PONDER for Specifying the Policies of Multimedia Sessions Management

A. Sekkaki[1], M. El Hamzaoui[2], B. Bensassi[3], C. B. Westphall[4], and C.M. Westphall[5]
[1,2,3]University Hassan II Aïn-Chok, Faculty of Sciences
P.O Box 5366, Maarif
Casablanca. Morocco.
[1]a.sekkaki@fsac.ac.ma, [2,3]{m_elhamzaoui, bahloul_bensassi}@yahoo.fr

[4,5]Federal University of Santa Catarina
Network and Management Laboratory
Caixa Postal 476, 88040-970, Florianopolis – SC – Brazil
[4,5]{westphal, carla}@lrg.ufsc.br

Abstract. Distributed multimedia systems allow a group of persons to see each other, communicate and work together on shared resources independently on their geographical positions. The multimedia systems rest on session concept to organize users in well defined groups. Many works realized inside the IETF (Internet Engineering Task Force) are devoted to the standardization of description, annoncement and initiation of the multimedia sessions on Internet. However, the problem of coordination within multimedia sessions is not dealt with in these works. In this paper, we will use the policy-based management to manage automatically the coordination inside teleconference systems sessions that belongs to multimedia distributed systems. Our approach is based exactly on the policies specification language Ponder to specify policies designed to manage the intra-session coordination

Keywords: Domains, Multimedia session, Ponder, SAP, SDP, SIP, Teleconference system.

(Received July 24, 2006 / Accepted January 03, 2007)

1. Introduction

Because of the important evolution of the multimedia domain and the appearance of high speed networks (compression advanced techniques and the exchange of any information type), the distributed multimedia applications are expansionary evolved. Consequently, the videoconference became currently possible on Internet thanks to this enormous evolution of distributed systems.

Distributed systems are composed of a certain number of autonomous processes. The preliminary stages of the distributed systems implementation must be assisted by analysis and conception methods. These methods are based on some formalisms such as Finite State Machine [02], Petri nets [15], Process Algebra [01] and Temporal Logic [19], that allow a formal description of the behaviour of the various entities interacting inside distributed systems. In this context, many formal specification languages such as ESTELLE (ISO-IS 9074) [26], LOTOS (ISO-IS 8807) [18] and SDL [16] were standardized to implement these formalisms. Moreover, the advent of the object-oriented approach led to the appearance of purely object-oriented specification languages such as Mondel [04]. In the same way, a lot of validation (Ved [17], Geode [03] and SPIN [14]) were developed to assist the various phases of a formal specification.

Concerning the IETF efforts devoted to the multimedia sessions, on one hand this organization could structure the description and annoncement of multimedia sessions on Internet through the protocols SDP (Session Protocol Description) [13] and SAP (Session Announcement protocol) [11] and on other

hand, IETF could also structured the establishment of the multimedia sessions through SIP protocol (Session Protocol Initiation) [23]. However, SIP protocol does not deal with both the intra and inter-multimedia sessions controls and coordination

In this work, we aim to employ the policy-based management to resolve the intra-multimedia session coordination problem and more precisely we will show through our approach how PONDER [06] could facilitate the coordination management tasks. Our approach will also rest on the various descriptions and terminologies of the SDP, SAP, and SIP protocols.

This paper will be presented as follows; in the second section we will present the principle of the policy-based management. The third section will briefly display the ponder policy language. Concerning the fourth section, it will include a presentation of the SDP, SAP and SIP protocols. Our approach will be the subject of the fifth section while related works will be presented in the sixth section. Finally, the conclusion and perspective of this work will be featured in the last section.

2. Principle of the policy-based management

The objective of the policy-based management is the optimisation as well as possible of administrators efforts. Thus, it first consists in determining the strategies and the tactics reflecting the managers' objective and also representing them in policies' form. Then, these policies must be presented as a set of rules to be understood by the management entities and stored in a Policy Repository (PR). The distribution and the application of these policies require to have these rules communicated to a PDP (Policy Decision Point) and to PEPs (Policy Enforcement Point) managed by this latter [27].

