

# Developing Pattern-Based Industrial Information Systems: Models, Tools, and Results

Dr. Sergey V. Zykov, PhD  
Software engineering department  
State university – Higher school of economics  
Moscow, Russia  
e-mail: szykov@hse.ru

## Abstract<sup>1</sup>

Developing enterprise software system (ESS) is dramatically hampered by exponentially growing data burden, and by heterogeneous nature of the system components in terms of architecture and structure. An integral approach to the enterprise software system lifecycle, which combines rigorous formal models and the supporting CASE tools, helps to obtain design patterns for the component-level metadata. Such patterns provide for more efficient enterprise software system lifecycle management due to increased software artifacts reuse. Thus, the enterprise software system implementations become more adaptive to rapidly changing business requirements. The paper outlines the pattern development approach for enterprise software systems and discusses its prospective implementations.

## 1. Introduction

The objective of the paper is a systematic outline of the new technology for developing large-scale integrated heterogeneous data warehouses. Currently, the multinational enterprises possess large, geographically distributed infrastructures, aimed at the same business goals. Each of the enterprises has accumulated a tremendous and rapidly increasing data burden, comparable to an avalanche. In certain cases, the data bulk exceeds petabyte size, and it tends to double every five years. Undoubtedly, warehousing of such data is a serious challenge. The problem becomes even more complicated due to heterogeneous nature of the stored data, which varies from well-structured relational

databases to non-normalized trees and lists, and to weak-structured multimedia data (scanned documents, audio and video data etc.). The technology, outlined in the paper is focused at more efficient heterogeneous enterprise data warehousing due to unification of the management procedures. The technology involves a set of novel mathematical models, methods, and the supporting software engineering tools for object-based representation and manipulation of heterogeneous enterprise warehouses data. The architecture of the technology is based on internet/intranet portals.

## 2. Industrial information system development: challenges and features

Unfortunately, a brute force application of the so-called "industrial" enterprise software development methodologies (such as IBM RUP, Microsoft MSF, Oracle CDM etc.) to heterogeneous enterprise data warehousing, without an object-based model-level theoretical basis, results either in unreasonably narrow "mono-vendor" solutions, or in inadequate time-and-cost expenses. On the other hand, the existing generalized approaches to information systems modeling and integration (category and ontology-based, Cyc and SYNTHESIS projects [5-7] etc.) do not result in practically applicable (scalable, robust, ergonomic) implementations since they are separated from state-of-the-art industrial technologies (CASE, RAD etc.). A number of international and federal research programs in the field supported by UN, UNESCO, USA, EU, Japan, Russia, proves that the technological problems of heterogeneous enterprise warehousing are really essential [9].

Thus, the suggested technology of integrated development and warehousing of heterogeneous internet-based enterprise software systems has been developed. The approach is based on rigorous mathematical models and it is supported by software engineering tools, which

---

Proceedings of the 12<sup>th</sup> international workshop on  
computer science and information technologies  
CSIT'2010, Moscow – Saint-Petersburg, Russia, 2010



provide integration to standard CASE tools, commonly used with "industrial" software development methodologies. The approach provides data duplication and contradiction within the integrated warehouses, thus increasing the robustness of the enterprise software systems (ESS). The technology takes into consideration quite a number of interrelated ESS development levels, such as data models, software applications, "industrial" methodologies, CASE, architecture, and database management.

The novel elements of the technology include the following:

- Conceptual framework of ESS development;
- A set of object models for ESS information representation and manipulation (warehousing);
- Engineering tools, which support semantic-oriented ESS development, warehousing and intelligent content management (ConceptModeller, ICMS) [16–18];
- Portal architecture, ESS rapid prototypes and full-scale implementations in a number of enterprises [19, 20].

### 3. The integrated model framework for pattern-based industrial information systems

For adequate modeling of heterogeneous ESS, a systematic approach has been developed, which includes object models for both data representation and data warehousing [16-20].

