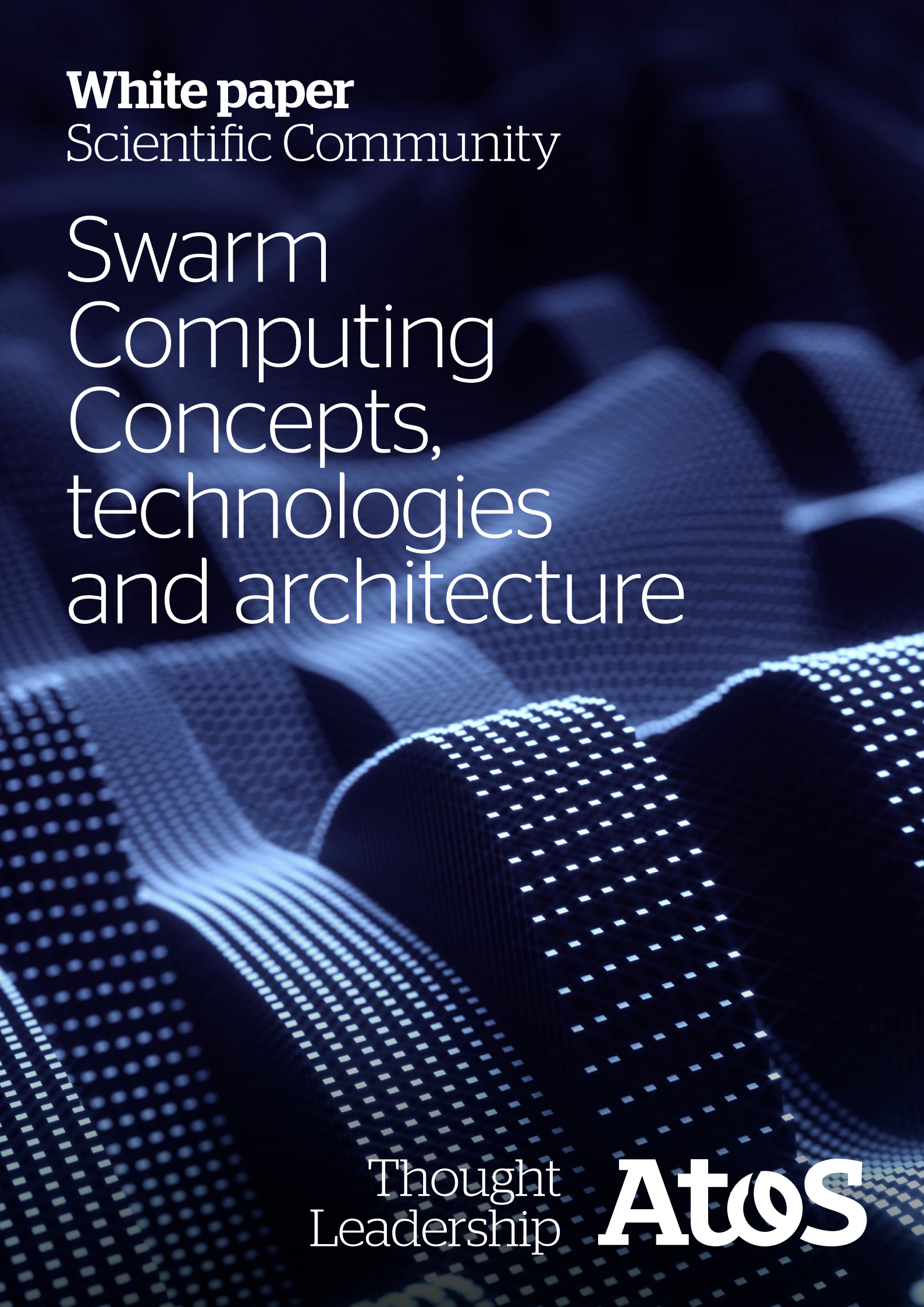
White paper
Scientific Community

Swarm Computing Concepts, technologies and architecture



Thought Leadership

Atos

Executive Summary

Swarm computing combines network and cloud principles to create an on-demand, autonomic and decentralized computing and storage management layer that transparently interoperates among diverse and disperse edge and cloud models and topologies.

It leverages on existing edge and cloud computing best practice, but with improved focus on mobility with data sharing and temporary control of connected devices.

We introduce in this paper a reference architecture for the swarm concept and illustrate it through two use cases:

- Robotics,
- and industrial swarms.

We have created a mapping of expected capabilities together with existing offerings in edge and cloud to identify the potential building blocks of the reference architecture.

About the Authors

The authors would like to thank the Atos Scientific Community, and the following members for their reviews and valuable comments to early versions of this white paper: John Hall, Hervé Barancourt, Vincent Dimitriou, Gabriel Kepeklian, Marc Llanes Badia, Yavuz Agca, Jordi Cuartero Esbri, Wim Vlak, Antoine Fressancourt, Jose Esteban Lauzan, Celestino Guemes, Ganesh Jagdale, Frédéric Oblé, Eric Monchalain and Thomas Hoberg.

This paper has been prepared by the Atos Scientific Community with contributions from:

Ana Juan Ferrer Atos Research and Innovation, Distinguished Expert
Corrado Iorizzo Atos Consulting Principal and CTO
Jeroen Staarink Solution Executive
Martin Gruber Global SAP Centre of Excellence
Wilfried Pichler Head of Engineering & Service Offerings CEE
Peter-Paul Kurstjens Account CTO, Architect & Senior Expert
Jordan Janeczko B&PS CoE PMO

Table of contents

Introduction

The emergence of IoT – the networked connection of people, process, data and things – is expected to significantly grow the number of connected devices worldwide, from the billions of units we have today, to the tens of billions expected to be deployed in the coming years.

Until now progress has been substantial in understanding IT support and services required to enable and operate such complex systems. Today, there is a clearer picture of what needs to be in place:

- the architecture needs to scale;
- big data and analytics must be advanced enough to work on these large data sets;
- and there must be a decrease in the cost of the components to support the products being connected.

However, one major change is taking place, which is re-opening some of those standard approaches and causing product managers and engineers to re-evaluate what until now has been considered best practice. As more and more devices get connected, the bigger the benefit will be from knowing what is close by and being able to interact with those connected devices, even when they are managed by a different operations' team or organization.

The obvious example is with smart home devices. People expect to be able to stream music from their phone to their smart speakers, and from their tablet to their large 4K television. When friends come over to your smart home, you'd like them to be able to stream their Netflix account to your 4K TV, or save photos from their phone on your computer, whilst also protecting smart devices such as your thermostat and oven.