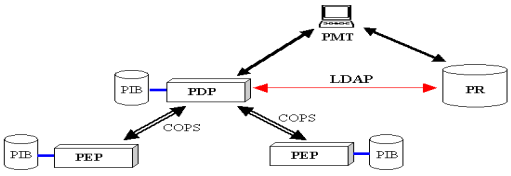

Figure 1: IETF Policy-based management Platform

Concerning this management platform (fig.1), both a Policy Management Tool (PMT) and a PR must be placed on the higher level to allow administrator to configure the application level policies and store them afterwards in the PR (network policies level). The policies stored in the PEP PIB (Policy Information Bases) are called equipment level policies while those stored in the PDP PIB are called network level policies.

A second terminology was employed in the works devoted to management inside Imperial College [28]. This terminology uses the notions of Subject and Target instead of those of PDP and PEP. On one hand, subjects indicate the manager objects and on other hand, targets indicate the managed objects. The relations between subjects and targets are defined through the management policies and they also depend on the type of these latter. Thus, obligation policies define what a subject must perform or not on the level of a target, while authorisation policies specify the access rights that could have a subject on the level of a target.

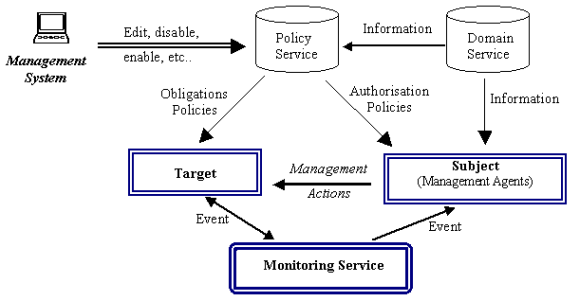

Figure 2: Management Platform based on the notions of subject and target

PDP/PEP and Subject/Target are not contradictory concepts but they are complementary. Thus, the PDP could play the subject role and in the same way the PEP could play the target role. The management platforms basing on the Subject and Target concepts (fig.2) use generally a Monitoring Service (MS) to automate the management. Moreover, the PR is replaced by both a domain service and a policy service. Concerning our work, the policies specification will be based on the Ponder language that appears actually as the best tool for specifying both security and management policies.

3. Ponder Policies Specification language

Ponder [06] is an object-oriented, declarative language for specifying security and management policies for distributed system. Like any object-oriented languages, Ponder provides reuse by supporting types definition, which can be instantiated for each specific situation by passing necessary parameters.

The Ponder is characterized by four basic specifications that are access control specification, obligation policy specification, constraints specification, and composite policies specification.

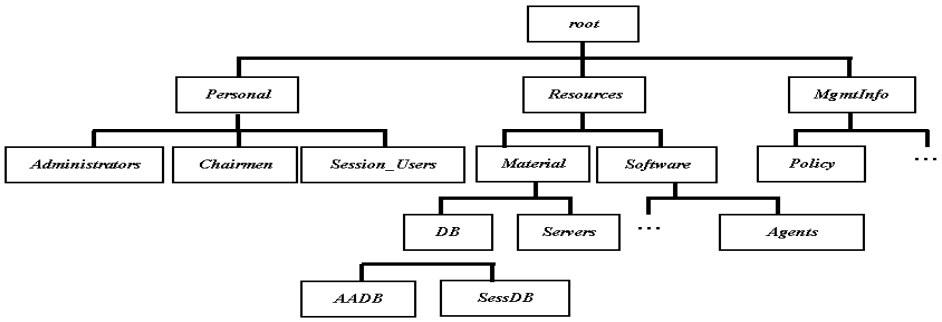

Figure 3: Organization in domains of our management environment

For Ponder, subjects, targets, and policies are all organised in domains. A domain is very similar to a directory or folder on a personal computer, and it is used to partition large systems according to some precise criteria [25]. Domains make the management of distributed systems very easy and flexible and give the possibility to modify the domains' components without altering management policies. The organisation in domains of the components of our security management environment is illustrated on figure 3.

