Ref. Ares(2021)1537606 - 28/02/2021

PHYSICS

OPTIMIZED HYBRID SPACE-TIME SERVICE CONTINUUM IN FAAS

D2.1 – MARKET ANALYSIS AND GAPS IDENTIFICATIONS

Lead Beneficiary	GFT
Work Package Ref.	WP2 – Requirements, Architecture and Technical Coordination
Task Ref.	T2.1 – Market Analysis and Gaps Identifications
Deliverable Title	D2.1 – Market Analysis and Gaps Identifications
Due Date	2021-02-28
Delivered Date	2021-02-28
Revision Number	3.0
Dissemination Level	Public (PU)
Type	Report (R)
Document Status	Release
Review Status	Internally Reviewed and Quality Assurance Reviewed
Document Acceptance	WP Leader Accepted and Coordinator Accepted
EC Project Officer	Mrs. Maria Tsakali

H2020 ICT 40 2020 Research and Innovation Action

This project has received funding from the European Union's horizon 2020 research and innovation programme under grant agreement no 101017047

CONTRIBUTING PARTNERS

Partner Acronym	Role[1]	Name Surname[2]
GFT	Lead Beneficiary	
FTDS	Contributor	
ISPRINT	Contributor	
DFKI	Contributor	
CYBE	Contributor	
HUA	Contributor	
RHT	Contributor	
UPM	Contributor	
INQ	Contributor	
INNOV	Peer Review	
DFKI	Quality Assurance	Volkan Gezer

REVISION HISTORY

Version	Date	Partner(s)	Description
0.1	2021-01-11	GFT	ToC Version
1.0	2021-02-08	GFT, FTDS, ISPRINT, DFKI, CYBE, HUA, RHT	1st Version
1.5	2021-02-15	GFT, HUA, RHT, INQ, INNOV	Version to Peer Review
1.7	2021-02-23	GFT, HUA, UPM	Version to second Peer Review
2.0	2021-02-26	GFT, DFKI	Version for Quality Assurance
3.0	2021-02-27	GFT	Version for Submission

[1] Lead Beneficiary, Contributor, Internal Reviewer, Quality Assurance
[2] Can be left void

LIST OF ABBREVIATIONS

Abbreviation	Complete Name
AI	Artificial Intelligence
APAC	Asia-Pacific
API	Application Programming Interface
BFSI	Banking, Financial Services and Insurance
BSN	Body Sensor Network
CaaS	Container-as-a-Service
CAGR	Compound Annual Growth Rate
CD	Continuous Delivery
CI	Continuous Integration
CLI	Command Line Interface
CNCF	Cloud Native Computing Foundation
CO2	Carbon Dioxide
CPU	Central Processing Unit
CRD	Custom Resource Definition
CRISPR	Clustered Regularly Interspaced Short Palindromic Repeats
CRON	Command Run On
CSP	Cloud Service Provider
CTO	Chief Technology Officer
DevOps	Development Operations
EMEA	Europe, Middle East and Africa
EU	European Union
FaaS	Function-as-a-Service
FoD	Features on Demand
GAFAM	Google, Apple, Facebook, Amazon, Microsoft
GDPR	General Data Protection Regulation
HPC	High Performance Computing
HTTP	HyperText Transfer Protocol
I4.0	Industry 4.0
IaaS	Infrastructure-as-a-Service
IEEE	Institute of Electrical and Electronics Engineers
IoT	Internet of Things
IT	Information Technology
ITES	Information Technology Enabled Services
M2M	Machine-to-Machine
MiFID	Markets in Financial Instruments Directive
MPI	Message Passing Interface
MVP	Minimum Viable Product
OCI	Open Container Initiative
OS	Operating System
P&L	Profit and Loss
PaaS	Platform-as-a-Service
PESTLE	Political, Economic, Social, Technological, Legal, and Environmental
PHYSICS	oPtimized HYbrid Space-time servIce Continuum in FaaS
PL4	Production Level 4
PSD2	Payment Services Directive 2

QC	Quality Control
RAM	Random-Access Memory
R&D	Research and Development
RWD	Real World Data
RWZ	Raiffeisen Waren-Zentrale Rhein-Main eG
SaaS	Software-as-a-Service
SAM	Serverless Application Model
SLA	Service Level Agreement
SME	Small and Medium Enterprise
SOA	Service-Oriented Architecture
SWOT	Strengths, Weaknesses, Opportunities, Threats
SysOps	System Operations
UCAS	Universities and Colleges Admissions Service (in the UK)
VM	Virtual Machine
WBAN	Wireless Body Area Network

EXECUTIVE SUMMARY

Deliverable D2.1 performs an analysis on the FaaS market in terms of size, growth, profitability, cost structures, trends and critical success factors identifying gaps and potential for impact. The document specifies and analyses the most valuable prospects, aiming to provide a factual introductory basis for the requirements, technical choices, architecture, and exploitation activities within WP2 and WP7.

The aforementioned task has been performed through comprehensive desk research based on market reports and scientific papers on top-tier databases and primary research. The authors conducted interviews with end-users and consultants with robust expertise of the market and customers' aspirations.

The deliverable's main findings have been synthesized in a SWOT analysis and in a list of core challenges and critical success factors. Together, these elements have been necessary to conceptualize the value proposition. This latter highlights how European FaaS Cloud framework could be offered to customers by providing access to a scalable, capacious, secure, compliant, and cost-effective services. Besides, it would allow users to benefit from the pay-per-use pricing model, leverage the auto-scaling of FaaS, and provide equal access to all players who wish to use the Cloud-based service. FaaS services providers, indeed, have been thoroughly analysed to identify the impact, gaps, and potential business models. Furthermore, the developed service would-be built-in compliance with GDPR inside the FaaS environment, relying on a cost-effective external, user-friendly efficient infrastructure with extended storage capacity for coding and development.

Lastly, the deliverables' findings will set the basis for both a comprehensive understanding of FaaS and the PHYSICS project's potential benefits. Specifically, the results of this report will serve as major inputs for both T2.2 dealing with reference scenarios for the PHYSICS platform and T7.2 that will focus on business modelling and the exploitation strategy of the project. As such the present document will be used as a useful background for the scientific, technical, and business exploitation work that will follow the project.

TABLE OF CONTENTS

1. INTRODUCTION

Deliverable D2.1 explores and outlines PHYSICS technologies' market and the potential for impact where gaps are identified. Building upon the insights derived from the analysis conducted in the first chapters, the document specifies core challenges to overcome and the corresponding critical success factors in order to offer a value proposition which can create gains and relief pains to end-users according to their needs and aspirations.

In the first chapter, the document's objective, and an introduction to the leading Cloud Computing Services and, more in detail, FaaS technology are briefly described. Then, the next three chapters of the document explore the following:

- The industry, sectors of interests, and trends (Chapter 2);
- Services offered by the leading players active in the market, users' needs, encountered problems, goals in different sectors, and stakeholders in the dimension of interest and power (Chapter 3);
- Infrastructure, architecture, and development tools of PHYSICS technologies (Chapter 4).

Next, the deliverable performs a SWOT analysis which synthesizes the insights of the previous chapters. Drawing from the SWOT analysis, the deliverable lists the core challenges which must be overcome, by obtaining specific critical success factors, to meet users' aspirations.

Conclusively, the document conceptualizes a value proposition which takes into account the analysis conducted, challenges outlined, and success factors identified.

Regarding the methodology, the deliverable's central insights have been obtained through desk research based on academic and industry sources. It has also been fundamental to conduct interviews with users and consulting professionals, who have a holistic view of different market sectors' needs. Popular frameworks such as PESTLE and SWOT analyses have been exploited to outline the external environment and synthesize insights, respectively. Finally, in order to visualize the value proposition designed, Osterwalder's canvas has been used.

1.1 Objectives of the Deliverable

The objective of the Deliverable is to analyse the FaaS market, identifying relevant gaps. The deliverable aims to explore this technology's impact in different market sectors, outline the leading players while mapping relevant stakeholders, and investigate critical success factors. The deliverable will summarise end-users' goals, issues and aspirations, and analyse these against the previous research through a SWOT analysis. Finally, a study on successful value propositions and associated business models will be conceptualised.

1.2 Cloud Computing Services

There are five main categories of Cloud Computing Services:

- IaaS (Infrastructure-as-a-Service) allows users to exploit only the fundamental cloud infrastructure. Therefore, it requires experienced engineers for its configuration and efficient utilisation.
- CaaS (Container-as-a-Service) allows users to exploit cloud's applications relying on the provider's orchestration platform.
- PaaS (Platform-as-a-Service) allows users to develop, run, and manage apps relying on the provider's infrastructures, saving platform maintenance costs.
- FaaS (Function-as-a-Service) allows users to run specific app's functions individually. Hence, it is labelled as serverless architecture.
- SaaS (Software-as-a-Service) allows users to access software "on-demand" without requiring any server maintenance.

Companies are moving to a public cloud for various reasons such as scalability, cost savings, data security and simplicity (no need to have your architecture and hire engineers to build, run and maintain it). Moreover, cloud services allow companies' employees to work from anywhere and increase virtual collaboration, becoming even more crucial during national lockdowns in 2020 and 2021 due to Covid-19 Pandemic (Scroczkowski, P. n.d.).

The main reason for utilizing these services is linked to cost savings and flexibility in accessing the amount of resources they need, whenever they need them and in a pay-as-you-go fashion. Indeed, companies with large cash assets could decide to build, run, and maintain their in-house infrastructure and servers instead of paying for the services listed above. However, this is happening exceptionally rarely. Indeed, even state governments rely on the services offered by the leading players of the market instead of running their platforms and building their infrastructure (IBM Developer, 2019).

1.3 Function-as-a-Service (FaaS)

The PHYSICS project aims to focus on one of the most promising services of Cloud Computing, the FaaS Services. FaaS allows functions to run in milliseconds and deals with individual requests and then the process ends. The main characteristics which distinguish FaaS from other cloud computing models are that it does not need to rely anymore on external remote servers and that this service is instantly scalable. The pricing model is another differentiating factor as the billing is based on actual consumption made and the execution of the functions. The FaaS model is generally used preferably to create micro services (web apps, data processing, chatbots and IT automation for instance, etc.)

For these reasons, companies in different industries are considering adopting FaaS cloud services to develop applications and deploy functionalities without having to deal with the underlying infrastructure. Moreover, during the Covid-19 Pandemic, the need for "work-from-home" services has dramatically increased the demand for FaaS services. Although new forecasts are not available at this time, we expect that recent estimates may predict an even higher growth rate for the future years.

A publication by Allied Market Research in 2018 estimated the FaaS market to be worth $3.01 billion. They estimated a year-on-year growth of 29.7% from 2020 to 2026, which would eventually reach an industry valuation of $24 billion (Allied Market Research, 2020). Other analyses forecast a Compound Annual Growth Rate (CAGR) of around 33%, which would value the market at $30 billion by 2026 (CB Insights, 2018).

Observing the deployment model, the largest industry share belongs to the public cloud segment, which accounted for nearly 60% of the total market in 2019. Even though it is expected to keep its dominant position, the hybrid cloud segment is expected to grow the most, with a forecasted CAGR of 34.8% from 2019 to 2026. Likewise, observing the user type, the largest share belongs to the developer-centric segment, which accounted for nearly 60% of the total market share in 2019. The dominant segment is expected to maintain its position. Still, the operator-centric segment is projected to grow the most, with a forecasted CAGR of 31.3% during the studied period (Allied Market Research, 2020).

2. INDUSTRY OVERVIEW

2.1 Market Sector Exploration

This section aims to explore FaaS impact on different industries on both existing markets and on markets with potentially high growth.

Following a publication by Markets and Markets, which estimated the FaaS market to be worth $7.72 billion by 2021, and the industry breakdown shown in the figure below, we estimated the percentage of each industry over the total market and the size in 2021 for each sector.

- Banking / financial services (25% - 1.93 billion in 2021)
- Telecommunication and ITES (21% - 1.544 billion in 2021)
- Consumer Goods and Retail (16% - 1.158 billion in 2021)
- Health care and Life Sciences (13% - 0.9264 billion in 2021)
- Government and Public Sector (10% - 0.772 billion in 2021)
- Others (15% - 1.158 billion in 2021)

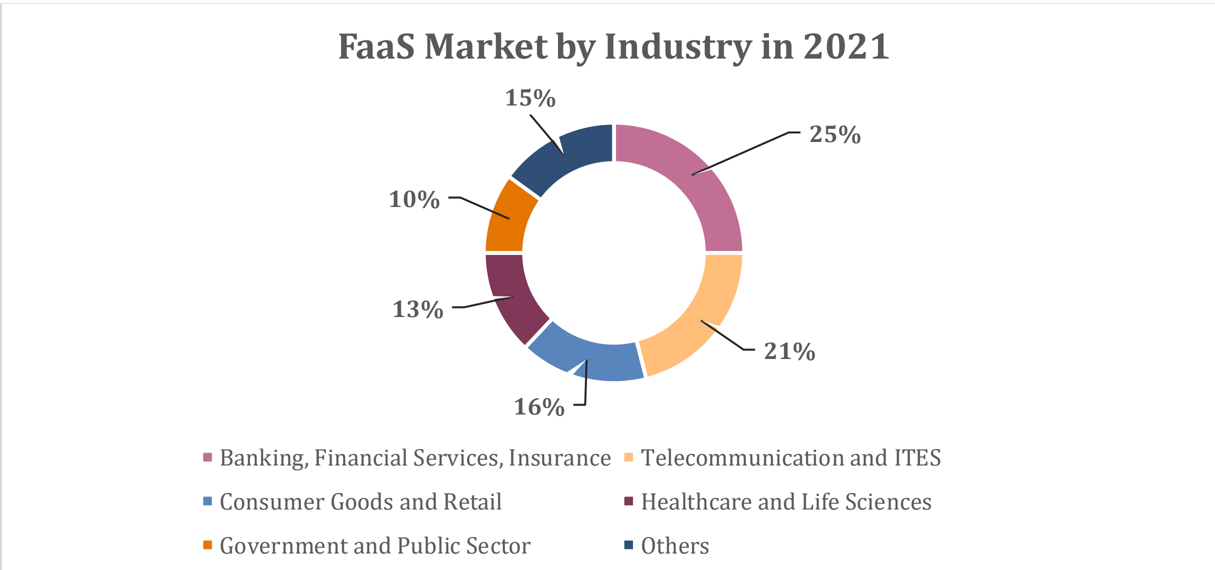

Figure 1 - FaaS Market by Industry. Source: Markets and Markets, 2017

<u>Financials</u>

The Financials sector encompasses the Banking, Financial Services (or Diversified Financials) and insurance industries: the BFSI segment. FaaS technology is widely adopted across the Financials sector. This segment is continually evolving and characterised by an intense competitive landscape, worldwide expansion, consolidation, varied regulatory norms, and ever-changing consumer demands. FaaS can support real-time applications and provide a scalable and secure platform for providing real-time information as required by the BFSI sector; hence, this sector offers substantial opportunities for the FaaS market's growth. Financial institutions are increasingly using FaaS for automation of routine operations and facilitating cost reduction. Moreover, FaaS services can enable a more comprehensive view of customers and financial products and further drive customer retention and acquisition. FaaS services allow the efficient integration of multiple delivery channels for banks, which has further helped drive these services' growth in the BFSI (CloudSecureTeach, 2016).

On further notice, observing the serverless nature of cloud and FaaS technology, an alternative universe of decentralized web is constituted by the today popular blockchain technology. A common element is the serverless digital nature of both. Indeed, nobody has responsibilities for decentralized blockchain networks. It should be noted that this technology is expected to grow in development and importance in the future, potentially leading to enhanced services and protocols that may develop similar characteristics to FaaS technology and that could, potentially, substitute it (Yuan, 2020.).

<u>Telecommunication and ITES</u>

The sector made up of telecommunications and internet service providers has experienced a sharp increase in cloud computing usage. By 2025, the cloud telecom industry is forecasted to be valued at $ 50.77 billion with a compounded annual growth rate (CAGR) of 20.89%. Specifically, FaaS cloud computing enables secure and hyper-personalised experiences with reduced costs as customers become more demanding. Since the liberalisation of the telecommunications markets, there has been a rapid increase in competition in this sector. To maintain competitive advantages, telecom companies use FaaS to develop and implement new business and payment models to ensure constant customer engagement and loyalty.

Consumer Goods and Retail [Consumer Discretionary and Staples]

For the sake of brevity, Consumer Discretionary and Staples are combined, when it comes to the use of FaaS in the Consumer Goods sector. On the one hand, the Consumer Discretionary sector, including Automobiles and components, Consumer durables and apparel, Consumer services, Retailing follows cyclical trends. It tends to be more income-sensitive than consumer staples due to its higher income elasticity of demand. On the other hand, the Consumer Staples sector encompasses the following industries: Food and Staples Retailing, Food, Beverage and Tobacco, Household and Personal Products.

The consumer goods and retail sector is one of the fastest-growing industry verticals owing to rising consumer purchasing power. With the advent of online retails, traditional retailers adopted more innovative technologies such as cloud computing, big data analytics, digital stores, and social networks. The vendors in a particular industry need a comprehensive cloud environment to offer their respective product and service portfolio. The Consumer Goods sector faces the challenge of managing instability and change in demand for various products due to high competition. FaaS helps seamlessly integrate the multiple shopping channels and presents a unique shopping experience to the connected consumer.

Healthcare and Life Sciences

The Healthcare sector includes two primary industries: the Healthcare equipment and services and the Pharmaceuticals, Biotechnology and Life sciences. The Covid-19 crisis amplified the rising demand for improved healthcare around the globe. For this purpose, benefits such as enhanced data usage, medical research, and lowering costs drive the market.

The sector is governed by the need for tools to rapidly create and implement business process integrations and healthcare data transformations. Efficient data management techniques are critical for different activities such as patient management, hospital resources management, doctor-patient relationship management, medical supplies management, and maintaining data for patients' health records. The industry has shifted into a model that collectively supports and coordinates the workflows and medical information. Clouds are efficient in storing extensive data, facilitating the sharing of information among physicians & hospitals, and increasing the data analysis & tracking features.

Public and private healthcare suppliers on all levels of care face a demand to develop their systems and improve the served population's health while reducing accounting costs and enhancing profitability indicators. Organisations may face the need to share healthcare data, further contributing to an efficient platform requirement. FaaS service types enable organisations to integrate their systems with other healthcare networks to share information and work together in an orchestrated manner to accommodate growth, improve agility, and further help healthcare providers optimise and increase the effectiveness of their services delivered to customers.

Before the Covid-19 pandemic, the e-Health sector experienced significant upsides in demand with seamless value offerings ranging from online consultations to patient management and preventative medical measures. As a part of the European Union political priority called "A Europe fit for the digital age", political opportunities are inferring growth prospects for e-Health application providers. For instance, the commissions' e-Health Digital Service Infrastructure will allow e-prescriptions and patient summaries to be exchanged between healthcare providers.

The aforementioned policies are a part of a larger framework - the Digital Single Market, which aims to improve cross-border connectivity and increase digitisation where application development is a significant keystone. One of the main drivers for e-Health is the fast access to a healthcare provider and the possibility of taking care of medical matters without forcing the consumers to visit the clinic. Telemedicine is expected to reach a global $180 billion valuation by 2026 and the total digital health market a valuation of $500 billion by 2025. Today, the values respectively amount to $45 and $88 billion.

In the current market, applications such as Livi[3], DoctoLib[4] and the National Health Service[5] are connecting medical staff with patients and given the big room for growth that is predicted to persist post-Covid due to convenience factors. The use of such applications is thus expected to increase. However, one pivotal factor to consider in e-Health is the risk of providing sensitive data to third parties. For this reason, an extra precaution in the design and choice of the cloud is highly relevant.

One key technology currently developed with FaaS in the Healthcare sector is WBAN - Wireless Body Area Network (also referred to as BSN - Body Sensor Networks). It aims at providing real-time healthcare monitoring services. As WBANs have limited memory, energy and computing power, a scalable high-performance computing and storage infrastructure, such as FaaS, is required to provide real-time data processing and storage.

