# SMART eLAB

VOLUME 6- ANNO 2015

ISSN 2282 - 2259

# SMART eLAB

# SOMMARIO
# Vol. 6, 2015

## Articoli


1-8 **Giuseppe Nantista, Andrea Lora, Augusto Pifferi** *ZFS: il file system del presente.*

9-11 **Giuseppe Nantista, Andrea Lora, Augusto Pifferi** *Pandora: la piattaforma di "storage in the cloud" dell'Area della Ricerca RM1 di Montelibretti.*

12-18 **Augusto Pifferi, Giuseppe Nantista, Sabina Ponzio, Francesca Vergari** *Comuni tra le Nuvole.*

19-23 **Giuseppe Nantista, Andrea Lora, Augusto Pifferi** *Autenticazione tramite social network per l'accesso a hotspot pubblici gratuiti.*

24-27 **Guido Righini, Luca Ianniello, Mirella Rondinelli, Augusto Pifferi.** *Progetto Romaforma: Interventi formativi in modalità blended a favore dei dipendenti capitolini.*



**Smart e-Lab**: http://smart-elab.mlib.ic.cnr.it
A peer-reviewed online resource, published by the Istituto di Cristallografia (CNR-IC)
EDITORS-IN-CHIEF : Michele Saviano, Augusto Pifferi - ASSOCIATED EDITOR : Guido Righini
GRAPHIC DESIGN : Claudio Ricci - EDITORIAL ASSISTANT : Caterina Chiarella
CNR - Istituto di Cristallografia, Strada Provinciale 35/d, I-00015 Monterotondo, Italy

Networking

# ZFS: il file system del presente.†

Giuseppe Nantista,[a] Andrea Lora,[a] Augusto Pifferi,[a]

In questo rapporto tecnico analizzeremo il file system ZFS, formalmente Zettabyte File System, sviluppato a partire dal 2002 e annunciato in un paper da Jeff Bonwick et al.[1] della Sun Microsystems nel 2003. Descriveremo i principi che sono alla base della sua formulazione teorica e le sue caratteristiche, decantate dagli stessi autori, di semplicità di amministrazione, scalabilità e integrità del dato salvato. Infine citeremo alcuni aspetti pratici di questo file system nella quotidianità del suo utilizzo, sottolineando le implicazioni dietro ogni possibile scelta architetturale.
**Keywords**: ZFS, zettabyte, storage, file system.

## 1 Introduzione

Quando nel 2003 Jeff Bonwick e i suoi colleghi[1] della Sun Microsystems annunciarono la prossima uscita di un file system rivoluzionario sottolinearono da subito come ZFS, lo Zettabyte File System, presentasse caratteristiche di forte integrità dei dati, immensa capacità e facile amministrazione. Di fatto ZFS ha rappresentato una netta svolta nella concezione di un file system di tipo general purpose, spazzando via una serie di credenze e costrizioni che erano (e ahimè talvolta sono ancora oggi) di comune convinzione nell'ambito dei file system ad uso locale.

Sinteticamente i 3 punti sopra citati vengono ottenuti nel seguente modo:

- forte integrità dei dati tramite meccanismi di checksum di ogni singolo dato scritto su disco;
- immensa capacità tramite uno spazio di indirizzamento a 128 bit;
- facile amministrazione tramite l'uso di un metalinguaggio che permetta di esprimere in maniera concisa cosa si vuole fare.

ZFS introduce inoltre alcuni concetti importanti, che sono quelli di pooled storage, modello transazionale copy-on-write e di checksum auto validanti, ottenendo, a volte indirettamente, numerosi vantaggi come la possibilità di creare snapshot del file system in maniera istantanea o la possibilità di non usare più i meccanismi di file system check (il comando fsck ben noto e temuto da qualunque sistemista unix che usi ext come file system) o ancora la possibilità di correggere un errore su file system all'atto della lettura del dato errato, oltre che a richiesta sull'intero file system tramite il comando di scrub.

## 2 Limiti dei vecchi file system

Nel concepire i vecchi file system si faceva riferimento ad assunzioni che risultano oggi obsolete, come il fatto che proteggere i dati con l'uso di un sistema di checksum sia troppo oneroso per un computer. Questo non è più vero grazie alle capacità di calcolo sempre maggiori dei computer moderni, ma anche se così fosse i dati sarebbero troppo preziosi per non essere protetti con un algoritmo di checksum anche "light"; di base ZFS usa l'algoritmo di Fletcher[2] a 64 bit, un algoritmo di checksum che si propone come ottimo rapporto fra efficacia ed efficienza, molto più semplice dal punto di vista computazionale di un codice CRC, ma più debole per quanto riguarda la sua capacità di individuare errori. Per capire quanto questo sia importante si faccia riferimento a uno studio approfondito condotto dal CERN[3] che ha evidenziato come l'incidenza di errori "silenti" sia distruttiva nei file system tradizionali, incoraggiando un utilizzo diffuso dei meccanismi di checksum.

Un altro limite dei file system tradizionali è la corrispondenza fra un file system e un "volume". Facciamo un passo indietro, inizialmente un file system era correlato a un singolo disco. Questo approccio molto limitativo fu superato adottando il concetto di volume, che di fatto è un dispositivo a blocchi virtuale, ottenuto con differenti metodi a partire da dispositivi fisici, ad esempio dal

[a] *Istituto di Cristallografia, C.N.R. via Salaria km 29.300, 00015 Monterotondo, Italy*

† Rapporto tecnico 2014/11 con protocollo CNR-IC n. 1383 del 01/08/2014

mirror di due dischi o dalla loro concatenazione o da un RAID5, meccanismo che usa n dispositivi fisici di uguale capacità per ottenere un dispositivo virtuale con capacità pari a quella di n-1 dischi. Rimanendo sull'argomento il meccanismo RAID5 prevede che il controller dei dischi abbia una cache di memoria non volatile (NVRAM), per tamponare eventuali mancanze di energia elettrica durante una scrittura a disco, evento che comprometterebbe il contenuto dell'ultimo blocco di informazioni che si stava scrivendo al momento del blackout.

Un ulteriore limite è relativo alla massima dimensione in byte del file system. Anche usando 64 bit per indirizzare i blocchi di un file system, così come fatto dal alcuni recenti sistemi (stiamo di proposito ignorando i file system con spazio di indirizzamento a 32 bit che sono già ad oggi insufficienti), la massima dimensione allocabile sarebbe di 16 exabyte, un valore non così difficilmente raggiungibile nei prossimi 10 o 20 anni. ZFS ha optato quindi per un indirizzamento a 128 bit, che rende di fatto irraggiungibile la dimensione massima di un file system.[4]

## 3 Principi di ZFS

Gli studi che sono alla base dello sviluppo di ZFS si posano su pilastri fondamentali, evidenziati da Bonwick e colleghi già in fase di design.

### 3.1 Amministrazione semplice

La sintassi usata da ZFS è stata studiata per rendere immediato e sicuro impartire un qualunque comando al sistema, così creare un pool, un file system, uno snapshot o un clone sono operazioni che possono essere eseguite con massima facilità. Anche se tutte le operazioni compiute a livello di file system sono in carico agli amministratori di sistema e sono eseguite raramente, non c'è motivo per non semplificarle riducendo al minimo la possibilità, in caso di emergenza, di commettere errori irrecuperabili.

### 3.2 Pooled storage

Il concetto qui descritto è forse uno dei più rivoluzionari introdotti da ZFS. Consiste nella possibilità di mettere a fattore comune tutti i dischi fisici a disposizione a formare un'unica risorsa da cui pescare spazio per creare i file system. In questa maniera non ci sono problemi nel far crescere lo spazio a disposizione di uno di essi, nel senso che di default ogni file system ha a disposizione tutto lo spazio del pool su cui esso è stato creato. Nulla vieta tuttavia di impostare delle quote (lo spazio massimo occupabile da un FS) o delle reservation (lo spazio minimo garantito a un FS). È evidente come il concetto stesso di pooled storage si porta dietro quello appena descritto di file system di dimensione variabile.

Un'altra operazione consentita da un approccio del genere è il grow di un pool, ossia la possibilità di aumentare le sue dimensioni dinamicamente con la sola aggiunta di dischi. In fig. 1 è mostrata in maniera abbastanza esplicita una operazione di grow di un pool. Mostriamo da subito un caveat di questa operazione, mentre ci si aspetta che le proprietà di ridondanza vengano mantenute, in realtà si perde la capacità del sistema di resistere, senza perdita di dati, al guasto di un disco. Questo per via di una precisazione che è da fare relativamente all'operazione cosiddetta di restripe, l'operazione con cui il contenuto di un pooled storage viene ridistribuito sulla totalità dei dischi che compongono il pool. Questa operazione non viene compiuta automaticamente, pertanto il contenuto del pool non viene riarrangiato per occupare anche il disco appena aggiunto. In certi casi questo è possibile con dei workaround, come ad esempio esportare e reimportare un file system su un altro pool locale. Nella figura è stato anche introdotto un concetto che è quello del virtual device (vdev), una struttura intermedia che di fatto assume il significato di logical volume manager. Un

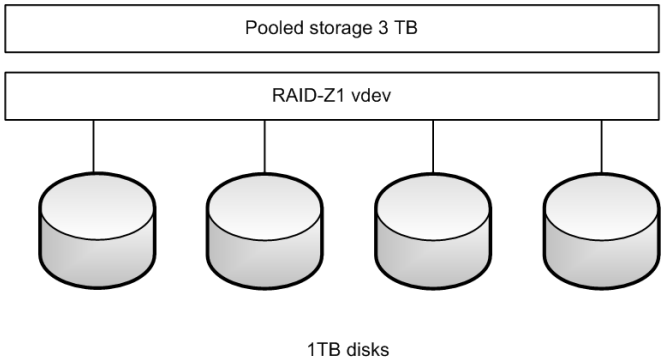



**Fig. 1** Grow di un pool mediante aggiunta di un disco: situazione iniziale.

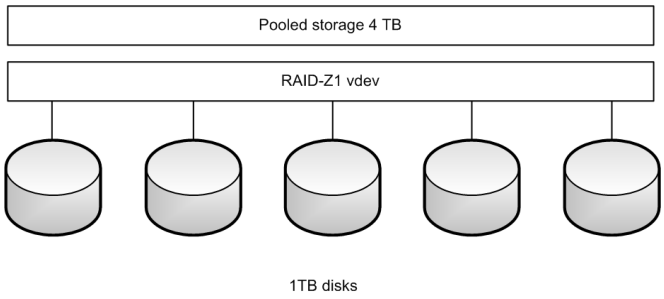



**Fig. 2** Grow di un pool mediante aggiunta di un disco: soluzione attesa

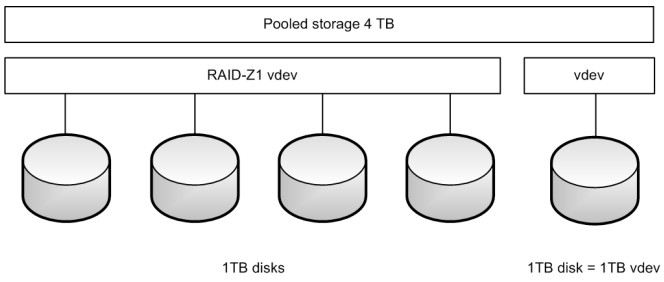



**Fig. 3** Grow di un pool mediante aggiunta di un disco: soluzione ottenuta

vdev può consistere di uno o più dischi legati secondo una logica, come il mirror, la concatenazione o il RAID per creare lo spazio del pool.

### 3.3 Consistenza dei dati su disco

Al fine di evitare situazioni di inconsistenza dei dati salvati su disco è fondamentale che il sistema transiti da uno stato consistente a un altro. Un modo efficace è quello di applicare la politica Copy On Write, che consiste nel copiare un blocco ogni qual volta si renda necessario modificarlo, aggiornando i puntatori ad esso solo quando è stata completata l'operazione di scrittura. L'approccio COW apre due scenari interessanti, la creazione degli snapshot e dei cloni.

Concettualmente sono due oggetti molto simili, uno snapshot è un'istantanea del file system al momento della creazione dello stesso. Ogni qual volta un blocco viene modificato il COW crea un blocco nuovo e aggiorna i puntamenti ad esso (i puntamenti sono riferimenti a quel blocco, di fatto sono quindi dati scritti su un altro blocco seguendo una struttura gerarchica). La differenza fra creare uno snapshot e modificare un blocco è che l'area usata dai blocchi modificati non viene liberata, ma rimane referenziata per una futura operazione di rollback alla situazione precedente. Per loro natura gli snapshot sono read only, perché una modifica pregiudicherebbe l'operazione di rollback. Invece sono in tutto e per tutto file system in lettura/scrittura i cloni, ottenuti a partire dal file system parent che vivono di vita propria.

Un'altra forma di consistenza è quella relativa alla correttezza dei dati scritti. ZFS arricchisce l'area dei metadati associati a un file con il campo checksum, ossia un codice di parità relativo al blocco stesso, capace di rivelare la corruzione del blocco cui si riferisce, attivando i meccanismi di correzione dell'errore (healing) leggendo il blocco da un mirror. Questa operazione di verifica viene compiuta ogni volta che un file viene letto, ma l'amministratore può decidere di lanciare il comando di ***scrub***, che operando a bassa priorità legge tutto il file system alla ricerca di errori nascosti e li corregge. I metadati di checksum sono salvati nel blocco parent del blocco in oggetto, pertanto allocando i blocchi sul disco in maniera sparsa si aumentano le probabilità che in caso di corruzione di un blocco il suo blocco parent sia corretto.

### 3.4 Immensa capacità

Come già discusso brevemente nell'introduzione ZFS consente la creazione di file system di capacità praticamente non raggiungibile. Un singolo file o un file system possono raggiungere la dimensione massima di 16 exabyte (1 EB = $2^{60}$ byte), una directory può contenere $2^{48}$ file, uno zpool può comprendere $2^{64}$ vdev e contenere $2^{64}$ file system.

I file vengono collocati all'interno di blocchi, ZFS consente di utilizzare blocchi di dimensione variabile da 512 a 128 Kbyte per ridurre gli sprechi legati alla operazione di fit di un file in N blocchi.

