ENGINEERING SCIENCE

VIRTUAL RESEARCH ENVIRONMENT "OPTIMAL ENGINEERING DESIGN AS A SERVICE ECOSYSTEM"

Petrenko Anatoly

National technical university of Ukraine "Kyiv Polytechnic Institute"
Kiev, Ukraine

ARTICLE INFO

Received 5 May 2016
Accepted 8 May 2016
Published 18 May 2016

KEYWORDS

Service-oriented computing (SOC),
Web-services,
Cloud Computing,
Engineering Design Platform

ABSTRACT

This paper provides a roadmap of development of the Engineering Design Environment, based on Service-oriented computing (SOC) and intended, in particular, for modeling and optimization of Nonlinear Dynamic Systems, based on components of different physical nature and being widely spread in different scientific and engineering fields.

1. INTRODUCTION

Porting engineering applications to the grid and cloud platform can be based on the paradigm of Service-Oriented Computing (SOC) which utilizes services as the basic constructs to support the development of rapid, low-cost and easy composition of distributed applications even in heterogeneous environments. The distinction between SOC and traditional computing (say, object-oriented computing) is that application builders no longer construct software from scratch using a programing language. Instead, they specify the application logic in a high-level specification language, utilizing standard services as components [1,2]. A Web service is a specific kind of service that is identified by a URI, whose service description and transport utilize open Internet standards. This original conception of Engineering SOC with Design procedures as web-services has no complete competitor worldwide [3].

2. DESCRIPTION OF A PROBLEM SOLUTION

To accelerate the process of applications porting the following research and development activities had to be conducted:

1).Investigating Engineering Design procedures as possible services in distributed environments instead of present attempts to migrate monolithic large CAE/CAD software systems into the grid/cloud infrastructure as it is done in [4-7]. To get this it is necessary:

- to implement novel service-oriented design paradigm in Engineering according to which all levels of design including components, circuit and system levels are divided into separate loosely coupled stages and procedures for their subsequent transfer to the form of standardized web-services;
- to analyze the existing mathematical modeling and optimal design software for the possible re-use of the best algorithms and design procedures implementations in the creating the depository of applied web services;
- to develop a container with interfaces for standardized individual web-services based on international standards and protocols which allow building compositions from these web-services as design (calculations) workflows.

2). Extending of service man¬agement and monitoring facilities in a cloud computing environ-ment by making these services to be more centralized and allowing them to use interconnected multiple distributed services databases. To realize this opportunity, it is necessary to incorporate in a cloud the service-based information similar to the type of information captured in UDDI directory services and provide cross-cloud connec¬tivity to facilitate the ability to openly discover the services residing within distributed databases.

3).Using Semantic approaches for service that allow meaningful definitions of information in cloud environments by offering solutions of many service providers who may reside

within the same infrastructure by agreement on linked ontology. Third-party software agents operating within a cloud might be able to derive ontological information from the stored data and operations. Service-oriented and cloud computing combined will indeed begin to challenge the way of enterprise computing development.

4). Comparing and integrating the procedure-oriented and resource-oriented servicestaking account their advantages and constraints and using Linked Data technology [8]for combining Web services, RESTful services and Semantic Web-Services on the base of known SPARQL, RDF and other standards.

5). Re-engineering the existing service workflow tools for composition and orchestration of heterogeneous web-services (including RESTful services, Semantic web-services and traditional WS* services) into a user defined computing scenario or a Design route, which comprises a set of ontology, domain-specific heuristics, and a knowledge base to support the semi-automatic workflow composition. In particular, the ontology will cover various aspects of Engineering Design and the composition will be based knowledge advanced matchmaking algorithms based on the assumption of concept types, the properties of inputs, outputs, and data. The workflow composition

6). Developing a distributed web-services repository which provides the access to autonomous, platform-independent Design procedures of CAE / CAD tools, say, for MEMS design (operations with large-scale mathematical models, steady state analysis, transient and frequency domain analysis, sensitivity and statistical analysis, parametric optimization and optimal tolerances assignment, solution centering, etc.) and supporting procedures (cross-domain mathematical model description translation, data formats translation etc.) based on innovative original numerical methods.Algorithms proposed for many design web-services are novel and unique (multi-criterion optimization, optimal tolerances assignment, yield maximization, stiff- and ill-conditional tasks solving, etc.).

7). Providing possibilities for different research teams to contribute in web-services repository development using different programming languages and planning to implement different data from distributed sources. Due to loosely coupled web-services feathers users can modify and adapt a composed application which is preserved when some web-services are changed. Design in Engineering becomes personalized and customized because users can build and adjust their design scenario and workflow by selecting the necessary web-services (as calculation procedures) to be executed on grid/cloud resources.

Repository Services are divided to Environment Supporting Services and Application support services. The Environment Supporting Services offer the standard operations for service management and hosting (e.g. cloud hosting, event processing and management, mediation and data services, services composition and workflow, security, connectivity, messaging, storage etc.). The Application support services provide different Engineering Design procedures and processes from which a user will be able to composite his own design route in the particular engineering field.

At the "Institute for Applied Systems Analysis" of the of the National Technical University of Ukraine “Kiev Polytechnic Institute” we have successful experience in developing such service-oriented design software for modelling and optimization of Nonlinear Dynamic Systems (www.alted.kpi.ua).

3. CONCLUSION AND FUTURE WORK

Solution in hand is designed primarily to meet the needs of small and medium enterprises in the modern toolkit design of complex technical objects and technological processes, as well as the small research laboratories to perform complex computational experiments. A long-term strategy for the Engineering Design is to create flexible networked simulation and modelling tools for “bottom-up” or “top-down/ bottom-up”. It seems to be suitable to requirements of the Horizon-2020 activity EINFRA-22-2016: USER-DRIVEN E-INFRASTRUCTURE INNOVATION.

REFERENCES

1. M. N. Huhns and M. P. Singh. Service-Oriented Computing: Key Concepts and Principles. -// Published in IEEE Internet Computing, Volume 9, Issue 1, 2005, pages 75-81.

2. Y. Chen, W.T. Tsai. Distributed Service-Oriented Software Development.- //Kendall Hunt Publishing, 2008, ISBN 978-0-7575-5273-1, 467 pages.

3. Petrenko A.I., V.Ladogubets, O. Finogenov, B. Bulakh. WebALLTED: InterdisciplinaryModelling inGridand Cloud.-// Computer Science, AGH, Krakow, 14(2),2013,ISSN 1508-2806, pages 295-306.

4. TINACloudproject home: http://www.tina.com /English/ tina/

5. PartSimproject home: http://www.partsim.com/ examples

6. RT-LAB project home: http://www.opal-rt.com/company/company-profile/

7. FineSim Pro project home: http:// www.automation.com/ content/magmas-latest-version-of-finesim-pro-delivers-3x-faster-runtime/

8. LinkedData technology project home: http:// linkeddatabook.com/editions/1.0/