The general technological framework of ESS development provides closed-loop, two-way construction with re-engineering. The latter feature is really critical for ESS verification, which essentially increases implementation robustness and reliability.

The general technological framework of ESS development contains stages, which correspond to data representation forms for heterogeneous software system components, communicating in the global environment. Such data representation forms include natural language, mathematical models, engineering tools integration, content management, data warehousing). The data representation forms are further detailed by the representation levels (such as objects, relationships, events, as well as instances of engineering tools and ESS components).

Content-oriented approach to ESS data warehousing allows data/metadata generalization on the common model basis, unified warehousing of heterogeneous objects, and adequate modeling of the global internet environment, which is critical for ESS robustness and reliability.

The object nature of the "class-object-value" model framework provides compatibility with traditional object-

oriented analysis and design approach (OOAD), as well as with other certain promising approaches (such as D. S. Scott's variable domains [12], V. E. Wolfengagen's conceptual method [14]) and helps to extend the mentioned approaches to model the ESS internet-based environments.

The following technological transformation sequence, according to the models developed, is suggested:

- A finite sequence object (e.g., a  $\lambda$ -calculus term) [1];
- A logical predicate (higher order logic is used);
- A frame (as a graphical representation) [11];
- An XML object (class definition generated by the ConceptModeller engineering tool [15]);
- A UML diagram (data scheme in terms of an industrial CASE tool) in the ESS (meta)data warehouse.

Therewith, the warehouse content representation is based on semantic network situation model, which provides intuitive transparency for problem domain analysts when they construct the problem domain description. The model can be ergonomically visualized through a frame-based notation. Warehouse content management is modeled as a state-based abstract machine and role assignments, which naturally generalizes the processes of similar engineering tools, such as (portal page template generation, portal page publication cycle, role/access management etc. Therewith, the major content management operations (declaration, evaluation, personalization etc.) are modeled by the abstract machine language. The language has got a formal syntax and denotational semantics in terms of variable domains (including content object generation, semantic functions and statements for these functions).

The technological transformation sequence of the model is as follows:

- A term of variable domain algebra (in terms of D. S. Scott's theory of computations) [12];
- A domain-based function (higher order logic is used) [12];
- A frame (a graphical notation) [11];
- An XML object (a template for a ICMS portal page);
- HTML code (ICMS portal page code) on the ESS warehousing portal.

The architecture of the integrated heterogeneous enterprise content warehouse provides unification due to generalized object association-based relationships at the data at metadata levels. On the other hand, the uniform heterogeneous ESS content management is based on a uniform portal foundation, which serves a meta-level enhancement over the enterprise data warehouse.



Therewith, assignments, implemented as software scripts, which change the states of IMCS virtual machine, provide dynamical, scenario-driven content warehousing within the portal architectural framework.

Another kinds of script implement personalized content warehousing, which is supported by a multi-parameter functional model and the IMCS virtual machine.

#### 4. Tools for pattern-based industrial information system development

The ConceptModeller engineering tool assists in semantically-oriented visualized development of heterogeneous ESS data warehouse scheme. Therewith, a semantic network-based model is suggested, which works in nearly natural-language terms, intuitively transparent to problem domain analysts. Model visualization is based on frame representation of the warehouse data scheme.

Thus, due to deep integration with mathematical models and state-of-the-art CASE toolkits, the ConceptModeller tool provides a closed-loop, continuous ESS development cycle (from formal model to data warehouse scheme) with a re-engineering feature. Therewith, frames are mapped into specific ordered lists.

