

FIWARE tutorial – First steps

Budapest, Hungary, 26th November 2014



Sándor Laki, József Stéger, Gergő Gombos
Wigner Research Centre for Physics
Contact: laki.sandor@wigner.mta.hu



<http://fiware.org>

<http://lab.fiware.org>



Understanding FIWARE (Open Standard Platform)

(advanced OpenStack-based Cloud + rich library of Generic Enablers)

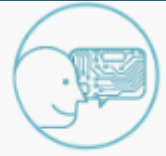


FIWARE Generic Enablers (GEs)

- A FIWARE Generic Enabler (GE):
 - set of general-purpose **platform functions** available through **APIs**
 - Building with other GEs a [FIWARE Reference Architecture](#)
- [FIWARE GE Specifications](#) are open (public and royalty-free)
- **FIWARE GE implementation (FIWARE GEi):**
 - Platform product that implements a given GE Open Spec
 - There might be multiple compliant GEis of each GE Open Spec
- **One open source reference implementation of each FIWARE GE (FIWARE GERi):**
 - Well-known open source license
 - Publicly available [Technical Roadmap](#) updated in every release
- Available FIWARE GEis, GERis and incubated enablers are published on the [FIWARE Catalogue](#)



FIWARE major differential features



FIWARE Catalogue

Cloud



- Federation of infrastructures (private regions)
- Automated GE deployment

Data/Services Delivery



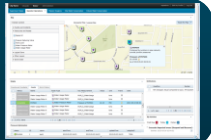
- Complete Context Management Platform
- Integration of Data and Media Content

IoT



- Easy plug&play of devices using multiple protocols
- Automated Measurements/Action \leftrightarrow Context updates

Apps



- Visualization of data (operation dashboards)
- Publication of data sets/services

Web UI



- Easy incorporation of advanced 3D and AR features
- Visual representation of context information

Security



- Security Monitoring
- Built-in Identity/Access/Privacy Management

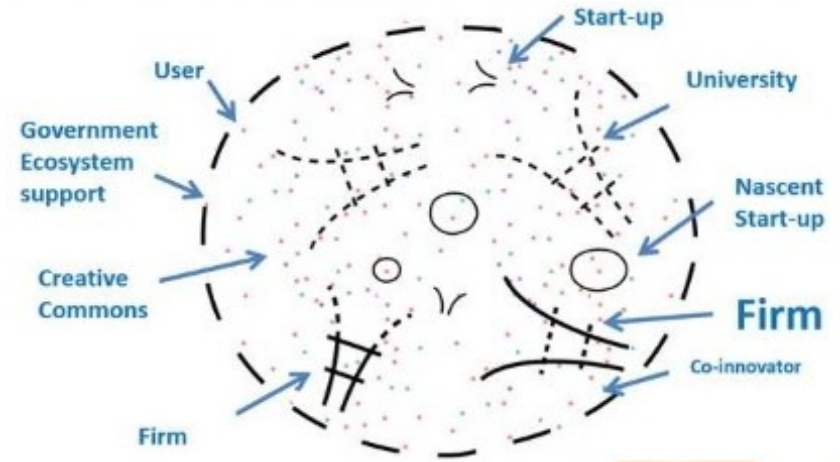
I2ND



- Advanced networking (SDN) and middleware
- Interface to robots



Ecosystem and platform: two tied concepts



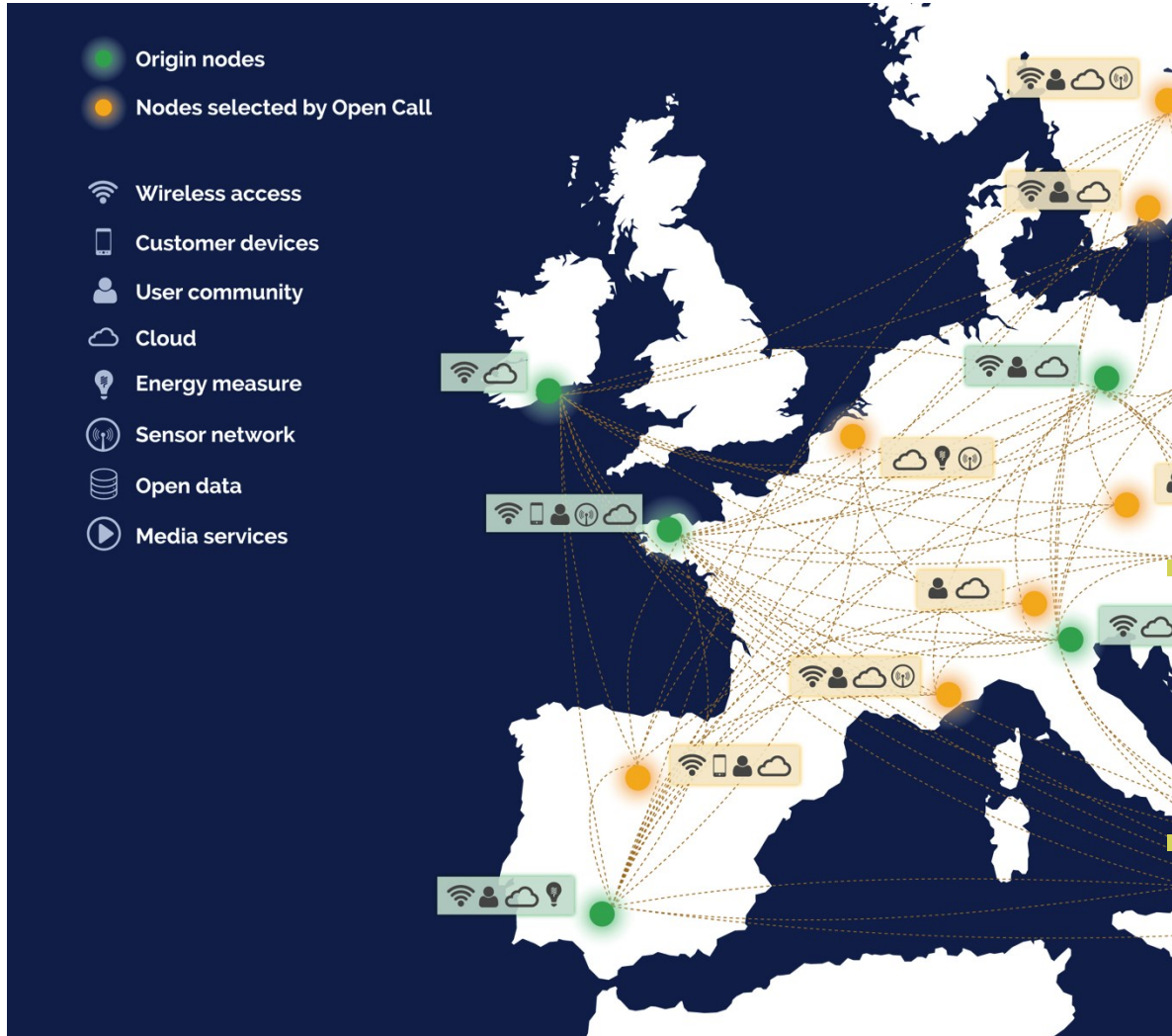
**Open Standard
Platform**



**Sustainable Open
Innovation Ecosystem**



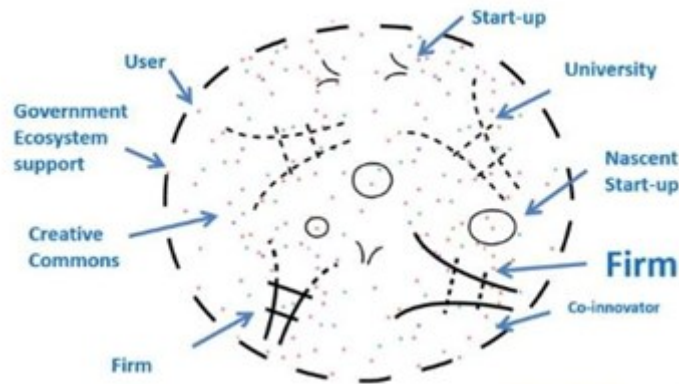
Extending the FIWARE Lab offering for service providers and developers



- 12 nodes in Europe providing up to 3000+ cores, 16TB+ Ram, 750TB+ HD
- Creation of nodes in Mexico (1000+ cores) and Brazil
- Level 1 and Level 2 support for the nodes
- Showcases for developers, infrastructures, smart businesses

