



CMDBuild

***Open Source Configuration and Management Database
Workflow Tutorial***

**Versione 0.6.1
Settembre 2007**

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, JasperReports, IReport
We are thankful for the great contributions that led to the creation of that products

CMDBuild è un progetto realizzato e gestito da:

Comune di Udine – Servizio Sistemi Informativi e Telematici



Tecnoteca S.r.l. (www.tecnoteca.com)



Cogitek S.r.l. (www.cogitek.it)



CMDBuild è rilasciato con licenza GPL (www.gnu.org/copyleft/gpl.html)
Copyright ©2006 Tecnoteca srl

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

Sommario

Introduzione.....	4
I moduli di CMDBuild.....	4
Descrizione del sistema di workflow.....	5
Obiettivi.....	5
Strumenti utilizzati.....	5
Terminologia.....	6
Attuali limitazioni del sistema.....	7
Scopo del manuale.....	8
Modalità di implementazione in CMDBuild.....	9
Definizione di un nuovo processo.....	9
Utilizzo di un nuovo processo.....	11
Editor visuale JPED.....	14
Generalità.....	14
Utilizzo integrato con CMDBuild.....	14
Tutorial.....	16
Esempio definizione processo per la creazione guidata nuovo utente.....	16
Fase a) – Creazione classe derivata da “Activity”.....	16
Fase b) – Esportazione “scheletro” XPDL da CMDBuild.....	17
Fase c) – Caricamento “scheletro” XPDL in JPED.....	18
Fase d) – Completamento definizione variabili workflow in JPED.....	20
Fase e1) – Attività “Inserimento richiesta”.....	20
Fase e2) – Attività “Notifica richiesta aperta”.....	22
Fase e3) – Attività “Valutazione richiesta”.....	27
Fase e4) – Attività “Associazione cespiti”.....	32
Fase e5) – Attività “Notifica richiesta valutata”.....	35
Fase e6) – Attività “Inizio lavorazione”.....	36
Fase e7) – Attività “Notifica inizio lavorazione”.....	38
Fase e8) – Attività “Fine lavorazione”.....	40
Fase e9) – Attività “Notifica richiesta chiusa”.....	42
Fase e10) – Attività “Notifica richiesta rifiutata”.....	43
Fase f) – Salvataggio definizione processo.....	44
Fase g) – Salvataggio schema grafico processo.....	45
Fase h) – Caricamento file XPDL e schema grafico in CMDBuild.....	45
Fase i) – Creazione istanza nuovo processo.....	45
Appendice.....	46
Lista metodi API.....	46

Introduzione

CMDBuild è una applicazione Open Source per la configurazione e gestione del database della configurazione (CMDB) degli oggetti in uso presso il Dipartimento IT di una organizzazione.

Gestire un Database della Configurazione 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.

CMDBuild si ispira alle "best practice" ITIL (Information Technology Infrastructure Library), ormai affermatesi come "standard de facto", non proprietario, per la gestione dei servizi informatici secondo criteri orientati ai processi.

Con CMDBuild l'amministratore del sistema può costruire autonomamente il proprio CMDB (da cui il nome del progetto), grazie ad un apposito programma di configurazione che consente di aggiungere progressivamente nel sistema nuove classi di oggetti, nuovi attributi e nuove tipologie di relazioni.

Tramite il sistema implementato per la gestione del workflow è poi possibile definire in modo visuale con un editor esterno nuovi processi operanti sulle classi trattate nel sistema, importarli in CMDBuild ed eseguirli secondo i criteri specificati.

Sono infine disponibili sistemi di interfaccia con applicazioni open source esterne specializzate in attività connesse a CMDBuild (Automatic Inventory, Help Desk e prossimamente Document Management).

I moduli di CMDBuild

Il sistema CMDBuild comprende due moduli principali:

- il Modulo Schema, dedicato alla definizione iniziale ed alle successive modifiche della struttura dati (classi e sottoclassi, attributi delle classi, tipologie di relazioni fra classi)
- il Modulo Gestione Dati, dedicato all'inserimento ed aggiornamento nel sistema dei dati descrittivi e delle relazioni funzionali fra le diverse entità, alla produzione di report e tabulati, nonché alla definizione e controllo dei processi per la gestione dei servizi informatici

Il presente manuale è dedicato alla descrizione di come è stata integrata in CMDBuild la gestione del workflow ed alla presentazione di un tutorial che illustra nel dettaglio i passaggi da svolgere per la creazione di un nuovo progetto (tutorial).

La definizione di nuovi workflow è riservata ad utenti evoluti quali i tecnici informatici cui sono demandate le attività di amministrazione del sistema.

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

- Overview concettuale del sistema
- Administrator Manual
- User Manual
- Technical Manual
- External Connectors

Descrizione del sistema di workflow

Obiettivi

Per poter supportare le indicazioni metodologiche di ITIL il sistema CMDBuild deve non solamente supportare l'aggiornamento dell'inventario degli asset e delle relazioni funzionali fra di essi, ma anche consentire la definizione e controllo dei processi per la gestione dei servizi informatici.

Un processo consiste di una o più attività, svolte da operatori e/o da applicazioni informatiche, ciascuna delle quali rappresenta un'azione da svolgere all'interno del processo, nel caso specifico relativamente alla gestione in qualità degli asset informatici.

Fra i processi fondamentali coperti da ITIL ci sono quelli del Service Management, comprendenti:

- Service Support: Incident Management, Problem Management, Change Management, Configuration Management, Release Management
- Service Level Management, Availability Management, Capacity Management, Continuity Management, Financial Management

Dati il numero elevato dei processi attivabili, le peculiarità organizzative dei singoli enti e le caratteristiche di estensibilità, flessibilità ed autonomia di gestione perseguite dal progetto CMDBuild, si è scelto di implementare non una serie di processi rigidi e predefiniti, ma un sistema generico tramite il quale utenti evoluti possano disegnare ed attivare autonomamente i workflow di proprio interesse.

Nel seguito del documento è sviluppato in modo semplificato sotto forma di tutorial un processo semplificato per la gestione delle attività di HelpDesk.

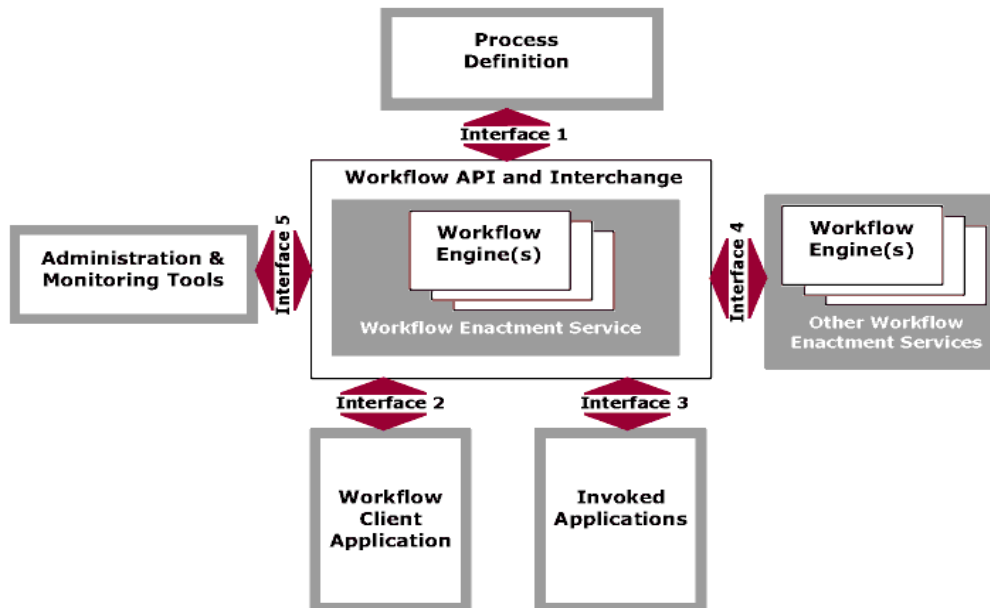
Osservandone la sintassi ed i meccanismi utilizzati l'amministratore del sistema CMDBuild potrà trarre indicazioni utili per definire nuovi processi secondo le proprie necessità, in ambito ITIL o anche per utilizzi diversi quali ad esempio la gestione degli acquisti o dei contratti di assistenza.

Strumenti utilizzati

Il sistema scelto in CMDBuild per la gestione del workflow utilizza i seguenti strumenti:

- XPDL come linguaggio di definizione (standardizzato dalla WfMC, WorkFlow Management Coalition sulla base del modello sotto riportato)
- il motore open source Enhydra Shark (<http://www.enhydra.org/workflow/shark/>), un framework estensibile che fornisce una implementazione completa e standard delle specifiche WfMC (<http://www.wfmc.org/>) e OMG, utilizzando al suo interno XPDL come linguaggio nativo
- l'editor visuale JPED (<http://jped.sourceforge.net/>) per il disegno del workflow e per la definizione dei meccanismi di integrazione con CMDBuild

Segue lo schema di riferimento per la gestione dei workflow secondo il modello standardizzato dal WfMC.



Terminologia

La terminologia utilizzata nel seguito del presente documento comprende i seguenti termini principali:

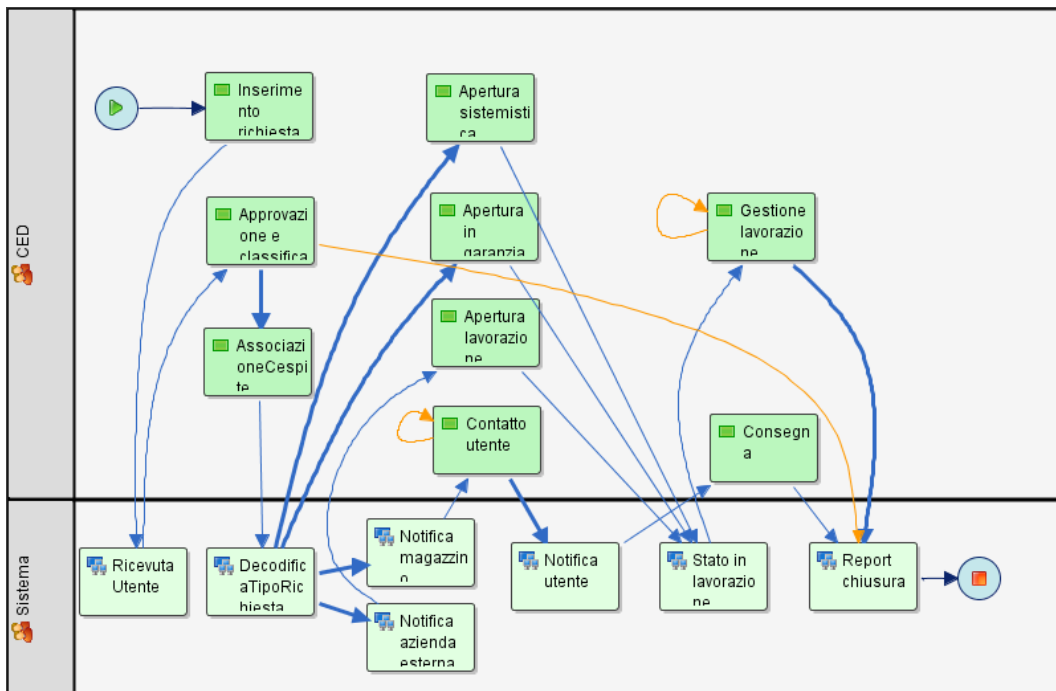
- con “processo” si intende una sequenza di passaggi (“attività”) descritti nel sistema per svolgere una determinata azione in forma guidata e secondo regole prestabilite
- con “attività” si intende uno dei passaggi che costituiscono il processo
- con “istanza di processo” si intende una specifica attivazione di un “processo”, effettuata tramite avvio e conferma del primo passaggio previsto
- con “istanza di attività” si intende una specifica attivazione di una attività, effettuata automaticamente dal sistema o manualmente da un operatore (tramite compilazione di suoi attributi e di eventuali ulteriori operazioni richieste e conferma finale)

I termini sopra indicati sono “tradotti” nel modello dati di CMDBuild secondo i seguenti criteri:

- ogni “processo” corrisponde ad una classe derivata della superclasse “Activity”, comprendente l’ “unione” degli attributi caratterizzanti le diverse attività
- ogni “istanza di processo” corrisponde ad una scheda della classe derivata da “Activity” (attività corrente) unita alla lista delle relative versioni storicizzate (attività concluse)
- ogni “istanza di attività” corrisponde ad una scheda della classe derivata da “Activity” (attività corrente) oppure ad una delle relative versioni storicizzate (attività concluse)

Ogni processo è caratterizzato da un nome, da uno o più partecipanti, da alcune variabili (fra cui lo stato corrente) e da una sequenza di attività e transizioni che lo realizzano.

Un processo è rappresentabile tramite un grafo, come ad esempio nell'immagine successiva:



Ogni attività è caratterizzata da:

- un nome
- un esecutore, che corrisponde obbligatoriamente ad un “ruolo” ed opzionalmente ad un operatore
- un tipo: iniziale, finale, attività eseguita da un operatore, attività eseguita automaticamente dal sistema
- eventuali attributi, provenienti da CMDBuild o interni al workflow, che saranno valorizzati nel corso della sua esecuzione
- eventuali metodi, associati ad API esterne (di CMDBuild, ma non solo) per poter essere eseguiti

Attuali limitazioni del sistema

Sono al momento note e saranno risolte nei rilasci successivi le seguenti limitazioni:

- nell'applicazione CMDBuild viene tracciato l'avanzamento delle attività eseguite da un operatore umano, ma non di quelle svolte automaticamente dal motore di workflow (“tool activity”)
- sempre per le attività svolte automaticamente dal motore di workflow non sono tracciati eventuali errori o eccezioni occorsi nell'esecuzione
- come per le altre classi di CMDBuild, non è possibile allegare un documento ad una attività (bisognerà attendere la release 0.70)

Scopo del manuale

Il presente manuale si propone di presentare i criteri di implementazione del workflow in CMDBuild e più in particolare di guidare l'utilizzatore del sistema nella creazione autonoma di nuovi processi personalizzati.

A tal fine si è scelto di far seguire ai primi capitoli introduttivi un completo e dettagliato tutorial in cui si descrive passo passo il meccanismo di gestione di un processo semplificato di HelpDesk (denominato appunto HelpDeskSemplice).

Tale processo comprende i seguenti passaggi:

- inserimento di una nuova richiesta (apertura nuovo ticket)
- aggiornamento stato processo ad “aperto” e notifica automatica via mail al richiedente
- valutazione della richiesta
- in caso di mancata approvazione aggiornamento stato processo a “chiuso” e notifica automatica via mail al richiedente
- in caso di approvazione associazione del ticket ad uno o più cespiti interessati
- aggiornamento stato processo a “valutato” e notifica automatica via mail al richiedente
- registrazione di inizio lavorazione
- aggiornamento stato processo a “in lavorazione” e notifica automatica via mail al richiedente
- registrazione di fine lavorazione
- aggiornamento stato processo a “chiuso” e notifica automatica via mail al richiedente

Il file XPDL risultante è stato inserito nel rilascio per agevolare la comprensione del presente tutorial.

Modalità di implementazione in CMDBuild

Definizione di un nuovo processo

Si è cercato di implementare la gestione del workflow tramite una struttura dati quanto più omogenea con i meccanismi di base già presenti in CMDBuild.