A **root** domain is designed, as it is schematised in fig.3, to organize the different components of our multimedia management environment in three main sub-domains: **Personal**, **System** and **MgmtInfo**.

The domain **Personal** contains the administrators, chairmen and sessions' users references. The domain **Resources** includes the references of all resources (material and software) of our multimedia management environment. Finally, the domain **MgmtInfo** regroups the references of the objects representing the necessary management information.

In what follows, we will restrict our discussion on obligation policies. Obligation policies specify what activities a subject (members of one or several domains) must do to a set of target objects (objects of one or several domains) and define the duties of the policy subject. Obligation policies are triggered by events and are normally interpreted by a manager agent at the subject. Ponder defines two syntaxes to specify obligation policies:

- Syntax for direct declaration of an obligation policy instance:

```
inst oblig  policyName    "{"
    on                        event-specification ;
    Subject       [<type>]    domain-Scope-Expression ;
    [ Target      [<type>]    domain-Scope-Expression ; ]
    do                        obligation-action-list ;
    [ catch                   exception-specification ; ]
    [ when                    constraint-Expression ; ]    "}"
```

The key word on specifies the required event. Subject and target are expressed in term of domains. The optional **catch-clause** specifies an exception that is executed if the actions fail to execute for some reason.

- Syntax for declaration and instantiation of an obligation policy type:

```
Type oblig  policyType "(" formalParameters ")"  "{" {obligationaction-policy-parts } "}"
inst oblig  policyName= policyType "(" actualParameters ")" ;
```

The obligation policy type is initially declared, then instantiated.

Ponder was used in many research works. Thus, Lymberopoulos et al. showed, in [21], how PONDER policies can be implemented and validated for Differentiated Services (DiffSer) by using CIM (Common Information Model) as the modeling framework for network resources as this device independent. They also used, in [20], Ponder language to realize a dynamic adaptation of policies in response to changes could occur within the managed environment. Finally, Damianou et al. presented in [05] the implementation of an integrated toolkit for the specification, deployment and management of policies specified in PONDER language. Concerning our research group, we used Ponder to specify the management of virtual laboratories security [08][24], manage the distribution of IPsec security policies [07], resolve the Diffie-Hellman protocol vulnerability problem [10], and manage the inter-domain communications [09]

4. SDP, SAP , and SIP protocols

A certain number of protocols specific to visioconference on Internet were developed. In this section, we present three main protocols among of them:

SDP protocol (Session Description Protocol) [13] describes textually the format and the contents of multi-media sessions, provides an informal specification of multimedia sessions information, and gives also a description of the data exchanged through multimedia

sessions. SDP defined a multi-media conference as a set of two or several users communicating between them via certain software while a multi-media session is defined as a set of transmitters, receivers and multimedia data exchanged between them.

The multimedia session is described on two levels. The first one is defined as session level while the second is media level. The SDP description of a multi-media session is based on the following parameters: SDP protocol version, session owner, session title, additional information on the session, URL to get information on the session, electronic addresses of the session responsible, used connection, session start and end moments, and the exchanged media via the session.

SAP protocol (Session Announcement Protocol) [11] rests on the session SDP description. It allows also to define the annoncement rules of a multimedia session on Internet by diffusing, in a periodically way, a SAP packet to well defined addresses and ports. A SAP packet is composed of a header and a Payload representing the session SDP description.

SIP Protocol [23] is an application level protocol that allows to create, modify and terminate multimedia sessions. The basic services of SIP protocol are SIP addressing, SIP server localisation, SIP invitation, users localisation, and registration service.

5. Our approach

5.1 Principle of our approach

Our approach rests on a global architecture of a teleconference system (fig.4) composed of a sub-system of coordination that coordinates intra and inter-session interactions and a sub-system of visioconference that deals with the media transmission and diffusion.

