SPARC

A SHARING ECONOMY OF COMPUTE RESOURCES

Corresponding author

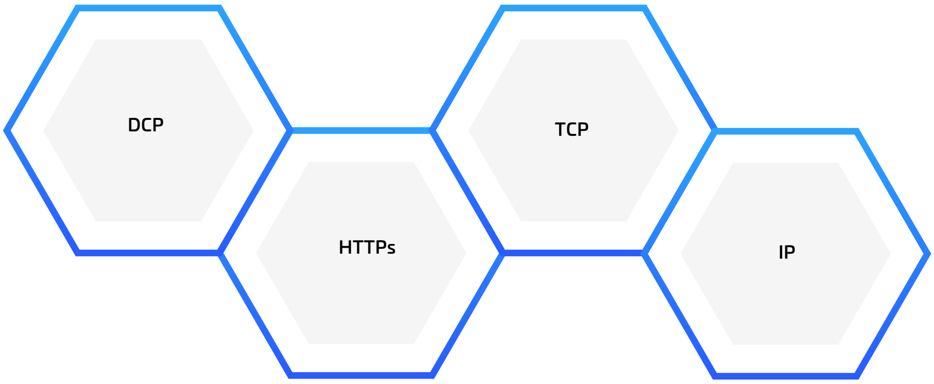
ddesjardins@sparc.network

Updated 2017/11/11

Contents

Executive summary

SPARC aims to become the de facto global standard for web-based distributed computing by seamlessly integrating its Distributed Compute Protocol (DCP) with the internet protocol suite. Specifically, SPARC will introduce the DCP dynamic layer on top of the HTTPS/TCP/IP suite effectively transforming computational power into a public utility. In doing so, the SPARC Foundation provides a specification for a sharing economy of compute resources. The foundation's mission is to accelerate compute-enabled research and innovation by providing developers, scientists and engineers with access to otherwise idle compute power through a collaborative compute network.



SPARC's web-based network protocol enables devices to perform computational work in exchange for SPARC tokens.

As computational problems spiral upwards in complexity and new high-tech fields emerge, developers are seeking new tools and methods to employ.

SPARC's network protocol will enable any connected device to contribute and consume computing resources from a compute grid, effectively transforming compute power into a public utility analogous to water and electricity.
The grid will distribute compute resources for client projects on a network using **SPARC's Compute Resource Allocation algorithm (CRAa)**. SPARC tokens are earned by completing computational packages distributed by the protocol. Tokens can be exchanged for computational work on the network or may be listed on cryptocurrency exchanges.

Key features of the SPARC initiative:

- JSON/Web Assembly based compute network;
- Client ready packages;
- Integration with Node Package Manager;
- Complementary tools and Software Development Kits;

The Future of Cloud Computing and SPARC's Vision

Computing revolution – Migrating to the cloud

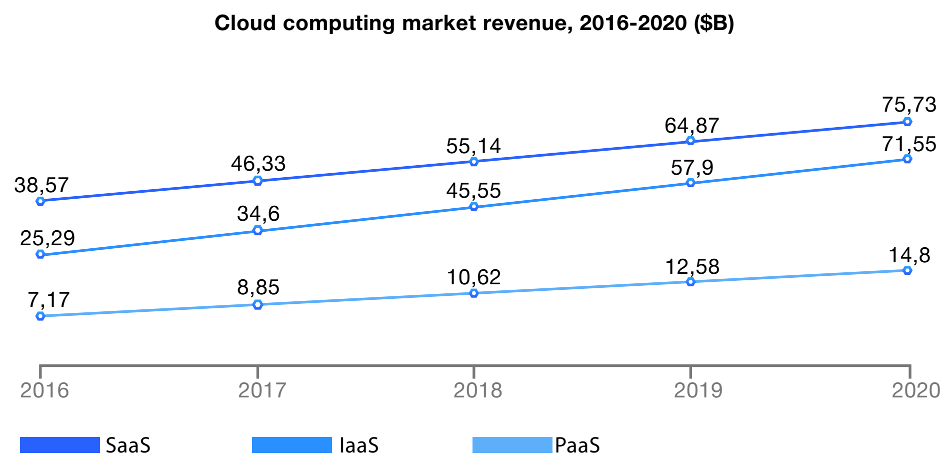
In this era of exponential data growth, industry and academia are realizing that they must achieve a fundamental transformation in how operations are run, and innovation is achieved. This transformation will require a change in thinking and an open-mindedness that may not have been present before. The commoditization of compute power will inevitably result from the convergence of digital currencies with ubiquitous wireless communication, machine learning, and embedded systems. Already, cloud computing has become a highly demanded service and is projected to grow exponentially.

Business and Institutions are gradually divesting their data centers and moving application workloads to the cloud. According to the CSA survey report, in 2016, 60.9% of applications workloads were still in enterprise data centers [1]. By the end of 2017, however, fewer than half (46.2%) will remain there. This is, in part, due to new applications primarily being deployed in the cloud, and because enterprises plan to migrate 20.7% of their existing applications to the public cloud [1]. SPARC is positioned to facilitate this transition.

Predicted growth of cloud computing market

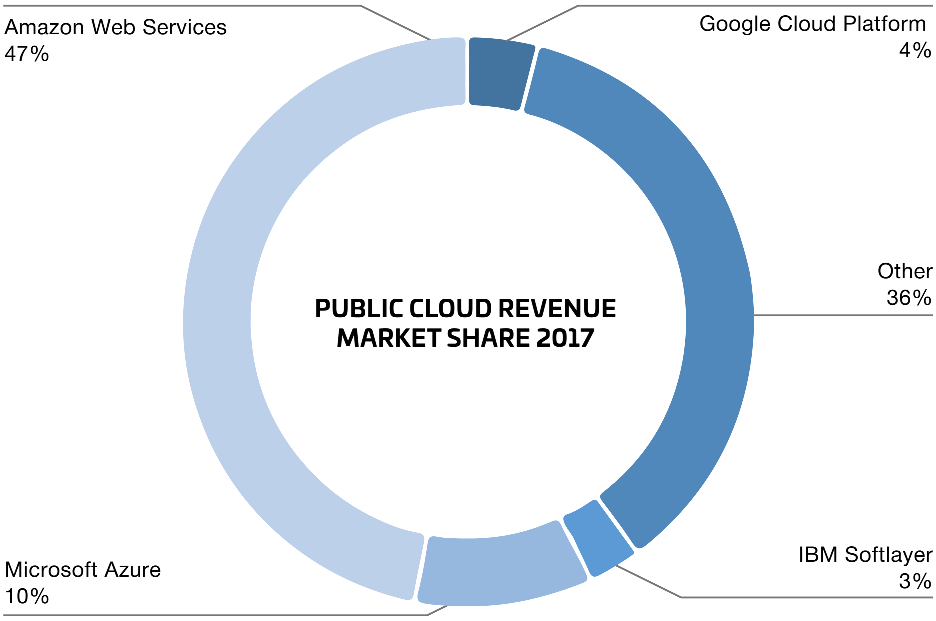
Cloud computing, which includes Software-as-a-Service (SaaS), Infrastructure-as-a-Service (IaaS), and Platform-as-a-Service (PaaS), is projected to increase from $67B in 2015 to $162B in 2020 attaining a Compounded Annual Growth Rate (CAGR) of 19% as shown in Figure 1 [2][3]. Notably, the International Data Corporation (IDC) forecasts that the High-Performance Computing (HPC) sector will experience a CAGR of 8% bringing total market to USD $31.4B by 2019 32[4].

High-Performance Computing (HPC) sector will experience a CAGR of 8% bringing total market to USD $31.4B by 2019 32[4].



[Figure 1]: Overview of Cloud Market in 2017 and Beyond. Source: Gartner 2017.

Centralized firms offering IaaS, examples listed in [Figure 2], are emerging in response to this growing global demand for cloud-based computing services.



[Figure 2]: Public cloud revenue market share 2017. Source: Gartner 2017.

> *Growing need for computational power and the urgent need for cloud computing*

Examples of computation-intensive sectors:

- **Big Data and Business Analytics (BDA)**
  Forecasts worldwide revenues for BDA will reach $150.8 billion in 2017, an increase of 12.4% over 2016. Commercial purchases of BDA-related hardware, software, and services are expected to maintain a CAGR of 11.9% through 2020 when revenues will be more than $210 billion.
- **Science, Technology, Engineering and Mathematics (STEM) researchers and industries**
  For example, the market for neural network software is projected to grow from USD $7.17B in 2016 to $22.55B by 2021 [5]. Driven by uncategorized and newly digitized data, neural networks are computation-intensive sorting machines.
- **Video and Image rendering**
  Rendering and visualization software is expected to grow at a compounded annual growth rate (CAGR) of 30.03% from 2016 to 2020 [6]. 3D enabled display devices, virtual and augmented reality, and high-end video games are driving this growth.
- **Advanced Risk Analytics**
  Risk analytics has a projected CAGR of 15.3% bringing the market from USD $17.60B in 2017 to $35.92B by 2022 [7]. Emerging technologies such as artificial intelligence enable institutions to improve underwriting decisions and increase revenues while reducing risk costs.

