Four key factors to design a Web of things architecture

Francesco Bruni[1] - Claudio Pomo[2] - Gaetano Murgolo[3]

ICWE - EnWoT - Rome June ‘17

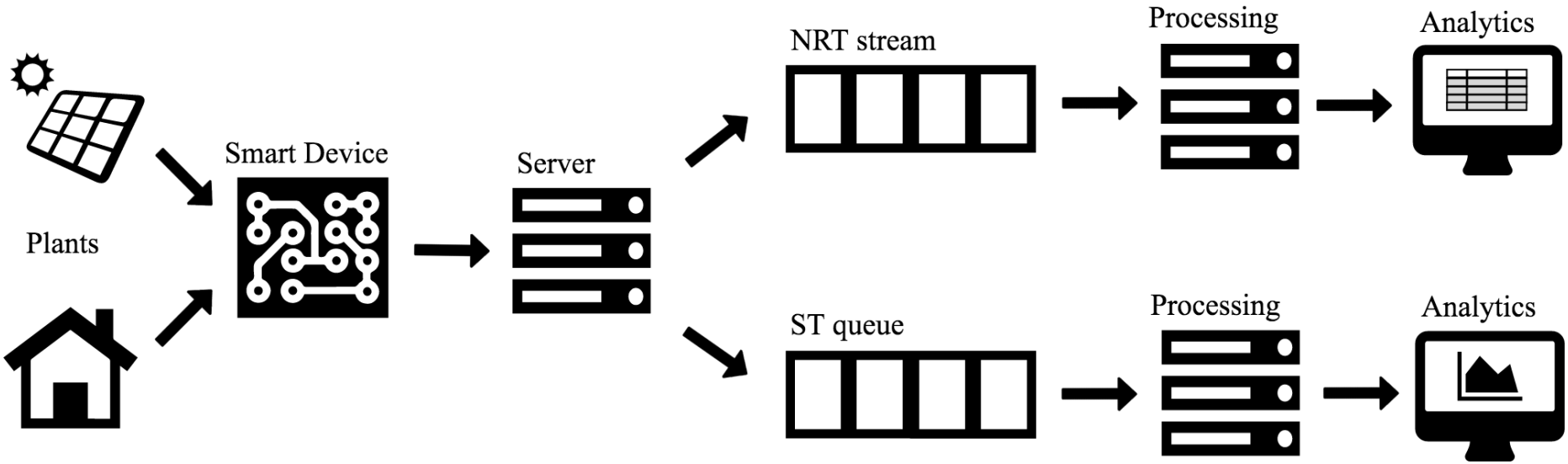
[1]Planetek Italia
[2]Polytechnic University of Bari

Outline

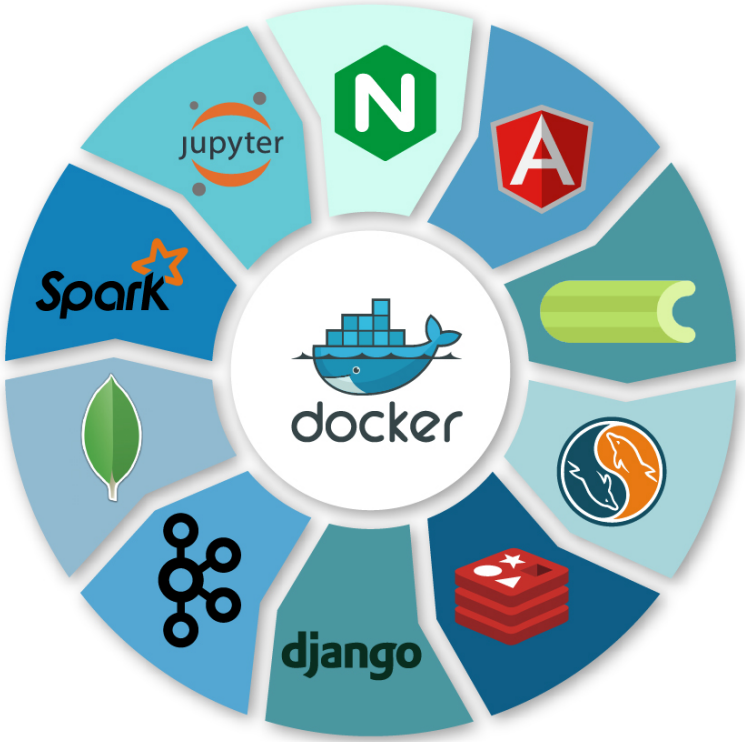
- The scenario
- The architecture
- The factors
  - Scalability
  - Per-user knowledge
  - Reactivity
- Conclusions

