

HighWay/eXtraWay Project - Schema sintetico delle configurazioni hardware Manuale di riferimento

Generato da Doxygen 1.5.1-p1

Fri Jan 4 11:10:14 2008

Indice

Capitolo 1

Schema sintetico delle possibili configurazioni HardWare

1.1 Introduzione alle configurazioni Hard Ware

Il presente documento vuole dare evidenza di quali possano essere gli scenari possibili per soddisfare le esigenze di progetti realizzati con il motore *eXtraWay Server*.

In primo luogo vale la pena di sottolineare che un'applicazione realizzata con la tecnologia *ExtraWay* prevede normalmente una struttura *3 tiers* ovvero la presenza di tre strati software che si occupano di:

- Presentazione delle informazioni raccolte e controllo delle informazioni inserite dall'utente (Web Server)
- Elaborazione delle informazioni inserite dall'utente e gestione delle comunicazioni con la banca dati (Application Server)
- Gestione dei dati vera e propria (DataBase Server)

Una simile struttura si presta naturalmente per consentire una scalabilità verticale e, se le condizioni lo consentono, anche orizzontale. Il primo ed il secondo strato software possono essere scalati orizzontalmente senza limitazioni mentre il terzo, vale a dire il DataBase Server, ovvero *ExtraWay*, può essere scalato orizzontalmente solo in alcuni casi.

Va inoltre ricordato che un ulteriore livello di separazione dei compiti può essere ottenuto distinguendo il "Back Office" di un'applicazione dal suo "Front Office" se la gestione dei due accessi ai dati può essere desincronizzato.

Di seguito vedremo tutti i possibili scenari soffermandoci sugli aspetti più significativi per il progetto in corso.

1.2 Gli scenari "di base" disponibili

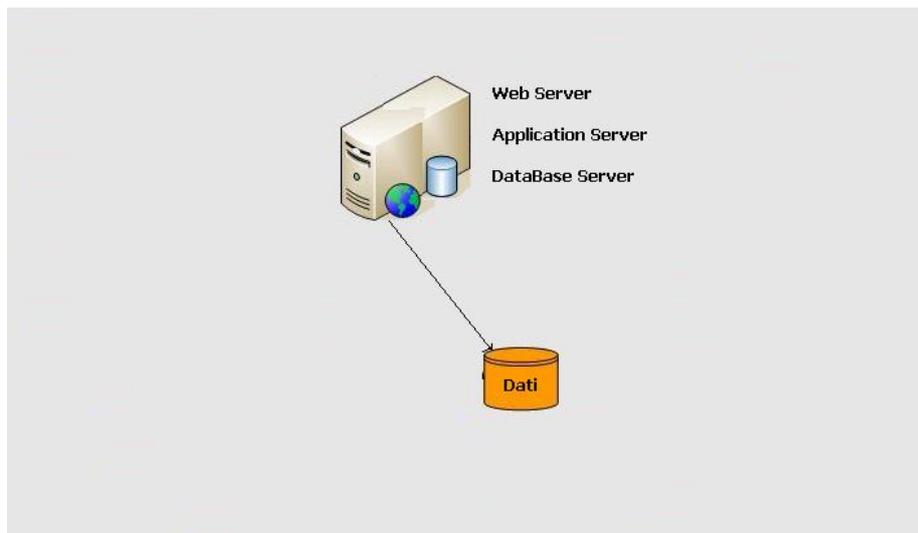
Si considerano scenari "di base" quelli che offrono il servizio minimo indispensabile senza garantire particolari livelli di performance, di continuità di servizio e di separazione tra settore redazionale e meri consultatori.

Nota:

La rappresentazione dei "Dati" separatamente dalla macchina è solo indicativa. Esclusivamente nei casi di Cluster tale separazione è effettivamente un requisito.

