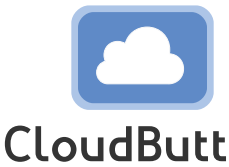

HORIZON 2020 FRAMEWORK PROGRAMME

CloudButton

(grant agreement No 825184)

Serverless Data Analytics Platform

D6.2 Communication Report

Due date of deliverable: 30-06-2020
Actual submission date: 04-08-2020

Start date of project: 01-01-2019 Duration: 36 months

Summary of the document

Document Type	Report
Dissemination level	Public
State	v1.0
Number of pages	13
WP/Task related to this document	WP6 / T6.1, T6.2
WP/Task responsible	RHAT
Leader	Tristan Tarrant (RHAT)
Technical Manager	Marc Sánchez (URV)
Quality Manager	Gerard París (URV)
Author(s)	Tristan Tarrant (RHAT), Gerard París (URV), Lara López (ATOS), Gil Vernik (IBM).
Partner(s) Contributing	RHAT, URV, ATOS, ANSWARE, IBM.
Document ID	CloudButton_D6.2_Public.pdf
Abstract	This document summarizes the communication and dissemination activities undertaken by the CloudButton project partners between July 2019 and June 2020.
Keywords	Communication, Dissemination, Conferences, Papers

History of changes

Version	Date	Author	Summary of changes
0.1	20-06-2020	Tristan Tarrant (RHAT)	First version.
0.2	22-07-2020	Gerard París (URV)	Reviewed. Added photos of dissemination events.
1.0	04-08-2020	Tristan Tarrant (RHAT)	Final version.

Table of Contents

List of Abbreviations and Acronyms

ACM	Association for Computing Machinery
ASMS	American Society for Mass Spectrometry
BDV	Big Data Value
EBDVF	European Big Data Value Forum
EMBL	European Molecular Biology Laboratory
EMBO	European Molecular Biology Organization
FaaS	Function as a Service
LiDAR	Light detection and ranging
MS	Mass Spectrometry
NDVI	Normalized difference vegetation index
RHAT	Red Hat
SaaS	Software as a Service
SLA	Service-level agreement
SME	Small and medium-sized enterprises
TPI	The Pirbright Institute
URV	Universitat Rovira i Virgili
WP	Work Package

1 Executive summary

This document summarizes the communication and dissemination activities undertaken by the CloudButton project partners during the second and third semesters of the project, between July 2019 and June 2020. It also includes initial exploitation plans by industrial partners.

2 Second-year communication activities

The CloudButton partners have been active on several communication fronts over the last year, promoting the project scope and results in different ways. The following is a list of activities that have taken place since the previous report:

2.1 Online presence/Social Media/Press releases

- Post about the project in the Answare web page: `https://www.answare-tech.com/en/id-2/`
- Dissemination of CloudButton and collaboration activities on Twitter (`https://twitter.com/metaspace2020`)
- 06/02/2019 Project presentation post on The Matrix Foundation Website: `https://bit.ly/2XupqYt`
- Added the CloudButton logo and link to the Infinispan homepage `https://infinispan.org`
- Press-release for PLANETIC (Spanish platform to foster the adoption of ICT) media channels (newsletter and website). Elaboration of a press release to advertise the project and first results within the Spanish R&D community using PLANETIC (Spanish platform for ICT promotion and adoption) media.
- Infinispan's new Container Image `https://infinispan.org/blog/2019/12/02/image/`

  Infinispan 10 introduced a new server, which does not utilise the same launch commands and configuration as the legacy 9.4 wildfly based server. Therefore, we decided that this was an excellent opportunity to rewrite our container image from scratch to better suite the capabilities of the new server and to provide all the functionality required by the Infinispan Operator. This post focuses on the server image's architecture. Future blog posts will focus on more advanced configurations, as well as example usage and deployment scenarios such as deploying a cluster using Kubernetes StatefulSets.
- Infinispan Operator 1.1.0 with Full Lifecycle support: `https://infinispan.org/blog/2020/01/24/infinispan-operator-1/`

  Infinispan Operator 1.1.0 is rated at the Full Lifecycle capacity level, which means the Operator now provides advanced cluster management capabilities and functionality to handle demanding workloads. One of the key new features in this release is graceful shutdown, which lets you bring clusters down safely to avoid data loss.
- Infinispan Operator improvements `https://infinispan.org/blog/2020/06/08/infinispan-operator-1/`
- Anchored keys - scaling up a cluster without transferring values `https://infinispan.org/blog/2020/07/22/anchored-keys/`