SPARC's Mission – A web-based sharing economy of computer power

SPARC is a not-for-profit foundation promoting the public sharing of compute resources. Known as collaborative consumption, (or a sharing economy), it will accelerate compute-enabled research and innovation by providing developers, scientists and engineers with access to vast reservoirs of under-utilized compute power. Specifically, SPARC's Distributed Compute Protocol (DCP) with integrate with the HTTPS/TCP/IP suite enabling standardized distribution of computations. To achieve this, we will:

Decentralized compute platform

As hardware performance and electronic device proliferation continue to grow exponentially with time, so to does the value of SPARC's network in accordance to Metcalf's law [8]. Traditional centralized computing infrastructure is static; it is expensive to purchase, operate, maintain and update, and becomes obsolete as computing technology evolves. SPARC's decentralized network will continuously grow in size and performance by connecting both the newest high-performance devices and computing infrastructure, as well as older devices, that would otherwise be discarded, to perform useful computations and earn SPARC. Furthermore, by networking secondary-purpose devices, the cost of computing will reduce to the base electrical costs, whereas centralized commercial services incur high costs relating to infrastructure, operations, and maintenance. Additionally, SPARC foresees that self-driving automated vehicles, each containing Massively Parallel Processing (MPP) units will network together, when not in operation, to form one of the most powerful distributed supercomputers on Earth. Simply put, SPARC is leveraging the Internet of Things [9] to allow any device to participate in a distributed computing network, earning SPARC for device owners and providing compute power for users. In doing so, SPARC expects to provide the most cost-effective and flexible distributed computing platform.

Browser-compatible technology

SPARC's browser-compatible platform is designed to operate across all intelligent devices: smartphones, tablets, laptops and computers will be connected by SPARC's Distributed Compute Protocol. Virtually no software installation or plug-ins are required; computations are automatically performed through a web-browser or command-line interface as a background process. In effect, participating in SPARC's computational network and mining SPARC tokens is accomplished by simply visiting a webpage. The convenience of SPARC's browser-compatible platform will greatly accelerate proliferation and adoption relative to other software-based compute platforms. In time, SPARC expects to capture a large percentage of the cloud-compute market share.

Micro-computations to replace Ads

SPARC's distributed compute platform will allow websites to generate revenue by performing micro-computations instead of displaying invasive web advertisements. Website owners can earn SPARC by distributing computational micro-tasks to visitors. These micro-computations would be performed automatically as a background process in the browser. In short, micro-computations enabled by SPARC's Distributed Compute Protocol will replace ads as a primary revenue-generating mechanism.

Recapturing wasteful blockchain computations

Existing blockchain currencies are extremely inefficient. For example, in the competition-based process of hashing a digital transaction on the Bitcoin network, approximately 12,000 peta hashes per second are discarded [10]. This amount of wasted compute power is equivalent to approximately 128,000x the combined power of the Top 500 Supercomputers. Instead of hashing to determine proof-of-work, SPARC will compensate participant nodes for actual work performed. The SPARC protocol will effectively recapture wasted computing resources and make them available to science and industry for the advancement of humankind.

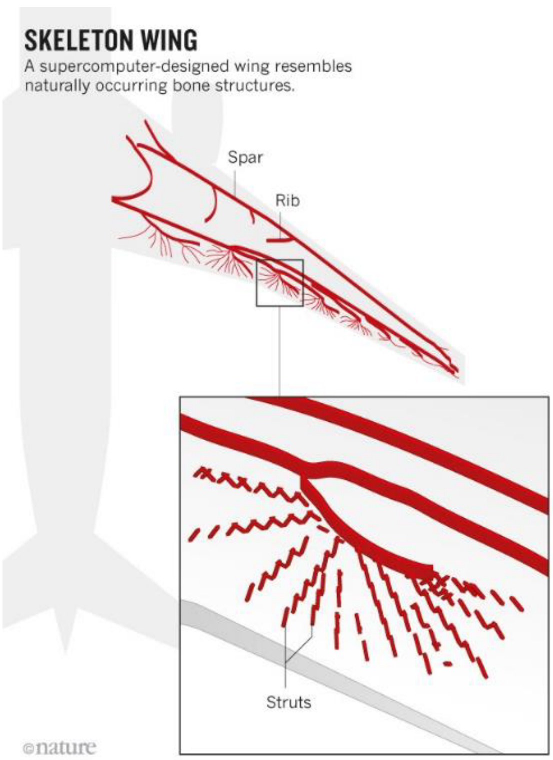
Impact on academic and scientific institutions

Sparc's network protocol is expected to have an immediate impact in academic and scientific institutions. Interviews with both senior and junior researchers at academic institutions across the US and Canada confirm that there is an imperative need for greater access to computational power. Despite this, a relatively limited number of researchers use commercial computational centers; instead, they let their laptops and personal computers crunch numbers for hours and days to obtain their results. Researchers interviewed by Sparc unanimously agree that a distributed compute protocol would be of immediate benefit to their scientific work. Sparc DCP is a network protocol designed by scientists, for scientists.

Use case examples:
Compute-enabled innovation

Web traffic and content delivery

SPARC's Distributed Compute Protocol is designed to distribute computational workloads over a network, making it ideal for web traffic and content delivery applications: "Israel-based Hola is a popular virtual private network (VPN) provider used by roughly 46 million users worldwide to make tracking their internet activity more difficult to track. The service is available in both a free and premium version. The free option routes traffic through other users of the free service, whereas the premium, paid-for alternative acts as a standard VPN. As resources are pooled between users, a free option is possible - but users must allow their Hola computers to contribute bandwidth and resources to Hola, which also powers the premium Luminati service." [11]



Supercomputer redesign of airplane wing

Engineers have used a supercomputing technique that mimics natural selection to design the internal structure of an aircraft wing from scratch. The resulting blueprint is not only lighter than existing wings, it also resembles natural formations, such as bird wing bones, that are not present in current aeroplanes. The organic-looking product is as stiff as a conventional aircraft wing but lighter, which could save up to 200 tonnes of fuel per year per plane [9].

> *This is a really nice illustration of how to employ computingbased optimization methods at immensely high resolution*

says Matthew Santer, an aerospace engineer at Imperial College London. The method could feed into the design process, although there are a number of hurdles to using it in aerospace applications in its present form, he adds [9].

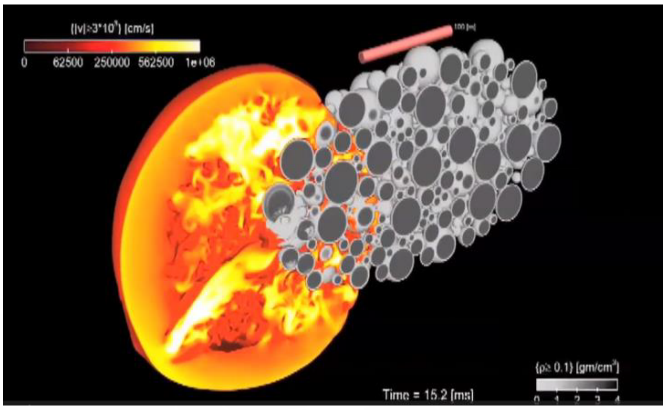
Stronger concrete to build the cities of tomorrow

SPARC will provide a convenient computing platform that will accelerate work performed by scientists and engineers like Oral Buyukozturk from MIT: "In a recent study, Oral Buyukozturk, a professor of civil and environmental engineering at MIT, and his colleagues analyzed a key property in concrete, at the level of individual atoms, that contributes to its overall strength and durability. The group developed a computer model to simulate the behavior of individual atoms which arrange to form molecular building blocks within a hardening material. The scientists describe a computer model that is part of a computational framework that they have developed to analyze how the atomic structure of concrete affects engineering properties. These models simulate the sliding and movement of clusters of particles at molecular scales within concrete. Their goal is to develop a stronger concrete that will be used to build the buildings of the future." [13]

Deflecting an earthbound asteroid with a nuclear bomb

A well-placed nuclear explosion could actually save humanity from a big asteroid hurtling toward Earth, just like in the movies, a new study suggests. Scientists at Los Alamos National Laboratory, a United States Department of Energy facility in New Mexico, used a supercomputer to model nukes' anti-asteroid effectiveness. They attacked a 1,650-foot-long (500-meter) space rock with a 1-megaton nuclear weapon [11]. In a recent video released by Los Alamos, scientist Bob Weaver stated:

> *Ultimately this 1-megaton blast will disrupt all of the rocks in the rockpile of this asteroid, and if this were an Earth-crossing asteroid, would fully mitigate the hazard represented by the initial asteroid itself,*



In the 3-D modeling study, run on 32,000 processors of the Cielo supercomputer, the blast went off at the asteroid's surface." [14]

Go to market strategies

Short term

- **Raise awareness:**
  - Alpha version of SPARC atop BOINC project provides visibility to over a million people, many of whom are already comfortable with the concept of distributed computing and easily convinced about the efficacy of the next step in this space;
  - Social media and public outreach using traditional advertising on Google and Facebook as well as growing SPARC's channels on Slack, Telegram, Twitter and other outlets.
  - Assemble an "A" list of advisors and influencers in the arenas of web development, ecma standards, telecoms and cloud providers for the promotion of SPARC's mission and longterm vision;

  Goal:
- - To be established as the leading brand for the sharing economy of compute resources;
  - SPARC Distributed Compute Protocol recognized as the first standards-conformant implementation.