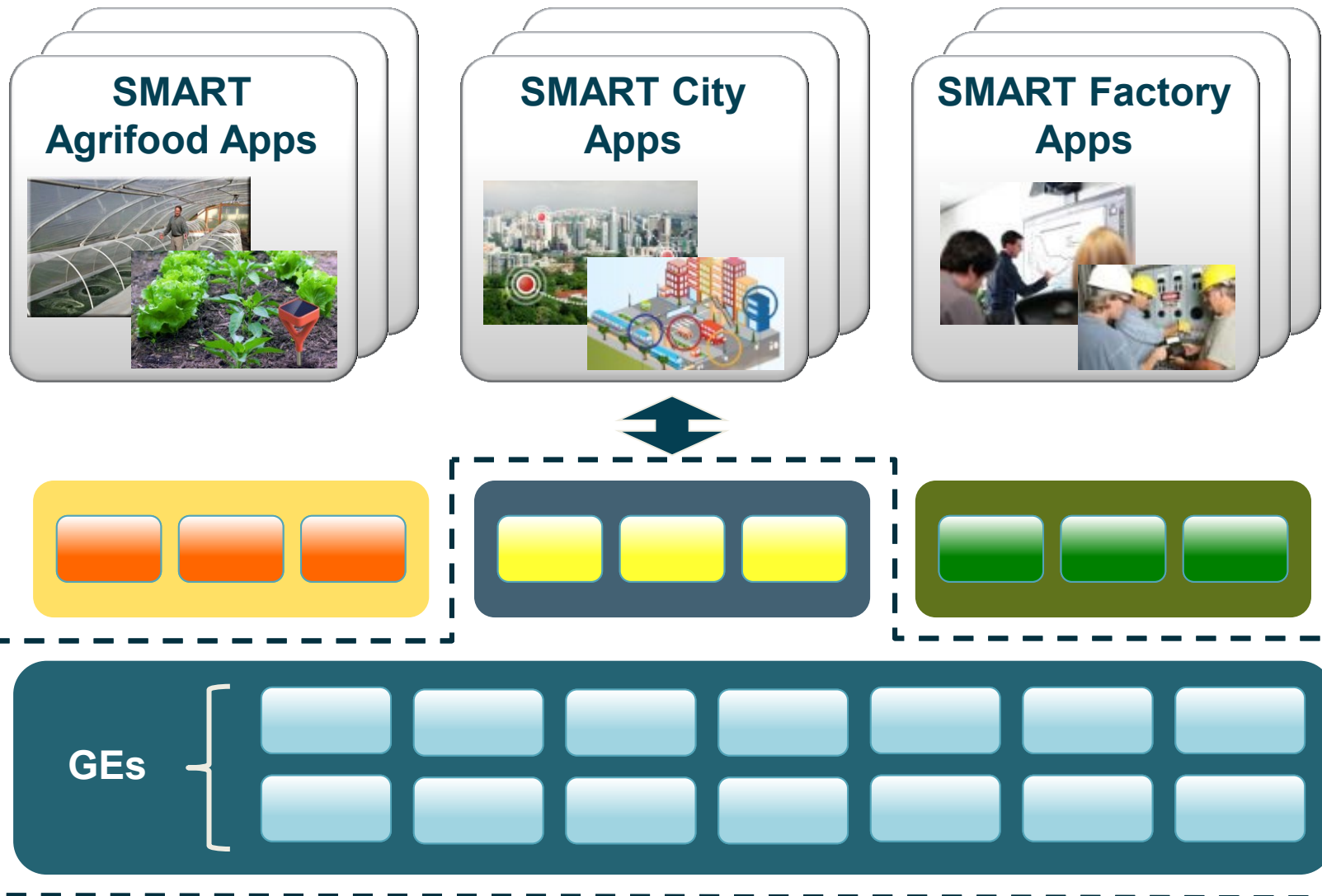
A live instance of FIWARE
available to developers for
free experimentation

Not only platform, tools...this is
powered by data!



Modularity; integration with legacies

Domain-specific platforms = FIWARE + specific enablers



FIWARE – Getting started

- Cloud portal: <https://cloud.lab.fi-ware.org>
- GE/GEri catalogue: <http://catalogue.fi-ware.org/enablers>
- Open data sources: <https://data.lab.fi-ware.org/>
- Mashup framework: <https://mashup.lab.fi-ware.org>
- FIWARE Academy: <http://edu.fi-ware.org/>
- SE catalogues:
 - E-Health: <http://catalogue.fi-star.eu/enablers>
 - Media & content: <http://mediafi.org/catalogue/>
 - Transport and agrifood – FISPACE: <http://dev.fispace.eu/doc/wiki/Home>
 - Manufacturing: <http://www.fitman-fi.eu/phase-iii-package/information-for-phase-iii-bidders>
 - Smart Energy: <http://finesce.github.io/>

L1 help desk:

fiware-lab-help@lists.fi-ware.org



Support and coaching

Home Available Courses My Courses



e-Learning Platform



<http://edu.fi-ware.org/>

1 Select a Category

Cloud Hosting (2)
Data/Context Management
Internet of Things (IoT) Services Enablement (1)
Applications and Services Ecosystem and Delivery

2 Select a Course

FI Application Project Management
This course introduces the use and usage of the FI Application Framework (FI-CoDE) to create and manage FI-based application projects, with...

3 Log in as Guest (if necessary)

Some courses may allow guest access

Login as - guest

4 Select the Course Topic

1. Data Collection

Data Collection

This tutorial explain how to set Eclipse and Trace Ana...
in order to collect the SAR and TCPDump data used by Trac...

5 Confirm the Course Topic

Mode: ☐ Preview ☒ Normal

Enter

6 Start the Course

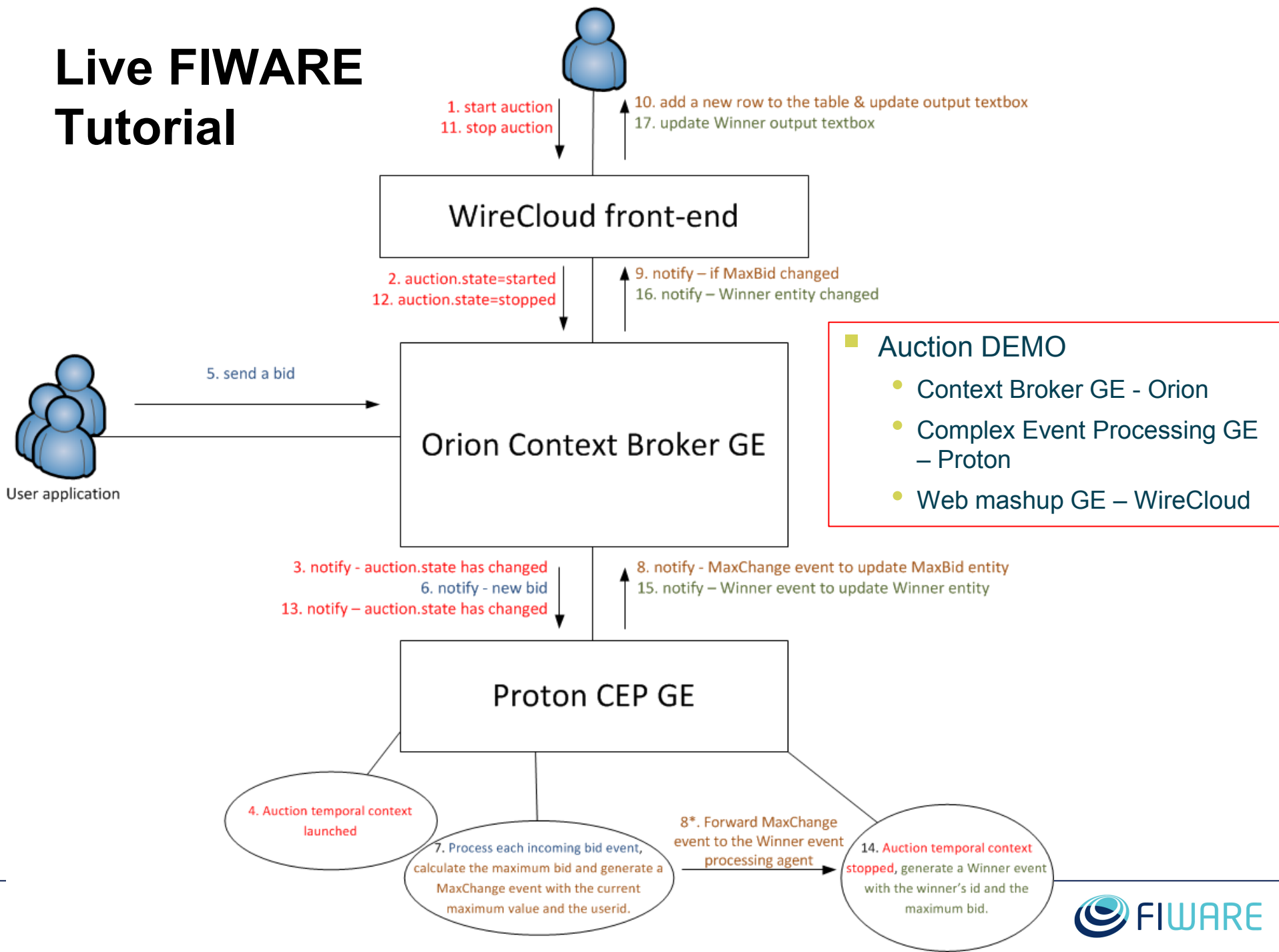
START

Watch all our WEBINARS

Learn how to use our most popular
Generic Enablers in just a few steps



Live FIWARE Tutorial



FIWARE Lab (<http://lab.fiware.org>)

The screenshot shows the FIWARE Lab website. At the top is a navigation bar with links: FIWARE Lab, Cloud, Store, Mashup, Account, and Help&info. Below this is a large video player titled "FI-WARE Cloud Blueprint templates" with a play button and the text "OPEN APIs FOR OPEN MINDS". To the right of the video player is a sidebar with a heading "Click on the links to see the FIWARE video tutorials" and sub-links: Cloud, Store, Mashup, Account. Below these are tabs for "Blueprints", "Instances", and "Object Storage". The "Blueprints" tab is active, showing text: "Blueprint Templates let you quickly create a template from which to build your application. You can specify the software you need in the Tier Templates and easily deploy all the instances with one click." Below the video player is a horizontal bar with four icons and links: "Need Help? Ask a question.", "Our GEs See our Catalogue.", "FIWARE Lab nodes Learn about FIWARE Ops.", and "eLearning Train yourself." Below this bar is a blog post titled "The challenges are closing" with the text "POSTED APRIL 22, 2014 BY ADMIN" and a paragraph: "The Smart Society Challenge and the FI-WARE Excellence Award are nearing the deadline. After an extension to make room for more ideas and contestants, the call for ideas is closing on the 28th." To the right of the blog post is a "Tweets" section with a "Follow" button and a tweet from @IMPACT_acc dated 12 Sep: "If u r a coder or an entrepreneur,".