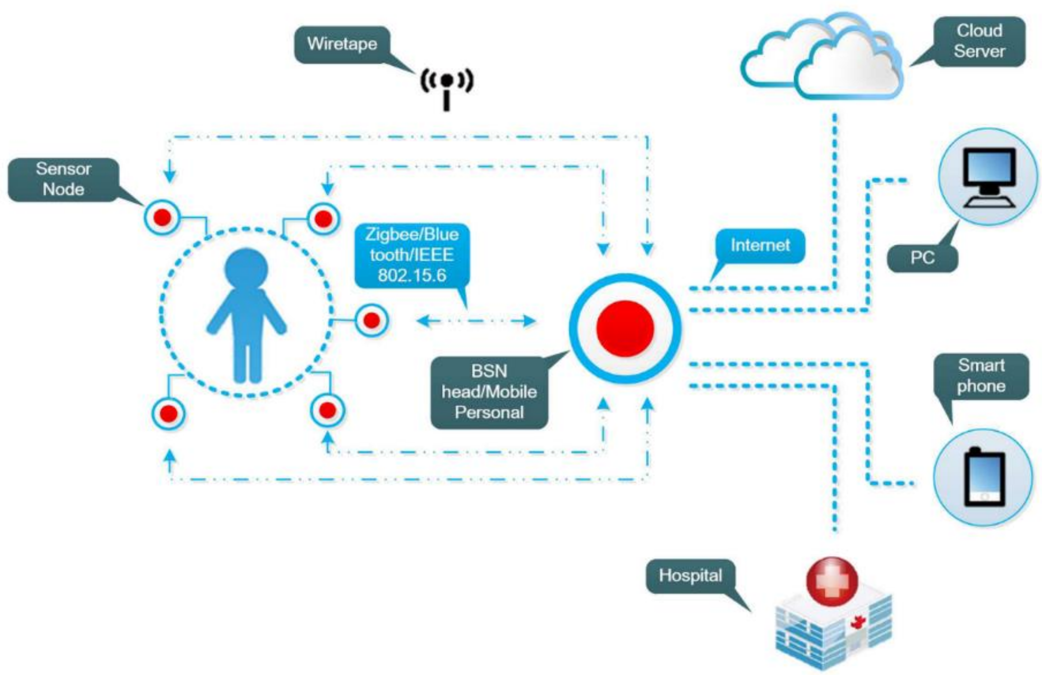

Figure 2 - WBAN - Wireless Body Area Network *Source: Foglets, 2019*

WBAN technology is predicted to be used in various areas such as remote health monitoring, home/health care, medicine (constantly measuring blood pressure, heart rate) and sports.

<u>Government and Public Sector</u>

This sector must deal with various issues such as tax collection, public interest, safety, and education. As consumers experience is enhanced by digitalisation in private sector businesses, users' expectations of government agencies' services demand superior service in terms of quality and efficiency. Hence, there has been a surge in government agencies moving to the cloud in order to deliver better services to citizens. The cloud facilitates these agencies to make fast and cost-effective capacity decisions, scaling up and down depending upon data traffic, seasonality, and business requisites. FaaS services enable government and

[3] https://www.livi.co.uk/

[4] https://www.doctolib.fr/

[5] https://www.nhs.uk/nhs-services/online-services/nhs-app/

public agencies to extend tailored services to handle mixed consumer demands. Government departments are increasingly using FaaS services to provide government alerts, manage air services, manage accommodations at national parks and many other such applications.

Others

We have identified several other promising market sectors which are not listed in the analysis above since their market size is still relatively minor, which are the following:

Media and Entertainment

The media & entertainment sector comprises businesses that produce and distribute motion pictures, television programs and commercials, streaming content, music and audio recordings, radio, book publishing, video games and supplementary services and products. The use of FaaS companies involved in this sector can manage their processes and workflows in an efficient, scalable, flexible and reliable manner. FaaS is deployed mainly to store and manage business-critical data due to new data formats with uncertain revenue and monetisation. Moreover, media and entertainment companies use FaaS to develop workflows that improve their digital supply chain while reducing operating expenses.

While many entertainment companies opt for established cloud players for their business, Netflix chose an alternative path (Gienow, 2018). Despite Amazon's renowned data storage application EC2[6], the largest video streaming company developed its own FaaS platform to power the next-gen Netflix API. This platform processes and orchestrates all Netflix clients' requests to backend systems, leveraging new technologies such as Docker, Node.js, and Amazon ECS. The main issues to be resolved by the FaaS platform were the need for non-required assembly, reliable and automated updates, observable metrics and managed operations. As a result, the platform consists of pre-assembled containers with all of the components needed for a production-ready service. By creating a development FaaS platform for local development, engineers can interactively test functions in seconds, reducing friction and increasing velocity. Therefore, the local FaaS platform can be integrated and routed within the Netflix cloud, enabling a seamless end to end testing. In terms of flexibility, auto-scaling is used to reduce costs further and increase availability. To resolve the visualisation issue, metrics and dashboards are automatically generated for each function to have full visibility.

Additionally, alerts are automatically generated based on metrics to promptly act on problems or even resolve them before causing any significant disruption. Being wholly owned by Netflix, the platform's infrastructure and operations and the application itself is handled by the centralised API platform team. UI teams are only responsible for managing individual functions.

Agriculture

Agriculture plays a crucial role in our society. It is estimated to represent a means of living for 70% of the world population. It is undeniable that agriculture will play a key role in Europe's near-future challenges: production efficiency in the context of ever-scarcer natural resources, rural people empowerment, and sustainable agri-food industries development.

As the perhaps most successful example of regional economic integration, the EU agricultural sector has generated job opportunities, made billions of meals, and contributed to a net added value of €181 billion according to Eurostat in 2018.

The lack of advanced technological integration concerning accounting cost has led to lower efficiency per spent Euro on the market – hence the great demand for Smart Agriculture. Inefficiency can be accounted for many reasons including lack of scale returns when looking at various current assets' indicators during

[6] https://aws.amazon.com/it/ecs/?whats-new-cards.sort-by=item.additionalFields.postDateTime&whats-new-cards.sort-order=desc&ecs-blogs.sort-by=item.additionalFields.createdDate&ecs-blogs.sort-order=desc

increased output levels. This issue is explained by the fact that the need for tangible assets is parallelly developing with an increase in demand - but marginal gain is insignificant.

With the development of precision agriculture and smart agriculture, productivity will be multiplied, and resources will be used more efficiently. Today, traditional cloud models used in the Agriculture sector are inadequate to handle the large amounts and variety of data generated by the IoT (Internet-of-Things) devices connected. Indeed, the usage of advanced technologies such as sensors, IoT satellites and drones to the profit of precision agriculture has substantially increased the data needed to be processed.

FaaS technology will lower resource allocation in production and effective plant management through precise agricultural FaaS services. The need for real-time and optimised management technology reduces dead weight, increases profitability, and contributes to a more stable and robust European food security supply. Indeed, latency is a crucial aspect as it must enable real-time decision making. Hence, the "Fog computing" technology, described as an extension of the cloud, provides agriculture with an attractive solution. Moving computation from the cloud to edge devices can support IoT services with fast response time and low bandwidth usage. It uses decentralised local network architecture to speed up the analysis and retrieval of data near the source. Hence, it moves the processing abilities closer to the data source, realizing more rapid M2M (Machine-to-Machine) communications. However, one downside is the lack of flexibility. Therefore, Function-as-a-Service (FaaS) appears as a promising programming model for Fog Computing to increase operational flexibility, improve the system performance and reduce ownership costs.

Manufacturing

Innovation and internet technologies have already started to disrupt the market with tangible effects already visible. Observing the services of companies such as Uber [7]or Lime[8], they have become possible thanks to increased connectivity, mainly through the broad adoption of internet and smartphones.

Observing the coming future, it is likely that future applications will be developed and imported into the mobility sector, changing the market that we see today. Established companies, such as Volkswagen, working with Ford, Daimler and BMW, have started partnerships with Google, IBM and Microsoft to create new offerings in a modern ecosystem of services. FaaS enters the game to deliver Features on Demand (FoD). In Volkswagen's vision, customers will be able to download and incorporate in their car's characteristics that were not included at the moment of purchase, adding new features as we do today for smartphones. As a consequence, applications must be developed to be potentially available for a wide range of users, and it will be paramount to rely on a framework capable of elasticity and scalability of resources. These features must be suited to satisfy the needs of few and, if they become mainstream, to satisfy a greater demand without compromising performances.

To give a glance at services already developed that are likely to be widely used in the future, Volkswagen has created functionalities such as We Parking, that allows finding free parking space. It offers a cashless way to pay for them. We Experience uses AI to deliver in-car merchant partners' offers, for instance, from coffee chains and petrol stations. Finally, We Deliver solves the problem to get packages securely placed in the car when there is nobody at home, turning the trunk of a vehicle into a parcel delivery station (Fisher, 2018).

Energy Sector

One of the newest technologies used in the Energy sector is Smart Grids. Smart Grids are developing networks of transportation lines, equipment, controls and new technologies working together to respond immediately to our 21st Century demand for electricity. More specifically, they represent a decentralised network of many small power producers, small-scale transmission lines and regional supply compensation,

[7] https://www.uber.com/it/it/

[8] https://www.li.me/electric-scooter

supporting the flow of energy in both directions and enabling the household's active participation in the system. This concept of Transactive Energy optimises energy production, energy transmission, and energy distribution by combining the traditional grid with next-gen hardware and software. The outcome is a far more cost-effective, efficient, reliable, and resilient power grid.

As this new system requires a considerable amount of data and information to operate, FaaS could be a viable alternative for many energy companies and grid operators. Companies would no longer need to build, manage, and maintain such a system's underlying infrastructure. They could prospectively solely focus on developing their energy business models and maximising their operations' business logic. By investing in serverless technology and using "pay-per-use " plans, ownership costs would consequently be reduced, and performances would be increased.
The most significant risks of using FaaS in the Energy sector, which has to be considered in such an analysis, are the loss of control over specific processes, the reduction of privacy/confidentiality, and more difficult debugging in system failures or computer-hacking incidents.

2.2 PHYSICS Focus on three prominent sectors:

Following the analysis of the different market sectors in which FaaS is currently deployed, we decided to highlight three key industries with high potential for FaaS Services: Healthcare, Agriculture and Manufacturing. The piloting applications in these three domains in the context of PHYSICS were selected based on criteria such as the following:

- Diverse application domains covering a multitude of environments affecting EU citizens, including the e-Health domain (especially in the light of recent and future pandemics), the Manufacturing domain (given the high shift of production towards external to EU countries) and the Agriculture domain, related to the need for key resources (water, electricity, GHG emissions, fertilizers) optimization
- Complementarity in terms of edge resources used, spanning from lightweight IoT edge and mobile devices (e-Health) to medium-sized on-site servers (Agriculture) and high-end edge nodes (Manufacturing)
- Complementarity in terms of application services needed in the interplay, from HPC related services (e.g. MPI simulations in Agriculture) to big data, analytics and machine learning (e-Health) and AI (Manufacturing)
- Complementarity in terms of application implementation state, spanning from more legacy type components with high "FaaSification" needs (e.g., MPI simulations in Agriculture), modern features but not of the FaaS model (e-Health) and more distributed SOA-based components (Manufacturing) Following, more details per scenario are portrayed.

Even though FaaS technology is already used in the Healthcare industry, it needs to take a step forward with the e-Health sector's opportunities and Wireless Body Area Network (WBAN). National governments will launch significant investments after the Covid-19 pandemic, and allocating parts of these investments to the sector digitalisation will be pivotal. Regarding the Manufacturing and Agriculture industries, the increasing need for flexibility, scalability, efficiency, and better allocation of resources makes them critical sectors for the deployment of FaaS. Remarkably, the perspectives offered by the combination of Function-as-a-Service and Fog computing would significantly enhance these industries. Hence, this section of the deliverable aims to provide three Use Cases on the industries highlighted above.

2.2.1 FOCUS – Agriculture Use Case

Greenhouses are the most sophisticated way to control plant environments to increase their production, reduce impact of climate uncertainty, provide physical barriers to diseases, and enable substantial reduction of chemical pesticides. However, they require more and more parameters to be set by the grower (e.g., 200 in a standard soil-less glasshouse used for tomatoes). Consequently, parameters are mostly set to default values, without adaptation to the location of the farm, the needs of the species and the cultivar, their potential in yield and quality (dry matter and sugar content). Thus, a more dynamic, online process should be pursued in order to gather collected data, model and continuously optimize parameter set in the greenhouse.

In previous works on greenhouse vegetables, developed greenhouse modelling solutions significantly improved crop management and yield estimation, including savings of 50-100 €/ha/day of CO2 (92% of CO2 cost) and reduced emission of liquid CO2 of 90% on tomato crops, while yield and quality estimations on salad crops reached 90-95% of precision. The applicability in greenhouses could be substantially increased by a more connected and more reactive "digital twin", processing in quasi-RT the greenhouse's meteorological data (hourly to daily reactions depending on the actuators). The uncertainty would be strongly reduced by automated data assimilation and simulation iteration.

The current software needs a full deployment in the cloud; it can only be used in R&D sites and not in real time. The latter leads to limitations both in the number of simulations run and model update in the production environment. In precision agriculture, SMEs wish to apply Data Science for easing the configuration of the computations' distribution between edges proposed by PHYSICS. The former would be an asset and abilities offered to customers in near real-time. The PHYSICS project aims to address the challenges faced to date with two key objectives:

- Enable the more straightforward adaptation of the complete application graph in the FaaS model and the process from a sporadic offline one to a continuous monitoring and operation function. The foreseen CSP Cloud Design Environment and relevant cloud design patterns will significantly enhance the ability to migrate existing modelling and simulation components that are based on legacy technologies and interconnect them with a distributed application, able to integrate the overall lifecycle, from data collection and forwarding to launching of the simulation, obtaining the results and applying them at the greenhouse level at a fine-grained, continuous monitoring and operation level.
- Optimize running costs, automate distribution and management of components and application performance. Automation of application management and the selection of proper external services on which it would be deployed, while linking and managing the various distributed elements at the greenhouse and externally.

Cybeletech is primarily a modelling company and does not have the expertise in distributed systems management and operation. Furthermore, cost and performance optimization from the FaaS model usage would significantly benefit the specific type of application, given that it consists of multiple, short-duration simulations (1 to 5 seconds). Therefore, it would directly benefit from the pay-as-you-execute mode of FaaS and performance-driven management achieved in a PHYSICS-enabled CSP resource used.

2.2.2 FOCUS – Manufacturing Use Case

SmartFactory-KL Production Level 4 (PL4) Demonstrator provides a modular architecture with service-based software elements and skill-based production modules. These properties enable a plug & produce manufacturing line following I4.0 procedure. The software behind the demonstrator already contains decoupled components (Figure 3).

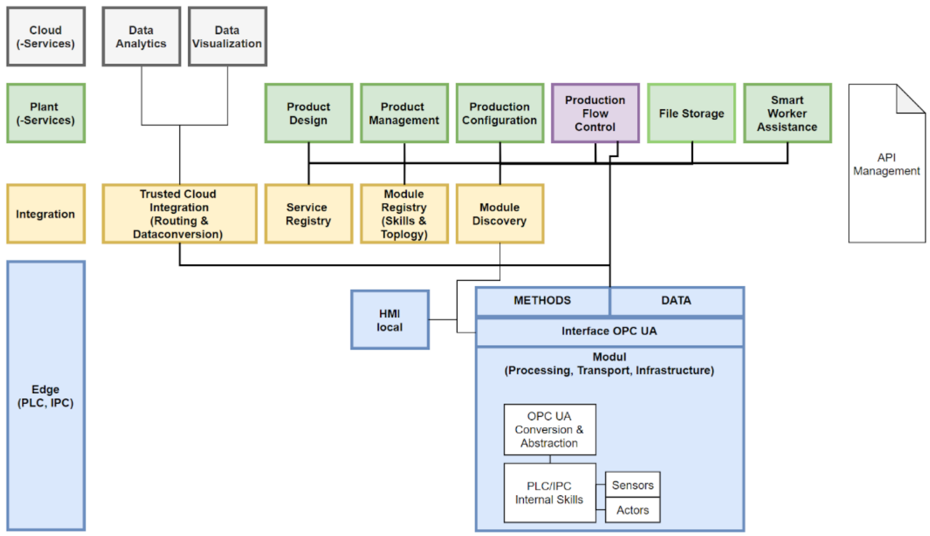

Figure 3 - SmartFactory-KL Demonstrator Software Components

The software components are hosted on servers, physically located at SmartFactory-KL premises. Each software component is individual and working independently on their single servers. However, as of today, there are no redundancy mechanisms which enable a fail-safe system in case of a breakdown. Moreover, no load balancing features were considered, which could otherwise help decrease the computation time. For example, the quality control (QC) module needs to perform Artificial Intelligence (AI) computations to evaluate the product status. If a new product arrives in that module before the previous computation is complete, the production rate will decrease. In an ideal case, the computations which will take longer to compute could be offloaded to more powerful servers/computers to keep the production rate stable, or even to increase it.

With the PHYSICS project, it will be possible to implement the aforementioned points to have an environment that is close to the ideal. By implementing orchestration mechanisms - no matter where the physical servers are – the production rate could be kept stable, and the reliability could be improved. The current implementation of the demonstrator enables an easy migration of the infrastructure into a FaaS infrastructure. A downside to be considered might be an increased number of integration and deployment pipelines. Therefore, the process must be automated in the future with CI/CD-approach. At the end of the project, the demonstrator is expected to be even more flexible, scalable, and more resource-aware to make the optimal decisions to execute its tasks. The project will also help the manufacturing domain to monitor the health of the services automatically, which are typically performed manually. Apart from these benefits, the acceptance might be critical as manufacturing is a high conservative domain requiring high SLAs and policies. Architectures are usually still monolithic and are still based on old concepts. Protocols and components are still very proprietary and designed for the specific application.

2.2.3 FOCUS – Healthcare Use Case

Technological advances are driving the uptake of e-Health services across the globe. Given an additional boost by the COVID-19 pandemic, and the need to reduce physical contact between individuals, more and more people are using remote monitoring and treatment tools to manage their health, whether suffering from mild- or chronic conditions or going through a process of rehabilitation. With e-Health technology, users (sometimes patients) can monitor their quality of life, keeping track of their daily health-related habits and adverse events like migraines, or athletic injuries and how those might affect lifestyle, health, or quality of life.

Innovation Sprint is a high-tech SME involved in e-Health through the provisioning of their Healthentia platform. This is a cloud platform that can collect individual health data from end-users, process and reason on this data, and provide derived knowledge or detailed analytics to professional users (e.g. healthcare professionals) through dashboards at the portal (web) app, or use this knowledge to provide feedback and coaching towards the end-user at the mobile app. Figure 4 below provides a high-level overview of the Healthentia solution, depicting the Mobile application and connected phone sensors, the Healthentia platform receiving data from 3rd party Wearables (IoT Edge resources) via their respective clouds (e.g. Fitbit or Garmin) and a number of possible FaaS services (Data Simulator, Clustering and Prediction).

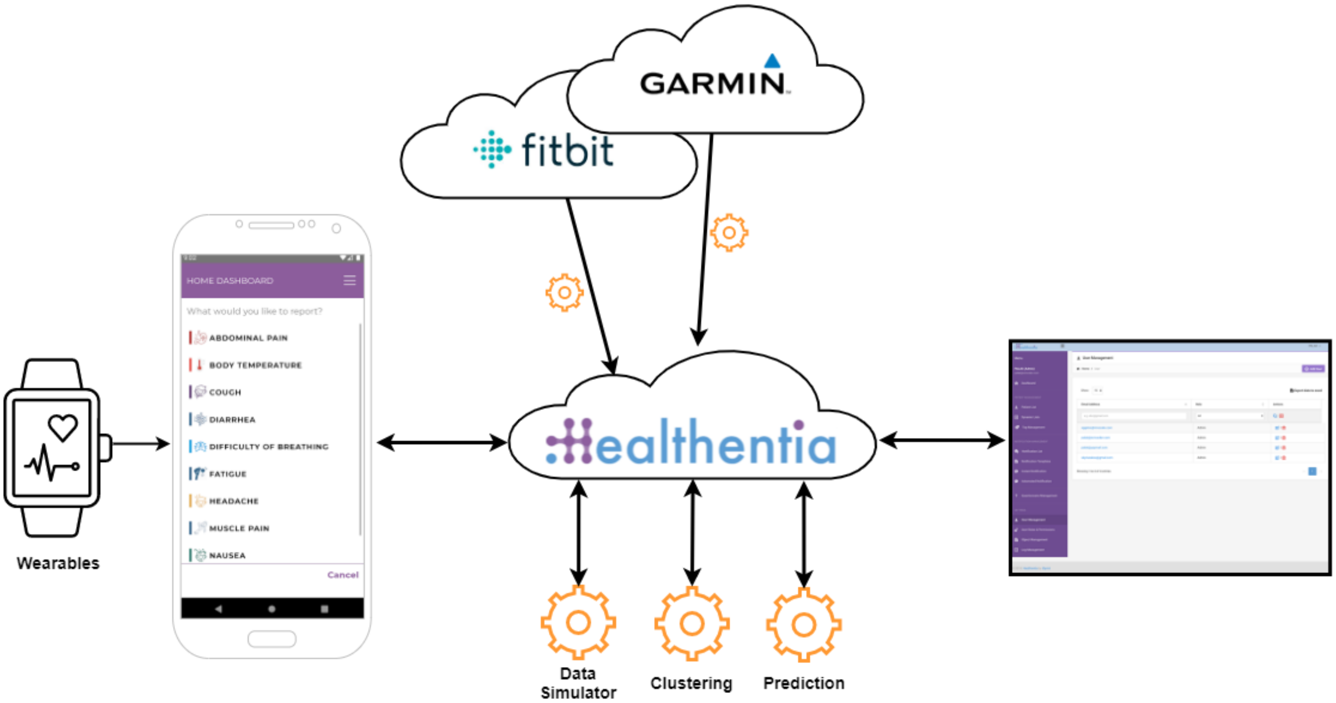

Figure 4 - *Schematic overview of the Healthentia e-Health platform, including mobile and portal (web) application, wearables, and potential FaaS services.*

Healthentia aims to employ FaaS oriented tools for analysing the collected patient-generated Real-World Data (RWD), making them actionable for decisions. The RWD will be processed to establish habits and trends, i.e., short-term variations from the long-term patterns. Machine learning and AI algorithms then derive each condition's behavioural biomarker under investigation and cluster the users into groups. The biomarker acts as predictor or regressor for the different outcomes (usually future objective RWD values). The analysis of its individual decisions identifies the essential attributes of the patient's behaviour, ultimately used to coach users by, e.g., pushing questionnaires, providing necessary feedback, or engaging with the user in a virtual coaching dialogue. The patient clusters are used to derive behavioural phenotypes, i.e., models of behaviour applicable to groups of people employed in the generation of synthetic data in the Data Simulator.

