



UNIVERSITÀ DI PARMA

DIPARTIMENTO DI SCIENZE MATEMATICHE, FISICHE E INFORMATICHE

Corso di Laurea Triennale in Informatica

Candidato

Saverio Mattia Merenda, 330503

Relatore

Prof. Vincenzo Arceri

Costruzione di Control-Flow Graph completi per bytecode EVM

Blockchain

Blockchain

Verifica della correttezza del codice

Blockchain

Verifica della correttezza del codice

Control-Flow Graph

Blockchain

Verifica della correttezza del codice

Control-Flow Graph

Interpretazione astratta

Blockchain

Verifica della correttezza del codice

Control-Flow Graph

Interpretazione astratta





Satoshi Nakamoto, 2008



Satoshi Nakamoto, 2008
Registro digitale



Satoshi Nakamoto, 2008

Registro digitale

Distribuito



Satoshi Nakamoto, 2008

Registro digitale

Distribuito

Catena di blocchi



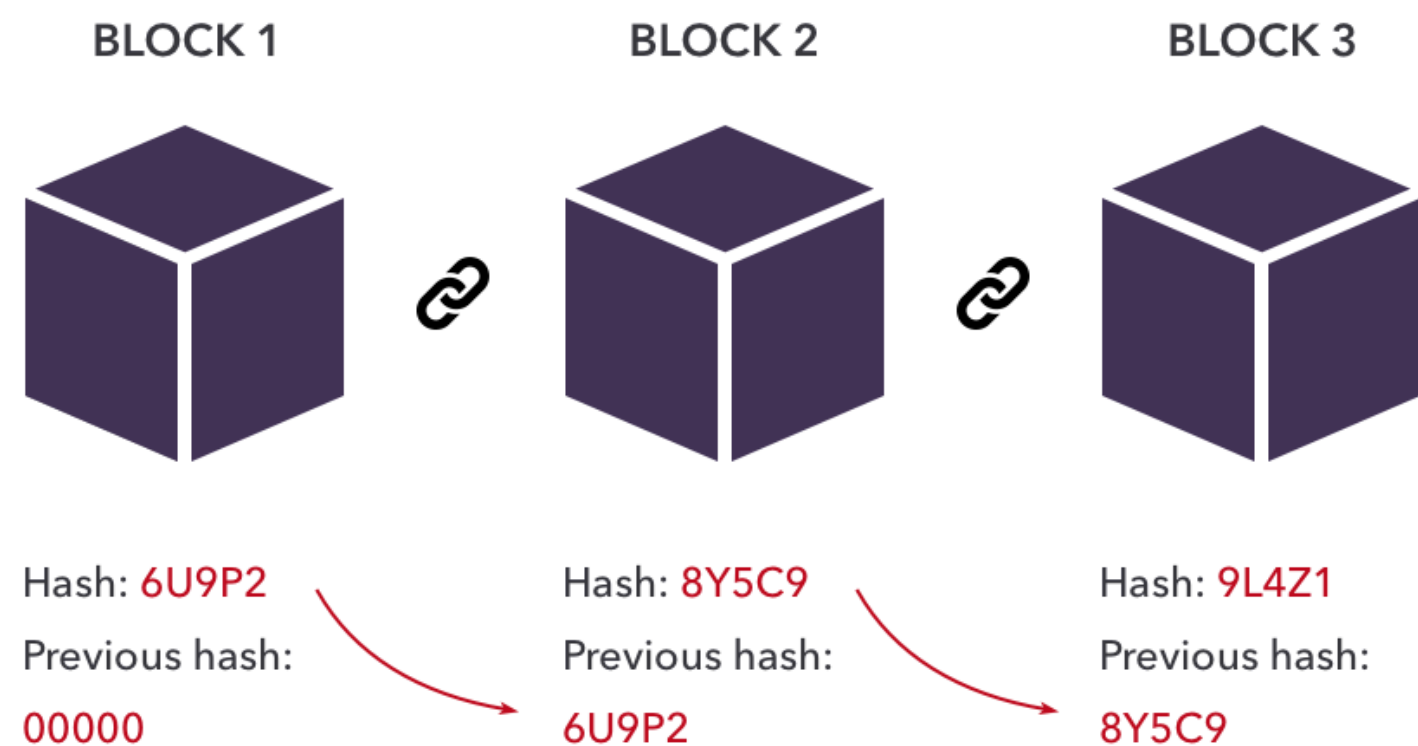
Satoshi Nakamoto, 2008

Registro digitale

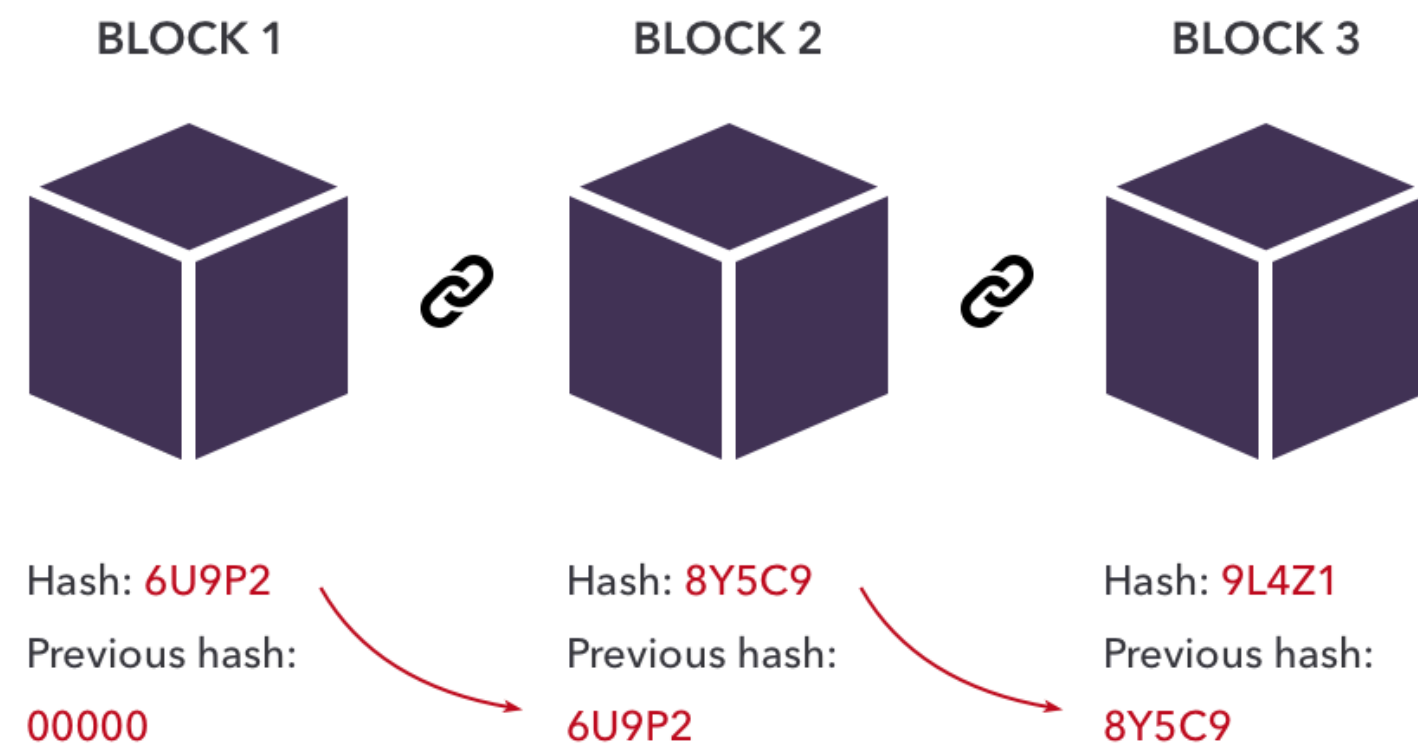
Distribuito

Catena di blocchi

Sicuro e affidabile

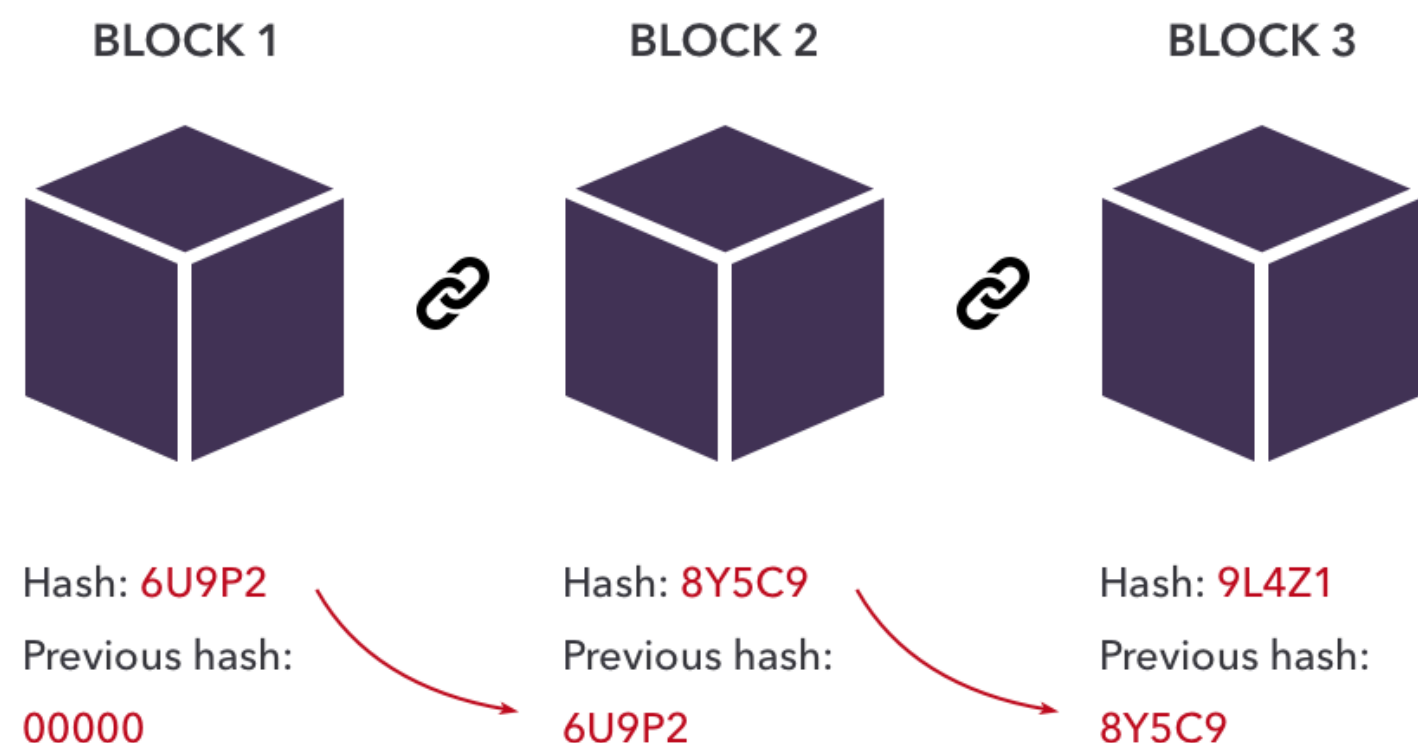


Blocchi interconnessi

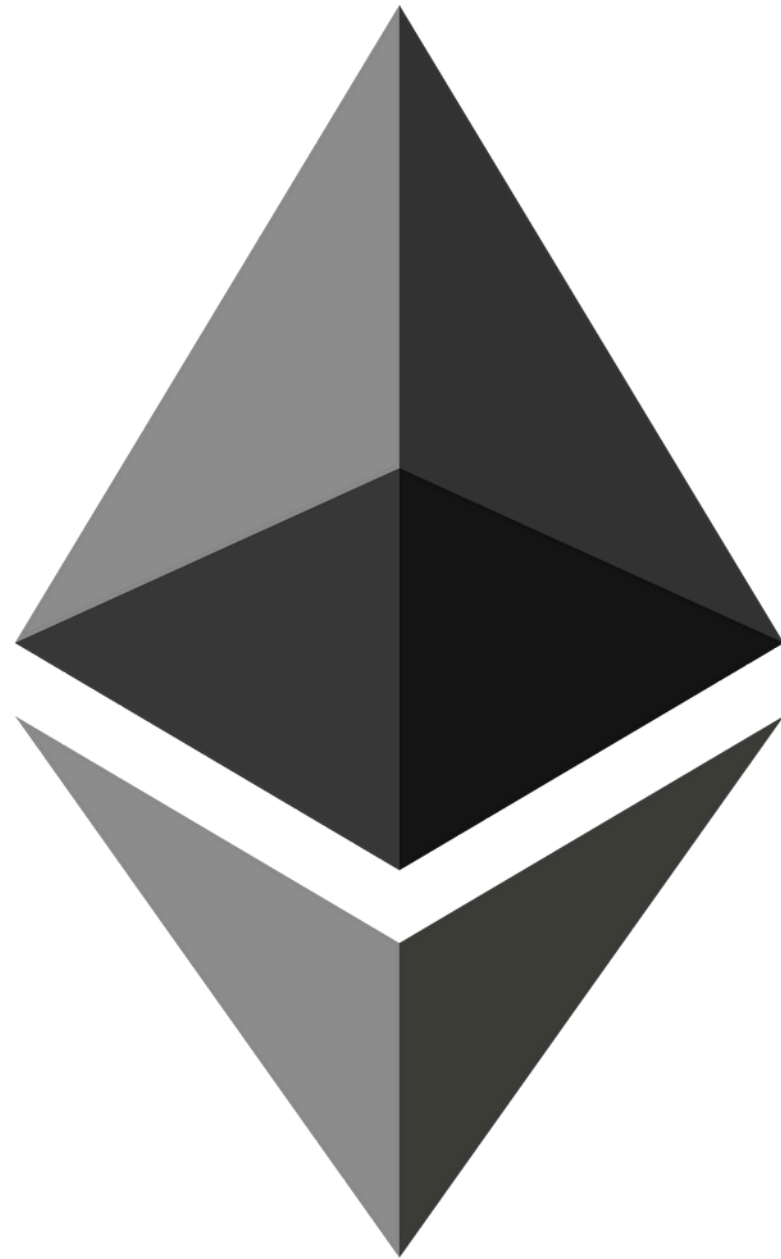


Blocchi interconnessi

Crittografia



Blocchi interconnessi
Crittografia
Catena inalterabile



Vitalik Buterin, 2013



Vitalik Buterin, 2013

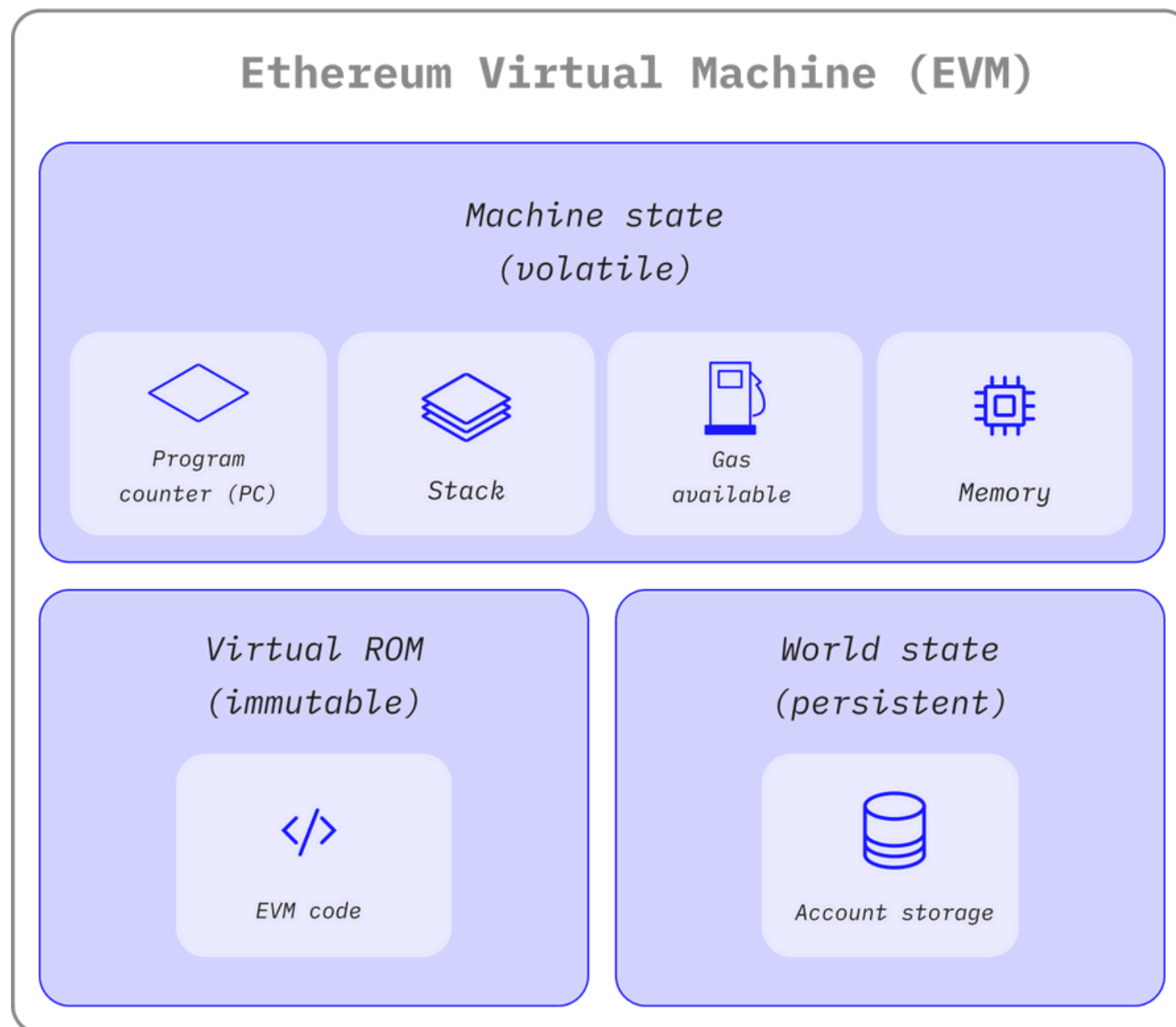
Pubblica & open-source



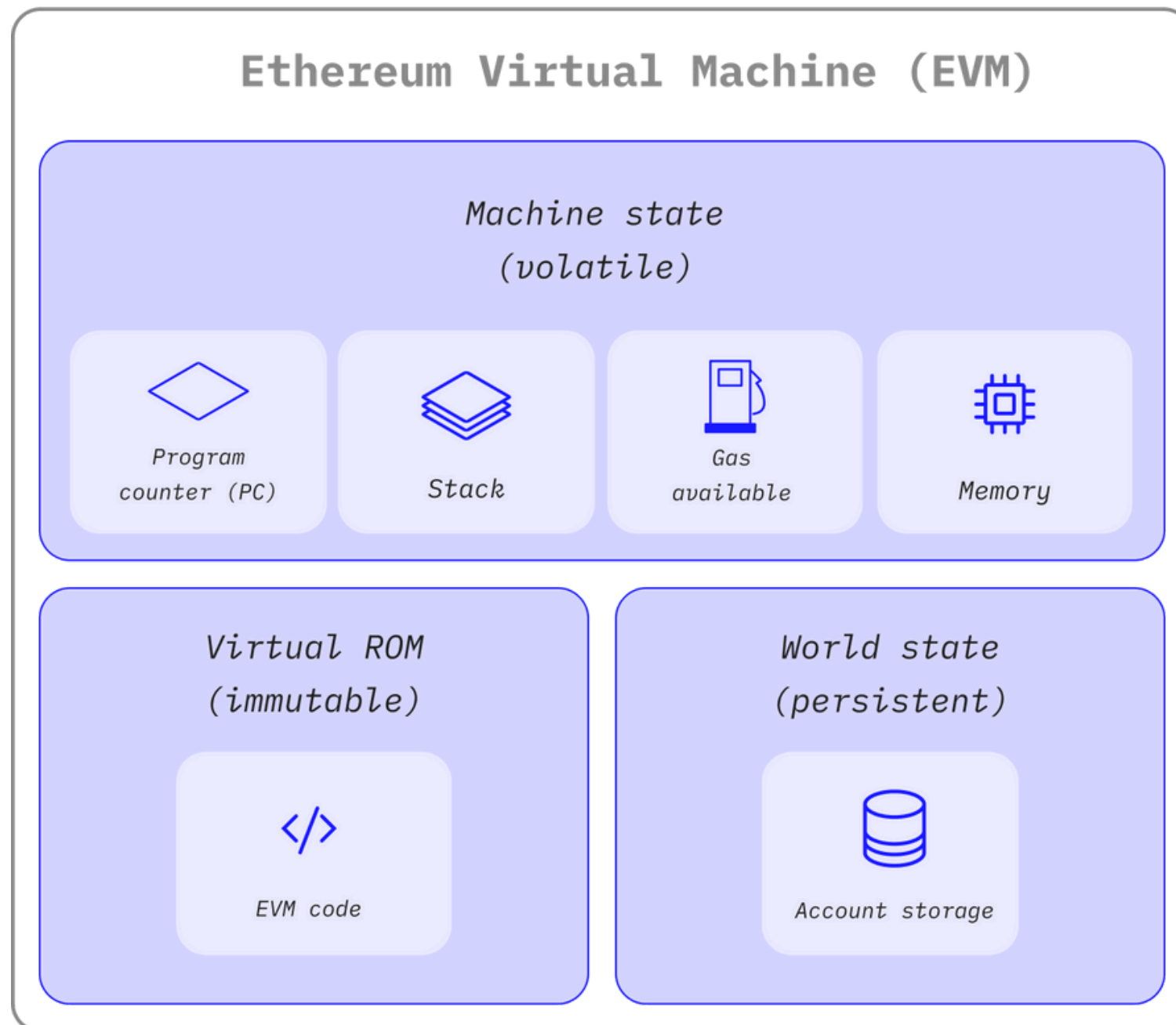
Vitalik Buterin, 2013

Pubblica & open-source

Smart contract

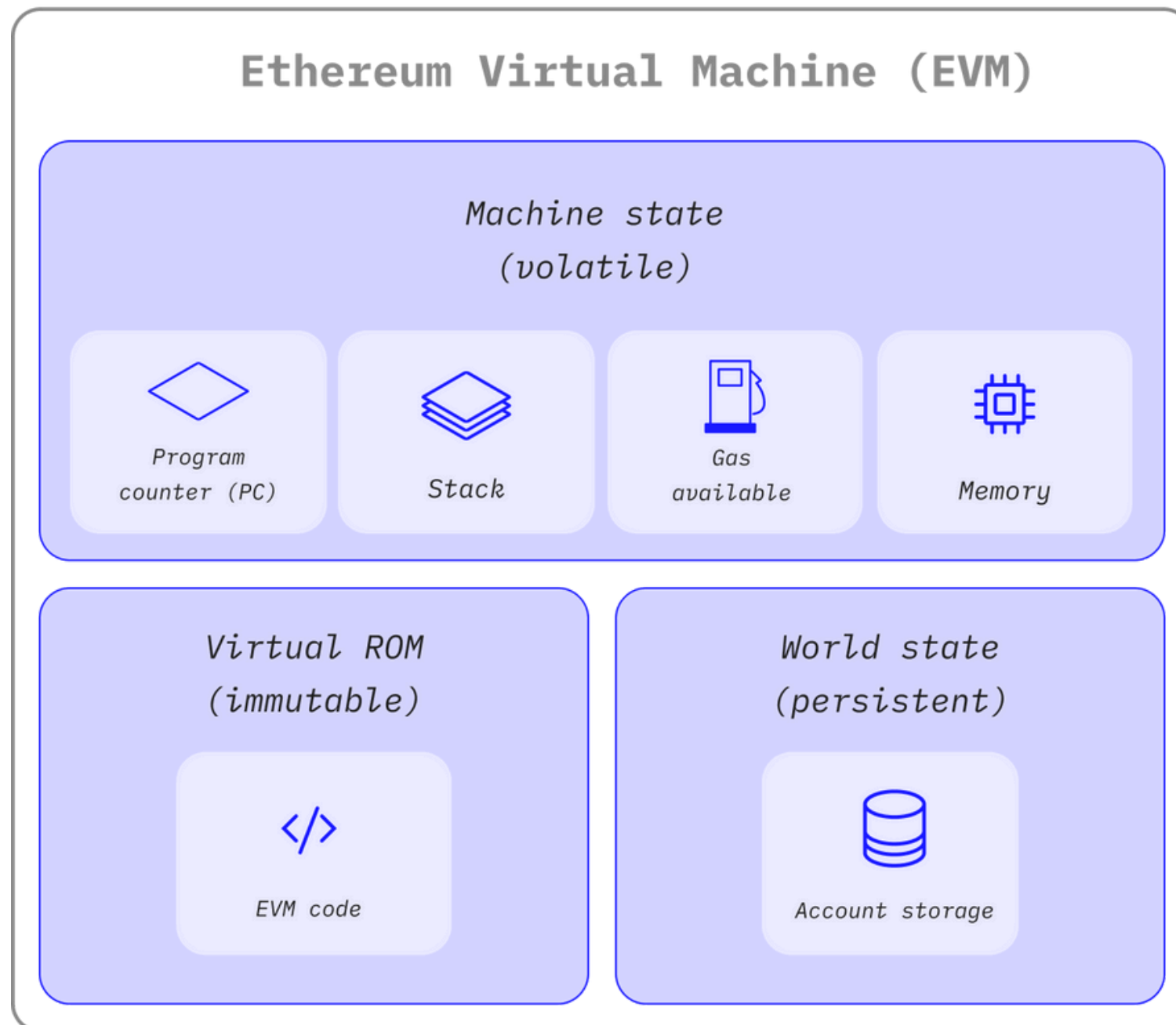


