WHITE PAPER | GIUGNO 2016

Andare oltre masking e subsetting: realizzare il valore del Test Data Management

Huw Price
CA Technologies

ca technologies

Sommario

Sezione 1

Introduzione

Forse il concetto di Test Data Management (TDM) non è nuovo, ma viene spesso trascurato, sottovalutato e mal compreso, il che impedisce di realizzarne l'intero valore potenziale per il business. Da molte aziende e molti vendor viene considerato solo come un problema ambientale che consiste nella copia, nel masking e nel subsetting dei dati di produzione. Questi dati migrati vengono quindi considerati come una "gold copy" da utilizzare in ambienti di QA e sviluppo; l'obiettivo di un TDM migliore diventa quindi la gestione più rapida di questa copia e della necessità di nuove copie.

La generazione di dati sintetici, se presente, avviene in isolamento, e per singoli progetti. Tutto questo, spesso, in base al presupposto che la generazione dei dati sia pratica valida per un singolo team o progetto, ma impraticabile a livello dell'intera azienda. Le imprese moderne dispongono di solito di grandi database complessi, con decine di milioni di record memorizzati in formati vari e set di strumenti disparati in uso. Di conseguenza, l'idea è che risulta necessariamente più facile copiare i dati di produzione esistenti, piuttosto che tentare di profilare e modellare dati così complessi, come sarebbe necessario per generare dati sintetici realistici.

Si tratta tuttavia di un presupposto discutibile: e le aziende che desiderano realizzare i vantaggi di un TDM migliore dovrebbero riconsiderare la generazione di dati sintetici a livello di impresa, così come il modo in cui avvengono memorizzazione, gestione e provisioning dei dati. La generazione di dati sintetici non solo è più efficace in termini di tempo, qualità e costi, ma spesso si dimostra anche più facile e più sicura del masking completo dei dati di produzione: purché associata alle modifiche adeguate in fatto di tecnologia, processi e struttura dei team.

In questo documento viene espressa una diversa visione dei problemi comuni e dei benefici potenziali del TDM da un punto di vista ambientale, considerando processi, tecnologie e strutture di team comunemente presenti in azienda[1].

Sezione 2

Gli svantaggi di un approccio puramente logistico al TDM

Per le aziende che si basano su dati di produzione per gli ambienti di test e sviluppo, masking e subsetting sono pratiche obbligatorie. Il masking è necessario al fine di soddisfare la legislazione vigente in materia di protezione dei dati, mentre il subsetting dei dati di produzione può aiutare a ridurre i costi di infrastruttura, sempre elevati.

Tuttavia, se il TDM si esaurisce nelle operazioni di masking e subsetting, il suo vero valore di business non viene realizzato. Questo può essere dimostrato partendo da due presupposti: in primo luogo, che i dati raccolti da qualsiasi azienda moderna conterranno informazioni sensibili che, per legge, non possono lasciare gli ambienti di produzione in forma riconoscibile; in secondo luogo, che i dati oggetto di masking sono destinati a scopi di test e di sviluppo, e devono quindi essere sufficienti per tutti i casi di test da eseguire, e per eventuali nuove funzionalità.

In base a questi presupposti, si dimostrerà in primo luogo perché il masking potrebbe in realtà non essere né il modo più semplice, né il più efficace, per eseguire il provisioning di dati sicuri, adatti agli scopi degli ambienti di test e sviluppo. Verranno quindi considerati strutture, tecnologie e processi del team implementati dall'azienda media, per sostenere che la scarsa copertura dei dati di produzione, in aggiunta alla loro modalità di gestione tipica, determina la persistenza di carenze note come ritardi di progetto, sforamento dai budget e difetti in produzione, se ci si affida unicamente a masking e subsetting.

Il masking non è la soluzione semplice

In primo luogo, allora, può essere messa in discussione l'idea che la copia e il masking sicuro dei dati di produzione per gli ambienti di test e sviluppo siano l'opzione più semplice in un contesto IT complesso. In realtà, lo sforzo richiesto dal masking dei dati di produzione mantenendo l'integrità referenziale spesso supererà quello richiesto per modellare i dati; mentre un masking efficace richiede comunque la preventiva profilazione dei dati. Inoltre, alcuni aspetti complessi dei dati vengono spesso esclusi dal masking, in una sorta di compromesso; il masking, allora, in molti casi potrebbe non essere nemmeno l'opzione più sicura.

