

La piattaforma per soluzioni di Asset Management

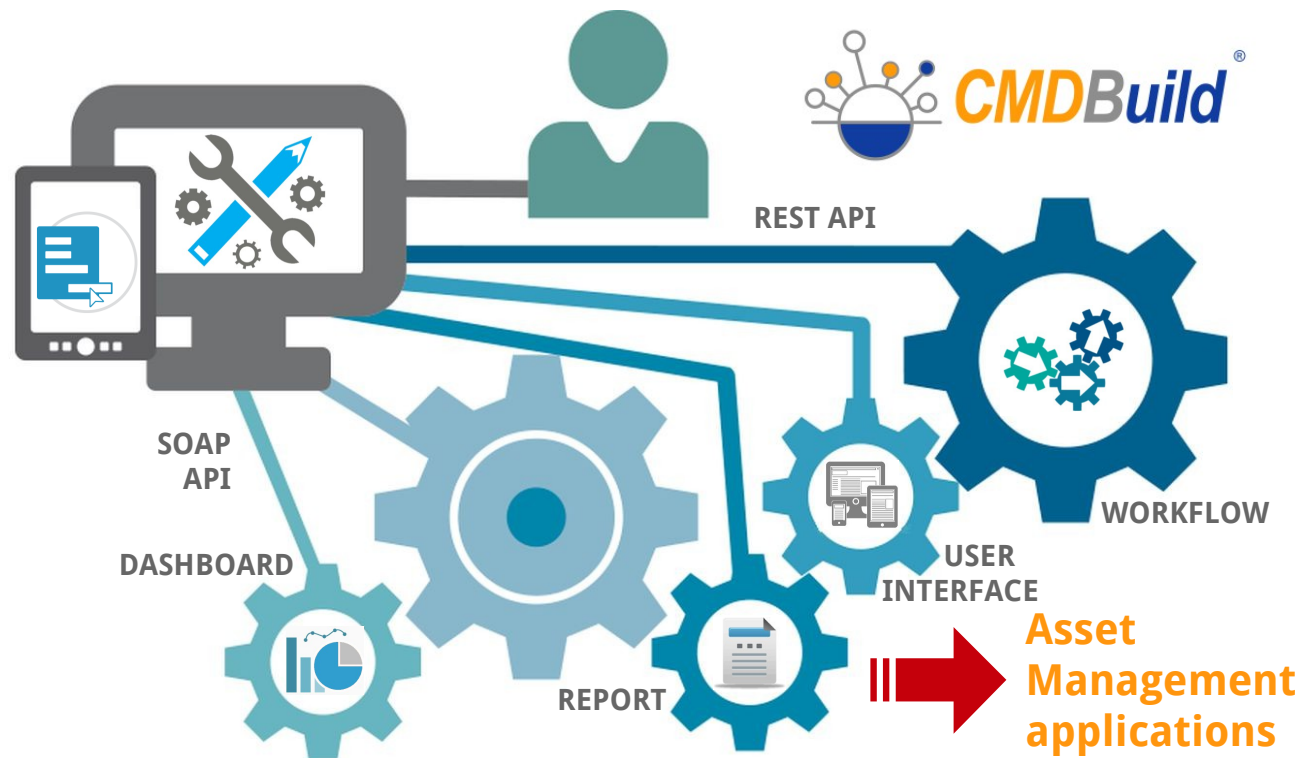


CMDBuild[®]

Cos'è CMDBuild

CMDBuild è una **piattaforma software** per la implementazione di applicazioni di Asset Management.