The ICMS engineering tool is based on an abstract machine model, and it is used for problem-oriented visualized heterogeneous ESS content warehousing and portal publication cycle. The ICMS tool features a flexible content management cycle and role-based mechanisms, which allow personalized content warehousing based on dynamically adaptive access profiles and portal page templates. Due to scenario-oriented content warehousing, the ICMS provides a unified portal representation of heterogeneous data and metadata objects, flexible content processing by various user groups (ordinary and privileged, enterprise and external), high data security (based on scenarios and access profiles), a higher ergonomics level (based on personal preferences) and intuitively transparent complex data object warehousing (incl. multimedia data). Therewith, the data object classes of the ESS warehouse are represented by order lists of *<attribute, type>* format, and templates – by ordered lists of *<attribute, type, value>* format.

#### 5. Building patterns for industrial information systems

The general ESS development framework (which contains 6 stages and 6 levels) [18–20] potentially allows the following benefits:

- Applying a "spiral-like" lifecycle to the general ESS development framework (sequential elaboration of ESS warehouse scheme after each iteration of the development cycle);

- ESS "tuning" (ESS software and data warehouse component-wise improvement) by applying a "spiral-like" lifecycle and subsequent verification;
- Requirement "tracing" (through reverse engineering and/or verification, followed by correction and/or optimization if required);
- Building a repository of ESS "meta-snapshots", with which the system and/or warehouse could be "reincarnated" to virtually any previous state using component-wise strategy;
- Building a "pattern catalogue" [4] for heterogeneous ESS, based on the integrated repository of various ESS state "meta-snapshots";
- Developing a repository of "branches" for "cloning" slight ESS variations for the "basis" (e.g., a portal "tree" for each company of a corporation);
- Developing a formal language specification (e.g, a DSL technology-based one) [2] for ESS requirement specification (let us call it RSL meaning Requirement Specification Language);
- "Adjusting" the existing ESS "meta-snapshot" repository components (i.e. the "new" product specifications) to match the new requirements;
- Reuse of the desired components ("best-practice" solutions for developers, "ready-made" components for the client).

Thus, the general ESS development framework implies software lifecycle variations according to waterfall, spiral, evolution, and incremental approach. The general ESS development framework tends to be iterative. However, in certain cases (relatively small ESS, fixed requirements, minor upgrades, as, let us say in case of systems for DoD), the waterfall model is possible and reasonable.

An essential feature of the general ESS development framework is its two-way organization. The approach provides reverse engineering possibility both for ESS in general, and their components in particular. The practical value of the approach is provided by the verifiability of heterogeneous ESS components (information systems, databases etc.) at the uniform level of the (formal) problem domain model, which is practically independent upon the hardware and software environment of the particular component (or ESS in general). Therewith, a major theoretical generalization is a possibility of mathematically rigorous verification of the heterogeneous ESS components by a function-based model (such as typed lambda expressions [1], categorical combinatory logic objects [2], polymorphic types in terms of variable domains with event-based models [12] etc.). A critical issue for engineering practice of huge and complex ESS, is that the models suggested are oriented at a very promising "pure" objects approach, which is a strategy of the state-of-the-art enterprise-level component



technologies of Microsoft .NET and Sun Java, where any program entity is an object.

An essential benefit of the approach suggested is a possibility of adaptive, sequential "fine tuning" of ESS heterogeneous component warehousing schemes in order to match the rapidly changing business requirements. Such benefit is possible due to the reverse engineering feature of the integrated general iterative framework of ESS development. The reverse engineering is possible down to model level, which allows rigorous component-wise (and even object-wise) ESS verification. Thus, conventional reengineering and verification can be enhanced by flexible correction and "optimization" of the target ESS in strict accordance with the specified business requirements. This is possible due to the suggested model-level generalization of the iterative, evolutionary ESS development framework.

Another benefit of the suggested ESS development framework is a possibility of building a "catalogue of templates for heterogeneous ESS", which is based on an integrated metadata warehouse, i.e., a "meta-snapshot" repository. Thus, the software development companies get a solution for storing relatively stable or frequently used configurations of heterogeneous enterprise software systems (or such configuration fragments). The solution potentially allows avoiding the integration problems of "standard" ESS components and/or combinations, which have been obtained previously, (e.g., integrating certain modules and/or versions of SAP R/3 and Oracle e-Business Suite ERP systems). The approach allows serious software engineering project savings (in terms of budget, terms, and labor) for clients, provided the ESS developer's "meta-snapshot" repository already stores a similar or an analogous integrated solution to the system required.

