

Enchaining

Enchaining is the process of selecting a subset of Chunks (or Belts) based on a certain Filter criteria. Once enchained Belts and Chunks become locked, the structural changes become minimized for the sake of consistent iterating.

C++ Workflow

That's pretty basic, really. You don't create (instantiate) Chains manually but those are actually managed by the 🤖 [Machine class](#). Assuming you've already assembled the needed [Filter](#) all you have to do is call a global (static) Machine method named 🤖 [Enchain](#) passing it a filter and a reference for a unique Chain identifier to be filled.

```
int32 ChainId;  
UMachine::Enchain(Filter, ChainId);
```

You are now ready to 🤖 [obtain](#) the chain designated by this unique identifier and use its methods later:

```
FChain& Chain = UMachine::ObtainChain(ChainId);
```

Make sure you're actually assigning it to a reference (&), and not copying an entire chain.

You're now ready to [iterate](#) the resulting Chain.

From:

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

Permanent link:

<http://turbanov.ru/wiki/en/toolworks/docs/apparatus/enchaining?rev=1624043837>

Last update: **2021/06/18 22:17**

