NETWORK DESIGN AND OPTIMIZATION UNDER USE OF CANDY FRAMEWORK

Andriy Luntovskyy, Dietbert Gütter, Alexander Schill
Dresden University of Technology, Department of Computer Science,
Hans-Grundig-Straße 25, 01062 Dresden, Germany
guetter@rn.inf.tu-dresden.de, http://www.inf.tu-dresden.de

Abstract - **CANDY, Computer-Aided Network Design utility** is a Java-framework for an efficient XML-based integrated network design environment developed via ***CANDY@TUD initiative*** [1]. This paper examines CANDY network design tools including graphical input and topology verification, cabling system tracing and AP/BS constellation (AP – Access Point, BS – Basis Station), performance analysis with feed-backs to re-engineering procedures. In the investigation focus stand cabling system geometry and costs optimization. The integrating component for these design tools is **NDML (Network Design Markup Language),** XML-based problem-oriented language developed for representation of design data and workflow consisting of the following special parts: NDML Basics, Topology, Performance, RadioNDML for Radio-networks etc.

I. CANDY Framework

The architecture of CANDY [2,4,5] is given in Fig.1. The development paradigm for CANDY is deployment of loose tools and components coupling, representation via open Internet projects with plug-ins and web services-ready functionality (like e.g. EJB, Web Services, Eclipse RCP). Different views on integrated office communication and automation networks are also considered, i.e. topology with use of structured cabling and WLAN/WiMAX-routes [3], cost bills, performance and QoS analysis. The integration of the tools is ensured using a common object model described in NDML and a project manager, unifying the project work flows. ***CANDY@TUD*** and ***NDML*** are registered at Dresden University of Technology. Today this project complexity exceeds approx. 10 human-years and 25,000 Java-code strings. CANDY tools can be easily encapsulated for different aims correspondently to the principle "CAD in CAD". Thus, CANDY Economics, CANDY Wireless, CANDY Automation can by delivered or their further derivations easily developed via contributive tele-working due to open [source] project. CANDY design descriptions are based specially on a document model, so called ***NDM, Network Design Model.*** NDM is a further development of relation algebra by Codd-Chen (1970-76) for OO- document-driven representation and interfaced to network design aims. The layers (CN sub-models) are important for 8-layered design. These layers are:

- conceptual with load/performance/costs-specifications
- document-based, i.e. NDM
- linguistic, i.e. NDML
- graph-based
- queuing modeling
- event simulation
- statistical
- sub]optimization based (generally 8 layers).

WWW, FTP, Email, data base transactions, deployment of modern distributed applications for a CN are important traffic specifications and can be considered conceptual sub-model with load/performance/costs-specifications. The properties of the NDM are:

- NDM can be mapped to each OO-structure or XML-based (Relational) DB
- NDM can be mapped into NDML
- NDM can be parsed via NDML with use of standard parsers for different "encapsulated CADs" and maintained design tools

NDML like each *ML (mark-up language) is flexible language with extensible grammar. Besides ***NDML*** is not only declarative but also procedural - RPC-able, Eclipse RCP or WebServices-able with workflow. So, NDML is not only a description language (declarative) but also is procedural (SMIL, PNML -Petri Net Markup Language).

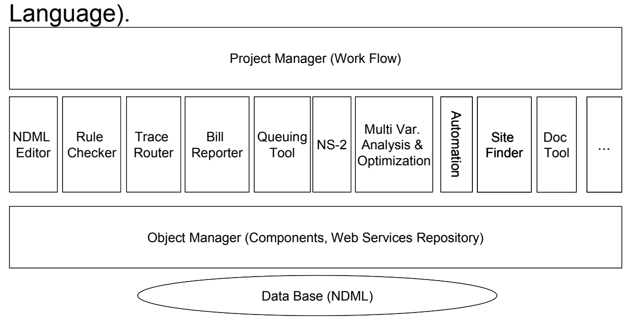


Fig.1. Extensible CANDY architecture

The brief characteristics overview of available CANDY tools is specified below:

