Ontological modeling of micro and macro architectures Co-Design of cyber-physical systems

Grigory Soldatov¹, Vladimir Ruchkin¹,²,³,⁴, Boris Kostrov² and Ekaterina Ruchkina²

¹Ryazan State University SC and Computer Department, Ryazan, 390000, Russia
²Ryazan State Radio-Engineering University CAD of computing facilities Chair
Ryazan, 390005, Russia
³Academy of the Federal Penal Service of Russia, Ryazan, 390000, Russia
⁴E-mail: v.ruchkin@365.rsu.edu.ru

Abstract. For effective design and modeling of modern mobile cyber-physical systems, it is proposed to use the theory of ontology, which has a number of important advantages: conceptual modeling of the target area in order to find hidden relationships, conflicts and description of incomplete (indefinite) information through abstraction at the physical or digital levels of the system, support for different styles modeling; modern interface of a large number of applications with a hierarchy of menus or work windows based on its developed conceptual model; the ability to combine two database applications or knowledge bases with completely different schemes of incompatible information systems into a single information space. Shown the importance of modern very high and is rapidly increasing mobile cyber-physical systems is determined by their wide range of application: almost all socially significant sectors of industry, economy, social systems: smart healthcare, smart home, smart city, smart e- and m-government. It is noted that the requirement for effective collaborative design and modeling leads to unusual complexity and ultra-high demands, application parallelism and heterogeneity, distribution of intelligence, computing resources for services and workloads in the Internet of Thinks hierarchy too. Suggested for modern complex applications require ultra-high performance and/or ultra-low power consumption использовать the authors propose using heterogeneous computing platforms, designing processors with parallel management multiprocessor macroarchitecture and parallel microarchitecture processors using various types of parallelism and application-specific parallel equipment based on super large-scale integrated circuit 1879VM6Y with high-performance processor core NMC4 of new generation architecture VLIW / SIMD.

1. Introduction
The main advantage of the Ontology Web Language (OWL) (www.w3.org/TR/owl-guide) is that it supports conceptual modeling of a specific domain. At the same time, OWL reasoning blocks can find hidden relationships, conflicts and missing elements, which allows you to describe incomplete information and interact with it through abstraction at the physical or logical levels of the information system. Another
advantage of the language is the support of different modeling styles: “top-down” or “bottom-up”, iterative or straightforward, focused on refinement or, conversely, generalization.

In addition, the modern interface of most applications is based on a hierarchy of different menus or work windows. Conceptual models based on OWL allow to build this hierarchy correctly. At the same time, for each class of users, its own ontology can be developed, corresponding to information needs, and on its basis, and its own interface. An important benefit of OWL is the ability to combine two database or knowledge base applications with completely different schemas, but with similar conceptual models. There is also the unification of incompatible information systems into a single information space. Thus, the listed advantages allow to increase the efficiency of modern cyber-physical systems at the design and simulation stages.

The modern cyber-physical system (CPS)\[1-2\] is a complicated set developed and integrated into even more complicated cybernetic and physical complexes and systems based on the existing components of CPS. As a result, a new set or complicated system appears that works as an independent unit respected to the outside world or other sets or systems.

A complex set of cyber-physical systems developed and integrated on the basis of existing components is a single whole with respect to the outer world or other sets or systems. Such sets are considered to be modern CPS.

2. Structuring analysis of multi-core organization macro-architectures

2.1. Distributed structure of the computing system

As a result of partitioning into clusters, a set of all possible structures $S_w \in S$ is determined, which allow a certain $j$-th processing program $PR$ to match a number of $CL_1$ clusters that are connected in series or in parallel, depending on the input algorithm:

$$S_w \in S: PR \Rightarrow \{CL_i\}; \forall i = \overline{1, L}; \forall w = \overline{1, W}$$

Expression $S_w \in S: PR \Rightarrow \{CL_i\}; \forall i = \overline{1, L}; \forall w = \overline{1, W}$ defines the already distributed structure $S_w \in S$ of the computing system being design.

In earlier publications, it was shown that expression $S_w \in S: PR \Rightarrow \{CL_i\}; \forall i = \overline{1, L}; \forall w = \overline{1, W}$ meets the requirements of reflexivity, symmetry, and transitivity and allows the original program to be divided into classes of non-overlapping sets called clusters or equivalence classes of a certain order or power. Further, these non-overlapping clusters can be combined into some structures in accordance with the algorithm.

