Mechanisms

Mechanisms are the main parts of the Machine which contain scoped Subject, Subjectives, Chunks and Belts. You enchain the matching Iterables via Mechanisms, so these also actually manage the Chains.

Each UE World, that has any Apparatus-specific entities is accompanied by a Mechanism automatically, you don't have to worry about that.

Blueprint Workflow

Mechanisms are managed automatically according to the current world object contexts.

C++ Workflow

To obtain a Mechanism for a World one would call API like:

```
AMechanism* Mechanism = UMachine::ObtainMechanism(GetWorld());
```

You can then use the instance to spawn a Subject for example:

```
FSubjectHandle Subject = Mechanism->SpawnSubject();
```

Creating your separate (possibly transient) Mechanisms is also supported and can be handy. You instantiate the Mechanism like you would any other UObject (NewObject<UMechanism>(...)):

```
AMechanism* Mechanism = NewObject<AMechanism>(GetTransientPackage());
```

Please, see the corresponding API reference for more details.