

Code Pinning

Code Pinning mechanism allows codes to be pinned to the memory.

This way code does not have to be loaded to memory on each execution thus makes ~x40 performance .

This is an estimation, needs to be benchmarked.

Code pinning is done through native chain governance.

Proposal

PinCodesProposal

```
// PinCodesProposal gov proposal content type to pin a set of code ids in the // wasmvvm cache. message
PinCodesProposal { // Title is a short summary string title = 1 [ (gogoproto.moretags) = "yaml:\"title\"" ]; // Description is a
human readable text string description = 2 [ (gogoproto.moretags) = "yaml:\"description\"" ]; // CodeIDs references the new
WASM codes repeated uint64 code_ids = 3 [ (gogoproto.customname) = "CodeIDs", (gogoproto.moretags) =
"yaml:\"code_ids\"" ]; } reference
```

You can create the proposal using client:

```
wasmd tx gov submit-proposal pin-codes 1 --from wallet --title "Pin code 1" --description "Pin code 1 plss"
```

UnpinCodesProposal

You can unpin codes:

```
// UnpinCodesProposal gov proposal content type to unpin a set of code ids in // the wasmvvm cache. message
UnpinCodesProposal { // Title is a short summary string title = 1 [ (gogoproto.moretags) = "yaml:\"title\"" ]; // Description is a
human readable text string description = 2 [ (gogoproto.moretags) = "yaml:\"description\"" ]; // CodeIDs references the
WASM codes repeated uint64 code_ids = 3 [ (gogoproto.customname) = "CodeIDs", (gogoproto.moretags) =
"yaml:\"code_ids\"" ]; } reference
```

```
wasmd tx gov submit-proposal unpin-codes 1 --title "Unpin code 1" --description "Unpin code 1 plss" --from wallet Previous
Migration Next Testing * Proposal * * PinCodesProposal * * UnpinCodesProposal
```