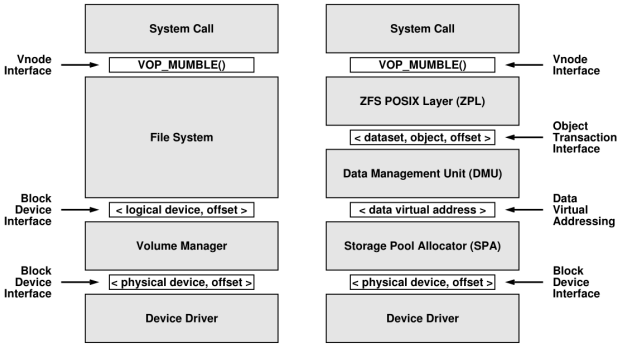



**Fig. 4** Il modello di storage di ZFS

Prevede inoltre meccanismi di compressione del dato, cosicché lo spazio occupato su un file system sia ulteriormente ridotto a scapito della capacità computazionale impiegata per comprimere ed espandere i dati. Sono possibili diversi algoritmi di compressione, inclusi quelli della famiglia gzip, ma di default quello usato è l'algoritmo LZJB, sviluppato appositamente da Jeff Bonwich partendo dal nucleo originale di Lempel e Ziv (di qui l'acronimo).

Infine il meccanismo di deduplica fa sì che due blocchi uguali occupino lo spazio di uno solo.

## 4 Caratteristiche di ZFS

### 4.1 Il modello di storage di ZFS

Il modello di storage adottato da ZFS è mostrato in fig. 4, dove viene confrontato con il modello standard degli altri file system. I file system tradizionali accedono a un block device, sia esso un disco fisico o un volume logico ed esportano una interfaccia di tipo vnode al sistema operativo, ZFS introduce invece un approccio differente di seguito descritto:

1. *Lo storage pool allocator (SPA)*

   Il livello più basso dell'architettura ZFS è lo storage pool allocator, che interfacciandosi con unità fisiche, i dischi, tramite un accesso a blocchi, fornisce allo strato superiore dell'architettura un'interfaccia per allocare spazio disco. Poiché accede ai blocchi fisici dei dischi è lo SPA a prendere in carico l'operazione di checksum su di essi, memorizzando l'informazione nel blocco parent di quello scritto. Se in fase di lettura lo SPA si accorge di un mismatch fra i dati letti e il loro checksum effettua la correzione leggendo l'informazione dai blocchi corretti.

   Tutte le operazioni che in un file system tradizionali sono affidate al volume manager vengono in ZFS eseguite dallo SPA, che realizza mirror logici, concatenazioni, ecc. fra dischi appartenenti allo stes-

so pool. Quando lo SPA esegue queste operazioni crea un vdev, un dispositivo virtuale ottenuto secondo le logiche di ridondanza desiderate a partire da dispositivi fisici e/o altri vdev.

La logica con cui lo SPA alloca i blocchi dati partendo dai vdev è di tipo round-robin, effettuato in maniera dinamica per default. Se un dispositivo è stato recentemente aggiunto a un pool le scritture convergeranno prevalentemente su di esso fino a pareggiare l'occupazione percentuale del vdev stesso.

2. *La data management unit (DMU)*

   La DMU è la porzione dell'architettura ZFS che accedendo alla struttura di "virtual addressed blocks" offerta dallo SPA offre allo strato superiore, lo ZPL, un'interfaccia transazionale a oggetti. Questi oggetti sono unità di memorizzazione dati, identificati da un indirizzo a 64 bit, contenenti un massimo di 264 byte di dati, che lo ZPL può creare, distruggere, leggere e scrivere.

   La DMU implementa la logica Copy On Write, occupandosi di gestire il puntamento ai blocchi secondo una logica padre-figlio, realizzando una struttura gerarchica in cui il blocco padre di tutta la struttura è il cosiddetto überblock, un blocco ovviamente trattato in maniera particolare dalla DMU, che ne conserva una copia di backup da ripristinare in caso di transazione non eseguita correttamente per colpa ad esempio di un blackout. L'insieme di scritture da eseguire in ogni transazione è variabile, perché la DMU le raggruppa insieme, così da limitare il numero di accessi in scrittura ai blocchi incluso l'überblock.

3. *ZPL, ZFS Posix Layer*

   Al sistema operativo viene offerta una interfaccia di tipo vnode tradizionale, secondo le direttive POSIX (Portable Operating System Interface for uniX), che di fatto rappresenta l'unico compromesso architetturale di ZFS.

### 4.2 RAID-Z

Oltre ai meccanismi classici di mirror, concatenazione e stripe ZFS supporta anche il RAID-Z, nelle varianti Z1, Z2 e Z3, dove la cifra dopo la Z indica il numero di dispositivi guasti a cui il sistema resiste senza perdita del dato. Il modello a cui si ispira il RAID-Z è lo stesso del RAID-5, che permetteva di usare n dischi di uguale dimensione per ottenere un volume di capacità pari a quella di n-1 dischi. Analogamente RAID-Zi usando n vdev di uguale dimensione ottiene uno storage pool di capacità pari a quella di n-i. È possibile anche usare vdev di dimensione differente perdendo parte della capacità dei dischi più grandi.

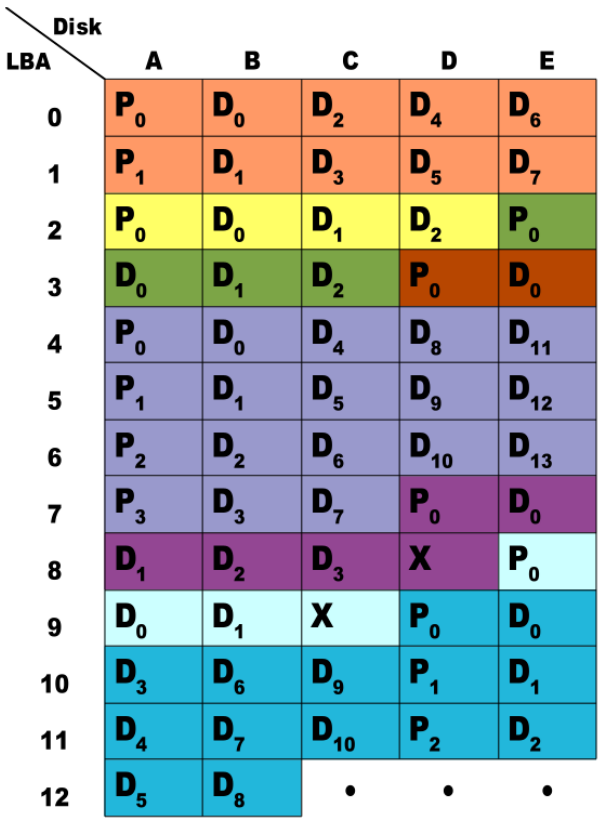



**Fig. 5** Distribuzione dei blocchi in RAID-Z

Il punto in cui RAID-Z si differenzia maggiormente dai suoi predecessori è la logica di distribuzione dei blocchi di parità, quelli cioè che consentono di recuperare l'informazione in caso di guasto. Il tradizionale RAID-5 effettua lo XOR (OR esclusivo) sui blocchi analoghi dei primi n-1 dischi e scrive il risultato nell'ultimo blocco, consentendo il recupero per differenza in caso di guasto, ma non in caso di errore di tipo "silent", situazione in cui il sistema rischia di correggere dati già corretti a partire da dati di parità inconsistenti. Un problema del tutto analogo si manifesta con il cosiddetto "write hole", la situazione cioè in cui fra la scrittura del dato e la scrittura della parità avviene un blackout; il modo comunemente utilizzato per resistere al write hole è l'uso di memorie NVRAM che mantengono il dato anche in caso di interruzione della corrente elettrica.

ZFS invece garantisce la coerenza dei dati grazie a sopra citato meccanismo "copy-on-write" che assicura l'atomicità di ogni singola scrittura, in aggiunta a questo il RAID-Z arricchisce il pool dei dati di informazioni di ridondanza che sono utili in caso di danneggiamento totale o parziale di un disco. Non è possibile come in RAID5 correggere dati validi usandone di inconsistenti, perché ZFS sa sempre quale dato è corretto e quale no grazie ai checksum.

Lo schema di RAID-Z è simile a quello di RAID5 nel senso che utilizza sempre un approccio di tipo dati + pa-

rità, la differenza sta nel fatto che ogni blocco che viene scritto ha una dimensione variabile, pertanto anche lo stripe, ossia l'insieme di blocchi fisici sui dischi, su cui una scrittura viene distribuita, ha una dimensione variabile. In fig. 5 viene illustrato il metodo di scrittura dei blocchi dati (evidenziati da una lettera D) e dei blocchi di parità (lettera P).

L'esempio in questione riporta il caso di 5 dischi (lettere A-E in ascissa) scritti con algoritmo di parità RAID-Z1, mentre in ordinata sono riportati i blocchi logici dei dischi (logical block addresses), ogni stripe di un colore differente rappresenta un blocco ZFS.

Nell'esempio sono stati anche riportati 2 errori localizzati, evidenziati con una lettera X, corrispondenti a un fallimento nella verifica del checksum, tali errori sono recuperabili tramite i dati corretti e le parità. Il massimo guasto tollerato da questo sistema è pari a un intero disco (una colonna).

### 4.3 Ditto Blocks

Quando un errore compromette la possibilità di leggere un blocco i dati in esso contenuti sono persi, a meno di meccanismi di ridondanza come quelli appena descritti. Ma mentre la perdita di un blocco dati ha conseguenze limitate, la perdita di un blocco intermedio nella struttura gerarchica del file system ZFS, contenente riferimenti e metadati, comporta l'impossibilità di accedere a tutti i blocchi da esso dipendenti.

Per salvaguardare l'integrità di questi blocchi, che da uno studio compiuto al riguardo occupano circa il 2% dello spazio disco, è possibile crearne delle copie in altri settori del disco, così da minimizzare la possibilità che un singolo guasto possa cancellare entrambe le copie. Se il sistema è munito di un solo disco, la distanza minima a cui tenere il blocco di copia è pari a 1/8 del disco stesso. Più è importante il blocco, ossia più in alto esso si trova nella struttura gerarchica, più copie possono esserne fatte per salvaguardarlo; ZFS di default assegna due locazioni (DVA, Data Virtual Address) per i blocchi intermedi e tre per i dati di accesso globale.

Nella terminologia ZFS vengono chiamati Ditto Blocks i blocchi in cui memorizzare le copie di ridondanza dei DVA.

## 5 Caratteristiche avanzate e precisazioni

Discutiamo in questo paragrafo alcuni aspetti che è importante conoscere prima di operare con il file system ZFS. Facciamo esplicitamente riferimento al documento di Andrew Galloway[5] del 2011, reperibile solo in rete.

Partiamo da una affermazione spiritosa: Il fatto che tu possa fare una cosa non implica che tu debba. Questo può far sorridere, ma effettivamente esprime bene il concetto per cui alcune soluzioni consentite da ZFS non rappresentano l'ottimo e vanno pertanto considerate con i loro pro e i loro contro. ZFS impone pochissimi limiti, ma il giusto compromesso non si ottiene mai spingendosi al massimo consentito dal sistema. Pensiamo ad esempio al fatto che si possano raggruppare $2^{64}$ vdev in un pool, questo è evidentemente un valore cui non bisogna nemmeno lontanamente avvicinarsi, pena la crescita smisurata del numero di operazioni di I/O al secondo (iops).

### 5.1 La deduplica

Questa tecnica consente di risparmiare grosse quantità di spazio su disco a discapito di un forte utilizzo di RAM per memorizzare le informazioni necessarie a sfruttarla e di CPU per valutare se un blocco è deduplicabile. Ogni volta che un blocco viene consegnato dallo ZPL alla DMU, questa ne calcola l'hash tramite algoritmo SHA256 e verifica se il blocco in questione può essere deduplicato, ossia se esiste già un blocco uguale (più precisamente un blocco con lo stesso hash, quindi attenzione perché c'è una seppur remota probabilità di collisione) scritto sul disco, nel qual caso non lo riscrive ma salva un riferimento al blocco già presente. Per funzionare il meccanismo fa uso di tavole di deduplica, che vengono tenute in RAM e vengono create e consultate in fase di scrittura e lettura dai file system. Questo può comportare un grosso dispendio di memoria, ben oltre il vantaggio di aver risparmiato spazio su disco. Le indicazioni di massima da parte di SUN sono quelle di compiere due valutazioni: prima tramite il tool zdb (ZFS debugger) valutare la possibile deduplication ratio, ossia il fattore di risparmio che si otterrebbe attivando la deduplica, se questo valore supera il fattore 2 allora ci può essere convenienza; la seconda verifica va compiuta sulla quantità di memoria necessaria ad attivare la deduplica a fronte di quella disponibile. La memoria richiesta si calcola moltiplicando il numero di blocchi allocati per 320, che è il valore in byte di una entry nella DDT. In appendice, in tab. 1, mostriamo un possibile scenario in cui il fattore di guadagno derivato dalla deduplica è nullo. In questo caso attivare la deduplica comporterebbe un consumo di RAM pari a 326,4 MB di memoria, pari a 320 byte x 1,02M (numero di blocchi in uso).

### 5.2 La compressione

Non ci sono dubbi a riguardo, la compressione va usata. Il guadagno in termini di I/O ottenuto è notevole e la perdita in termini di risorse CPU è trascurabile, specie se la compressione utilizzata è la LZJB, la versione di Jeff Bonwich della famiglia di algoritmi di compressione Lempel Ziv, più leggero e di conseguenza meno oneroso per il processore. È tuttavia possibile utilizzare l'algoritmo gzip selezionando il flag di "compression level" al fine di trovare il giusto rapporto fra costo computazionale e risparmio di spazio.

### 5.3 L'uso di memoria

ZFS usa tutta la memoria che gli viene messa a disposizione. Una feature di ZFS molto utile per l'aumento delle prestazioni è la ARC, Adaptive Replacement Cache, un meccanismo molto avanzato di caching dei dati. In aggiunta è possibile usare la L2ARC, che fa uso di unità disco molto veloci (ad esempio dischi SSD) per avere una quantità superiore di memoria cache, seppur meno prestante della RAM del sistema.

### 5.4 Amministrazione semplice ma "not for dummies"

I comandi di ZFS sono immediati, ma è necessario sapere cosa si sta facendo prima di lanciare un comando. Questo ci riconduce alla prima precisazione di questo paragrafo, il fatto che tu possa fare una cosa non implica che tu debba.

