PoC Platform Partnership Proposal: Cardano

2021

Christopher Swenor (chris@reach.sh)
Jay McCarthy (jay@reach.sh)
Brandon Rodriguez (brandon@reach.sh)

1. Executive Summary: Preferred Platform Partnership

Reach has the ability to drastically lower the barrier to entry into DApp development by making it easy for any developer to build safe applications on the decentralized network.

Our platform's competitive advantages directly support your ecosystem's mission of achieving interoperability by making it easier to actually build protocol agnostic, universal, and safe applications, and for developers to experience the value and accessibility of a universal blockchain ecosystem.

Reach's large global developer community has the potential to add value to your platform as a Preferred Reach Integration Partner, allowing the over 2,100 (and growing) developers in our community and those of other protocols to begin developing universal applications immediately on your network.

We will demonstrate Reach's potential by building and launching a blockchain-enforced information trade onto your network. Through a funded proof of concept Reach's will demonstrate its edge in the following areas:

1. Ease of development
2. Speed of development
3. Protocol agnosticism
4. Enhanced safety
5. Reduced development cost

As the **first step** to becoming a Preferred Reach Platform Partner, we will build a proof of concept (POC) for your platform. The POC will establish the technical integrations needed to later fully support a complete partnership. This POC will also serve the purpose of proving that Reach's abstraction will work with your platform's infrastructure.

Ultimately, our goal is for this proof of concept to serve as a foundation for a long-term multi-year partnership, aligned through a shared incentive to see your platform and token reach their full potential.

2. Executive Team

Chris Swenor (co-founder and CEO):

- +15 years of experience managing technical teams around the world
- + 2 years of experience in blockchain ventures
- Successfully “exited” multiple start-ups; including 25 person software services company, which developed and launched numerous successful applications over 5 years
- Previously a mentor at Harvard Business School, TechStars, and MassChallenge.

  LinkedIn: https://www.linkedin.com/in/chrisswenor/

Jay McCarthy (co-founder and CTO)

- Ph.D. in Computer Science from Brown University.
- Assistant Professor: Starting in 2008, he was an assistant professor of computer science, first at Brigham Young University, and then at Vassar College
- Associate Professor: Currently, computer science professor at UMass Lowell.
- Expertise: Since 2005, he has conducted research in formal verification, programming language design and implementation, and cryptographic protocol analysis.
- Accomplishments: + 25 peer-reviewed publications, +400 citations. Co-creator of the Racket programming language and winner of the 2018 SIGPLAN Programming Languages Software Award, an honor shared only with Scala, V8, Z3, GCC, Coq, the Jikes RVM, GHC, and LLVM.
- Community Leader: Not just a researcher, he is an active creator of and contributor to open source projects, with over 150 repositories on GitHub, over 500 stars, over 250 followers, across 13 different languages ranging from C to Haskell and Coq.

  LinkedIn: https://www.linkedin.com/in/jay-mccarthy-34a6b95/
  GitHub: http://github.com/jeapostrophe/
  Google Scholar: https://scholar.google.com/citations?user=imyMtsYAAAAJ&hl=en
  Publications: https://jeapostrophe.github.io/home/ (local copies of PDFs available)

3. Problem Statement

Cardano's mission is to "*ensure the positive advancement of the Cardano protocol, while also contributing to the positive advancement of blockchain as a world-changing technology. In everything we do, we enable, empower, facilitate and accelerate progress in the blockchain space."*

However, building decentralized applications on a blockchain protocol is still a challenge. In addition to this is the added challenge new protocols face with growing their developer communities.

A Multifaceted Problem

1. **Developer Lockout**
   DApp development requires detailed knowledge of particular networks and cryptographic techniques. This knowledge is not pervasive among traditional full-stack developers and it is too expensive and risky to acquire because no blockchain platform is perceived as "safe" or dominant. Thus, traditional developers are locked out of the market.

2. **Component Consistency**
   DApps involve many cooperating components to deploy successfully and safely. The details of the smart contract, front-ends, and correctness proofs must all agree on every level, from the protocol details (like data formats and method API) to the logical operations of the program.

3. **High Cost of Correctness**
   DApps require a high standard of correctness because mistakes can automatically put great financial resources at risk (c.f. DAO hack).