  Anchored keys - scaling up a cluster without transferring values For background, the preferred way to scale up the storage capacity of a Infinispan cluster is to use distributed caches. A distributed cache stores each key/value pair on num-owners nodes, and each node can compute the location of a key (aka the key owners) directly. Infinispan achieves this by statically mapping cache keys to num-segments consistent hash segments, and then dynamically mapping segments to nodes based on the cache's topology (roughly the current plus the historical membership of the cache). Whenever a new node joins the cluster, the cache is rebalanced, and the new node replaces an existing node as the owner of some segments. The key/value pairs in those segments are copied to the new node and removed from the no-longer-owner node via state transfer. The basic idea is to skip the static mapping of keys to segments and to map keys directly to nodes. When a key/value pair is inserted into the cache, the newest member becomes the anchor owner of that key, and the only node storing the actual value. In order to

make the anchor location available without an extra remote lookup, all the other nodes store a reference to the anchor owner. That way, when another node joins, it only needs to receive the location information from the existing nodes, and values can stay on the anchor owner, minimizing the amount of traffic.

- Infinispan Native Server Image `https://infinispan.org/blog/2020/06/16/native-server-announce/`
- September 19: P. Sutra, CloudButton: le Big Data à portée de clic. I'MTech, online blog. `https://blogrecherche.wp.imt.fr/2019/10/10/cloudbutton-big-data-a-portee-de-clic`

Figure 1: CloudButton at BigMedilytics Event, Valencia, Spain, September 2019.

Figure 2: CloudButton at European Big Data Value Forum, Helsinki, Finland, October 2019.

2.2 Conferences, Meetings and Workshops

- P. Sutra, On the correctness of Egalitarian Paxos, Workshop on Verification (w. NETYS'19), invited talk, July 2019. `http://goto.ucsd.edu/~gleissen/vds-test`

  In this talk, we present a problem in both the TLA + specification and the implementation of the Egalitarian Paxos protocol. This problem is related to how replicas switch from one ballot to another when computing the dependencies of a command. We show that it may lead replicas to diverge and break the linearizability of the replicated service.

- Pedro Garcia Lopez, "CloudButton: Serverless Data Analytics". CloudButton presentation in IBM Watson Research. Open by videoconference to all IBM locations. 11 July 2019.

- BigMedilytics event (Valencia, Spain). Josep Sampé presented CloudButton project in the event "Big Data: Fueling the transformation of Europe's Healthcare Sector". 5 September 2019.

- EMBO Workshop Lipid Function in Health and Disease, Sep 2019, Dresden, DE

  Spatial metabolomics and lipidomics in tissues and single cells

- EMBL Partnership Conference Perspectives in Translational Medicine, Sep 2019, Barcelona, ES

- German Pharmaceutical Society Symposium, Sep 2019, Heidelberg, DE (keynote)

- M4I Workshop on Mass Spectrometry Imaging, Sep 2019, Maastricht, NL

- Talk at Strata Data Conference – "Your easy move to serverless computing and radically simplified data processing". `https://conferences.oreilly.com/strata/strata-ny/public/schedule/detail/77226` Strata NY, September 2019

  Abstract Suppose you wrote Python code for Monte Carlo simulations to analyze financial data. The general process involves writing the code and running a simulation over small set of data to test it. Assuming this all goes smoothly, you now must run the same code at a massive scale, with parallelism, on terabytes of data, doing millions of Monte Carlo simulations. Clearly

