Polkadot Runtime Environment Spec Glossary (WIP)

January 16, 2020

Basics

Symbol	Description	Defined
b	a sequence of bytes of length n	$b := (b_0, b_1, ..., b_{n-1})$ such that $0 \leq b_i \leq 255$
$\mathbb{B}_n$	the set of all byte arrays of length n	$\mathbb{B} := \bigcup_{i=0}^{\infty} \mathbb{B}_i$
I	little-endian representation of a non-negative integer	$I = (B_n...B_0)_{256}$
B	byte array	$B = (b_0, b_1, ..., b_n)$ such that $b_1 := B_i$
Enc_{LE}		$Enc_{LE} : \begin{array}{lll} \mathbb{Z}+ & \rightarrow & \mathbb{B} \\ (B_n...B_0)_{256} & \rightarrow & (B_0, B_1, ..., B_n) \end{array}$
C	a blockchain is a directed path graph. Each node of the graph is called Block and indicated by B	
$P(B)$	the parent of block B	$B_{n+1} := P(B_n)$
$\mathcal{N}$	the set of the nodes of the Polkadot state trie	
N	an individual node in the trie	$N \in \mathcal{N}$
$\mathcal{N}_b$	a branch node which has one child or more (max 16)	$\mathcal{N}_b := \{N \in \mathcal{N} \mid \text{N is a branch node}\}$
$\mathcal{N}_l$	a leaf node is a childless node	$\mathcal{N}_l := \{N \in \mathcal{N} \mid \text{N is a leaf node}\}$
SV_N	the subvalue of the given node	$SV_N := \begin{cases} Enc_{SC}(StoredValue(k_N)) & \text{N is a leaf node} \\ ChildrenBitmap(N) \| Enc_{SC}(H(NC_1))...Enc_{SC}(StoredValue(k_N)) & \text{N is a branch node} \end{cases}$
$StoredValue$	function to retrieve the value stored under a specific key in the state storage	$StoredValue \quad \mathcal{K} \rightarrow \mathcal{V}$ $k \rightarrow \begin{cases} v & if(k, v) \text{ exists in state storage} \\ \phi & otherwise \end{cases}$ where $K \subset \mathbb{B}$ and $\mathcal{V}w \subset \mathbb{B}$ are respectively the set of all keys and values stored in the state storage

Symbol	Description	Defined
$KeyEncode(k)$	function to encode keys for labeling brnaches of the Trie	$k_{enc} := (k_{enc1}, ..., k_{enc2n}) := KeyEncode(k)$
k_{enc}		such that: $KeyEncode(k) : \begin{cases} \mathbb{B} & \rightarrow Nibbles^4 \\ k := (b_1, ..., b_n) := & \rightarrow (b_1^1, b_1^2, b_2^1, b_2^2, ..., b_n^1, b_n^2) \\ & \rightarrow := (k_{enc_1}, ..., k_{enc2n}) \end{cases}$ where $Nibble^4$ is the set of all nibbles of 4-bit arrays and b_i^1 and b_i^2 are 4-bit nibbles, which are the big endian representation of b_i: $(b_i^1, b_i^2) := (b_i/16, b_i, mod 16)$ where mod is the remainder and / is the integer division operators.
pk_N	TODO	
pk_N^{Agr}	TODO	
$HeadN$	the node header of node N	
v_N	the node value which is stored by the node $N \in \mathcal{N}$	$v_N := Head_N \parallel Enc_{HE}(pk_N) \parallel SV_N$
$ChildrenBitmap$		$ChildrenBitmap : \begin{matrix} \mathcal{N}_b \rightarrow \mathbb{B}_2 \\ N \rightarrow (b_{15}, ..., b_8, b_7, ...b_0)_2 \end{matrix}$ where $b_i := \begin{cases} 1 & \exists N_c \in \mathcal{N} : k_{Nc} = k_{N_b} \parallel i \parallel pk_{Nc} \\ 0 & otherwise \end{cases}$
$H(N)$	the Merkle value of N	$H : \mathbb{B} \rightarrow \mathbb{B}_{32}$ $H(N) : \begin{cases} v_N & \parallel v_N \parallel < 32 \\ Blake2b(v_N) & \parallel v_N \parallel \geq 32 \end{cases}$ Where v_N is the node value of N and $0_{32-\parallel v_N \parallel}$ an all zero byte array of length $32- \parallel v_N \parallel$. The Merkle hash of the Trie is defined as: $Blake2b(H(R))$ where R is the root of the Trie.