Se si esegue il masking dei dati per utilizzarli in ambienti non di produzione, le relazioni tra le colonne dei dati originali devono essere mantenute, anche quando il contenuto sensibile è oggetto di masking. A un livello base, questo significa mantenere l'integrità referenziale. A un livello più complesso, tuttavia, significa mantenere i rapporti complessi tra le colonne. Ad esempio, in presenza di un campo "totale", con il calcolo del totale di due o più colonne, queste colonne dovranno rimanere allineate nei dati oggetto di masking. Le più complesse sono le relazioni di dati temporali e le relazioni causali, ad esempio un totale basato sul tempo, di cui è estremamente difficile eseguire il masking in modo coerente.

Non solo: queste relazioni non solo devono essere oggetto di masking in modo coerente all'interno di un database (intra-sistema) ma, per la situazione tipica di una grande azienda, con più tipi di database e applicazioni composite e complesse, devono anche essere mantenute tra database diversi (inter-sistema). Più complessi sono i dati, più difficile diventa tutto questo mentre, al contempo, i dati diventano più facili da violare, dato che aumenta il volume di informazioni che è possibile mettere in correlazione tra loro.

Il più delle volte, il masking dei dati si concentra principalmente sui contenuti, e sulle relazioni intra-sistema e inter-sistema, nella misura in cui è necessario mantenere l'integrità referenziale. La complessità legata all'esigenza di conservare contemporaneamente relazioni tra contenuti, tra colonne e temporali, significa, di solito, che uno di questi elementi viene trascurato; spesso, ad esempio, le relazioni temporali tra colonne rimangono invariate. Sebbene non sensibili da un punto di vista tecnico o legale, queste informazioni conservate possono essere utilizzate per identificare dati sensibili, mettendole in correlazione con informazioni esterne ai dati.

Prendiamo l'esempio di un insieme di registri delle transazioni oggetto di masking: l'autore di un attacco sa che una certa persona ha svolto una transazione di un determinato importo in un momento specifico. Anche se questa informazione non è inclusa nei contenuti sensibili, che sono stati oggetto di masking, nel database oggetto di masking potrebbero essere ancora presenti informazioni temporali, poiché il masking coerente di contenuti, totali numerici e tempi delle transazioni è estremamente difficile. Una volta che identificate nella prima istanza, le informazioni sensibili possono esserlo in più database e sistemi, dato che l'integrità inter-sistema e intra-sistema è stata mantenuta. Questo comporterà un'effettiva de-anonimizzazione dei dati.

Anche se l'attività di profilazione e di masking dei dati è stata svolta, in tutta la sua complessità, non è ancora possibile considerare i dati adeguatamente sicuri, perché le informazioni continueranno a essere mantenute nei loro aspetti più complessi. Ad esempio, l'autore di un attacco potrebbe identificare gli effetti causali a cascata di un registro delle transazioni, combinando informazioni inter-sistema, intra-sistema e causali con metodi temporali, ad esempio un confronto tra istantanee. Sarebbe allora possibile dedurre in che modo i dati si sono modificati durante un'operazione, e perfino eventuali effetti causali a cascata (ad esempio, trigger che insistono sul database). Anche in questo caso, i dati oggetto di masking sono sicuri quanto il loro elemento più debole: una volta identificata questa informazione, l'autore di un attacco può farsi strada attraverso le relazioni tra le colonne, decifrando requisiti sensibili dal punto di vista commerciale o, peggio, dati personali.

A partire dal presupposto che i dati di qualsiasi azienda o quasi conterranno informazioni personali, il masking non costituisce un modo sicuro, o efficace, per il provisioning dei dati agli ambienti non di produzione. Questo è particolarmente vero alla luce della crescente pressione che le aziende subiscono a tutelare pienamente i loro dati: la sanzione media per una violazione dei dati è cresciuta del 13%, arrivando fino a 3,5 milioni di dollari, nel 2013 [5, Ponemon, 2014], mentre l'imminente entrata in vigore del regolamento generale sulla protezione dei dati UE dovrebbe aumentare l'esecutività della legislazione in vigore, che vieta l'utilizzo di informazioni personali per qualsiasi ragione diversa da quelle per cui sono state raccolte.

