

Appendix B - DeFacto Standards

- de facto Standards Bodies
- Amazon
- Apache Software Foundation (ASF)
- Apache: Log4cxx
- Apache: Log4j
- Apache: log4scala
- Apache: log4net
- Apache: log4php
- Apple
- Apple: Darwin
- Apple: iOS
- Apple: MacOS
- Bitcoin
- Bitcoin: Bitcoin Improvement Proposals (BIPs)
- BIP 0011 - M-of-N Standard Transactions
- BIP 0013 - Address Format for pay-to-script-hash
- BIP 0014 - Protocol Version and User Agent
- BIP 0016 - Pay to Script Hash (soft fork)
- BIP 0021 - URI Scheme
- BIP 0022 - getblocktemplate - Fundamentals
- BIP 0023 - getblocktemplate - Pooled Mining
- BIP 0030 - Duplicate transactions (soft fork)
- BIP 0031 - Pong message
- BIP 0034 - Block v2, Height in Coinbase (soft fork)
- BIP 0035 - mempool message
- BIP 0037 - Connection Bloom filtering
- BIP 0042 - A finite monetary supply for Bitcoin (soft fork)
- BIP 0061 - Reject P2P message
- BIP 0065 - OP_CHECKLOCKTIMEVERIFY (soft fork)
- BIP 0068 - Relative lock-time using consensus-enforced sequence numbers (soft fork)
- BIP 0070 - Payment Protocol
- BIP 0071 - Payment Protocol MIME types
- BIP 0072 - bitcoin: uri extensions for Payment Protocol
- BIP 0073 - Use "Accept" header for response type negotiation with Payment Request URLs
- BIP 0091 - Reduced threshold Segwit MASF (soft fork)
- BIP 0112 - CHECKSEQUENCEVERIFY (soft fork)
- BIP 0113 - Median time-past as endpoint for lock-time calculations (soft fork)
- BIP 0137 - Signatures of Messages using Private Keys
- BIP 0141 - Segregated Witness (Consensus layer) (soft fork)
- BIP 0143 - Transaction Signature Verification for Version 0 Witness Program (soft fork)
- BIP 0144 - Segregated Witness (Peer Services)
- BIP 0145 - getblocktemplate Updates for Segregated Witness
- BIP 0147 - Dealing with dummy stack element malleability (soft fork)
- BIP 0148 - Mandatory activation of segwit deployment (soft fork)

- [Bitcoin: Developer's Guidance](#)
- [Bitcoin: Guide 1 Blockchain](#)
- [Bitcoin: Guide 2 Transactions](#)
- [Bitcoin: Guide 3 Contracts](#)
- [Bitcoin: Guide 4 Wallets](#)
- [Bitcoin: Guide 5 Payment Processing Guide](#)
- [Bitcoin: Guide 6 Operating Modes](#)
- [Bitcoin: Guide 7 Peer-to-Peer Networks](#)
- [Bitcoin: Guide 8 Mining](#)
- [Bitcoin: Bitcoinj Developer's Documentation](#)
- [Consortium for Information & Software Quality \(CISQ\)](#)
- [Ethereum](#)
- [ethereum_solidity](#)
- [ethereum_vm](#)
- [Ethereum: Ethereum Improvement Proposals \(EIPs\)](#)
- [EIP 20: ERC-20 Token Standard](#)
- [EIP 55: Mixed-case checksum address encoding](#)
- [EIP 137: Ethereum Domain Name Service - Specification](#)
- [EIP 141: Designated invalid EVM instruction](#)
- [EIP 155: Simple replay attack protection](#)
- [EIP 162: Initial ENS Hash Registrar](#)
- [EIP 165: ERC-165 Standard Interface Detection](#)
- [EIP 181: ENS support for reverse resolution of Ethereum addresses](#)
- [EIP 190: Ethereum Smart Contract Packaging Standard](#)
- [EIP 191: Signed Data Standard \(DRAFT\)](#)
- [EIP 211: New opcodes: RETURNDATASIZE and RETURNDATACOPY](#)
- [EIP 214: New opcode STATICCALL](#)
- [EIP 721: ERC-721 Non-Fungible Token Standard](#)
- [EIP 777: ERC-777 Token Standard](#)
- [EIP 1167: Minimal Proxy Contract](#)
- [EIP 1820: Pseudo-introspection Registry Contract](#)
- [EIP 107: safe "eth_sendTransaction" authorization via html popup \(DRAFT\)](#)
- [EIP 234: `blockHash` to JSON-RPC filter options \(DRAFT\)](#)
- [EIP 695: Create `eth\\_chainId` method for JSON-RPC \(DRAFT\)](#)
- [EIP 712: Ethereum typed structured data hashing and signing \(DRAFT\)](#)
- [EIP 758: ERC-NN Subscriptions and filters for completed transactions \(DRAFT\)](#)
- [EIP 1102: Opt-in account exposure \(DRAFT\)](#)
- [EIP 1186: RPC-Method to get Merkle Proofs - eth_getProof \(DRAFT\)](#)
- [EIP 1193: Ethereum Provider JavaScript API \(DRAFT\)](#)
- [EIP 1474: Remote Procedure Call \(RPC\) specification \(DRAFT\)](#)
- [EIP 1767: GraphQL interface to Ethereum node data \(DRAFT\)](#)
- [EIP 1803: ERC-NN Rename opcodes for clarity \(DRAFT\)](#)
- [EIP 1898: ERC-NN Add `blockHash` to JSON-RPC methods which accept a default block parameter \(DRAFT\)](#)
- [Ethereum: Clients](#)
- [Ethereum: cpp Project](#)
- [Ethereum: Ethereumh Project](#)
- [Ethereum: Ethereumjs-lib Project](#)

- Ethereum: Ethereum_j Project
- Ethereum: Go-ethereum Project
- Ethereum: Parity Project
- Ethereum: Pyethapp Project
- Ethereum: Ruby-ethereum Project
- Google
- Google: Android
- Google: Go (software language)
- Google: gRPC
- Google: Protocol Buffers
- IOTA
- Linux Foundation
- Linux Foundation: Hyperledger
- Linux Foundation: OpenJS Foundation
- Kubernetes
- Node.js
- Linux Foundation: Open Middleware Agnostic Messaging API (OpenMAMA)
- Linux Foundation: Open Messaging
- ISO/IEC The Linux Standard Base 5 Specification Series (LSB 5)
- Microsoft
- Microsoft: Windows API
- Oracle
- Oracle: The Java® Language Specification SE 8 Edition
- Oracle: The Java® Virtual Machine Specification JVM
- Oracle: Java logger API
- Talk Openly Develop Openly (TODO)
- TODO: How to create an open source program
- TODO: Measuring your open source program's success
- TODO: Tools for managing open source programs
- TODO: Using open source code
- TODO: Participating in open source communities
- TODO: Recruiting open source developers
- TODO: Starting an open source project
- TODO: Improve your open source development impact
- TODO: Shutting down an open source project
- TODO: Building leadership in an open source community
- TODO: Setting an Open Source Strategy
- GIT (Revision Control)
- InterPlanetary File System (IPFS)
- Jenkins (Continuous Delivery)
- Jira (Bug tracking system)
- Participating in Open Source Communities
- ZeroMQ Distributed Messaging
- ZeroMQ Message Transport Protocol (ZMTP)

From:

<https://omgwiki.org/dido/> - **DIDO Wiki**

Permanent link:

https://omgwiki.org/dido/doku.php?id=wiki:ebook:app_b_-_defacto&rev=1636491105



Last update: **2021/11/09 15:51**