In particolare sono stati utilizzati i seguenti accorgimenti:

- è stata creata una nuova superclasse “riservata” (proprietà non modificabili dall’amministratore di CMDBuild) denominata “Activity” e contenente alcuni attributi comuni agli specifici workflow definibili (vedi schema sottostante)
- è richiesta la creazione di una sottoclasse della classe “Activity” per ogni specifico workflow da attivare
- è stato utilizzato il meccanismo della “storia” per tracciare gli stati di avanzamento di un processo
- è stato mantenuto il meccanismo delle “relazioni” per creare collegamenti automatici o manuali in forma “guidata” fra le classi degli oggetti trattati in CMDBuild o fra queste e l’istanza di processo che le ha aggiornate

Segue la definizione della superclasse “Activity” (relativamente agli attributi non specifici):

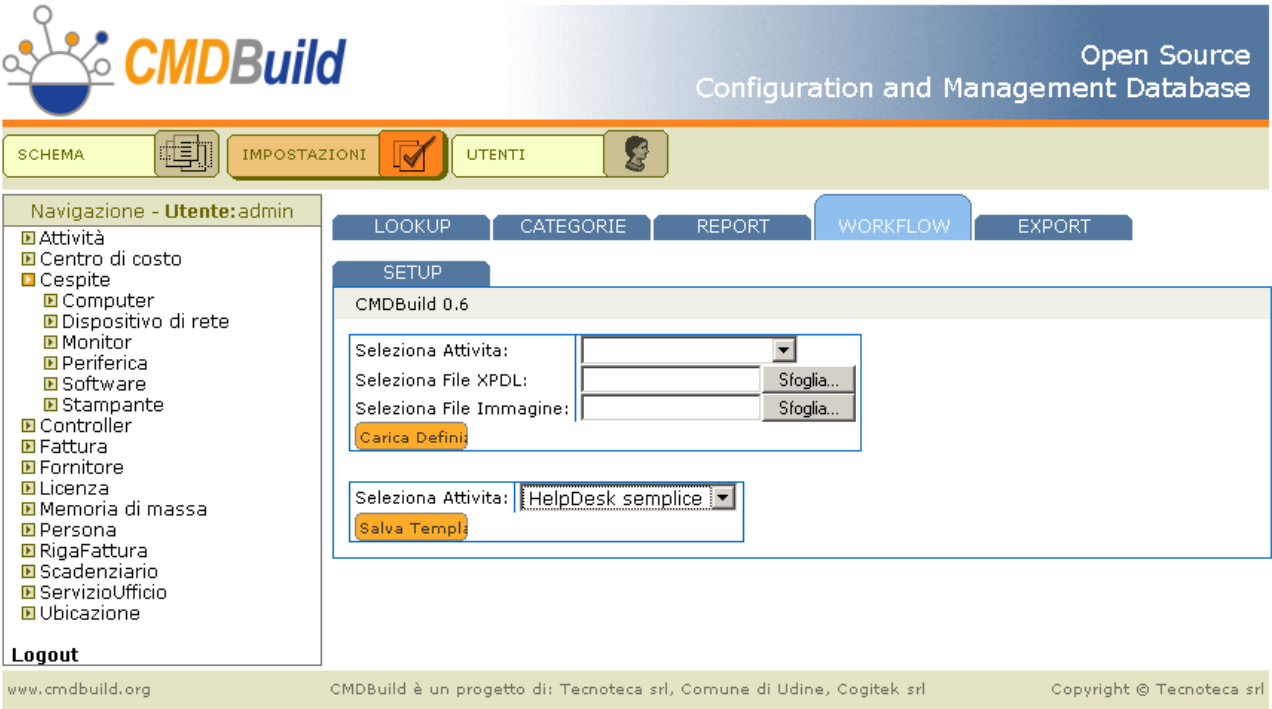
Attributo	Tipo	Modo	Note
Code	String	write	Nome attività XPDL
Description	String	write	Testo richiesta di avvio del processo
AssignedTo	Reference (su tabella “User”)	write	Operatore cui viene assegnata l’attività, ove applicabile (da implementare)
Priority	Integer	write	Priorità
FlowStatus	Integer	reserved	Stato istanza workflow, per uso interno
ActivityDefinitionId	String	reserved	Id istanza attività Enhydra Shark, per uso interno
ActivityName	String	reserved	Nome istanza attività Enhydra Shark, per uso interno
ProcessCode	String	reserved	ID processo, per uso interno
ActivityDescription	String	reserved	Per uso interno

Il meccanismo previsto per la creazione di un nuovo schema di processo comprende la seguente sequenza di passaggi:

- analisi del nuovo processo da implementare, al fine di individuare:
 - la lista degli attributi descrittivi del processo nell’insieme delle sue attività costitutive
 - la lista delle chiamate dei metodi API richiesti dal processo (nell’ambito dei metodi disponibili descritti in Appendice)
 - la lista dei diversi ruoli (gruppi di utenti) coinvolti nel sistema
- creazione della nuova sottoclasse di Activity, con il nome del nuovo processo, che dovrà

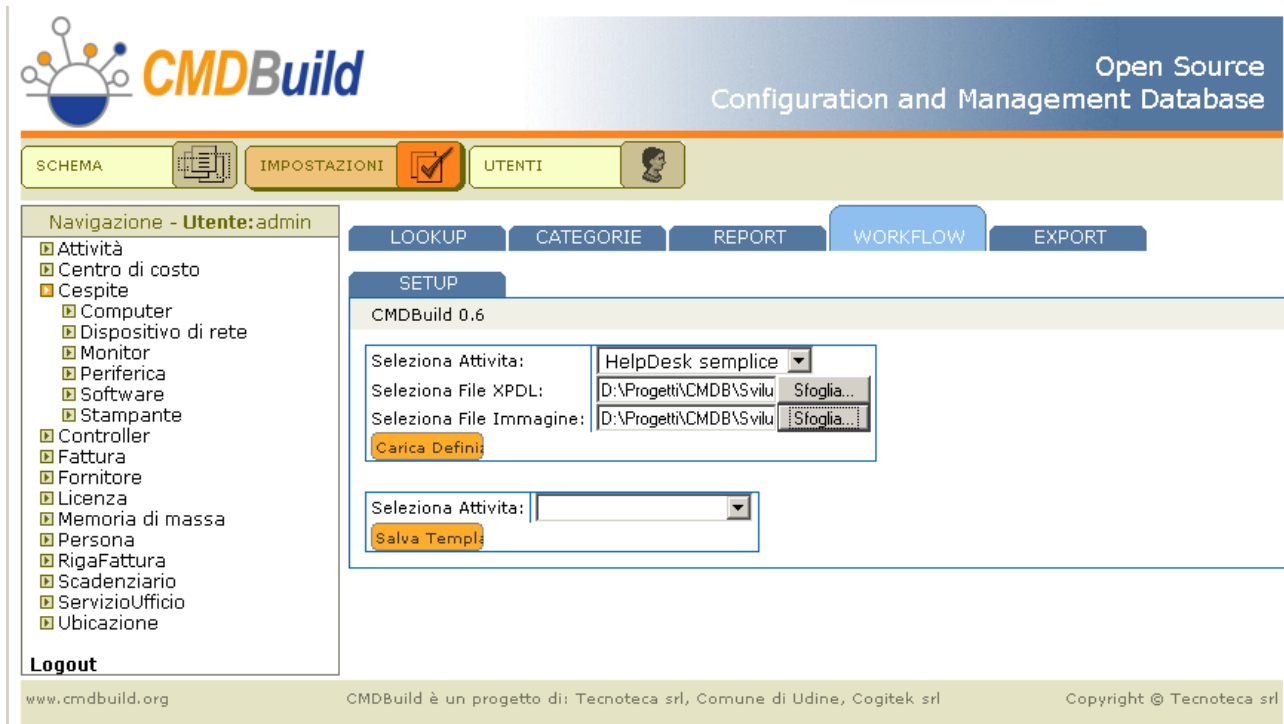
essere definita tramite il Modulo Schema, derivandola dalla superclasse “Activity” ed aggiungendo gli attributi specifici individuati al passo precedente

- creazione dei gruppi di utenti mancanti, che dovrà essere definita tramite il Modulo Schema
- esportazione con il Modulo Schema (pulsante “Salva Template”, vedi form successiva) dello “scheletro” del nuovo schema di processo, costituito appunto da:
 - nome del processo
 - lista degli attributi del processo
 - lista delle API utilizzabili nel processo
 - lista degli “attori” (gruppi di utenti) partecipanti al processo (cui viene aggiunto il ruolo “fittizio” denominato “Sistema” per l’assegnamento delle attività automatiche)



The screenshot shows the CMDBuild web interface. At the top, there is a navigation bar with tabs for SCHEMA, IMPOSTAZIONI, UTENTI, and a user profile icon. Below this is a secondary navigation bar with buttons for LOOKUP, CATEGORIE, REPORT, WORKFLOW (selected), and EXPORT. The main content area is titled 'SETUP' and contains a form for 'CMDBuild 0.6'. The form has two sections: the first for selecting an activity and file (with 'Carica Defini' button), and the second for selecting an activity (with 'Salva Template' button). A left sidebar shows a tree view of categories like 'Attività', 'Centro di costo', 'Cespite', etc. The footer contains the website URL, project information, and copyright notice.

- disegno del workflow, tramite utilizzo dell’editor esterno JPED (o altro editor XPDL compatibile, ad esempio JaWE) con cui verrà completato lo “scheletro” esportato da CMDBuild
- salvataggio dall’editor esterno JPED
 - dello schema del processo disegnato (in formato XPDL)
 - della rappresentazione grafica del processo (in formato JPG)
- importazione in CMDBuild dello schema del processo e della sua rappresentazione grafica, tramite l’apposita funzione del Modulo Schema (pulsante “Carica Definizione”, vedi form successiva)



Una volta terminate le operazioni sopra descritte il nuovo processo è disponibile per essere utilizzato nel Modulo Gestione di CMDBuild (TAB “Attività”), che interpreterà ed eseguirà lo schema del processo tramite il supporto del motore Enhydra Shark.

Le operazioni descritte possono essere eseguite anche più volte a fronte della necessità di modifica di un processo già importato, con l'unica avvertenza che le modifiche saranno recepite solo dalle nuove istanze del processo che verranno avviate.

Utilizzo di un nuovo processo

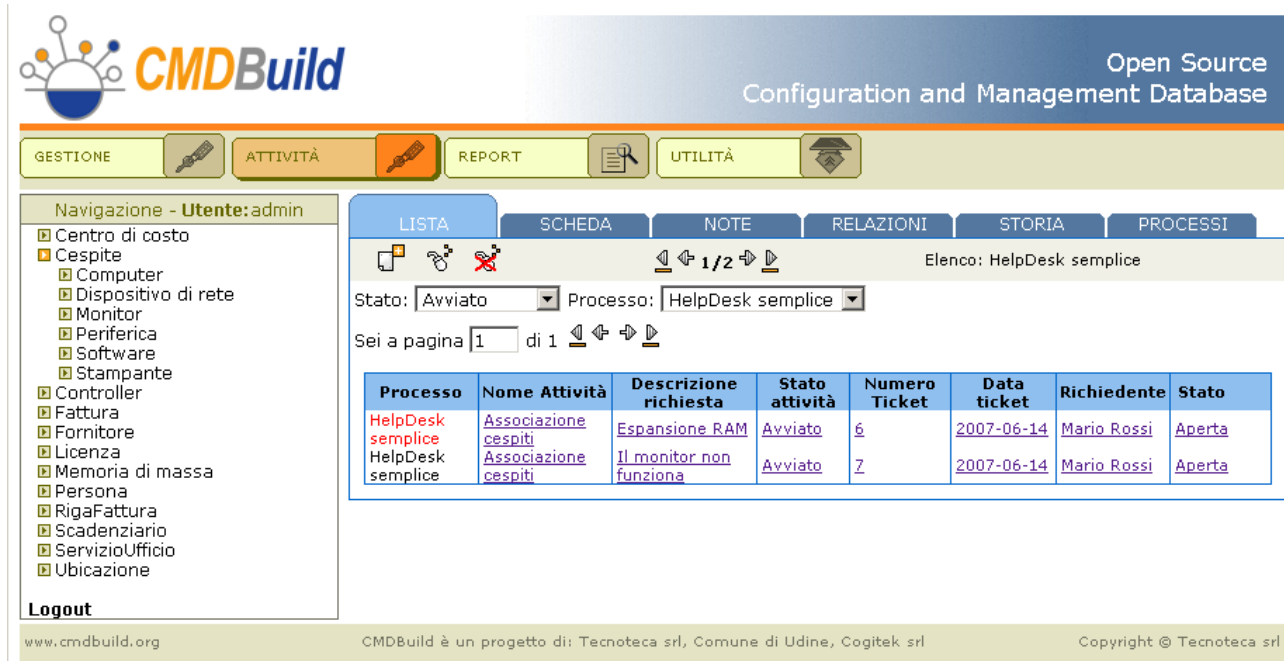
L'applicazione CMDBuild è stata anche estesa nel Modulo Gestione con la possibilità di interpretare, tramite il supporto del motore Enhydra Shark, i processi disegnati esternamente con JPED e poi importati tramite il Modulo Schema.

Sempre con l'obiettivo di mantenere la massima coerenza con le preesistenti funzionalità di CMDBuild, l'interfaccia utente del Modulo Gestione è stata estesa secondo i seguenti criteri:

- è stata creata una nuova sezione del Modulo Gestione, denominata “Attività” (nuovo TAB di primo livello)
- la nuova sezione mantiene le gestioni standard già presenti nel TAB di primo livello “Gestione”, basate sui TAB “Lista”, “Scheda”, “Note”, “Relazioni”, “Storia”, cui è stato aggiunto il TAB informativo “Processi”
- nel TAB “Lista” sono visualizzate le istanze delle attività in cui l'utente è coinvolto (perché partecipa a quell'attività o ad altre attività di quel processo) con:
 - filtri per stato (avviato, completato, interrotto, tutti), tipo processo (lista dei processi definiti nel Modulo Schema), tipo visualizzazione (solo attività corrente o tutte, da implementare)
 - filtro standard sugli attributi dell'attività e delle classi in relazione

- area dati con visualizzazione tabellare delle informazioni (il nome del processo, il nome dell'attività, la descrizione della richiesta, lo stato del processo e gli ulteriori attributi definiti come "display base" nel Modulo Schema), rese "cliccabili" per l'accesso guidato alla scheda di gestione dell'attività, nel caso questa richieda l'intervento dell'utente collegato
- pulsanti per creare una nuova attività o per approvare / rifiutare in forma veloce una singola attività di "approvazione" o tutte le attività evidenziate
- nel TAB "Scheda" è possibile visualizzare o compilare gli attributi previsti per quell'istanza di attività (l'accesso in scrittura o sola lettura è specificabile nell'ambito dell'editor JPED) oppure effettuare eventuali ulteriori operazioni richieste quali la creazione di nuove schede o di relazioni (specificate tramite l'editor JPED)
- nel TAB "Note" è possibile visualizzare o inserire annotazioni sull'istanza di attività
- nel TAB "Relazioni" è possibile visualizzare o inserire relazioni fra l'istanza dell'attività e istanze di altre classi ("schede")
- nel TAB "Storia" è possibile visualizzare le versioni precedenti di quell'istanza di attività (istanze già eseguite)
- nel TAB "Processi" sarà visualizzata l'eventuale rappresentazione grafica del processo esportata dall'editor visuale JPED ed importata in CMDBuild

La lista delle attività da eseguire viene ad esempio presentata tramite la seguente form:



The screenshot shows the CMDBuild web interface. At the top, there is a navigation bar with tabs: GESTIONE, ATTIVITÀ, REPORT, and UTILITÀ. Below this, there is a sidebar with a tree view of categories like 'Centro di costo', 'Cespiti', 'Computer', etc. The main content area is titled 'Elenco: HelpDesk semplice' and shows a list of activities. The list has columns for 'Processo', 'Nome Attività', 'Descrizione richiesta', 'Stato attività', 'Numero Ticket', 'Data ticket', 'Richiedente', and 'Stato'. Two activities are listed, both with the state 'Avviato'.

Processo	Nome Attività	Descrizione richiesta	Stato attività	Numero Ticket	Data ticket	Richiedente	Stato
HelpDesk semplice	Associazione cespiti	Espansione RAM	Avviato	6	2007-06-14	Mario Rossi	Aperta
HelpDesk semplice	Associazione cespiti	Il monitor non funziona	Avviato	7	2007-06-14	Mario Rossi	Aperta

Lo svolgimento di un'attività viene effettuato tramite una form simile a quella sotto riportata:


 Open Source
 Configuration and Management Database

GESTIONE
ATTIVITÀ
REPORT
UTILITÀ

Navigazione - Utente: admin

- ▣ Centro di costo
- ▣ Cespite
 - ▣ Computer
 - ▣ Dispositivo di rete
 - ▣ Monitor
 - ▣ Periferica
 - ▣ Software
 - ▣ Stampante
- ▣ Controller
- ▣ Fattura
- ▣ Fornitore
- ▣ Licenza
- ▣ Memoria di massa
- ▣ Persona
- ▣ RigaFattura
- ▣ Scadenziario
- ▣ ServizioUfficio
- ▣ Ubicazione

Logout

LISTA
SCHEDA
NOTE
RELAZIONI
STORIA
PROCESSI

Classe HelpDesk semplice ▾

conferma
annulla

Nome Attività InserimentoRichiesta

Descrizione richiesta Non funziona la posta elettronica

Richiedente Mario Rossi ▾

conferma
annulla

www.cmdbuild.org
CMDBuild è un progetto di: Tecnoteca srl, Comune di Udine, Cogitek srl
Copyright © Tecnoteca srl

In alcuni casi può essere opportuno che alcuni passaggi del processo (ad esempio proprio una nuova richiesta di HelpDesk) siano inseriti da un utente esterno all'organizzazione del CED (ad esempio dall'utilizzatore stesso del cespite), non sufficientemente esperto per l'utilizzo sia pure limitato di CMDBuild. Tale eventuale necessità può essere risolta tramite sviluppo di una pagina web personalizzata che potrà essere raggiungibile da un eventuale portale intranet preesistente ed alimentare poi l'archivio delle attività di CMDBuild.

Editor visuale JPEd

Generalità

JPEd è un editor visuale per il disegno e formalizzazione di processi in linguaggio XPDL, basato sul codice sorgente di JaWE Community Edition, cui sono state aggiunte alcune utili caratteristiche.

L'editor JPEd consente il disegno visuale di un workflow, producendone il corrispondente file XPDL poi interpretato da CMDBuild tramite il motore Enhydra Shark.

Nel seguito si descriveranno sommariamente i meccanismi di integrazione utilizzati, rimandando al capitolo successivo per la consultazione del "tutorial" sulla creazione completa di un processo.

