

MODELE STANDARDIZATE DE GESTIUNE A REȚELELOR

dr. ing. Liliana Dobrică

dr. ing. Daniela Saru

Universitatea Politehnica, București

Rezumat: Scopul acestui articol este de a analiza conceptele de bază pentru modelarea standardizată a aplicațiilor de gestiune a rețelilor de interconectare deschise OSI. În general, pentru sistemele de gestiune, se dezvoltă standarde care structurează și modelează informațiile astfel încât să poată fi aplicate pentru diverse rețele de interconectare. Datorită nivelului mare de generalitate, aceste standarde sunt foarte complexe. În prima parte, se analizează structura informației de gestiune, apoi se prezintă arhitectura unui sistem pentru gestiunea sistemelor deschise. Se analizează, de asemenea, serviciile suport pentru manipularea evenimentelor și logging și se menționează funcția de control al accesului.

Cuvinte cheie: sisteme de gestiune OSI, obiect gestionat, manager, agent, servicii de gestiune OSI

1. Introducere

Standardele OSI s-au dezvoltat în scopul gestionării comunicațiilor din sistemele deschise. Ele se ocupă cu:

- stabilirea regulilor de bază pentru transmiterea eficientă a șirurilor de biți pentru echipamente;
- asignează semnificație sintactică și semantică șirurilor de biți în funcție de aplicație.

Datorită diversității organizațiilor care asigură și utilizează servicii de prelucrare și comunicație de date, al domeniului larg al aplicațiilor de prelucrare a informațiilor pe care-l necesită și a complexității echipamentului folosit, gestiunea mediului de comunicație OSI este supusă standardizării.

Modelul de referință OSI (ISO 7498-1) a fost dezvoltat în anii 1980. Scopurile de bază au fost:

- cercetarea și stabilirea cerințelor de gestiune;
- deducerea unui set general de funcții de gestiune și servicii de comunicație;
- cercetarea de noi caracteristici prin care să se specifice acele funcții și servicii.

Standardele de gestiune au stabilit o imagine abstractă a resurselor prin operațiile de gestiune, care pot fi efectuate pe ele și prin comunicațiile necesare, când acele resurse sunt gestionate de la distanță. Aceasta se identifică în:

- interfețe;
- natura interacțiunilor testabile, care pot avea loc prin intermediul acelor interfețe;

- comportamentul presupus al resurselor ce sunt accesate prin interfețe.

În urma cerințelor de standardizare, a rezultat un model de referință de bază. Relația dintre acesta și modul în care este folosit să comunice date despre gestiunea resurselor OSI, este stabilită în [2]. Conform acestui standard, când nu se poate garanta toată stiva OSI, există **gestiune de nivel** dacă schimburile de date între entitățile de pe nivelurile de sub cel al aplicației pot fi folosite în cazuri speciale de inițializare a sistemului sau de reparare a defectelor. Se definesc **sistemele de gestiune** ca reprezentând sisteme bazate pe schimburile de date între entitățile de gestiune realizate la nivelul aplicației.

Se specifică, de asemenea, o tehnică de modelare a resurselor de comunicație sub formă de date aflate în sistemul deschis, care asigură o interfață locală cu acestea. Datele reprezintă abstractizarea, din punct de vedere al gestiunii, a comportamentului și stării resurselor. Se definește baza informației de gestiune (termenul în engleză este MIB-Management Information Base) ca fiind acea informație din sistemul deschis care poate fi transferată sau modificată prin protocoale de gestiune OSI. Datele pot reprezenta fie resursele și activitățile lor specifice unuia din cele șapte niveluri OSI, fie alte date ce sunt relevante pentru activitățile de comunicație ale sistemului.

2. Zonele funcționale ale cadrului de gestiune OSI

Cadrul de gestiune OSI este împărțit în cinci zone funcționale și anume:

- i. **gestiunea defectelor**, care identifică existența unui defect, înlătură cauza și efectuează teste de credibilitate pentru verificarea înlăturării defectului;
- ii. **gestiune de configurare**, pentru a monitoriza starea resurselor și a relațiilor dintre ele (de exemplu, pentru a descrie topologia rețelei); pentru a schimba starea și relațiile între resurse fie prin simple comenzi sau prin încărcarea altor versiuni de software de comunicație sau tabele de configurare;

