Integrating Machine Learning and Multi-Agent Systems for Fully Enabling Device-Edge-Cloud Continuum in Complex IoT Worlds

Giancarlo Fortino

Full Professor of Computer Engineering @
Department of Informatics, Modeling, Electronics and Systems (DIMES)
University of Calabria (Unical), Rende (CS), ITALY

IEEE Fellow and Highly Cited Researcher 2020-22
Rector’s Delegate for International Relations, Unical
Chair of PhD Program in ICT, Unical
Director of the SPEME Lab, Unical
Distinguished Lecturer of IEEE Sensors Council
Co-founder and CEO of SenSysCal S.r.l. (www.sensyscal.com)
Co-founder and vice-CEO of Bigtech S.r.l.
Associate senior researcher at ICAR-CNR Institute, Italy
Distinguished Professor/Scientist at WUT, HUST, SIAT, HZAU, SMU, NIT (China)
Co-Director of Joint China-Italy IoT Labs (WUT, SMU, HZAU)
Visiting Researcher/Professor @ ICSI-University of Berkeley and Queensland University of Technology

Email: g.fortino@unical.it
WWW: https://labs.dimes.unical.it/speme/people/giancarlo-fortino/

IoTBDS 2023 and COMPLEXIS 2023, Prague.
Outline

1. Internet of Things: where we are and the next future!*
2. Agents meet the IoT!!!**
3. The Device-Edge-Cloud continuum Paradigm***
4. From Cloud-based Machine Learning to the Embedding Machine Learning and back via the Edge****
5. Integrating ML and Multi-agent Systems: the EU MLSysOps Project (https://mlsysops.eu/)
6. Concluding Remarks


****G. Fortino, M. Zhou, M.M Hassan, M. Pathan, S. Karnouskos, Pushing Artificial Intelligence to the Edge: Emerging trends, issues and challenges, Engineering Applications of Artificial Intelligence, 103, 2021,

*****C Savaglio, G Fortino, A Simulation-driven Methodology for IoT Data Mining Based on Edge Computing, ACM Transactions on Internet Technology (TOIT) 21 (2), 1-22, 2021
An introduction to IoT

The Internet of Things (IoT) usually refers to a world-wide network of interconnected heterogeneous objects (sensors, actuators, smart devices, smart objects, RFID, embedded computers, etc) uniquely addressable, based on standard communication protocols.

- Internet of Things

Everything is able to be networked, discovered and exploited [MSDC2012]

- Smart Objects

Real world objects with embedded smartness [KKFS2010] [SF21]


An introduction to IoT

Services are the real IoT drivers, not devices!

Bigger revenues not from IoT devices selling, but from their provided services ("product-service hybrids").
Current Status & Future Prospect of IoT

“Change is the only thing permanent in this world”
IoT History

1999
The IoT Gets a Name
Kevin Ashton coins the term “Internet of things” and establishes MIT’s Auto-ID Center, a global research network of academic laboratories focused on RFID and the IoT.
IoT: Smart Objects

- IoT: which “things”? 
  - Objects 
  - Machines 
  - Buildings 
  - Infrastructures 
  - Vehicles 
  - Pets 
  - People 
  - Plants 
  - ... almost everything ...
• Beyond the abovementioned network-oriented IoT definition, in this talk we will focus on the definition of IoT as a loosely coupled, decentralized system of cooperating smart objects (SOs).

• An SO is an autonomous, physical digital object augmented with sensing/actuating, processing, storing, and networking capabilities.

• SOs are able to sense/actuate, store, and interpret information created within themselves and around the neighboring external world where they are situated, act on their own, cooperate with each other, and exchange information with other kinds of electronic devices and human users.
IoT and Smart Objects
Towards systems of systems

A Smart Object Architecture

https://inter-iot.eu/
How IoT works?

THE INTERNET OF THINGS LIFECYCLE

COLLECT  COMMUNICATE  ANALYZE  ACT
How IoT works?