1.2.1 Configurazione "minima"

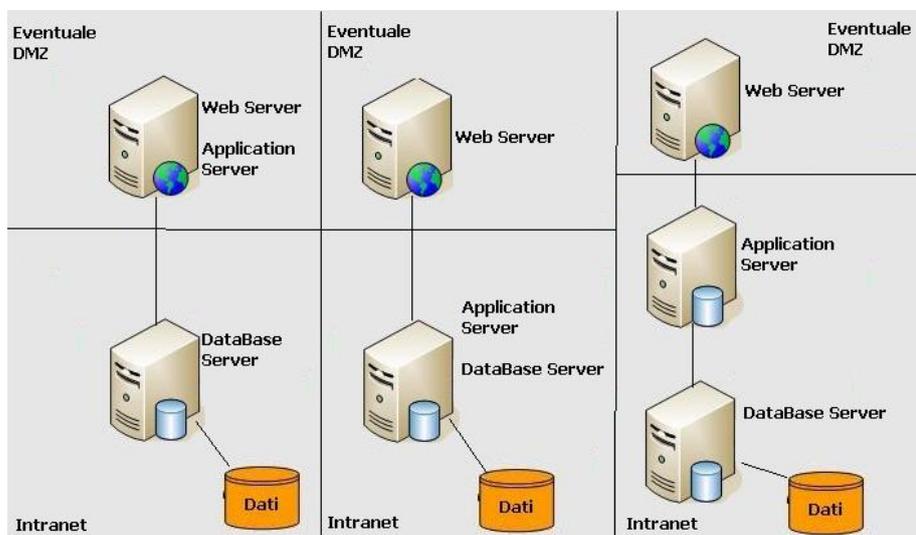
Lo scenario "minimo" va concepito con la compresenza sullo stesso server di tutti e tre gli strati software descritti precedentemente. Non sfrutta alcuna possibilità di distribuire il carico di lavoro rappresentato dai singoli strati scalando l'hardware verticalmente e non garantisce una continuità nel servizio in caso di guasto.



1.2.2 Configurazione scalata verticalmente

Deriva direttamente dalla configurazione minima e consiste nell'aver due o tre strati hardware, non replicati tra loro, che consentano di distribuire verticalmente il carico di lavoro. Sfrutta quindi la possibilità di distribuire il carico (da una configurazione minima su due livelli ad una *canonica* su tre) ma non garantisce una continuità nel servizio in caso di guasto.

Di seguito i tre esempi possibili. In due casi si sfruttano solo due macchine per distribuire i tre livelli di software mentre il terzo caso rappresenta la condizione più ampiamente distribuita e quella solitamente adottata quando si decide di compiere una scalatura verticale dell'hardware.



1.3 Scenari complessi

Negli scenari complessi trovano posto tutte le diverse configurazioni che possono risultare maggiormente garantiste per prestazioni, continuità del servizio e separazione tra settori.

In particolare si prenderanno in esame i sistemi che sfruttano una tecnologia di Load Balancing e/o Fault Tolerance avvalendosi di uno o più Cluster.

1.3.1 Configurazioni con l'uso di Cluster

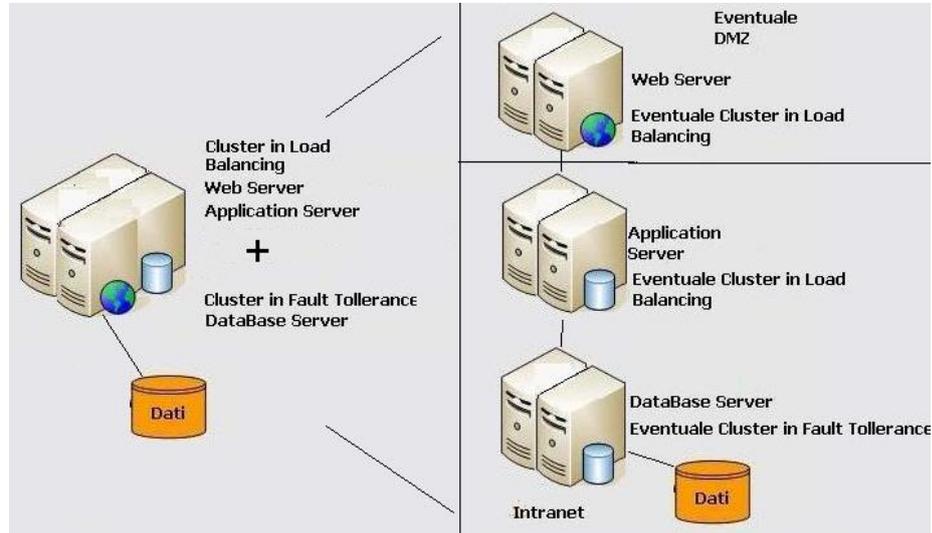
Le configurazioni che si possono avvalere di sistemi di Cluster possono essere viste come dirette derivazioni degli scenari "di base" descritti precedentemente. In tal caso, in virtù della suddivisione "verticale" degli strati software sull'hardware posto in gioco, si possono assoggettare a Cluster gli strati preferiti. Si può infatti pensare di porre in Load Balancing (quindi in cluster di macchine tutte operative) il primo e/o il secondo strato software coinvolto (Presentazione ed Elaborazione) mentre per il terzo strato, la Gestione Dati, è necessario fare affidamento ad un sistema di Fault Tolerance.

