SECTIUNEA 1

RAPORTUL STIINTIFIC SI TEHNIC (RST)

FAZA DE EXECUTIE NR. VI

CU TITLUL "Implementarea MonALISA pe infrastructura de GRID locala"

- **RST – raport stiintific si tehnic in extenso***
- **PVAI – proces verbal de avizare interna**
- **PVRLP – procese verbale de receptie a lucrarilor de la parteneri****

* pentru modulul 3 se va utiliza modelul din Anexa 1
**forma si continutul se stabilesc de catre conducatorul proiectului, tinand seama de cele continute in PVAI

Indicatori de realizare a fazei (conform specificului fiecarui program/proiect)

Denumirea indicatorilor	**Numar**	
	Planificat	**Realizat**
• organizatii si respectiv numar de personal de cercetare implicate in proiect	3	3
○ tipuri de organizatii; INCD,U.P., SC, Univ.	INCD, 2 UNIV	INCD, 2UNIV
○ nr. cercetatori/ proiect/ module	33	33
•număr de studii studii privind starea şi perspectiva domeniului	1	1
• sisteme, structuri, procese, metode, mecanisme implementate/ aplicate (pe categorii)		
○ produse/ tehnologii/ servicii noi realizate		
○ produse/ tehnologii/ servicii modernizate		
° servicii noi realizate in cadrul programului, aliniate la standardele internationale		3
• produse program noi/modernizate/aliniate/ certificate/aplicate/valorificate	1	1
• agenti economici angrenati in parteneriate		
• platforme tehnologice integrate dezvoltate la nivelul programului		
• valoarea dotarilor noi pe program	40 925 lei	40 925 lei
• brevete de inventie propuse/ acceptate		
• articole/ carti publicate		
- *Carti tehnice*		
- *Cataloage*		
- *Dicţionare*		
- *Pliante*		
- *Postere*		
- *Standard European*		
- *Standard Internaţionale*		
- *Standard naţional*		
- *Documentaţii*		
- *Studii*		
- *Studii de piaţa*		
- *Studii de fezabilitate*		
- *Caiet de sarcini*		
- *Concepte*		
- *Metode*		
- *Ghiduri*		

- *Proceduri* - *Manual de utilizare* - *Rapoarte de verificare/testare* - *Proiecte/ Desene de execuţie modele, instalaţie pilot , prototip* - *Planuri de afaceri* • comunicari stiintifice • organisme ale infrastructurii de evaluare a conformitatii dezvoltate in cadrul programului: ○ laboratoare de incercari ○ laboratoare de etalonare ○ organisme de certificare • organisme de evaluare a conformitatii care isi desfasoara activitatea in domeniile reglementate prin directivele Uniunii Europene, din care: ○ produse industriale care intra sub incidenta marcajului CE; ○ produse agro- alimentare. ○ nr. de specialisti formati/instruiti pentru evaluarea conformitatii; • programe postdoctorale create la nivel national • cercetatori romani avand titlul de doctori in stiinte obtinut in strainatate sau stagiipostdoctorale efectuate in strainatate reveniti in tara si angajati in unitati de cercetare • specialisti formati/ instruiti im managementul si administratia cercetarii • manifestari stiintifice sau promotionale cu participare internationala reprezentative; • vizite de lucru si stagii de lunga durata ale unor personalitati stiintifice din strainatate; • propuneri de proiecte transmise la programe internationale; • propuneri de proiecte internationale aprobate; • platforme tehnologice integrate in platforme tehnologice europene. • parteneriate nou create • *Software* • *Baze de date* • *Pagini web*	2	2 11 1 1 2 2 1
▪ *Consultanta, Asistenta tehnica* ▪ *Cursuri de pregatire organizate*		1 2
Constructii institutionale si formare continua: • *linii de invatamant* • *programe de masterat* • *formare continua* • *Pregatire post doctorala* • *Pregatire manageriala*		2

• *Formarea de personal specializat*		

CUPRINS

Obiective generale

Prezentul proiect e un proiect de mare complexitate care si-a propus mai multe obiective:

I. Obtinerea de noi rezultate experimentale si teoretice privind structura nucleara si fazele materiei nucleare

II. Dezvoltarea si constructia de sisteme avansate de detectie

III. Proiectarea si constructia de electronica front-end (FEE) asociata sistemelor avansate de detectie

IV. Dezvoltarea unui sistem de calcul distribuit de tip GRID pentru calculatoare de anvergura

V. Aplicatii in alte sectoare de activitate.

Prin activitatile desfasurate in prezenta faza de executie se aduc contributii la realizarea celui de al patrulea dintre obiectivele generale enumerate mai sus.

Obiectivele fazei de executie

Implementarea si testarea pe ferma locala de computere, componenta a ALICE-GRID, a unui pachet "middleware" bazat pe MonALISA in scopul unei administrari si monitorizari eficiente a intregii structuri ALICE GRID.

Rezumatul fazei

Obiectivul principal al etapei raportate este de a implementa MonALISA pe infrastructura GRID locala de la IFIN. Obiectivul este in concordanta cu obiectivul general al proiectului carea are ca scop realizarea unor cercetari privind **Fizica interactiilor nucleare si a fazelor materiei hadronice.** Realizarea obiectivului general implica o cercetare interdisciplinara cu participarea specialistilor din domeniul fizicii si din domeniul stiintei calculatoarelor.

Specialistii din domeniul Fizicii impreuna cu specialistii din domeniul Stiinta Calculatoarelor au asigurat infrastructura de cercetare din punctul de vedere al resurselor de calcul (clustere de calculatoare) al comunicatiei (legaturi de mare viteza) software

suport pentru aplicatii distribuite (midleware) sisteme de monitorizare si algoritmi specifici prelucrarii distribuite a informatie.

Avand in vedere caracterul eterogen al aplicatiilor suportate de un middleware Grid si avand la dispozitie o platforma generica de monitorizare, in aceasta etapa s-a dezvoltat o interfata de programare (API) care sa puna la dispozitia dezvoltatorilor de aplicatii Grid (sau a portatorilor de aplicatii Grid) un mijloc comod de a raporta la nivelul infrastructurii Grid o serie de parametrii preluati din sensorii locali, accesibili aplicatiei. In continuare s-a implementat aceasta interfata de programare in diverse limbaje de programare C, C++, Java, Perl, Python.

Scopul proiectului este acela de a stabili o procedura eficienta de monitorizare distribuita a resurselor din infrastructura Grid locala IFIN. Exista un numar insemnat de solutii de monitorizare pentru resursele infrastructurilor Grid. Totusi, nici una dintre aceste solutii nu a ajuns la maturitatea necesara monitorizarii transparente si distribuite a intregii infrastructuri Grid. In concluzie, este nevoie de stabilirea unei proceduri de monitorizare a resurselor Grid, aplicata pentru cazul concret al infrastructurii nationale.
Noutatea domeniului si aparitia recenta a specificatiilor GGF GMA fac ca aceasta directie sa fie putin abordata in Romania. In general institutiile de cercetare si academice au desfasurat proiecte locale.

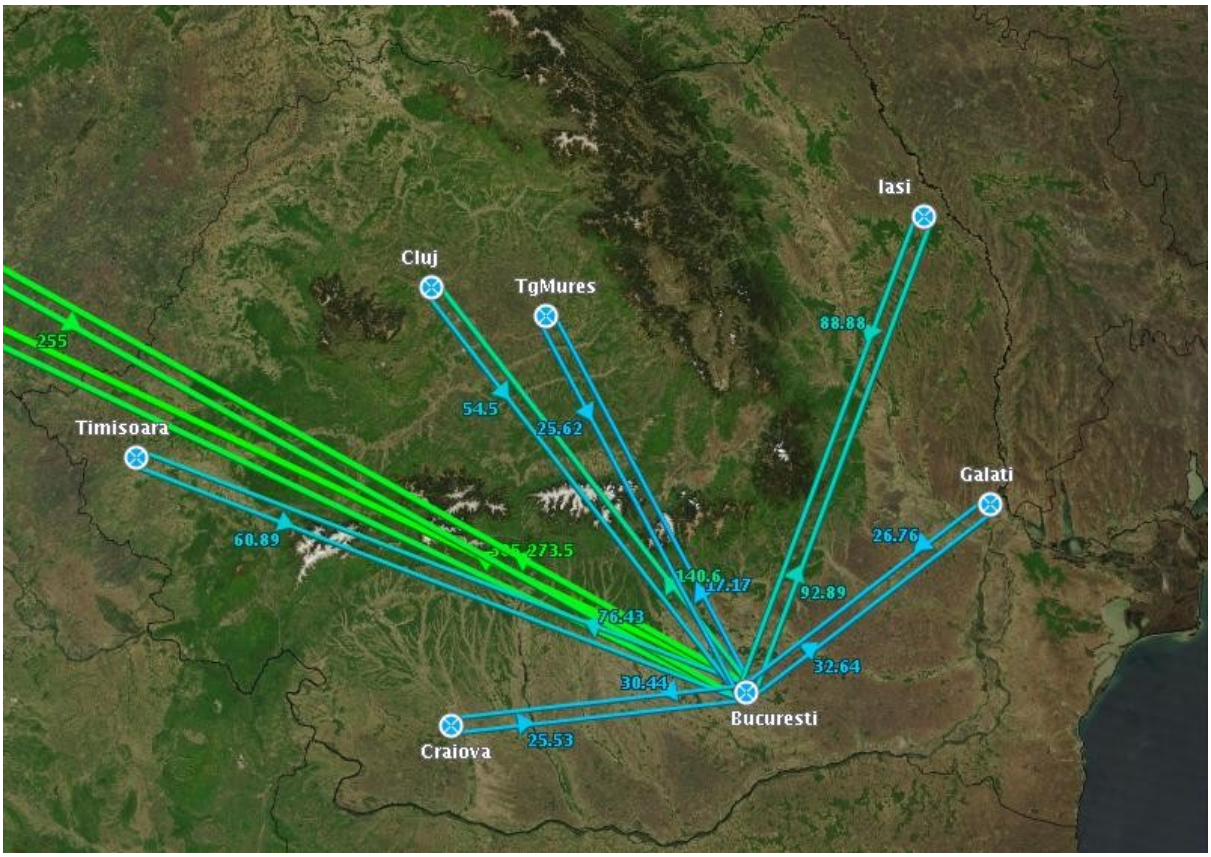
Tematica monitorizarii infrastructurii Grid a fost abordata de colectivul care lucreaza la acest proiect, in colaborare cu echipa de cercetare de la Universitatea CALTECH si grupul de monitorizare de la CERN.
In plus, progresul infrastructurilor Grid nationale face ca monitorizarea adecvata a acestora, cu trimitere la o integrare eficienta in structurile europene si internationale, sa devina un obiectiv national imediat.

In cadrul acestei faze s-a realizat partea de adaptare a sistemului de monitorizare care include cararcteristicile:

- Sistem distribuit format din servicii dinamice autonome, multi-threaded, self-describing care colaborează între ele.
- Suportul pentru monitorizarea unor aplicații distribuite de mari dimensiuni, clustere, griduri.
- Furnizarea de informații real-time şi history.
- Framework pentru dezvoltarea de servicii inteligente pe baza informațiilor real-time din sistem.

S-a realizat implementarea **pachetului „middleware”** pentru monitorarea si coordonarea activitatii pe **sistemul de calcul distribuit** al grupului in cadrul **ALICE GRID si t**estarea lui pentru monitorarea si coordonarea activitatii pe sistemul de calcul distribuit al grupului in cadrul ALICE/GRID. Sistemul in varianta curenta, el fiind in permanenta dezvoltare, se foloseste in cadrul proiectelor dezvoltate la UPB, IFIN, ICI, INCAS, Univ Timisoara, Univ Cluj Napoca. De asemena se utilizeaza de catre CALTECH si CERN. In figura se prezinta infrastructura de comunicatie RoEduNEt si conexiunea externa la GEANT vizualizata in cadrul sistemului de monitorizare

MonaLISA.



Descrierea tehnica a cercetarii – fundamentele sistemului de monitorizare- se face in continuare si in anexele asociate (Anexa1 – MonaLISA arhitectura, Anexa 2- MonaLISA utilizare). Anexele sunt in limba engleza deoarece se adreseaza tuturor cercetatorilor care doresc sa utilizeze acest sistem de monitorizare.

Descrierea stiintifica si tehnica

1. Introducere

In **infrastructura Grid** pot sa apara urmatoarele categorii de resurse:

- resurse de calcul – clustere formate din calculatoare de birou, *blade*-uri de calcul, alte resurse gen *rack-clusters*, solutii de stocare, memorii, dispozitive de intrare/iesire;

- resurse de date – baze de date/cunostinte, in general disponibile in urma unor colaborari cu oragnizatii de profil;
- resurse de retea – subretele/canale de viteza medie (Fast Ethernet) si viteza mare (Gigabit Ethernet, ATM etc.) folosite in transferuri de date pentru aplicatii de inalta performanta (HPC).

Modelul Grid propune o metoda promitatoare de partajare pe scara larga a resurselor informatice, strategia nationala in directiile imbunatatirii componentelor de interconectare si asezarii a centrelor de calcul/stocare fiind vitala. Aceasta strategie nationala depinde in mare masura de detectarea punctelor-cheie curente si/sau de viitor; cu alte cuvinte, strategia nationala de dezvoltare a infrastructurii Grid depinde de monitorizarea adecvata a acesteia.

Resursele de interconectare reprezinta una dintre cele mai importante componente ale infrastructurii Grid. In Romania, reteaua educationala constituie o parte importanta din infrastructura de conectare prevazuta pentru construirea de medii Grid. Prin urmare, este nevoie de monitorizarea retelei educationale folosind unelte de monitorizare de tip Grid. Nu in ultimul rand, este important ca datele de monitorizare obtinute sa fie centralizate si filtrate, apoi oferite spre analiza specialistilor. Dezvoltarea de aplicatii bazate pe GRID impune utilizarea informatiilor obtinute prin monitorizare. S-au exploatat resursele deja existente in institutiile implicate in proiect, in primul rand.