**COLLECTION**

Devices and Sensors are collecting data everywhere.

- At your home
- In your car
- At the office
- In the manufacturing plant
How IoT works?

COMMUNICATION

Sending data and events through networks to some destination

- A cloud platform
- Private data center
- Home network
How IoT works?

ANALYSIS

Creating information from the data

- Visualizing the data
- Building reports
- Filtering data (paring it down)
How IoT works?

**ACTION**

Taking action based on the information and data

- Communicate with another machine (m2m)
- Send a notification (sms, email, text)
- Talk to another system
Few Applications of IoT

✓ Building and Home automation
✓ Manufacturing (Industry 4.0)
✓ Medical and Healthcare systems (WCS)
✓ Media
✓ Environmental monitoring
✓ Infrastructure management
✓ Energy management
✓ Transportation (ITS)
✓ Better quality of life for elderly (AAL)
✓ Urban Computing
✓ ... ... ...

You name it, and you will have it in IoT!
Agents meet the IoT: Why Agents?

• To deal with the IoT system development challenges, we promote exploiting an Agent-based Computing (ABC) paradigm, which is focused on the concept of "agent", as well-defined software engineering and distributed computing paradigm for programming, deploying and managing IoT systems.

• The ABC paradigm models distributed software systems in terms of multi-agent systems (MAS), where agents are networked software entities that can perform specific tasks for a user and have a degree of intelligence that permits them to perform parts of their tasks autonomously by interacting with other agents and with their environment in a useful manner.

• Agents have been to date effectively used in many application domains to analyze and build robust and dynamic distributed systems and applications.
Agents meet the IoT: Why Agents?

• We thus claim that their characteristics also perfectly fit those of IoT systems and their components:

  • Autonomy: smart objects as agents should be able to perform the majority of their problem solving tasks without direct intervention of humans or other agents, and they should have a degree of control over their own actions and their own internal state.

  • Social ability: smart objects as agents should be able to interact, when they deem appropriate, with other smart objects and even humans in order to complete their own problem solving tasks and to help other smart objects in their activities where appropriate.

  • Responsiveness: smart objects as agents should perceive (be aware of) the environment, in which they are situated and which may be the physical world, a user, a set of other smart objects, and respond in a timely manner to changes which occur in it.

  • Proactiveness: smart objects as agents should not simply act in response to their environment, but they should also be able to exhibit opportunistic, goal-directed behavior and take the initiative where and when is appropriate.

  • Mobility: in order to fulfill physically distributed tasks, mobile smart objects should be able to physically move from one location where they act to another (as mobile software agents can migrate from one environment (machine) to another in a logical environment.
The ACOSO middleware

• The ACOSO middleware allows for the development and management of CSOs, which are modeled as agents that can cooperate with each other and with non-agent cyber-physical entities to fulfill specific goals.

• An ecosystem of CSOs therefore forms a Multi-Agent System (MAS).

• Website: http://acoso.dimes.unical.it
The ACOSO middleware

• ACOSO currently relies on JADE that provides an effective agent management and communication support.

• Specifically, CSOs can be implemented as either JADE or JADEX agents, atop both Java-based and Android-based devices.

• JADE-based CSOs can cooperate by a direct coordination model based on ACL message passing and/or by a spatio-temporal decoupled coordination model relying on a topic-based publish/subscribe mechanism.
The ACOSO middleware: Layers


The (J)ACOSO middleware: JADE-based Architecture
The Device-Edge-Cloud continuum

The symbiotic relationship between cloud computing, edge computing and IoT is reflected on the term cloud/edge/IoT computing continuum. The latter is a way of visualizing the relationship between cloud computing, edge computing, and the Internet of Things (IoT). It represents the different layers of the computing infrastructure that work together to support different applications. Specifically:

- At the top of the continuum is cloud computing, which provides a highly scalable and flexible infrastructure for storing, processing, and analyzing data based on a single or multiple clouds (e.g., hybrid clouds).
- In the middle of the continuum is edge computing, which brings the computing and storage resources closer to the IoT devices themselves.
- At the bottom of the continuum is the IoT itself, which consists of a vast network of connected devices that generate and transmit data. IoT includes a wide variety of devices, from sensors and wearables to industrial machinery and smart appliances.

