Towards Modeling the Modern Distributed Systems Fabric

Keynote at Modelsward 2021
10 Feb 2021

Schahram Dustdar

dsg.tuwien.ac.at
Smart Evolution – People, Services, and Things

- Elastic Systems & Processes
- Smart Energy Networks
- Smart eGovernments & eAdministrations
- Smart Transport Networks
- Smart Homes
- Smart Health & Smart Health networks

- Game Machine
- STB
- TV
- PC
- Telephone
- Audio
- DVD
Ecosystems: People, Systems, and Things

Complex system with networked dependencies and intrinsic adaptive behavior – has:

1. Robustness & Resilience mechanisms: achieving stability in the presence of disruption
2. Measures of health: diversity, population trends, other key indicators
3. Built-in coherence
4. Entropy-resistence
Ecosystems for Distributed Systems
Observation

There are new families of applications that require:

• (Soft) Real-time location-based access to data from the environment at different levels of fidelity

• Appropriate compute and storage resources in close proximity to data producers and consumers
Dynamic Analytics (e.g., Smart City)
Rethinking Divide and Conquer—Towards Holistic Interfaces of the Computing Stack

IEEE Internet Computing, Vol 24., Issue 6, Nov/Dec, pp. 45-57

Schahram Dustdar
TU Wien

Nandita Vijaykumar
University of Toronto

Onur Mutlu
ETH Zürich
Perspectives on Distributed Systems infrastructures

(a) A cloud-centric perspective: Edge as “edge of the cloud”
(b) An Internet-centric perspective: Edge as “edge of the Internet”

Cloud-centric perspective

Assumptions

• Cloud provides core services; Edge provides local proxies for the Cloud (offloading parts of the cloud’s workload)

Edge Computers

• play supportive role for the IoT services and applications

• Cloud computing-based IoT solutions use cloud servers for various purposes including massive computation, data storage, communication between IoT systems, and security/privacy

Missing

• In the network architecture, the cloud is also located at the network edge, not surrounded by the edge

• Computers at the edge do not always have to depend on the cloud; they can operate autonomously and collaborate with one another directly without the help of the cloud
Internet-centric perspective

Assumptions

• Internet is center of IoT architecture; Edge devices are gateways to the Internet (not the Cloud)
• Each LAN can be organized around edge devices autonomously
• Local devices do not depend on Cloud

Therefore

• Things belong to partitioned subsystems and LANs rather than to a centralized system directly
• The Cloud is connected to the Internet via the edge of the network
• Remote IoT systems can be connected directly via the Internet. Communications does not have to go via the Cloud
• The Edge can connect things to the Internet and disconnect traffic outside the LAN to protect things -> IoT system must be able to act autonomously
IoT/Edge/Fog/Cloud Continuum: A high level view

- **Fog Domain**
  - Mobile/access network edge
  - Edge of the (mobile) network
  - Low latency to end device
  - Close to/collocated with 4G/5G base stations
  - General purpose compute infrastructure
  - Standards-based architectures & management/orchestration stacks

- **Edge Domain**
  - Core network, Internet
  - Telecom operator controlled

- **Cloud Domain**
  - Central clouds
  - “Unlimited” compute/storage resources
  - Full spectrum of cloud services
  - High availability
  - Lower cost
  - Higher latency vs. edge/fog

---

**Low reliability**

**Volatility**

**Mobility**

(Mostly) Wireless connectivity

**Small form factor**

**Battery constraints**

Mobile, IoT, smart home, vehicles, ...

**User/Service provider controlled**

---
Vertical vs. Horizontal Edge/Fog/Cloud Architecture
Computing Continuum (horizontal | vertical)

Sun Modular Datacenter  
Mini-ITX Servers ¹  
Ubuntu Orange Box (Intel NUC cluster)  
“Micro Clouds” ²

Server Computers

SOC & Single Board Computers


Specialized Compute Platforms

Write once run anywhere™?
City-Scale Edge Computing Fabric

https://arrayofthings.github.io

Huawei PoleStar2.0
Software-intensive Edge Systems

Total rethink necessary to support design and operation in an environment that changes

Fundamental conflicting system factors critical to system requirements satisfaction include:

• **Latency**, as delays of data or control command transfers is a factor arising from the platform and networks heterogeneity and the inherent traditional division of Cloud-IoT, and may affect timeliness and performance;

