



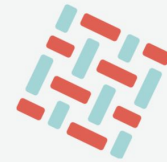
Applying Blockchain to Energy Delivery Systems

Anthony Cosimo, Dakota Moore, Katie Ringgenberg, Keegan Bloedel, Steven Rein, and Jacob Dawson



Integration of Blockchain with API

- Selecting identity from wallet
- Connecting to network gateway
- Accessing network
- Constructing request
- Submitting transaction
- Processing response
- All works locally, but will it be just easy after deploying?



HYPERLEDGER
FABRIC



Technical Challenges We Have Faced

```
// Obtain the smart contract with which our application wants to interact
const wallet = await Wallets.newFileSystemWallet(walletDirectoryPath);
const gatewayOptions: GatewayOptions = {
  identity: 'user@example.org', // Previously imported identity
  wallet,
};
const gateway = new Gateway();
await gateway.connect(commonConnectionProfile, gatewayOptions);
const network = await gateway.getNetwork(channelName);
const contract = network.getContract(chaincodeId);

// Submit transactions for the smart contract
const args = [arg1, arg2];
const submitResult = await contract.submitTransaction("transactionName", ...args);

// Evaluate queries for the smart contract
const evalResult = await contract.evaluateTransaction("transactionName", ...args);

// Create and submit transactions for the smart contract with transient data
const transientResult = await contract.createTransaction(transactionName)
  .setTransient(privateData)
  .submit(arg1, arg2);
```

- Initial integration between API and Blockchain Network
 - Solution: Hyperledger Fabric Node SDK
- Hyperledger Composer is deprecated
 - Composer is a collaboration tool for building blockchain business networks



Technical Challenges We Expect

- Deploying blockchain network to virtual machines.
- Configuring networking between the virtual machines.
- Piping test data from virtual devices provided by PowerCyber into the blockchain network.
 - Ideal: Execute smart contracts from PowerCyber devices
 - Not ideal: Having the API act as a middle man to execute smart contracts