Toate aceste resurse trebuie monitorizate intr-un mod adecvat, in cadrul mai multor medii de test. UPB ofera mai multe medii de test, intre care un mediu de lucru Globus instalat intre mai multe laboratoare, un sistem cluster bazat pe MOSIX si un sistem de servere de mare performanta pe care ruleaza SUN Grid Engine. Institutia RoeduNet ofera o gama larga de facilitati de interconectare, cu produse high-end CISCO si 3COM, precum si cu produse specifice pietei low-end din Romania. In concluzie,infrastructura de test acopera o gama larga de posibili utilizatori, in concordanta cu tendintele Europene si internationale, de adaptare a mediilor Grid existente la cerintele unei multitudini de tipuri de consumatori.

2. Medii GRID

GRID este un termen propus la mijlocul anilor ’90 pentru definirea unei infrastructuri de calcul distribuite, dedicată în special cercetarilor ştiinţifice cu cerinţe computaţionale ridicate. Deşi există multe domenii conexe cu dezvoltarea sistemelor GRID – atât reţelele de calculatoare cât şi sistemele de calcul distribuite au deja o istorie a lor – noua infrastructură impune utilizarea de tehnologii dedicate.

GRID se doreşte a fi un set de aplicaţii şi servicii partajate, împreună cu regulile de operare asupra lor. Aceste reguli vor permite GRID să se adapteze la modificările unui context dinamic prin intermediul serviciilor şi aplicaţiilor care supraveghează nivelul infrastructurii.

În plus, GRID poate fi considerat un context semantic care poate fi construit dinamic, în funcţie de profilul aplicaţiilor care îl solicită. În această abordare au fost

definite mai multe tipuri de GRID: grid de date (*data grid*), grid computațional (*computing grid*), grid HDTV (*high definition television grid*), etc. Alte tipuri de GRID vor trebui definite în continuare, pentru a defini alte contexte semantice absolut necesare integrării unui astfel de domeniu în societatea informațională. Dezvoltarea tehnologiilor GRID trebuie să îmbunătățească accesul utilizatorilor la informațiile de care au nevoie şi să fie totodată un motor în plus pentru dezvoltarea societații informaționale în sine. Asocierea de componente grid de diverse tipuri trebuie să fie accesibilă utilizatorilor şi aplicațiilor lor în forma unui portal GRID (*grid portal*).

Din punct de vedere al utilizatorilor, cerința fundamentală pe care trebuie să o respecte GRID este accesarea de informații şi cunoştințe la cerere, la orice moment de timp şi din orice loc. Este echivalent cu a pune un ştecher în priză: nu interesează de unde vine tensiunea, e important să fie suficientă şi trebuie plătit ceea ce se consumă. Din această analogie devine clar că GRID-ul nu înlocuieşte Internetul, ci îi dă o nouă întrebuințare.

Monitorizarea resurselor in medii GRID

3.1 Standardizare

Capacitatea de a monitoriza si de a gestiona elementele arhitecturilor pentru calcul distribuit reprezinta o componenta critica de luat in considerare in realizarea de astfel de sisteme de mare performanta. Datele monitorizate sunt necesare pentru determinarea surselor generatoare de probleme si pentru a imbunatati sistemele si aplicatiile care ruleaza pe aceste sisteme pentru ajungerea la performante din ce in ce mai bune. Detectarea erorilor si mecanismele de recuperare din astfel de erori au nevoie de datele monitorizate pentru determinarea disponibilitatilor unor servere din sistem sau in cazul intalnirii unor servere cu probleme daca e nevoie doar de repornirea unui astfel de server sau daca problema e de mai mare amploare eventual transferarea serviciilor intr-o alta directie.

Exista mai multe grupuri de lucru care incearca dezvoltarea unor sisteme de monitorizare de arhitecturi de calcul de tip Grid pentru rezolvarea unor astfel de probleme, insa de curand aceste grupuri au inceput sa simta din ce in ce mai mult nevoia de interoperabilitate a rezultatelor obtinute intre grupurile respective. Pentru obtinerea acestui obiectiv a fost creata o arhitectura de monitorizare orientata special pentru arhitecturi de sisteme de tip Grid. Un sistem de monitorizare de tip Grid se diferentiaza de un sistem general de monitorizare prin faptul ca acest sistem trebuie sa fie scalabil pe retele de mari intinderi si sa includa o mare varietate de resurse eterogene. Mai trebuie de asemenea sa fie integrat in alte componente Grid in ceea ce priveste problemele de nume si de securitate. Acest sistem ce are numele de Grid Monitoring Architecture (GMA) se ocupa de toate aceste probleme si este destulde general pentru a putea fi adaptat spre a fi folosit in medii distribuite altele decat cele de tip Grid. De exemplu se poate folosi cu ferme de procesare de mari dimensiuni care necesita monitorizare constanta pentru a asigura faptul ca toate nodurile implicate functioneaza corect.

Daca ne gandim la potentialul adus de mii de resurse aflate in diverse locatii geografice si zeci de mii de utilizatori Grid este evidenta importanta gestionarii corecte a datelor pentru scalare si in acelasi timp protejarea datelor impotriva interferentelor. Pentru a permite scalarea atat in ceea ce priveste administrarea, cat si in ceea ce priveste impactul asupra performantelor pentru un astfel de sistem elementul decizional asupra a ceea ce se vrea a se monitoriza, asupra frecventei de masurare a diverselor marimi implicate si asupra modului in care datele obtinute sunt facute publice acest sistem a fost create spre a fi distribuit si dinamic. Deci, in locul unei componente de gestiune centralizata sistemul este compus din mai multe componente de gestiune independente sincronizate prin intermediul unui serviciu de directoare, care la randul lui poate fi distribuit. Gestionarea datelor astfel ajuta de asemenea la minimizarea efectelor datorate erorilor de retea sau datorate caderilor unor statii de lucru, permitand in acelasi timp o robustete sporita sub efectul conditiilor pe care tocmai le incearca a le detecta. In astfel de modele, cum ar fi CORBA Event Service, toate erorile de comunicatie daca ar trece printr-o unica componenta centrala ar duce in mod sigur la o gatuire. In sistemul acesta de asemenea datele monitorizate care sunt direct legate de performanta sistemului, ceea ce constituie de fapt majoritatea traficului implicat, circula direct de la producatorii acelor date la consumatorii lor. In acest mod perechile individuale producator-consumator pot aplica algoritmi de potrivire pe baza necesitatilor de negociere si cantitatea de date care cirula prin sistem poate fi controlata intr-o maniera precisa si localizata pe baza unor considerente de incarcare. Design-ul permite de asemenea replicarea si reducerea datelor in cadrul unor componente intermediare ce actioneaza asemenea unor zone tampon sau filtre de tip producator-consumator. Folosirea acestor componente intermediare micsoreaza incarcarea producatorilor acelor date care pot fi de interes pentru mai multi consumatori, avand de asemenea influenta si asupra reducerii traficului de retea si aceasta deoarece intermediarii pot fi plasati in apropierea consumatorilor de date. Serviciul de directoare contine doar metadate legate de datele de performanta si de componentele de sistem si este accesat destul de putin, ceea ce are ca efect reducerea sanselor unor gatuiri.

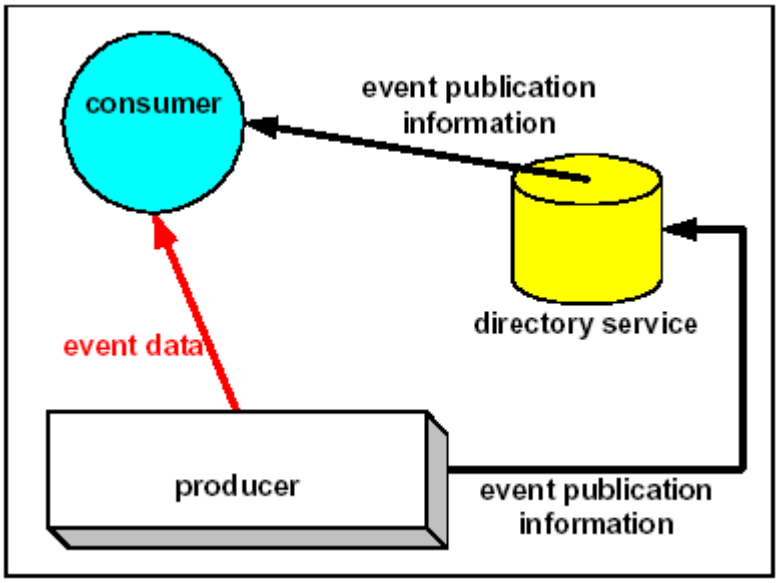
Initial pentru monitorizarea datelor s-a pornit de la o solutie bazata pe SNMP, dar aceasta a fost in scurt timp rejectata deoarece modelul bazat pe GET/SET oferit de SNMP nu este destul de bogat si de asemenea nu exista implementat un suport pentru subscriere. De asemenea modelul de securitate oferit nu se mapeaza bine pe Infrastructura de Securitate Grid.

Arhitectura sistemului suporta doua modele, primul fiind cel de tip producator-consumator, fiind similar celor existente deja in cadrul unor sisteme cum ar fi CORBA Event Service, iar cel de-al doilea fiind un model de tip intrebare-raspuns. Pentru fiecare dintre aceste doua modele producatorii si consumatorii care accepta conexiuni isi fac cunoscute intentiile prin intermediul unui serviciu de directoare. Consumatorii folosesc serviciul de directoare pentru a localiza unul sau mai multi producatori care genereaza tipurile de date de care sunt acestia interesati. Fiecare consumator subscrie apoi sau interogheaza direct unul dintre potentialii producatori descoperiti. Intr-o maniera similara un producator poate interofa serviciul de directoare pentru a localiza unul sau mai multi consumatori care sunt dispusi sa accepte si sa proceseze datele intr-o anumita maniera – de exemplu un consumator care e dispus sa arhiveze datele pentru analize viitoare. Odata

ce un consumator corespunzator este descoperit producatorul il contacteaza si initiaza un transfer al acelor date.

Arhitectura este alcatuira din mai multe componente (dupa cum se poate observa si in figura) si anume:

- consumatori
- producatori
- serviciul de directoare



Prin definirea a trei interfete (cea intre consumator si producator, cea intre consumator si serviciul de directoare si cea intre producator si serviciul de directoare) se pot construi servicii de monitorizare de tip grid care sa poata interopera.

Serviciul de directoare

Pentru localizarea, numirea si descrierea caracteristicilor structurale ale tuturor datelor disponibile s-a impus existenta unui serviciu de directoare distribuit in care sa se poata publica aceste informatii. Scopul principal al acestui serviciu de directoare este acela de a permite consumatorilor de informatii (utilizatorii, instrumentele de vizualizare, programele) descoperirea si intelegerea caracteristicilor informatiilor disponibile. Suplimentar producatorii de informatii trebuie sa poata fi capabili de a actualiza informatiile spre a reflecta corect starea sistemului. Directorul de servicii contine o lista a tuturor datelor disponibile si a producatorilor acestor date. Acest aspect permite clientilor descoperirea datelor care sunt disponibile, care sunt caracterisiticile acestor date si de asemenea contactarea producatorilor ce pot furniza acele date. Presupunem ca numele si caracteristicile asociate cu datele de performanta dinamice se schimba cu o frecventa relativ scazuta (in contrast cu datele de performanta).

Consumatorul

Un consumator reprezinta orice program care primeste date de la un producator. Consumatorii care accepta cereri asincrone de la producari pot publica aceste informatii in serviciul de directoare. Functiile suportate de catre consumator sunt:

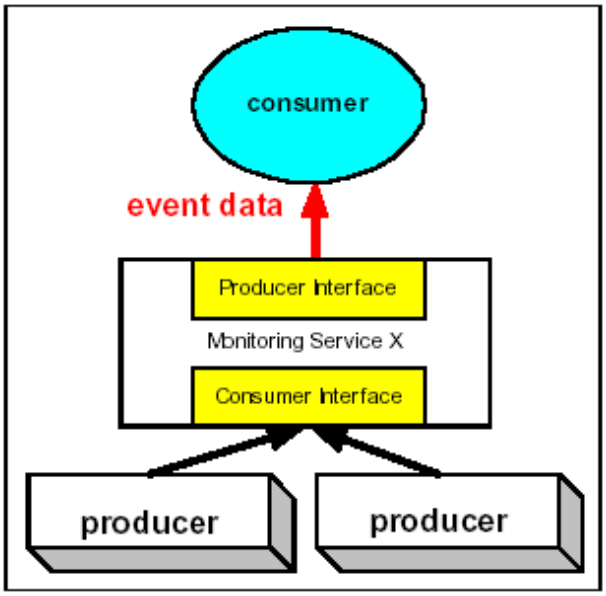
1. Autorizarea catre producator – Consumatorul contacteaza un producator si ii dovedeste acestuia identitatea sa. Acest lucru se realizeaza o singura data pe fiecare sesiune, dar se poate folosi si un mecanism de autorizare pe fiecare cerere.
2. Autorizarea de la producator – Consumatorul accepta cereri de autorizare de la producator si face verificarea identitatii acestuia.
3. Interogare – Consumatorul primeste unul sau mai multe seturi de date de la producator. Optional se pot folosi filtre pentru indicarea interesului pentru o submultime de date.
4. Inscrierea initiata de consumator – Consumatorul stabileste o conexiune cu producatorul pentru primirea de date.
5. Dezinscrierea initiata de consumator – Consumatorul informeaza producatorul in legatura cu incheierea unei inscrieri.
6. Inscrierea initiata de producator – Consumatorul accepta inscrieri ale producatorilor care doresc trimiterea de date.
7. Dezinscrierea initiata de producator – Consumatorul accepta cereri de incheiere a unei inscrieri de la producator.
8. Autorizarea in serviciul de directoare – Consumatorul contacteaza serviciul de directoare si isi dovedeste propria identitate.
9. Verificare – Consumatorul initiaza o cerere catre serviciul de directoare pentru stabilirea producatorilor ce pot furniza anumite tipuri de date.