Vogliamo fare riferimento all'operazione, riportata al paragrafo 3 e in fig. 1, di grow di un pool; se fosse possibile realizzarla così come auspicato l'operazione manterrebbe il livello di affidabilità iniziale, ossia la resistenza del sistema al guasto di uno qualunque dei dischi fisici; invece quello che succede, e che viene scherzosamente definito "hating your data", è che i dati scritti dopo l'operazione di grow saranno distribuiti, in funzione dello spazio disponibile, su due vdev, uno con affidabilità di un disco guasto su quattro e uno senza alcun livello di affidabilità, con conseguente perdita di tutti i dati in esso contenuti in caso di guasto fisico. Un modo per evitare questo è quello di creare a parte un nuovo pool, esportare i file system, distruggere la struttura precedentemente creata e ricrearla da capo con l'aggiunta del nuovo disco.

Ci sono poi particolari features del sistema che vanno apprese pena il fallimento degli intenti. Per esempio nelle operazioni send/receive incrementale del file system è importante che il filesystem ricevente non sia modificato. Eseguire una semplice visualizzazione di un file, che non portano modifiche ad esso, sono però sufficienti a cambiare il contenuto del filesystem, poiché sono cambiati, in questo esempio, i metadati di accesso. La risoluzione del problema qui e' settare il parametro readonly nel filesystem, che ne impedisce qualsiasi modifica.

### 5.5 ZIL: ZFS Intent Log

Una discussione estesa di cosa sia ZIL è fuori dallo scopo di questo documento. Lo scopo dello ZIL è quello di proteggere i dati in caso di fallimento della macchina. Per questo potremmo assimilarlo ad un sistema di journaling di un file system, ma come ha fatto notare Toponce[6] è più simile ad RDBMS che rispetti le specifiche ACID. Qualunque scrittura sincrona richiesta a ZFS verrà scritta sullo ZIL, e solo dopo l'ACK di questo verrà mandata la conferma allo strato applicativo. ZFS poi eseguirà il commit secondo le sue transazioni sullo strato fisico. In una situazione in cui non venga specificato un device apposito, lo ZIL corrisponde ad una zona del pool corrente, con gli stessi tempi di accesso. L'esistenza o la non esistenza dello ZIL non cambia il funzionamento e le performance del sistema. ZFS permette di inserire nel pool uno SLOG (Separate Log Device) che sarà la nuova locazione dello ZIL. In caso di scrittura sincrona quindi ZFS scriverà sullo SLOG i dati, ricevuto l'ACK dallo SLOG provvederà a trasmetterlo allo strato applicativo, nel frattempo potrà scrivere i dati sui suoi dischi. Se il dispositivo che funge da SLOG è un dispositivo ad alte performance (SSD o RAM con backup a batteria) tutte le scritture sincrone beneficeranno di tempi di esecuzione ben inferiori a quelli che avrebbero ottenuto in una configurazione senza SLOG. È impossibile stimare i miglioramenti ottenuti da uno SLOG, visto che cambiano in base al carico di lavoro, ma se ZFS è utilizzato per scritture sincrone (un server DB ne è un esempio) il suo utilizzo è fortemente consigliato. Secondo il workflow sopra illustrato se ci fosse un problema alla macchina tra quando lo SLOG ha inviato l'ACK ma lo strato fisico ancora non ha terminato il commit lo SLOG è il solo contenitore di informazioni che lo strato applicativo dà per garantite. E' quindi norma comune configurare lo SLOG come mirror di due dischi, poiché la rottura del componente può generare perdita di dati.

## 6 Licenze d'uso

Una annosa questione relativa a ZFS è il suo mancato inserimento all'interno del kernel di Linux, per via di un conflitto fra le licenze sotto cui i due prodotti open source vengono rilasciati.

ZFS è licenziato secondo la CDDL, Common Development and Distribution License, che impedisce esplicitamente una redistribuzione del software secondo altre condizioni, mentre il kernel di Linux è distribuito sotto GNU GPLv2, General Public License versione 2, che impone che un software che venga distribuito insieme ad esso debba essere licenziato GPL, anche se giudicato separato e indipendente. Un siffatto comportamento viene spesso indicato come "virale" e impedisce che i due progetti possano essere distribuiti congiuntamente.

Questo aspetto è stato trattato più approfonditamente in altre sedi[7], in questo rapporto tecnico si vuole solo dare un'idea sommaria del perché un prodotto così ben strutturato e open non è stato integrato in quello che è il sistema operativo open source per antonomasia.

## 7 Conclusioni

Il file system ZFS è stato sviluppato ormai oltre un decennio fa e rappresenta ad oggi il miglior file system per sistemi stand alone. Le sue caratteristiche di affidabilità, scalabilità e semplicità di amministrazione lo rendono ineguagliabile al confronto con i sistemi analoghi usati in ambito personal computer e server, tuttavia una difficoltà

di carattere legislativo ne ha impedito lo sviluppo diffuso. Installare ZFS in ambiente linux è possibile, distribuire i binari di installazione in un sistema con kernel licenziato GPL no. Il progetto ZFS on Linux ornisce, nei limiti del possibile tutto il supporto necessario per adottare questo file system in ambienti open source tradizionali.

Perché "il file system del presente" e non del futuro? La risposta è semplice, il futuro dei sistemi informatici vede sempre più affermarsi le architetture virtualizzate e lo storage distribuito dei dati, con strutture scalabili orizzontalmente all'infinito senza i limiti fisici dello chassis in cui risiede un server. ZFS non è un file system distribuito, anche se dal suo sviluppo hanno visto la luce progetti paralleli, come Lustre, che si pongono come obiettivo quello di applicare ai sistemi distribuiti le valutazioni compiute durante lo studio e lo sviluppo portati avanti da Bonwick e colleghi.

Il nostro gruppo di lavoro ha utilizzato ZFS come file system di storage per altri sistemi realizzati, come ad esempio Mercurio, il server di posta elettronica dell'Area della Ricerca RM1 del CNR o Pandora, la piattaforma di storage in the cloud.

Per il server di mail sono state utilizzate pesantemente le funzionalità di snapshot di ZFS. Affiancando alle feature uno script di automatizzazione è stato possibile realizzare degli snapshot incrementali e granulari: vengono creati gli snapshot ogni quarto d'ora (con retention degli ultimi 4), ogni ora (con retention 24), ogni giorno (con retention 30), ogni mese (retention 12) e ogni anno (retention illimitata). Lo spazio effettivamente consumato dai backup aumenta al crescere delle email cancellate dagli utenti, visto che le mail che attualmente sono presenti nelle caselle non pesano sul backup. Lo stesso approccio senza l'uso di ZFS avrebbe richiesto spazi di storage decisamente superiori.

Per eseguire il backup dei pool ci affidiamo alle capabilities di send/receive di ZFS, che ci permettono di eseguire il salvataggio dei dati delle piattaforme in produzione senza nessuna interruzione. Abbiamo attivato la compressione sul pool del ricevente, visto che gli attributi possono essere settati indipendentemente, beneficiando di un risparmio di spazio.

ZFS offre anche il supporto allo ZVOL, un dispositivo a blocchi creato sopra il pool ZFS, che poi può essere esportato tramite un protocollo che gestisca questo tipo di device, come ad esempio iSCSI. Questo consente di utilizzare ZFS come backend disco di sistemi di virtualizzazione o di usarlo come block device di un sistema operativo sempre tramite iSCSI.

## 8 Appendice

Riportiamo di seguito alcuni comandi previsti dalla sintassi ZFS, così da mostrare praticamente uno dei punti fondamentali del file system, la sua semplicità di amministrazione. Come creare uno storage pool denominato "tank" usando come vdev il mirror dei due dischi fisici c2d0 e c3d0

```
# zpool create tank mirror c2d0 c3d0
```

Come creare un file system denominato "home" come figlio di "tank" ed esportarlo al mountpoint /export/home

```
# zfs create tank/home
# zfs set mountpoint=/export/home tank/home
```

Come creare le home directory per alcuni utenti. Per l'ereditarietà della operazione di export queste saranno automaticamente esportate ai mountpoint /export/home/<nome>

```
# zfs create tank/home/giuseppe
# zfs create tank/home/andrea
# zfs create tank/home/augusto
```

Come aggiungere al pool altro spazio, un vdev ottenuto dal mirror di c4d0 e c5d0

```
# zpool add tank mirror c4d0 c5d0
```

Come esportare automaticamente tutte le home directories via NFS

```
# zfs set sharenfs=rw tank/home
```

Attivare la compressione sul pool "tank"

```
# zfs set compression=on tank
```

Limitare la quota di Giuseppe a 10 gigabyte

```
# zfs set quota=10g tank/home/giuseppe
```

Garantire ad Andrea una reservation di 20 gigabyte

```
# zfs set reservation=20g tank/home/andrea
```

Creare uno snapshot della home directory di Augusto

```
# zfs snapshot tank/home/augusto@tuesday
```

Effettuare l'operazione di roll back a uno snapshot precedente

```
# zfs rollback tank/home/augusto@monday
```

Accedere in lettura a una versione precedente di un singolo file. Wednesday è il nome dello snapshot.

```
$ cat ~maybee/.zfs/snapshot/wednesday/foo.c
```

```
# zdb -S tank
Simulated DDT histogram:

bucket              allocated                       referenced
______   ______________________________   ______________________________
refcnt   blocks   LSIZE   PSIZE   DSIZE   blocks   LSIZE   PSIZE   DSIZE
------   ------   -----   -----   -----   ------   -----   -----   -----
     1    1.00M    126G    126G    126G    1.00M    126G    126G    126G
     2    11.8K    573M    573M    573M    23.9K   1.12G   1.12G   1.12G
     4      370    418K    418K    418K    1.79K   1.93M   1.93M   1.93M
     8      127    194K    194K    194K    1.25K   2.39M   2.39M   2.39M
    16       43   22.5K   22.5K   22.5K      879    456K    456K    456K
    32       12      6K      6K      6K      515    258K    258K    258K
    64        4      2K      2K      2K      318    159K    159K    159K
   128        1     512     512     512      200    100K    100K    100K
 Total    1.02M    127G    127G    127G    1.03M    127G    127G    127G

dedup = 1.00, compress = 1.00, copies = 1.00, dedup * compress / copies = 1.0

DDT memory needed = 326,4 MB
```

Tab. 1 Verifica della possibilità di deduplica del file system.

## Riferimenti


1 J. Bonwick, M. Ahrens, V. Henson, M. Maybee, M. Shellenbaum, The Zettabyte File System (2003).

2 J. Fletcher, An arithmetic checksum for serial transmissions, IEEE Transactions on Communications 30 (1) (1982) 247–252. doi:10.1109/TCOM.1982.1095369.

3 B. Panzer-Steindel, CERN/IT, Data integrity, Draft 1.3 8. (April 2007).

4 S. Lloyd, Ultimate physical limits to computation, Nature 406 (2000) 1047–1054.

5 A. Galloway, Things about ZFS that nobody told you (2011).

6 A. Toponce, ZFS Administration, Appendix A - Visualizing the ZFS Intent Log (2013).

7 R. Williams, R. Townsend, (Attorneys at Law), ZFS on Linux: Copyright and Licensing Issues.

Networking

# Pandora: la piattaforma di "storage in the cloud" dell'Area della Ricerca RM1 di Montelibretti.†

Giuseppe Nantista,[a] Andrea Lora,[a] Augusto Pifferi,[a]

In questo rapporto tecnico descriveremo il setup di una piattaforma di cloud storage accessibile dagli utenti dell'Area della Ricerca RM1 che hanno già un account email attivo sul server di posta elettronica locale, Mercurio. Dopo la fase iniziale di setup abbiamo collegato una share NFS ospitata su uno storage esterno su cui memorizzare i dati degli utenti, quindi configurato il bridge LDAP per l'autenticazione single sign on mediante le credenziali di posta elettronica.
**Keywords**: OwnCloud, cloud storage, single sign on, collaboration tools

## 1 Introduzione

La collaborazione fra membri di un gruppo di lavoro ha bisogno di opportuni strumenti, in questo rapporto tecnico discutiamo di uno strumento software che da la possibilità di accedere da remoto ai dati condivisi del gruppo di lavoro, senza precluderne l'accesso da terminali mobili come tablet o smartphone.

Esistono servizi come Dropbox o Google Drive, tanto per citare i più diffusi, che mettono a disposizione questa possibilità anche in maniera gratuita con spazi allocati di modesta entità. Tuttavia la necessità di allocare spazio in quantità elevata, unita alla preferenza per ospitare dati sensibili della ricerca scientifica su piattaforme a gestione interna, ci ha spinti a sviluppare un sistema analogo basato su software open source reperibili in rete e opportunamente supportati dalla comunità.

Il progetto Pandora si basa su software di cui il core è costituito da ownCloud e permette la memorizzazione di file sui server dell'Area di Ricerca RM1 di Montelibretti, per poi accedervi dovunque si abbia una connessione ad internet.

Lo scopo di Pandora è quello di offrire agli utenti un modo facile, veloce ed affidabile di conservare e condividere file, per ulteriori informazioni sul software, le guide sull'utilizzo e per scaricare i client di sincronizzazione è possibile far riferimento al sito del prodotto: http://owncloud.org/

Cos'è in pratica Pandora? Un sistema di storage remotizzato, che permette di mantenere i propri file memorizzati in sicurezza su una cartella ospitata presso un server remoto, accessibile via web da qualunque parte del mondo. Non solo, l'accesso tramite Sync Client offre la possibilità di mantenere una copia di tutti i file sui propri PC, replicando le modifiche su tutte le copie attive contemporaneamente, così da poter modificare i propri documenti da più postazioni senza dover salvare su supporti non affidabili (le classiche chiavette USB) e consentendo anche il lavoro off line nei momenti in cui l'accesso a internet non è disponibile.

## 2 Setup

Abbiamo approntato un server con distribuzione Debian Wheezy su cui abbiamo installato i package di base richiesti da owncloud, quindi apache2, php5, mysql server, curl e altre librerie richieste.

Data la riservatezza dei dati in transito e delle credenziali di login degli utenti è fondamentale che tutte le comunicazioni siano cifrate. Per questo abbiamo installato dei certificati SSL ottenuti gratuitamente tramite l'accordo con il GARR e Terena Certification Authority.

L'installazione del pacchetto owncloud si effettua semplicemente estraendo dentro la cartella /var/www il package scaricato dal sito http://owncloud.org, lo step successivo è stato quello di collegare la cartella /var/www/-data, dove verranno memorizzati i dati degli utenti, allo storage esterno tramite share NFS.

