Attachment XII – Architecture Mapping of Ontology

Section 1 Summary

Platform summary	
Platform ID:	*ONT*
Status/Revision:	V 1.7.0
Type:	*Public – Ontology* *Consortium – Ontology 2B*
Domain:	Blockchain infrastructure

Section 2 Governance & Compliance Functions

Platform governance - Ontology	
Governance Type:	*Permissionless*
Chain Network Admin:	*Ontology community*
Pledge (cost of malicious action):	*Stake*
Description:	*Node with Stake can contribute to consensus network, the stake can be frozen on malicious actions, community vote to decide upon malicious actions.*

Platform trust endorsement policy - Ontology	
Type:	*Tokenomics*
Tool:	ONT/ONG
Policy:	*ONT as stake to become node, and ONG as basic DLT service fee. ONT can be frozen on malicious actions.*

Platform governance – Ontology 2B	
Governance Type:	*Permissioned*
Chain Network Admin:	*Entity*
Pledge (cost of malicious action):	*Liquidated damages*

Description:	*Agreement will be placed to buy in the node into network, and any malicious action will go to law process*

Platform trust endorsement policy – Ontology 2B	
Type:	*Law / Agreement*
Tool:	*ONT ID (with CA) + agreement*
Policy:	-

Economic Model (optional)	
Price Model to Deploy Contracts and do Transactions	*ONG as utility token for gas inside Ontology.* *Ref., fee model[1], deployment[2].*
Who pays the costs of the network	*Users*
Monetary Policy of Tokens	*1 billion ONT total, 1 billion ONG bound with ONT, ONG unbinding curve equation manage ONG unbinding per second[3].*
Rights of Tokens	*ONT as Stake and ONG as utility token inside Ontology multi-chain network*

Section 3 Application

Platform Smart Contract mechanism	
Language	*NEOVM: Python; C#; Javascript;* *WASM: C++; Rust*
Turing Complete?	*Yes*
Compiler	*NEOVM: Python; C#; Javascript;* *WASM: C++; Rust*
Runtime VM:	*NEOVM; WASM*

[1] https://github.com/ontio/ontology-smartcontract/blob/master/smart-contract-tutorial/feemodel_en.md

[2] https://ontio.github.io/documentation/Smart_Contract_Deployment_en.html#calculate-the-gas-consumed-by-deploying-a-smart-contract

[3] https://medium.com/ontologynetwork/triones-node-incentive-model-dbcb175f4728

DevTools	*SDK[4], SmartX[5], Punica Suite[6]*
Extra Tool(s):	*Explorer[7]*
Lifecycle	*Live within an app call*
Description:	*Support NEOVM and WASM with multiple programming languages compiler, as well as some language translators, from bytecode to NEOVM/WASM bytecode.*

Section 4 Protocol

Platform AAA Management	
Account type:	*Identity; address*
Distributed ID:	*ONT ID*
AAA support:	*ONT ID suite (ONTID[8], OntPass[9], TrustAnchor connector[10])*
Description:	*ONT ID as identifier of entities.*

Platform consensus mechanism	
Algorithm:	*VBFT (Byzantine Fault Tolerance with Verifiable Randomness)*
Consensus mode:	*Event*
Management solution:	*Internal*
Description:	*VBFT achieves chain scalability by consensus node selection with VRF, anti-attack ability by randomness and PoS, and fast state finality with BFT.* *Plus, in Ontology 2B, use predefined stake in agreement as PoS in consensus*

4 https://dev-docs.ont.io/#/docs-en/Punica/punica

5 https://dev-docs.ont.io/#/docs-en/SmartX/00-overview

6 https://dev-docs.ont.io/#/docs-en/SDKs/00-overview

7 https://explorer.ont.io/

8 https://pro-docs.ont.io/#/docs-en/ontid/overview

9 https://pro-docs.ont.io/#/docs-en/ontpass/overview

10 https://pro-docs.ont.io/#/docs-en/taconnector/overview

Platform ledger management			
Model:	*balance*	**Extra:**	*MPT on sub-chain and sharding*
Description:	*By default, Ontology uses balance model to store data. Can support UTXO in sub-chain(s). To support SPV, apply MPT in sub-chain and sharding.*		

