Protocol upgrade process

The Sui protocol, framework, and execution engine are frequently extended to include new functionality and bug fixes. This functionality is added in the form of new code which is released to validator operators as part of our regular software releases. The Sui protocol, however, requires that all Sui validators agree about the results of executing each transaction.

This poses the following challenge: How do we release code that changes transaction execution, given that it is not possible to ensure that all operators upgrade their software at the same instant? Further, how do we ensure that all Sui transaction history can be replayed even after functionality has changed?

To solve this problem, Sui uses a process called protocol upgrades.

An outline of the process used for protocol upgrades includes the following steps:

Full nodes follow a similar process, however, they do not participate in voting. Instead, they follow the actions that validators recorded.

When validators switch to a new protocol version, they do so by recording the new version number in the special end-of-epoch transaction. Full nodes execute this transaction as they are replaying the chain history, and are thus able to switch to the new protocol version at the right time.

Not all new Sui functionality comes in the form of changes to the validator code. There are also changes to the Sui framework. For instance, Sui developers periodically add new native functions to the framework to expose new functionality to smart contracts. The process for framework updates is similar to protocol upgrades.

Instead of using feature flags, however, Sui objects are used to coordinate framework changes. The Sui framework is a special object with id 0x2 . The Move source code for the framework is built into the validator binary.

If the validator notices that its built-in framework is different from the framework in object 0x2 , it signals to the other validators that it would like to upgrade the framework to a new version. Just as with changes to ProtocolConfig , if enough validators agree to perform the upgrade, the new framework object is written at the end of the current epoch. Then, transactions that are executed in the new epoch use the new version of the framework.