Per consentire l'accesso tramite la login della propria casella e-mail è stato quindi necessario abilitare il plugin LDAP e configurarlo opportunamente. È necessario configurare non solo l'accesso in lettura all'albero LDAP relativo agli account utente, ma anche distinguere un utente abilitato ad avere il suo spazio su Pandora da un utente

[a] *Istituto di Cristallografia, C.N.R. via Salaria km 29.300, 00015 Monterotondo, Italy*

† Rapporto tecnico 2014/21 con protocollo CNR-IC 2213 del 11/12/2014

che non ne ha diritto.

Questa operazione si ottiene filtrando gli utenti in base al servizio attivo "owncloud".

User Filter:

```
(&
  (objectclass=mailUser)
  (enabledService=owncloud)
)
```

Login Filter:

```
(&
 (&
  (objectclass=mailUser)
  (enabledService=owncloud)
 )
 (
   (mailPrimaryAddress=%uid)
   (mail=%uid)
 )
)
```

Poiché la base dati usata dal nostro sistema di posta elettronica è di tipo LDAP è stato necessario aggiungere un "enabledService" a tutti gli utenti abilitati al servizio e, in fase di login, effettuare una verifica sulla presenza dello stesso flag nella porzione di albero relativa all'utente che si sta loggando.

## 3 Configurazioni Avanzate

Di default a un utente che effettua il login tramite credenziali LDAP[1] verrà assegnato un uid (user identifier) esadecimale e i suoi file verranno memorizzati nella directory /var/www/data/uid. Per ragioni di semplicità nel management abbiamo fatto sì che il percorso dati coincidesse con la login dell'utente, modificato il campo "Attributo nome utente interno" affinché coincidesse con l'indirizzo email.

Un'altra personalizzazione è relativa al campo email associato a ogni utente. Nonostante esso sia di default assegnato al valore "mail" della struttura LDAP, fintanto che l'utente non effettua il primo login tale valore è vuoto. Come conseguenza ogni qual volta un utente condivide un file della propria area con un collega che non ha mai effettuato il login la mail di notifica non viene consegnata. Con uno script ad hoc abbiamo popolato tale tabella, inoltre lo script è stato inserito in crontab per essere eseguito ogni giorno. Si veda in appendice il listato dello script.

I file gestiti dalla piattaforma sono memorizzati su una porzione di disco offerta da una SAN Solaris montata via nfs4 con le opzioni **hard** e **initrd**. La quota è stata configurata a 2 TB.

È necessario evitare che il demone apache parta prima del nfs, questo è evitabile aggiungendo l'opzione **netdev** al file **/etc/fstab**. Inoltre, per evitare qualsiasi problema, è bene modificare il file **/etc/init.d/apache** affinché esso impedisca l'avvio del servizio qualora la share NFS non fosse montata. A tal proposito è sufficiente creare un file sulla share NFS che contiene i dati (nel nostro caso **/var/www/data** ), mediante il comando **touch we_are_on_nfs** ed eseguire prima dell'avvio del servizio una verifica sull'esistenza del file. Il seguente snippet di codice fornisce una possibile implementazione

```
if [ ! -f /var/www/data/we_are_on_nfs ]
then
    echo Missing nfs mount
    exit 1
fi
```

Di default i log vengono salvati nella directory **/var/www/data**. Ciò può portare diversi problemi di performance a causa dei numerosi accessi I/O. Una possibile soluzione è quella di spostare il file di log in altra posizione. Questa posizione è configurabile attraverso il file di configurazione con il parametro logfile.

Esempio:

$$\text{"}\mathbf{logfile}\text{"} => \text{"}/\mathbf{var}/\mathbf{log}/\mathbf{owncloud}.\mathbf{log}\text{"}$$

## 4 Online Editor

La piattaforma da anche la possibilità di condividere documenti di testo in formato odt (open document text) con l'aggiunta di un sistema online di editing, che permette quindi di accedere in modalità lettura/scrittura sullo stesso file contemporaneamente senza imbattere in conflitti di accesso e con in più la possibilità di vedere in diretta le modifiche che stanno facendo i partecipanti alla sessione di editing. Le modifiche del singolo editore vengono evidenziate con colori differenti.

## 5 Monitoraggio della Piattaforma

Tutti i server gestiti dal nostro gruppo viene tenuto sotto controllo tramite una piattaforma di monitoring basata sul software open source Zabbix[2], che si occupa di tenere sotto controllo lo stato del sistema e dei processi tramite l'uso di un agent.

La procedura di installazione dell'agent prevede 3 passi:

```
# wget http://repo.zabbix.com/zabbix/2.0/
debian/pool/main/z/zabbix-release/
zabbix-release_2.0-1wheezy_all.deb
# dpkg -i zabbix-release_2.0-1wheezy_all.deb
# apt-get update
# apt-get install zabbix-agent
```

La configurazione del template linux è sufficiente a monitorare una gran quantità di parametri. Resta da configurare il controllo sullo spazio disco della share NFS. Avendo assegnato una quota in fase di configurazione basterà interrogare il sistema operativo tramite df, il cui output è riportato in appendice.

Quindi un check con parametro

$$[\mathbf{vfs.fs.size}[/\mathbf{var}/\mathbf{www}/\mathbf{data}, \mathbf{pfree}]]$$

fornirà le informazioni richieste in zabbix.

## 6 Appendice

Script di popolazione della tabella mysql oc_preferences

```
#!/bin/bash
tmp=$(mktemp)
if [ -f $tmp ];
then
    echo "Il file con le istruzioni sql e' $tmp"
else
    echo "$tmp non e' stato creato"
    exit
fi
lista_utenti=$(ldapsearch -h *ip_ldap* -p 389 -D "cn=Manager,dc=mlib,dc=cnr,dc=it" -w *password* \
-b "o=domains,dc=mlib,dc=cnr,dc=it" enabledService=owncloud mail | grep "mail: " | cut -d " " -f 2) \
for utente in $lista_utenti
do
echo
"INSERT
IGNORE
into
oc_preferences
(userid,appid,configkey,configvalue)
VALUES
(\"$utente\", \ \"settings\",\"email\",\"$utente\");" >> $tmp
done
echo File $tmp contiene gli statement SQL
```

Output del comando df

```
# df -h /var/www/data
File system
Dim. Usati Dispon. Uso% Montato su
server_nfs:/volumes/tank/data 2,0T 47G 2,0T 3% /var/www/data
```

## Riferimenti


1 G. Nantista, G. Righini, L. Ianniello, A. Lora, A. Pifferi, Servizi DNS e DHCP con Backed LDAP in Business Continuity, SMART eLAB 1 (2013) 1–8. doi:10.30441/smart-elab.v1i0.15.

2 A. Pifferi, G. Nantista, L. Ianniello, A. Lora, M. Simonetti, Analisi e implementazione di sistemi per il monitoraggio della rete wireless relativa al progetto add (anti digital divide) e delle infrastrutture di campus adr rm1., SMART eLAB 2 (2013) 1–9. doi:10.30441/smart-elab.v2i0.46.

Networking

# Comuni tra le nuvole.†

Augusto Pifferi,[a] Giuseppe Nantista,[a] Sabina Ponzio,[a] Francesca Vergari.[a]


Con il codice dell'amministrazione digitale si è dato un impulso decisivo all'importante processo di informatizzazione e digitalizzazione della pubblica amministrazione avviato anni fa ma mai portato a termine. Siamo, ora, in una fase importante del processo, fase che vede come attori principali gli enti locali coinvolti nei progetti di innovazione avviati. In questo rapporto tecnico viene descritta una piattaforma, nella logica del "cloud", per la gestione di uno sportello SUAP (Sportello Unico per le Attività Produttive e SUE (Sportello Unico per l'Edilizia) e per i servizi al cittadino a domanda individuale. La piattaforma contiene inoltre un SIT (Sistema Informativo Territoriale) che oltre ad essere la base per la geolocalizzazione dei dati può essere usato come strumento per la correlazione degli stessi al fine di estrapolare informazioni utili agli amministratori per il contrasto all'evasione ed elusione fiscale. Il progetto è stato finanziato dall'Unione di Comuni della Bassa Sabina per l'importo di 85.000 euro.

**Keywords**: Codice Amministrazione Digitale, SUAP, SUE, SIT, Open Source, Business Intelligence

## 1 Introduzione ed Obiettivi

La caratteristica del progetto è quella di permettere alle Amministrazioni Comunali che date le loro ridotte dimensioni in termini di popolazione, di risorse finanziare e tecnico-professionali, vogliano attraverso l'aggregazione di più comuni accedere a strumenti tecnologici e organizzativi tali da offrire i servizi di una piattaforma Enterprise ai propri cittadini e imprese del territorio.

Tutto ciò per dimostrare capacità organizzativa e funzionale per intraprendere un percorso sostenibile di innovazione e servizi, da attuarsi attraverso una gestione associata dei servizi sempre più ampia, integrata, efficiente ed efficace, divenuta ormai obiettivo importante del governo e soprattutto imprescindibile dei piccoli comuni italiani.

Obiettivo tecnologico prioritario è quello di sviluppare una piattaforma in grado di prefigurare uno sviluppo dei servizi informativi verso il cloud computing, che rappresenta una tendenza evolutiva dell'outsourcing che si fonda su i seguenti principi: Focalizzare l'attenzione su ciò che le tecnologie consentono di realizzare, piuttosto che sulle tecnologie in sé, con il risultato che il software non è più un asset gestito in proprio dall'utilizzatore ma è quest'ultimo che sceglie i servizi di cui usufruire, in base alle proprie reali necessità.

Questo ci sembra essere il modello che meglio si addice ai Comuni e/o Enti di piccole e medie dimensioni che potranno meglio utilizzare le proprie risorse economiche, focalizzandole sul servizio e non sulle infrastrutture.

La gestione associata, come da dettato legislativo, dovrà interessare i più importati e strategici servizi Comunali, garantendo:

- costante miglioramento della loro qualità e quantità;
- razionalizzazione delle spese;
- incremento delle entrate attraverso la lotta all'evasione/elusione fiscale anche con forme di cooperazione tra i vari livelli di governo;
- valorizzazione e formazione del personale interno;
- utilizzo sempre più spinto delle nuove tecnologie informatiche.

Queste problematiche vanno affrontate con una visione che tenga conto principalmente dei portatori di interesse più significativi del territorio.

Partire dalle esigenze dei Comuni e dei Portatori di Interesse è da ritenersi un fattore strategico per la diffusio-

[a] *Istituto di Cristallografia, C.N.R. via Salaria km 29.300, 00015 Monterotondo, Italy*



† Rapporto 2014/20 con Protocollo CNR-IC 2145 del 03/12/2014

ne dell'innovazione, avviando un percorso di riorganizzazione della pubblica amministrazione con una visione nuova e moderna.

Investire risorse nella formazione e nella comunicazione prestando sempre maggiore attenzione alle esigenze del cittadino, utilizzando le informazioni come indicatori fondamentali per una revisione periodica delle strategie in conseguenza dei cambiamenti esterni.

In sintesi una buona amministrazione è quella che sa ascoltare le esigenze dei cittadini e riesce a dare loro risposte concrete.

Il progetto pone le sue basi sull'aggregazione dei seguenti servizi:

**Gestione dei Servizi a Domanda Individuale**

- Gestione Rette Scolastiche
- Gestione dei Trasporti
- Gestione Asili Nido

**Gestione dei Servizi associati**

- Comunicazione e Corrispondenza
- Flusso degli incassi e dei pagamenti
- Monitoraggio dei servizi

**Gestione della Rete di Sportelli**

- Sportello Unico per l'Edilizia
- Sportello Unico Attività Produttive
- Reti Amiche

**Gestione del Territori**

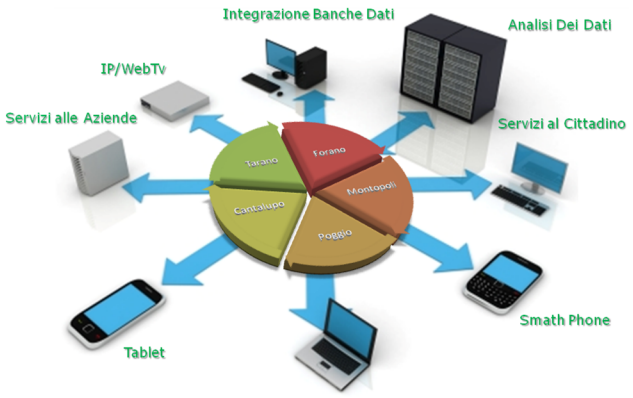
- Sistema Informativo Territoriale
- Gestione del Personale
- Gestione centralizzata delle Paghe

**Connettività e centralizzazione delle risorse**

- Connettività a banda larga
- Server Farm

**I vantaggi del modello sono rappresentati da:**

- Riorganizzazione dei servizi verso un processo di semplificazione e riduzione di inefficienze
- Riduzione dei Costi diretti ed indiretti sui servizi (es. riduzione dei costi di fornitura derivanti dall'aumento di volumi rappresentati dall'unione e non dal singolo ente, liberare risorse umane per la semplificazione del servizio)
- Aumento degli strumenti di controllo sui flussi di incasso, pagamento e accertamento
- Semplificazione dei servizi per il cittadino ed aumento della loro disponibilità di accesso (pagamento online, URP virtuale)
- L'ampliamento delle possibilità di accesso ai servizi erogati ai cittadini attraverso l' ntegrazione di sportelli fisici e sportelli virtuali usufruendo delle Reti Amiche.
- La riorganizzazione del personale interno alla PAL.
- La digitalizzazione dei procedimenti amministrativi
- L'utilizzo di applicazioni informatiche avanzate con tecnologia ASP - Application Service Provider orientata ai Web Services
- Creazione di una Banca Dati Centralizzata che ne aumenta l'efficacia dell'informazione diminuendo i costi di gestione
- La realizzazione di eventuali infrastrutture di rete wireless per la diffusione della "larga banda" sul territorio Comunale




## 2 Architettura della Soluzione

L'architettura qui rappresentata descrive la soluzione nella sua interezza, ma all'interno sono previste le integrazione a molteplici sottosistemi.

