

Versione

2.5



» Connectors Manual

Novembre 2017

Author Tecnoteca srl

www.tecnoteca.com

ITA

www.cmdbuild.org

No part of this document may be reproduced, in whole or in part, without the express written permission of Tecnoteca s.r.l.

CMDBuild ® leverages many great technologies from the open source community: PostgreSQL, Apache, Tomcat, Eclipse, Ext JS, JasperReports, IReport, Enhydra Shark, TWE, OCS Inventory, Liferay, Alfresco, GeoServer, OpenLayers, Prefuse, Quartz, BiMserver. We are thankful for the great contributions that led to the creation of that products.

CMDBuild ® è un prodotto di Tecnoteca S.r.l. che ne ha curato la progettazione e realizzazione, è maintainer dell'applicazione e ne ha registrato il logo.



Al progetto ha anche partecipato come committente iniziale il Comune di Udine – Servizio Sistemi Informativi e Telematici.



CMDBuild ® è rilasciato con licenza open source AGPL (<http://www.gnu.org/licenses/agpl-3.0.html>)

CMDBuild ® è un marchio depositato da Tecnoteca Srl .

In tutte le situazioni in cui viene riportato il logo di CMDBuild® deve essere esplicitamente citato il nome del maintainer Tecnoteca Srl e deve essere presente in modo evidente un link al sito del progetto:

<http://www.cmdbuild.org>.

Il marchio di CMDBuild ®:

- non può essere modificato (colori, proporzioni, forma, font) in nessun modo, nè essere integrato in altri marchi
- non può essere utilizzato come logo aziendale nè l'azienda che lo utilizza può presentarsi come autore / proprietario / maintainer del progetto,
- non può essere rimosso dalle parti dell'applicazione in cui è riportato, ed in particolare dall'intestazione in alto di ogni pagina.

Il sito ufficiale di CMDBuild è <http://www.cmdbuild.org>

Sommario

Introduzione.....	4
Documentazione disponibile.....	5
Interoperabilità.....	6
Alimentazione del CMDB.....	6
Caratteristiche generali di un connettore con sistemi esterni.....	6
Soluzioni disponibili nel sistema CMDBuild.....	7
Wizard Connector.....	8
Generalità.....	8
Modalità di funzionamento.....	8
Modalità di configurazione.....	8
Licenza di utilizzo.....	9
Advanced Connector.....	10
Generalità.....	10
Modalità di funzionamento.....	10
Modalità di configurazione.....	12
Esecuzione.....	12
Limitazioni di utilizzo.....	13
Esempi di connettori.....	14
Generalità.....	14
OCS Inventory.....	15
Active Directory.....	19
VMware VCenter.....	21
Apparati LLDP.....	23
Zabbix.....	25
Nagios – GroundWork – NetEye.....	26
Archi.....	31
APPENDICE: Glossario.....	37

Introduzione

CMDBuild è una applicazione Open Source finalizzata a supportare la gestione della configurazione degli oggetti e dei servizi informatici in carico al Dipartimento ICT di una organizzazione e a guidarne i processi di controllo, eventualmente secondo le “best practice” ITIL.

Gestire un Database della Configurazione (CMDB) significa mantenere aggiornata e disponibile per gli altri processi la base dati relativa agli elementi informatici utilizzati, alle loro relazioni ed alle loro modifiche nel tempo.

Con CMDBuild l'amministratore del sistema può costruire ed estendere autonomamente il proprio CMDB (da cui il nome del progetto), modellandolo su misura della propria organizzazione tramite un apposito Modulo di Amministrazione che consente di aggiungere progressivamente nuove classi di oggetti, nuovi attributi e nuove tipologie di relazioni. E' anche possibile definire filtri, “viste” e permessi di accesso ristretti a righe e colonne di ciascuna classe.

CMDBuild è in grado di fornire un completo supporto all'adozione delle “best practice” ITIL, ormai affermatesi come "standard de facto", non proprietario, per la gestione dei servizi informatici secondo criteri orientati ai processi.

Tramite un apposito sistema di gestione dei workflow è possibile definire in modo visuale, con un editor esterno, nuovi processi operanti sulle classi modellate nel database, importarli in CMDBuild ed eseguirli secondo i flussi previsti e con gli automatismi configurati.

E' disponibile un task manager integrato nell'interfaccia utente del Modulo di Amministrazione che consente di gestire in background diverse tipologie di operazioni (avvio di processi, ricezione e invio di mail, esecuzione di connettori) e di controlli sui dati del CMDB (eventi sincroni e asincroni) a fronte delle quali eseguire notifiche, avviare workflow ed eseguire script.

CMDBuild consente la stampa di report tramite il motore open source JasperReports, sia di tipo tabulare prodotti tramite un wizard interno, che di maggiore complessità ottenibili importando template disegnati tramite un apposito editor visuale esterno.

Possono essere poi definite delle dashboard costituite da grafici che mostrino in modo immediato la situazione di alcuni indicatori dello stato corrente del sistema (KPI).

Grazie all'integrazione con il diffuso sistema documentale open source Alfresco è inoltre possibile allegare documenti, immagini, video ed altre tipologie di file alle schede archiviate in CMDBuild.

E' anche possibile utilizzare funzionalità GIS per il georiferimento degli asset e la loro visualizzazione su una mappa geografica (servizi mappe esterni) e / o sulla planimetria di un ufficio (server locale GeoServer) e funzionalità BIM per la visualizzazione di modelli 3D in formato IFC.

Sono poi inclusi nel sistema un webservice SOAP ed un webservice REST, utili per implementare soluzioni di interoperabilità con architettura SOA.

CMDBuild comprende di base due framework denominati Basic Connector e Advanced Connector, che tramite il webservice SOAP sono in grado di sincronizzare le informazioni registrate nel CMDB con fonti dati esterne, ad esempio con sistemi di automatic inventory (quali lo strumento open source OCS Inventory) o con sistemi di virtualizzazione o di monitoraggio.

Un ulteriore strumento, il CMDBuild GUI Framework, consente invece tramite il webservice REST di pubblicare su portali esterni pagine web personalizzate in grado di interagire con il CMDB.

E' infine disponibile una interfaccia utente ottimizzata per strumenti “mobile” (smartphone e tablet), implementata come “app” multiplatforma (iOS, Android) e anch'essa collegata a CMDBuild tramite il webservice REST.

Documentazione disponibile

Il presente manuale è dedicato alla configurazione di connettori tramite cui sincronizzare in CMDBuild, nelle diverse modalità tecniche attualmente disponibili, informazioni gestite in applicazioni e database esterni.

Sono disponibili sul sito di CMDBuild (<http://www.cmdbuild.org>) ulteriori specifici manuali dedicati a:

- system overview ("Overview Manual")
- utilizzo del sistema ("User Manual")
- amministrazione del sistema ("Administrator Manual")
- installazione e gestione tecnica del sistema ("Technical Manual")
- configurazione del workflow ("Workflow Manual")
- utilizzo del webservice per l'interoperabilità con sistemi esterni ("Webservice Manual")

Interoperabilità

Alimentazione del CMDB

La gestione dei servizi IT in enti ed aziende di medie o grandi dimensioni è necessariamente effettuata con l'ausilio di più sistemi informatici specializzati che devono poter cooperare nella gestione delle proprie attività e delle proprie informazioni.

La raccolta ed il controllo manuale delle informazioni gestite nel CMDB possono introdurre problemi di ritardo o imprecisione nell'aggiornamento dei dati ed è quindi conveniente, ove possibile, aggiornarle in modalità automatica.

Diventa quindi importante poter configurare connettori con sistemi esterni per sincronizzare in CMDBuild (sistema CMDB centrale) dati la cui gestione principale ("master") avviene su altre applicazioni specialistiche, fra cui ad esempio:

- sistemi di automatic inventory, per confronto automatico dei dati tecnici degli asset e del software installato e gestione delle differenze rilevate (OCS Inventory o altri prodotti)
- sistemi di controllo (virtualizzazione, ecc), da cui ricavare informazioni sulla configurazione corrente dell'infrastruttura IT
- sistemi di rilevamento di informazioni relative alla esecuzione di servizi e applicazioni server
- sistemi di monitoraggio, da cui rilevare malfunzionamenti sistemistici ed avviare workflow di Incident Management
- directory LDAP, come repository per l'archivio del personale
- sistemi HR, come soluzione alternativa per ricevere variazioni all'elenco del personale
- sistemi ERP, per acquisire dati amministrativi (cespiti, fornitori, ecc)

Caratteristiche generali di un connettore con sistemi esterni

In generale un connettore con sistemi esterni deve essere in grado di:

- contattare il sistema di interesse (tool di automatic inventory, directory LDAP, ecc), attraverso uno fra i possibili canali di comunicazione (accesso diretto al DB, webservice SOAP, webservice REST, altre API, mail strutturate, file CSV, ecc)
- accedere alle specifiche informazioni da sincronizzare (lista di asset, lista di persone, ecc), estratte tramite specifiche query SQL oppure chiamate di metodi SOAP oppure utilizzo di API, ecc
- applicare gli opportuni criteri di "mapping" fra le informazioni di origine e le corrispondenti informazioni da riportare in CMDBuild, considerando:
 - chiave univoca fonte dati di origine <=> chiave univoca classe CMDBuild
 - entità / informazione fonte dati di origine (ad esempio tabella / colonna database oppure elemento / proprietà restituiti da un metodo SOAP) <=> entità / informazione di destinazione (classe / attributo CMDBuild)
 - eventuali elementi di logica applicativa da applicare nell'ambito dell'operazione di "mapping" (ad esempio applicazione di regole di aggregazione / disaggregazione di informazioni, esecuzione di operazioni quali log o invio di notifiche, registrazione di informazioni nel sistema esterno, ecc)

Un connettore potrà poi limitarsi alla sincronizzazione dei dati non aggiornati rispetto al sistema “master” esterno, oppure potrà sottoporre la necessità di variazione di quelle informazioni ad un processo autorizzativo di Change Management (suggerito per le differenze classificate come “critiche”).

In una stessa istanza CMDBuild può essere utile attivare più connettori con diversi sistemi esterni, ciascuno con proprie regole di elaborazione e proprie regole di schedulazione.

Soluzioni disponibili nel sistema CMDBuild

CMDBuild mette a disposizione due diverse soluzioni per attivare e schedulare connettori di sincronizzazione dati con sistemi esterni:

- una prima soluzione interna a CMDBuild denominata Wizard Connector, completamente configurabile da interfaccia utente tramite “wizard”, utilizzabile nei casi più semplici in cui le regole di “mapping” non richiedano una specifica logica applicativa
- una seconda soluzione esterna a CMDBuild denominata Advanced Connector, basata su un framework realizzato in linguaggio Java e programmabile tramite il linguaggio di scripting Groovy, utilizzabile nei casi più complessi (diverse fonti dati gestite, logica applicativa complessa, volumi di dati elevati)

Nelle precedenti versioni del presente manuale era documentata una ulteriore soluzione denominata Basic Connector, configurabile tramite il linguaggio di trasformazione XSLT / XPATH e basata sull'accesso diretto a fonti dati esterne. Tale soluzione non è più supportata da Tecnoteca.

E' ovviamente sempre possibile rinunciare alle librerie ed alle primitive messe a disposizione dall'Advanced Connector, e realizzare da zero soluzioni di interoperabilità esterne a CMDBuild che implementino le logiche desiderate accedendo al suo database tramite i webservice pubblici REST e SOAP.

Wizard Connector

Generalità

La versione del connettore basata sul wizard interno consente di risolvere i casi più semplici di sincronizzazione dati, in cui le regole di “mapping” non richiedano una specifica logica applicativa.

Essendo completamente configurabile da interfaccia utente ha anche ovviamente il vantaggio di poter essere utilizzato in modo immediato, senza necessità di complesse operazioni di attivazione.