At each layer, different technologies and architectures are used to meet the specific requirements of enterprise applications. By working together, these layers form a continuum that enables the seamless and efficient processing of data from IoT devices. Modern enterprises are expected to invest in the different layers of the continuum to meet the requirements of their applications and improve their business results.
The Device-Edge-Cloud continuum

The Device-Edge-Cloud continuum

The Device-Edge-Cloud continuum

A Deep Learning Approach on Edge-Fog-Cloud Framework for Driving Behavior Detection and Monitoring

A Deep Learning Approach on Edge-Fog-Cloud Framework for Driving Behavior Detection and Monitoring

Simulation-Driven Platform for Edge-Based AAL Systems

Simulation-Driven Platform for Edge-Based AAL Systems

Fig. 7. Layout of the simulation environment: a care facility.

Fig. 8. Well-being services performances with multiple care facility units served by one edge node.

TABLE III
DETAILED eHEALTH SERVICE PERFORMANCE WITH RESPECT TO THE DIFFERENT DEPLOYMENTS SUPPORTED BY E-ALPHA AND A 3G-BASED CONNECTIVITY

<table>
<thead>
<tr>
<th>Deployment Type</th>
<th>Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Service Time (s)</td>
</tr>
<tr>
<td>OnlyEdge</td>
<td>13.1</td>
</tr>
<tr>
<td>OnlyCloud</td>
<td>10.3</td>
</tr>
<tr>
<td>FullPlatform</td>
<td>11.0</td>
</tr>
</tbody>
</table>
From Current to Future World of Wearables:  
Conventional WCS Architecture and data provided by WCS

<table>
<thead>
<tr>
<th>Categories</th>
<th>Data Signals</th>
<th>Application Foci</th>
<th>Main Application Domains</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inertial</td>
<td>Acceleration, Orientation, Magnetometer values</td>
<td>Gesture/Movement recognition, Fall detection</td>
<td>Sports, Healthcare, Wellness, Emergency response</td>
</tr>
<tr>
<td>Visual</td>
<td>Image &amp; depth map, Image</td>
<td>Aiding low-vision people, Fall detection, environment recognition</td>
<td>Healthcare, Emergency response</td>
</tr>
<tr>
<td>Audio</td>
<td>Environmental sounds, Throat sounds, Voice</td>
<td>Ingestive behavior monitoring, Activity and Voice recognition</td>
<td>Healthcare, Emergency response</td>
</tr>
<tr>
<td>Strain</td>
<td>Strain</td>
<td>Activity recognition, Ingestive behavior monitoring</td>
<td>Healthcare, Fitness, Sport</td>
</tr>
<tr>
<td>Force</td>
<td>Force</td>
<td>Gait analysis, Activity recognition</td>
<td>Healthcare, Fitness, Sport, Emergency response</td>
</tr>
<tr>
<td>RFID</td>
<td>RF signal</td>
<td>Activity recognition, Man-Environment interaction</td>
<td>Manufacturing, Logistics</td>
</tr>
</tbody>
</table>

![Figure 1.1: Conventional WCS Architecture and data provided by WCS](image-url)
SPINE Body-of-Knowledge

https://projects.dimes.unical.it/spine-bok/

- The **SPINE Body of Knowledge (BoK)** has been created over the last 15 years in the context of the open-source SPINE project, and includes models, methods, algorithms, frameworks, tools and systems for the systematic and full-fledged development of wearable computing systems based on body sensor networks.

- SPINE covers many different **application domains**: Healthcare, Fitness, Sport, Factory, Transportation, Gaming, Social interactions, Defence.