VM provisioning

fi-ware Dashboard

Project: demo_project

Images

Name	Status	Visibility
Centos-6.3-etc	active	public
Centos-6.3-etc	active	public
Puppet-Awsume-6	active	public
Ubuntu-12.0-etc	active	public
Ubuntu-12.0	active	public
puppet-awsume	active	public
puppet-awsume7	active	public
sdcc4RegularUpdates	active	private
sdcc7Regul		
sdcc7Regul		
snapshot-c		
testUC		
testUC		

Instances

Instance Name	IP Address	Size	Keypair	Status	Task
blueprint04-tomcat5-1	10.00.20.5	2048 MB RAM 1 VCPU 20GB Disk		SHUTOFF	None
		2048 MB RAM 1 VCPU 20GB Disk		SHUTOFF	None
		2048 MB RAM 1 VCPU 20GB Disk		SHUTOFF	None
		2048 MB RAM 1 VCPU 20GB Disk		SHUTOFF	None
		2048 MB RAM 1 VCPU 20GB Disk		SHUTOFF	None
		2048 MB RAM 1 VCPU 20GB Disk		SHUTOFF	None
		2048 MB RAM 1 VCPU 20GB Disk		ACTIVE	None
		2048 MB RAM 1 VCPU 20GB Disk		SHUTOFF	None

Launch Instances

1. Details | 2. Access & Security | 3. Networking | 4. Post-Creation | 5. Summary

Instance Name *
awsume

Flavor
m1.tiny

Description
Specify the details for launching an instance. The chart below shows the resources used by this project in relation to the project's quotas.

Flavor Details

Name	m1.tiny
VCPUs	1
Root Disk	0 GB
Ephemeral Disk	0 GB
Total Disk	0 GB
RAM	512 MB

Project Quotas

Resource	Used	Available
Instance Count (0)	4	4 Available
VCPUs (0)	14	14 Available
Disk (20 GB)	1424	1876 GB Available
Memory (512 MB)	3092	3092 MB Available

* Mandatory fields.

Launch Instances

1. Details | 2. Access & Security | 3. Networking | 4. Post-Creation | 5. Summary

Selected Networks
net1 demonetwork

Available Networks
storage_network

Description
Control access to your instance via keypairs, security groups, and other mechanisms.

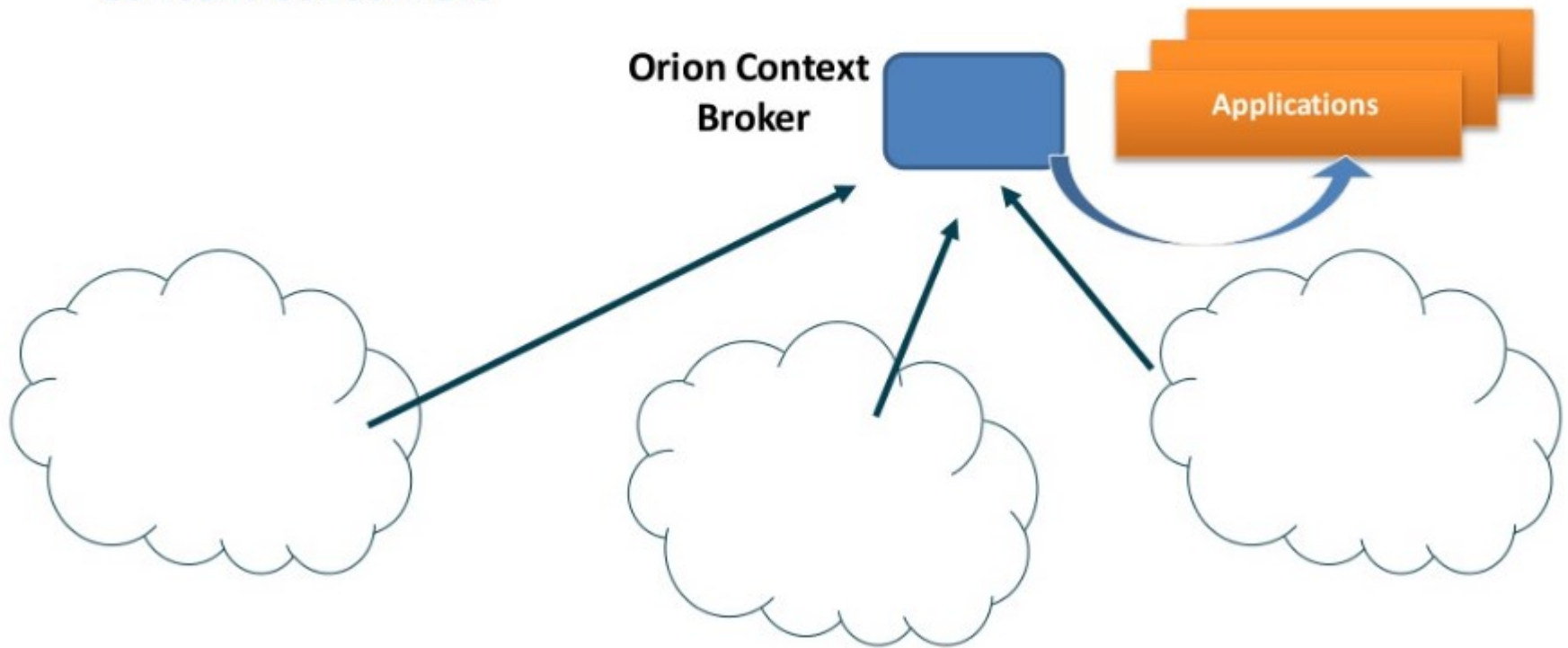
* Mandatory fields.

Actions

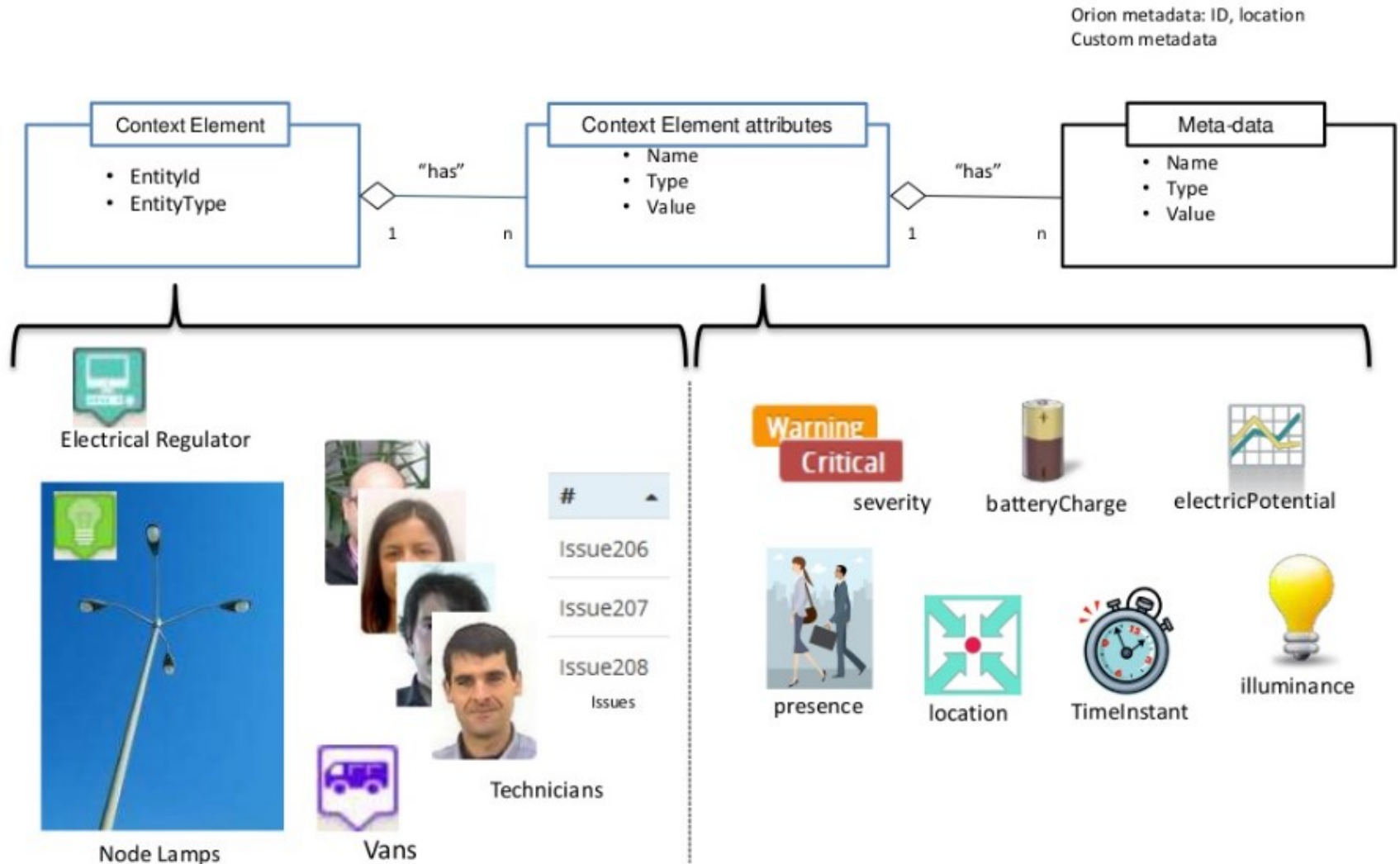
- Edit Instance
- Connect to Instance
- View Log
- Create Snapshot
- Pause Instance
- Unpause Instance
- Suspend Instance
- Resume Instance
- Change Password
- Reset Instance
- Terminate Instance