Smart factories - where shop floor machines can tell robots to load raw materials, unload finished products, and have sensors so they can turn off machinery if a person approaches too closely - have become easy to design. Combining a few pieces of IT magic: edge devices and gateways, cloud computing, high-bandwidth end-to-end connectivity, and the ability to store large amounts of data for analysis to improve service. However, with today's approach, these examples are only easy if all devices are owned and managed by one person or department.

The challenge, which is requiring new strategies, is that these connected devices are mobile.

In the Smart Factory, when service technicians fix a broken milling machine or an industrial robot, they want to connect their smart diagnostic devices to analyze the logged data from the previous days and identify the problem; or understand if other equipment under their warranty is predicted to fail soon and requires attention.

This issue is only going to increase with robotics. Here are some examples:

- Smart warehouses rely on Autonomous Guided Vehicles (AGVs) – a robot dressed as a car – to help automate warehouse management. But there is no standard practice to coordinate AGVs from different companies in the same warehouse.
- When fleets of smart trucks arrive at warehouses, they are guided to the right dock for unloading and signal to forklifts what products are to be found where in the trailer, using a Blockchain Supply Chain Management System to confirm delivery, and driving away.

Robots are inherently IoT devices, but because of mobility, they will need to substantially increase their ability to become autonomous. Indeed mobile robots will need more sensors to understand their surroundings (video, LiDar, radar, Bluetooth, GPS – the list goes on); more CPU power to combine and process that information; and they need to be able to get continuous updates from and use advanced analytic algorithms to improve their actions. They will need to interact with their immediate surroundings, even if the connected devices around them belong to other people or other companies.

Because objects are connected to their own cloud, data sharing between mobile and connected devices need to support multi-cloud scenarios. Due to a significant amount of real-time processing needed with low-latency answers, either the connected device (or an edge device close by) will require enough compute power to run more advanced analytics and have their algorithms updated based on changes to their surroundings.

Swarm computing, which leverages on existing edge and cloud computing best practice, but with an improved focus on mobility, data-sharing and temporary control of smart devices, can provide a solution to this issue.

Swarm computing concept

Swarm computing combines network and cloud capabilities to create on-demand, autonomic and decentralized computing, thus taking the functionality and flexibility delivered by IoT ecosystems to a new level.

- Edge computing and multi-cloud architectures embracing the swarm vision will evolve into a set of computing and storage platforms able to provide low latency and near real-time responses with security capabilities focusing on both physical and digital entities.
- Within the swarm vision, cloud computing will evolve into a set of flexibly interoperable cloud platforms with high computing power, able to process huge data volumes with complex algorithms to devise the intelligence and deep learning from data collected by the edge platform. The processing required for immediate action and extreme low latency will happen at the edge of the network, while other processes will run in a range of cloud models. This will all be executed in an automated, self-organized and self-managed model called swarm computing.
- According to us the potential two-level polarization between edge and cloud is superseded by the concept of swarm which enables a more advanced cloud cooperation model. Looking ahead to the anticipated evolution of cloud technologies that leads to a multitude of granular services such as micro-cloud and personal cloud, we believe that complex and dynamic relationships will be established to support the cooperation with diverse edge and cloud models. This will be communicated on the basis of swarm rules and driven by the principle of subsidiarity so as to govern intelligence decentralization.

A computing swarm will therefore become an opportunistic service network. It will consist of heterogeneous resources capable of linking dynamically, flexibly and autonomously to provide assets (i.e. services, content and resources) to relevant applications and participants. Participant resources can be IoT, edge and cloud resources. Swarms are service networks deployed and executed in response to requirements, which may emerge from application / user requests for services or information, or from opportunities offered in response to data- or event-driven activities.

It is important to note that, although taking from it inspirational concepts, swarm computing is not directly related to swarm intelligence, widely employed and studied in artificial intelligence and robotics.

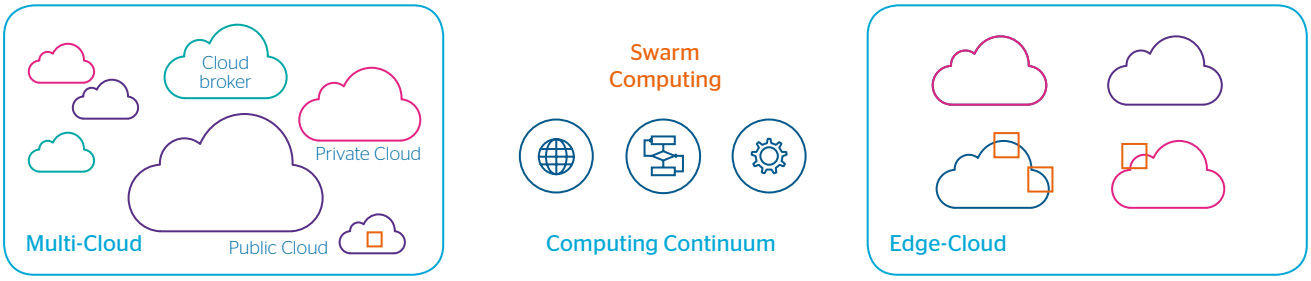


Figure 1: Swarm Computing Concept

Swarm computing reference architecture

Atos defines the swarm computing concept driven by the following IoT principles:

Aware

A swarm asset must be able to sense something about its physical or digital environment.

As examples, this might be location; proximity; altitude; light levels; motion at physical level; connectivity to peer devices; or an associated cloud richer computing environment at digital level.

Autonomous

The data gathered from a swarm asset may be processed locally or transferred to another specialized central processing application automatically — either at a set time, or when a condition is triggered (e.g. a threshold passed). This decision would be taken on site by default rather than being a manual process.

Actionable

Edge isn't just about gathering data; it's about using it to make better predictions or prescriptive decisions, and providing real-time automation.

Regardless of whether the operation of a swarm is manual or highly automated, analysis of the data must be integrated into related business processes and can involve diverse edge and cloud computing environments.

