

L'evoluzione dei sistemi di telecontrollo: metodologie, approcci e soluzioni

Maurizio Bigoloni, Franco Corti, Maria Antonietta Di Jonno, Damiano Manocchia

ABSTRACT

I paradigmi e le architetture dei sistemi di telecontrollo, sempre più, si stanno avvicinando a quelli adottati per sistemi informativi aziendali, le funzioni di monitoraggio e supervisione vengono estese con funzioni di business process management (BPM) e business intelligence (BI), gli strumenti e le tecnologie di controllo si integrano con i sistemi di supporto alle decisioni (Decision Support System), si instaura una crescente collaborazione tra i vari settori aziendali con lo scopo di definire un modello di integrazione che sia frutto di una visione di insieme.

Il panorama attuale vede la coesistenza dei classici sistemi di telecontrollo “monolitici”, dei più recenti sistemi distribuiti, la cui architettura prevede diverse tipologie di stazioni (unità di elaborazione, front end di comunicazione, postazione operatore, etc), e si sta affacciando sui moderni sistemi “service oriented”, in cui il sistema di telecontrollo, integrandosi con piattaforme di più alto livello, mette a disposizione servizi verticali disponibili anche su Internet.

L'inerzia con la quale il mondo del telecontrollo affronta tale evoluzione è in parte giustificata dall'elevato livello di complessità dei sistemi stessi, dalle criticità intrinseche dei sistemi real time, dalla protezione degli investimenti dei sistemi legacy attualmente in uso, e talvolta da alcune metodologie e approcci che hanno portato a esperienze che non hanno soddisfatto appieno le esigenze di carattere funzionale.

Un approccio top-down o, comunque guidato dall'innovazione tecnologica risulta avere degli impatti molto rilevanti sulle architetture tradizionali, mentre un approccio bottom-up, favorisce l'integrazione delle nuove tecnologie mantenendo il focus sui requisiti di affidabilità, disponibilità e sicurezza caratteristici dei sistemi real-time.

Nell'ambito della memoria saranno analizzati alcuni case study in cui l'evoluzione architettonica dei sistemi di telecontrollo affronterà svariati temi tra cui:

- Introduzione della virtualizzazione, una tecnologia che permette di sostituire le stazioni fisiche con stazioni virtuali, di astrarre il concetto di macchina in concetto di servizio, di garantire e migliorare le prestazioni in termini di scalabilità, disponibilità, balancing.
- Evoluzione verso architetture a servizi (SOA/ESB): in particolare, l'introduzione di metodologie che favoriscano l'evoluzione di un sistema legacy ad architettura distribuita verso un'architettura orientata al servizio.
- Evoluzione verso il paradigma cloud computing e verso il business model “software as a service” (SaaS): la concezione del sistema di telecontrollo nella sua più ampia visione, in particolare l'apertura del sistema SCADA verso le risorse della Intranet aziendale (interfacciamento con altri sistemi enterprise) e verso Internet (pubblico, clienti, etc.).

INTRODUZIONE

La presente memoria ha lo scopo di analizzare le evoluzioni dei sistemi di telecontrollo in parallelo alle trasformazioni dei paradigmi architetturali tipici dell'Information Technology.

Scorrendo l'attuale situazione fino a quello che è il futuro prossimo prospettato, analizzeremo le soluzioni architetturali che, rappresentano una prospettiva fattibile in ambito del telecontrollo, del monitoraggio di impianti e del controllo di processo.

L'evoluzione tecnologica si sposta sempre di più verso paradigmi tecnologici definiti "cost reflective", ovvero soluzioni che permettano di ottimizzare i benefici e di minimizzare i costi delle risorse impegnate.

Nel corso della trattazione verrà individuato un paradigma infrastrutturale a tendere: l'utility computing.

Le varie tecnologie verranno declinate nel mondo del telecontrollo e verranno presentati i vantaggi e i limiti di ciascuna di esse, per poi andare a presentare lo scenario più prossimo: l'integrazione del Cloud nei sistemi SCADA.

I DRIVER EVOLUTIVI

In un'analisi di questo tipo è fondamentale identificare i principali driver che hanno guidato l'evoluzione del telecontrollo e, più in generale, dell'ICT.

Davanti la fase di scelta di una nuova tecnologia, un'azienda valuta, innanzitutto, i benefici economici e finanziari legati alla tecnologia stessa, principalmente in termini di riduzione di costo e aumento dei ricavi.