Producatorul

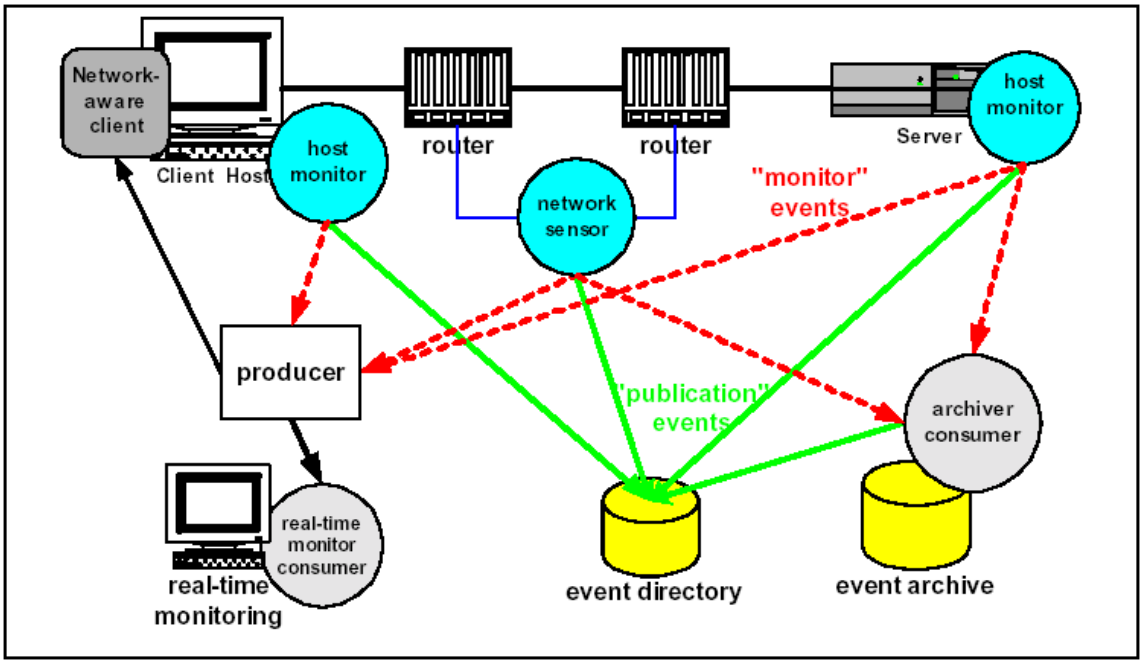
Producatorii sunt responsabili cu furnizarea datelor catre consumatori, fie prin cereri, fie in mod asincron. Producatorii isi publica seturile de date disponibile in serviciul de directoare. Functiile suportate de catre producator sunt:

1. Autorizarea din partea consumatorului – Producatorul stabileste identitatea unui consumator si permisiunile de acces ale acestuia. Autorizarea poate fi combinata cu inscrierea si efectuarea de cereri sau poate fi efectuata ca proces separat.
2. Autorizarea catre consumator – Producatorul contacteaza un consumator si dovedeste acestuia propria identitate.
3. Interogarea – Producatorul returneaza unul sau mai multe seturi de date ca raspuns la o interogare a unui client.

In cadrul sistemului mai pot exista de asemenea componente ce pot fi atat consumatori, cat si producatori. De exemplu un consumator poate colecta date de la mai multi producatori si apoi poate folosi aceste date pentru generarea unor noi seturi de date (de exemplu prin derivarea datelor initiale), care date pot fi furnizate unor alti consumatori, dupa cum se poate vedea si din figura urmatoare.



Un exemplu de folosire a sistemului poate fi vazut in figura urmatoare. Datele sunt colectate pentru fiecare sistem gazda si pentru fiecare router de retea si sunt trimise unui producator. Producatorul inregistreaza disponibilitatea acelor date in serviciul de directoare. Un consumator de monitorizare real-time se inscrie pentru toate datele pentru vizualizarea acestora real-time si pentru efectuarea de analize asupra acelor seturi de date. Producatorul este capabil de efectuarea unor calcule de sumarizare a throughput-ului si a latentei de exemplu, permitand unui alt client setarea optima a dimensiunii stivei TCP.O submulime de date ale producatorului este de asemenea trimisa pentru colectare.



3.2 Ierarhii de caracteristici

Principiu

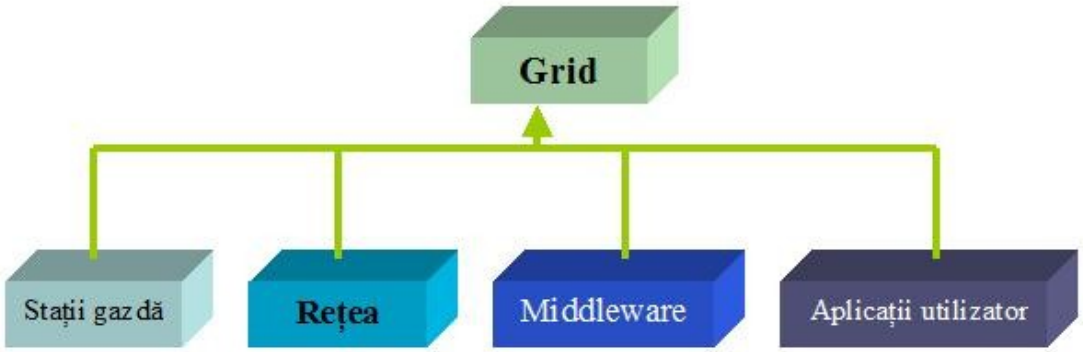
Pot fi monitorizate in acelaşi timp un numar de oricâte proprietăți de obiecte Grid. De aceea are sens gruparea datelor într-o formă ierarhică, relativ la un obiect , în special cand se lucrează cu obiecte complexe, care cuprind alte obiecte complexe, această formă repetându-se de mai multe ori. Ierarhia formată urmărind această viziune poate fi extinsă până când va cuprinde întregul sistem Grid. Deoarece definirea Grid-ului este ambiguă, am adoptat definiția dată de Forumul Global de Grid (Grid Global Forum):

"... grupuri de resurse computaționale conectate care pot să lucreze împreună ca o singură entitate, Grid".

Cu această definire, cererea de date de la un nod din ierarhie poate să returneze rezultate de la toate ramurile cuprinse în acel nod, dând o viziune completă asupra stării nodului. Deoarece numărul de proprietăți este mare, nu este recomandată o cerere de genul "date de la întreg sistemul Grid".

3.2.1 Taxonomia ierarhiilor de caracteristici:

Într-o ierarhie de caracteristici care acoperă întregul sistem Grid, nodul rădăcină este însuşi sistemul Grid. Primul nivel de sub nodul rădăcină defineşte componentele sistemului Grid, aşa cum este văzut de sistemele de monitorizare (şi de administratorul de sistem). Există patru tipuri de componente Grid: statiile gazdă, rețeaua, middleware-ul şi aplicațiile utilizator.



Este foarte important să existe o ierarhie de caracteristici completă, standardizată, pentru toate cele patru componente ale sistemului Grid. Acest subiect este departe de a fi finalizat, fiind foarte combatut asupra caracteristicilor ce se pentru fiecare componentă a Grid-ului. A fost propusă o ierarhie completa pentru stațiile gazdă si pentru rețea.

Grid

Caracteristicile pentru stațiile gazdă:

Stații gazdă
Rețea
Middleware
Aplicații utilizator

Caracteristicile pentru stațiile gazdă sunt folosite pentru descrierea stațiilor în cadrul mediului Grid. Aceste stații pot să fie resurse computaționale, sisteme de stocare de date, sau alte resurse. Cele mai importante caracteristici din această categorie sunt:

Încărcarea pe CPU - fracțiunea de timp de procesor utilizant pentru procesare într-un interval de timp; cu cât este mai mic intervalul de referință, cu atât este mai exactă valoarea acestui parametru; de obicei fereastra de timp aleasă este între o secundă şi câteva minute.

Uptime-ul de sistem – timpul total cat sistemul nu a fost idle ; acest parametru poate fi măsurat ca o valoare pentru o perioadă de timp (asemănător cu încărcare pe procesor, dar fereastra de timp se alege între un minut şi câteva ore) sau de la ultima pornire a sistemului.

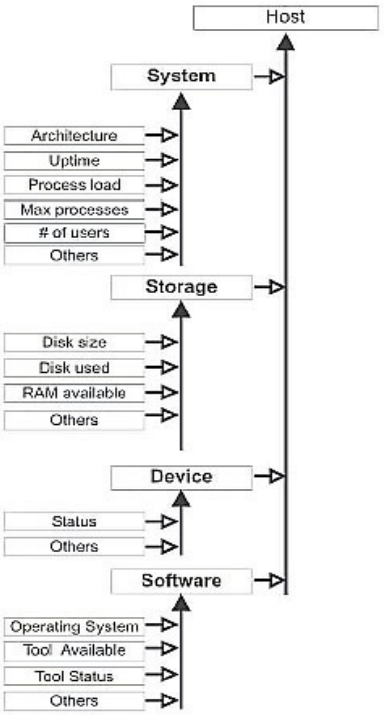
Dimensiunea discului – spațiul total de pe disc sau de pe discurile existente ;

Spațiul liber de pe disc – spațiul disponibil de pe discul / discurile din sistem ;

Memoria disponibilă – memoria totală volatilă / nevolatilă ;

Arhitectura de sistem – arhitectura sistemului gazdă ;

Sistemul de operare – sistemul de operare de pe maşina gazdă



Caracteristicile de rețea:

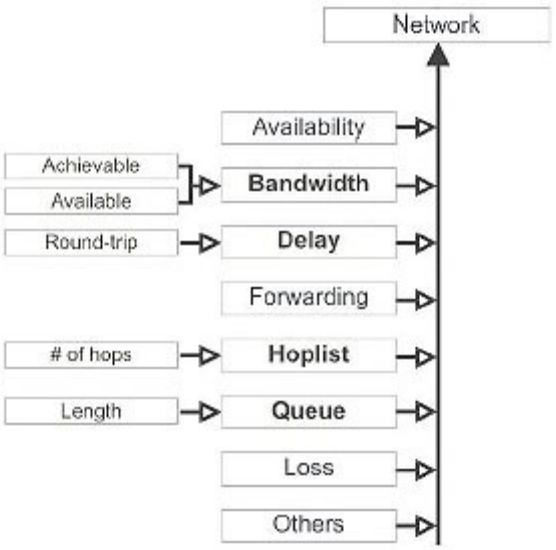
Caracteristicile de rețea sunt folosite pentru a descrie elementele de rețea care interconectează sistemele gazdă din mediile Grid. Printre elementele de rețea sunt routerele, switch-urile şi alte dispozitive. Cele mai importante caracteristici din această categorie sunt:

Lungimea de bandă TCP disponibilă – lungimea de bandă maximă disponibilă atunci când se foloseşte numai protocolul de comunicare TCP .

RTT (Round Trip Time) – timpul total consumat de un mesaj minimal, trimis utilizând protocolul ICMP, pentru a ajunge de la sursă la destinație si înapoi la sursă

Numărul de hopuri – numărul de noduri (hopuri) care definesc o legătură de rețea pentru un nivel ISO OSI dat între o sursă şi o destinație pe legătura .

Pierderea de pachete – procentul din numărul de pachete trimise care nu au ajuns la destinație.

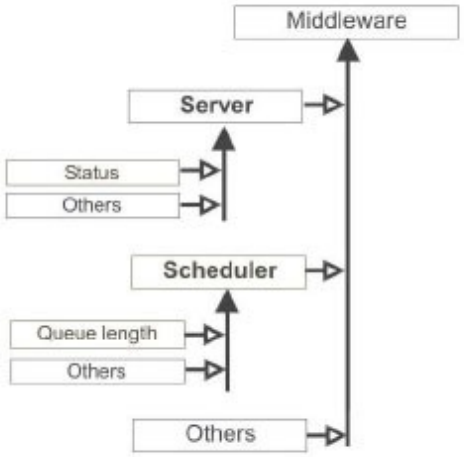


Caracteristici de middleware

Caracteristicile de middleware sunt folosite pentru a descrie evolutia middleware-ului în cadrul Grid-ului. Printre elementele de middleware se află şi componentele de securitate şi planificatoarele. Cele mai importante caracteristici din această categorie sunt.

Lungimea cozii planificatorului .

Starea componentei de middleware – starea curentă a unei componente de middleware (în execuție, oprită, etc.).



Caracteristici ale aplicațiilor utilizator

Aceste caracteristici sunt folosite pentru a descrie aplicațiile utilizator în cadrul mediului Grid. Nu există nici un consens la ce caracteristici sa fie definite la acest nivel. Utilizatorul este liber sa decidă care sunt caracteristicile folosite. Există o caracteristică luat în seamă de mai toți utilizatorii, **starea de execuție a aplicației.**

3.3 Metode de culegere a informatiei.

3.3.1 Etape generale ale monitorizarii

Achizitia datelor

Obtinerea datelor ridica multe probleme de cercetare printre ingineri. Datele trebuie sa fie culese dinamic, la timp, si cu acuratetea ceruta. Datele trebuie sa fie relevante pentru sistemul minitorizat si sa il descie corect. Cererea datelor poate sa perturbe sistemul monitorizat si sa schimbe insasi rezultatul procesului de monitorizare. Odata culese, rareori sistemul de monitorizare poate folosi in mod automat, chiar datele respective, interventia omului fiind deseori necesara. De asemenea dupa efectuarea procesului de monitorizare, aproape in toate cazurile, modificarile sunt facute de factorul uman.

Vizualizarea

Vizualizarea datelor ajuta foarte mult la detectarea greselilor si la interventia celor care supreavegheaza sistemul. Pe masura ce sistemele Grid tind sa-si mareasca dimensiunea (pe masura ce sunt adaugate totmai multe clustere si resurse), se aduna o cantitate mare de date, ceea ce face din vizualizare un factor cu atat mai important. Fara metodele corespunzatoare de reprezentare a datelor, toata aceasta cantitate da date poate deveni inutila. Tehnicile de vizualizare se refera la reprezentarea organizatiilor virtuale, starea resurselor (de exemplu incarcarea masinii sau a clusterului si tipurile de incarcare), reprezentarile ierarhice ale resurselor, diagramele conform topologiei si asa mai departe.

Arhivarea

Toata cantitatea de date de monitorizare venita de la diverse componente ale sistemelor Grid ar trebui de asemenea stocata pentru o perioada de timp, pana cand fie devine invechita, fie devine prea mare pentru sistemul de stocare, caz in care sistemul poate sa ofere o posibilitate de a-i ajusta dimensiunea, controland granularitatea informatiei stocate sau eliminand datele cele mai vechi.