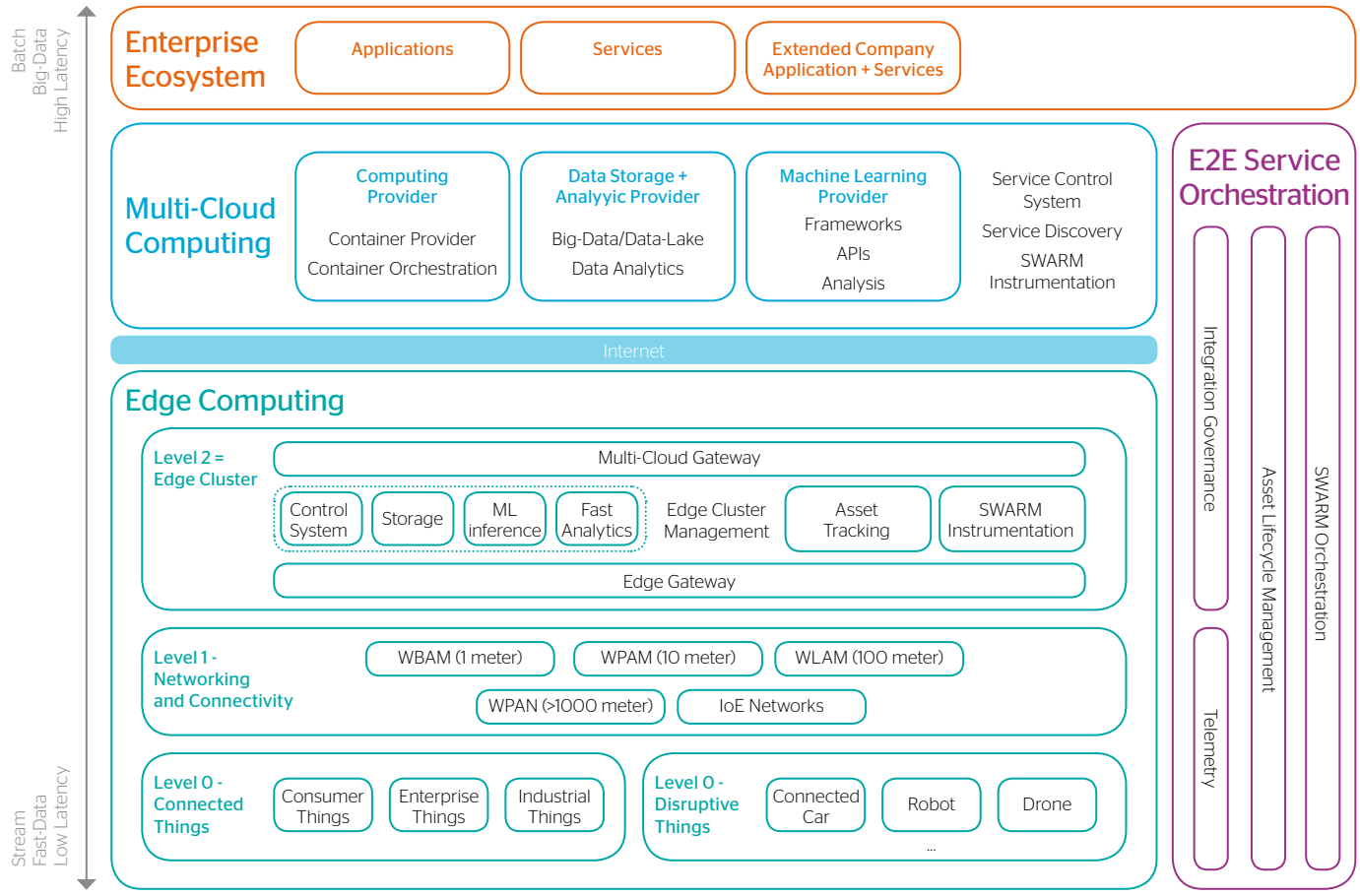


Figure 2: Swarm Computing Reference Architecture

We believe that a reference architecture for swarm computing (Figure 2) must be based upon three cooperating elements:

Edge computing

Edge computing is the combination of digital capabilities which are connecting, integrating and interacting with physical devices to collect data and track events. Based upon specific application control, physical phenomenon behavior is managed and influenced in almost real-time (e.g. industrial internet, vehicles security, robots) using actuators rather than providing specific insights to end-users aiming to take or influence decisions.

- Edge computing is characterized as a large network of interconnected devices exchanging and interacting among themselves to fulfil a collective goal (swarm coordination).
- The topology of an edge system is complex and multiple edge systems can be combined and inter-connected among them and to diverse clouds.
- The configuration, behavior and technology behind edge computing are aimed at optimizing two main critical resources: latency and asset management.

Multi-cloud

Multi-cloud works as a combination of public, private or hybrid cloud solutions at IaaS and PaaS levels, working together as a whole to deliver digital capabilities.

- Multi-cloud provides unlimited computing, processing and storage capacity; however such unlimited potential is constrained by the laws of physics, since typically a latency of 80-100 micro sec is incurred to reach a cloud provider[1].
- Multi-cloud platforms are particularly beneficial for computing intensive processing such as analyzing large amounts of data, for instance geographic information (e.g. geo-marketing, behavioral trends per territories, etc.).
- Multi-cloud platforms can also be used to provide and publish centralized services (APIs, data-lakes, micro-services) supported by the asset management capability and the centralized coordination of the overall platforms.
- Multi-cloud environments provide the freedom of choice depending on specific needs by not binding execution to a specific cloud offering.

Swarm computing

Swarm computing is the natural evolution of cloud and IoT, while at the same time, being a combination of complex multi-cloud architectures with edge computing.

We believe that the swarm approach is a disruption to existing computing paradigms forcing existing offerings that have emerged as part of a centralization paradigm, to evolve to a pure hybrid decentralized and autonomously managed model. A key characteristic of a swarm is being a temporary, limited organization, automatically assembled on-demand to address specific needs.

Swarm computing combines network and cloud principles to create an on-demand, autonomic and decentralized computing and storage management layer that transparently interoperates among diverse and disperse edge and cloud models and topologies.

Swarms are envisaged to model and abstract structural, componential, functional, and behavioral dimensions of collaborative networks, as well as actions and patterns of interactions. These will be used to create value-added, service-based outcomes that exploit the potential of information and service sources and processing capabilities of various devices.

It is important to note that swarm computing set-ups will be managed using the principle of subsidiarity of intelligence, meaning that decision making will happen at lowest appropriate level, in which the different cooperating environments are autonomous. Intelligence and knowledge about the overall status of the swarm is decentralized and spread across diverse participating instances.

At the same time the economics of participation will be managed to capitalize individuals' and communities' willingness to engage, identify and deliver assets. Time-constrained reservation, adaptive selection, conflict resolution and techniques to consider the volatility and uncertainty (introduced by real-world dynamics) will be developed to enable efficient and reliable service provision.

