Orchestrating the Edge: the “Guardian Angel” concept

IoT and Edge Computing: Future directions for Europe

HiPEAC vision 2021 editorial board
Marc Duranton (CEA)

Since 2004, HiPEAC has provided a hub for European researchers and industry representatives in computing systems (HW and SW).
Vision on the evolution...  
... of the Web and of IoT
Evolution of the WEB

Web 1.0: One way road: Contents and presentation imposed

Web 2.0: Bidirectional contents
Presentation imposed
Evolution of the WEB

Less use of web browsers, instead
Specialized Apps for **services**
(e.g. App from an Hotel to *book a room*)
Aggregators apps to select the service
Even Aggregators of aggregators
...
Overflow of information, of choices, etc....
Industry/users only want “**services**”

More natural and contextual interfaces (AI)
Aware of the environment
Adapt input and output to the situation
Horizontal IoT Platform

Various application/industry domains
Domain specific Platforms
(also company specific)

Heterogeneous IoT devices

Heterogeneous physical quantities

Heterogeneous physical world
Domain specific Platforms (also company specific)
Various application/industry domains
“PROGRAMS”, “APPLICATIONS” NOW ARE EXECUTED IN THE “CONTINUUM”

MULTIPLE DISTRIBUTED FUNCTIONS (SERVICES)
+ GLUE (ASSEMBLING, ORCHESTRATING) = THE “NEW” PROGRAM STRUCTURE

Even at programming level
E.g. Python
(glue between specialized libraries)

- This initiative currently involves:
  - ETP4HPC, ECSO, BDVA, 5GIA,
  - EUMATHS, CLAIRE, HiPEAC...
  - More to come...

Edge processing, Fog, ...
Plug & play, self-management, orchestration with (virtually) no human intervention

- Auto-discovery
- Self-configuration
- Self-matchmaking
- Auto-description
- Self-optimization
- Self-healing
- Energy-harvesting

- Auto-scaling
- Continuous-deployment
The challenge: Interoperability and Composability (Orchestration)

- **Overcome the fragmentation of vertically-oriented closed systems**
  Move towards open platforms and standards.

- **Heterogeneity/Interoperability**
  How to handle the numerous types of devices, protocols, standards, *non functional requirements*?

- **Scalability**
  How to handle the big number of connections/big data coming from millions of devices?

- **Dynamicity**
  Plug&play, self-configuration, self-management, self-matchmaking

- **Privacy**
  Ensuring that the confidential data of the company, the individuals don’t leak

Creating the *next Web*, Intertwining Cyber and Physical worlds for industrial and personal use

*Build on top of existing technologies*
- Dynamic construction of applications from distributed Services (local or remote)
- Taking into account non-functional requirements (locality, energy, privacy, ...)

Build on a trustable distributed meta-level “OS”

Guardian Angel

Physical quantities

Physical world

Company/private data

Running on edge hardware

trusted computing base
Hints on how to realize the “Guardian Angels”
Pictorial representation

As-a-service (open-source) Apps written by Software after Human specification, verification and approval

Contract-based services REST-API

In and Out-bound data protection

Natural interface to orchestration scripts

The soldering robot 23 should take the blue car door and ....
Having software write software

• Application code should be written by software – humans should give specifications
  – Most such Apps will run on near-edge:edge devices, but also on traditional computers
• The Apps themselves, highly-mobile (not resident), serverless-like, and sandboxed, will aggregate other services
  – The communications and interactions needed for orchestration will all occur over-and-above HTTPS
  – The individual services candidate for orchestration will expose rich contract-based interfaces and will reside anywhere
  – The development of such Apps by orchestration will not need programmers, but programming engines capable of voice-and-artefact interaction with humans:
    – Humans will express requirements, preferences, and properties to be satisfied
    – Development will be iterative until satisfaction, sanctioned with (auto-generated) proof of compliance
    – They should be correct by construction

• Their execution environment (the Trusted Computing Base) will be an evolution of the engine of modern browsers without the display and rendering part.
  – The TCB will also allow orchestrating the activation and cooperation of Apps as needed
  – The TCB builds user preferences (adaptively) and preserve them against attacks, bias and tampering
Continuum of computing

• Continuum of computing: a uniform platform capable of hosting as-a-service software at the place most appropriate to the use(r) of it, for latency, resource capacity, etc.
• The continuum infrastructure is composed of two types of software
  - The TCB written, verified and validated (and approved by humans), and delivered in open source
  - The user-level Apps (resulting from orchestrating service), not written by humans, but produced by (virtual-assistant) software in response to oral or other natural way to express specifications and guided assisted iterations (including automated verification and validation) before delivery and use

• The platform of that continuum could be built as a trusted, principled, overlay over the current OS technologies: this will form the Trusted Computing Base (TCB)
  - It could entirely replace traditional OSes on the edge, where technology is less dependent on legacy
  - Open source, adherent to European values for data privacy and fairness, treated as a “public good”
  - The economic revenue of open-sourcing the continuum infrastructure should come from private enterprises developing and commercializing proprietary value-added services that run on the TCB but are unable to break and breach it

• The external interface of the continuum should be a flexible, configurable, programmable, orchestrator and aggregator of service Apps
  - Where Apps should be consumed in an as-a-service or serverless model (no need for installation)
  - With sandboxed execution, as with container-based microservices

• An emblematic notion of this continuum platform is a constellation of cooperating personalized virtual assistants running on edge nodes (handheld, wearable devices) near the user.
Structuring European efforts: define a cross domain moonshot project

Promote a collaboration that synergizes:

• Interoperability and a contract based approach
  • Contracts, assertions between entities
• Exposing non-functional properties
  • Managing the complexity (of choices)
• Controlling in-bound and out-bound data
  • According to user’s profile(s)
• Edge processing and federation of local resources
• Open-source “orchestrator” and TCB
• Natural interfaces and context aware (using AI)
• ”Natural programming” (for orchestration)
• Containers for legacy and migration of code/data
• “Supervisor” for non trusted/legacy parts
• Cybersecurity everywhere
• Running on European hardware platforms
Past, Present and Future

For a more fun introduction of the concept of Guardian Angels...