1. ***NDML Editor*** allows the graphical input of building contour and campus map as well as office communication and automation network (PC, gateways, routers, switches, hubs, AP, cables, automation nodes etc.).
2. ***Rule Checker*** controls the common design rules like network configuration, use of network components (switches, routers etc.) with coupled transfer media (fiber optic, cooper cables, radio routes) as well as further workload constraints
3. ***Trace Router*** allows optimization of tracing and implementation of structured cabling system at the building for Ethernet LAN IEEE802.3 with considering of wireless routes via WLAN IEEE802.11.
4. ***Site Finder*** enables WLAN Access Point constellation optimization.
5. ***Queuing Tool and NS-2*** facilitate both the detailed performance analysis for complex networks. Asymptotical prediction of network behavior is made via ***Queuing Tool*** (throughput, latencies). Accurate performance and QoS (data rate, delays, and jitter) simulation under TCP/IP protocols use is carried out via ***NS-2 standard freeware simulator*** with ***NDML front end***.
6. ***Bill Reporter*** generates costs overview for whole system.

7. ***Multivariate Analysis and Optimization*** block is aimed to prediction of network performance and increasing of "performance-cost" –ratio.
8. ***Doc Tool*** allows the consistent retrieving of distributed project data for CANDY in NDML descriptions mapped on data bases as well as other target formats like PDF, HTML etc. and persistent backup at Repository of *CANDY-specific objects, components and Web Services*.
9. ***Automation network*** design tool allow integration of automation issues into CANDY framework. The integrated design of LON-based building automation networks and dedicated wireless/wired gateways together with wireless/wired building or campus networks became actually of great meaning.

II. Solving the design and optimization problems via CANDY

Communication wired/wireless networks (CN) constructed on the basis of modern standards (SCS – structured cabling system, Ethernet, ATM, WLAN, WiMAX) have as a Design Object complex hierarchical and heterogeneous character (Fig. 2).**The main design problems for CN are:**

- ***Composition problem, arranging or area combining*** for filial firms, campus, buildings, floors, recreations, ISP, telco operators
- ***Constellation problem for active network devices***, like PC, NIC, Eth-switches, IP-routers, firewalls, WLAN 802.11 AP and WiMAX 802.16 BS
- ***Tracing problem*** for structured cabling subsystems in the LAN 802.3.

The similar problems are solved also for multiple other fields, for instance, by PCB/VLSI design, mechanical engineering, building construction, architecture, planning, logistics and business management. Generally all above mentioned problems have high computational complexity (NP-full) and only heuristic issues. Among the CN most significant parameters are: throughput/data rate ***DR,*** delays and jitter effects, reliability, quality of service ***QoS*** and costs ***K***. The common CN design methodology is aimed to the optimization (mini-max-problem solving) of overall costs [6] under certain performance constraints (data rate, delays, jitter) obtained via graph-based geometrical, queuing and/or event-driven simulation for Ethernet- and WLAN/WiMAX- routes of building (campus) network. One of general ideas to CN project optimization (by U.Herzog) is the treatment of overall costs as the generalized optimization criteria (Fig.3) how it is shown below [7]:

$$\mathrm{Min}\ K(N,L,DR,t)\,|\ \mathrm{Max}\ DR$$

$$K(N,L,DR,t) = (a + \alpha \cdot L + \beta \cdot N + DR \cdot (\gamma + \eta \cdot L)) \cdot \mu(t) \quad (1)$$

$$\mu(t) = 1 + ((1+z)^{t} - 1)^{-1} + w/z$$

where ***K*** – overall cost function for enterprise network, ***N*** – number of used network devices, ***L*** – common cabling system length, ***DR*** – data rate, ***a*** – constant investment, ***μ(t)*** – yearly actual cash value (mapping of deployment, amortization, modification, operation phases*),* ***z*** – yearly amortization ratio*,* ***w*** – yearly operation costs*,* ***T*** – average life time.

III. Modified algorithm by Dijkstra

Especially the tracing problem for cabling sub-systems in CN can be solved via use of the following algorithms:

- Dynamic Programming Algorithm by Bellman-Ford
- Minimal Trees Construction by Dijkstra and other „greedy"-class-algorithms
- A*-algorithm
- Lee Wave Algorithm, Penalty Algorithms etc.

As the common constraints and limitations act:

- **LAN 802.3:** building plans, SCS-norms, ISP requirements
- **WLAN 802.11:** EM-wave propagation conditions
- **WiMAX 802.16:** EM-wave propagation conditions, digital maps, clearance models and ray-optical methods.