3.3.2 Cererinte ale sistemelor de monitorizare

Standardizarea si deschiderea

Este dificil, daca nu imposibil sa se asigure instalarea aceluiasi software in organizatii diferite. Prin urmare, o unealta de monitorizare unica, dedicata, nu este o solutie potrivita pentru mediile Grid. Din aceasta cauza, vom analiza unelte care urmeaza o cale libera in dezvoltare. Aceasta cale a fost stabilita de Grid Performance Working Group al Global Grid Forum, in cadrul incercarii lor de definire a unei Arhitecturi de Monitorizare in Grid (GMA).

Scalabilitatea

Ca pentru orice sistem distribuit, scalabilitatea este o problema foarte importanta in dezvoltarea aplicatiilor de monitorizare pentru Grid. Resursele sunt raspandite, deseori pe arii largi, ceea ce adauga constrangeri de latenta cand e vorba de cantitati mari de date (stocarea si regasirea datelor). Cu toate acestea exista o alta latura a scalabilitatii Grid-ului : factorul uman. Exista o diferenta intre plasarea unui sistem de monitorizare distribuit in interiorul propriei infrasctructuri si plasarea aceluiasi sistem intr-un Grid care se intinde peste mai multe organizatii, de cele mai multe ori avand propriile strategii de suport pentru monitorizare. Cea mai mare problema in ceea ce priveste scalabilitatea o constituie asigurarea unor stari de lucru sigure pentru infrastructura de monitorizare, chiar daca unele conditii nedorite cum ar fi defectarea unor statii de lucru.
Speculand putem spune ca solutia la aceasta problema a scalabilitatii ar putea veni din domeniul cercetarilor in domeniul procesarii punct-la-punct, care se axeaza pe probleme precum localizarea automata a resurselor dispersate si supravietuirea sistemului in absenta unor membrii din sistem.

Securitatea

Securitatea reprezinta una din probleme cele mai importante ale oricarui sistem Grid. Organizatiile virtuale adesea se bazeaza pe diverse politici de securitate si monitorizarea unor asemenea medii trebuie sa faca fata oricaror astfel de situatii. Datele pot sa nu fie oferite decat unor anumiti utilizatori acreditati, punand un accent special pe problema unei „singure inscrieri" (utilizatorii nu ar trebui fortati sa se autentifice decat o singura data la inceputul unei sesiuni de lucru pentru a accesa datele la care au acces, chiar daca aceste date sunt situate in mai multe locatii fizice chiar apartinand unor organizatii diferite) si pe problema „delegatiei" (utilizatorii ar trebuie sa poate sa transmite drepturile proprii de autentificare altor utilizatori si/sau altor programe). Propunerea GGF_GMA sugereaza folosirea certificatelor de identitate X.509 si a protocolului de autentificare Secured Socket Layer (SSL) pentru implementarea problemei autentificarii.
Sistemele de monitorizare ar trebui de asemenea sa fie capabile de o adaptare dinamica a strategiilor de lucru in functie de drepturile disponibile intr-un subsistem particular (de exemplu un instrument de monitorizare ar putea avea drepturi de citire intr-un subsistem, dar nici un fel de drept intr-un altul, in functie de politica de securitate a organizatiei virtuale din care face parte subsistemul respectiv; intr-un astfel de caz datele ar trebui sa fie obtinute de catre sistemul de monitorizare folosind alt instrument de monitorizare sau sub alta forma).

Datele tinta

Exista patru niveluri de date pentru sistemele de monitorizare Grid: gazda, retea, middleware si aplicatii utilizator.
Host Nivelul gazda se refera la infrastructura de procesare si stocare a sistemelor Grid. Din aceasta categorie fac parte mainframe-uri, supercomputere, clustere formate din statii de lucru obisnuite, centre de tip „blade', sisteme de stocare, sisteme I/O, senzori, fiecare avand nevoi de monitorizare speciale.
Retea Infrastructura de tip retea reprezinta o componenta cheie pentru infrastructura de procesare Grid, prind urmare deservind propriul nivel. Din acest nivel fac parte subretelele, conexiunile de mare viteza, Gateway-uri, rutere si chiar switch-uri.
Middleware Componentele middleware ale sistemelor Grid actioneaza, de unde si numele, la nivelul de mijloc intre aplicatiile utilizatorilor si infrastructura Grid. Subparti ale acestui nivel includ elementele de securitate, alocarea resurselor si altele, fiecare avand nevoi speciale de monitorizare.
Aplicatii utilizator Aplicatiile utilizator intalnite in sistemele Grid sunt de o mare diversitate. Prin urmare cum si ce trebuie monitorizat depinde de tipurile aplicatiilor respective. Cu toate acestea un subset minim de date monitorizare precum starea de rulare/oprire a aplicatiilor, ar trebui sa poata fi monitorizate.
Pentru ca un sistem de monitorizare Grid sa fie complet toate aceste patru nivele de date trebuie sa fie cu succes atinse. Aceasta include probleme generale de monitorizare speciale precum achizitia, vizualizarea si pastrarea datelor, dar si adaugarea unor selectii de date de monitorizare corespunzatoare (datorita caracterului eterogen al componentelor

Grid). De asemenea pe masura de sistemele Grid prind amploare s-ar putea ajunge la concluzia ca stocarea sau senzorii de date trebuie vazute ca nivele noi de date.

3.3.3 Tehnici de monitorizare

In ultimii ani a aparut un consens in ceea ce priveste elementele de baza care tin de monitorizare. Ceea ce trebuie insa gasit pentru fiecare sistem este reprezentat de *caracteristicile* componentelor sistemelor ce sunt monitorizate.
Instrumentele de monitorizare trebuie sa stranga date ale caracteristicilor cerute sub presupunerea unor medii eterogene. Acestea trebuie sa furnizeze instrumente de filtare si analizare a datelor stranse. De asemenea ele trebuie sa reduca incarcarea cat mai mult posibil, aceasta in contextul unor operatii cat mai compacte.
Includerea dinamica a unor masini ca gazde monitorizate este de asemenea o cerinta necesare. Tehnicile de monitorizare precum inregistrarea, pastrarea si vizualizarea datelor trebuie alese cu mare atentie.
Avand toate aceste cerinte de indeplinit aceasta sectiune incearca definirea a ceea ce inseamna o caracteristica si care caracteristici sunt in mod obisnuit puse sub supraveghere intr-un sistem de monitorizare Grid. Se face o distinctie clara intre dooua tehnici generale de efectuare a masuratorilor: activ si pasiv.

Caracteristici

Prin extinderea definitiei GGF_NMWG o caracteristca resprezinta o *proprietate intrinseca a unui obiect, sistem sau proces care descrie modul in care apare sau actioneaza in anumite contexte de mediu.* De exemplu o *caracteristica de retea* reprezinta o *proprietate intrinseca a unei portiuni din retea care este cumva inrudita cu performanta sau siguranta Internetului*. In acest exemplu obiectul reprezinta o portiune din retea, in timp ce contextul inconjurator este reprezentat de sistemul Grid monitorizat, iar obiectele reprezinta componentele mediului care ofera partea de metaprocesare: reteaua, calculatoarele care proceseaza, partea de middleware (software-ul ce permite aplicatiilor utilizatorilor folosirea retelei si a statiilor de lucru) si aplicatiile utilizator.

Masuratori

Principiul

Datele pentru o anumita caracteristica sunt obtinute prin intermediul *procesului de masurare*. Nume alternative ale acestui proces ar fi *probare* sau *observare*. Procesul de masurare este condus intotdeauna (pentru a obtine informatii corecte) dupa o metodologie de masurare, aceasta reprezentand o tehnica pentru estimarea masurarii unei caracteristici. In acest context *observatiile* pot fi *atomice*, stranse laolata (*probe*) sau derivate dintr-o proba (*statistice*).

Instrumente specializate, numite *senzori*, sunt cele care realizeaza observatiile. In functie de domeniul de aplicabilitate exista cel putin patru tipuri de senzori: de retea, de

statie, pentru middleware si pentru aplicatiile utilizator. In afara de datele obtinute senzorii pot adauga o inregistrare de timp fiecarui eveniment inregistrat. Aceasta poate permite realizarea intr-un viitor a procesarii, posibil statistice, a datelor stranse. Daca se foloseste inregistrarea timpului aceasta trebuie sa fie consistenta, cu alte cuvinte trebuie asigurat un ceas global, de exemplu prin folosirea unor servere de retea specializate (de exemplu servere de Network Time Protocol).

Senzorii

Un senzor reprezinta orice program care genereaza un eveniment incadrat in timp al unei marimi de performanta monitorizata. Ca exemple de senzori avem:

- incarcarea procesorului;
- incarcarea memoriei;
- incarcarea retelei.

Senzorii pot fi de asemenea folositi pentru monitorizarea conditiilor de eroare, cum ar fi:

- un proces server mort;
- erori de CRC intr-un router.

Senzorii pot fi grupati in functie de urmatoarele categorii:

a. Senzori gazda

Acesti senzori realizeaza orice tip de monitorizare pentru o statie de lucru, cum ar fi:

- incarcarea procesorului;
- memoria disponibila;
- retransmisiile TCP.

Acestia pot rula la distanta de statia monitorizata (de exemplu monitorizarea bazata pe SNMP: Simple Network Management Protocol).

b. Senzori de retea

Acestia efectueaza interogari SNMP cu oricare dispozitiv de retea, cum ar fi;

- un router;
- un switch.

c. Senzori la nivel de procese

Acestia genereaza evenimente atunci cand exista o schimbare in starea unui proces, cum ar fi:

- pornirea procesului;
- oprirea normala a procesului;
- oprirea anormala a procesului;
- un prag dinamic este atins (de exemplu numarul de utilizatori pe o anumita perioada de timp).

d. Senzori la nivel de aplicatii

Acesti senzori sunt integrati in cadrul unor aplicatii. Ei pot genera evenimente in diverse condirii, cum ar fi:

- conectarea/deconectarea unui utilizator;
- schimbarea parolei unui utilizator;
- -atingerea unui prag static;
- primirea unui semnal UNIX;
- alte functii definite de utilizator.

Masuratori active, pasive si putin deranjante

In functie de modul de colectare a datelor exista doua directii in monitorizare: monitorizarea prin *masuratori active* si monitorizarea prin *masuratori pasive*. Monitorizarea prin masuratori active implica faptul ca sistemul monitorizat in mod sistem genereaza proble care afecteaza intregul sistem pentru estimarea valorilor caracteristicilor, cum ar fi cazul lansarii de pachete pentru estimarea lungimii de banda sau folosirea unor semnale heart-beat pentru detectarea statiilor functionabile.

Monitorizarea prin masuratori pasive necesita ca nici o astfel de proba generata sistematic sa nu fie folosita. Sistemul de monitorizare se bazeaza in acest caz complet pe datele stranse de software-ul de nivel mai scazut, cum ar fi sistemul de operare, pentru gasirea valorilor caracteristicilor de performanta.

Presupunand ca ambele tipuri de masuratori ar obtine aceleasi clase de rezultate, fie ca sunt sigure sau exacte sau amandoua, ar fi de preferat folosirea masuratorilor pasive. Din pacate nu intotdeauna se poate acest lucru deoarece instrumentele pe care se bazeaza masuratorile pasive pot duce la unele probleme grave. Cercetari recente in domeniul retelelor (Gygabit, Myrinet, fibre optice) si in domeniul procesarii (procesoarele Intel si AMD) fac ca masuratorile active sa devina din ce in ce mai „ieftine”.

3.3.4 Monitorizarea Grid

Folosirea masuratorilor active induc o alta problema delicata: *efectul de probare*. Efectul de probare inseamna elementele perturbatorii asupra functiilor sistemului monitorizat introduse de catre chiar sistemul de monitorizare. Prin urmare sistemul de monitorizare inregistreaza nu doar informatii care tin de sistemul tinta, ci chiar informatii care tin de sistemul in sine. Acest lucru este de nedorit, mai ales daca perturbatiile nu pot fi tinute intre anumite limite. Prin urmare scopul curent al cercetatorilor este acela de a obtine datele de monitorizare prin tinerea datelor *cat mai putin posibil* sub influenta elementelor perturbante.

Metode de transmitere a informatiilor in sistemele Grid

Grid-ul pune la dispozitie comunitatilor de cercetatori si nu numai, un mediu de partajare, replicare si organizare a unor seturi largi de date. In cele mai multe cazuri datele sunt distribuite din punct de vedere geografic, astfel incat pentru preluarea sau stocarea datelor, utilizatorii trebuie sa acceseze diferite noduri din aceasta retea globala.
Aceasta combinatie intre volumul mare de date, distributia geografica a acestora si a utilizatorilor precum si a caracterului intensiv din punct de vedere computational al analizei acestor date, a dus la cerinte ce nu sunt satisfacute de sistemele actuale.

Pentru o utilizarea eficienta a resurselor din Grid trebuie asigurate : un acces sigur si rapid la datele partajate, mecanisme de descoperire, stocare, abstractizare, transformare, organizare, integrare, distribuire, publicare, securizare, refacere si intergorare ale acestor date.

In ceea ce priveste modurile de transfer al datelor, in sistemele Grid se disting urmatoarele tipuri: transfer paralel de inalta performanta, transfer de date distribuite, transfer partial de date, posibilitati de aplicare a unor functii de reducere si selectie asupra datelor transferate.

Transfer paralel de date in Grid

Transfer intre diferite puncte ale retelei ce utilizeaza mai multe canale TCP.
In retelele WAN, utilizarea de canale TCP multiple in paralel (chiar si intre doua puncte) poate imbunatati largimea de banda agregata.

Transfer de date distribuite

Transfer de date intre m noduri sursa si n noduri destinatie. Datele sunt distribuite(partionate) pe cele m noduri, utilizand sisteme de fisiere specializate (ex."Distributed-Parallel Storage System").

Transfer partial de date

Transfer de date ce suporta trasferarea de subseturi sau regiuni de date.
Acest tip de transfer, este in general utilizat in aplicatiile din domeniul cercetarii care necesita accesul la subseturi, relativ mici, din datele globale.

Transfer controlat de la distanta

Transfer ce se realizeaza intre doua noduri A si B fiind initiat si controlat de un utilizator aflat intr-un punct C.