### 2.1 Funzionalità della Piattaforma Base del Progetto

La piattaforma base è corredata dalle seguenti funzionalità trasversali all'attivazione di qualsiasi servizio:

- Sicurezza
- Caricamento Dati
- Front-End
- Analisi e reportistica dei Dati
- Sistema di Pagamento
- Sistemi di Notifica

### 2.2 Gestione dei Servizi a Domanda Individuale

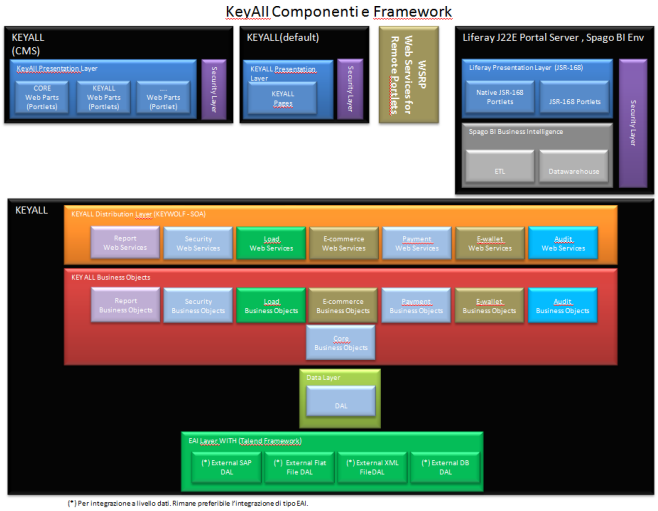
Osmosit KEY REF è la soluzione per la gestione del servizi a domanda Individuale mense di scuole, trasporti e Asilo Nido.

Uno strumento pensato per i gestori delle forniture mense e per gli Enti pubblici e/o aziende destinatarie del servizio.

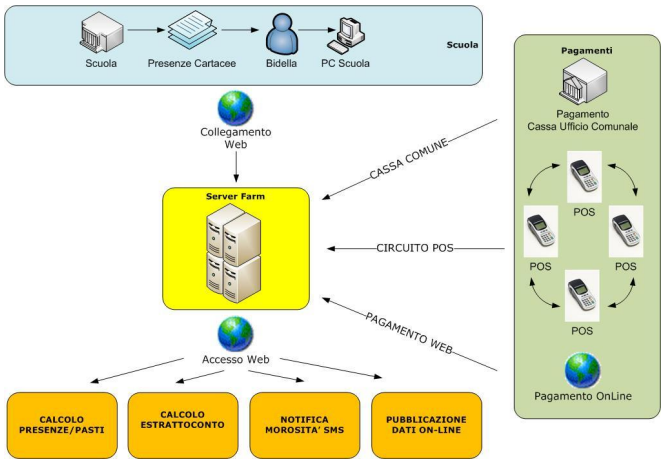
Permette:

- All'Ente/Azienda un controllo costante dei costi e delle presenze del servizio, con un sistema di pagamento integrato che agevola l'incasso e la riscossione del servizio;

- Al Fornitore del servizio di offrire all'ente/azienda un sistema di controllo e gestione accurato e efficiente;



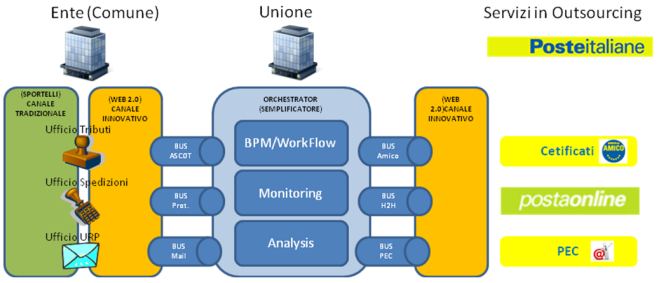

Il sottoscrittore del servizio mensa inoltre, può accedere al proprio profilo per verificare le presenze, i pagamenti e i crediti nei confronti di chi offre il servizio.




## 2.3 Gestione dei Servizi associati

Nexus KEY ALL è una piattaforma multicanale di accesso, gestione, controllo del sistema di:

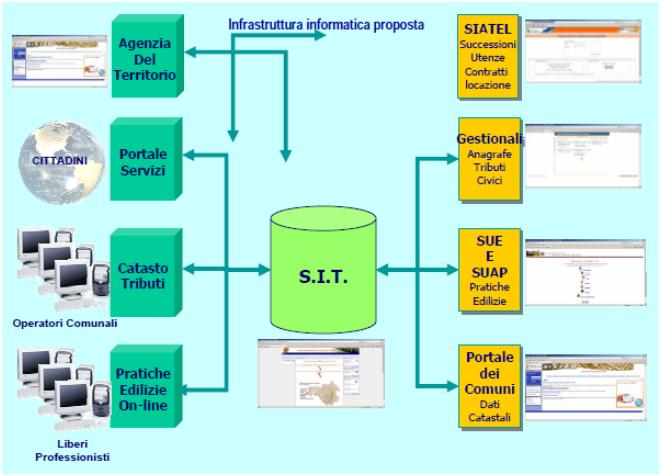
- Comunicazione/Corrispondenza
- Semplificare i rapporti tra cittadino e la pubblica amministrazione, intesi come, semplificare le comunicazioni (pec), le richieste (certificati tramite reti amiche), i pagamenti (pagamenti online);
- Ottimizzare e accertare la spesa pubblica: centralizzando la gestione degli incassi, anticipando i pagamenti, moltiplicando i punti di accesso ai pagamenti e ai dati;
- Garantire la qualità dei servizi e la trasparenza verso il cittadino;
- Moltiplicare i canali di accesso del cittadino verso la PA per tutti i servizi dell'ente: documentale, informativo o dei pagamenti.




## 2.4 Gestione della Rete di Sportelli

L'infrastruttura informatica proposta è la seguente:

- Mantenimento dei gestionali Anagrafe e Tributi già presenti nei Comuni
- Adozione applicazione informatica per Sportello Unico per l'Edilizia -SUE
- Adozione applicazione informatica per Sportello Unico Attività Produttive - SUAP
- Adozione Sistema Informativo Territoriale per integrazione banche dati - SIT
- Erogazione servizi on-line




## 2.5 Integrazione Rete Amica

La proposta Reti Amiche da l'opportuità di realizzare un partenariato tra parti diverse (soggetti pubblici o privati, forze economiche e sociali) per la realizzazione di interventi finalizzati al miglioramento della vita quotidiana del cittadino che in diversi momenti della sua giornata si trova ad interloquire con la Pubblica Amministrazione.

L'obiettivo di valorizzare al massimo l'infrastruttura tecnologica dei servizi esistenti facendo dell'integrazione e cooperazione delle piattaforme la strategia di ampliamento e miglioramento dei servizi erogabili al cittadino da parte dell'Ente, ottenendo allo stesso tempo un'ottimizzazione dei Costi aumentando la disponibilità del servizio.

## 2.6 I Vantaggi

- Riduzione delle Code agli sportelli, con diminuzione del flusso cittadini presso gli uffici più in sovraccarico, con possibilità di liberare risorse verso funzioni più strategiche.

- Aumentare la disponibilità del servizio sia in termini di orario che di sportelli e questo soprattutto per quelle tipologie di Comuni Piccoli, Comuni di Montagna o Comuni con molte frazioni.

- Riduzione della circolazione del Cartaceo, con conseguente riduzione di spostamenti dei cittadini ed anche relativa diminuzione di fattori inquinanti o di rischio

- L'introduzione del servizio in multicanalità su una piattaforma che poi permetta l'attivazione di N servizi, utilizzando sempre le stesse metodologie di accesso per il Cittadino


## 2.7 Gestione del Territorio

Esc CityExplorer sfrutta le più recenti tecnologie per consentire un flusso di informazioni in tutta l'organizzazione dell'amministrazione.

Il database territoriale, che raccoglie e struttura le informazioni di vari settori, viene reso disponibile attraverso la rete Intranet/Internet dell'ente. I vari utenti accedono al sistema tramite interfacce ed applicazioni personalizzate che rispecchiano le diverse necessità di consultazione, analisi e modifica del dato stesso.

Il progetto che ne deriva risulta quindi un Sistema Integrato e non dedicato esclusivamente a singoli processi (Ambientale, Urbanistico, ecc.). Ogni processo che compone l'attività dell'amministrazione viene affrontato pertanto con applicazioni dedicate che estendono i pacchetti software attualmente in possesso verso l'uso del dato geografico.

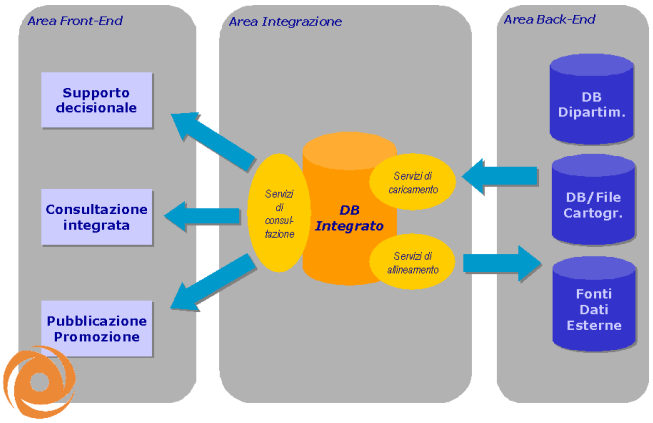
L'accesso alla banca dati avviene attraverso la rete Intranet/Internet dell'ente.

Per la pubblicazione in Internet di tali dati è inoltre possibile prevedere una cooperazione tra l'applicazione WEB-GIS e il portale/sito dell'ente.

## 2.8 Connettività e centralizzazione delle risorse

Un complesso di servizi come quelli presentati necessita di risorse hardware e connettività adeguate al fine di garantire:

- velocità di accesso,
- continuità del servizio,
- sicurezza fisica dei sistemi,
- sicurezza dei dati,
- ridondanza degli apparati per un rapido disaster recovery
- capacità di storage.




E' quindi indispensabile, per l'erogazione ottimale dei servizi offerti dalla piattaforma, che i sistemi siano centralizzati in una server farm remota in grado di ospitare su sistemi ad alta prestazione e ridondati il software di gestione. Per usare un termine oggi "up to date" realizzare un sistema di cloud computing ove i client potranno accedere in modalità web al server centralizzato. Non è da escludere che in una seconda fase i server potranno essere clonati e dislocati in farm diverse per una migliore garanzia di continuità ed affidabilità del servizio.

Nella Server Farm del Consiglio Nazionale delle Ricerche presso l'Area della Ricerca RM1 di Montelibretti saranno operative le macchine sulle quali verranno installati i servizi elencati. I dati locali e copia dei dati remoti saranno conservati in unità di Storage di alta capacità con dischi ridondati in modalità raid 5.

Il secondo aspetto fondamentale per l'utilizzo della piattaforma sono i collegamenti di rete che devono avere caratteristiche di velocità adeguate. Ove fisicamente raggiungibili dai trasmettitori wireless del Consiglio Nazionale delle Ricerche nell'AdR RM 1 di Montelibretti, i comuni potranno dotarsi di una linea di comunicazione verso il Centro di gestione. Questa linea potrà essere utilizzata sia per i collegamenti verso il mondo internet che, per mezzo di una Virtual Private Network (VPN), per i collegamenti dei client dei Comuni e dell'Unione di Comuni verso la piattaforma. In alternativa potranno essere utilizzate linee dati di gestori terzi con adeguate specifiche di performance soprattutto in upload al fine

di trasmettere i file sulla data storage remota in tempi rapidi.

## 3 Funzionalità

### 3.1 Gestione Funzionalità della Piattaforma Base del Progetto

La piattaforma base è corredata dalle seguenti funzionalità trasversali all'attivazione di qualsiasi servizio:

**Sicurezza**

- Sistema di SSO e Profilatura Utenti/Ruoli/Funzioni
- Gestione anagrafica e sistemi di Registrazione
- Modalità di Autenticazione con CIE-CNS-CRS

**Caricamento Dati**

- Gestione ETL/Carichi Banche Dati
- Gestione dei Processi

**Front-End**

- Estratto Conto Cittadino
- Consultazione posizione
- Consultazione Storico
- Comunicazione Ente/Cittadino

**Analisi e reportistica dei Dati**

- Gestione Rendicontazione
- Gestione posizione utenti
- Cruscotto Monitoraggio

**Sistema di Pagamento**

- Carrello Dei Pagamenti (Entrate, Bollettino Premarcato, Bollettino Bianco)
- Estratto Conto Cittadino
- Pagamenti OnLine (Porta dei pagamenti, Gateway Bancari)
- Gestione crediti e rendicontazioni

**Sistemi di Notifica**

- via SMS
- via WEB
- via APP-SmartPhone

### 3.2 Gestione dei Servizi a Domanda Individuale

KEY REF permette di gestire ogni fase del processo dei servizi a domanda individuale, mense, trasporti scolastici e asili nido:

- L'iscrizione al servizio: con il rilascio di una tessera di riconoscimento univoca, e la profilatura dell'utente per tipo di esenzione;
- Il sistema di pagamento: completamente integrato al sistema, può essere reso a mezzo Bollettino Postale; lo scambio di tracciati record integrato con i sistemi di Poste Italiane permette l'allineamento in tempo reale dei dati;
- Gestione Crediti: l'applicazione permette il controllo dei crediti degli utenti e l'eventuale notifica in SMS di fine credito con avviso o sollecito;
- Gestione Prenotazioni e Presenze ed utilizzo del servizio;
- Gestione Report: l'applicazione permette il controllo, fruizione del servizio e dei crediti e i pagamenti al fine della consuntivazione del servizio per tutti gli attori coinvolti nel processo.

### 3.3 Gestione del territorio

L'integrazione in un unico ambiente delle diverse basi dati e la creazione delle relative chiavi di collegamento consentono di "navigare" attraverso i diversi archivi: l'utente può infatti ricercare per via grafica o alfanumerica un numero civico, visualizzare la posizione in cartografia e consultare le informazioni di dettaglio, quindi visualizzare l'elenco dei soggetti con residenza sul numero civico, verificare quale soggetto residente è anche intestatario al catasto e quali sono gli immobili intestati, se e su quali immobili versa il corrispondente tributo e così via.