Medium term

- **Approach current telecommunication providers to generate interest in implementing SPARC's Distributed Compute Protocol.**
- **Attain critical mass:**

  Attracting miners
  - Appealing to their economic interest by subsidizing their device purchase cost;
  - Appealing to the basic desire to help scientific progress by contributing their resources to the collective;
  - Encouraging users of compute power themselves to add to the network;
  - Organic advertising using traditional channels to attract miners.
- **Attracting researchers and users**
  - Leveraging the first scientific peer-reviewed papers that use the network to achieve publishable results;
  - Growing a team of trusted experts reach out and establish significant beachheads among leading consumers of compute power in arenas such as Artificial Intelligence, Big Data Analytics, and Insurance underwriting.
- **Goal:**
  - To be established as the leading brand for the sharing economy of compute resources;
  - SPARC Distributed Compute Protocol achieving general adoption by the web standards community.

Long Term

- **The Sparc foundation will continue to maintain and improve the specification and Software Development Kits for the network.**
- **SPARC tokens will be the standard currency for the purchase of utility compute power, and will also be associated with economies surrounding ECMA modules and other dependency trees.**

  Periodical featuring peer-reviewed SPARC projects on sparc.community website.
- **SPARC in Academia:**
  - Direct sponsorships, grants, and establishment of academic chairs for research projects involving compute power in emerging fields;
  - Sparc will become the students' and researchers' natural environment for meeting their computational requirements.
- **SPARC in Business and Industry:**
  - Protocol is ubiquitous and self-promoting;
  - Module and related ecosystems are supported by major tech companies.
- **Goal:**
  - SPARC Distributed Compute Protocol will be the de facto global standard for compute power and seamlessly integrated with the internet protocol suite.

SPARC Cryptocurrency – The SPARC Token

Purpose and technical description of the token

SPARC's Distributed Compute Protocol is designed to convert web-compute power into a utility, and SPARC tokens act as the currency.

SPARC is fractionally divisible, fungible and is long-term inflationary where new circulation is donated as compute power to humanitarian research projects. SPARC tokens are transferable as they may be traded on cryptocurrency exchanges. Miners will gain SPARC tokens by computing discrete compute packages and will be rewarded based on market-driven valuation.

SPARC will be implemented on the public Ethereum blockchain in accordance with the ERC-20 (Ethereum Token Standard). Any token implemented according to the ERC-20 standard is compatible with the existing infrastructure of the Ethereum ecosystem. Advanced functions and high activity within the Ethereum ecosystem make it the most suitable platform for issuing SPARC.

SPARC Token usage opportunities

Earning and spending SPARC:

USE SPARC TO

- Mine SPARC tokens by completing compute packages on the network
- Tip es6 module creators, such as the creators of useful math functions
- Charge software licensing fees, such as digital rights management
- Purchase freelance development/stock images/assets
- Pay bounties for bugs
- Get access to research papers/equipment

Overview of the SPARC Ecosystem

Basic concept of operations

The three principal components of a distributed computing protocol are: **Miners, Hosts** and **Boards.**

The Miner is the person who is connecting a device to the network with the intention of contributing compute resources in exchange for SPARC. The Miner will access the network and register the device using either a standard web browser or a mobile App built specifically for the type of device and its operating system. The miner workflow is described in the technical description of the platform.

The Host is a person who wants to deploy a computational project to the network. A package is created which describes the job that needs to be done, its component tasks, and other required metadata, as well as an adequate SPARC token allocation. Access to the Sparc ecosystem is available via the command-line, the official SPARC implementation or third-party interfaces. The Host maintains an exposed folder on an internet server for sending and receiving requests and tasks from miners. The host workflow is described in the technical description of the platform.

The Board is a stateless Application Programing Interface (API) resource which uses a database to store data associated with tasks, miners, hosts and other related information. In addition to storing information, the Board balances miner availability and performance with Host task requirements as well as collecting and distributing SPARC tokens used. Boards may interface with other boards to communicate miner availability and characteristics.

**Any single node may run one or each of the above components simultaneously. i.e. a Host could serve, or even mine, its own tasks.*

Internet Environment and SPARC

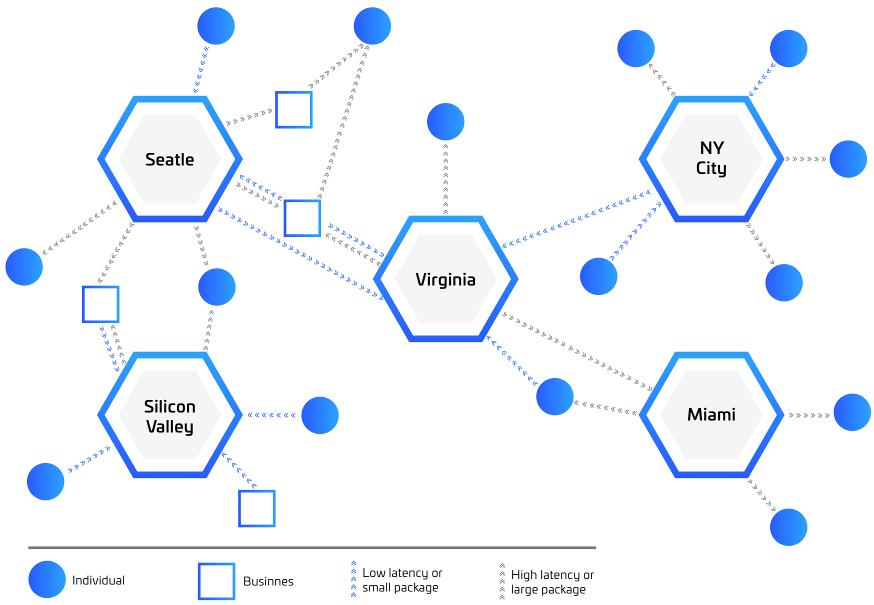
Similar in the way TCP/IP routes and balances static assets, the network utilizes a standards-based specification which enables the routing and balancing of dynamic assets. In this way the static assets can be understood to be commoditized.

Commoditization allows the network to balance latency with device availability and package performance requirements. Additionally, it permits infrastructure owners in more ideal network locations to enter the dynamic compute market. Miners residing at less ideal locations will still compute lower priority packages and gain SPARC tokens. This will be discussed in greater detail in the technical description of the platform.

The key features of this concept also make it

- Modular: NPM packages can be used as dependencies;
- Efficient: common dependencies are cached instead of being re-packaged with each job;
- Secure: tasks compiled as strings and compartmentalized in workers to only have the access that is legitimately granted.

SPARC aims to promote and favour miners and hosts in ideal network locations relative to one another as depicted in Figure 3.

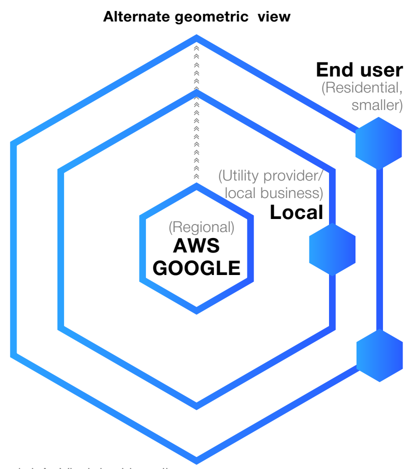


[Figure 3]: SPARC's network topology illustrated for high and low latency computational tasks.

ECMAScript

ECMAScript, often referred to as JavaScript, is a memory safe language that all websites and web applications depend upon. It has been a fast-growing language since its inception, and has now become the largest language by number of developers [15].