A key objective, but also a challenge for swarm systems, is breaking the interoperability barriers and fences which characterize the technology vendor ecosystems fostering a heterogeneous integration with standardized technologies and pluggable systems.

It is worth noting that although security is a crucial element for the implementation of swarm computing, the topic is not yet addressed by this work.

The following subsections present in more detail components and features envisaged at the three core-building blocks that represent these three cooperating levels in the swarm reference architecture.

1 Choy, S., Wong, B., Simon, G., & Rosenberg, C. The brewing storm in cloud gaming: A measurement study on cloud to end-user latency. In Proceedings of the 11th annual workshop on network and systems support for games (p. 2). IEEE Press. http://perso.telecom-bretagne.eu/gwendalsimon/data/edgecloud.pdf

Swarm computing reference architecture

Edge computing

The edge computing level is responsible for the provision of computational and storage capabilities of the resources located at the edge of the network. It is important to note that the locality of such deployment is typically bound to a specific space. This allows for extreme low latency that enables (near) real-time data process enabling data consumption and actuation in connected devices. As opposed to existing commercial offerings where edge installations rely on a single device, our architecture considers the possibility to establish clusters of these edge devices (which could be nested). The edge layer is structured into three different levels:

- Level 0 represents IoT devices and sensors that are the main data sources, but are also enabled with rapid actuation capacities;
- Level 1 components offer the connectivity services that permit extreme low latency;
- Level 2 components enable elastic computing and storage capacities at the edge, as well as access to traditional cloud computing capacities.

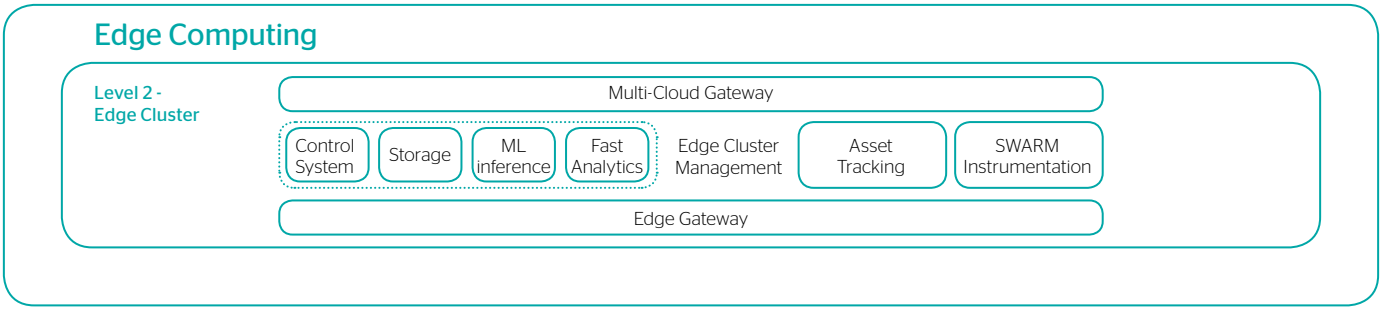
One of the main characteristics of edge computing is the heterogeneity of its devices. Particularly the envisaged characteristics in swarm computing target the so-called large devices, as in the following classification[2]:

- Tiny: very limited devices (8 and 16 bit micro controllers with less than 64kB program memory and 4kB of data memory). Example: Arduino UNO.
- Small: devices with a specific OS and restricted hardware characteristics (less than 128kB program memory and less than 64kB data memory).
- Large: devices supporting general purpose OS. Examples: Raspberry PI and Android. Although still in its infancy, it is expected that under this classification we will soon find mini-HPC, bringing high performance computation at the edge of the network[3].

Tiny and small devices are considered part of Level 0, therefore engaged with as data sources.

In order to have the ability to set-up edge clusters of the so-named large edge devices at Level 2, we foresee the following components and functionalities:

- Edge gateway: offering an abstraction layer among connected devices and other interacting components. This component must provide a unified interface that enables the collection of monitoring information about sensors' status, pulling and pushing data, interacting with actuation methods through a series of extensible plug-ins that cover the widest possible range of technologies and standards. For cases in which nested edge clusters are required, this component will oversee the provision of edge federation capabilities.
- Edge cluster management: this set of components offers complete functionality for edge computing and storage capacities.
  - Control system: this allows configuration of behavior (controls, scheduling, access, fault tolerance and service life-cycle management) within the edge cluster, over storage and analytics jobs. These could be packaged among others as containers or software functions. Specific challenges of the detailed functionalities are related to the characteristics of edge devices and their potential mobility. Device mobility and associated unreliability of connectivity, increment the so-called device churn rates; thus bringing new issues not traditionally considered in cluster and cloud computing systems. These specific resource characteristics will be crucial in the design of the control system itself. If we take the example of scheduling processes, historical information will be necessary to determine resource availability patterns in order to decide if a service can be accepted into the cluster, and which would be the most adequate placement among available resources.
  - Storage: relying on existing object storage and NoSQL databases, these components offer different storage services to be used by Level-0 sensors to push and pull data, acting as an event hub for all objects aligned to the specific edge cluster.
  - Fast analytics: the cluster needs to offer pre-configured solutions for diverse open source data-analytics frameworks to allow users to perform diverse operations with data collected by Level-0 sensors. These must consider, amongst others: data transformation across diverse formats; data aggregation and computation (to check such things as averaging data across multiple devices and performing simple computations); data enrichment (to combine generated data with other accessible metadata). Metrics gathered for data analytics are used to manage the scalability of jobs across available computation resources in the cluster, allowing interactions with cloud services if configured to do so.
  - Machine learning inference: although traditionally edge computing resources have been considered too constrained



2 HEADS Project, D3.3. Final Framework of resource-constrained devices and networks, http://heads-project.eu/sites/default/files/HEADS%20D3.3%20V1.0.pdf
3 European Processor Initiative, https://ec.europa.eu/digital-single-market/en/news/european-processor-initiative-consortium-develop-microprocessors-future-supercomputers

to be able to execute machine learning processes and workloads, increasingly solutions start to be available. Relevant examples of these are Microsoft EdgeML[4] and, notably, Google Federated Learning[5], which enables collaborative machine learning with decentralized training data. Nowadays these are only available for certain machine learning approaches and not yet widely integrated into edge offerings. However these are promising solutions for advanced edge usage models.

