Deploying Fog applications: How much does it cost btw?

Antonio Brogi, Stefano Forti and Ahmad Ibrahim

Service-oriented, Cloud and Fog Computing Research Group
Department of Computer Science
University of Pisa, Italy

Life on the edge

The era of the cloud’s total dominance is drawing to a close

The rise of the “Internet of things” is one reason why computing is emerging from the centralised cloud and moving to an “edge” of networks and intelligent devices.

Blowing away the cloud

Connected devices, worldwide, bn

- Internet of things*
- Mobile phones
- PC, laptop & tablet
- Fixed phones

Source: Ericsson

* Everyday objects connected to the internet

Economist.com
Fog Computing

• The Cloud is here to stay but it cannot support the IoT momentum alone.
• There is a need for filtering and processing before the Cloud.
Deploying to the Fog

Application with hw/sw, QoS and IoT requirements

Infrastructure with available hw/sw and (varying) QoS

Deployer’s desiderata and constraints

Processing should occur wherever it is best-placed for any given IoT application.
Deployment cost

- Financial considerations will influence deployment choice.
- Cloud ≈ few large providers
  Fog ≈ many small to medium players
- Pricing strategies for Fog computing are an open research challenge.
Related Work

• Literature focus either on
  - Cloud
    e.g., (Dìaz et al., 2017)
    *pay-per-use, or
    *subscription-based
  - IoT
    e.g., (Niyato et al., 2016), (Markus et al., 2017)
    *type and number of sensors,
    *number of data requests

We extend Cloud cost models to the Fog scenario and integrate them with costs from the IoT.
Cost model

Determine **cost-effective** deployments

Design **billing/pricing** of new services

Estimate **revenues and outflows** beforehand
SmartBuild

DataStorage

IoTController

Dashboard

fire_sensor
lights_control
thermostate
weather_station
videocamera

Sat.
3G/4G
VDSL
Stakeholders’ desiderata

System Integrators

sell the deployed solution for €1,500 monthly
plan to **spend at most** €850 / month
to keep it up and running

Customers

want a guarantee of the app being up and running fine at least **98% of the time**
Concretely

**Is there any deployment that complies with QoS requirements 98% of the times and doesn’t exceed €850?**

**How can free Fog resources be maximised?**

**After upgrading to 4G (+€20), will the system integrators gain more money?**
Our Solution

**Modelling** of IoT apps, Fog infrastructures

**Algorithms** to determine deployments

**Cost model** for IoT+Fog+Cloud scenarios
Application model

DataStorage

IoTController

Dashboard

Software: Linux, PHP
Hardware: small

<140 ms, (0.4 Mbps, 0.9 Mbps)>
<100 ms, (0.3 Mbps, 1.5 Mbps)>

Software: Linux, SQL
Hardware: large

<160 ms, (0.5 Mbps, 3.5 Mbps)>
<140 ms, (0.4 Mbps, 0.9 Mbps)>

Software: Linux
Hardware: tiny

fire_sensor: 1 invoke/min, 100 ms, (0.1, 0.5) Mbps
lights_control: 3 invokes/h, 200 ms, (0.9, 1) Mbps
thermostat: 2 invokes/h, 2 s, (0.1, 0.1) Mbps
weather_station: 5 invoke/day, 5 s, (0.1, 0.5) Mbps
video_camera: stream, 50 ms, (0.1, 5) Mbps

Software: Linux

<table>
<thead>
<tr>
<th>VM Type</th>
<th>vCPUs</th>
<th>RAM (GB)</th>
<th>HDD (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>tiny</td>
<td>1</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>small</td>
<td>1</td>
<td>2</td>
<td>20</td>
</tr>
<tr>
<td>medium</td>
<td>2</td>
<td>4</td>
<td>40</td>
</tr>
<tr>
<td>large</td>
<td>4</td>
<td>8</td>
<td>80</td>
</tr>
<tr>
<td>xlarge</td>
<td>8</td>
<td>16</td>
<td>160</td>
</tr>
</tbody>
</table>
Infrastructure model

