



Powering Dynamic M2M Event Processing with OSGi

Dynamic Complex Event Processing and OSGi

6/12/2014

Hitachi Communication Technologies America, Inc.

Walt Bowers

Chief Architect OSGi Solutions

**Human Dreams.
Make IT Real.**

Powering Dynamic M2M Event Processing with OSGi **Dynamic Complex Event Processing and OSGi.**

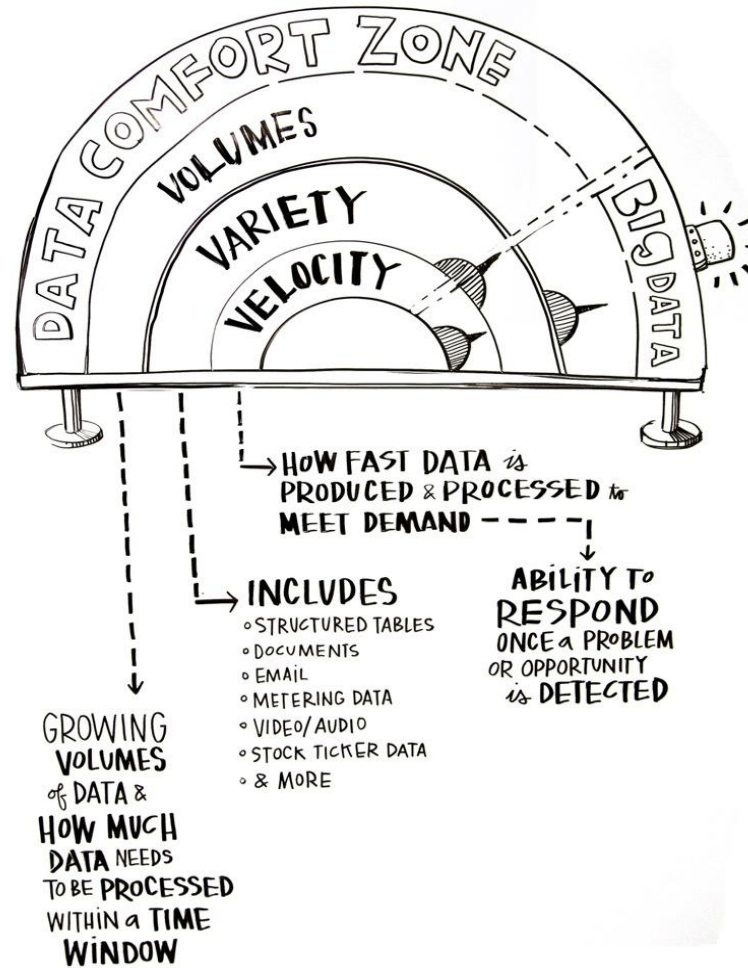
Contents

1. The Vs of Big Data
2. Complex Event Processing
3. Dynamic Complex Event Processing
4. Demo

Powering Dynamic M2M Event Processing with OSGi

Dynamic Complex Event Processing and OSGi.

1. The Vs of Big Data



Extracting **VALUE** from **VIABLE** Data
WHERE It Matters and *WHEN* It Matters



Rise of The Intelligent Device



1

Intelligent Devices

Always-on devices connected to variety of sensors and running multiple software applications

2

Real-Time Analytics

High-frequency data analysis for instant decision making and automation of information flows

3

Big Data

Integration of data from connected devices with enterprise applications and historical data

- Local analytics and business rules are controlled by global analytics
 - In-flight data analytics on the device
 - Near real time response on the device
- Global Analytics for the Big Patterns
 - Big Data post processing
 - Discover Hidden Patterns/dependencies
- Dynamically Adjust the Rules
 - Update new rules to the local device
 - Enhances the devices local analytics
- Rinse and Repeat

Transportation Example

1 Location tracking (LBS)



2 Route monitoring (Time of day optimization)



3 Passenger Counter



4 Fraud detection (Passenger counter, camera, fare collection)

7



Digital signage (Information and advertising - Location aware)

6 Vehicle diagnostics (Sensors throughout vehicle)

5 Camera (Normal operation, local storage; emergencies such as Amber alert, streaming to emergency authorities)

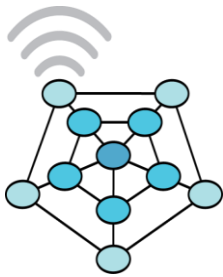


Powering Dynamic M2M Event Processing with OSGi

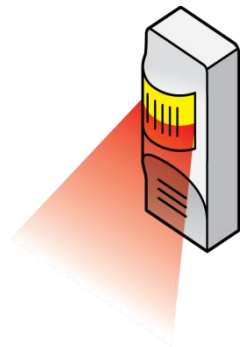
Dynamic Complex Event Processing and OSGi.