Cuore di Ethereum



Cuore di Ethereum

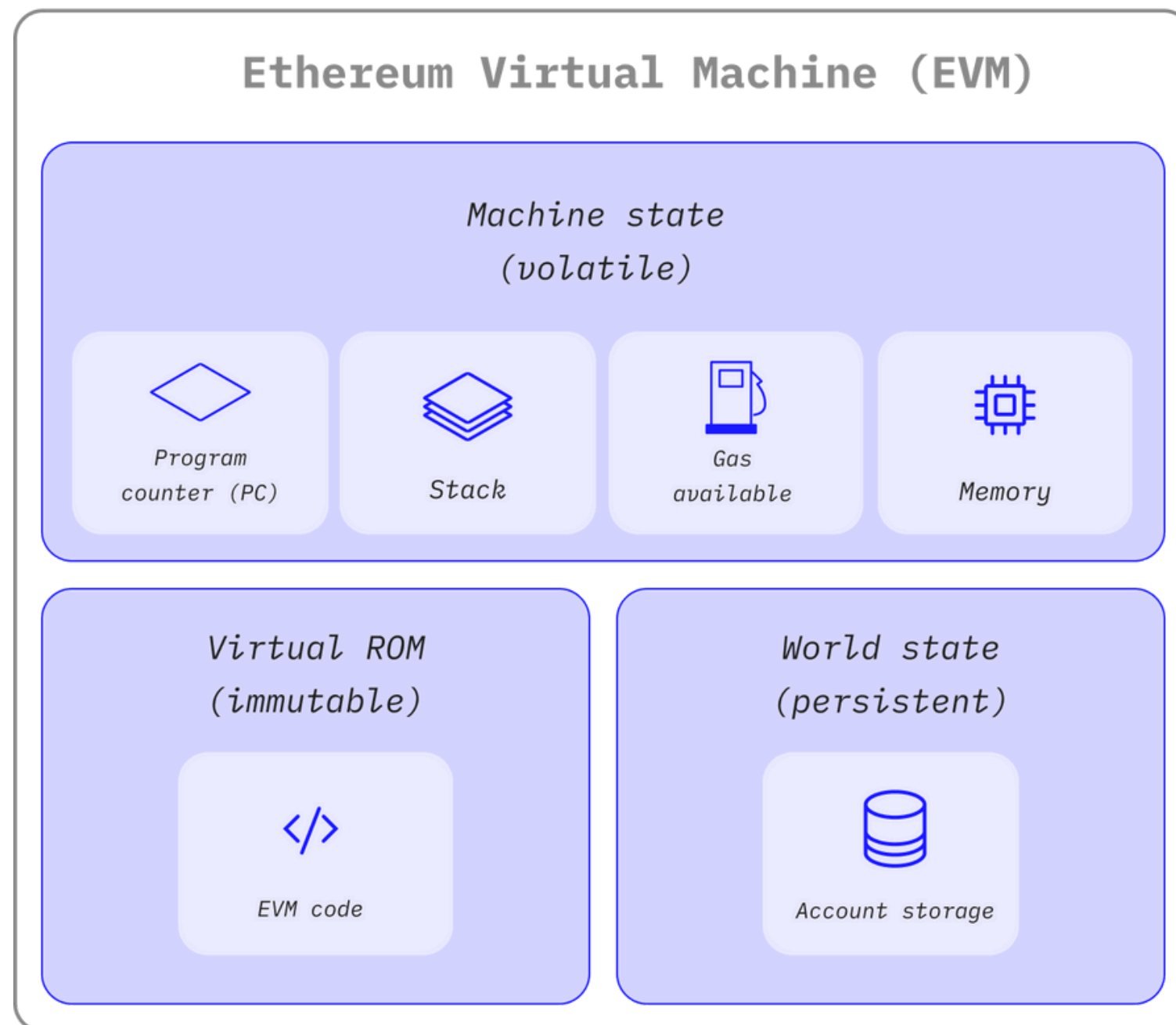
Gestione smart contract



Cuore di Ethereum

Gestione smart contract

Funzione matematica

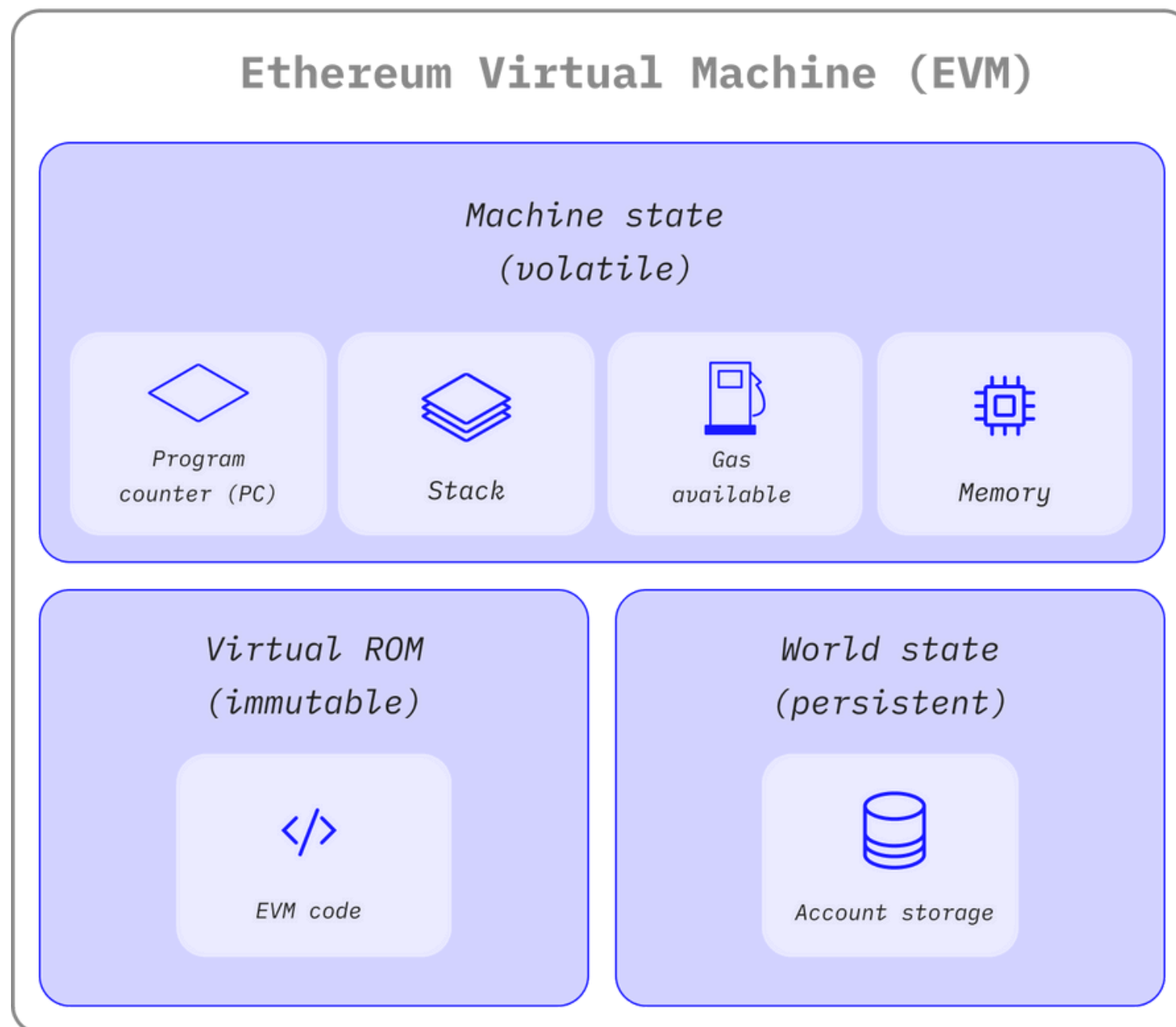


Cuore di Ethereum

Gestione smart contract

Funzione matematica

Stack, Memory & Storage



Cuore di Ethereum

Gestione smart contract

Funzione matematica

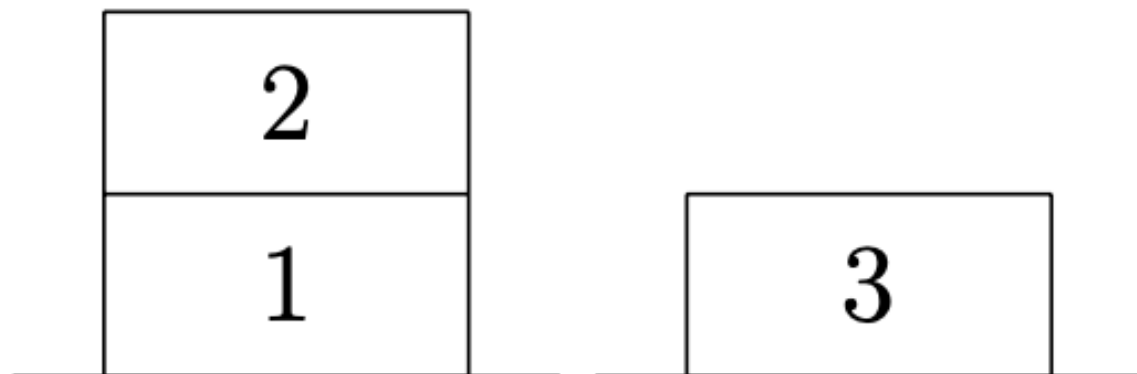
Stack, Memory & Storage

Terminazione garantita

```
PUSH1 0x01  
PUSH1 0x02  
ADD
```

Basso livello

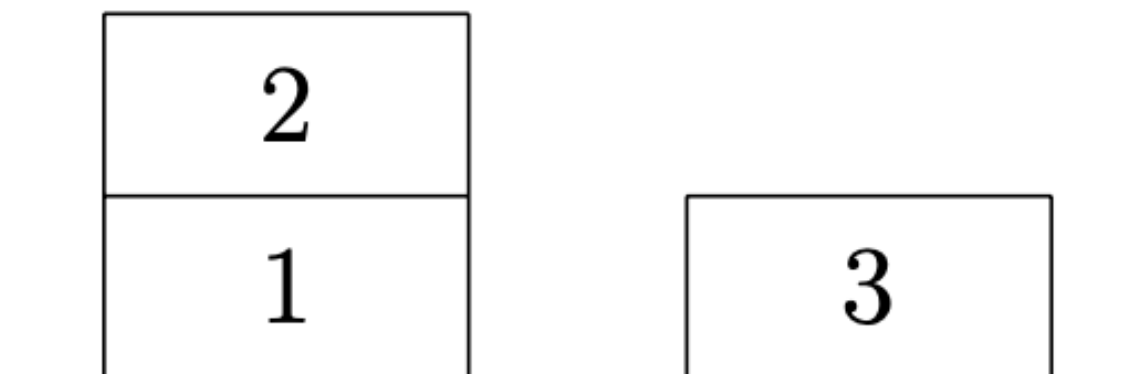
```
PUSH1 0x01  
PUSH1 0x02  
ADD
```



Basso livello

Basato su stack


```
PUSH1 0x01  
PUSH1 0x02  
ADD
```

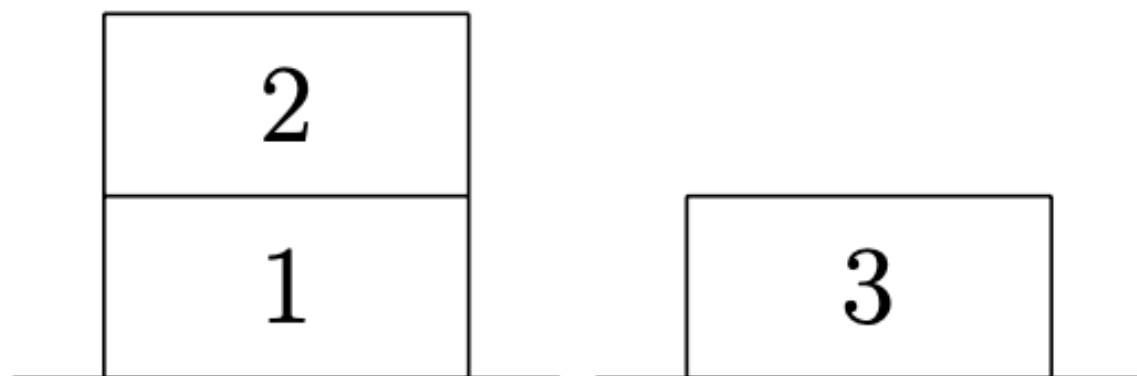


