CONCEPT OF INTEGRATED ENVIRONMENT FOR NETWORK DESIGN

Andriy Luntovskyy, Dietbert Gütter, Alexander Schill, Ulrich Winkler
Department Computer Science
Dresden University of Technology
E-mail:uw469478@mail.inf.tu-dresden.de

Abstract - This work examines CANDy and NDML concepts. CANDy, Computer Aided Network Design utility is a concept for an efficient XML-based integrated network design environment. Its integrating components are NDML (Network Design Markup Language), an XML-based problem-oriented language developed for representation of information components related to network projecting within documents. Its design supports interoperability with the XML family of tools and protocols. It provides a set of components to meet the needs of network design.

.

I. Introduction

Presently computer networks are mostly designed by experts, using their high amount of network related knowledge and experiences. Apart from creative work there is also a lot of routine work.

It exist the possibility to design computer networks with the use of some existing design tools, but the automated tasks are unfortunately few. A concept for an efficient XML-based design system is examined in this work. The following aspects are essential: to analyze and to apply basic logical-mathematical models, standards, languages, interfaces and design applications, to create a design workflow and to develop a concept of a design workbench for computer networks.

All the above mentioned distinguished features unfortunately exist only partially in the presently used tools, without clear interfaces and special design languages.

Our aim is to describe an *Integrated Design Environment* with the following properties: openness due the use of XML interchange data format, design stage inheritance in the frame of existing workflows, design stage inheritance in the frame of existing workflows and openness of applied simulation tools [1],[2],[4].

II. The CANDy Approach

Existing stages of network designing are development of network concepts using existing patterns and heuristics for LANs, visualization, structure analysis, design rule checking, testing, proposals for administration and network management [5]. Additional quantitative models could be used to simulate the load and capacity behavior and to gain accurate cost estimation [3]. The result of the whole design process is a set of different documents - either in human or computer readable formats. With that design the network can be installed by a company. The network model could be used for testing and validating the installation and be the basics for monitoring tools supporting the work of the administrator. The whole process is supported by workflows, tools and methods for content management and creation, and business process management.

The central elements in our approach is a informational model for computer networks, which we call *Network Design Model (NDM)* and a XML based, design language *NDML (Network Design Markup Language).* Both are used by CANDy (*Computer-Aided Network Design utility).* CANDy will provide a workbench for designing computer networks.

III. NDML

As a integrating component for design applications acts a XML based, problem-oriented *NDML - Network Design Markup Language*, which organizes mapping of the documents for corporate networks design as well as their wireless sub-systems.

NDML is developed for representing information components related to network projecting within documents. It's designed to support interoperability within the XML family tools and protocols. It provides a set of components to meet the needs of network design.

XML will soon be Esperanto for design applications due to increasing support by modern software products [3]. The XML language can be simultaneously used as integrating component and communication interface because the model of a computer network can be described via XML, a common format facilitates the cooperation of several tools and XML organizes mapping of design objects and communication protocols to design applications. Therefore, XML can be used as an integration component as well as a communication interface.

NDML1.x dialects are developed in the frame of CANDY-Project (2002-2005) and had the following important disadvantages: first of all it was initially defined as a completely declarative description language (not-procedural, then e.g. no workflows descriptions) and we used only the web standards HTML, XML, XSL, XSLT, XPath.

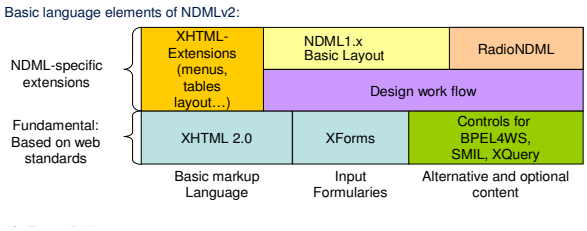

Figure 1 NDMLv2 as integration component (Orlando, 2004-2005)

In the future NDMLv2 will become procedural (with design workflows statements) and will be based on XHTML 2.0, XForms, SMIL, BPEL4WS and XQuery standards. Mapping into XML-DB will be optionally provided in the new NDML-Language-Profile. Modified NDMLv2 can be used by mobile design applications on end-device of different types. XSLT-architecture elements are aimed for:

- *Device Recognition* – end-device supporting
- *Reducer* – document text reducing & normalization
- *Rules Engine* – document semantic checking
- *Paginator/Navigator* – document splitting into displayed segments and segment navigation

- *XForms-Processor* – forms (formulary) processing for e.g. DB-queries
- *Markup Mapper* – trans-coding into target formats.

