| **Cosmos SDK** | **Provenance** |
| --- | --- |
| Baseapp -  Ref - https://docs.cosmos.network/master/intro/sdk-app-architecture.html#abci  Baseapp is the boilerplate implementation of a Cosmos SDK application. It comes with an implementation of the ABCI to handle the connection with the underlying consensus engine. Typically, a Cosmos SDK application extends Baseapp by embedding it in [app.go](https://docs.cosmos.network/master/basics/app-anatomy.html#core-application-file).  ABCI -  Tendermint passes transactions to the application through an interface called the [ABCI](https://docs.tendermint.com/master/spec/abci/)  [(opens new window)](https://docs.tendermint.com/master/spec/abci/)  Note that - Tendermint only handles transaction bytes | Provenance rename the folder as “app”  App . go ->  In app.go they are not keeping all the cosmos codes, they are importing from cosmos github.  .i.e : bankkeeper "github.com/cosmos/cosmos-sdk/x/bank/keeper"  Provenance excluded all the test cases from their app folder. |
| Client - | Client - docs / Rosetta folders only  Missing lot of files and folders |
| Missing cmd folder | cmd |
| Codec  Amino\_codec.go - same as - <https://github.com/provenance-io/provenance/blob/v0.0.0/x/marker/types/codec.go>  But they commented all the code | Missing codec folder  https://github.com/provenance-io/provenance/blob/v0.0.0/x/marker/types/codec.go  compresses and decompresses large amounts of data  They removed codec functions. |
| Contrib - images migrate folders extra | Contrib |
| Cosmovisor  crypto | Cosmovisor / crypto folders missing |
| Docs | Docs multiple files folders are missing and new |
|  | internal / networks folders new added |
|  | Proto folder new files added |
|  | protoBindings folder new added |
| scripts | Scripts lot of new files |
| Server  Simapp  Snapshots  std  store  telemetry  tests  types  Version | Server  Simapp  Snapshots  std  store  telemetry  tests  types  Version  Folders missing |
| third\_party/proto | third\_party/proto New files added |
| X | X lot of files missing (deleted ) |
|  | gon.json  Sims-state-listening.mk  sims.mk |
| Dockerfile | Dockerfile missing |
| Makefile | Makefile lot of changes |
| docker-compose.yml | Config.yml some changes |
| build.sh | Build.sh file missing |
| buf.yaml | Buf.work.yaml changes |

Important Links -

Inherited Modules from Cosmos -

* <https://docs.provenance.io/modules/inherited-modules>
* [Provenance Launches New Public, Decentralized Blockchain for Financial Services | Business Wire](https://www.businesswire.com/news/home/20210519005633/en/Provenance-Launches-New-Public-Decentralized-Blockchain-for-Financial-Services)

Built using Cosmos SDK/Tendermint, the new open-source Provenance reduces the need for third-party intermediation, thus drastically reducing costs and freeing up capital in financial transactions.

**COSMOS -**

* <https://cosmos.network/>
* <https://tutorials.cosmos.network/>
* The Internet of Blockchains (IOB) Dual-layer Blockchain Ecosystem
* There are currently more than 265 apps and services in the Cosmos network, including Binance Chain, Terra, Crypto , Ethereum, Bicoin.
* Cosmos apps and services connect using IBC, the Inter-Blockchain Communication protocol. This innovation enables you to **freely exchange assets and data across sovereign**
* The Cosmos SDK (opens new window)is an **open-source framework for building multi-asset public Proof-of-Stake** (PoS) blockchains, like the Cosmos Hub, as well
* Cosmos Hub is a kind of **Marketplace , Security provider , Router** (to connect chains by establishing IBC connections with compatible chains and operating decentralized bridges with chains like Ethereum and Bitcoin.)
* **IBC** - Inter-Blockchain Communication Protocol -> allows independent blockchains to directly communicate and trade assets
* Each individual blockchain maintain control of its own governance, but is interportable with other blockchains on the network.
* Blockchain that do not utilize **BFT**(Byzantine fault-tolerance) **algorithm** can be connected to the cosmos network via adaptor blockchains.
* Cosmos is not designed for one particular use case, but to be adaptable to suit many different use cases.