Historically, ECMAScript has been limited to browser-side execution environments. However, it has recently gained wide adoption for server-side events as developers want to be able to blend their environment and work using a single language [16]. Therefore, ECMAScript is - and will continue to be - the dominant language for client and server traffic. In particular, ECMAScript has been organized in such a way that it has a common dependency tree called the Node Package Manager, and these dependencies are designed to run as self-contained modules.



Regional: LA, Virginia, Hawaii.
Local: cogeco, time warner, university, etc.
End user: personal, residential, small business, etc.

> *ECMAScript is the perfect environment for distributed computing.*

Additionally, many servers today are turning to stateless technology, and interfacing with databases through tokens. This technology blends well with both cryptographic currencies and distributed computing, allowing us to further gain efficiency and connectivity. In the future, SPARC foresees the addition of optional typing to ECMAScript demonstrated today in TypeScript [17], or TurboScript [18]. Naturally, this will lead to Ahead-of-Time compiling to WebAssembly [19] for various static or compute-heavy modules and continue to improve JavaScript’s performance.

The SPARC Foundation

Goals

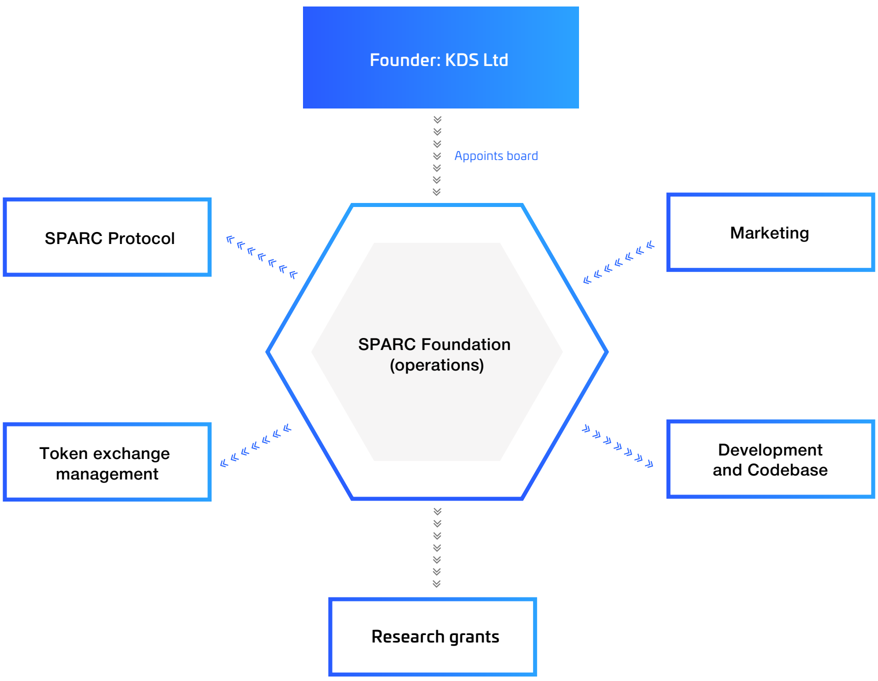
The Foundation maintains the protocol and provides all auxiliary services and support necessary for smooth, secure and predictable operations. The goal of the SPARC Foundation is to establish a decentralized global public compute utility. SPARC seeks to offer a reliable, low-cost way for compute package transactions to be to be distributed across a network in the most efficient manner. Such a decentralized system will be capable of delivering reliable, efficient, and ever-expanding compute capacity. The mission is to accelerate the transition to a decentralized world made up of both large and small suppliers and consumers of compute resources forming the Internet-of-Things

By organizing as a foundation, SPARC will be permitted to engage in proper stewardship of the collective compute resource and the protocol which governs it. As a not-for-profit entity, it is effectively disarmed and will be seen as such by other entities which have competitive interests in the space. This will encourage adoption of the protocol and expansion of the SPARC network.

Foundation structure

- Kings Distributed Systems (KDS) Ltd. is the Founder;
- KDS contributes and manages the assets of the foundation including the intellectual property as well as the protocol;
- KDS appoints and controls the board;
- The SPARC Foundation Board of Directors controls:
  - The protocol which enables the commoditization of the global compute grid;
  - Daily operations of the Foundation;
  - The SPARC token contract and token supply;
  - Promotion of the SPARC network;
  - Management of road-map and open-commit codebase;
  - Work with researchers and other causes to help them leverage idle compute power.

SPARC's corporate structure and governance schema is summarily depicted in Figure 4 below.



[Figure 4]: SPARC's corporate structure and governance schema.

SPARC ICO

The ICO will be conducted in two phases. Presale and crowd sale.

Phase 1 - Presale

SPARC token begins with a Presale to raise funds towards running the Alpha, marketing the ICO, and getting listed on exchanges. SPARC is raising funds for the continued development and scaling of the network. In return for Ethereum, buyers will receive an initial supply of SPARC that can be traded on the exchange or held until release for use on the network.

This limited Presale is designed to benefit early adopters and provide SPARC with funds necessary to launch the protocol. Round 1 buyers will receive double the exchange rate of tokens compared to Round 2. In addition, buyers who participate in both rounds will receive another 100% bonus of their second-round purchase. This second round bonus is capped at the amount of tokens received during the first round.

Example: First round purchase of 1 Ethereum generates 20 000 SPARC at purchaser's address. Second round purchase of 3 Ethereum generates 30 000 SPARC + 20 000 Bonus SPARC at same address.

**Please do not send your purchase from an exchange address. If you do, your tokens can be recovered, but this is a manual process and may take several days involving proving your ownership of that address.*

Start Date	July 27th 2017 11AM EST
Accepted Currencies	ETH
Maximum Cap	1000 ETH
Exchange Rate	1 ETH : 20000 SPARC
Bonus	100% Over ICO
Round 2 Bonus	Additional 100%
Round 2 Bonus Cap	SPARC received in Round 1
Presale Contract Address	Etherscan
Token Contract Address	Etherscan
Required Gas	60 000
Presale Address	013FBAED9c80CC4422FF50D71F4Dc622bC7c8AE2

[Table 1]: ICO presale information.

Phase 2 – Crowd sale

SPARC's second round is designed to open funding up to the broader public now that the SPARC Alpha has gained traction.

SPARC is raising funds to support the continued development of the protocol and associated Software Development Kits. Q1 2018 is SPARC's projected release for the minimum viable product. This release is targeted at bringing new users and agile projects onto the system. Moving forward, SPARC will be expanding support to niche cases and developing extensions for proprietary programs such as MatLab and COMSOL.

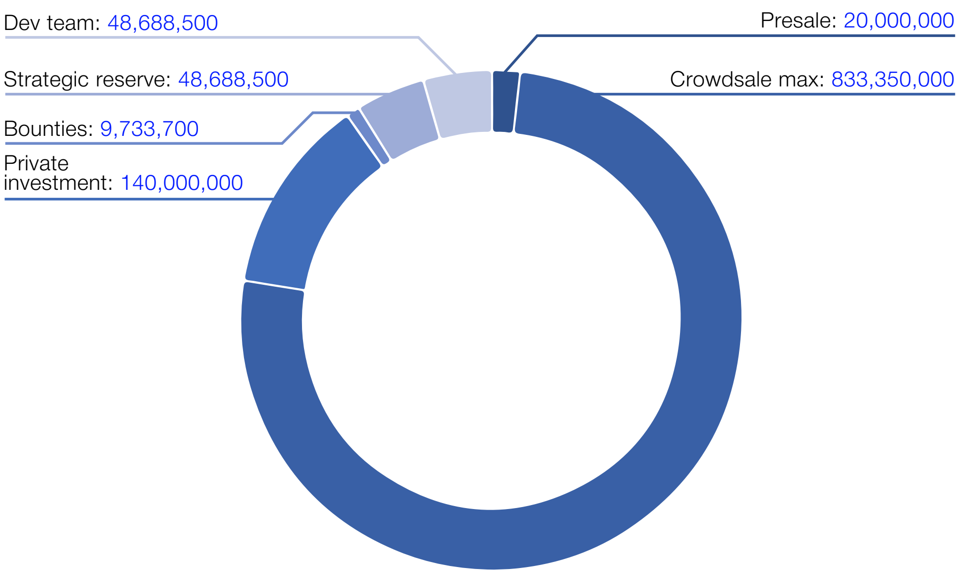
SPARC's goal is to establish a long-term budget to ensure continuation of the network. Miners and project hosts will be confident in moving onto SPARC's platform.