Considerando inoltre lo "State of Privacy Report 2015" di Symantec, che ha rilevato come i problemi di sicurezza dei dati determinano le scelte dell'88% dei consumatori europei in termini di dove e come fare acquisti, il rischio dell'utilizzo dei dati oggetto di masking in ambienti non di produzione diventa insostenibile. Ora che abbiamo evidenziato i rischi legali della migrazione dei dati di produzione negli ambienti di test e sviluppo, possiamo considerare l'efficacia e l'efficienza di questa operazione.

Persone

Dipendenze dei dati

È raro che un'azienda disponga di un team centrale responsabile per il TDM, e di singoli gruppi responsabili della gestione, della ricerca e della creazione dei propri dati. I team lavorano quindi in isolamento gli uni dagli altri, nei propri ambienti di test o di sviluppo, ma basandosi spesso sulle stesse origini di dati. Questa mancanza di centralizzazione e di collaborazione porta a vincoli di dipendenza dei dati: la modifica apportata a un database da un team influenza tutti gli altri. Ne derivano frustrazioni, ritardi e rilavorazioni: i test falliscono senza motivo apparente e i team sono incapaci di stabilire se la causa sia un difetto nel codice o un errore nei dati. Spesso, inoltre, i team mancano della capacità di eseguire il versioning o la parametrizzazione dei dati di test, e di eseguire il roll back dei database se un altro team li utilizza in modo di fatto esclusivo.

Il ciclo di vita di sviluppo del software viene visto quindi come una serie di fasi lineari, in cui un team completa un'attività e quindi passa il lavoro al team successivo. Di conseguenza, i team di sviluppo e di test troppo spesso si trovano ad aver a che fare con progetti in stallo a causa di ritardi "a monte". Rimangono in attesa della disponibilità dei dati, o del completamento di specifiche e feed di dati da parte di altri team: una quantità di tempo notevole all'interno del ciclo di vita, quindi, è dedicata potenzialmente ad attendere i dati.

Questa mancanza di funzionamento in parallelo è in netto contrasto con la spinta, presente oggi in molte aziende, verso la continuous delivery, in cui i team necessitano dei dati in ogni fase del ciclo di vita. Essa rende impossibile, ad esempio, eseguire efficacemente lo sviluppo end-of-life su versioni precedenti di un sistema, utilizzando al contempo i dati esistenti per le nuove versioni. I team si trovano invece a lavorare in un nuovo ambiente di sviluppo, privo dei dati necessari al suo interno.

Tecnologia

Dipendenze di sistema

Oltre alle dipendenze dei dati, un punto dolente ben noto agli attuali team di test e sviluppo sono i vincoli hardware e di sistema. Le applicazioni sono diventate sempre più complesse nel corso degli ultimi due decenni, con una serie sempre crescente di dipendenze con altri sistemi, sia all'interno che all'esterno dell'azienda. Questo porta spesso a ostacoli al momento di testare un sistema, perché i servizi potrebbero essere instabili o non disponibili. Un'altra preoccupazione dei team di test e sviluppo è la mancanza di ambienti full size. Un team potrebbe scoprire che un altro team ha la priorità sull'ambiente che intende utilizzare o potrebbe non riuscire a ottenere i dati corretti perché sono già utilizzati da un altro team. Anche se in un primo momento questi vincoli potrebbero non sembrare direttamente rilevanti per i dati, si vedrà successivamente come una migliore infrastruttura TDM rappresenti una fase necessaria verso la loro risoluzione.

Storage di dati

Le aziende archiviano grandi quantità di dati di produzione. Disporre di copie di questi database ed eseguirle su macchine di sviluppo risulta lento e costoso. I costi elevati dell'infrastruttura sono collegati allo storage di tali dati, inclusi hardware, licenze e oneri di supporto. Inoltre, i server sono soggetti a richieste crescenti, dovendo fornire volumi elevati di dati ai processi, e contemporaneamente dovendo eseguire connessioni aperte, supportare i file aperti e gestire le injection di dati[2].