Basso livello

Basato su stack

Oltre 150 opcode

```
PUSH1 0x01  
PUSH1 0x02  
ADD
```



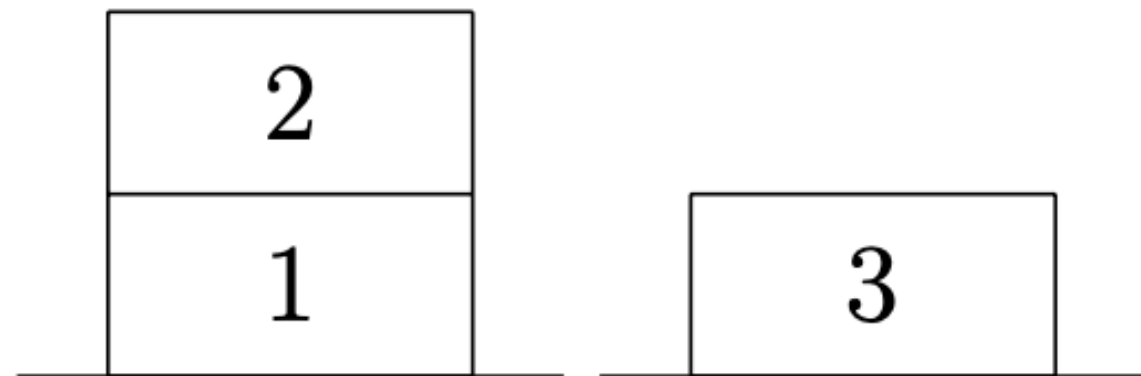
Basso livello

Basato su stack

Oltre 150 opcode

Molte operazioni

```
PUSH1 0x01  
PUSH1 0x02  
ADD
```



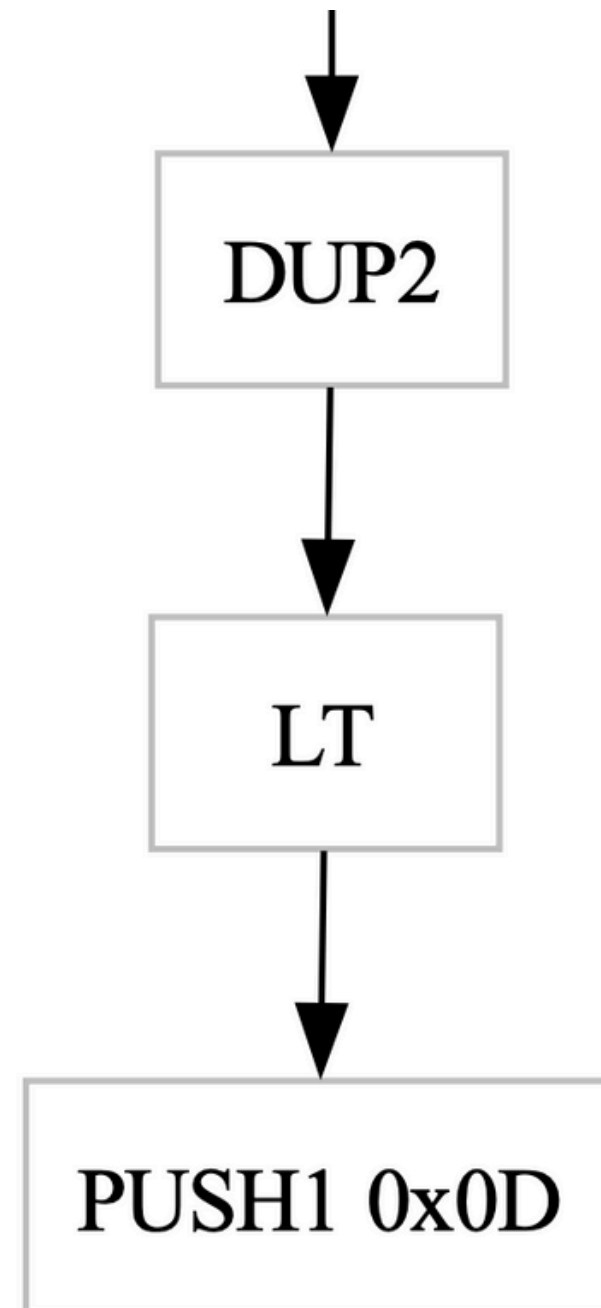
Basso livello

Basato su stack

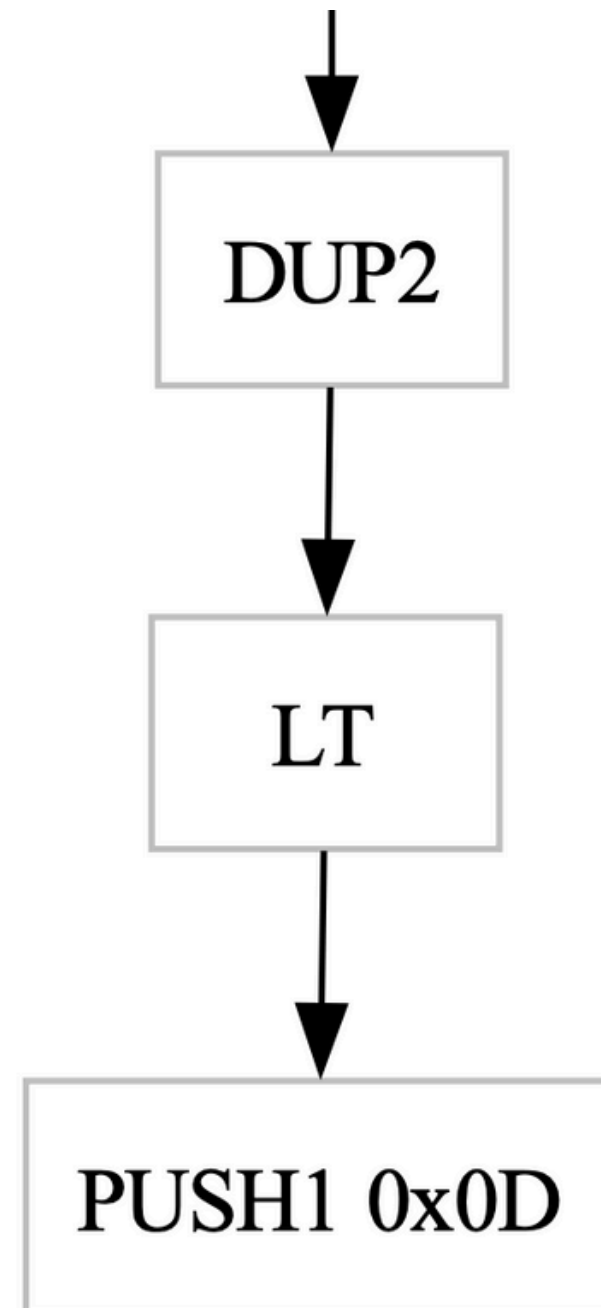
Oltre 150 opcode

Molte operazioni

Accesso ad informazioni

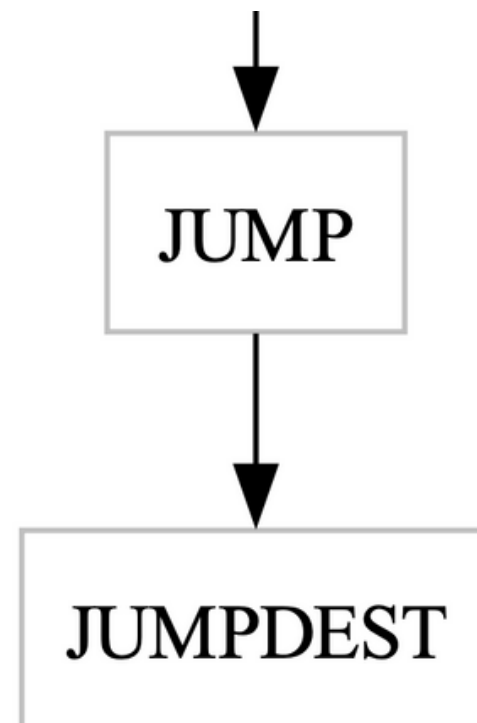


Flusso sequenziale



Flusso sequenziale
JUMP & JUMPI

```
PUSH1 0x01  
PUSH1 0x02  
JUMP
```

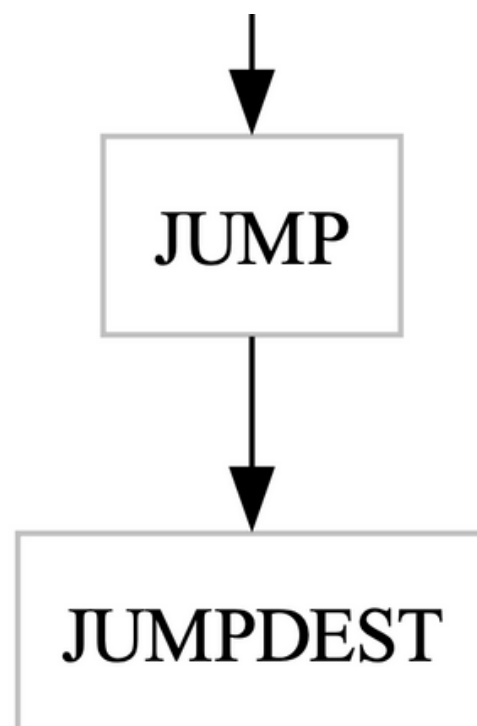


Flusso sequenziale

JUMP & JUMPI

Salto incondizionato

```
PUSH1 0x01  
PUSH1 0x02  
JUMP
```