- The SPINE project was **originally established in 2006** at the Telecom Italia/Pirelli Wireless Sensor Networks Lab in Berkeley (CA). The founders were University of Calabria (G. Fortino), Telecom Italia/Pirelli WSN Lab (M. Sgroi), Telecom Italia Lab (F. Bellifemine), and University of Berkeley (A. Sangiovanni-Vincentelli). Since 2013, the project is **fully driven and managed by the Prof. Fortino’s research group** at University of Calabria. Many R&D groups from Academia and Industry contributed to SPINE BoK, both with research contributions and with contribution to the open-source code.

- In particular, the **SPINE BoK includes**: the SPINE framework and related methodology (SPIME-DM), the SPINE extension frameworks (SPINE2, C-SPINE, A-SPINE, SPINE-*), the BodyCloud infrastructure and related methodology, the BodyEdge infrastructure, and a rich set of application-specific multi-sensor data fusion algorithms.
Overall, the SPINE research and dissemination activities produced 100+ papers, notably 40+ in top-level journals (e.g. IEEE-THMS, IEEE-SensorsJ, IEEE-IoTJ, IEEE-Network, IEEE-WCM, IEEE-TASE, INFFUS, FGCS, JNCA), 5000+ number of citations according to Google Scholar with an h-index=30, and five highly cited papers according to Web of Science.


Prof. Fortino is Distinguished Lecturer of IEEE Sensors Council disseminating the SPINE BoK with lectures on “Wearable Computing Systems based on Body Sensor Networks: State-of-the-art and Future Research Challenges”.
The **SPINE (Signal Processing In-Node Environment)** Project

[Diagram of SPINE architecture]

  
  **Award: 2014 Andrew P. Sage Best IEEE SMC Transactions (from Web of Science Core Collection) Highly Cited Paper**

- **G. Fortino, S. Galzarano, R. Gravina, Wearable Systems and Body Sensor Networks: from modeling to implementation, Wiley, USA, 2018.**

[Website link: http://spine.deis.unical.it]

**Project Contributors:**

- Telecom Italia (F. Bellifemine)
- UT Dallas (R. Jafari)
- UC Berkeley (A. Sangiovanni-Vincentelli)
- ...
EDGE-CLOUD C-SPINE-based Architecture

1) **Neighbor Detection**: it detects neighbor CBSNs among co-located people in a specific range.

2) **Sensor Selection and Service activation**: when neighbors have been detected, all of BSNs will activate/ select sensing on their own nodes and necessary (processing) services will be activated. This step will allow the system saving energy which means sensor will only activated when it is needed.

3) **Potential Activity**: once a one-sided potential activity occurs on a BSN, the corresponding BS will be notified.

4) **Advertisement**: the coordinator will send a message to all neighbor CBSNs to request if there is a corresponding reaction for detecting multi-user activity.

5) **Collaborative Information Fusion**: low-level data and recognized individual activity will be sent to BodyEdge; the edge layer will perform decision-level fusion according to specific classification algorithms to detect the multi-user activity.

6) **Remote Access**: if large amounts of data are needed for computation or storage, the BodyCloud layer will provide proper support.

---


1. **Statistical significance** is always critical (spatio-temporal series)
2. **Data fusion** is challenging (heterogeneity of objective & subjective sources)
3. **Data is intrinsically unreliable and incomplete** (collection constraints)

Intelligent processes must start at the micro-edge (concurrent sensing and learning)
Machine Learning & IoT

Typical data science workflow: from data ingestion to modelling

- Raw data
- Data exploration
- Data cleaning
- Data curation
- Data transformation

Model building

Model evaluation

Analysis & Insights

Visualization & explanation

Clean and curated

appropriate dimension

Tools | Methods | trial & error | domain specific | time consuming

from DATA to VALUE
Machine Learning & IoT

The ‘art’ of turning data into actionable insights

MODEL PLANNING

- Data transformation
- Data curation
- Data cleaning

Variety of tools and techniques

- Model building
- Model evaluation
- Analysis & Insights
- Visualization & explanation

from DATA → 70% effort → 30% effort → to VALUE
Machine Learning & IoT: Cloud-based Approach