A meno di riconsiderare la modalità di gestione dei dati, è improbabile che questi costi diminuiscano. La quantità di dati raccolti e conservati dal medio business raddoppia ogni anno[3], e l'avvento dei "Big Data" ha portato le aziende a parlare di gestione non più di terabyte ma di petabyte di dati. Di conseguenza, lo storage dei dati è diventato una delle quote del budget IT a più rapida crescita; c'è chi sostiene che la crescita annua del settore dello storage dei dati, in tempo recenti, potrebbe essere arrivata al 20%[4]. Le aziende devono quindi valutare la necessità di ogni copia dei dati di produzione: dato che alcune tendono ad avere più copie di un singolo database, la risposta è probabilmente no.

Estrazione e profilazione dei dati

Senza una tecnologia automatizzata, in tutto o in parte, l'individuazione dei dati è una delle sfide più impegnative affrontate dai team che intendono erogare software completamente testato nel rispetto dei tempi, in particolare in un contesto di sviluppo Agile o in un framework di continuous delivery. I tester arrivano a dedicare fino a metà del loro tempo alla ricerca dei dati, e sono costretti a trovare un compromesso tra eseguire tutti i test richiesti, per prevenire il passaggio in produzione di difetti costosi, e fornire il software nel rispetto dei tempi.

Questa situazione è aggravata dall'incoerenza della memorizzazione dei dati in fogli di lavoro non controllati (con riferimenti incrociati o indicizzazione scarsa o nulla, ad esempio), piuttosto che in un deposito o repository centralizzato. Inoltre, i database di solito non sono adeguatamente documentati, a nelle aziende spesso mancano dizionari centrali degli attributi dei dati di test e delle query SQL correlate. L'estrazione e l'allocazione dei dati risultano quindi compromesse, e i team non possono richiedere i dati in base a modelli o moduli standardizzati, o in base ai criteri specifici di cui hanno bisogno. Di conseguenza, dovranno spesso trovare manualmente un insieme limitato di dati idonei da associare ai singoli casi e requisiti di test; un processo lungo e soggetto a errori.

La mancanza di strumenti automatizzati di profilazione e di estrazione dei dati aumenta anche il rischio di mancata compliance. Se i dati vengono memorizzati in fogli di calcolo non controllati, le informazioni sensibili sono potenzialmente reperibili ovunque, ad esempio in una colonna Note, utilizzata quando non era disponibile un campo di colonna adatto, o se tutti i campi rilevanti erano già pieni. Senza la capacità di ricercare campi specifici, è meno probabile che queste informazioni siano reperibili, ed ecco perché potrebbero arrivare in ambienti non di produzione. Questo viola la normativa sulla protezione dei dati, che prevede che i dati possano essere utilizzati solo per il motivo per cui sono stati raccolti; il rischio è di pesanti sanzioni, in media pari a 3,5 milioni di dollari[5], e di un grave danno per la redditività dell'azienda.

Processi

I dati di produzione non sono una "gold copy"

L'incapacità di trovare dati adatti per casi di test e requisiti ci porta al problema principale collegato all'utilizzo di dati di produzione in ambienti non di produzione: la semplice impossibilità di soddisfare la maggioranza delle richieste di dati che un'azienda può ricevere, in particolare dagli ambienti di sviluppo. Come descritto, i team di test e di sviluppo hanno bisogno dei dati in ogni fase dello sviluppo. Un database "gold copy" propriamente detto deve

quindi contenere un set standard di dati da testare più volte, e i dati necessari per soddisfare ogni test possibile. Inoltre, dovrebbe essere nello stato di produzione e aggiornato, includendo dati non validi e tutti i dati precedenti. I dati di produzione soddisfano solo due di queste condizioni: sono nello stato di produzione, ovviamente, e possono includere tutti i dati precedenti. Conseguentemente, non costituiscono una "gold copy".

Molti dati di produzione sono altamente simili, essendo relativi alle operazioni di ordinaria amministrazione, e per loro stessa natura risultano "bonificati", cioè non includono i dati che potrebbero causare il collasso del sistema. Pertanto, non includono dati non validi; implicano, di conseguenza, che nuovi scenari e percorsi negativi non verranno presi in considerazione durante i test. Tuttavia, sono questi risultati imprevisti, valori anomali e condizioni di limite che normalmente causano il collasso del sistema: e l'obiettivo di sviluppo e test dovrebbe essere proprio quello di testare i casi limite, i percorsi non corretti e gli scenari imprevisti. Quando ci si affida unicamente ai metodi di campionamento, i difetti arriveranno inevitabilmente in produzione, dove correggerli costa fino a 1000 volte di più[6], e può richiedere fino a 50 volte più tempo[7].