The areas identified in Figure 4 with the orange cog wheels indicate those processes that would benefit from the scalability and ease of development offered by the FaaS paradigm. Data connectors, retrieving and processing 3rd party data (e.g., from Fitbit or Garmin platforms) need to run reliably and often for each user in the platform, where especially new users with large amounts of existing data pose potential threats to these services' availability and performance. The Data Simulator, Clustering and Prediction algorithms are all complex stand-alone services that benefit from dynamic scaling and a transparent and decoupled error handling and debugging process.

Healthentia operates within the highly regulated medical domain and is classified as a Class I medical device. As such, all data processing operations need to adhere to strict guidelines regarding, e.g. privacy preservation and security. When moving into the FaaS paradigm, extra care should be taken in handling such aspects.

2.3 Trends Outline

<u>Trends and Growth</u>

The global cloud computing industry was valued at $266.0 billion in 2019, and the market is expected to further grow with a Compound Annual Growth Rate (CAGR) of 14.9% on a 7-year basis. Driving factors behind this growth includes but is not limited to general digital transformation among the global industries, increased penetration of IoT items across consumers, and the increased demand and consumption for big data. In terms of FaaS which is on the pathway for higher growth prospects with a CAGR of 29,7% and a $24 billion industry valuation by 2026, main future growth drivers are within IoT and cloud infrastructure services.

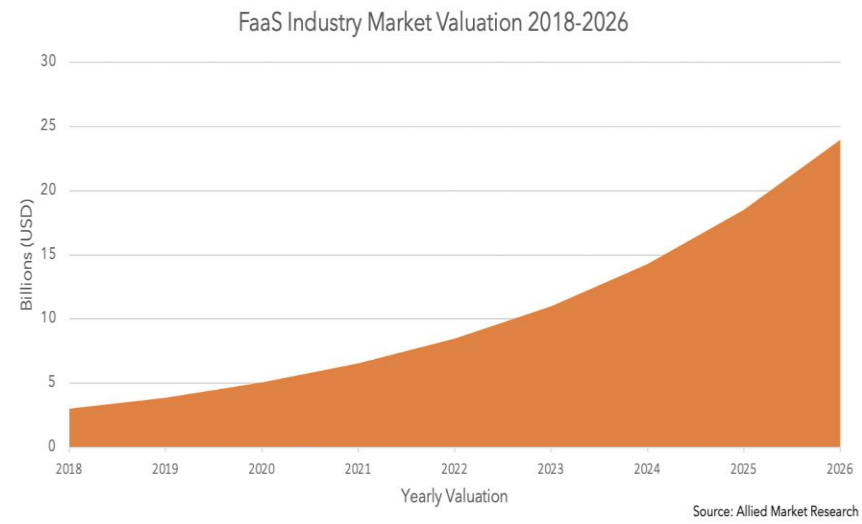

Figure 5 - Total Valuation FaaS Industry

Concerning trends, two main factors seem to be significantly demanded: (1) ease of deployment and (2) the ability to manage current cloud infrastructure assets. The need for ease of deployment stems from the variation in programming languages and the lack of functions to be transposed in an Application Programming Interface (API) to ensure resource allocation successfully. Due to the nature of operations being easily scalable as they are event-driven, positive synergies and value creation are two critical metrics expected to increase using a FaaS provider.

Accounting for the bifurcation in service provider performance in a Covid-19 business environment, demand towards an asset-light and serverless infrastructural approach has led to FaaS providers' growth. As digital transformation is increasingly growing among traditional sectors (where the use of digital solutions is commonly low) and human capital is scattered from one central production point, FaaS can accommodate the continued development of application services and solutions without the need to construct and maintain infrastructure and servers.

When looking at deployment models concerning segments, the public cloud segments made up 75% of 2019 revenue and is expected to maintain its position as the largest segment for the forecasted period 2019-2026. Nevertheless, the highest CAGR of 34.8% is found in the hybrid cloud segment. The hybrid cloud model addresses a higher security and scalability level - two major justifications for choosing FaaS and mitigating risk. As enterprises increasingly need to keep their sensitive data secure in private clouds, while benefiting from the on-demand scalability and cost-effectiveness of public clouds, the hybrid cloud market segment is the fastest-growing deployment model. Due to concerns about data security and privacy, the hybrid cloud is the most preferred cloud deployment method in Europe, especially in Germany, France, and the United Kingdom. As companies realise the advantages of hybrid cloud and other cloud computing models (such as enhanced business agility and secure operations), the adoption of hybrid cloud is expected to grow substantially, increasing market share overall. Regarding user type, the developer-centric segment made up 50% of FaaS market share, while the operator-segment is poised for the highest growth at 31.3%.

PESTLE-Analysis Framework

Table 1 – PESTLE Analysis

Category	Determinant	Prices	Costs	Profitability
POLITICAL	EU-Competitiveness framework	↑	↓	↕
	No new major Data Regulation or Political Reforms	-	↓	↑
ECONOMIC	Increased demand of applications	↑	↓	↑
	Low cyclical association	-	-	↑
SOCIAL	Democratisation of services	↑	↓	↑
	Covid-19 Application Development Rationale	↑	↓	↑
TECHNOLOGICAL	Scalable deployment and liberalisation of FaaS	-	↓	↑
	Effectivize private clouds	↓	↓	↑
LEGAL	Data Compliance Design (GDPR, BFSI-regulations)	-	↑	↓
	Risk exposure related to third party API's	-	↑	↓
ENVIRONMENTAL	Digital transformation positive externalities	-	↓	↑
	Power-efficient by design	-	↓	↑

Political

European firms are too small to sustain competitive operations on a singular basis and need cooperation and support from the European Union, which shortens the relative distance between governments and businesses. APAC has a comparative advantage in the cost of production and human capital resources from a cost-based perspective. At the same time, US-counterparts cover the largest market share, most robust business intelligence with respect to implementation on the market, and have the most extensive balance sheets of assets and attractive value offerings to customers as per aforementioned data. For European firms to compete, diversification and innovation have to be the leading comparative advantages globally as the APAC and US comparative advantages are less realistic on the European market structure. Niche initiatives such as the PHYSICS project and integrations of digital economies via the Digital Single Market as a whole are pivotal political forces to foster a competitive environment and innovation.

Compliance and Regulations are both aspects of the Political and Legal parts of the PESTLE. With respect to Political, the focus is on the political processes in the European Union and the intended political effects of such proposals. With more restrictive data regulations comes a political environment that strengthens the individual's consumer rights while imposing a more regulatory burden on the business. It can thus far be inferred that the EU, compared to other political bodies, are more interventionist in their approach to data given the extensiveness in the latest set regulations in relation to equivalent governing legislatures.
As of right now, GDPR-compliance as a single regulation aspect is no major factor affecting FaaS due to early adaptation and a design environment that allows enhanced logging and privacy-preserving functions. As of now, there are no substantial present political risks related to FaaS that would disrupt operations. There are no anticipated major data regulations in the pipeline that would lead to drastic FaaS compliance changes. Furthermore, no regulatory aspects on a national and EU-level have supported any significant political element. Therefore, the risk associated with political effects and intentions from the EU or other

political institutions cannot be credited as extensive based on currently available information. The legal effects based on passed legislation are, however, more substantial and are discussed in the Legal section.

Economic

The Covid-19 crisis has, without reasonable doubt, affected all economies on a global scale. When looking at the monetary, fiscal, and financial indicators concerning FaaS, no significant risk or impacting factor is present due to FaaS lack of association with capital market trends, interest rate policies, and fiscal policies. It should still be recalled that FaaS clients end users may face economic distress which would directly impact FaaS delivery and revenue opportunities. Looking at the current growth despite Covid-19, the trend suggests that such a liquidity risk is unlikely to occur.

Nevertheless, the indirect effects of resource allocation regarding human capital have led to an increase in demand for FaaS services. As the general workforce is scattered, serverless development provides an efficient solution to the increased demand for digital services from companies and end-users. When looking at the aforementioned trends for FaaS, the cyclical nature of the crisis can, to some extent, explain a current upside on demand. Nevertheless, growth in the FaaS field is still to remain post-Covid.

Social

As a high concentration of SMEs characterises the European market, and as the general market regardless of industry is moving towards increasing technology use in their operations, FaaS catalyses an increase in available opportunities for non-expert users. Irrespective of the level of liquidity or market share, the pursuit of opportunity is positively inferred by FaaS functionalities. Two major factors are the ease of deployment and pay-per-use philosophy.

In general, market demand suggests a shift toward serverless computing. Henceforth, increased demand for FaaS services by FaaS providers is inferred, and growing market capture opportunities are present. Because of this, the distinction between development and operations may not be as clear. Due to FaaS services' nature, automation, and abstraction lead to decreased demand for operations management over the development side in the processes. The rationale has changed in terms of application development.

Technological

This factor is, without reasonable doubt, the most significant and central part of the analysis. The Technology that FaaS offers on the market can effectively processes for users and successfully scale deployment processes.

FaaS technology is still relatively novel in the Cloud Computing market and is still emerging. Nevertheless, the critical feature of scalable deployment and flexible models is a significant pull factor that converts a wide range of stakeholders to adopt FaaS technology. Regardless, technological constraints need to be addressed in order for FaaS technology to scale as a technology. While the PHYSICS project aims to reduce vendor lock-in due to open-source technologies, the clouds and environments among FaaS-providers infer a high vendor lock-in risk. To mitigate this, investments into automation of application translations processes, deployment and abstraction using FaaS middleware are pivotal.

Touching based on security and the preference to utilise private cloud elements (usually in a hybrid manner), cost and efficiency effects have to be considered in developing the FaaS technologies as such constraints may lead to less scale of returns among desired target users and developers. By creating more sophisticated security features in the public cloud domain and reducing operational costs related to private clouds, the economic and social effects derived from this technological aspect may improve profitability indicators and increase production scale.

Legal

If looking at the legal effects and implications of imposed data regulations on EU-based FaaS-providers, compliance per se is not an element that extensively affects the daily operations. Providers of FaaS are generally showing adequate levels of adaptation to GDPR, but the increased regulatory requirements are inferring an increased risk concerning breaches of data, as well as an increased risk among third party APIs. For providers, the legal sanction of not complying with regulations is sometimes a larger risk itself than any obstacles related to GDPR implementation due to the drastic liquid and reputable effect of such data breach incidents. Consequently, risk mitigation is usually done through private cloud solutions dependent on traditional data centre staffing and maintenance. For some sectors where data is highly subjected to GDPR, such as patient information, the development of services has to ensure full security.

There are also sector-specific regulations such as the Payment Services Directive 2 (PSD2) or Markets in Financial Instruments Directive II (MiFID II) which, if looking at relevant aspects for FaaS, refers to increased consumer data protection measures in payments and more extensive reporting standards of trades respectively. These regulations are more commonly imposed and updated according to legal demand. Based on historical regulations, these regulations infers a higher risk for sectors such as the BFSI, which have been subject to both major non-sector specific regulations such as GDPR, as well as sector-specific regulations. Therefore, the need to design by compliance is pivotal, which is over-all successful among FaaS-providers due to the agility in the development of the FaaS environment.

Environmental

The current ongoing digital transformation supported by FaaS services leads to positive externalities as manufacturing, agriculture, and healthcare sectors operate more efficiently, automated, and digitally. As FaaS enables digital opportunities that may not have been previously feasible, it supports a tech-and-environmental integration.

Typical servers may have an unconscious state of 30% while their annual usage rates are dramatically low, sometimes as low as 5 to 15% per annum. The environmental footprint of software and servers can be substantially improved thanks to the intrinsic advantages of the FaaS models pay-per-use rationale. Furthermore, increased server consolidation combined with an optimised provider level scheduling may reduce power consumptions for data centres - currently making up 3% of worldwide total current power consumption which constitutes around 416 terawatts in generated electricity.

2.4 Business Model Synthesisation

Insight to the FaaS Business Model

A FaaS-provider business model is constructed to reduce the DevOps-demand of developers and cater a full-service experience to the developer in a serverless environment with high scalability and low costs. The business model is asset-light, and a rented service as users do not take ownership of the technologies but rather utilize it as consumers. Therefore, the providers can maintain stable revenue streams as customers are dependent on the providers, while users of the technology enjoy a more efficient and streamlined experience. In the FaaS business model, the main risk is placed on the FaaS provider as the following elements:[9]

- Data storage,
- Middleware,
- Containers,
- Operating System (OS),
- Virtualization and
- Hardware

are abstracted by the vendor whilst functions are the customer-managed unit of scale, and the application code is the customer-managed element. The latter leads to a broader range of customers given the comparative ease concerning technical knowledge and the non-existent infrastructural demand to develop applications.

[9] the list is not exhaustive

The PHYSICS project will provide three main tools known as CSP Cloud Design Environment, CSP Optimized Platform Level FaaS Services Toolkit and CSP Backend Optimization Toolkit. These tools will support providers in offering optimized services. The FaaS provider in PHYSICS will, in turn, have to provide tools such as Hardware or Containers.

The Main Common Industry Elements of the Business Models

Given the scope of the PHYSICS project and lack of public access to the business intelligence of individual FaaS providers, this part will address main characteristics observed among the previously defined and relevant market players rather than presenting the exact business model for each provider. The overview will not extensively account for production pricing of FaaS service nor operational costs due to the dedicated paragraphs later in chapter 2.4.

Element 1: All-inclusive Value Offer

The leading providers on the market seek to make life for the developers more focused on developing application code for the functions, and are therefore taking steps to make the process more seamless. Thus, elements that used to be integrated into DevOps are implemented in the FaaS environment to create an all-inclusive value offer.

Element 2: Pay-Per-Use Pricing Method

The pay-per-use rationale is one of the most vital elements of the general FaaS value offering. It is described more in detail in the “pricing and profitability” paragraph of this chapter. In short, it facilitates low-cost deployment, efficient production, and lack of ownership of assets.

Element 3: No application language barriers nor frameworks

The FaaS model allows a wide diversity of programming languages and does not require coding to be written according to specific frameworks due to the third-party services’ ability to handle the code. The deployment process differs from traditional options as the front-end code is uploaded on the FaaS service which then manages the backend processes. For instance, the provisioning and instantiating virtual machines. Developers enjoy flexibility and ease in their process of developing applications.

Element 4: No need for maintenance nor development of infrastructure

Developers may spend more time on design, development, and execution of functions related to the end-product rather than developing and maintaining an in-house set-up that does not generate value. The latter reduces time and increases efficiency, something fundamentally crucial for smaller developer teams without the means to maintain the infrastructure themselves adequately.

Increased Productivity

When looking at the properties of FaaS, the cost-of-service production is dynamic, and therefore better corresponding to the output levels of the service's consumption. While cost predictability is less clear due to the lack of fixed-expenses elements on the income statement, the dynamic pricing model ensures general efficiency in terms of financial expenditures. Furthermore, dynamic scaling reduces idle time and enables effects linked to economies of scale.

Operational Risk Assessment

While FaaS is poised for substantial growth, there are associated risks and uncertainty related to third-party APIs. Another non-liquid operational risk is interlinked with security concerns, which is especially pivotal among companies operating under EU-jurisdiction due to the high level of European data regulation. As risk is placed on the FaaS provider, all cloud components are under the provider's responsibility. Given that serverless functions utilise plenteous event sources such as HTTP Application Programming Interfaces and cloud storage assets, standardised web application firewalls may not effectively inspect the range of protocols and message structures.

In terms of the General Data Protection Regulation (GDPR), providers have constructed fully compliant data models. Nevertheless, developers still need to provide end-users with accessible and transparent ways to comply with the four foundational pillars of GDPR:

- The Right to Data Portability: users have the right to have a copy of their stored personal data.
- The Right to Be Forgotten: users have the right to have their personal data deleted.
- Privacy by Design: the security policies should be taken into account since the earliest stages of development.
- Notifications about Breaches: all breaches must be reported within 72 hours.

As serverless functions are yet the status quo of application development, visualisation and general monitoring opportunities are somewhat less user-friendly. Therefore, comprehensive logging is an essential component of troubleshooting.

Pricing and Profitability

Generally, the pricing models for FaaS services are harmonized, meaning that pricing is not a competitive-advantage differentiation strategy among FaaS providers. Production pricing consists of two main parts, the first coefficient is requested, and the second coefficient is GB-seconds. GB-seconds are the seconds a function runs multiplied by the amount of Random-Access Memory (RAM) consumed.
The price function is the following: Price = (Request Price Coefficient * Y) + (#GBs * Duration Price) where Y is every 1M request post any free tier offering as providers usually provide a free-tier option for certain amounts of requests and then charge for execution beyond the tier.

Providers such as AWS Lambda use a geographical open price discrimination strategy. The segmentation is not limited to APAC and EMEA regions, but rather segmented and priced on a country-to-country basis. For instance, in London (EMEA), the request is priced at $0.2 per 1M requests and $0.0000166667 for every GB-second. The same pricing model in Milan (EMEA) is $0.23 per 1M requests and $0.0000195172 for every GB-second. Providers such as AWS Lambda, Azure, IBM, and Google functions are typically more expensive than bare-metal FaaS providers like Heroku as the latter do not work with the same amount of abstraction such as including load-balancing, debugging and local development tools in their value offer. They seek to eliminate the demand for DevOps, hence making the value offer more all-inclusive.

As aforementioned and will be explained in the next part of this analysis, FaaS users face several positive financial effects by utilising FaaS technology. A direct consequence is the efficient pricing model, low cost of assets, and therefore lower need for investment and/or financial liability to finance assets as such. In a more forward-looking manner, indirect effects related to ease of deployment and general development facilities revenue opportunities with impact in which the additional revenue surpasses additional cost, thus making the P&L's of firms incurring opportunities for profit, not accounting for other revenues or expenses. In other terms, the pricing model leads to increased value for money and requires no or little initial cash contribution. Businesses such as those in agriculture may improve profitability by making more accurate assessments using data-driven and auto-scalable solutions, mitigating inventory impairment and non-realised revenue transactions.

Looking at non-asset, human capital resources, the serverless computing paradigm allows developers to focus on producing code without considering the provisioning, configuration, management, and manual scaling of any back-end infrastructure. Henceforth, the realisation of deployment and execution is faster, and the time spent on the processes decreases as the complexity threshold decreases when creating scalable applications. When assessing profitability, effects are expected throughout the entire spectrum, meaning that the application's data size and complexity are not deviating modalities. Nevertheless, momentum using FaaS can be especially important for the profitability indicators for Minimum Viable Products (MVPs) and smaller applications as the development costs are in relation to the value they might create.

With respect to the predictability of expenditures which leads to a more accurate profitability assessment, the FaaS model both leads to cost reductions and an improved opportunity to oversee fees due to the variable cost based on usage. As operations expenditures are proportional to use, the general business model of a product built on application code is more comfortable assessing profitability. According to Tim Wagner, former GM of AWS Lambda, FaaS may yield a 4:1 to 10:1 cost compression ratio for a typical workload.

Cost Structure Analysis

One can look at the cost structure in FaaS from two different perspectives, those being from the developer and provider perspectives. FaaS technology infers a more tailored and efficient spending opportunity while providers face some risk in facilitating the serverless infrastructure model's operations.

For FaaS developers (a consumer in business terms), the main factor for implementation is the pay-per-use rationale. Simply put, there is no need to pay for infrastructural assets that may not be optimized fully, but instead, maximize the use of FaaS providers and their services. But how do FaaS users minimize their expenses in their income statement by this asset-light production model? The main explanation lies in the fact that FaaS providers facilitate more efficient data transportation. To understand how cost is incurred in FaaS, the main cost is found in charges outside of the price per function invocation which benchmarks at approximately $0.0000002.