Implementation:
Data-intensive processes are virtually centralized

Dataset | Internet | Cloud computing

From data | | Data insights
Machine Learning & IoT:
Cloud-based Approach
Machine Learning & IoT: Hybrid Device- and Cloud-based Approach

IoT intelligence must start at micro-edge level

Data streams | IoT device | Internet | Cloud computing

**Short-long term anomaly detection in wireless sensor networks based on machine learning and multi-parameterized edit distance** [https://doi.org/10.1016/j.infus.2018.11.010]

Machine Learning & IoT: Embedded Machine Learning

In-node shallow Learning:
Concurrent sensing and learning for anomaly detection

TelosB mote specs
- TI MSP430 controller, 8Mhz
- 10KB RAM
- IEEE 802.15.4 compliant
- 250 kbps data rate radio
- TinyOS 1.1.10 or higher

Shallow learning algorithms
- Sliding window mean
- Recursive Least Sq.
- Extreme learning machines
- Polynomial Function Approximation
- Ensembles
Machine Learning & IoT: Embedded Machine Learning

In-node reinforcement learning: Spectrum and energy efficiency

Transmission power convergence

- ATxmega256A3U
- CPU: 32 MHz
- Flash memory: 256 Kbytes
- SRAM: 16 Kbytes
- IEEE 802.15.4
- Low-power transceiver (11mA in receiving mode)
- Development platform: atmel studio (20% of avail. memory)

Intelligence at the Edge of Complex Networks: The Case of Cognitive Transmission Power Control
IEEE Wireless Communications https://doi.org/10.1109/MWC.2019.1800354

Machine Learning & IoT: Federated Learning Edge-Cloud (FL, FTL, FD)

IoT Data Mining @ the Edge: The EdgeMiningSim Methodology

C Savaglio, G Fortino, A Simulation-driven Methodology for IoT Data Mining Based on Edge Computing, ACM Transactions on Internet Technology (TOIT) 21 (2), 1-22, 2021.
Does the straightforward use of traditional Data Mining technologies over smart devices allow an effective and efficient analysis of the data generated at the IoT Edge?

<table>
<thead>
<tr>
<th>Data Mining</th>
<th>IoT Data Mining</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Objective</strong></td>
<td>Knowledge Disclosure</td>
</tr>
<tr>
<td><strong>Task</strong></td>
<td>Descriptive, Predictive</td>
</tr>
<tr>
<td><strong>Goal</strong></td>
<td>Technical significance</td>
</tr>
<tr>
<td><strong>Exploited technique</strong></td>
<td>Automatic, semi-automatic</td>
</tr>
<tr>
<td><strong>Process</strong></td>
<td>Data-driven</td>
</tr>
<tr>
<td><strong>Locus of computation</strong></td>
<td>Servers, Cloud Servers</td>
</tr>
<tr>
<td><strong>Data sources</strong></td>
<td>Computing systems, sensors</td>
</tr>
<tr>
<td><strong>Data</strong></td>
<td>Disk-resident transactional, refined dataset</td>
</tr>
<tr>
<td><strong>Resources availability</strong></td>
<td>High and stable</td>
</tr>
<tr>
<td><strong>Importance of Simulation</strong></td>
<td>Limited</td>
</tr>
</tbody>
</table>
IoT Data Mining: Motivations

Does the straightforward use of traditional Data Mining technologies over smart devices allow an effective and efficient analysis of the data generated at the IoT Edge?