Vedremo infatti che anche in tutti gli scenari ad eccezione della separazione tra Back Office e Front Office, la Gestione Dati non può essere amministrata correttamente solo con una configurazione in Load Balancing.

Nulla vieta, comunque, che anche in presenza di una sola coppia di macchine in cluster tra loro, i diversi strati di software possano essere amministrati separatamente usufruendo della distribuzione di carico almeno per i primi due di essi.

L'immagine seguente vuole dare evidenza del fatto che le macchine possono essere poste in cluster quale che sia la scalatura verticale scelta secondo le ipotesi avanzate nel precedente capitolo. In particolare l'esempio di destra indica come "eventuale"

ogni cluster in quanto si può decidere a quale livello esso è necessario ed applicarlo solo in tal caso. In caso di Cluster sul DataBase Server l'adozione di unità dischi esterna condivisibile dalle due macchine è pressoché inevitabile.



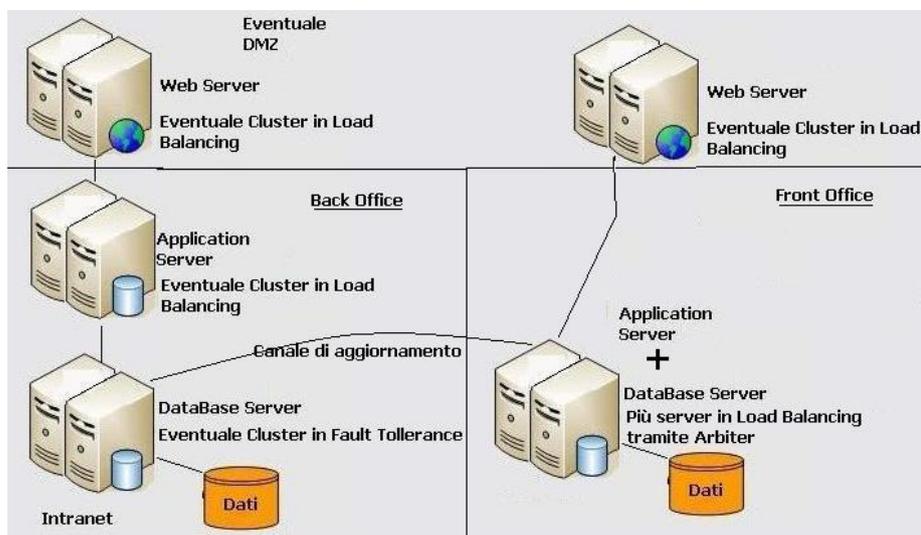
1.3.2 Configurazione separata tra Back e Front Office

In una simile configurazione c'è la massima disponibilità di scalare l'hardware coinvolto. Questa configurazione deriva dalla precedente ma replica su due serie di macchine i ruoli redazionali (Back Office) e di consultazione (Front Office) con una sincronizzazione tra i due ambienti da definire.