Il software mette a disposizione dell'utente funzionalità espressamente dedicate all'interrogazione ed analisi dell'informazione catastale ed urbanistica, fra cui:

- certificato di destinazione urbanistica: ottenibile in diversi formati (PDF, RFT) corredato da estratti di mappa e norme tecniche;
- analisi di destinazione urbanistica;
- piani particellari;
- localizzazione e disegno del file Pregeo;
- visure catastali per intestatari ed immobili;
- ricerca per via cartografica ed alfanumerica;
- esportazione delle informazioni in diversi formati (xls, csv, dxf, shp);
- reportistica di dettaglio e sintesi;
- analisi statistiche;
- funzioni di misura (area e perimetro);
- funzioni di disegno (punti, linee, poligoni, testi, ecc.) con possibilità di esportazione degli oggetti;
- funzioni di buffering;
- funzioni di interrogazione di modelli numerici del terreno (quote, livelli, sezioni);
- funzioni di stampa.

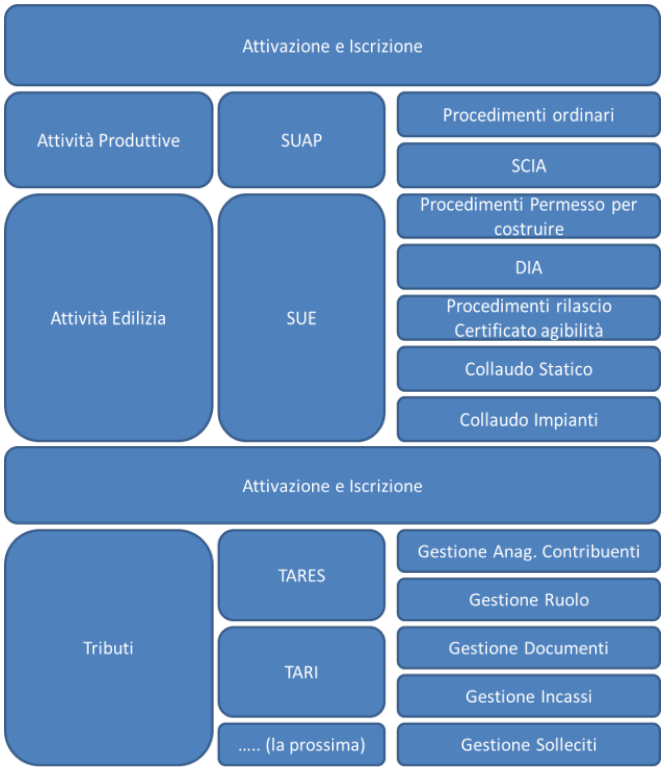
## 4 Fasi di Sviluppo del Progetto

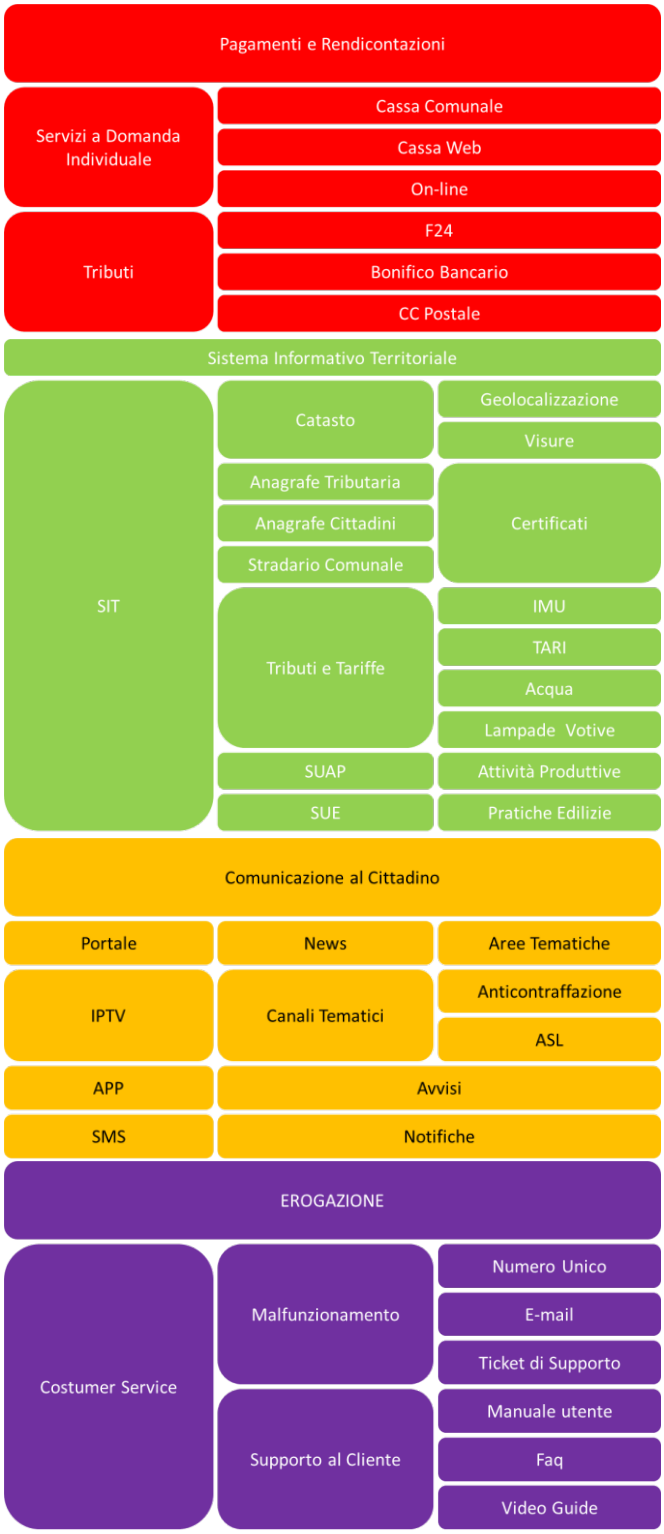
a) Raccolta dei Requisiti
b) Analisi e Progettazione Esecutiva
c) Installazione, Configurazione Personalizzazione
d) Integrazione delle Banche Dati
e) Integrazione Con Servizi Terzi
f) Collaudo
g) Formazione
h) Attivazione dei Servizi e degli sportelli

i) Supporto Operativo

## 5 I Servizi Attivati


Nella tavola sono indicati i servizi attualmente gestiti



## 6 Conclusioni

Dei 6 comuni aderenti all'Unione di Comuni della Bassa Sabina non tutti hanno attivato tutte le funzionalità della piattaforma. Il processo di adeguamento ai nuovi sistemi digitalizzati è lento e non privo di difficoltà soprattutto per il diverso approccio che necessita di una adeguata formazione degli operatori che andranno ad utilizzare il servizio.

Nella tabella sottostante sono illustrati i dati indicativi della gestione di tre comuni in un anno di utilizzo.

| Indicatori | Servizio | Dettaglio | Valore |
|---|---|---|---|
| POPOLAZIONE | Totale | Anagrafiche | 6.371 |
| | Tributi | Contribuenti | 3.302 |
| | Servizi a Domanda Individuale | Mensa - Genitori ed Alunni | 1.727 |
| Pagamenti e Incassi | TARES | Dovuto | € 1.116.295,00 |
| | | Incassato | € 832.963,00 |
| | | N. Pagamenti | 8.575 |
| | | N. Bollettini | 3.254 |
| | | F24 | 3.245 |
| | | Sollecitato | € 259.918,00 |

In futuro è prevista la realizzazione di un unico sistema in grado di raccogliere i dati originati dalle varie applicazioni fiscali e di elaborarli direttamente senza un preprocessamento manuale.

Questo progetto è stato realizzato in collaborazione con l'Azienda Nexus srl con la quale è stata formalizzata una Associazione Temporanea di Scopo.

Networking

# Autenticazione tramite social network per l'accesso a hotspot pubblici gratuiti.†

Giuseppe Nantista,[a] Andrea Lora,[a] Augusto Pifferi.[a]


In questo rapporto tecnico studieremo la possibilità di autenticare gli utenti di un hotspot pubblico gratuito tramite gli account che essi hanno registrato sui principali social network. Il caso preso in esame fa uso delle API di Facebook per accedere a un hotspot realizzato con hardware Mikrotik, tuttavia il principio alla base del funzionamento è generalizzabile ad altre piattaforme hardware/software e ad altri network che forniscano API di autenticazione. Descriveremo i principi alla base della soluzione realizzata e le limitazioni che essa presenta in termini di effettiva identificazione dell'utente e navigazione anonima.
**Keywords**: Social hotspot, Facebook, Mikrotik

## 1 Introduzione

Quando un utente accede a un hotspot pubblico gratuito il fornitore del servizio deve tutelarsi dalle azioni che l'utente compirà, tenendo traccia di tutte le connessioni effettuate e riservandosi la possibilità di ricondurre ogni singola connessione a un individuo. Il metodo più diffuso in questi casi consiste nell'inviare un SMS sul cellulare dell'utente contenente username e password di accesso all'hotspot, associando così le connessioni a un numero di cellulare che, in caso di contestazioni da parte delle autorità giudiziarie, possa essere usato come riferimento univoco all'individuo che ha effettuato le connessioni incriminate. Tale meccanismo prevede da parte dell'utente il fastidio di memorizzare nuove credenziali di accesso al servizio e da parte del fornitore un costo (l'invio degli SMS) non desiderato laddove si offre un servizio gratuito. In altri casi si preferisce identificare *de visu* l'utente tramite l'esibizione di un documento di identità e di conseguenza si provvede ad emettere un ticket individuale con le credenziali di accesso, tuttavia questo caso comporta l'accesso a informazioni riservate e la necessità di impegnare una persona in questa attività.

Demandare a terzi l'autenticazione dell'individuo comporta il già citato vantaggio da parte dell'utente di non dover memorizzare le ennesime credenziali di accesso a un servizio, liberando il fornitore del servizio dagli oneri e dalle responsabilità correlate all'erogazione del servizio stesso.

Questo documento vuole indicare una possibile soluzione mediante l'impiego di un hotspot gateway Mikrotik, di un server web abilitato PHP e di un account sviluppatore Facebook.

## 2 Operazioni Preliminari

Il primo passo da compiere è la registrazione di un account da sviluppatore di Facebook, è sufficiente accedere con il proprio account all'indirizzo https://developers.facebook.com/ e creare la prima applicazione di tipo "website". Verrà assegnato un "app- id" alla nostra applicazione, della quale dovremo anche specificare l'url alla quale verrà ospitata, opzionalmente è possibile specificare una url per applicazioni mobile. Per iniziare Facebook mette a disposizione un tutorial reperibile nell'area per sviluppatori. [1]

Le applicazioni sono di uso generale, a noi in questa fase interessa la funzione di login, ovvero sia di validazione delle credenziali di accesso, da inserire nella nostra pagina di autenticazione dell'hotspot.

A tal proposito abbiamo configurato una routerboard Mikrotik per svolgere le funzioni di Hotspot e sostituito la pagina di login di default con una che rimanda a una pagina di autenticazione esterna secondo quanto indicato nella wiki di Mikrotik stessa. [2] Nell'hotspot così creato andranno aggiunte alcune destinazioni nel Walled Garden, la raccolta di tutte le destinazioni che un utente non ancora autenticato può visitare; innanzitutto l'indirizzo del nostro web server di supporto e poi due indirizzi generici per raggiungere le pagine di autenticazione di facebook:

```
*facebook*
*akamai*
```

Nella pagina va personalizzato il link di redirect, che dovrà puntare al nostro server web, abilitato PHP, il cui indirizzo coinciderà con quello specificato nel "site url" della applicazione Facebook, in caso contrario otterremo un errore di indirizzo di provenienza errato.

Ora è necessario, affinché un utente possa autenticarsi e navigare, seguire i seguenti passi:

1. L'utente clicca sul pulsante "login with Facebook";

[a] *Istituto di Cristallografia, C.N.R. via Salaria km 29.300, 00015 Monterotondo, Italy*

† apporto tecnico 2015/06 prot. CNR-IC 815 del 24/04/2015

2. Accede alla pagina di login del social network con riferimento alla app in questione, si autentica e autorizza l'applicazione ad accedere alle sue informazioni di base, profilo e email;
3. Ottiene da Facebook un "token" che garantisce che egli si sia correttamente autenticato;
4. Invia questo token al web server che effettua la verifica di validità del token sul sito di Facebook;
5. Se la verifica va a buon fine la routerboard Mikrotik viene istruita a creare una utenza temporanea con una password generata casualmente;
6. Il web server invia una istruzione di redirect verso la pagina di login, inserendo nella url username e password dell'utenza appena creata, cosicché l'utente risulti già in sessione.

Riassumiamo visivamente il flusso di informazioni appena descritto in figura 1.

L'hotspot Mikrotik Mikrotik mette a disposizione un meccanismo guidato per la creazione di un hotspot, che include anche le regole firewall di redirect in caso di utente non autenticato e le eccezioni relative al walled garden. Il redirect iniziale viene rimandato all'url

```
http://ip-dell-hotspot-mikrotik/login
```

La funzione login è hard-coded e invoca l'apertura del file login.html contenuto nella directory radice dell'hotspot; alla funzione login vengono passati alcuni parametri come ad esempio il mac-address e l'ip del client che ha tentato la connessione nonché l'url a cui egli intendeva accedere prima di essere "catturato". Sostituendo il file login.html con quello indicato in [2] è possibile rimandare l'utente a una pagina di login esterna.

Quando, ad autenticazione completata, il web server crea una utenza sull'hotspot Mikrotik fa uso delle API PHP RouterOS, scaricabili liberamente all'indirizzo [3].

Queste funzioni vengono eseguite con una utenza che deve essere preventivamente creata e abilitata all'interno del router con i seguenti comandi:

```
ip service set api disabled=no
/user group add name=gapi policy=api,read,write
/user add name=uapi password=apipasswd group=gapi
```

## 3 Il web server di supporto

Per realizzare l'infrastruttura proposta in questo progetto è indispensabile un server web con funzionalità PHP abilitate, che interviene in due fasi dell'attività di login dell'utente.

1. *La presentazione della pagina di login*
   La pagina di per se utilizza linguaggio javascript per caricare il pulsante di login, tuttavia l'uso di PHP è fondamentale per "portarsi dietro" i parametri relativi all'utente guest che fa richiesta di accesso e alla url richiesta. In appendice riportiamo una versione semplificata di tale pagina.
2. *L'immissione delle credenziali di accesso*
   Quando l'utente effettua il login su Facebook e ottiene il token relativo alla sua sessione lo comunica al web server, che tramite la pagina "confirm.php", anch'essa riportata in appendice, verifica la validità del token, crea l'utente sull'hotpost e infine rimanda l'utente alla pagina di login immettendo per suo conto i dati di accesso e la url di destinazione.