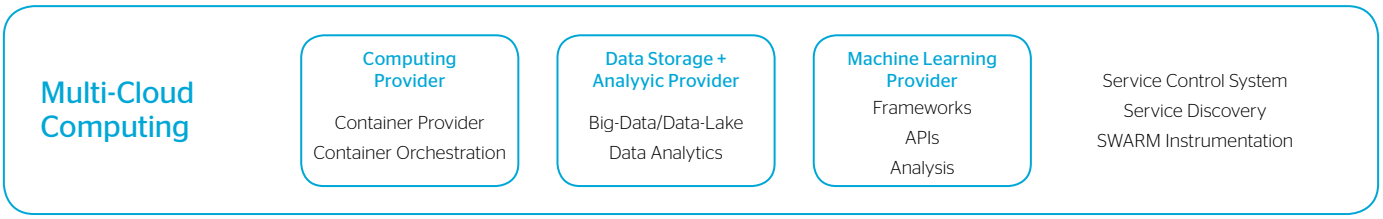
- Asset tracking: acts as the registry of devices, that keeps track of edge cluster associated devices and computing and storage resources, as well as their capabilities.
- Multi-cloud gateway: allows for a single access point to interact with both computing and storage services in traditional cloud set-ups. It is worth mentioning that it is expected to offer cloud interoperability mechanisms, similar to the ones offered by jclouds[6], for handling diverse cloud providers with the same interface to avoid being locked into a specific platform choice.
- Swarm instrumentation: provides the means for Level 0 devices and edge computing and storage resources to participate in a warm's service network. It manages edge cluster level operation of the established swarms (including processes for monitoring, evaluation, runtime adaptation, etc.). The main aim of this component is to deliver a unified description of the edge resource characteristics and capabilities, and their offered assets (i.e. resources, services and data), so to develop an abstraction layer to manage these resources. By means of such abstraction, the different interface implementations and characteristics of the diverse resources can be handled uniformly from the swarm orchestration platform.

Multi-cloud computing

The multi-cloud computing level aims at enabling interoperability among diverse cloud computing offerings while bringing highly performant resources for computation and data lakes. The components envisaged in this layer include:

- Service control system: offers multi-cloud orchestration execution capabilities for a range of cloud providers. It orchestrates resources in a diverse type of resource offering ranging from computing, data storage, analytics and machine learning services. In detail:
  - Computing provider: offering traditional IaaS and PaaS cloud capabilities.
  - Data storage and analytic provider: offering both block storage, databases as a service and data analytics capabilities.
  - Machine learning provider: services from diverse providers related to machine learning and other techniques.

  To enable the automatic selection of the most suitable cloud provider and services, multi-cloud provisioning mechanisms have to be enhanced. These mechanisms need to decide which cloud solution offers the best deployment of each service. The selection algorithms can be optimized based on the type of service to be deployed and the different application execution requirements, or by taking into account previous application execution experiences. This will allow swarms to take advantage of specific cloud providers' services that complement certain application components.
- Service discovery: there is a need to automatically discover and compose cloud services at different levels (e.g. infrastructure, data management and analytics) in order to satisfy application requirements. This enables not only the fast development of applications but also their runtime adaptation, when the respective need arises. Such an automatic discovery and composition have to be based on well-defined patterns in order to ensure the efficiency of the solution. Scalability in this context requires a decentralized approach for cloud service discovery that focuses on the distribution of data and computational load.
- Swarm instrumentation: this provides the means for multi-cloud orchestration platforms to abstract the services offered by the different cloud providers' services. Similar to the equivalent functionality at the edge level it offers a common abstraction employed by E2E swarm capabilities. Specifically, swarm instrumentation at multi-cloud level has to offer a description framework for the diverse cloud service characteristics and capabilities, providing an abstraction layer for all types of cloud services participating in the swarm platform. This will raise the need for defining a service description language, which must be able to cover both functional and non-functional aspects of different cloud services. This language should be able to describe them in different levels of abstraction and quality of service characteristics, going beyond existing standards.



4 Microsoft EdgeML, https://www.microsoft.com/en-us/research/project/resource-efficient-ml-for-the-edge-and-endpoint-iot-devices/
5 Google Research Blog, Federated Learning: Collaborative Machine Learning without Centralized Training Data, https://research.googleblog.com/2017/04/federated-learning-collaborative.html
6 The Java Multi-Cloud toolkit, https://jclouds.apache.org/

Swarm computing reference architecture

E2E service orchestration

End-to-end service orchestration (E2E) capabilities address the need for overall resource and service orchestration as the main aspect to be addressed in the swarm reference architecture.

Main components for end-to-end capabilities are:

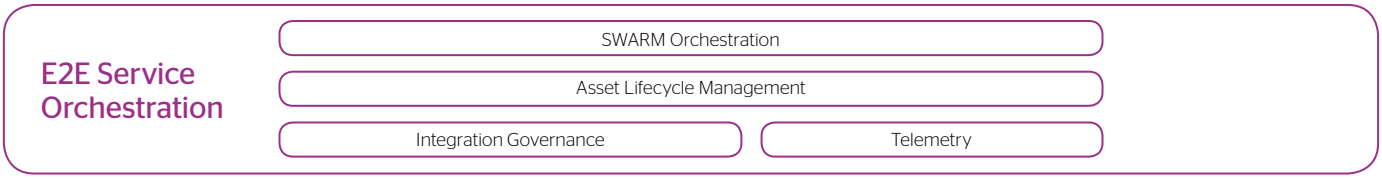
- Swarm orchestration has to be provided as an open, distributed runtime environment for decentralized management and coordination of multiple and diverse devices. This aims to overcome inefficiencies of centralized management mechanisms given the number of devices, their locality, and the fact that they can be governed by different administrative rules and principles. Swarm orchestration relies on edge and multi-cloud swarm instrumentation components in order to implement swarm coordination actions at local level. Swarm orchestration will provide a decentralized management framework that will base decisions on collaborative attributes and an entity's objective lifecycle properties. Among these we consider trust-, administrative-, location-, relationship-, information-, asset-, contextual- and environmental- lifecycle properties. It will be based on the principle of subsidiarity of intelligence, meaning that decision making will happen at lowest appropriate level, safeguarding autonomous behavior of the participating environments. Runtime adaptation and swarm repurposing mechanisms need to incorporate real-time events analysis to trigger swarm evolvement according to emerging situations, application requests, context and environment characteristics. Initially, potential swarm members will be identified via time-constrained services that will allow resources to be discovered, filtered and reserved as required by the event- or opportunity- related requirements. Discovery will seamlessly integrate techniques to consider the volatility and uncertainty introduced due to real-world dynamics, which affect the reliability of the resources (e.g. actual bandwidth of connections, reliability of the device, appropriateness of a participant for the targeted task, etc). Based on adaptive selection and conflict resolution mechanisms, the final swarm members will be identified and reflected in a concrete model that will be used for the foundation (constitution and set up) of the swarm instance. The concrete model will capture information on the software and hardware resources, data flows and network links. Swarm orchestration will also support the operation of established swarms by means of mechanisms to control, influence, monitor and predict ad-hoc interactions between swarm members. During operation, events will be detected and analyzed in real-time based on the information being exchanged, triggering actions with respect to resources (e.g. network bandwidth) and service provision driving swarm adaptation, if necessary.