Per contro il Wizard Connector ha alcune limitazioni:

- gestisce solo regole di mapping semplificate
- accede solamente a fonti dati esterne di tipo database relazionale (PostgreSQL, MySQL, Oracle, SQLServer)
- ogni istanza può sincronizzare solamente una classe CMDBuild ed eventuali classi di dettaglio di primo livello collegate con domini 1:N
- non gestisce la possibilità di avviare processi autorizzativi di Change Management per la risoluzione delle variazioni riscontrate sui dati importati

Modalità di funzionamento

Il Wizard Connector è implementato come funzionalità interna all'applicazione CMDBuild ed è basato sui seguenti criteri:

- deve essere disponibile l'accesso ad una fonte dati esterna della tipologia prevista
- nella fonte dati esterna è disponibile un sottoinsieme di informazioni di interesse di CMDBuild, già organizzate in modo da poter essere “mappate” in modo diretto sulle classi / attributi di CMDBuild (tabelle o viste o altre strutture dati per fonti dati diverse da database relazionali)
- l'amministratore di CMDBuild configura da interfaccia utente i criteri di “mapping” delle informazioni
- a intervalli definiti in fase di schedulazione del connettore viene eseguito un task che:
 - legge le informazioni presenti nelle viste configurate
 - le confronta con le informazioni corrispondenti in CMDBuild
 - esegue le operazioni di aggiornamento
 - al termine invia una mail di notifica oppure esegue un workflow oppure esegue uno script, secondo le modalità di configurazione previste

Modalità di configurazione

Il Wizard Connector è una delle diverse tipologie di task gestite in CMDBuild con il Task Manager.

La sua configurazione avviene quindi tramite un apposito wizard che richiede passo dopo passo tutte le informazioni necessarie per schedulare ed eseguire il task.

Segue la pagina iniziale del wizard.

The screenshot displays the CMDBuild administration interface. At the top, the user is identified as 'Bottega Fabio' with a 'Logout' link, and the group is 'Multi group' with a link to the 'Data management module'. The interface is titled 'Task manager - All' and features a table of tasks. A dropdown menu is open, showing options for 'Connector', 'E-mail', 'Event', 'Process', and 'Others'. The table lists several tasks, including 'SchedulerContatti', 'NotificaRisposte', and various 'Gestione Richieste' tasks. Below the table, there are buttons for 'Modify task', 'Remove task', and 'Clone task'. The main configuration area shows the 'Type' set to 'Connector', the 'Description' as 'Wizard Connector example', and the 'Start on save' checkbox unchecked. At the bottom, there are 'Previous', 'Save', 'Cancel', and 'Next' buttons. The footer contains the URL 'cmdbuild.tecnoteca.com/cmdbuild/administration.jsp#', 'Info & Support', and 'Copyright © Tecnoteca srl'.

Connector	Description	Active
E-mail	SchedulerContatti	✓
E-mail	NotificaRisposte	✓
Event	Gestione Richieste - Notifica risposte clienti	✓
Process	Gestione Richieste - Web - Ricezione ticket	✓
Others	Gestione Richieste - Web - Notifica risposte clienti	✓
E-mail	Gestione Richieste - Ricezione ticket	✓
Event synchronous	Notifica incremento richiesta 2	✗

Per una descrizione completa e più dettagliata delle modalità di configurazione del connettore si rimanda al Manuale di Amministrazione di CMDBuild.

Licenza di utilizzo

Trattandosi di una funzionalità dell'interfaccia utente di CMDBuild si applica la stessa licenza open source AGPL 3.0 dell'applicazione base.

Advanced Connector

Generalità

L'Advanced Connector è anch'esso costituito da una applicazione esterna a CMDBuild, pur essendo poi “schedulabile” sia tramite l'apposito servizio di sistema, che tramite il Task Manager interno a CMDBuild.

E' basato su un framework Java predefinito che implementa le logiche principali utili per attività di sincronizzazione e richiede poi che il comportamento specifico di ciascuna istanza del connettore venga configurato tramite il linguaggio di scripting Groovy.

Anche l'Advanced Connector può essere configurato per interagire con un processo di gestione (normalmente un workflow di Change Management in ottica ITIL) che sottoponga le modifiche potenzialmente critiche all'approvazione di un utente responsabile (normalmente il Change Manager).

E' la soluzione più completa fra quelle disponibili con CMDBuild, garantendo:

- la possibilità di operare con qualsiasi applicazione esterna che renda accessibile il proprio database oppure disponga di webservice / API oppure produca file di testo / mail contenenti le informazioni da sincronizzare
- possibilità di sincronizzazione contestuale di strutture dati anche complesse
- buone prestazioni anche con volumi di dati elevati
- capacità illimitata di personalizzazione, essendo il connettore configurabile tramite un linguaggio di programmazione standard

Unica limitazione dell'Advanced Connector è proprio quella di richiedere competenze di programmazione per la configurazione di ciascuna diversa istanza.

Modalità di funzionamento

L'Advanced Connector è composto da una serie di moduli che vengono eseguiti in sequenza secondo quanto definito nel file di configurazione:

```
{CONNECTOR_HOME}/conf/connector.properties
```

Tali moduli possono fra l'altro:

- compiere operazione di input/output leggendo dalla fonte dati estera e scrivendo sul database di CMDBuild attraverso i suoi webservices
- inviare eventuali mail di notifica con report allegati
- elaborare opportunamente le informazioni lette in input per normalizzare i dati o per renderli compatibili con la struttura dati di CMDBuild

Le operazioni che esegue ogni modulo sono descritte nei file di configurazione definiti nella cartella:

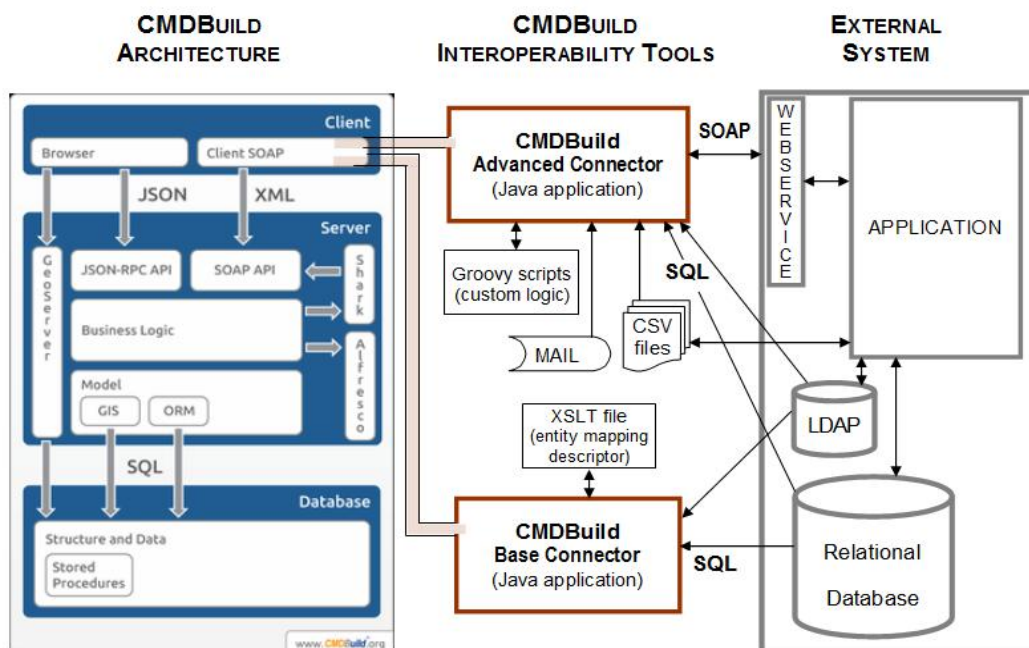
```
{CONNECTOR_HOME}/conf
```

Ogni file di configurazione consiste in uno script in linguaggio Groovy che viene opportunamente interpretato dal connettore il quale quindi esegue la sequenza di istruzioni definite nello script stesso.

Nel dettaglio i moduli da configurare sono i seguenti:

1. schema module: connector-schema.groovy
Definisce lo schema della struttura dati di CMDBuild su cui il connettore mantiene aggiornati i dati letti dalla sorgente esterna.
2. sql module: connector-sql.groovy
E' il modulo di input che esegue la lettura delle informazioni dalla sorgente dati esterna ed esegue una prima normalizzazione delle informazioni lette. Tale modulo inoltre esegue un controllo sui dati letti verificando eventuali anomalie o inconsistenze tali da necessitare l'invio di una notifica.
3. modify on remove module: connector-mor.groovy
E' il modulo di output che semplicemente imposta il connettore in modifica anche sulle informazioni non più presenti nella base dati esterna (DBO.DATI) che normalmente verrebbero rimosse dal database di CMDBuild (in questo modo invece vengono solamente impostate come non attive e non rimosse dalla struttura dati).
4. mail module: connector-mail.groovy
E' il modulo che permette di leggere una mailbox e di inviare eventuali mail di notifica se necessario.
5. È presente inoltre un ulteriore modulo di output (implicito) che definisce l'end-point dei webservices di CMDBuild, utilizzato dal connettore per scrivere le informazioni su CMDBuild nelle tabelle opportunamente definite nel modulo "schema".

Si riporta di seguito uno schema di come l'Advanced Connector ed il Basic Connector interagiscano con CMDBuild e con l'applicazione / sistema esterno con cui è richiesta la sincronizzazione dei dati.



Modalità di configurazione

Struttura del file system

Il connettore può essere posizionato in una cartella a piacere nel filesystem del server, che nel seguito sarà riferita con il nome di `${CONNECTOR_HOME}`.

Si riporta di seguito la struttura di default del filesystem.

```
-- ${CONNECTOR_HOME}
|
|--bin
|   |--connector.sh
|
|--config
|   |--connector-extension.properties
|   |--connector.properties
|   |--connector-sql-context.xml
|   |--log4j.properties
|   |--template
|   |   |-- ...
|
|-- lib
|   |--....
```

Il file bin/connector.sh

E' lo script necessario per eseguire l'applicazione. Contiene la configurazione ed i parametri di default necessari all'esecuzione.

Il file config/connector.properties

Il file descrive i moduli presenti e la loro organizzazione (catene):

```
modules.schema=
modules.in.source=
modules.in.cmdbuild=ws
modules.out=
module.ws.classname=org.cmdbuild.connector.ws.WsModule
```

Devono essere specificati per lo meno un modulo schema, ed un modulo di input ed un modulo di output. Di default è già presente il modulo necessario per la comunicazione con CMDBuild tramite web service.

La directory config/template

La directory contiene i template dei file di configurazione per tutti i moduli disponibili.

I file di configurazione dei moduli

Per ogni modulo aggiunto, all'interno della directory `${CONNECTOR_HOME}/config` dovranno essere presenti dei file "connector-`{MODULE_NAME}.properties`" e "connector-`{MODULE_NAME}.groovy`", dove "`{MODULE_NAME}`" rappresenta il nome del modulo come specificato all'interno del file.

Esecuzione

Una volta editati i file sopra descritti, il connettore potrà essere avviato eseguendo lo script:

```
connector.sh
```

L'esecuzione del connettore può essere programmata sia tramite il Task Manager interno a

CMDBuild che tramite lo scheduler di sistema (il servizio “cron” in ambiente Unix / Linux).

Per il primo caso si rimanda all'Administrator Manual.

Per il secondo caso si suggerisce di inserire nella cartella `/etc/cron.d/` un file aggiuntivo, ad esempio “connectors” in cui specificare, secondo la sintassi cron, quando eseguire i diversi connettori.

Un esempio di schedulazione è il seguente:

```
00 20 * * * root sh ${CONNECTOR_HOME}/bin/connectors.sh
```

Si ricorda di riavviare il servizio cron una volta inserita la schedulazione.

Limitazioni di utilizzo

L'Advanced Connector è reso disponibile con licenza non open source, che ne consente l'utilizzo solamente a chi ha sottoscritto un servizio di manutenzione sull'applicazione CMDBuild con Tecnoteca Srl e fino a che tale servizio è attivo.

L'Advanced Connector viene comunque fornito con il codice sorgente di tutti i suoi componenti a chi ha sottoscritto un servizio di manutenzione con Tecnoteca Srl.

Esempi di connettori

Generalità

Essendo CMDBuild un ambiente del tutto personalizzato da parte di ciascun utilizzatore, non è possibile rendere disponibili connettori standard già funzionanti su una generica istanza dell'applicazione di cui non è nota la configurazione.

Dal momento che l'obiettivo del connettore è quello di sincronizzare informazioni sulla base di criteri di "mapping" fra il modello dati del sistema esterno ed il modello dati di CMDBuild, la completa libertà nella configurazione di quest'ultimo rende di fatto necessario procedere anche ad una corrispondente configurazione del connettore.