- iii. *gestiunea conturilor* pentru adunarea datelor despre utilizarea unei resurse și asignarea acelei utilizări unui subscritor a cărui capacitate de folosire trebuie contabilizată. Controlul este exercitat prin strângerea acestor date și prin împărțirea și alocarea resursei utilizate subscritorilor.
- iv. *gestiunea performanței* pentru adunarea datelor despre utilizarea unei resurse și combinarea lor pentru realizarea statisticilor de performanță;
- v. *gestiunea securității* care se ocupă cu generarea rapoartelor despre încercările de violare a securității sistemului. Sunt două aspecte importante: (i) gestionarea mediului de securitate a unei rețele incluzând detecția violărilor de securitate și păstrarea informației raportate; (ii) efectuarea procesului de gestiune a rețelei într-un mod securizat.

Din analiza structurării în zone funcționale a sistemelor de gestiune OSI în [13] se ridică următoarele probleme:

- i. zonele funcționale identifică cerințele funcționale ale sistemelor de gestiune, dar ele nu reprezintă o bază de standardizare, pentru că lista funcțiilor ce pot apărea într-o zonă este deschisă. Un standard care încearcă să acopere o zonă nu va fi niciodată complet, fiind într-o continuă actualizare;
- ii. zonele funcționale pot avea entități și operații efectuate asupra lor comune. De exemplu, sistemele de gestiune a defectelor și sistemele de gestiune a securității răspund la și controlează fluxul datelor de alarmă, care este același. Fluxul de date se obține cu aceeași funcție, dar se supune unor prelucrări diferite. Astfel, este mai bine ca entitățile, operațiile și comportamentul lor să aibă definiții generale, decât să li se asigneze funcții specifice.
- iii. gestiunea este o activitate de prelucrare distribuită, care are nevoie ea însăși de gestiune, pentru ca operarea rețelei să fie eficientă. De exemplu, un defect poate genera mai multe semnale de alarmă dacă diferitele componente ale rețelei sunt influențate de acesta. Sistemele de gestiune trebuie să identifice care dintre alarmele primite este semnificativă și să-și controleze fluxul datelor de alarmă, când rapoartele defectelor excesive, ca număr, congestionează rețeaua. Deoarece aceste

date pot fi distribuite, funcțiile trebuie restricționate astfel încât să se păstreze consistența datelor.

Împărțirea din punct de vedere funcțional a gestiunii sistemelor permite analiza funcțiilor fiecărei zone conform specificației standardelor [1]. Există astfel definite funcțiile din zona gestiunii configurării [9], [10], [11], funcțiile de raportare a alarmelor [12], funcțiile pentru gestiunea defectelor pentru testare și alte funcții standardizate sau în curs de standardizare.

3. Arhitectura sistemelor de gestiune OSI

Din analiza dezvoltării standardelor OSI rezultă trei consecințe importante. Prima este necesitatea specificării unui model informatic cu date prin care să se abstractizeze, cu un punct de vedere comun, resursele de comunicație. Realizarea în acest scop, o reprezintă faptul că interfața cu resursele poate fi descrisă după tehnica modelării orientate obiect, având caracteristicile unui obiect gestionat. O altă consecință o reprezintă modul în care standardele rezultate sunt documentate și ce relație există între ele. Aceasta a condus la dezvoltarea unui număr de standarde funcționale specifice și a unei mulțimi de standarde ce prezintă funcțiile de gestiune. *Modelul de referință OSI* reprezintă punctul de referință pentru stabilirea cadrului de gestiune OSI și a protocoalelor pentru comunicarea datelor între sistemele de gestiune. Totuși, pentru înțelegerea celorlalte standarde despre sistemele de gestiune, baza este prezentată în [3]. Standardele se împart în două categorii, unele ce se ocupă de funcționalitate și altele de reprezentarea informației. A treia consecință este dezvoltarea unei arhitecturi pentru sistemele de gestiune. Deși este parțial definită, după cum se susține în [3], arhitectura din [2] ajută la înțelegerea standardelor publicate și a modului lor de aplicare.

Principiile modelării resurselor pentru gestiune

Puterea, generalitatea și eficiența gestiunii sistemelor OSI sunt mari datorită utilizării unui model informatic de gestiune, comun pentru specificarea gestionării resurselor de orice tip [6]. Modelul se bazează pe conceptul de obiect gestionat și adoptă principiul de moștenire pentru a permite reutilizarea și extinderea într-un mod controlat a specificației.