Inoltre, i dati campionati dalla produzione non saranno quasi mai aggiornati, date le modifiche ambientali e le esigenze di business in costante mutamento che devono essere gestite dai team IT. Poiché i dati non vengono archiviati in modo indipendente dagli ambienti, la modifica di un ambiente richiede un aggiornamento del sistema, o di versione. Questo può portare a un utilizzo esclusivo di dati e scenari basati su più origini di dati di produzione, e determinare la perdita di set di dati utili; set di dati che dovranno quindi essere laboriosamente ricreati a mano. Questi aggiornamenti di sistema, in genere, tendono anche a essere estremamente lenti. Ad esempio, abbiamo lavorato con aziende in cui il completamento di questo processo ha richiesto fino a 9 mesi.

La creazione manuale dei dati può fornire una soluzione a breve termine, consentendo l'esecuzione immediata dei casi di test a portata. Tuttavia, poiché tali dati sono costituiti sulla base di requisiti o casi di test specifici, diventeranno obsoleti quasi immediatamente. Un buon esempio di questo è rappresentato dai tassi di cambio o dai modelli di trading. In quei casi i dati diventano obsoleti ogni giorno, il che significa che i dati creati manualmente non possono essere riutilizzati e così, di solito, sono "bruciati". I nuovi dati devono quindi essere creati faticosamente per ogni test, il che genera ritardi progressivi dei progetti, mentre i test vengono rimandati allo sprint successivo, o alla fase successiva del ciclo di sviluppo.

Se un masking efficace su più database richiede che i dati vengano prima sottoposti a profilazione, ma non garantisce ancora che i dati forniti siano sicuri o di qualità sufficiente, una domanda sorge spontanea: perché, una volta che i dati di produzione sono stati profilati con successo, un'azienda non dovrebbe limitarsi a generare sinteticamente i dati richiesti?

Sezione 3

L'alternativa ideale: persone, processi e tecnologia

Una migliore policy TDM consiste nell'adozione di un approccio strutturato e, soprattutto, centralizzato alla gestione dei dati a livello aziendale. Esso non solo risolve molte delle carenze sopra descritte, ma spesso costituisce, a tutti gli effetti, un modo più economico, più efficiente e perfino più semplice per eseguire il provisioning dei dati agli ambienti di test e sviluppo, con la giusta tecnologia.

L'archiviazione dei dati utili come risorse esterne agli ambienti, e la loro fornitura a richiesta, elimina le problematiche ambientali del TDM, che conseguentemente si concentra sulla modalità di restituzione dei dati. Questa operazione può a sua volta essere eseguita efficacemente mediante un approccio centralizzato allo storage dei dati, in cui questi vengono modellati come risorse riutilizzabili, che è possibile estrarre come sottoinsiemi precisi, a richiesta. Anziché

accontentarsi di masking e subsetting come operazioni necessarie, la generazione di dati sintetici rappresenta quindi un obiettivo strategico che, una volta inserito all'interno di una più ampia policy TDM, consente la delivery di software completamente testato, nel rispetto dei tempi e del budget.

Tecnologia

Profilazione automatizzata dei dati

È già stato detto che, per eseguire efficacemente il masking dei dati di produzione, questi devono prima essere profilati; ma, anche in quel caso, i dati oggetto di masking non saranno completamente sicuri, e non soddisferanno le esigenze di test e sviluppo. Tuttavia, una tecnologia automatizzata per ridurre lo sforzo richiesto dalla profilazione dei dati in un contesto IT complesso esiste. Per profilare i dati è necessario prima "registrarli", raccogliendo il maggior numero possibile di metadati. Questi metadati comprendono nomi di tabella, nomi di colonna e tipi di colonna, ad esempio, e sono presenti perfino per i sistemi di database non relazionali, sotto forma di copybook per i sistemi mainframe e di documenti di mapping per file flat delimitati o a larghezza fissa.