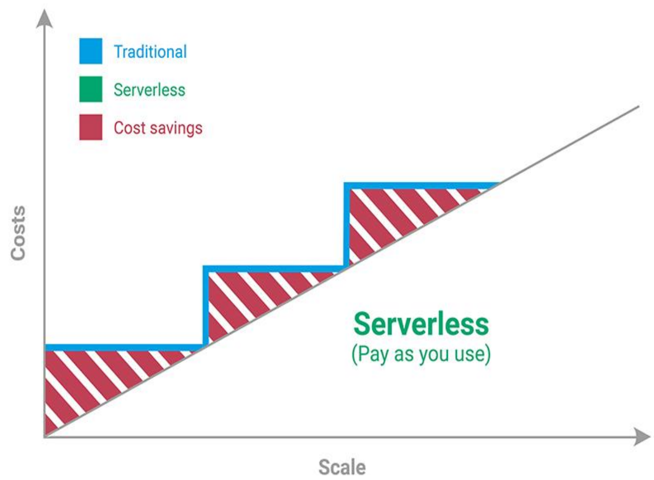

Figure 6 - Linear cost-reduction representation of FaaS v. Traditional Computing (Edelhoff, 2019)

Frankly, operational costs related to the execution of functions make up the largest incurred cost ratio due to computational resources usage. When looking at the two major providers AWS Lambda and Azure, they both charge around $0.00001667 per Gigabyte-second (seconds a function run multiplied by the amount of Random-Access Memory consumed). Depending on the function execution demand level, the total cost will vary with respect to the allocated memory configuration, which is usually subject to the composition in the range of 128 MB to 1.5 GB. Regardless, FaaS provides a low-cost solution in relation to an asset-based function execution model as costs are usually lower than for IaaS-models.

Nevertheless, a key consideration for any consumer is the feasibility of each technology based on their demand. Serverless computing services may not be the most efficient technology for all types of computing. Generally, the following characteristics of computing operations are considered as cost-efficient:

- Ease of division to smaller independent units;
- Non-linear data traffic patterns;
- Short start-up time frames;
- Demand of either higher developer velocity or short time to each market.

While workloads carrying the following elements may not have FaaS as the most efficient technology:

- Tasks in demand for incense computation power that are long-run;
- Workloads in demand of direct execution with no risk for delays.

From this, it can be concluded that FaaS is more cost-effective for execution that is not dependent on any traffic delays. As pricing is solely based on execution time instead of process idle time, higher scalability can be achieved at a low cost at the expense of increased latency. Finally, the operational risk of running non-suitable invocations may lead to overhead and retries in the FaaS environment. As the invocation of functions fails, retries will occur until the event is retained. Automated monitoring efforts to quickly spot scaling costs are therefore recommended to leave room for cost mitigation.

3. PROVIDERS, USERS AND STAKEHOLDER ANALYSIS

3.1 Main Cloud Providers' Structural Sources and Executions

Among the companies offering the most popular services in the market, AWS Lambda is the one with the largest market share (Synergy Research Group, 2020).

This section will perform a market analysis of the main providers focusing primarily on the structural sources and execution capabilities at the core of the services provided by AWS Lambda, Google Cloud Functions and Microsoft Azure Functions. In other words, it will be investigated the effort of the companies to improve their frameworks in one or more characteristics.

Afterwards, it will be considered that a FaaS platform does not necessarily run on a serverless environment, such as AWS Lambda. Still, many FaaS implementations such as OpenFaaS, Fission and OpenWhisk, allow us to deploy and run FaaS on users' hardware. In this reality of open-source frameworks, most of them use licenses such as MIT and Apache 2.0, which allow users to use the software for any use with minimal limitations.

OpenFaaS, for example, uses an MIT license, which allows users to use code for any purposes, even if the code is part of proprietary software, with the only limitation of including the original copy of the MIT license.

While this chapter will focus on a market analysis of the providers, section 4 will perform a more technical analysis of the open sources providers' infrastructures

3.1.1 Amazon AWS Lambda

According to New Relic's Serverless Technology Semi-annual Report, the rise in serverless practice adoption among businesses is exponential as the average weekly invocations were increasing by 206% in 2019. Once again, Amazon Web Services played a vital role in this segment when it introduced Lambda, an extension from its existing AWS product range in 2014. Even though Lambda was not the first mover on the FaaS market nor the very first serverless compute service, it still was a successful entrant that soon became the go-to serverless cloud compute service provider and a model to replicate for other prominent players in the cloud computing market.

Simply put, AWS Lambda is a serverless computing service that facilitates the customer to run code without provisioning or managing servers, to create workload-aware cluster scaling logic, maintaining event integrations, or managing runtimes. With Lambda, developers may run code for virtually any type of application or backend service. Customers upload their code as a ZIP file or container image, and Lambda automatically and precisely allocates execution power and runs the code based on the incoming request or event, for any scale of traffic. You can set up your code to automatically trigger from 140 AWS services or call it directly from any web or mobile app. Furthermore, customers can write Lambda functions in all types of languages (Node.js, Python, Go, Java, and more) and use both serverless and container tools, such as AWS SAM or Docker CLI, to build, test and deploy your functions. Lambda thus makes it possible to run the code without having to take care of deploying the servers anymore.

Lambda is Amazon Web Service's main feature when it comes to serverless. Lambda customers use a unit of code for a function or a task to achieve specific results. The customer leases this piece of code for a certain amount of time until the required tasks are carried out. AWS then charges for the memory used to carry out the function, and for the time, this function or service is active. Simple and highly effective.

The possibility to integrate Lambda with a multitude of services made available and hosted by Amazon Web Services (CloudWatch, API Gateway, S3, Dynamo DB, Alexa, Kinesis) explains why AWS Lambda still has to date a competitive edge against the likes of Google Cloud Run and Azure Functions.

Furthermore, it works more efficiently with code. AWS Lambda provides users with the infrastructure to upload their code. It takes care of maintaining the code and triggers the code whenever the required event happens. It allows choosing the memory and the timeout required for the code.

On the one hand, AWS Lambda supports unlimited functions per project and will allow 1,000 executions per account for each region. On the other hand, Google Cloud Functions provides a cap of 1,000 tasks per project, with a maximum of 400 performances. However, concerning execution time, AWS Lambda will timeout after five minutes, while Google Cloud Functions may execute for up to nine minutes, which may be more critical for long-running processes, such as video composition and transcoding.

Soon after Amazon Web Services introduced Lambda, its main competitors also started developing and commercialising their own serverless FaaS frameworks without experiencing Lambda's same success. Their offering was simply catching up to what AWS had previously delivered and achieved through Lambda without necessarily bringing additional added value or features. Given the difference in their timeline, AWS Lambda holds an advantage over the other platforms as it provides scalability and fully automated administration with concurrency controls and event source mapping.

3.1.2 Microsoft Azure

When a company is looking for the best cloud infrastructure provider, it should look at the leading player in such a market (namely AWS) or try to take inspiration from its competitors in the choice they made for such a service. Thus, even though Amazon Web Services is the leading cloud provider worldwide with ⅓ of the cloud infrastructure market share (33%, way ahead of its competitors, Synergy Research Group, 2020), a company should look for the provider that best fits its needs. Part of the main criteria that constitute the specifications when making a call for bids include requirements for performance, availability, security, storage, and workload, to name a few.

Microsoft launched Azure in January 2010. A decade later, Azure is AWS's strongest competitor with an 18% market share (Synergy Research Group, 2020) as it is closing the gap against Amazon's cloud infrastructure solution. Azure thus presents a lot of critical strengths which make it a fierce challenger to AWS in this market. Therefore, it is increasingly difficult for companies to choose the better service between the two GAFAM Giants and other providers such as Alibaba and IBM. However, there are still some differences in the quality of the offerings and in the technical characteristics that can be spotted.

Microsoft Azure leverages Microsoft's existing customer base using Office 365 to fuel its rapid and constant growth over the last few years in the cloud infrastructure market. While rapidly growing and gaining market share in the cloud market, the outlook also looks promising for Azure. It has made several exciting moves in recent years that might give it a competitive edge. For instance, Microsoft's cloud solution prevailed to win a 10-billion-dollar deal from the Pentagon for its cloud computing services. Azure also agreed on a substantial contract with the NBA and Blackrock, the world's most prominent asset management firm for its well-known Aladdin platform. The long-term agreement sealed with AT&T is another significant move from Azure which can be mentioned as Microsoft is moving fast in the cloud universe.

These milestones emphasise how Microsoft can benefit from its existing products such as Windows and other Microsoft tools and software to build on these existing partnerships that it can leverage for its Azure service. Microsoft Azure, therefore, stands as the more straightforward go-to solution for businesses' executives to have an all-in-one place package through one provider, through a combination of Microsoft services (Azure, Office 365, Teams, and many others) to run all their applications in one single cloud, not to mention other players it is associated with (Salesforce, Adobe, SAP, Oracle). Furthermore, despite its late mover advantage in this market, Microsoft also leverages the multiple on-premises software it has developed throughout its existence and repurposed it for Azure, making it a severe cloud computing provider.

Azure benefited from Microsoft Software-as-a-Service (SaaS) footprint it has earned throughout the years on an external site. Also, it helped Azure internally as, from an execution and technology perspective, the learning curve from Windows has been a critical success factor according to Microsoft Azure EVP Jason Zander.

In terms of scalability, performance, reliability and security, Microsoft Azure is known as one of the best solution providers even though it is not a differentiating factor against Amazon Web Services or Google's GCP for instance. As mentioned by Jason Zander, Microsoft Executive Vice President, it is also a central focus. Additionally, Microsoft Azure possesses more robust hybrid options than AWS, while also offering more specialised storage options (e.g., Data Lake) (Varonis, 2020). However, on the other hand, Azure is weaker than AWS in terms of the depth of its offerings, as AWS offers a broader range of services to its customers.

Moreover, even though Microsoft has an extensive experience in serving corporate clients with its various offerings, Azure still reportedly underperforms in technical support, training and breadth of the ISV partner ecosystem, and documentation. Finally, it can improve its cloud offering as the learning curve can be qualified as steep, making Azure more complicated than its direct competitors and harder to use and manage.

With its recent acquisitions of 5G specialists Metaswitch Networks and Affirmed Networks, Microsoft makes a critical move to improve its 5G cloud offering, thus strengthening its Azure solution's capabilities. This latter will grow at scale through enhanced capabilities via a more secure, broad and efficient ecosystem (Khalidi, 2020).

Microsoft will be looking in the future to leverage AI capabilities for Azure by integrating the Brainwave Project (Deep Learning system). This area is still catching up with Google, which already made critical investments in AI and machine learning. Out of the main tech giants, Google has indeed invested the most in AI, as it has invested around $3.9 billion since 2016, thus far ahead of Amazon and Microsoft, respectively second and fifth most prominent investors in Artificial Intelligence to date (according to a research conducted by RS Components, 2018).

3.1.3 Google Cloud Services

In September 2017, Google strategically acquired Apigee Corp for $625 million (Clement, 2018). Apigee is a provider of application programming interface (API) management. Various companies already use their services, such as Burberry, Walgreens, Live Nation, etc.

The acquisition of the API oriented tech business was a crucial development strategy for Google, as it gave it a competitive advantage. According to the research, 84% of the tech industry experts state that API implementation is critical or somewhat critical for their business strategy and further growth (Marklein, 2019). Research shows that US companies alone have spent nearly $3 billion on API management (Greene, 2016).

The current industry leaders are implementing APIs in various ways to develop their business, with 55% of them using API as a revenue stream. The examples of companies with API-based business models, including Google with Google Maps, Analytics, Calendar, Contacts, or Facebook, Spotify or PayPal, are mainly API-based, connecting their services to millions of third-party websites and apps. Also, Japanese Sony has an API for developing applications integrated into their devices (Vector ITC, 2019).

The main benefits of introducing API to Google's offer was to improve its cloud offering focused mostly on corporate clients (Trefis Team, 2016) by providing the following advantages to their service. The main benefit of adding API Apigee to Google Cloud is accelerating moving the customers into high-quality digital interactions. The service will allow faster and easier APIs implementation and publishing with excellence (Greene, 2016). That means that the customers will be able to, for example, enable their developers to work on the code of their application while maintaining the stable interface in the apps and services. Google has chosen this particular company because this API fulfils most of the requirements: supporting security and allowing the developers to select the development environment they want to work in. It includes testing supports and usage analytics.

A few years ago, such an investment into the API management systems was indeed a bold move as API is now at the core of everything digital. This market is currently valued at $1.97 billion by revenue. Furthermore, as shown in the graph above, it is predicted to reach $6.1 billion in 2024.

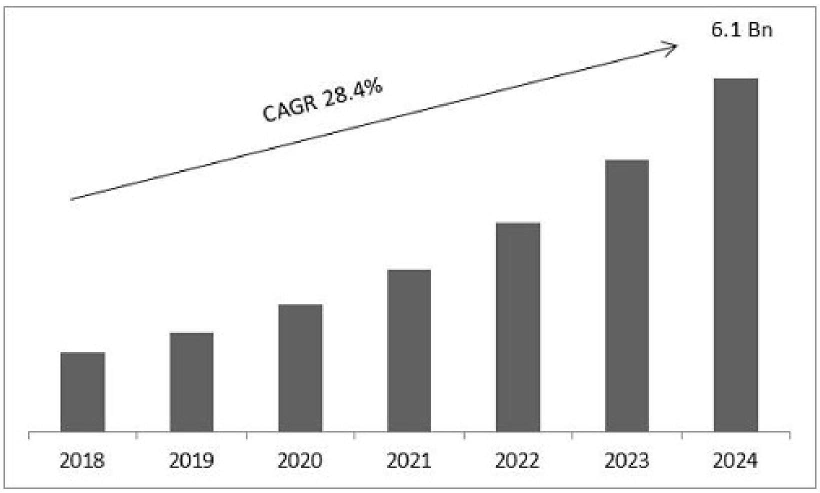

Figure 7 - Global API Management Market Size *Source: KBV Research, 2019*

The main benefits of implementing API in Google are that it drives efficiency and accelerates time to market and, as visible above, it has substantial growth potential. The efficiency comes from how the apps are built, without API's they were made monolithically, but smaller teams can work on different parts of the app at their speed. Therefore, the customers of Google Cloud are launching their projects quicker and more efficiently. Also, API allows Google's partners to unlock new business models and revenue opportunities. For example, AccuWeather and Pitney Bowes have pursued API monetisation strategy with Google by selling the data and functionalities captured by API and selling them to third parties.

Another aspect is that Google's API allows its customers to activate data and inject intelligence into business processes. It lets the enterprises connect their digital assets to the APIs that provide machine learning services running in the cloud to develop their business utilities. Also, the API allows combining the code to be reused and incorporated for different cases, making it modular and composable in the IT systems, creating resiliency (Hood & Kasiviswanathan, 2020).

3.1.4 Open-Source Frameworks

This deliverable aims to cover the characteristics of open-source frameworks, which benefit from a large community of developers that increase these services' spread.

Among all of them, OpenFaaS is one of the most popular ones, with more than 19k GitHub stars (a mark of appreciation of users) way more than other popular frameworks such as Kubeless [10](6.4k), OpenWhisk (5.2k), Fission (5.9k) and Knative (2.7k) (GitHub, 2021).

In the following development four of the main open-source FaaS framework will be analysed in their most important aspects, mainly but not exclusively technical, and in their characteristic competitive advantages, in particular:

- OpenFaaS
- Fission
- OpenWhisk
- Knative

OpenFaaS

OpenFaaS allows users to utilise OCI and Docker image format (meaning a system that can make an application work as if it is separate from the rest of the OS, creating the illusion that the application is getting its OS instance) and convert any process into a serverless function.

If compared to AWS Lambda, OpenFaaS is developed with a greater extent of customizability: while Lambda supports only Node.js (JavaScript), Python, Java, Ruby, PowerShell, C# (.NET Core) and Go, OpenFaaS can run codes in any language for Linux and Windows just by specifying them in the Working Environment, the place where functions are built and run. It is not limited to the 300 seconds for longer-running processes, as the cloud timeout can be overridden. Finally, it can be run in a system with Kubernetes or Docker, even on-premise, hybrid cloud, private cloud or multi-cloud (OpenFaaS introduction, 2021).

OpenFaaS overcomes the problem of continuous integration (CI) of many large organisations which manually organise every microservices and, for every single one, define the deployment pipeline. OpenFaaS structures all the functions under a single CI system, and using build-in metrics developers can see where the demand is.

Another OpenFaaS characteristic comes when working with pre-trained machine learning models or heavy Python libraries: there is no limit in the functions' size. For instance, this feature is paramount for Anisha Keshavan, a neuroinformatic researcher at the University of Washington that uses OpenFaaS to support her brain scans analyses. Only during the last few years, other frameworks improved on this dimension. AWS Lambda memory allocation has reached a maximum of 10240 MB, while Google Functions can configure a maximum of 8 GB.

According to Alex Ellis's opinion, the creator of the framework, OpenFaaS has been designed primarily for developers. Among the main advantages, a Command Line Interface that has received attention from the community, the User Interface connected to a function store to easily implement popular functions in the code.

Secondly, it aimed to support operation departments. Indeed, it allows running OpenFaaS with one command and without waiting for more than one minute.

Thirdly, OpenFaaS is community centric. Based on the assumption that top-down adoption is not efficient, the framework relies on the spread of OpenFaaS through developers' appreciation, who make IT decisions in their organisations and influence people around them (Contino, 2018).

[10] https://kubeless.io/

Fission

Like OpenFaaS, Fission is an open-source project that launched its alpha stage in 2017, focusing on developer productivity and high performance. As it is built on a Kubernetes cluster and works under the Apache License, it can function anywhere: in a private data centre, a public cloud, or a personal laptop.

Fission especially shines with its low cold-start latencies. While AWS Lambda takes on average around 3-5 seconds to cold start (sc. Lohika, 2021), Fission manages to start running after around 100 milliseconds. The latter is due to the maintenance of so-called “warm” containers that each hold a dynamic loader. When a function is called for the first time, it is chosen from an already running container, which results in lower latencies.

In terms of supported languages, Fission supports Python, NodeJS, Go, C# and PHP as of February 2021, but custom containers can be built at any time to add additional languages.

One of the most significant advantages of Fission is the integration of live-reload. Fission automatically deploys the written code into a live Kubernetes test cluster, allowing developers to run testing cycles more quickly. Bugs are therefore detected more rapidly, and early bug fixing can prevent critical damage in higher environments.

Furthermore, Fission is the first serverless project that incorporates an out-of-the-box Record-replay function to simplify testing and troubleshooting. This functionality will save events that engage serverless functions and replay these events with exact information about the application use and the functions' in- and outputs. The latter makes it reasonably easier for developers to reproduce complex failures and troubleshoot issues.

Fission provides fully automated and easy configurable Canary Deployments to prevent more significant damage as the first open-source serverless framework. DevOps and Site Reliability Engineers can now more efficiently and safely deploy new versions of applications, as it is first deployed to only a small percentage of the total traffic, before being gradually rolled out to the entire framework. If a failure sets in, the application can be easily rolled back to a previous, more stable version.

Lastly, Fission offers businesses an Azure-like serverless service on an infrastructure of their choice. They can choose to use cheaper public clouds or even running their own data centres through this option. Fission’s cost and performance optimization controls allow for specific CPU and Memory resource usage limits, minimum and maximum number of instances and auto-scaling parameters. The company can decide by itself, which are the most critical functions in order to allocate resources according to their business plan.

OpenWhisk

OpenWhisk was introduced as an open-source project by Adobe and IBM in 2016 and donated to the Apache Foundation to encourage community involvement. It is a robust and scalable platform that supports thousands of concurrent triggers and invocations.

Being open source, it can be deployed on any infrastructure but works exceptionally well on the IBM Bluemix platform. OpenWhisk stands out with its extensive language compatibility as it already supports NodeJS, Go, Java, Scala, PHP, Python, Ruby, Swift, Ballerina, .NET, and Rust. Additional languages can be added by executing Zip Actions which run on the Docker runtime.

Another great benefit of OpenWhisk is the integrated ability to create Sequences of Actions. This built-in feature allows for chains of functions in which the previous one's output will be the output of the next one. In addition to this, OpenWhisk is built with a library of Actions to automate 3rd party integrations such as GitHub, Slack, Weather, Watson, Cloudant, WebSockets and Mobile Push. Using this shared library of utility functions makes connectivity and cross-integration more straightforward.

Lastly, OpenWhisk includes native HTTP integration which allows developers not to spend time exposing actions via HTTP. HTTP endpoints are automatically created for each action and can be invoked via POST requests and Basic Authentication headers.