<table>
<thead>
<tr>
<th>Dash Type</th>
<th>Profile</th>
<th>Latency</th>
<th>Download</th>
<th>Upload</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satellite 14M</td>
<td>40 ms</td>
<td>98%: 10.5 Mbps 2%: 0 Mbps</td>
<td>98%: 4.5 Mbps 2%: 0 Mbps</td>
<td></td>
</tr>
<tr>
<td>3G</td>
<td>54 ms</td>
<td>99.6%: 9.61 Mbps 0.4%: 0 Mbps</td>
<td>99.6%: 2.89 Mbps 0.4%: 0 Mbps</td>
<td></td>
</tr>
<tr>
<td>4G</td>
<td>53 ms</td>
<td>99.3%: 22.67 Mbps 0.7%: 0 Mbps</td>
<td>99.4%: 16.97 Mbps 0.6%: 0 Mbps</td>
<td></td>
</tr>
<tr>
<td>VDSL</td>
<td>60 ms</td>
<td>99%: 60 Mbps</td>
<td>99%: 6 Mbps</td>
<td></td>
</tr>
<tr>
<td>Fibre</td>
<td>5 ms</td>
<td>99.9%: 1000 Mbps</td>
<td>99.9%: 1000 Mbps</td>
<td></td>
</tr>
<tr>
<td>WLAN</td>
<td>15 ms</td>
<td>99%: 32 Mbps  10%: 16 Mbps</td>
<td>99%: 32 Mbps  10%: 16 Mbps</td>
<td></td>
</tr>
</tbody>
</table>

Costs per service:

- fire_sensor_1: €0.01 per invoke
- lights_control_1: €0.3 per invoke
- thermostate_1: €0.01 per invoke
- videocamera_1: €30 per month
- fire_sensor_2: €0 per invoke
- lights_control_2: €0 per invoke
- thermostate_2: €0 per invoke
- videocamera_2: €0 per month
- weather_station_3: €0.2 per invoke
Determine a deployment?

**Software:** Linux, PHP

**Hardware:** small

- CPUs: 4
- RAM: 12 GB
- HDD: 128 GB

**Software:** Linux, PEP

**Hardware:** small

- CPUs: 2
- RAM: 4 GB
- HDD: 32 GB

**Software:** Linux

**Hardware:** tiny

- CPUs: 2
- RAM: 2 GB
- HDD: 32 GB

---

**Fire Sensor 1:** €0.01 per invoke

**Lights Control 1:** €0.03 per invoke

**Thermostat 1:** €0.01 per invoke

**Video Camera 1:** €30 per month

**Fire Sensor 2:** €0 per invoke

**Lights Control 2:** €0 per invoke

**Thermostat 2:** €0 per invoke

**Video Camera 2:** €0 per month

**Weather Station 3:** €0.2 per invoke
VM Cost

\[ p(H, n) = \begin{cases} 
  c(H, n) & \text{if } H \text{ is a default VM} \\
  \sum_{\rho \in R} [H.\rho \times c(\rho, n)] & \text{if } H \text{ is an on-demand VM}
\end{cases} \]

Default VM:
tiny, small, medium, ...

On Demand VM:
\(<RAM: 16 \text{ GB}, CPU: 8, HDD: 1 \text{ TB}>\)

<table>
<thead>
<tr>
<th>Hardware</th>
<th>resource</th>
<th>monthly cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>€ 4.0 /core</td>
<td></td>
</tr>
<tr>
<td>RAM</td>
<td>€ 6.0 /GB</td>
<td></td>
</tr>
<tr>
<td>HDD</td>
<td>€ 1.0 /GB</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VMs</th>
<th>VM type</th>
<th>monthly cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>tiny</td>
<td>€ 7.0</td>
</tr>
<tr>
<td></td>
<td>small</td>
<td>€ 25.0</td>
</tr>
<tr>
<td></td>
<td>medium</td>
<td>€ 50.0</td>
</tr>
<tr>
<td></td>
<td>large</td>
<td>€ 100.0</td>
</tr>
<tr>
<td></td>
<td>xlarge</td>
<td>€ 200.0</td>
</tr>
</tbody>
</table>
Software Cost

Software bundle: containing a pre-assembled suite

Per software license: e.g., RH Linux, SQL Server

\[ p(S, n) = \begin{cases} 
    c(S, n) & \text{if } S \text{ is a bundle} \\
    \sum_{s \in S} c(s, n) & \text{if } S \text{ is on-demand}
\end{cases} \]

<table>
<thead>
<tr>
<th>Software</th>
<th>name</th>
<th>monthly cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>linux</td>
<td>€ 50.0</td>
</tr>
<tr>
<td></td>
<td>PHP</td>
<td>€ 0.0</td>
</tr>
<tr>
<td></td>
<td>SQL</td>
<td>€ 60.0</td>
</tr>
<tr>
<td></td>
<td>java</td>
<td>€ 0.0</td>
</tr>
</tbody>
</table>
IoT Cost

