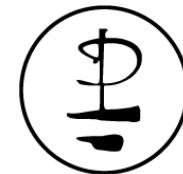


Internet Of Things:
Eclipse SmartHome con OpenHAB

Giuseppe Salinaro

Bio

- Student @ Department of Computer Science - University of Bari "Aldo Moro"
- Utente GNU/Linux sin dal 2006
- Membro del team di traduzione della openSUSE 11.0 (2008)
- **Il Software Libero nella Pubblica Amministrazione** (2010)



L'ascesa delle macchine: che cosa significa la Internet of Things per il business

Un report recente ha evidenziato 3 punti chiave circa l'ascesa dell' Internet of Things.

- 1) Il mercato per IoT crescerà 347% nei prossimi quattro anni.
- 2) la sua crescita eclisserà l'uso "umano" di internet prima della fine del decennio
- 3) I veri vincitori IoT potrebbero non essere le tipiche compagnie tecnologiche.



L'ascesa delle macchine

Servizi igienici, cucine, televisori, dispositivi mobili, automobili, pompe di benzina, pacchetti, frigoriferi, distributori automatici, sistemi SCADA: saranno presto strumenti di comunicazione vitali contribuendo a rendere la nostra vita più facile / più sana / più produttiva.

L'IoT non è solo la raccolta dei dati; è anche di analisi e l'utilizzo di tali dati.

Eclipse IoT



Eclipse IoT è un ecosistema di aziende e privati che stanno lavorando insieme per creare una Internet delle cose basata su tecnologie aperte.

Eclipse IoT

CANONICAL

GENERATIVE
SOFTWARE



openHAB
FOUNDATION

m2mTM
alliance

IBM



bitreactive





Eclipse IoT

La comunità è composta da più di 200 collaboratori attivi che lavorano su 24 diversi progetti. Questi progetti sono costituiti da oltre 2 milioni di linee di codice e sono stati scaricati oltre 500.000 volte.

La Eclipse IoT Working Group comprende 30 aziende associate che collaborano per fornire elementi di software in forma di implementazioni open source degli standard, i servizi e le strutture al fine di creare una Open Internet of Things.

Eclipse IoT

La stragrande maggioranza delle soluzioni IoT di oggi sono stati progettati in modo ad hoc.

A seconda del dominio aziendale e della piattaforma di destinazione (OS, H/W, ...) vengono implementate architetture molto diverse e spesso incompatibili tra loro

Eclipse IoT fornisce gli elementi costruttivi basati su standard e protocolli aperti e forniscono anche servizi e framework aggiuntivi per la gestione dei dispositivi, la comunicazione wired/wireless, soluzioni verticali come la domotica,..

Eclipse IoT

Costruire soluzioni interoperabili IoT è una vera sfida. Da sensori e attuatori sul campo di sistemi back-end, ci sono molti aspetti di soluzioni end-to-end in cui è importante fare affidamento su standard:

- Protocolli usati per implementare le comunicazioni device-to-device o device-to-server,
- Protocolli di gestione di dispositivi per consentire il controllo dei dispositivi IoT e dei gateways,
- Gateways e interfacce Servers .

Mentre Open Standards sono fondamentali, ritengono che sia importante anche per rendere disponibili implementazioni open-source di tali standard, per incoraggiare l'adozione di tali standard sia da parte di sviluppatori IoT e sia dalle industrie IoT.

Eclipse IoT

Il gruppo fornisce le implementazioni open source dei protocolli dell'IoT - COAP, ETSI SmartM2M, MQTT e LwM2M - ed esegue servizi estensibili e strutture per favorire lo sviluppo delle applicazioni dell'IoT con APIs. I 4 progetti includono:

- Eclipse Kura , un framework costruire IoT gateways.
- Eclipse Paho, una client-based implementazione dei protocolli di messaging MQTT e MQTT-SN.
- Eclipse SmartHome, un framework per la creazione di soluzioni intelligenti per la casa concentrandosi su ambienti eterogenei, progettate per i dispositivi embedded come Raspberry Pi, BeagleBone nero e Intel Edison.
- Eclipse OM2M, che implementa lo standard oneM2M, che fornisce servizi IoT orizzontali per favorire lo sviluppo di soluzioni IoT che sono indipendenti dalle reti sottostanti.

Eclipse SmartHome



Il progetto SmartHome è un framework che permette la creazione di soluzioni di casa intelligente che hanno una forte attenzione su ambienti eterogenei, cioè soluzioni che si occupano dell'integrazione di diversi protocolli o standard. Il suo scopo è quello di fornire un accesso uniforme ai dispositivi e informazioni e per facilitare diversi tipi di interazioni con loro. Tale framework è costituito da un insieme di bundle OSGi che possono essere distribuiti su un runtime OSGi e che definisce i servizi OSGi come estensioni.