2.2. The problem of explicit parallelization of algorithms

Thus, the $S_w$ structure of the designed computing system of data processing correlates a certain $j$-th processing program $PR$ with a set of independent and unequal $CL_1$ clusters, their number equaling the $L$ number of equivalence classes, and the $(CL_1)$ multiplicity is determined by the $(a_j)$ order of the equivalence class, the type of information exchange between clusters inside the class, and the type of exchange of information between clusters of various classes:

$$\forall o = \overline{1, N}, PR^{(j)} \Rightarrow \{(CL_q)^o\}; \forall q = \overline{1, a_j}; \forall l = \overline{1, L}$$

2.3. Core Programming

As a result, the problem of explicit parallelization of algorithms to a set of clusters and organization of information processing in the form of a data transmission structure in series or in parallel between clusters is solved.
∀j = \overline{1}NA^{(j)} \Rightarrow \cdots \forall o = \overline{1}N, PR^{(j)} \Rightarrow \{(CL_i)^q\}, \forall q = \overline{1}, a_i; \forall l = \overline{1}, L \\

Each cluster is a \textit{ROi} subroutine of a \textit{PR} processing program in machine instructions \textit{MK}_i and can be implemented on one of the cores of a given cluster. Otherwise, they can be programmed on the free cores of another single-type cluster.

2.4. Performance Assessment and Co - Design
Solving problem ∀j = \overline{1}, NA^{(j)} \Rightarrow \cdots \forall o = \overline{1}, N, PR^{(j)} \Rightarrow \{(CL_i)^q\}, \forall q = \overline{1}, a_i; \forall l = \overline{1}, L, we compute the resources or technical characteristics in the form of the set: speed 1/\textit{T}_R, firmware memory \textit{MP} \forall, number \textit{N}_0 of microprocessor cores of super large-scale integrated circuit (SLSIC) 1879BM6Y with high-performance processor core \textit{NMC}4 of new generation architecture \textit{VLIW} / \textit{SIMD} [4-5] and additive function of structure utility \textit{U}_1. This is decided advantage of \textit{decision of problem Co – Design hard ware and soft ware} architecture of CPS [3].

2.5. Implicit (fuzzy) cluster parallelism
Expression ∀j = \overline{1}, NA^{(j)} \Rightarrow \cdots \forall o = \overline{1}, N, PR^{(j)} \Rightarrow \{(CL_i)^q\}, \forall q = \overline{1}, a_i; \forall l = \overline{1}, L defines an already distributed control structure \textit{S}_w \in \textit{S} for computing resources, is the ratio of the equivalence of partitioning into independent modules, and represents an explicit parallelism procedure. To preserve the semantics of the original program, explicit (clear) and implicit (fuzzy) parallelism of clusters is introduced. Next, we consider the following possible solutions to problem ∀j = \overline{1}, NA^{(j)} \Rightarrow \cdots \forall o = \overline{1}, N, PR^{(j)} \Rightarrow \{(CL_i)^q\}, \forall q = \overline{1}, a_i; \forall l = \overline{1}, L in the form of a frame model of an expert system.

2.6. Multycore distributed management structure of computing resources with for clusters
Expression \textit{S}_w \in \textit{S}: PR \Rightarrow \{(CL_i)e\}; \forall l = \overline{1}, l; \forall w = \overline{1}, W states the already distributed management structure of computing resources. It expresses the partitioning equivalence into independent modules and presents an explicit parallelism structure, for example structure \textit{SBIS} 1879VM8Y [4-5].

3. Ontological Modeling
Modeling of the proposed methodology is described on the basis of ontologies with the formalism of descriptive logic in RDFS language with the description of dictionaries and the possibility of \textit{rdfs:subClassOf}. This language is a standard metadata of arbitrary structure, meaning, and can fix statements about resources or tactical and technical parameters: Speed 1/\textit{T}_R, firmware memory \textit{MP}, number of microprocessor cores \textit{N}_0, exchange properties: \textit{sequential1} and \textit{parallel1} between the cores of one cluster or \textit{sequential2} and \textit{parallel2} different clusters and the values of these properties with Co – Design [6-8].