The scenario

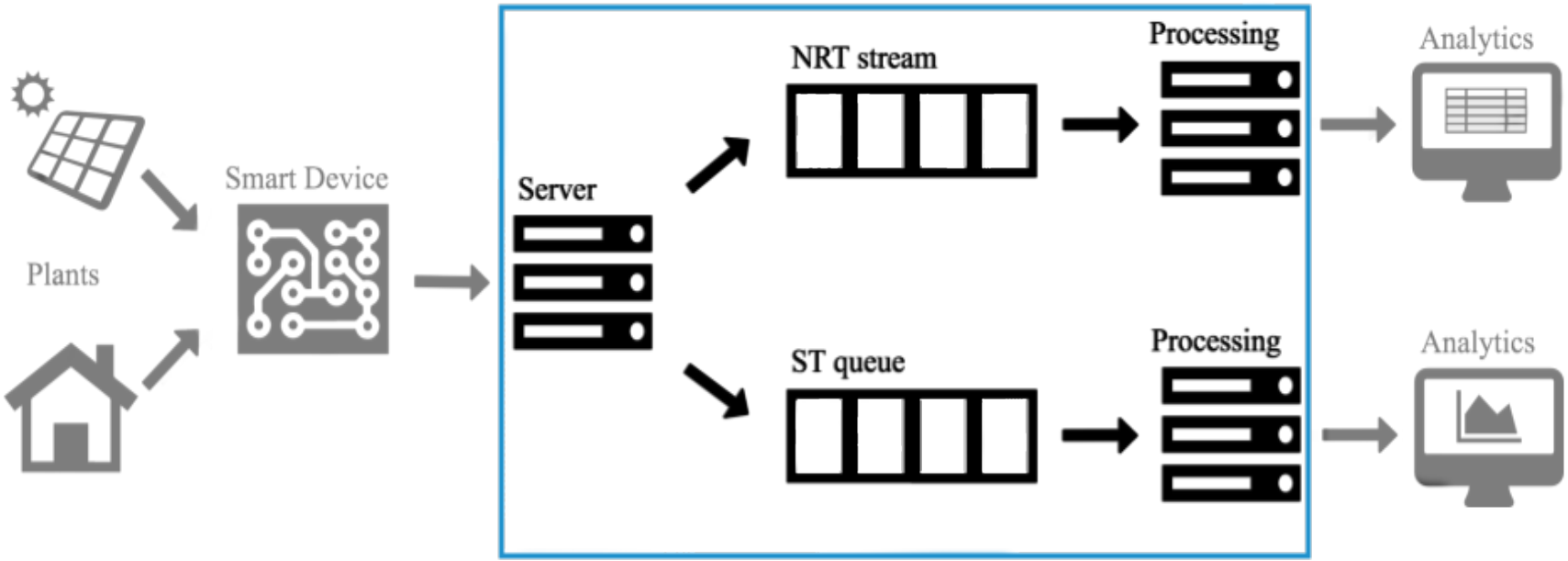
The architecture



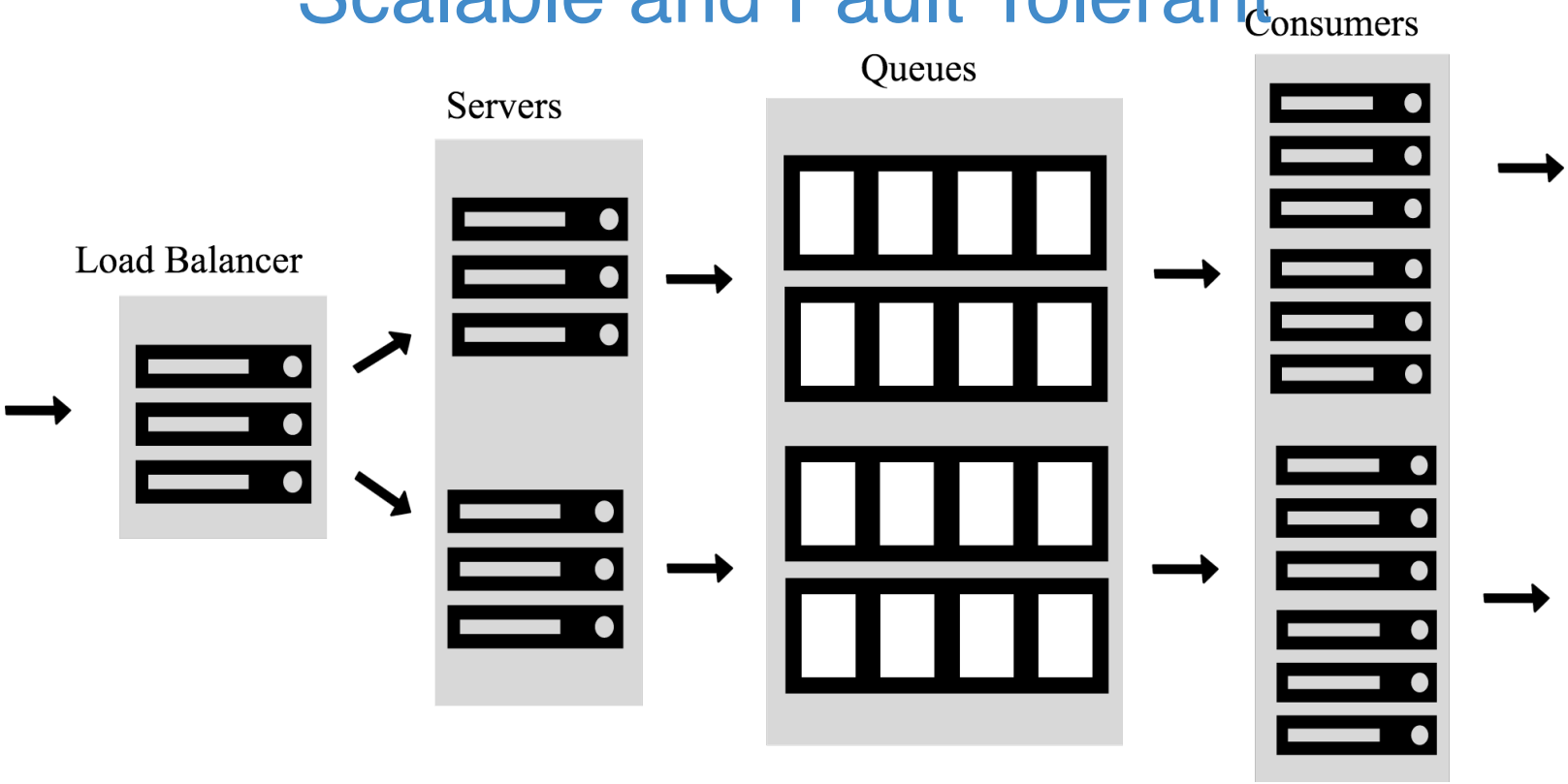
Multiple services