On the level of the coordination sub-system, participants are organized in groups that correspond to sessions. Each session has a chairman while each participant is defined by a process that enables him to interact with sub-system of teleconference. On the level of the sub system of teleconference, to each process of the coordination sub-system is corresponded a videoconference process as well as a connection. The visioconference processes and connections are grouped in the form of sessions on the teleconference sub system level.

Our objective is, as we already mentioned, the specification of the management policies of the intra-sessions coordination using Ponder language. Thus, in the following sub-section we give a presentation of our platform of the intra-multimedia sessions coordination management.

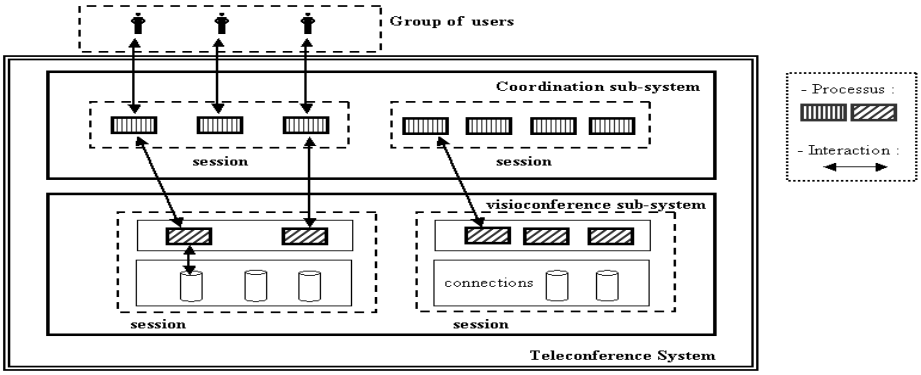

Figure 4: Global architecture of our teleconference system

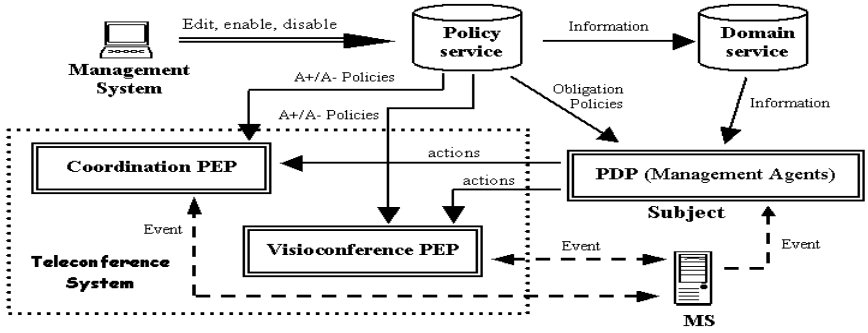

Figure 5: Proposed platform to policy-based manage the intra-session coordination

```
inst oblig  AA_Reg_Session {
   on        EvtRegistry(user[],hote[],session_id[]) ;
   Subject   Resources/soft/Agents/AAAgt ;
   Target    t = Personal/ChairMan ;
   do        AA[] = Extract.AAinf(user[]) -> RespAA[] = auth_author(AA[]) -> ChrmRef[] = ExtractChmRef(Session_id)
             -> RespChrm[] = t.ChrmAgr(ChrmRef[],RespAA[],user[],hote[]) -> session[] = ExtractSess(session-id[],ChrmRef[])
             -> MS.TriggerEvts.RegistryUserEvt(RespChrm[],user[],hote[],session[]) -> RegJournal(); }
```

