked corrections.

## REFERENCES

[1] The program "Digital Economy of the Russian Federation". *The Government of the Russian Federation*. Order of July 28, 2017 No. 1632-r

[2] M. Sneps-Sneppe. “On telecommunications evolution: Pentagon case and some challenges”. In *9th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, IEEE, 2017.

[3] Global Information Grid. Architectural Vision for a Net-Centric, Service-Oriented DoD Enterprise. Department of Defense. June 2007.

[4] U.S. Army Unified Capabilities Reference Architecture Version 1.0. October 2013.

[5] U.S. Army CIO/G-6. Common Operating Environment Architecture, Appendix C to Guidance for ‘End State’ Army Enterprise Network Architecture, 1 Oct 2010. https://www.us.army.mil/suite/doc/38201362 Retrieved: Feb, 2018

[6] Army Information Architecture (AIA). Version 4.1, Office of the Army Chief Information Officer, 5 June 2013. https://www.us.army.mil/suite/doc/38201362 Retrieved: Feb, 2018

[7] Engineering a Common Operating Environment for the US Army, 29 February 2012 http://www.ieee-stc.org/proceedings/2012/pdfs/2921JoyceTokar.pdf Retrieved: Feb, 2018

[8] Army Common Operating Environment (COE), March 11, 2015 http://www.afcea-aberdeen.org/files/presentations/AFCEA_Aberdeen_COE%20Update_11March2015.pdf. Retrieved: Feb, 2018

[9] The Chief Information Officers Council (1999). Federal Enterprise Architecture Framework, Version 1.1. September 1999.

[10] GAO. 2002. "Information Technology: Enterprise Architecture Use Across the Federal Government Can Be Improved," GAO-02-6, Government Accountability Office, Washington, DC.

[11] S.B. Gaver (2010) "Why Doesn’t the Federal Enterprise Architecture Work?" *Technology Matters*, McLean, VA.

[12] Lifecycle Modeling Language (LML) SPECIFICATION http://www.lifecyclemodeling.org/spec/LML_Specification_1_0.pdf Retrieved: Feb, 2018

[13] Lifecycle Modeling Language (LML) and Systems of Systems (SoS) https://www.acq.osd.mil/se/webinars/2015_05_19-SoSECIE-Dam-brief.pdf Retrieved: Feb, 2018

[14] Kupriyanovsky, Vasily, et al. "The new paradigm of the digital railway–assets life cycle standardization." *International Journal of Open Information Technologies* 5.2 (2017): 64-84.

[15] Vaneman, Warren K. “Enhancing model-based system engineering with the LifeCycle Modeling Language.” *System Conference (SysCon)*, 2016 Annual, IEEE. 2016

[16] Kupriyanovsky, Vasily, et al. "A holistic model of transformation in the digital economy-how to become digital leaders." *International Journal of Open Information Technologies* 5.1 (2017): 26-33.

[17] Schneps-Schneppe, Manfred, et. al. “Wired Smart Home: energy metering, security, and emergency issues”. In *Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2012 4th International Congress on*. IEEE, 2012

[18] Dassault Systems http://www.3ds.com Retrieved: Feb, 2018

[19] Innoslate http://innoslate.com Retrieved: Feb, 2018

[20] D. Namiot, M. Sneps-Sneppe. "On IoT Programming". *International Journal of Open Information Technologies* 2.10 (2014): 25-28.

[21] D. Namiot, M.Sneps-Sneppe. "On Internet of Things and Big Data in university courses". *International Journal of Embedded and Real-Time Communication Systems* 8.1 (2017): 18-30.