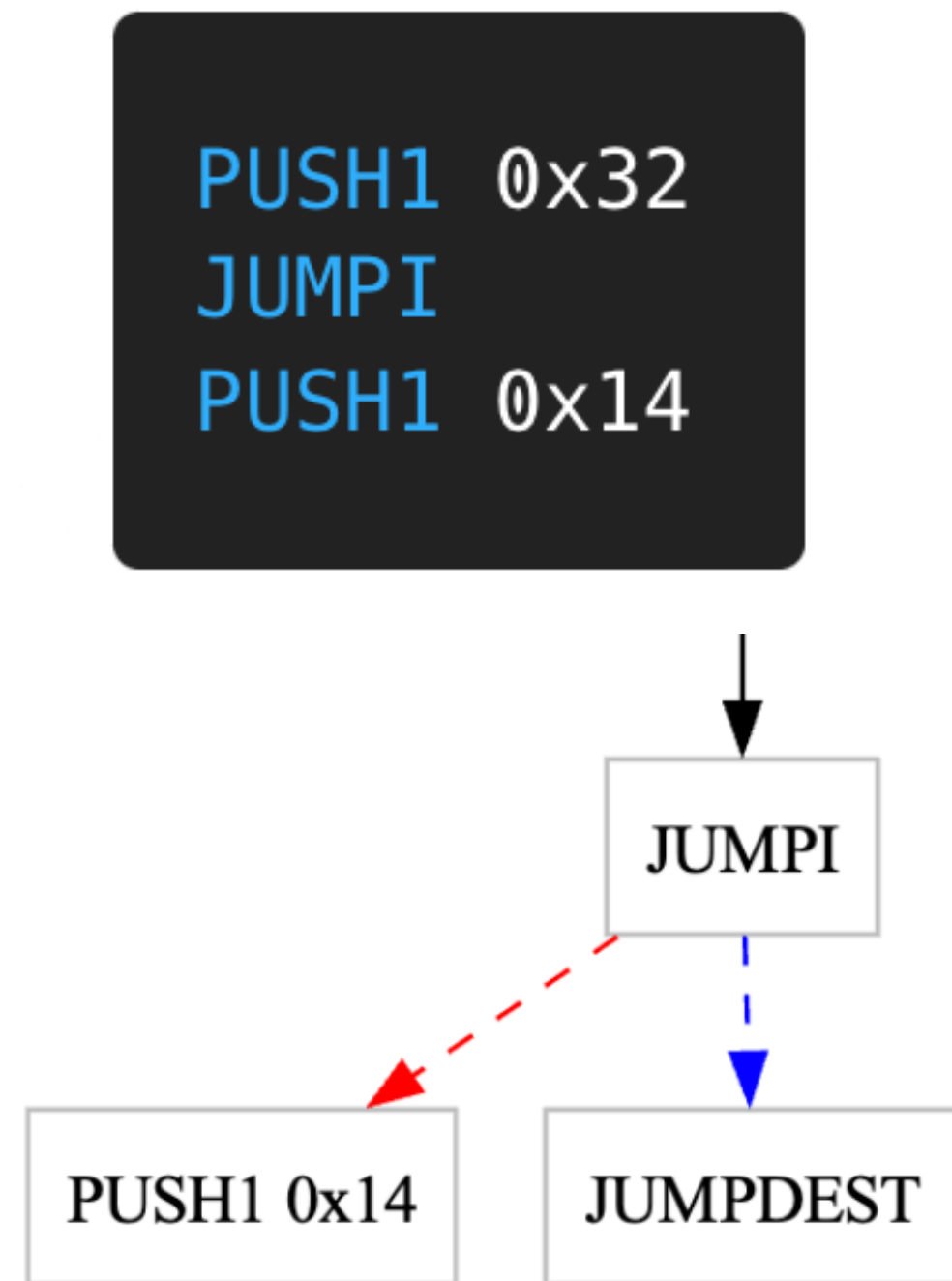


Flusso sequenziale

JUMP & JUMPI

Salto incondizionato

JUMPDEST



Flusso sequenziale

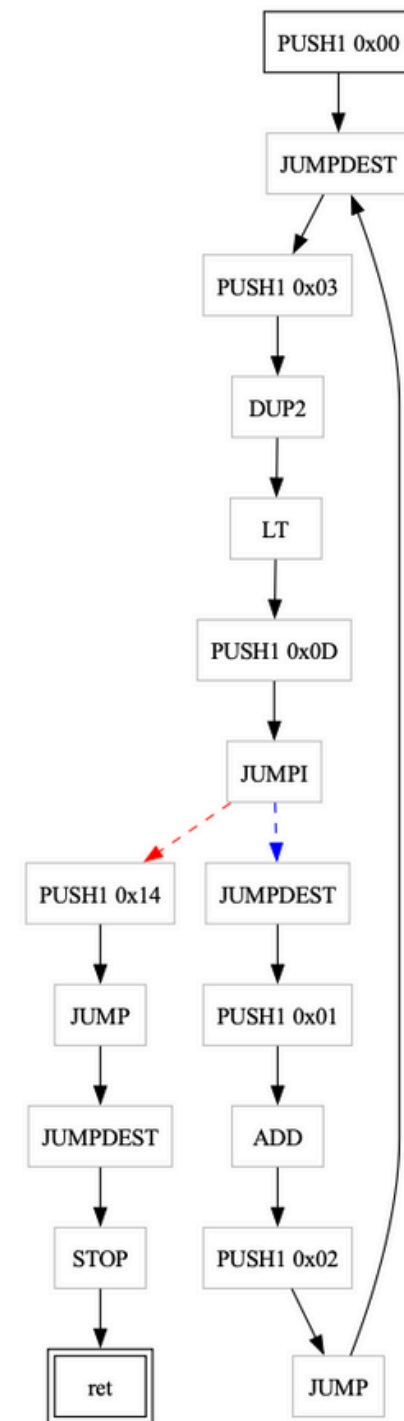
JUMP & JUMPI

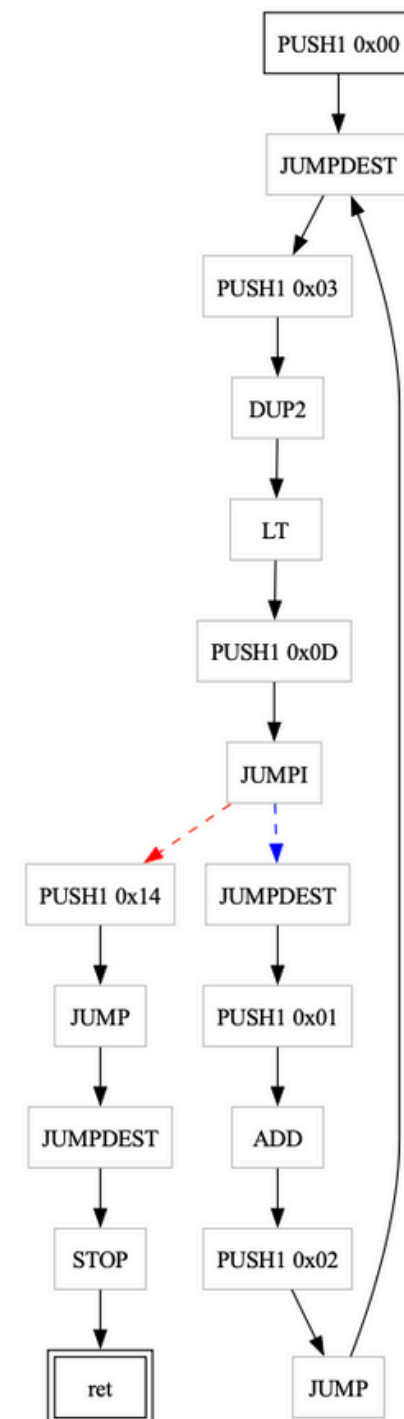
Salto incondizionato

JUMPDEST

Salto condizionato

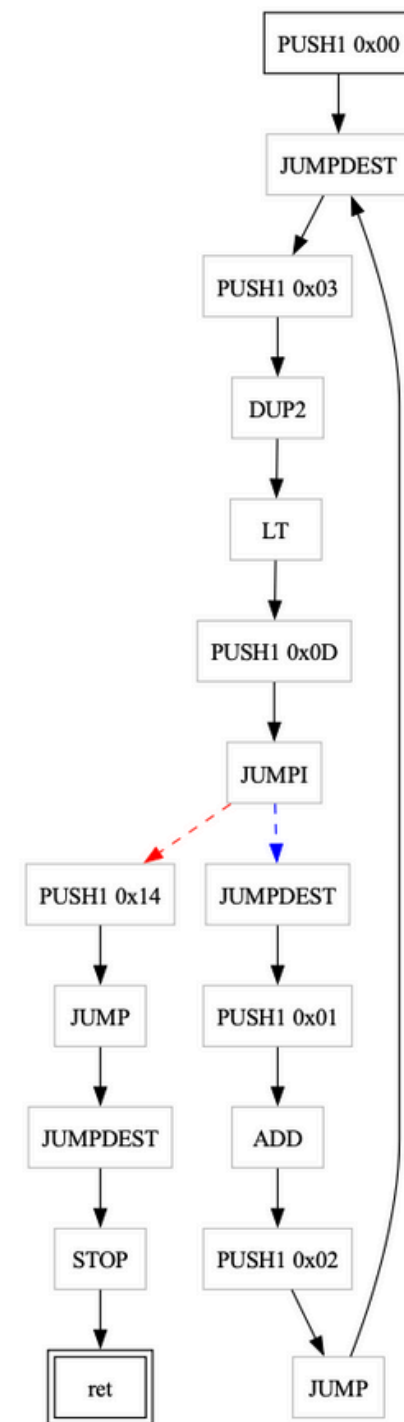
Control-flow Graph





Control-flow Graph

Grafo dei possibili percorsi di esecuzione



Control-flow Graph

Grafo dei possibili percorsi di esecuzione

Sound per un'analisi corretta

JUMP & JUMPI

```
PUSH1 0x01  
PUSH1 0x02  
JUMP
```

JUMP & JUMPI
Pushed jumps

```
PUSH1 0x0A  
PUSH1 0x0C  
ADD  
JUMP
```

JUMP & JUMPI
Pushed jumps
Orphan jumps

```
PUSH1 0x0A  
PUSH1 0x0C  
ADD  
JUMP
```