Orion Context Broker GErI

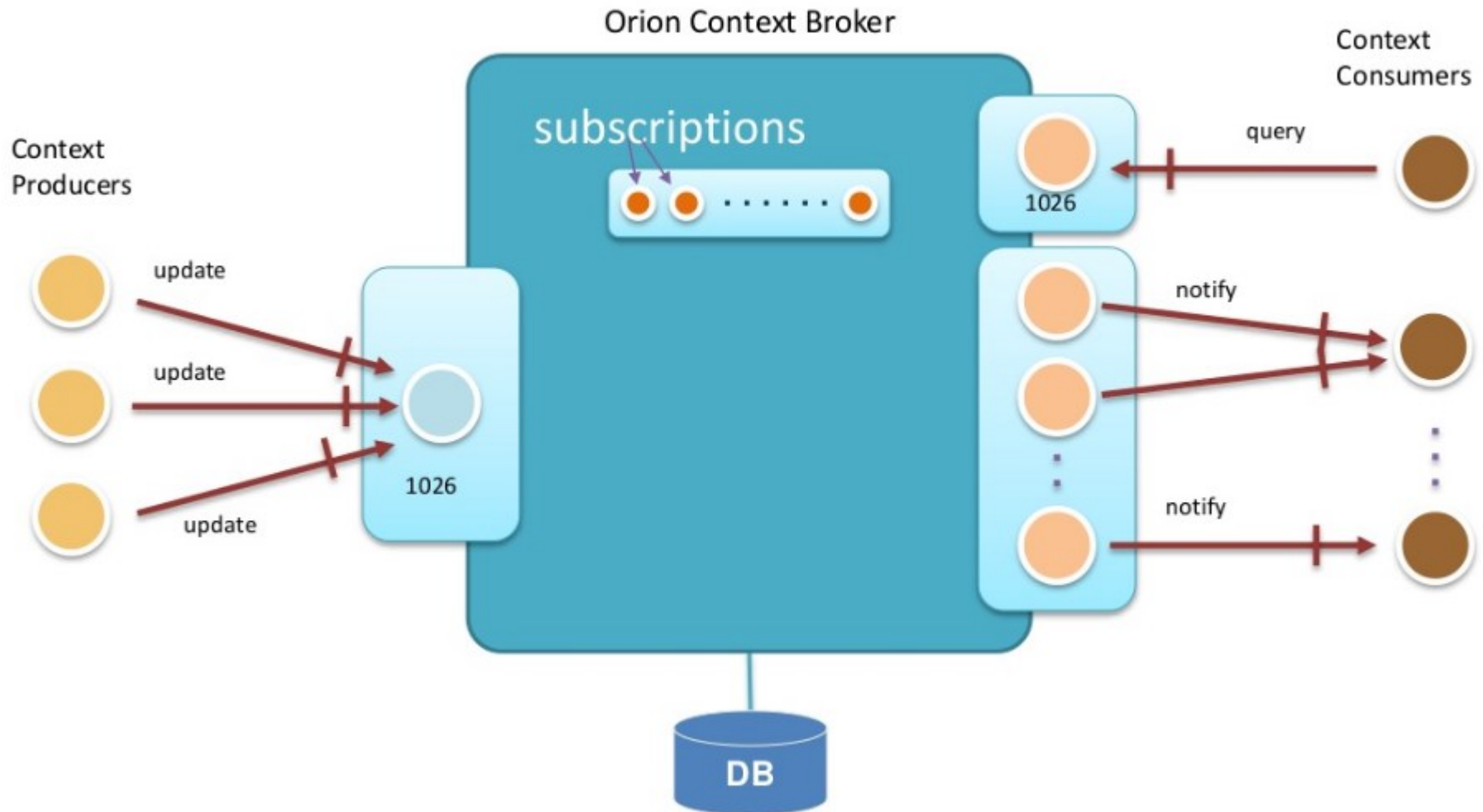
- Context Management in FI-WARE is about management of Context data (aka Context Information)
- Context Information is always relevant to “entities”, although entities can be anything (applications, users, things, ...)
- Orion Context Broker intermediates between **context producers** and **context consumers**



Orion Context Broker GErI NGSI information model



Orion Context Broker GErI



Orion Context Broker GEri

Create an entity

```
url = 'http://<IP_of_orionCB>:1026/NGSI10/updateContext'
payload = {
  "contextElements": [
    {
      "type": "Bid",
      "isPattern": "false",
      "id": "BidDEMO9",
      "attributes": [
        {
          "name": "bid",
          "type": "integer",
          "value": "0"
        }
      ]
    }
  ],
  "updateAction": "APPEND"
}

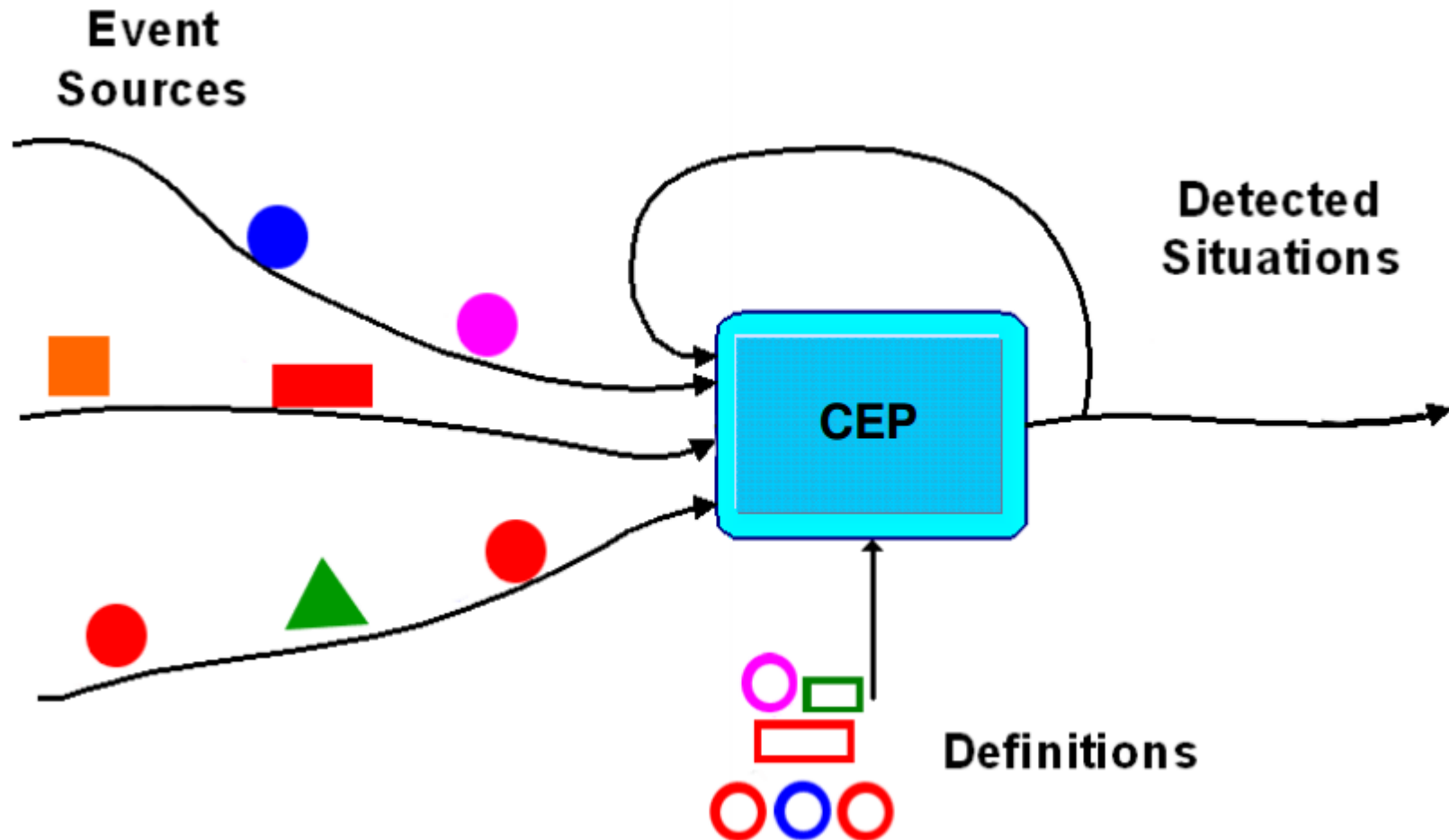
headers = { 'Accept' : 'application/json',
            'content-type' : 'application/json'
          }

r = requests.post(url, data=json.dumps(payload), headers=headers)
```

