

**Applicazione lenta ?  
E' colpa della rete ! O, forse NO !  
Come sciogliere velocemente  
ogni dubbio ?**

**Indice**

- 1. Introduzione
- 2. End User Experience Monitoring
- 3. La strategia
- 4. Back-in time analysis
- 5. Baselines e notifiche
- 6. Summary

**1. Introduzione**

Applicazione lenta ?

Spesso la rete viene additata come la responsabile di tutto quanto accade alle performance di una applicazione, ma non è sempre detto che lo sia effettivamente. Accade molte volte di assistere ad un palleggio di responsabilità dove, ognuno che è responsabile per una porzione degli assets coinvolti, cerchi di scaricare la responsabilità sugli altri; nel migliore dei casi accade che più divisioni siano chiamate ad investigare lo stesso problema, con inevitabile spreco di risorse preziose.



Figura 1

Questo è ciò che accade quando si implementano sistemi di network management tradizionali, che non sono di particolare aiuto in situazioni in cui devono dirimere una questione legata alle performance degradate di una applicazione.

Verificare che un servizio/applicazione sia disponibile è estremamente facile ed è un compito che i sistemi di network management assolvono da tempo in modo soddisfacente, ma quali informazioni utili sono in grado di fornirci nel caso in cui ad esempio degli utilizzatori di un sito lamentano, a ragione o a torto, un degrado di performance relativo alla applicazione che stanno utilizzando ?

A questo punto dovremmo porci le seguenti domande :

- a. E' vero che l'applicazione è degradata ?
- b. Se si, quanto è degradata rispetto alla norma ?
- c. Quali sono le performance percepite dagli altri utilizzatori ?
- d. Perché nessun altro si lamenta ?
- e. E' un problema di quei singoli utenti di quel sito o è diffuso all'intero sito ?
- f. Come mai le schermate del sistema di network management non indicano alcun problema ?
- g. I responsabili della applicazione e dei server dicono che è tutto ok, perché allora lamentarsi ?

e.. potremmo continuare a lungo nel porci domande.

Un sistema tradizionale di SNMP polling, ideale per conoscere l'utilizzo in termini quantitativi di una determinata risorsa di rete, interfaccia fisica/logica, circuito, VLAN, memoria, CPU ... nulla ci dice circa le performance percepite da chi utilizza una determinata applicazione.

Un eventuale plug-in specifico con funzionalità di active test, atto a controllare che una applicazione/servizio sia raggiungibile, non in grado di correlare quanto da lui stesso misurato con quanto stanno sperimentando realmente gli utilizzatori della applicazione, che vi accedono da remoto.

Cosa fare dunque per iniziare a dare delle risposte concrete ?

## 2. End User Experience Monitoring

Solo ponendo l'utente finale al centro della nostra strategia sarà possibile trovare le risposte ai quesiti che ci siamo posti in precedenza, ovvero l'attenzione deve essere spostata verso chi utilizza un servizio, che per poter essere erogato ed acceduto, utilizza in molti casi innumerevoli risorse aziendali e non solo.

Controllare singolarmente i singoli elementi, senza una visione complessiva, di insieme, che ci fornisca la visione reale di chi sta utilizzando un determinato servizio/applicazione, non ci consente di dare alcuna risposta efficace, ecco perché i tradizionali sistemi di Network Management da soli risultano essere insufficienti.

Misurare l'End User Experience, significa conoscere in ogni istante esattamente la percezione che ha un utilizzatore del nostro servizio/applicazione, ovunque si trovi nel mondo e qualunque modalità di accesso utilizzi, permettendoci di acquisire baselines automatiche che evidenzino il livello di performance percepito normalmente, così facendo sarà estremamente agevole identificare le cause di un eventuale degrado ed attivare dei sistemi di notifica automatica verso una struttura di help desk fornendo non solo la segnalazione del problema ma, anche i dettagli relativi alla specifica metrica responsabile del problema.