- Asset lifecycle management: asset management will consider a scalable and dynamic object discovery and reservation framework. This aims to provide an up-to-date image of the available assets that are candidates for a swarm, and allow objects to join, leave, or change their participation level in the candidate group. The discovery mechanisms will allow objects to join the resource pool and register their assets (ie resources, data and services) as candidates for utilization. Object registration and resource discovery products will be stored in a distributed data store. The data store will be optimized for the kind of queries expected from the resource in different layers, managed by swarm instrumentation and management and coordination layers at swarm orchestration.
- Integration governance: governance approaches in swarm management and orchestration have to rely on self-governance and participant-based reputation rating, targeted at exploiting to full capacity the assets provided by the swarm participants. Models for evaluating, classifying and incentivizing the participants will be necessary for the realization of different service-based situations, in addition to mechanisms to monitor and evaluate participant behaviors that target engagement.
- Telemetry: telemetry and monitoring frameworks consist of a set of agents that collect measurements and statistics from the entities to be monitored and are typically co-located with them. The monitor collects the statistics from all agents and aggregates them, producing the monitoring output. The current gap in the state of the art lies in monitoring frameworks for very large-scale infrastructures (from tens of thousands to millions of entities) able to correlate all the monitored entities in a holistic way. Swarm orchestration requires of a framework able to monitor and correlate metrics of very large numbers of entities recording numerous metrics per second for a large number of swarm participating elements. Furthermore, the framework will be distributed and highly elastic, to serve the uniform growth from small-scale to large-scale swarms.

The concept of swarm first emerged in the field of robotics[7]. Swarm robotics refers to the coordination of multi-robot systems consisting of large numbers of simple robots. Interactions among the robots with elements of the environment cause the emergence of collective behaviors between simple robots focusing on the same task.

Using this concept as baseline and taking into account existing developments in IoT, edge and cloud computing, swarm computing aims to bring this concept to the computing world.

In this section we will use two examples to illustrate the concept.



Swarm realization

Example 1: the robotics swarm

Generally speaking, robots are constrained devices in terms of computational, storage and energy resources. Robots' designers and producers are under pressure to provide increasingly complex behaviors and skills at competitive costs. Robots progressively require more and more sophisticated software systems to provide ever broader spectrums of functionalities, and this may exceed their on-board processors' capacities. Increasing on-board computation of robots raises their costs and energy demand while reducing autonomy. Emerging next-generation mobile robotics trends point in the direction of using small, general purpose cheap on-board processors and software defined robots as means of achieving new skills, cognitive capabilities, and greater flexibility.

For robots to perform a task such as moving a part, recognizing or grasping an object requires a significant amount of processing power as well as contextual and stored data and information. Capacities of robots with regards to computation and storage capacities are limited.

A robot can be seen as an aggregated set of resources encompassing sensors, actuators and computing and storage capacity that collectively implement a cognitive loop. The external extension of these capabilities with additional computational, storage and service solutions is not a new concept and is being investigated under the cloud robotics field. They can also be isolated entities. They will need to be able to have interactions beyond the robot system and establish mechanisms for data and knowledge crowdsourcing and sharing so as to be able to acquire knowledge about their operational environment.

The advent of even more sophisticated systems such as humanoid-like robots will exacerbate these needs. To this end, the robot swarm concept is articulated though three computational levels: on-board computation on the robot; computation at edge (providing closer computational capacities); and finally, resource richer multi cloud computing environments. Through the orchestration of these three elements a swarm emerges in which knowledge and intelligence is created.

Aware

Robots are equipped with autonomous navigation systems, such as localization, map building and path planning relying on robots' odometry, inertial and LIDAR sensors, and considering highly dynamic and changing environments.

Autonomous

The data processed by the robot may be processed locally and / or transferred to a central processing application automatically – either at a set time, or when a certain condition is met or a threshold passed.

Required services have to be deployed as close as possible to access point. Service management is autonomic and distributed, including replication and migration. Behavior can be improved with access to fast processing and environmental data.

Multi-robot coordination and cooperation between them alongside edge and cloud services can support reliability.

Actionable

Actuation not only requires data gathering; it's about using it to make better decisions and provide real-time automation.

It requires trade-offs among:

- dynamic and real-time response to fast processing (i.e. for simulation and optimization tasks i.e. image recognition processes),
- as well as access to vast amounts of data for simulation and algorithm training.

7 Beni, G., Wang, J. "Swarm Intelligence in Cellular Robotic Systems". Proceed. NATO Advanced Workshop on Robots and Biological Systems, Tuscany, Italy, June 26-30 (1989). doi:10.1007/978-3-642-58069-7_38.

Swarm realization

The three layers permit an incremental acquisition of computational capabilities by off-loading processes among the three computational levels which enrich the computational capacities available at each level. But also, remarkably, they enable the management of three levels of knowledge about the environment:

- The individual robot's knowledge, acquired by its own perception capabilities;
- The edge knowledge level, which encapsulates knowledge acquired by robots deployed in a certain location under the influence of a specific edge environment. This allows multiple robots to share gathered knowledge, thereby building edge-awareness, and this can be further enriched via sensor network data in the surrounding environment.
- Swarm cloud level, knowledge gathered by underlying levels is consolidated into a platform knowledge, enabling high level knowledge sharing, learning and exploitation of gathered information by using rich computing environments in cloud.

In robotic applications, managing global information about the environment requires a great capability for saving and processing the data. This may include:

- Building maps of the environment to localize or navigate;
- detecting changes and dynamism;
- generating global plans in order to achieve objectives;
- adapting to new situations that may appear during a task;
- and the ability to re-plan tasks or objectives according to those new situations.