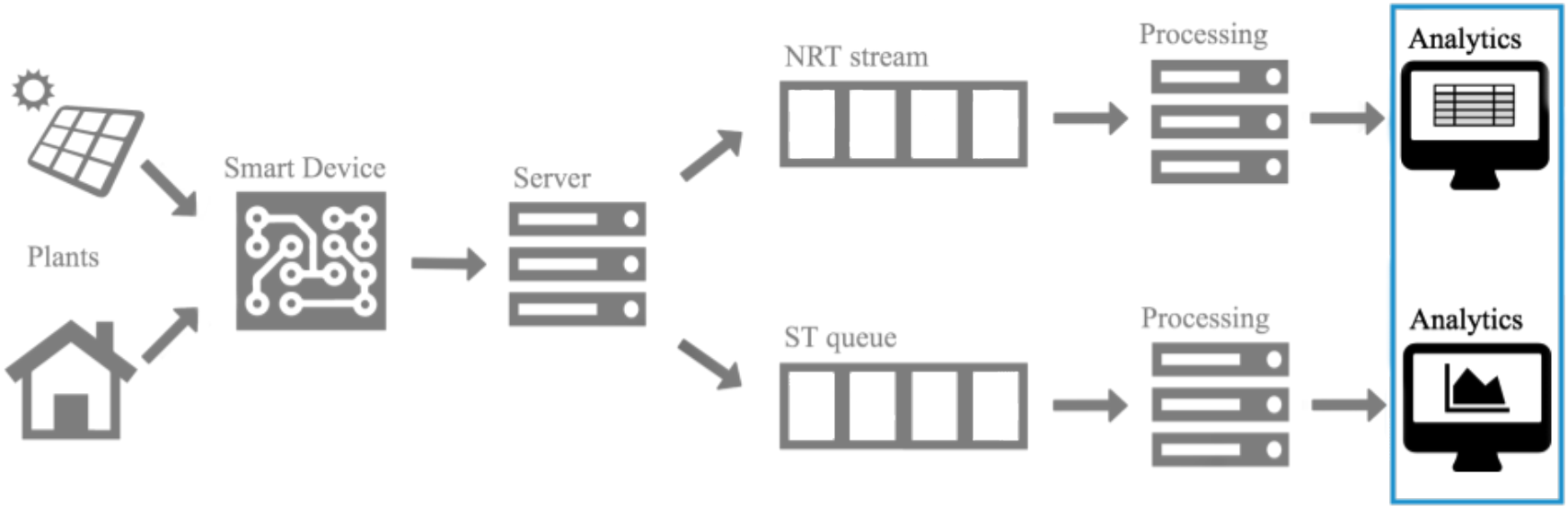
Scalable and Fault Tolerant



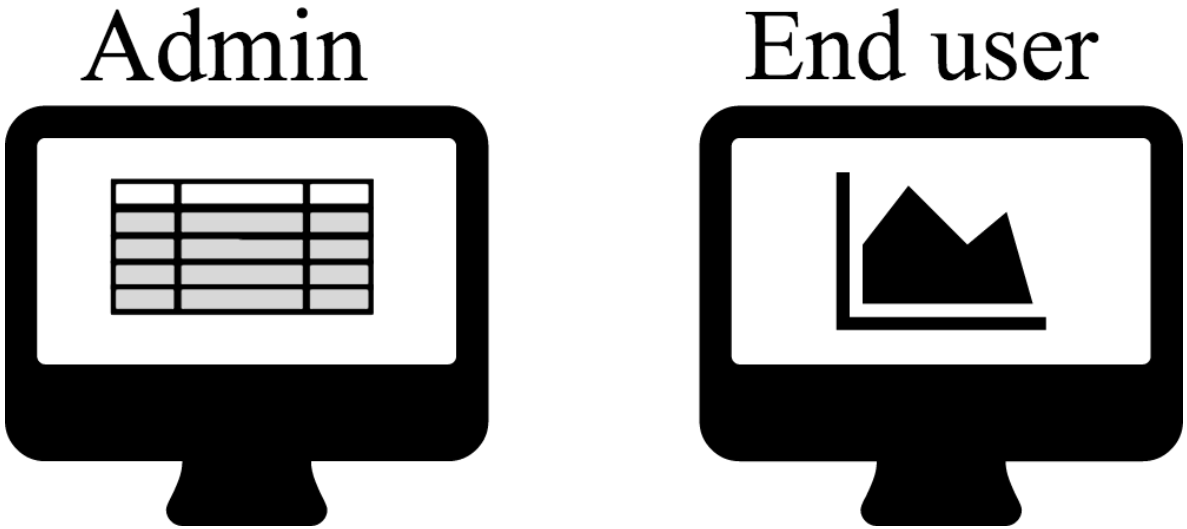
Scalable and Fault Tolerant