In seconda istanza viene valutata la "business agility" della soluzione, ovvero la possibilità di esplorare nuove opportunità di business senza rivoluzionare l'infrastruttura adottata. Un terzo driver è dato dalla capacità di ottimizzare le risorse, quindi la possibilità di minimizzare quelle sottoutilizzate e, contemporaneamente, di

far fronte ad eventuali richieste di picco. Sicuramente anche l'evoluzione delle reti di telecomunicazione è un fattore cruciale per l'innovazione tecnologica, soprattutto nello scenario di architetture distribuite o su sistemi sparsi su grandi aree geografiche. Al fine di eleggere una data tecnologia è, inoltre, opportuno valutare i relativi aspetti in termini di legislazione, di privacy e di security. Infine, è importante sottolineare il ruolo fondamentale e gli aspetti sociali economici e politici legati allo scenario di riferimento.

Non c'è dubbio che la tecnologia contribuisca pesantemente ai cambiamenti nella nostra società: si pensi, ad esempio, alla trasformazione della comunicazione a seguito della diffusione delle email e dei social network, alla diffusione dei PC a seguito dell'introduzione delle interfacce grafiche come quelle dei sistemi operativi Windows e della Apple GUI, l'introduzione dei motori di ricerca, etc.

È opportuno prestare attenzione a temi più recenti e molto sentiti come l'efficienza energetica e la sostenibilità: una svolta importante è stata data da Siemens nel 2009 introducendo la figura di Chief Sustainability Officer nel *managing board*.

Nonostante l'attualità di questi temi, numerosi studi riportano che, ad oggi, l'utilizzo medio dei server non supera il 15% delle proprie potenzialità; il 3% dei consumi di elettricità a livello mondiale sono impegnati dall'ICT; il 2% dell'emissione globale di CO₂ è imputabile alla componente IT (praticamente pari all'industria dell'aviazione).

Contestualmente le aziende vedono la crescita del proprio numero di server e del costo dell'elettricità, e un'attenzione sempre maggiore alla così detta "carbon footprint" (l'ammontare dell'emissione di anidride carbonica attribuibile ad un'organizzazione).

PARADIGMI ARCHITETTURALI

Prima di entrare nel merito della trattazione è opportuno introdurre il paradigma di **utility computing**.

Il concetto di utility computing è stato teorizzato da John McCarthy nel 1960 durante la sua esperienza al Massachusetts Institute of Technology e poi esteso da Douglas Parkhill nel 1966.

Nel suo libro, *The Challenge of the Computing Utility*, Parkhill esamina la natura delle utenze come acqua, gas ed elettricità e la relativa fornitura per poi trasportarle nel mondo dell'Information Technology. Ad esempio, quando pensiamo alla fornitura di energia elettrica, si tende a dare per scontato che la potenza elettrica necessaria nelle nostre abitazioni sarà sempre disponibile: gli utenti possono usufruire della potenza elettrica necessaria collegando i dispositivi elettrici alle prese a muro. Periodicamente la società incaricata della fornitura elettrica fattura i consumi, e gli utenti pagano quanto consumato.

Alla base dello scenario descritto c'è un assunto: a meno di casi eccezionali, non ci aspettiamo che la fornitura di elettricità sia interrotta.

Nel paradigma utility computing le risorse informatiche sono viste come se fossero fornite da un'utility: tutte le risorse sono rese disponibili quando necessarie, in questo modo la fornitura appare illimitata. Infine, l'utente si aspetta di pagare solo per le risorse effettivamente utilizzate, spostandosi da una fornitura di risorse a una fornitura di servizi.

Il paradigma utility computing abbraccia tutti gli aspetti rilevanti della nostra analisi (business agility, ottimizzazione delle risorse, sostenibilità, etc) e, in questa trattazione, rappresenta il modello di riferimento a tendere.

Di seguito presentiamo l'evoluzione dei sistemi di telecontrollo rapportati all'evoluzione dei paradigmi architetturali tipici dell'Information Technology.

