Self-configuration of Multiservice IP Networks

SDDI 2005

Formalize SLSs

- SLS XML Schema
- SLS XML Examples
- SLS XML Validation
- SLS XML to CLIPs

System Configuration

- CLIPs Rules For Network Management
- CLIPs Rules For Topology
- CLIPs Rules For Diffserv Mappings
- CLIPs Rules For MPLS Mappings
- CLIPs Extensions to Create Network Simulator Scripts

Abstract

The use of Internet as a global communication infra-structure and as the base technology for a large number of applications and services, presented new demands and challenges to the quality of service provided by ISPs (Internet Service Providers). The Internet is a datagram network. Thus, it was not conceived to provide levels of QoS (Quality of Service) suitable to such services and applications demands providing by default a best-effort sevice.The network is not capable to distinguish traffic streams to fulfill specific requirements of its delay and loss. This behavior is more evident when there is a high usage of the network. Performance will drop significantly.

To overcome this problem, TCP/IP stack has been evolving to be a platform of convergence, integrating existing and future communication services. Particularly, IETF (Internet Engineering Task Force) has been presenting several proposals of models and mechanisms to provide Internet with the adequate QoS levels. The most important proposals are Integrated Services model [1], Differentiated Services model [2, 3], IP/MPLS (Multi-Protocol Label Switching) [4], GMPLS (Generalized MPLS) [5, 6], Traffic Engineering [7] and Constraint Based Routing [8]. The service class model is the most feasible model to implement because of scalability. Assembling network traffic into several classes is simpler to manage than having to deal with a large number of traffic flows. Therefore, ISPs may offer a limited set of different QoS classes. ISPs are responding to the market needs while keeping a technologically feasible solution. Furthermore, the main switch and router builders have already implemented, partially or totally, solutions based on QoS classes. Examples of such networks are the Internet2 [9] in the U.S.A and project Geant [10] in Europe.

Implementation and support for QoS in IP networks demands a larger traffic control and network management, though. The different QoS classes have to be assured, and, at the same time, a efficient usage of the network resources is required. Although the class QoS model [2] focus the complexity on the domain's borders, where network traffic is marked according to its class, inside the domain, such control is relieved because of performance issues. Therefore, it is difficult to assure those QoS classes inside the domain [11], i. e., it is difficult to ensure the compliment of the agreed QoS parameters negotiated thru SLAs (Service Level Agreements). If the QoS parameter's specification is done by customer and not by class, assuring QoS is a more difficult problem.

The main objective of this work is to automate network management and optimize configuration in order to guarantee the QoS performance parameters specified in the SLAs. Thus, several steps are required such as: (i) formalizing the technical parts of SLAs called Service Level Specifications (SLSs); (ii) mapping the SLSs into network configurations independently of the underlying technology (DiffServ, MPLS, others); (iii) mapping and testing configurations in a simulated environment; and (iv) optimize device configurations.

Network Monitoring

- CLIPs Extensions to Monitorize QoS Performance Values From Network Simulator
- CLIPs Extensions and Rules to Trigger Optimization Algorithms

Network Optimization

- Configuration and Implementation of Evolutionary Computation Algorithms to Optimize Network Configurations

References

[1] R. Braden, D. Clark, and S. Shenker. Integrated Services in the Internet Architecture: an Overview. IETF RFC1633, 1994.
[2] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Services. IETF RFC 2475, 1998.
[3] D. Grossman. New Terminology and Clarifications for Diffserv. IETF RFC3260, April 2002.
[4] R. Callon, E. Rosen, and A. Viswanathan. Multiprotocol Label Switching Architecture. IETF RFC3031, January 2001.
[5] E. Mannie. GMPLS Architecture. IETF draft: draft-ietf-ccamp-gmpls-architecture-07.txt (work in progress), May 2003.
[6] A. Banerjee et al. Generalized Multiprotocol Label Switching: An Overview of Routing and Management Enhancements. IEEE Communications Magazine, 39(1), January 2001.
[7] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao. Overview and Principles of Internet Traffic Engineering. IETF RFC3272, May 2002.
[8] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus. Requirements for Traffic Engineering Over MPLS. IETF RFC 2702, 1999.
[9] Internet2 QoS Working Group Charter, March 2001.
[10] DANTE - Geant Project. URL: http://www.dante.net/server/show/nav.007, 2000.
[11] K. Nichols and B. Carpenter. Definition of Diff. Services Per Domain Behaviors and Rules for their Specification. IETF RFC 3086, 2001.

Pedro Miguel Félix Alípio

Supervisors:
Paulo Manuel Martins de Carvalho
José Carlos Ferreira Maia Neves