you'd prefer not to need to learn the intricacies of setting up virtual machines, suffer long setup times for the virtual machines, nor become an expert in scaling up Python code. This is exactly where serverless computing could come to the rescue. With serverless computing, you don't need to set up the computing environment and only pay for the actual amount of resources your application consumes rather than prepurchased units of capacity. Here you'll learn how to easily gain these benefits. Gil Vernik takes a deep dive into the challenge of how serverless computing can be easily used for a broad range of scenarios, like high-performance computing (HPC), Monte Carlo simulations, and data preprocessing for AI. You'll focus on how to connect existing code and frameworks to serverless without the painful process of starting from scratch and or learning new skills. To achieve this, you're based on the open source PyWren framework that introduces serverless computing with minimal effort, and its new fusion with serverless computing brings automated scalability and the use of existing frameworks into the picture. You can simply write a Python function and provide an input pointing to the dataset in a storage bucket. Then PyWren does the magic by automatically scaling and executing the user function as a serverless action at massive scale. Gil demonstrates how this capability allowed IBM to run broad range of scenarios over serverless, including Monte Carlo simulations to predict future stock prices and hyperparameter optimizations for ML models. IBM managed to complete the entire Monte Carlo simulation for stock price prediction in about 90 seconds with 1,000 concurrent invocations, compared to 247 minutes with almost 100% CPU utilization running the same flow over a laptop with 4 CPU cores. He'll also show you how to bond TensorFlow and serverless for the data-preparation phases. Existing TensorFlow code can be easily adapted and benefit serverless with only minimal code modifications and without users having to learn serverless architectures and deployments.

- ASMS Asilomar Conference on Mass Spectrometry, Oct 2019, Pacific Grove, CA, USA

  Spatial Metabolomics: From Big Data to Single Cells,
- Munich Metabolomics Meeting, Oct 2019, Munich, DE (keynote)
- Bayer MS Imaging Workshop, Oct 2019, Berlin, DE
- OurCon conference on Imaging Mass Spectrometry, Oct 2019, St. Malo, France
- Talk at EBDVF in Helsinki, "Your easy move to serverless computing and radically simplified data processing", `https://www.european-big-data-value-forum.eu/program-day-2/` European Big Data Value Forum, October 2019
- Talk at Haifa tech meetup – "Your easy move to serverless computing". `https://www.meetup.com/Haifa-Tech-Talks/events/261393733/`
- Presentation to Intel Labs of the current status of the Faasm runtime, October 2019.
- Pedro Garcia Lopez, Talk: "The next revolution in Distributed Computing". IBM Watson Research. 21 January 2020.

Figure 3: Daniel Barcelona's talk at the ACM/IFIP International Middleware Conference 2019, UC Davis, California.

2.3 Conference papers, journal articles and preprints

- Raúl Gracia Tinedo, Marc Sánchez Artigas, Pedro García López, Yosef Moatti, Filip Gluszak: Lambda-Flow: Automatic Pushdown of Dataflow Operators Close to the Data. 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID 2019), Larnaca, Cyprus, May 14-17, 2019. pp. 112-121.

  Abstract Modern data analytics infrastructures are composed of physically disaggregated compute and storage clusters. Thus, dataflow analytics engines, such as Apache Spark or Flink, are left with no choice but to transfer datasets to the compute cluster prior to their actual processing. For large data volumes, this becomes problematic, since it involves massive data transfers that exhaust network bandwidth, that waste compute cluster memory, and that may become a performance barrier. To overcome this problem, we present λFlow: a framework for automatically pushing dataflow operators (e.g., map, flatMap, filter, etc.) down onto the storage layer. The novelty of λFlow is that it manages the pushdown granularity at the operator level, which makes it a unique problem. To wit, it requires addressing several challenges, such as how to encapsulate dataflow operators and execute them on the storage cluster, and how to keep track of dependencies such that operators can be pushed down safely onto the storage layer. Our evaluation reports significant reductions in resource usage for a large variety of IO-bound jobs. For instance, λFlow was able to reduce both network bandwidth and memory requirements by 90% in Spark. Our Flink experiments also prove the extensibility of λFlow to other engines.

- Pedro García López, Alberto Montresor, Anwitaman Datta: Please, do not Decentralize the Internet with (Permissionless) Blockchains!. 39th IEEE International Conference on Distributed Computing Systems, ICDCS 2019, Dallas, TX, USA, July 7-10, 2019. pp. 1901-1911.

  Abstract The old mantra of decentralizing the Internet is coming again with fanfare, this time around the blockchain technology hype. We have already seen a technology supposed to change the nature of the Internet: peer-to-peer. The reality is that peer-to-peer naming systems failed, peer-to-peer social networks failed, and yes, peer-to-peer storage failed as well. In this paper, we will review the research on distributed systems in the last few years to identify the