Block Format

Symbol	Description	Defined
H_p	the 32-byte Blake2b hash of the header of the parent of the block	
H_i	the interger representing the index of the current block in the chain. It is equal to the number of the ancestor blocks. The genesis block has number 0	
H_r	the root of the Merkle trie, whose leaves implement the storage for the system	

Symbol	Description	Defined
H_e	the field which is reserved for the Runtime to validate the integrity of the extrinsics composing the block body. The extrinsics_root is set by the runtime and its value is opaque to Polkadot RE	
H_d	used to store any chain-specific auxiliary data	$H_d(B) := H_d^1, ..., H_d^n$ where H_d^i's are digest items
$H_h(B)$	the hash of the header of block B by codec	$H_h(B) := Blake2b(Enc_{SC}(Head(B)))$
$H_h(B)$	Block hash	
$H_i(B)$	Block number	
$Body(B)$	the body of block B	$Body(B) := Enc_{SC}(E_1, ..., E_n)$ where each $E_i \in \mathbb{B}$ is a SCALE encoded extrinsic

SCALE Codec

Symbol	Description	Defined
A	Byte array	$A := b_1, b_2, ...b_n$
T	Tuple where A_i's are values of different types	$T := (A_1, ..., A_n)$
S	Sequence where Ai's are values of the same type (and the decoder is unable to infer value of n from the context)	$S := A_1, ..., A_n$
τ	Varying data type (TODO)	$T = \{T_1, ..., T_n\}$
$Enc_{SC}(A)$	SCALE encoding of byte array A such that $n < 2^{256}$	$Enc_{SC}(A) := Enc_{SC}^{Len}(\| A \|) \| A$
$Enc_{SC}(T)$	SCALE encoding of tuple T	$Enc_{SC}(T) := Enc_{SC}(A_1) \| Enc_{SC}(A_2) \| ... \| Enc_{SC}(A_n)$
$Enc_{SC}(S)$	SCALE encoding of sequence S	$Enc_{SC}(S) := Enc_{SC}^{Len}(\| S \|)Enc_{SC}(A_1) \| Enc_{SC}(A_2) \| ... \| Enc_{SC}(A_n)$
Enc_{SC}^{Len}	SCALE length encoding aka. compact encoding of non-negative interger numbers of varying sized prominently in an encoding length of arrays	$Enc_{SC}^{Len} : \mathbb{N} \to \mathbb{B}$ $n \to b \begin{cases} l_1 & 0 \le n < 2^6 \\ i_1 i_2 & 2^6 \le n < 2^{14} \\ j_1 j_2 j_3 & 2^{14} \le n < 2^{30} \\ k_1 k_2 ... k_m & 2^{30} \le n \end{cases}$ in where the least significant bits of the first byte of byte array b are defined as follows: $l_1^1 l_1^0 = 00$ $i_1^1 i_1^0 = 01$ $j_1^1 j_1^0 = 10$ $k_1^1 k_1^0 = 11$ and the rest of the bits of b store the value of n in little-endian format in base-2 as follows: $\left.\begin{array}{ll} l_1^7 ... l_1^3 l_1^2 & n < 2^6 \\ i_2^7 ... i_2^0 i_1^7 ... i_1^2 & 2^6 \le n < 2^{14} \\ j_4^7 ... j_4^0 j_3^7 ... j_1^7 ... j_1^2 & 2^{14} \le n < 2^{30} \\ k_2 + k_3 2^8 + k_4 2^{2\times 8} + ... + k_m 2^{(m-2)\times 8} & 2^{30} \le n \end{array}\right\} := n$ such that: $k_1^7 .. k_1^3 k_1^2 := m - 4$