Come già descritto alcune informazioni trattate da JPEd sono rese automaticamente disponibili tramite lo "scheletro" XPDL esportato da CMDBuild, fra cui in particolare:

- il nome del processo
- gli attributi non riservati definiti nel processo (superclasse "Activity" e sottoclasse specifica)
- i ruoli definiti nel sistema (attori del processo), cui è aggiunto automaticamente per convenzione il ruolo denominato "Sistema" (al quale per convenzione dovranno essere assegnate le attività automatiche del processo)
- i metodi (API) resi disponibili da CMDBuild o da Enhydra Shark per l'automazione parziale o totale di un'attività (creazione nuova scheda, modifica scheda, creazione relazione, invio email, ecc)

Utilizzo integrato con CMDBuild

All'interno dell'editor JPEd sarà quindi possibile, ovviamente rimanendo nell'ambito della sintassi prevista per il formato XPDL:

- aprire lo "scheletro" XPDL esportato da CMDBuild per agevolare la definizione di un nuovo processo
- definire variabili di processo che svolgano ad esempio la funzione di memoria dell'esito di un'attività da utilizzarsi poi come input per un'attività successiva
- definire attività di diversa tipologia:
 - inizio processo (già creato per default nel file XPDL esportato da CMDBuild)
 - fine processo (già creato per default nel file XPDL esportato da CMDBuild)
 - attività senza implementazione automatica
 - semplici, per le quali è richiesta la valorizzazione di attributi del processo o solamente la conferma di avvenuta esecuzione off-line (ad esempio la sostituzione della scheda di rete in un computer)
 - complesse, rese da CMDBuild tramite il meccanismo degli "extended attributes" XPDL, per le quali è richiesta l'esecuzione di azioni ulteriori tramite l'interfaccia di CMDBuild (creazione nuova scheda, modifica scheda, creazione relazione, ecc)
 - attività automatiche, gestite tramite il meccanismo standard XPDL dei "tools" (esecuzione di metodi esterni automatici che non prevedono interfaccia utente, ad esempio l'invio di una mail)

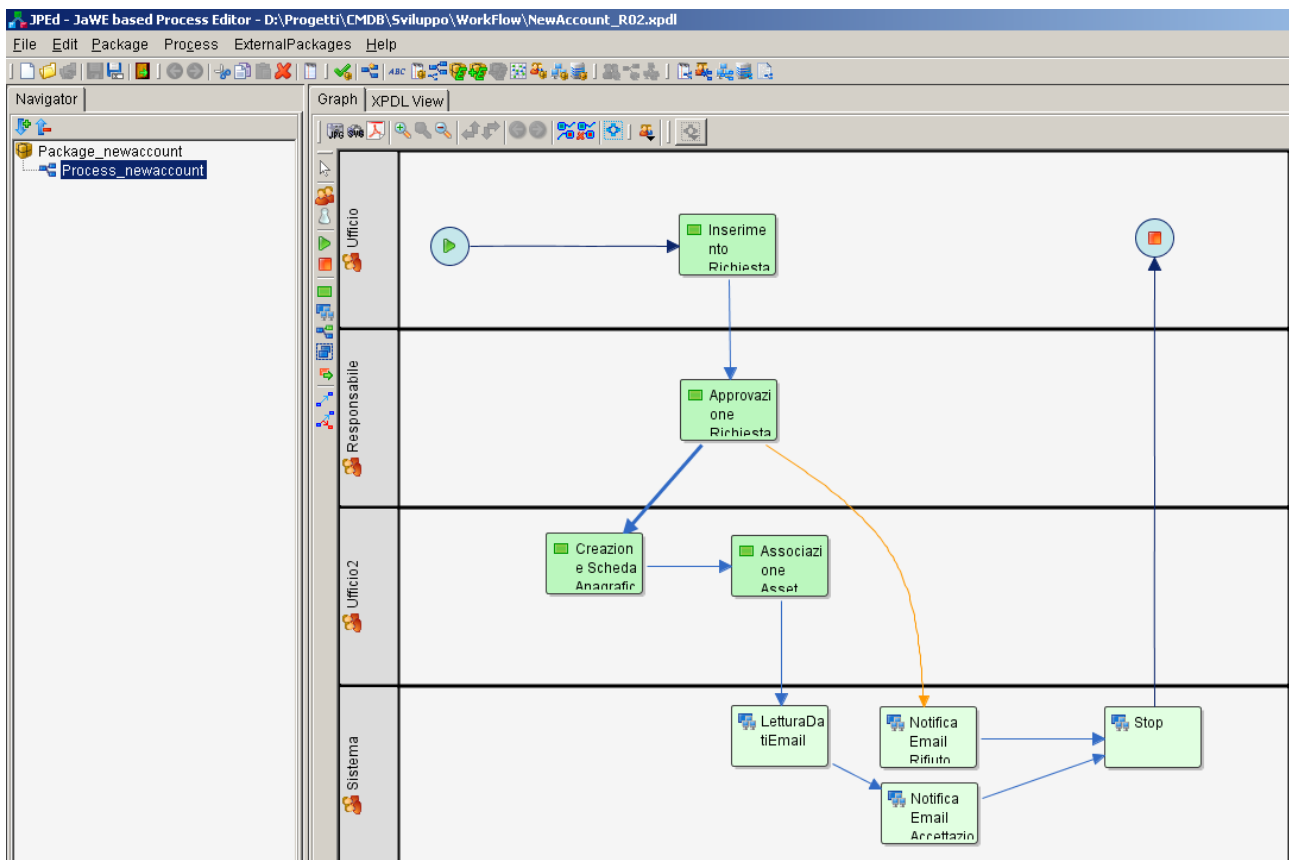
- definire per ogni attività e per ogni attributo del processo importato da CMDBuild, se quell'attributo dovrà trattato come modificabile o di sola lettura o neppure da visualizzare nella form con cui CMDBuild gestirà l'esecuzione dell'attività stessa (TAB Activity form editor)
- descrivere, per le attività che prevedano scelte fra più alternative, le condizioni di percorrenza delle transizioni previste
- definire per ogni attività complessa la lista dei metodi (API) da richiamare in CMDBuild nell'ambito della form di gestione con cui l'attività sarà eseguita, memorizzando eventuali valori di ritorno nelle variabili di processo definite (tramite "extended attributes")
- definire per ogni attività automatica la lista dei metodi (API) da richiamare per l'esecuzione automatica dell'attività, memorizzando eventuali valori di ritorno nelle variabili di processo definite (tramite "tools")
- salvare il file XPDL corrispondente al processo disegnato
- salvare la rappresentazione grafica del processo, prodotta da JPEd con una apposita funzione del menu

Una volta concluso il disegno del processo in JPEd rimane solamente da:

- caricare in CMDBuild tramite il Modulo Schema il file XPDL ottenuto
- utilizzare il TAB di primo livello "Attività" nel Modulo Gestione per attivare una nuova istanza del processo e gestirla tramite l'interfaccia utente prevista

JPEd ammette il raggruppamento di più processi in uno stesso "package", ma tale possibilità non è supportata dal meccanismo di integrazione con CMDBuild.

Segue un esempio dell'interfaccia utente dell'editor JPEd:



Tutorial

Esempio definizione processo per gestione semplificata servizio HelpDesk

Nell'esempio seguente vengono descritti i singoli passaggi da eseguire per la definizione di un processo che controllerà la gestione semplificata di un servizio di HelpDesk.

Il processo prevede le seguenti attività:

- inserimento richiesta (da parte dell'operatore del servizio di HelpDesk che riceve la telefonata dell'utente)
- notifica apertura ticket (invio mail all'utente con gli estremi del ticket) e aggiornamento stato ticket (attività automatica)
- valutazione richiesta (da parte di un operatore dell'Ufficio IT)
- nel caso la richiesta sia rifiutata (problema non riscontrato, utente non abilitato alla funzionalità richiesta, ecc), notifica via mail al richiedente, aggiornamento stato ticket e chiusura dell'istanza del processo (attività automatica)
- nel caso la richiesta sia approvata, associazione della richiesta ai cespiti coinvolti
- inoltro mail al richiedente con notifica di approvazione della richiesta e aggiornamento stato ticket (attività automatica)
- inizio lavorazione (attività di sola registrazione dell'operazione)
- inoltro mail al richiedente con notifica di inizio lavorazione e aggiornamento stato ticket (attività automatica)
- fine lavorazione (attività di sola registrazione dell'operazione)
- inoltro mail al richiedente con notifica di chiusura richiesta e aggiornamento stato ticket (attività automatica)

Fase a) – Creazione classe derivata da “Activity”

Per la gestione del processo viene creata tramite il Modulo Schema la classe “HelpDeskSemplice” derivata dalla superclasse “Activity”.

La classe conterrà nell'esempio i seguenti attributi specifici:

Codice Attributo	Descrizione attributo	Tipo	Note
NumTicket	Numero ticket	Integer	La gestione richiede la creazione manuale sul database di una sequence utilizzata per la numerazione automatica ALTER TABLE "HelpDeskSemplice" ALTER COLUMN "NumTicket" SET DEFAULT nextval('helpdesk_semplice_num_seq');
DataTicket	Data ticket	Data	Per una attuale limitazione del Modulo Schema va impostato manualmente nel database il default a “now” ALTER TABLE "HelpDeskSemplice" ALTER COLUMN "DataTicket" SET DEFAULT now();

Richiedente	Richiedente	Reference su tabella "Assegnatario"	Persona che esegue la richiesta da cui inizia il processo
Valutazione	Valutazione	Boolean	Approvazione o rifiuto della richiesta
Motivazione	Motivazione	Text	
DataPrevistaChiusura	Data prevista chiusura	Date	
NoteInizio	Note inizio lavorazione	Text	
DataChiusura	Data chiusura	Date	
TempoImpiegato	Tempo impiegato (ore)	Integer	
NoteFine	Note fine lavorazione	Text	
StatoRichiesta	Stato	LookUp su lista "Stato richiesta helpdesk"	Stato "applicativo" della richiesta (specifico del tipo di processo, in contrapposizione con lo stato dell'attività che assume gli stessi possibili valori per tutti i processi)

Si suppone siano già preesistenti nel modello dati di CMDBuild:

- le classi "Assegnatario" (assegnatari degli asset informatici) ed "Asset" (beni IT, con relative sottoclassi)
- i domini:
 - "CespitemHelpDeskSemplice" fra "Asset" e "HelpDeskSemplice" (per correlare all'istanza del processo corrente i cespiti coinvolti)
 - "RichiedenteHelpDeskSemplice" fra "Assegnatario" e "HelpDeskSemplice" (per correlare la richiesta al dipendente)
- la tabella di LookUp "Stato richiesta helpdesk" con valori "Aperta", "Valutata", "In lavorazione", "Chiusa"

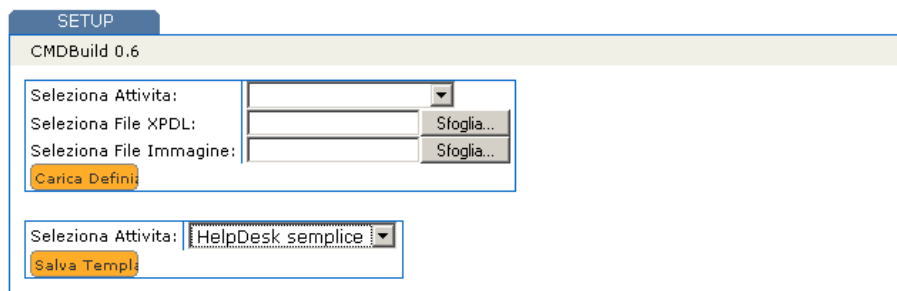
Si suppone infine siano già stato definito con il Modulo Schema l'unico ruolo coinvolto nel workflow (nell'esempio del presente tutorial denominato "CED"), il ruolo fittizio "Sistema" viene invece aggiunto automaticamente da CMDBuild nello scheletro di file XPDL esportato.

Per l'esecuzione del processo possono essere poi utilizzati i ruoli specifici sopra riportati, oppure può essere utilizzato il solo ruolo "Administrator" che può "impersonare" qualunque ruolo.

Fase b) – Esportazione "scheletro" XPDL da CMDBuild

Sempre tramite il Modulo Schema viene esportato lo "scheletro" di file XPDL comprendente:

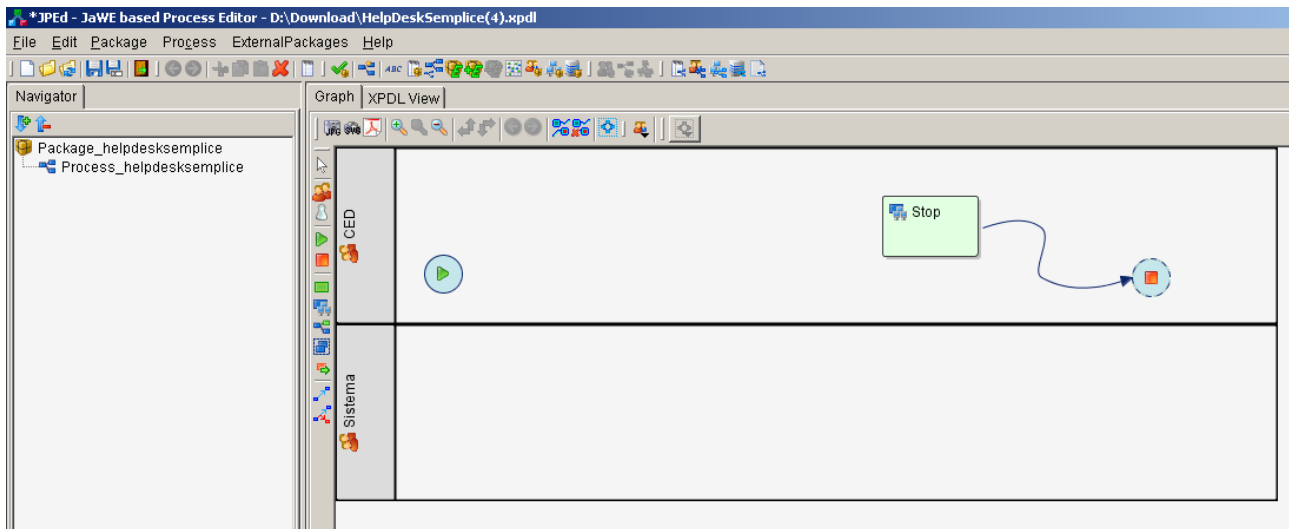
- il nome del processo
- la lista degli attributi non riservati presenti nella classe di gestione del processo
- la lista dei metodi utilizzabili
- la lista dei ruoli definiti nel sistema



Fase c) – Caricamento “scheletro” XPDL in JPED

Avviando l'applicazione JPED si procede aprendo il file XPDL esportato da CMDBuild.

Lo schema grafico iniziale proposto è il seguente:

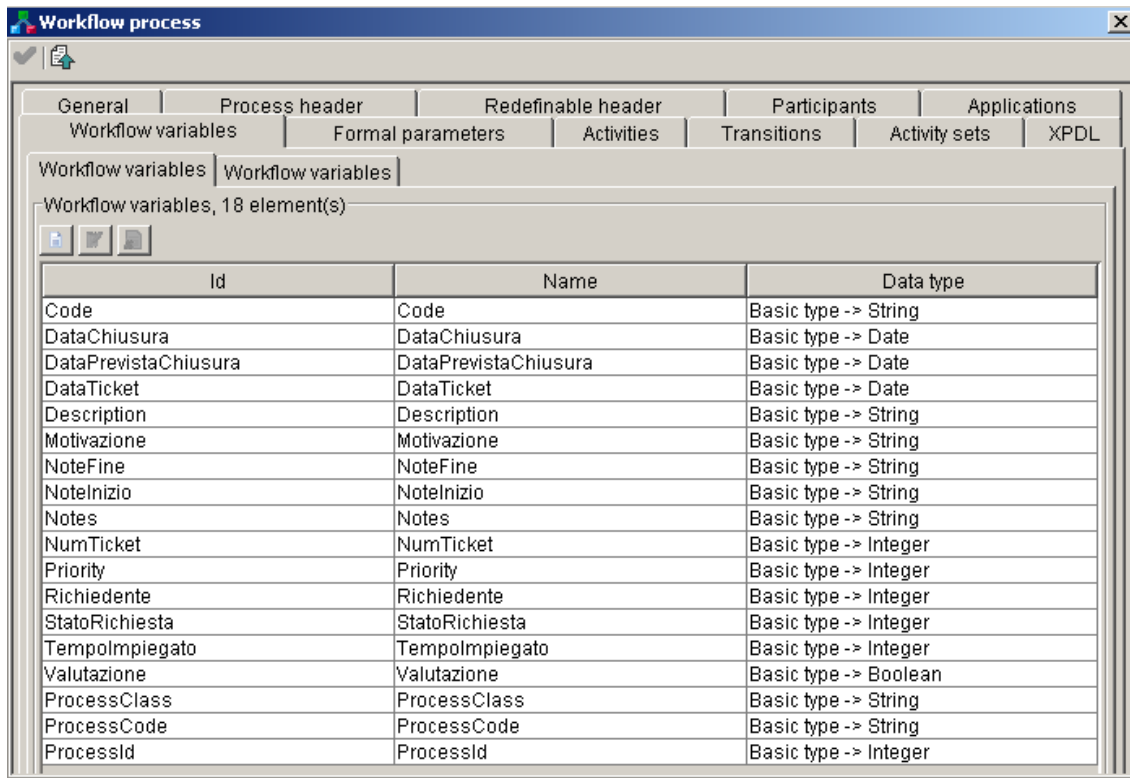


Le attività “Start” e “End” sono le attività iniziali e finali obbligatorie nella sintassi XPDL.

L'attività “Stop” viene proposta per comodità dell'utente, non prevedendo la sintassi XPDL due archi che si connettano all'attività finale (cerchio con quadrato arancione) ed essendo quindi necessario disporre di una precedente attività fittizia ove riunificare tali eventuali percorsi alternativi.

