Methods and Means for Automated Information Systems Development based on Ontology «Software and Hardware Complexes Quality Management»

Samonov A.V. <a.samonov@mail.ru>
Introduction:

Relevance of the problem

The main part:

1. Conceptual bases and basic elements of implementation of software-controlled process of AIS development.
2. Models and methods of development and verification of complex requirements to AIS.
3. Methods and means of development and verification of architecture and design solutions to AIS

Conclusion:

Directions of development and implementation of the proposed solutions in practice
Relevance of the problem of quality assurance AIS CII

Relevance is conditioned by:

1) high requirements to functional and operational characteristics of AIS CII (correctness, reliability, performance and security);
2) the need to integrate newly developed systems with existing and legacy ones;
3) the need to operate in a heterogeneous environment on multiple hardware platforms;
4) high level of novelty, which limits the possibility of using standard algorithms of functioning and design solutions.

Chaos reports
* The causes of the problems and risks

Obstacles:
- the main reasons for this state of Affairs is the lack of participants in the process of creating AIS CII (user, designer and programmer-developer) a single terminological and conceptual framework, adequate logical and mathematical apparatus,
- there are no effective means of supporting the processes of formation and analysis of the two most important artifacts of the life cycle (LC) of AIS IIC: a set of requirements and design solutions.

There is a high level of risk to obtain AIS that does not meet the requirements of the Customer and Users, while there is a high risk of not meeting the limits of the allocated time and financial resources in connection with the detection of defects in system artifacts at the later stages of the Software LC.
Models, methods and means of industrial software and hardware development

Specifications OMG (Object Management Group)

FUML (Semantics of a Foundational Subset for Executable UML Models), ReqIF (Requirements Interchange Format), OCL (Object Constraint Language), UTP (UML Testing Profile), Action Languages (ALF, Scylla, ...), ...

Support tools:

Rational Rhapsody Developer, Sparx Enterprise Architect, MASIW (ISPRAS in collaboration with GosNIIAS (ГосНИИАС); Rodin, CPN Tools, SPIN; Eclipse Modeling Framework, Graphical Editing Framework, Modelio, Papyrus, GEMOC Studio, ...

Monographs, manuals and other publications:


5. Advanced and efficient execution trace management for executable domain-specific modeling languages Erwan Bousse, Tanja Mayerhofer, Benoit Combemale, Benoit Baudry.


Unresolved issues and constraints:

1) objective complexity of the task of constructing a formal representation of the requirements for AIS CII on the basis of their initial informal representation in the terms of Reference;

2) availability of a wide range of languages and tools offered for building AIS CII at various stages of creation (justification of requirements, architecture development, implementation of hardware and software complex), in the absence of clear and specific rules and recommendations for the use of these languages and tools.

Proposed Ways solve the problem in order to ensure the required quality of creation and maintenance of AIS CII at all Development LC stages are:

- building a unified model-language and software-information environment for development and verification of AIS CII, and
- development of algorithmic, instrumental and methodological support for the implementation of software-controlled process of development and verification for requirements, architecture and implementation formal models of AIS CII.
The following means to build a unified model-language and information-software environment for development and verification of AIS CII are offered:

1) FUML, SysML, OCL, ALF, Scram model language,

2) development methods and means of subject-oriented ontology of AIS CII,

3) libraries and software products that are implemented in the framework of the Eclipse project: Eclipse Modeling Framework, Graphical Editing Framework, Papyrus, Modelio.

The choice of these models, languages and tools is due to the fact that, firstly, their development is actively supported by leading enterprises developers and consumer organizations SPTS, and secondly, both the technologies themselves and the tools based on them are open and available for study, application and improvement.
Example of quality management ontology fragment
«Характеристики качества требований»

Классификация характеристик качества требований к СППС

- Адекватность
- Внутренние
  - Однозначность
  - Непротиворечивость
  - Систематичность
  - Неизбыточность
- Внешние
  - Проверяемость
  - Отслеживаемость
  - Модифицируемость
Fragments of ontology describing the composition and structure of the system, ways of interaction between objects and subjects, functional requirements for all its components:
The Ontology of "AIS quality Management" describes the quality requirements of both the system as a whole and each structural component.
Development of requirements formal model

The development of the preliminary RC model is realized by means of:

1. generation functional requirements for programs complex in the form of UCdiagrams;
2. description of the composition and structure of the system as a set of diagrams of blocks and classes (d_blocks, d_class);
Development of requirements formal model

The unified UseCase&Classes metamodel - it is a preliminary model of functional requirements.

Example of ontology fragment OverviewClass diagram "Основная концепция решения задачи БО"
Example of quality management ontology fragment «Требования к функциональным возможностям АИС КИИ»

Development of quality requirements for the AIS function implementation
Methods and tools for verification and validation formal models of requirements

Formal model of requirements

- bdd
  - attribute1 : Type
  + function1(Type) : void
  - function2() : Type

- reqs
  maxvalue
  reliability
  performance
  response
  Time

- UC
  \( e_1 \)

  UC (n,d,a,s,b,a)

  init

  result

- Active Class
  - attribute1 : Type
  + operation1(Type) : void
  - operation2() : Type

Translator from FUMIL&OCL to CPN ML

CPN Tools

Defects of the requirements complex

Critical Events Analysis

Event
- timestamp : long
+ getTimestamp() : long

TraceEvent
- activityExecutionID : int
+ getActivityExecutionID() : int

Next tasks for VM fUML
> execute
> next step
> resume
Methods and means of development architecture and design solutions

Example of a sequence diagram metamodel: Ввод суточного плана

Example of a State Machine diagram: PlanString
*Methods and means for verification of architecture and design solutions*

**Formal architecture model**

- **bdd**
  - Allocated from value

- **regs**
  - reliability
  - security

- **Active Class**
  - attribute1 : Type
  - operation1(Type) : void
  - operation2() : Type

**fUML, OCL, ALF**

**d_sm Analysis**
- Event-B translator
- Rodin

**d_seq Analysis**
- Promela translator
- SPIN

**d_act Analysis**
- CPN ML translator
- CPN Tools

**Virtual machine for execution, validation and verification fUML (ALF) models**

**Executor**

```java
public class Executor {
  public ParameterValueList execute (Behavior behavior) {
    Execution execution = this.locus.false
    ... execution execute ();
    ...
  }
}
```

**Locus**

**ExecutionFactory**

**Critical Events Analysis**

- Event
  - timestamp : long
  + getTimestamp() : long

- TraceEvent
  - activityExecutionID : int
  + getActivityExecutionID() : int

- Next tasks for VM fUML
  - execute
  - next step
  - resume
The Implementation and application of this Complex will allow:

1) timely detect and eliminate defects of complex requirements and design solutions through their validation and formal verification;
2) improve the quality and efficiency of AIS development and maintenance processes that have the necessary functional and operational characteristics, as well as meet the requirements of regulatory and technical documents and operating conditions;
3) improve economic performance in terms of reducing the financial and time costs associated with the implementation of additional work, both in case of detection of any defects, and when changing the requirements or operating conditions of AIS.
The presentation is end.

Thanks for your attention.