The minimum cap is set to ensure that SPARC has critical mass to launch protocol. If the minimum cap is not met, SPARC will continue to build support on its Alpha and develop the network protocol with the intention of conducting a successful funding round in the future.

**Please do not send your purchase from an exchange address. If you do, your tokens can be recovered, but this is a manual process and may take several days involving proving your ownership of that address.*

Tentative Start Date	Jan 15 2018
End Date	30 days after Start Date
Assumed USD to ETH exchange	$300.00
Minimum Cap	$5M (16,667 ETH = 166,670,000 SPARC)
Maximum Cap	$25M (83,335 ETH = 833,350,000 SPARC)
Exchange Rate	1 ETH = 10,000 SPARC
Bounty tokens	1% (8,333,500 Max)
Developer team tokens	5% (41,667,500 Max)
Strategic reserve tokens	5% (41,667,500 Max)

[Table 2]: ICO Crowd sale information.



[Figure 5]: Distribution of tokens from ICO crowd sale.

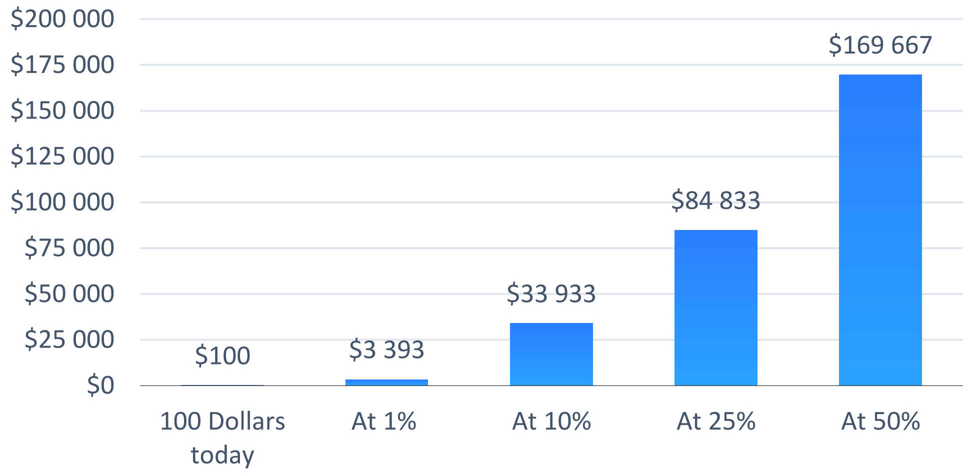
SPARC Market Potential

As the SPARC protocol gains market usage, the SPARC token increases in value. The first percent of cloud computing market share captured will represent a token value gain of approximately 3400% relative to the ICO token value.

Assumed price of ETH in USD	$300.00
Value of cloud computing market in 2018 (USD)	110 Billion
Total SPARC economy token count	1,080,440,700
SPARC tokens per ETH	10,000
Today's price of SPARC in USD	$0.03
SPARC tokens for $100.00 USD	3,333
Value increase with each 1% of market share captured	$1.02
SPARC value increase with first 1% of market share captured	3394%

[Table 3]: SPARC market potential, estimation based on market share.

As SPARC's Distributed Compute Protocol gains widespread adoption, SPARC will capture an increasingly larger percentage of the cloud computing market share shown in Figure 1. For example, $100.00 USD of SPARC tokens will increase as a function of market share capture as shown in Figure 6.



[Figure 6]: Value of $100.00 of SPARC tokens as a function of cloud computing market share captured.

Cryptonomos

Cryptonomos is a successful hosting and marketing platform for worthy partners in the blockchain space. SPARC takes advantage of Cryptonomos' technology and expert guidance at every stage of the token sale to ensure the utmost security and success of the ICO.

Technology

- Secure user account and wallet;
- Secure collection of funds;
- Token distribution oversight;
- Smart contract and token development.

Framework

- Token economics;
- Legal analysis;
- Token holder agreements and disclosures;
- Design and presentation;
- Escrow services.

Marketing

- Early-backer engagement;
- Marketing and Public Relations campaigns;
- Community channel management.

The SPARC team is able to focus its attention on developing the protocol and technology while relying on Cryptonomos to provide ancillary services at a high level of quality and security. The result of this partnership will be a greater assurance of success in reaching the brand goals as well as the deployment of the promised project on a timely basis.

Technical description of the SPARC Protocol

SPARC's Distributed Compute Protocol (DCP) connects project hosts with miners who interact with a board. A technical description of the Host, Miner, and Board workflows is presented here.

Host workflow

- Install npm package (or from client app for small jobs);
- Create or load private key;
- Place job description/custom functions into exposed folder;
- Place per-task data into exposed folder;
- Selection, automatic or manual, of board to distribute tasks;
- Send job description to board;
- Accept requests from miners for custom functions/data;
- Accept 'expectation of result' from board with miner id/task id/job id/checksum;
- Accept results directly from miner;
- Confirm results receipt and send rating change to board (who may accept/weight/discard host's rating).

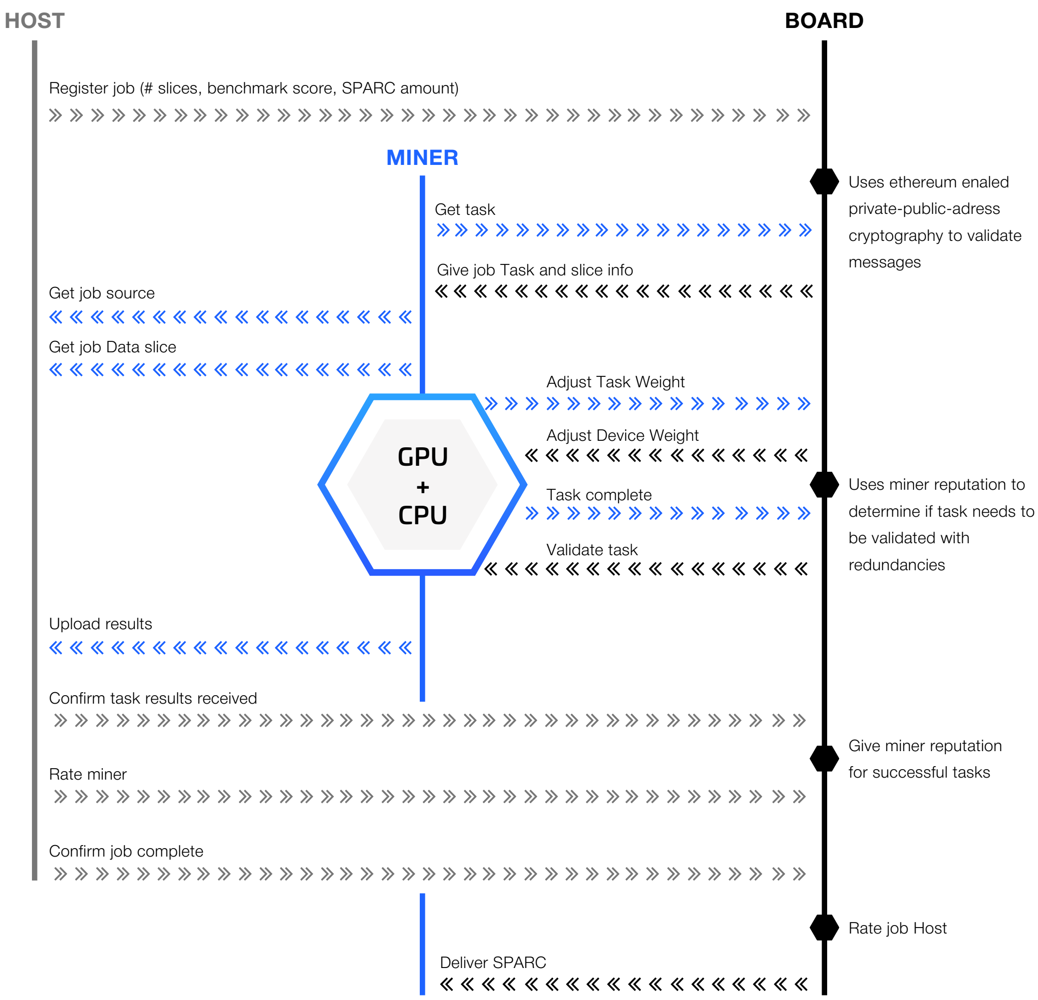
Miner workflow

- Visit website or download application;
- Selection, automatic or manual, of board to connect to;
- Create or load private key;
- Request tasks;
- Complete tasks by performing computations;
- Send checksum to board;
- Board notifies host of good reputation checksum;
- Board decides miner is not well-weighted and task is sent again to another miner to verify checksum; (miners split SPARC share, but get reputation bonus);
- Upload results directly to host with checksum;
- Successfully completed tasks increase reputation score.