Gestiunea sistemelor OSI intenționează să fie suficient de generală pentru a gestiona un număr

mare de resurse. Deși aceste resurse diferă mult între ele, este foarte important să poată fi gestionate folosind serviciul de comunicații CMIS [4] standard suportat de protocolul CMIP [5] și de avea în comun un număr de concepte fundamentale.

Se cere astfel ca:

- definițiile despre cum se gestionează resursele să fie compatibile cu CMIS;
- trebuie să fie o structură de nume globale, comună, astfel încât gestiunea diferitelor resurse să poată fi combinată într-un sistem și să fie referită fără ambiguitate;
- tipuri similare de informație să fie definite în moduri similare;
- trebuie să fie un mod standard de extindere a definițiilor despre cum se gestionează resursele și despre "împrumutul" părților de specificație din altă parte pentru reutilizarea lor;

Modelul abordează aceste cerințe prin definirea conceptului de **obiect gestionat**, care are diferite proprietăți, ce încorporează cerințele enunțate mai sus. Obiectele gestionate sunt definite ca fiind "imaginea de gestiune" a resurselor pe care le reprezintă. Această definiție nu trebuie implementată pasiv deoarece "imaginea de gestiune" a resursei nu este chiar o consecință naturală a implementării ei și, de aceea, este nevoie de o atentă analiză. Partea importantă din specificarea pentru gestiune este de a decide ce aspecte ale resursei pot fi monitorizate și controlate de către manager. Astfel, un obiect gestionat nu este un alt cuvânt pentru o resursă gestionată, ci definește posibilitatea de gestiune a resursei.

Obiectul gestionat este prezentat ca o sferă opacă, ce învăluie resursa reală, cu o fereastră prin care se poate pasa informația de gestiune. Informația ce traversează suprafața constă din operații de gestiune pe obiectul gestionat, din rezultatele operațiilor care se întorc și din notificările ce pot fi generate spontan de către obiectul gestionat când se întâmplă ceva semnificativ cu sau în resursă. Astfel, resursele pot fi cunoscute sau controlate numai prin operațiile, rezultatele și notificările ce traversează suprafața obiectului gestionat.

În general, există două abordări distincte ale comportării unei resurse. Una va specifica "comportamentul operațional" al resursei, adică ce face independent de orice gestiune; aceasta poate defini un protocol de tipul protocolului de transport OSI, ori poate stabili cum se comportă un modem sau poate defini operarea unui releu. Cealaltă specificație, definiția obiectului gestionat, va stabili cum poate fi gestionată resursa și cum se leagă

operațiile de gestiune de comportarea operațională. Definiția obiectului gestionat se scrie în conformitate cu cerințele modelului informatic de gestiune.

În [13] se susține că specificarea completă a unei arhitecturi ar trebui:

- să arate toate aspectele de gestiune, care afectează interconectarea sistemelor deschise, iar mediul de comunicare să poată fi realizat pe baza standardelor;
- să acopere activitatea de prelucrare distribuită a gestiunii sistemelor, arătând cum pot fi coordonate interacțiunile de gestiune între sisteme și cum pot fi combinate datele din mai multe surse;
- să definească modul de comparare a datelor obținute din monitorizarea sistemelor deschise cu cerințele politicii;
- să arate cum, nepotrivirile dintre comportamentul așteptat și cel observat, generează acțiuni de control pentru îndeplinirea cerințelor;
- arhitectura să fie independentă de dimensiunea rețelei și de tehnologia de calcul și comunicație.

Arhitectura prezentată în standard [3] este limitată la comunicația dintre o pereche de entități aplicație de gestiune a sistemelor. De asemenea, se abordează un tip particular de interacțiune de gestiune, în care procesul manager interacționează cu un sistem la distanță pentru a-și exercita controlul asupra resurselor de la distanță. În figura 1, un proces de gestiune dintr-un sistem deschis are rol de **manager**. Entitatea sa pereche din celalalt sistem are rol de **agent** pentru a asigura accesul la reprezentarea de obiect gestionat a resurselor gestionate. Primul sistem este uneori referit ca "sistem care gestionează", în timp ce al doilea este "sistem gestionat". Împreună asigură serviciul de gestiune. Asignarea rolurilor sistemelor nu este permanentă. Distincția între cele două roluri se face în funcție de natura activității de comunicație. Pentru a o realiza, este necesară anticiparea caracteristicilor obiectelor gestionate.

Obiectele gestionate asigură o interfață prin care (i) operațiile pot fi efectuate și (ii) evenimentele ce apar asincron într-un obiect gestionat pot fi emise sub forma de notificări.