limits of open peer-to-peer networks. We will address issues like system complexity, security and frailty, instability and performance. We will show how many of the aforementioned problems also apply to the recent breed of permissionless blockchain networks. The applicability of such systems to mature industrial applications is undermined by the same properties that make them so interesting for a libertarian audience: namely, their openness, their pseudo-anonymity and their unregulated cryptocurrencies. As such, we argue that permissionless blockchain networks are unsuitable to be the substrate for a decentralized Internet. Yet, there is still hope for more decentralization, albeit in a form somewhat limited with respect to the libertarian view of decentralized Internet: in cooperation rather than in competition with the superpowerful datacenters that dominate the world today. This is derived from the recent surge in interest in byzantine fault tolerance and permissioned blockchains, which opens the door to a world where use of trusted third parties is not the only way to arbitrate an ensemble of entities. The ability of establish trust through permissioned blockchains enables to move the control from the datacenters to the edge, truly realizing the promises of edge-centric computing.

- Pedro García López, Marc Sánchez Artigas, Simon Shillaker, Peter R. Pietzuch, David Breitgand, Gil Vernik, Pierre Sutra, Tristan Tarrant, Ana Juan Ferrer: ServerMix: Tradeoffs and Challenges of Serverless Data Analytics. arXiv CoRR abs/1907.11465 (July 2019).

  Abstract Serverless computing has become very popular today since it largely simplifies cloud programming. Developers do not need to longer worry about provisioning or operating servers, and they pay only for the compute resources used when their code is run. This new cloud paradigm suits well for many applications, and researchers have already begun investigating the feasibility of serverless computing for data analytics. Unfortunately, today's serverless computing presents important limitations that make it really difficult to support all sorts of analytics workloads. This paper first starts by analyzing three fundamental trade-offs of today's serverless computing model and their relationship with data analytics. It studies how by relaxing disaggregation, isolation, and simple scheduling, it is possible to increase the overall computing performance, but at the expense of essential aspects of the model such as elasticity, security, or sub-second activations, respectively. The consequence of these trade-offs is that analytics applications may well end up embracing hybrid systems composed of serverless and serverful components, which we call Servermix in this paper. We will review the existing related work to show that most applications can be actually categorized as Servermix. Finally, this paper will introduce the major challenges of the CloudButton research project to manage these trade-offs.

- Katja Ovchinnikova, Alexander Rakhlin, Lachlan Stuart, Sergey Nikolenko, Theodore Alexandrov, ColocAI: artificial intelligence approach to quantify co-localization between mass spectrometry images, BioRxiv, `https://www.biorxiv.org/content/10.1101/758425v1`, September 2019.

  Abstract Motivation Imaging mass spectrometry (imaging MS) is a prominent technique for capturing distributions of molecules in tissue sections. Various computational methods for imaging MS rely on quantifying spatial correlations between ion images, referred to as co-localization. However, no comprehensive evaluation of co-localization measures has ever been performed; this leads to arbitrary choices and hinders method development.

  Results We present ColocAI, an artificial intelligence approach addressing this gap. With the help of 42 imaging MS experts from 9 labs, we created a gold standard of 2210 pairs of ion images ranked by their co-localization. We evaluated existing co-localization measures and developed novel measures using tf-idf and deep neural networks. The semi-supervised deep learning Pi model and the cosine score applied after median thresholding performed the best (Spearman 0.797 and 0.794 with expert rankings respectively). We illustrate these measures by inferring co-localization properties of 10273 molecules from 3685 public METASPACE datasets.

- Alexey Gotsman, Anatole Lefort, Gregory V. Chockler. White-Box Atomic Multicast. 49th An-

nual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2019, Portland, OR, USA, June 24-27, 2019,. pp. 176-87.

Abstract Atomic multicast is a communication primitive that delivers messages to multiple groups of processes according to some total order, with each group receiving the projection of the total order onto messages addressed to it. To be scalable, atomic multicast needs to be genuine, meaning that only the destination processes of a message should participate in ordering it. In this paper we propose a novel genuine atomic multicast protocol that in the absence of failures takes as low as 3 message delays to deliver a message when no other messages are multicast concurrently to its destination groups, and 5 message delays in the presence of concurrency. This improves the latencies of both the fault-tolerant version of classical Skeen's multicast protocol (6 or 12 message delays, depending on concurrency) and its recent improvement by Coelho et al. (4 or 8 message delays). To achieve such low latencies, we depart from the typical way of guaranteeing fault-tolerance by replicating each group with Paxos. Instead, we weave Paxos and Skeen's protocol together into a single coherent protocol, exploiting opportunities for white-box optimisations. We experimentally demonstrate that the superior theoretical characteristics of our protocol are reflected in practical performance pay-offs.