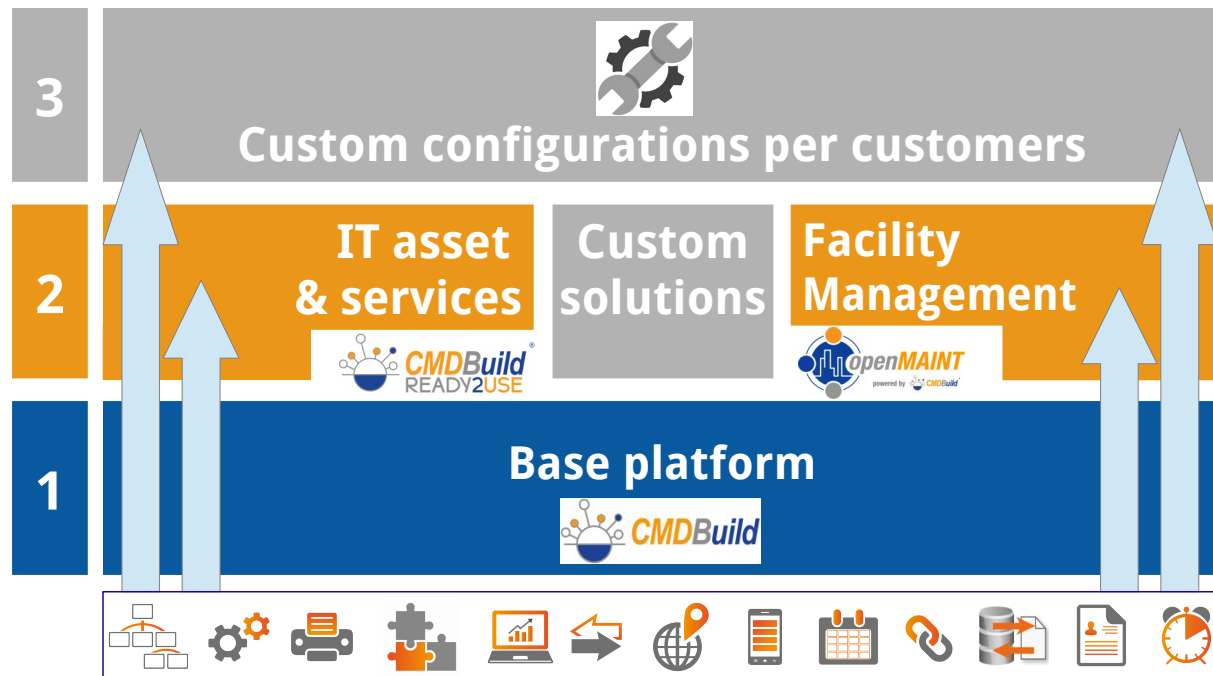
Mette a disposizione **meccanismi** per la configurazione di funzioni applicative.



Progettazione a tre livelli

E' il livello più basso di una progettazione a tre livelli:

- livello 1: **piattaforma** di base **CMDBuild**
- livello 2: **soluzioni applicative** verticali preconfigurate (**CMDBuild READY2USE** per l'IT, **openMAINT** per il Facility) oppure altre custom
- livello 3: configurazioni **personalizzate** per ciascun cliente