JUMP & JUMPI

Pushed jumps

Orphan jumps

Vulnerabilità ad attacchi

Smart contract sono software

Smart contract sono software

Soggetti a vulnerabilità ed errori

Smart contract sono software

Soggetti a vulnerabilità ed errori

Analisi statica

Smart contract sono software

Soggetti a vulnerabilità ed errori

Analisi statica

Correttezza senza eseguirlo

Teoria formale

Teoria formale

Approssima oggetti matematici

Teoria formale

Approssima oggetti matematici

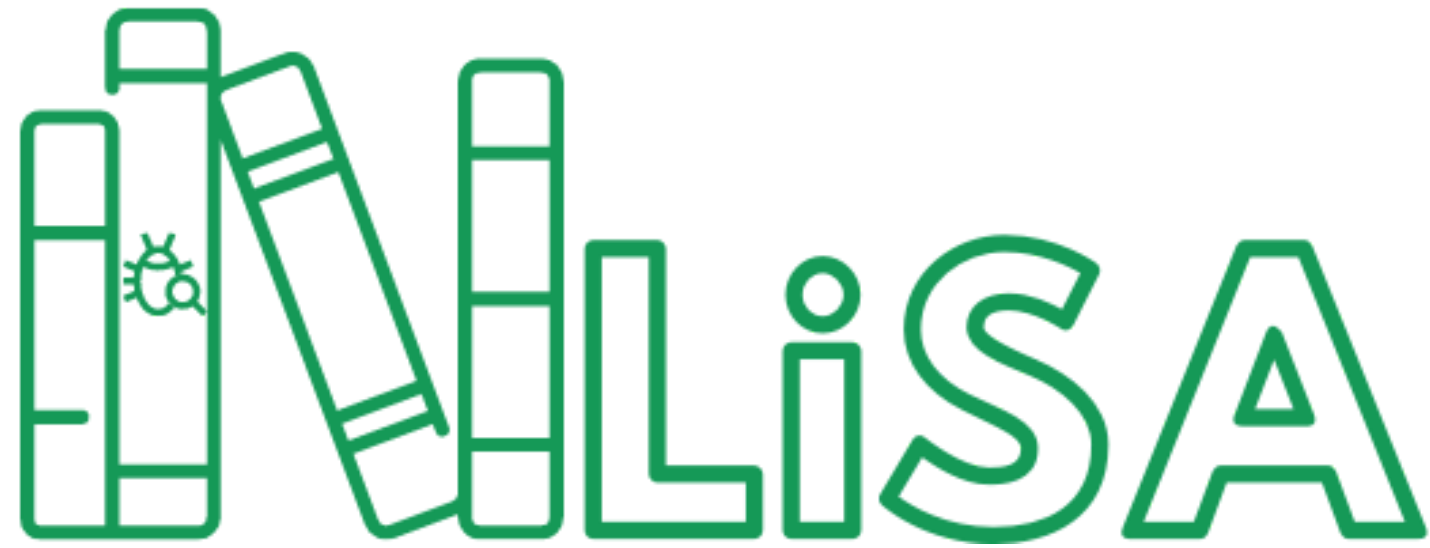
Relazioni

Teoria formale

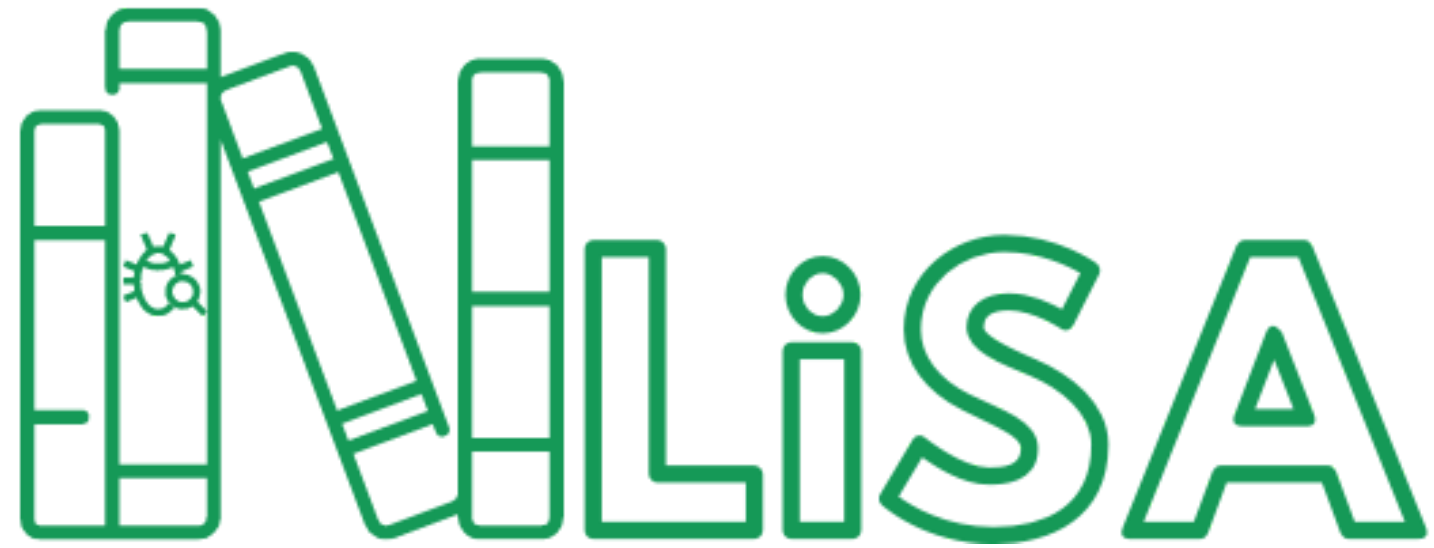
Approssima oggetti matematici

Relazioni

**Approssimare comportamenti concreti
lavorando sull'astrazione**

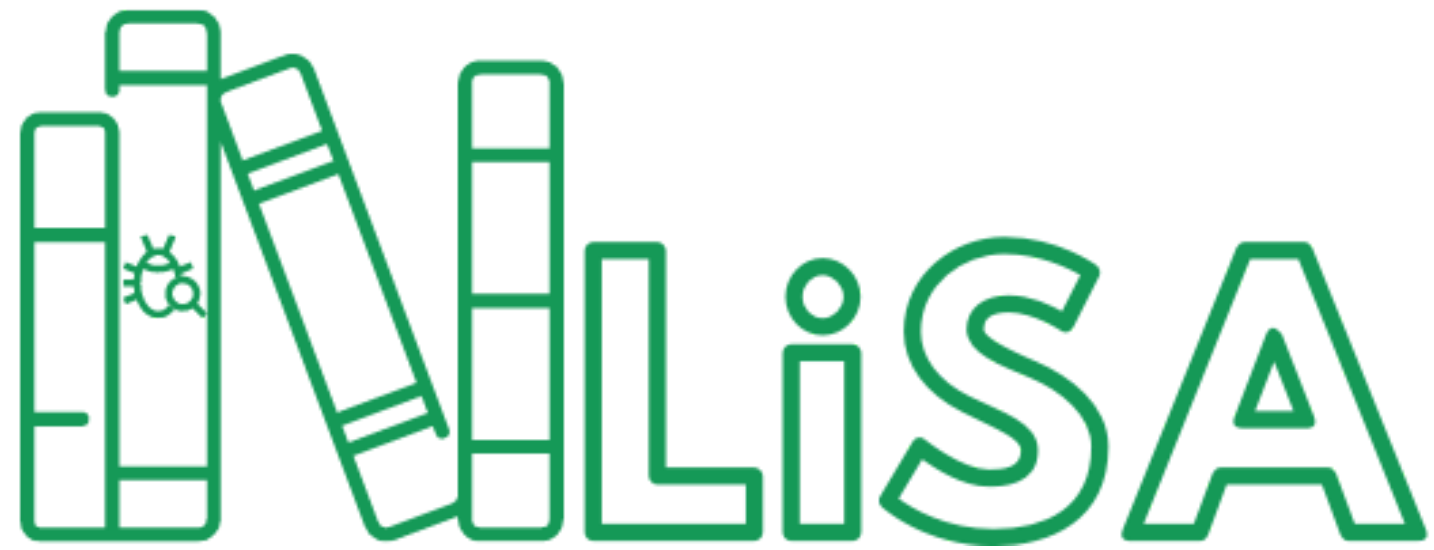


Library for Static Analysis



Library for Static Analysis

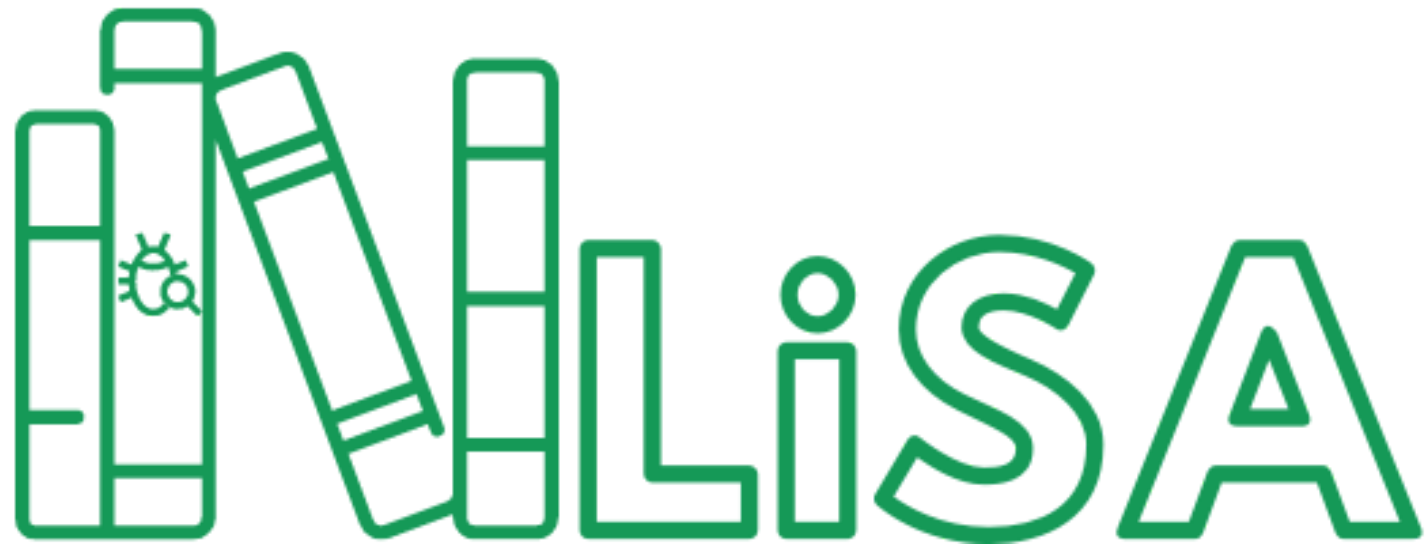
Analizzatori statici



Library for Static Analysis

Analizzatori statici

CFG di un programma

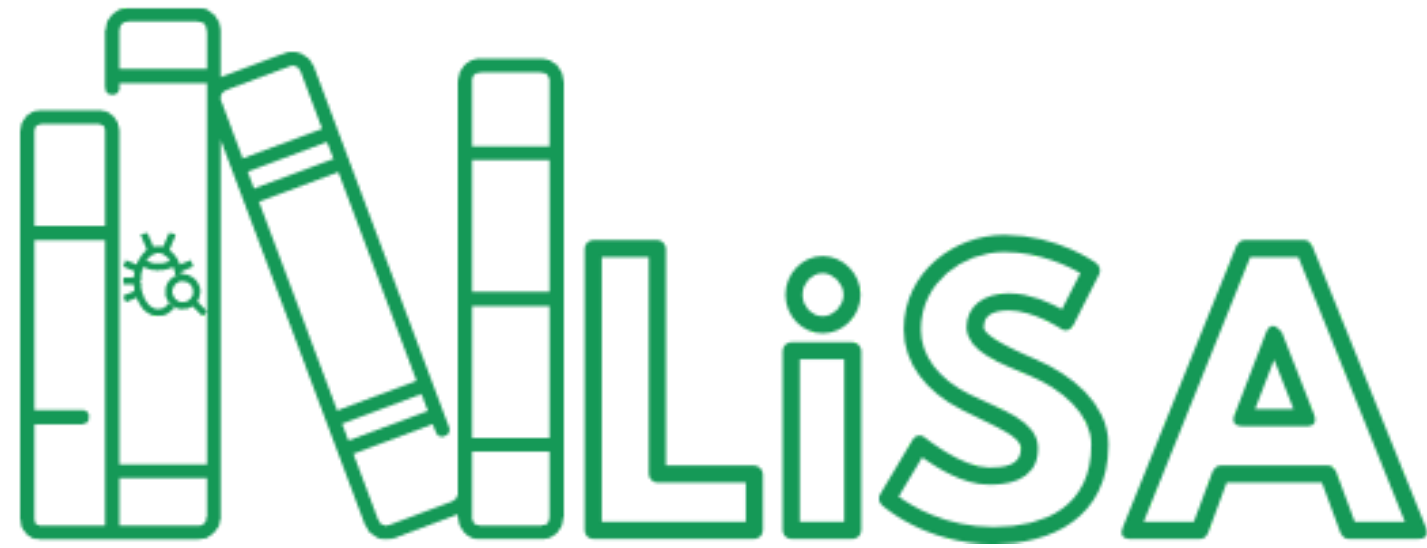


Library for Static Analysis

Analizzatori statici

CFG di un programma

Verifica delle proprietà



Library for Static Analysis

Analizzatori statici

CFG di un programma

Verifica delle proprietà

EVM Bytecode

Legge il bytecode EVM

Legge il bytecode EVM

Analizza il flusso di esecuzione

Legge il bytecode EVM

Analizza il flusso di esecuzione

JUMP genera un *Sequential edge*

Legge il bytecode EVM

Analizza il flusso di esecuzione

JUMP genera un *Sequential edge*

JUMPI genera un *True edge* e un *False edge*

Legge il bytecode EVM

Analizza il flusso di esecuzione

JUMP genera un *Sequential edge*

JUMPI genera un *True edge* e un *False edge*

JUMPDEST

EVM usa stack volatile per elaborare istruzioni

EVM usa stack volatile per elaborare istruzioni

Stack astratto

EVM usa stack volatile per elaborare istruzioni

Stack astratto

Elementi dello stack astratto

$$\text{Ints}_k \triangleq \langle \mathcal{P}_{\leq k}(\mathbb{Z}) \cup \{\top_{\text{Ints}_k}\}, \sqcup_{\text{Ints}_k}, \sqcap_{\text{Ints}_k}, \top_{\text{Ints}_k}, \emptyset \rangle$$

$$\text{Ints}_k \triangleq \langle \mathcal{P}_{\leq k}(\mathbb{Z}) \cup \{\top_{\text{Ints}_k}\}, \sqcup_{\text{Ints}_k}, \sqcap_{\text{Ints}_k}, \top_{\text{Ints}_k}, \emptyset \rangle$$

Insieme di k numeri interi

$$\text{Ints}_k \triangleq \langle \wp_{\leq k}(\mathbb{Z}) \cup \{\top_{\text{Ints}_k}\}, \sqcup_{\text{Ints}_k}, \sqcap_{\text{Ints}_k}, \top_{\text{Ints}_k}, \emptyset \rangle$$

