Request for Proposal:

Ethers.js v5 to v6 Upgrade and SDK Agnosticism for Cow Protocol

Preamble:

Requests for proposals are not intended to be prescriptive or exhaustive. The community is encouraged to submit proposals that expand upon the ideas presented in this post. The scope of the project may change based on the proposals received. The primary intent of this document is to provide a starting point to achieve the outlined goals, and the final implementation may differ from the initial proposal.

All applications will follow the standard Grants DAO process. This request should not be interpreted as an offer.

Motivation

The Cow Protocol ecosystem relies heavily on Ethers.js for its operations. Upgrading from Ethers.js v5 to v6 is critical for leveraging modern JavaScript practices, improving performance, and ensuring future-proof compatibility. While Viem offers certain advantages, migrating a large codebase tightly coupled with Ethers.js to a different library would be resource-intensive. Therefore, this grant aims to fund:

1. Seamlessly transition Cow Protocol's projects to Ethers.js v6.

2. Make the Cow SDK agnostic to Ethereum libraries, enabling greater flexibility and choice in the future.

Benefits

1. Modernized Codebase:

2. Embrace ES Modules for better compatibility with contemporary tools.

3. Enhanced TypeScript support for robust development.

4. Simplified API and Maintenance:

5. Cleaner, more intuitive API in Ethers.js v6.

6. Elimination of deprecated patterns improves maintainability.

7. Optimized Performance and Security:

8. Faster execution and reduced bundle size.

9. Updated dependencies ensure better security and compliance.

10. New Features and Debugging:

11. Support for the latest Ethereum standards (e.g., EIP-1559).

12. Improved debugging tools for a smoother developer experience.

13. Future-Proofing:

14. Active maintenance of Ethers.js v6 ensures long-term viability.

Suggested Scope of Work

The upgrade should be implemented across the following Cow Protocol projects:

1. Cow SDK

2. Bump @cowprotocol/contracts

to the latest version.

- Transfer previously external functions (e.g., signOrder

, computeOrderUid

) into the SDK code.

- Update the SDK codebase to comply with Ethers.js v6 changes.

- Implement library-agnostic design, replacing Ethers.js-specific entities with native equivalents (BigNumber

→ bigint

).

- Create an abstraction layer for blockchain interactions, with adapters for both Ethers.js v6 and Viem.
- Release a new major version of the SDK.
- Cowswap
- Update @cowprotocol/cow-sdk

and ethers

to the latest versions.

- Refactor the codebase to align with the updated SDK and Ethers.js v6.
- Replace imports from @cowprotocol/contracts

with @cowprotocol/cow-sdk

.

- Apply similar changes to related sub-projects like explorer

.

1. Ethers.js and Viem Adapters
2. Develop modular adapters for blockchain interactions:
3. Contract calls, EIP-712 signatures, and other core functionalities.
4. Contract calls, EIP-712 signatures, and other core functionalities.
5. Implement adapters for Ethers.js v6 and Viem, ensuring compatibility with multiple libraries.

Suggested Deliverables

1. Updated Cow SDK (v6-compatible and library-agnostic) released as a new major version.
2. Refactored Cowswap and related projects to work seamlessly with the updated SDK and Ethers.js v6.
3. Fully functional Ethers.js and Viem adapters for blockchain interactions.
4. Comprehensive documentation for the new SDK and migration guides for developers.

Suggested Timeline

Phase 1: Initial Upgrades (2-3 Weeks)

- Update cow-sdk

and cowswap

to work with Ethers.js v6.

- Integrate necessary functions from @cowprotocol/contracts

into cow-sdk

.

Phase 2: Library-Agnostic Design (2-3 Weeks)

- Replace Ethers.js-specific entities with native alternatives in cow-sdk

.

- Build and test abstraction layers and modular adapters.

Phase 3: Testing and Deployment (2-3 Weeks)

- Thorough testing of updated projects in staging environments.

- Finalize documentation and release migration guides.

- Deploy and monitor updated versions.

Conclusion

This grant will modernize Cow Protocol's ecosystem, enhancing performance, security, and maintainability. By introducing library agnosticism, it also sets the stage for future flexibility, empowering developers to choose the best tools for their needs.

Values of Grants DAO and its Grants:

These values may evolve and are listed in no particular order:

- Open Source:

Integrations should be open source.

- Milestones

: Milestones should be attainable and well-defined to ensure easy verification of completion.

- Price Transparency

: Pricing should be broken down into optional and core metrics/deliverables to allow selective implementation.

- Sustainability:

Address the sustainability of deliverables (e.g., who will manage, maintain, and for how long, including associated costs).

- Simplicity

: Aim for simplicity. Completion is often more valuable than striving for perfection—except for critical components, which must meet the highest standards.

- Documentation

: Provide solid documentation to ensure that others can build on your work smoothly, where applicable.

- Flexibility

: We recognize that some processes require flexibility (e.g., adding new features, adapting to changes in technology or infrastructure). Open communication is encouraged to adapt to these changes. Scope extensions and pricing changes typically require a committee vote.

Call for action

Proposals should be submitted until January 31st 2025, using the standard [Grants Program template (deadline will be double checked with the GrantsDAO committee)](#). Ideally each application posts a reply in this thread, linking their application.

Selection process

The selection will be made by the GrantsDAO committee at their discretion. The committee will consider the above values, cost, timing, quality, and scope in their decision-making. Committee members may ask questions or make a decision independently. Purposfully, no expected funding is included in this RFP.

Currently, there are no official rejection criteria. If the forum discussion does not provide a clear outcome, an applicant can post their proposal to the [GrantsDAO Snapshot space](#) and request a committee vote, if needed.