Prima della nascita dei sistemi di telecontrollo le operazioni di manovra sui dispositivi collocati in campo per la regolazione di un processo avvenivano soprattutto manualmente e con l'intervento

di addetti specializzati. Con l'avvento e la diffusione dei calcolatori (computer dedicati) sono stati realizzati i primi sistemi SCADA (sistemi monolitici), d'ausilio agli operatori per l'esecuzione da remoto di comandi e manovre sui dispositivi di campo. In seguito, i sistemi di telecontrollo si sono orientati verso un'architettura distribuita, in cui sono definite più tipologie di stazioni quali: le unità di elaborazione (Back End), i dispositivi di Front End, le postazioni operatore etc. Recentemente, il mondo del telecontrollo si sta affacciando verso i più moderni sistemi "service oriented", in cui il sistema si integra con piattaforme di più alto livello e mette a disposizione servizi verticali disponibili sulla Intranet aziendale e su Internet.

SERVICE ORIENTATION

Il termine "service oriented" tende a delimitare una funzione che opera per soddisfare un particolare obiettivo. Alla base di questo paradigma c'è il concetto di astrazione: i servizi offerti all'utente finale sono aggregati di servizi di granularità inferiore che cooperano per realizzare una data funzione aziendale.

Le architetture service oriented (SOA) si avvalgono dei principi del "service orientation" per soddisfare le esigenze organizzative di un'azienda, questo implica che la tecnologia sia eletta in modo da integrare e supportare un modello architetturale definito come un insieme di servizi. L'introduzione della SOA permette di integrare facilmente le nuove applicazioni software (Business Service Component) non strettamente legate alle funzioni SCADA, ma indispensabili per soddisfare le esigenze aziendali, ad esempio i software per la georeferenziazione, outages management, supervisione, decision support system etc.

L'Enterprise Service Bus (ESB), lo strato software di interconnessione, avvalendosi di web service e protocolli standard, facilita l'interconnessione tra i vari servizi e, grazie ad opportuni adattatori, offre la possibilità di

integrare anche i cosiddetti sistemi legacy, ovvero i sistemi *stand alone* o nativamente non orientati ai servizi.

Implementare un'architettura a servizi è un'operazione relativamente semplice quando si progetta partendo da zero; diversamente, la migrazione di un sistema esistente di tipo monolitico/distribuito in un'applicazione a servizi basata sul concetto di bus rappresenta uno scenario più sfidante. L'approccio proposto consiste nell'iniziare a creare nuove funzionalità creando nuovi servizi che saranno progettati in modo che siano il più possibile disaccoppiati tra loro; ognuno di questi servizi espone le proprie funzionalità tramite interfacce standard (Web Services, RESTful API) ad applicazioni di livello più alto (Client) e tramite un bus (ESB) agli altri servizi. Questi servizi non comunicano direttamente con il sistema legacy; al contrario le funzionalità del sistema legacy vengono incapsulate da appositi *wrapper* ed esposte tramite l'Enterprise Service Bus di riferimento.

Il passaggio successivo consiste nell'identificazione e il disaccoppiamento di componenti fortemente interconnessi all'interno del sistema legacy. I nuovi moduli indipendenti comunicano tra loro tramite l'ESB (per mezzo dei *wrapper* sopra descritti).

Una volta che l'intero sistema (i nuovi moduli e la parte legacy) è stato realmente decomposto in servizi disaccoppiati si potranno sfruttare a pieno i vantaggi della SOA e del ESB in termini di affidabilità e scalabilità.

A differenza dei sistemi tradizionali questo non è gestito a livello applicativo dal sistema ma dalla piattaforma SOA/ESB, tipicamente un Application Server che fornisce funzionalità di comunicazione tra servizi (secondo le diverse modalità request-response, public-subscribe, o message-queue) e funzionalità di bilanciamento e alta affidabilità. Ogni singolo servizio potrà quindi essere configurato secondo specifiche logiche di:

- bilanciamento e scalabilità: i servizi più critici dal punto di vista delle performance possono essere lanciati più volte con una logica di bilanciamento configurabile;
- affidabilità: i servizi possono essere eseguiti su macchine diverse in modo trasparente agli utilizzatori delle funzionalità esposte.

I vantaggi principali di questo approccio sono:

- riutilizzo delle funzionalità e di grandi porzioni di codice;
- possibilità di esplorare velocemente e con costi contenuti nuove opportunità di business (per far fronte ad una nuova richiesta è sufficiente introdurre nuove funzioni);
- facilità di monitoraggio di indicatori sulla performance dei processi aziendali quali qualità dei servizi (QoS), lead times, defect rates, etc;
- facilità di integrazione di più sistemi, ad esempio tra il sistema informativo aziendale con quello dei fornitori, delle aziende satellite, etc.