OpenHAB



Kai Kreuzer, che ha iniziato openHAB nel 2010, sta conducendo il progetto, ed ha contribuito il nucleo di openHAB per creare il framework Eclipse SmartHome.

openHAB stessa ha già un certo numero di dispositivi e tecnologie che possono essere integrate, dando solo un esempio di ciò che è possibile.

openHAB 2.0 è stato creato utilizzando SmartHome framework.

OpenHAB

- Insteon e vecchi dispositivi X-10.
- Nest termostato, è possibile ottenere informazioni dal Nest e utilizzarlo per configurare tapparelle, ventilatori da soffitto, o altri dispositivi HVAC.
- È anche possibile aggiungere un contatto a una finestra (che funge anche da un contatto di sicurezza) e quando la finestra è aperta, impostare il Nest in modalità "Away".
- Philips HUE, Wemo, Plex, Z-Wave
- è tutto modulare. Puoi configurare le associazioni ed integrazioni che si desidera, e quindi si può avere eventi di innesco su un oggetto verso un'altro.

OpenHAB 2 vs OpenHAB1

- Il ramo 1.x di openHAB - il fuoco che è sulle binding e altri add-on.
- openHAB 2 ha un diverso obiettivo: il comfort per l'utente. Mentre in openHAB 1.x è necessario configurare tutto in file di testo e capire la sintassi corretta e le possibilità di un certo binding nella wiki, openHAB 2 permette binding e altri add-on di autodescrivere la loro configurazione, in modo che sia possibile offrire interfacce utente per l'installazione e la configurazione del sistema. Inoltre, il rilevamento automatico (ad esempio tramite UPnP, AllJoyn, ecc) è offerto, in modo che i nuovi dispositivi possono essere aggiunti con un semplice click di un pulsante.

OpenHAB 2 vs OpenHAB 1

- Un secondo obiettivo di progettazione importante della openHAB 2.x è l'ottimizzazione per piattaforme embedded.
- Il runtime di base e le sue API cambiano radicalmente.
- Di conseguenza, il repository dei sorgenti di openHAB 2 avrà inizio senza binding in modo da incoraggiare gli sviluppatori della comunità a migrare le loro associazioni di 1.x esistenti per i concetti 2.x, una volta che questi sono completamente a posto.
- Per il momento, vi è un "pacchetto di compatibilità 1.x" che permette di utilizzare openHAB 1.x componenti aggiuntivi con il runtime openHAB 2 - questo non funzionerà per tutti appena installati, ma per la maggior parte di essi

OpenHAB 2

- Il file di configurazione centrale "openhbab.cfg" è andato. Invece, ora è possibile avere un file separato per ogni componente aggiuntivo. Questo migliora chiaramente la panoramica sopra i parametri di configurazione del file di configurazione non verrà riempita di informazioni sui componenti aggiuntivi che non si sta utilizzando.
- L'intera struttura della directory è stata revisionata - ora ci sono tre cartelle principali.
- Questa nuova struttura rende gli aggiornamenti di sistema molto più semplici e facilita anche l'installazione su sistemi embedded, dove non si vuole utilizzare la memoria flash per la scrittura di file di log continuo etc.

OpenHAB 2

Le cartelle principali:

"Runtime" che contiene i file binari e altri contenuti che è necessario per far funzionare il sistema.

"Conf", che contiene tutte le configurazioni e le personalizzazioni personali (come ad esempio le icone personalizzate).