Figure 6: Ponder specification of the registration of a user in a multimedia session.

```
inst oblig  Direct_Reg_User_Session {
   on        RegistryUserEvt(RespChrm[],user[],hote[],session[]);
   Subject   Resources/soft/Agents/UserRegAgt ;
   Target    t = Resources/Material/DB/SessionsDB ;
   do        decision[] = RegistryDec(RespChrm[]) -> SIPAdres[] = CalculateAdres(user[],hote[])
             -> SuppInfo[] = ExtractSupplInfo(user[],hote[],session[]) -> t.RegistryUser(SIPAdres[], SuppInfo)
             -> PEPTelc.Registry(SuppInfo[]) -> RegJournal(); }

inst oblig  Start_InitInvit_Session {
   on        EvtInitInvitSession(ChrmRef[],session_id[]) ;
   Subject   Resources/soft/Agents/SessCtrlAgt ;
   Target    t = Resources/soft/Agents/AAAgent ;
   do        AAResp[] = t.identification(ChrmRef,session_id[]) -> MS.TriggerEvts.TriggSession(AAResp[],ChrmRef[],session_id[])
             -> RegJournal(); }

inst oblig  Init_Invit_Session {
   on        TriggSession(AAResp[],ChrmRef[],session_id[]);
   Subject   Resources/soft/Agents/SessInitInvitAgt ;
   Target    t = Resources/Material/DB/SessDB ;
   do        dec[] = decision(AAResp[]) -> InitActivSession(ChrmRef,session_id[])
             -> UsersSIPAdres[] = t.ExtractSIPAdrs(ChrmRef,session_id[])
             -> invitation(UsersSIPAdres[]) -> PEPTelc.initDif(ChrmRef,session_id[]) -> RegJournal(); }

inst oblig  Invit_Users {
   on        EvtInvitUser(ChrmRef[],session_id[],user_info[]);
   Subject   Resources/soft/Agents/UserInvitAgt ;
   Target    t = Resources/Material/DB/SessDB ;
   do        AAResp[] = identification(ChrmRef[],session_id[]) -> ChrmName[] = t.Chrmname(ChrmRef[],session_id[])
             -> SIPAdr[] = t. ExtractSIPAdr(user_info) -> invitUser(SIPAdr[],ChrmName[],session_id[])
             -> PEPTelc.Diff(SIPAdr[],ChrmRef,session_id[]) -> RegJournal(); }
```

Figure 7: Ponder specifications of the basic multimedia session management activities.

5.2 Policy-based management Platform

Our platform of the intra-multimedia sessions coordination management with the possible interactions that could take place between its components are illustrated on figure5. The proposed policy-based management platform is constituted of a certain number of agents. These latter permit to take management decisions on the PDP level, set configuration tasks and control access on the on the PEPs level. Moreover, the proposed platform is characterized by the integration of a MS to automate the management. Concerning both the domain service and the policy service, they contain the references of the domains and those of management policies respectively.

5.3. Functioning of our proposed management platform

The basic multimedia session management activities discussed in this section are user registration, multimedia session initiation and activation, user invitation, speech queue management, and multimedia session closing. These activities respect the descriptions of the SDP, SAP and SIP protocols. Concerning our multimedia environment users, each user has to possess an identifier and a password to identify inside the environment.

To be registered in a multimedia session, the multimedia environment users must follow the following procedure:

Firstly, the user sends a registration message with his parameters (user own parameters, host parameters, and the desired multimedia session identifier) to the MS. With the reception of this message, the MS triggers the adequate obligation management policies to perform the necessary management actions. The corresponding Ponder specification is presented on figure 6. Thus, with the reception of the event EvtRegistry() (fig.6), the subject AAAgt (Authentication Authorization Agent) extracts, through the method ExtractAAinf(), the necessary information to identify the user. The user identification is realised through the authentication and

authorization method auth_author(). Then, the subject invokes the methods ExtractChmRef() and ChrmAgr() to get the session chairman reference and agreement respectively. Finally, the necessary information on the considered session are extracted through the method ExtractSess() and the policy managing the users direct registration in a multimedia session is triggered via a MS event. This operation is recorded in the management journal through the method RegJournal().