Update an entity

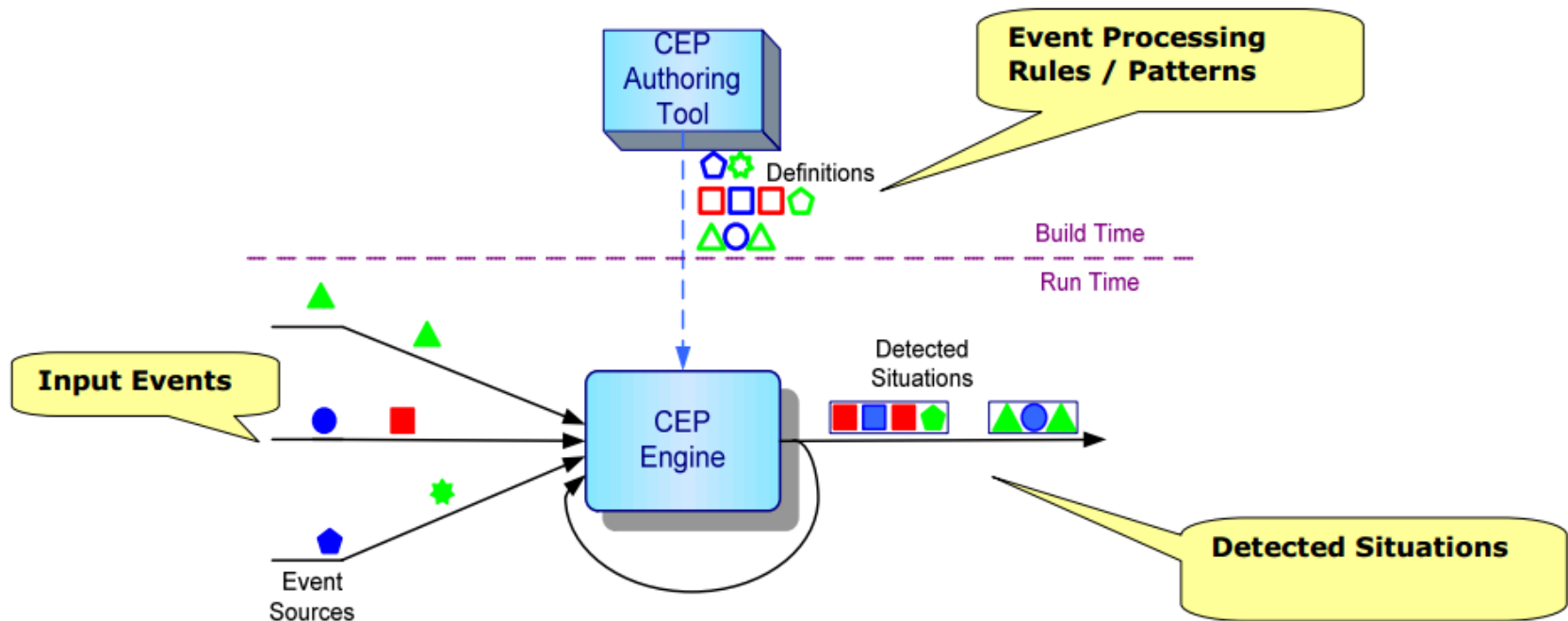
```
payload = {
  "contextElements": [
    {
      "type": "Bid",
      "isPattern": "false",
      "id": "BidDEMO9",
      "attributes": [
        {
          "name": "bid",
          "type": "integer",
          "value": "1000"
        }
      ]
    }
  ],
  "updateAction": "UPDATE"
}
```

Proton Complex Event Processing GErI

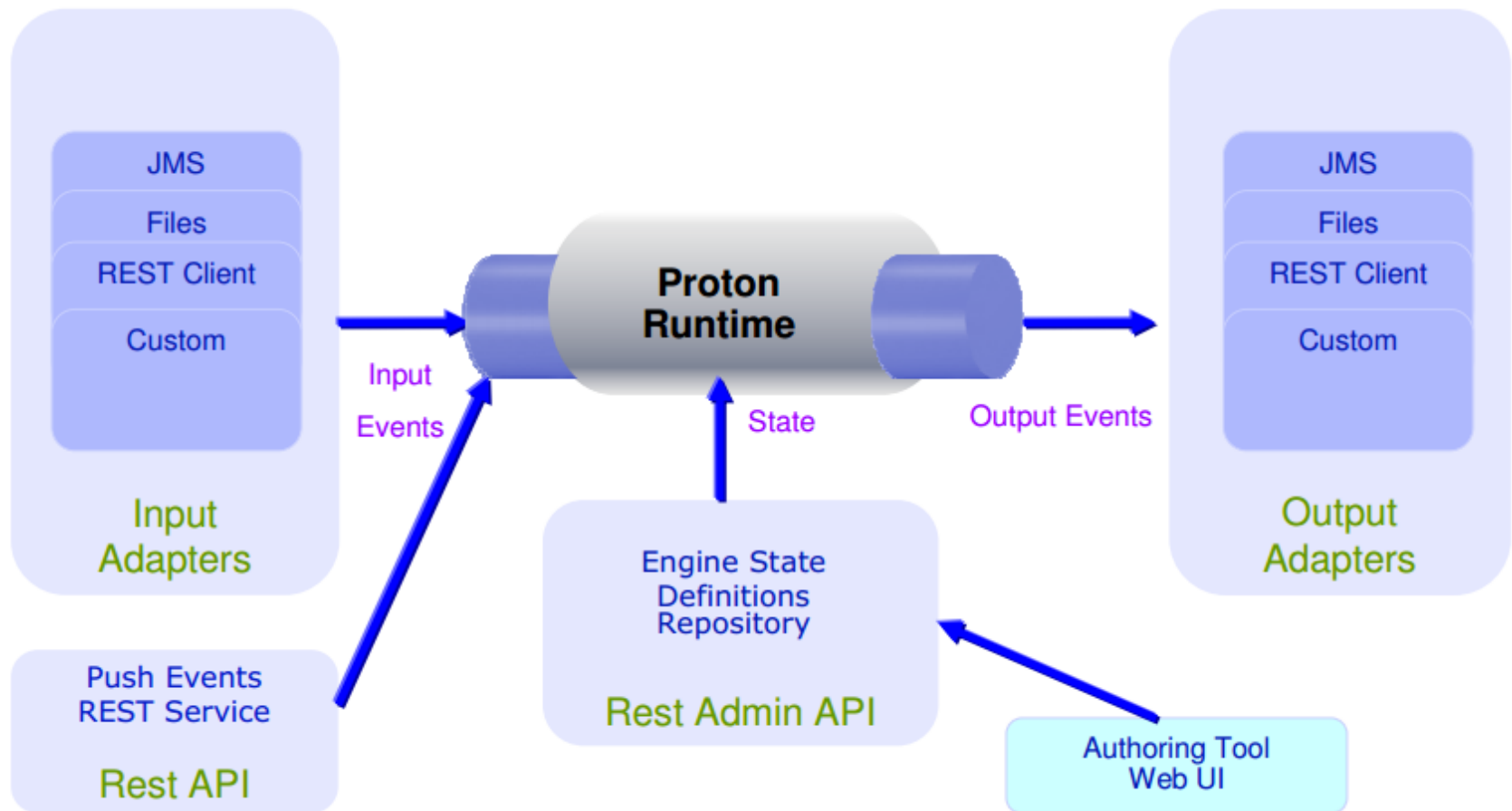


Proton Complex Event Processing GErI

- From **Event-Condition-Action** to **Pattern-Condition-Action**
- In certain scenarios, single events are insignificant, a CEP engine can detect combinations of events, and generate derived events, called **situations**, which are meaningful



Proton Complex Event Processing GEri



STEP 3:

Proton Authoring Tool

http://<CEP_GE_IP>:8080/AuthoringTool

The screenshot displays the Proton Authoring Tool interface. At the top, there is a toolbar with buttons: Open Project, New, Verify, Save, Save and Export, Import project..., Draw EPN, and Delete Project. On the left, a tree view shows the project structure under 'VotingDEMO', including folders for Events, EPAs, Contexts, Consumers, Producers, and Actions. The 'EPAs' folder is expanded, and 'MaxChangeEPA' is selected. The main workspace shows the configuration for 'MaxChangeEPA' with tabs for General, Event Selection, Participant Events, Computed Variables, Condition, and Derivation. The 'Event Selection' tab is active, showing a table of participant events. Below it, the 'Computed Variables' tab is also visible, showing a table of computed variables. At the bottom, a status bar indicates 'Number of errors and warnings = 0'.

