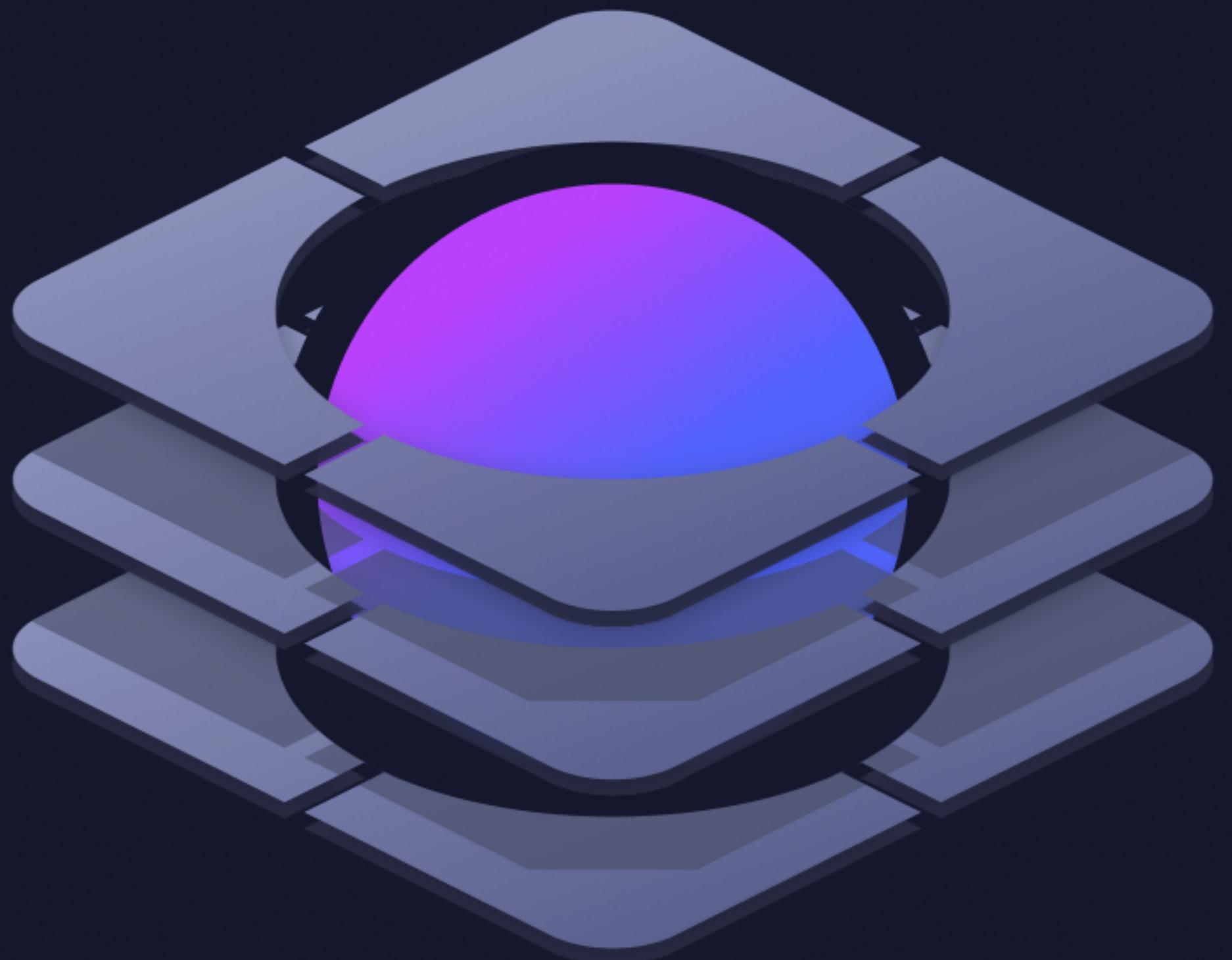


Cosmos SDK

Scalable Blockchain
Development



Jae Kwon

Founder of Cosmos

KVStore

KVStore

```
type KVStore interface {
    Store
    Get(key []byte) []byte
    Has(key []byte) bool
    Set(key, value []byte)
    Delete(key []byte)
    Iterator(start, end []byte) Iterator
    ReverseIterator(start, end []byte) Iterator
}
```

[cosmos-sdk/types/store.go](#)

Lines 91 to 121 in 3fc7200

Multiple KVStores

```
app.MountStoresIAVL(app.keyMain,  
app.keyAccount, app.keyIBC,  
app.keyStake, app.keySlashing)
```

[cosmos-sdk/examples/basecoin/app/app.go](https://github.com/cosmossdk/examples/blob/main/basecoin/app/app.go)

Line 90 in 3fc7200

iavlStore

```
func (st *iavlStore) Query(req  
abci.RequestQuery) (res  
abci.ResponseQuery) {
```

[cosmos-sdk/store/iavlstore.go](https://github.com/cosmossdk/store/iavlstore.go)