These are some of the capabilities that autonomous robots must have.

In many applications, for example logistics in warehouses or hospitals, multi-robot teams are able to carry out tasks more efficiently - but they need to share information captured by each of them to optimally distribute the tasks, compute global plans for the team in a centralized way and build a situational awareness through knowledge sharing in order to execute plans as a distributed system in a swarm environment.

Example 2: the industrial swarm

IoT for manufacturing, known as industrial internet of things (IIoT) in an industry 4.0 context, is a fundamental part of modern manufacturing today. Introduction of IoT into factories and supply chains is enabling the evolution of smart product service systems which raise production efficiency. Industry 4.0, the factory of the future and IIoT are important trends all focusing on the digitization of the manufacturing sector. They consider embedded sensors in virtually all product components and manufacturing equipment, ubiquitous cyber-physical systems (CPS), and analysis of all relevant data[8].

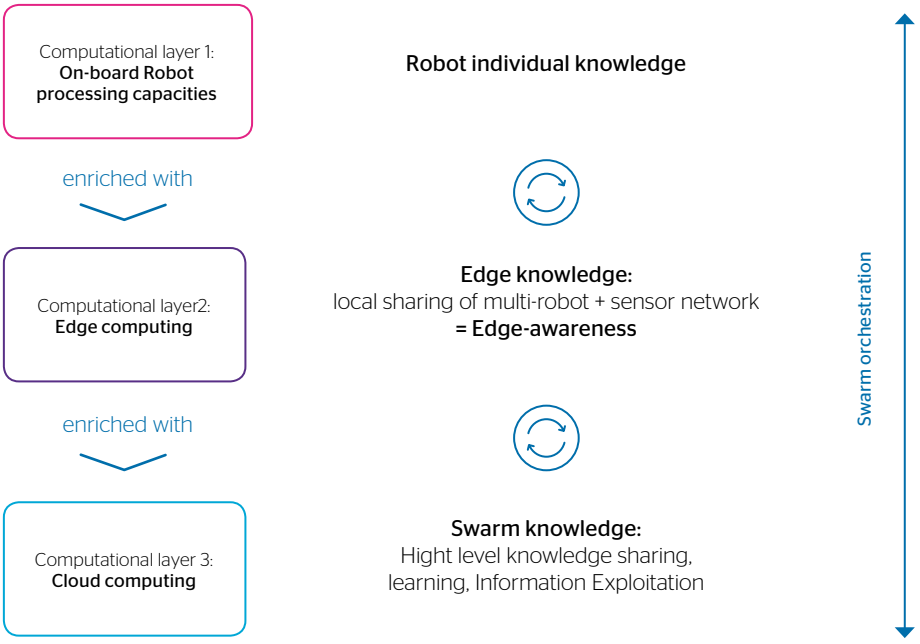


Figure 3: Computational levels for Robot Swarm

Aware

Manufacturers and suppliers require accurate and up-to-date visibility at each point of the supply chain in order to optimize transportation and logistics processes.

Supply chain managers need to get data from warehouses, vehicles, packages, and materials. They must be able to handle equipment from any device, such as GPS and RFID readers.

Autonomous

Fueled by collected information from sensors and data-enhanced supply chains, supply chain managers need to be able to predict, prevent and correct situations that could or are affecting the smooth functioning of the supply chain. These issues can range from failed equipment and inventory outages to safety risks. Access to this information can ensure optimal efficiency of the supply chain at all times.

Actionable

In IIoT context, actuation presents clear needs for interconnection and interoperation within existing systems. Examples of these can be: product lifecycle management system is enhanced so to add in-service product performance measuring and monitoring; enterprise resource planning systems are capable of detecting failures and issue work orders for components needing replacement or repair; customer relationship management platforms can improve forecasting based on production status and better anticipate supply chain adjustment needs.

CPSs have been traditionally designed to make use of the resources available in-house. Increasingly CPS, whilst considering safety first, are now looking at cloud and edge solutions in order to reduce costs and enhance scalability. Cyber-physical cloud computing aims at addressing these needs by providing "a system environment able to rapidly build, modify and provision cyber-physical systems, composed of a set of cloud computing-based sensor, processing, control and data services". The emergence of this trend raises new challenges. On the one hand, factors such as dynamic aggregation / disaggregation of behaviors, consolidation, economic incentives, and auto-scaling of resources would have to be considered in CPS design and operation. On the other, cloud and edge computing would require advancements in specific aspects such as QoS management, reliability and fault management, data-clustering, security, safety, ethics and real-time support in order to enable the specific need of CPS. Here is where swarm computing comes into play, by providing a more dynamic execution environment, offering coordinated management across industrial IoT devices, edge and diverse cloud environments. Swarm computing could then become a key technology for industrial automation, in line with IIoT objectives.

In this context, edge computing pushes the intelligence, processing power and communication capabilities of an edge gateway or appliance directly into devices (i.e. programmable automation controllers) and ensures it is closer to where the data originates from (for example the sensors, pumps, motors, relays, etc). Whilst this takes place, multi-cloud environments offer rich computing networks as well as interoperability with existing elements in the industrial automation software stack. Swarm computing could provide the mechanisms for smart metering of material flows through smart coordination of edge and cloud capabilities. This would enable the detailed monitoring of material outside and inside the plant; this would help to better predict the expected arrival time of materials and allow deliveries to be "just in time" (i.e. in connection with process control system warehouse management). This would provide benefits in terms of cost optimization for the whole production system.