Entitatea cu rol de agent este intermediară între comunicarea operațiilor către obiecte și comunicarea notificărilor entității manager. Entitatea manager răspunde de una sau mai multe activități de gestiune, generând operațiile de gestiune

și recepționând notificații. Astfel, canalul de comunicație dintre manager și agent suportă două tipuri de servicii de transfer al datelor de gestiune, **operații de gestiune** (cerute de manager agentului) și **notificații** (trimise de agent managerului). Astfel, rolul luat de fiecare entitate aplicație este dependent total de tipul și de direcția de transfer a datelor. Operațiile și notificațiile sunt comunicate folosind protocoalele de gestiune OSI [5].

sistem, pentru sincronizarea operațiilor și pentru controlul accesului la obiectele gestionate. De asemenea, asigură mecanisme ce pot filtra selectiv operațiile de efectuat sau pot controla fluxul de date de la notificații. Dacă aceste caracteristici asigurate de către agent pot fi ele însele gestionate, aceste posibilități de gestiune se realizează tot prin operații asupra obiectelor gestionate. Prin aceste servicii, standardele de gestiune OSI se caracterizează printr-o funcționalitate cu proprie gestiune (self-management).

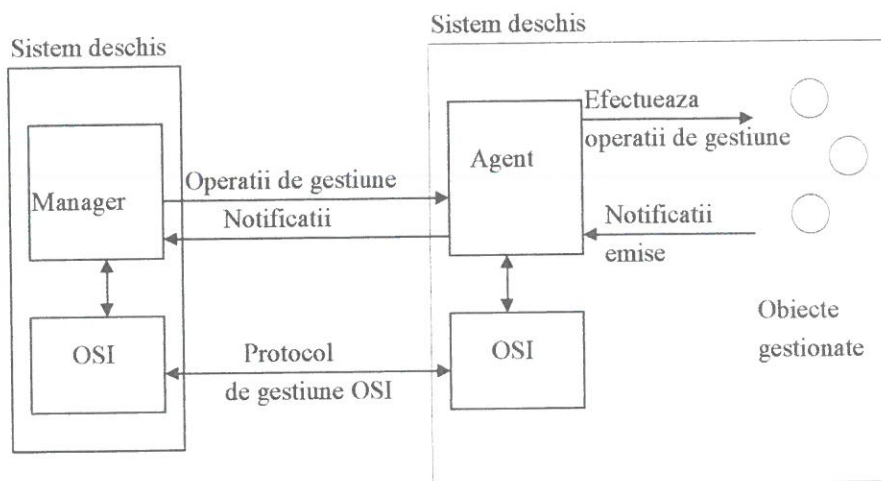


Figura 1. Arhitectura unui sistem de gestiune

De asemenea, [3] specifică schimburile de date ce pot avea loc când două entități aplicație formează o **asociație** pentru efectuarea funcțiilor de gestiune. Primul schimb care stabilește **contextul de gestiune** identifică dacă o entitate va opera numai ca manager, agent sau ambele pe durata asocierii. Celălalt, care identifică **unitățile funcționale de gestiune** a sistemelor, este folosit pentru stabilirea funcționalității asociate schimburilor de date ulterioare. Unitățile funcționale servesc pentru indicarea posibilităților funcționale asociate și pentru limitarea domeniului de schimburi la cele acordate între entitățile ce comunică.

4. Servicii suport pentru gestiune

Agentul asigură, pe de o parte, serviciile pentru suportul gestiunii și, pe de altă parte, maparea între interfața obiectului gestionat și interfața de comunicație. Astfel, el permite localizarea obiectelor gestionate accesate de către manager prin intermediul comenzilor specifice infrastructurii de comunicație și asigură și detecția evenimentelor și avansarea notificațiilor către managerii înregistrați să le primească.

În particular, el dispune de mecanisme pentru adresarea grupurilor de obiecte gestionate din

Astfel de funcții specificate în standarde asigură gestiunea funcțiilor de **raportare a evenimentelor**, de **log control**, de **control al accesului** și de **planificare**. Figura 2, ilustrează imaginea de gestiune în cazul agentului. Se prezintă obiectele gestionate ce răspund la o **operație** prin emiterea unei **notificații**, ambele fiind manipulate de către agent. Diagrama arată, de asemenea, obiectele suport, gestionate care-și exercită controlul local asupra funcționării agentului. Planificarea depinde de timpul determinat de la un ceas sistem local. Fluxul de date de la agent la log-ul local este accesat de către obiectul funcției log control, astfel încât, acesta din urmă să-și poată efectua activitatea de suport al gestiunii.