Allo stesso modo gli eventuali meccanismi di approvazione dei Change più critici consentiti dalla logica del connettore dovranno essere adattati ai corrispondenti workflow configurati in CMDBuild.

Per questo motivo le tre tipologie di connettori descritte alle pagine precedenti prevedono tutti delle opzioni di configurazione, sia pure con gradi di difficoltà diversi (interfaccia utente => editing di file di testo => scrittura di codice di programmazione).

Alle pagine successive vengono descritti alcuni connettori già implementati, che vanno quindi intesi come esempi di implementazione e non come tool già pronti e direttamente utilizzabili.

Del connettore con OCS Inventory, essendo disponibile anche come file scaricabile dal sito di CMDBuild vengono descritti più nel dettaglio tutti i criteri di configurazione.

OCS Inventory

Generalità

Il controllo e aggiornamento manuale delle variazioni nell'inventario degli asset IT richiede risorse importanti che non sempre sono disponibili.

Si consiglia quindi di automatizzare la sincronizzazione di quelle informazioni tramite utilizzo di un tool di automatic inventory ed in particolare OCS Inventory (<http://www.ocsinventory-ng.org/en/>) è il prodotto suggerito agli utilizzatori di CMDBuild, considerandone sia le caratteristiche tecniche che lo licenza open source.

OCS Inventory consente di:

- attivare dei programmi “agent” sui computer da controllare (Windows, Linux o Mac OS)
- raccogliere informazioni relative a BIOS, processore, RAM, device di input, controller, periferiche, schede aggiuntive, impostazioni rete, sistema operativo, applicazioni software, ecc
- inviare periodicamente a server tali informazioni sotto forma di file XML
- archiviare tali informazioni in un database relazionale MySQL
- consultare ed modificare tali informazioni da una apposita applicazione web di gestione

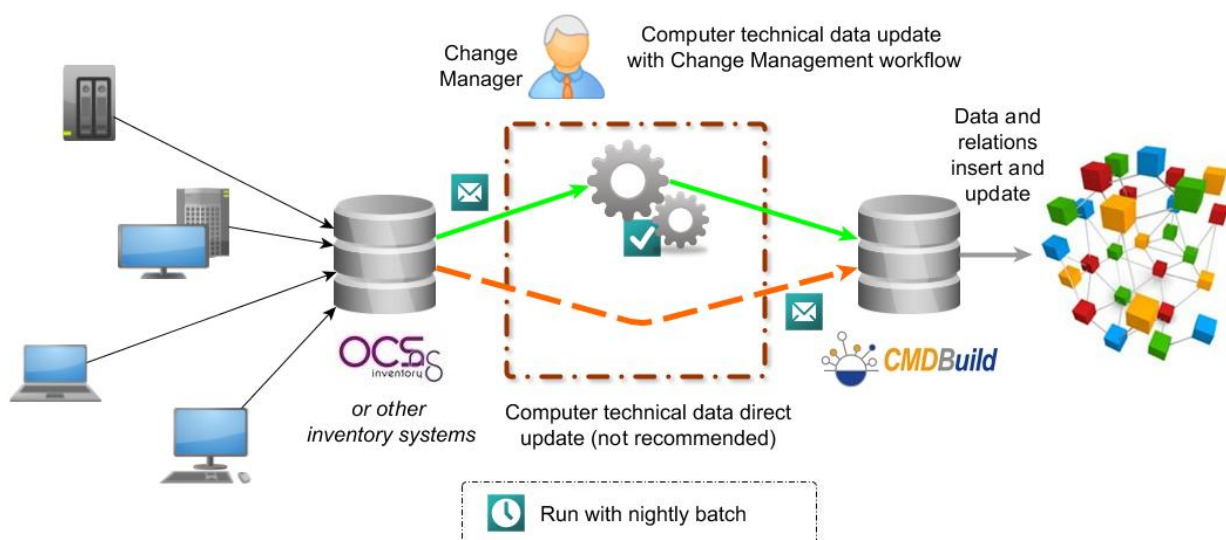
Connettore CMDBuild

Obiettivo del connettore è quello di sincronizzare l'archivio dei computer registrati in CMDBuild con i dati disponibili nel database di OCS Inventory.

In alcune implementazioni (ad esempio in quella inclusa nell'applicazione CMDBuild `READY2USE`) il connettore si occupa anche di sincronizzare le istanze del software “controllato” (di cui si vuole verificare le licenze) rilevate da OCS Inventory con i dati disponibili nel database di CMDBuild.

Al termine il connettore può inviare un report con la lista delle modifiche apportate nel CMDB.

Si riporta di seguito uno schema del connettore.



Attributi disponibili ed esempi di “mapping”

Si riporta di seguito la lista delle entità ed informazioni rilevati da OCS Inventory fra le quali dovranno essere individuate quali “mappare” in CMDBuild, ed eventualmente su quali classi ed attributi (ultima colonna).

OCS Inventory	Vista I/F CMDBuild	Descrizione	Esempio	CMDBuild	
Hardware.Id AccountInfo.Tag -- Hardware.Name Hardware.Workgroup Hardware.UserDomain -- -- Hardware.OSName Hardware.OSVersion Hardware.OSComments Hardware.ProcessorT Hardware.ProcessorS Hardware.ProcessorN Hardware.Memory Hardware.WinProdKey	CMDBuild_device	Id Tag Serial Name Workgroup UserDomain Manufacturer ProductName OSName OSVersion OSComments ProcessorS ProcessorN Memory WinProdKey	Id Chiave Numero di serie Nome Gruppo lavoro Dominio Produttore Nome prodotto Nome sistema operativo Versione sistema operativo Service Pack Velocità processore Numero processori Memoria RAM Chiave prodotto	2 lisa HUB5390NQK Kelly Tecnoteca Kelly Hewlett-Packard HP Compaq nc6120 (PN936AV) Microsoft Windows XP Professional 5.1.2600 1832 1 512	
Bios.Hardware_Id Bios.SManufacturer Bios.SModel Bios.SSN Bios.Type Bios.BManufacturer Bios.BVersion Bios.BDate	CMDBuild_bios	Device_Id SManufacturer SModel SSN Type BManufacturer BVersion BDate	Riferimento macchina Produttore MotherBoard Modello MotherBoard Numero di serie M.B. Tipo chassis Produttore Versione Data	2 VIA K7Upgrade-600 SYS-1234567890 Desktop American Megatrends Inc. AMIINT-10-SMBiosVersion: P1.50 N/A	
Controllers.Hardware_Id Controllers.Manufacturer Controllers.Name Controllers.Caption Controllers.Description Controllers.Version Controllers.Type	CMDBuild_controller	Device_Id Manufacturer Name Caption Description Version Type	Riferimento macchina Produttore Nome Titolo Descrizione Versione Tipo	2 VIA Technologies, Inc. Controller IDE VIA Bus Master Controller IDE VIA Bus Master Controller IDE VIA Bus Master N/A IDE Controller	
Drives.Hardware_Id Drives.Letter Drives.Type Drives.FileSystem Drives.Total Drives.Free Drives.NumFiles Drives.VolumN	CMDBuild_drive	Device_Id Letter Type FileSystem Total Free NumFiles VolumN	Riferimento macchina Lettera Tipo File System Spazio totale Spazio libero Numero files Nome volume	2 C:/ Hard Drive NTFS 39997 15708 0	
Inputs.Hardware_Id Inputs.Type Inputs.Manufacturer Inputs.Caption Inputs.Description Inputs.Interface Inputs.PointType	CMDBuild_input	Device_Id Type Manufacturer Caption Description Interface PointType	Riferimento macchina Tipo Produttore Titolo Descrizione Interfaccia Tipo puntatore	2 Pointing (Periferiche standard di sistema Periferica USB Human Interface Periferica USB Human Interface USB N/A	

Memories.Hardware_Id Memories.Caption Memories.Description Memories.Capacity Memories.Purpose Memories.Type Memories.Speed Memories.NumSlots	CMDBuild_memory	Device_Id Caption Description Capacity Purpose Type Speed NumSlots	Riferimento macchina Titolo Descrizione Dimensione Scopo Tipo Velocità Numero slots	2 Memoria fisica DIMM1 (Other ECC) 512 Reserved Unknown N/A 1	
Modems.Hardware_Id Modems.Name Modems.Model Modems.Description Modems.Type	CMDBuild_modem	Device_Id Name Model Description Type	Riferimento macchina Nome Modello Descrizione Tipo		
Monitors.Hardware_Id Monitors.Manufacturer Monitors.Caption Monitors.Description Monitors.Type Monitors.Serial	CMDBuild_monitor	Device_Id Manufacturer Caption Description Type Serial	Riferimento macchina Produttore Titolo Descrizione Tipo Numero di serie	2 Unknown manufacturer code ACR AL1716 RGB color	
Networks.Hardware_Id Networks.Description Networks.Type Networks.TypeMIB Networks.Speed Networks.MCAddr Networks.Status Networks.IPAddress Networks.IPMask Networks.IPGateway -- Networks.IPSubnet Networks.IPDHCP	CMDBuild_network	Device_Id Description Type TypeMIB Speed MACAddr Status IPAddress IPMask IPGateway IPDNS IPSubnet IPDHCP	Riferimento macchina Descrizione Tipo TipoMIB Velocità Macaddress Stato Indirizzo IP Maschera IP IP Gateway IP DNS IP sottorete IP DHCP server	2 Scheda Fast Ethernet VIA compat. Ethernet EthernetCsmacd 100 Mb/s 00:0B:6A:AE:DE:6B Up 192.168.2.199 255.255.255.0 192.168.2.1 192.168.2.100 192.168.2.0 255.255.255.255	
Ports.Hardware_Id Ports.Type Ports.Name Ports.Caption Ports.Description	CMDBuild_port	Device_Id Type Name Caption Description	Riferimento macchina Tipo Nome Titolo Descrizione	2 Serial Porta di comunicazione (COM1) Porta di comunicazione (COM1) Porta di comunicazione	
Slots.Hardware_Id Slots.Name Slots.Description Slots.Designation Slots.Purpose Slots.Status Slots.PShare	CMDBuild_slot	Device_Id Name Description Designation Purpose Status PShare	Riferimento macchina Nome Descrizione Codice Utilizzo Stato Condivisione	2 Slot di sistema Slot di sistema PCI1 OK 1	
Softwares.Hardware_Id Softwares.Publisher Softwares.Name Softwares.Version Softwares.Folder Softwares.Comments Softwares.FileName Softwares.FileSize Softwares.Source	CMDBuild_software	Device_Id Publisher Name Version Folder Comments FileName FileSize Source	Riferimento macchina Produttore Nome Versione Cartella Commento Nome file Dimensione file Sorgente	2 OpenOffice.org OpenOffice.org 2.0 2.0.9044 OpenOffice.org 2.0 (en-US)(Build:9044) N/A 0 1	

Sounds.Hardware_Id Sounds.Manufacturer Sounds.Name Sounds.Description	CMDBuild_sound	Device_Id Manufacturer Name Description	Riferimento macchina Produttore Nome Descrizione	2 Silicon Integrated Systems [SiS] AC'97 Multimedia audio controller rev a0	
Storages.Hardware_Id Storages.Manufacturer Storages.Name Storages.Model Storages.Description Storages.Type Storages.DiskSize	CMDBuild_storage	Device_Id Manufacturer Name Model Description Type DiskSize	Riferimento macchina Produttore Nome Modello Descrizione Tipo Dimensione disco	2 (unità disco standard) Generic Flash Disk USB Device //./PHYSICALDRIVE1 Unità disco Removable media other than floppy 117	
Videos.Hardware_Id Videos.Name Videos.Chipset Videos.Memory Videos.Resolution --	CMDBuild_video	Device_Id Name Chipset Memory Resolution RefreshRate	Riferimento macchina Nome Chipset Memoria Risoluzione Frequenza refresh	2 RADEON 9200 PRO Family (Microsoft C.) RADEON 9200 PRO AGP (0x5960) 128 1280 x 1024 64	

E' importante valutare con attenzione quale attributo utilizzare come campo univoco del computer: un numero di cespide interno (da specificare "una tantum" all'installazione dell'Agente OCS tramite l'apposito flag "tag"), il MAC Address della scheda di rete (non necessariamente invariante in caso di una sua sostituzione), ecc.

Active Directory

Generalità

Pur essendo una tipologia di informazione normalmente trattata da sistemi dedicati, è in genere utile disporre anche in CMDBuild dell'archivio del personale, consentendo in tal modo di correlare ad esempio i computer agli assegnatari, le attività di assistenza ai richiedenti, le stanze agli occupanti, etc.