- Daniel Barcelona Pons, Marc Sánchez Artigas, Gerard París, Pierre Sutra, Pedro García López: On the FaaS Track: Building Stateful Distributed Applications with Serverless Architectures. 20th International Middleware Conference, Middleware 2019, Davis, CA, USA, December 9-13, 2019. pp. 41-54.

  Abstract Serverless computing is an emerging paradigm that greatly simplifies the usage of cloud resources and suits well to many tasks. Most notably, Function-as-a-Service (FaaS) enables programmers to develop cloud applications as individual functions that can run and scale independently. Yet, due to the disaggregation of storage and compute resources in FaaS, applications that require fine-grained support for mutable state and synchronization, such as machine learning and scientific computing, are hard to build. In this work, we present Crucial, a system to program highly-concurrent stateful applications with serverless architectures. Its programming model keeps the simplicity of FaaS and allows to port effortlessly multi-threaded algorithms to this new environment. Crucial is built upon the key insight that FaaS resembles to concurrent programming at the scale of a data center. As a consequence, a distributed shared memory layer is the right answer to the need for fine-grained state management and coordination in serverless. We validate our system with the help of micro-benchmarks and various applications. In particular, we implement two common machine learning algorithms: k-means clustering and logistic regression. For both cases, Crucial obtains superior or comparable performance to an equivalent Spark cluster.

- Daniel Barcelona Pons, Pedro García López, Álvaro Ruiz Ollobarren, Amanda Gómez-Gómez, Gerard París, Marc Sánchez Artigas: FaaS Orchestration of Parallel Workloads. 5th International Workshop on Serverless Computing, WOSC@Middleware 2019, Davis, CA, USA, December 09-13, 2019. pp. 25-30.

  Abstract Function as a Service (FaaS) is based on a reactive programming model where functions are activated by triggers in response to cloud events (e.g., objects added to an object store). The inherent elasticity and the pay-per-use model of serverless functions make them very appropriate for embarrassingly parallel tasks like data preprocessing, or even the execution of MapReduce jobs in the cloud.

  But current Serverless orchestration systems are not designed for managing parallel fork-join workflows in a scalable and efficient way. We demonstrate in this paper that existing services like AWS Step Functions or Azure Durable Functions incur in considerable overheads, and only Composer at IBM Cloud provides suitable performance.

  Successively, we analyze the architecture of OpenWhisk as an open-source FaaS systems and its

orchestration features (Composer). We outline its architecture problems and propose guidelines for orchestrating massively parallel workloads using serverless functions.

- Vitor Enes, Carlos Baquero, Tuanir França Rezende, Alexey Gotsman, Matthieu Perrin, Pierre Sutra: State-Machine Replication for Planet-Scale Systems. EuroSys'20: Fifteenth EuroSys Conference 2020, Heraklion, Greece, April 27-30, 2020. pp 24:1-24:15.

  Abstract Online applications now routinely replicate their data at multiple sites around the world. In this paper we present Atlas, the first state-machine replication protocol tailored for such planet-scale systems. Atlas does not rely on a distinguished leader, so clients enjoy the same quality of service independently of their geographical locations. Furthermore, client-perceived latency improves as we add sites closer to clients. To achieve this, Atlas minimizes the size of its quorums using an observation that concurrent data center failures are rare. It also processes a high percentage of accesses in a single round trip, even when these conflict. We experimentally demonstrate that Atlas consistently outperforms state-of-the-art protocols in planet-scale scenarios. In particular, Atlas is up to two times faster than Flexible Paxos with identical failure assumptions, and more than doubles the performance of Egalitarian Paxos in the YCSB benchmark.

- P. Sutra: On the Correctness of Egalitarian Paxos, Information Processing Letters, Vol. 156, April 2020.

  Abstract This paper identifies a problem in both the TLA+ specification and the implementation of the Egalitarian Paxos protocol. It is related to how replicas switch from one ballot to another when computing the dependencies of a command. The problem may lead replicas to diverge and break the linearizability of the replicated service.

- Pedro García López, Aleksander Slominski, Simon Shillaker, Michael Behrendt, Bernard Metzler: Serverless End Game: Disaggregation enabling Transparency. arXiv CoRR abs/2006.01251 (June 2020).