În aplicațiile de gestiune, cel mai frecvent utilizate dintre funcțiile standardizate sunt cele care îndeplinesc:

- serviciul de raportare a evenimentelor;
- serviciul de logging.

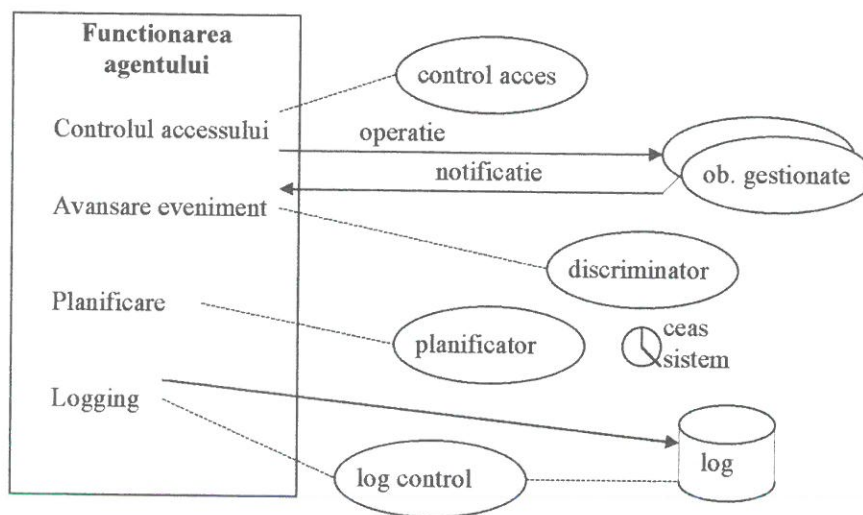


Figura 2. Funcționarea agentului

Serviciul de raportare a evenimentelor

Acesta este subliniat în [3], aspectele sale funcționale fiind specificate în **funcția de raportare a evenimentelor**. Scopul acestei funcții este de a asigura un mecanism prin care să se controleze avansarea notificărilor de la obiectele gestionate. Modelul de gestiune a evenimentelor este unul în care toate notificările pot genera potențiale rapoarte de gestiune a evenimentelor. Toate notificările sosite în sistem sunt, din punct de vedere conceptual, subiectul controlului *discriminatorilor* pentru avansarea evenimentului (eng. *event forwarding discriminator* - EFD) al cărui task este de a selecta anumite notificări și de a le înainta destinațiilor corespunzătoare. O notificare care este avansată devine raport al evenimentului.

În figura 3, *notificările* sunt receptionate de către o componentă de *detectare și de prelucrare a evenimentului* care mai adaugă momentul evenimentului, clasa obiectului și obiectul generator la mesajul de notificare pentru a forma un *potențial raport* al evenimentului. Acestea sunt conceptual trimise tuturor EFD-urilor locale. Un

EFD are un constructor discriminator, care definește criteriile folosite să selecteze care din rapoartele potențiale sunt înaintate la adresa destinație, memorată în EFD. Un EFD trimite rapoarte unei singure destinații. EFD-ul este un obiect gestionat, care poate genera și el *notificări EFD*, ce sunt avansate de un alt EFD local.

EFD este o clasă de obiect gestionat definită în [7]. Ca obiect este folosit să asocieze managerii cu evenimentele de care sunt interesați și să avanseze evenimentele de la nodul gestionat la stația care gestionează. Două dintre atributele sale obligatorii (citire/scriere) sunt *constructorul discriminator* și *destinația*. Primul reprezintă un predicat care determină criteriile după care vor fi înaintate notificările, deoarece volumul și diversitatea evenimentelor ce au loc în rețea poate fi foarte mare, făcând dificilă separarea informației utile de zgomot. El acționează ca un filtru și parametri săi sunt combinații logice de constructori simpli, ce testează tipul și/sau prezența notificării, relația aritmetică (mai mare, mai mic, egal cu), relații pe mulțimi, toate pentru datele generate de notificare.