E' quindi conveniente attivare un connettore per la sincronizzazione automatica dei dati fra l'archivio "master", che molto spesso è gestito con il servizio Active Directory (o comunque con sistemi LDAP), e CMDBuild.

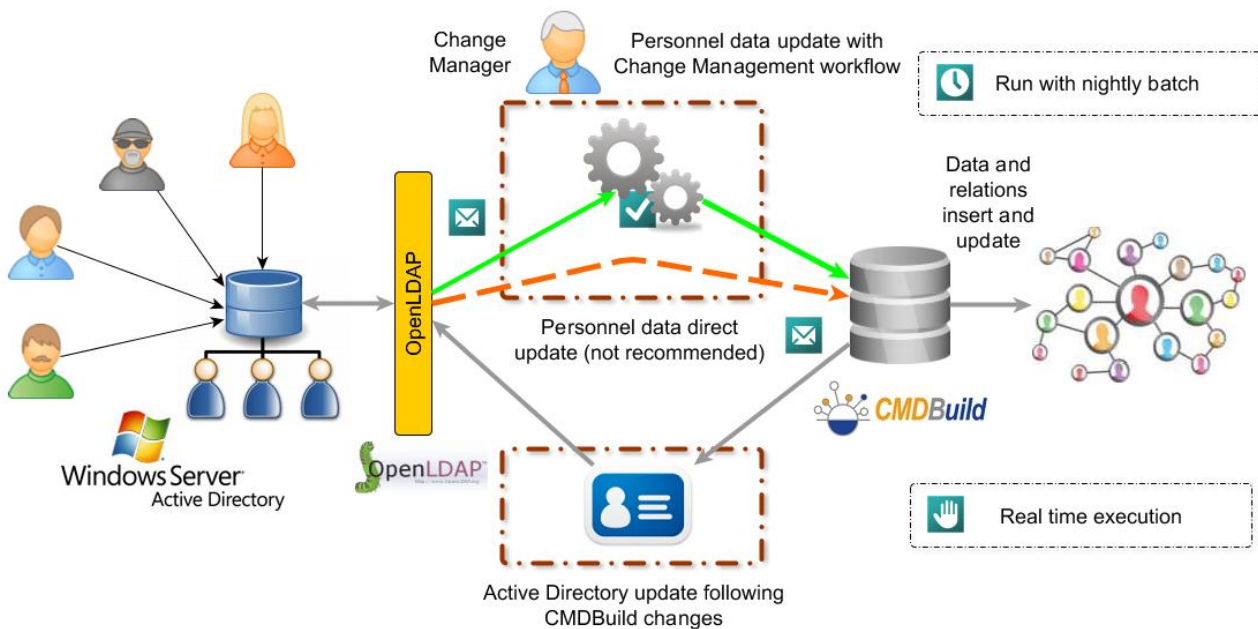
Connettore CMDBuild

Obiettivo del connettore è quello di sincronizzare l'archivio del personale registrato in CMDBuild con i dati disponibili nel database LDAP.

In alcune implementazioni il connettore si occupa anche di sincronizzare da LDAP la lista degli uffici aziendali cui il personale appartiene (organigramma).

Al termine il connettore può inviare un report con la lista delle modifiche apportate nel CMDB.

Si riporta di seguito uno schema del connettore.



Attributi disponibili ed esempi di "mapping"

Segue un esempio di mapping tratto dal connettore LDAP incluso nell'applicazione CMDBuild READY2USE:

CAMPO ACTIVE DIRECTORY	CLASSE CMDBUILD	ATTRIBUTO CMDBUILD	CHIAVE
uid	InternalEmployee	Code	sì

givenName	InternalEmployee	FirstName	
sn	InternalEmployee	LastName	
mail	InternalEmployee	Email	
telephonenumber	InternalEmployee	Phone	
'Active'	InternalEmployee	State	

.

VMware VCenter

Generalità

VMware è uno degli ambienti di virtualizzazione leader di mercato, mentre VCenter è il tool messo a disposizione da VMware per la gestione dell'infrastruttura.

Il connettore descritto al presente paragrafo ha lo scopo di sincronizzare in CMDBuild l'inventario sempre aggiornato dei server virtuali in uso nell'infrastruttura VMware, con:

- informazioni relative alle risorse assegnate (sistema operativo, RAM, CPU, etc)
- informazioni relative allo stato del server
- altre informazioni gestite in modo personalizzato da ciascuna organizzazione utilizzando l'interfaccia grafica dell'applicazione (referente tecnico, cluster di appartenenza, ecc)

E' vero che anche i tool di automatic inventory (fra cui OCS Inventory) sono in grado di rilevare alcune informazioni tecniche relative ai server virtuali, ma tali informazioni non sono complete quanto quelle messe a disposizione da ciascuno specifico sistema di virtualizzazione.

Il connettore può sincronizzare in CMDBuild tutte le macchine virtuali connesse a VMware VCenter, a prescindere dal loro stato (acceso, spento, etc).

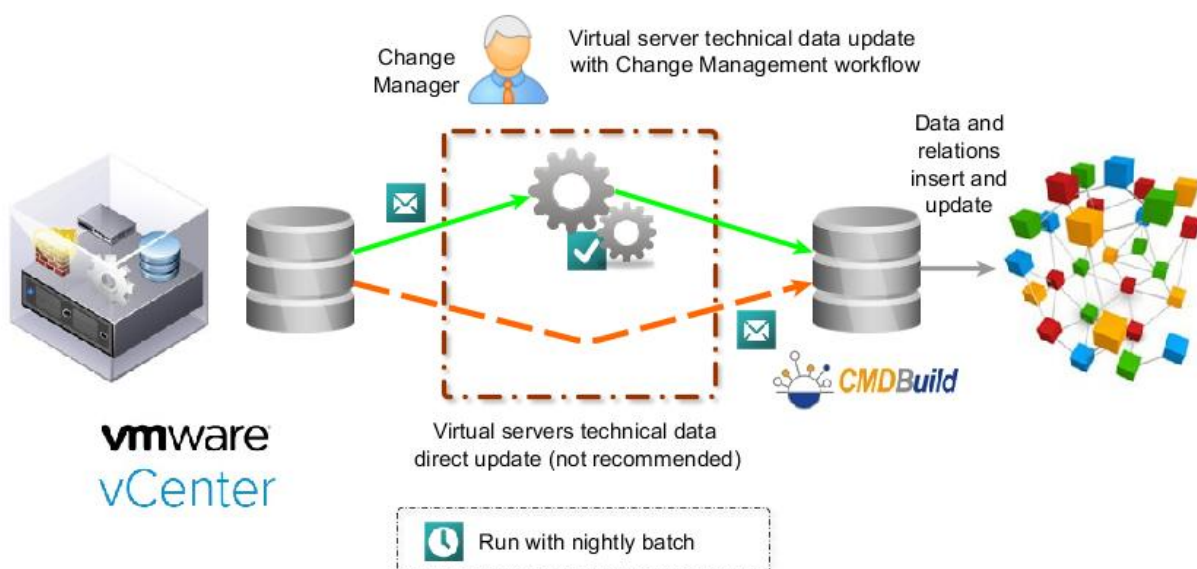
Connettore CMDBuild

Obiettivo del connettore è quello di sincronizzare le informazioni relative ai server fisici che erogano risorse all'ambiente di virtualizzazione, ai server virtuali gestiti nell'infrastruttura ed ai cluster a cui i server virtuali appartengono.

Al termine il connettore può inviare un report con la lista delle modifiche apportate nel CMDB.

Il connettore incluso nell'applicazione CMDBuild `READY2USE` accede a VMware VCenter tramite le API rese disponibili da VMware Infrastructure (VI) SDK 2.5.0 (`com.vmware.vim25` e `com.vmware.vim25.mo`).

Si riporta di seguito uno schema del connettore.



Attributi disponibili ed esempi di “mapping”

Segue un esempio di mapping tratto dal connettore VMware VCenter incluso nell'applicazione CMDBuild READY2USE:

INFORMAZIONE VMWARE	CLASSE CMDBUILD	ATTRIBUTO CMDBUILD	CHIAVE
Descrizione cluster	Cluster	Description	sì
Nome cluster	Cluster	Name	
Descrizione server virtuale	VirtualServer	Description	sì
Hostname server virtuale	VirtualServer	Hostname	
RAM server virtuale	VirtualServer	RAM	
Numero CPU server virtuale	VirtualServer	TotalCPUs	
Sistema operativo server virtuale	VirtualServer	OSVersion	
Note server virtuale	VirtualServer	Notes	
Stato server virtuale	VirtualServer	VSState	
Cluster esecuzione server virtuale	VirtualServer	ExecutionCluster	
Descrizione server fisico	PhysicalServer	Description	sì
Hostname server fisico	PhysicalServer	Hostname	
Marca server fisico	PhysicalServer	Brand	
Modello server fisico	PhysicalServer	Model	
Sistema operativo server fisico	PhysicalServer	OS	
Versione sistema operativo server fisico	PhysicalServer	OSVersion	
RAM server fisico	PhysicalServer	RAM	
Dimensione dischi server fisico	PhysicalServer	HDSize	
CPU server fisico	PhysicalServer	CPU	
Cluster servito dal server fisico	PhysicalServer	ServedCluster	
Descrizione interfaccia di rete	NetworkInterface	Description	sì
Codice interfaccia di rete	NetworkInterface	Code	
Nome interfaccia di rete	NetworkInterface	Name	
IPAddress interfaccia di rete	NetworkInterface	IPAddress	
Device cui è collegata l'interfaccia di rete	NetworkInterface	Hardware	

INFORMAZIONE LLDP / SNMP (OID)	CLASSE CMDBUILD	ATTRIBUTO CMDBUILD	CHIAVE
iso.0.8802.1.1.2.1.4.1.1.9	Hardware	Hostname	sì
iso.0.8802.1.1.2.1.3.3.0	NetworkDevice	Hostname	sì
iso.0.8802.1.1.2.1.3.4.0	NetworkDevice	Description	
iso.0.8802.1.1.2.1.3.7.1.3	NetworkDevice	PortNumber	
iso.0.8802.1.1.2.1.4.1.1.8	NetworkInterface	Name	sì
iso.0.8802.1.1.2.1.4.2.1.3	NetworkInterface	IPAddress	
iso.0.8802.1.1.2.1.4.1.1.7	NetworkInterface	MACAddress	
Reference a card NetworkDevice	NetworkInterface	Hardware	
Reference a card NetworkInterface	NetworkInterface	DstPort	

Zabbix

Generalità

Zabbix è uno strumento open source utilizzato per effettuare il monitoraggio di reti e sistemi informatici e per notificare eventuali malfunzionamenti.

Tali notifiche sono di particolare interesse per CMDBuild, che deve raccogliere e gestire eventuali interruzioni ai servizi erogati agli utenti.

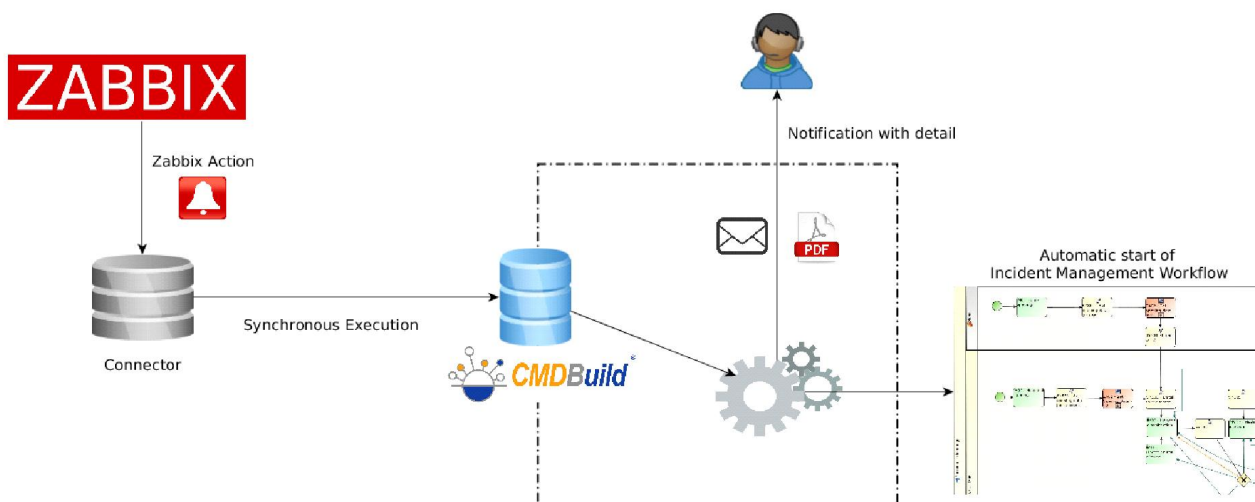
Connettore CMDBuild

Obiettivo del connettore incluso nell'applicazione CMDBuild Ready2Use è quello di:

- raccogliere le notifiche, tramite le API 3.0 di Zabbix
- archiviare gli eventi in una apposita nuova classe del CMDB
- inviare ai referenti tecnici un report riepilogativo dei CI coinvolti
- avviare un nuovo processo di Incident Management
- chiudere automaticamente il processo di Incident se il guasto si dovesse risolvere prima che il processo venga preso in carico

Se alcuni host o servizi dipendono da altri e non si vogliono aprire tanti Incident quanti sono gli asset che hanno il problema ma solo quello "padre", va configurata lato Zabbix la dipendenza tra Trigger.