Using the ontology representation language Web Ontology Language (OWL) in the form of a dictionary of terms defined by RDFS expressive dialect Full allows you to analyze the relationship between properties \textit{owl:equivalentProperty} is an equivalent property, \textit{owl:inverseOf} is an inverse property, \textit{owl:FunctionalProperty} is a unique property, \textit{owl:Inverse-FunctionalProperty} for the inverse functional property with the corresponding logical characteristics: \textit{owl:SymmetricProperty} for symmetric, \textit{owl:TransitiveProperty} for transitive and \textit{owl:ReflexivityProperty} for reflexive.

3.1. Description of the procedure for analysis, clustering, and nuclear facilities of microstructure and many nuclear facilities of macrostructures
As a result of the ontology presentation, all the mechanisms of the procedure for analysis and clustering of software microstructure nuclear facilities and many macrostructure nuclear facilities are described by the following characteristics:
• owl:ClusterizationProperty;
• owl:CompilationProperty;
• owl:ExplicitParallelismProperty;
• owl:MicroArchitectureProperty;
• owl:MacroArchitectureProperty;
• owl:MicroArchitectureProperty;
• owl:StructureProperty;

3.2. Description of possible macrostructure options
As a result of the analysis, possible variants of macrostructures are determined by owl:MacroArchitectureProperty:

• PipelineStructura – owl:PipelineStructuraProperty,
• NearPipelineStructura – owl:NearPipelineStructuraProperty.
• PipelineVectorStructura – owl:PipelineVectorStructuraProperty.
• NearPipelineVectorStructura – owl:NearPipeline VectorStructuraProperty,
• MatrixStructura – owl:MatrixStructuraProperty.
• NearVectorPipelineStructura – owl:NearVectorPipelineStructuraProperty.
• VectorPipelineStructura – owl:VectorPipelineStructuraProperty.
• NearVectorStructura – owl:NearVectorStructuraProperty.
• VectorStructura – owl:VectorStructuraProperty.

3.3. Owl:disjointWith construct
The disjointness of a set of classes can be expressed using the owl:disjointWith construct. This ensures that an individual who is a member of one class cannot simultaneously be a representative of a designated other class (figure 1).

```xml
<owl:Class rdf:ID="MacroStructura">
    <rdfs:subClassOf rdf:resource="# Structura"/>
    <owl:disjointWith rdf:resource="# NearPipelineStructura"/>
    <owl:disjointWith rdf:resource="# PipelineVectorStructura"/>
    <owl:disjointWith rdf:resource="# NearPipelineVectorStructura"/>
    <owl:disjointWith rdf:resource="# MatrixStructura"/>
    <owl:disjointWith rdf:resource="# NearVectorPipelineStructura"/>
    <owl:disjointWith rdf:resource="# VectorPipelineStructura"/>
    <owl:disjointWith rdf:resource="# NearVectorStructura"/>
    <owl:disjointWith rdf:resource="# VectorStructura"/>
</owl:Class>
</owl:ObjectProperty>
```

3.4. Getting the main tactical and technical characteristics
The disjointness of a set of classes can be expressed using the owl:disjointWith construct. This ensures that an individual who is a member of one class cannot simultaneously be a representative of a designated other class (figure 1).

```xml
<owl:ObjectProperty rdf:ID="haveSpeed">
    <rdfs:subPropertyOf rdf:resource="# resources/technical characteristics"/>
    <rdfs:range rdf:resource="# Speed"/>
    <rdfs:range rdf:resource="# firmware memory"/>
    <rdfs:range rdf:resource="# number of microprocessor cores"/>
</owl:ObjectProperty>
```
3.5. Interface description
As a result of the analysis, in order to select the best for certain technical parameters, the main parameters of analysis, clustering and structuring, which are given by the program for comparison with the obtained results, as shown in figure 2.

4. Results
During the experiment using this technique, a user interface was developed for modeling an expert selection system on a specific platform of a heterogeneous multiprocessor system on an 1879VM6Y chip [9-11].