The Ponder specification of the direct registration of a user in a multimedia session is presented on figure 7 (obligation policy ***Direct_Reg_User_Session***). With the reception of the event RegistryUserEvt(), the subject UserRegAgt (User Registration Agent) decides on the user registration, through the method RegistryDec(), basing on the Chairman response. Then, the subject calculates the user SIP address and extracts the necessary registration information through the methods CalculateAdres() and ExtractSupplInfo() respectively. Finally, the methods RegistryUser() and Registry() are invoked to registry the user in the desired session and ask the teleconference PEP (PEPTelc) to reserve a process for the considered user, respectively. This management operation is recorded in the management journal through the invocation of the method RegJournal().

The Ponder specification for the initiation and activation of multimedia sessions is also given on figure 7 (obligation policy ***Start_InitInvit_Session***): With the reception of the event EvtInitInvitSession(), the subject SessCtrlAgt (Session Control Agent) identifies firstly the considered chairman and then triggers, through the MS, the adequate obligation policy to start the multimedia session. Finally, this management operation is recorded in the management journal.

The Ponder specification of the multimedia session triggering is given on figure 7 (obligation policy ***Init_Invit_Session***): With the reception of the event TriggSession(), the subject SessInitInvitAgt (Session Initiation and Invitation Agent) will trigger the initiation of the considered multimedia session if the decision is positive. Afterwards, the subject extracts the SIP addresses of the considered session users invoking the method ExtractSIPAdrs() and invites them, through the method invitation(), to initiate the multimedia session. Finally, the subject asks the teleconference PEP for beginning the diffusion and records this management operation in the management journal.

Chairmen could invite users to join a multimedia session. The Ponder specification is presented on figure 7 (policy ***Invit_Users***): With the reception of the event EvtInvitUser(), the subject UserInvitAgt (Users Invitation Agent) identifies the chairman and extracts his name through the methods identification() and Chrmname() respectively. Then, The subject extracts the user SIP address and invites him to join the session invoking the methods ExtractSIPAdr() and invitUser(). Finally, the subject asks the teleconference PEP for reserving a teleconference process for the considered user and records this management intervention in the management journal.

The speech organisation is an important task of the multimedia session management. Our proposed Ponder specification of the management of a multimedia speech queue is given on figure 8 (obligation policy ***Token_Mgmt***): With the reception of the event AskToken(), the subject TokenMgmtAgt (Token Management Agent) extracts the user priority and demands the session chairman agreement through the methods ExractPriority() and ChrmAgreemt(), respectively. Then, the subject invokes the method WaitingFunction() to manage the speech queue according to the chairman agreement. Finally, this management operation is recorded in the management journa

```
inst oblig Token_Mgmt {
  on        AskToken(user_id[],UserSIPAdr[],session_id[]);
  Subject   Resources/soft/Agents/TokenMgmtAgt ;
  Target    t = Resources/Material/DB/SessDB ;
  do        priority[] = t.ExractPriority(user_id[],session_id[]) -> RespChrm[] = ChrmAgreemt(priority[], user_id[],session_id[])
            -> WaitingFunction(RespChrm[],UserSIPAdr[],session_id[])) -> RegJournal(); }
```

```
inst oblig SessionClose {
  on        EvtCloseSession(ChrmRef[],session_id[]);
  Subject   Resources/soft/Agents/CloseSessAgt ;
  Target    t = Resources/Material/DB/SessDB ;
  do        AAResp[] = identification(ChrmRef[],session_id[]) -> UsersSIPAdrs[] = t.ExtractSIPAdr(ChrmRef[],session_id[])
            -> InforUsers(AAResp[],UsersSIPAdrs[]) -> CloseSession(ChrmRef, session_id[])
            -> PEPTelc.StopDiff(ChrmRef[],session_id[]) -> RegJournal(); }
```

Figure 8: Procedures of both speech organization and multimedia session close.

.

Finally, the procedure to close a multimedia session is specified on figure 8 (obligation policy ***SessionClose***): With the reception of the event EvtCloseSession(), the subject CloseSessAgt (Session Closing Agent) identifies the chairman, extracts the SIP addresses of the considered session users and sends them an information message. Then, the subject triggers the session closing and asks the teleconference PEP for stopping the diffusion. Finally, this management operation is recorded in the management journal.