GRANDPA

Symbol	Description	Defined
v	GRANDPA Voter	
k_v^{pr}	ED25519 private key of v	
v_{id}	ED25519 public key of v	
$\mathbb{V}$	set of all GRANDPA voters	
$\mathbb{V}_B$	set of all GRANDPA voters for a given block	
$\mathbb{V}_{id}$	is an incremental counter tracking membership, which changes in V	
GS	GRANDPA state	$GS := \{\mathbb{V}, id_{\mathbb{V}}, r\}$
$V(B)$	GRANDPA vote	$V(B) := (H_h(B), H - I(B))$
$V_v^{r,pv}$	pre-vote	
$V_v^{r,pc}$	pre-commit	
r	Voting round number	
V_id	Incremental counter tracking membership	
$V_v^{r,stage}(B)$	equivocatory vote	
$\mathcal{E}^{r,stage}$	set of all equivocators voters in sub-round "stage" of round r	
$\mathcal{E}_{obs(v)}^{r,stage}$	set of all equivocators voters in sub-round "stage" of round r observed by voter v	
$VD_{obs(v)(B)}^{r,stage}$	the set of observed direct votes for block B in round r	
$V_{obs(v)}^{r,stage}$	the set of total votes observed by voter v in sub-round "stage" of round r	
$V_{obs(v)}^{r,stage}(B)$	set of all observed votes by v in the sub-round stage of round r for block B	$V_{obs(v)}^{r,stage}(B) := \bigcup_{v_i \in \mathbb{V}, B \geq B'} VD_{obs(v)}^{r,stage}(B')$
$B_v^{r,pv}$	The current pre-voted block	$H_n(B_v^{r,pv}) = Max(H_n(B) \| \forall B : \#V_{obs(v)}^{r,pv}(B) \geq 2 \setminus 3 \| \mathbb{V} \|)$

Voting Messages Specification

Symbol	Description	Defined
$M_v^{r,stage}$	A broadcasted message by the voter v casting his vote to the network	$M_v^{r,stage} := Enc_{SC}(r, id_{\mathbb{V}}, Enc_{SC}(stage, V_v^{r,stage}, Sig_{ED25519}(Enc_{SC}(stage, V_v^{r,stage}, r, V_{id}), v_{id})))$
$J^r(B)$	The justification for block B in round r	The justification is a vector of pairs of the type $(V(B'), (Sign_{vi}^{r,pc}(B'), v_{id}))$ in which either $B' \geq B$ or $V_{vi}^{r,pc}(B')$ is an equivocatory vote
$Sign_{vi}^{r,pc}(B')$	The signature of voter v, broadcasted during the pre-commit sub-round of round r	
$M_v^{r,Fin}(B)$	The finalizing message broadcasted by voter v to the network indicating that voter v has finalized bock B in round r	$M_v^{r,Fin}(B) := Enc_{SC}(r, V(B), J^r(B))$

Cryptographic keys

Symbol	Description	Defined
Account key (sk^a, pk^a)	A keypair of type of either SR25519, ED25519, secp256k1	

Hex encoding

Symbol	Description	Defined
$Enc_{HE}(PK)$	hex encoding	$Enc_{HE}(PK) :=$ $\begin{cases} Nibbles_4 \to \mathbb{B} \\ PK = (k_1, ..., k_n) \to \begin{cases} (16k_1 + k_2, ..., 16k_{2i-1} + k_{2i}) & n = 2i \\ (k_1, 16k_2 + k_3, ..., 16k_{2i} + k_{2i+1}) & n = 2i + 1 \end{cases} \end{cases}$