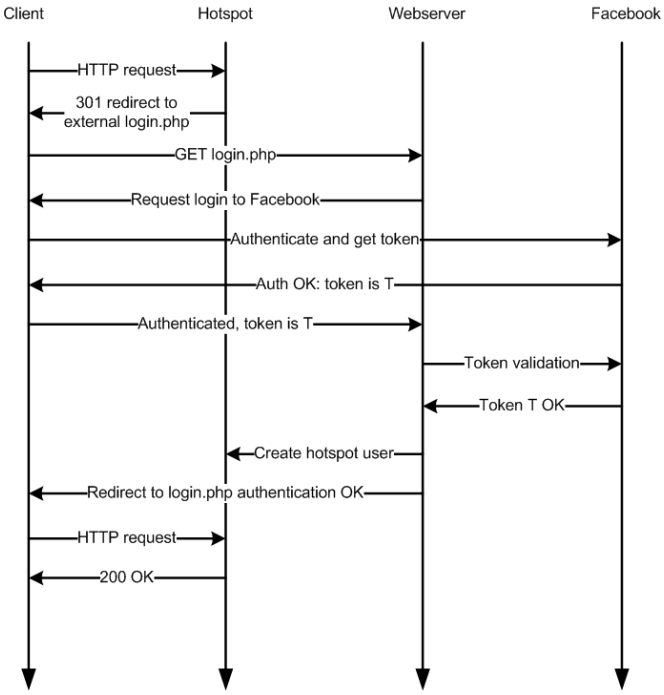



**Fig. 1** Diagramma di flusso del processo di autenticazione

## 4 Affidabilità del sistema

Facebook tiene traccia di un certo numero di domini di posta elettronica che non garantiscono la vera identità di un utente, ad esempio quei domini di "disposable mailbox", tuttavia è possibile raggirare il meccanismo di verifica e creare un account fasullo così da poterlo sfruttare in una circostanza come l'accesso anonimo a una rete, tuttavia riteniamo che il livello di confidenzialità sia sufficiente per la normativa italiana relativa alla tracciabilità delle connessioni degli utenti.

Di recente è stato introdotto il concetto di verifica dell'account, che consente di aggiungere fra i dati personali oltre all'indirizzo email usato per registrare l'account anche un numero di cellulare, il che ricondurrebbe la tracciabilità dell'account all'operatore telefonico che ha rilasciato la scheda sim in questione. In questo senso quindi è possibile consentire l'accesso alla rete solo a quegli utenti che hanno un account verificato da Facebook; il valore in questione viene fornito da Facebook senza che l'utente conceda ulteriori autorizzazioni, il suo nome è "verified" ed è di tipo boolean.

Un'altra limitazione già indicata è la possibilità per un utente non ancora autenticato di navigare all'interno del sito Facebook perché consentito dal walled garden.

Se l'account email dell'utente non è stato confermato Facebook potrebbe anche consentire l'accesso al suo network, tuttavia al momento in cui l'applicazione chiede conferma a Facebook il valore "email" non viene restituito, rendendo di fatto impossibile la creazione dell'utenza temporanea, che usa come username proprio l'indirizzo email dell'utente.

## 5 Conclusioni

La questione discussa in questo documento tecnico rientra nel discorso della "Open Authentication", una tematica che ha visto la luce nel 2006, quando fu elaborato da Blaine Cook e Chris Messina il primo protocollo di autenticazione aperta, dal nome oauth, pubblicato da Hammer e Lahav nel 2010 come

RFC 5849[4] e recentemente reso obsoleto dal RFC 6749.[5]

L'idea di base[6] è quella di autorizzare terze parti a gestire documenti privati senza condividere la password. La condivisione della password infatti presenta molti limiti a livello di sicurezza, come ad esempio non garantisce supporto per singoli privilegi su determinati file o operazioni, ma rende accessibile l'intero account e il pannello di amministrazione. Inoltre vige l'impossibilità di revocare l'accesso nel futuro se non cambiando la password dell'intero account. OAuth è nato quindi con il presupposto di garantire l'accesso delegato ad un client specifico per determinate risorse sul server per un tempo limitato, con possibilità di revoca.

OAuth presenta alcune limitazioni a livello di sicurezza, il server infatti raccoglierà informazioni riguardanti l'utente, il client e la loro interazione e le terrà in memoria per un limitato arco di tempo, il protocollo inoltre non garantisce confidenzialità né sulle richieste effettuate, né sui contenuti scambiati, ad esempio non garantisce che l'uso delle risorse autorizzate rimanga nell'ambito specificato. Per questo motivo OAuth suggerisce al server di proteggere le risorse tramite il protocollo TLS.

Un ulteriore problema noto è quello del phising. Il client potrebbe indirizzare l'utente and una pagina di accesso fasulla del server per richiedere l'autenticazione e ottenere le credenziali dell'utente.

## 6 Appendice

Il file login.php

```
    $ip=$_POST['ip'];
    $username=$_POST['username'];
    $linklogin=$_POST['link-login'];
    $linkorig=$_POST['link-orig'];
    $error=$_POST['error'];
    $chapid=$_POST['chap-id'];
    $chapchallenge=$_POST['chap-challenge'];
    $linkloginonly=$_POST['link-login-only'];
    $linkorigesc=$_POST['link-orig-esc'];
    $macesc=$_POST['mac-esc'];
?>

..Omissis...

<script>
//statusChangeCallback viene invocata da checkLoginState
    function statusChangeCallback(response) {
        if (response.status === 'connected') {
            mikAPI(response);
        } else if (response.status === 'not_authorized') {
            } else {
        }
    }

// checkLoginState è la prima funzione chiamata dal pulsante di login
    function checkLoginState() {
        FB.getLoginStatus(function(response) {
            statusChangeCallback(response);
        });
    }

    window.fbAsyncInit = function() {
    FB.init({
    appId   : 'app-id',
    cookie  : true,
    xfbml   : true,
    version : 'v2.2'
    });

    FB.getLoginStatus(function(response) {
        statusChangeCallback(response);
            });
    };

    (function(d, s, id) {
    var js, fjs = d.getElementsByTagName(s)[0];
```

```
    if (d.getElementById(id)) return;
    js = d.createElement(s); js.id = id;
    js.src = "//connect.facebook.net/en_US/sdk.js";
    fjs.parentNode.insertBefore(js, fjs);
    }(document, 'script', 'facebook-jssdk'));

// mikAPI è la funzione creata da noi che chiede il token e lo passa alla pagina confirm.php
    function mikAPI(myResp) {
        console.log('Welcome! Fetching your information.... ');
        FB.api('/me', function(response) {
                                            window.location.replace("http://ip-server-web/confirm.php?token="+
myResp.authResponse.accessToken+"&dst="+"<?php echo $linkorig; ?>");
    });
    }
</script>

...Omissis...

<fb:login-button scope="public_profile,email" size="xlarge" onlogin="checkLoginState();"> </fb:login-button>

...Omissis...
```

Il file confirm.php

```
<?php
    require('routeros_api.class.php');

    $token = $_GET["token"];
    $dst = $_GET["dst"];
    $linkFb = "https://graph.facebook.com/v2.2/me?access_token=".$token."
&fields=id%2Cname%2Cemail&format=json&locale=it_IT&method=get&pretty=0&suppress_http_code=1";
    $result = json_decode(file_get_contents($linkFb),true);
    $email = $result["email"];
    $passwd = substr(str_shuffle('abcdefghijklmnopqrstuvwxyz0123456789'),0,8);
    $nome = $result["name"];
    echo "Benvenuto ".$nome;

    if($email) {
        $API = new routeros_api();
        if ($API->connect('mikrotik-hotspot-ip', 'uapi', 'apipasswd')) {
            $API->comm("/ip/hotspot/user/add", array (
                "name" => $email,
                "profile" => "facebook-prof",
                "limit-uptime" => "00:60:00",
                "password" => $passwd,
            ));
            $API->disconnect();
        }
    }
    header("Location: http://hotspot.local/login?username=$email&password=$passwd&dst=$dst");
    die();
?>
```

## Riferimenti


1 https://developers.facebook.com/docs/facebook-login/login-flow-for-web/v2.3.

2 http://wiki.mikrotik.com/wiki/HotSpot_external_login_page.

3 http://wiki.mikrotik.com/wiki/API_PHP_class.

4 E. Hammer-Lahav, The OAuth 1.0 Protocol, April 2010 https://tools.ietf.org/html/rfc5849.

5 D. Hardt, The OAuth 2.0 Authorization Framework, October 2012 https://tools.ietf.org/html/rfc6749.

6 http://it.wikipedia.org/wiki/OAuth.

E-Learning & Scientific Communication

# Progetto Romaforma: Interventi formativi in modalità blended a favore dei dipendenti capitolini.†

Guido Righini,[a] Luca Ianniello,[b] Mirella Rondinelli,[c] Augusto Pifferi.[b]

Il progetto Romaforma, realizzazione di una piattaforma informatica di formazione dei dipendenti della PA "Roma Capitale", ha consentito di sperimentare una configurazione tecnologica non convenzionale mirata alle Alte Prestazioni ed all'Alta Affidabilità del servizio. Il presente rapporto descrive l'infrastruttura tecnologica adottata. La piattaforma, denominata Romaforma ha consentito a circa trecento dipendenti di Roma Capitale di seguire i corsi di formazione presso le diverse sedi della PA durante l'orario lavorativo nei tempi previsti e senza interruzioni tecniche.
**Keywords**: Formazione in modalità blended, Moodle, Alta Affidabilità.

## 1 Introduzione

Nel maggio 2012 l'amministrazione pubblica **Roma Capitale** e l'associazione **Pianeta Formazione** hanno affidato al gruppo di lavoro "Smart eLab" dell'Istituto di Cristallografia del C.N.R. l'incarico di realizzare una piattaforma informatica dedicata alla formazione in modalità mista (frontale e on-line) per i dipendenti capitolini. Nel presente articolo saranno descritte le specifiche richieste e le soluzioni tecniche adottate per la realizzazione della piattaforma informatica Romaforma.

## 2 L'intervento Formativo "Media e Comunicazione Pubblica"

L'amministrazione Capitolina ha affidato all'Associazione Pianeta Formazione la realizzazione e l'esecuzione di un intervento formativo a favore dei dipendenti capitolini. Destinatari dell'intervento formativo sono stati gli operatori URP, i redattori web e gli operatori della comunicazione pubblica. Nello specifico i corsi erano mirati ai seguenti argomenti:

- Cultura generale degli strumenti comunicativi dell'essere umano, impronta teorica sulle recenti teorie della comunicazione fino ad arrivare alle prerogative fissate dalla Legge 150;
- Interventi formativi diretti a ridurre il divario culturale delle conoscenze tecnologiche, "Digital Divide";
- La comunicazione web 2.0, teoria e pratica degli strumenti interattivi web;
- Scrittura e linguaggi nella comunicazione web.

Una delle specifiche dell'intervento formativo è la suddivisione della didattica nelle modalità frontale in aula e modalità on-line con strumenti web 2.0. Per entrambe le modalità erano previste lezioni alternate a esercitazioni, simulazioni pratiche e lavori su casi reali. Questa modalità di intervento formativo viene definita *blended*. Per la realizzazione di corsi in modalità blended è fondamentale disporre di una piattaforma informatica con cui coordinare le attività didattiche in aula e in rete.

Altra specifica tecnica richiesta è la fruizione delle attività didattiche on-line presso le sedi di lavoro del dipendente con la certificazione dei periodi temporali dedicati. I dipendenti capitolini partecipanti al corso appartengono a diverse sedi distribuite sul territorio comunale connesse tra loro dalla rete *intranet* che a sua volta è connessa alla rete *internet*. I server con il software di gestione dei corsi sono stati collocati presso il centro di calcolo dell'Area di Ricerca di Roma 1 del CNR ove la connettività ad internet è elevata con garanzia di alta prestazione e alta affidabilità del servizio.

I corsi realizzati sono stati i seguenti:

- **Scrittura e Linguaggi nella Comunicazione Web**; numero di partecipanti: 96;

[a] *Istituto di Struttura della Materia - C.N.R., via Salaria km 29,3, 00015 Monterotondo*
[b] *Istituto di Cristallografia - C.N.R., via Salaria km 29,3, 00015 Monterotondo*
[c] *Roma Capitale, Dipartimento Progetti di Sviluppo e Finanziamenti Europei, U.O. Coordinamento Programmazione e Progettazione Comunitaria, Via del Tempio di Giove n. 3, 00186 Roma*

† Rapporto 2015/07 con Protocollo CNR-IC 976 del 22/05/2015

- **Digital Divide e Web 2.0**; numero di partecipanti: 100;
- **Media e Comunicazione pubblica**; numero di partecipanti: 110.

Dopo una indagine preliminare sui software utilizzati dalle pubbliche amministrazioni per le loro iniziative di formazione del personale dipendente e dalla lettura dei risultati pubblicati dal CNIPA[1] sulle piattaforme di *Learning Management System*, si è constatato che anche in questo ambito Moodle risulta essere la scelta preferita.[2]

I principali vantaggi riscontrati dagli utilizzatori di Moodle sono:

- forte attenzione dedicata agli aspetti pedagogici dell'apprendimento mediato dalle tecnologie;
- un sistema intuitivo, semplice da utilizzare, flessibile e idoneo per le diverse modalità di erogazione della didattica;
- ottimo supporto per il *blended learning* perché riesce a soddisfare pienamente la sincronizzazione di un flusso eterogeneo di eventi didattici;
- integrazione delle attività svolte in presenza e/o a distanza;
- consente di organizzare e gestire in maniera integrata e complementare materiali didattici di diversa natura, offrendo all'utente la possibilità di individuarne facilmente le relazioni e la tipologia;
- un'interfaccia intuitiva e semplice da utilizzare, che agevola e aiuta gli utenti a creare e gestire le relazioni attraverso l'uso di strumenti di comunicazione e collaborazione differenti.

Il software moodle può essere arricchito con funzionalità aggiuntive a seconda delle esigenze didattiche dei corsi. Nel caso della PA Roma Capitale si è reso necessario l'uso di due moduli aggiuntivi per la certificazione della partecipazione dei dipendenti alle attività didattiche dei corsi durante l'orario di servizio.

Nella figura 1 un esempio di rapporto della frequenza di un partecipante al corso.