GRID COMPUTING

Il paradigma grid computing è emerso negli anni 90 quando Ian Foster e Carl Kesselman suggerirono di accedere alle risorse del computer nello stesso modo in cui si accede alla rete elettrica.

I supercomputer capaci di elaborare un numero enorme di dati erano finanziariamente proibitivi per molte aree della ricerca, pertanto nacque l'alternativa di connettere e coordinare molti PC come una rete (grid) per ottenere un'unica organizzazione virtuale (virtual organization).

L'esperienza più significativa è stata senz'altro il progetto SETI@Home (Search for Extraterrestrial Intelligence, lanciato nel 1999 da Dan Verthimer), volto ad analizzare i segnali radio in cerca di segni di intelligenze extraterrestri. Il software di calcolo distribuito sfrutta la potenza di elaborazione inutilizzata dai PC su cui viene

installato, realizzando il più grande calcolo nella storia e raggiungendo una capacità di calcolo di circa 770 TeraFLOPS.

Questo paradigma, pur rappresentando una tappa importante del percorso evolutivo delle architetture informatiche, non ha avuto esperienze significative nel mondo del *real time* e del telecontrollo dove i fattori critici di successo sono caratterizzati dall'affidabilità e dai tempi di risposta.

VIRTUALIZZAZIONE

La virtualizzazione è una tecnologia che, sfruttando il continuo incremento della potenza dei processori, permette di creare istanze di macchine virtuali che possono girare su infrastrutture fisiche di vario tipo. L'Istituto nazionale per gli standard e la tecnologia degli Stati Uniti d'America (NIST, National Institute of Standards and Technology) definisce la virtualizzazione "come la simulazione del software e/o dell'hardware su cui viene eseguito altro software. L'ambiente simulato è chiamato virtual machine (VM)".

L'architettura prevede un livello fisico su cui poggia uno strato software chiamato hypervisor. Lo scopo di tale software è permettere alle varie istanze virtuali di accedere e comunicare con lo strato hardware senza installare un sistema operativo. Sull'hypervisor vengono installate le virtual machine (VM), ognuna di esse rappresenta una risorsa computazionale indipendente e ben definita pur non esistendo fisicamente. Su ogni macchina virtuale è installato un sistema operativo che permette di eseguire le tradizionali applicazioni.

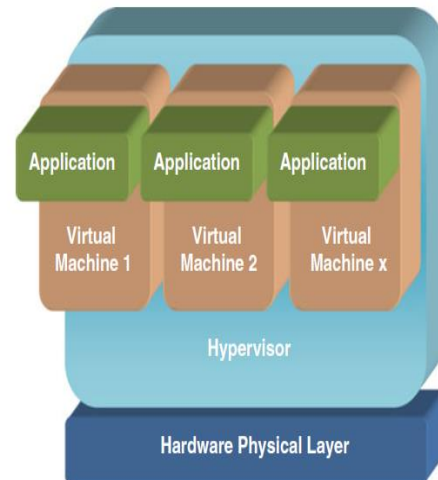


Figura 1. Architettura virtualizzata

Introducendo la virtualizzazione nei sistemi di telecontrollo è possibile ridurre il numero di macchine fisiche necessarie in modo da ottimizzare l'utilizzo delle risorse hardware e ridurre i consumi e l'*overhead* relativo alla gestione dei server.

I principali vantaggi offerti da tale paradigma architetturale sono:

- incremento dell'efficienza a seguito della riduzione del numero complessivo dei server, della massimizzazione dell'utilizzo dell'hardware e riduzione del Total Cost of Ownership (TCO);
- flessibilità e della scalabilità del sistema, potendo il numero e la dimensione delle macchine virtuali variare velocemente;
- incremento dell'indice di disponibilità e affidabilità, facilitando le configurazioni in alta affidabilità e di disaster recovery;
- maggiore sicurezza e segregazione, potendo isolare eventuali macchine causanti effetti dannosi o inefficienze;
- possibilità di realizzare snapshot, cioè di creare e memorizzare vere e proprie "fotografie" di ciascuna macchina virtuale.