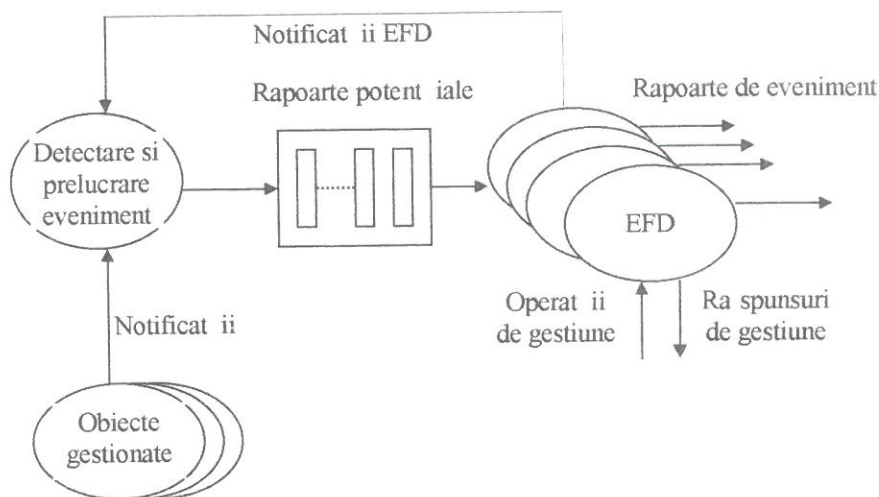


Figura 3. Modelul serviciului de raportare eveniment

Atributul destinație conține destinația aleasă pentru raportul de eveniment selectat fie sub forma de "titlu entitate aplicație", fie ca adresă de grup, dacă se alege una sau mai multe destinații. EFD are, de asemenea, un atribut obligatoriu de stare administrativă prin care operația pe obiectul gestionat poate fi suspendată sau reluată.

Suplimentar, EFD are anumite atribute opționale. Unul dintre acestea poate defini un set de *destinații alternative*, ordonate după priorități ce pot fi selectate ca destinații de backup, dacă nu e posibilă trimiterea unui raport de eveniment locației specificate în atributul destinație. O altă opțiune este setarea atributului de *mod*. Acesta determină dacă EFD așteaptă confirmarea de la destinația aleasă pentru a vedea că a primit raportul evenimentului. Deși nu este o procedură standard, această posibilitate de confirmare se poate utiliza în conjuncție cu atributul destinației backup și un parametru local de timeout (de asemenea, nespecificat în standard) pentru a determina dacă notificarea a ajuns la țintă în timpul predefinit. Dacă nu, notificarea poate fi regenerată destinației backup selectată. EFD suportă opțional raportare selectivă în timp. Această facilitate este asigurată prin planificarea timpului la care să se termine raportarea evenimentelor cu noțiuni temporale de tip zilnic sau săptămânal. O *planificare zilnică* are o listă a intervalelor de timp de perechi {start time, stop time} când se activează avansarea evenimentelor. *Planificarea săptămânală* o extinde pe aceasta prin definirea unei măști a săptămânii pentru a rezulta zilele din săptămână în care se aplică planificarea zilnică.

Deoarece EFD-urile sunt obiecte gestionate, ele pot emite notificări. Cele definite de standard sunt crearea sau ștergerea unui EFD sau modificarea unui atribut al EFD. Aceste notificări sunt trimise, ca și celelalte, tuturor EFD-urilor din sistem pentru a determina dacă și cărei destinații trebuie trimis raportul evenimentului.

Serviciul de logging

Deși standardele nu precizează destinația rapoartelor evenimentelor (RE), o destinație ce trebuie să existe este **log-ul**. Log-ul este un device ce poate memora selectiv RE în **înregistrări de log** (eng. log records). Standardul *Log Control Function* definește cum se realizează acest lucru. Datorită faptului că procesul de gestiune trebuie să fie capabil să controleze parametrii de bază ai activității de logging și pe cei pentru obținerea înregistrărilor din log, se definesc obiectele log (L) și log records (LR).

Log-ul se poate găsi fie în sistemul cu rol de manager, fie alături de obiectele gestionate ale căror notificări trebuie să le păstreze (log). Din acest motiv, L are multe caracteristici comune cu EFD. Modelul presupune că toate notificările de la obiectele gestionate sunt trimise la L și la EFD. Atributul *discriminator construct* selectează care date trebuie puse în LR. Ca și EFD, L are atributul de *stare administrativă*; operațiile pe acest atribut controlează suspendarea sau reluarea activității de logging. Opțional, se pot aplica și pentru L restricții de planificări temporale ca la EFD.