3. 5 Probleme ridicate de monitorizarea resurselor in medii Grid

Intr-un mediu distribuit, dinamic, masuratorile trebuie efectuate permanent, fara pierderi, deoarece evenimentele sunt unice, irepetabile. Trebuiesc astfel inregistrate cat mai multe informatii despre starea componentelor sistemului pentru a putea mai tarziu identifica cauzele problemelor aparute sau pentru a estima performantele sistemului in viitor.

Problemele care pot aparea referitor la monitorizare propriu-zisa sunt:

- pierderea datelor de monitorizare. Cauzele pot fi variate: pierderea conexiunii prin care se transferau datele pentru a fi stocate, defectarea mediului de stocare, erori umane samd. Pentru a evita astfel de probleme ar trebui sa existe mai multe nivele de monitorizare si arhivare a datelor care sa permita recuperea informatiei pierdute la un anumit nivel. Pentru exemplificare putem lua cazul unui echipament de retea activ de tip router. El poate monitoriza activ transferurile de date care il implica, dar nu poate stoca aceste informatii pentru o perioada indelungata. Din acest motiv ar trebui ca informatiile sa fie culese de un serviciu de monitorizare care sa pastreze si un istoric al evenimentelor pe o perioada indelungata. Informatiile din mai multe astfel de sisteme ar trebui apoi combinate pentru a avea o vedere de ansamblu asupra sistemului. Pastrarea integritatii datelor este vitala pentru a putea lua o decizie corecta la un nivel superior de monitorizare, din acest motiv trebuiesc prevazute mecanisme de revenire din starea de eroare prin care sa se recupereze cat mai multe informatii utile cu putinta. Tot pe baza cazului anterior considerat, daca se pierde legatura intre router si sistemul de monitorizare trebuie ca sistemul sa semnalizeze cat mai curand acest lucru iar cand legatura devine din nou functionala sa aduca cat mai multe din informatiile stocate temporar in router. La fel si in cazul pierderii legaturii intre sistemele de monitorizare, nivelul superior trebuie sa poata reface istoricul global prin combinarea istoricului din nivelele inferioare de monitorizare. Ceea ce ne duce la urmatoarea problema a monitorizarii;
- transferul datelor. Monitorizarea activa a sistemelor poate duce la crearea unor cantitati foarte mari de informatii care trebuiesc transferate continuu, stocate si analizate pentru a avea o vedere globala a sistemului. In cazul unui sistem mare, cu multi parametrii de functionare monitorizati, aceste cantitati pot fi de ordinul zecilor sau sutelor de kb pe secunda. Pentru a putea monitoriza continuu un astfel de sistem trebuie sa fie rezervate permanent latimile de banda necesare;
- Stocarea datelor de monitorizare este o alta problema importanta. Trebuie gasit un echilibru intre nevoia de a avea cat mai multi parametrii, rezolutia datelor, performanta sistemelor de arhivare si analiza a lor samd. Nu exista o solutie universala pentru toate arhitecturile ci pentru fiecare caz in parte se stabilesc valori limita pentru acesti parametrii ai monitorizarii iar sistemul de monitorizare trebuie sa fie suficient de flexibil incat sa se poata adapta oricaror conditii (numar de resurse / parametrii / rezolutie a datelor). Alta diferenta majora provine din tipul datelor de monitorizare. De exemplu in cazul analizei traficului de retea avem nevoie de urmarirea continua a parametrilor, cu rezolutie cat mai mare a datelor. Dar daca monitorizam un Grid de procesare de date trebuiesc urmarite doar evenimentele din sistem (momentul inceperii unui job, momentul

terminarii lui, timpul de procesor folosit, cantitatea de date transferata, spatiul ocupat pe disc etc).

Procedura de monitorizare distribuita a resurselor din infrastructura GRID

4.1 Arhitectura generală a sistemului

4.1.1 Condiții de proiectare impuse sistemului

Pornind de la faptul că sistemul de monitorizare este în principal destinat monitorizării de noduri din *Grid* asupra proiectării s-au impus următoarele condiții:

- nodurile din *Grid, testbed*-uri se pot afla oriunde în Internet
- traficul peste WAN trebuie minimizat având în vedere că de obicei comunicația se face pe Internet, ceea ce presupune o lățime de bandă posibil cu un orin de mărime sub cel din LAN
- datele trebuie exportate sub formă de servicii
- clienții pot descoperi ce servicii sunt disponibile şi pot fi înstiințați de apariția unor noi servicii şi/sau dispariția unora existente
- sistemul trebuie să suporte diverse tehnologii de stocare a datelor, existând posibilitatea folosirii de soluții diferite de stocare în diferite centre de monitorizare, cu mențiunea că unitatea de stocare este unică într-un centru de monitorizare
- eventuala încărcare dinamică a claselor ce realizează persistența datelor, menționarea parametrilor specifici unei anumite soluții de stocare la Runtime, de exemplu unui URL(Uniform Resource Locator) ce poate referi fie o resursa(fişier sau director, etc) locală fie una externă(WebServer, FTP, etc)
- posibilitatea configurării dinamice a ce anume se monitorizează; acasta presupune existența unor module de monitorizare
- să permită folosirea unor sisteme deja existente de monitorizare, acolo unde este posibil
- interogarea datelor să se poată face într-un mod inteligent
- să existe posibilitatea interogării atât retroactive pentru datele colectate
- clientul poate primi date din prezent oferindu-i-se şi posibilitatea filtrării acestora, o aşa zisă *filtrare activă*
- suport WSDL/SOAP pentru clienți non-Java
- sistemul va fi tolerant la defecte, în cazul în care în timpul monitorizării unul din noduri nu mai este disponibil funcționarea sistemului nu trebuie să fie perturbată
- să permită o monitorizare în paralel a mai multor noduri
- posibilitea încărcării dinamice, ca şi în cazul solțiilor de stocare, a unor module de monitorizare
- interfațare cu **MDS** Metacomputing Directory Service, un modul din Glonbus care gestionează diverse informații despre *Grid*

Având în vedere cele enunţate anterior, în principal a condiţiilor referitoare la localizarea testbed-urilor şi a minimizării traficuluipeste WAN s-a pornit de la o arhitectură ierarhică cu centre locale în fiecare punct(de obicei testbed) de monitorizare. Aceste centre locale au fost denumite **FarmMonitor** şi sunt coordonate de către **RCMonitor**(Regional Center).

4.1.2 Componentele de bază ale arhitecturii

Noduri fizice

Pot fi simple calculatoare(host), DB Servere, Switch-uri, Router-e, etc; in principiu orice nod dintr-o retea ce poate fi monitorizat cu un *Monitoring Module(modul de monitorizare)* Pe aceste noduri pot rula agenti SNMP, servere de rsh, in general orice poate "discuta" un acelasi protocol cu modulul pereche de pe FarmMonitor.

Module de monitorizare (Monitoring Modules)

Principala componentă care introduce date în sistem, translatându-le în acelaşi timp şi în formatul intern înţeles de aplicaţie, este modulul de monitorizare, care este o clasă Java ce poate fi încărcată dinamic, locaţia putând fi specificată printr-un URL. În principiu datele sunt obţinute printr-o parsare specifică fiecărui modul, datele fiind întoarse într-un format standard pentru orice modul de monitorizare. În pricipiu pe lângă datele numerice, ce reprezintă efectiv valorile monitorizate, ce sunt parsate de către modul se mai adaugă si informaţii referitoare la nod, cum ar fi numele acestuia, cluster-ul şi ferma din care face parte. De obicei aceste module sunt rulate la intervale regulate de timp, cu ajutorul unei cozi de priorităţi. Pot în pricipal extrage date prin SNMP, rula scripturi prin rsh, sau ssh, acolo unde este posibil acest lucru, etc. Pentru a reuşi menţinerea sistemelor mari şi foarte mari la zi şi pentru a putea aduce eventuale noi funcţionalităţi celor existente este de dorit ca aceste module să poată di încărcate şi instanţiate dinamic de la anumite URL-uri, eventual prestabilite. Iată câteva exemple de astfel de module:

- Un modul ce foloseşte librărie(în cazul Java un pachet) SNMP pentru a culege diverse informaţii referitoare la I/O, încărcarea unui procesor. Mai întâi se face un request, după care se aşteaptă rezultatul.
- Un modul ce foloseşte comandă shell pe sistemul local, de ex */bin/*cat pentru a parsa diferite informaţii conţinute în */proc*, pentru sisteme de tip Unix, Linux, cum ar fi memoria liberă, etc.
- Un modul care realizează cam aceeaşi funcţie cu cel anterior doar că foloseşte de exemplu *rsh* sau *ssh* pentru execuţia scriptului, sau a comenzii.

Farm Monitor

Această componentă este responsabilă pentru configurarea şi monitorizarea unui cluster de noduri de reţea. El este cel care poate încărca dinamic orice modul pe baza unor anumite URL-uri, sau sisteme de fişiere distribuite.

Configuarea unui astfel de Farm Monitor este realizată de către RC Monitor(Regional Center Monitor), componentă ce va fi descrisă mai jos în proiect.
Un "*pool*" de fire de execuție(thread-uri) este folosit de către Farm Monitor pentru a rula modulele de monitorizare pentru fiecare nod. Această abordare, folosirea unui "*pool*" de fire de execuție, duce la o folosire eficientă a resurselor. Justficarea ar fi următoarea: perioada de rulare a unui modul de monitorizare este de obicei scurtă, în comparație cu perioada cât acesta este liber, nefolosit. Având în vedere că sistemul va fi folosit pentru monitorizări pe un număr semnificativ de noduri, adăugând şi faptul că fiecărui nod îi pot fi destinate mai multe module de monitorizare, ar fi total ineficient pornirea unui thread pentru fiecare modul. În plus, dacă se întâmplă ca o anumită monitorizare să eşueze celelalte nu sunt afectate. În schemă mai intervine un thread ce face periodic verificări şi opreşte thread-urile blocate din cauza unor erori.

Cluster

Este o colecție logică de noduri fizice. Se mai numesc şi unități funcționale, deoarece organizarea de noduri fizice în astfel de grupări se face pe un fir logic, după funcțiile pe care le îndeplinesc în întreaga structură monitorizată. A fost definit în principal pentru a fi posibilă extragerea de informații referitoare la grupări logice de noduri din rețea. Se pot grupa de exemplu echipamentele de rețea într-un cluster separat față de nodurile care procesează date în *Grid*.

Regional Center Monitor (RC Monitor)

Este unitatea centrală care coordonează unitățile de tip Farm Monitor şi extrage informații de la acestea. Acesta este cel care trimite configurațiile (modulele pentru fiecare nod, parametrii, frecvența de măsurare) către unitațile de monitorizare creând conexiuni peer – to – peer cu fiecare dintre ele.

Există posibilitatea pornirii şi opririi de unități Farm Monitor. Aşadar structura logică de până acum ar fi următoarea:

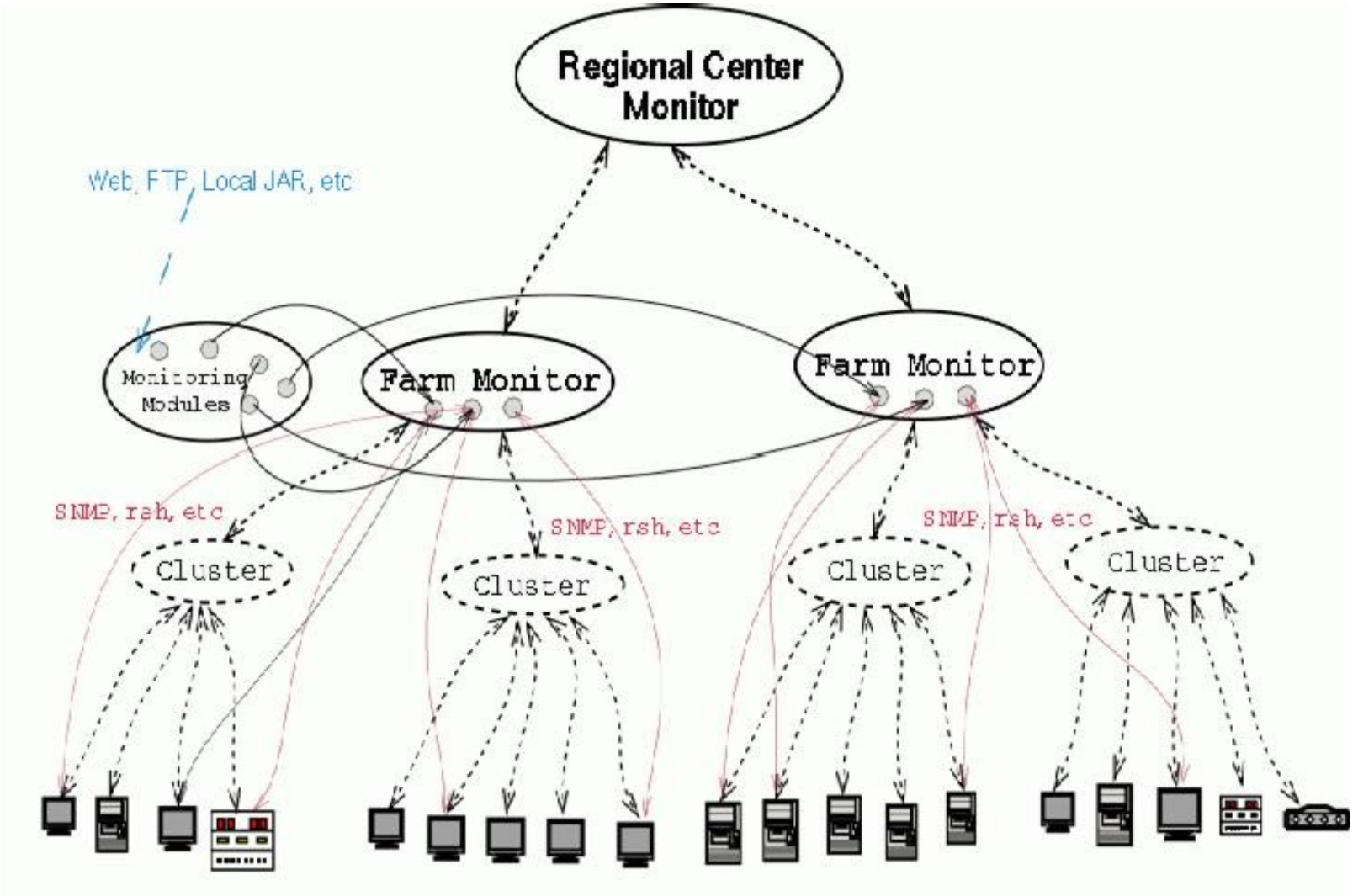
RC conține o listă de "ferme" (unități de tip Farm Monitor)

Fiecare fermă are o listă de unități functionale – este vorba despr Cluster-e , de exemplu servere de Baze de Date, WebServere, Routere, etc

Fiecare unitate functională (Cluster) are o listă de noduri fizice (calculatoare, switch-uri, etc)

Fiecare system (Nod) are asociată o listă de module şi o frecvență de execuție a acestora

Fiecare modul de monitorizare poate obține o serie de parametrii (valori monitorizate) de la un sistem.