The above consideration clears the way for "meta-snapshot" repository development, which stores the chronological sequence of ESS solutions as a tree with the "baseline" version and slightly different "branches" for ESS variations. This is analogous to software engineering tools for version control. The approach allows a reasonable selection of most valuable deliverables of the ESS lifecycle phases (e.g., design, implementation, maintenance), and organization of similar solution "cloning". Therewith, the "clones" may be created both for different client enterprises, and for different companies of a single enterprise (a visual illustration of the latter solution is a "tree" of company portals, which is based on a shared ESS data warehouse embracing heterogeneous components of ERP and CRM modules).

Further discussion could cover the prospective areas of "meta-snapshot" repository development. First of all, to describe the metadata warehouses and the related enterprise-level business requirements it seems reasonable to develop new DSL-type [2] problem-

oriented meta-languages. Let us call them the MetaWarehouse Description Language (or MWDL) and the Requirement Specification Language (or RSL) respectively. Further, the formal models, outlined in the paper and given a more detailed coverage in [17–20], allow interrelating the RSL and MWDL entities (and, in future, potentially, to arrive to a more rigorous mapping).

Semantic-oriented search mechanisms based on semantic networks with frame visualization will assist in revealing the components of ESS "meta-snapshot" repository, which provide the closest matching to the new requirements. The approach potentially allows terms-and-cost-effective and adequate transforming of the existing ESS components in order to match the new requirements with minimum corrections effort and, consequently, with minimum labor expenses. Therewith, the global perspective (in case of a rich "meta-snapshot" repository) it becomes possible to reuse certain ESS components (or their certain combinations) for current or new clients. Selection criteria for such "basic" components may be percentage of reuse, ease of maintenance, client satisfaction, degree of matching business requirements etc.

## 6. Results, conclusion, and prospects

Implementation of the suggested approach allowed to developing a unified ESS, which integrates a number of heterogeneous components: state-of-the-art Oracle-based ERP modules for financial planning and management, a legacy HR management system and a weak-structured multimedia archive. The implementation of internet and intranet portals, which manage the heterogeneous ESS warehouse content, provided a number of successful implementations in diversified ITERA International Group of companies, which has around 10,000 employees in nearly 150 companies of over 20 countries.

The systematic approach to ESS framework development (including formal models, software engineering tools, portal architecture) provides integration with a wide range of state-of-the-art CASE tools (such as IBM Rational, Microsoft Visual Studio .NET, Oracle Developer) and standards (UML, XML) of ESS development.

Strategic benefits of the approach as compared to methodologies of the above mentioned vendors are achieved due to model and software engineering tools orientation on integrated heterogeneous portal-based ESS warehousing. The challenging nature of the heterogeneous ESS results from their high complexity, versatile architectures and data object structures. Qualitative assessments of the approach implemented functional features are approved by comparison of major macro level indices – total cost of ownership, return on investment and implementation costs. Implementation results in ITERA Group exceed the leading commercially available solutions by the average of 30-40%.



ESS and curricula based on the research results have been successfully implemented in a number of governmental and commercial enterprises, such as ITERA, Russian Ministry for Industry and Energy, Institute of Control Systems of Russian Academy of Science et al. [19,20].