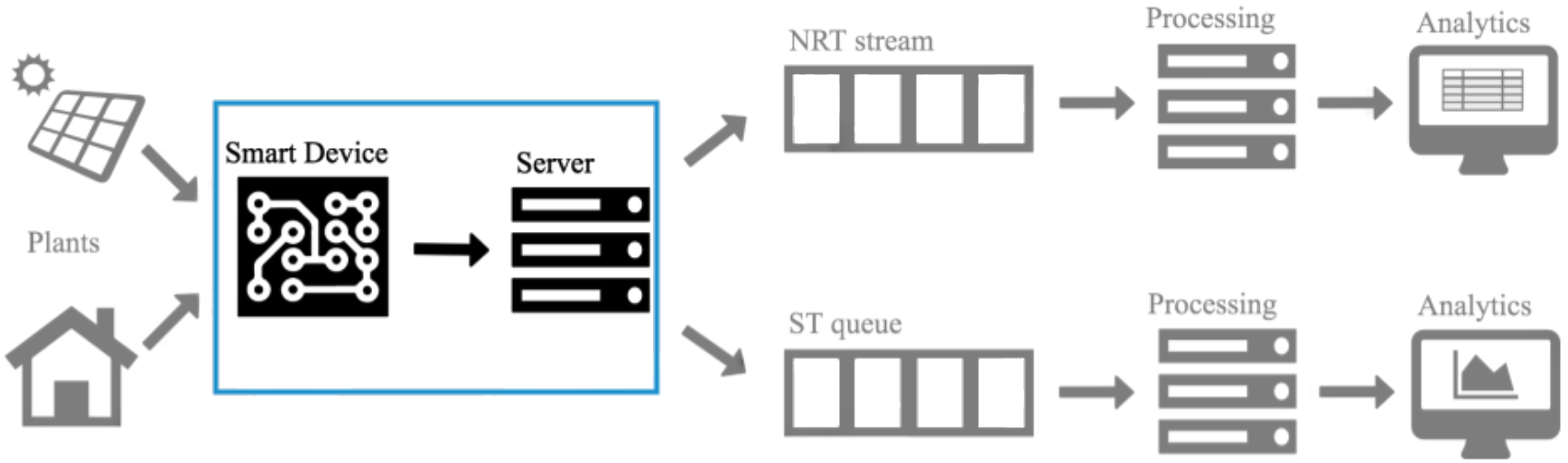
Per-user knowledge



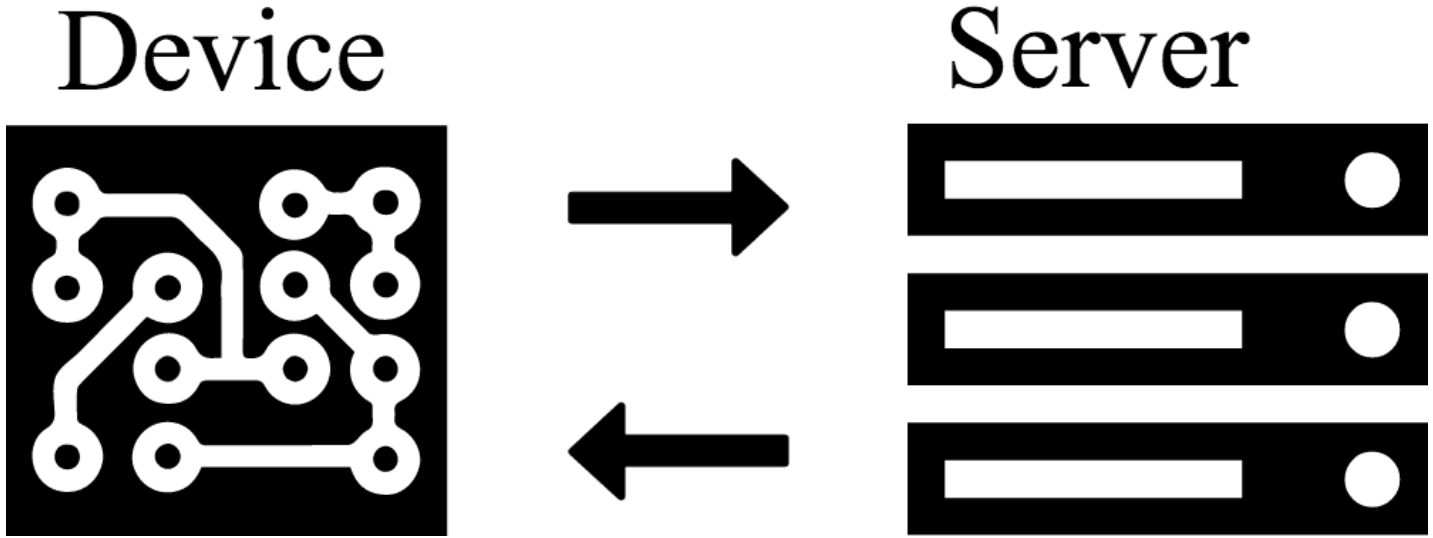
Per-user knowledge



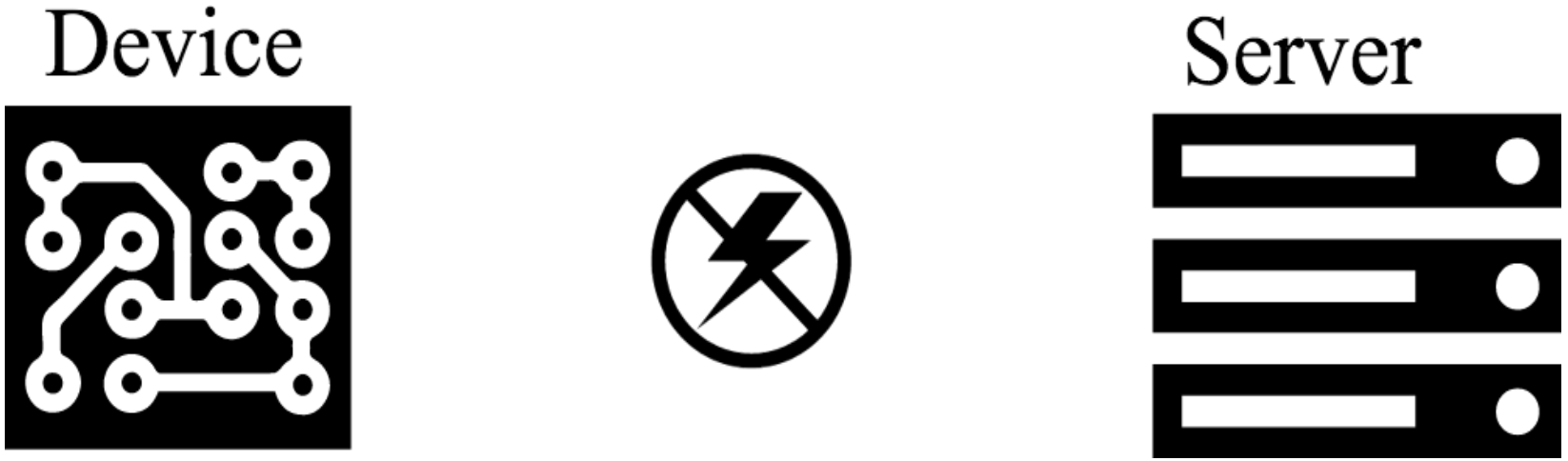