Board Workflow

- Install npm package;
- Create or load private key;
- Create database to store records (uses file buckets by default);
- Accept job header and place in available jobs;
- Accept task requests, and assigns tasks using Compute Resource Allocation algorithm;
- Receive checksum of results from miner;
- Assess miner reputation, trust/discard result;
- Send checksum to host, instruct miner to upload data;
- Host verifies receipt of job when upload complete;
- Exchange tokens and update miner reputation score/statistics.

The network flow diagram is depicted in Figure 7 below.



[Figure 7]: Network flow diagram between the Host, Miner and Board.

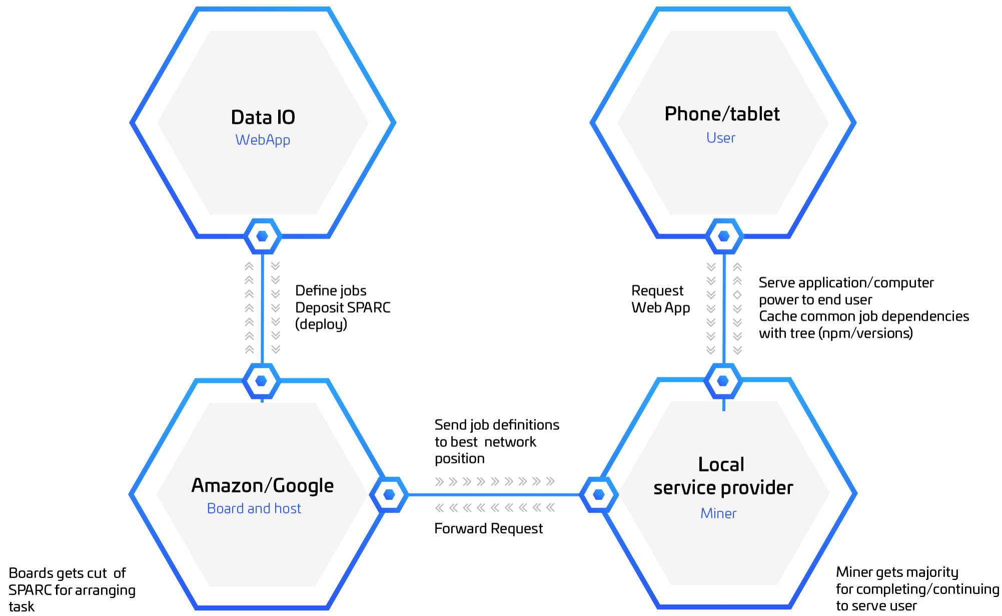
SPARC's protocol is versatile in it's ability to accommodate a variety user/project requirements and constraints such as security, speed, and latency. Three use case examples of protocol implementation are depicted below.

Use case 1: Web Traffic

This use case demonstrates how Google and other service providers can challenge Amazon's monopoly of cloud compute services through the commoditization of server power.

WEB Traffic

Allows Google and smaller service providers to challenge Amazons monopoly through comoditization of server powers.



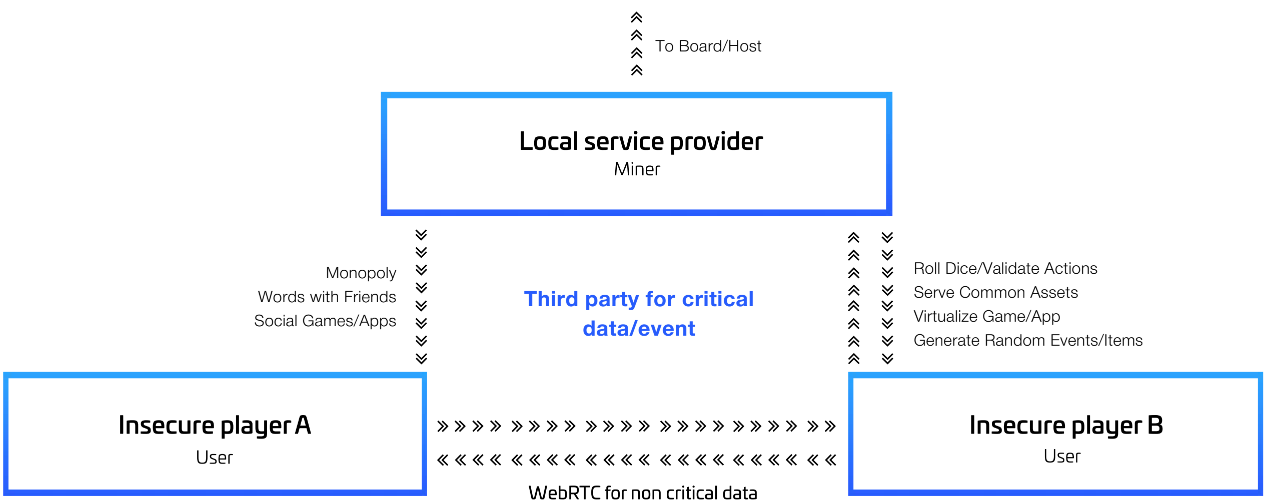
[Figure 8]: Web traffic implementation of Sparc's Distributed Compute Protocol.

- Whereas TCP serves static assets, DCP serves dynamic assets;
- Layered levels of security;
- The local service provider is this use case is a regional utility provider or another high-bandwidth intercepting node because of its ideal network location.

Some addresses can be verified secure such as Amazon or Google and others are rated for security by obscurity.

Use case 2: Secure third party

This use case outlines applications requiring third-party security. Nodes in less ideal locations can still serve secure applications.

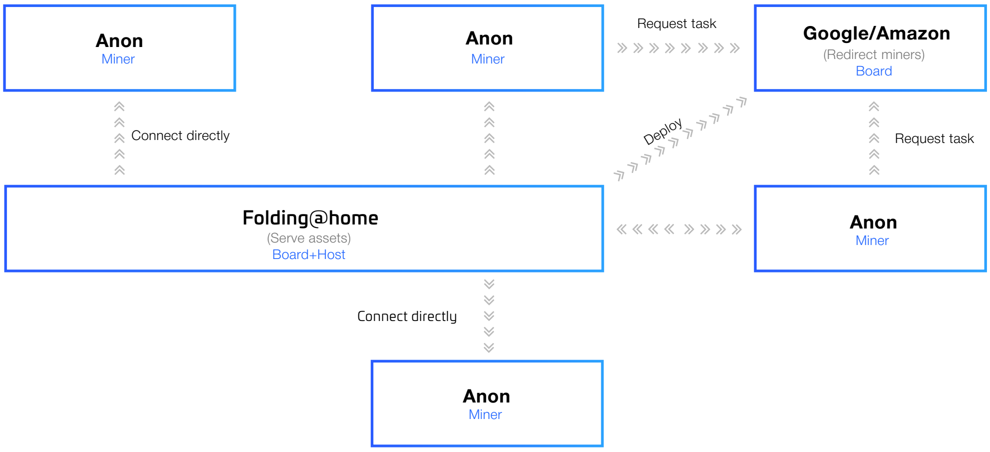


[Figure 9]: Secure third party implementation of SPARC's Distributed Compute Protocol.

- Some applications require third parties to complete semi-secure tasks such as generating an event;
- Example: a random roll of a dice;
- Security by obscurity;
- Nodes serving semi-secure applications do not have to be in ideal network locations.

Use case 3: Massively Parallelizable Tasks

This use case outlines the protocol's ability to distribute computational tasks - such physics, mathematics, and rendering - over a parallel network.



[Figure 10]: Massively parallelizable tasks implementation of SPARC's Distributed Compute Protocol.

- Higher latency applications, such as aircraft wing re-design, concrete molecular properties and atomic detonation simulations, are processed in this scenario;
- Nodes serving higher applications do not have to be in ideal network locations.