Different requirements, constraints and goals for different IoT domains

→ systematic guidelines needed

<table>
<thead>
<tr>
<th>IoT Domain</th>
<th>Example of Data Mining Task</th>
<th>Main Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>Smart Video Surveillance</td>
<td>Classification-based object recognition</td>
<td>Bandwidth Efficiency</td>
</tr>
<tr>
<td>Smart Agriculture</td>
<td>Clustered WSNs</td>
<td>Energy Efficiency</td>
</tr>
<tr>
<td>Smart Health</td>
<td>Outline detection from ECG signal</td>
<td>Responsiveness</td>
</tr>
<tr>
<td>Smart Home</td>
<td>Cluster-based user behavior analysis</td>
<td>Privacy</td>
</tr>
<tr>
<td>Smart Transportation</td>
<td>Classification-based fatigue detection systems</td>
<td>Safety</td>
</tr>
<tr>
<td>Smart Grid</td>
<td>TS-based prediction on consumer’s energy expenditure</td>
<td>Accuracy</td>
</tr>
<tr>
<td>Social IoT</td>
<td>Community detection among heterogeneous IoT devices</td>
<td>Interoperability</td>
</tr>
<tr>
<td>Smart City</td>
<td>Classification-based parking management</td>
<td>Scalability</td>
</tr>
</tbody>
</table>
EdgeMiningSim

- A simulation-driven methodology inspired to software engineering principles for enabling the IoT Data Mining.
- Supports manifold (algorithmic, infrastructural and contextual) aspects, so far only partially or individually analyzed in the narrow literature of IoT Data Mining.
- General purpose (widely stable across varying applications), interactive (strategic decisions taken by the user), iterative (possibility of backtracking to previous steps) and tool/technique independent but supportable (like C.R.I.S.P.).

Goal: driving (through simulation) the domain experts in disclosing actionable knowledge, namely descriptive or predictive models for taking effective actions in the IoT scenario.
Table 3. Breakdown of the proposed methodology in its constitutive elements (specific activities are starred)

<table>
<thead>
<tr>
<th>Phase (ID)</th>
<th>Activity (ID)</th>
<th>Task</th>
<th>Work-product (ID)</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>IoT Domain Analysis (P1)</td>
<td>Application Characterization (P1.a)</td>
<td>UML modelling, User storytelling</td>
<td>Domain Report (WP1)</td>
<td>Business expert, Technical expert</td>
</tr>
<tr>
<td></td>
<td>Requirements Identification (P1.b)</td>
<td>Joint Interview, Requirements ranking</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Device Characterization (P1.c)</td>
<td>Datasheet analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Scenario Characterization (P1.d)</td>
<td>Network infrastructure analysis, Topology analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IoT Data Analysis (P2)</td>
<td>Data format understanding (P2.a)</td>
<td>ASIS HL7 standard</td>
<td>Data Report (WP2)</td>
<td>Business expert, Technical expert</td>
</tr>
<tr>
<td></td>
<td>Data lifecycle understanding (P2.b)</td>
<td>DFD modelling</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data quality problem identification</td>
<td>Outline detection, Missing data analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>(P2.c)*</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data preparation (P2.d)*</td>
<td>Data cleaning, Data aggregation,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Mining Setting (P3)</td>
<td>DM Goal identification (P3.a)</td>
<td>DM Framework analysis</td>
<td>Data Mining Report (WP3)</td>
<td>Technical expert</td>
</tr>
<tr>
<td></td>
<td>DM Task identification (P3.b)</td>
<td>DM Framework analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DM Algorithm identification (P3.c)</td>
<td>DM Framework analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DM Algorithm customization (P3.d)*</td>
<td>Heuristic, Algorithm optimization</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## EdgeMiningSim

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Devices Modelling (P4.b)</td>
<td>Mobility modelling, Energy modeling</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Application Modelling (P4.c)</td>
<td>Task generation modelling</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simulator Selection (P4.d)</td>
<td>SOTA-analysis</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simulator Customization (P4.e)*</td>
<td>Models set extension, Statistics edit</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DM Algorithm Preliminary Setting (P4.e)*</td>
<td>Time-window assessing, Clusters’ SSE analysis, Training-set definition</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simulation execution (P4.f)</td>
<td>Performance criteria specification, Results plotting</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Evaluation and Validation (P5)</th>
<th>Simulation Result evaluation (P5.a)</th>
<th>Comparative analysis respect WP1</th>
<th>DM Project Setting (WP5)</th>
<th>Business expert, Technical expert</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trade-off Management (P5.b)</td>
<td>Settings tuning Backtracking</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simulation Result validation (P5.c)</td>
<td>Testbed design, Discussion with Business expert</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**EdgeCloudSim** is a simulator built upon CloudSim to address the specific demands of Edge Computing research and support necessary functionality in terms of computation and networking abilities.

