

Apparatus Features

The current list of features includes:

- ECS at its maximum. A complete Unreal Engine integration of the [data-oriented](#) workflow.
- Loyal to the ecosystem. Both C++ and Blueprint development is supported.
- Best of two worlds. UObject-based Details and UStruct-based Traits.
- Pedal to the metal. The Chunk-based data storage is hardware-first. Quite a bit of performance optimizations on top: Filter-based caching, fast bit-array masking, mainline Component indexing and more.
- Power at hand. [C++ Lambda](#)-based operations with an automatic argument delivery.
- Querying with ease. A versatile Component-including and Component-excluding filtering for the Mechanics.
- Bitwise [Flagmark](#) filtering support for boolean-based states.
- Straightforward editing. Assign and customize your [Trait/Detail](#) blocks right from the [Details Panel](#) in Unreal Editor.
- Multiple Details of the same class on a single Subjective are supported. All of the available Detail combinations are processed in the operating body.
- Inheritance at best. The Detail classes can be inherited for extra modularity and flexibility.
- Dedicated in-Editor user experience touches for some clear reading, validation and overall ease of use.
- Full concurrent (multi-threaded) iterating support with a dedicated compile-time [Solid](#) semantic.
- Ready to speak. An elegant and versatile custom [networking](#) solution built upon the Unreal's replication and RPC functionality.
- [Steady-ticking](#) implementation for some extra stability of your game frame-dependent logic.
- Thoroughly documented [API](#) with a dedicated [user manual wiki pages](#).

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

Permanent link:
<http://turbanov.ru/wiki/en/toolworks/docs/apparatus/features?rev=1673451794>

Last update: **2023/01/11 15:43**