Specific log-ului este faptul că acesta are o capacitate limitată pentru înregistrări și se poate

întâmpla ca, după un timp, acesta să se umple. Standardul identifică două tipuri de log:

- acelea care resetează începutul când log-ul este plin și supraînscrie înregistrările mai recente;
- acelea care blochează procesul de logging-ul când se umplu.

Se pot genera notificații pe măsură ce logg-ul se umple, la anumite limite de prag definite, pentru ca managerul să poată acționa în consecință. Deoarece log-ul are o dimensiune finită, există atribute care dau dimensiunea (în octeți) a sa și cât de mult este umplut (octeți și număr de înregistrări).

LR sunt obiecte cu atribute read-only. Aceste atribute au un *identificator de înregistrare*, *timpul la care s-a introdus înregistrarea în log și data asociată raportului evenimentului ce se log-ează*.

Structura serviciului de log este prezentată în figura 4. *Procesul de colectare* primește rapoartele de eveniment de la obiectele gestionate locale sau la distanță și le formatează în *potențiale înregistrări de log* adăugându-le un identificator de înregistrare în log și momentul de timp de logging.

Mulțimea criteriilor pentru selecția potențialelor înregistrări ce pot fi memorate este specificată prin operații de comparare cu datele din rapoartele de eveniment, planificări zilnice sau săptămânale etc.

Dacă log atinge capacitatea maximă, există două opțiuni. Fie se oprește activitatea de log, iar noile înregistrări sunt ignorate, fie se pierde cele mai vechi înregistrări. Se poate genera un *eveniment limită de prag pentru capacitate* pentru a avertiza manager-ul înainte ca log-ul să fie complet plin. Fiecare log trebuie să suporte comportament de *halt* (nu se vor loga alte înregistrări, iar cele rămase vor fi păstrate), iar cel de *wrap* (vechile înregistrări din log, determinate după identificatorul de înregistrare vor fi șterse pentru eliberarea resurselor pentru crearea de noi înregistrări) este opțional.

Log-urile sunt ele însele obiecte gestionate și starea lor poate fi controlată de manageri pentru activarea/dezactivarea funcției de logging. De asemenea, planificările de timp, definite intern, pot modifica starea lor operațională. Atributele de log specifică logID, numărul maxim de înregistrări din log, dimensiunea curentă în octeți, numărul curent de înregistrări, acțiunea la umplere a log-ului, limita de prag pentru generarea evenimentelor de

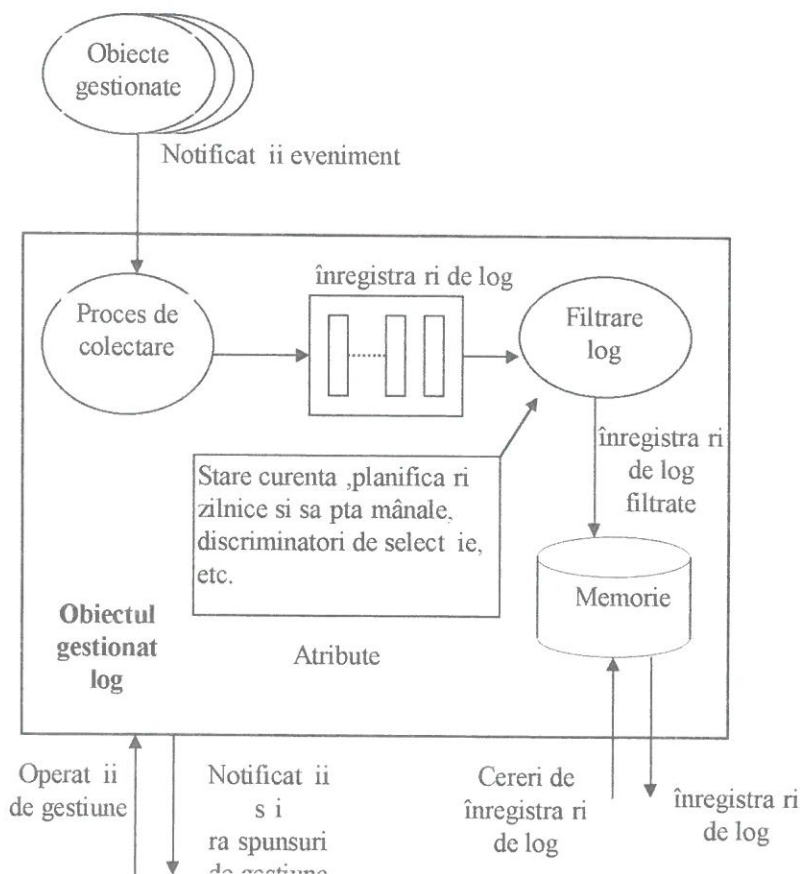


Figura 4. Serviciul de log

avertizare la umplere și atributele care controlează criteriile de selecție pentru înregistrări. Serviciul de logging este foarte flexibil pentru că managerii pot schimba criteriile de filtrare prin modificarea atributelor din log.