The implemented Dijkstra algorithm belongs to the class of sub-optimal „greedy algorithms"and calculates all shorter paths from the start node to all other nodes. Therefore, the criteria ***L*** as well as objective ***K*** for overall costs are minimized. The algorithm complexity is ***$O(n^2)$***. The method use unlike "Dijkstra pur" the coded adjacency matrix represented in Triple-Form (Fig.4):

$$G = G(V, E, Cost) \rightarrow AdjacencyMatrix \rightarrow$$
$$\rightarrow TripleListS = \{(<Start>, <Target>, <Cost>)\} (2)$$

where ***Cost*** – corresponds to the integrated criteria for performance, maximal length and loading [8].

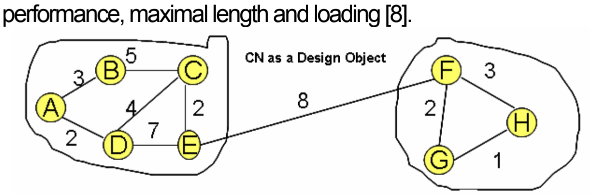


Fig.2. CN as a Design Object: hierarchical character

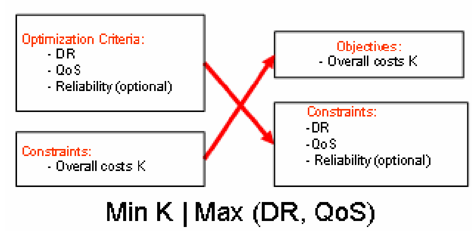


Fig.3. Overall costs as criteria and performance parameters as constraints

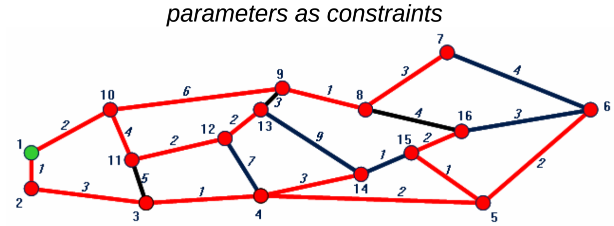


Fig.4. Overall costs as criteria

The modified algorithm by Dijkstra is implemented in ***CANDY Trace Router*** tool and is given on pseudo-code the below:

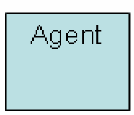
INPUT: Starttriple s, V-set
RESULT: shortest path S
DATA: int length, Triple search_triple, TripleSet D, TripleList S
START:
// Init block
length = 0
search_triple = (s.start, s.start, 0)

```
// To find all from start node reachable nodes
FOREACH v from V DO
IF s.from == v.from THEN D = D [ {v}
// Short pathes from Start
WHILE D.size > 0 or s.to != search_triple.to DO
//function minimum() finds from the triple list S one with
the minimal cost
// the order in the list S is important; ranging is
necessary
search_triple = minimum(D)
D = D exclude {search_triple}
E = E union {search_triple}
FOREACH v from V DO
IF search_triple.to == v.from THEN
FOREACH d from D DO
IF v notfrom E and (v notfrom D or (v.costs < d.costs
and v.target == d.target)) THEN
D = D union {v}
// To find the path from Start to Target
search_triple = (s.target, s.target, 0)
WHILE s.start != getFirst(S).start DO
FOREACH e from E DO
IF search_triple.from == e.to THEN
S = PutFirst(S, e)
length = length + e.costs
search_triple = e
break
// evaluation
IF S.size == 0 THEN
return "NO PATH CALCULATED"
IF length > s.costs THEN
return "THE FOUND PATH TOO LANG!"
ELSE return S
```

OUTPUT: ***shortest path S***

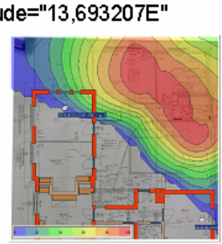
Certain examples of the wired/wireless network descriptions via NDML/RadioNDML are given below.