What this Means

These problems are pervasive across the entire blockchain space but are particularly poignant for newer blockchain networks.

> *EXAMPLE*
> *DApps require a high standard of correctness, due to the fear of attacks. Because of this, considerable effort must be invested to ensure correctness in code. . However, this is more difficult due to the many cooperating components in a DApp. Not to mention, it is too risky for existing DApp authors to switch from a dominant chain, like Ethereum, to a new chain.*

Such challenges prevalent in blockchain prevent new entrants from joining the community, and even when they do, they are far more likely to connect with a dominant chain over a new chain.

How Reach will Help:

Through a Reach integration, new protocols can:

1. **Acquire Developers:** Rapidly increase developer following because Reach makes it easier for new developers to join the blockchain space and has 2100 developers to bring to your protocol with a community that is on average growing 10% week-after-week.
2. **Component Consistency:** Developers can ensure that all components of their DApp are consistent and can communicate together, without needing to heavily invest in learning the details of your chain.
3. **Assure Correctness:** Developers can experiment with a variety of networks, collaborate, and have the assurance that their software is correct and protected from attacks.
4. **Increase Mindshare:** Blockchain must increase its mindshare in the DApp marketplace and will, therefore, benefit from being a platform of choice for building secure and reliable DApps.

We believe Reach is the solution to this problem. In this proposal, we will build a blockchain-enforced information trade DApp as a proof of concept on how Reach can successfully address the key challenges described above.

4. Solution Approach:

Proposal Proof of Concept: Reach will build a blockchain-enforced information trade application as a proof of concept of Reach's ability to seamlessly work with your protocol; a first step to demonstrating Reach's ability to support your platform's mission.

The Reach platform provides three essential services via a domain-specific language (DSL) for specifying DApps and a specialized compiler that projects the specification into each of the output components while performing automatic verification of correctness properties.

1. ***Ease of Network Adoption***
   Our DSL uses a subset of JavaScript to specify the DApp at an abstract level involving potentially infinite interactions between individual participants performing finite computations over consensus data in the presence of rely-guarantee assertions. This enables us to abstract over particular blockchain networks, while remaining faithful to the interfaces offered by actual networks.

Since Reach abstracts the low-level details of the blockchain network, it enables DApp authors to develop on one platform (or purely in simulation) and deploy on another network. This is especially valuable for platforms in development that would otherwise need to onboard developers and coax them into rewriting their software to use the new system.

There are many existing DApps that are not built on Cardano and would require significant effort and energy to port. The Reach platform offers a concrete migration path via its abstract model of backend networks; this will increase day-1 adoption of Cardano.

2. ***Ease of Component Coordination***
   The Reach compiler uses type-checking, A Normal-Form transformation, information-flow security, and end-point projection to derive each component correctly from the single specification.

3. ***Ease of Correctness***
   The compiler is integrated with a satisfiability-modulo-theories (SMT) theorem prover (e.g. Z3) to automatically check the correctness of the application via developer-specific predicates, as well as automatically generated properties. Reach enables formal guarantees to be embedded in the source code and automatically verified. Auditors of Reach programs can objectively assert the correctness of DApps by adding more guarantees to the source code and demonstrating that these new properties are met (or not). Anything that increases the trustworthiness of DApps will be a boon for the blockchain community, which is reeling from the lack of confidence inspired by high-profile attacks.

In summary, Reach builds on decades of research in formal verification, compilation, and optimization of cryptographic protocols. The robustness of these aforementioned techniques will be demonstrated in the building out of the POC blockchain-enforced information trade application on your platform. All of our software (Reach language and compiler) is open-source and freely available to your developer community.

We anticipate that scaling the results of this proof of concept into a full integration will allow us to bring three transformative impacts on the blockchain ecosystem:

- **Growth**: We expect dramatic growth in the size of Cardano's DApp marketplace.
- **Auditing Methodology**: We expect the development of an objective DApp auditing methodology.
- **Transition to NextGen Blockchain**: We expect an increased and easier transition to Cardano.