Insieme di k numeri interi

Parzialmente ordinati

$$\text{Ints}_k \triangleq \langle \wp_{\leq k}(\mathbb{Z}) \cup \{\top_{\text{Ints}_k}\}, \sqcup_{\text{Ints}_k}, \sqcap_{\text{Ints}_k}, \top_{\text{Ints}_k}, \emptyset \rangle$$

Insieme di k numeri interi

Parzialmente ordinati

\emptyset elemento inferiore (errore / irraggiungibile)

$$\text{Ints}_k \triangleq \langle \wp_{\leq k}(\mathbb{Z}) \cup \{\top_{\text{Ints}_k}\}, \sqcup_{\text{Ints}_k}, \sqcap_{\text{Ints}_k}, \top_{\text{Ints}_k}, \emptyset \rangle$$

Insieme di k numeri interi

Parzialmente ordinati

\emptyset elemento inferiore (errore / irraggiungibile)

\top_{Ints_k} valore non rappresentabile

$$\mathcal{S}_{\text{Ints}_k, h} \triangleq \{[s_0, s_1, \dots, s_{h-1}] \mid \forall i \in [0, h-1] . s_i \in \text{Ints}_k, h, k > 0\}$$

$$\mathcal{S}_{\text{Ints}_k, h} \triangleq \{[s_0, s_1, \dots, s_{h-1}] \mid \forall i \in [0, h-1] . s_i \in \text{Ints}_k, h, k > 0\}$$

Contiene i primi h elementi Ints_k dello stack

$$\mathcal{S}_{\text{Ints}_k, h} \triangleq \{ [s_0, s_1, \dots, s_{h-1}] \mid \forall i \in [0, h-1] . s_i \in \text{Ints}_k, h, k > 0 \}$$

Contiene i primi h elementi Ints_k dello stack

$$\text{St}_{k, h}^\# \triangleq \langle \mathcal{S}_{\text{Ints}_k, h} \cup \{ \perp_{\text{St}_{k, h}^\#} \}, \sqcup_{\text{St}_{k, h}^\#}, \sqcap_{\text{St}_{k, h}^\#}, \top_{\text{St}_{k, h}^\#}, \perp_{\text{St}_{k, h}^\#} \rangle$$

$$\mathcal{S}_{\text{Ints}_{k,h}} \triangleq \{[s_0, s_1, \dots, s_{h-1}] \mid \forall i \in [0, h-1] . s_i \in \text{Ints}_k, h, k > 0\}$$

Contiene i primi h elementi Ints_k dello stack

$$\text{St}_{k,h}^\# \triangleq \langle \mathcal{S}_{\text{Ints}_{k,h}} \cup \{\perp_{\text{St}_{k,h}^\#}\}, \sqcup_{\text{St}_{k,h}^\#}, \sqcap_{\text{St}_{k,h}^\#}, \top_{\text{St}_{k,h}^\#}, \perp_{\text{St}_{k,h}^\#} \rangle$$

$\perp_{\text{St}_{k,h}^\#}$ **elemento inferiore (*bottom*)**

$$\mathcal{S}_{\text{Ints}_k, h} \triangleq \{ [s_0, s_1, \dots, s_{h-1}] \mid \forall i \in [0, h-1] . s_i \in \text{Ints}_k, h, k > 0 \}$$

Contiene i primi h elementi Ints_k dello stack

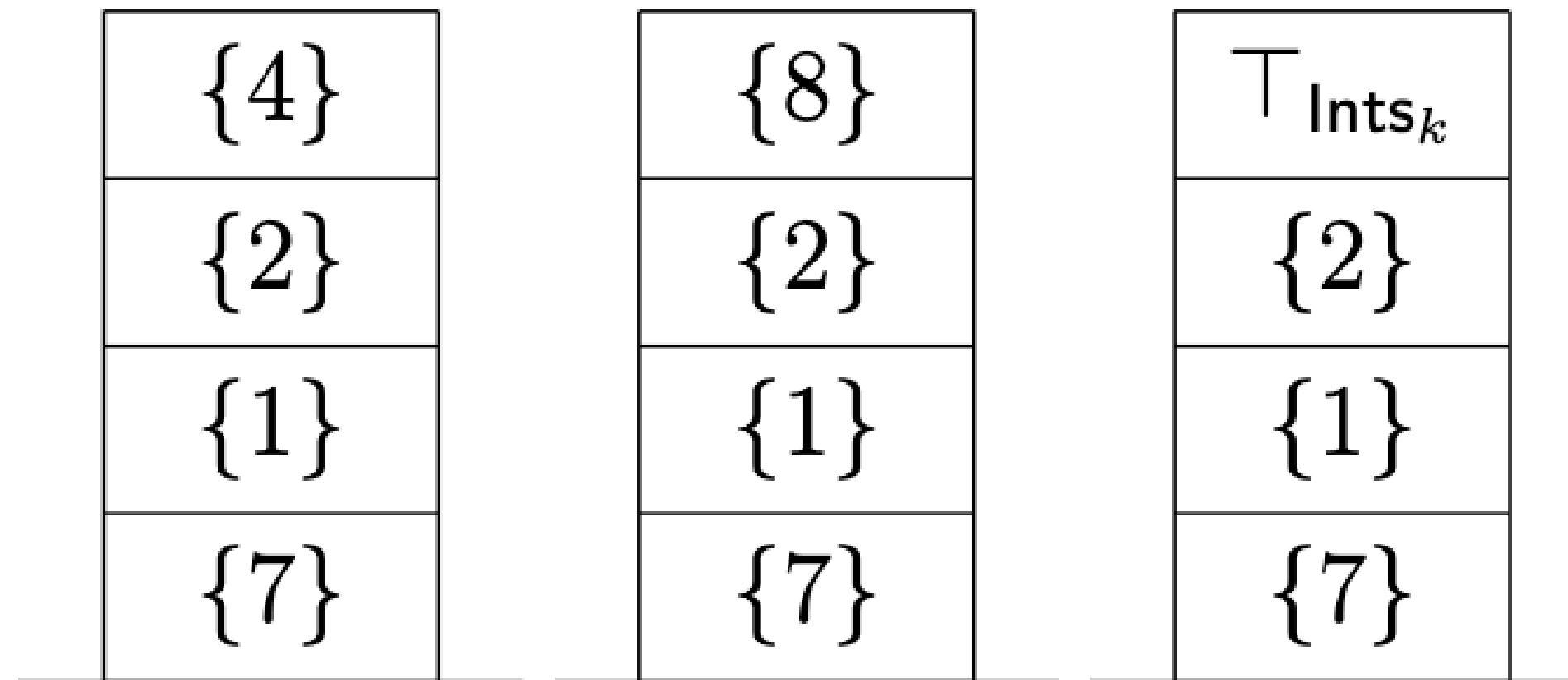
$$\text{St}_{k, h}^\# \triangleq \langle \mathcal{S}_{\text{Ints}_k, h} \cup \{ \perp_{\text{St}_{k, h}^\#} \}, \sqcup_{\text{St}_{k, h}^\#}, \sqcap_{\text{St}_{k, h}^\#}, \top_{\text{St}_{k, h}^\#}, \perp_{\text{St}_{k, h}^\#} \rangle$$