Compute Resource Allocation Algorithm

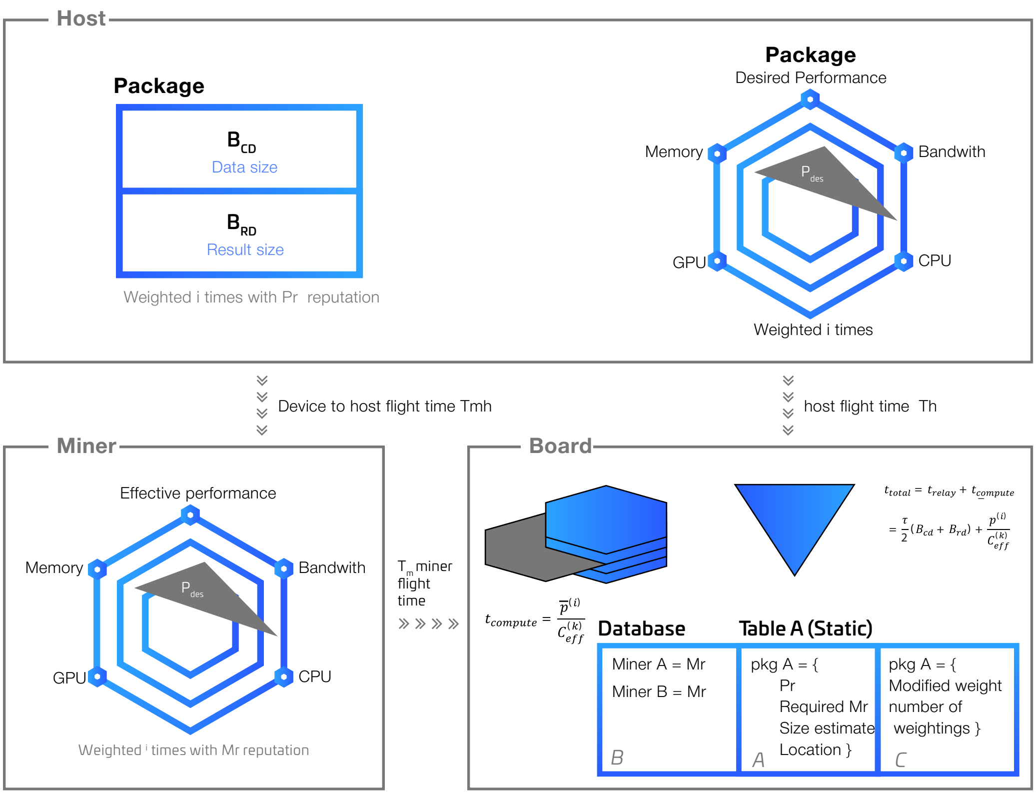
Mathematical Model

A SPARC board will intelligently distribute work packages to miners for computation. SPARC's Compute Resource Allocation (CRA) algorithm accounts for the proximity of the miner within the network (affects time of flight between nodes), the effective performance of the miner's device (device performance when computing a package with specified performance requirements), the reputation of the miner, the weight of the package, and the reliability of the miner.

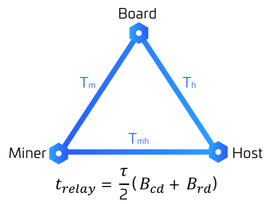
A graphical depiction of the algorithm is presented in [Figure 11] and the full mathematical derivation follows.

$\tau^{(k)}$	Round-trip ping time per byte weighted k times [s]
B_{cd}	Size of compute data and instructions [bytes]
B_{rd}	Size of result data [bytes]
$\overline{p}^{(i)}$	Average package size weighted i times [effective operations]
$C_{eff}^{(k)}$	Effective device performance weighted k times [effective operations/s]
p^{+}	Lifetime number of completed packages by device
p^{-}	Lifetime number of dropped packages by device

[Table 4]: List of select variables



Rep	Secure pfg	Obscure pfg	Open pkg
<10	no	no	yes
<100	no	yes	yes
Verified	yes	yes	yes



[Figure 11]: Graphical depiction of the Compute Resource Allocation algorithm.

Scoring Algorithm

The total work corresponding to a job is defined as W where the units of work are effective operations. Parameters such as memory capacity, CPU speed, GPU speed and bandwidth will affect a device's ability and rate of performing work. The job can be subdivided into N tasks of work, labelled pi , that can be distributed over a network. The total work is then written as the sum of the individual tasks:

$$W = \sum_{i=1}^{N} p_i \qquad [1]$$

This sum can be simplified by considering the average task size $\bar{p}$

$$W = N\bar{p} \qquad [2]$$

The average task size is then

$$\bar{p} = \frac{W}{N} \qquad [3]$$

The value W is seldom explicitly known before completing all computations, but it can be estimated after a first task is computed $\bar{p}^{(1)}$, and the estimate is updated and refined as subsequent tasks (numbered i) are computed $\bar{p}^{(1)}$). The average task size will depend on the specific job and must be re-evaluated for different jobs.

A miner's device performance D depends on

- CPU speed, variable a with units of effective operations per second;
- GPU speed, variable b with units of effective operations per second;
- Memory capacity, variable c with units of B (bytes);
- Bandwidth, variable d with units of B/s (bytes per second);

and can be represented by a four-dimensional vector

$$\mathbf{D} = a\hat{\mathbf{e}}_1 + b\hat{\mathbf{e}}_2 + c\hat{\mathbf{e}}_3 + d\hat{\mathbf{e}}_4 \qquad [4]$$

Different jobs will require different performance criteria. The task performance requirements R is also represented by a four-dimensional vector

$$\mathbf{R} = e\hat{\mathbf{e}}_1 + f\hat{\mathbf{e}}_2 + g\hat{\mathbf{e}}_3 + \mathrm{h}\,\hat{\mathbf{e}}_4 \qquad [5]$$

The effective computational performance C_{eff} is a function of the task performance requirements and the device performance

$$C_{eff} = f(\mathbf{D}, \mathbf{R}) \qquad [6]$$

For which a first estimate $C_{eff}^{(1)}$ can be computed theoretically or by computing a small benchmark test. Subsequent task iterations (numbered k) that are processed by the device update the effective device performance $C_{eff}^{(k)}$. The effective performance corresponds to a specific job; it must be re-evaluated for different jobs. The approximate time to compute an average task $\bar{p}^{(i)}$ based on the device's effective compute performance $C_{eff}^{(k)}$. is then

$$t_{compute} = \frac{\bar{p}^{(i)}}{C_{eff}^{(k)}} \qquad [7]$$

The approximate time for relaying task compute data B_{cd} to the node and receiving task result data Brd from the node given a round-trip ping time per byte τ is

$$t_{relay} = \frac{\tau}{2}(B_{cd} + B_{rd}) \qquad [8]$$

The theoretical total time required to transmit, remotely compute and receive result data for an average task is

$$t_{total} = t_{relay} + t_{compute} \quad [9]$$

$$= \frac{\tau}{2}(B_{cd} + B_{rd}) + \frac{\bar{p}^{(i)}}{C_{eff}^{(k)}} \quad [10]$$

Shorter time estimates yield a favourable time score t* according to

$$t^* = (t_{total})^{-1} \quad [11]$$

Nodes will also have a reputation score *R** and a reliability score *r**. A node's reputation and reliability scores will increase with completed tasks p^+, will decrease with dropped tasks p-, and will be reset or barred if deceitful behaviour such as task result spoofing is detected. These values are cumulative for the device across different jobs.

$$r^* = \frac{p^+}{p^+ + p^-} \quad [12]$$

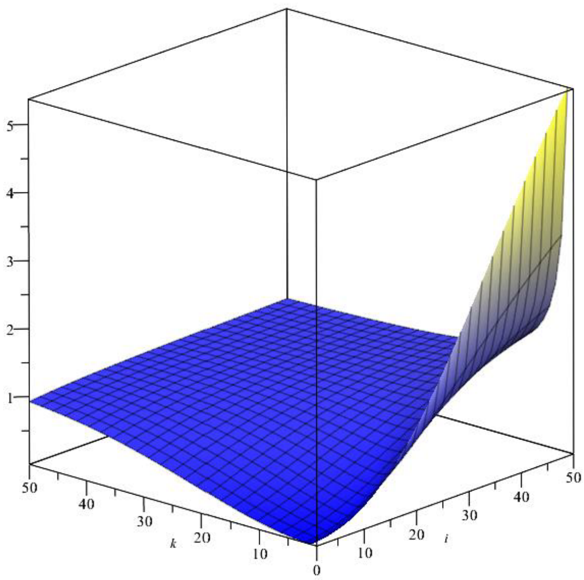
$$R^* = p^+ - p^- \quad [13]$$

Devices that are new to the network will have relatively uncharacterized computational performance coefficients $C_{eff}^{(k<10)}$ and will need to receive well-characterized tasks. Conversely, uncharacterized tasks should not be assigned to uncharacterized devices. Expressed mathematically, given $\bar{p}^{(i)}$ (ith task from a job W) and $C_{eff}^{(k<10)}$ (kth task completed by a device), the formula for the associativity score that simultaneously

- Discourages association between poorly-weighted tasks with poorly-weighted devices;
- Encourages associations between strongly-weighted tasks with poorly-weighted devices;
- And leaves the remainder unchanged, is given as

$$a^* = \left(\frac{0.1 \cdot i}{k} + 1\right)\left(1 - e^{-0.001 \cdot (k+i)^2}\right) \quad [14]$$

A 3D plot of equation (14) is presented in Figure 12 below.