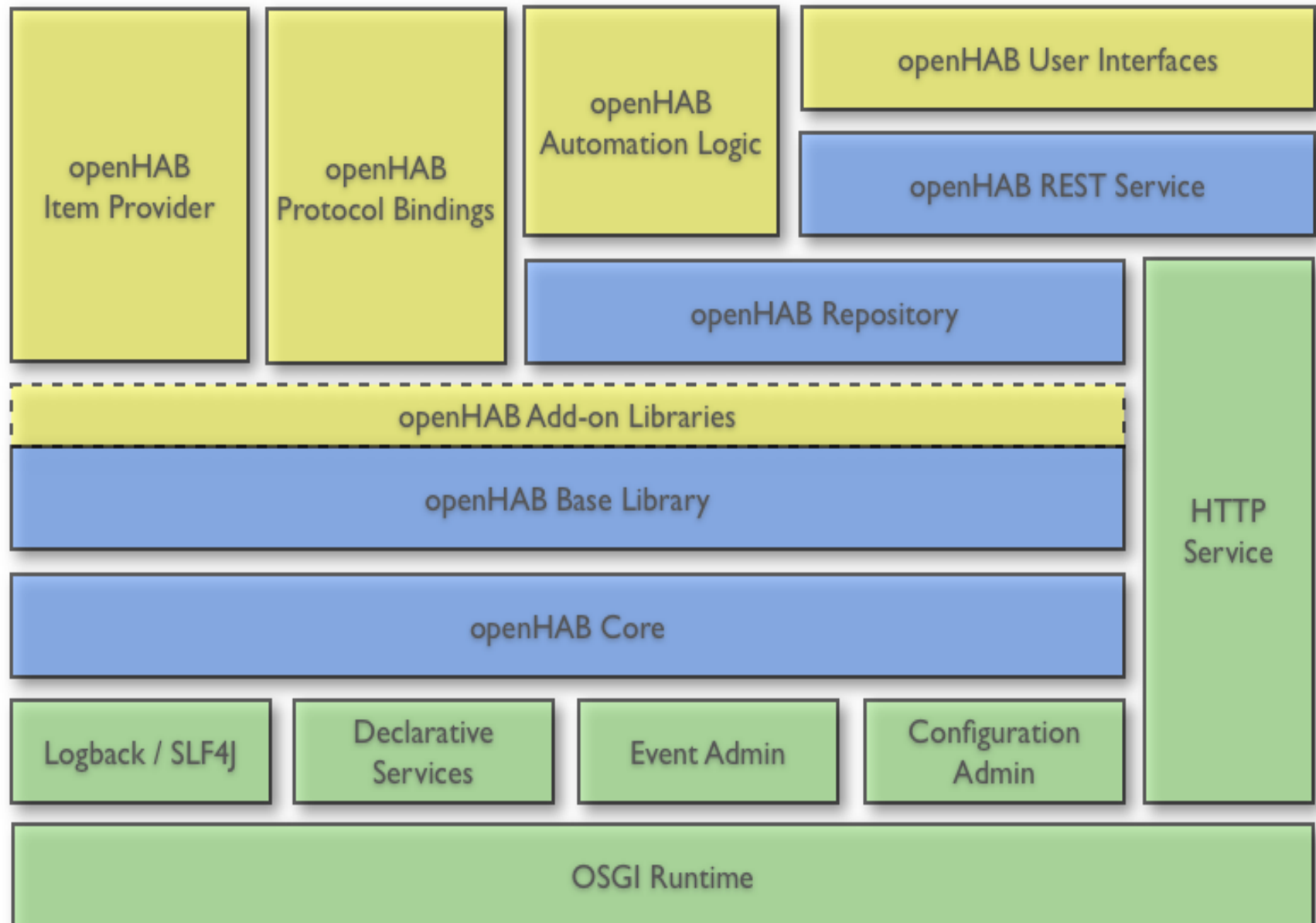
"Userdata", che è l'unica directory che il sistema scrive attivamente (file, database, ecc log).

OpenHAB

- Un concetto fondamentale per openHAB è la nozione di un "Item". Un item(elemento) è un elemento atomico funzionale incentrato sui dati - si può pensare ad esso come ad una "capacità".
- Tutte le funzionalità offerte da openHAB usano questa astrazione "item", che significa che non si troverà alcun riferimento a cose specifiche dei dispositivi (come gli indirizzi IP, ID, ecc) nelle regole di automazione, le definizioni di interfaccia utente e così via.
- Un aspetto molto importante di architettura di openHAB è il suo design modulare. E 'molto facile aggiungere nuove funzionalità (come l'integrazione con un altro sistema attraverso una "binding") ed è possibile aggiungere e rimuovere tali caratteristiche in fase di esecuzione.

openHAB Architecture Overview

- openHAB Add-ons
- openHAB Core Components
- OSGi Framework



Sitemap

openHAB viene fornito con una configurazione testuale generica per le sue interfacce utente: Il cosiddetto Sitemap. La mappa del sito è una struttura ad albero di widget, che definiscono le diverse pagine di un utente e il loro contenuto. I widget possono essere associati agli item(elementi), per essi dovrebbero mostrare lo stato e/o gli elementi di controllo.

DEMO INSTALLAZIONE

OpenHAB

openHAB si presenta come un file zip indipendente dalla piattaforma, è solo necessario estrarlo in una certa cartella.