5. Detailed Technical Plan and Statement of Work

Project Task Outline: Reach Language and Compiler

5.1 Cardano Connector

Context

Reach works by compiling users' DApps to an intermediate representation based on a finite-state machine where transitions correspond to protocol actions. It assumes that a contract can control the state and mediate when transitions are permissible. We refer to the process that compiles the state machine to the particular chain, the connector. We will need to build a connector from the Reach intermediate language to Cardano's smart contract language.

Considerations

Cardano has multiple smart contract languages: Plutus Core, Plutus IR, Plutus Tx, and Marlowe. Part of this task will be to select the appropriate target language for Reach's connector. We presently have connectors that target high-level languages (like Go), connectors that target "medium"-level languages (like Solidity), and connectors that target low-level languages (like AVM assembly.) The choice of what level to target can greatly impact what the connector can do and its long-run maintainability. For example, Marlowe cannot do everything that Plutus Core can do, so it is a bad idea for us to target it. Plutus IR may be easier to work with, but if it is not documented with an externally obeyed interface, then it is not wise to use unless there is tight integration over time between the Cardano team and the Reach Cardano connector team.

Suggested Approach

We assume that we will target Plutus Tx and generate Haskell programs that are compiled by the Plutus compiler into Plutus Core. We are Haskell experts (Reach is implemented in Haskell) so it is within our existing experience to do this. There are some downsides to this though, such as requiring our users to have access to a Haskell compiler (perhaps by embedding it into the Reach connector) and generating high-level code, when our code is already at a low-level. We expect that if there is too much friction with this, then we will switch and directly compile Plutus Core without Plutus Tx.

5.2 Cardano Reach Runtime

Context

Reach enables developers to author front-ends for their DApps using JavaScript and traditional full-stack Web development tools. This is crucial to enabling a wide range of developers access to blockchain technology. The key ingredient of this is a standard client-side library that provides uniform access to different Reach compatible chains at runtime. Most of these connector-specific runtimes rely on special compilation artifacts custom to the particular consensus network, such as interface ABIs and so on, that are produced at the same time as compiled backend. The meat of this component is a JavaScript library that communicates to the consensus network nodes to launch, query, and interact with the ledger. For example, on Ethereum, this library uses Ethers to contact Ethereum nodes running e.g. geth.

Considerations

Cardano has different pieces presently than networks that Reach already supports. Specifically, based on our current understanding, Cardano uses the PAB, Chain Index, and Alonzo node as three separate components that are likely to be hosted together. In contrast, on a network like Ethereum, the Chain Index and Alonzo node are integrated together into the geth node and the PAB exists partially on the client-side and partially in the network-side; while on a network like Algorand, the Chain Index and Alonzo node are also separate components and the PAB exists locally through a Reach-produced JavaScript compilation target of the Algorand VM code. Part of this proof-of-concept will be to determine an architecture that makes sense for Reach and Cardano.

Suggested Approach

We expect that our initial version will allow work by having a client-side JavaScript runtime that is designed to speak to a PAB instance and aid in the creation of a hosted PAB instance. For example, the Reach compiler may produce a Plutus contract and a PAB Docker container with a standard interface, while the Reach runtime will know how to connect to that PAB instance and will provide instructions and automation for launching PAB Docker containers on cloud or other services. We expect that this will be a true proof-of-concept and that a future version will default to using the expected client-side/in-browser version of the PAB, but allow easy access to hosted PABs for more expert users.

Proof of Concept Deliverable

Within 3-months we will deliver a fully functioning blockchain-enforced information trade DApp running on Cardano. The code for this application can be found here on our GitHub.

This Proof of Concept will demonstrate the ability of Reach's abstraction to seamlessly work with Cardano. To fully understand Reach's competitive advantages in ease and speed of development, built-in safety, and cost-efficacy, several more Reach DApps will need to be built and demoed on Cardano. Our hope is that this POC will provide an opportunity to further showcase the full extent of Reach's capabilities through a full Reach-Cardano integration and partnership

6. Beyond POC: Preferred Platform Partnership

The Future of Blockchain

Reach is confident in our ability to deliver on not only the Proof of Concept but on our vision of the future of blockchain. We are looking to Cardano as a key long-term partner in making blockchain and DApps mainstream.

