Appendix – Behavior Interface Model

Table 1 shows the Behavioral Interfaces of the BlockVoke/ACME Extension.

Table 1: Behavioral Interface Model of the BlockVoke/ACME Extension

Activity	**Trigger**	**Precondition(s)**	**Postcondition(s)**
Register ACME Account	CO wants to register an ACME account	Generated ACME Account key–pair	ACME account registered
Create ACME registration request		CO does not already have an ACME account	ACME registration request created (unsigned)
Sign ACME registration request		Generated ACME account key–pair, ACME registration request created	ACME registration request signed
ACME CA Verifies registration request signature		ACME registration request signed by CO, ACME registration request signature is valid	ACME account registered
Generate Certificate	ACME CA wants to generate CO's Certificate	CO's CSR with information relevant to the Certificate, CO's (wallet) public key, personal private key (for signing), personal (wallet) public key, CO's ACME account is registered	Generated Certificate ready to be verified by end–user
Communicate newly signed certificate		Certificate Generated	CO has communicated newly signed certificate with end–user and their organization
Sign certificate		CA's public key, Multisignature address generated	Certificate signed by CA
Compute certificate fingerprint		Certificate signed by CA	Certificate fingerprint computed and added by CA

Generate Multisignature address		CO's (wallet) public key, CA's (wallet) public key	1-of-2 Multisignature address generated
Verify CSR Signature		CO's CSR	CSR verified
Send CSR to CA		CSR identifiers validated	CO has sent CSR to ACME CA
ACME Validation	CO wants to create CSR	ACME account registered, CO's Bitcoin wallet key–pair, CSR identifiers to be validated	CSR Identifiers Validated
Send ACME order			Signed ACME order sent to CA
Verify ACME order signature		ACME order created, ACME account registered	ACME order verified
Send ACME validation challenge request for CO Bitcoin address public key		Random number generated, ACME order verified	CO receives ACME validation challenge request
CO create validation challenge response		CO has received ACME validation challenge request, CO Bitcoin key–pair	Validation challenge response created
CO signs validation challenge response		CO has created ACME validation challenge response, CO's ACME account key–pair	Validation challenge response created
CO sends validation response		CO has created and signed ACME validation challenge response	ACME CA receives signed ACME validation response
Verify validation response signature		CA has received signed ACME validation response	ACME validation challenge response signature verified
Verify ACME validation response		CA has verified ACME validation response signature, CO's Bitcoin address public key	CSR identifiers validated

Verify Certificate	End–User wants to verify a CO's Certificate	Signed Certificate, CA's public key	Certificate has been verified by end user
Check Certificate Fingerprint		Signed Certificate with computed Fingerprint	Certificate Fingerprint Verified by End–User
Check Certificate Signature		Signed Certificate, Associated CA's Public Key	CA's Signature on Certificate Verified by End–User
Revoke Certificate	CA or CO wants to revoke a certificate that they have signed/own respectively	Crypto wallet with Small credit amount, Signed Certificate, RFC 5280 Code, Optional CA Identifier	Certificate has been revoked
Create Revocation transactions			Revocation transactions for a Certificate have been created
Create Tx:Revoke transaction		Hash of Tx:Fund transaction for same Multisig address, Signed Certificate, RFC 5280 Code, Optional CA Identifier, Tx:Fund transaction has been created	Tx:Revoke transaction spending the funds sent to Multisig address in Tx:Fund
Add previous output hash of input address (Multisig address)		Hash of Tx:Fund transaction for same Multisig address	Previous output hash of Multisig address added to Tx:Revoke transaction
Add Input address (Multisig address)	CA or CO wants to revoke a certificate that they have signed/own respectively	Multisignature address associated with a Signed Certificate	Input address added to Tx:Revoke transaction
Add output address		An output address for spending the funds in the multisig address	Output address added to Tx:Revoke transaction

Add OP_RETURN script		Certificate Signature, Date of Issuance, RFC 5280 Revocation code	OP_RETURN script added to Tx:Revoke transaction
Add Certificate Signature		Certificate Signature	Certificate Signature added to OP_RETURN script of Tx:Revoke transaction
Add Certificate Date of Issuance		Certificate Date of Issuance in days since 2020-02-02	Certificate Date of Issuance added to OP_RETURN script of Tx:Revoke transaction
Create Tx:Fund transaction		Small credit amount for funding Multisig wallet, Input wallet address containing the funding amount, Multisignature address associated with the Signed Certificate	Tx:Fund transaction has been created
Add previous output hash of Input address		Hash of previous transaction with Input address of Tx:Fund transaction as output	Previous output hash of Input address added to Tx:Fund transaction
Prepare Funds		Small Credit amount	Small credit amount prepared and added to Input address of Tx:Fund transaction
Add Input Address		Input Wallet address	Input address added to Tx:Fund transaction
Add output address (Multisig address)		Multisignature address associated with a Signed Certificate	Output address added to Tx:Fund transaction

Send Revocation transactions		Created Tx:Fund and Tx:Revoke transactions	Revocation transactions have been sent to the blockchain network
Add unconfirmed Revocation transactions into mempool	Blockchain network receives Revocation transactions	Revocation transactions scrutinised by the Blockchain Network	Revocation Transactions added to Unconfirmed transaction List (mempool)
Scrutinise Revocation transactions			Revocation Transactions scrutinised by Blockchain Network
Propagate mined blocks with confirmed Revocation transactions	Blockchain network receives a newly mined blocks with confirmed Revocation transactions	Newly mined blocks	Mined blocks with confirmed Revocation transactions propagated on Blockchain Network
Mine Revocation transactions	Miner receives Revocation transactions from Blockchain Network	Unconfirmed Revocation Transactions in mempool	Revocation transactions mined into a new block and sent to Blockchain Network
Create new block with Revocation transactions		Revocation transactions in Block’s transaction list, nonce	Block containing Revocation transactions created
Find nonce		Transaction List containing revocation transactions, Previous block hash	Nonce for block found
Select Revocation transactions from mempool		Revocation transactions in Blockchain Network’s mempool	Revocation transactions added to Block’s transaction list from mempool
Mark Certificate as ’Revoked’	User witnesses Tx:Revoke transaction for a Certificate	Confirmed Tx:Revoke transaction	User marks Certificate as Revoked

Communicate Revocation Transactions to Users	Tx:Revoke transaction for a certificate has been confirmed and appears in a block	Certificate fingerprint	User has witnessed a Tx:Revoke transaction on the blockchain with the certificate fingerprint in the OP_RETURN script