• **Computation as an Edge resource**, traditionally performed on cloud infrastructures now may be located closer to end devices, raising an abundance of complex issues associated with distributed systems such as **safety, and security**;

• **Locality and mobility** within administrative domains introduces novel challenges with respect to **privacy, software configuration and system evolution**
Question

Which characteristics of edge computing systems should be abstracted as first-class citizens into the underpinning model?
Hypothesized Answer

• Proximity
• Context
• Capabilities
• Energy

→

• Elastic diffusion
• Intelligent resource allocation
• Efficient operations
Elasticity (Resilience)

(Physics) The property of returning to an initial form or state following deformation.

- **stretch** when a force stresses them
  
e.g., *acquire* new resources, *reduce* quality

- **shrink** when the stress is removed
  
e.g., *release* resources, *increase* quality
Elastic Computing > Scalability

High level elasticity control

#SYBL.CloudServiceLevel
Cons1: CONSTRAINT responseTime < 5 ms
Cons2: CONSTRAINT responseTime < 10 ms
WHEN nbOfUsers > 10000
Str1: STRATEGY CASE fulfilled(Cons1) OR fulfilled(Cons2): minimize(cost)

#SYBL.ServiceUnitLevel
Str2: STRATEGY CASE ioCost < 3 Euro: maximize(dataFreshness)

#SYBL.CodeRegionLevel
Cons4: CONSTRAINT dataAccuracy>90% AND cost<4 Euro


Elasticity Model for Edge & Cloud Services


- **Elasticity Pathway functions**: to characterize the elasticity behavior from a general/particular view

- **Elasticity space functions**: to determine if a service unit/service is in the “elasticity behavior”
Elastic Diffusion, aka Osmotic Computing

In chemistry, “osmosis” represents the seamless diffusion of molecules from a higher to a lower concentration solution.

Dynamic management of (micro)services across cloud and edge infrastructures
- deployment, networking, and security, ...
- providing reliable IoT support with specified levels of QoS.

Towards Edge Intelligence

Computational Fabric

- dispersed resources allow training, monitoring, serving of models
- Heterogeneity of applications and models requires
  - (1) flexible and modular infrastructure and
  - (2) intelligent operations mechanisms (due to the scale of the infrastructure)

Operationalization

- Automated AI application lifecycle management to the Edge

1. **Sensing (Sensor Data as a Service)**
   - Large number, dynamic and mobile nature of publishers/subscribers of sensor data + QoS requirements of edge intelligence
     - rethink centralized messaging services such as AWS IoT or MS Azure IoT Hub
   - Management and governance of such a distributed/decentralized sensing infrastructure

2. **Edge computer network with modular AI capabilities**
   - New AI accelerators for edge devices (e.g., Google Edge TPU with an application specific integrated circuit; MS BrainWave with field-programmable gate arrays (FPGAs); Intel Neural Compute Stick; Baidu Kunlun, Huawei Atlas AI Platform)

3. **Intelligent orchestration mechanisms for decentralized and distributed infrastructure**
Edge Intelligence Fabric


Training on data **directly on remote devices**...

...without revealing the data themselves

Sending the outcome of local training to server (local updates)

Server aggregates these updates into a global model

Makes the model available to devices

Applications

● For mobile devices
  ○ Next-word prediction, face detection, voice recognition
  ○ Train on data from smartphone text editors, cameras, mics
  ○ Users do not wish to reveal their messages, photos, and videos
  ○ Also, they don’t want to waste bandwidth and MBs from their data plan

● For organizations
  ○ Organizations such as hospitals have data, but should not expose them
  ○ Federating such data in a private way to apply ML for medical and other research

● For environmental, transportation, smart home, and other applications
  ○ Measurement devices with sensors (e.g., for air pollution) mounted on cars
  ○ Sensors in a smart home
  ○ Pushing data to servers for centralized training might leak driver patterns, daily habits, etc.
Current research challenges

**Device recruitment strategies:** Which subset of the devices to assign a learning task at any given round? Processing, storage, battery, trustworthiness, data quality and other criteria to consider