  Abstract For many years, the distributed systems community has struggled to smooth the transition from local to remote computing. Transparency means concealing the complexities of distributed programming like remote locations, failures or scaling. For us, full transparency implies that we can compile, debug and run unmodified single-machine code over effectively unlimited compute, storage, and memory resources. We elaborate in this article why resource disaggregation in serverless computing is the definitive catalyst to enable full transparency in the Cloud. We demonstrate with two experiments that we can achieve transparency today over disaggregated serverless resources and obtain comparable performance to local executions. We also show that locality cannot be neglected for many problems and we present five open research challenges: granular middleware and locality, memory disaggregation, virtualization, elastic programming models, and optimized deployment. If full transparency is possible, who needs explicit use of middleware if you can treat remote entities as local ones? Can we close the curtains of distributed systems complexity for the majority of users?

- Pedro García López, Aitor Arjona, Josep Sampé, Aleksander Slominski, Lionel Villard: Triggerflow: Trigger-based orchestration of Serverless Workflows. DEBS '20: The 14th ACM International Conference on Distributed and Event-based Systems, Montreal, Quebec, Canada, July 13-17, 2020. pp. 3-14.

  Abstract As more applications are being moved to the Cloud thanks to serverless computing, it is increasingly necessary to support native life cycle execution of those applications in the data center. But existing systems either focus on short-running workflows (like IBM Composer or Amazon Express Workflows) or impose considerable overheads for synchronizing massively parallel jobs (Azure Durable Functions, Amazon Step Functions, Google Cloud Composer). None of them are open systems enabling extensible interception and optimization of custom

workflows. We present Triggerflow: an extensible Trigger-based Orchestration architecture for serverless workflows built on top of Knative Eventing and Kubernetes technologies. We demonstrate that Triggerflow is a novel serverless building block capable of constructing different reactive schedulers (State Machines, Directed Acyclic Graphs, Workflow as code). We also validate that it can support high-volume event processing workloads, auto-scale on demand and transparently optimize scientific workflows.

- Simon Shillaker, Peter Pietzuch: Faasm: Lightweight Isolation for Efficient Stateful Serverless Computing, USENIX Annual Technical Conference (USENIX ATC), 15-17 July 2020, Boston, MA, USA.

  Abstract Serverless computing is an excellent fit for big data processing because it can scale quickly and cheaply to thousands of parallel functions. Existing serverless platforms isolate functions in ephemeral, stateless containers, preventing them from directly sharing memory. This forces users to duplicate and serialise data repeatedly, adding unnecessary performance and resource costs. We believe that a new lightweight isolation approach is needed, which supports sharing memory directly between functions and reduces resource overheads. We introduce Faaslets, a new isolation abstraction for high-performance serverless computing. Faaslets isolate the memory of executed functions using software-fault isolation (SFI), as provided by WebAssembly, while allowing memory regions to be shared between functions in the same address space. Faaslets can thus avoid expensive data movement when functions are co-located on the same machine. Our runtime for Faaslets, Faasm, isolates other resources, e.g. CPU and network, using standard Linux cgroups, and provides a low-level POSIX host interface for networking, file system access and dynamic loading. To reduce initialisation times, Faasm restores Faaslets from already-initialised snapshots. We compare Faasm to a standard container-based platform and show that, when training a machine learning model, it achieves a 2x speed-up with 10x less memory; for serving machine learning inference, Faasm doubles the throughput and reduces tail latency by 90%.

- Gerard París, Pedro García López, Marc Sánchez-Artigas. Serverless Elastic Exploration of Unbalanced Algorithms, 13th IEEE International Conference on Cloud Computing, CLOUD 2020, October, 2020.

  Abstract In recent years, serverless computing and, in particular the Function-as-a-Service (Faas) execution model, has proven to be efficient for running parallel computing tasks. However, little attention has been paid to highly-parallel applications with unbalanced and irregular workloads. The main challenge of executing this type of algorithms in the cloud is the difficulty to account for the computing requirements beforehand. This places a burden on scientific users who very often make bad decisions by either overprovisioning resources or inadvertently limiting the parallelism of these algorithms due to resource contention. Our hypothesis is that the elasticity and ease of management of serverless computing can help users avoid such decisions, which may lead to undesirable cost-performance consequences for unbalanced problem spaces. In this work, we show that with a simple serverless executor pool abstraction, which can combine local resources with remote cloud functions, one can achieve a better cost-performance trade-off than a Spark cluster of static size and large EC2 VMs. To support this conclusion, we evaluate two unbalanced algorithms: the Unbalanced Tree Search (UTS) and the Mandelbrot Set using the Mariani-Silver algorithm. For instance, our serverless implementation of UTS is able to outperform Spark by up to 55% with the same cost. This provides the first concrete evidence that highly-parallel, irregular workloads can be efficiently executed using purely stateless functions with almost zero burden on users — i.e., no need for users to understand non-obvious system-level parameters and optimizations.