2. Complex Event Processing

Complex Events Processing
enables real-time business insights from edge devices



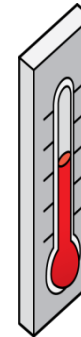
**Communication
Events**



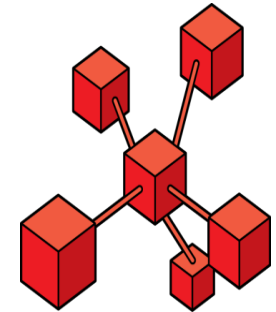
**Machine
Events**



**Security
Events**



**Environmental
Events**

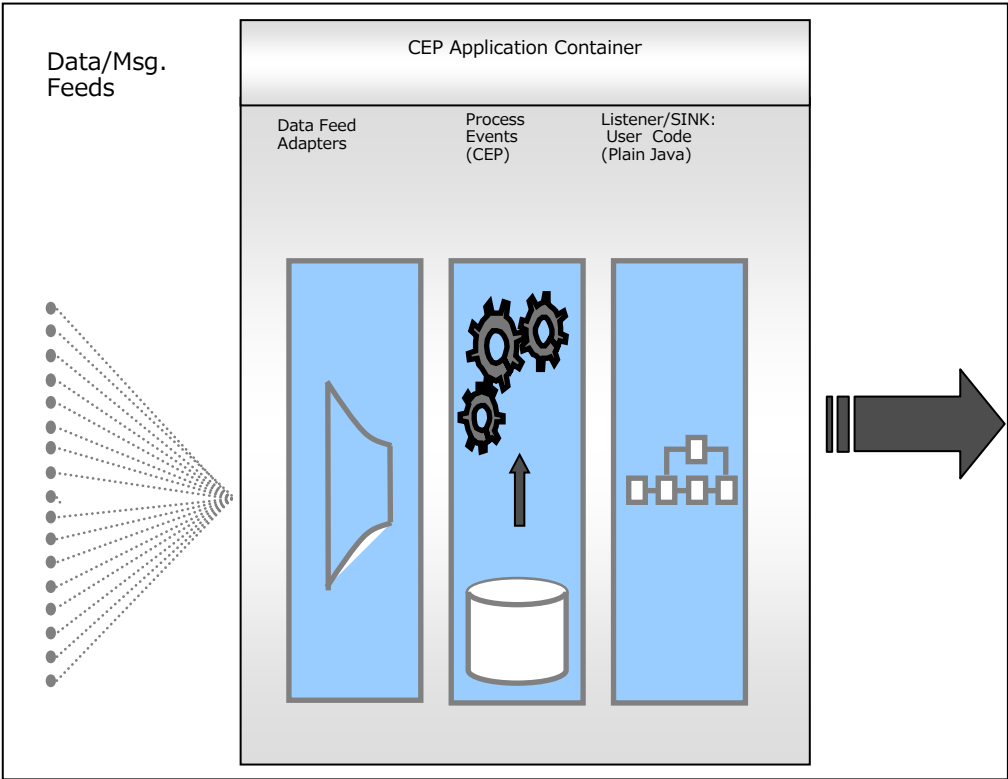


**Business
Logic
Events**

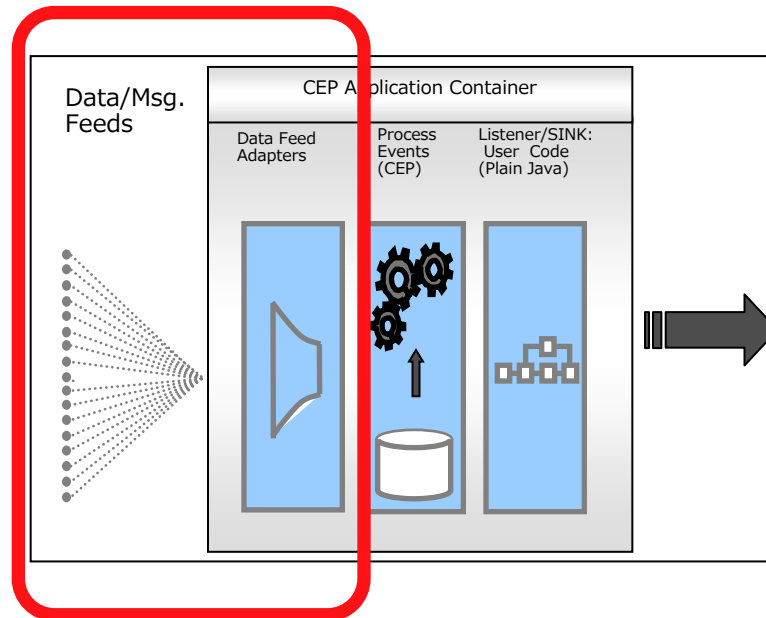
- Event-driven Architecture
- A generic data management infrastructure for processing in-flight data before data is potentially stored to deliver results in near real-time
- Programming language for defining rules
- It allows users to Aggregate/Correlate/Enrich/Detect Patterns in high speed streaming data



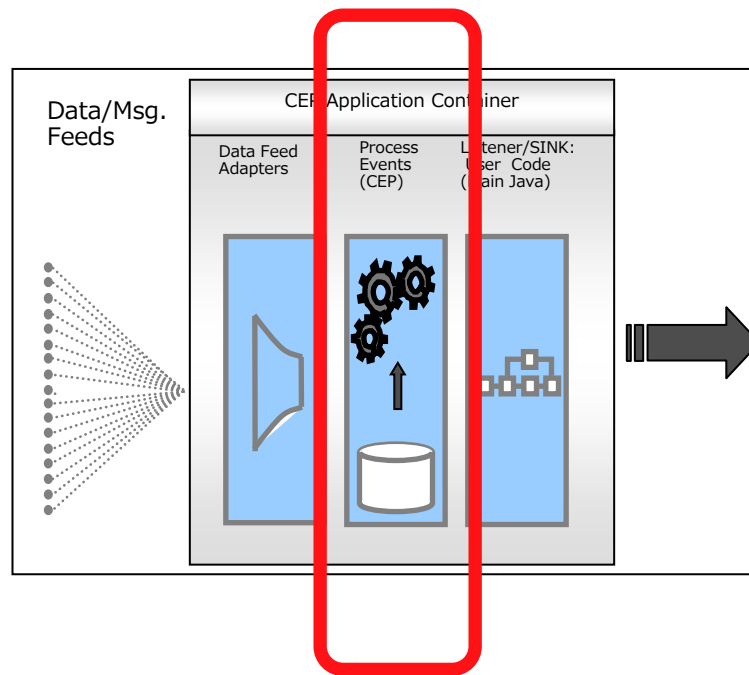
Complex Event Processing (CEP)



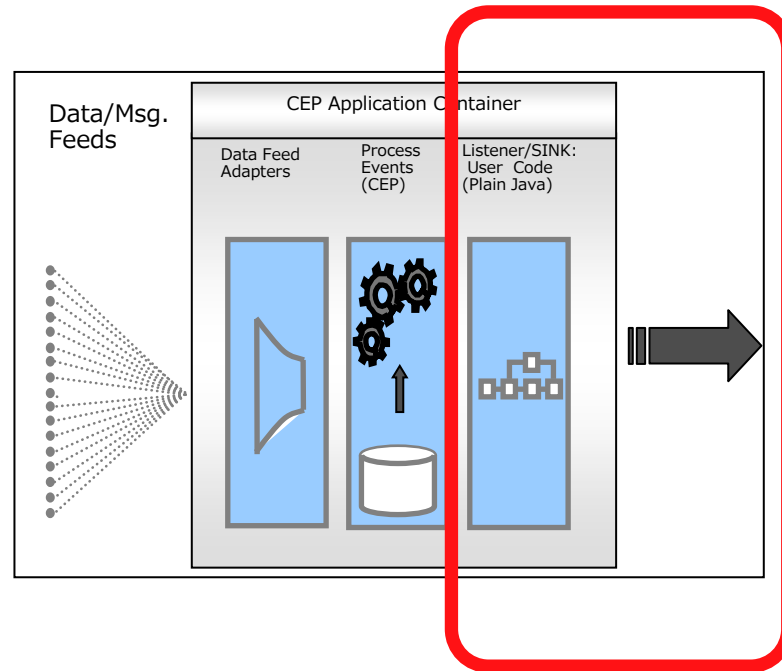
- Events generated at sources
- Adaptor captures event and sends it into the Event Processing Network