In terms of OpenWhisk's technical limitations, it is similar to OpenFaaS and Fission. The average latency is between 0.3 (median) and 0.54 (95%) seconds, and its maximum memory is 256 MB per action. The maximum size of the code per action is limited to 46 MB, and the default maximum runtime for a container is 60 seconds, which is comparably short compared to the other services. This timeout limit can be changed when creating the action to a maximum technical runtime of 300 seconds.

Knative

FaaS offerings were typically based on small pieces of source code that ran as functions - one small file and one command and you have a function. There is no need to code any HTTP handlers/routes, the function has one entry and exit point and is all built into the platform. Knative disrupted all that by making any service available as a function, by allowing a service to scale to zero after a configured period. In other words, the service stops running, which means no CPU cycles, no disk activity, and no billing activity during its idle time.

Knative is a Kubernetes-based platform to deploy and manage modern serverless workloads, originally by Google with over 50 different companies' contributions. It is an open-source platform that is just starting to gain popularity but is of great interest to developers today. Indeed, it has been adopted by the Cloud Native Computing Foundation, which means no vendor lock-in, which is a considerable limitation of the current cloud-based FaaS solutions. Knative can run on any Kubernetes cluster.

Knative builds on top of Kubernetes, abstracting away the complex details and enabling developers to focus on what matters. It allows you to develop and deploy container-based server applications that can be easily ported between cloud providers. Whether on-premises, in the cloud, or a third-party data centre, it follows the best practices shared by successful real-world implementations, solving the "boring but difficult" parts of deploying and managing cloud-native services. Additionally, it focuses on an idiomatic developer experience, supporting common development patterns such as GitOps.

Therefore, while the operators can focus on managing the Kubernetes cluster and installing and maintaining Knative instances, the developers can focus on building and deploying applications using the Knative API interface. That gives any organization immense power, as now different teams can focus on their area of expertise without stepping on each other's shoes.

3.2 Users Identification and Exploration

3.2.1 BFSI

As seen in section 2.1.1, in the BFSI sector, FaaS technology is already widely adopted, and some examples of FaaS usage will now be illustrated.

Financial Engines is an end-user of the FaaS product developed by Amazon: AWS Lambda. It is the largest independent investment advisor in the United States in terms of asset under management, providing technology-enabled portfolio-management services, financial planning, and investment advice. The main reason which urged the company to adopt FaaS technology is the improvement of the scalability of their core engine component (the Integer Programming Optimizer) performing calculations used to evaluate and optimize investment portfolios as there was a continuous increase in demand for their services. They were also able to boost their system's resilience.

They opt for AWS Lambda specifically rather than Amazon Elastic Compute Cloud because the former can precisely scale with the workflow. Hence, "With AWS Lambda, we don't need to worry about spikes or growing or shrinking the farm." says the Chief Architect at Financial Engines, Allen Aubuchon.

Regarding the insurance industry, both big companies like AXA group and small start-ups like Branch Insurance started using AWS Lambda services.

On the one hand, AXA group, a French insurance company, partnered with Uber to develop a specific vehicle interruption insurance for Uber drivers. They describe this new service as "the next level of insurances". It includes a pay-as-you-go concept: in the case of a car crash with an Uber driver, they are insured through Uber for their car, but there is an income problem as the driver does not have a car anymore. Therefore, AXA developed an "on-demand" concept with this insurance, directly integrated into the Uber App that the driver can switch on and off. This specific insurance covers drivers at the exact moment when they are driving. The insurance will also be calculated depending on Uber driver's characteristics, hence getting data directly from the Uber App. Even though Uber internally developed its FaaS service to have more introspection inside the system, AXA and Uber used AWS to develop this new kind of insurance for Uber drivers.

On the other hand, Branch Insurance is an insurance start-up selling home and auto insurance online. According to its co-founder, it would be relatively easy to implement several automated and managed services in the insurance industry as it has a sort of simplicity which makes it easier. Hence, Branch Insurance decided to use AWS, specifically AWS Lambda, as the start-up aims to have a coding environment that is understandable for all their developers, thus being modifiable by any of them. "The biggest benefit of serverless is that it gives us true infrastructure as code in a way that all of our developers can understand and maintain pretty easily." Says Joseph Emison, the co-founder and CTO of Branch Insurance.

Interviewing with a Partner for Finance Strategy at Deloitte, the Consulting Sector's perspective on the current stage of Cloud Computing and especially FaaS arose.

According to the interviewee, the Consulting Sector is heavily familiar with the technology, and Deloitte has entire cloud teams to offer clients the perfect solutions. In terms of current limitations for the widespread use of FaaS, he sees especially security and privacy issues still to be tackled, which is why many traditional sectors such as banking and the public sector have yet to open up to these new possibilities. On the contrary, manufacturing, automotive, utilities and telecommunications sectors are pioneers in creating use cases for serverless technology. The main issues for every company are the lack of professionals (so-called "purple people") with serverless expertise, which is why companies like Deloitte have been expanding their portfolio, offering End-to-End Consulting services.

3.2.2 Telecommunications

T-Mobile (Wireless Services Provider). As T-Mobile experimented with Amazon Web Services (AWS) serverless technologies for its APIs, microservices, and time- and event-based processing, it quickly uncovered new value from optimized resources, simpler scaling, reduced patching and time savings—all of which enabled an even stronger focus on exploration and innovation and resulted in increased agility in responding to customer demands. As a result, the company now has a "serverless first" policy for developing new services, with serverless technologies like AWS Lambda playing a pivotal role in such mission-critical T-Mobile applications as the company website and the T-Mobile My Account smartphone app. However, to get there, T-Mobile needed a way to enable its development teams to adopt and use serverless computing while also complying with company-wide security, operational, deployment, and cost requirements.

AWS Lambda is used to build "Jazz" an open-source platform for building, deploying, monitoring, and debugging cloud-native APIs, functions, and static websites with serverless architectures. With AWS Lambda, developers could focus only on writing codes. The latter led to an increase in the developers' room for creativity and thus, satisfaction. As a result, employees can better focus on creating value for the company and customer without worrying about infrastructure management and maintenance. Satish Malireddi, Principal Cloud Architect at T-Mobile, further mentions that “serverless computing on AWS has cut out many time-consuming development and deployment steps, where the saved time can be spent on innovating and iterating on the solutions customers want.” Moreover, the pay-per-use model used led to increased cost-efficiency in the firm.

3.2.3 Consumer Goods

The Coca-Cola Company initially used the AWS Lambda function in order to develop the Coke.com Vending Pass Programme. It consists of drink rewards when purchasing at vending machines equipped to support mobile payments. In this program, AWS Lambda is used to initiate a pair of calls to some existing backend code to count the vending points and update the participant’s record.

After this project’s success, Coca-Cola kept using AWS Lambda for other projects such as building serverless solutions to publish nutrition information to their food service partners using Lambda and developing a prototype application to analyse stream of photos on social networks and extract trends in tastes and flavours. There are two main benefits of using AWS for projects at Coca-Cola. Firstly, developers no longer need to wait for servers to be provisioned and hence, they have more time for creation and innovation processes. Secondly, it enables Coca-Cola to improve the scalability, functionality, and reliability of their applications.

3.2.4 Healthcare

A successful example of FaaS services usage in the Healthcare industry is Benchling, a Californian science software company currently using AWS Lambda services.

They developed a software platform providing researchers with diverse services to perform their experiments. One essential benefit is the CRISPR technique's support: a breakthrough technique that worldwide researchers use to modify parts of a genome with extreme precision, build disease models, and screen for drug targets. Benchling’s goal was to split up a CRISPR search across several AWS Lambda tasks to reduce costs and boost scalability in order to support more users of the platform.

The use of AWS Lambda inside the process is as follows: a Benchling web server receives a researcher’s request to conduct a CRISPR search on a specific genome. The web server then splits up the genome into smaller tasks and invokes the AWS Lambda function for each task. Lambda then downloads the genome data stored in Amazon S3 (another AWS service), performs the query, combines the results, and returns them to the researcher. Using Amazon Lambda, Benchling does not need to maintain several servers to perform searches anymore.

“By making this analysis faster, scientists using our platform can spend more time focusing on science. That’s what we want to do—enable faster searches so their research isn’t hindered by performance limitations. The industry standard for these searches is minutes to hours, and we’ve been able to improve that tremendously.” says Vineet Gopal, Engineering Manager at Benchling.

3.2.5 Government

Some public institutions also use cloud services for diverse purposes. For instance, the Home Office department in the United Kingdom renewed its deal with Amazon Web Services in January 2020 for four years. The organisation handling the application process for British Universities: UCAS also uses AWS. It is the only service to apply for higher education. Therefore, the website has to handle massive surges of users logging on the website when the results are released. Hence, using AWS allows for flexible capacity: UCAS can scale its servers to accommodate demand and then scale back down to normal loads while paying only for its needs. It is a crucial cost-decision for the organisation, as the release of results only happens once a year, hence needing excessive capacity only once a year.

3.2.6 Media & Entertainment

Within the Media & Entertainment sector, FaaS is widely used in the Mobile Gaming industry. There, it helps Game developers like Square Enix to accelerate in-game operations and image processing.

Dragon Quest X which is one of their flagship video-game titles is connected to a portal site and a smartphone app that provides players with the ability to exchange messages, make purchases, and manage characters in the game. It also features a function for taking in-game screenshots, which customers can view and control from the portal site. This function has proven extremely popular. Customers take a massive number of screenshots, especially during events held in the game that coincide with occasions such as Christmas or New Year's Eve.

This image processing is quite resource-intensive, especially if there is a spike in server load when many people take screenshots at the same time, such as during events. Usually, 200 to 300 images are received for processing each minute, but during New Year's Eve in-game events, this can climb as high as 6,000 images per minute. “Accelerating this process and boosting customer satisfaction was one of the challenges Square Enix faced in growing its customer base”, mentions Technical Director Daisuke Agata. He further explains that “AWS Lambda had a striking effect. Image processing that used to take several hours was finished in a little over 10 seconds. We were also able to reduce costs to about one-twentieth of those for performing the same processing on-premises."

3.2.7 Manufacturing

In the Manufacturing sector, the automotive industry seems to use FaaS technology widely. Indeed, several examples of large automotive players such as Volkswagen, BMW, and Toyota are currently FaaS services users. Renault and BMW are two of the most active players in the sector as they respectively launched a partnership with Google Cloud from July 2020 and Amazon Web Services from December 2020. BMW initially partnered with AWS to better predict customer demand for its production, switching from gut-driven to data-driven decision making. It also aims at centralizing data from the 3 BMW brands: BMW, Mini, and Rolls-Royce.

From a manufacturing point of view, it enables to optimize car production and logistics and better predict maintenance. Using AWS cloud computing platform will increase efficiency, performance, and sustainability across its vehicles' lifecycle, from design to after-sales services.

In 2018, BMW also released a new service called Unified Configurator Platform on AWS. It allows customers to design the perfect BMW car with multiple combinations possible. The platform consolidates the vehicle products as services. In this specific case, AWS Lambda is used to import data from primary data sources continuously. From the Amazon CloudWatch Events data, AWS Lambda creates up to hundreds of jobs to import the data in the AWS batch.

3.2.8 Agriculture

To gain insights, an interview with Dominik Heinen, Manager in the Digitalisation department at Raiffeisen Waren-Zentrale Rhein-Main eG (RWZ), Germany’s third-largest agribusiness, has been conducted.

According to the interviewee, cloud applications usage is continuously growing in the sector, specifically in large agribusinesses. Nonetheless, small agricultural producers and farms have yet to follow the steps implemented by companies like RWZ.

As the reason for the transformation reluctance of most farmers, the interview mentioned that it is relatively hard to convince the current generation to implement radical changes. Indeed, traditionally, this sector has always been conservative. The main issues they have with cloud-computing are data protection-related questions and general distrust towards new technologies that require know-how outside of their expertise.

However, the potential created for farmers and agribusinesses by cloud computing is enormous. First of all, cloud computing would heavily reduce farmers' costs due to the pay-per-use models provided and further would create more flexibility. The latter is particularly the case for farmers using central cloud-based platforms and applications. Many inefficiencies are created due to incompatibilities between farmers, equipment providers, and distributors.

More specifically, FaaS is not widely used in agriculture yet, but there are possible applications already. For example, code concerning weather predictions with their direct implications on certain agricultural products (e.g., fungus), could be developed and maintained by the farmer or agribusiness. The infrastructure would be kept at the provider, implying that the individual agriculture entity holds the intellectual property and can thus profit from the FaaS structure while scaling and selling it to other businesses.

3.3 Users Identification and Exploration

3.3.1 EU

Having the purpose of shaping the European digital future, the EU aims to strengthen Europe's position in data-driven innovation, improve competitiveness and cohesion, and help create a Digital Single Market in Europe. Aligned with this goal, the European Cloud Initiative was created to provide European science, industry, and public authorities with a world-class data infrastructure to store and manage data, high-speed connectivity to transport data, and an extraordinarily powerful High-Performance Computers to process data (European Commission, 2020).

3.3.2 Cloud Users

Identified also as the final customers of the cloud service, their behaviour is most affected by breaches or lack of performances in the cloud environment. According to the security mechanisms and the recommendations provided by ITU-T Recommendation X.1601 and followed by the Institute of Electrical and Electronics Engineers (IEEE) the significant needs to be addressed include the protection of the access and, especially, data. Only validated and authorised entities must have granted access to the platform and the services, and confidential information must be protected both from errors and attacks by third entities. Additionally, control of the data and availability at all time and place are additional features of a sensitive matter and, even though rarely applied, the possibility to migrate easily from one vendor to another without facing vendor lock-in constraints. Finally, problems may arise from performance limitations, due to multi-tenancy and time restrictions of functions (even though it varies on the FaaS framework adopted) (SpringerLink, 2020).

3.3.3 Cloud Provider

As the entity that owns the responsibility of delivering cloud services to the Cloud User and handling a vast amount of data from the users, their service's efficiency relies on the trust that performances and security are granted from Cloud Users. Therefore, proper instruments and technologies must be designed to support the service's managing and protection and sensitive information. Following the ITU-T Recommendation X.1601, significant threats and challenges include the danger of hacking attacks to internal servers and secure access granted only to the administrators. Regarding granting performances, the service's reliability must be protected from types of attacks such as DoS and DDoS, which can compromise the FaaS framework's continuity (SpringerLink, 2020).

3.3.4 Cloud Auditor

In order to evaluate the performance, privacy and control of the services of Cloud Providers, the stakeholder has roles of performance audit, privacy audit and security audit. The critical aspects that it observes are the transparency of the management of data and security. The function includes auditing the SLA (Service-Legal Agreement) documents prepared by Cloud Providers that ensure a minimum level of protection to the cloud customers. Changes and scaling of technologies are certified by the Cloud Auditor not to affect the security and the services granted to the Cloud Users. Finally, the Cloud Auditor confirms if the technology used to encrypt customers' information gives a sufficient protection level (SpringerLink, 2020).

3.3.5 Cloud Broker

Cloud Brokers are a category working between Cloud Customers and Cloud Service Providers, offering additional services to Cloud Customers such as interfaces with various integrated services and aggregation, arbitrage and intermediation. Working with different types of Cloud Customers, sometimes on the same platform requires assurance from data leakage and privacy. Finally, a proper channel with Cloud Service Providers guarantees correct and efficient customer data flow. It verifies the role, position, and permission of Cloud Brokers in the chain of data processing and access from the cloud (SpringerLink, 2020).

3.3.6 Cloud Carrier

The entities that help provide the cloud service through the internet network, telecommunication and other devices belong to this category. Their function is to create and manage the connection between Cloud Service Providers and Cloud Users, often using encryption technologies to secure data during its transportation. To provide the service, the SLA, mentioned above under the Cloud Auditor description, must be prepared between the Cloud Service Provider and the Cloud Carrier before delivering the service to protect from data-stealing, leaking, and everything else that can represent a significant loss to the Cloud User and can damage the business and image of the Cloud Service Provider (SpringerLink, 2020).

4. INFRASTRUCTURE, ARCHITECTURE AND DEVELOPMENT TOOLS

The market concerning Cloud Technologies, Server Infrastructure, and Application Development is continuously changing and evolving. As DevOps (Development Operations) linked crucial gaps from SysOps (System Operations) concerning factors such as but not limited to configuration, monitoring, and deployment and its ability to improve application development, the rise of Function-as-a-Service (FaaS) is taking application development to a new level of efficiency.
Before being taken singularly, consider the following points to be analysed for each of the three sections inside this chapter (meaning FaaS Architecture, FaaS Development Tools, FaaS Cloud Infrastructure):

- Overview of today's situation and environment of the FaaS market
- Identification of the most relevant gaps in the market
- Suggestions/ideas/guides on how to fill the gaps mentioned above.

4.1 FaaS Architectures

FaaS is a particular case of serverless architectures. Serverless was initially used to describe applications that significantly or fully incorporate third-party, cloud-hosted applications and services, to manage server-side logic and state. The latter is also known as Back-end as a Service (BaaS) (Roberts, 2018). Serverless also includes the development of server-side logic. FaaS follows this serverless model. However, the code runs in stateless containers that are *event-triggered*, ephemeral (they run for a short time) and managed by a third party (a FaaS provider). FaaS is a way to build applications based on serverless architecture. The platform provider makes functions available and manages resource allocation when the function is triggered. If there are several concurrent invocations to a function, several instances are created. That is, dynamic scaling is automatically provided by the vendor paying for the resources that are used. Functions should be designed to do a small piece of work associated with an event. Cloud providers limit the size of the code and memory of functions to be loaded fast. Also, Cloud providers may limit the execution duration of a function (e.g., five minutes), which means that long-running applications may need to be redesigned and split into several functions. Those do not preserve the state across invocations. If the state needs to be persisted across requests, the state should be externalized to a file, cache, or database.

Some of the main gaps in current FaaS platforms affect the performance of applications.

- The performance of functions is affected by several factors.

Namely, initialization time, queueing time and execution time of functions. The initialization time is the time for preparing the container running the function (cold start time). Queueing time is the time spent in the FaaS system before the function is executed, for instance, waiting for available resources. The execution time will depend on the hardware, concurrency level (multitenancy), and function type. The former times are inherent to FaaS deployments. Applications should be enriched with metadata to define their requirements in terms of hardware, memory, and CPU needs for achieving the desired performance target. This information should be used for deciding on the deployment of the application on the FaaS platform.

- The stateless nature of functions limits the adoption of the FaaS model.

Sharing state across function invocations must be done through external components (e.g., a cache, database), which is costly and affects the function performance. The placement of the external state in the nodes where the FaaS platform runs has a significant impact on the performance of functions using that state. A central data repository will decrease performance since data may be far from the function, while a distributed data repository may produce consistency problems. This model will also compete for the platform resources at the nodes where it is deployed. A local data cache may improve the performance of functions as far as the memory assigned to the function store the cached data. If this is not the case, an external data cache is needed, which will slow down functions.

- Heterogenous geo-distributed infrastructure.

The deployment of FaaS applications is currently generally done in a cluster of homogeneous hardware. However, many applications may need to be geo-distributed, even running in a multi-cloud environment. The latter may happen for the nature of the application (e.g., some functions may need to be executed at the edge running on small devices) or because of regulations (e.g., data must be stored in a given country). The runtime FaaS infrastructure should consider the heterogeneity of resources, geo-distributed nature of the application and take care of the reconfiguration actions needed to avoid performance degradation. The infrastructure should provide monitoring features of functions and trigger reconfiguration actions if necessary.

4.2 FaaS Development Tools

Adaptation to the FaaS model can bring considerable benefits for the application developer regarding application maintainability, deployment, server maintenance, and cloud elasticity exploitation. However, it comes with the need for an application adaptation and the ability to combine different computing models such as legacy code, microservices and function implementations. Automation and Integration Services will play a vital role in this transition. Organizations require these services to help them manage several platforms and optimize the micro-services that drive the FaaS market (Reports and Data, 2020). However, there are still many significant issues that hinder harvesting of benefits from its adoption. The major challenges faced currently in FaaS environments with relation to development environments include lack of tooling support, e.g., testing, deployment, (55%), vendor lock-in (32%), managing state in functions (27%), little support for reusing functions (14%) and little support for composition of functions (12%) (Leitner, et al., 2019).

In terms of major commercial FaaS platforms, these typically do not come with a UI for workflow definition (Grohmann, et al., no date), except Apache Airflow, which includes the incorporation of operators to incorporate typical cloud services or processes. One drawback of Airflow is that these operators are typically provider-specific and cannot be reused while amplifying the vendor lock-in. Fission workflows are mainly programmatically defined while heavily linked with the Kubernetes environment. Proprietary solutions also exist with an extensive list of accompanying services such as the IBM Cloud (formerly Bluemix) environment and Google Composer (for managing Airflow related workflows) that offer integrated services design, deployment, and composition, however tightly coupled with the associated vendor lock-in. Regarding the European CSPs, it is evident that solutions targeting ordinary public provider

FaaS frameworks (AWS Lambda, GCF, Azure etc.) are not helpful towards attracting applications that can run primarily on their infrastructures. The latter is true for many serverless frameworks and related services, e.g., AWS Amplify, Up, Apache Airflow, Architect. Other frameworks (Claudia.js, Sigma, Riff) are primarily focused on code management and deployment, or focused on infrastructure as code (Pulumi), and enable multiple location deployments and cooperation with optimization frameworks for deciding function placement.