Invocations bundle:
e.g., 5000 invocations at € 5/month

Per invocation (Thing usage + data transfer):
e.g., € 0.01 per invocation

\[
p(T, t) = \begin{cases} 
  c(T, t) & \text{if } T \text{ is subscription based} \\
  T \cdot k \times c(t) & \text{if } T \text{ is pay-per-invocation}
\end{cases}
\]

weather_station_3: € 0.2 per invoke
<table>
<thead>
<tr>
<th>Resource</th>
<th>Monthly Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>€ 2.0 /core</td>
</tr>
<tr>
<td>RAM</td>
<td>€ 3.0 /GB</td>
</tr>
<tr>
<td>HDD</td>
<td>€ 1.0 /GB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource</th>
<th>Monthly Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux</td>
<td>€ 0.0</td>
</tr>
<tr>
<td>PHP</td>
<td>€ 0.0</td>
</tr>
<tr>
<td>SQL</td>
<td>€ 45.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource</th>
<th>Monthly Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>€ 4.0 /core</td>
</tr>
<tr>
<td>RAM</td>
<td>€ 6.0 /GB</td>
</tr>
<tr>
<td>HDD</td>
<td>€ 1.0 /GB</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VM Type</th>
<th>Monthly Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tiny</td>
<td>€ 7.0</td>
</tr>
<tr>
<td>Small</td>
<td>€ 25.0</td>
</tr>
<tr>
<td>Medium</td>
<td>€ 50.0</td>
</tr>
<tr>
<td>Large</td>
<td>€ 100.0</td>
</tr>
<tr>
<td>XLarge</td>
<td>€ 200.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resource</th>
<th>Monthly Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linux</td>
<td>€ 3.0</td>
</tr>
<tr>
<td>PHP</td>
<td>€ 0.0</td>
</tr>
<tr>
<td>SQL</td>
<td>€ 60.0</td>
</tr>
<tr>
<td>Java</td>
<td>€ 0.0</td>
</tr>
</tbody>
</table>

Fog 1
CPUs: 2
RAM: 4 GB
HDD: 32 GB

Fog 2
CPUs: 2
RAM: 2 GB
HDD: 32 GB

Fog 3
CPUs: 4
RAM: 12 GB
HDD: 128 GB

Cloud 1

Cloud 2

Fire sensor_1: € 0.01 per invoke
Lights control_1: € 0.03 per invoke
Thermostat_1: € 0.01 per invoke
Video camera_1: € 30 per month

Fire sensor_2: € 0 per invoke
Lights control_2: € 0 per invoke
Thermostat_2: € 0 per invoke
Video camera_2: € 0 per month

Weather station_3: € 0.2 per invoke
Deployment cost

• Given a mapping $\Delta$ between components of $A$ and Cloud/Fog nodes and a mapping $\vartheta$ between Thing requirements and Things:

$$\text{cost}(\Delta, \vartheta, A) = \sum_{\gamma \in A} \left[ p(\gamma.\bar{H}, \Delta(\gamma)) + p(\gamma.\bar{\Sigma}, \Delta(\gamma)) + \sum_{r \in \gamma.\bar{\Theta}} p(r, \vartheta(r)) \right]$$

- VM Cost
- SW Cost
- IoT Cost

• Still misses how to select a «best offer» for each item of expenditure.
An example

Hardware requirements

Cloud 2

```
CPU : 1
RAM : 1GB
HDD : 20GB
```

**on_demand** = 1 CPU x €4/core + 1 GB RAM x €6/GB + 20 GB HDD x €1/GB = €30/month

**small** = €25/month
Requirement-to-offering matching

\( p_m(r, n) \) matches a requirement \( r \) to the estimated monthly cost of the offering that will support it, according to some policy.

\[
\text{cost}(\Delta, \vartheta, A) = \sum_{\gamma \in A} \left[ p_m(\gamma \cdot \bar{H}, \Delta(\gamma)) + p_m(\gamma \cdot \bar{\Sigma}, \Delta(\gamma)) + \sum_{r \in \gamma \cdot \bar{\Theta}} p_m(r, \vartheta(r)) \right]
\]

- VM Cost
- SW Cost
- IoT Cost

*best-fit lowest-cost policy*
Monte Carlo Simulation

1: procedure MONTECARLO(A, I, ϑ, δ, n)
2:     D ← ∅ ▷ dictionary of ⟨Δ, counter⟩
3:     parallel for n times
4:         I_s ← SAMPLELINKSQoS(I)
5:         E ← FINDDEPLOYMENTS(A, I_s, ϑ, δ)
6:         D ← UNIONUPDATE(D, E)
7:     end parallel for
8:     for Δ ∈ keys(D) do
9:         D[Δ] ← D[Δ]/n
10:    end for
11:    return D
12: end procedure

QoS-assurance = percentage of runs a certain deployment was output

cost & Fog resource consumption
Output

QoS-assurance
How likely is a deployment to meet QoS requirements?

Fog resource consumption
Will there be free resources?

Cost
How much will the deployment cost?

QoS-aware
context-aware
cost-aware
Results (1)

Is there any deployment that complies with QoS requirements 98% of the times and doesn’t exceed € 850?