Reactive and adapting system



Reactive and adapting system



Handling anomalies



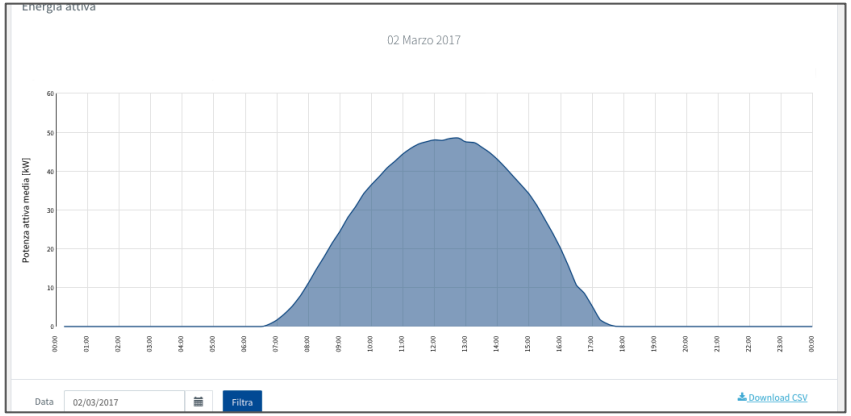
Deployed environment

In [9]:

```
pdf['hour'] = pdf['hour'].map(lambda item: int(item))
pdf.sort_values('hour')
pdf.set_index('hour')
```