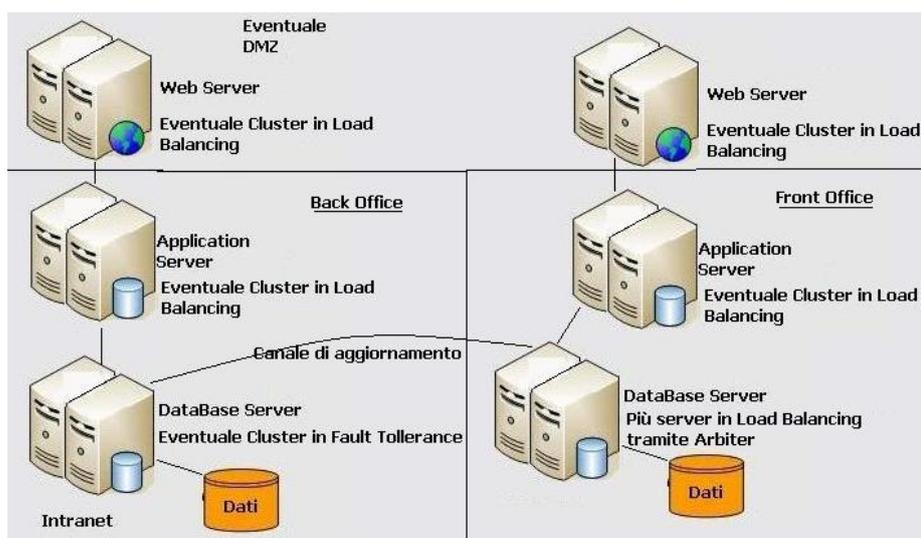
Lo scenario prevede quindi che il Back Office, strutturato in una qualsiasi delle precedenti modalità descritte, con o senza Cluster, mantenga aggiornato con uno strumento di distribuzione (Spreader) una batteria di macchine di Front Office (idealmente da due in su) che non hanno la necessità di essere poste in cluster. Sarà il procedimento di distribuzione, avvalendosi di un processo detto "arbiter" a sincronizzare tutte le macchine del Front Office. L'arbiter, a sua volta, potrà esporre all'utenza le sole macchine effettivamente sincronizzate.

L'esempio che segue deriva dall'applicazione di un Cluster completo, scalato nell'hardware sui 3 livelli software (parte di sinistra). In sostanza, per il nostro esempio, il Back Office può essere strutturato in una qualsiasi delle modalità sin ora viste, quella prescelta è solo esemplificativa. A destra vediamo invece l'aspetto che avrebbe il front office con l'uso di Web Server distinti ed una batteria di due o più macchine tutte contemporaneamente in grado di fungere tanto da Application Server quanto da DataBase Server grazie all'intervento di un Arbiter.

La prima immagine rappresenta il caso tipico mentre la seconda mostra un caso più complesso nel quale si sia voluta compiere un'eventuale suddivisione del carico di lavoro separando anche lo strato software inerente l'Application Server.



Nella seconda immagine, fondamentalmente equivalente alla prima, si introduce solo l'Application Server in Load Balancing.



1.4 Note sullo "Spreader"

Come detto in precedenza, una separazione tra Back Office e Front Office può essere ottenuta in diversi modi ed esistono senza alcun dubbio strumenti in grado di compiere una replicazione più o meno *complessiva* dello stato di dati ed indici (gli archivi) dalla macchina del Back Office a quelle del Front Office, assumendo che esse siano due macchine in Load Balancing. Condividere via NFS o altro strumento di rete i dischi di un server verso altri server per ampliare la batteria di macchine a disposizione del-

l'utenza è una pratica percorribile ma che, in particolare a fronte di un utenza di ampie dimensioni, risulta scarsamente performante.

A tale scopo 3D Informatica ha realizzato uno strumento denominato "Spreader", avente lo scopo di distribuire alle macchine di un Front Office uno o più archivi provenienti dal Back Office operando in modo differenziale e, collaborando con l'Arbiter che compie la distribuzione di carico tra le macchine Front, mantenendo aggiornato lo stato delle macchine in modo sempre allineato.

Cerchiamo quindi di spigarne il comportamento tramite un esempio. Abbiamo un server di BackOffice (BO1) e 5 server di Front Office (FO1, FO2, FO3, FO4 ed FO5).

Quando l'amministratore decide di compiere l'aggiornamento dal Back Office al Front Office, ovvero tramite un automatismo a tempo, lo Spreader controlla richiedendo le informazioni alle macchine del Front Office, quale sia il loro stato di aggiornamento.

Supponiamo in un primo momento che esse siano tutte aggiornate al giorno precedente.