**EdgeCloudSim (event-driven) architecture:**

- Mobile SOs dynamically organized in LAN
- Static Edge Server (e.g., RaspberryPi), one per LAN, provided with a wireless Access Point
- Both SO and EdgeServers can talk with each other and with the Cloud
- If needed, EdgeServers can be coordinated by an Orchestrator (e.g., load balancing)
EdgeMiningSim: A Use Case

A case study related to Smart Environment for showing the application of EdgeMiningSim and the trade-off analysis among Cloud and Edge Mining.

Fig. 4. The full convergence of K-Means (100% accuracy) demands for 30 iterations.

Fig. 5. (a) K-Means energy depletion according to the Cloud- and Edge-based version (with both message-passing models) and the number of algorithm iterations (b) Bandwidth depletion according to both the message-passing models of Edge-based K-Means version and the number of algorithm iterations.

Fig. 6. K-Means execution times, with separated computation and communication contributions, according to the Cloud- and Edge-based version (with both message-passing models) and the performed iterations.
### EdgeMiningSim: A Use Case

<table>
<thead>
<tr>
<th><strong>Domain Report</strong> (WP1)</th>
<th><strong>Data Report</strong> (WP2)</th>
<th><strong>Data Mining Report</strong> (WP3)</th>
<th><strong>Simulation Report</strong> (WP4)</th>
<th><strong>DM Project Setting</strong> (WP5)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>P1.a:</strong> Smart Monitoring, indoor environments</td>
<td>P2.a: 2.3 million readings, REAL timestamped data, temperature, humidity, light, voltage values</td>
<td>P3.a: descriptive model</td>
<td>P4.a: Cloud- and Edge-based deployment</td>
<td>P5.a: see Figs. 4-6</td>
</tr>
<tr>
<td><strong>P1.b:</strong> accuracy, bandwidth, energy efficiency</td>
<td>P2.b: packets of 65 bytes every 31 sec</td>
<td>P3.b: clustering task</td>
<td>P4.b, P4.c: according to WP1, WP2 and WP3</td>
<td>P5.b: Edge-based deployment, 18 iterations, ring message-passing model</td>
</tr>
<tr>
<td><strong>P1.c:</strong> 54 MICA2Dot sensors, AWS server</td>
<td>P2.c: noisy and missing data</td>
<td>P3.c: distributed K-Means</td>
<td>P4.d: EdgeCloudSim</td>
<td>P5.c: see Fig. 7</td>
</tr>
<tr>
<td><strong>P1.d:</strong> grid deployment</td>
<td>P2.d: ring and flooding message-passing models</td>
<td>P3.d: energy model of [29]</td>
<td>P4.e: $k=3$, $n=4$</td>
<td></td>
</tr>
</tbody>
</table>

Table 4. WPs resulting from the application of the proposed methodology on the case study.
The MLSysOps Project (see external slide)

Use Cases

SMART AGRICULTURE

SMART CITIES
Conclusions

• IoT-Edge-Cloud continuum is becoming a best practise!!!

• Pushing intelligence and machine learning to the IoT Edge is becoming a “must”!!!

• From embedded machine learning to distributed machine learning at the edge, limiting data mining on the cloud!!!

• Methodologies are key for the next-generation IoT development!!!

• Towards autonomic systems based on the Integration of IoT-Edge-Cloud continuum with Multi-Agent Systems
Thank you!

Question?

Contact: g.fortino@unical.it