Section 5 Resources

Node Management	
Node Role	*Candidate node; consensus node*
Joining	*synchronized node with hardware and software installed* - *certain ONT as stake in address, peer admin address/wallet* - *address for node operating, peer runtime address/wallet* - *ONT ID combine with addresses above* *register candidate, have ONT staked, approved by operator role (manually by Ontology Foundation for first network size, delegate to AI robot contract later)*
Leaving	*Quit node and withdraw ONT staked*
Role changing	*Stake to certain rank and upgrade from candidate node to consensus node; for lower rank, downgrade from consensus node to candidate node*
Description:	-

Platform data protection - core	
Mass storage mitigation[11]	*Pay on data storing*
Decentralized Data Storage Support	*No*
Data Privacy Solution	*ZKP POC done, MPC in research*
Tamper Proof (tamper cost):	*stop service, average PoS * 1/3 network scale (nodes)* *tamper,*

[11] On chain storage cost much, solution/mechanism to resolve the problem of large cost of mass storage from node perspective.

	*average PoS * 2/3 network scale for data tamper*
Description:	*Ontology-crypto lib, supports multiple signature schemas[12] and anonymous credential[13]*

Platform Network hypothesis	
Node Scalability:	*Up to 50,000 nodes*
Byzantine Node Accepted? :	*Yes;*
Network Structure	*Flexible*
P2P? :	*Yes*
Network Discovery Protocol	*DHT*
Data Exchange Protocol	-
Description:	*Theoretically there's no limitation of node count. However, to satisfy Byzantine failure tolerance. Node scalability shall satisfy hypergeometric distribution.* *Consensus node count per block < 200, error rate < 0.00000001.*

Section 6 Utils

Platform Messaging Mechanism	
Protocol Type	*RPC; RESTful*
Description:	-

Platform Crypto Libraries	
Secure Network Connection Type	*TLS*
Cipher Suites	*Key types: ECDSA; SM2; EdDSA* *Signature schemes: SHA224withECDSA; SHA256withECDSA; SHA384withECDSA; SHA512withECDSA; SHA3-224withECDSA; SHA3-*

[12] https://github.com/ontio/ontology-crypto

[13] https://github.com/ontio/ontology-crypto/wiki/Anonymous-Credential

	256withECDSA; SHA3-384withECDSA; SHA3-512withECDSA; RIPEMD160withECDSA; SM3withSM2; SHA512withEdDSA
Description:	*Cryptography Library for Ontology Network is referenced*[14]

Section 7 Operation & Maintenance

Platform system management – node	
Log	*yes*
Monitoring	*explore*[15]
Description:	*[Operation and Maintenance] -*

Platform system management – chain network	
Permission Control:	*Yes*
Auditing:	*N/A*
Supervisory Support:	*N/A*
Description:	*[Operation and Maintenance] Native auth and global parameter contract.*

Section 8 External Resource Management

Platform external data exchange – application service	
Interoperation solution:	*ONT ID + data token solution to map data with token.* *Semantic web solution (ontology data model) to support data interoperability*
Description:	-

Section 9 Extensions

Platform Extensions
[the following list can be duplicated for multiple extensions]

[14] https://github.com/ontio/ontology-crypto

[15] https://explorer.ont.io/

Name	*Ontology sharding*
Extension type[16]	*Internal*
Solution	*sharding*
Extension mode[17]	*horizontal*
Serve domain	*Computing capability*
Description:	*Ontology sharding supports shard on state, shard on transaction and shard on network* [18]
Name	*Ontology sidechain / ecochain*
Extension type	*External*
Solution	*side-chain*
Extension mode	*horizontal*
Serve domain	*Cross domain (chain) applications*
Description:	*Ontology ecochain serves the requirement of multiple domain requirement with different governance model*
Name	*Ontology oracle and state channel*
Extension type	*External*
Solution	*Layer 2 + oracle*
Extension mode	*Horizontal and vertical*
Serve domain	*Non-DLT applications and hybrid storage system*

16 Standing from DLT system instance perspective, any extension inside the instance is marked as "internal", while any extension outside the instance is marked as "external"

17 All extension instances are equal (with similar capability and functional features), targeting for the scalability of DLT instance, marked as "horizontal"; extensions with different functional features, targeting to enforce the capability of DLT instance, marked as vertical. Extension type and mode pair(s) is/are used to describe the extension as to the whole DLT system. E.g., sharding (internal – horizontal), lightening – BTC (external – vertical), Corda Contract (internal – vertical).

18 https://github.com/ontio/documentation/tree/master/sharding

Description:	*Ontology oracle and state channel provides the requirement of on-chain / off-chain applications and extend the performance of on-chain applications with lower cost*