Lo Spreader richiede al Server BO1 di valutare le differenze intercorse in termini di dati ed indici dalla data ed ora (time stamp completo sino ai millisecondi) di ultimo aggiornamento *comune*. Il file prodotto, chiamato *delta* viene in seguito usato per l'aggiornamento delle singole macchine.

L'aggiornamento avviene secondo questa tecnica.

Lo Spreader chiede all'Arbitro di porre in "stand by" FO1 che da quel momento non verrà più usata per distribuire il carico di lavoro richiesto dall'utenza. Poi invia a FO1 il *delta* e gli chiede di applicarlo. Una volta applicato il server FO1 viene mantenuto in "stand by".

Poi fa altrettanto con FO2 e con FO3. Avendo superato "la metà" dei server disponibili sul Front Office, lo Spreader richiede all'Arbitro, con un'operazione atomica, di porre in "stand by" le macchine FO4 ed FO5 e di riportare "on line" FO1, FO2 ed FO3 che essendo tutte aggiornate ed in "maggioranza" possono meglio servire l'utenza.

Di seguito provvede a compiere l'aggiornamento su FO4 ed FO5 richiedendo che vengano poste "on line" appena aggiornate per rimpinguare il pool di macchine disponibili all'utenza.

Se nel procedimento descritto una macchina risulta non raggiungibile, spenta, precedentemente posta in "stand by" e non rimessa "on line" a causa di un errore di aggiornamento, lo Spreader stabilisce quale sia la combinazione migliore di macchine da lasciare "on line" per garantire il miglior servizio al miglior stato d'avanzamento. Se le macchine FO sono allineate in modo diverso, il *delta* che viene calcolato comprende il materiale necessario a condurle tutte allo stato d'aggiornamento corrente anche a costo di applicare aggiornamenti già applicati in parte (senza con questo impattare sulla validità degli archivi).

1.5 Estensione degli Scenari, il "Disaster Recovery"

Gli scenari, semplici e complessi, sin ora descritti consentono di costruire una struttura con diversi livelli di sicurezza già in essa. Ad esempio, un Back Office configurato con un Cluster è ragionevolmente solido ed ha un Back Up naturale nel Front Office. Una

struttura che preveda più Front Office ha in essa una sicurezza intrinseca di continuità del servizio.

Se a questi scenari vogliamo affiancare un ulteriore livello di sicurezza entriamo nell'ambito del *Disaster @ Recovery* per il quale valgono le considerazioni già fatte negli scenari descritti. Anche in quel caso possono essere ipotizzati scenari semplici e complessi per il nodo *secondario* indipendenti dalla configurazione data a quello *primario*.

Esso consiste, solitamente, nella replicazione dell'hardware (nella forma più opportuna) e di tutti i dati disponibili ed allineati in una struttura differente. Per "struttura", in questo caso, facciamo riferimento "fisicamente" ad un luogo diverso, servito differentemente sia dal punto di vista elettrico che per ogni altro aspetto logistico.

Quello che può essere replicato in questa differente struttura è un qualsiasi scenario "minore o uguale" a quello primario che può essere mantenuto allineato con degli automatismi con cadenza giornaliera o con quella ritenuta più opportuna.

Gli strumenti a disposizione, dal trasferimento e l'applicazione di un backup all'uso dello Spreader, consentono di avere un allineamento ragionevolmente "fresco" della copia presso la struttura secondaria ed essa può essere concepita come sola salvaguardia dei dati (quindi una copia fedele del Back Office consultabile) sia come una completa replicazione della configurazione primaria.