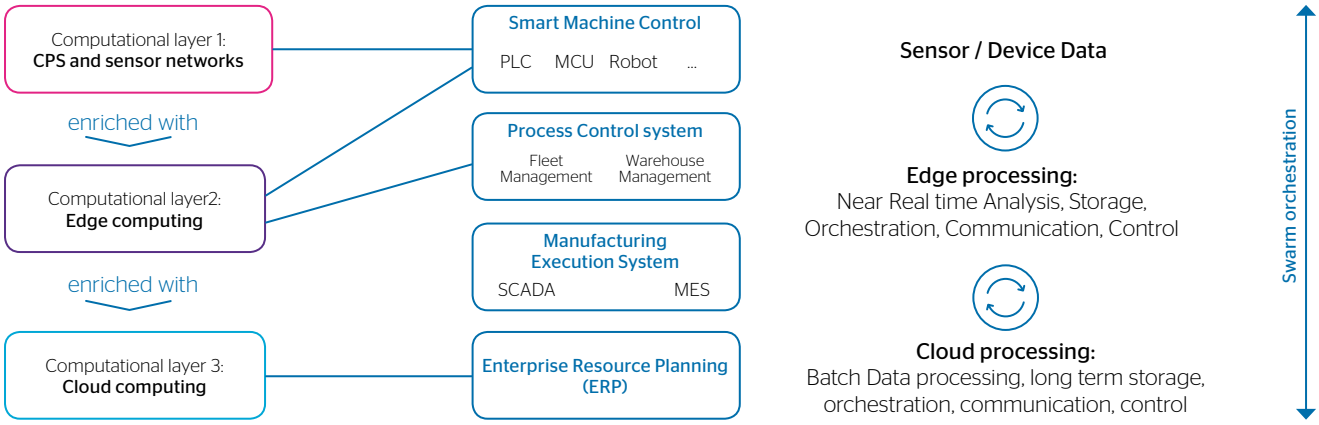


Figure 4 Computational levels of Industrial Swarm

8 D. Wee, R. Kelly, J. Cattell, J. Cattell and M. Breunig, "Industry 4.0 - How to navigate digitization of the manufacturing sector," McKinsey Digital, 2015.

Swarm status: existing market products

While forecasts for the expected number of connected devices and things are controversial[9], IoT adoption is a reality that will obviously only increase in the near future. Based on this expected evolution, vendors are increasingly building specific IoT cloud and edge computing offerings which complement established cloud offerings and can provide the necessary solutions for the realization of swarm computing.

To implement the swarm concept, we do not aim to build it from scratch, instead, we plan to rely on existing features and develop these. With this purpose we have analyzed IoT (edge and IoT cloud) solutions available in May 2018 from seven vendors to assess their characteristics and potential contributions to the swarm. In each of them we have checked available features at edge and cloud layers. To the best of our knowledge, multi-cloud features at cloud IoT service level are not yet present in market offerings. Findings of this analysis are summarized in the table below detailing characteristics of products available at the time of writing this whitepaper. It should be noted that this analysis does not aim to provide a complete market overview of edge and cloud IoT services but to share examples of available offerings that could act as swarm computing concept building blocks.

	Edge Capabilities						Cloud Capabilities				
Vendor	Control system	Asset Tracking	Storage Management	Fast Analysis	ML inference	Cloud Gateway	Control System	Computing Provider	Data Storage	Analysis Provider	ML Learning Provider
CISCO IoT Edge Compute[10]	●	●	● Limited	● Depends on user application							
Dell Edge gatewas for IoT[11]	●	●	● Limited	●			●	● Cloud Foundry	●		
Intel, Evolution of Edge C.[12]	●	●		●		●	●	● Not clear details			
Azure Stack IoT[13] / IoT Edge[14]	●	●	●	● Azure Functions	●	●	●	●	●	●	●
AWS IoT[15] / Greengrass[16]	●	●	●	● AWS Lambda functions	●	●	●	●	●	●	●
Android Things[17] / Google IoT[18]		●			●	●	●	●	●	●	●
SAP Leonardo[19]			●	●	● Planned	● SAP Cloud Native	●	● Cloud Foundry	●	●	●
Siemens MindSphere[20]	●	●	●	●	●	●	●	● Cloud Foundry	●	●	●

9 Popular Internet of Things Forecast of 50 Billion Devices by 2020 Is Outdated, http://spectrum.ieee.org/tech-talk/telecom/internet/popular-internet-of-things-forecast-of-50-billion-devices-by-2020-is-outdated
10 CISCO IoT Edge Compute https://blogs.cisco.com/developer/iot-edge-compute-at-its-best-apps-and-deployment-on-cisco-iot-gateways
11 Dell Edge gateways for IoT, http://www.dell.com/us/business/p/edge-gateway
12 Intel, Evolution of Edge Computing, https://www.intel.es/content/www/es/es/communications/evolution-of-edge-computing-paper.html?wapkw=edge+computing
13 Azure IoT Suite, https://azure.microsoft.com/en-us/suites/iot-suite/
14 Azure IoT Edge, https://azure.microsoft.com/en-us/services/iot-edge/
15 AWS IoT, https://aws.amazon.com/iot/
16 AWS Greengrass, https://aws.amazon.com/greengrass/
17 Android Things, https://developer.android.com/things/hardware/index.html
18 Google Cloud IoT, https://cloud.google.com/solutions/iot/
19 SAP Leonardo Edge Computing, https://www.sap.com/documents/2017/03/068181cb-ae7c-0010-82c7-eda71af511fa.html
20 Siemens Mindsphere Whitepaper, https://www.plm.automation.siemens.com/media/global/en/Siemens_MindSphere_Whitepaper_tcm27-9395.pdf

Conclusion: challenging but essential

Swarm computing aims to address new business models around digital markets by creating novel opportunities at the intersection between people, businesses and things. Connecting people, devices and services into digital ecosystems has been recognized as challenging but essential for the coming years[21].

New research is required on how to deploy, operate and manage these heterogeneous and highly distributed services, which will require high levels of automation and innovative skills. In addition, future complex digital ecosystems will increase the need for solutions able to adapt to specific user needs at scale in a highly dynamic way.

To meet the needs of industry, Atos is further investigating swarm computing with specific emphasis on management and architectural approaches. Examples of these research works are collaborations in projects such as mF2C[22] and DITAS[23].

Future activities focus on further progress in certain technological aspects such as security and data management not yet addressed in this work. Furthermore, we will study requirements and opportunities in determined vertical sectors such as energy grids.

21 Gartner Top 10 Strategic Technology Trends for 2018, https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2018/
22 mF2C Project, http://www.mf2c-project.eu/
23 DITAS Project, https://www.ditas-project.eu/

About Atos

Atos is a global leader in digital transformation with 120,000 employees in 73 countries and annual revenue of € 13 billion. European number one in Cloud, Cybersecurity and High-Performance Computing, the Group provides end-to-end Orchestrated Hybrid Cloud, Big Data, Business Applications and Digital Workplace solutions through its Digital Transformation Factory, as well as transactional services through Worldline, the European leader in the payment industry. With its cutting-edge technologies and industry knowledge, Atos supports the digital transformation of its clients across all business sectors. The Group is the Worldwide Information Technology Partner for the Olympic & Paralympic Games and operates under the brands Atos, Atos Syntel, Unify and Worldline. Atos is listed on the CAC40 Paris stock index.

Find out more about us
atos.net/scientific-community

Let's start a discussion together

CT_J514_181012_RY_WP_SWARM