- Events processed in flight
- Merging multiple event sources and types
- Data enrichment by accessing external data sources (e.g. databases)



- Processing produces events
- Adaptor receives event and sends it into the downstream clients



Powering Dynamic M2M Event Processing with OSGi

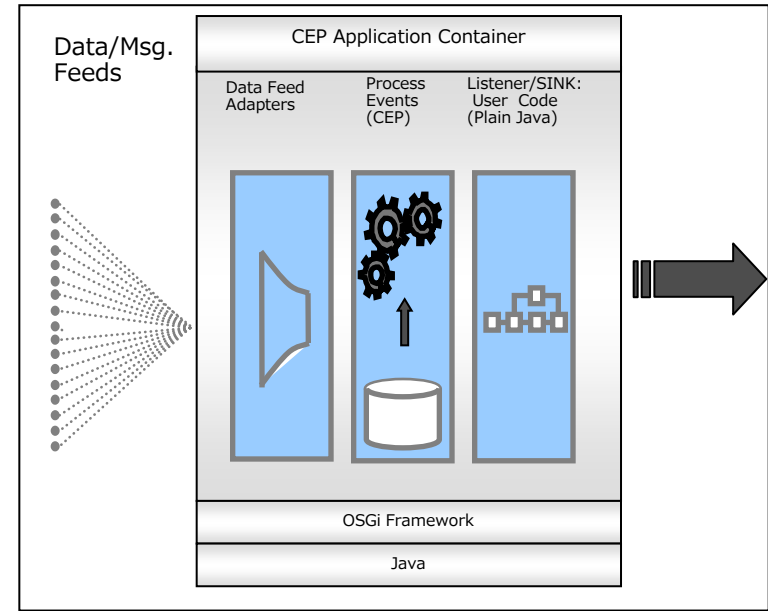
Dynamic Complex Event Processing and OSGi.

3. Dynamic Complex Event Processing

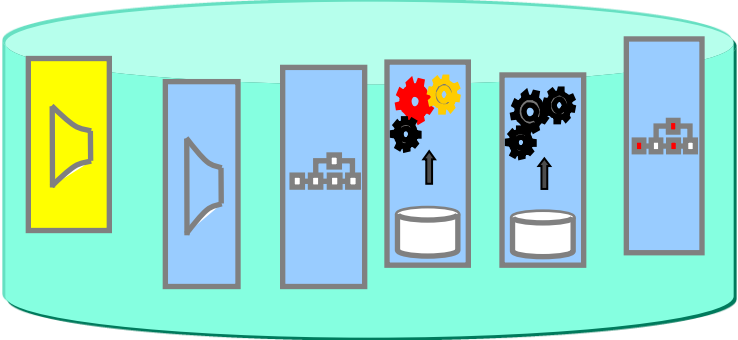
- Devices do not operate in a static environment
- Inputs change
- Knowledge is gained from analytics
- Additional systems want to receive the output

- Our Complex Event Processing engine needs to be dynamic
- Ability to change behavior without stopping the flow
- Allow a higher level system to change the processing rules

- Dynamic Modular System for Java
- Mature Lightweight Application Framework
 - Ideal for embedded environments
- Supports Module Lifecycle
 - Install/start/stop/uninstall/upgrade
 - Remotely manageable
 - Versioning
- Services Model
 - Advertise and discover services
 - Modules are dependent on service not implementation

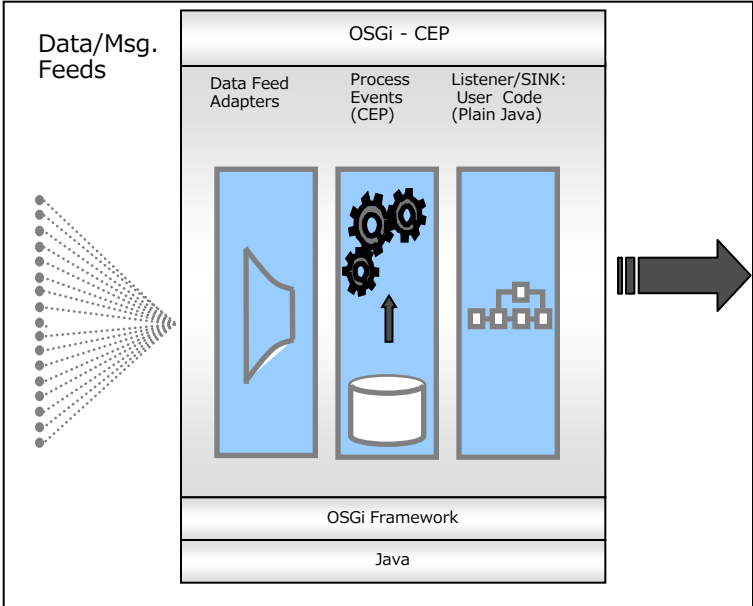


Powered by  **OSGi**
Alliance

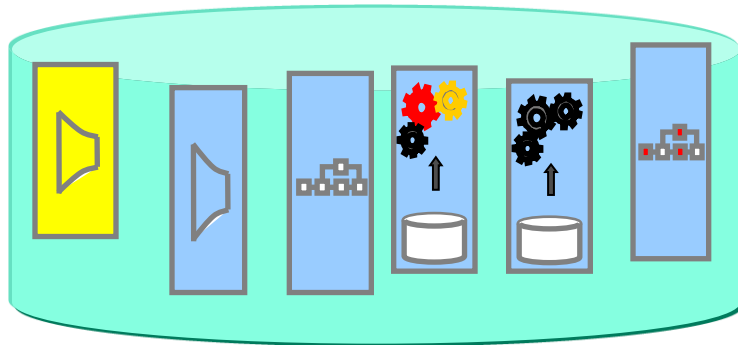


OSGi Management System & Repository

Remote Device

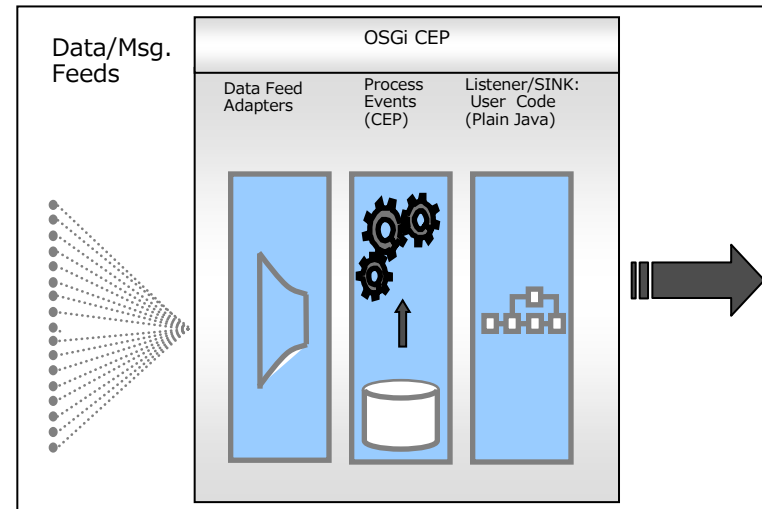


Deployed System. Happily processing...