**Volatility:** Devices can “disappear” or join at any time

**Asynchrony:** Algorithms face challenges when end devices do not submit their data in a timely manner

**Non independent and identically distributed data:** inaccuracies, personalization lost

**Heterogeneity in the volume of training data per device:** A device that contributes a lot may lead to a biased model

**Preventing privacy leaks:** Some private information may be inferred even if devices do not transmit the actual data

**Incentives to misbehave:** Why waste battery when I can let the others do all the work?

Research Roadmap – Quality of Experience

1. Performance
   E.g., the ratio of computation offloading

2. Cost
   Computation | Communication | Energy consumption costs

3. Privacy & Security
   Federated learning, i.e., aggregating local machines models from distributed edge devices

4. Efficiency
   Excellent performance with low overhead, e.g., model compression, conditional computation

5. Reliability
   Relates to model upload and download and wireless network congestion
AI for Edge

1. Topology
   • Edge orchestration and coordination with small base stations
   • Unmanned Aerial Vehicles (UAVs) and access points

2. Content
   Lightweight service frameworks for QoS-aware services, e.g., on mobile devices

3. Service
   Computation offloading, User profile migration and mobility management
Grand Challenges – AI for Edge

• **Model Establishment – restraining the optimization model**
  - Stochastic Gradient Descent (SGD)
  - MBGD (Mini-Batch Gradient Descent)

• **Algorithm Development**
  - Selection of *which* edge device should be responsible for deployment and execution in an online manner
  - SOTA formulates combinatorial and NP-hard optimization problems with high computational complexity

• **Trade-off between optimality and efficiency**
  - Consider resource constraint devices

Al on Edge

• Data Availability
  • Challenge of lack of availability and usability of raw training data for model training and inference
  • Bias of raw data from various end user/mobile devices

• Model Selection
  • SOTA requires selection of need-to-be trained AI models has challenges
  • Threshold of learning accuracy and scale of AI models for quick deployment and delivery
  • Selection of probe training frameworks and accelerator architectures under limited resources

• Coordination Mechanisms
  • Coordination between heterogeneous edge devices, cloud, and various middlewares and APIs

Managing the AI Lifecycle

AI lifecycle pipeline with a rule-based trigger \( e \) that monitors available data and runtime performance data to form an automated retraining loop.
## AI Operations Workflows – Edge to Cloud

<table>
<thead>
<tr>
<th>Data characteristics</th>
<th>Model characteristics</th>
<th>Enabling technologies</th>
<th>Example use cases</th>
</tr>
</thead>
</table>
| **C2C**              | - Training data is centralized  
- Massive data sets | - Models are large  
- Huge number of inferencing requests need to be load balanced | - Scalable learning infrastructure [39]  
- Data warehousing | - Image search  
- Recommender systems |
| **C2E**              | - Training data is centralized  
- Inferencing data may be sensitive | - Inferencing may need to happen in near-real time  
- Large number of model deployments  
- Models run on specialized hardware | - Model compression [42]  
- Latency/accuracy tradeoff [43]  
- Distributed inferencing [44]  
- Transfer learning [45] | - Surveillance systems  
- Self driving cars  
- Fieldwork assistants |
| **E2C**              | - Training data is distributed  
- Training data may be sensitive | - Models can be centralized  
- Huge number of inferencing requests need to be load balanced | - Decentralized/federated learning [41] | - Volunteer computing  
- Novel Smart City use cases |
| **E2E**              | - Training data is distributed  
- Training and inferencing data may be sensitive | - Inferencing may need to be near-real time | - Decentralized/federated learning  
- Distributed inferencing | - Industrial IoT (e.g., predictive maintenance)  
- Privacy-aware personal assistants  
- Novel IoT use cases |

Conclusions

• Leverage the Computing “Continuum“ from IoT->Edge->Fog->Cloud

• Differentiate between AI for Edge and AI on Edge. Both bring their distinct research challenges

• Need for an Edge Intelligence AI Fabric and a “clear“ distributed systems ecosystems understanding
Thanks for your attention

Prof. Schahram Dustdar
IEEE Fellow
Member and Section Chairman
“Informatics” at Academia Europaea
ACM Distinguished Scientist
ACM Distinguished Speaker
IEEE TCSVC Outstanding Leadership Award in Services Computing
IEEE TCSC Award for Excellence in Scalable Computing
IBM Faculty award
Distributed Systems Group
TU Wien, Austria
dsg.tuwien.ac.at