- Tuanir França Rezende, Pierre Sutra. Leaderless State-Machine Replication: Specification, Properties, Limits, 34th International Symposium on Distributed Computing, DISC 2020, October, 2020.

Abstract Modern Internet services commonly replicate critical data across several geographical locations using state-machine replication (SMR). Due to their reliance on a leader replica, classical SMR protocols offer limited scalability and availability in this setting. To solve this problem, recent protocols follow instead a leaderless approach, in which each replica is able to make progress using a quorum of its peers. In this paper, we study this new emerging class of SMR protocols and states some of their limits. We first propose a framework and compelling set of definitions that capture the essence of leaderless state-machine replication (Leaderless SMR). Then, we introduce a set of desirable properties for these protocols: (R)eliability, (O)ptimal (L)atency and (L)oad Balancing. We show that protocols matching all of the ROLL properties are subject to a trade-off between performance and reliability. We also establish a lower bound on the message delay to execute a command in protocols optimal for the ROLL properties. This lower bound explains the persistent chaining effect observed in experimental results.

3 Initial exploitation plans

While the project consortium has still been mostly focused on the research and development aspect, the industrial partners have begun exploring the potential avenues for exploitation of the project results in the context of their specific businesses. These initial plans will be further expanded and refined over the course of the second half of the project.

3.1 Red Hat

Red Hat's product strategy is to build technology collaboratively either by initiating or by participating in existing open source communities. These communities drive innovation at a very fast pace, building on the continuous feedback of their users and experimenting with new ideas and implementations. Based on internal assessments, a reduced set of the community driven technologies are integrated into battle tested, mature and multi-year supported products. These products are what ends-up in the hands of Red Hat customers.

The improvements made to Infinispan in the context of the CloudButton project will have a number of very significant benefits for both the current offerings as well as adoption in new areas. Red Hat currently includes Infinispan in a number of products and services: Red Hat Data Grid (RHDG), Red Hat Enterprise Application Platform (EAP), Red Hat Single Sign On (RHSSO), and more. Several CloudButton-driven technologies and improvements are being integrated into RHDG. This integration has been done at a faster than usual pace due to their usefulness and their alignment to market needs.

Highlights of the work done within the CloudButton efforts:

- the work done on the Infinispan Kubernetes Operator, including service provisioning, autoscaling and metrics will make RHDG much more attractive to customers in terms of manageability and monitoring
- the reduced memory footprint and startup time will allow for improved performance and higher density deployments, especially in the context of Red Hat's Serverless product, where RHDG will be deployed to offer high-performance in-memory storage alongside other, more traditional persistent storage.
- the new memory scaling algorithm, currently named *Anchored Keys* upstream, will be particularly useful in the context of Red Hat OpenShift Serverless Eventing and Serving.

We can already claim that Infinispan's participation in the project has been a booster for RHDG's usefulness to Red Hat customers. Red Hat will continue its efforts to push these technologies the communities and, if proven useful, push them up to its product line.

3.2 Atos

ATOS, as a large industry, is more focused on business impact. ATOS foresees a huge potential on SLA commercialization, based on dynamic information and infrastructure monitoring for data intensive applications, as it is a real market need. The SLA manager is currently considered as an asset to be included in Cloud/Edge offerings. Thanks to the work developed within CloudButton, it is considered as a complementarity for the Connected Intelligence offering within the Life Sciences portfolio.

3.3 IBM

IBM plans to integrate CloudButton Toolkit as an API for IBM Code Engine service. In addition, IBM plans to explore integration of CloudButton Toolkit with Red Hat OpenShift offerings.

3.4 Answare

Answare, an SME, is exploring ways to sell NDVI calculation as a service (SaaS model), highlighting the benefits of serverless technologies, in particular the increase of productivity achieved through parallelization of processes.