Il personale di help desk dovrà semplicemente cliccare sul link allegato alla email di notifica per visualizzare le informazioni specifiche del problema, addirittura nel caso in cui la metrica coinvolta fosse una transazione della applicazione utilizzata, avrebbe a disposizione gli statements associati alla request/response tra client e server.

Per maggiori informazioni relativamente a come misurare l'End User Experience, si rimanda allo specifico whitepaper dal titolo : [Come misurare l'end user experience e quali sono le metriche che contano ?](#)

Quale strategia adottare per costruire un sistema di gestione realmente efficace ?

## 3. La strategia

Al fine di realizzare un sistema di gestione integrato, che risulti essere realmente efficace al bisogno, allineato con le esigenze di business di oggi, è necessario implementare una serie di moduli, ognuno con le proprie specificità.

La caratteristica principale che deve possedere detto sistema, è quella di garantire una correlazione e sincronizzazione delle informazioni, che potranno essere acquisite utilizzando i sistemi più diversi, ma che dovranno essere necessariamente integrate tra loro, mediante l'utilizzo di un mediation & correlation database.

Se ciò non fosse, le informazioni risulterebbero slegate, prive di significato ed inutilizzabili, è come se si disponesse di una pluralità di sistemi, con overlapping e carenze.

Dunque, gli elementi funzionali che necessitiamo sono i seguenti :

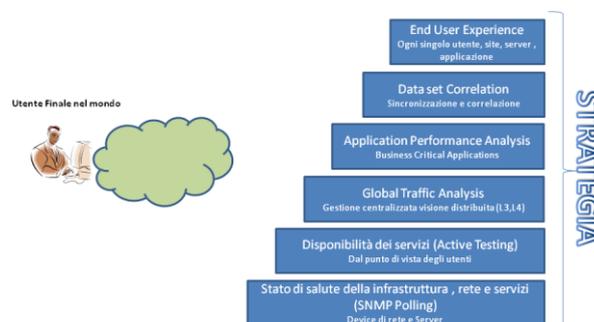


Figura 2

### a. SNMP Polling

Un sistema di SNMP polling è assolutamente essenziale per tenere sotto controllo lo stato di salute della infrastruttura di rete e dei servizi, rilevando utilizzi ed errori.

### b. Active Testing

Verificare la disponibilità di un server con il classico ICMP ping è di per sé necessario ma non sufficiente, come del resto non è sufficiente controllare la disponibilità di un certo servizio all'interno della rete da cui viene erogato.

Sarebbe molto meglio utilizzare delle tecniche che ne verificano la disponibilità dal punto di vista degli utilizzatori.

#### c. Global Traffic Analysis

Tramite l'SNMP polling si ottengono informazioni di tipo quantitativo, (es. quanto è utilizzata la WAN) ma non di tipo qualitativo, ovvero quale traffico abbiamo su di WAN, chi lo sta generando, chi parla e con chi, con quale QOS ecc.

Pertanto necessitiamo di utilizzare delle soluzioni in grado di analizzare il traffico, tramite delle acquisizioni anche distribuite ma con una gestione centralizzata.

Tecnologie come NetFlow, sFlow, JFlow, IPFIX, cFlow .. sono particolarmente idonee per assolvere questo compito.

#### d. Application Performance Analysis

Le soluzioni di traffic monitoring basate sulle tecnologie precedentemente elencate, molto utili per analizzare il traffico in rete, non sono in grado di analizzare le performance delle applicazioni; ecco perché è necessario disporre di appliance e soluzioni ad-hoc specificatamente realizzate per questo tipo di analisi.

Il focus deve essere la transaction analysis e non la packet analysis, ovvero non stiamo parlando dei classici analizzatori di protocollo che sono generalmente destinati ad altri scopi.

#### e. Data Set Correlation

Quanto raccolto dai moduli precedentemente descritti, deve essere visualizzato in modo che le informazioni risultino non solo integrate tra loro, ma anche correlate sulla base di un sample time omogeneo.

La granularità auspicabile è quella di poter disporre di informazioni con un sample minimo di 1 minuto.