MaxChangeEPA x WinnerEPA x AuctionTemporalContext x EPN x

General

Event Selection

Participant Events Show Required Only Add New

Event	Alias	Condition	Consumption
BidContextUpdate		BidContextUpdate.bid>0	Reuse

Computed Variables Add New Refresh

Name	Aggregation Type	BidContextUpdate Expr
maxval	Max	BidContextUpdate.bid

Condition

Derivation

Number of errors and warnings = 0

Orion Context Broker GEri

Registering the CEP instance

Register CEP to get notification on the changes of AuctionStatus:

```
(curl <IP_of_orionCB>:1026/NGSI10/subscribeContext -s -S --header 'Content-Type: application/xml' -d @- |
xmllint --format -) <<EOF
```

```
<?xml version="1.0"?>
<subscribeContextRequest>
  <entityIdList>
    <entityId type="AuctionStatus" isPattern="false">
      <id>AuctionDEMO</id>
    </entityId>
  </entityIdList>
  <attributeList>
    <attribute>status</attribute>
  </attributeList>
  <reference>http://10.0.8.109:8080/ProtonOnWebServer/rest/events</reference>
  <duration>P1M</duration>
  <notifyConditions>
    <notifyCondition>
      <type>ONCHANGE</type>
      <condValueList>
        <condValue>status</condValue>
      </condValueList>
    </notifyCondition>
  </notifyConditions>
  <throttling>PT6S</throttling>
</subscribeContextRequest>
EOF
```

For CEP the only acceptable subscription format is XML!
Otherwise notifications will not be handled by CEP.

IP of the CEP instance

Response:

```
<?xml version="1.0"?>
<subscribeContextResponse>
  <subscribeResponse>
    <subscriptionId>546e18fba85d637c37b205bc</subscriptionId>
    <duration>P1M</duration>
    <throttling>PT6S</throttling>
  </subscribeResponse>
</subscribeContextResponse>
```

Orion Context Broker GEri

The Orion entity for a situation generated by CEP

url = 'http://<IP_of_orionCB>:1026/NGSI10/updateContext'

payload = {

"contextElements": [

```
{
  "type": "MaxBid",
  "isPattern": "false",
  "id": "MaxBidDEMO",
  "attributes": [
```

```
{
  "name": "Name",
  "type": "string",
  "value": "0"
```

```
,
{
  "name": "Certainty",
  "type": "percentage",
  "value": "0"
```

```
,
{
  "name": "OccurrenceTime",
  "type": "date",
  "value": "0"
```

```
,
{
  "name": "DetectionTime",
  "type": "date",
  "value": "0"
```

```
],
```

```
{
  "name": "Duration",
  "type": "double",
  "value": "0.0"
```

```
,
{
  "name": "Cost",
  "type": "double",
  "value": "0.0"
```

```
,
{
  "name": "EventId",
  "type": "string",
  "value": "0"
```

```
,
{
  "name": "maxbid",
  "type": "integer",
  "value": "0"
```

```
],
```

```
{
  "name": "maxentity",
  "type": "string",
  "value": "0"
```

```
},
```

```
]
```

```
,
"updateAction": "APPEND"
}
```

```
headers = { 'Accept' : 'application/json',
            'content-type' : 'application/json'
          }
```

```
r = requests.post(url,
data=json.dumps(payload),
headers=headers)
```

Attributes defined by us...

All the situations generated by CEP have these attributes.

How to deploy a CEP project?

1. *export it to the external repository*

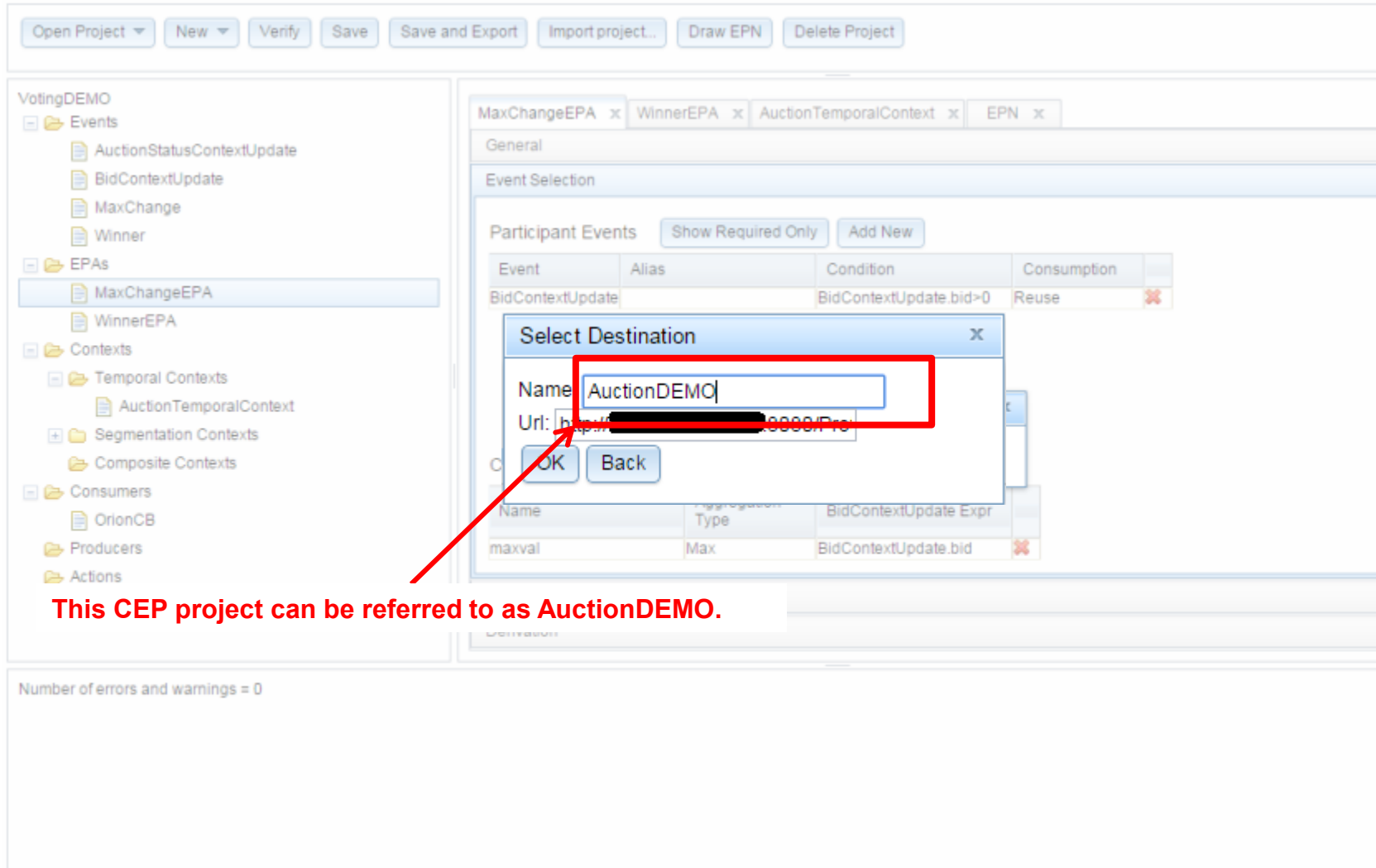
The screenshot displays the FIWARE CEP IDE interface. At the top, a toolbar contains buttons: 'Open Project', 'New', 'Verify', 'Save', 'Save and Export' (highlighted with a red box), 'Import project...', 'Draw EPN', and 'Delete Project'. A red arrow points from the 'Save and Export' button to the text 'First click on Save and Export'. On the left, a project tree for 'VotingDEMO' shows folders for 'Events', 'Contexts', 'Consumers', 'Producers', and 'Actions', with sub-items like 'AuctionStatusContextUpdate', 'BidContextUpdate', 'MaxChange', 'MaxChangeEPA', 'WinnerEPA', 'Temporal Contexts', 'AuctionTemporalContext', 'Segmentation Contexts', 'Composite Contexts', 'OrionCB', and 'Actions'. The main workspace shows a 'General' tab with 'Event Selection' and 'Participant Events' sections. A 'Select Destination' dialog box is open, showing 'Download file' and 'Export to external repository' buttons, with the latter highlighted by a red box and a red arrow pointing to the text 'And then on Export to external repository'. The dialog also shows a table of 'Computed Variables' with columns 'Name', 'Aggregation Type', and 'BidContextUpdate Expr'. The table contains one row: 'maxval', 'Max', and 'BidContextUpdate.bid'. The bottom status bar indicates 'Number of errors and warnings = 0'.

First click on Save and Export

And then on Export to external repository

How to deploy a CEP project?

1. *export it to the external repository*



The screenshot displays the FIWARE CEP project interface. On the left, a tree view shows the project structure for 'VotingDEMO', including Events, EPAs, Contexts, Consumers, Producers, and Actions. The 'MaxChangeEPA' is selected under EPAs. The main panel shows the 'General' tab for 'MaxChangeEPA', with an 'Event Selection' section. A 'Participant Events' table is visible, showing 'BidContextUpdate' with a condition 'BidContextUpdate.bid>0' and a consumption of 'Reuse'. A 'Select Destination' dialog box is open, with the 'Name' field set to 'AuctionDEMO'. A red arrow points from the 'AuctionDEMO' text in the dialog to a red text box below the interface.