**What is Cosmos ?**

* Cosmos is a project with an ambitious mission: To create a network of distributed ledgers (blockchains) that will solve long-standing problems in the cryptocurrency and blockchain communities.
* The Cosmos network consists of many independent, parallel blockchains, called zones, each powered by classical Byzantine fault-tolerant (BFT) consensus protocols like Tendermint. Some zones act as hubs with respect to other zones, allowing many zones to interoperate through a shared hub. The architecture is a more general application of the Bitcoin sidechains concept, using classic BFT and Proof-of-Stake algorithms, instead of Proof-of-Work.

References - <https://www.youtube.com/watch?v=3ohaGWMYgSY>

**Provenance Blockchain -**

* <https://docs.provenance.io/>
* [https://www.provenance.io/](https://docs.provenance.io/)
* Provenance is designed and developed to **support financial service industry** needs by providing a ledger, registry, and **exchange across multiple financial assets and markets**. The new, public, decentralized Provenance replaces the prior permissioned Provenance blockchain **launched in 2018**
* The Provenance Blockchain is **based on the Cosmos SDK** and the **Tendermint blockchain**
* **Tendermint** - Tendermint Core is a blockchain application platform; it provides the equivalent of a web-server, database, and supporting libraries for blockchain applications written in any programming language. Like a web-server serving web applications, Tendermint serves blockchain applications.

**Cosmos 40.0 fork -**

- Built using Cosmos SDK/Tendermint, the new open-source Provenance reduces the need for third-party intermediation,

thus drastically reducing costs and freeing up capital in financial transactions

--------------------------------------------------

Git repos ->

<https://github.com/cosmos/cosmos-sdk>

<https://github.com/provenance-io/provenance>

smart contract -

[https://docs.provenance.io/modules/provwasm-smart-contract](https://docs.provenance.io/modules/provwasm-smart-contracts)

-------------------------------------------------------------------------------------------------------------

Important Links -

Provenance Launches New Public, Decentralized Blockchain for Financial Services | Business Wire

Built using Cosmos SDK/Tendermint, the new open-source Provenance reduces the need for third-party intermediation, thus drastically reducing costs and freeing up capital in financial transactions.

<https://docs.cosmos.network/master/intro/sdk-app-architecture.html#abci>

CosmWasm -

CosmWasm is written as a module that can plug into the Cosmos SDK. This means that anyone currently building a blockchain using the Cosmos SDK can quickly and easily add CosmWasm smart contracting support to their chain

ProvWasm

Smart Contracts

Web assembly (Wasm) smart contracts for the Provenance Blockchain.

The CosmWasm VM only integrates with the banking and staking modules by default. However, it allows for adding custom encoders and queriers.

A package was added in the Provenance Blockchain that contain extensions for the following Provenance Blockchain modules:

Account

Marker

Name

Metadata

-------------------------------------------------------------------------------------------------------------

**- NOTE: Not yet implemented**

**These extensions are compiled into the blockchain at build time.**  
  
**In file** **app.go  
  
Line number 90 to 114, provenance self modules**

**Line number 123 to 136 added home default home directories for the application daemon.**

**switch to provenance legacy migration cmd -**

legacy "github.com/provenance-io/provenance/internal/legacy"

**Add metadata modules -**

"github.com/provenance-io/provenance/x/metadata"

metadatakeeper "github.com/provenance-io/provenance/x/metadata/keeper"

metadatatypes "github.com/provenance-io/provenance/x/metadata/types"

**Change the prefix module for accounts -**

Provenance Blockchain testnet addresses begin with **tp** whereas mainnet addresses begin with **pb**

766 - paramsKeeper.Subspace(markertypes.ModuleName)

**Marker** - A special type of Token represented by a contract that maintains an ownership structure and an optional collection of assets.

<https://docs.provenance.io/modules/inherited-modules>