#### f. End User Experience

Il punto di vista come detto deve essere l'utilizzatore finale, il sistema deve quindi essere in grado di presentare statistiche e dati avendo il singolo utente,

che è parte di un sito, come elemento privilegiato di fonte di rappresentazione.

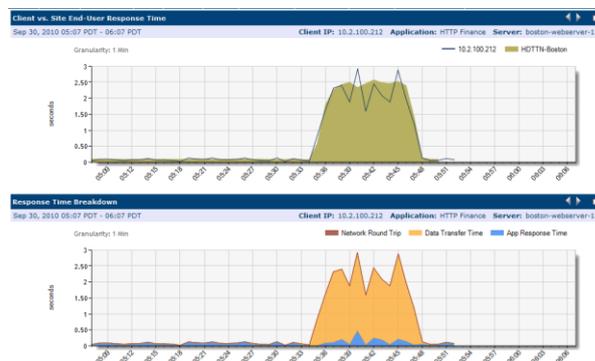


Figura 3

Consentendo di poter confrontare le sue performance con quelle percepite dal sito al quale appartiene, in questo modo risulterebbe estremamente agevole verificare se uno specifico degrado è limitato al singolo utente o è generalizzato all'intero sito di appartenenza.

### 4. Back-in Time Analysis

Generalmente quando si utilizza quale sistema di notifica, di alerting, non un sistema proattivo ma l'utente che chiama l'help desk, il problema segnalato si è già manifestato, quindi se non si disponeva di un sistema già completamente attivo, in grado di raccogliere informazioni nel tempo, si è persa la chance di poter analizzare quanto segnalato.

Ecco dunque che un sistema di gestione integrato, così come descritto al paragrafo precedente, deve possedere tra le proprie caratteristiche anche quella di consentire l'analisi dei dati a ritroso, come se si fosse in real time, da qui il termine back-in time che rende molto bene il concetto.

Naturalmente, maggiore è il tempo che possiamo scorrere all'indietro e maggiore saranno le possibilità di analizzare con serenità il problema, disponendo non solo dei dati statistici, ovviamente, ma anche delle transazioni per ogni singolo utente, delle applicazioni monitorate, eventualmente corredate all'occorrenza anche delle packet trace associate.

Quanto descritto è spesso elemento distintivo di una soluzione rispetto ad un'altra.

## 5. Baselines e notifiche

Disporre di un sistema proattivo, in grado di allertare chi di dovere sull'insorgere di un problema, significa ottimizzare le risorse aziendali evitando palleggi di responsabilità e sprechi.

Tutto ciò è possibile solo se il sistema è in grado di raccogliere in modo automatico ed autonomo le baselines di metriche significative e consente una personalizzazione mediante l'utilizzo di business group ad-hoc al fine di distribuire alert mediante notifiche mirate e non generalizzate.

I business group devono servire anche per consentire la personalizzazione e la creazione di dashboard finalizzate al controllo ed alla distribuzione delle informazioni acquisite.

## 6. Summary

Identificare la causa reale di un degrado delle performance di una applicazione, percepito da alcuni utenti, necessita che il focus sia posto sulla End User Experience.

L'approccio tradizionale a silos, a compartimenti stagni, è un modello del tutto superato, in quanto una applicazione/servizio coinvolge molteplici assets, non necessariamente tutti riconducibili sotto il diretto controllo della stessa organizzazione, pertanto è necessario disporre di una visione che rappresenti la vista di insieme; cioè la End User Experience.

Una analisi dei singoli elementi, condotta in modo separato, non integrata e correlata, ci preclude di poter identificare la reale root cause di un problema, di discriminare cioè se si tratta di client, rete, server o applicazione, di identificare ed isolare quello che si chiama problem domain.

*Autore : Maurizio Malinconi*



**NetPerF** Consulting  
more control for better performances !

Via A. Diaz, 30  
20035 Lissone (MB)  
Tel. +39 02 40047334  
Fax +39 039 2781283  
Email [info@netperf.it](mailto:info@netperf.it)  
Web [www.netperf.it](http://www.netperf.it)