This CEP project can be referred to as AuctionDEMO.

How to deploy a CEP project?

2. the (project) definition is available in the repo.

■ HTTP GET request for receiving the stored project definitions

- GET <IP_of_CEP_Instance>:8080/ProtonOnWebServerAdmin/resources/definitions

■ Response

```
[  
  {  
    "name": "/opt/repositories/tomcat10/DoSAttack.json",  
    "url": "/ProtonOnWebServerAdmin/resources/definitions/DoSAttack"  
  },  
  {  
    "name": "/opt/repositories/tomcat10/DoSAttack2.json",  
    "url": "/ProtonOnWebServerAdmin/resources/definitions/DoSAttack2"  
  },  
  {  
    "name": "/opt/repositories/tomcat10/Test.json",  
    "url": "/ProtonOnWebServerAdmin/resources/definitions/Test"  
  },  
  {  
    "name": "/opt/repositories/tomcat10/AuctionDEMO.json",  
    "url": "/ProtonOnWebServerAdmin/resources/definitions/AuctionDEMO"  
  }  
]
```

How to deploy a CEP project?

3. *Activate AuctionDEMO*

- HTTP PUT request sent to the following end-point

- PUT <IP_of_CEP_Instance>:8080/ProtonOnWebServerAdmin/resources/instances/ProtonOnWebServer

- **Content:**

```
{  
    "action": "ChangeDefinitions",  
    "definitions-url": "/ProtonOnWebServerAdmin/resources/definitions/AuctionDEMO"  
}
```

- **Response** : 200 OK

How to deploy a CEP project?

4. *Start and stop AuctionDEMO*

Start the CEP instance

- HTTP PUT request sent to the following end-point
 - PUT <IP_of_CEP_Instance>:8080/ProtonOnWebServerAdmin/resources/instances/ProtonOnWebServer
 - **Content:**

```
{"action":"ChangeState","state":"start"}
```
 - **Response** : 200 OK
-

Stop the CEP instance

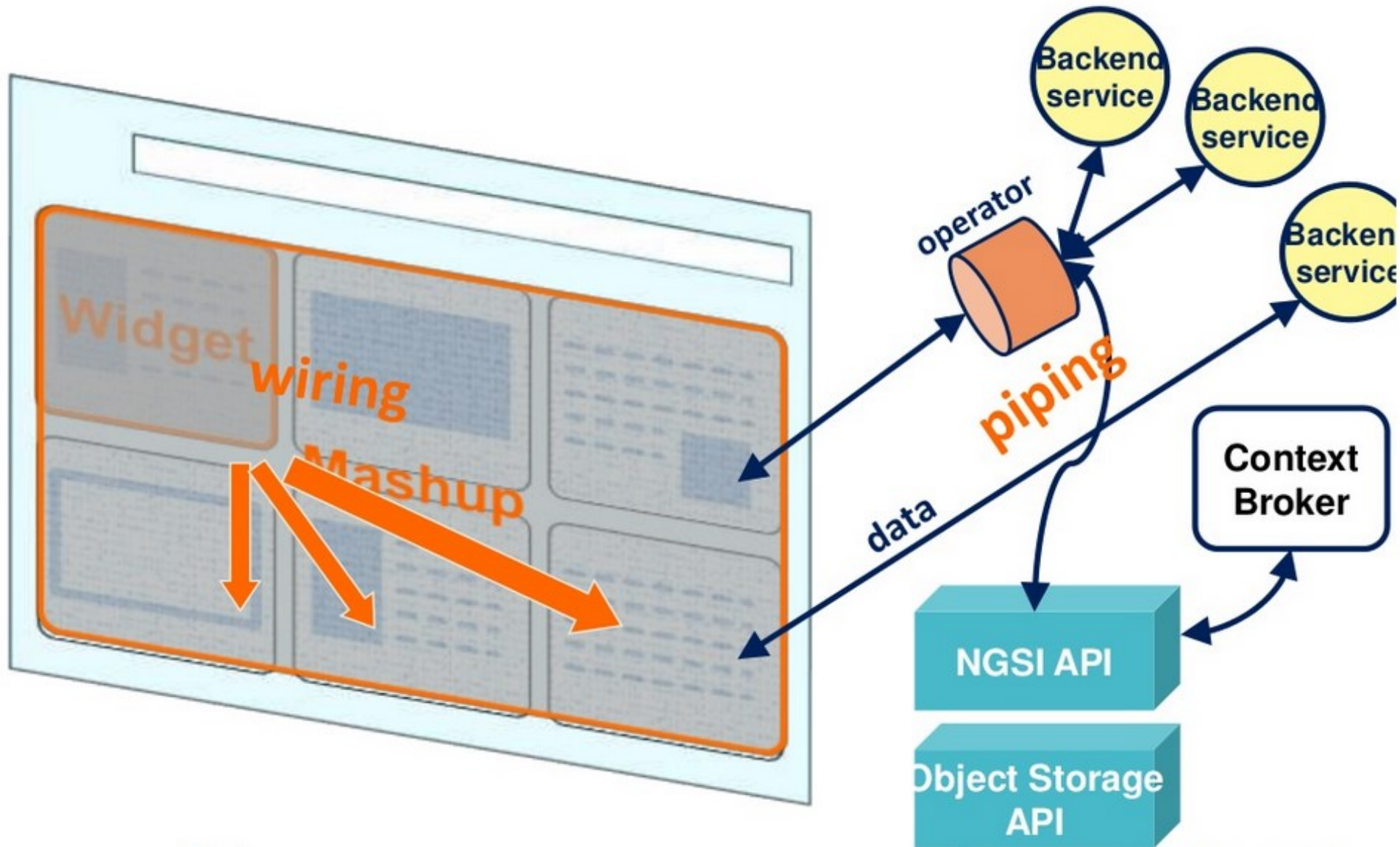
- HTTP PUT request sent to the following end-point
 - PUT <IP_of_CEP_Instance>:8080/ProtonOnWebServerAdmin/resources/instances/ProtonOnWebServer
 - **Content:**

```
{"action":"ChangeState","state":"stop"}
```
 - **Response** : 200 OK
-

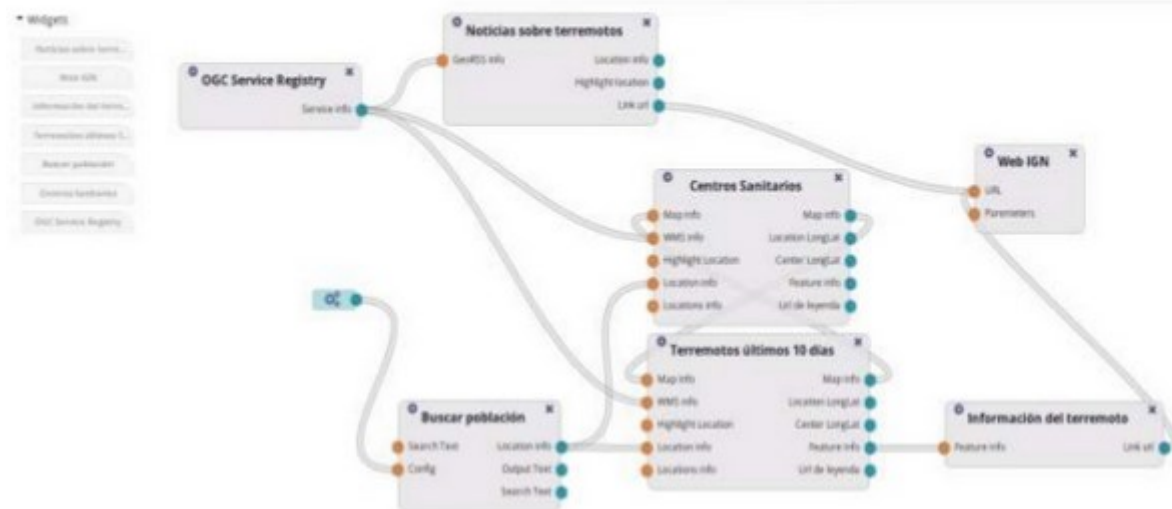
Get status of the instance

GET <IP_of_CEP_Instance>:8080/ProtonOnWebServerAdmin/resources/instances/ProtonOnWebServer

WireCloud Web Mashup GEri



WireCloud Web Mashup GEri



How to create a WireCloud Widget...