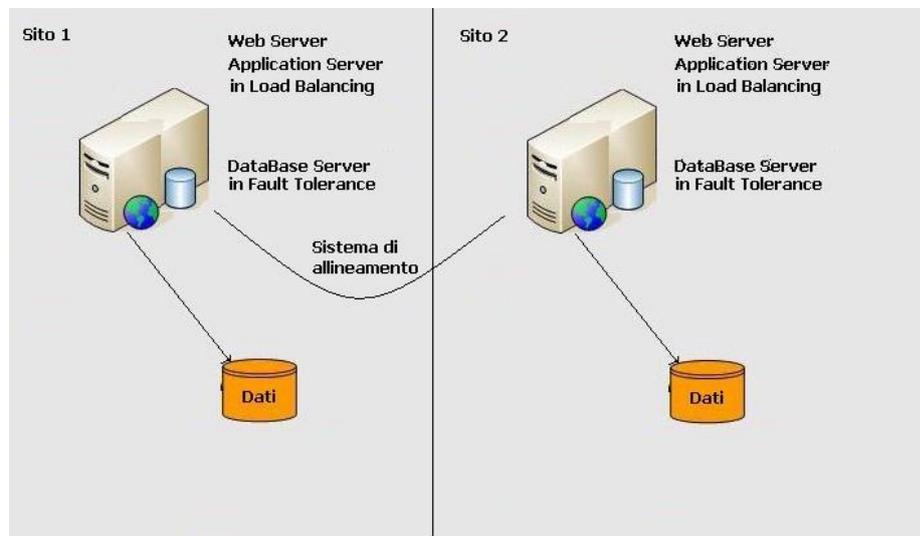
Anche se per il nodo *primario* si fosse scelta una configurazione minima in essa dovrebbe trovare posto per lo meno uno spreader che provvederà a mantenere aggiornata una delle repliche che seguono.

La replicazione dello scenario dovrebbe essere integrale. La replicazione integrale va vista come la riproduzione, presso diversa struttura, di una configurazione minima o scalata come dagli scenari descritti. Essa può essere una versione "ridotta" (ovvero priva di scalatura orizzontale o verticale o con una scalatura ridotta) di quella primaria ma comunque pienamente funzionante.

Tramite spreader (considerando il BackOffice dell'installazione *secondaria* al pari di un Front Office da aggiornare da parte del nodo *primario*) si può mantenere aggiornato il secondo nodo ed in esso si provvederà, se anche il nodo secondario è scalto verticalmente, ad aggiornare il proprio Front Office avvalendosi di un'installazione "locale" dello spreader.

In caso il nodo *secondario* venga chiamato in causa per indisponibilità del *primario* esso prevederà un ambiente completo che consentirà di proseguire tanto nell'attività redazionale quanto in quella di fruizione del sistema mantenendo eventualmente distinte le due realtà.

Si tratta dello scenario più completo ed esaustivo.



Possono essere presi in esame altri scenari in cui il nodo *secondario* ha una replicazione parziale del nodo *primario*, replicando in esso il solo Back Office o il solo Front Office. In tal caso la fruibilità del nodo *secondario* potrebbe risultare limitata ed esso avrebbe principalmente lo scopo di ricreare quanto prima possibile le macchine del servizio primario piuttosto che sostituirsi efficacemente ad esse.

1.6 Conclusioni

Concludendo, gli scenari che si possono realizzare sono stati ampiamente descritti e possono essere ottenuti combinando in vario modo le possibilità esposte. La scelta sulla configurazione più opportuna deve tenere in considerazione due fattori evidenti: la scalabilità orizzontale garantisce di adeguarsi alle performance richieste dal sistema in virtù del carico di lavoro previsto sulla base dell'utenza che ne farà uso, potenzialità in parte ottenibile anche con la scalabilità verticale, ma rispetto a quest'ultima offre tipicamente maggiori risultati e da garanzie di continuità nell'erogazione del servizio che la sola scalabilità verticale non può offrire.

L'unico reale vincolo si ha nella realizzazione di un Cluster per il DataBase Server del BO (o globale se non v'è distinzione tra le due parti) in quanto non è possibile adottare una configurazione in Load Balancing ma solo in Fault Tolerance per lo meno sino a quando alcune limitazioni esistenti nella piena accessibilità delle risorse condivise non saranno superate dai produttori di hardware e software di base.

Questo, per lo meno, sulla base dell'esperienza sino ad ora accumulata in proposito.

1.7 Cronologia del documento

- Il presente documento è stato scritto da Roberto Tirabassi di 3D Informatica nei giorni 18 e 19 luglio 2006.

- Il paragrafo inerente il *Disaster Recovery* è stato aggiunto da Roberto Tirabassi di 3D Informatica il giorno 22 gennaio 2007.

Autore:

Tirabassi Roberto