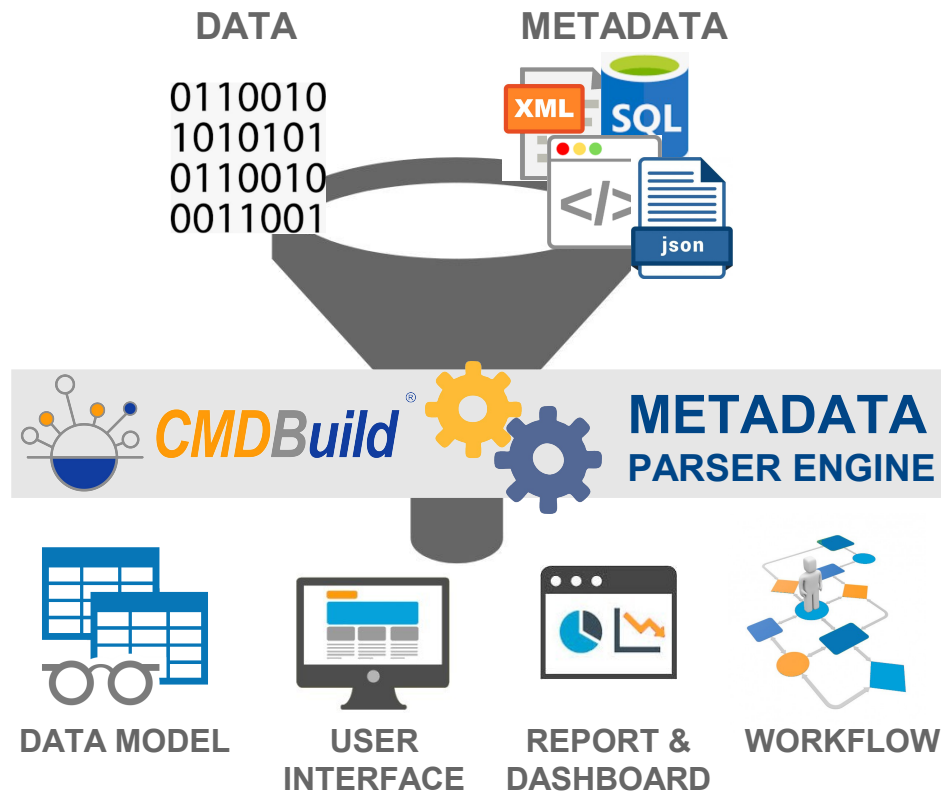


Dati e Metadati

Come può funzionare una architettura di questo tipo ?

Separando il "motore" dalle "logiche di business", e gestendo le implementazioni di livello 2 e di livello 3 tramite servizi descritti da **metadati**:

- parametri semplici
- descrittori XML
- descrittori JSON
- strutture SQL

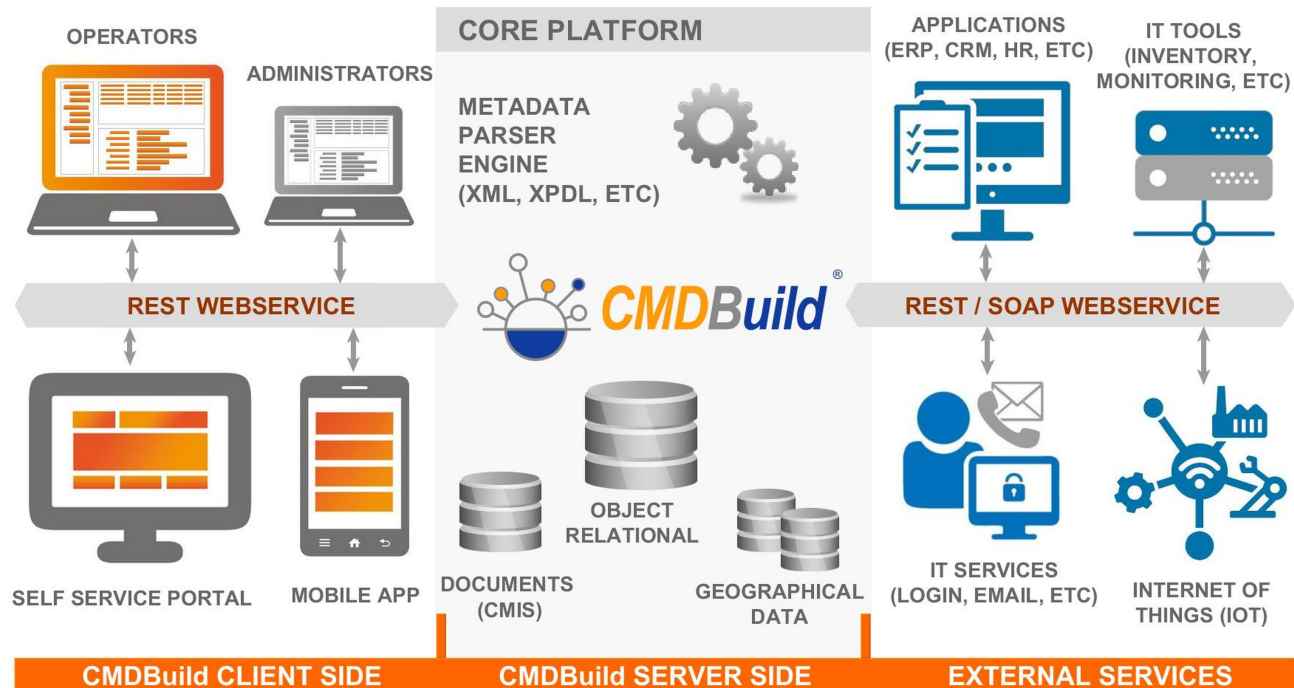


Interoperabilità

CMDBuild è una piattaforma per l'**interoperabilità**:

- componenti server
- interfacce client
- collaborazione con servizi esterni

Utilizza **protocolli** di comunicazione **standard e aperti**.



Meccanismi disponibili

Meccanismi principali (GUI e metadati)



Disegno del
modello dati



Editor e motore
di workflow



Editor e motore
di report



Configurazione
di dashboard



Configurazione di
logiche e GUI custom

Funzionalità native della piattaforma



Paradigma
"a relazioni"



Archivio
documentale



Storicizzazione
dei dati



Profilazione
utenti e sicurezza



Gestione Email
(input e output)