La lista delle variabili del workflow rese inizialmente disponibili in JPED (vedasi la form seguente) comprende:

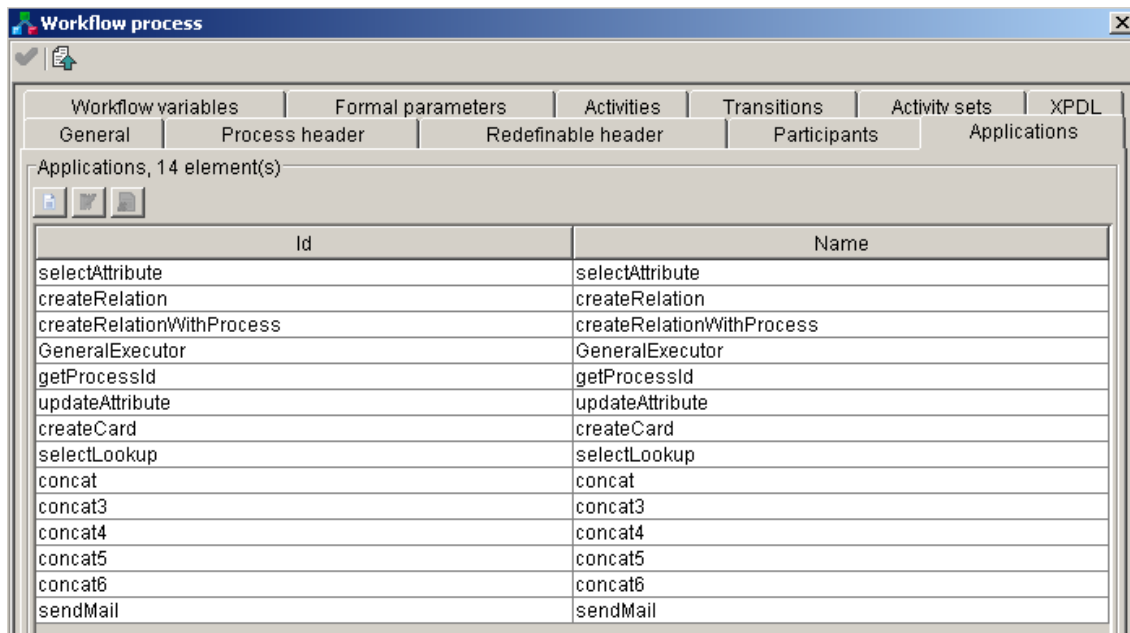
- gli attributi di “Activity” e della classe derivata per il processo specifico
- gli attributi di sistema che identificano il processo corrente



Id	Name	Data type
Code	Code	Basic type -> String
DataChiusura	DataChiusura	Basic type -> Date
DataPrevistaChiusura	DataPrevistaChiusura	Basic type -> Date
DataTicket	DataTicket	Basic type -> Date
Description	Description	Basic type -> String
Motivazione	Motivazione	Basic type -> String
NoteFine	NoteFine	Basic type -> String
NoteInizio	NoteInizio	Basic type -> String
Notes	Notes	Basic type -> String
NumTicket	NumTicket	Basic type -> Integer
Priority	Priority	Basic type -> Integer
Richiedente	Richiedente	Basic type -> Integer
StatoRichiesta	StatoRichiesta	Basic type -> Integer
TempoImpiegato	TempoImpiegato	Basic type -> Integer
Valutazione	Valutazione	Basic type -> Boolean
ProcessClass	ProcessClass	Basic type -> String
ProcessCode	ProcessCode	Basic type -> String
ProcessId	ProcessId	Basic type -> Integer

Le variabili importate da CMDBuild sono soggette ad aggiornamento automatico del proprio valore nel corso del workflow. Nel caso si volesse utilizzare le informazioni contenute in tali variabili non sarà quindi necessario utilizzare metodi di lettura nel database (quali ad esempio il tool “selectAttribute”).

La lista delle “Applications” del workflow (tools) è invece la seguente:



Id	Name
selectAttribute	selectAttribute
createRelation	createRelation
createRelationWithProcess	createRelationWithProcess
GeneralExecutor	GeneralExecutor
getProcessId	getProcessId
updateAttribute	updateAttribute
createCard	createCard
selectLookup	selectLookup
concat	concat
concat3	concat3
concat4	concat4
concat5	concat5
concat6	concat6
sendMail	sendMail

Il metodo “concat” è riportato in più versioni per disporre del numero di argomenti necessario (non è gestito nel formalismo XPDL l'overload dei metodi).

Fase d) – Completamento definizione variabili workflow in JPED

Per successivi utilizzi nel corso del processo vengono definite le seguenti ulteriori variabili del workflow:

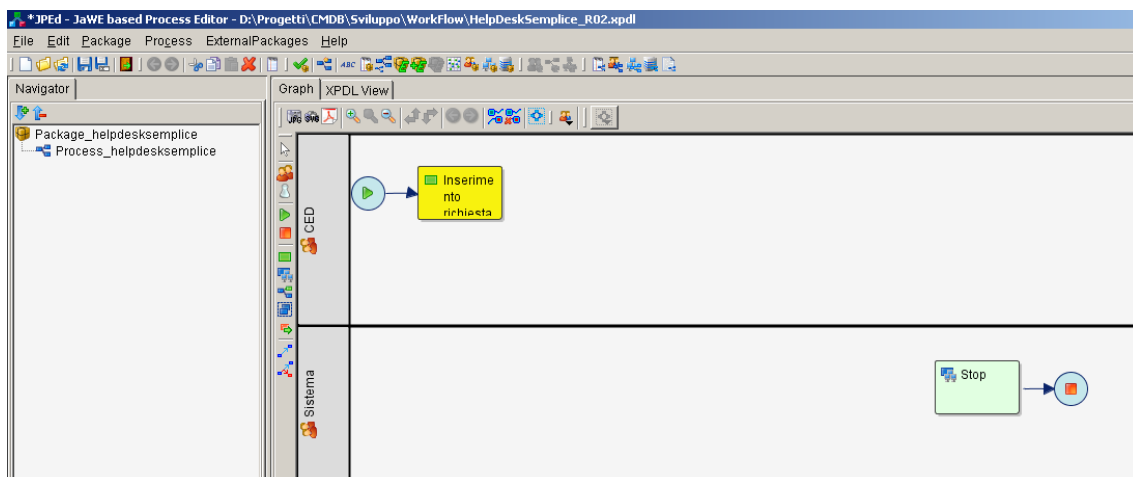
- Ret, ove memorizzare il risultato di ritorno dei metodi booleani richiamati
- MailAddress, ove memorizzare l'indirizzo di mail del richiedente
- MailContent, ove memorizzare il testo da inviare via email
- Richiedente_decodifica, ove memorizzare il nome completo del richiedente (nella variabile "Richiedente" è invece presente l' "Id" della scheda collegata tramite il campo "reference")
- NumTicket_stringa, trasformato in stringa dal metodo "selectAttribute" ed utile per essere concatenato nel testo delle mail di notifica

Fase e1) – Attività "Inserimento richiesta"

Tramite l'attività iniziale di inserimento di una richiesta viene avviato una nuova istanza del processo di HelpDesk semplificato, oggetto del presente tutorial.

Operazioni lato JPED

Deve essere inserita la nuova attività nell'area corrispondente al nome del ruolo che dovrà svolgerla, nel caso in oggetto "CED". Ricordiamo che i ruoli devono essere predefiniti in CMDBuild per essere poi esportati nel file XPDL utilizzato da JPED.



La nuova attività viene inserita tramite il pulsante "Insert activity without implementation" (vedasi spiegazione più sotto) e segue l'attività iniziale del workflow cui deve essere collegata tramite il pulsante "Insert transition".

Alla nuova attività dovranno essere assegnati un "Id" (stringa senza spazi) ed un "Name", nell'esempio "InserimentoRichiesta" ed "Inserimento richiesta".

La definizione di attributi / metodi per ogni attività descritta con JPED viene effettuata tramite apposite form cui si accede con un "doppio click" sull'attività in questione (mentre la definizione di attributi / metodi generali del workflow viene effettuata con un "doppio click" sul processo in questione).

La definizione dell'attività di "Inserimento richiesta" in JPED prevede la sola indicazione (TAB Activity Form Editor) di quali saranno gli attributi da mostrare nella form di gestione attività lato

CMDBuild, cioè “Description” e “Richiedente”.

Non sono richiesti in questo passaggio automatismi ulteriori, né attività su altre classi.

Metodi automatici (gestiti tramite “Tools”)

Non sono richiesti

Metodi con intervento utente (gestiti tramite “Extended Attributes”)

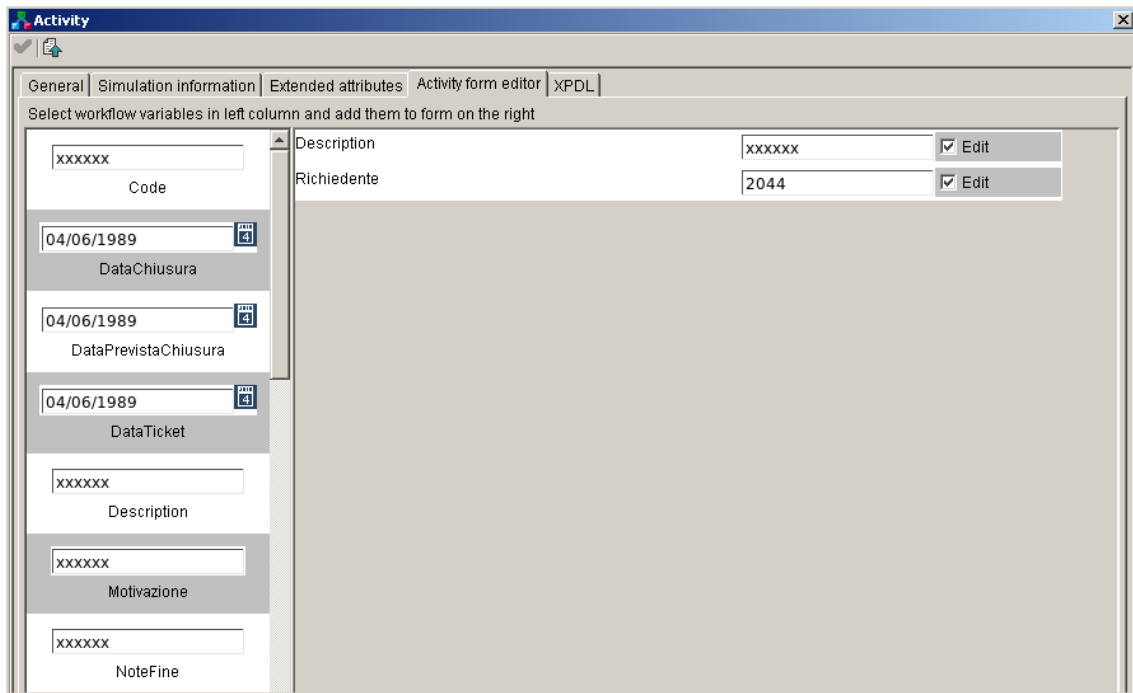
Non sono richiesti

Gestione attributi


Utilizzando l’interfaccia disponibile nel TAB “ActivityFormEditor” vengono impostati i permessi sugli attributi da gestire lato CMDBuild:

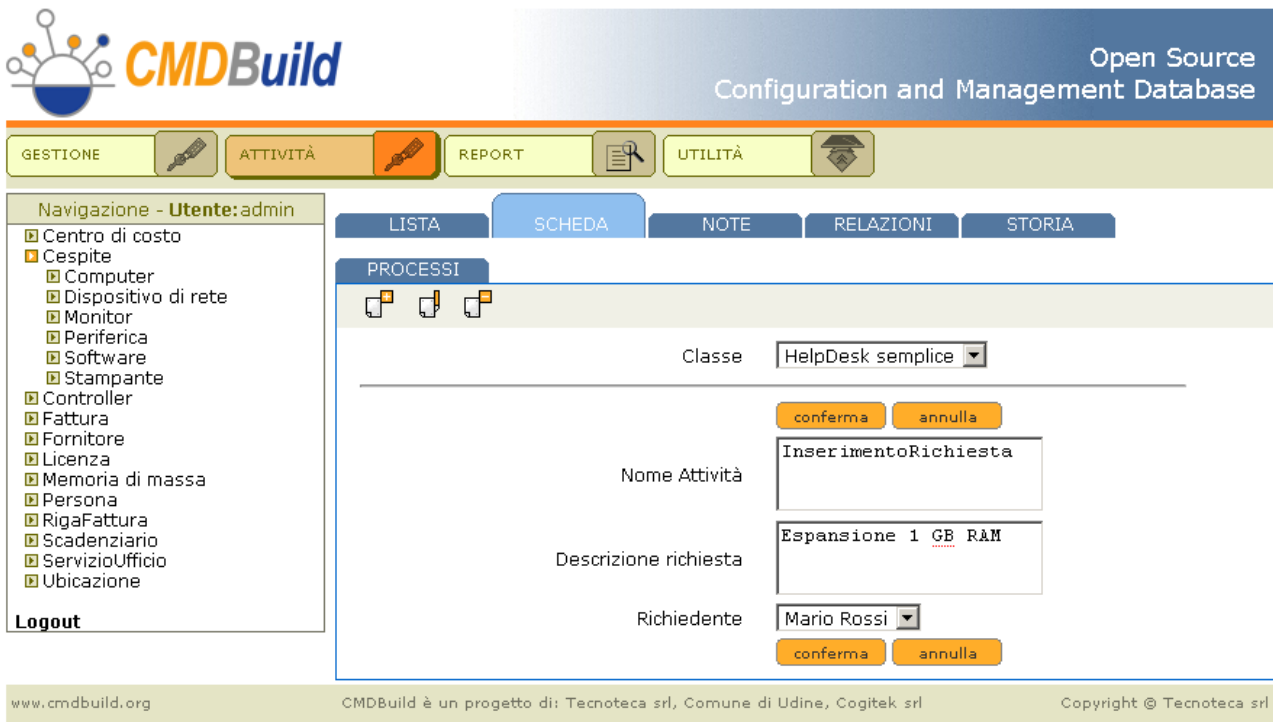
Nome	Modo	Note
Description	write	Campo descrittivo della richiesta
Richiedente	write	Persona che ha effettuato la richiesta (per default l’utente che compila la form)

Segue un esempio dell’interfaccia utente predisposta in JPEd (“Form editor”) per l’impostazione dei permessi sugli attributi.



Effetto lato CMDBuild

Le impostazioni sopra descritte produrranno in CMDBuild, all’avvio di una nuova istanza di processo (effettuabile premendo il pulsante  alto a sinistra nel TAB “Lista”), la presentazione della seguente form:



Open Source Configuration and Management Database

GESTIONE ATTIVITÀ REPORT UTILITÀ

Navigazione - Utente: admin

- Centro di costo
- Cespite
 - Computer
 - Dispositivo di rete
 - Monitor
 - Periferica
 - Software
 - Stampante
- Controller
- Fattura
- Fornitore
- Licenza
- Memoria di massa
- Persona
- RigaFattura
- Scadenziario
- ServizioUfficio
- Ubicazione

Logout

LISTA SCHEMA NOTE RELAZIONI STORIA

PROCESSI

Classe: HelpDesk semplice

conferma annulla

Nome Attività: InserimentoRichiesta

Descrizione richiesta: Espansione 1 GB RAM

Richiedente: Mario Rossi

conferma annulla

www.cmdbuild.org CMDBuild è un progetto di: Tecnoteca srl, Comune di Udine, Cogitek srl Copyright © Tecnoteca srl

Solamente gli operatori abilitati al ruolo cui appartiene l'attività in questione (nell'esempio il ruolo "CED") potranno eseguire la nuova attività.

Fase e2) – Attività "Notifica richiesta aperta"

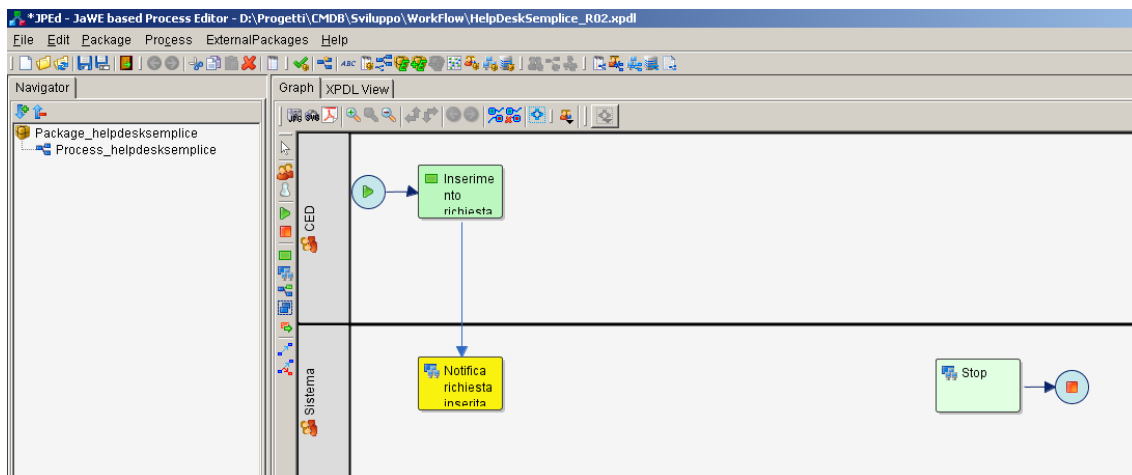
La nuova attività è un'attività automatica e ha lo scopo di:

- impostare il nuovo stato applicativo della richiesta ("Aperta")
- inviare al richiedente una notifica di creazione del nuovo ticket, comprensiva di stato, richiedente, numero ticket (nel frattempo attribuito in automatico tramite il valore di default impostato sul database e la apposita sequence creata nel sistema), descrizione della richiesta

Operazioni lato JPED

L'attività viene inserita con il pulsante di JPED "Insert tool activity" (viene creata un'attività automatica) e segue l'attività "Inserimento richiesta" cui deve essere collegata tramite il pulsante "Insert transition".