Troverete le seguenti cartelle:

- Conf: Questa contiene tutti i file di configurazione utente specifico.
- Runtime: contiene i binari openHAB, di norma devono sussistere alcun bisogno di toccare nulla qui dentro - l'intera cartella può essere considerato di sola lettura.
- Userdata: Qui troverete tutti i dati che si genera durante il runtime: i file di log, file di database, ecc In teoria, questo dovrebbe essere l'unica cartella in cui openHAB ha bisogno del permesso di scrittura.
- Addons: Qui si può inserire componenti aggiuntivi (o qualsiasi altro bundle OSGi) che si desidera distribuire nella propria istanza. Questi possono essere add-on per openHAB 1.x e 2.x indistintamente.

OpenHAB

Il pacchetto demo è costituito da file di configurazione di esempio e campioni di add-on e interfacce utente. Per installare la demo, è necessario modificare il file 'conf/services/addons.cfg'. Rimuovere il commento dalla linea `package` e impostarlo su `demo` come illustrato di seguito:

```
# The base installation package of this openHAB instance (default is "standard")
# Valid options:
# - minimal : Installation only with dashboard, but no UIs or other addons
# - standard : Typical installation with all standard UIs
# - demo : A demo setup which includes UIs, a few bindings, config files etc.
package = demo
...

```

OpenHAB

Avvio del runtime

Una volta configurato il runtime come sopra, avviare il runtime openHAB dal terminale chiamando `./start.sh` (`start.bat` su Windows)

Puntare il browser su `http:// <nome host>: 8080` (consenti al runtime po 'di tempo per iniziare prima che il server HTTP è disponibile, soprattutto sul primo avvio) e si sarà accolto dal openHAB Dashboard, che è il punto di ingresso alle diverse interfacce web

OpenHAB

Usa della shell

openHAB usa Apache Karaf ed è dotato di una potentissima shell per gestire la installazione. Esempio di comandi utili:

log:tail: Mostra l'output del live logging, finisce premendo ctrl+c.

log:set DEBUG org.openhab.binding.sonos: Attiva il debug logging per un certo binding.

feature:list: elenca tutte le funzioni disponibili e mostra il loro stato.

feature:install openhab-binding-knx: installa una certa add-on (KNX).

bundle:list -s: elenca tutti i bundle installati con il loro nome simbolico.

logout: chiude openHAB.



HABmin

HABmin è un'interfaccia utente moderna, professionale e portatile per openHAB, fornisce sia le funzioni amministrative e sia quelle utente (ad esempio sitemap per gli utenti, e l'utility di configurazione per facilitare l'installazione).

HABmin

Caratteristiche:

Responsive. Dovrebbe funzionare bene su tutti i dispositivi. Naturalmente alcune funzioni possono essere rimossi o essere difficili da utilizzare su dispositivi di piccole dimensioni (ad esempio, l'editor regola grafica).

Theme-able. temi multipli sono disponibili a vostra scelta (attualmente 3 temi).

Charting. Moderno, grafici veloci dei dati storici.

Graphical rule editor. Nessuna necessità di imparare le regole sintattiche..

International support.

Disponibile come applicazione nativa per **Android**.

openHABian

- Hassle-free openHAB 2 Raspbian è un'immagine di installazione automatica minima “netinstall” per il Raspberry Pi..
- L'immagine fornita di soli 64 MB contiene un sistema di avvio minimo. Questo sistema sarà quindi possibile installare Raspbian seguito da openHAB e una serie di strumenti utili. Tutti i pacchetti verranno scaricati nella loro versione più recente.

openHABian

- openHAB 2 latest snapshot (package repository)
- Oracle Java 8 (build 1.8.0_101, needed for my.openhab)
- Samba (preconfigured)
- custom .bashrc and .vimrc files
- openHAB syntax highlighting in vim and nano
- uses whole SD card by default (8GB or 16GB SD card sufficient)
- 16MB GPU memory split
- git based versioning of etc by the help of etckeeper
- useful packages like screen, mc, htop ...