Main.js

```
var InputBox = function InputBox() {
  this.textPreference = "";
  MashupPlatform.widget.context.registerCallback(function (newValues) {
    if ("heightInPixels" in newValues) {
      repaint.call(this);
    }
  }).bind(this));
  MashupPlatform.prefs.registerCallback(handlerPref.bind(this));
  // Input callbacks:
  MashupPlatform.wiring.registerCallback("inputMsg", handlerInputMsg.bind(this));
};
```

Index.html

```
<html>
<head>
  <meta http-equiv="Content-Type,,
    content="text/html; charset=UTF-8">
  <title>TextBox</title>
  <!-- CSS -->
  <link id="css_style" href="css/style.css"
    type="text/css" rel="stylesheet" />
  <link id="css_theme" type="text/css" rel="stylesheet" />
  <!-- JS -->
  <script type="text/javascript" src="js/widget/TextBox.js">
  </script>
  <script type="text/javascript" src="js/widget/main.js">
  </script>
</head>
<body>
</body>
</html>
```

Config.xml

```
<Template
xmlns="http://wirecloud.conwet.fi.upm.es/ns/template#">
  <Catalog.ResourceDescription>
    <Vendor>CoNWeT</Vendor>
    <Name>output-box</Name>
    <DisplayName>Output Box</DisplayName>
    <Author>sblanco</Author>
    <Version>1.0</Version>
    <Mail>sblanco@conwet.com</Mail>
    <Description>It sends a message written...</Description>
  </Catalog.ResourceDescription>
  <Platform.Wiring>
    <InputEndpoint name="inputMsg" type="text" label="Msg"
      description="We get the message"
      friendcode="textMsg"/>
  </Platform.Wiring>
</Template>
```

...or a WireCloud Operator...

```
<?xml version='1.0' encoding='UTF-8'?>
<operator xmlns="http://wirecloud.conwet.fi.upm.es/ns/macdescription/1"
vendor="CoNWeT" name="ngsidentity2table" version="1.0">
  <details>
    <title>NGSI Entity To Table Data</title>
    <authors>wigner</authors>
    <image>images/catalogue.png</image>
    <description>Convert NGSI entities coming from...</description>
  </details>
  <wiring>
    <outputendpoint name="tableOutput" type="text" label="Data"
      description="Transformed data from the received entity" friendcode="data"/>
    <inputendpoint name="entityInput" type="text" label="Entity"
      description="Received entity will be transform to a Table" friendcode="entity"/>
  </wiring>
  <scripts>
    <script src="js/main.js"/>
  </scripts>
</operator>
```

Config.xml

Main.js

```
MashupPlatform.wiring.registerCallback("entityInput", function (entityString) {
  var entity = JSON.parse(entityString);
  var structure = [];
  var structureString = [];
  var dataString = [];
  for (var key in entity) {
    structure.push(key);
    structureString.push("{\"id\":\""+key+"\"}");
    dataString.push("\""+key+"\":\""+entity[key]+"\"");
  }
  structure = "\"structure\":["+structureString.join()+"]";
  var data = "\"data\":["+dataString.join()+"]";
  var output = "{"+structure+"",""+data+"}";
  MashupPlatform.wiring.pushEvent("tableOutput", output);
});
```

What can you do with FIWARE?



350 connected sensors for the management of public resources through a single connectivity platform and enhancing various areas such as **transport, energy efficiency and environmental services**



24.10.2013 12:45	
Supermarkt#	
Luxemburger Straße 2 50675 Köln Bertin n. d. Gasse	
Mo - Sa 08.00 - 20.00	
Lieferung 24.10.2013	
Milch	30%
Tomaten	30%
Cherry	30%
28.10.2013	
Frische Schinken Bio-Milch	15%
28.10.2013	
Milch	20%
Mozzarella	20%
28.10.2013	
Milch	30%
Joghurt	30%
31.10.2013	
Milch	50%
Naturschokolade	50%
30.10.2013	
Milch	40%
Joghurt mild	40%
(verschiedene Sorten)	40%
30.10.2013	
Amerikaner	30%
US-Rib-Eye-Steak	30%
28.10.2013	



HELP TAXI DRIVERS



REAL-TIME FORECAST

Success Stories? Get inspiration!



Building Streaming Media Applications with FI-WAR...

41 visualizaciones Hace 1 mes



Creating Interactive 3D Applications and Shared...

20 visualizaciones Hace 1 mes



The FI-Content use case project - Social and...

14 visualizaciones Hace 1 mes



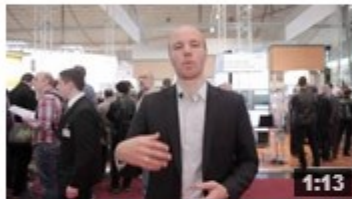
Naevatec (Kurento Generic Enabler) - Interview with...

23 visualizaciones Hace 1 mes



Magic Box Interview with Javier Herrero at CeBIT...

27 visualizaciones Hace 1 mes



Thorsten Sandfuchs (SAP) interview at CeBIT 2014

22 visualizaciones Hace 1 mes



FI-WARE Business Framework by Thorsten...

24 visualizaciones Hace 1 mes



Connecting to the Internet of Things (IoT) with FI...

23 visualizaciones Hace 1 mes



Connecting to the Internet of Things (IoT) with FI...

12 visualizaciones Hace 1 mes



FI-WARE and FI-Content in the EU Future Internet...

33 visualizaciones Hace 1 mes



Magic Box & FI-WARE, Javier Herrero at CeBIT...

15 visualizaciones Hace 1 mes



CeBIT Future Talks 2014

15 visualizaciones Hace 1 mes



SmartTaxi: The Value for a Developer - FI-WARE in...

119 visualizaciones Hace 2 meses



FI-LAB: A guide to the most exciting Future Internet...

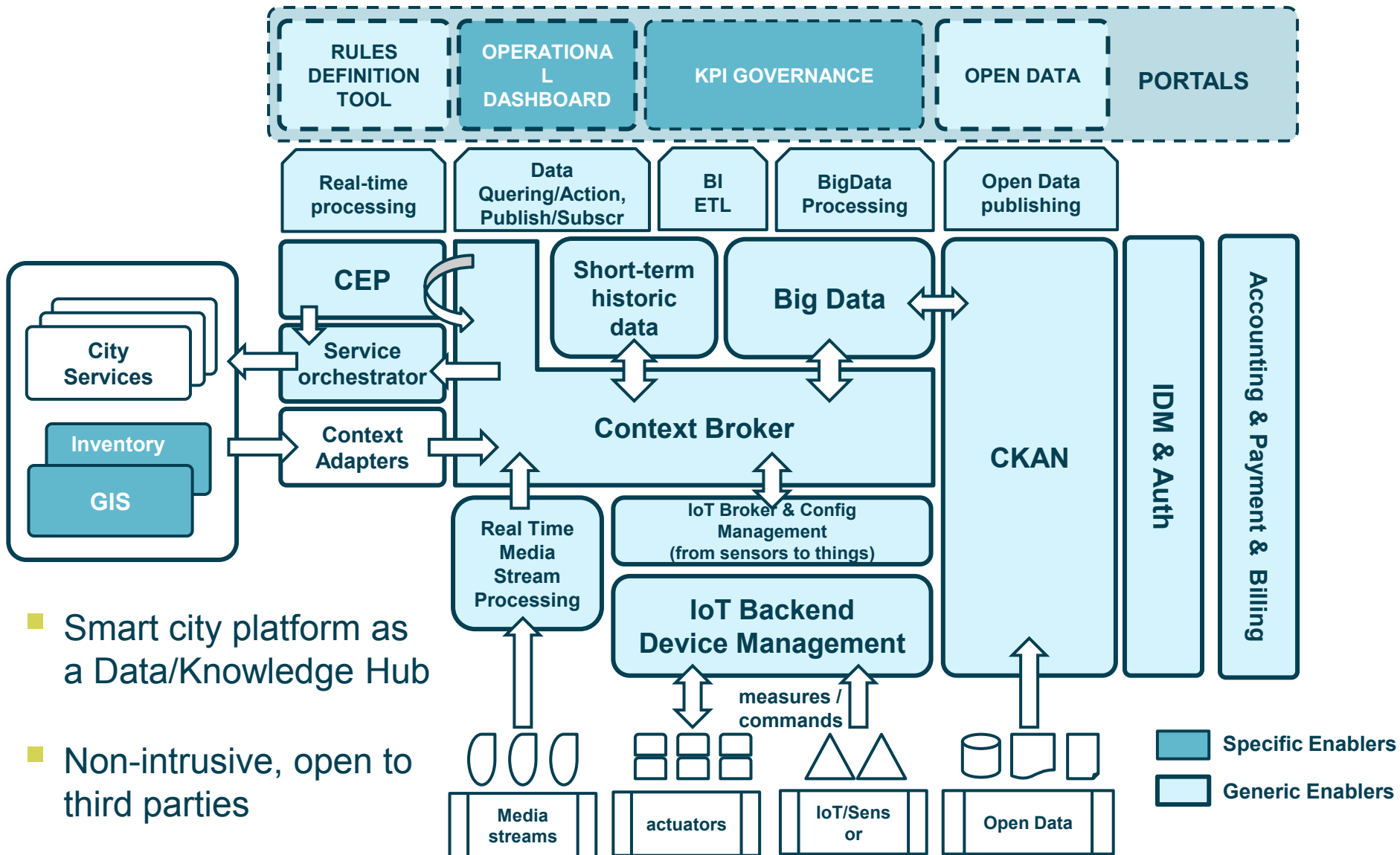
133 visualizaciones Hace 2 meses



FI-OPS: access from everywhere in Europe by...

20 visualizaciones Hace 2 meses

Example: Smart City platform



Join us!

<http://fiware.org>

<http://lab.fiware.org>

Follow @Fiware on Twitter !