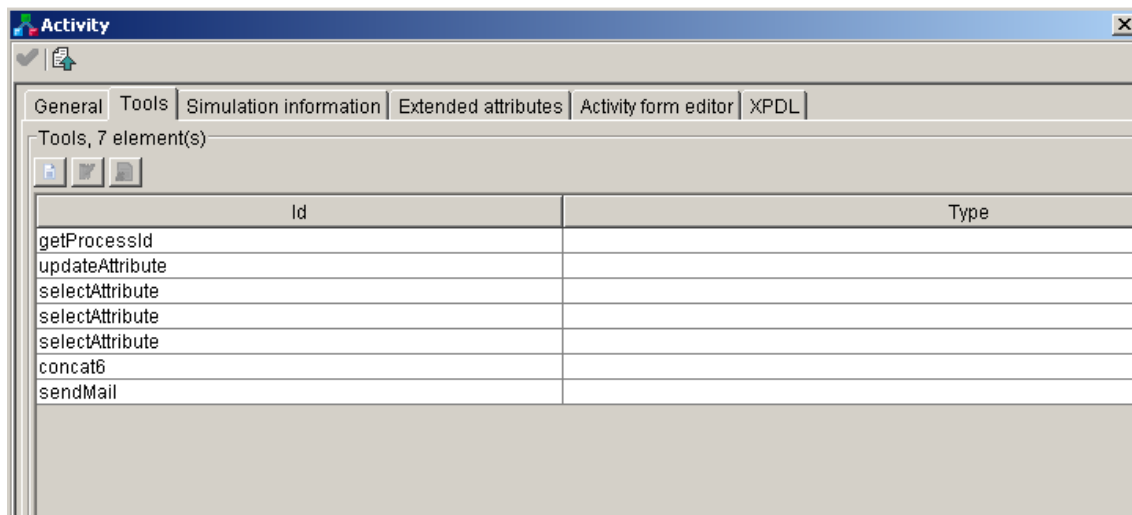
Il ruolo interessato è in questo caso quello fittizio di "Sistema".



Metodi automatici

Viene richiamata l'esecuzione della seguente sequenza di metodi (vedi sintassi in Appendice):

- `getProcessId`, per recuperare l' "Id" dell'istanza del processo corrente
- `updateAttribute`, per aggiornare lo stato applicativo dell'istanza di processo
- `selectAttribute`, per recuperare il nome completo del richiedente (nella variabile "Richiedente" è invece presente l' "Id" della scheda collegata tramite il campo "reference")
- `selectAttribute`, per recuperare l'email del richiedente (nella scheda "Assegnatario")
- `concat6`, per concatenare le stringhe (nel numero di 6) che comporranno il testo della mail
- `sendmail`, per l'invio della mail di notifica all'utente che ha aperto il ticket



Id	Type
<code>getProcessId</code>	
<code>updateAttribute</code>	
<code>selectAttribute</code>	
<code>selectAttribute</code>	
<code>selectAttribute</code>	
<code>concat6</code>	
<code>sendMail</code>	

Seguono i dettagli dei parametri impostati per ogni metodo chiamato.

Tool

Id:

Type:

Formal parameters, 3 element(s)

Id	Mode	Data type
ProcessClass	In	Basic type -> String
ProcessCode	In	Basic type -> String
ProcessID	Out	Basic type -> Inte...

Actual parameters, 3 element(s)

Actual parameter
ProcessClass
ProcessCode
ProcessId

Description:

Extended attributes, 0 element(s)

Name	Value

Tool

Id:

Type:

Formal parameters, 5 element(s)

Id	Mode	Data type
ClassName	In	Basic type -> String
AttributeName	In	Basic type -> String
ObjId	In	Basic type -> Inte...
AttributeValue	In	Basic type -> String
Done	Out	Basic type -> Bool...

Actual parameters, 5 element(s)

Actual parameter
ProcessClass
"StatoRichiesta"
ProcessId
"Aperta"
Ret

Description:

Extended attributes, 0 element(s)

Name	Value

Tool

Id:

Type:

Formal parameters, 4 element(s)

Id	Mode	Data type
ClassName	In	Basic type -> String
AttributeName	In	Basic type -> String
ObjId	In	Basic type -> Inte...
AttributeValue	Out	Basic type -> String

Actual parameters, 4 element(s)

Actual parameter
"Assegnatario"
"Description"
Richiedente
Richiedente_decodifica

Description:

Extended attributes, 0 element(s)

Name	Value

Tool

Id:

Type:

Formal parameters, 4 element(s)

Id	Mode	Data type
ClassName	In	Basic type -> String
AttributeName	In	Basic type -> String
ObjId	In	Basic type -> Inte...
AttributeValue	Out	Basic type -> String

Actual parameters, 4 element(s)

Actual parameter
"Assegnatario"
"Email"
Richiedente
MailAddress

Description:

Extended attributes, 0 element(s)

Name	Value

Tool

Id:

Type:

Formal parameters, 4 element(s)

Id	Mode	Data type
ClassName	In	Basic type -> String
AttributeName	In	Basic type -> String
ObjId	In	Basic type -> Inte...
AttributeValue	Out	Basic type -> String

Actual parameters, 4 element(s)

Actual parameter
"HelpDeskSemplice"
"NumTicket"
ProcessId
NumTicket_stringa

Description:

Extended attributes, 0 element(s)

Name	Value
------	-------

Tool

Id:

Type:

Formal parameters, 7 element(s)

Id	Mode	Data type
InputString1	In	Basic type -> String
InputString2	In	Basic type -> String
InputString3	In	Basic type -> String
InputString4	In	Basic type -> String
InputString5	In	Basic type -> String
InputString6	In	Basic type -> String
OutputString	Out	Basic type -> String

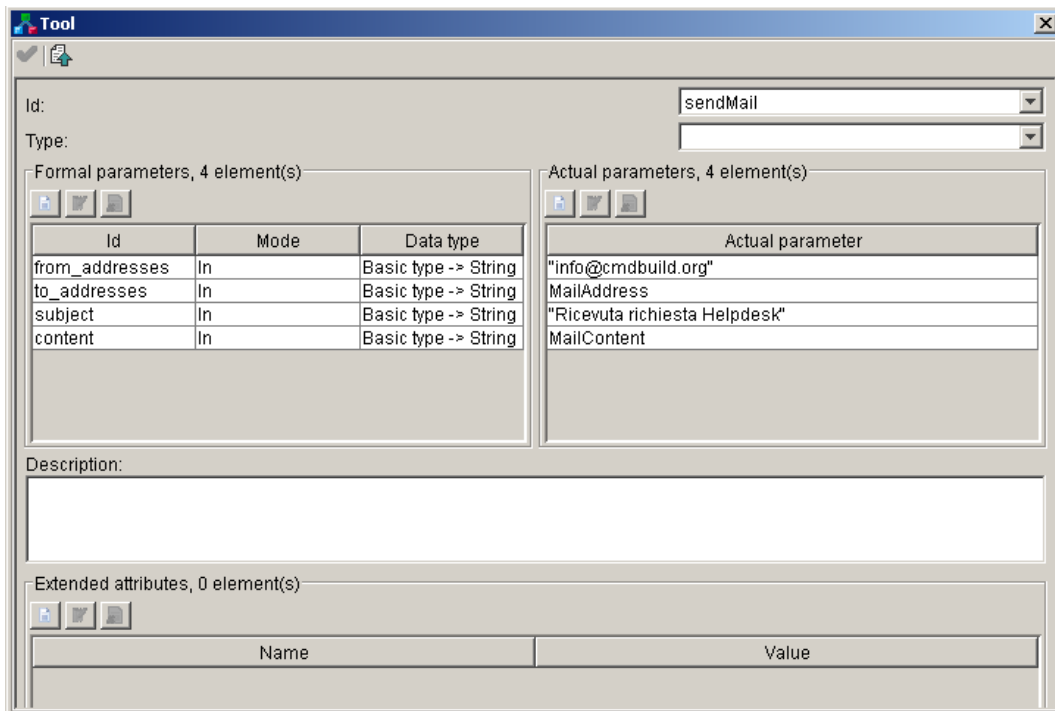
Actual parameters, 7 element(s)

Actual parameter
"Stato richiesta: Aperta - Richiedente: "
Richiedente_decodifica
" - Ticket: "
NumTicket_stringa
" - Richiesta: "
Description
MailContent

Description:

Extended attributes, 0 element(s)

Name	Value
------	-------



Tool

Id:

Type:

Formal parameters, 4 element(s)

Id	Mode	Data type
from_addresses	In	Basic type -> String
to_addresses	In	Basic type -> String
subject	In	Basic type -> String
content	In	Basic type -> String

Actual parameters, 4 element(s)

Actual parameter
"info@cmdbuild.org"
MailAddress
"Ricevuta richiesta Helpdesk"
MailContent

Description:

Extended attributes, 0 element(s)

Name	Value
------	-------

Metodi con intervento utente

Non sono richiesti

Gestione attributi

Non applicabile

Effetto lato CMDBuild

Non applicabile

Fase e3) – Attività “Valutazione richiesta”

Tramite l'attività di valutazione l'operatore incaricato del CED approva e fa proseguire la lavorazione della richiesta, oppure ne rifiuta l'approvazione, motivando comunque la decisione.

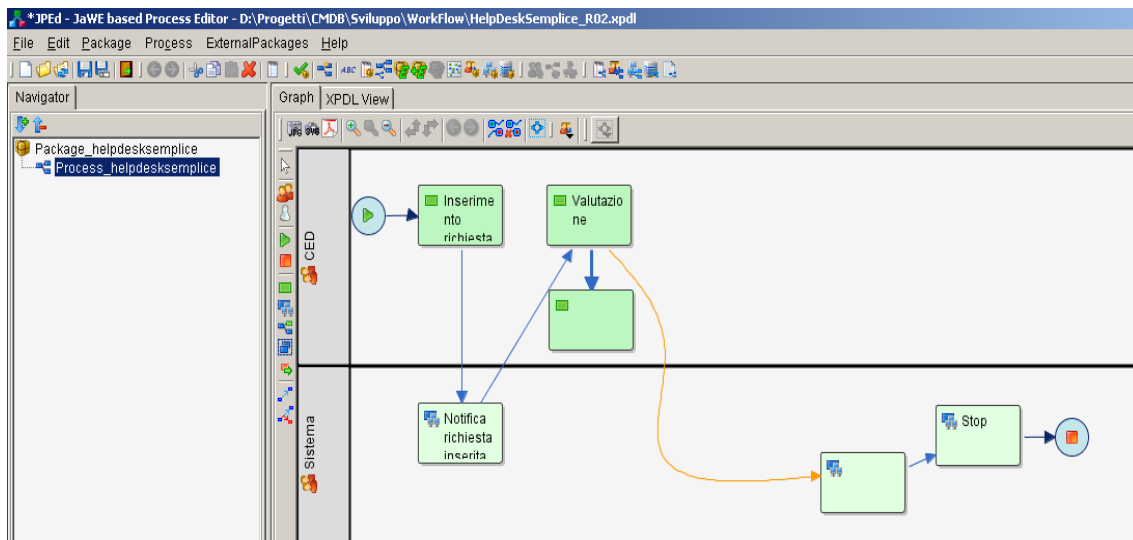
Operazioni lato JPEd

La nuova attività viene inserita con il pulsante di JPEd “Insert route activity” (viene creato un “bivio” nel caso rispettivamente di approvazione o rifiuto della richiesta) e segue l'attività “Inserimento richiesta” cui deve essere collegata tramite il pulsante “Insert transition”.

Alla nuova attività vengono assegnati l' “Id” “Valutazione” ed il “Name” “Valutazione”.

La definizione dell'attività di “Valutazione richiesta” in JPEd prevede:

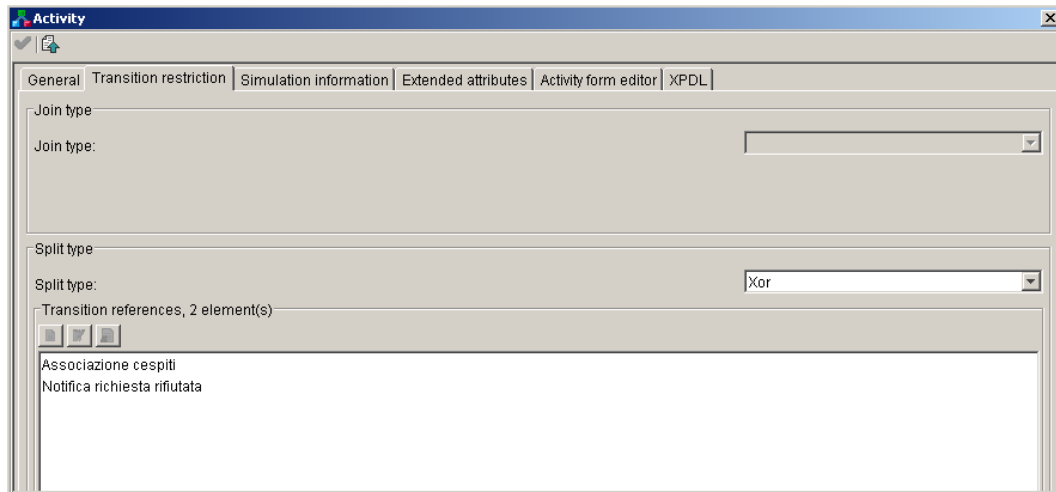
- l'indicazione (TAB ActivityFormEditor) di quali saranno gli attributi da mostrare nella form di gestione attività lato CMDBuild, cioè “Description” e “Richiedente” (come descritto più avanti)
- le condizioni di percorrenza delle due uscite previste (TAB Transition restriction) nel caso di approvazione o rifiuto della richiesta
- l'attributo esteso (Extended Attribute) che identifica questa attività come attività eseguibile in forma “veloce” direttamente dalla lista della attività in CMDBuild (pulsante di “spunta”)



Il ruolo interessato è ancora quello di “CED”.

Anche per questo passaggio non sono richiesti automatismi ulteriori da svolgere poi in CMDBuild, né attività su altre classi.

Vanno a questo punto specificate in JPEd le condizioni di percorrenza delle due uscite previste (approvazione / rifiuto), come indicato nello screenshot sotto riportato (la presenza delle due attività di destinazione, che verranno riprese nel dettaglio più avanti, è stata anticipata al passaggio corrente per poterne descrivere i meccanismi di transizione previsti).



Per i due archi di transizione fra “Valutazione” e le attività successive, secondo i due percorsi previsti, devono essere specificate le condizioni di percorrenza accedendo alle relative form di specifica parametri (vedi screenshot successivi) previste da JPEd, ed in particolare:

- fra le proprietà del primo arco va selezionato come tipo di condizione “condition” e come dettaglio il fatto che tale arco verrà percorso nel caso l'attributo “Approval” assuma valore “true” (specificato tramite la notazione “Valutazione.booleanvalue()”, la condizione “uguale a true” è implicita essendo un attributo “boolean”)
- fra le proprietà del secondo arco va selezionato come tipo di condizione “otherwise”

Transition [XPDL]

Id:

Name:

From:

To:

Condition

Type:

`Valutazione.booleanValue();`

Description:

Extended attributes, 0 element(s)

Name	Value

Transition [XPDL]

Id:

Name:

From:

To:

Condition

Type:

.

Description:

Extended attributes, 0 element(s)

Name	Value

Metodi automatici

Non sono richiesti

Metodi con intervento utente

Viene richiamato il metodo API “quickAccept” di CMDBuild che richiederà per quell'attività la disponibilità dei pulsanti di approvazione / rifiuto veloce nel TAB “Lista” dell'interfaccia utente di CMDBuild.

Nome	Parametri input	Parametri output
quickAccept	Valutazione	

Gestione attributi

Utilizzando l'interfaccia disponibile nel TAB “ActivityFormEditor” vengono impostati i permessi sugli attributi da gestire lato CMDBuild:

Nome	Modo	Note
DataTicket	read	Data di inserimento della richiesta, valorizzata in automatico sul database tramite l'impostazione del valore di default “now()”
NumTicket	read	Numero ticket, valorizzato in automatico sul database tramite la “sequence” appositamente predisposta
Richiedente	read	Utente che ha aperto il ticket
Description	read	Descrizione della richiesta
Valutazione	write	Approvazione o rifiuto richiesta
Motivazione	write	Motivazione della decisione

La form presentata in CMDBuild visualizzerà in sola lettura i primi quattro attributi e richiederà la compilazione degli ultimi due.

Effetto lato CMDBuild

Alla conferma dell'attività “Inserimento richiesta” l'avanzamento del processo tramite il motore di workflow produrrà nel TAB “Lista” la presenza della nuova riga corrispondente all'attività di approvazione (nell'esempio a fine lista).



GESTIONE ATTIVITÀ REPORT UTILITÀ

Navigazione - Utente: admin

- Centro di costo
- Cespite
 - Computer
 - Dispositivo di rete
 - Monitor
 - Periferica
 - Software
 - Stampante
- Controller
- Fattura
- Fornitore
- Licenza
- Memoria di massa
- Persona
- RigaFattura
- Scadenziario
- ServizioUfficio
- Ubicazione

Logout

LISTA SCHEDA NOTE RELAZIONI STORIA PROCESSI

Elenco: HelpDesk semplice

Stato: Processo: HelpDesk semplice

Sei a pagina 1 di 1

Processo	Nome Attività	Descrizione richiesta	Stato attività	Numero Ticket	Data ticket	Richiedente	Stato	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
HelpDesk semplice	Inizio lavorazione	Sostituire il toner della stampante	Avviato	8	2007-06-17	Mario Rossi	Valutata	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
HelpDesk semplice	Valutazione	Espansione 1 GB RAM	Avviato	10	2007-06-17	Mario Rossi	Aperta	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
HelpDesk semplice	Associazione cespiti	Il PC non si collega a internet	Avviato	9	2007-06-17	Mario Rossi	Aperta	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

www.cmdbuild.org CMDBuild è un progetto di: Tecnoteca srl, Comune di Udine, Cogitek srl Copyright © Tecnoteca srl

La nuova riga (evidenziata in rosso sulla prima colonna) sarà visibile con link attivo solo per gli operatori abilitati al ruolo cui è riservata l'attività in questione (nell'esempio il ruolo "CED"), che potranno quindi eseguirla.