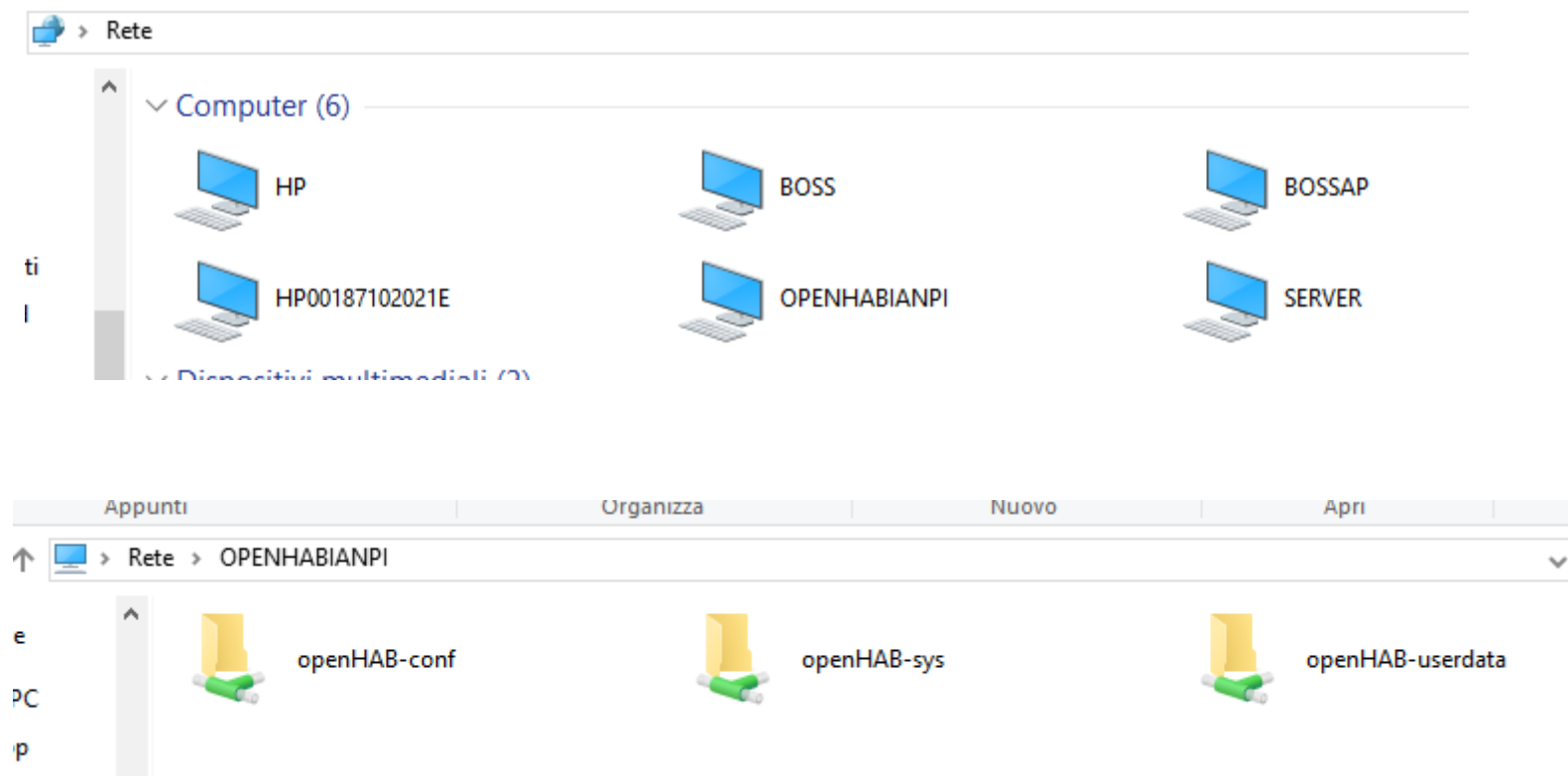
openHABian

Setup

- Connect Ethernet, SD card and power to your Raspberry Pi
- Wait up to 45 minutes (setup depends on your downlink as almost everything is downloaded live)
- Green LED will indicate when setup is finished
 - Irregular blinking: setup in progress...
 - Steady "heartbeat": setup successful
 - Fast blinking: error while setup, check `/var/log/raspbian-ua-netinst.log`, create GitHub Issue
- Connect to the openHAB 2 portal (available after another 15 minutes): `http://openhabianpi:8080`
- Connect via ssh with `pi:raspberrypi`
- Connect to the Samba network share with `openhab:habopen`

openHABian

SMB (samba)



“È chiaro che altri problemi come il fondamentalismo religioso, la sovrappopolazione, i danni all'ambiente e il dominio del business su governi, scienza, pensiero e società sono molto più grandi del software non libero. Ma molta altra gente ci sta già lavorando, e io non ho nessuna grande capacità o idee per indirizzarle. Così sembra che sia meglio che io continui a lavorare sul problema del software libero. Ad ogni modo, il software libero contrasta uno degli aspetti del dominio dell'economia sulla società.”

Richard Matthew Stallman(FSF)



salinaro@ld16bari :-> umount /dev/talk