Fiecare operație asupra unui obiect gestionat, care este cerută de către manager, poate fi subiectul controlului accesului, care este o funcție a agentului, exercitată prin prelucrare. Definind această funcție ca obiect gestionat, agentul poate citi și modifica parametrii săi. În acest fel, agentul este informat de (setul de) obiecte și de operații cărora li se aplică un control la acces, precum și de entitățile manager ce pot efectua operații asupra obiectelor.

5. Concluzii

Arhitectura analizată reprezintă numai cadrul, definit ca fundament științific, pentru crearea sistemelor de gestiune interoperabile. Realizarea sistemelor cu astfel de funcții depinde de:

- selecția protocoalelor prin care se desfășoară schimbul de informații;
- selecția serviciilor pentru transferul informațiilor și a funcțiilor ce reprezintă problematica specifică a implementării;
- resursele ce trebuie gestionate;
- definițiile obiectelor gestionate, ce reprezintă resursele.

Este de reținut faptul că, la nivel internațional, există o tendință comună pentru globalizarea informațiilor. Pentru aceasta se încearcă structurarea într-o formă comună a reprezentării resurselor ce se gestionează. Pe lângă analiza și stabilirea resurselor ce se gestionează, procesul de dezvoltare a sistemelor din acest domeniu trebuie să definească structura și forma informațiilor ce se comunică în rețea. Transferul informațiilor are loc pe baza definirii funcțiilor interfețelor de comunicație. După definirea acestor funcții are loc realizarea la nivelul aplicației a sistemului format din entitățile manager și agent, care implementează aceste funcții.

Bibliografie

1. DOBRICĂ, L.: Analiza funcțională a modelului de gestiune OSI, Referat de Doctorat, aprilie 1997.
2. CCITT Recommendation X.700 (1993) | 7498-4, Management Framework for Open Systems Interconnection (OSI) for CCITT Applications.
3. CCITT Recommendation X.701 (1992) | ISO/IEC 10040:1992, Information technology –

Open Systems Interconnection – Systems management overview plus Technical Corrigendum 1 (1994) plus Technical Corrigendum 2 (1995).

4. CCITT Recommendation X.710 (1992) | ISO/IEC 9595:1993, Information technology – Open Systems Interconnection Common Management Information Service Definition.
5. CCITT Recommendation X.711 (1992) | ISO/IEC 9596-1:1993, Information technology – Open Systems Interconnection Common Management Information Protocol Specification for CCITT Applications.
6. CCITT Recommendation X.720 (1992) | ISO/IEC 10165-1:1992, Information technology – Open Systems Interconnection – Structure of management information: Management Information Model.
7. CCITT Recommendation X.721 (1992) | ISO/IEC 10165-2:1992, Information technology – Open Systems Interconnection – Structure of management information: Definition of management information, 1994.
8. CCITT Recommendation X.722 (1992) | ISO/IEC 10165-4:1992, Information technology – Open Systems Interconnection – Structure of management information: Guidelines for the definition of managed objects.
9. CCITT Recommendation X.730 (1992) | ISO/IEC 10164-1:1993, Information Technology - Open Systems Interconnection - Systems Management: Object Management Function.
10. CCITT Recommendation X.731 (1992) | ISO/IEC 10164-2:1993, Information Technology - Open Systems Interconnection - Systems Management: State Management Function
11. CCITT Recommendation X.732 (1992) | ISO/IEC 10164-3:1993, Information Technology - Open Systems Interconnection - Systems Management: Attributes for representing relationship
12. CCITT Recommendation X.733 (1992) | ISO/IEC 10164-4:1993, Information Technology - Open Systems Interconnection - Systems Management: Alarm Reporting Function.
13. ALWYN, L.: *OSI Management Model and Standards*. În: Network and Distributed Systems Management, Addison-Wesley Publishing Company, 1994, pp. 69-93