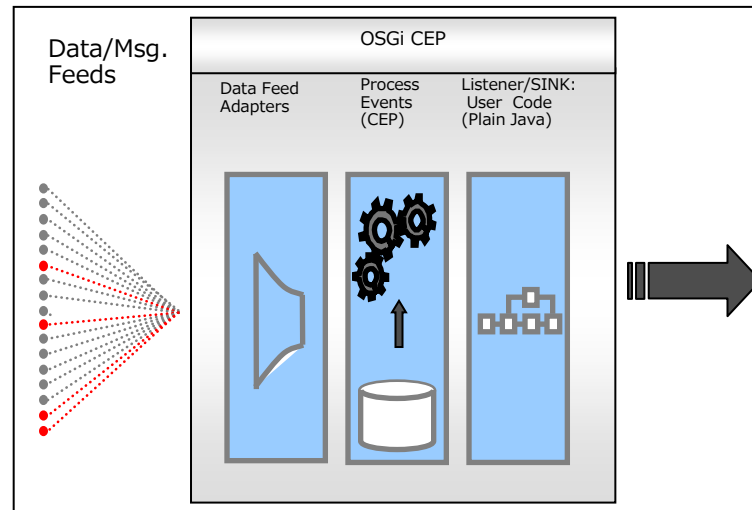


OSGi Management System
&
Repository

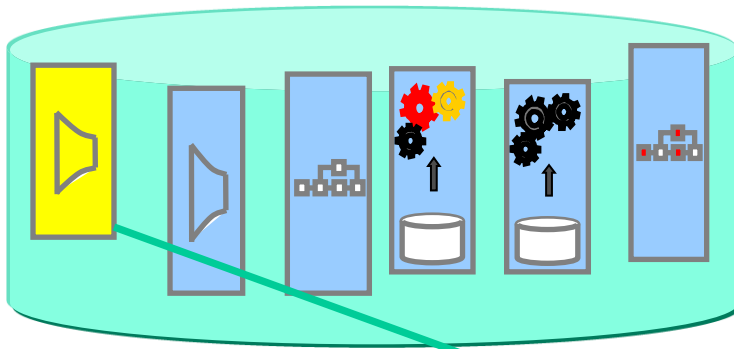
Remote Device



The Data Inputs Change

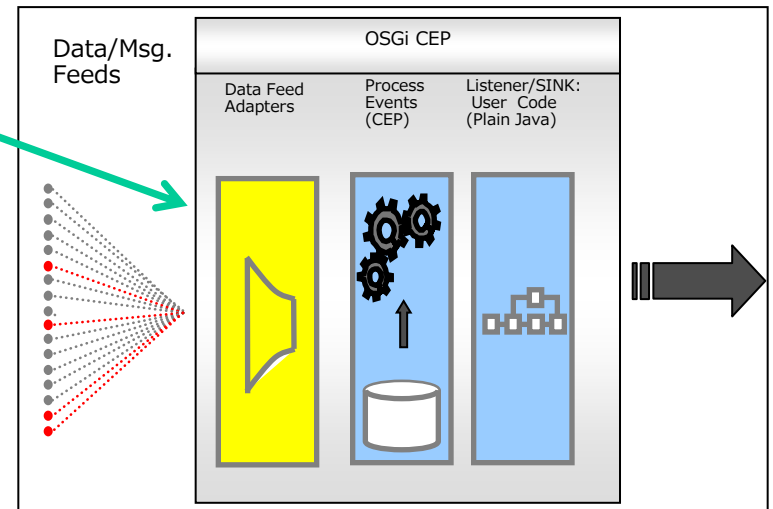


Update the adaptor

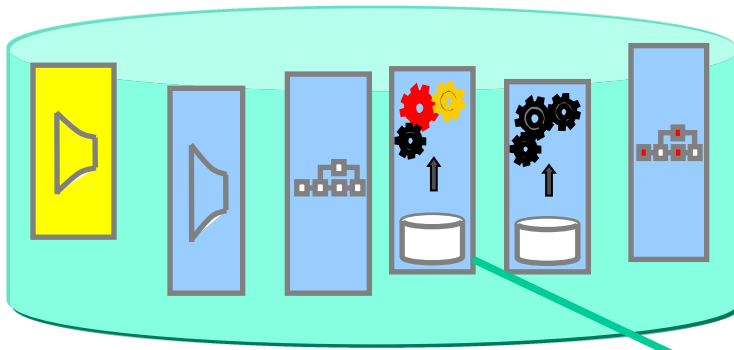


OSGi Management System
&
Repository

Remote Device

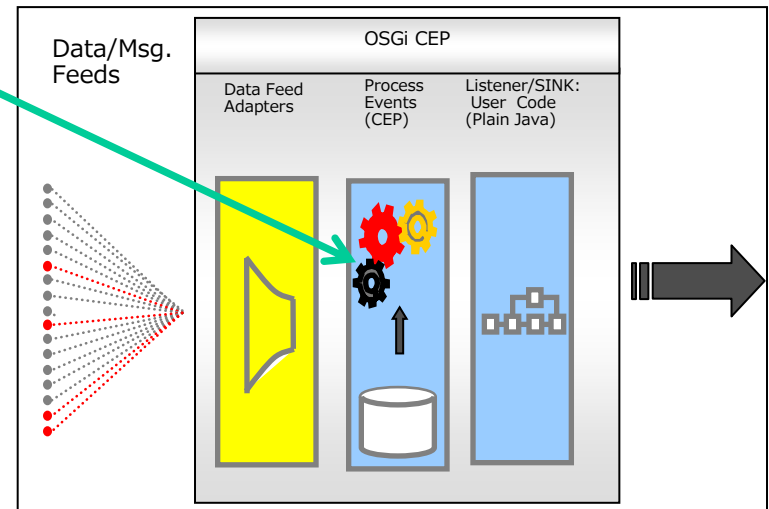