4.1. Description of the expert system user interface
The paper proposes a description of the user interface that allows you to control input data, obtain results with a graphical representation of the resulting micro- and macrostructures. The interface includes three windows: Data Entry, Method of on Indistinct Fuzzy Logical Output and Graphical Representation. First window points to Number of Equivalence Clusters, Order of Equivalence Clusters, Data Interchange between clusters. The second window provides information on Structure Realization, General Parameters:
Number of Neuroprocessor Cores, Memory Sizes, High-Speed Performance, Scaled Estimates, Scaled Estimates, Normalized Weights, Quality Standards of the Significance, Quality Standards of the Usefulness. Third window gives Graphical Representation of MicroArchitecture or MacroArchitecture.

To implement and encapsulate the logic of the application, the class ComputerSystemStructure was written, which can determine the type of neural computer network needed. The class itself consists of methods, each of the consisting of the above rules.

The user interface consists of three parts. In the first part of the analysis, the input data for the ComputerSystemStructure class is entered.

4.2. Platform Description

The module is a digital device with interface capabilities, designed to be used as a universal hardware and software platform for receiving, processing, storage and transmission of large data streams in real time as part of embedded computers and specialized high-performance systems, as well as for building a wide class digital signal processing systems, machine vision and deep neural networks (NeuroMatrix Deep Learning – nmDL) on MC121.01 и MC 127.05 modules (figure 3).

SLSIC 1879VM6Ya is a high-performance processor [3-4], the architecture of which is based on the use of a new generation VLIW / SIMD processor core NMC4 and consists of:

- Two processing cores NMPU0 and NMPU1.
- 32/64 - bit RISC processing core with 32/64 - bit co - processor with floating point NMPU0.
- 32/64 - bit RISC processing core with 64 - bit co - processor with vector - matrix operation over integer data of variable length from 1 to 64 bits NMPU1.
- 12 two ports of banks NMPU1.
- Internal SRAM memory (16Kx64 bits each).
- 4 Mbits distributed memory.
- 1Kx64 bits cashe command.
- 32 - bits interface with exit memory DDR2 @ 400 MGc (EMI).
- Fore batches communication ports with bandwidth up to 1Gbits / sec each (CP0… CP3).
- 16 ports GPIO (GPIO).
- JTAG controller and port for debugging and testing (JTAG).
- Two independence RAP controllers (XDMAC & BPOMC).
- Controller exit / inter interrupt (EXTITC).
- Bit depth of data 32 bits.
- The bit width of command is 32 and 64 bits; size of address space - 4Gx32 bits.
- 3 scalar instruction for cecle (ALU operation, operation of modification of address, input / output).
- Performance - 1000 MIPS (3000 MOPS).
- Programmable data length from 2 to 64 bits (64 bits packed word length).
- The basic operation is integer matrix multiplication.
5. Conclusion

Effective joint design and modeling of modern mobile cyber-physical systems is proposed to be carried out on the basis of ontology theory, which has a number of important advantages of conceptual modeling of a specific subject area of a given subject area in order to find hidden relationships, conflicts and description of undefined fuzzy logic information by abstraction with are classified on a set-theoretic basis at the physical or logical levels of the information system. In addition, support is provided for different modeling styles with a modern interface for a large number of applications with a hierarchy of different menus or work windows based on their developed conceptual model. Moreover, it is possible to combine two database applications or knowledge bases with completely different schemes of incompatible information systems into a single information space. The authors propose The importance of modern very high and is rapidly increasing mobile cyber-physical systems is determined by their wide range of application: almost all socially significant sectors of industry, economy, social systems: smart healthcare, smart home, smart city, smart e- and m-government. The authors propose It is noted that the requirement for effective collaborative design and modeling leads to unusual complexity and ultra-high demands, application parallelism and heterogeneity, distribution of intelligence, computing resources для services and workloads in the Internet of Things hierarchy also. It is proposed for modern complex applications that require ultra-high performance and/or ultra-low power consumption to use the authors propose using heterogeneous computing platforms, designing processors with parallel control multiprocessor macroarchitecture and parallel microarchitecture processors using various types of parallelism and application-specific parallel equipment on based on super large-scale integrated circuit 1879ВМ6Я with high-performance processor core NMC4 of new generation architecture VLIW / SIMD. The purpose of this classification is to ensure the versatility of the cyber-physical system. The authors propose a methodology for cluster analysis of a single-nuclear organization of the CPS micro architecture and, based on the cluster approach of explicit and implicit parallelism, a structural analysis of the multi-nuclear organization of the CPS macro architecture is carried out.

