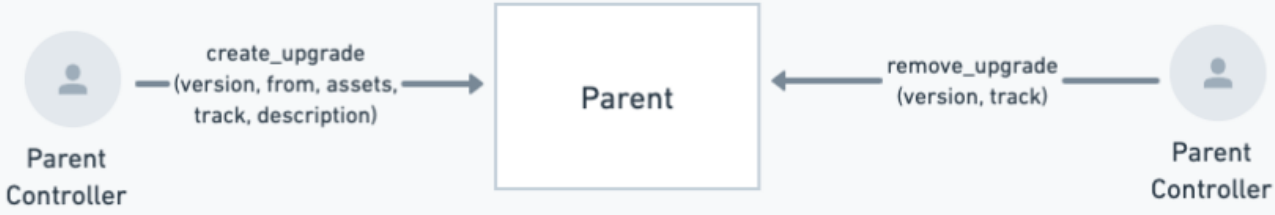
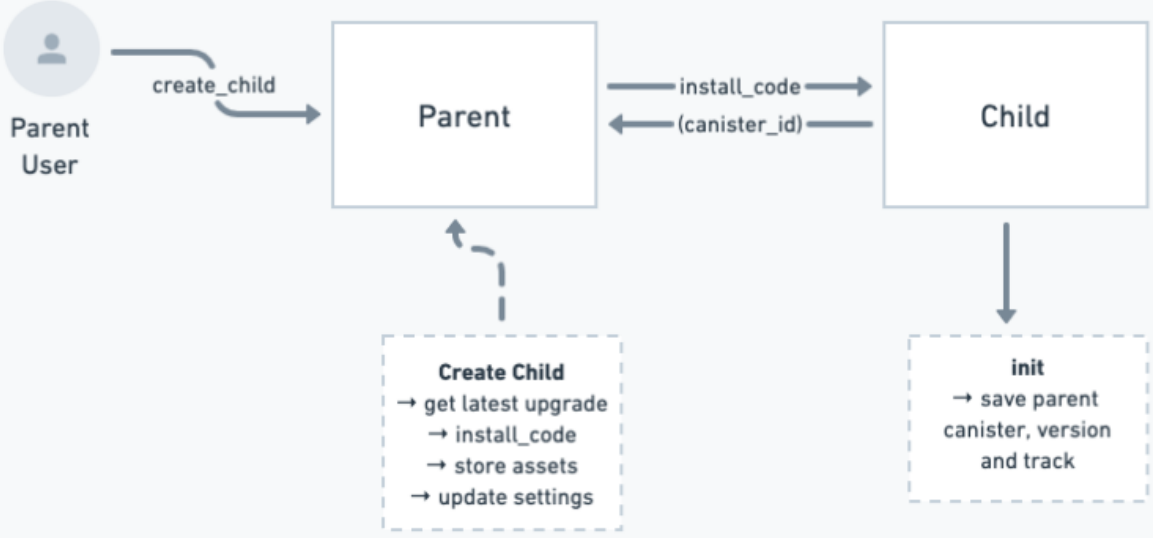
Summary

1. The system has a parent canister that has two functions:
   a. Create new community canisters
   b. Coordinate upgrades
2. To create a new community users use the parent canister that creates a new child canister for them. That canister is self-contained (ie. has both backend and frontend). and will eventually follow NFT standards to support easier management and transfer of ownership.
3. Community canisters support decentralized logins with Ethereum and Solana wallets. Users can sign a message with the wallet to give access to a new randomly generated principal that is stored locally in their browser. The can do this again to login from another computer.
4. The parent canister can propose upgrades to child canisters and the owners of the child canisters can choose to upgrade or not. To avoid locked in the parent canister supports multiple upgrade tracks so users can choose the upgrade track the want to follow. When the parent canister is decentralized propose upgrades will be done through a DAO vote.

Create Upgrade (Parent)

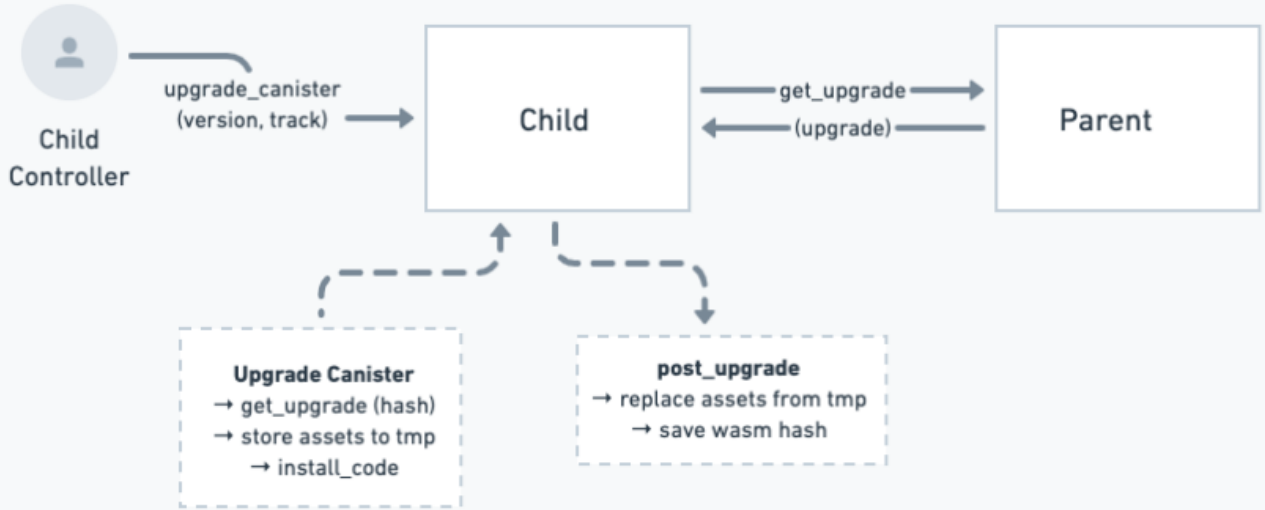


Create Community (Parent)



Get Next Upgrade (Child)

Upgrade Canister (Child)



Ethereum / Solana

(wallet login)

sign_message
→ Creates a random private key for a new principal and is saved locally
→ The user signs a message with that principal with their Ethereum or Solana wallets
→ The local principal will corresponds to the Ethereum or Solana address

update_profile
→ Checks that Ethereum or Solana signature
→ If the signature is correct it assigns the local principal to the user profile
→ The local principal is used for validation in update calls
→ Can be called again to sign in with a different computer