Abstraction layers are critical for the multi-platform nature of the current landscape. These enable a single point of (application) definition, supporting the transformation of its deployment towards multiple environments, for instance. Furthermore, integration capabilities between different systems, technologies and stacks are needed, especially in the current continuum approach, in which diverse applications and systems need to cooperate in order to achieve the common end. This integration should be abstract to the developer, who should easily compose workflows of functionalities, embedding code of different languages, and other structures (e.g., microservices, legacy code etc.).

The Serverless Framework is an open-source project that can be used to manage the lifecycle of a serverless architecture (build, deploy, update, delete) and deploy functions, events, and their required resources together via provider resource managers towards multiple providers or FaaS frameworks (including AWS, Openwhisk, Knative etc.). Thus, it can act as an abstraction layer for comfortable support of various frameworks. However, it is still a command line tool that should be enhanced by relevant workflow creation and dynamic code inclusion capabilities to ease application development. Moreover, ready-made combined functionalities that can help enrich applications management (e.g., from non-functional aspects such as performance, cost, security etc.) by a simple drag and drop manner would significantly optimize the final deployment and operation.

In recent years, visual environments have emerged as a user-friendly and abstract means of development that can speed up application development. Typically, these environments are based on flow programming, based on asynchronous event-driven languages such as JavaScript or Golang, and offer palettes of readymade nodes or operators that incorporate the major functionalities needed. Furthermore, they encompass means of extension for these nodes and external repositories in which such nodes or in general flows can be stored and shared by the community, while easily imported into existing flows. Environments such as open-source Node-RED[11] for event-driven applications typically are designed for a specific domain (e.g., Node-RED for the IoT). They do not include an end-to-end rationale, from design to development and deployment in one step, nor do they have ready-made patterns for exploiting the cloud model (yet they can be extended or adapted to do so).

In the cloud design patterns domain, due to the increasing complexity of Cloud Services, Big Vendors have promoted Pattern-Based development through new programming and deployment paradigms to build value-added services. This development methodology has the goal of providing complex services and resources by the interaction of simpler ones. It can be used to define proper orchestration actions across the cloud architecture layers following a description of the composite service (Amato, et al., 2017.). The pattern-based development has also proven remarkably successful in other domains, such as the Kubernetes environment, in which aspects such as the operators have helped extensibility, reusability, and integration of components, leading to a rich ecosystem of available services and functionalities. Kubeflow is an exciting stack of technologies that has a direct relation to PHYSICS. It includes a set of functionalities through which a developer can write python code primarily for AI tasks, package it and then deploy it (through Knative plugins) directly on a FaaS environment. Defined functions can also be reused, even in some elementary workflows. Kubeflow is positioned as an ML toolkit above Kubernetes while importing existing arbitrary code does not seem very flexible. Furthermore, it is tightly coupled with Knative as an underlying FaaS framework.

[11] http://nodered.org

One case here would be to enable the developer to define their own set of orchestrated actions, exploiting existing patterns to embed them into their application operation directly. Some of these patterns may exist in implementation (even directly in the FaaS model or enabling transformations to this) (Carvalho, et al., 2019). However, they need either full parameterization (so that they can be automatically configured) and/or wrapping around the core design framework to achieve maximum abstraction.

4.2.1 Gaps Analysis

Conclusively, a list of current gaps can be identified with relation to this domain:

End to end visual workflows for supporting design and deployment

Current design environments focus primarily on singular vertical stacks, targeting either a specific provider or application domain. Furthermore, they lack a unified and vertical approach to enable application definition, enhancement with features, and cloud deployment specification creation in an integrated manner. Either deployment specification or workflow specification are supported but not combined.

Application enrichment, encapsulation, and integration with FaaS services and functionalities

Another drawback is that current environments cannot aid an application in exploiting cloud benefits through suitable and ready-made supporting structures that enhance functional and non-functional aspects. They do not include design patterns and functionality automatically incorporated and configured in the application graph. Many of the proposed operators are providers and increase vendor lock-in. Wrappers of implementations for them to be used in the visual flow environments (e.g., through library subflows or function nodes) are needed, as well as relevant code injection processes, containerized implementations and interfaces in order to insert parameterized configuration information dynamically (e.g. base functions used/connected in the pattern). Through this approach, instances supporting these patterns will be automatically included and configured in the application graph.

Cost aspects in the FaaS model

Another aspect of application migration to the FaaS model refers to the aspect of cost, which is more dynamic and unpredictable in FaaS deployments. In this case it is based on aspects such as number of function invocations as well as performance issues (memory and execution time of a function) in contrast to typical server-based deployments in which the cost can be more easily calculated. Therefore, suitable mechanisms should exist in order to indicate a provider's performance to a given function (not necessarily at the design and development level) as well as mechanisms or structures that could enhance cost aspects from a pattern point of view and included as options in the design environment,

Exploitation of multicloud/edge facilities in a single environment

Ability to exploit the existence of multiple facilities for executing an application by splitting it into relevant chunks should be abstract to the application developer. While this approach is not necessarily driven by the design environment (more from the underlying platform services), the way the application is structured, defined and translated to a deployment graph should be abstract enough in the design environment in order to enable further flexibility at the platform side during instantiation. This implies the existence of an abstraction layer that can achieve that goal.

Link of application graph with infrastructure stack and management

The link between application-level components (even at the function level) and according infrastructure level features would enable an integrated environment in which the developer could directly embed in the graph various requirements and management aspects (e.g. scalability) at a fine grained level. These features could also be semantically enriched so that the environment can infer from these descriptions the most appropriate course of action or configuration.

4.3 FaaS Cloud Infrastructure

This section will perform a more technical analysis of the services' providers explored throughout section 3.1, focusing on the infrastructures of the aforementioned players.

Serverless computing is a cloud computing execution model in which the cloud provider allocates machine resources on demand, taking care of the servers on behalf of the users. Both the serverless applications as well as the functions are still running in some infrastructure, composed of several layers: e.g., servers running some Linux distribution, being part of a Kubernetes cluster that run on VMs on top of a cloud provider such as AWS or Azure, or an on-premise one based on OpenStack.

4.3.1 Kubernetes/OpenShift and the Hybrid cloud

Nowadays, the trend is to run applications by relying on containers as the deployment unit, where most of the code is packaged as container images. Today's de-facto standard for container orchestration is Kubernetes, and those container images are usually deployed as Kubernetes pods (a group of containers).
Kubernetes is an open-source container orchestration platform for automating the deployment, scaling and lifecycle management of podified applications. During the last couple of years, Kubernetes has gained industry acceptance which made it the foundation of modern infrastructure. OpenShift is the Red Hat's production-grade distribution of Kubernetes (in the same way as RHEL is a Linux distribution) where extra features are included to manage the OpenShift cluster lifecycle itself, as well as additional tools for developers such as source-to-image, integrated registry, routes, etc. Besides, it comes with support for Operators [12]so that other applications/infrastructure can be set up easily on top, in our case, the tooling related to serverless infrastructure.

OpenShift provides advantages with zero limitations, following a traditional hosting model and advanced traffic management techniques that help you connect the applications as needed. If you are already using Kubernetes, serverless is the next logical step, going the extra mile by providing developers with running code without worrying about infrastructure and saves on infra costs by virtually scaling your application instances from zero.

Additionally, with OpenShift, you can leverage the full potential of hybrid cloud computing, where you can consistently deploy the same infrastructure (OpenShift) across bare-metal, virtual, private, and public clouds. This unlocks a set of capabilities to achieve speed, agility, and portability, such àas scale-up/down without rebuilding applications, considering different locations, and enhancing stability and security.
There are some differences between Containers-as-a-Service (CaaS) and FaaS. CaaS based on Kubernetes and FaaS follows different approaches in deploying and executing code. FaaS is built on the philosophy of on-demand execution, per-second billing, and event-driven invocation while CaaS is designed to deliver scale and reliability of applications. While CaaS can be deployed within an enterprise data centre, FaaS is still mostly confined to the public cloud.

4.3.2 Knative

Knative, an open-source project developed by Google, IBM, Pivotal, Red Hat, and SAP, aims to make Kubernetes the best platform for running microservices and serverless applications. It is an abstraction layer for Kubernetes hiding the complexity involved in packaging and deploying applications. As an open-source cloud-native platform, it enables you to run your serverless workloads on Kubernetes, providing all Kubernetes capabilities, plus the simplicity and flexibility of serverless.

Behind the scenes, Knative is an amalgamation of a lot of CNCF and open-source products — such as Kubernetes, Istio, Prometheus, Grafana, and event-streaming engines, such as Kafka and Google Pub/Sub. It relies on a service mesh like Istio to manage traffic routing, revisions, and metrics.

[12] https://operatorhub.io/

Developers targeting Knative do not need to understand the intricacies of Kubernetes such as pods, deployments, services, and ingress. They package a container image as a microservice and deploy it to Knative which will handle the rest. In the CaaS world, Knative can be thought of as the correct way to perform autoscaling on Kubernetes. It solves the scale-to-zero use case that is a requirement if you want to autoscale correctly. It also plugs in essential components like a servicemesh (Istio) and the ability to perform logging and tracing in the correct fashion. If Kubernetes is the infrastructure, Knative is the platform component of the stack. It is designed to enhance the developer experience, deliver the serverless execution model, and to run event-driven code within Kubernetes. You can self-install Knative components to any Kubernetes cluster, and there are cloud providers that offer Knative-as-a-service, such as OpenShift. Knative allows developers to write loosely coupled code with its eventing framework that provides the universal subscription, delivery, and management of events. That means you can declare event connectivity, and your apps can subscribe to specific data streams. It integrates quite well with the Eventing engine, and it provides users with a seamless experience in designing decoupled systems. Users' applications code can remain completely free of any endpoint configuration, and it can publish and subscribe to events by declaring configurations in the Kubernetes layer. That is a considerable advantage for complex microservice applications. To deliver this, Knative offers two core building blocks — serving and eventing:

- Knative Serving: it is responsible for exposing, hosting, scaling, and managing the lifecycle of a microservice packaged as a container image. It supports scale-to-zero feature where a service automatically gets terminated when there are no requests for some time. After getting terminated, if the service receives a new request, the serving resource will immediately launch a new pod to handle it. Parameters such as the timeout, cooling period, and the maximum number of instances can be defined through Knative configuration associated with the service.
- Knative Eventing: Developers can write code to respond to events that are packaged as a container image and deployed as a Knative service. It comes with a broker that acts as a conduit between the event producers and consumers. Producers treat the broker like a hub by publishing all the messages to it. A trigger binds a consumer to the publisher through the broker. This loosely-coupled architecture makes it possible to deploy highly scalable eventing infrastructure.

Finally, Knative exposes the kn API interface using Kubernetes operators and CRDs. Using this, users can deploy their applications with the command line. In the background, Knative will create all of the Kubernetes resources (such as deployment, services, ingress, etc.) required to run your applications without you having to worry about it.
In short, Knative is more of a platform rather than merely an implementation. It gives you a broad and robust foundation for the implementation of your functions. So Knative does not offer FaaS service but gives you the tools to build your own FaaS solution on top of Kubernetes, using Knative's features.

4.3.3 OpenFaaS

As relates to FaaS, OpenFaaS is a cloud-native FaaS framework that can be deployed by a user on many different cloud platforms and bare-metal servers. OpenFaaS can be installed on Kubernetes and Docker swarm. The framework includes an API Gateway, asynchronous queue worker, and monitoring with Prometheus. The installation process is automated and deploys all the components necessary for running on the chosen platform. The Watchdog lives inside the function container. Docker is not the only runtime available in Kubernetes so that others can be used.

The architecture of OpenFaaS is relatively simple. The API Gateway can be invoked synchronously or asynchronously via Kafka, SNS, CloudEvents, CRON and other triggers. NATS Streaming handles asynchronous invocations. The autoscaling is performed using Prometheus and Prometheus Alertmanager, but it appears that it could also be swapped out to use Kubernetes' HorizontalPodAutoscaler. The OpenFaaS Gateway provides a route to the services deployed with OpenFaaS. A consumer calls the Gateway and their request is then routed through to the correct function. Metrics are automatically collected in Prometheus, and these metrics can be used to auto-scale your functions to deal with changes in load.

4.3.4 Multicluster - Submariner

As edge or multiple cluster deployment models become more common. In order to facilitate features like geo-redundancy, scale, fault isolation for the applications (or the services/functions in the CaaS/FaaS model), there is a need for having Kubernetes cluster that can span multiple cloud providers, data centres or regions. What's more, it is also needed to support different types of Kubernetes deployments. For instance, in some edge location, the resources available may not be strong enough for a standard HA Kubernetes deployment, and a lighter solution may be needed (single node OpenShift, k3s, etc.).
In order to add flexibility to the application/functions deployment on a multicluster/edge infrastructure, there is a need for further enhancements on the connectivity/discovery between applications running at different edges/clouds. To that end, Submariner is an Open Source project focused on enabling direct networking between Pods and Services in other Kubernetes clusters, either on-premise or in the cloud. It is CNI agnostic and can link different Kubernetes distributions together by providing:

- cross-cluster L3 connectivity using encrypted VPN tunnels
- service discovery across clusters

5. MAIN INSIGHTS AND RECOMMENDATIONS

5.1 SWOT Analysis: Consequences on Different Market Sectors Analysis

The SWOT analysis performed below was aimed to assess and analyse the strengths, weaknesses, opportunities and threats of FaaS both for the developers on the one hand and for serverless providers on the other hand. The aim of conducting this analysis is to define the pros and cons of this cloud computing model in the FaaS providers' shoes such as AWS Lambda and Microsoft Azure and in the customer side (developers for businesses), while foreseeing the main drivers through an outlook of the threats and weaknesses that could potentially impact players on both sides (developers and providers). The insights from the analysis conducted across this deliverable are synthesized in the SWOT analysis performed below:

Table 2 – SWOT Analysis

Strengths
1.1 Increased level of efficiency, simplicity and productivity for businesses as it avoids the constraints of the underlying infrastructure
1.2 Automatic and rapid scaling and deployment: the services run smoothly and rapidly even when there are numerous requests.
1.3 Rapid deployment : less time between the project ideation and its execution
1.4 Flexible and reliable while avoiding the constraints of the underlying infrastructure
1.5 Cost-saving through the "pay-per-use" pricing model
1.6 Less administration overhead and constraints
1.7 Availability of various coding languages
1.8 Developed and broad availability of open sources framework

Weaknesses
2.1 Security-related risks
2.2 More latency in the execution phase
2.3 Constraints with limitations for executing the functions
2.4 Vendor lock-in
2.5 Risk of high operational costs
2.6 Highly competitive market segment

Opportunities
4.1 High market potential in many growing industries and Industry 4.0
4.2 Integration into new business models
4.3 Continuous growth of the cloud technology
4.4 Increased use of FaaS by Financial institutions
4.5 Providing FaaS services compliant with GDPR will make them even more attractive to SMEs.

Threats
3.1 (GDPR) Security policies compliance and vulnerabilities
3.2 The development of Blockchain technologies could outperform FaaS technologies on the long-run

<u>Strengths</u>

Overall, making cloud computing serverless through FaaS enables increased efficiency, simplicity, and productivity for businesses. Indeed, one of the main benefits of FaaS lies in the fact that it avoids the constraints of the underlying infrastructure. The increased demand for asset-light and serverless infrastructural approach has consequently led to a growing demand for FaaS providers in a more and more digitalized era for traditional businesses and industries. On the other hand, however, this requires FaaS providers strong internal capabilities and to come up with technology almost foolproof sure in order to ensure the quality of the service and that operations run smoothly.

Moreover, serverless cloud computing providers succeeded in making their product run smoothly and rapidly even when there are numerous requests, enabling automatic and rapid scaling. These vendors (FaaS providers) are also in charge of managing the scaling part, which takes a thorn out of businesses' side.

Another aspect which makes FaaS a rapidly growing segment in the cloud computing market increasingly popular, is the rapid deployment it enables. For businesses, there is less time between the project ideation and its execution. Serverless cloud computing is also flexible and reliable: the serverless provider takes care of the setting and executing the application and is able to do it in a qualitative way as code runs smoother.

Furthermore, given the pay-per-use pricing model, businesses only need to pay to execute the functions and the resources used (memory, storage, CPU…), and not anymore for idle time. Therefore, this pricing model results in cost savings in most cases for businesses as it does not imply extra costs such as acquiring and installing the servers, maintenance costs, operating system management costs, etcetera. Hence, it corresponds better to the output levels of the service's actual consumption.

For FaaS providers, on the other hand, it implies that forecasting their revenues from such service is harder due to the lack of fixed-bundles because of the pay-per-use pricing model they offer to businesses.

Moreover, with the increased traffic, deploying and executing functions efficiently and automatically is a real challenge for serverless cloud computing providers. In serverless cloud computing, automatic scaling is a critical success factor, and players like AWS Lambda and Azure can satisfy the demand in this area as the functions' execution runs smoothly. The systems automatically adjust the capacity to maintain steady and predictable performance at the lowest cost.

The core purpose of FaaS implies less administration overhead and constraints. Going serverless means that businesses do not need to manage the servers anymore and to hire specialized staff in charge of managing infrastructure, thus saving costs and reducing complexity.

Current most prominent players in the FaaS segment have deployed their product while enabling high availability of various coding languages. Hence, despite going serverless, the main players' products still allow coding in any language required (HTML, Java, Node.js, Python, etc.).

Open source serverless frameworks are highly available and developed. Its main advantage lies in the fact that it can be deployed on any infrastructure.

Weaknesses

One of the main weak spots in FaaS cloud computing is related to security risks. Security-related vulnerabilities need to be addressed as a few issues remain in this domain and can therefore harm businesses and their willingness to go serverless (function event data injection, broken identification, insecure serverless deployment configuration, insecure application secrets storage, etc.).

Latency is another major drawback with regards to serverless cloud computing. Latency in the execution phase might occur. Improving efficiency and execution of the functions, addressing the latency issue, could be a pivotal area of improvement for market players.

Some other issues may be mentioned regarding the execution of the functions in FaaS. There are some constraints aside from the latency with limitations when executing the functions. For instance, the lack of memory is the main one when it comes to coding size.

Vendor lock-in can potentially become a drawback as being stuck with one cloud provider may become harmful and tend to decrease the agility in businesses' operations. In other words, once one begins working with a vendor (Cloud Service Provider) and that it starts executing functions and moving workloads into the cloud, it might be hard afterwards to drive them away and take control back. Businesses' data is critical, and with this data being managed by external vendors, cloud customers become highly dependent on these external providers.

Despite the pricing model being on paper appealing and beneficial, it is yet key to monitor and mitigate the potential risks of high operational costs related to overhead and retrials due to data not being optimal for FaaS.

On the vendor side, the main issue in this market is that with the cloud adoption among businesses becoming more and more mainstream, it becomes increasingly difficult and challenging for new players to penetrate the market. With huge existing market players already operating in this ever-growing market in terms of size (volume and value) and the competition intensifying with the China giants such as Alibaba also breaking into the market with extensive internal capabilities and strong financing capabilities, the cloud computing industry can already be referred as a red ocean.

Threats

FaaS frameworks will need to satisfy GDPR in Europe. Developers still need to provide end-users with accessible and transparent ways to comply with foundational pillars of GDPR. The increased regulatory requirements are inferring an increased risk concerning breaches of data. Additionally, in some sectors where data is highly subjected to GDPR such as patient information, development of services will have to ensure full security.

As observed in chapter 2.1, under the focus on *Financials*, the blockchain growing trend has the potential to maintain its current development and importance and, ultimately, develop functionalities that can substitute the FaaS platform. While this scenario is yet to be clearly defined, it is an eventuality that has to be kept into account as it may disrupt FaaS long-term survival.

Opportunities

First of all, it is crucial to consider the high market potential in many growing industries and overall Industry 4.0. Indeed, advancements in new technologies have sped up the transition towards a more digitalized world. The development of data-driven technologies and connected objects such as Machine Learning, AI, Big Data, IoT and 5G in many sectors represent a significant growth opportunity for FaaS.

Secondly, the potential integration into new business models. The COVID-19 pandemic has sped up a transition towards smart technologies. In the healthcare sector, benefits such as enhanced data usage, medical research and lowering costs, drive the market. The e-Health sector relies mainly on cloud technology and is forecasted to grow at an exponential pace. Smart agriculture and manufacturing are primarily driven by IoT and require large amounts of data to be processed. Lower resource allocation and the need for real-time management technology will create immense opportunities for FaaS integration.