For the purpose of further research, an ontological description is made and modeling of CPS micro and macro architectures is carried out by means of the formalism of descriptive logic in the RDFS language and the OWL ontology presentation language. In the process of obtaining experimental data, simulation results are presented with a description of the user interface of the proposed expert system based on ontologies and the presentation of both tactical and technical characteristics.

The ontology for expert system has been developed with a interface for analyzing the resulting clear and fuzzy structures for various technical characteristics according to the selection strategy: speed 1/Tₘ, firmware memory (|MP|), number of microprocessor cores N₀ etc.

The article proposes set-theoretic clustering of explicit concurrency, on the basis of several frame models of knowledge in the form of decision-making rules as a machine of logical inference. To implement and encapsulate the application logic, a ComputerSystemStructure class was written, which can determine the type of neural network.

Development of new informational, operational and algorithmic support for multiprogrammable cyber-physical systems results in reduction of risks during decision-making in handling emergencies and using data from aerospace, ground and physical monitoring, analysis of problem dynamics and construction of
forecast contours. Environmental conditions at the time of emergencies (of natural or anthropogenic character) are usually unstable or uncertain; thus, using smart computing in processing of such data will increase efficiency of telecommunications in real time. Adaptation of the algorithms developed on the basis of NeuroKS software complex for the software used in commercially available MC 127.05 modules will enable improvement and automation in gathering and processing of primary monitoring data (LPM). SLSIC 1879VM6Y is a high-performance processor [4-5], the architecture of which is based on the use of a new generation VLIW / SIMD processor core NMC4.

Acknowledgements

The paper was prepared as a result of the research conducted under two grants Russian Foundation Basic Research (RFBR) № 14-07-00261, 14-07-00261 (a) “Clustering and organization of cloud and distributed computing systems on the multiprocessor base” and the scientific grant № 2.9519.2017/8.9.

References

[1] Jozwiak L 2019 Introduction to Modern Cyber-Physical Systems and their Quality-Driven Design (Budva, Montenegro: MECOnet)
[2] Jozwiak L 2017 Advanced mobile and wearable systems Microprocessors and Microsystems 50 202-21
[3] Bykovsky S V, Gorbachev Y G, Platunov A E, Kluchev A O and Penskoi A V 2016 Hardware/Software Co-Design part 1 (St. Petersburg, Russia: ITMO)
[4] Chernikov V, Vixne P and Shelukhin A 2015 High-performance nmc4 vector processor core for fixed and floating point calculations Proc. 6th Moscow Supercomputing Forum (Russia, Moscow: Technosphere) 13-4
[5] Chernikov V, Vixne P and Shelukhin A 2015 New core of signal processor core nmc4 of set neuro matrix Proc. 6th Moscow Supercomputing Forum (Russia, Moscow: Technosphere) pp 12-3
[6] Zlobin V K and Ruchkin V N 2011 Neural Networks and Neurocomputers (Russia, St. Petesburg: BHX Petesburg)
[7] Ruchkin V N, Kostrov B V, Kolesenkov A N and Ruchkina E V 2014 Arthropogenic situation express monitoring on the base of fuzzy neural networks Proc. 3nd Mediter. Conf. on Embedded Comp. MECO (Budva, Montenegro: MECOnet) pp 166-9
[8] Ruchkin V N, Kostrov B V, Ruchkina E V, Kolesenkov A N and Taganov A I 2017 Means of neural network for the anthropogenic environment analysis Proc. Int. Conf. on Mech., System And Control Engin. (Russia, St. Petesburg: IEEE) 352-6
[9] Ruchkin V N, Kostrov B V, Kolesenkov A N and Ruchkina E V 2015 Algorithms of fire seat detection, modeling their dynamics and observation of forest fires via communication technologies Proc. 4th Mediter. Conf. on Embedded Comp. MECO (Budva, Montenegro: MECOnet) pp 166-9
[10] Kostrov B V and Ruchkin V N 2015 Method of smart, detection, modeling and follow-up of fires TulGU News 5(2) 266-74
[11] Ruchkin V, Ruchkina E, Soldatov G, Koryachko A and Kostrov A 2020 Conceptual model of hardware & software co-design for multicore systems on chip Proc. Mediter. Conf. on Embedded Comp. MECO (Budva, Montenegro: MECOnet) pp 1-4