Si riporta di seguito uno schema del connettore.



Nagios – GroundWork – NetEye

Generalità

Fra i tool di gestione adottati per il controllo dei sistemi informativi un posto di primo piano è occupato dalle soluzioni per il monitoraggio degli apparati e la gestione degli eventuali allarmi (ricordiamo in ambito open source le applicazioni Nagios, NetEye, GroundWork, etc).

Spesso però l'utilizzo di tali tool non prevede in input integrazioni con il CMDB aziendale ed è limitato in output al solo invio di mail di notifica.

Come spunto per chi deve configurare connettori di questa tipologia, riportiamo di seguito la descrizione di due utilizzi reali.

Ottimizzare le attività del Service Desk

Obiettivo di questa prima soluzione è quello di agevolare gli operatori nella valutazione di eventuali malfunzionamenti ed avviare contestualmente i processi di Incident Management che li risolvano.

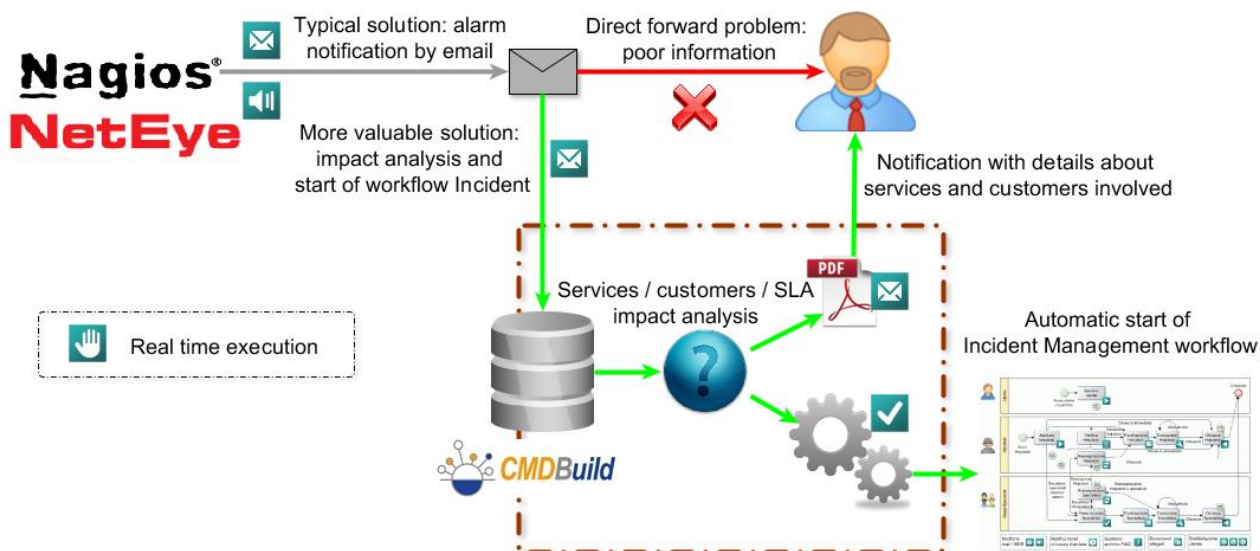
La soluzione più diffusa prevede infatti che il sistema di monitoraggio invii mail del tipo “Host xy down”. In molti casi però il tecnico che riceve la notifica non è in grado di capire la gravità del problema e, ad esempio se fuori azienda nelle giornate di reperibilità, non dispone in modo immediato di informazioni aggiuntive tramite cui approfondire la valutazione.

Il connettore suggerito per risolvere problematiche di questo tipo si comporta nel seguente modo:

- intercetta le mail inviate dal sistema di monitoraggio alle caselle di posta (IMAP) del supporto tecnico
- analizza il testo della mail, che essendo prodotto dal tool di monitoraggio è normalmente basato su un pattern predefinito e configurabile, ed estrae le informazioni utili (ad esempio la gravità del problema e l'hostname del server)
- utilizza il webservice di CMDBuild per accedere al CMDB e recuperare le informazioni di interesse che, in funzione della completezza del modello dati configurato, possono includere:
 - informazioni tecniche aggiuntive sul server coinvolto (memoria, sistema operativo, ecc)
 - lista dei servizi tecnici attivi su quel server
 - lista dei servizi utente coinvolti
 - lista dei clienti che li hanno sottoscritti
 - lista degli SLA previsti per ogni cliente su un dato servizio
 - criticità particolari
- produce un report contenente le informazioni sopra elencate e lo invia all'operatore tecnico, in allegato alla mail originale di segnalazione del problema
- avvia in CMDBuild un workflow di Incident Management precompilando le informazioni disponibili
- registra l'evento di down in una apposita entità di CMDBuild (normalmente configurata come classe “semplice”), correlandolo al server coinvolto
- segnala un “down” in corso sulla scheda del server coinvolto

La configurazione di un connettore di questo tipo, dotato di una estesa logica applicativa, richiede l'utilizzo dell'Advanced Connector.

Si riporta di seguito uno schema grafico della soluzione:



Per ulteriori approfondimenti:

L'implementazione è descritta in un case history presentato in occasione del “CMDBuild Day 2012”, tenutosi a Bologna presso la sede della Regione Emilia Romagna lo scorso 10 maggio 2012.

Si veda l'intervento dal titolo “Monitoraggio dei servizi e gestione degli allarmi con CMDBuild” (video e slide) a questo link (in lingua italiana):

<http://www.cmdbuild.org/it/diffusione/cmdbuild-day/cmdbuild-day-2014/2012>

Ottimizzare la configurazione del sistema di monitoraggio

Le attività di configurazione di un sistema di monitoraggio sono tutt'altro che banali, dovendo produrre la lista completa degli host da controllare, i servizi da monitorare su ciascuno di essi e le dipendenze rispetto agli altri host della rete (per evitare che un guasto ad un device produca centinaia di allarmi inviati da ogni altro host da esso dipendente).

Molto spesso tali attività sono svolte manualmente duplicando informazioni che dovrebbero essere già presenti nel CMDB, e soprattutto obbligando ad una doppia attività di aggiornamento nel caso di modifiche alla configurazione della rete (con conseguenti elevate probabilità di errore).

La soluzione proposta da un contribuente esterno (vedi riferimenti più sotto), attualmente in versione “beta”, è quella di arricchire il modello dati di CMDBuild con elementi aggiuntivi che consentano di ottenere l'esportazione automatica dei file di configurazione di:

- Nagios / NetEye
- NagVis, plugin che permette di visualizzare lo stato dei check all'interno di mappe e diagrammi
- Nagios Business Process, plugin che implementa due importanti viste:
 - Business View: aggrega il risultato dei check Nagios per visualizzare lo stato di servizi di alto livello

- Business Impact Analysis: permette di effettuare analisi di impatto simulando il down di uno o più componenti e visualizzandone gli effetti sui servizi di alto livello

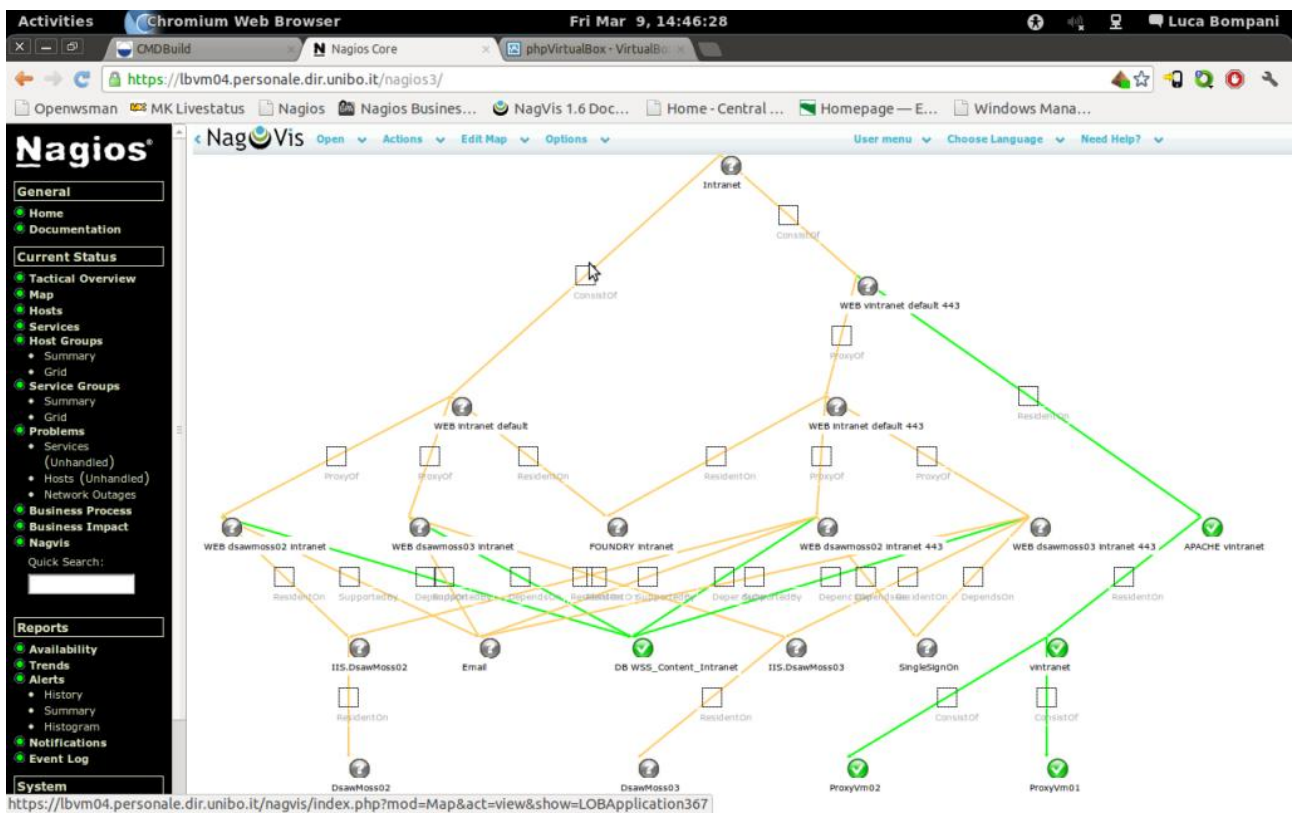
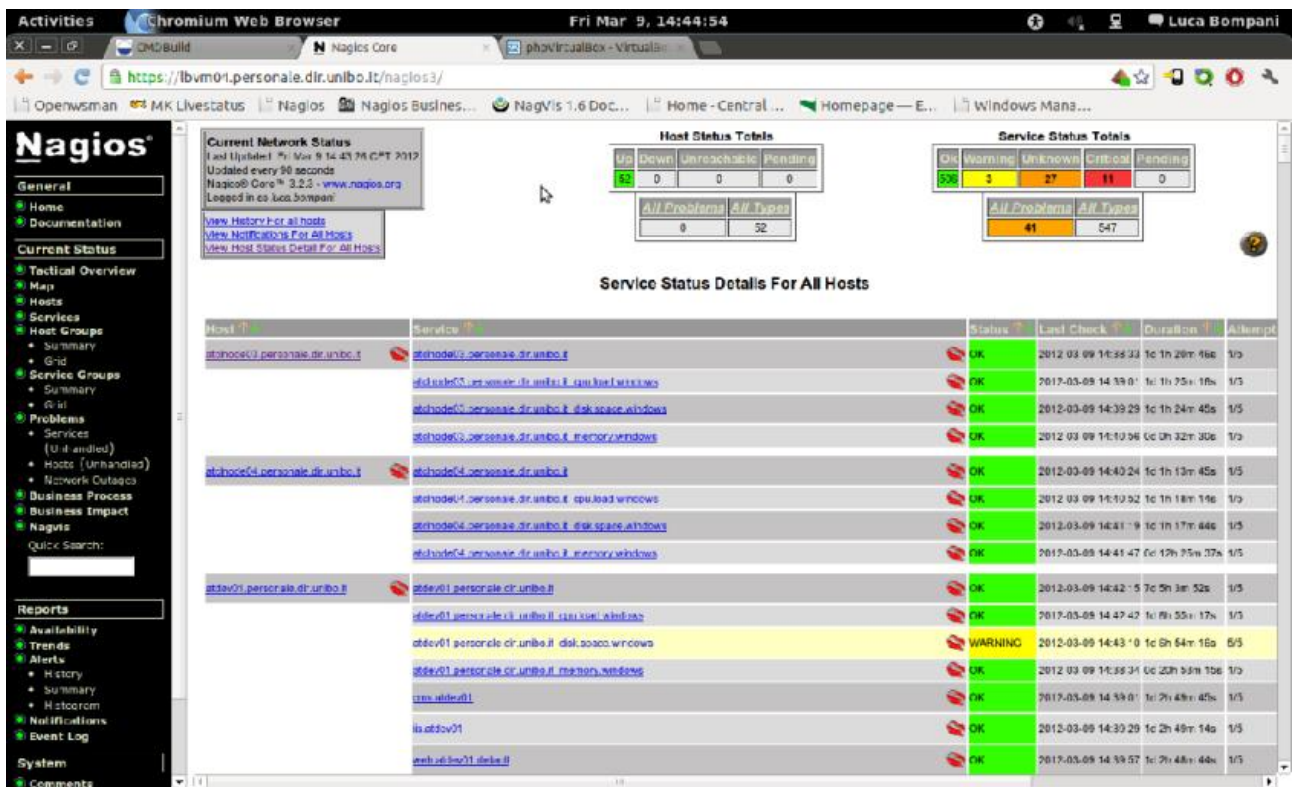
Il connettore è costituito da una applicazione Java, diversa dal Basic Connector e dall'Advanced Connector descritti in precedenza, che opera nel seguente modo:

