Allow projects to use CalVer or SemVer

Status	RESOLVED
Blocking	
Outcome	Approved
Minutes Link	2020 09 17 TSC Minutes

Overview of Proposal

Allow projects to switch from SemVer (Semantic Versioning) for their versioning to CalVer (Calender Versioning) for their versioning.

Besu is different from a lot of the Hyperledger project in that it is beholden to a standard (Ethereum Mainnet compatibility) that it does not control. While it participates in co-ordination of that standard Besu does not have simple control over it like the other DLT projects do.

When working with public networks such as Mainnet there is a critical need to use the most current version of Besu, no matter what the semantic compatibility is. Not dealing with a breaking change is the same as dropping the service. Mainnet is fairly good about maintaining backwards compatibility, but there are other smaller changes such as P2P network protocol upgrades and RPC APIs that update at an infrequent cadence.

Besu feels that a CalVer versioning for the main DLT release itself would better signal how aligned we are to the continually moving target of mainnet compatibility. We think it will also more clearly communicate multiple release streams should not be expected.

Formal Proposal(s)

Update the Release Taxonomy to allow project to opt into using Calendar Versioning (CalVer) instead of Semantic Versioning MAJOR.MINOR.PATCH versioning.

Projects that opt in must meet these criteria:

- The project has a regular or semi-regular release schedule that is oriented around release dates
- If there are different points in time where different release processes are used, those need to be documented.
- The project has an established policy for introducing breaking changes
    - This policy must allow users sufficient time to adapt to the changes
    - This policy must include the channels used to communicate this (such as prominent placement in CHANGELOG files)
    - Some period of backward compatibility should exist.

For example, Besu's policy would be:

- Besu has triannual to quarterly major releases. These are numbered YY.M.0, such as 20.9.0.
    - These releases undergo a two week beta and two week RC cycle,
    - All supported public networks are synchronized and validated to stay in sync
- Besu has fortnightly patch releases. These are numbered YY.M.PATCH, such as 20.9.1
    - These releases come off of continuous integration releases
    - A smaller number of select networks are re-synchronized, and previously nodes are upgraded and observed for 24 hours prior to release.
- Breaking changes are only introduced in quarterly changes
    - These should have an entire quarter of notice, for example if notice of a breaking change is given in 20.2.4 then 20.5.0 should not implement the change but instead warn that 20.8.0 will implement the change
    - These must be placed prominently at the top of the CHANGELOG for the quarterly release and each patch release for the notice period should reference it.
- If Besu exposes external libraries (such as JNI libraries for cryptography), those should continue to use SemVer.
- Internal versioning of build modules will use CalVer, providing a clear delineation as to what is an API and what isn't.

Action Items

☑ Danno Ferrin to submit a corresponding PR to update the Release Taxonomy

Reviewed By

☑ Angelo De Caro

☑ Arnaud J Le Hors

☑ Christopher Ferris

☑ Dan Middleton

☑ Gari Singh

☑ Hart Montgomery
☑ Mark Wagner
☑ Nathan George
☑ Swetha Repakula
☑ Tracy Kuhrt
□ Troy Ronda