Gruppi : classe 1c
Ruoli : Student
Equivalent training time: 5 giorni 19 ore (0)
Time in activities: 3 giorni 1 ora
Work sessions: 165

**Time spent out of structure**

| | |
|---|---|
| Course global areas | 1 giorno 22 ore (0) |

**In courses**

**Sito**

Total time : 1 giorno 6 ore
Hits : 670

**E-Learning**

| Corso | Total time | Hits |
|---|---|---|
| Media e Comunicazione pubblica | 3 giorni 5 ore | 2302 |
| Scrittura e Linguaggi nella Comunicazione Web | 1 giorno 7 ore | 490 |

**Working sessions (real guessed times)**

| Session start | Session end | Duration |
|---|---|---|
| mercoledì, 17 ottobre 2012, 11:01 | mercoledì, 17 ottobre 2012, 11:05 | 8 min. 58 secondi |
| mercoledì, 17 ottobre 2012, 11:06 | mercoledì, 17 ottobre 2012, 11:07 | 3 min. 5 secondi |
| mercoledì, 17 ottobre 2012, 13:54 | mercoledì, 17 ottobre 2012, 13:54 | 4 secondi |

**Fig. 1** Esempio di rapporto sull'attività di uno studente.

## 3 Infrastruttura hardware della Piattaforma Informatica

Il gruppo Smart eLab disponeva già di una infrastruttura tecnologia hardware ad alta prestazione fortemente ridondata, il software su cui si basa e VMWare. VMWare permette la creazione di macchine virtuali (VirtualHost) sulle quali sono state costruite strutture LAMP in alta affidabilità (VirtualHosting in High-Availability)[3] tra cui una completamente dedicata alla didattica realizzata ad hoc per il Progetto Minerva.[4] L'aggiunta della nuova piattaforma ci ha spinto a sperimentare nuovi accorgimenti tecnici per migliorare le prestazioni, anche in previsione di picchi di traffico determinati dalla concentrazione di orario di svolgimento dei corsi (prevalentemente durante l'orario di lavoro).

Ogni corso era così strutturato:

- 30 ore di lezione frontale, con un impegno di sei ore a settimana;
- 30 ore di attività on-line da distribuire durante l'orario di lavoro per l'intera durata del corso a scelta del partecipante;
- i partecipanti suddivisi in sei classi da 20;
- Le attività online delle classi devono essere separate e non visibili;

Come parametri da utilizzare nella valutazione del raggiungimento delle prestazioni tecniche richieste dai committenti sono stati scelti i seguenti:

- Tempi di latenza a seguito di richieste di servizio multiple e simultanee;
- Continuità di servizio, anche in presenza di guasti tecnici imprevisti a una delle componenti dell'infrastruttura tecnologica;
- procedure di Backup dati giornaliere e tempi di conservazione delle copie fino a un massimo di cinque giorni;
- Tempi di ripristino dati dei server per il recupero documentazione erroneamente rimossa;
- Tempi di assenza servizio per operazioni di aggiornamento software per rimozione criticità della sicurezza.

La piattaforma informatica Moodle per svolgere il compito di gestione dei contenuti didattici utilizza la classica pila LAMP. Quest'ultima è stata suddivisa in comparti separati per ogni servizio.[4] Questa soluzione consente di ottimizzare i sistemi operativi dei server in base al servizio affidato, mantenendo alte le prestazioni anche in caso di grandi volumi di traffico concentrato temporalmente. Grazie a questa impostazione tecnica di base questa soluzione da la possibilità di realizzare più piattaforme di e-Learning che condividono le stesse risorse hardware. Attualmente sono in funzione contemporaneamente no-

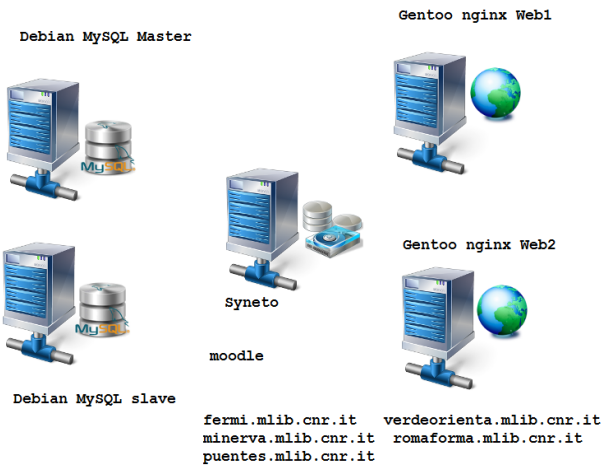



**Fig. 2** infrastruttura tecnologica piattaforma di elearning CNR

ve piattaforme informatiche per Istituti Scolastici Superiori, Dottorati di Ricerca, Istituti di Ricerca e per Roma Capitale.

La ridondanza delle risorse è stata ottenuta installando due server fisici in due CED separati. Sono state attivate pratiche per il bilanciamento di carico tra i due server in modo da ripartire il traffico tra di loro aumentando cosi' le prestazioni e, in caso di guasto ad uno di essi, continuità di servizio in quanto il traffico viene automaticamente inoltrato verso il server rimasto attivo. La struttura realizzata consente di aggiungere ulteriori server per aumentare le prestazioni del sistema. Il compito di bilanciare il traffico è delegato al firewall Stonegate dell'Area della Ricerca RM 1. I dati registrati dai due server web sui server database e data storage vengono mantenuti sincronizzati onde garantire la coerenza delle informazioni prodotte. Per la descrizione tecnica approfondita dell'infrastruttura si rimanda all'articolo pubblicato su questa rivista in quanto questa soluzione tecnica è stata adottata anche per la gestione dei servizi esclusivi web.[3] In figura 3 è riportato uno dei grafici dei test di prestazione della configurazione hardware adottata, in condizioni simulate di alto traffico. Si sono simulati 500 utenti con 50 richieste di servizio contemporanee. La bontà dei risultati ha convinto il gruppo di lavoro all'adozione della soluzione tecnica proposta. Fase successiva della sperimentazione è stata lo svolgimento dei corsi e il continuo controllo delle prestazioni della piattaforma e l'analisi di ogni possibile anomalia nella erogazione del servizio.

## 4 Risultati.

I corsi di formazione della PA Roma Capitale si sono svolti da ottobre 2012 e maggio 2013. Durante questo periodo non si sono verificati disservizi che potessero interrompere l'attività di formazione on-line. In figura 4 sono riportati gli utenti connessi e le pagine richieste durante il periodo di svolgimento dei corsi. Dal grafico è evidente l'utilizzo della piattaforma prevalentemente nel-

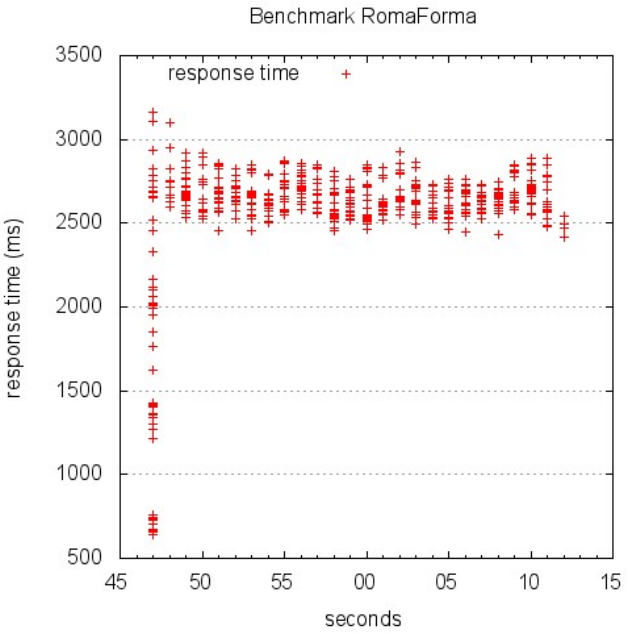



**Fig. 3** Risultati della simulazione dei tempi di risposta della piattaforma in situazione di alto traffico.

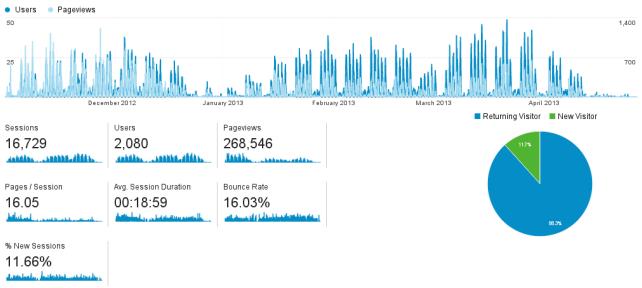



**Fig. 4** Numero di utenti e pagine richieste durante lo svolgimento dei corsi.

l'orario di lavoro. Le operazioni di backup di sicurezza sono state svolte ogni giorno alle ore 03:00 con possibilità di roll-back massima di 5 giorni. Gli aggiornamenti periodici del software moodle hanno comportato brevi sospensioni del servizio tra i 10 e 20 minuti per un totale di circa un'ora nei sei mesi dell'attività formativa.

Nonostante il numero degli iscritti ai corsi fosse di circa 300 unità, con una attività formativa prevalente durante l'orario lavorativo e distribuite su più sedi, diffuse nel territorio municipale, la piattaforma ha sostenuto il servizio informatico, come riportato nei grafici seguenti, senza defezioni. Il software Moodle corredato di alcuni specifici plugin di rendicontazione ha consentito di registrare e certificare l'attività di ogni singolo corsista, sia ai fini del conseguimento degli obiettivi didattici del corso sia per la giustificazione dei permessi per formazione usufruiti.

Di seguito sono riportati i diagrammi delle attività svolte durante il periodo formativo per i tre corsi di formazione.

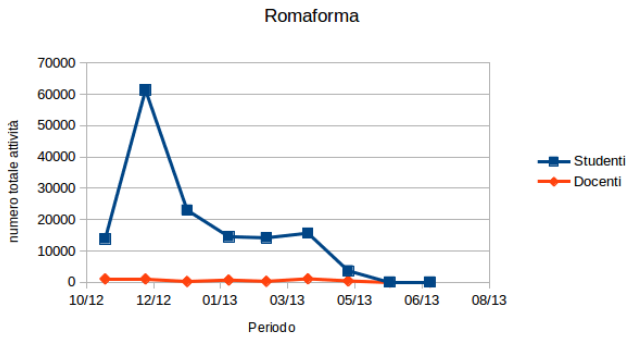



**Fig. 5** Numero totale di attività svolte sulla piattaforma informatica Romaforma.

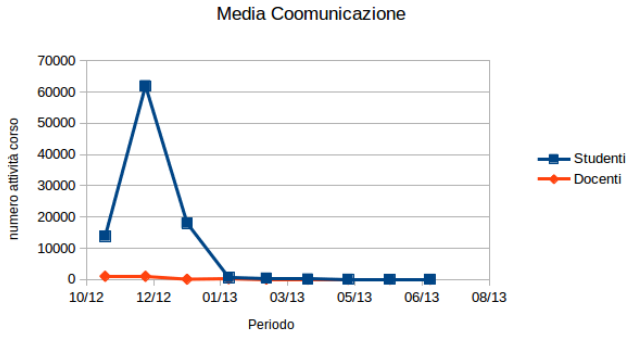



**Fig. 6** Numero attività svolte durante il corso “Media e Comunicazione nella PA”.

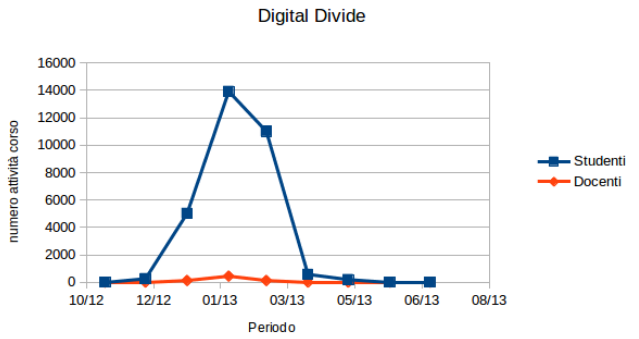



**Fig. 7** Numero di attività svolte durante il corso “Digital Divide”

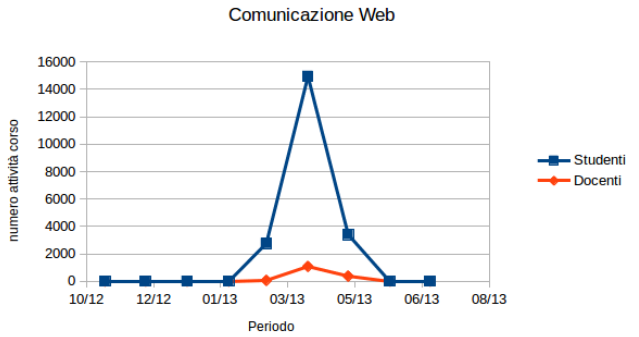



**Fig. 8** Numero di attività svolte durante il corso “Comunicazione Web”.

## 5 Conclusioni

L’infrastruttura hardware realizzata per accogliere la piattaforma informatica della formazione per i corsi della PA Roma Capitale si è dimostrata essere rispondente alle specifiche tecniche richieste. Questa infrastruttura tecnologica è stata adottata per gestire tutte le nove piattaforme informatiche commissionate all’Istituto di Cristallografia da parte di Istituti Scolastici Superiori , di Ricerca, Università e PA con requisiti tecnici di Alta Prestazione, Alta Affidabilità e Alta Disponibilità.

## Riferimenti

1 AA. VV., Vademecum per la realizzazione di progetti formativi in modalità e-learning nelle pubbliche amministrazioni, I Quaderni - CNIPA 32 (2007) 128–133, http://www2.cnipa.gov.it/site/_files/cnipa_quad_32.pdf.

2 S. De Lorenzis, tudio per la realizzazione di un corso di formazione ambientale attraverso la piattaforma e-learning moodle, Relazione Tirocinio, ISPRA http://www.isprambiente.gov.it/contentfiles/00005100/5128-delorenzis.zip (2010).

3 L. Ianniello, G. Nantista, A. Lora, A. Pifferi, Progetto virtual hosting in high-availabity per l’area della ricerca rm1 del cnr., SMART eLAB 3 (2014) 25–33. doi:10.30441/smart-elab.v3i0.85.

4 G. Righini, L. Ianniello, G. Nantista, A. Lora, A. Pifferi, Progetto minerva: La piattaforma di e-learning dell”area della ricerca rm 1., SMART eLAB 1 (2013) 13–25. doi:10.30441/smart-elab.v1i0.24.