- accede al CMDB tramite un nuovo webservice "CMDBf compliant" (<http://www.dmtf.org/standards/cmdbf>), implementato dallo stesso contributore
- interpreta gli asset e le relazioni contenute in CMDBuild in modo dinamico, facendosi guidare da alcuni metadati appositamente predisposti sotto forma di attributi aggiuntivi sulle classi e sui domini coinvolti (vedi sotto)
- crea un rappresentazione dei dati in memoria sotto forma di grafo, utilizzando appositi tool opensource
- utilizza algoritmi standard, per l'ordinamento e la visita dei nodi di un grafo e per generare i file di configurazione del monitoraggio

I metadati inseriti nel CMDB sotto forma di attributi aggiuntivi sono associati:

- alle classi
 - "host = true" significa che per quell'istanza deve essere generato un host Nagios
 - "serviceGroup = true" significa per quell'istanza deve essere generato un serviceGroup Nagios
- ai domini
 - "checkAssociation = true" significa che quella relazione deve essere utilizzata per associare i service Nagios agli host
 - "checkDependency = true" significa che quella relazione deve essere utilizzata per generare le dipendenze tra i service Nagios
 - "checkPropagation = true" significa che quella relazione deve essere utilizzata per propagare i check associati ad un asset ad un'altro asset
 - "internalDependency = true" significa che quella relazione deve essere utilizzata nella generazione delle mappe NagVis
 - "clusterAttribute = Cluster" significa che quella relazione deve essere utilizzata per raggruppare le relazioni in cluster: il check su un cluster fallisce solo se falliscono tutti i componenti del cluster

Si riportano di seguito tre screenshot relativi rispettivamente all'integrazione con Nagios / NetEye, NagVis e NagiosBP.



The screenshot shows the Nagios Core web interface in a Chromium browser. The main content area displays a "Short Summary: All Business Processes" with three sections for different priority levels. Each section contains a table of business processes and their status.

Priority 1
Alerting round the clock (24 x 7)

Business Process	Status	Status Information
Intranet	UNKNOWN	
Portale Internet	UNKNOWN	

Priority 2
Alerting Monday to Sunday 7:00 to 22:00

Business Process	Status	Status Information
Academic Projects	UNKNOWN	
Corsi di studio	UNKNOWN	
Dipartimenti	UNKNOWN	
Facolta	UNKNOWN	
SpaziVirtual	UNKNOWN	
UniboMagazine	UNKNOWN	

Priority 3
Alerting Monday to Thursday 7:00 to 17:00, Friday 7:00 to 15:00

Business Process	Status	Status Information
Altri siti web	UNKNOWN	

At the bottom of the summary, there are links: [\[All Priorities\]](#) [\[Priority 1\]](#) [\[Priority 2\]](#) [\[Priority 3\]](#) [\[Show Traffic Lights\]](#)

Per ulteriori approfondimenti:

L'implementazione è descritta in un case history presentato in occasione del "CMDBuild Day 2012", tenutosi a Bologna presso la sede della Regione Emilia Romagna lo scorso 10 maggio 2012.

Si veda l'intervento dal titolo "Un prototipo di integrazione fra CMDBuild e NetEye / Nagios" (video e slide) a questo link (in lingua italiana):

<http://www.cmdbuild.org/it/diffusione/cmdbuild-day/cmdbuild-day-2014/2012>

Archi

Generalità

TOGAF ¹ è un framework aperto di supporto alla progettazione e gestione di architetture informatiche enterprise, Archimate ² è un linguaggio di modellazione complementare a TOGAF per la descrizione delle architetture IT, Archi ³ è da tempo lo strumento open source di riferimento per il disegno di modelli Archimate.

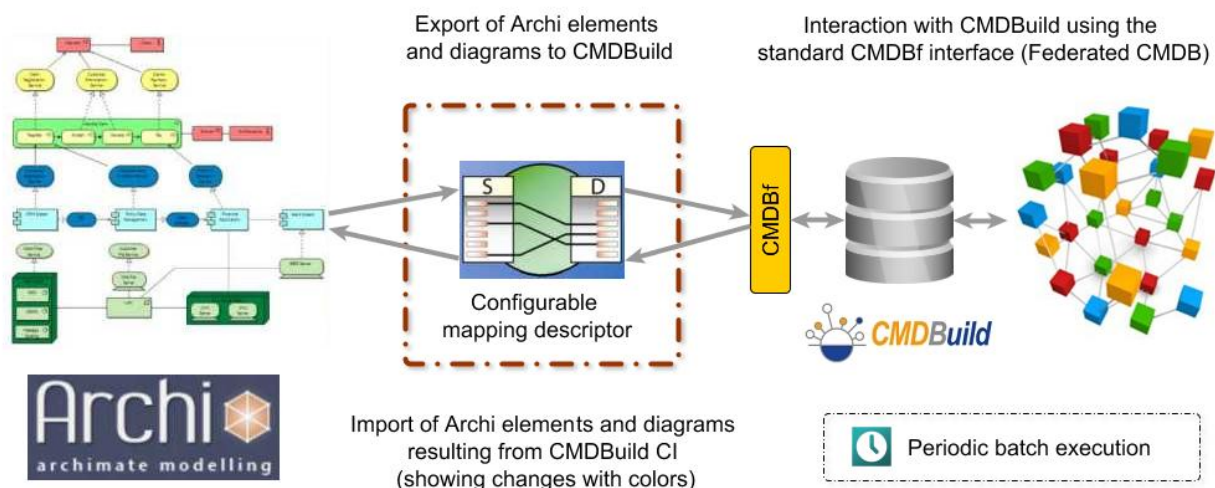
Archi consente di avere una visione grafica d'insieme dei CI, di personalizzare le viste con raggruppamenti e colori e di navigare graficamente le relazioni, ma non dispone di un database in cui archiviare i propri dati.

Per contro CMDBuild è un repository completo e strutturato, ma non fornisce una visione grafica di insieme dei componenti infrastrutturali, tramite cui poter analizzare gli impatti e navigare graficamente le relazioni.

Partendo da queste esigenze due enti diversi ⁴ hanno deciso di implementare un plugin per la sincronizzazione dei dati di CMDBuild in Archi, e viceversa. Essendo venuti a conoscenza dei due progetti abbiamo messo in contatto i due enti e questo ha portato ad unire gli sforzi ottenendo un prodotto unico realizzato in collaborazione.

Il plugin è costituito da una applicazione Java rilasciata dai due contributori con licenza open source, liberamente scaricabile a questo indirizzo:

<https://bitbucket.org/bompani/cmdbuild-unibo/downloads>



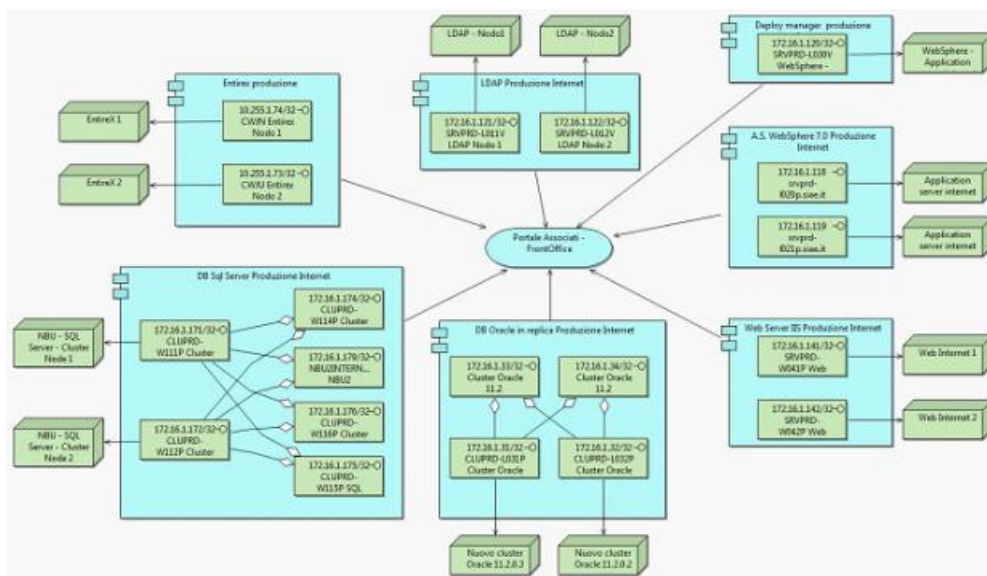
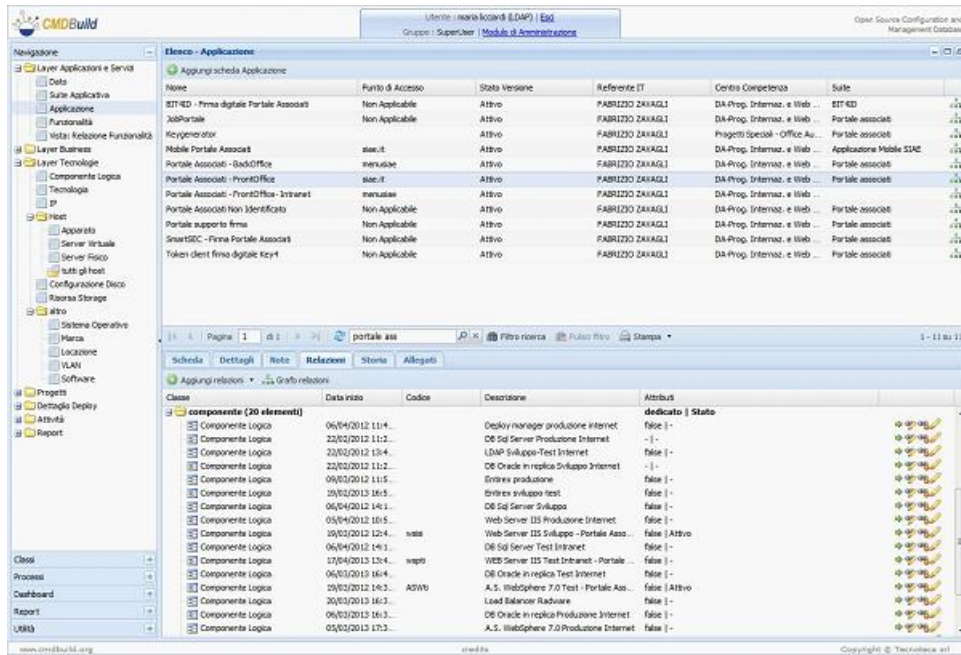
1 <http://www.opengroup.org/togaf/>

2 <http://www.opengroup.org/subjectareas/enterprise/archimate>

3 <http://archi.cetis.ac.uk/>

4 Università degli Studi di Bologna e SIAE Roma

Riportiamo di seguito due screenshot significativi di come le stesse informazioni infrastrutturali sono presentate e rese disponibili nelle due interfacce CMDBuild ed Archi.