La nuova riga sarà anche visibile (ma con link non attivo e quindi non eseguibile) agli operatori abilitati ad altri ruoli coinvolti nel processo, che potranno quindi consultare l'avanzamento del processo in quanto cointeressati alla sua esecuzione, ma non potranno intervenire nell'attività corrente.

Le impostazioni effettuate in JPED produrranno in CMDBuild, seguendo il link sull'attività corrente, l'apertura della seguente form, tramite cui il responsabile provvederà ad approvare o rifiutare la richiesta:

GESTIONE ATTIVITÀ REPORT UTILITÀ

Navigazione - Utente: admin

- Centro di costo
- Cespite
 - Computer
 - Dispositivo di rete
 - Monitor
 - Periferica
 - Software
 - Stampante
- Controller
- Fattura
- Fornitore
- Licenza
- Memoria di massa
- Persona
- RigaFattura
- Scadenziario
- ServizioUfficio
- Ubicazione

Logout

LISTA SCHEDA NOTE RELAZIONI STORIA PROCESSI

classe HelpDesk semplice

conferma annulla

Nome Attività Valutazione

Descrizione richiesta Espansione 1 GB RAM

Numero Ticket 10

Data ticket 2007-06-17

Richiedente Mario Rossi

Valutazione S

Motivazione ok, si approva la richiesta

conferma annulla

www.cmdbuild.org CMDBuild è un progetto di: Tecnoteca srl, Comune di Udine, Cogitek srl Copyright © Tecnoteca srl

Il alternativa, solamente per le attività di approvazione, si potrà utilizzare la “forma veloce” di conferma / rifiuto, tramite gli appositi pulsanti presenti sul TAB “Lista” (vedi screenshot pagina precedente).

Fase e4) – Attività “Associazione cespiti”

L'attività in questione ha l'obiettivo di guidare l'operatore alla correlazione fra il processo in questione (ticket di HelpDesk) ed i cespiti interessati.

La correlazione fra processo (classe HelpDeskSemplice) e richiedente (classe Assegnatario) viene invece creata in automatico dal sistema grazie al campo reference che collega le due classi.

Nel caso della correlazione con la classe “Cespiti” si è scelto di utilizzare la forma dell'inserimento manuale guidato delle relazioni, potendo essere presenti più cespiti interessati allo stesso ticket (nel caso di tipologie di relazioni N:M non è ovviamente possibile utilizzare campi reference).

Tale scelta consente anche di arricchire il presente tutorial con un caso di operazione semiautomatica, cioè svolta dall'operare ma in modo guidato dal sistema (in CMDBuild è il caso tipico di utilizzo degli “extended attribute”).

Sia le relazioni inserite in modo esplicito con gli asset coinvolti, sia la relazione inserita in modo implicito con chi ha aperto il ticket, saranno poi normalmente consultabili e navigabili nel sistema (per analisi su frequenze di guasto fra i cespiti, sull'impegno richiesto al CED dai diversi uffici, ecc).

Operazioni lato JPEd

L'attività viene inserita nell'area corrispondente al ruolo “CED” tramite il pulsante “Insert activity without implementation”. Si utilizzerà poi un “extended attribute” per richiamare il metodo API che proporrà all'utente il link per la creazione guidata delle relazioni.

Metodi automatici

Non sono richiesti

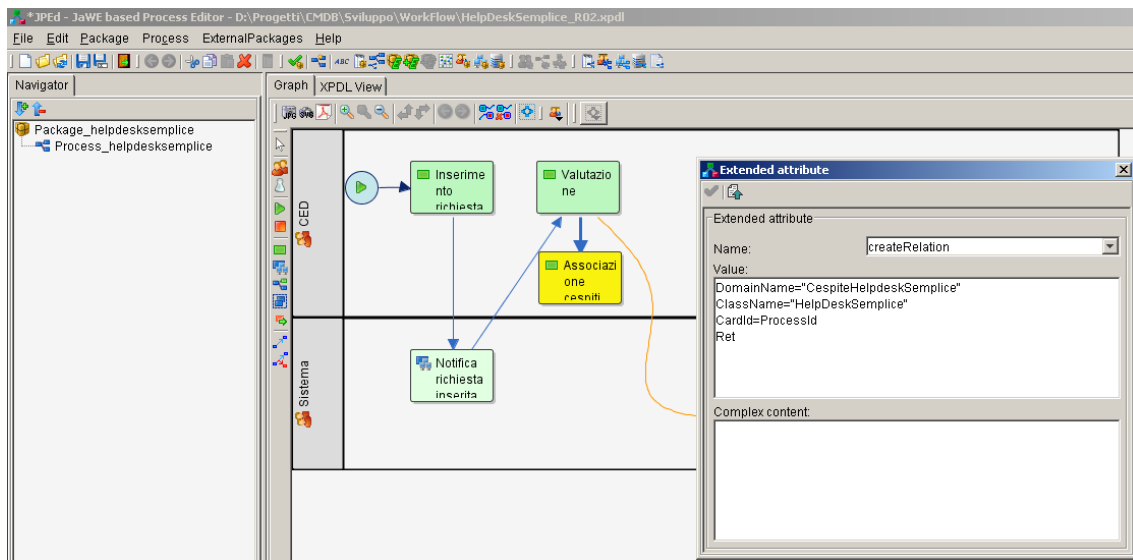
Metodi con intervento utente

Viene richiamato il metodo API “createRelation” di CMDBuild, che guiderà l'utente alla creazione di nuove relazioni fra l'istanza del processo corrente e una o più schede da selezionare sulle sottoclassi di “Asset”, e che presenta la seguente sintassi:

Nome	Parametri input	Parametri output
createRelation	DomainName = "CespitiHelpdeskSemplice" ClassName = "HelpDeskSemplice" CardId = ProcessId	Ret

L'associazione di tale metodo all'attività corrente viene effettuato tramite il TAB “Extended attribute” dell'attività stessa.

Segue lo screenshot della form di specifica del metodo in JPEd:



Come effetto delle impostazioni sopra descritte, in CMDBuild sarà presentata la form di inserimento relazioni tramite un apposito link predisposto sulla form di gestione attività.

Gestione attributi

Anche in questo caso, utilizzando l'interfaccia disponibile nel TAB "ActivityFormEditor", vengono impostati i permessi sugli attributi da gestire lato CMDBuild nell'attività corrente:

Nome	Modo	Note
DataTicket	read	Data di inserimento della richiesta, valorizzata in automatico sul database tramite l'impostazione del valore di default "now()"
NumTicket	read	Numero ticket, valorizzato in automatico sul database tramite la "sequence" appositamente predisposta
Richiedente	read	Utente che ha aperto il ticket
Description	read	Descrizione della richiesta
Valutazione	read	Approvazione o rifiuto richiesta
Motivazione	read	Motivazione della decisione

La form presentata in CMDBuild visualizzerà in sola lettura tutti e sei gli attributi, mantenendo abilitato il solo link per la creazione delle relazioni con schede della classe "Assegnatario".

Effetto lato CMDBuild

Alla conferma dell'attività ("Valutazione richiesta"), avanzando nel processo verrà presentata nel TAB "Lista" della gestione attività una nuova riga corrispondente all'attività di associazione al ticket di uno o più cespiti individuati nelle sottoclassi di "Asset".

La nuova riga sarà visibile con link attivo solo per gli operatori abilitati al ruolo cui appartiene l'attività in questione (nell'esempio il ruolo "CED").

Le impostazioni effettuate in JPEd produrranno in CMDBuild, seguendo il link sull'attività corrente, l'apertura di una form con il link per la creazione della relazione prevista.



Open Source
Configuration and Management Database

GESTIONE ATTIVITÀ REPORT UTILITÀ

Navigazione - Utente:admin

- Centro di costo
- Cespite
- Controller
- Fattura
- Fornitore
- Licenza
- Memoria di massa
- Persona
- RigaFattura
- Scadenziario
- ServizioUfficio
- Ubicazione

Logout

LISTA SCHEDA NOTE RELAZIONI STORIA PROCESSI

3/3

classe HelpDesk semplice

conferma annulla

Nome Attività Associazione cespiti

Descrizione richiesta Espansione 1 GB RAM

Numero Ticket 10

Data ticket 2007-06-17

Richiedente Mario Rossi

Valutazione Si

Motivazione ok, si approva la richiesta

Creare una relazione [Crea una relazione](#)

conferma annulla

www.cmdbuild.org CMDBuild è un progetto di: Tecnoteca srl, Comune di Udine, Cogitek srl Copyright © Tecnoteca srl

Tramite il link guidato si accede alla form di creazione delle nuove relazioni:

GESTIONE ATTIVITÀ REPORT UTILITÀ

Navigazione - Utente:admin

- Centro di costo
- Cespite
- Controller

LISTA SCHEDA NOTE RELAZIONI STORIA PROCESSI

3/3

classe HelpDesk semplice

conferma annulla

Nome Attività Associazione cespiti

Descrizione richiesta Espansione 1 GB RAM

Numero Ticket 10

Data ticket 2007-06-17

Richiedente Mario Rossi

Valutazione Si

Motivazione ok, si approva la richiesta

Creare una relazione [Crea una relazione](#)

conferma annulla

http://localhost:8080 - Mozilla Firefox

Inserisci nuova relazione

Classe Computer

Dominio relativo a cespite

Oggetto

conferma annulla

Completato

www.cmdbuild.org CMDBuild è un progetto di: Tecnoteca srl, Comune di Udine, Cogitek srl Copyright © Tecnoteca srl

Fase e5) – Attività “Notifica richiesta valutata”

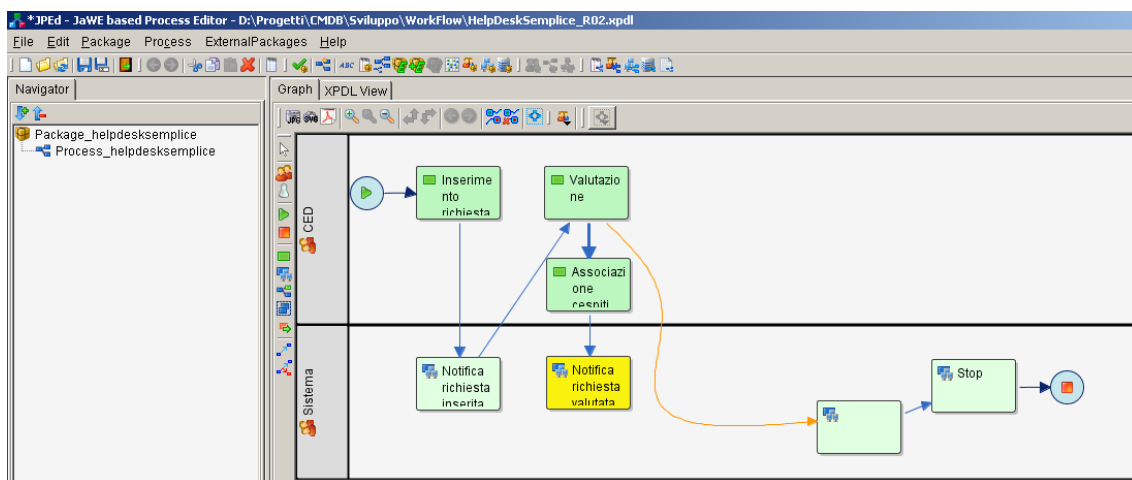
La nuova attività è un'attività automatica e ha lo scopo di:

- impostare il nuovo stato applicativo della richiesta (“Valutata”)
- inviare al richiedente una notifica di avanzamento del processo

Operazioni lato JPED

L'attività viene inserita con il pulsante di JPED “Insert tool activity” (viene creata un'attività automatica) e segue l'attività “Associazione cespiti” cui deve essere collegata tramite il pulsante “Insert transition”.

Il ruolo interessato è in questo caso quello fittizio di “Sistema”.

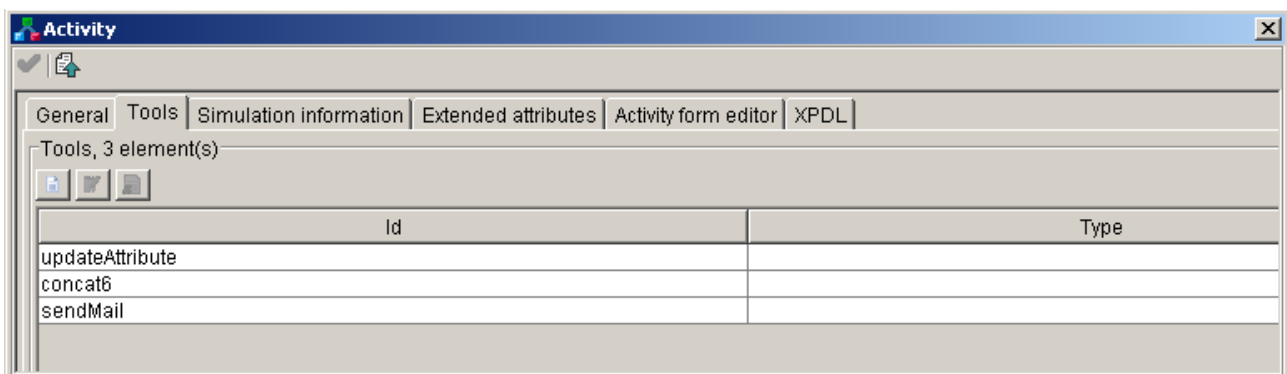


Metodi automatici

Viene richiamata l'esecuzione della seguente sequenza di metodi (vedi sintassi in Appendice):

- updateAttribute, per aggiornare lo stato applicativo dell'istanza di processo
- concat6, per concatenare le stringhe (nel numero di 6) che comporranno il testo della mail
- sendmail, per l'invio della mail di notifica all'utente che ha aperto il ticket

Si ricorda che le informazioni già acquisite sul richiedente e sul numero ticket nel corso della precedente attività automatica di notifica rimangono disponibili e non necessitano quindi di essere richieste nuovamente.



I dettagli dei metodi richiamati non vengono riproposti essendo del tutto simili a quelli già presentati nell'ambito dell'attività e2).

Metodi con intervento utente

Non sono richiesti

Gestione attributi

Non applicabile

Effetto lato CMDBuild

Non applicabile

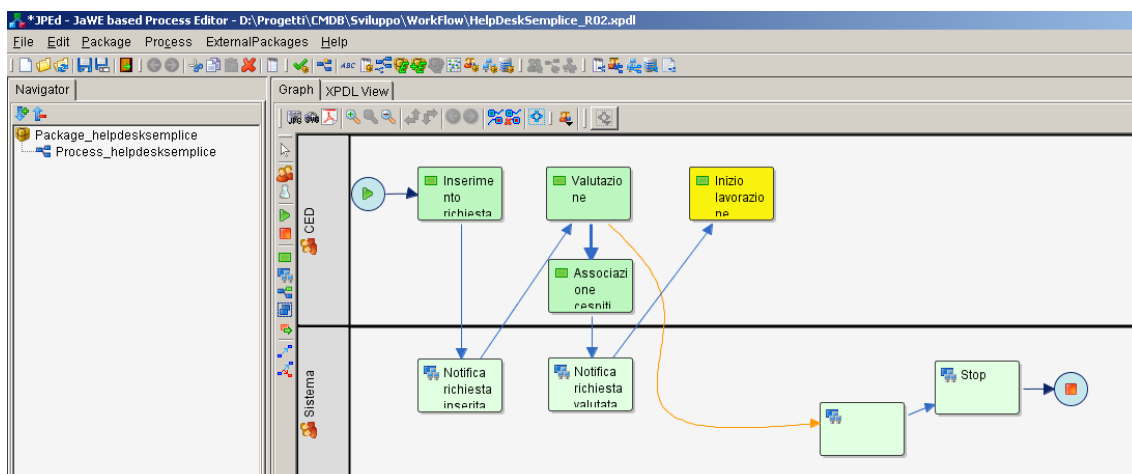
Fase e6) – Attività “Inizio lavorazione”

Tramite l'attività di inizio lavorazione l'operatore del CED segnala l'avvio dei lavori necessari alla risoluzione del ticket di HelpDesk.

Nella stessa occasione l'operatore specifica anche la data prevista di fine lavori ed eventuali note di inizio lavorazione.

Operazioni lato JPEd

L'attività viene inserita nell'area corrispondente al ruolo “CED” tramite il pulsante “Insert activity without implementation”.



Metodi automatici

Non sono richiesti

Metodi con intervento utente

Non sono richiesti

Gestione attributi

Utilizzando l'interfaccia disponibile nel TAB "ActivityFormEditor" vengono impostati i permessi sugli attributi da gestire lato CMDBuild nell'attività corrente:

Nome	Modo	Note
DataTicket	read	Data di inserimento della richiesta, valorizzata in automatico sul database tramite l'impostazione del valore di default "now()"
NumTicket	read	Numero ticket, valorizzato in automatico sul database tramite la "sequence" appositamente predisposta
Richiedente	read	Utente che ha aperto il ticket
Description	read	Descrizione della richiesta
Valutazione	read	Approvazione o rifiuto richiesta
Motivazione	read	Motivazione della decisione
DataPrevistaChiusura	write	Data prevista chiusura intervento
Notelinizio	write	Note di inizio lavorazione

La form presentata in CMDBuild visualizzerà in sola lettura i primi sei attributi, richiedendo all'operatore di specificare la data prevista per la chiusura del ticket ed eventuali note di inizio lavorazione.