Process the Events Differently

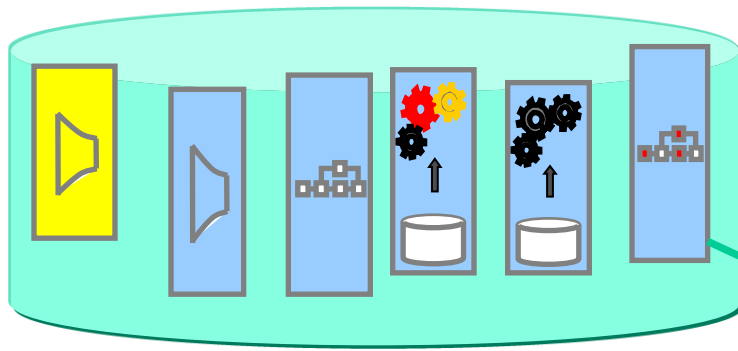


OSGi Management System
&
Repository

Remote Device

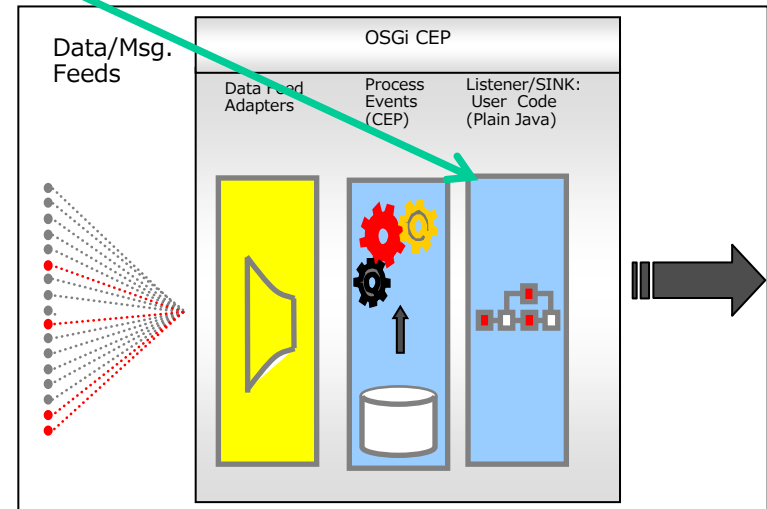


Forward to additional locations for processing

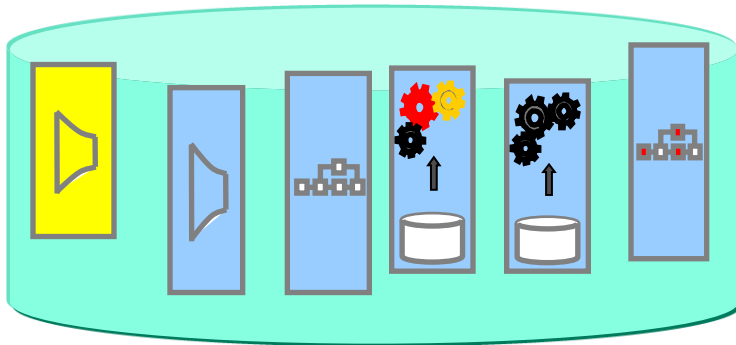


OSGi Management System
&
Repository

Remote Device

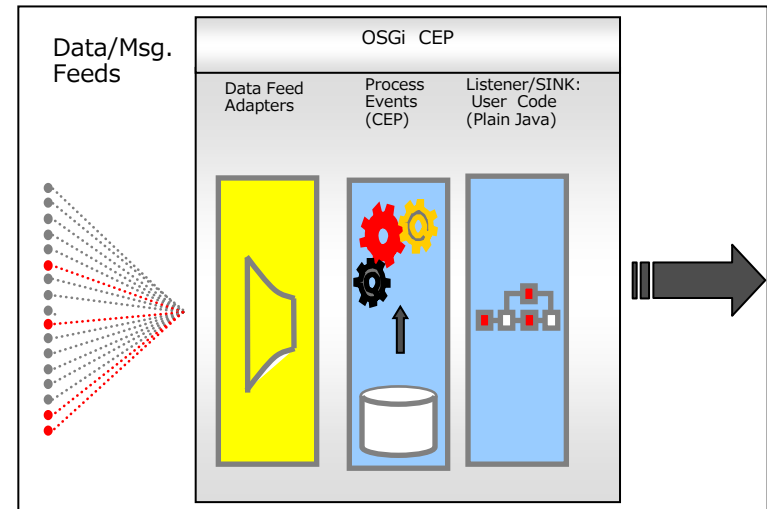


Happily processing again...