Per ulteriori approfondimenti:

L'implementazione è descritta in un case history presentato in occasione dell'evento "TOGAF, Archi e CMDBuild", tenutosi a Firenze presso la sede del Consiglio Regionale Toscana lo scorso 28 maggio 2013.

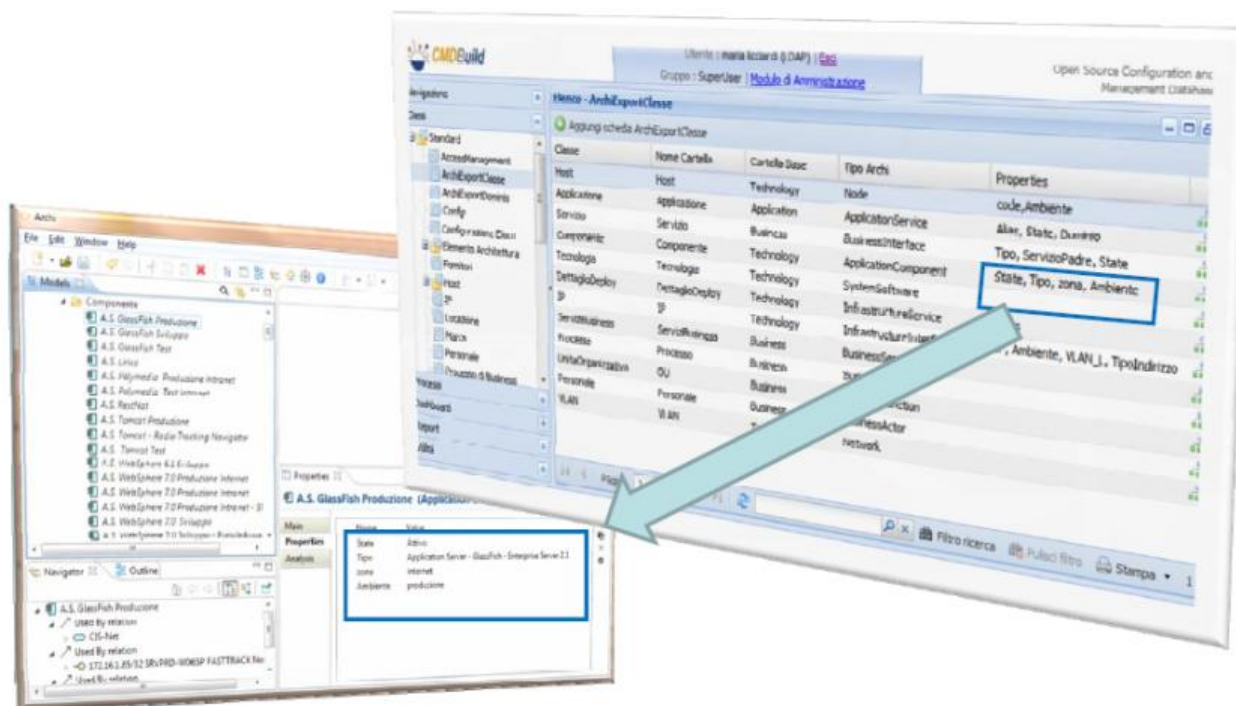
Si veda l'ultimo intervento (video e slide) a questo link (in lingua italiana):

<http://www.cmdbuild.org/it/diffusione/convegni-e-workshop/togaf-archi-e-cmdbuild>

Modalità di funzionamento

Anche in questo caso la funzionalità principale del connettore è quella di mantenere sincronizzate le informazioni gestite nei due ambienti, avendo definito regole di "mapping" fra:

- entità Archi e classi CMDBuild
- relazioni Archi e domini CMDBuild



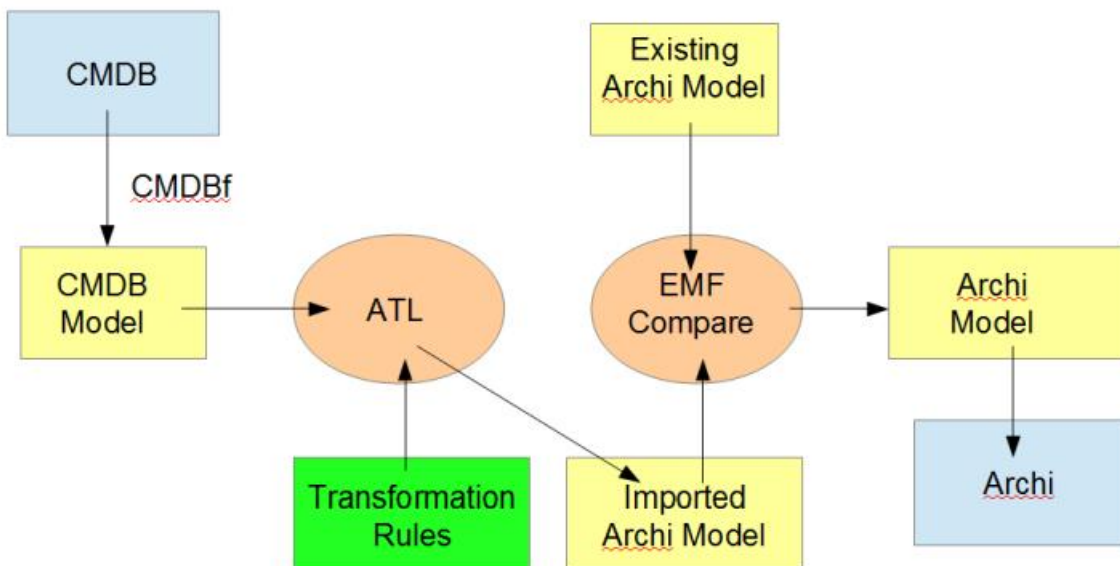
Caratteristiche principali del connettore sono:

- sincronizza dati e diagrammi (in particolare importa in CMDBuild i diagrammi Archi, sotto forma di documento allegato alla scheda dati)
- accede ai dati di CMDBuild attraverso un webservice compatibile con la standard CMDBf (<http://www.dmtf.org/standards/cmdbf>), realizzato anch'esso da uno dei due contributori del connettore
- si appoggia al linguaggio standard ATL per la definizione dei criteri di mapping Archi <=> CMDBuild; ATL (Atlas Transformation Language - <http://www.eclipse.org/at/>) è a sua volta basato su EMF (Eclipse Modeling framework - <http://www.eclipse.org/modeling/emf/>)

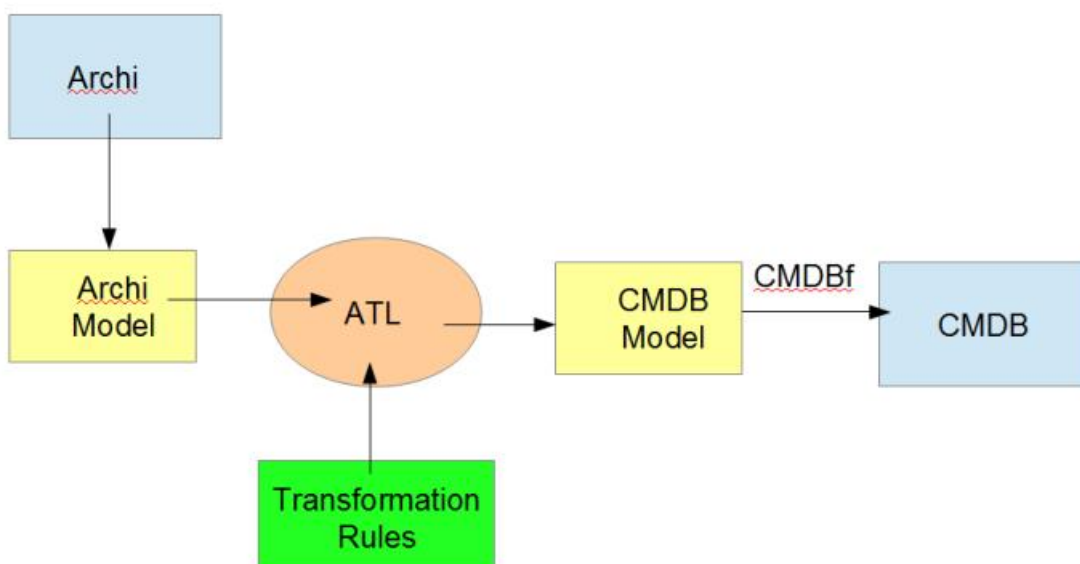
- esegue il confronto di modelli EMF tramite la libreria di Eclipse EMF Compare
- gestisce la sincronizzazione bidirezionale
- evidenzia gli eventuali asset che il connettore eliminerebbe, trovandoli nel sistema di destinazione ma non in quello di partenza
- visualizza le differenze riscontrate fra i due sistemi

Seguono gli schemi di funzionamento delle due tipologie di sincronizzazione.

Sincronizzazione CMDBuild => Archi



Sincronizzazione CMDBuild Archi => CMDBuild



Configurazione

Si riportano di seguito alcune indicazioni per la configurazione del connettore.

Per maggiori informazioni si faccia riferimento a quanto indicato nel file di download del connettore.

Installazione del plugin

L'installazione del plugin richiede le seguenti operazioni:

- scaricare il file `archimate-cmdbuild-plugin.zip`, disponibile nel repository Bitbucket (<https://bitbucket.org/bompani/cmdbuild-unibo/downloads>)
- espandere l'archivio ZIP del plugin
- espandere l'archivio ZIP `plugins.zip` nella cartella `plugins` di Archi
- copiare il file `archimate-cmdb-plugin.jar` nella cartella `plugins` di Archi
- importare i file `.xsd` per creare le classi ed i domini in CMDBuild, disponibili in <http://localhost/cmdbuild/services/cmdb-schema/>
- riavviare dopo aver eseguito l'upload

File di configurazione del plugin

La directory di configurazione di Archi `{HOME}/uk.ac.bolton/Archi` contiene i file di configurazione del plugin:

- `cmdbf.xml`
- `archi2cmdbuild.atl`
- `cmdbuild2archi.atl`

File `cmdbf.xml`

Si riporta un esempio del file di configurazione `cmdbf.xml`:

```
<configuration>
  <cmdb id="http://www.cmdbuild.org">
    <queryWSDL>
      http://localhost:8080/cmdbuild/services/soap/CMDBfQuery?wsdl
    </queryWSDL>
    <registrationWSDL>
      http://localhost:8080/cmdbuild/services/soap/CMDBfRegistration?wsdl
    </registrationWSDL>
    <query>
      ... {CMDBf Query}
    </query>
    <archi2cmdb>
      file:/home/luca/uk.ac.bolton/Archi/archi2cmdbuild.atl
    </archi2cmdb>
    <cmdb2archi>
      file:/home/luca/uk.ac.bolton/Archi/cmdbuild2archi.atl
    </cmdb2archi>
  </cmdb>
</configuration>
```

```
</cmdb>  
</configuration>
```

Configurazione del plugin

Per configurare il plugin:

- installare in Eclipse il plugin ATL SDK
- editare i file `archi2cmdbuild.atl` e `cmdbuild2archi.atl` per definire le trasformazioni fra il metamodello ArchiMate ed il metamodello del CMDB

APPENDICE: Glossario

ALLEGATO

Per “allegato” si intende un qualunque file associabile ad una scheda dati inserita nel sistema.