Regional Center Monitor
Web, FTP, Local JAR, etc
Monitoring Modules
Farm Monitor
Farm Monitor
SNMP, rsh, etc
SNMP, rsh, etc
SNMP, rsh, etc
Cluster
Cluster
Cluster
Cluster

4.2 Componente critice pentru realizarea procedurii de monitorizare

4.2.1 SNMP

4.2.1.1 SNMP şi administrarea reţelei

Administrarea unor reţele complexe, care folosesc tehnologii Internet, se poate face folosind un protocol SNMP (Simple Network Management Protocol). SNMP este folosit şi ca termen generic pentru standardele care se referă la administrarea reţelelor folosind tehnologii Internet. SNMP rămâne, totuşi, doar o componentă a setului de standarde referitoare la administrarea unor astfel de reţele. Cadrul oferit pentru administrare - Internet-standard Network Management Framework - mai include, pe lângă SNMP, şi SMI (Structure for Management Information) şi diferitele baze de date de administrare MIB (Management Information Base).

IAB (Internet Activities Board) controlează elaborarea de standarde pentru Internet. Aşa cum se specifică în RFC-ul (Request For Comments) corespunzător, etapele standardizării sunt: cea de propunere de standard, apoi un proiect de standard (draft) şi, la urmă, cel de standard. SNMP a trecut de procesul de standardizare, mai întâi sub forma SNMPv1. Datorită evoluţiei rapide a domeniului, a devenit imperios necesară o nouă versiune SNMPv2. Cele care urmează se ocupă de cele două versiuni SNMP şi modul în care sunt utilizate pentru realizarea administrării reţelelor.

4.2.1.2 Protocolul SNMP

Internet-ul a crescut rapid şi există deja zeci de mii de reţele conectate şi sute de mii de echipamente de interconectare, (rutere, comutatoare), care asigură interconectarea a milioane de echipamente, de la chioşcuri de informare şi terminale bancare ATM, la sisteme mari sau complexe, ciorchini de servere (cluster). Pe aceste echipamente rulează diferite sisteme de operare şi aplicaţii care provin de la un număr mare de furnizori. Complexitatea foarte mare a acestei reţele mondiale nu ar permite o funcţionare acceptabila fără ca să se respecte o serie de standarde şi protocoale.
Acestea sunt utilizate şi pentru reţelele interne ale diferitelor instituţii, organizaţii sau firme - intranet-urile - unde evoluţia tehnologică a dus la existenţa de echipamente şi software de la o mulţime de furnizori.

Conlucrarea echipamentelor diferite nu este posibilă fără respectarea standardelor, chiar dacă reţeaua nu este conectată la Internet. În plus, fără administrare corespunzătoare, aceste reţele nu ar funcţiona acceptabil şi nu ar fi posibilă extinderea, scalabilitatea, fără a provoca perturbaţii cu efecte inestimabile.

TCP/IP s-a impus ca protocol pentru reţelele care sunt conectate pe Internet. Încă din februarie 1988, s-a format un comitet ad-hoc la cererea IAB pentru a evalua posibilităţile de administrare a reţelelor care folosesc tehnologii Internet. Ca rezultat, s-a adoptat un protocol SNMP (Simple Network Management Protocol) şi un cadru (Framework) pentru utilizarea acestuia pe Internet. SNMP se bazează pe SGMP (Simple Gateway Management Protocol), un protocol utilizat anterior la administrarea unor reţele regionale legate la Internet. În 1993, SNMP şi Internet-standard Network Management

Framework au fost actualizate, pentru o mai bună adecvare la noile cerințe. A luat astfel naştere SNMPv2.

Modelul de administrare pe care se bazează SNMP este format din staţii de administrare şi elemente de rețea.

- Staţiile de administrare de rețea (manager) asigură execuţia protocolului de administrare a rețelei şi a aplicaţiilor de administrare care urmăresc şi controlează elementele rețelei.
- Elementele din rețea, numite şi elemente administrate (managed resources), sunt echipamente de tipul sistemelor gazdă (host), punți, rutere, comutatoare, echipamente care au agenți care îndeplinesc funcțiile de administrare solicitate de către staţiile de administrare sau, eventual, semnalează diferite evenimente.

SNMP asigură modalitatea de comunicare între staţiile de administrare şi elementele de rețea. Este conceput ca un protocol simplu, care dă posibilitate administratorului rețelei să inspecteze sau să modifice variabile la un element de rețea, de la o staţie aflată la distanță. Implementarea SNMP se remarcă prin relativa simplitate şi prin faptul că necesită resurse reduse din partea rețelei.

Strategia urmărită la implementarea SNMP este ca toată administrarea să poată fi făcută la staţia de administrare, cu excepţia unor situaţii rare. Staţia de administrare interoghează (poll) elementele de rețea pentru a obține informaţii sau pentru a modifica variabile la elementul de rețea. Elementul de re-țea va iniția comunicaţia cu staţia de administrare (prin Trap) doar in situaţii deosebite. Mesajele Trap nesolicitate pot fi trimise de la un element de rețea pentru informare sau pentru corectarea temporizării la interogări de la staţia de administrare. Numărul mesajelor Trap este însă limitat pentru a se evita traficul intens şi solicitările rețelei pentru activitatea de administrare.

4.2.1.3 Simplitatea protocolului SNMP

IAB a considerat ca fiind deosebit de importantă păstrarea simplităţii protocolului, pentru a nu se genera trafic excesiv de administrare pe rețea. În acest scop s-au specificat criterii pentru proiectare pentru cei care implementează SNMP.

- Toate acţiunile necesare administrării trebuie să fie implementate sub forma unor operaţii de scriere/citire la/de la variabile din elementele de rețea. Ca efect, numărul funcţiilor este limitat, fiind nevoie doar de operaţii care atribuie valoare unei variabile şi de operaţii care examinează valoarea unei variabile.
- Nu trebuie să existe comenzi imperative, care să impună elementului de rețea să execute o anumită acţiune. Operaţia trebuie implementată sub forma atribuirii unei valori la o variabilă a elementului şi acţiunea va fi determinată prin inspectarea variabilei, pe baza valorii citite.
- Trebuie limitat numărul mesajelor ne-solicitate (Trap sau evenimente). Astfel se va limita traficul şi staţia de administrare va putea păstra controlul, nefiind perturbată de traficul excesiv generat de mesaje de semnalare a unor situaţii deosebite.

- Trebuie utilizat un protocol simplu de transport, protocol fără conexiune - de preferință UDP (User Datagram Protocol) - pentru a păstra la minimum complexitatea agentului de administrare. Astfel, fiecare mesaj se va putea implementa sub forma unei datagrame simple pentru transport.

Elementele administrate nu vor fi nevoite să păstreze informații de stare (de exemplu Blocat, în aşteptarea unui răspuns). Datorită acestei simplități, SNMP poate să facă față şi în situații când există multe pachete pierdute şi se generează retransmisii în cascadă.

4.2.1.4 RFC-uri SNMPv1

Inițial, definițiile pentru SNMPv1 au fost stabilite în trei documente, dar în timp aceste documente au fost completate de o serie de alte documente.

Definirea protocolului s-a făcut în RFC 1157, regulile de descriere a variabilelor pentru administrare în RFC 1155 şi setul de variabile de bază în RFC 1156. Mai târziu, regulile de descriere au fost extinse prin două documente, RFC 1212 şi RFC 1215. S-a extins şi numărul variabilelor de bază (core variables) şi definiția originală, numită MIB-I, a fost înlocuita cu cea MIB-II.

- **RFC 1155 - SMI - Structura informațiilor de administrare**
  Acest RFC specifică structurile şi schema de identificare pentru definirea informațiilor de administrare. Această definiție include descrierea unui model al obiectului de informație şi un set de tipuri generice care descriu informația de administrare
- **RFC 1156 - MIB-I - baza de informații de administrare**
  Acest RFC oferă specificația obiectelor administrate, inclusiv modul de denumire, sintaxă, definirea, accesul şi starea. Acest MIB oferea descrierea pentru 114 obiecte, prin reguli prezentate de documentul SMI. MIB-I nu mai este actual.
- **RFC 1157 SNMP - Protocol simplu de administrare rețea**
  Acest RFC prezintă protocolul prin care se face comunicarea dintre administratori şi agenți.
- **RFC 1212 - Definiții concise MIB**
  Acest RFC prezintă modul de redactare al unui MIB concis şi descriptiv. Acesta este un format nou de scriere a definițiilor MIB, care permite eliminarea informațiilor redundante. Acest RFC extinde definițiile SMI.
- **RFC 1213 - MIB-II - Baza de date cu informații de administrare**
  Acest RFC este o revizie pentru RFC 1156, care extinde vechile definiții de la 114 la 171 variabile. Aceste variabile sunt doar extensii MIB-I şi se asigură compatibilitatea înapoi. MIB-I devine perimat prin acest RFC.
- **RFC -1215 - Definiții concise pentru TRAP**
  Acest RFC prezintă modul în care trebuie scrise specificațiile Trap SNMP.

4.2.1.5 RFC-uri SNMPv2

Aceste specificaţii sunt frecvent utilizate azi, dar utilizarea lor trebuie făcută cu respectarea SNMPv1 care este standard aprobat şi ţinând seama de orientările din SNMPv3, în curs de specificare. Trebuie amintite următoarele:

- Introducere la SNMPv2 care oferă o prezentare pentru Internet Standard Network Management Framework, adică un cadru de administrare
- SMI pentru SNMPv2 defineşte un sub-set din ASN.1 de la OSI, pentru a putea fi utilizat la definirea obiectelor MIB
- Convenţii textuale SNMPv2 propune un set nou de tipuri similare celor definite de SMI, cu aceeaşi sintaxă dar cu semantica mai precisă.
- Declaraţii de conformitate pentru SNMPv2 care definesc cerinţele minime de implementare pentru un grup de obiecte ale unui modul MIB. Se definesc şi modalităţile de documentare pentru nivelul actual de implementare care s-a realizat în agent.
- Model de administrare pentru SNMPv2 prezintă modul de încapsulare a mesajelor SNMPv2 pentru păstrarea integrităţii datelor, autentificarea originii, asigurarea intimităţii etc.
- Protocoale de securitate pentru SNMPv2 este un RFC care prezintă modul de utilizare MD5 (message digest algorithm) pentru asigurarea integrităţii şi autentificare, în cazul în care se utilizează SNMPv2 cu securitate. Se prezintă şi algoritmul DES pentru criptare.
- MIB-uri pentru participanţi (party) la SNMPv2 descriu obiectele administrate definind proprietăţi asociate participanţilor la SNMPv2, contextul şi politici de control acces.
- Operaţii protocol pentru SNMPv2 descrie unităţile PDU (Protocol Data Units) pentru SNMPv2, care transmit sau recepţionează mesaje.
- Mapări de transport SNMPv2 este un RFC care defineşte protocoalele de nivel transport permise şi prin care se transmit mesajele SNMPv2. Maparea preferata este UDP, ca şi în cazul SNMPv1.
- MIB pentru SNMPv2 defineşte obiectele administrate care descriu operaţiile executabile de un agent de administrare sau staţie de administrare.
- MIB manager la manager. La SNMPv2 o staţie poate avea atât rol de manager cât şi de agent, asigurând schimbul de informaţii de administrare cu alte staţii de administrare. Acest RFC defineşte obiectele administrate care specifică operaţiile executate de o entitate SNMPv2 care participă la administrare atât cu rol de staţie de administrare, cât şi ca agent.
- Coexistenţa SNMPv1 şi SNMPv2 este descrisă de un RFC în termenii unor module MIB, declaraţii de conformitate şi de capacităţi (capabilities).

4.2.1.6 Arhitectura SNMP

Grupul de lucru care a realizat arhitectura SNMP a avut în permanență în vedere asigurarea simplității protocolului printr-o structură simplă a arhitecturii pentru administrare. Administrarea este organizată astfel încât complexitatea se află la stația de administrare şi implementarea elementelor de administrare, a agenților, este simplă fără să presupună activități deosebite de executat.

Prin plasarea complexității şi a responsabilităților în administrare la stația de administrare s-a asigurat calea unei dezvoltări simple, pentru o perioadă îndelungată. Aplicațiile de administrare pot astfel îndeplini sarcini din ce în ce mai complexe şi pot duce la eliminarea necesității intervențiilor umane, în multe situații. De asemenea, aplicațiile pot fi izolate de detaliile specifice pentru protocolul de administrare şi protocolul de comunicație. Aplicațiile se pot preocupa astfel de funcțiile care trebuie asigurate şi nu sunt legate de alte detalii de protocol.

Abordarea minimalistă pentru partea elementului administrat permite realizarea administrării pentru o mare varietate de echipamente, fără a fi nevoie de modificarea acestora, în mod simplu şi eficient. Nu vor fi necesare resurse deosebite de partea agentului.
Administrarea rețelei folosind SNMP impune existența mai multor elemente care conlucrează. Acestea se împart în elemente administrate (agenți) şi stații de administrare (administratori) şi trebuie asigurată comunicarea dintre acestea de către protocol (SNMP).