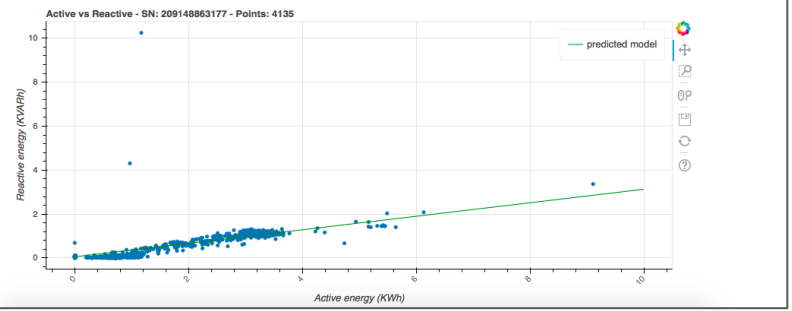
Out[9]:

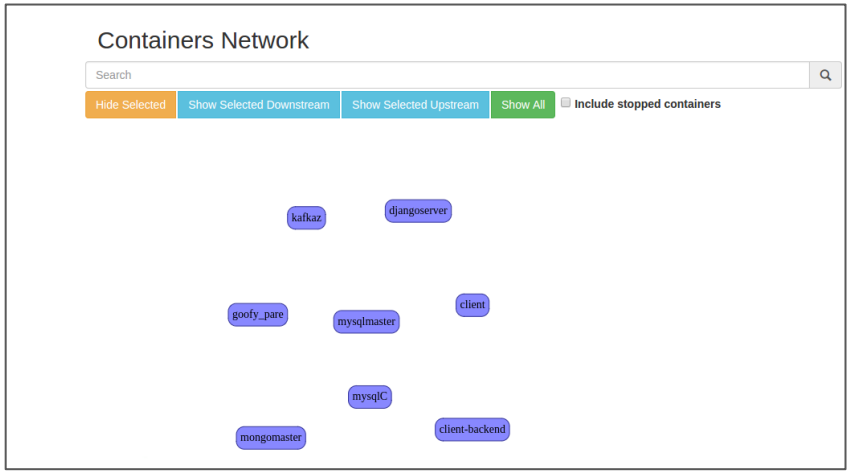
	min(WHRL-MEAN)	avg(WHRL-MEAN)	max(WHRL-MEAN)	stddev_samp(WHRL-MEAN)	count(WHRL-MEAN)
hour					
7	0.009333	0.450333	1.361000	0.492923	17
15	0.003000	1.221301	4.401667	0.923636	174
11	0.003333	1.178847	2.842667	0.920330	213
3	0.013667	0.014127	0.017000	0.000641	184
8	0.007667	1.037897	4.139667	0.852202	71
22	0.013667	0.014793	0.017000	0.000925	227
16	0.000667	0.876853	4.524000	1.096655	301
0	0.013667	0.014173	0.017000	0.000601	185



In [8]:

```
import numpy as np
values = np.linspace(0, 10, 4000)
p = figure(width=900, height=400, title="Active vs Reactive - SN: 209148863177 - Points: %s" %len(act_react['WHRL-MEAN'
p.circle(x=act_react['WHRL-MEAN'], y=act_react['FVARHRL-MEAN'])
p.yaxis.axis_label = "Reactive energy (KVARh)"
p.xaxis.axis_label = "Active energy (KWh)"
p.line(x=values, y = lrModel.coefficients[0]*values + lrModel.intercept, color="green", legend="predicted model")
p.xaxis.major_label_orientation = pi/4
show(p)
```





Conclusions

- Raw sensed data into a market valuable knowledge
- 4 key factors driven architecture design
  - Being scalable
  - User oriented metrics and control
  - Adapt to external inputs
  - Notify connectionless/non working device

Future works

- Microservices architecture
- Websocket connection
- Data based anomalies detection
- Harvesting multiple data sources

Questions?