6. Related works

Multimedia domain was the subject of various works. On one hand, Harroud et al. proposed an approach for the policy-based management for multimedia collaborative services [12]. This approach is characterised by the use of the concept of Virtual Team (V-Team) that has emerged as an important new teamwork model. V-Team consists of a dynamic collection of individuals, a set of collaborative services and network facilities that ensure a flexible and secure coordinated resource sharing. In this approach, authors describe the main components of V-Team, the principles of context customisation and the system monitoring approach via policies. On other hand, Plagemann [22] presented an important discussion concerning multimedia and middleware while basing on four research works. These four works describe: novel solutions for integrated runtime support in QoS-aware middleware; dynamic end-to-end QoS management middleware; proxy architectures for collaborative media streaming; and abstractions for multimedia streaming.

Concerning our work, it is characterized by the use of the policy based management concept to manage the coordination inside multimedia sessions, respecting some protocols descriptions (SDP, SAP, and SIP). Preciously, we used Ponder to specify the various policies management. Our approach is very simple and returns the management actions sequences very clear and easy to understand. Consequently, the responsibilities of the multimedia environment management are assigned to autonomous agents. This assignation well automates the management and minimizes the sessions chairman and administrators efforts.

7. Conclusion

Visioconference systems belong to distributed multimedia systems that allow persons to see each other, communicate and work together on shared resources through sessions independently on their physical places. Many works were devoted, by the IETF, to the standardization of description, annoncement and initiation of multimedia sessions on Internet. However, they did not consider the problem of the intra and inter-sessions coordination. This paper is a contribution in the efforts devoted to this area. Our approach was based on the policy-based management that became currently the tendency of all type of management because the management traditional methods resting on both the physical presence and displacement of administrators are very exceeded. Our approach employed also the policies specification language Ponder that appears currently to be the adequate tool for specifying the distributed systems management policies. In our Ponder specifications, we respected the various descriptions and standards suggested by the protocol dedicated to the multimedia sessions; SDP, SAP and SIP. Our perspective will be the implementation of this approach with the integration of the quality of service management.

8. References

[1] Baeten J.C.M., Over 30 years of process algebra: Past, present and future. In L. Aceto, Z. Ésik, W.J. Fokkink, and A. Ingolfsdottir, editors, Process Algebra: Open Problems and Future Directions, volume NS-03-3 of BRICS Notes Series, pp. 7–12, 2003.

[2] Beauquier J., Bérard B. , Système d'exploitation. McGraw-Hill, 1991.

[3] Bellosta M.J., Bessede A., Darrieumerlou C., Gruber O., Pucheral P., Steffen H., Thévenin J.M., GEODE: Concepts and Facilities. 6èmes journées Bases de Données Avancées (BDA), Monpellier, Septembre 1990.

[4] Bochmann G. V., and e. al., System specification with MONDEL and relation early with other formalisms. Progress Report No. 13 for CRIM/BNR project, June 1990.

[5] Damianou N., Dulay N., Lupu E.C. Sloman M.S., Tools for Domain-Based Management of Distributed Systems. IEEE/IFIP Network operations and management symposium (NOMS2002), Florence,Italy, pp.213-218, April 15-19, 2002.

[6] Damianou N., Dulay N., Lupu E.C. Sloman M.S., The Ponder Policy Specification language. Proc. Policy 2001, International Workshop on Policies for Distributed Systems and Networks, Bristol, United Kingdom, January 29-31, 2001.

[7] El Hamzaoui M., Sekkaki A., Bensassi B., Infrastructure to manage the IPSec Security Policies. In Proc. GRES'2006 - Gestion de REseau

et de Service, 7ème Colloque Francophone, Bordeaux-France, pp.315-328, May 9-12, 2006.