OSGi Management System
&
Repository

Remote Device

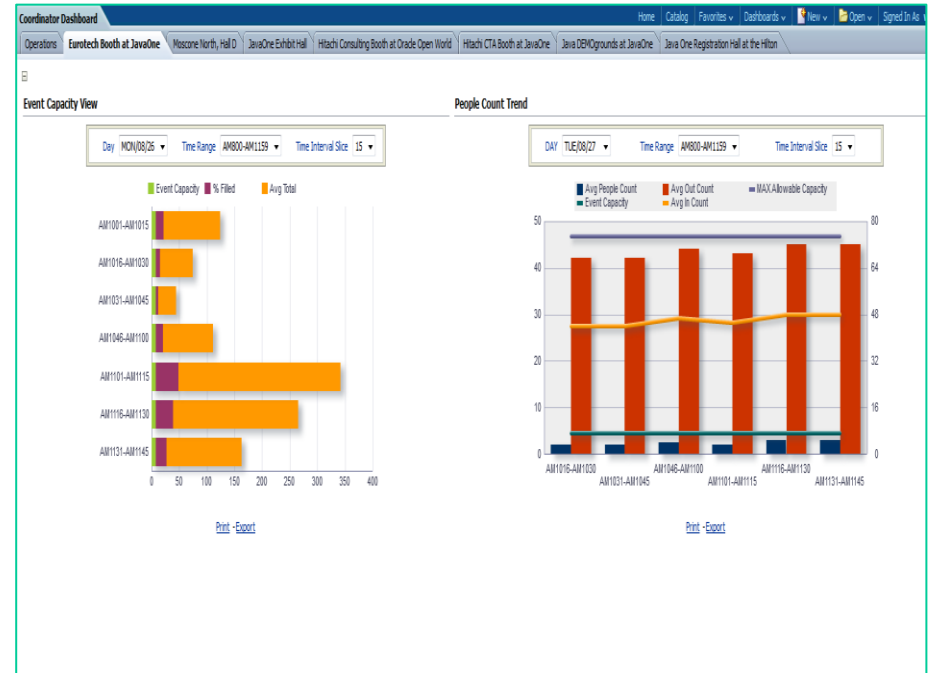
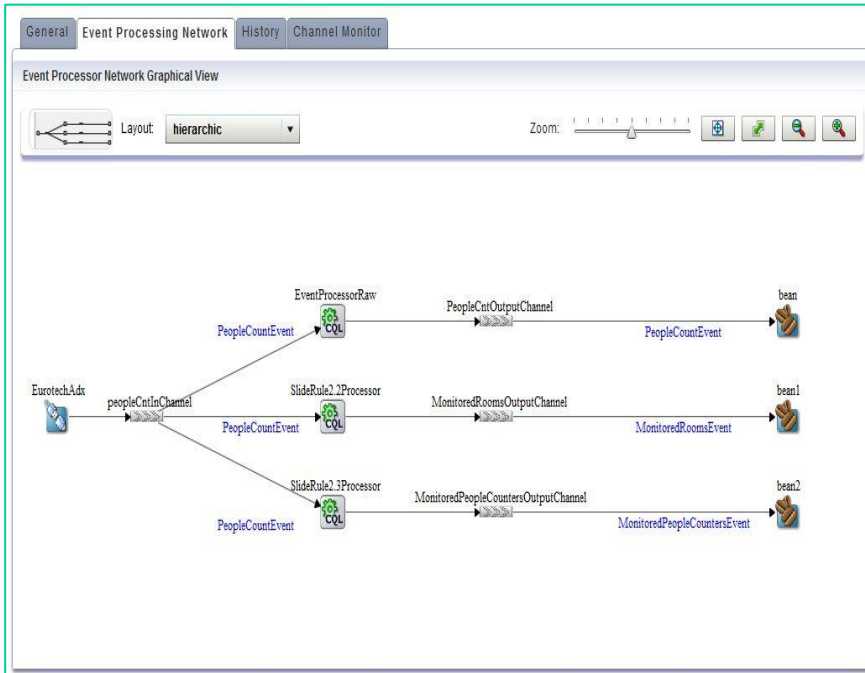


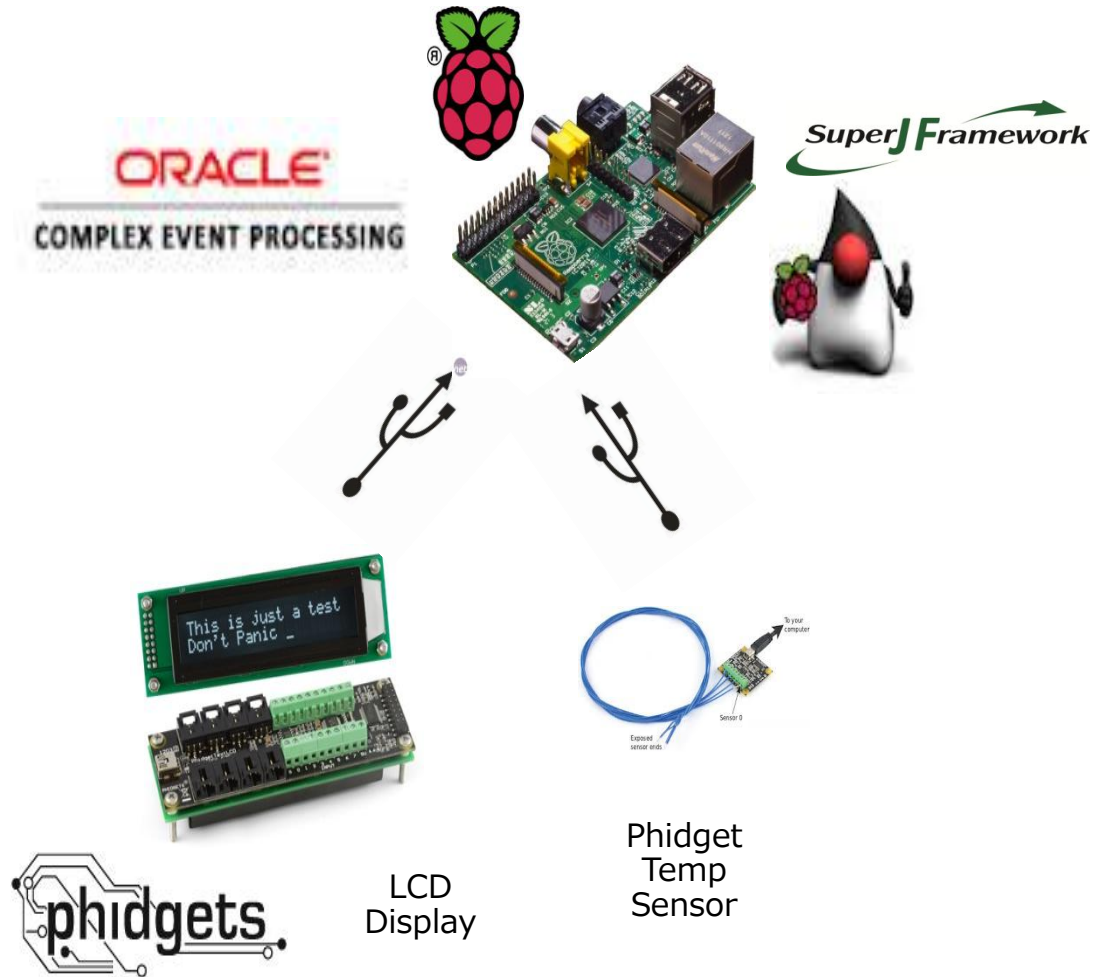
Powering Dynamic M2M Event Processing with OSGi

How OSGi and Java enables smart data on M2M aggregators and gateways.