Una volta eseguita la registrazione, è possibile applicare algoritmi di rilevazione dati di tipo matematico. Con CA Test Data Manager (in precedenza Data Maker di Grid-Tools), ad esempio, questa operazione avviene per i singoli schemi, identificando i dati sensibili e, se necessario, eseguendo il reverse engineering delle relazioni di database. Una volta eseguita questa operazione, i sistemi possono essere uniti. In tal modo, CA Test Data Manager utilizza viste "cubiche" per profilare anche le relazioni più complesse all'interno dei dati, creando un insieme di dati multidimensionale in cui ogni dimensione del "cubo" rappresenta un attributo dei dati. Questa profilazione consente all'azienda di comprendere esattamente quali dati sono presenti, dove sono archiviati e di individuare eventuali lacune della copertura funzionale.

Generazione di dati sintetici

Dopo aver costruito un quadro preciso dei dati esistenti e aver identificato i dati aggiuntivi necessari per gli ambienti di test, è possibile generare automaticamente eventuali dati mancanti. Ogni elemento del mondo reale può essere pensato come un punto di dati, quindi i dati possono essere modellati e creati per coprire il 100% delle variazioni funzionali. Questi dati includono scenari futuri mai verificatisi prima, così come "dati non validi", valori anomali e risultati imprevisti. Questo consente un testing negativo efficace e lo sviluppo di un nuovo sistema o sottosistema. Fornisce un modo sistematico per eseguire i test, tale per cui risultati imprevisti e scenari che potrebbero non venire immaginati dai tester non causano il collasso del sistema, mentre i difetti vengono individuati prima di entrare in produzione.

Se i dati non sono sufficienti per ripetere i test più volte, è inoltre possibile creare grandi volumi di dati utilizzando la tecnologia automatizzata. CA Test Data Manager fornisce uno strumento automatico che funziona direttamente con RDBMS o i livelli API ERP, per generare dati ulteriori con l'unico limite della velocità consentita dalla potenza di elaborazione. L'uso di script di massa può raddoppiare la quantità di dati disponibili a un'azienda, alla massima velocità che l'infrastruttura è in grado di gestire. In sintesi, questa tecnologia automatizzata consente la creazione rapida di dati nella situazione di produzione, con tutti i dati necessari per eseguire i test, inclusi test negativi, e dati sufficienti per eseguire test ripetuti.

Gestione centralizzata dei dati

Grazie ai miglioramenti tecnologici citati, l'azienda si è già spinta molto avanti sulla strada della costituzione di una "gold copy", come definita in questo documento (consultare la sezione "I dati di produzione non sono una "gold copy"). Mediante l'ulteriore archiviazione dei dati, modellati come oggetti riutilizzabili, in un deposito di dati di test, diventa possibile anche iniziare ad acquisire la capacità di identificare rapidamente sottoinsiemi di dati specifici per gli ambienti di test e sviluppo, su richiesta.

Clonazione dei dati

Una volta che i dati sono stati modellati come oggetti in un "test mart", o deposito di dati di test, e sono stati definiti dizionari delle risorse di dati e delle query associate, è possibile identificare, clonare e inviare agli ambienti di test e sviluppo sottoinsiemi di dati specifici.

Il modulo di clonazione dati di CA Test Data Manager, ad esempio, estrae insiemi coerenti di dati di test di piccole dimensioni da più sistemi di produzione e sviluppo, tra loro collegati, sostituendo il processo lento e costoso di copia e di spostamento di database grandi e complessi. La capacità di estrarre, copiare e rendere disponibili solo i dati necessari significa che l'azienda non è più costretta a conservare varie copie full size dei database di produzione.

Disporre di un deposito di dati centralizzato ed essere in grado di clonare i dati può rimuovere ulteriormente le dipendenze di dati tra i team, separando provisioning e utilizzo dei dati. Questo significa che i dati possono essere clonati ed erogati a più team in parallelo, eliminando i ritardi dovuti all'attesa che i dati "a monte" diventino disponibili, e impedendo che le modifiche ai dati da parte di un team influenzino negativamente gli altri.

La modellazione e lo storage centralizzato dei dati come oggetti riutilizzabili e malleabili consentono inoltre di riprodurre facilmente bug e scenari di interesse. Con i dati estratti in modo esplicito nelle segnalazioni di bug, la tecnologia di clonazione rapida e flessibile consente la ripetizione di test complessi e rari, senza consumare i dati. Questo è particolarmente utile in caso di esecuzione di un aggiornamento dei dati, perché significa che i dati non devono essere uniti e che insiemi di dati interessanti non andranno perduti.