<table>
<thead>
<tr>
<th>DeploymentID</th>
<th>IoTController</th>
<th>DataStorage</th>
<th>Dashboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Δ1</td>
<td>Fog 2</td>
<td>Fog 3</td>
<td>Cloud 2</td>
</tr>
<tr>
<td>Δ2</td>
<td>Fog 2</td>
<td>Fog 3</td>
<td>Cloud 1</td>
</tr>
<tr>
<td>Δ3</td>
<td>Fog 3</td>
<td>Fog 3</td>
<td>Cloud 1</td>
</tr>
<tr>
<td>Δ4</td>
<td>Fog 2</td>
<td>Fog 3</td>
<td>Fog 1</td>
</tr>
<tr>
<td>Δ5</td>
<td>Fog 1</td>
<td>Fog 3</td>
<td>Cloud 1</td>
</tr>
<tr>
<td>Δ6</td>
<td>Fog 3</td>
<td>Fog 3</td>
<td>Cloud 2</td>
</tr>
<tr>
<td>Δ7</td>
<td>Fog 3</td>
<td>Fog 3</td>
<td>Fog 2</td>
</tr>
<tr>
<td>Δ8</td>
<td>Fog 3</td>
<td>Fog 3</td>
<td>Fog 1</td>
</tr>
<tr>
<td>Δ9</td>
<td>Fog 1</td>
<td>Fog 3</td>
<td>Cloud 2</td>
</tr>
<tr>
<td>Δ10</td>
<td>Fog 1</td>
<td>Fog 3</td>
<td>Fog 2</td>
</tr>
</tbody>
</table>
How can free Fog resources be maximised?
Results (3)

After upgrading to 4G (+€20), will the system integrators gain more money?
Results (4)

<table>
<thead>
<tr>
<th>DeploymentID</th>
<th>IoTController</th>
<th>DataStorage</th>
<th>Dashboard</th>
</tr>
</thead>
<tbody>
<tr>
<td>Δ1</td>
<td>Fog 2</td>
<td>Fog 3</td>
<td>Cloud 2</td>
</tr>
<tr>
<td>Δ2</td>
<td>Fog 2</td>
<td>Fog 3</td>
<td>Cloud 1</td>
</tr>
<tr>
<td>Δ3</td>
<td>Fog 3</td>
<td>Fog 3</td>
<td>Cloud 1</td>
</tr>
<tr>
<td>Δ4</td>
<td>Fog 2</td>
<td>Fog 3</td>
<td>Fog 3</td>
</tr>
<tr>
<td>Δ5</td>
<td>Fog 1</td>
<td>Fog 3</td>
<td>Cloud 1</td>
</tr>
<tr>
<td>Δ6</td>
<td>Fog 3</td>
<td>Fog 3</td>
<td>Cloud 2</td>
</tr>
<tr>
<td>Δ7</td>
<td>Fog 3</td>
<td>Fog 3</td>
<td>Fog 2</td>
</tr>
<tr>
<td>Δ8</td>
<td>Fog 3</td>
<td>Fog 3</td>
<td>Fog 1</td>
</tr>
<tr>
<td>Δ9</td>
<td>Fog 1</td>
<td>Fog 3</td>
<td>Cloud 2</td>
</tr>
<tr>
<td>Δ10</td>
<td>Fog 1</td>
<td>Fog 3</td>
<td>Fog 2</td>
</tr>
<tr>
<td>Δ11</td>
<td>Fog 1</td>
<td>Fog 3</td>
<td>Fog 1</td>
</tr>
<tr>
<td>Δ12</td>
<td>Fog 2</td>
<td>Cloud 2</td>
<td>Fog 1</td>
</tr>
<tr>
<td>Δ13</td>
<td>Fog 2</td>
<td>Cloud 2</td>
<td>Cloud 1</td>
</tr>
<tr>
<td>Δ14</td>
<td>Fog 2</td>
<td>Cloud 2</td>
<td>Cloud 2</td>
</tr>
<tr>
<td>Δ15</td>
<td>Fog 2</td>
<td>Cloud 1</td>
<td>Cloud 2</td>
</tr>
<tr>
<td>Δ16</td>
<td>Fog 2</td>
<td>Cloud 1</td>
<td>Cloud 1</td>
</tr>
<tr>
<td>Δ17</td>
<td>Fog 2</td>
<td>Cloud 1</td>
<td>Fog 3</td>
</tr>
</tbody>
</table>
Conclusions

Determine, simulate and compare eligible deployments

QoS-, context- and cost aware deployments

Estimate revenues and outflows beforehand
Future Work

Include **energy consumption, mobility, security**

Improve the **search**, exploit multi-criteria optimisation

Engineer and assess over **case studies**
Deploying Fog applications: How much does it cost *btw*?

Antonio Brogi, Stefano Forti and Ahmad Ibrahim

Service-oriented, Cloud and Fog Computing Research Group
Department of Computer Science
University of Pisa, Italy