# Soluções Open Source para Dores CBDC Restantes

## 1. MIT OpenCBDC (Project Hamilton)

### Status: RESOLVE 3/4 dores restantes

**Performance**: 1.7M TPS com two-phase commit vs 125 TPS do Drex

**Arquitetura**:

```
COMPONENTE        | DREX ATUAL   | OPENCBDC     | MELHORIA
-------------------|---------------|---------------|----------
Transaction Processor| 125 TPS      | 1,700,000 TPS | 13,600x
Privacy Model       | ZK+TEE       | UHS Hash      | Simples
Authority Control   | Limitado     | Full Control  | ✅
Consensus          | QBFT (5s)    | 2PC (<100ms)  | 50x
```

✅ **RESOLVE**: B1 (Controle de Autoridade), C2 (SLA), D2 (Segregação)

## 2. Digital Asset DAML CBDC

### Status: Smart Contract Compliance

**Features**:

- Privacy-preserving programmable money
- Built-in compliance rules
- Interoperability framework

✅ **RESOLVE**: B3 (Auditoria compliance automática)

## 3. Hyperledger Fabric + Idemix

### Status: Privacy + Auditability

**Capabilities**:

- Zero-knowledge identity proofs
- Selective disclosure
- Regulatory oversight built-in

✅ **RESOLVE**: B1 (Privacy vs Authority), D1 (Threat model)

## 4. Consensys Quorum + Tessera

### Status: Enterprise Privacy

**Architecture**:

- Private state channels
- Regulator node access
- Transaction-level permissions

✅ **RESOLVE**: D2 (Role-based access), B3 (Audit trails)

## Implementações de Referência:

### MIT OpenCBDC Core (C++)

```cpp
// Authority Override for Emergency Actions
class AuthorityController {
  bool canOverride(const Transaction& tx, const Authority& auth) {
    return auth.hasEmergencyPowers() &&
        tx.requiresRegulatorIntervention();
  }

  void executeOverride(const Account& account,
            const Amount& amount,
            const string& justification) {
    // Bypass normal privacy constraints for regulatory action
    auditLog.record(AuthorityAction{account, amount, justification});
    ledger.forceTransfer(account, centralBankAccount, amount);
  }
};
```

### DAML Privacy Contract

```haskell
```

```
template CBDCToken
 with
  issuer : Party    -- Central Bank
  owner : Party     -- Current holder
  amount : Decimal
  regulatorView : Bool -- Can regulator see this?
 where
  signatory issuer, owner
  observer if regulatorView then [regulator] else []

  choice Transfer : ContractId CBDCToken
   with
    newOwner : Party
    withRegulatorOversight : Bool
   controller owner
   do
    create this with
     owner = newOwner
     regulatorView = withRegulatorOversight
```

## Score de Resolução por Solução:

| SOLUÇÃO | B1 | B3 | D1 | D2 | TOTAL |
|---|---|---|---|---|---|
| **MIT OpenCBDC** | ✅ | ✅ | ✅ | ✅ | 4/4 |
| **DAML CBDC** | ✅ | ✅ | × | ✅ | 3/4 |
| **Hyperledger** | ✅ | ✅ | ✅ | × | 3/4 |
| **Quorum+Tessera** | ✅ | ✅ | × | ✅ | 3/4 |

## Recomendação: Hybrid Architecture

**Base**: MIT OpenCBDC (performance core)

**Privacy**: Hyperledger Fabric (regulatory compliance)

**Smart Contracts**: DAML (programmable compliance)

**Monitoring**: Custom authority override layer