Therefore, the *NDML*-content for design can be invariant represented on mobile end-devices with limited screen resolution and navigational abilities (PDA, mobile phone).

A special data format *RadioNDML for WLAN* is used in the frame of actually created modification of the language - NDML2.0 (fig.5)

IV. NDM

The NDM is built on the basis of the topology descriptions of the network in NDML augmented with functionality of each concrete design application.

The NDM is defined by the Model M

$$NDM = M(\alpha, F, L, B, T, R)$$

where

α	design document algebra
F	formula calculus for documents
L	design language
B	type of design document retrieving
T	syntax tree obtained on the basis of ERA- representation normalizing
R	corresponding relations

The following components of given sextuple are analyzed

- (α, F) – manipulation component of NDM for design documents
- L – linguistic component of NDM, represented via NDML
- (B, T, R) – OO–relational structures of NDM retrieving

The basic operations for α - algebra for documents *(theoretical-set- and specific document operations)* are introduced and their properties are described in our next publication. The elaborated model NDM enables to manipulate with XML-based design documents on the basis of proposed *algebra and formula calculus*, as well as to retrieve the above mentioned documents in the *repository* in form of *XML-based data structure (syntax tree)* or, dual, *OO-relational data base.* The α - algebra can be definitely mapped onto *design language L*.

The position of NDM within the general design workflow is shown in figure 3.

V. CANDy

CANDy, as the Design Utility, consists both of new components such as NDML and its tools as well as of existing tools such as NS-2 that are tightly integrated via appropriate interfaces.

The architecture of the CANDY's concept of the Integrated Development Environment for network design is given in figure 2.

Besides some management tools like the ProjectManger, a basic element of CANDy is the NetEditor, a graphical user interface helping the planer to develop a basic net concept. Supported by a rule-checker, optimizing- and simulating tools the network concept will be iterative improved and corrected and almost optimal design could be reached. The BillReport supports the Manager to get an idea of the costs. At eacht step is it possible to go back and improve and adapt the network design.

The combination of developed tools by the CANDY-team and third-party tools will give us enough flexibility to test and improve further development stages and ideas and helps us to stay in touch with today's state-of-the-art technologies and current developments in the field of network design. Furthermore the CANDY-Framework provides for us a playground for scientific works and educational purposes.

An overview over all tools provided or supported by CANDy is given in table 1.

All tools and components developed by our CANDy team will be written in JAVA and will be released under an Open Source Licenses (which one is not decided yet). In fig.3,4 the architecture of *CANDY* as an XML/XSL–based CAD–Tool for Network Design is represented.

Tools provided by CANDy	Third party tools
Project Manager NetEditor NDML-Editor Rule Checker Bill Reporter Queuing Analytical Tool Multivariate Statistical Analysis Linear Optimization Documentation Tool Repository/Database XSLT-Converters	*MS Visio, Sourceforge Dia* *AutoDesk / AutoCAD* NS-2 Acrobat-PDF *Export/Import-Tools like MS Office*

Tabel 1 - CANDY provided tools (or in planning)

For the future developing further front-ends for the following important simulators are planed: Berkley Network Simulator NS-2, OPNET IT Guru, Modeler, WDM Guru, Ekahau Positioning Engine, Site Survey, Wi-Fi Tag, HP/Compuware Corporation COMNET III.

Furthermore an optimization of the CANDY-database will be achieved by use of OO-relational XML-DB driven by XQuery.

VI. References

[1] *A. Luntovskyy, A. Schill, D. Gütter, G. Pfeifer, A. Panchenko,* CANDY: Integrated Environment for Network Design. Proceedings of SCI 2004, Orlando, pp. 252-257, ISBN 980-6560-13-2

[2] *Luntovskyy, A., Schill, A., D. Gütter, G. Pfeifer, A. Panchenko*, V. Vasyutynskyy: Computer Network Modeling and Analysis Using XML-Descriptions; The 9th World Multi-Conference on SYSTEMICS, CYBERNETICS AND INFORMATICS (WMSCI 2005), July 10-13, 2005, Orlando, Florida, USA, pp. 283-288, ISBN 980-6560-54-3

[3] *Buchholz, P.* Kapazitätsplanung und Leistungsbewertung verteilter Systeme, Manuskript zur Lehrveranstaltung an der Professur für Modellierung und Simulation (Institut für Angewandte Informatik), 2003