Effetto lato CMDBuild

Alla conferma dell'attività ("Valutazione richiesta"), avanzando nel processo verrà presentata nel TAB "Lista" della gestione attività una nuova riga corrispondente all'attività di inizio lavorazione.

La nuova riga sarà visibile con link attivo solo per gli operatori abilitati al ruolo cui appartiene l'attività in questione (nell'esempio il ruolo "CED").

Le impostazioni effettuate in JPEd produrranno in CMDBuild, seguendo il link sull'attività corrente, l'apertura della seguente form:



Open Source
Configuration and Management Database

GESTIONE ATTIVITÀ REPORT UTILITÀ

Navigazione - Utente: admin

- Centro di costo
- Cespite
- Controller
- Fattura
- Fornitore
- Licenza
- Memoria di massa
- Persona
- RigaFattura
- Scadenziario
- ServizioUfficio
- Ubicazione

Logout

LISTA SCHEDA NOTE RELAZIONI STORIA PROCESSI

classe HelpDesk semplice

conferma annulla

Nome Attività Inizio lavorazione

Descrizione richiesta Espansione 1 GB RAM

Numero Ticket 10

Data ticket 2007-06-17

Richiedente Mario Rossi

Valutazione Si

Motivazione ok, si approva la richiesta

Data prevista chiusura 2007-06-21

Note inizio lavorazione tutto ok

conferma annulla

www.cmdbuild.org CMDBuild è un progetto di: Tecnoteca srl, Comune di Udine, Cogitek srl Copyright © Tecnoteca srl

Fase e7) – Attività “Notifica inizio lavorazione”

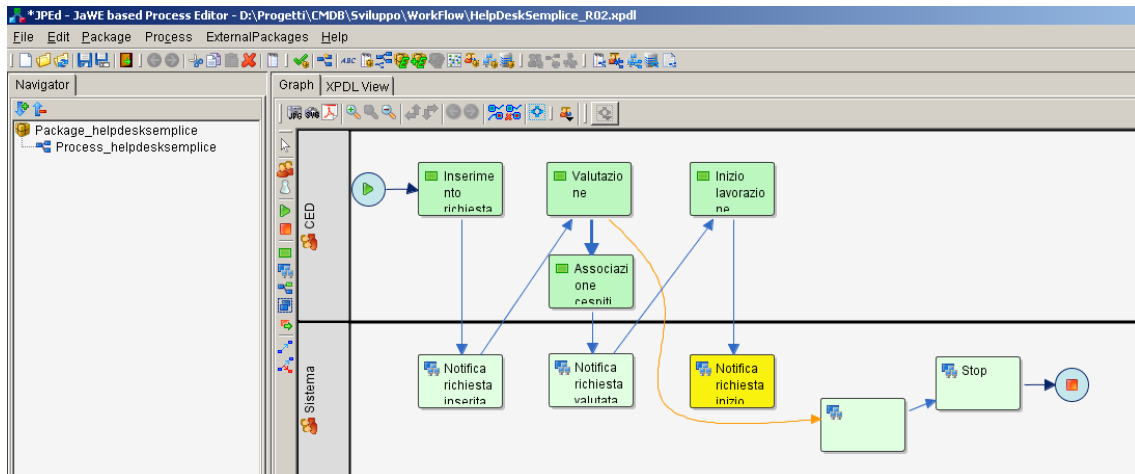
La nuova attività è un'attività automatica e ha lo scopo di:

- impostare il nuovo stato applicativo della richiesta (“In lavorazione”)
- inviare al richiedente una notifica di avanzamento del processo

Operazioni lato JPED

L'attività viene inserita con il pulsante di JPED “Insert tool activity” (viene creata un'attività automatica) e segue l'attività “Inizio lavorazione” cui deve essere collegata tramite il pulsante “Insert transition”.

Il ruolo interessato è in questo caso quello fittizio di “Sistema”.



Metodi automatici

Viene richiamata l'esecuzione della seguente sequenza di metodi (vedi sintassi in Appendice):

- updateAttribute, per aggiornare lo stato applicativo dell'istanza di processo
- concat6, per concatenare le stringhe (nel numero di 6) che comporranno il testo della mail
- sendmail, per l'invio della mail di notifica all'utente che ha aperto il ticket

Si ricorda nuovamente che le informazioni già acquisite sul richiedente e sul numero ticket nel corso della precedente attività automatica di notifica rimangono disponibili e non necessitano quindi di essere richieste nuovamente.

Id	Type
updateAttribute	
concat6	
sendMail	

I dettagli dei metodi richiamati non vengono riproposti essendo del tutto simili a quelli già presentati nell'ambito dell'attività e2).

Metodi con intervento utente

Non sono richiesti

Gestione attributi

Non applicabile

Effetto lato CMDBuild

Non applicabile

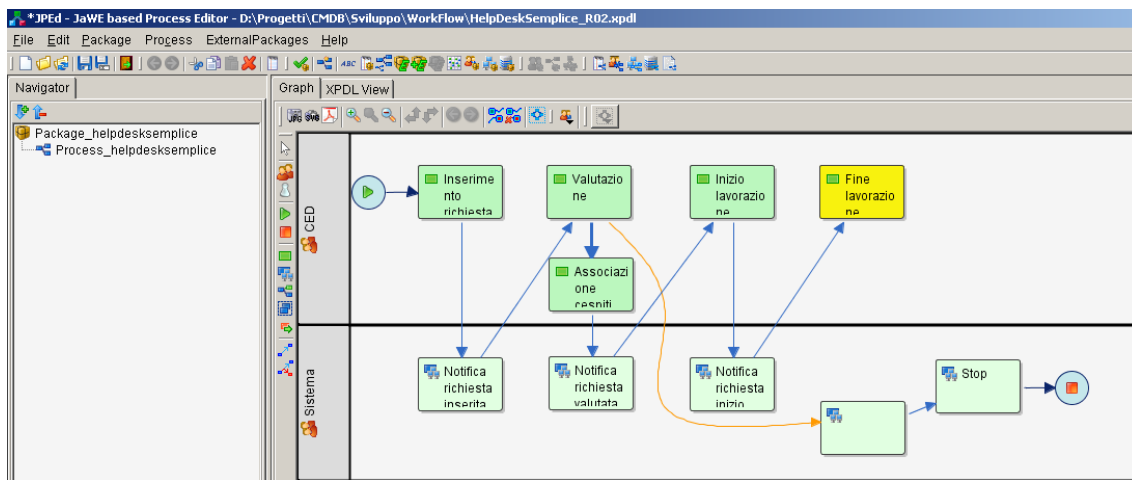
Fase e8) – Attività “Fine lavorazione”

Tramite l'attività di fine lavorazione l'operatore del CED segnala la conclusione dei lavori necessari alla risoluzione del ticket di HelpDesk.

Nella stessa occasione l'operatore specifica anche la data effettiva di fine lavori, l'impegno richiesto in ore ed eventuali note di fine lavorazione.

Operazioni lato JPEd

L'attività viene inserita nell'area corrispondente al ruolo “CED” tramite il pulsante “Insert activity without implementation”.



Metodi automatici

Non sono richiesti

Metodi con intervento utente

Non sono richiesti

Gestione attributi

Utilizzando l'interfaccia disponibile nel TAB “ActivityFormEditor” vengono impostati i permessi sugli attributi da gestire lato CMDBuild nell'attività corrente:

Nome	Modo	Note
DataTicket	read	Data di inserimento della richiesta, valorizzata in automatico sul database tramite l'impostazione del valore di default “now()”
NumTicket	read	Numero ticket, valorizzato in automatico sul database tramite la “sequence” appositamente predisposta
Richiedente	read	Utente che ha aperto il ticket
Description	read	Descrizione della richiesta
Valutazione	read	Approvazione o rifiuto richiesta
Motivazione	read	Motivazione della decisione
DataPrevistaChiusura	read	Data prevista chiusura intervento

NoteInizio	read	Note di inizio lavorazione
DataChiusura	write	Data effettiva chiusura intervento
TempoImpiegato	write	Tempo di lavorazione effettiva (in ore)
NoteFine	write	Note di inizio lavorazione

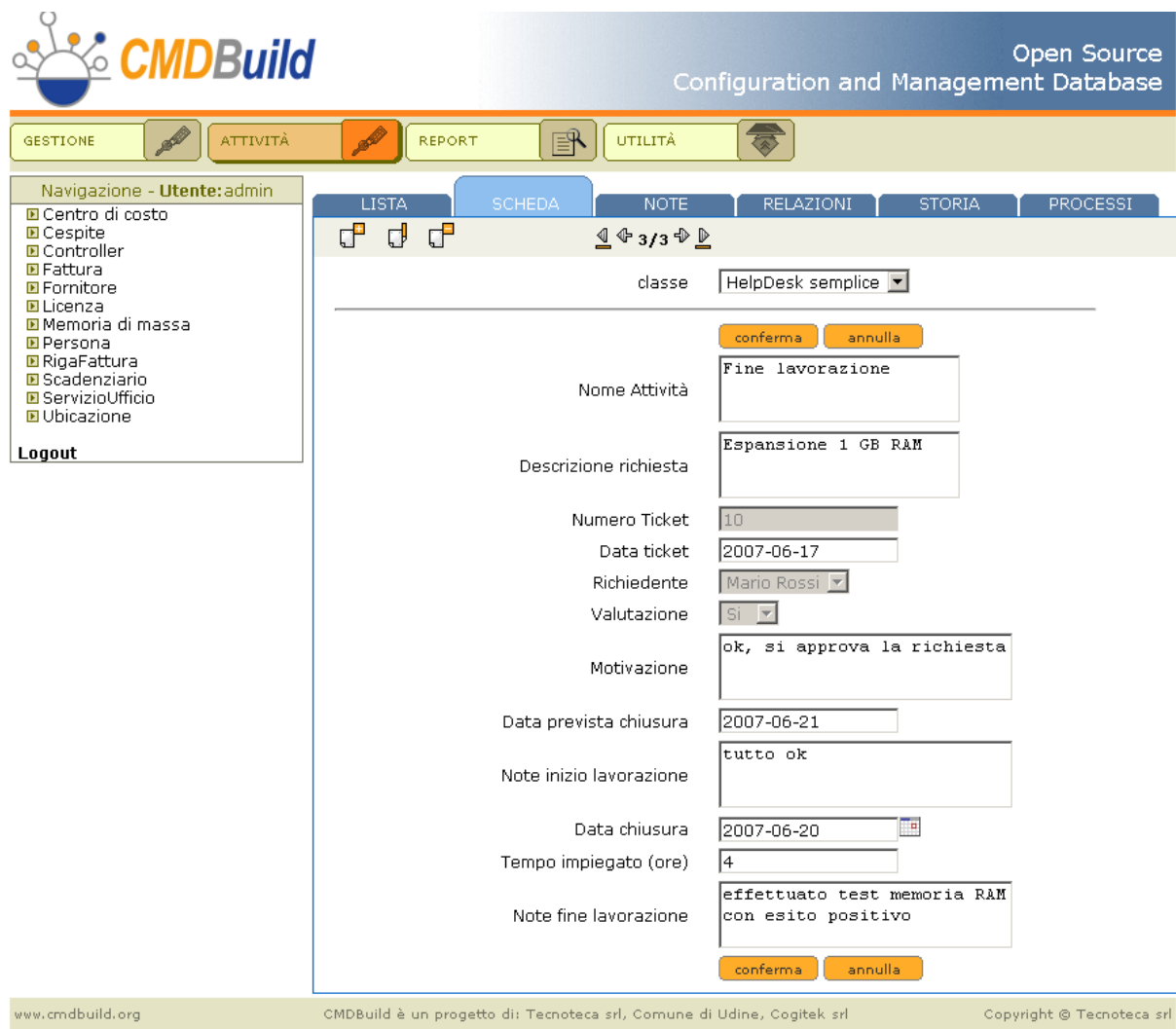
La form presentata in CMDBuild visualizzerà in sola lettura i primi otto attributi, richiedendo all'operatore di specificare la data effettiva di chiusura del ticket, l'impegno richiesto in ore ed eventuali note di fine lavorazione.

Effetto lato CMDBuild

Alla conferma dell'attività ("Valutazione richiesta"), avanzando nel processo verrà presentata nel TAB "Lista" della gestione attività una nuova riga corrispondente all'attività di fine lavorazione.

La nuova riga sarà visibile con link attivo solo per gli operatori abilitati al ruolo cui appartiene l'attività in questione (nell'esempio il ruolo "CED").

Le impostazioni effettuate in JPEd produrranno in CMDBuild, seguendo il link sull'attività corrente, l'apertura della seguente form:



The screenshot displays the CMDBuild web application interface. At the top, there is a navigation bar with tabs for 'GESTIONE', 'ATTIVITÀ', 'REPORT', and 'UTILITÀ'. Below this, a sidebar menu shows a tree structure of categories like 'Centro di costo', 'Cespite', 'Controllor', etc. The main content area is titled 'SCHEDA' and shows a form for the 'Fine lavorazione' activity. The form fields are as follows:

- Nome Attività: Fine lavorazione
- Descrizione richiesta: Espansione 1 GB RAM
- Numero Ticket: 10
- Data ticket: 2007-06-17
- Richiedente: Mario Rossi
- Valutazione: SI
- Motivazione: ok, si approva la richiesta
- Data prevista chiusura: 2007-06-21
- Note inizio lavorazione: tutto ok
- Data chiusura: 2007-06-20
- Tempo impiegato (ore): 4
- Note fine lavorazione: effettuato test memoria RAM con esito positivo

Buttons for 'conferma' and 'annulla' are present at the top and bottom of the form. The footer contains the website URL 'www.cmdbuild.org', project information, and copyright notice.

Fase e9) – Attività “Notifica richiesta chiusa”

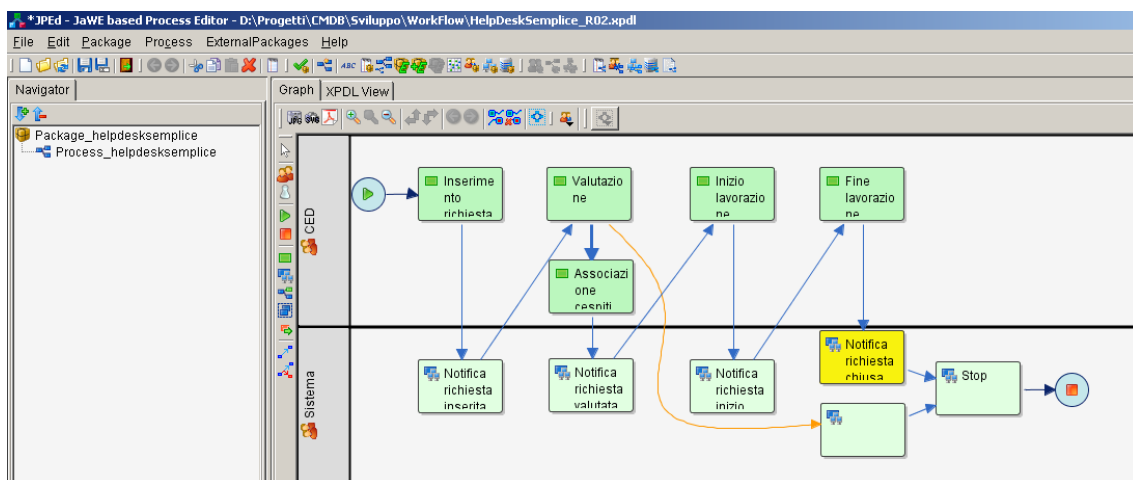
La nuova attività è un'attività automatica e ha lo scopo di:

- impostare il nuovo stato applicativo della richiesta (“Chiusa”)
- inviare al richiedente una notifica di conclusione positiva del processo

Operazioni lato JPED

L'attività viene inserita con il pulsante di JPED “Insert tool activity” (viene creata un'attività automatica) e segue l'attività “Fine lavorazione” cui deve essere collegata tramite il pulsante “Insert transition”.

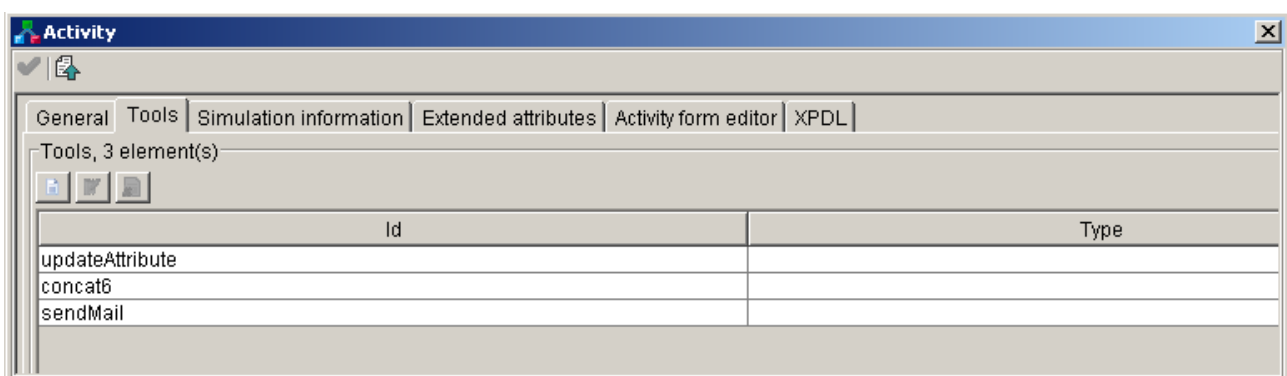
Il ruolo interessato è in questo caso quello fittizio di “Sistema”.