Next, the continuous growth of the cloud technology, (15% CAGR over 7 years), whether used in mobile apps, streaming platforms, storage services or web applications, is correlated directly with the growth of FaaS technology, as the pricing model of FaaS can translate into drastic cost savings for service providers and consequently for consumers.

Another pivotal opportunity lies in the increased use of FaaS by Financial institutions. FaaS technology is already widely adopted by the financial sector for multiple reasons. Indeed, financial institutions are increasingly using FaaS for automation of routine operations and facilitating cost reduction. As investments in the financial sector towards digitalisation keep increasing, so does the industry's demand for FaaS services.

Conclusively, providing FaaS services compliant with GDPR will make them even more attractive to SMEs. SMEs often struggle to comply with GDPR due to its complexity. Some studies even argued that GDPR harms competition within the EU as start-ups tend to be more challenged by regulation than Big-Tech companies. For this reason, providing FaaS services compliant with GDPR will make them even more attractive to SMEs.

5.2 Core Challenges to overcome

Building upon the Strengths, Weaknesses, Opportunities and Threats identified in the deliverable, and synthesised in the SWOT in the previous section of this chapter (5.1), some core challenges which providers will have to overcome to thrive in the future have been drafted. Below, they are shown in the figure (figure 7). It illustrates five core challenges that aim to minimize threats and weaknesses, maximise opportunities, and leverage strengths, following the TOWS analysis fundamentals. Overall, opportunity one (0:4.1), the high market potential for FaaS technologies in many growing industries, is at the core of most of the core challenges listed.

	Core Challenge	Opportunities/ Threats	Strengths/ Weaknesses
Challenge 1	How to exploit the technology advancing and disruptive trend by leveraging on FaaS efficiency, simplicity and productivity without lacking in distinctiveness of the offering and security.	O: 4.1	S: 1.1 W: 2.6
Challenge2	How to address the high market potential in many growing industries leveraging on the "pay-per-use" pricing model while minimizing the risk of high operational costs.	O: 4.1	S: 1.5 W: 2.5
Challenge3	How to tackle latency in data execution in relation to client demand of more reliable/direct deployment as this is currently limiting growth prospect due to current use of alternative cloud computing services for such.	O: 4.2	W: 2.2
Challenge4	How to face flexible demand for solutions through FaaS characterized auto-scaling nature maintaining performances high despite latency and execution constraints.	O: 4.1	S: 1.2 W: 2.2, 2.3
Challenge5	Ensuring operational compliance for FaaS users by a stronger integration of privacy-by-design in the FaaS environment to grow in various highly regulated sectors such as eHealth and Financial Services.	O: 4.4 T: 3.2	S: 1.1 W: 2.1

Figure 8 – Core Challenges

5.3 Critical Success Factors

After establishing the core challenges, the deliverable aims to explore the critical success factors needed to overcome those challenges.

- IT capabilities

FaaS is one of the most promising cloud computing services and is increasingly popular among businesses in various industries. Developing the aforementioned service for enterprises requires robust internal capabilities on the provider side due to the complex infrastructure behind operating the product. Developing this technology is not an easy task. Indeed, the competition is intense, and the cloud computing industry keeps growing with FaaS adoption becoming more and more widespread. Consequently, it becomes increasingly challenging for new providers to break into the market to develop and provide the appropriate technology offering that fits users' demands. Therefore, it requires consistent IT capabilities to build a robust FaaS solution for businesses on the provider side.

- Financial capabilities to support heavy investments and development costs

Building a robust FaaS solution comes with strong financial capabilities required to support the heavy development costs needed to build, run and operate the serverless cloud computing infrastructure. Currently, leading players such as AWS Lambda, Google Cloud Functions, Microsoft Azure, and emerging players such as Alibaba and IBM, share financing capabilities from their status as worldwide powerhouses.

- Technical features as a competitive edge

In order to provide a comprehensive service as well as value for money, FaaS cloud providers need to offer a cloud platform with reliable technical characteristics as these features may represent a vital differentiating factor against the increasingly fierce competition. When choosing their preferred serverless cloud computing provider, users demand several technical features, such as scalability and deployment to execute functions under the cloud platform, the flexibility of the offer (execution on-demand), storage capacity and the ability to program in any language. The ability to deploy and execute functions efficiently, and at scale, is challenging due to increasing traffic and can indeed be a competitive advantage for providers who will succeed to deliver it. Automatic scaling is also pivotal; services such as those offered by AWS Lambda and Azure can satisfy the demand in this area as the functions' execution runs smoothly. Therefore, all these features represent a critical success factor as successfully developing the cloud offering will be a crucial component in order to become a successful player in the FaaS market.

- IT Management & Implementation of serverless infrastructure

Developing FaaS requires IT management of the underlying infrastructure to make the cloud service run and operate efficiently. Infrastructure management should be integrated into the continuous delivery pipeline so that infrastructure stacks are updated together with any code changes. Moreover, dependencies between separate stacks have to be understood and handled to make infrastructure management more effective. In the meantime, when offering their service to businesses, transparency becomes increasingly essential and needs to be tackled by FaaS vendors. This feature mainly relates to the storage, maintenance and usage of client sensitive information reinforcing client trust and ensuring regulatory compliance. Vendors should take into account customer expectations of their platforms, as businesses exploiting this service rely more and more on their hosting capabilities.

- Cost and pricing model management

One of the most significant advantages of Serverless lies in its pricing model. Indeed, the pay-per-use model and the asset-light rationale infers a considerable financial benefit for its users because they pay exclusively when executing the service and do not need to invest directly into assets using their own cash assets or debt. Different providers offer different pricing models, but they generally have similar market-clearing prices, and they all allow customers to access various services, and layers within those services, paying only for actual computing usage. FaaS offers dramatic cost efficiencies and removes operational complexities. Consequently, developers can focus on their code without worrying about computing constraints or operating costs.

- External regulation

On one side, with increasing data regulation in the EU, FaaS vendors need to be aware of and comply with continuous amendments in the respective field to avoid significant penalties due to possible data breaches. Any breach or data leak would result in a lack of trust for both the customers, especially in industries with sensitive information (e.g., Healthcare, Financial Services) and external regulators. On the other side, governing decision-makers need to provide tailored frameworks in which the aforementioned vendors can operate. The EU should attempt to incorporate the promising future and implications of FaaS technology to lay out a legal and fair playing field. Policymakers should seek to find solutions which provide vendors and developers with enough freedom, where possible, to be able to generate an internationally competitive platform.

5.4 Value Proposition

Conclusively, the deliverable conceptualized a value proposition that can guarantee pain relievers and gain creators to overcome the core challenges detected, leveraging the outlined critical success factors (figure 8). On the one hand, the figure shows the most significant pains perceived in the document, especially in the users' exploration section (3.2) and in the interviews conducted. On the other hand, it indicates the aspirations (gains) of end-users exposed in the document. The FaaS cloud offering, which takes into consideration all the insights explored throughout the deliverable would: "guarantee access to a scalable, capacious, secure, cost-effective, European Cloud Framework, compliant with GDPR". More details regarding pains, gains, pain relievers, and gain creators can be observed in the figure.

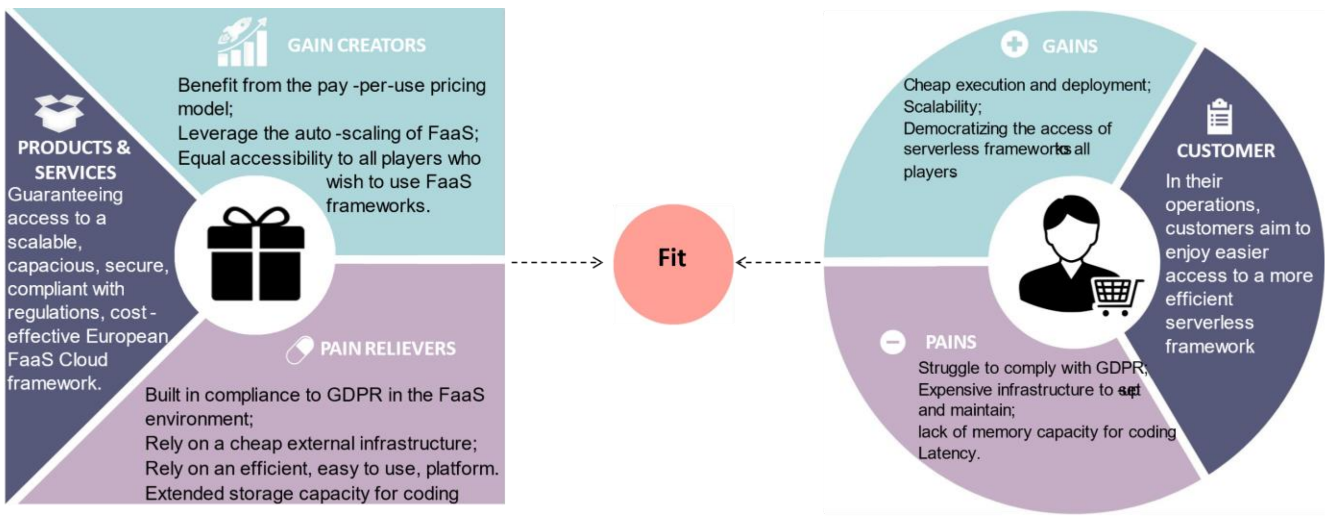

Figure 9 – Value Proposition

6. CONCLUSION

Deliverable D2.1 reviewed the FaaS market to evaluate the potential of the Cloud Computing Service. It identified the most pivotal areas to address, and the critical characteristics of FaaS to develop, to exploit market trends and growing demands predicted over the foreseeable future. Examining the short and long-term prospects in more detail, the aforementioned growth implies a robust positive trajectory due to the main pull-factors around FaaS deployment, execution, and development of functions and application solutions.

Among the critical aspects of the FaaS-technology to leverage, cloud-based technology allows a simple implementation and development that does not require expensive and complicated infrastructure to be managed. Frankly, the technology is an asset-light application development method and requires no servers for the users – hence, the term serverless. For this reason, FaaS technology is perfectly tailored for users with limited resources and technical skills as the FaaS environment is carefully developed to regulatory, technological, and user-experience requirements. Nevertheless, it is suitable for international firms seeking to pursue economies of scale in their operations effectively.

Among the wide range of solutions that FaaS provides, applications in the e-Health and Smart Agriculture sector provide considerable operational cost reduction concerning the economies of scales and synergies by FaaS deployment. Here as well, relying on external providers for serverless infrastructure saves on operating expenses and reduces the need for acquiring noncurrent assets at the expense of large upfront cash investments with additional executional and depreciative costs.

A successful strategy needs to consider core pitfalls to mitigate risks for non-conversions for customer targets. For example, internal R&D strategies regarding the security of the framework, the public cloud sphere, and compliance with relevant data protection regulations are deemed distinctly relevant. If well developed, FaaS technology will undoubtedly be used to support growing technologies that are becoming increasingly important in a sector and stage-agnostic manner.

According to our analysis results, we have indicated five core challenges to address in order for FaaS providers to leverage on the expected market growth and facilitate the most outstanding value offering. One key finding is the notion of leveraging the cost and executional efficiency and disruptiveness of FaaS without compromising on data integrity. This demand-factor is especially prevalent in e-Health. As political and corporate ambitions to consolidate patient data cross-entity and digitize medical treatments via applications increase, risks regarding data breach management and general security are somewhat hindering such a growth.

Concerning the elastic demand for solutions through FaaS' auto-scaling nature, key considerations involve maintaining efficiency on performances despite technology limitations involving latency and execution constraints. There is a significant presence of established players in the manufacturing industry, specifically automotive, companies such as Volkswagen, BMW and Toyota engaged in partnerships with Google Cloud and Amazon Web Services and active users of FaaS technologies. At the same time, in the future, FaaS is expected to achieve even higher importance by delivering the so-called Features on Demand (FoD). As mentioned in the previous chapter, it would bring to concreteness the vision of allowing customers to download and incorporate in the vehicles services and characteristics not included in the purchase moment.

Complemented with the main insights and suggestions identified throughout the document, this deliverable's scope includes mainly identifying critical guidelines to cover existing gaps in the market. For this purpose, the described indications hope to contribute to creating a solution capable of innovating and enhancing the market's technology landscape.

Closing, this deliverable will be directly connected with the deliverable D2.2 in the context of an initial market analysis of the available technologies and solutions which underline and complement the initially identified gaps and shortcomings that the PHYSICS project aims to fill and resolve. With this in hand, the deliverable D2.2 will build on top of the information provided by this deliverable and create a state-of-the-art analysis on each technological and research axis of PHYSICS by thoroughly assessing the information provided by D2.2. Furthermore, D2.1 will identify the initial user and system requirements considering the needs and the aspects that occur in the current communities and solutions analysed in this deliverable while also combining them with the needs of the project use cases and overall research and system goals.

REFERENCES

A Cloud Guru News., 2021. Does AWS Lambda keep its serverless marketing promise of continuous scaling? A Cloud Guru. Available at: https://acloudguru.com/blog/engineering/does-aws-lambda-keep-its-serverless-marketing-promise-of-continuous-scaling

A., 2018. AWS vs Azure : Pros & Cons for Your Business. Squadex. Available at: https://squadex.com/insights/aws-vs-azure-pros-cons-for-business/

A., 2018. AWS vs Azure : Pros & Cons for Your Business. Squadex. Available at: https://squadex.com/insights/aws-vs-azure-pros-cons-for-business/

Allied Research, 2020. Function-as-a-Service (FaaS) Market to Reach $24.00 Billion by 2026: Allied Market Research (AMR). Available at: Function-as-a-Service (FaaS) Market to Reach $24.00 Billion by 2026: Allied Market Research (AMR)

Amato, F. and Moscato, F., 2017. Exploiting cloud and workflow patterns for the analysis of composite cloud services. Future Generation Computer Systems, 67, pp.255-265.

Amazon Web Services, 2014 : Case Studies - UCAS, Interview with James Munson | AWS – Lambda. Available at: https://aws.amazon.com/solutions/case-studies/ucas/

Amazon Web Services, 2016 : Case Studies - Benchling | AWS – Lambda. Available at: https://aws.amazon.com/solutions/case-studies/benchling/

Amazon Web Services, 2019 : Case Studies - Financial Engines | AWS. Available at: Lambdahttps://aws.amazon.com/solutions/case-studies/financial-engines/

Amazon Web Services, 2020 : AWS Executive Insights: Interview with Roland Scharrer, the CTIO of AXA Group | AWS – Lambda. Available at: https://aws.amazon.com/fr/executive-insights/france/

Ammar Albayati, Nor Fadzilah Abdullah, Asma Abu-Samah, Ammar Hussein Mutlag and Rosdiadee Nordin, 2020 : Serverless Advanced Metering Infrastructure Based on Fog-Edge Computing for a Smart Grid: A Comparison Study for Energy Sector in Iraq | Research Gate. Available at https://www.researchgate.net/publication/344805466_A_Serverless_Advanced_Metering_Infrastructure_Based_on_Fog-Edge_Computing_for_a_Smart_Grid_A_Comparison_Study_for_Energy_Sector_in_Iraq

Apache Airflow workflow definition and monitoring tool, available at: https://airflow.apache.org/

Artefact., n.d. Are Cloud and AI inseparable? Available at: https://www.artefact.com/news/are-cloud-and-ai-inseparable/

Assist-software.net. 2019. Pros and Cons of Serverless Computing. FaaS comparison: AWS Lambda vs Azure Functions vs Google Functions. Available at: https://assist-software.net/blog/pros-and-cons-serverless-computing-FaaS-comparison-aws-lambda-vs-azure-functions-vs-google
Available at: https://doi.org/10.1007/s42979-020-00353-2
Available at: https://ec.europa.eu/digital-single-market/en/european-cloud-initiative
Available at: https://www.globenewswire.com/news-release/2019/04/08/1798931/0/en/Study-55-Percent-Say-API-Integration-Is-Critical-to-Business-Strategy.html

AWS Lambda – Product Features., 2021. Amazon Web Services, Inc. Available at: https://aws.amazon.com/lambda/features/

AWS Lambda vs Google Cloud Functions | What are the differences? (2021). StackShare. Available at: https://stackshare.io/stackups/aws-lambda-vs-google-cloud functions#:%7E:text=AWS%20Lambda%20supports%20unlimited%20functions,a%20maximum%20of %20400%20executions.

Azeez, B., & Azeez, B., 2020. Pros and Cons of AWS (Amazon Web Services) Lambda. The Best Software Team across Asia. https://www.ateamindia.com/pros-and-cons-of-aws-lambda/

Baumgart, B., 2020. Serverless or Containers? Why Not Both?. Extreme Networks. Available at:https://www.extremenetworks.com/extreme-networks-blog/serverless-or-containers-why-not-both/

Belatrix Software, n.d. Mitigating serverless and FaaS security threats. Available at: https://www.belatrixsf.com/whitepapers/serverless-FaaS-security-threats/

Brode, B., 2020. Why the Serverless Revolution Has Stalled. Available at: https://www.infoq.com/articles/serverless-stalled/.

Business, I., 2021. Les 10 tendances Cloud en 2021. IT for Business. Available at: https://www.itforbusiness.fr/les-10-tendances-cloud-en-2021-41991

Cardoso, J., 2020. Google Cloud Run vs. AWS Lambda : Is Cloud Run a Serverless Game Changer ? (Part 1). IOD. Available at: https://iamondemand.com/blog/google-cloud-run-vs-aws-lambda-is-cloud-run-a-serverless-game-changer-part-1/

Cardoso, J., 2020. Google Cloud Run vs. AWS Lambda : Is Cloud Run a Serverless Game Changer ? (Part 3). IOD. Available at: https://iamondemand.com/blog/google-cloud-run-vs-aws-lambda-is-cloud-run-a-serverless-game-changer-part-3/

Cardoso, J., 2020. Google Cloud Run vs. AWS Lambda : Performance Benchmarks (Part 2). IOD. Available at: https://iamondemand.com/blog/google-cloud-run-vs-aws-lambda-performance-benchmarks-part-2/

Carey, S., 2020. AWS vs Azure vs Google Cloud : What's the best cloud platform for enterprise? Available at: Computerworld. https://www.computerworld.com/article/3429365/aws-vs-azure-vs-google-whats-the-best-cloud-platform-for-enterprise.html

Carey, S., 2020. AWS vs Azure vs Google Cloud : What's the best cloud platform for enterprise? Computerworld. Available at: https://www.computerworld.com/article/3429365/aws-vs-azure-vs-google-whats-the-best-cloud-platform-for-enterprise.html

Carvalho, L. and de Araújo, A.P.F., 2019. Framework Node2FaaS: Automatic NodeJS Application Converter for Function as a Service. In Proceedings of the 9th International Conference on Cloud Computing and Services Science (Vol. 1, pp. 271-278).