- Elementele administrate - Agenți, sunt elemente din rețea ca sisteme gazdă, punți, rutere, comutatoare, repetoare multi-port, servere de terminale şi alte echipamente inteligente, pe care se poate executa softul de agent de administrare. Prin aceşti agenți stațiile de administrare pot comunica cu elementele administrate folosind SNMP. Agenții realizează funcțiile de administrare solicitate de stația de administrare.
- Stațiile de administrare - Administratori execută aplicațiile de administrare care urmăresc şi controlează elementele administrate din rețea. Stația de administrare permite raportarea către administratorul uman printr-o interfață utilizator.

- Baza de informatii de administrare – MIB consta dintr-o colectie de obiecte administrate de pe un element administrat. Este o structură de informații cu privire la elementul administrat, un echipament din rețea, o tabelă de rutare IP, o adresă în sub-nivelul MAC, etc. Obiectele definesc diferitele caracteristici ale unui echipament administrat.

- Protocolul simplu de administrare rețea - SNMP este un protocol simplu cerere/răspuns care permite mişcarea informațiilor de administrare între stația de administrare şi agentul aflat pe elementul administrat. Protocolul nu conține definițiile pentru obiectele care se pot administra. Astfel, SNMP poate fi utilizat cu orice variabilă de administrare care poate fi inspectată sau modificată.

- Autentificarea este o modalitate de asigurare a securității prin care agentul SNMP validează cererea provenită de la o anumită stație de administrare, înainte de a transmite răspunsul la cerere. Autentificarea este domeniul în care există diferențe mari între versiunile SNMP.

4.2.1.7 Agenți SNMP

Aplicațiile SNMP pot urmări şi controla elementele de rețea prin utilizarea agenților. Agenții poartă răspunderea îndeplinirii funcțiilor de administrare pentru elementul administrat, prin îndeplinirea cererilor primite de la stațiile de administrare. Agenții se află, deci, pe elementele de rețea şi au acces la variabilele MIB.

Agenții sunt sub-sisteme simple din elementele de rețea şi de multe ori sunt pasive, lucrând doar conform dispozițiilor primite de la aplicația de administrare. Doar în cazul apariției unor condiții de eroare bine definite va putea să preia un agent inițiativa şi să acționeze. Aceste evenimente declanşează capcane (traps) şi utilizarea lor este limitată.

Agenții administrează doar un set specific de resurse existente la un element dat de rețea. Aceste resurse sunt specificate de obiecte MIB. Această focalizare pe scopuri restrânse face ca agentul să poată fi păstrat relativ simplu şi să fie uşor de implementat. O implementare tipica a unui agent este formata din patru componente.

- Transport/Legătură - Asigură transmisia (emisia şi recepția) datagramelor pe rețea. De obicei acest serviciu există deja, fiind realizat sub formă generală ca serviciu pentru alte procese existente pe elementul de rețea.
- Motor SNMP - Implementează protocolul SNMP şi răspunde de schimbul de mesaje de la egal la egal dintre manager şi agent, de codificarea/decodificarea cererilor sau răspunsurilor într-o formă neutră față de platformă.
- Instrumentar - oferă acces pentru protocolul de administrare la variabilele agentului, care prezintă interes. Acest lucru este realizat, de regulă, printr-un mecanism intern de comunicare, care permite accesul la structurile de date de pe echipament, structuri care pot fi eventual manipulate la cererea protocolului de administrare.
- Profilul de administrare - este format dintr-un set de reguli care definesc accesul. Fiecare obiect din profil are specificat un mod de acces, care pate fi doar în citire, citire/ scriere, sau inaccesibil.

4.2.1.8 Agenți de proximitate

Agenții de proximitate (proxy) oferă funcționalități performante pentru SNMP. Ei îndeplinesc rolul de translator între SNMP şi sistemele de administrare proprietar, de la diferite firme. În acest rol, agenții proxy SNMP primesc directivele de administrare de la procesul de administrare, traduc directivele în operații proprietar, adună informații şi răspund către procesul de administrare prin mesaje şi capcane standard SNMP.

Agenții de proximitate pot avea şi rolul de zid protector pentru elemente critice ale rețelei. În acest caz agentul de proximitate se află pe un alt element din rețea, element diferit de cel critic şi poate administra un cache în numele elementului critic. Agentul de proximitate păstrează în cache cererile referitoare la elementul critic, eliberându-l de sarcina tratării cererilor în număr mare venite de la multe stații de administrare. Acest

lucru permite ca elementul critic să comunice doar prin intermediul agentului de proximitate.

Există o serie de motive pentru care se implementează agenții de proximitate:

- Resursele de memorie sau de procesare ale elementului din rețea sunt limitate şi nu poate asigura suport pentru agentul necesar administrării.
- Protocolul de acces pentru administrare a elementului nu este suportat de stația de administrare, deci este nevoie de translatare.
- Elementul de rețea are cerințe deosebite în ceea ce priveşte securitatea.
- Protocolul de transport utilizat nu oferă o cale directa intre manager şi agent, din cauza de rutere, de exemplu.

4.2.1.9 Stația de administrare SNMP

Un administrator SNMP este o colecție de aplicații şi baze de date care permit urmărirea şi controlul unui grup de agenți SNMP. Administratorii pot determina agenții individuali sa furnizeze informații pentru administrare sau să modifice modul de funcționare pentru un element dat.

Administratorii au o complexitate mare față de agenți, ei pot lua informații de la un grup agenți şi apoi pot trimite directive pentru coordonarea funcționării grupului de agenți.

Implementările tipice de administrator au în general cinci componente:

- Interfața utilizator - permite administratorului uman urmărirea rețelei, introducerea unor comenzi pentru administrare şi recepționarea de răspunsuri la cerere sau ne-solicitate.
- Aplicația de administrare - asigură analiza informațiilor de administrare rețea, primite de la agenți.
- Baza de date - conține toate datele privind denumirile, configurațiile, performanțele, topologia şi datele provenite din auditare. Trebuie să remarcm diferența dintre MIB şi alte baze de date. MIB defineşte variabilele de interes, existente la toate sistemele administrate care formează rețeaua.

Celelalte baze de date obişnuite oferă posibilități de stocare pentru instanță ale obiectelor MIB sau alte tipuri de date colectate şi folosite pentru funcțiile de administrare. Putem avea:

- Baza de date MIB;
- Baza de date cu elemente de rețea;
- Baze de date ale aplicațiilor de administrare, ca cele de mapare, de jurnale evenimente sau jurnale de urmărire.
- Motor SNMP - procesul care implementează SNMP, asigură schimbul de mesaje SNMP şi permite sistemului să acceseze de la distanță informațiile de administrare aflate pe elementele administrate din rețea.
- Transport/Legătură - oferă accesul la căile de comunicații implementate la nivelurile de dedesubt.

4.2.1.10 Baza de date de informații de administrare - MIB

MIB este o colecție structurată de informații despre toate obiectele controlate de un anumit agent sau manager. Este organizat sub forma unui tabel foarte simplu şi oferă acces uşor şi eficient la aproape orice sistem.
RFC 1213 defineşte MIB-2, dar mai există o serie de RFC-uri suplimentare pentru alte MIB-uri standard sau definite de firme.

- Obiectele de administrare sunt definițiile resurselor reale administrate în mediul SNMP. Există peste 1000 de obiecte definite şi înregistrate ca fiind standard Internet MIB. Nu toate obiectele sunt însă adecvate pentru orice element de rețea. De exemplu, obiectele MIB pentru punți sunt adecvate doar pentru bridge-uri.
- Grupuri de obiecte MIB sunt colecții de obiecte înrudite SNMP şi gruparea acestora uşurează funcțiile de administrare. Nu toate grupurile de variabile definite sunt obligatorii pentru toate componentele Internet. Este însă obligatoriu să existe suport pentru toate variabilele unui grup, dacă există suport pentru un element dintr-un grup. Este de aşteptat să apară grupe MIB şi variabile noi, pe măsură ce necesitățile de administrare cresc şi Internet-ul continuă să se dezvolte. Lista de grupuri de MIB-uri de bază incluse în MIB-II cuprinde:
    - Grupul sistem
    - Grupul interfața
    - Grupul de translatare a adreselor
    - Grupul IP
    - Grupul ICMP
    - Grupul TCP
    - Grupul UDP
    - Grupul EGP
    - Transmisia
    - SNMP

4.2.1.11 Unități de bază pentru date-protocol SNMP -PDU

În RFC 1157, referitor la operațiile SNMP şi SNMPv2, se specifică faptul că se va utiliza un număr minim de tipuri de mesaje sau PDU (Protocol Data Units) pentru a se oferi funcționalități de administrare pentru rețele şi sisteme diverse.
SNMP utilizează servicii de transport fără conexiune UDP (User Datagram Protocol) pentru a transfere PDU-uri între manageri şi agenți.

Fiind un serviciu de transport fără conexiune, UDP reflectă natura tranzacțională a interacțiunilor de administrare. Fiecare mesaj SNMP este conținut complet într-o singură datagramă, pentru transport. Managerii pot trimite mai multe cereri către un agent şi pot vedea fiecare răspuns primit pentru a restabili ordinea, dacă au sosit într-o altă ordine. Un contor de timp (timer) simplu poate să detecteze cererile care nu au primit răspuns, pentru a fi transmise din nou.

Există următoarele tipuri de PDU, pentru diferitele funcţii:

- **GetRequest** - acest tip de PDU cere agentului să returneze valorile atributelor, pentru lista specificată de obiecte administrate. Staţia de administrare trimite către agent câte o cerere o dată, cu lista de denumiri pentru fiecare obiect de care este interesat. Ca răspuns, agentul trimite răspunsul în care se indică succesul sau eşecul cererii. Dacă cererea s-a derulat cu succes, mesajul conţine şi valorile pentru toate obiectele solicitate.
- **GetNextRequest** - acest tip de PDU permite parcurgerea unei tabele de obiecte. Deoarece atributele obiectelor sunt păstrate în ordine lexicografică (similară cu sistemul Dewey pentru biblioteci), rezultatul unei cereri PDU GetNextRequest anterioare poate fi folosit ca argument în următorul GetNextRequest. Astfel, administratorul poate să parcurgă tabele de lungime variabilă, până ce extrage informaţiile din toate rândurile tabelei. Acest PDU nu este limitat la citirea rândurilor din tabele, ci poate fi utilizat pentru regăsirea tuturor informaţiilor de administrare, disponibile la un agent.
- **SetRequest** - acest tip este folosit de staţia de administrare pentru a cere unui agent să stabilească (Set) valorile pentru atribute pentru obiectele selectate. Staţia de administrare trimite atât lista cu numele obiectelor, cât şi valorile.
- **GetResponse** - este un tip de PDU folosit de agentul SNMP pentru a răspunde la o cerere GetRequest, GetNextRequest sau SetRequest.
- **Trap** este un PDU trimis de agent pentru raporta unele condiţii sau schimbări de stare către procesul de administrare. SMNP asigură suportul pentru un număr limitat de Trap-uri care pot să vină de la elementele din reţea.

4.2.1.12 Mediul de administrare SNMP

Între cele două versiuni SNMP există diferenţe importante.

4.2.1.12.1 SNMPv1

Din motive de securitate, agentul SNMP verifică cererile, înainte de a răspunde la ele. Acest lucru împiedică administratorii ne-autorizaţi să vadă sau să schimbe configuraţia la agent. Detaliile procesului depind de implementarea specifică. Totuşi, toate implementările recurg la valoarea câmpului şirului de identificare a comunităţii: community string.

Community - În SNMP comunitatea este relaţia dintre un agent şi unul sau mai mulţi administratori. Toţi membrii unei comunităţi date dispun de aceleaşi drepturi de acces. Agentul poate fi configurat ca numai administratorii care fac parte dintr-o comunitate cunoscută să poată trimite cereri şi sa primească răspuns. O comunitate este identificată printr-un şir de octeţi. De obicei, şirul este format din caractere ASCII tipăribile. Toate schimburile de mesaje SNMP se compun dintr-un nume de comunitate şi un câmp de date care conţine operaţiile SNMP şi operanzii corespunzători.

Schema de autentificare SNMP permite, în acest moment, doar autentificare simplă. Numele de comunitate este plasat în fiecare cerere SNMP. Acesta nu este criptat sau codificat.

Dacă numele de comunitate din mesajul SNMP este cunoscut pentru agent, se consideră că administratorul este autentic şi se permite accesul permis membrilor comunității. Există două moduri de acces:

- Read-only : numai în citire
- Read-write: pentru citire şi scriere

4.2.1.12.2 SNMPv2

S-au constatat o serie de aspecte mai slabe la SNMPv1, mai ales la autentificare şi la controlul accesului. SNMPv2 abordează aceste aspecte sesizate de cei care lucrează cu administrarea rețelelor. În plus, s-au introdus şi mecanisme noi de criptare a datelor (pentru asigurarea intimității - Privacy) şi pentru administrarea securității.

Participanți SNMP (party) este conceptul de bază pentru modelul de securitate SNMPv2. Este un mediu în care agenții şi administratorii SNMPv2 pot să comunice între ei prin protocol SNMP, folosind un anumit serviciu pentru transport, ca UDP, existând posibilități de utilizare a serviciilor pentru autentificare şi intimitate. Fiecare administrator sau agent SNMP comunică ca şi participant (party) cu un alt administrator sau agent SNMPv2. Fiecare participant SNMPv2 constă din:

- O identitate de participant
- O locație logică din rețea, unde se execută participantul, cum ar fi portul UDP 161
- Un protocol de autentificare
- Un protocol de control al intimității poate asigura criptarea datelor

Pot coexista mai mulți participanți şi pot fi folosiți să comunice cu administratorul SNMP sau cu agenți. Fiecare agent sau administrator SNMPv2 comunică utilizând o identitate de participant distinctă. Acest lucru permite să se facă identificarea fără ambiguități pentru emițător şi receptor, la fiecare mesaj SNMPv2, rezultat care nu putea fi obținut folosind şirurile de comunitate din SNMPv1.

Entitate SNMPv2 - este un proces care generează mesaje SNMPv2 de genul GetRequest sau GetResponse. Aceasta entitate îndeplineşte rolul unei anumite comunități, atunci când comunică cu alte entități. Prin a acționa ca o anumită comunitate, entitatea SNMPv2 este limitată la sub-setul de operații definite pentru acea comunitate. Fiecare entitate SNMPv2 administrează trei baze de date locale. Prima conține lista tuturor participanților recunoscuți. A doua conține o colecție de obiecte administrate, accesibile de către entitatea SNMPv2; aceasta se numeşte baza de date Context SNMPv2. A treia conține drepturile de acces, permise pentru toți participanții SNMPv2 cunoscuți. Deci o entitate SNMPv2 este un proces de pe un element din rețea care generează mesaje SNMPv2.