Grafo visuale per
analisi di impatto



Georiferimenti
GIS e BIM



Scheduler per
task automatici



Import / Export
da GUI e batch



Gestione
scadenziario

Opzioni di utilizzo



Diversi protocolli
di autenticazione



Multitenant



Configurazione
cluster



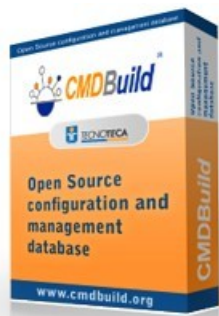
Appliance
preconfigurata



Servizio
SaaS

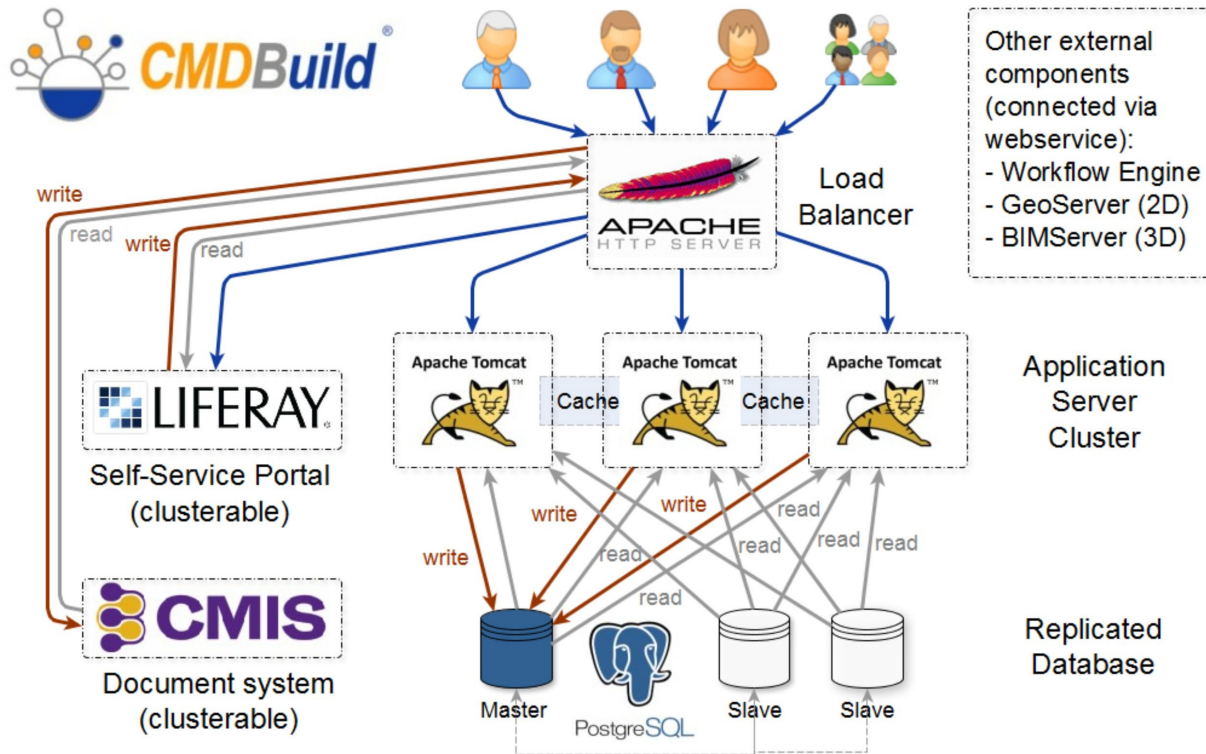
Caratteristiche tecniche - Componenti

Tutti i componenti del sistema **CMDBuild**, sono **open source** e basati sullo stack **Java**



Caratteristiche tecniche - Cluster

Configurazione
in **cluster**
per garantire
la **continuità operativa**
e la **scalabilità**
del sistema

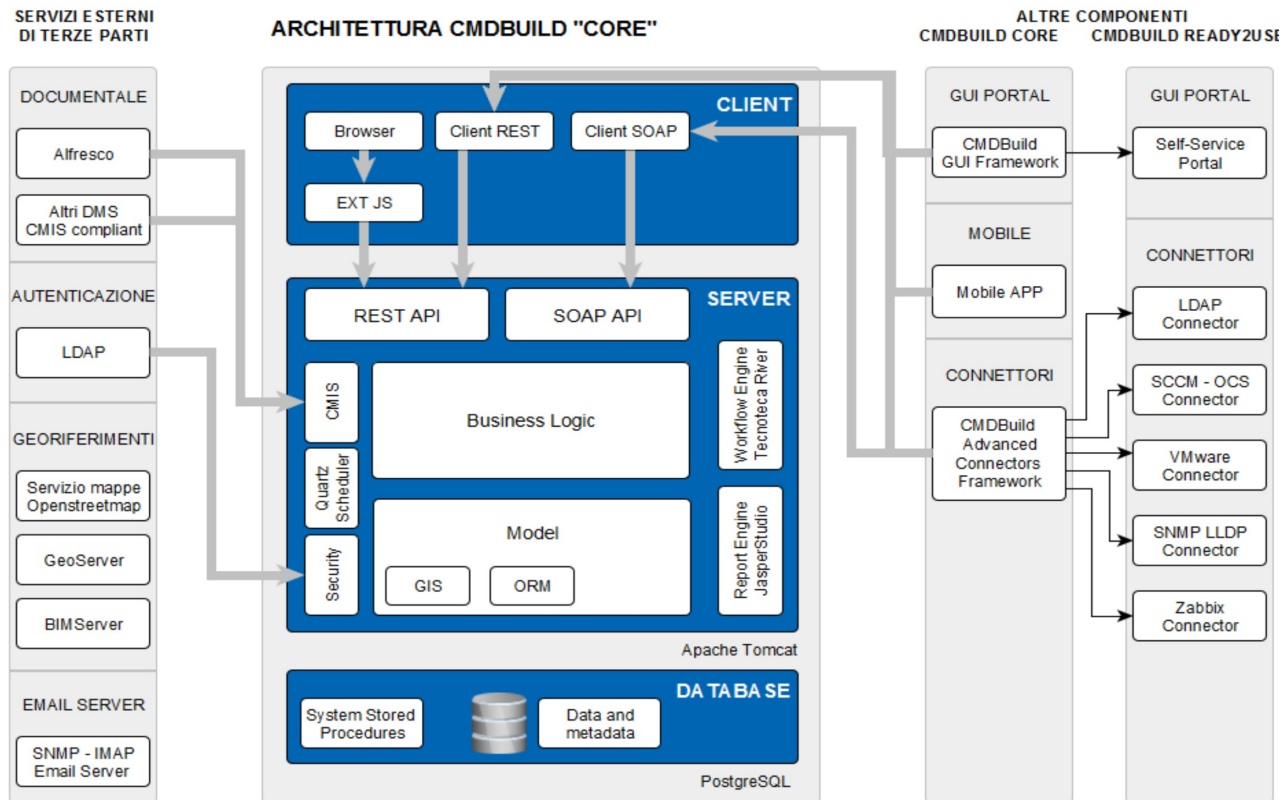


Caratteristiche tecniche - Architettura

Architettura **three-tier**
(frontend, backend, DB)

