

Solidity

Apparatus implements an additional compile-time and runtime semantic which is called *solidity* (or being *solid*).

The solidity state is assigned per each Mechanism of the Machine individually. This is managed automatically, during the [enchain](#)ing operation. When the Mechanism is in a solid state, you are not allowed to perform any structural changes to your Subjects and Subjectives, i.e. you can't add or remove Traits, can't enable or disable Details. Spawning new Subjects and Subjectives within a solid Mechanism is also prohibited.

While comprising aforementioned limitations this uniform state provides for some very important features like direct memory access to Traits data (essentially eliminating the copying) and being able to operate on Subjects and Subjectives concurrently in a threaded manner.

In its turn if you've enchain to a solid Chain you would get solid Cursors only, essentially safeguarding your Mechanic iterating/operating.

The opposite of solidity is *liquidity* (or being *liquid*) which is the default behavior. If the liquid (non-solid) enchain is active on the Mechanism, it can't be made solid and vice versa.

From:
<http://turbanov.ru/wiki/> - **Turbopedia**

Permanent link:
<http://turbanov.ru/wiki/en/toolworks/docs/apparatus/solidity?rev=1635586468>

Last update: **2021/10/30 09:34**