Context SNMPv2 - Un context este o colecție de obiecte administrate accesibile pentru o entitate SNMPv2. Un context defineşte obiectele administrate în termeni de orizont (view) MIB.

Orizont MIB (view) - este un sub-set din toate instanțele tuturor tipurilor de obiecte definite în SMI. Fiecare orizont MIB se bazează pe una sau mai multe familii de sub-arbori ai orizontului, care pot fi incluşi sau excluşi din orizontul MIB. În cadrul fiecărei entități SNMPv2 există o tabelă administrată local care defineşte orizontul MIB asociat fiecărui context SNMPv2 care are referințe la obiectele locale administrate.

Politica de control acces - O politică de control acces se defineşte în termenii unor perechi de participanți SNMPv2 care pot să comunice între ei în cadrul contextului specificat, fiind limitați la operațiile care sunt autorizate să aibă loc între cei doi participanți, operații ca GetRequest sau GetNextRequest. Politica de control acces se aplică la entitatea SNMPv2 care recepționează mesaj SNMPv2.

Participanților SNMPv2 li se poate cere de către alți participanți să efectueze operații de administrare. În mod similar, participanții SNMPv2 pot cere altor participant să execute anumite operații de administrare prin trimiterea de comunicații de mesaje de administrare.

4.3 Sisteme de monitorizare

MRTG (http://www.mrtg.org/) : acronim pentru Multi Router Traffic Grapher, este un program open source scris in C si Perl care culege informatii din retea prin SNMP, stocheaza informatiile pe disc in fisiere de dimensiune fixa si genereaza la fiecare rulare grafice pentru fiecare din marimile masurate. Acest soft de monitorizare este foarte folosit, dar are cateva dezavantaje majore: graficele sunt pentru o singura marime masurata, cu maxim doua ipostaze ale sale (de exemplu grafice pentru traficul de intrare si cel de iesire de pe o interfata de retea, memoria RAM si swap ocupata samd), nu se pot compara deci mai multe marimi de acelasi tip; fisierele contin toate datele culese pana la un moment dat, pentru maxim un an, desi datele mai vechi nu sunt utile pentru analiza deoarece nu este aratata decat o medie a lor. La fiecare rulare se regenereaza imaginile pentru toate marimile masurate si pentru toate intervalele disponibile, ceea ce poate duce la incarcarea inutila a masinii mai ales daca aceste date nu sunt consultate frecvent. Lipseste un sistem de vizualizare a limitelor intre care s-a incadrat o marime observata pe un anumit interval de timp. Granularitatea masurarilor este fixa, datele sunt stocate la intervale de minim 5 minute, sau mai rar, dar nu se poate alege de exemplu crearea unui grafic cu rezolutie de 1 minut.

RRDtool : acronim pentru Round Robin Database. De asemenea open source, rezolva cateva din dezavantajele lui MRTG: elimina datele irelevante (mediaza valorile vechi) si poate arata limitele (min/max) intre care s-a incadrat o valoare masurata. De asemenea graficele sunt mult mai complexe, se poate alege o forma cat mai intuitiva dintre mai multe tipuri de grafice disponibile, sunt prezentate seriile de date cu culori diferite. Totusi intervalele sunt fixe si imaginile se genereaza la fiecare valoare primita. Important de retinut de aici a fost sistemul ales pentru stocarea datelor vechi: pentru un interval oarecare de timp (o luna, un an) se mentin atatea valori cate sunt suficiente pentru a afisa acel grafic (deci doar cateva zeci de valori). Astfel pentru desenarea graficelor nu vor mai trebui mediate toate valorile instantanee de-a lungul intregii perioade dorite (la MRTG de

exemplu se poate ajunge la 100000 de inregistrari pentru o marime masurata intr-un an), ci pur si simplu se deseneaza cele cateva zeci de valori care sunt reprezentative pentru acea perioada de timp.

AstroFlow (http://www.netsoft.co.za/) : sistem de monitorizare comercial orientat spre gestionarea latimii de banda.

Big Brother (http://www.bb4.com/) : monitorizeaza mai multe sisteme si tine loguri ale diferitelor evenimente (resetare calculator, legatura la internet intrerupta etc). Sistemul are o arhitectura client-server, poate trimite notificari in cazul anumitor evenimente, poate fi extins prin module proprietare.

Aceste programe sunt in general destinate monitorizarii unui numar mic de sisteme, cu numar mic, constant, de parametri. Folosirea lor este foarte punctuala, pentru sisteme izolate sau retele mici pot fi foarte utile, insa nu se pot folosi in sisteme de tip Grid. Exista insa si arhitecturi mai complexe, unele din ele bazate pe solutii enumerate mai sus, care ofera insa suport pentru modele de retea de dimensiuni mari. Cateva dintre acestea sunt:

Ganglia (http://ganglia.sourceforge.net/) : proiect open source care a fost pornit de catre Universitatea Berkeley, California. Este orientat in principal catre monitorizarea clusterelor si sistemelor de tip Grid. Protocolul de comunicatie este bazat pe multicast in interiorul clusterului iar intre clustere se foloseste o structura ierarhica de conexiuni punct-la-punct. Datele sunt transferate in format XML prin XDR, stocarea si analiza lor se face folosind RRDtool. Arhitectura este destul de flexibila, poate fi adaptata cam oricaror cerinte de monitorizare.

OpenNMS (http://www.opennms.org/) : proiect open source bazat pe platforma Linux. Pentru fiecare nod configurat pentru a fi monitorizat se incearca determinarea serviciilor care ruleaza pe el prin incercare directa de stabilire a unei conexiuni cu acel serviciu. OpenNMS se comporta ca un client normal pentru serviciul respectiv. Odata la 24 de ore se incearca din nou toate nodurile din configuratie pentru a vedea daca cumva exista un nou serviciu activat intre timp. Serviciile cunoscute se verifica la fiecare 5 minute, daca se detecteaza ca un serviciu nu ruleaza se scade intervalul de verificare la 30 de secunde. Daca serviciul nu ruleaza dupa 12 ore atunci se presupune ca problema nu se rezolva curand si intervalul este crescut la 10 minute. Dupa 5 zile in care un serviciu nu a raspuns acesta este scos din lista serviciilor active presupunandu-se ca a fost oprit definitiv. Ideea de baza aici este determinarea cat mai exacta a intervalului de timp in care serviciul nu a fost disponibil. Alta parte de monitorizare se face prin SNMP. Aici se pot configura diferite nivele de monitorizare, de exemplu pentru serverele importante se poate face monitorizare la minut iar pentru statiile de lucru se poate alege un interval de 5 minute de exemplu. Daca valorile receptionate depasesc anumite limite superioare sau inferioare se pot genera evenimente pentru a atentiona administratorul. Se pot astfel lua imediat masuri in cazul unui server care are incarcare prea mare, sau nu mai are spatiu pe

disc etc. Stocarea datelor se face folosind baza de date relationala open source PostgreSQL (http://www.postgresql.org/) iar afisarea se face folosind Tomcat (server de web open source, in acelasi timp si mediu de rulare pentru servlet-i si jsp-uri - http://jakarta.apache.org/tomcat/). Imaginile sunt generate cu RRDtool. Este o arhitectura buna pentru monitorizat reteaua unei firme, scuteste administratorul de sarcina monitorizarii serverelor, dar nu poate lua masuri cand se intampla ceva (de exemplu daca un server de web nu mai raspunde atunci sa incerce mai intai sa-l reporneasca) si nu poate masura valori si sa arate grafice pentru valorile stocate, este limitat la evenimente si notificari in urma evenimentelor.

HP OpenView (http://www.openview.com/) : solutie comerciala de monitorizare, ofera solutii complete pentru companii prin monitorizare oricaror echipamente de retea (servere de baze de date, servere de web, mail, echipamente de retea - switch-uri, router-e -, echipamente de stocare. Acest produs ofera numeroase facilitati: descoperirea automata a arhitecturii retelei (atat la nivel logic -3-, cat si la nivel fizic -2-), determinarea automata a cauzei unei defectiuni, analiza protocoalelor folosite in retea (IP, IPv6, OSPF, HSRP), afisarea corespunzatoare a VLAN-urilor. Este utilizat de cei mai mari furnizori de Internet din Statele Unite, HP comunica urmatoarele cifre de folosire: 135.000 de instalari, 19 milioane de utilizatori se folosesc de aceste servicii, 67% din furnizorii de Internet din SUA il folosesc. Totusi aceasta solutie are si un dezavantaj: fiecare client in parte inseamna o configurare specifica care poate dura destul de mult, de exemplu integrarea OpenView la Vodafone a inceput in Decembrie 1996 si a durat pana in Octombrie 1999.

Mai exista si alte solutii comerciale
al caror cost este in general ridicat, de exemplu: CiscoWorks, IBM Trivoli, Copmuter Associates Unicenter, Micromuse Netcool, OpenService NerveCenter. Acestea se implementeaza in general pentru fiecare client in parte diferite, fiecare din ele este orientata spre o anumita arhitectura si este nevoie de personal cu instruire speciala pentru a le folosi.
Dintre solutiile comerciale cu cost mai redus, dar si cu facilitati mai reduse:

Microsoft SMS (System Management Software) : se adreseaza in principiu monitorizarii de produse Microsoft, servere si statii de lucru pe care ruleaza sisteme de operare Windows bazate pe kernel NT. Nu exista posibilitatea monitorizarii prin SNMP a unor echipamente de retea cum ar fi switch-uri si router-e si nici a altor noduri de retea pe care ruleaza alte sisteme de operare.

LanWare NMS Platinum : functioneaza doar pe Windows si ofera suport SNMP, dar si monitorizare a unor aspecte specifice serverelor Microsoft (procese si servicii NT).

5. Concluzii

Putem afirma ca in cadrul etapei 6 a proiectului s-au realizat:

- **Implementarea MonALISA pe infrastructura GRID locala IFIN**
- **Conectarea la retelele informatice nationale si informationale**

Si s-au implementat urmatoarele obiective propuse:
Analiza si identificarea cerintelor unui sistem de monitorizare distribuit pentru intretinerea si evaluarea performantelor infrastructurii Grid.
Elaborarea unei proceduri eficiente de monitorizare distribuita a resurselor din infrastructura Grid.

Sistemul de monitorizare, este utilizat de comunitatea de cercetare din domeniul sistemelor distribuite de mare dimensiune. Dezvoltatorii de aplicatii distribuite folosesc sistemul de monitorizare pentru parametrizarea calitatii resurselor de comunicare si a resurselor disponibile in cadrul aplicatiei distribuite. Sistemul in varianta curenta, el fiind in permanenta dezvoltare, se foloseste in cadrul proiectelor dezvoltate la UPB, IFIN, ICI, INCAS, Univ Timisoara, Univ Cluj Napoca. De asemena se utilizeaza de catre CALTECH si CERN.
Sistemul este functional si este pus la dispozitia utilizatorilor la adresa
http://monalisa.caltech.edu/

Developers: developers@monalisa.cern.ch

- **Mihaela Toarta-Dediu (UPB)**
- **Corina Stratan (UPB)**
- **Catalin Cirstoiu (CERN)**
- **Costin Grigoras (UPB)**
- **Ramiro Voicu (CERN)**
- **Adrian Muraru (UPB)**
- **Ciprian Dobre (UPB)**
- **Lucian Musat (UPB)**
- **Alexandru Costan (UPB)**
- **Alexandru Herisanu (UPB)**
- **Iosif Legrand (CALTECH)**

Rezultatele obtinute au fost comunicate la conferinta internationala **Eurocon 2007, 9-12 September, Warsaw, Poland:**

TOWARDS A COMMUNICATION FRAMEWORK BASED ON BALANCED MESSAGE FLOWDISTRIBUTION
Mugurel I. Andreica, *Polytechnic University of Bucharest*
Iosif C. Legrand, *California Institute of Technology*
Nicolae Tapus, *Polytechnic University of Bucharest*

FAULT-TOLERANT SCHEDULING FRAMEWORK FOR MEDIOGRID SYSTEM
Florin Pop, *University Politehnica of Bucharest*
Dacian Tudor, *Politehnica University of Timisoara*

Valentin Cristea, *University Politehnica of Bucharest*
Vladimir Ioan Cretu, *Computing Science Faculty*

AN AGENT BASED FRAMEWORK TO MONITOR AND CONTROL HIGH PERFORMANCE DATA TRANSFERS.

Ciprian M. Dobre, *Politehnica University of Bucharest, Romania*
Ramiro Voicu, *California Institute of Technology, USA*
Adrian Muraru, *European Center for Nuclear Research - CERN, Geneva, Switzerland*
Iosif C. Legrand, *California Institute of Technology, USA*

De asemenea au fost facute mai multe comunicari la Workshop on Global Computing 19-22 April 2007, Sibiu, Romania:

C. Carstoiu(CH) – Monitoring the ALICE Grid with MonALISA
V. Cristea (RO) – SOA and Grid Technologies
C. Stratan, F. Pop (RO) - Scheduling Grid Applications using DIOGENES
C. Dobre (RO) – A Distributed Agent Based System to Control and Coordinate Large Scale Data Transfers
M. Andreica (RO) – Balanced Message Flow Distribution Over Multiple Paths
E. Slusanschi, L.E. Duta (RO) – From Simulation to Optimization of Large-Scale Scientific Applications
C. Stratan, A. Costan (RO) – Resource Usage Accounting in the Open Science Grid
C. Rentea (DE) – Performance Constraints in Business Processes
St. Trausan-Matu (RO) – Semantic GRID

Obiectivele lucrarii au fost realizate integral si corespund cerintelor contractuale ceea ce asigura conditiile pentru realizarea activitatilor etapelor VIII si IX ale prezentului proiect. Etapa urmatoare, etapaVII, are un caracter aplicativ si contribuie la realizarea obiectivul general 5 din enumerarea de la inceoutul prezentului RST.

6. Anexe

Anexa 1: Arhitectura sistemului MonALISA

Anexa 2: MonALISA utilizare