[8] El Hamzaoui M., Sekkaki A., Bensassi B., The Policy-Based Management of a Virtual Laboratory Security. In Proc. GRES'2005 - Gestion de REseau et de Service, 6ème Colloque Francophone, Luchon-France, pp.183-198, 28 February-02 March 2005.

[9] El Hamzaoui M., Sekkaki A., Bensassi B., Policy-Based Management of the inter-Domain communications Security. In Proc. IEEE/IFIP 4th Latin American Network Operations and Management Symposium (LANOMS), Porto Alegre, Brazil, pp. 269-274, 2005.

[10] El Hamzaoui M., Sekkaki A., Bensassi B., Policy-based Resolution of the Diffie-Hellman protocol vulnerability. In Proc. IEEE/IFIP 4th Latin American Network Operations and Management Symposium (LANOMS), Porto Alegre, Brazil, pp. 277-282, 2005.

[11] M. Handley, C. Perkins, and E. Whelan., Session Announcement Protocol. RFC 2974, October 2000.

[12] Harroud H., Lakhdissi M., Karmouch A., Grossner C., Policy-Based Management for Multimedia Collaborative Services. In Proc. 4th IFIP/IEEE International Conference on Management of Multimedia Networks and Services, MNS 2001, Chicago, IL, USA, October 29 - November 1, 2001.

[13] Handley M., Jacobson V., SDP: Session Description Protocol. RFC 2327, April 1998.

[14] Gerard J. Holzmann., The model checker SPIN. IEEE Transactions on Software Engineering, 23(5):279–295, May 1997.

[15] Girault C., Valk R., Petri Nets for Systems Engineering. Springer Verlag, 2002.

[16] ITU-T Recommendation Z.100(11/99), Specification and description language (SDL). Telecommunication Standardization Sector of ITU, Printed in Switzerland, Geneva, 2000.

[17] Jard C., Monin JF., Groz R., Development of VEDA- a Prototyping Tool for Distributed Algorithms. IEEE trans. on Software Engineering, Vol.14, N. 3, Mars 1988.

[18] Kenneth J. Turner., Using Formal description techniques: An introduction to ESTELLE, LOTOS, and SDL. John Wiley, 1993.

[19] Lamport L.A., The Temporal Logic of Actions. Research Report 79, SRC, April 30, 1994.

[20] Lymberopoulos L., Lupu E.C., Sloman M., An Adaptive Policy-Based Framework for Network Services Management. Journal of Network and Systems Management, Vol.11, No.3, September 2003.

[21] Lymberopoulos L., Lupu E.C., Sloman M., ONDER Policy Implementation and Validation in a CIM and Differentiated Services Framework. 9th IEEE/IFIP Network Operations and Management Symposium (NOMS 2004), Seoul, Korea, May 2004.

[22] Plagemann T., Introduction to Multimedia special issue on multimedia middleware: Middleware + multimedia = multimedia middleware?. Multimedia Systems 8: 395–396 (2002), Digital Object Identifier (DOI) 10.1007/s00530-002-0060-5. Multimedia Systems, Springer-Verlag, 2002.

[23] Rosenberg J., Schulzrinne H., Camarillo G., Johnston A., Peterson J., Sparks R., Handley M., Schooler E., SIP: Session Initiation Protocol. RFC 3261, June 2002.

[24] Sekkaki A., El Hamzaoui M., Bensassi B., Policy-based Management of a Virtual Laboratory Communications Security. In Proc. The First IEEE International Workshop on Broadband Convergence Networks (BCN2006), Vancouver, Canada, pp.199-204, April 2006.

[25] Sloman M., Twidle K., Domains: A Framework for Structuring Management Policy. In Network and distributed systems management, ed. Morris Sloman, Addison-Wesley, pp.433-453, 1994.

[26] Wayne McCoy., Implementation Guide For The ISO Transport Protocol. RFC 1008, June 1987.

[27] Yavatkar R., Pendarakis D., Guerin R., A Framework for Policy-based Admission Control, RFC 2753, January 2000.

[28] Imperial College : http://www.doc.ic.ac.uk/, last update : 2003.