CLOUD

L'Istituto nazionale per gli standard e la tecnologia degli Stati Uniti d'America (NIST, National Institute of Standards and Technology) definisce il cloud computing

come “un modello per l’accesso ubiquo, conveniente e su richiesta ad un pool condiviso di risorse configurabili (ad esempio reti, server, storage, applicazioni e servizi) che possono essere rilasciate rapidamente con una gestione ed interazione minima da parte del gestore del servizio”. Inoltre, l’istituto definisce le caratteristiche essenziali del modello cloud, i modelli di servizio e di sviluppo.

Le cinque caratteristiche essenziali definite dal NIST sono:

1. self service su richiesta, le richieste per ulteriori risorse devono essere automaticamente e dinamicamente fornite in tempo reale, senza l’intervento umano;
2. ampio accesso alla rete, cioè usando meccanismi e protocolli di rete standard mediante dispositivi standard (persona computer, smartphone, tablet etc);
3. resource pooling, presentandosi all’utente come un insieme omogeneo di risorse, senza preoccuparsi della localizzazione fisica o della granularità delle risorse;
4. elasticità dinamica, al fine di far apparire la capacità delle risorse illimitata;
5. servizi misurati tramite un servizio di monitoraggio ad un livello di astrazione rilevante per l’utente finale.

I service model del cloud computing, al di sopra del layer fisico, ovvero dell’insieme dei dispositivi che compongono fisicamente l’infrastruttura, sono rappresentati nella seguente figura e sono:

1. Infrastructure as a Service (IaaS), il livello più basso del cloud computing che offre l’accesso all’infrastruttura virtuale sulla quale girano il sistema operativo e le applicazioni software;
2. Platform as a Service (PaaS), il sistema operativo e i tool relativi alla piattaforma (ad esempio i compilatori) gestiti dal fornitore del servizio cloud;
3. Software as a Service (SaaS), il livello applicativo secondo il quale le applicazioni sono disponibili via thin

client (web browser, applicazioni per smartphone, etc).

Spesso gli utenti hanno avuto la possibilità di creare nuove opportunità di business attraverso la creazione di un ulteriore strato, definito Business Process as a Service (BaaS), ma, secondo il NIST, tale definizione ricade all’interno del layer SaaS, e, in questa trattazione ci limiteremo ai tre service model enumerati in precedenza.

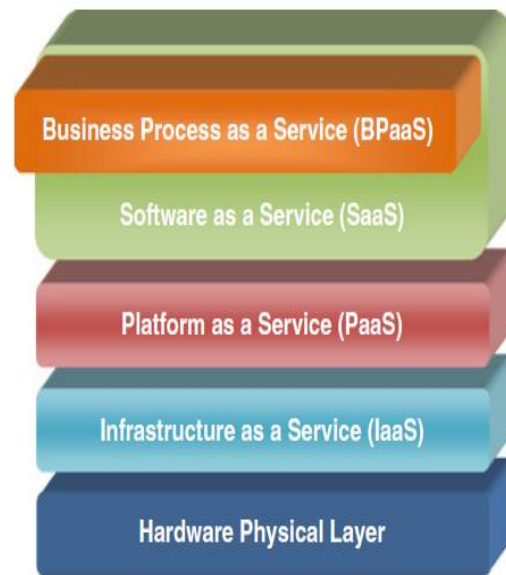


Figura 2. Architettura Cloud Computing

Uno dei concetti più rilevanti del cloud computing è quello di elasticità: non c’è bisogno di investire in risorse hardware utilizzate per far fronte ai picchi e sottoutilizzate per la maggior parte del tempo.

Il rapido approvvigionamento di nuove risorse senza impegno di capitali rende il paradigma cloud “cost effective” e permette di rimuovere tradizionali barriere quali il reperimento di capitali, i tempi dovuti ai processi di acquisto, la disponibilità delle risorse umane necessarie.

Come ogni metodologia e tecnologia, ci sono dei limiti ai benefici del cloud, e nuovi metodi di lavoro possono introdurre rischi aggiuntivi e il cloud computing è un modello in corso di maturazione. Alcuni degli aspetti più sentiti dagli utenti sono i

temi “security” e “trust”, in particolare la definizione di una locazione fisica ove i dati degli utenti sono memorizzati.

Nel caso in cui i dati siano soggetti a legislazioni che risentono della locazione fisica degli stessi, è possibile che si definisca uno scenario in cui i dati siano memorizzati in una nazione fuori dalla portata della giurisdizione di riferimento.