[4] Schill, A.: Vorlesungsskripte an der Professur RN, http://www.rn.inf.tu-dresden.de/scripts_lsrn/Lehre/, 2005

[5] Sloman, M.: NETWORK AND DISTRIBUTED SYSTEMS MANAGEMENT/ Imperial College of Science, Technology and Medicine at University of London/ Addison-Wesley, 1994, 665 pages

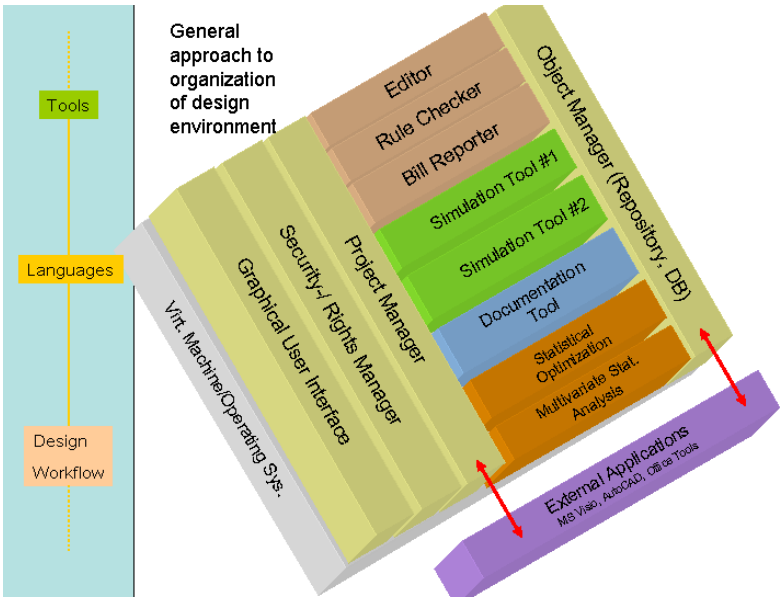

Figure 2. Development Environment, Orlando 2004

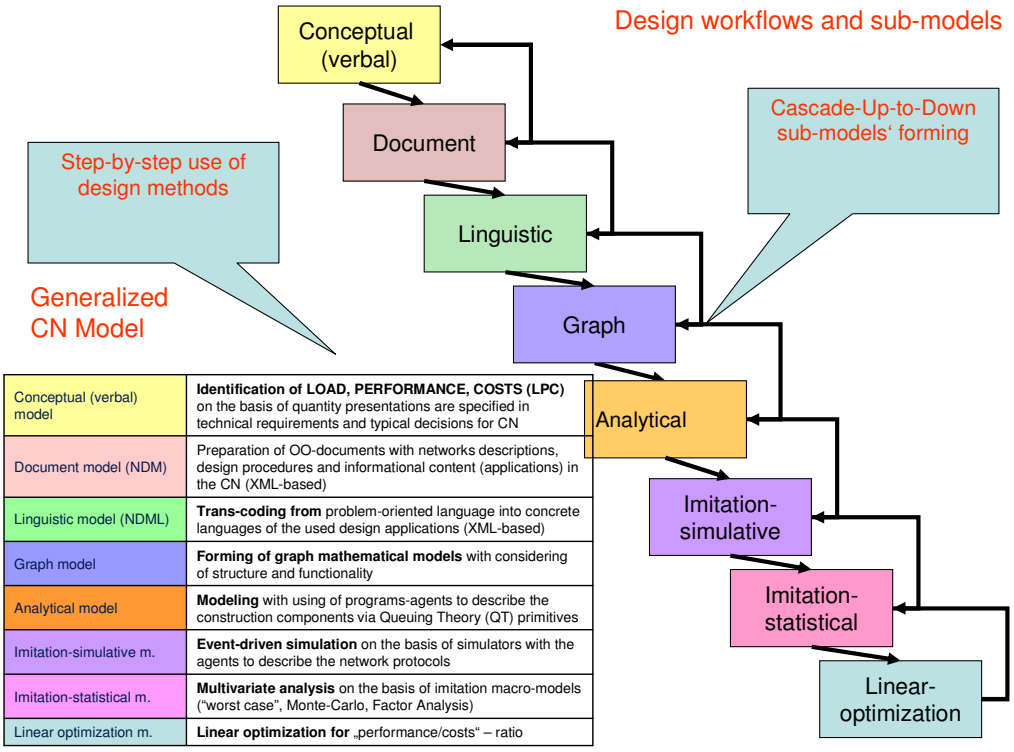

Figure 3. NDM and NDML as of the general design workflow