Following the successful completion of the POC, we expect your excitement towards the possibility of growing your ecosystem via the Reach Developer Community will lead us to formalize a long-term commitment to bring more and more developers creating on the blockchain through Cardano.

Building a Future-Proof Partnership

With businesses and industries as dynamic as ours, it becomes difficult to anticipate the specific benchmarks and obligations of the future. Things change day-to-day. To solve for this, instead of wasting effort to define the unknowable, Reach believes deeply in the power of aligned incentives and shared outcomes.

We see this happening via a **meaningful token grant that vests over the life of the partnership.**

Through a long-term token-funded partnership, Reach and Cardano will have a shared incentive to engage in activities and initiatives that are mutually beneficial and not based on some heavily prescribed legal document.

We anticipate that much of the value of the partnership will fall into two categories: Technical & Commercial.

Technical Integration

Technologically, a partnership would incentivize both parties to work closely on a continual basis to deliver on the promises to developers to make the creation of DApps on Reach and Cardano as accessible as possible.

As partners we will no longer be “porting” the current state of Reach to Cardano, instead, we will be continuing to maintain the connector as more and more features are added to Reach.

Roadmap Examples: Features we will be working on in 2021

- Migration and upgrade-enabled contracts as an automatic byproduct of using Reach
- Automatic derivation of zero-knowledge proof circuits as a basic primitive in Reach programs
- Verified foreign (i.e. non-Reach) contract interaction using rely-guarantee reasoning and runtime verification of the foreign party

While it is impossible to predict the extent and exact timing of our roadmap multiple years out, shared incentives ensure it is in both of our best interests that integrations are maintained and optimized across the Reach and Cardano platforms.

Commercial Incentives

As with the technological obligations, instead of attempting to anticipate a dynamic and unpredictable future, we believe that the partnership lends itself to create the proper incentives for Reach to promote Cardano’s partnership and platform to the entirety of our rapidly expanding developer network.

As a Preferred Reach Platform Partner, the entire Reach community will be introduced to Cardano and our marketing efforts will be oriented towards promoting Cardano as much as promoting Reach.

We anticipate this will manifest in a number of ways, particularly in documentation and marketing efforts.

Documentation:
Reach has extensive documentation that describes all aspects of the Reach language and development tools, including examples and deployment instructions for each network. As a partner, Cardano would be prioritized within our documentation.

Documentation as Marketing:
This documentation is not just instructional, but part of a marketing and developer outreach effort that includes weekly videos, walkthroughs, tutorials, and workshops.

Co-Marketing Initiatives:
In our joint efforts to raise awareness, on-board developers, and ensure they push DApps to production, there will be numerous opportunities to co-brand content, including events, hackathons, contests, speaking engagements, panels, etc. that mutually benefit our respective communities and shared prospect pools.

And More…

While we have outlined some possibilities, we fully anticipate that the opportunities to collaborate will extend far beyond what we are able to anticipate, however via this partnership, binding our fates together will enable Reach to commit the resources necessary to do what is best for the partnership, whatever those initiatives might be in the distant future.

7. Budget and Costs

The budget for this Proof of Concept consists of the labor costs for a single full-time Architect and three Engineers that work on Reach's standard library, runtime, and compiler. Please note these are US-Based engineers and are being paid market rate.

3-month proposed POC Budget:

Architect Full Time ($21,700/month)	$65,100
Engineers x3 ($13,500/month each) ● Standard Library ● Runtime ● Compiler	$121,500
TOTAL	$186,600

8. Reporting Obligations

During the term of the grant, we will provide quarterly progress & financial reports and discuss with the appropriate Cardano representative the progress of our Proof of Concept. We shall provide full access to all materials, documents, and information relating to this project to allow Cardano to monitor development, the use of the grant and/or verify the progress update.

The following individuals shall be the Project lead representatives for the purposes of this SOW who will be the primary contact and Project leader and have full authority to make all decisions on behalf of the party which it represents. All communications between the parties under this SOW will be carried out through the Project leads set out below.

Awardee		[PARTNER]	
Name	Christopher Swenor	Name	
Title	CEO	Title	
Address	145 Brook Rd, Sharon, MA 02067, USA	Address	
Email	chris@reach.sh	Email	