## References

1. Barendregt H.P. *The lambda calculus* (rev. ed.), Studies in Logic, 103, North Holland, Amsterdam, 1984.
2. Cook S., Jones G., Kent S., Wills A.C. *Domain-Specific Development with Visual Studio DSL Tools*, Pearson Education, Inc, 2008, 524 pp.
3. Curry H.B., Feys R. *Combinatory logic*, Vol.1, North Holland, Amsterdam, 1958.
4. Fowler M. *Analysis Patterns: Reusable Object Models*, Addison Wesley, 1997, 223 pp.
5. Forbus K., Birnbaum L., Wagner E. et al. Combining analogy, intelligent information retrieval, and knowledge integration for analysis: A preliminary report. In: *2005 International Conference on Intelligence Analysis*, McLean, Virginia, USA, 2005.
6. Kalinichenko L.A., Stupnikov S.A. Heterogeneous information model unification as a pre-requisite to resource schema mapping. In: A. D'Atri and D. Sacca (eds.), *Information Systems: People, Organizations, Institutions, and Technologies. Proc. of the V Conference of the Italian Chapter of Association for Information Systems itAIS*, Berlin-Heidelberg: Springer Physica Verlag, 2009, pp. 373-380.
7. Lenat D., Guha R.V. *Building Large Knowledge-Based Systems: Representation and Inference in the Cyc Project*. Addison-Wesley, 1990.
8. Masters J., Güngördü Z. Structured Knowledge Source Integration: A Progress Report. In: *Integration of Knowledge Intensive Multiagent Systems*, Cambridge, MA, USA, 2003.
9. Nishizawa H., Fujiwara M., Yokoyama M., Kanazawa S. R&D Trends for Future Networks in the USA, the EU, and Japan. *NTT Technical Review*, Vol. 7 No. 5 May 2009, p.p.1-6.
10. Reed S., Lenat D. Mapping Ontologies into Cyc. In: *AAAI 2002 Conference Workshop on Ontologies For The Semantic Web*, Edmonton, Canada, 2002.
11. Roussopoulos N.D. *A semantic network model of databases*. Toronto Univ., 1976.
12. Scott D.S. *Lectures on a mathematical theory of computations*. Oxford University Computing Laboratory Technical Monograph. PRG-19, 1981, 148 pp.
13. Witbrock M., Panton K., Reed S.L., et al. Automated OWL Annotation Assisted by a Large Knowledge Base. In: *Workshop on Knowledge Markup and Semantic Annotation at the 3rd International Semantic Web Conference (ISWC 2004)*, Hiroshima, Japan, 2004, pp. 71-80.
14. Wolfengagen V.E. Event Driven Objects. In: *Proc. CSIT'99*. Moscow, Russia, 1999, p.p.88-96.
15. Zykov S.V. ConceptModeller: Implementing a Semantically-Based Toolkit for Enterprise Applications In: *Proc. CSE-2006*, Lviv, Ukraine, October 2006, Lviv Polytechnic National University Publishers, 2006, pp.23-26.
16. Zykov S.V. Enterprise Content Management: Bridging the Academia and Industry Gap In: *Proc. i-Society 2007*, Merrillville, Indiana, USA, Oct. 7-11, 2007, Vol.I, p.p.145-152.
17. Zykov S.V. Enterprise Content Management: Theory and Engineering for Entire Lifecycle Support. In: *Proc. CSIT'2006*, Vol.1, Ufa State Aviation Technical University, USATU Publishers, Karlsruhe, Germany, 2006, pp. 86-92.
18. Zykov S.V. Integrated Methodology for Internet-Based Enterprise Information Systems Development. In: *Proc. WEBIST2005*, Miami, FL, USA, May 2005, p.p.168-175.
19. Zykov S.V. An Integral Approach to Enterprise Content Management. In: Callaos N., Lesso W., Zinn C.D., Zmazek B. (Eds.), *Proc. of 11<sup>th</sup> International World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI 2007)*, Orlando, FL, U.S.A., July 8-11, 2007, Vol.I, p.p. 212-216.
20. Zykov S.V. The Integrated Methodology for Enterprise Content Management. In: *WMSCI 2009*, July 10-13, 2009 – Orlando, FL, USA , p.p.259-264.