4. Demo

Example: JavaOne IoT In Motion







- OSGi based
- Continuous Query Language (CQL) for defining rules
- <http://www.oracle.com/us/technologies/java/embedded/event-processing/overview/index.html?ssSourceSiteId=opn>



- Arm Based Linux platform
- <http://www.raspberrypi.org/>

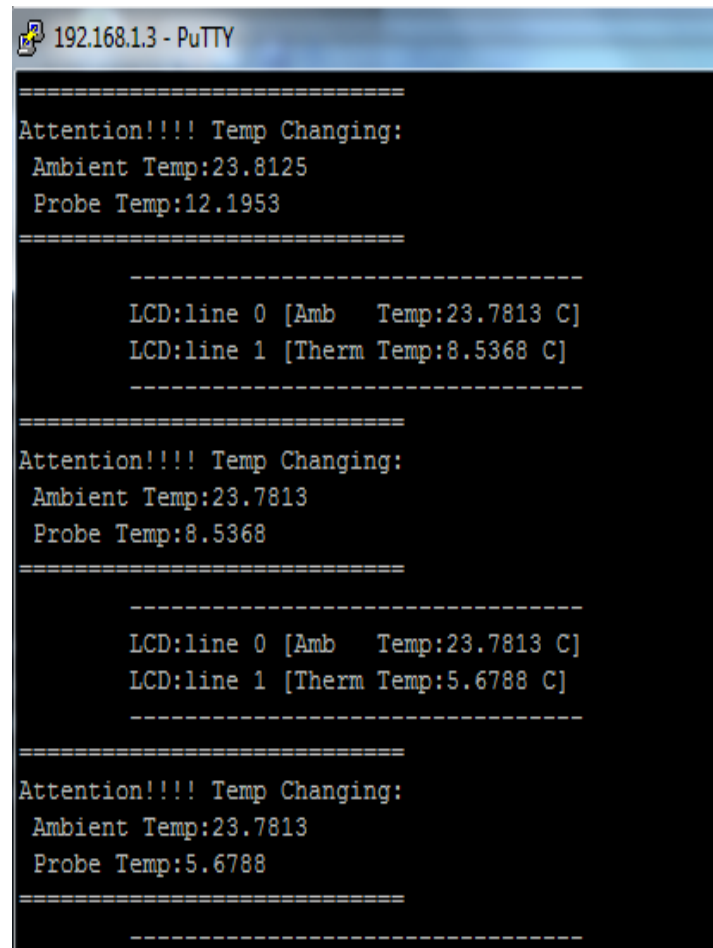


- USB hardware devices
- Open and inexpensive
- <http://www.phidgets.com/>



- Hitachi's OSGi Framework

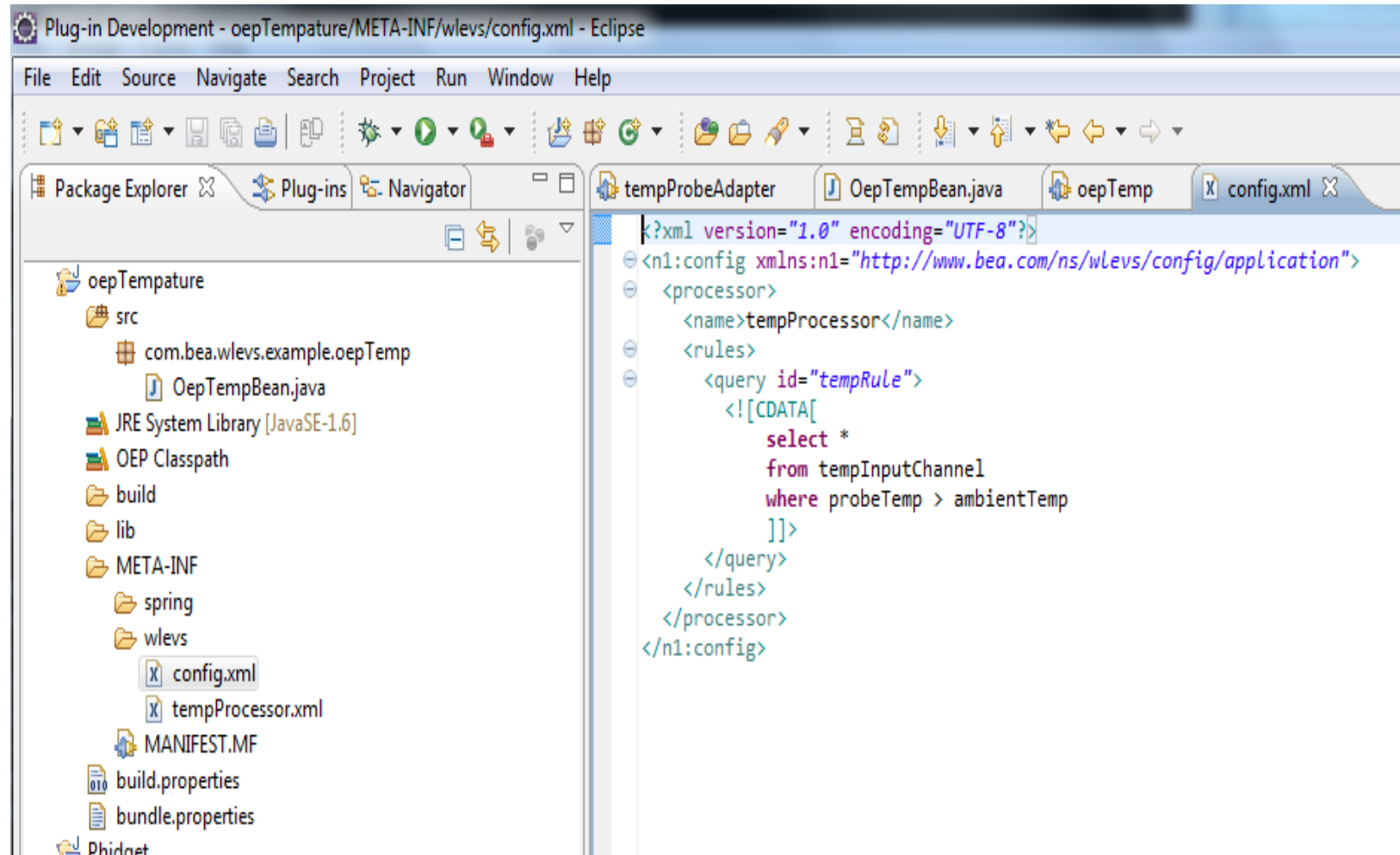
Start reporting temperature changes below ambient temperature