Per la gestione degli allegati CMDBuild utilizza in modalità embedded un qualunque sistema documentale compatibile con il protocollo standard CMIS (oppure il DMS Alfresco fino alla versione 3 tramite il proprio webservice nativo).

La gestione degli allegati supporta il versioning di file caricati più volte, con numerazione automatica.

ATTIVITA'

Per “attività” si intende uno dei passaggi che costituiscono il processo.

Una attività è caratterizzata da un nome, un esecutore, un tipo, eventuali attributi, eventuali metodi associati ad API di CMDBuild per poter essere eseguiti.

Per “istanza di attività” si intende una specifica attivazione di una attività, effettuata automaticamente dal sistema o manualmente da un operatore.

Vedi anche: Processo

ATTRIBUTO

Il termine indica nel sistema CMDBuild la generica tipologia di informazione descrittiva di una determinata classe.

CMDBuild consente tramite il Modulo Schema di creare nuovi attributi in una classe o in un dominio e di modificarne alcune caratteristiche.

Nella classe “Fornitore” gli attributi sono ad esempio il nome, l'indirizzo, il numero di telefono, ecc.

Ogni attributo corrisponde nel Modulo di Gestione a campi di inserimento dati sulla apposita scheda di gestione della classe e a colonne della corrispondente tabella nel database.

Vedi anche: Classe, Dominio, Relazione, Superclasse, Tipo di attributo

BIM

Metodologia che si pone l'obiettivo di supportare l'intero ciclo di vita di un edificio, dall'idea iniziale alla fase di costruzione, di utilizzo e manutenzione, fino alla eventuale demolizione finale.

La metodologia BIM (Building Information Modeling) è supportata da numerosi programmi informatici che possono interagire tramite un formato aperto di scambio dati denominato IFC (Industry Foundation Classes).

Vedi anche: GIS

CI

Si definisce Configuration Item (Elemento della Configurazione) ogni elemento che concorre a fornire il servizio IT all'Utente, considerato ad un livello di dettaglio sufficiente per la sua gestione tecnica e patrimoniale.

Esempi di CI sono: server, workstation, programma applicativo, sistema operativo, stampante, ecc

Vedi anche: Configurazione

CLASSE

Il termine rappresenta un tipo di dati complesso caratterizzato da un insieme di attributi che nel loro insieme descrivono quel tipo di dato.

Una classe modella una tipologia di oggetto da gestire nel CMDB, quale ad esempio un computer, una applicazione software, un servizio, un fornitore, ecc

CMDBuild consente all'Amministratore del Sistema, attraverso il Modulo Schema, di definire nuove classi e di cancellare o modificare la struttura di classi già definite.

Una classe è rappresentata a video da una apposita scheda di gestione dati e nel database da una tavola generata automaticamente al momento della definizione della classe.

Vedi anche: Scheda, Attributo

CONFIGURAZIONE

Il processo di Gestione della Configurazione ha lo scopo di mantenere aggiornata e disponibile per gli altri processi la base di informazioni relativa agli oggetti informatici gestiti (CI), alle loro relazioni ed alla loro storia.

E' uno dei principali processi gestiti dal sistema ITIL.

Vedi anche: CI, ITIL

DASHBOARD

Una dashboard corrisponde in CMDBuild ad una raccolta di grafici di diversa tipologia, tramite cui avere immediata evidenza di alcuni parametri chiave (KPI) relativi ad un particolare aspetto di gestione del servizio IT.

Vedi anche: Report

DATABASE

Il termine indica un insieme di informazioni strutturato ed organizzato in archivi residenti sull'elaboratore server, nonché l'insieme dei programmi di utilità dedicati alla gestione dei tali informazioni per attività quali inizializzazione, allocazione degli spazi, ottimizzazione, backup, ecc.

CMDBuild si appoggia sul database PostgreSQL, il più potente, affidabile e completo database Open Source, di cui utilizza in particolare le sofisticate funzionalità e caratteristiche object oriented.

DOMINIO

Un dominio rappresenta una tipologia di relazione fra una coppia di classi.

E' caratterizzato da un nome, dalle descrizioni della funzione diretta ed inversa, dai codici delle due classi e dalla cardinalità (numerosità degli elementi relazionabili) ammessa, nonché dagli eventuali attributi configurati.

CMDBuild consente all'Amministratore del Sistema, attraverso il Modulo Schema, di definire nuovi domini e di cancellare o modificare la struttura di domini già definiti.

E' possibile caratterizzare ciascun dominio tramite definizione di attributi custom.

Vedi anche: Classe, Relazione

FILTRO DATI

Un filtro dati è una restrizione della lista degli elementi contenuti in una classe, ottenuta specificando condizioni booleane (uguale, diverso, contiene, inizia, ecc) sui possibili valori assumibili da ciascun attributo della classe.

I filtri dati possono essere definiti ed utilizzati “una tantum”, oppure possono essere memorizzati dall'operatore e richiamati successivamente (dallo stesso operatore o da operatori di altri gruppi di utenti ai quali l'Amministratore del sistema abbia concesso l'utilizzo).

Vedi anche: Classe, Vista

GIS

Un sistema GIS è un sistema informatico in grado di produrre, gestire e analizzare dati spaziali associando a ciascun elemento geografico una o più descrizioni alfanumeriche.

Le funzionalità GIS implementate in CMDBuild consentono di creare attributi geometrici, in aggiunta a quelli testuali, tramite cui rappresentare su scala locale (planimetrie) o su scala più estesa (mappe esterne) elementi puntuali (ad esempio gli asset IT), poligonali (ad esempio linee dati) o aree (piani, stanze, ecc).

Vedi anche: BIM

GUI FRAMEWORK

E' una interfaccia utente completamente personalizzabile e orientata a fornire un accesso semplificato all'applicazione, pubblicabile su portali web di qualsiasi tecnologia ed interoperabile con CMDBuild tramite il webservice REST standard.

Vedi anche: Mobile, Webservice

ITIL

Sistema di "best practice" ormai affermatosi come "standard de facto", non proprietario, per la gestione dei servizi informatici secondo criteri orientati ai processi (Information Technology Infrastructure Library).

Fra i processi fondamentali coperti da ITIL ci sono quelli del Service Support, comprendenti l'Incident Management, il Problem Management, il Change Management, il Configuration Management ed il Release Management.

Per ogni processo considera la descrizione, i componenti di base, i criteri e gli strumenti consigliati per la misura della qualità del servizio, i ruoli e le responsabilità delle risorse coinvolte, i punti di integrazione con gli altri processi (per eliminare duplicazioni e inefficienze).

Vedi anche: Configurazione

LOOKUP

Con il termine “LookUp” si indica una coppia di valori del tipo (Codice, Descrizione) impostabili dall'Amministratore del Sistema tramite il Modulo Schema.

Tali valori vengono utilizzati dall'applicazione per vincolare la scelta dell'utente, al momento della compilazione del relativo campo sulla scheda dati, ad uno dei valori preimpostati.

Il Modulo Schema consente la definizione di nuove tabelle di “LookUp” secondo le necessità dell'organizzazione.

MOBILE

E' una interfaccia utente ottimizzata per strumenti "mobile" (smartphone e tablet), implementata come "app" multiplatforma (iOS, Android) ed interoperabile con CMDBuild tramite il webservice REST standard.

Vedi anche: GUI Framework, Webservice

PROCESSO

Per "processo" (o workflow) si intende una sequenza di passaggi ("attività") descritti nel sistema per svolgere in forma guidata e secondo regole prestabilite una determinata azione.

Per ogni processo saranno avviate in CMDBuild una serie di "istanze di processo", una per ogni necessità di effettiva esecuzione dell'azione corrispondente, che avrà luogo su "asset" specifici e sarà svolta da utenti specifici.

Una "istanza di processo" viene attivata tramite avvio e conferma del primo passaggio previsto e termina alla esecuzione dell'attività finale prevista nella definizione.

Vedi anche: Attività

RELAZIONE

Per "Relazione" si intende in CMDBuild un collegamento effettivo di due schede appartenenti a due classi, o in altri termini una istanza di un dato dominio.

Una relazione è quindi definita da una coppia di identificativi univoci delle due schede collegate e dall'identificativo del dominio utilizzato per il collegamento, nonché dalla valorizzazione degli eventuali attributi previsti nel dominio.

CMDBuild consente agli operatori del Sistema, attraverso il Modulo Gestione Dati, di definire nuove relazioni fra le schede archiviate nel database.

Vedi anche: Classe, Dominio

REPORT

Il termine indica in CMDBuild una stampa (in formato PDF o CSV) riportante in forma analitica le informazioni estratte da una o più classi fra le quali sia definita una catena di domini.

I report possono essere generati e modificati dagli operatori di CMDBuild tramite una apposita funzione del Modulo di Gestione Dati e la relativa definizione viene memorizzata nel database per poter essere riutilizzata successivamente.

Vedi anche: Classe, Dominio, Database

SCHEDA

Con il termine "Scheda" in CMDBuild si riferisce un elemento archiviato in una determinata classe.

Una scheda è caratterizzata da un insieme di valori assunti da ciascuno degli attributi definiti per la sua classe di appartenenza.

CMDBuild consente agli operatori del Sistema, attraverso il Modulo Gestione Dati, di archiviare nuove schede nel database e di aggiornare schede già archiviate.

Le informazioni di ogni scheda saranno memorizzate nel database alle opportune colonne di una riga della tavola generata per la classe di appartenenza della scheda.

Vedi anche: Classe, Attributo

SUPERCLASSE

Una superclasse è una classe astratta utilizzabile per definire una sola volta attributi condivisi fra più classi. Da tale classe astratta è poi possibile derivare classi reali che conterranno i dati effettivi e che comprenderanno sia gli attributi condivisi (specificati nella superclasse) che quelli specifici della sottoclasse.

Ad esempio è possibile definire la superclasse “Computer” con alcuni attributi base (RAM, HD, ecc) e le sottoclassi derivate “Desktop”, “Notebook”, “Server”, ciascuna delle quali con i soli attributi specifici.

Vedi anche: Classe, Attributo

TIPO DI ATTRIBUTO

Ogni attributo definito per una determinata classe è caratterizzato da un “Tipo” che determina le caratteristiche delle informazioni contenute e la loro modalità di gestione.

Il tipo di attributo viene definito con il Modulo Schema e può essere poi modificato entro alcuni limiti dipendenti dalla tipologia dei dati già archiviati.

CMDBuild gestisce i seguenti tipi di attributo: “Boolean” (booleano, Si / No), “Date” (data), “Decimal” (decimale), “Double” (virgola mobile in doppia precisione), “Inet” (indirizzo IP), “Integer” (numero intero), “LookUp” (tabellato da lista configurabile in “Impostazioni” / “LookUp”), “Reference” (riferimento o foreign key), “String” (stringa), “Text” (testo lungo), “TimeStamp” (data e ora).

Vedi anche: Attributo

VISTA

Una vista è un insieme di schede definito in modo “logico” anziché dal fatto di costituire l'intero contenuto di una classe nel CMDB.

In particolare una vista può essere definita in CMDBuild applicando un filtro ad una classe (quindi conterrà un insieme ridotto delle stesse righe) oppure specificando una funzione SQL che estragga attributi da una o più classi correlate.

La prima tipologia di vista mantiene tutte le funzionalità disponibili per una classe, la seconda consente la sola visualizzazione e ricerca con filtro veloce.

Vedi anche: Classe, Filtro

WEBSERVICE

Un webservice è un'interfaccia che descrive una collezione di operazioni, accessibili attraverso una rete mediante messaggistica XML.

Tramite un webservice una applicazione può rendere accessibili le proprie funzionalità ad altre applicazioni operanti attraverso il web.

CMDBuild dispone di un webservice SOAP e di un webservice REST.

WIDGET

Un widget è un componente grafico di una interfaccia utente di una applicazione software, che ha lo scopo di facilitare all'utente l'interazione con l'applicazione stessa.

CMDBuild prevede l'utilizzo di widget sotto forma di “pulsanti” posizionabili su schede dati o su schede di avanzamento di processi. I pulsanti aprono finestre di tipo “popup” tramite cui inserire se richiesto informazioni aggiuntive e visualizzare poi l'output della funzione richiamata.