Un altro aspetto da considerare è il così detto “vendor lock-in”, ovvero lo scenario in cui risulta impossibile trasferire la piattaforma e le applicazioni fuori dal cloud del fornitore (a livello IaaS), o, a livello PaaS o SaaS, lo scenario in cui la portabilità della applicazioni risulti problematica a causa di procedure non standard per la memorizzazione dei dati o di accesso ai dati. Tra gli scenari catastrofici, è da valutare la possibilità di collasso di un fornitore di servizi cloud con la relativa ripercussione sul business dell’utente finale. Per concludere l’overview sul cloud computing vengono riassunte di seguito le principali differenze tra i modelli cloud privati e pubblici, senza entrare nel merito del particolare modello o del modello ibrido.

	Cloud Pubblico	Cloud Privato
Rete	Internet	Privata
Localizzazioni e dei server	Globale	All’interno dell’azienda
Costi e tariffazione	Per uso o gratuito	Meccanismo interno, spesso per capacità e processore
Tenancy	Multipla	Singola
Orientamento	Verticale (es. focus sull’utente)	Orizzontale (es. focus sull’applicazione)
Criteri di scelta/ Punti di forza	Costi	Sicurezza

Dal punto di vista aziendale spesso l’attenzione non si sofferma sulla forma con cui l’Information Technology supporta il business, ma sulla riduzione dei costi o l’aumento di profittabilità che un percorso informatico comporta. Il trend tecnologico è diretto verso l’adozione del paradigma cloud computing perché esso offre benefici finanziari e flessibilità del business e non perché rappresenti un paradigma architetturale “migliore” per definizione.

Il Cloud Computing nella sua accezione IaaS può essere visto come un’evoluzione della Virtualizzazione: le macchine virtuali non risiedono su un’infrastruttura fisica (che deve essere anch’essa acquistata e dimensionata preventivamente) ma risiedono su un’infrastruttura virtuale che può crescere in modo flessibile e, soprattutto, fornita “on demand”. Flessibilità e riduzione dell’investimento iniziale sono indubbi vantaggi del Cloud-IaaS rispetto alla Virtualizzazione; tuttavia permane il limite che l’elemento flessibile è l’intera macchina. Il passaggio architetturale successivo è senza dubbio il Cloud nella modalità PaaS dove il concetto di macchina (seppur virtuale) viene completamente superato. Questo approccio comporta ulteriori vantaggi: oltre ad azzerare le attività (e relativi costi) di manutenzione hardware, vengono annullate le attività di manutenzione software in quanto lo stack applicativo (dal sistema operativo all’application server) viene fornito e mantenuto dal provider Cloud PaaS.

Come per la Virtualizzazione il Cloud-IaaS è facilmente applicabile anche a sistemi con architetture di tipo distribuito; al contrario il Cloud-PaaS è integrabile solo con sistemi con un’architettura a servizi.

I diversi ambienti Cloud-PaaS forniscono soluzioni per la comunicazione, la scalabilità e la ridondanza dei servizi.

Nelle nostre sperimentazioni sono in corso vari progetti che prevedono l’adozione del Cloud per diversi aspetti del telecontrollo, per quanto riguarda le funzionalità più critiche (tipicamente quelle legate al tempo reale) è preferibile continuare a seguire metodologie più consolidate.

CONCLUSIONI

Nella memoria sono state analizzate diverse tecnologie, più o meno consolidate; tra queste il Cloud Computing è senza dubbio l'elemento più innovativo e più interessante.

Esso può essere considerato consolidato per applicazioni del mondo web (molti servizi B2C legati a siti e/o *app* mobile hanno il back end su Cloud) e sta entrando nel mondo delle applicazioni ICT tradizionale. È chiaro che il mondo dei sistemi di controllo seguirà questa evoluzione in modo molto più lento, ma è altrettanto evidente che porterà a vantaggi indiscussi. Nel caso di applicazioni SCADA un altro elemento critico è la mappatura del sistema in servizi: l'individuazione di servizi che per forza di cose rimarranno on-premise (tipicamente quelli legati ad esigenze di tempo reale) e quelli che migreranno verso il Cloud.

La transizione qui proposta parte da sistemi attuali di tipo distribuito e descrive un processo di migrazione che vede la Virtualizzazione come primo passo poiché questa può essere fatta mantenendo sostanzialmente inalterati i sistemi tradizionali.

Il passo successivo è la migrazione verso architetture a servizi che rappresenta il fattore chiave e abilitante per l'approccio al Cloud Computing.