Vincoli hardware

In combinazione con un kit di strumenti di virtualizzazione, la generazione di dati sintetici può aiutare a risolvere anche i problemi legati a vincoli hardware e di sistema. È possibile simulare i livelli di messaggio, utilizzando i metadati di un sistema per profilare con precisione e generare messaggi di servizio realistici (inclusi file MQ, REST, SOAP, così come file flat). In questo caso un motore automatizzato di creazione dei dati risiede sotto una macchina virtuale, per compilare risposte ai messaggi realistiche, come ad esempio coppie richiesta/risposta.

La virtualizzazione di intere macchine significa inoltre che è possibile creare più ambienti di sviluppo. Questo implica che i team possono lavorare negli ambienti, anche quando i componenti interdipendenti non sono disponibili, evitando ritardi a monte, mentre anche i costosi sistemi legacy e hardware possono essere virtualizzati a scopo di test.

Processi

Parallel development e riutilizzabilità

Oltre alle questioni ambientali, la capacità di individuare e clonare i dati sulla base di attributi di dati specifici risolve anche l'altra preoccupazione essenziale del TDM: ovvero, come eseguire in modo efficiente il provisioning dei dati ai singoli tester o team di test. Essa consente di richiedere, condividere e riutilizzare i dati in parallelo, a richiesta.

L'accesso a un portale di "dati di test a richiesta" centralizzato e basato sul web, come quello fornito con CA Test Data Manager, ad esempio, consente a tester e sviluppatori di richiedere esattamente i dati di cui hanno bisogno per l'attività da svolgere. Quando vengono presentati criteri specifici (ad esempio, attributi dei dati di test), il portale invia un processo al motore di batch, che individuerà i dati appropriati dai sistemi di back-end, o provvederà a clonare i dati e a erogarli. Questo elimina la necessità di cercare manualmente i dati, oppure di crearli a mano, riducendo così in modo significativo il tempo necessario per soddisfare le richieste di dati.

Più standardizzate sono le domande sul modulo, meglio è, dato che questo consente ai team un miglior riutilizzo del lavoro reciproco. Ad esempio, se si dispone di dati sintetici creati in precedenza, è possibile parametrizzare l'input ed esporlo attraverso elenchi a discesa nel portale, consentendo a chiunque di richiedere i dati, anche se il caso di test è diverso. Oltre ai dati di test, framework di creazione di dati, test unitari, risorse virtuali e script di automazione possono essere memorizzati e utilizzati come basi per il lavoro futuro.

Controllo delle versioni

La solida funzione di controllo delle versioni consente il parallelismo necessario per sviluppare continuamente nuovi sistemi, nel rispetto di tempi e budget. Ad esempio, il deposito di dati di test di CA Test Data Manager consente a un team di copiare i dati da un archivio, che in effetti li "eredita", unitamente ai puntatori alle versioni precedenti. Questo permette di gestire l'evoluzione dei dati tra più versioni, in quanto blocca i dati per un determinato team, consentendone il roll back o il roll forward in modo semplice, nonché la riconciliazione con versioni diverse.
Le eventuali modifiche apportate in una posizione si trasmettono tra le versioni, mentre l'originale rimane intatto. Supponiamo che un team abbia necessità di aggiungere una nuova colonna a un intero database: con CA Test Data Manager, se in possesso dei genitori hard-coded, il team può individuare tutti i figli collegati e impostare valori predefiniti, oppure generare dati utilizzando sequenze o funzioni standard.

Persone

Infine, i cambiamenti strutturali dei team spesso possono andare a completare questi miglioramenti tecnologici e di procedura, contribuendo ulteriormente a risolvere le carenze ambientali citate. La centralizzazione del TDM, affidato a un team dedicato, entra in correlazione con la disponibilità di una risorsa centrale di provisioning, gestione e storage dei dati, in grado di soddisfare con maggiore efficienza le necessità dell'impresa. Questo team potrà essere responsabile del provisioning dei dati, nonché della loro gestione, ma anche della creazione di nuovi dati e della loro profilazione, se necessarie.