Accesso a **servizi di terzi**

Framework e **strumenti esterni** specializzati



Caratteristiche tecniche - Processo di sviluppo

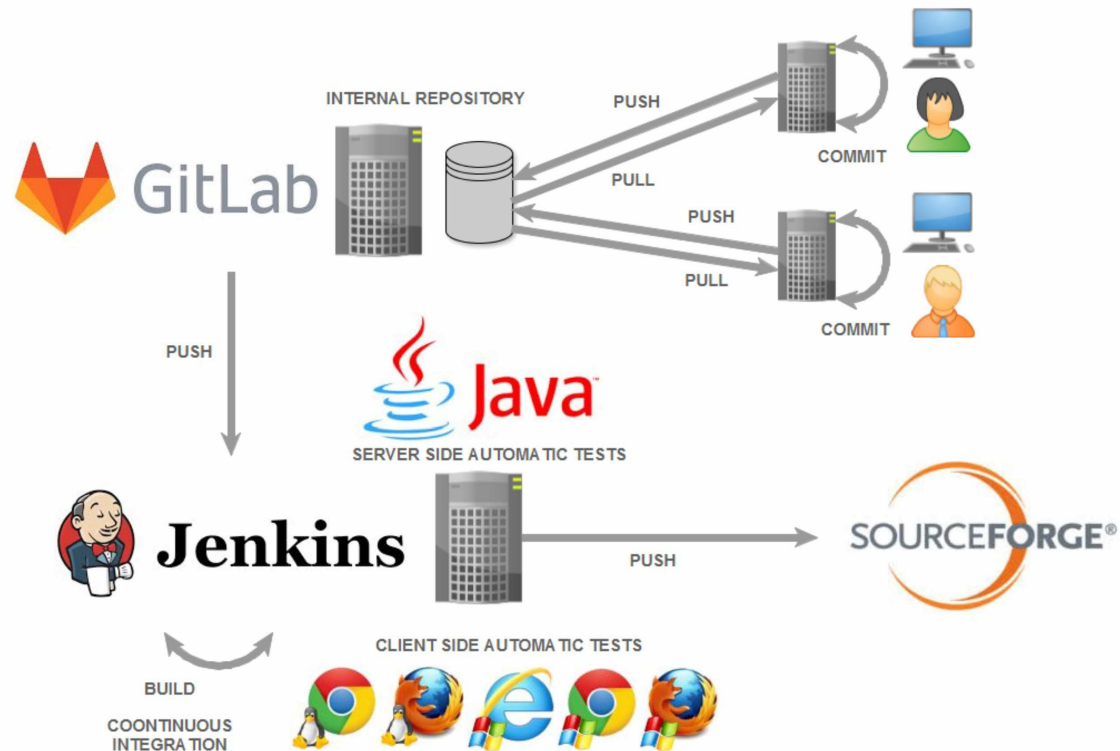
Gestione del progetto con tecnologie "Agile".

Gitlab come repository del codice sorgente e gestione degli issue.

Jenkins per il merge progressivo dei "commit" degli sviluppatori (continuous integration).

JUnit e per il test automatico lato server, **Selenium** per il test automatico lato client.

Sourceforge come repository pubblico del codice open source.