The screenshot shows a PuTTY terminal window titled "192.168.1.3 - PuTTY". The terminal output consists of three identical blocks of text, each separated by a line of equals signs. Each block starts with "Attention!!!! Temp Changing:", followed by "Ambient Temp:23.8125" and "Probe Temp:12.1953". Below this is a dashed line, then "LCD:line 0 [Amb Temp:23.7813 C]" and "LCD:line 1 [Therm Temp:8.5368 C]", followed by another dashed line. The second and third blocks show the same "Attention!!!! Temp Changing:" message, but with "Ambient Temp:23.7813" and "Probe Temp:8.5368" (or "5.6788" in the third block), and the LCD line 1 temperature is "5.6788 C".

```
=====  
Attention!!!! Temp Changing:  
Ambient Temp:23.8125  
Probe Temp:12.1953  
=====  
  
-----  
LCD:line 0 [Amb Temp:23.7813 C]  
LCD:line 1 [Therm Temp:8.5368 C]  
-----  
  
=====  
Attention!!!! Temp Changing:  
Ambient Temp:23.7813  
Probe Temp:8.5368  
=====  
  
-----  
LCD:line 0 [Amb Temp:23.7813 C]  
LCD:line 1 [Therm Temp:5.6788 C]  
-----  
  
=====  
Attention!!!! Temp Changing:  
Ambient Temp:23.7813  
Probe Temp:5.6788  
=====  
  
-----
```


Change the rules and redeploy remotely



Now reporting temperature changes above ambient temperature

```
192.168.1.3 - PuTTY
=====
Attention!!! Temp Changing:
Ambient Temp:23.8594
Probe Temp:25.7484
=====

-----
LCD:line 0 [Amb Temp:23.8672 C]
LCD:line 1 [Therm Temp:26.3546 C]
-----

=====
Attention!!! Temp Changing:
Ambient Temp:23.8672
Probe Temp:26.3546
=====

-----
LCD:line 0 [Amb Temp:23.8672 C]
LCD:line 1 [Therm Temp:25.8299 C]
-----

=====
Attention!!! Temp Changing:
Ambient Temp:23.8672
Probe Temp:25.8299
=====

-----
```

END

Powering Dynamic M2M Event Processing with OSGi
Dynamic Complex Event Processing and OSGi.

6/12/2014

Hitachi Communication Technologies America, Inc.

Walt Bowers

Chief Architect OSGi Solutions

Walt.bowers@hitachi-cta.com

**Human Dreams.
Make IT Real.**

HITACHI
Inspire the Next

Appendix

2012

3 006 477 107 200 GB added to the “digital universe”
~30% of it generated by machines

2020

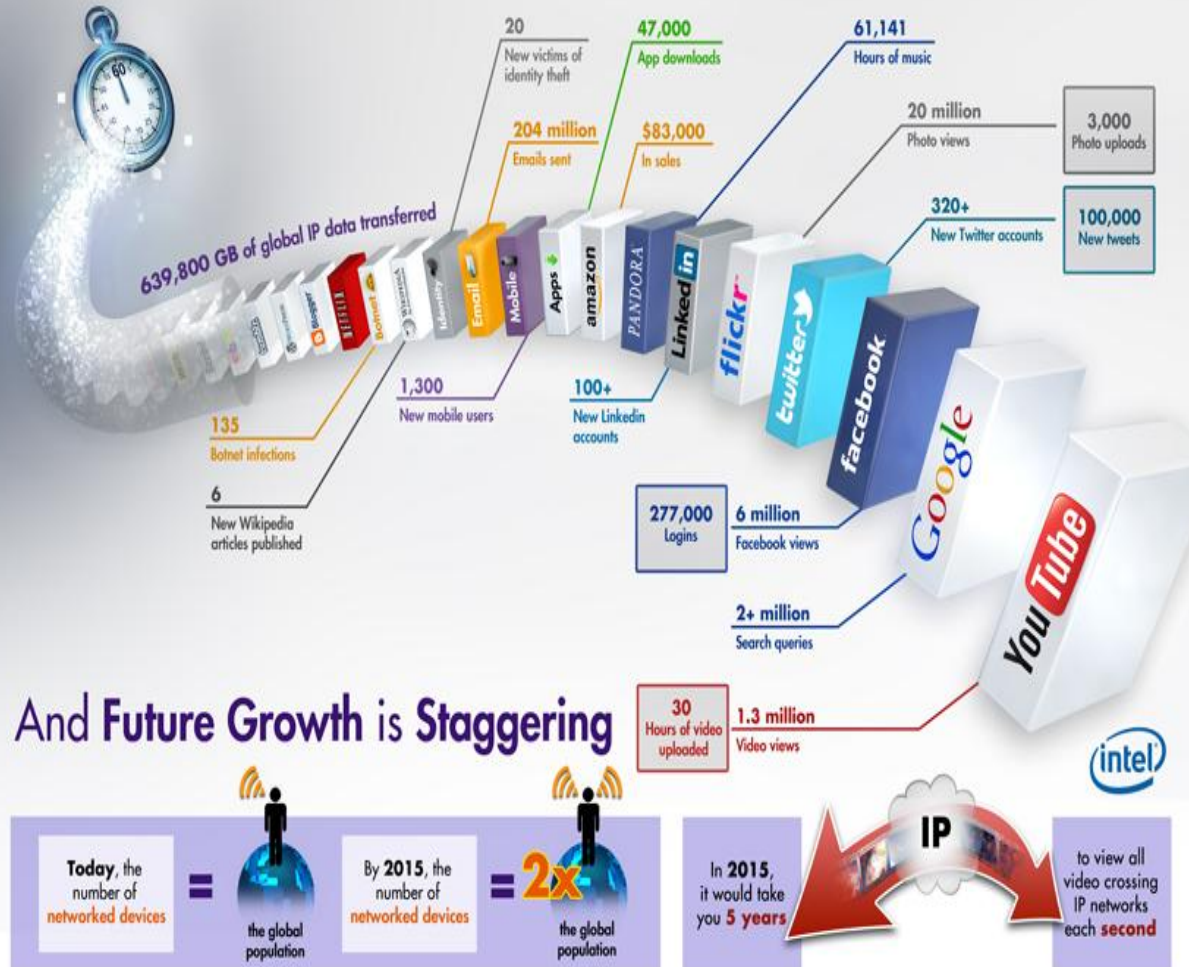
Assuming
2GB/hr it will
take 2.3 billion
years to
download



42 949 673 000 000 GB – 15 x increase
42 % will be generated by devices

According to IDC's “Digital Universe in 2020” study published in December 2012

What Happens in an Internet Minute?



Web and Social Media

- *Clickstream Data*
- *Twitter Feeds*
- *Facebook Postings*
- *Web Content*

Machine-to-Machine

- *Smart Meters Readings*
- *RFID Readings*
- *Oil Rig Sensors*
- *GP Signals*

Big Transaction Data

- *Healthcare Claims*
- *Telecommunication Call Details Record*
- *Utility Billing Records*

Biometrics

- *Facial Recognition*
- *Genetics*

Human Generated

- *Call Center Voice Recording*
- *Email*
- *Electronic Medical Records*