[Figure 12]: 3D plot of the associativity score;Task weighted *i* times associated with a device weighted *k* times.

Figure 12 clearly shows that uncharacterized tasks (low i) cannot be assigned to uncharacterized devices (low k), only well-characterized tasks (high i) are assigned to uncharacterized devices (low i), any other association of med-high i with med-high k has the same score.

The overall score useful for node prioritization is the product of the time, reliability and associativity scores

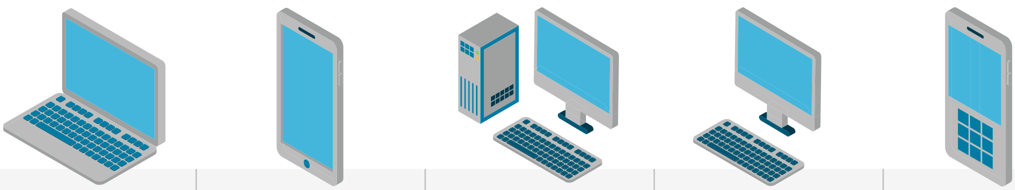
$$s^* = t^* r^* a^* \qquad [15]$$

Substituting equations (11), (12) and (14) into equation (15) yields the final desired expression for the Compute Resource Allocation algorithm:

$$\left(\frac{\tau^{(k)}}{2}(B_{cd} + B_{rd}) + \frac{\overline{p}^{(i)}}{C_{eff}^{(k)}}\right)^{-1} \left(\frac{p^+}{p^+ + p^-}\right)\left(\frac{0.1 \cdot i}{k} + 1\right)\left(1 - e^{-0.001 \cdot (k+i)^2}\right) \qquad [16]$$

Example SPARC Network Device Scoring

Suppose that the average task size for a given computational job is $\overline{p}$=10 Gflop, the Board is located in Kingston ON, and that it's the 200th task in a job series (i = 200), and. The algorithm produces the following s* scores which dictates the priority sequence for distributing that task.



Location	Kingston	Miami	Munich	Xi'an	Montreal
IP address	199.246.2.51	50.73.157.178	188.174.109.118	219.144.222.232	174.142.39.136
τ	4 ms	105 ms	132 ms	268 ms	14 ms
C_{eff}	43 Gflop/s	11 Gflop/s	257 Gflop/s	107 Gflop/s	6 Gflop/s
t^*	4.227290602	0.986104886	5.85101539	2.766573586	0.595001983
p^+	632	84	794	1245	7
p^-	4	1	24	32	0
r	0.993710692	0.988235294	0.970660147	0.974941269	1
R	628	83	770	1213	7
i	200	200	200	200	200
k	39	17	79	34	1
a	1.512820513	2.176470588	1.253164557	1.588235294	2
s	6.35491098	2.120978536	7.11715694	4.283862503	12.49504165

[Table 5]: Example application of the Computational Resource Allocation algorithm.

Conclusion

SPARC provides a specification for a sharing economy of compute resources. SPARC's Distributed Compute Protocol (DCP) integrates with the HTTPS/TCP/IP suite and effectively transforms computational power into a public utility. The protocol enables any connected device to contribute and consume computing resources from a compute grid. The grid distributes compute resources for client projects on a network using SPARC's Compute Resource Allocation algorithm. Devices then perform computational work in exchange for SPARC tokens that can be exchanged for compute power through the network, or listed on a cryptocurrency exchange.

SPARC's mission is to accelerate compute-enabled research and innovation by providing developers, scientists and engineers with access to idle compute power through a collaborative compute network. By connecting their devices to the network, miners will accelerate science and technology while earning SPARC.

Together, we will advance humanity…

SPARC addendum

The SPARC team

We are an experienced group of researchers and developers who have strong technology and project implementation backgrounds. Our group also has team members and partners who have deep experience in marketing and commercializing technology products to help us connect with both sides of our market. Additionally, we are proud to have a growing advisory board of experts who are helping us forge the way forward in this new and exciting space. See our website for more information.

Legal Disclaimer

This is a pre-final version of the document and is made available to the public for informational purposes only. Nothing in this document shall be construed as an offer to enter into any transaction with any of the companies or individuals mentioned herein. Nothing in this document is to be considered as advice, consultation or recommendation, and shall not be relied upon as such by any person.

Any action taken by any person in reliance on this document is made by them at their own risk and based on their own assessment, and Kings Distributed Systems Ltd. and their affiliated parties shall not bear any liability for the consequences of such actions. Comprehensive terms and conditions of acquisition of SPARC will be made available in the final version of the present document.

References

[1] Cloud Security Alliance. (2017) Custom Applications and IaaS Trends 2017 Report. Retrieved from http://info.skyhighnetworks.com/rs/274-AUP-214/images/wp-csa-survey-custom-apps-iaas-survey-report.pdf

[2] Pettey C. and Goasduff L. (2017, Feb 22). Gartner Says Worldwide Public Cloud Services Market to Grow 18 Percent in 2017 [Press release]. Retrieved from http://www.gartner.com/newsroom/id/3616417

[3] Columbus L. (2017, Apr 29). Roundup Of Cloud Computing Forecasts, 2017 [Article]. Retrieved from https://www.forbes.com/sites/louiscolumbus/2017/04/29/roundup-of-cloud-computing-forecasts-2017/#25949c6131e8

[4] Feldman, M. (2016, Nov 30). IDC's Latest Forecast for the HPC Market: "2016 is Looking Good" [Article]. Retrieved from https://www.top500.org/news/idcs-latest-forecast-for-the-hpc-market-2016-is-looking-good/

[5] CISION PR Newswire. (2017, Jan 23). Neural Network Software Market by Type, Vertical, and Region - Global Forecast to 2021 [Article]. Retrieved from http://www.prnewswire.com/news-releases/neural-network-software-market-by-type-vertical-and-region---global-forecast-to-2021-300395057.html

[6] Business Wire. (2016, Mar 31). Visualization and 3D Rendering Software Market 2016-2020 - High-End Video Games, Architectural and Product Visualization, Training Simulation, Marketing and Advertisement - Research and Markets [Article]. Retrieved from http://www.businesswire.com/news/home/20160331005465/en/Visualization-3D-Rendering-Software-Market-2016-2020--

[7] Markets and markets. (2017). Risk Analytics Market worth 35.92 Billion USD by 2022 [Article]. Retrieved from http://www.marketsandmarkets.com/PressReleases/risk-analytics.asp

[8] Saran C. (2016, Mar 24). Metcalfe's law of connections will drive digital business value [Article]. Retrieved from http://www.computerweekly.com/news/450279842/Metcalfes-law-of-connections-will-drive-digital-business-value

[9] Columbus L. (2017, Jan 29). Forbes Tech: Internet Of Things Market To Reach $267B By 2020 [Article]. Retrieved from https://www.forbes.com/sites/louiscolumbus/2017/01/29/internet-of-things-market-to-reach-267b-by-2020/#3593f782609b

[10] Blockchain Info (2017). Has Rate TH/s [Infographic] Retrieved from https://blockchain.info/charts/hash-rate

[11] Osbourne C. (2015, May 29). Hola: A free VPN with a side of botnet [Article]. Retrieved from http://www.zdnet.com/article/hola-a-free-vpn-with-a-side-of-botnet/

[12] Gibney E. (2017, Oct 4). Nature News : Supercomputer redesign of aeroplane wing mirrors bird anatomy [Article]. Retrieved from http://www.nature.com/news/supercomputer-redesign-of-aeroplane-wing-mirrors-bird-anatomy-1.22759?WT.ec_id=NEWSDAILY-20171006

[13] Chu J. (2017, Sep 29). MIT News Office : A concrete solution [Article]. Retrieved from http://news.mit.edu/2017/concrete-solution-microscopic-properties-stronger-sustainable-0929

[14] Wall M. (2012, Mar 12) Space.com : How a Nuclear Bomb Could Save Earth From an Asteroid. [Article]. Retrieved from https://www.space.com/14857-asteroid-nuclear-bomb-explosion-video.html

[15] GitHub (2017) The State of the Octoverse 2017 [Website]. Retrieved from https://octoverse.github.com

[16] VanToll TJ (2016, Feb 2) What To Expect From JavaScript In 2016 – Beyond the Browser [Article]. Retrieved from https://developer.telerik.com/featured/what-to-expect-from-javascript-in-2016-beyond-the-browser/

[17] TypeScript (2017) [Website]. Retrieved from https://www.typescriptlang.org

[18] GitHub (2017) [Website]. Retrieved from https://github.com/01alchemist/TurboScript

[19] WebAssembly.org (2017) WebAssembly or wasm is a new portable, size- and load-time-efficient format suitable for compilation to the web [Website]. Retrieved from http://webassembly.org/