Metodi automatici

Viene richiamata l'esecuzione della seguente sequenza di metodi (vedi sintassi in Appendice):

- updateAttribute, per aggiornare lo stato applicativo dell'istanza di processo
- concat6, per concatenare le stringhe (nel numero di 6) che comporranno il testo della mail
- sendmail, per l'invio della mail di notifica all'utente che ha aperto il ticket



I dettagli dei metodi richiamati non vengono riproposti essendo del tutto simili a quelli già presentati nell'ambito dell'attività e2).

Metodi con intervento utente

Non sono richiesti

Gestione attributi

Non applicabile

Effetto lato CMDBuild

Non applicabile

Fase e10) – Attività “Notifica richiesta rifiutata”

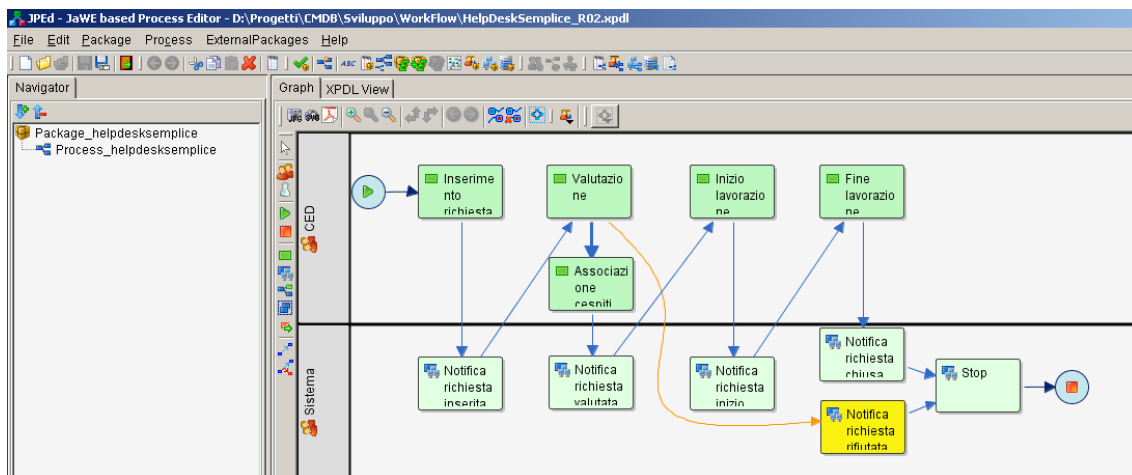
La nuova attività è un'attività automatica, viene eseguita nel caso la valutazione di cui all'attività e3) risulti negativa, ed ha lo scopo di:

- impostare il nuovo stato applicativo della richiesta (“Chiusa”)
- inviare al richiedente una notifica di conclusione negativa del processo

Operazioni lato JPEd

L'attività viene inserita con il pulsante di JPEd “Insert tool activity” (viene creata un'attività automatica) e segue l'attività “Valutazione richiesta” cui deve essere collegata tramite il pulsante “Insert transition”.

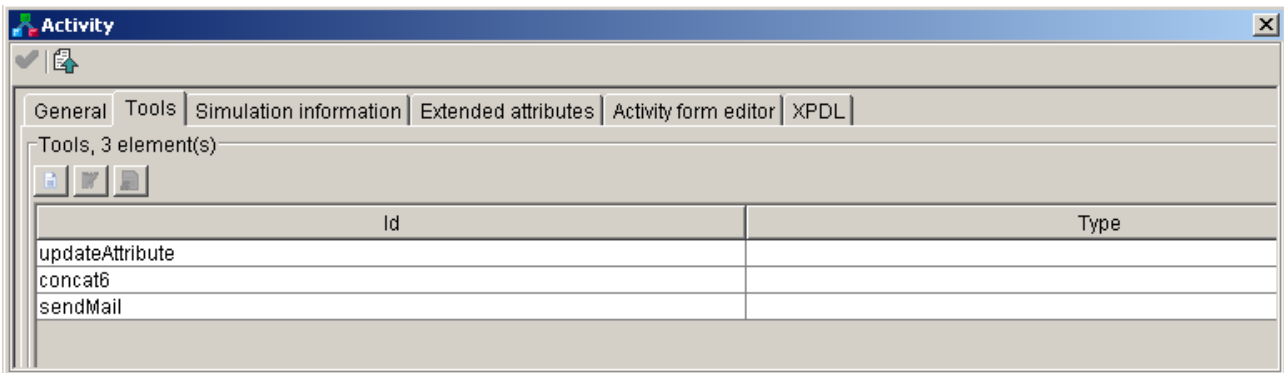
Il ruolo interessato è in questo caso quello fittizio di “Sistema”.



Metodi automatici

Viene richiamata l'esecuzione della seguente sequenza di metodi (vedi sintassi in Appendice):

- updateAttribute, per aggiornare lo stato applicativo dell'istanza di processo
- concat6, per concatenare le stringhe (nel numero di 6) che comporranno il testo della mail
- sendmail, per l'invio della mail di notifica all'utente che ha aperto il ticket



I dettagli dei metodi richiamati non vengono riproposti essendo del tutto simili a quelli già presentati nell'ambito dell'attività e2).

Metodi con intervento utente

Non sono richiesti

Gestione attributi

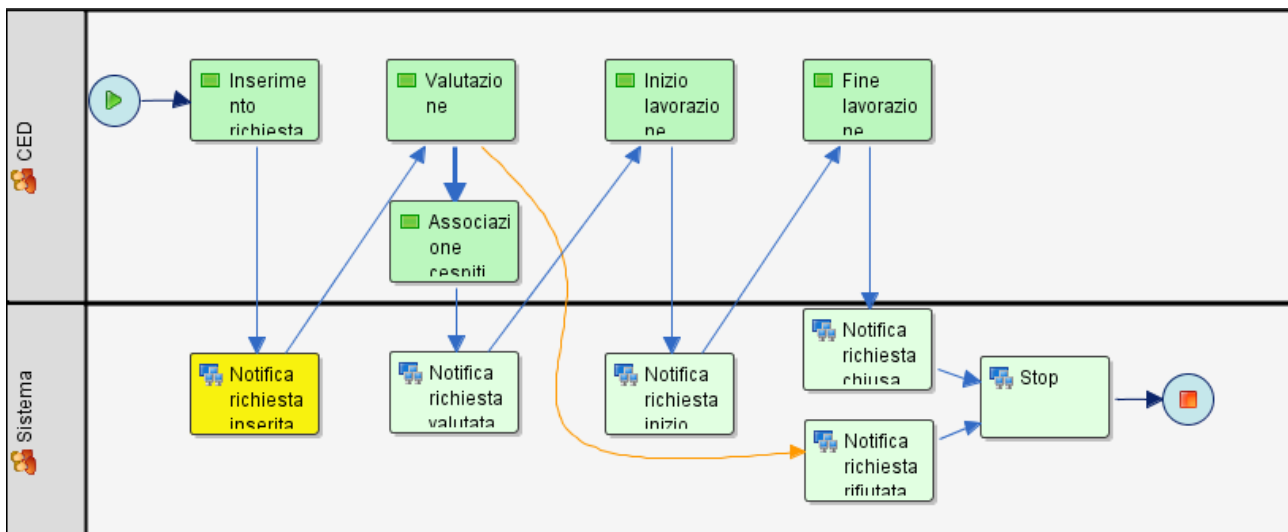
Non applicabile

Effetto lato CMDBuild

Non applicabile

Fase f) – Salvataggio definizione processo

Il nuovo processo è a questo punto completato:



Il processo creato va a questo punto salvato tramite il comando “File – Salva” (o l'equivalente pulsante sulla toolbar).

Il file viene salvato da JPEd in formato XPDL.

Fase g) – Salvataggio schema grafico processo

E' possibile salvare anche lo schema grafico del processo in formato immagine "jpeg".

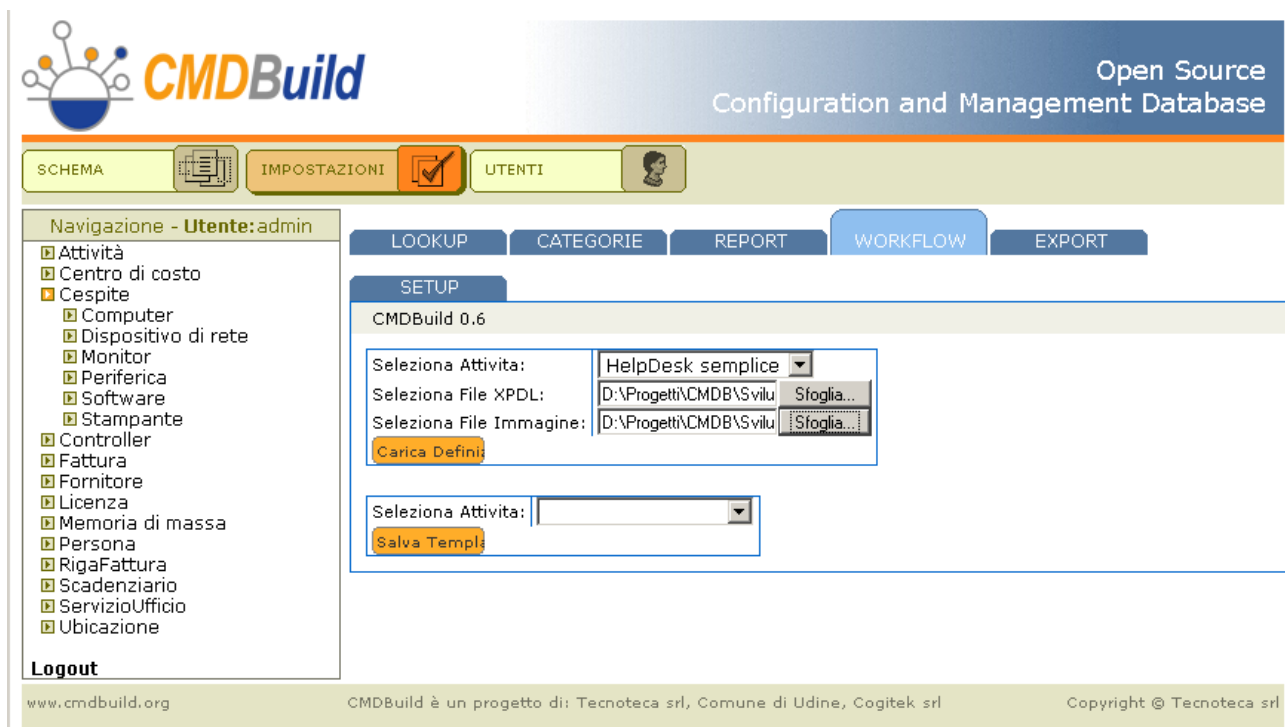
Tale immagine può essere poi caricata e visualizzata in CMDBuild come documentazione del processo in corso.

Fase h) – Caricamento file XPDL e schema grafico in CMDBuild

La definizione del nuovo processo deve essere ora caricato in CMDBuild tramite la funzione "Impostazioni – Workflow" disponibile nel Modulo Schema.

Tramite la stessa funzione è possibile caricare in CMDBuild anche lo schema grafico del processo esportato in formato immagine "jpeg".

Segue lo screenshot della funzione resa disponibile nel Modulo Schema:



Fase i) – Creazione istanza nuovo processo

E' possibile a questo punto avviare l'utilizzo fra gli operatori del nuovo processo generato, che è ora disponibile nel Modulo di Gestione di CMDBuild.

A tale scopo sarà sufficiente spostarsi nel TAB "Lista" all'interno del TAB di primo livello "Attività" e, avendone i permessi, avviare una prima istanza del nuovo processo tramite il pulsante "+" in alto a sinistra nello stesso TAB "Lista".

Appendice

Lista metodi API

Metodi automatici (implementati tramite “tools”)

L'interfaccia di definizione in JPEd prevede una lista di parametri formali ed una lista di parametri attuali, differenziati fra parametri di input e di output.

Gli eventuali argomenti di tipo “costante” devono essere racchiusi fra doppi apici.

Metodo	Parametri input	Parametri output	Note
createRelation	DomainName = <i>value</i> ClassName1 = <i>value</i> ClassName2 = <i>value</i> ObjId1 = <i>value</i> ObjId2 = <i>value</i>	<i>Done</i>	crea una relazione per il dominio e le classi / sottoclassi specificate fra le schede specificate; il metodo ritorna il valore booleano 'true' se termina correttamente
selectAttribute	ClassName = <i>value</i> AttributeName = <i>value</i> ObjId = <i>value</i>	<i>AttributeValue</i>	consente di recuperare il valore di un attributo presente nella scheda specificata; ritorna il valore dell'attributo richiesto
updateAttribute	ClassName = <i>value</i> AttributeName = <i>value</i> ObjId = <i>value</i> AttributeValue = <i>value</i>	<i>Done</i>	il metodo ritorna il valore booleano 'true' se termina correttamente
createCard	ClassName = <i>value</i> CardCode = <i>value</i> CardDescription = <i>value</i>	<i>CardID</i>	crea una scheda nella classe specificata, impostando gli attributi "Code" e "Description" con i valori passati gli eventuali ulteriori attributi vanno valorizzati con il metodo "updateAttribute" ritorna l'ID della scheda inserita
selectLookup	LookupType = <i>value</i> LookupID = <i>value</i>	<i>LookupDescription</i>	ritorna la descrizione decodificata del record della lista Lookup specificata e con ID specificato
sendMail	from_addresses = <i>value</i> to_addresses = <i>value</i> subject = <i>value</i> content = <i>value</i>		invia una mail alla lista di indirizzi specificata con il mittente, il soggetto ed il testo specificati
nextInt	Input_int = <i>value</i>		incrementa di uno la variabile passata in input (ed utilizzata poi anche in output)
int2string	InputInt	<i>OutputString</i>	converte un intero in stringa
date2string	InputDate	<i>OutputString</i>	converte una data in string

bool2string	InputBool	<i>OutputString</i>	converte un boolean in stringa
float2string	InputFloat	<i>OutputString</i>	converte un float in stringa
concat	InputString1 = <i>value</i> InputString2 = <i>value</i>	<i>OutputString</i>	concatena nella stringa di output le due stringhe passate in input
concat3	InputString1 = <i>value</i> InputString2 = <i>value</i> InputString3 = <i>value</i>	<i>OutputString</i>	concatena nella stringa di output le tre stringhe passate in input
concat4	InputString1 = <i>value</i> InputString2 = <i>value</i> InputString3 = <i>value</i> InputString4 = <i>value</i>	<i>OutputString</i>	concatena nella stringa di output le quattro stringhe passate in input
concat5	InputString1 = <i>value</i> InputString2 = <i>value</i> InputString3 = <i>value</i> InputString4 = <i>value</i> InputString5 = <i>value</i>	<i>OutputString</i>	concatena nella stringa di output le cinque stringhe passate in input
concat6	InputString1 = <i>value</i> InputString2 = <i>value</i> InputString3 = <i>value</i> InputString4 = <i>value</i> InputString5 = <i>value</i> InputString6 = <i>value</i>	<i>OutputString</i>	concatena nella stringa di output le sei stringhe passate in input

Il metodo “concat” è riportato in più versioni per disporre del numero di argomenti necessario (non è gestito nel formalismo XPDL l'overload dei metodi).

Metodi con interazione utente (implementati tramite “extended attributes”)

L'interfaccia di definizione in JPEd prevede una unica lista di parametri di input e di output, i primi seguiti dal simbolo “=” e dal valore del parametro, i secondi privi di tale estensione.

Gli eventuali argomenti di tipo “costante” devono essere racchiusi fra doppi apici.

Metodo	Parametri input	Parametri output	Note
createCard	ClassName = <i>value</i> Message = <i>value</i>	<i>value</i>	produce la visualizzazione nella form di gestione attività in CMDBuild di un link (con nome = “Message”) tramite cui creare una istanza della classe specificata; ritorna l'id della nuova istanza
modifyCard	ClassName = <i>value</i> ObjId = <i>value</i> Message = <i>value</i>	<i>ret</i>	produce la visualizzazione nella form di gestione attività in CMDBuild di un link (con nome = “Message”) tramite cui modificare una istanza

			della classe specificata; il metodo ritorna il valore booleano 'true' se termina correttamente
createRelation	DomainName = <i>value</i> ClassName = <i>value</i> ObjId = <i>value</i> Message = <i>value</i>	<i>ret</i>	produce la visualizzazione nella form di gestione attività in CMDBuild di un link (con nome = "Message") tramite cui creare una relazione per il dominio specificato e sulla scheda specificata;
quickAccept	Attribute = <i>value</i>		richiede per l'attività corrente la disponibilità dei pulsanti di approvazione / rifiuto veloce nel TAB "Lista" dell'interfaccia utente di CMDBuild
listClass	classes = <i>lista</i>	<i>selected</i>	visualizza nella form di gestione attività in CMDBuild una lista fra cui selezionare una classe; ritorna il nome della classe selezionata; la variabile "lista" è una stringa formattata nel seguente modo: DescrizioneClasse1=Classe1, DescrizioneClasse2=Classe2, ecc

Nuovi metodi potranno essere resi disponibili nei prossimi rilasci dell'applicazione.