**FAC Solution Summary**

|  |  |
| --- | --- |
| **Topic** | **FAC Solution** |
| Industry / Market | Financial Services / Mutual Fund Servicing |
| Value Proposition | Significant reduction in operating costs within the fund operations value chain through shared data, removing duplicated process as well as the need for reconciliations and messaging protocols |
| Primary use cases | **Launch:** Fund managers can launch tokenized funds onto the network via fund servicing nodes.  **Transact:** Investors can transact with fund servicers to buy and sell tokenized shares in funds.  **Settle:** Atomic settlement on ledger between investor and fund results in immediate finality.  **Maintain Register:** Primary register of fund shareholders maintained on ledger.  **Distribute income:** Funds can distribute income to investors with tokenized cash. |
| Blockchain Platform | Corda Enterprise (v4.5) |
| Rationale for this choice | Corda Enterprise is designed to meet the needs of financial organizations for privacy, performance and scalability. It is supported by R3 and a growing ecosystem of developers and service providers. |
| Development Language | Java/Kotlin |
| Cryptographic suite | **Node identity and network map:** Pure EdDSA using the ed25519 curve and SHA-512. **Certificate authorities and TLS:** ECDSA using the NIST P-256 curve (secp256r1) and SHA-256 (NIST recommended and HSM compatible) |
| Network Design | FAC is a private permissioned network |
| Nodes | Fund Manager (multiple)  Fund Servicing (multiple)  Investor/Distributor (multiple)  Cash Exchange (single)  Asset Exchange (single)  Regulator (single)  Business Network Operator (BNO) (single) |
| Participant Access | Participants operate their own nodes and are permissioned onto the network by the Business Network Operator. |
| User Authentication | FAC nodes operate a role based user access control service using an Open LDAP service. |
| User interaction | General access by web browser based GUI. Sone nodes can be remotely operated by API |
| Network Services | The BNO operates Corda generic services for the notary, network directory and network management.  Third party providers will operate exchanges, KYC and digital custody services on the network |
| Messaging Layer | Corda uses point to point messaging instead of global broadcast. The messaging layer is AMQP/1.0 over TLS between nodes which is currently implemented using Apache Artemis, an embeddable message queue broker. Building on established MQ protocols gives us features like persistence to disk, automatic delivery retries with back off and dead-letter routing, security, large message streaming and so on. |
| Integration | The FAC Cordapp supports an API that allows integration to third party systems for investor servicing, token issuance and reconciliation. The FAC API layer uses an AMQP broker and HTTP calls to interact with third party systems. |
| Consensus mechanism | Transactions are validated by their participants and then checked by an independent network notary service to prevent double spend before committing to the chain.  A standard Corda non-validating notary service is used. In production this will be operated by the BNO as a centralized high availability cluster using a Raft consensus mechanism. |
| Ledger | Unlike other blockchain solutions, in Corda there is **no single block chain**. Instead, each node maintains its own record of the transactions it has participated in. No node contains the entire blockchain.  Corda uses a UTXO (unspent transaction output) model where every state on the ledger is immutable. The ledger evolves over time by applying transactions to change states |
| Value on Ledger | Both cash and fund assets are tokenized on ledger using a solution based on the Corda token SDK. FAC issues fungible tokens for both cash and assets. This supports atomic settlement of transactions providing immediate finality. Corda uses a UTXO model to represent |
| Smart Contracts | FAC uses Corda contracts to validate transactions. In the future there is a possibility to introduce fund specific smart contracts that enforce the dealing terms and legal contracts for fund investors. |
| Data Privacy | Personal information is held off chain on the investors node and syndicated on a case by case basis to other nodes permissioned to receive it. |
| Key Management | In production node operators will have the option of an HSM key management solution. FAC intends to support third party key management / custody solutions when these are provided for Corda. |
| Node technology stack | Ubuntu  Corda  PostGreSQL  Java  Webpack/ReactJS/MaterialUI  Jasper Reports  Open LDAP |
| Database | FAC uses a PostgreSQL database. Data Schema’s are directly addressable via JDBC. Corda chain data (states) are stored in a reserved section of the databased referred to as the vault. Corda uses object serialization and stores data in the database as binary objects. FAC also uses Corda Queryable states to parallel save data to SQL tables for analytical and reporting purposes. |
| How is resilience achieved? | Corda is built on tried and tested technologies such as Java Virtual Machine and SQL, it supports the use of commercial RDBMSs and Cloud. This makes it possible to utilise existing approaches to ensuring nodes operate with high availability and redundant capability.  Should a node fail the flow framework guarantees atomicity of processing incoming events. This means that a flow or the node may be stopped at any time, even during processing of an event and on restart the node will reconstruct the correct state of the flows and will proceed as if nothing happened. |
| Deployment options | Container based deployment on cloud servers (Microsoft Azure).  For web GUI access the following firewall ports to be opened: 9090, 9080, 9070, 8090, 8080, 80, 17702, 17604, 17606, 17602,17706, 17704,17722,17718,17710 |

**For more information** please contact Chris Baldwin, FAC Delivery Manager, chris.baldwin@fundadminchain.com