Line 135 in 3fc7200

go-amino

BaseAccount

```
type BaseAccount struct {
    Address    sdk.Address      `json:"address"`
    Coins      sdk.Coins        `json:"coins"`
    PubKey     crypto.PubKey   `json:"public_key"`
    Sequence   int64           `json:"sequence"`
}
```

[cosmos-sdk/x/auth/account.go](#)
lines of code in what commit

Register crypto

```
func MakeCodec() *wire.Codec {
    var cdc = wire.NewCodec()
    wire.RegisterCrypto(cdc) // Register crypto.
    sdk.RegisterWire(cdc)    // Register Msgs
    bank.RegisterWire(cdc)
    stake.RegisterWire(cdc)
    slashing.RegisterWire(cdc)
    ibc.RegisterWire(cdc)

    cdc.RegisterInterface((*auth.Account)(nil), nil)
    cdc.RegisterConcrete(&types.AppAccount{}, "basecoin/Account", nil)
    return cdc
}
```

[cosmos-sdk/examples/basecoin/app/app.go](https://github.com/cosmos-sdk/examples/blob/basecoin/app/app.go#L101)

Line 101 in d309abb

More on go-amino

github.com/tendermint/go-amino

Keys, Keepers, and Mappers

Hold a key — not a reference

```
type AccountMapper struct {
    // The (unexposed) key used to access the
    // store from the Context.
    key sdk.StoreKey
    // The prototypical Account concrete type.
    proto Account
    // wire codec for binary encoding of accounts.
    cdc *wire.Codec
}
```

[cosmos-sdk/x/auth/mapper.go](#)

Lines 14 to 24 in fc0e401

Define the accountMapper

```
app.accountMapper =  
auth.NewAccountMapper(  
    cdc,  
    app.keyAccount,           // target store  
&types.AppAccount{}, // prototype  
)
```

[cosmos-sdk/examples/basecoin/app/app.go](https://github.com/cosmos-sdk/examples/blob/basecoin/app/app.go#L65-L70)

Lines 65 to 70 in d309abb

Enable access with associated key

```
func (am AccountMapper) GetAccount(ctx sdk.Context,  
addr sdk.Address) Account {  
    store := ctx.KVStore(am.key)  
    bz := store.Get(addr)  
    if bz == nil { return nil }  
    acc := am.decodeAccount(bz)  
    return acc  
}
```

[cosmos-sdk/x/auth/mapper.go](#)

Lines 44 to 53 in fc0e401

Tx, Msg, Handler, and AnteHandler

StdTx

```
// A standard way to wrap a Msg
// with a Fee and Signatures.

type StdTx struct {
    Msg         sdk.Msg           `json:"msg"`
    Fee         StdFee            `json:"fee"`
    Signatures []StdSignature   `json:"signatures"`
}
```

[cosmos-sdk/x/auth/stdtx.go](#)

Lines 12 to 18 in fc0e401

Register Msgs

```
func MakeCodec() *wire.Codec {
    var cdc = wire.NewCodec()
    wire.RegisterCrypto(cdc) // Register crypto.
    sdk.RegisterWire(cdc)    // Register Msgs
    bank.RegisterWire(cdc)
    stake.RegisterWire(cdc)
    slashing.RegisterWire(cdc)
    ibc.RegisterWire(cdc)

    cdc.RegisterInterface((*auth.Account)(nil), nil)
    cdc.RegisterConcrete(&types.AppAccount{}, "basecoin/Account", nil)
    return cdc
}
```

[cosmos-sdk/examples/basecoin/app/app.go](https://github.com/cosmos-sdk/examples/blob/basecoin/app/app.go#L102-L106)

Lines 102 to 106 in d309abb

Register message routes

```
app.Router().  
    AddRoute("auth", auth.NewHandler(  
        app.accountMapper)).  
    AddRoute("bank", bank.NewHandler(  
        app.coinKeeper)).  
    AddRoute("ibc", ibc.NewHandler(  
        app.ibcMapper, app.coinKeeper)).  
    AddRoute("stake", stake.NewHandler(  
        app.stakeKeeper))
```

[cosmos-sdk/examples/basecoin/app/app.go](#)

Lines 78 to 83 in d309abb

EndBlocker

Process validator provisions

```
// XXX assuming in seconds, confirm
blockTime := ctx.BlockHeader().Time
if pool.InflationLastTime+blockTime >= 3600 {
    pool.InflationLastTime = blockTime
    pool = k.processProvisions(ctx)
}
```

[cosmos-sdk/x/stake/handler.go](#)

Lines 32 to 37 in 3fc7200

SDK - Staking Module

[github.com/cosmos/cosmos-sdk/tree/
develop/x/stake](https://github.com/cosmos/cosmos-sdk/tree/develop/x/stake)

Learn more

tendermint.com

cosmos.network

We're hiring!

COSMOS

INTERNET OF BLOCKCHAINS