Questo non solo aiuta a evitare i vincoli di dipendenza dei dati tra i team, ma significa anche che le richieste di dati e il reporting dei bug possono essere accorpati sotto un'unica competenza. Diventa allora possibile applicare vincoli qualitativi, mentre la titolarità dei dati potrà essere centralizzata e attribuita al team di sicurezza IT. La creazione dei moduli dinamici offerta dal portale di dati di test a richiesta di CA Test Data Manager supporta tutto questo, andando oltre i diritti di accesso basati sui ruoli ed eseguendo il provisioning dei dati sensibili solo al personale autorizzato che ne fa richiesta.

Sezione 4

Riferimenti

1 Gli svantaggi dell'utilizzo effettivo dei dati di produzione in ambienti non di produzione sono stati illustrati altrove. Si veda, ad esempio, Huw Price, Ridurre il time-to-market con il Test Data Management e Perché migliorare il Test Data Management è l'unico modo per arrivare alla continuous delivery

2 Jacek Becla and Daniel L. Wang, Lessons Learned from managing a Petabyte, P. 4. Estratto il 19/02/2015 da http://www.slac.stanford.edu/BFROOT/www/Public/Computing/Databases/proceedings/

3 Lessons Learned from managing a Petabyte

4 http://www.computerweekly.com/feature/Meeting-the-demand-for-data-storage

5 http://www.ponemon.org/blog/ponemon-institute-releases-2014-cost-of-data-breach-global-analysis

6 http://benderrbt.com/Bender-Requirements%20Based%20Testing%20Process%20Overview.pdf

7 http://www.softwaretestingclass.com/why-testing-should-start-early-in-software-development-lifecycle/

Sezione 5

Il vantaggio di CA Technologies

CA Technologies (NASDAQ: offre soluzioni di gestione IT che aiutano i clienti a gestire e proteggere ambienti IT complessi, per supportare servizi di business agili. Le aziende utilizzano il software e le soluzioni SaaS di CA Technologies per accelerare l'innovazione, trasformare l'infrastruttura e proteggere dati e identità, dal data center al cloud. L'impegno di CA Technologies è orientato a garantire che i clienti, attraverso l'impiego della sua tecnologia, ottengano i risultati attesi e il business value previsto. Per ulteriori informazioni sui programmi a supporto dei nostri clienti, visita **ca.com/customer-success**. Per ulteriori informazioni su CA Technologies, visitare **ca.com/it**.

Sezione 6

L'autore

Nel corso di una carriera di oltre 30 anni, Huw Price ha ricoperto il ruolo di architetto di infrastruttura per diverse aziende software statunitensi ed europee e ha fornito supporto a progetti per l'architettura di rete di alto livello per banche multinazionali, fornitori di servizi pubblici e sanitari. Eletto "IT Director dell'anno" per il 2010 da QA Guild, Huw nel corso degli anni si è specializzato in strumenti di automazione dei test e ha lanciato numerosi prodotti innovativi che hanno ridefinito il modello di test adottato nel settore del software. Interviene spesso in importanti eventi internazionali e i suoi contributi sono pubblicati da numerose riviste quali Professional Tester, CIO Magazine e altre pubblicazioni tecniche.

L'ultima avventura di Huw, Grid-Tools, è stata acquistata da CA Technologies nel giugno del 2015. Da quasi dieci anni è impegnato a ridefinire le modalità di approccio alle strategie di testing della grande azienda. Grazie all'approccio visionario e alla leadership di Huw, l'azienda ha introdotto un solido orientamento al testing centrato sui dati, proponendo nuovi concetti da lui ideati come "oggetti di dati", "eredità dei dati" e "deposito di dati di test centralizzato".

ca technologies

Entra in contatto con CA Technologies all'indirizzo ca.com/it

CA Technologies (NASDAQ: CA) crea software che promuove l'innovazione all'interno delle aziende, consentendo loro di cogliere le opportunità offerte dall'economia delle applicazioni. Il software rappresenta il cuore di qualsiasi business, in ogni settore. Dalla pianificazione allo sviluppo, fino alla gestione e alla sicurezza, CA Technologies lavora con le aziende di tutto il mondo per cambiare il nostro modo di vivere, interagire e comunicare, in ambienti mobile, cloud pubblici e privati, distribuiti e mainframe. Per ulteriori informazioni, visita il sito **ca.com/it**.

 CS200-205127