IV. NDML examples

1	Workstation 	<device id="dev_ws25" name="ArbStat25" description="Sun Ultra 10"> </device>
2	NIC 	<nic id="nic1" home="ArbStat25" ...> <l_nicType l_wireless="OFF" .../> <technology>100 Base T</technology> </nic>
3	Medium 	<medium id="medeth01" name="stp cat. 5"> <connectedNic id=„nic1"/> <connectedNic id=„..."/> </medium>
4	Agent 	<device ...> <l_agent .. l_type="Null"/> <l_agent ... l_type="UDP" l_target="..."> <l_application .../> </l_agent> </device> <device id="dev_ws25" name="ArbStat25" description="Sun Ultra 10"> <l_agent id="tcpsender2" l_target="tcpreceiver4" l_type="TCP"> <l_application id="ftp2" l_type="FTP" ... /> </l_agent> <l_agent id="tcpreceiver2" l_type="TCPSink"/> </device>

V. RadioNDML examples

```
<project id="rn01" ... >
<scenario height="100000" width="100000"
ltlatitude="51,031551N" ltlongitude="13,693207E"
rblatitude="51,032551N"
rblongitude="13,694207E">
<devices> ... </devices>
<users> ... </users>
<nics> ... </nics>
<enviroment> ...</enviroment>
</szenario>
</project>
```



(a) Coordinates for AP/BS

```
<device id="ap01" type="Accesspoint"
xposition="80000" yposition="65000"
zposition="8000" hdirection="270"
vdirection="0">
<ssid>Wireless</ssid>
<antenna rid="a01"/>
<product rid="p01"/>
<c_user>
<user number="10" uid="u01"/>
<user number="4" uid="u02"/>
</c_user>
</device>
```

(b) Relative coordinates AP/BS

<antenna id="a02" name="Huber + Suhner
Omni" gain="8" horizontalangle="360"
verticalangle="20">
<costs id="c02">
<unit_price>90,00</unit_price>
<price>90,00</price>
</costs>
</antenna>

(c) Antenna type

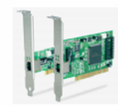
<product bandwidth="54" id="p01"
name="WRT54g"
technologie="802.11g" type="Accesspoint"
vendor="Linksys">
<transmitpower>15</transmitpower>
<costs id="c03">
<unit_price>60,00</unit_price>
<price>60,00</price>
</costs>
</product>

(d) AP type

<users>
<utype id="u01" name="student" presence="20"
ratio="10:90" t_avr="30" t_max="200"
t_min="10"/>
<utype id="u02" name="scientist"
presence="35" ratio="40:60" t_avr="20"
t_max="100" t_min="3"/>
</users>

(e) User profile

<nics>
<nic home="ap01" id="nicap01">
<l_nicType l_satelliteOriented="OFF"
l_wired="ON" l_wireless="ON"/>
</nic>
<nic home="ap02" id="nicap2">
<l_nicType l_satelliteOriented="OFF"
l_wired="ON" l_wireless="ON"/>
</nic>
</nics>



(f) Wired/Wireless NIC type

Conclusions

The CANDY Framework can be deployed for solving of complex design and optimization problems for CN built on the basis of IEEE 802.3, 802.11, 802.16 as well as EN50173 SCS-standards (constellation, tracing tasks). The used CN design methodology is aimed to the mini-max- optimization problem for overall costs under certain performance constraints like data rate, delays, and jitter. The solution is obtained via use of implemented CANDY tools for graph-based geometrical, queuing and event-driven simulation.

References

1. CANDY@TUD Learning Project: http://www.rn.inf.tu-dresden.de
2. Luntovskyy, A., Gütter, D., Schill, A., Winkler, U.: Concept of an Integrated Environment for Network Design; IEEE CriMiCo Conference, Sevastopol, Sept. 2005, pp. 959-961 (ISBN966-7968-79-0)
3. Luntovskyy, A., Gütter, D., Schill, A., Winkler, U.: Design Particularities for Wireless Networks; IEEE CriMiCo Conference, Sevastopol, Sept. 2005, pp. 955-958 (ISBN966-7968-79-0)
4. 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
5. 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
6. M.Sloman. NETWORK AND DISTRIBUTED SYSTEMS MANAGEMENT/ Imperial College of Science, Technology and Medicine at University of London/ Addison-Wesley, 1994, 665p.
7. W.Lehnert: Planung & Optimierung v. Telekomm.-Netzen; Manuskript TUD, WS 04/05
8. Modified Dijkstra: http://www.orchid.inf.tu-dresden.de/gdp/