Casalboni, A., 2019. Bluemix OpenWhisk: IBM's Function as a Service. Available at: https://cloudacademy.com/blog/ibm-bluemix-openwhisk-serverless/

Casale, G., Artač, M., Van Den Heuvel, Wj. Et Al. Radon, 2020: Rational Decomposition And Orchestration For Serverless Computing. Sics Softw.-inensiv. Cyber-phys. Syst. 35, 77–87. Available at:Https://Doi.Org/10.1007/S00450-019-00413-w

Castellazzi, L., Avgerinou, M. and Bertoldi, P., 2017. Trends in data centre energy consumption under the European Code of Conduct for Data Centre Energy Efficiency. Available at: https://publications.jrc.ec.europa.eu/repository/bitstream/JRC108354/kjna28874en

Clement, J., 2018. Most expensive acquisitions by Google as of September 2017. Statista. Available at: https://www-statista-com.revproxy.escpeurope.eu/statistics/199275/most-expensive-acquisitions-by-google-since-2006/

Cloudflare, n.d., What is Function-as-a-Service (FaaS)? | Cloudflare. Available at: https://www.cloudflare.com/fr-fr/learning/serverless/glossary/function-as-a-service-FaaS/

Cloud Industry Forum, 2021. What's all the FaaS about? | Cloud industry forum. Available at: https://www.cloudindustryforum.org/content/whats-all-FaaS-about

Code Project, 2019. Serverless Computing: An Introduction to FaaS with Machine Learning. Available at: https://www.codeproject.com/Articles/1426308/Serverless-Computing-An-Introduction-to-FaaS-wi-11

Deloitte, 2019. Artificial intelligence and DevOps foster agile cloud computing | Deloitte Insights. Available at: https://www2.deloitte.com/us/en/insights/focus/signals-for-strategists/artificial-intelligence-and-devops-for-cloud-computing.html

Dignan, L., 2021. Top cloud providers in 2021 : AWS, Microsoft Azure, and Google Cloud, hybrid, SaaS players. ZDNet. Available at: https://www.zdnet.com/article/the-top-cloud-providers-of-2021-aws-microsoft-azure-google-cloud-hybrid-saas/

Dignan, L., 2021. Top cloud providers in 2021 : AWS, Microsoft Azure, and Google Cloud, hybrid, SaaS players. ZDNet. Available at: https://www.zdnet.com/article/the-top-cloud-providers-of-2021-aws-microsoft-azure-google-cloud-hybrid-saas/

Ellis, A., 2021. Introducing the Template Store for OpenFaaS. Available at: https://www.openFaaS.com/blog/template-store/

Ermanno Guardo, Alessandro Di Stefano, Aurelio La Corte, Marco Sapienza, Marialisa Scatà, 2018 : A Fog Computing-based IoT Framework for Precision Agriculture | Research Gate. Available at https://www.researchgate.net/publication/328599954_A_fog_computing-based_IoT_framework_for_precision_agriculture

Fauna, 2021. A Comparison of Serverless Function (FaaS) Providers. Fauna. Available at: https://fauna.com/blog/comparison-FaaS-providers

Fisher, P. 2018. The We generation: Volkswagen has big plans for its digital app ecosystem including retail tie-ups and Function as a Service (FaaS) in near future. Available at : https://www.automotiveit.com/connected-car/the-we-generation-volkswagen-has-big-plans-for-its-digital-app-ecosystem-including-retail-tie-ups-and-function-as-a-service-FaaS-in-near-future/5708.article?adredir=1

Fission Open source Kubernetes-native Serverless framework, available at: https://fission.io/

Fission.Io, 2020. Serverless Functions for Kubernetes. Available at: https://fission.io/features/

Foglets, 2019: Wireless Body Area Network (WBAN) | Foglets. Available at: https://foglets.com/wireless-body-area-network-wban/

Foley, M. J., 2019. How Microsoft's Azure organization makes the Windows sausage. ZDNet. Available at: https://www.zdnet.com/article/how-microsofts-azure-organization-makes-the-windows-sausage/

Function-as-a-Service (FaaS) Market By User Type, By Service Type, By Application, By Deployment Mode, and By Industry Vertical, 2016-2026, Reports and Data, 2020.

Gancarz, R., 2021. The economics of serverless computing: A real-world test | TechBeacon. Available at:https://techbeacon.com/enterprise-it/economics-serverless-computing-real-world-test#:~:text=When%20looking%20at%20the%20price,cents%20for%201%20million%20invocations>

Gartner, n.d. The CIO's Guide to Serverless Computing. Available at: https://www.gartner.com/smarterwithgartner/the-cios-guide-to-serverless-computing/

Geinow, M. 2018. Why Netflix rolled its own node.js Function-as-a-Service for its API Platform. The *NewStack.* Available at: https://thenewstack.io/why-netflix-rolled-its-own-node-js-functions-as-a-service-runtime/

Github - Fission. 2021. Available at: https://github.com/fission/fission

Github - Kubeless. 2021. Available at: https://github.com/kubeless/kubeless

Github - OpenFaaS. 2021. Available at: https://github.com/openFaaS/FaaS

Github - Openwhisk. 2021. Available at: https://github.com/apache/openwhisk

Google Composer offering, available at: https://cloud.google.com/composer

Gr, Y. and montagne, 2018. FaaS (Functions-as-a-Service) : la révolution serverless. Available at: https://itsocial.fr/enjeux-it/enjeux-cloud-computing/cloud-public-prive-hybride/FaaS-functions-as-a-service-revolution-serverless/

Grand View Research, 2021. Cloud Computing Market Size, Share | Industry Report, 2020-2027. Grandviewresearch.com. Available at: https://www.grandviewresearch.com/industry-analysis/cloud-computing-industry

Greene, D., 2016. Google to acquire Apigee. Google Cloud Blog. Available at: https://cloud.google.com/blog/products/gcp/google-to-acquire-apigee

Gupta, S. K., 2020. Amazon SWOT 2020 | SWOT analysis of Amazon. Business Strategy Hub. Available at: https://bstrategyhub.com/swot-analysis-of-amazon-amazon-swot/

H. Shafiei, Member, IEEE, , A. Khonsari, and P. Mousavi, 2019: Serverless Computing: A Survey of Opportunities, Challenges and Applications | Research Gate Available at https://www.researchgate.net/publication/338108229_Serverless_Computing_A_Survey_of_Opportunities_Challenges_and_Applications

Harbridge, R., 2016. Microsoft Azure and the Competitive Public Cloud Industry : What You Need to Know. Slideshare. Available at: https://fr.slideshare.net/rharbridge/microsoft-azure-and-the-competitive-public-cloud-industry-what-you-need-to-know

Harvey, C. H., 2020. AWS vs. Azure vs. Google : 2021 Cloud Comparison. Datamation. Available at: https://www.datamation.com/cloud-computing/aws-vs-azure-vs-google-cloud-comparison.html

Harvey, C. H., 2020. AWS vs. Azure vs. Google : 2021 Cloud Comparison. Datamation. Available at: https://www.datamation.com/cloud-computing/aws-vs-azure-vs-google-cloud-comparison.html

Hood, C., & Kasiviswanathan, B., 2020. What is API-first? 5 opportunities to create business value. Google Cloud Blog. Available at: https://cloud.google.com/blog/products/api-management/api-first-digital-transformation

IBM Cloud Docs., 2020. System details and limits. Available at: https://cloud.ibm.com/docs/openwhisk?topic=openwhisk-limits

Israel, T., 2018. Top 10 Security Risks in Serverless Architectures - DZone Security. Available at: https://dzone.com/articles/top-10-security-risks-in-serverless

IT for Business., 2018. Atouts et limites du Serverless Computing…. Available at: https://www.itforbusiness.fr/atouts-et-limites-du-serverless-computing-18219

Jaber, O. (n.d.). Serverless ou FAAS (Function As A Service), une révolution ? Available at: www.theodo.fr. Available at: https://www.theodo.fr/digital-et-strategie/serverless-FaaS-une-revolution

Jeff Barr, 2017 : Things Go Better With Step Functions - Coca Cola | Amazon Web Services. Available at https://aws.amazon.com/blogs/aws/things-go-better-with-step-functions/

Jones, E. J., 2020. Google Cloud vs Azure in 2021 (Comparing the Giants). Kinsta. Available at: https://kinsta.com/blog/google-cloud-vs-azure/

Jones, E., 2020. Cloud Market Share – a Look at the Cloud Ecosystem in 2021. Kinsta. Available at: https://kinsta.com/blog/cloud-market-share/

Jones, E., 2020. Cloud Market Share – a Look at the Cloud Ecosystem in 2021. Kinsta. Available at: https://kinsta.com/blog/cloud-market-share/

Khalidi, Y., 2020. Microsoft announces definitive agreement to acquire Metaswitch Networks, expanding approach to empower operators and partner with network equipment providers to deliver on promise of 5G. Available at: https://blogs.microsoft.com/blog/2020/05/14/microsoft-announces-definitive-agreement-to-acquire-metaswitch-networks-expanding-approach-to-empower-operators-and-partner-with-network-equipment-providers-to-deliver

Kindgeek, 2019. Serverless: what, why, when?. Medium. Available at: https://kindgeek.medium.com/serverless-what-why-when-e725940308fe

Kinsta., 2020. Azure Market Share : Revenue, Growth & Competition (2021). Available at: https://kinsta.com/azure-market-share/

Kubeflow Stack, Available at: https://www.kubeflow.org/

Kubernetes Operator hub, Available at: https://operatorhub.io/

Kubernetes Operators, Available at: https://kubernetes.io/docs/concepts/extend-kubernetes/operator/

Kubernetes.Io, 2018. Fission: Serverless Functions as a Service for. Available at: https://kubernetes.io/blog/2017/01/fission-serverless-functions-as-service-for-kubernetes/

Lack, S. and Dandeneau, J., n.d. An Astadia CloudGPS Whitepaper EXECUTIVE SUMMARY. Available at: https://uploads-ssl.webflow.com/5be08f6f96d344ce891bdbea/5c33a70bf87503b6a92edf25_7-Critical-Success-Factors-for-Cloud-Whitepaper-Astadia.pdf.

Lee, E., 2021. Function-as-a-Service (FaaS) Market Research Report Including SWOT Analysis, PESTEL Analysis, Drivers, Restraints, Global Industry Outlook and Key Players Analysis By 2027, NeighborWebSJ

Linthicum, D., 2020. 3 post-pandemic cloud architecture trends. InfoWorld. Available at: https://www.infoworld.com/article/3541355/3-post-pandemic-cloud-architecture-trends.html

M.S. de Brito, S. Hoque, R. Steinke, A. Willner, T. Magedanz, 2017 : Application of the Fog computing paradigm to Smart Factories and cyber-physical systems | Research Gate. Available at: https://www.researchgate.net/publication/317171634_Application_of_the_Fog_computing_paradigm_to_Smart_Factories_and_cyber-physical_systems

Marcus Williams, 2020 : BMW to use AWS cloud computing software to improve supply chain visibility | Automotive Logistics. Available at: https://www.automotivelogistics.media/news/bmw-to-use-aws-cloud-computing-software-to-improve-supply-chain-visibility/41392.article

Markets and Markets, 2021. Cloud Computing Market Size, Share and Global Market Forecast to 2025 | COVID-19 Impact Analysis | MarketsandMarkets. Marketsandmarkets.com. Available at: https://www.marketsandmarkets.com/Market-Reports/cloud-computing-market-234.html

Markets and Markets, 2021. Function-as-a-Service Market - Scope, Size, Share, Forecast Report 2025. Available at :https://www.tmrresearch.com/function-as-a-service-market

Marklein, T., 2019. Study: 55 Percent Say API Integration Is “Critical” to Business Strategy. GlobeNewswire Maximize Market Research, 2021. Maximizemarketresearch.com. Available at: https://www.maximizemarketresearch.com/market-report/global-function-as-a-service-market/63231/

McCaskill, S., 2020. Microsoft strengthens 5G cloud offering with Metaswitch Networks purchase. TechRadar. Available at: https://www.techradar.com/news/microsoft-strengthens-5g-cloud-offering-with-metaswitch-networks purchase#:%7E:text=Microsoft%20has%20strengthened%20the%205G,the%20acquisition%20of%20Metaswitch%20Networks.&text=5G%20promises%20ultrafast%20speeds%2C%20enhanced,business%2C%20consumer%20and%20government%20applications

Md Al Shayokh, Jin Woo Kim, Soo Young Shin, 2015 : Cloud Based Software Defined Wireless Body Area Networks Architecture for Virtual Hospital | EUDL European Union Digital Library] Available at https://eudl.eu/doi/10.4108/eai.28-9-2015.2261508

Miell, I., 2019. AWS Lambda's Major Limitations. IOD. https://iamondemand.com/blog/aws-lambdas-major-limitations/

Monocello, M., 2020. Pros and Cons of Function as a Service (FaaS). DevPro Journal. Available at: https://www.devprojournal.com/software-development-trends/pros-and-cons-of-function-as-a-service-FaaS/

Msv, J., 2020. An Architectural View of Apache OpenWhisk. Available at: https://thenewstack.io/behind-scenes-apache-openwhisk-serverless-platform/
New Relic., 2020. For the Love of Serverless. Available at: https://newrelic.com/resources/ebooks/serverless-benchmark-report-aws-lambda-2020

Nigam, R., 2020. Apache OpenWhisk (Serverless for Kubernetes) - FAUN. Available at: https://medium.com/faun/apache-openwhisk-serverless-for-kubernetes-820f62534f24

Olaph, W., 2019. When To Use Iaas, FaaS, Paas, And Caas. Ibm Developer. Available At: Https://Developer.Ibm.Com/Depmodels/Cloud/Articles/When-to-use-iaas-FaaS-paas-and-caas

OpenFaaS Introduction. 2021. Available at: https://docs.openFaaS.com/

Openwhisk.Apache.Org, 2020. Apache OpenWhisk is a serverless, open source cloud platform. Available at: https://openwhisk.apache.org

P., 2018. Fission Open Source Serverless Framework Offers Live-Reload, Canary Deployments, Record-Replay, More Features to Improve the Quality and Simplify Release Automation of Serverless Applications on Kubernetes. Available at: https://www.prnewswire.com/news-releases/fission-open-source-serverless-framework-offers-live-reload-canary-deployments-record-replay-more-features-to-improve-the-quality-and-simplify-release-automation-of-serverless-applications-on-kubernetes-300731242.html

Paakkunainen, O., 2021. What the FaaS? Cut down your computing costs with serverless computing.Available at: https://qvik.com/news/serverless-FaaS-computing-costs/

Panettieri, J., 2020. Cloud Market Share 2020 : Amazon AWS, Microsoft Azure, Google, IBM. ChannelE2E. Available at: https://www.channele2e.com/channel-partners/csps/cloud-market-share-2020-amazon-aws-microsoft-azure-google-ibm/

Panettieri, J., 2020. Cloud Market Share 2020 : Amazon AWS, Microsoft Azure, Google, IBM. ChannelE2E. Available at: https://www.channele2e.com/channel-partners/csps/cloud-market-share-2020-amazon-aws-microsoft-azure-google-ibm/

Pangkaj Chandra Paul, John Loane , Fergal McCaffery, and Gilbert Regan, 2019 : A Serverless Architecture for Wireless Body Area Network Applications | SpringerLink Available at: https://link.springer.com/chapter/10.1007/978-3-030-32872-6_16

ParamitaGhosh,2018. Serverless Computing and Serverless Architecture: An Overview of BaaS, FaaS, and PaaS. DATAVERSITY. Available at: https://www.dataversity.net/serverless-computing-serverless-architecture-overview/

Parker, B., 2020. Microsoft SWOT Analysis 2019 | SWOT Analysis of Microsoft. Business Strategy Hub. Available at: https://bstrategyhub.com/swot-analysis-of-microsoft/

Patrick Lanners & Julian Roedig, 2018 : Building BMW's new Unified Configurator Platform on AWS | AWS Summit Berlin 2018 | Youtube, Amazon Web Services Deutsch. Available at: https://www.youtube.com/watch?v=7wpqmJZV9uU

Pavlovic, U., 2020. Serverless Computing: How it Works, AWS vs. Azure vs GCP Comparison. Hentsu. Available at: https://hentsu.com/serverless-computing-aws-vs-azure-vs-gcp-cloud-comparison/

Petters, J., 2020. AWS vs Azure vs Google : Cloud Services Comparison - Varonis. Inside Out Security. Available at: https://www.varonis.com/blog/aws-vs-azure-vs-google/

Petters, J., 2020. AWS vs Azure vs Google : Cloud Services Comparison - Varonis. Inside Out Security. Available at: https://www.varonis.com/blog/aws-vs-azure-vs-google/

Philipp Leitner, Erik Wittern, Josef Spillner, and Waldemar Hummer 2019 A mixed-method empirical study of Function-as-a-Service software development in industrial practice. Journal of Systems and Software, 146:340–359,

Projexion., 2020. Multicloud : quels sont les facteurs clés de succès de sa mise œuvre ? Available at: https://www.projexion.com/carrefour-apprentissage/rex/des-avantages-aux-facteurs-cles-de-succes-comment-mettre-en-place-et-optimiser-son-approche-multicloud/

R., V., 2021. What are the advantage of using AWS Lambda. Edureka Community. Available at: https://www.edureka.co/community/35439/what-are-the-advantage-of-using-aws-lambda

Rehemägi, T., 2020. Serverless Platforms : Ultimate Competitor Matrix. Dashbird. Available at: https://dashbird.io/blog/ultimate-serverless-benchmark-2019/

Reports and Data, 2020 : Function As A Service FaaS Market | Reports and Data. Available at: https://www.reportsanddata.com/report-detail/function-as-a-service-FaaS-market

Roberts, M., 2018. Serverless Architectures. *MartinFlowers.com* Available at: https://martinfowler.com/articles/serverless.html.

Roy Illsley, R. I., 2020. SWOT Assessment: Microsoft Azure. Omdia. Available at: https://omdia.tech.informa.com/-/media/tech/omdia/assetfamily/2020/04/30/swot-assessment-microsoft-azure/swot-assessment-microsoft-azure-pdf.pdf

Sergiu Tudose, 2020 : BMW Expands Partnership With Amazon, Will Use AI To Predict Consumer Demand | Carscoops. Available at : https://www.carscoops.com/2020/12/bmw-expands-partnership-with-amazon-will-use-ai-to-predict-consumer-demand/

Shilkov, M., 2020. AWS Lambda vs. Azure Functions : 10 Major Differences. IOD. Available at: https://iamondemand.com/blog/aws-lambda-vs-azure-functions-ten-major-differences/

Shuper, A., n.d. Serverless Security: The Importance of FaaS Risk Assessment. Available at: blog.aquasec.com. Available at: https://blog.aquasec.com/serverless-security-FaaS-risk-assessment

Siapno, N., 2020. Everything You Need to Know About AWS Lambda - The Startup. Medium. Available at: https://medium.com/swlh/everything-you-need-to-know-about-aws-lambda-e0b214224b0f

Solanki, J., 2021. AWS Lambda Performance Tuning & Best Practices (2021). Insights on Latest Technologies - Simform Blog. Available at: https://www.simform.com/aws-lambda-performance/

Spark Consulting., 2020. Le serverless open source : plus de liberté. Available at: https://spark-consulting.fr/2020/02/25/serverless-open-source-plus-de-liberte/

Sroczkowski, P., N.D. Cloud: Iaas Vs Paas Vs Saas Vs Daas Vs FaaS Vs Dbaas | Brainhub. Brainhub. Available At: Https://Brainhub.Eu/Blog/Cloud-architecture-saas-FaaS-xaas

Stalcup, K., 2020. AWS vs Azure vs Google Cloud Market Share 2020 : What the Latest Data Shows. ParkMyCloud. Available at: https://www.parkmycloud.com/blog/aws-vs-azure-vs-google-cloud-market-share/

Stalcup, K., 2020. AWS vs Azure vs Google Cloud Market Share 2020 : What the Latest Data Shows. ParkMyCloud. Available at: https://www.parkmycloud.com/blog/aws-vs-azure-vs-google-cloud-market-share/

Sucharitha. V, Prakash. P , Ganesh Neelakanta Iyer, 2019 : AgriFog- A Fog Computing based IoT for Smart Agriculture | Research Gate. Available at https://www.researchgate.net/publication/332559113_Agrifog-a_fog_computing_based_IoT_for_smart_agriculture

Synergy Research Group. n.d. Quarterly Cloud Spending Blows Past $30B; Incremental Growth Continues to Rise | Synergy Research Group. Available at: https://www.srgresearch.com/articles/quarterly-cloud-spending-blows-past-30b-incremental-growth-continues-rise.

Tannahill, B., 2018. What Is OpenFaaS and How Can It Drive Innovation? An Interview with Creator Alex Ellis. Available at: https://www.contino.io/insights/what-is-openFaaS-and-why-is-it-an-alternative-to-aws-lambda-an-interview-with-creator-alex-ellis

TechRepublic Staff., 2020. Microsoft Azure : A cheat sheet. TechRepublic. Available at: https://www.techrepublic.com/article/microsoft-azure-the-smart-persons-guide/

The European Cloud Initiative - Shaping Europe's digital future - European Commission., 2021.

Titienok, S., 2020. Lessons Learned from Building a Low-Latency System Based on AWS Lambda. Available at: https://www.lohika.com/aws-lambda-lessons-learned

Trefis Team., 2016. Here's Why Google Is Acquiring Apigee. Forbes. Available at: https://www.forbes.com/sites/greatspeculations/2016/09/13/heres-why-google-is-acquiring-apigee/?sh=7b3cec203ed5

Usine-digitale.fr. (n.d.). Les dix facteurs clés qui vont façonner le paysage du cloud en 2017. Available at: https://www.usine-digitale.fr/article/les-dix-facteurs-cles-qui-vont-faconner-le-paysage-du-cloud-en-2017.N461397

Van Eyk, E., Grohmann, J., Eismann, S., Bauer, A., Versluis, L., Toader, L., Schmitt, N., Herbst, N., Abad, C. and Iosup, A., 2019. The SPEC-RG Reference Architecture for FaaS: From Microservices and Containers to Serverless Platforms. IEEE Internet Computing.

Vector ITC., 2019. A 55% companies use APIs as a revenue stream. Vector ITC Group. Available at: https://www.vectoritcgroup.com/en/tech-magazine-en/software-trends-en/55-of-companies-use-apis-as-revenue-streams/

Verma, G., Adhikari, S., 2020. Cloud Computing Security Issues: a Stakeholder's Perspective.

Yuan, M., 2020. Blockchain is the Ultimate Serverless Platform. Available at: https://www.informit.com/articles/article.aspx?p=3006828#