Alcuni numeri sul progetto



2006

Anno primo rilascio



52

Rilasci "core"



48

Anni uomo "core"



32

Localizzazioni
primo livello



30.000

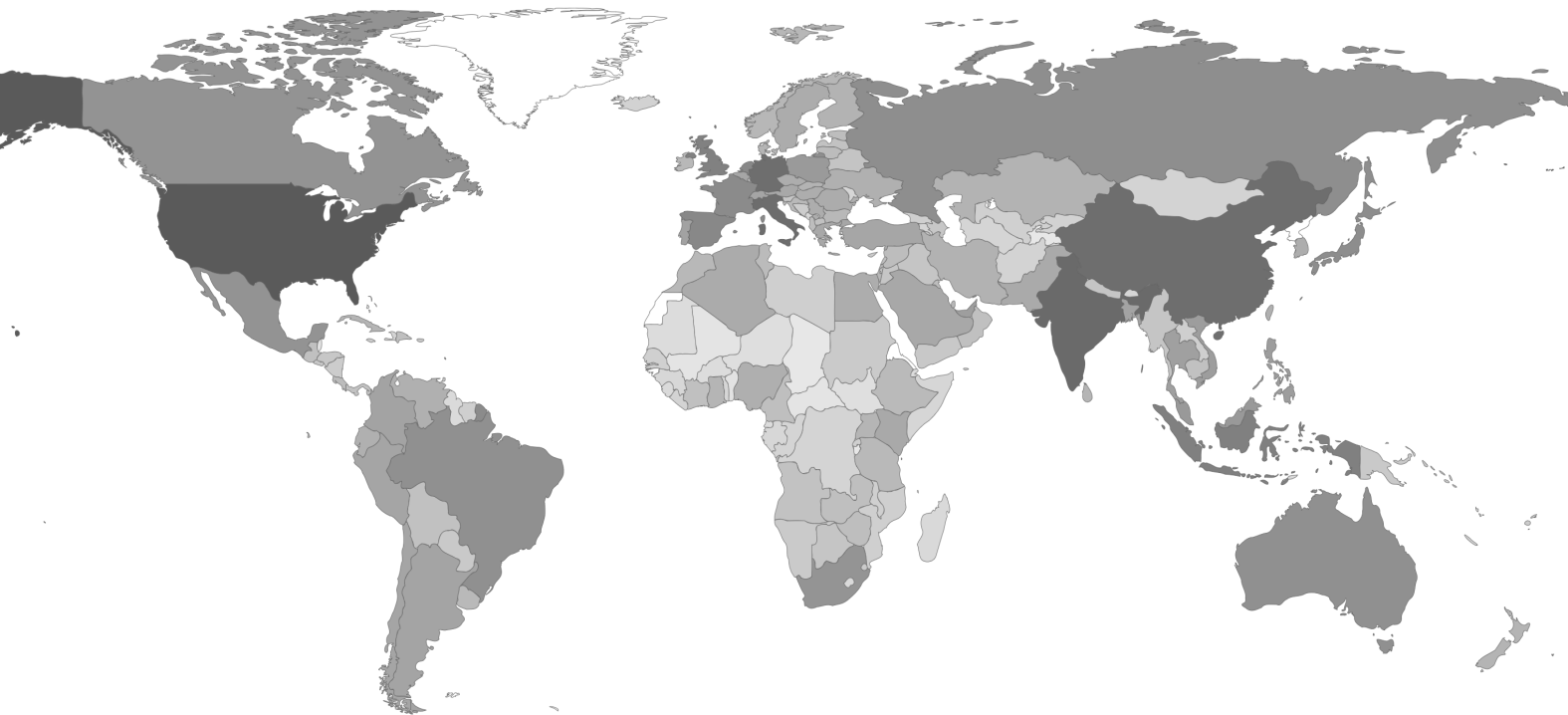
Download annui



6

CMDBuild Day
(con 76 case history)

Download da SourceForge



Country	Total
1. United States	18,803
2. India	11,140
3. Italy	10,063
4. China	10,051
5. Germany	9,655
6. Indonesia	5,091
7. United Kingdom	4,726
8. Spain	3,653
9. Netherlands	3,617
10. France	3,150