$\perp_{\text{St}_{k, h}^\#}$ **elemento inferiore (*bottom*)**

$$\top_{\text{St}_{k, h}^\#} = [\top_{\text{Ints}_k}, \top_{\text{Ints}_k}, \dots, \top_{\text{Ints}_k}]$$

Maggiore efficacia nella risoluzione delle jump

Maggiore efficacia nella risoluzione delle jump



(a)

(b)

(c)

$$k = 1$$

Maggiore efficacia nella risoluzione delle jump

$$\text{SetSt}_{k,h,l}^{\#} \triangleq \langle \wp_{\leq l}(\mathcal{S}_{\text{Ints}_{k,h}}) \cup \{\top_{\text{SetSt}_{k,h,l}^{\#}}\}, \sqcup_{\text{SetSt}_{k,h,l}^{\#}}, \sqcap_{\text{SetSt}_{k,h,l}^{\#}}, \top_{\text{SetSt}_{k,h,l}^{\#}}, \emptyset \rangle$$

Maggiore efficacia nella risoluzione delle jump

$$\text{SetSt}_{k,h,l}^{\#} \triangleq \langle \wp_{\leq l}(\mathcal{S}_{\text{Ints}_{k,h}}) \cup \{\top_{\text{SetSt}_{k,h,l}^{\#}}\}, \sqcup_{\text{SetSt}_{k,h,l}^{\#}}, \sqcap_{\text{SetSt}_{k,h,l}^{\#}}, \top_{\text{SetSt}_{k,h,l}^{\#}}, \emptyset \rangle$$

Contiene al più l elementi $\mathcal{S}_{\text{Ints}_{k,h}}$

Maggiore efficacia nella risoluzione delle jump

$$\text{SetSt}_{k,h,l}^{\#} \triangleq \langle \wp_{\leq l}(\mathcal{S}_{\text{Ints}_{k,h}}) \cup \{\top_{\text{SetSt}_{k,h,l}^{\#}}\}, \sqcup_{\text{SetSt}_{k,h,l}^{\#}}, \sqcap_{\text{SetSt}_{k,h,l}^{\#}}, \top_{\text{SetSt}_{k,h,l}^{\#}}, \emptyset \rangle$$

Contiene al più l elementi $\mathcal{S}_{\text{Ints}_{k,h}}$

top quando sforiamo la dimensione l

Risoluzione orphan jumps

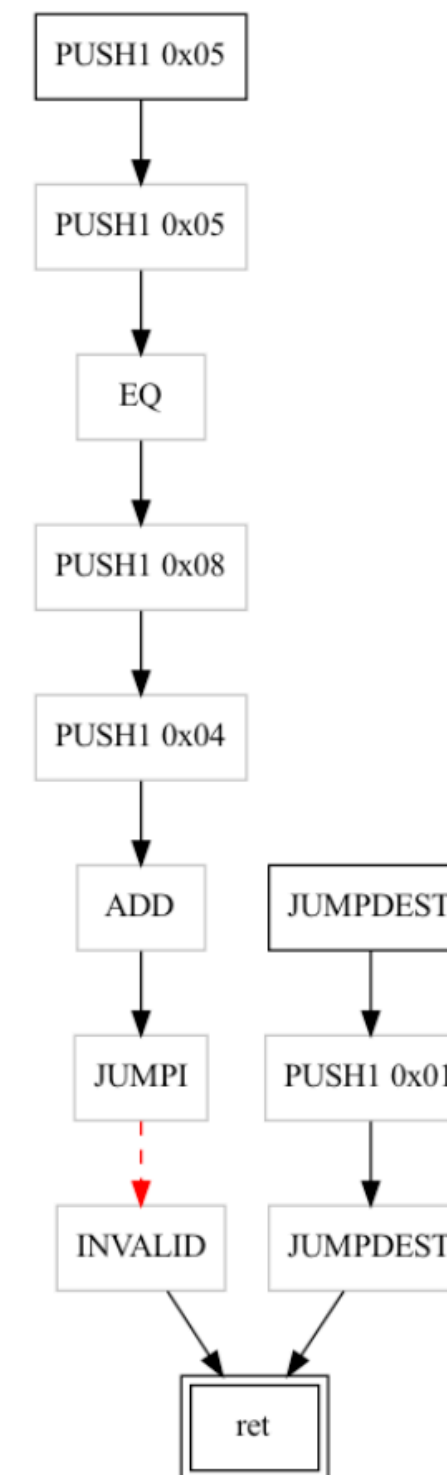


```
[00]    PUSH1 0x05
[02]    PUSH1 0x05
[04]    EQ
[05]    PUSH1 0x08
[07]    PUSH1 0x04
[09]    ADD
[0a]    JUMPI // orphan jump
[0b]    INVALID
[0c]    JUMPDEST
[0d]    PUSH1 0x01
[0f]    JUMPDEST
```

Risoluzione orphan jumps



```
[00] PUSH1 0x05
[02] PUSH1 0x05
[04] EQ
[05] PUSH1 0x08
[07] PUSH1 0x04
[09] ADD
[0a] JUMPI // orphan jump
[0b] INVALID
[0c] JUMPDEST
[0d] PUSH1 0x01
[0f] JUMPDEST
```



Risoluzione orphan jumps



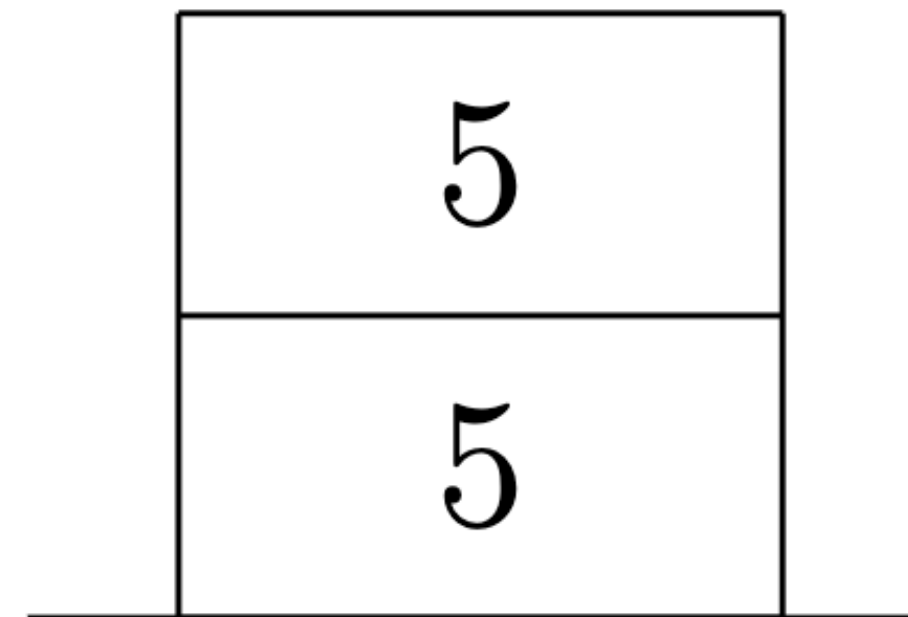
```
[00] PUSH1 0x05
[02] PUSH1 0x05
[04] EQ
[05] PUSH1 0x08
[07] PUSH1 0x04
[09] ADD
[0a] JUMPI // orphan jump
[0b] INVALID
[0c] JUMPDEST
[0d] PUSH1 0x01
[0f] JUMPDEST
```



Risoluzione orphan jumps



```
[00] PUSH1 0x05
[02] PUSH1 0x05
[04] EQ
[05] PUSH1 0x08
[07] PUSH1 0x04
[09] ADD
[0a] JUMPI // orphan jump
[0b] INVALID
[0c] JUMPDEST
[0d] PUSH1 0x01
[0f] JUMPDEST
```



Risoluzione orphan jumps



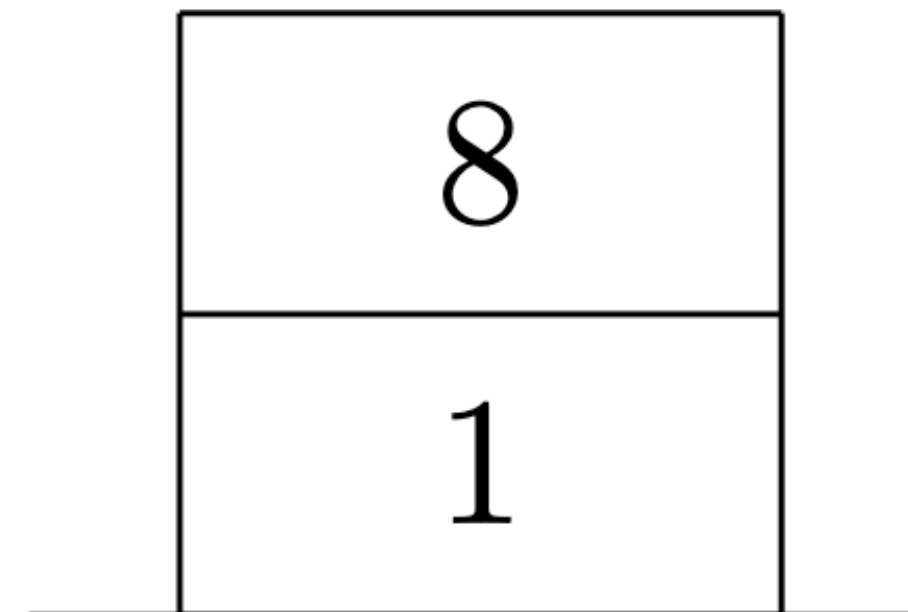
```
[00] PUSH1 0x05  
[02] PUSH1 0x05  
[04] EQ  
[05] PUSH1 0x08  
[07] PUSH1 0x04  
[09] ADD  
[0a] JUMPI // orphan jump  
[0b] INVALID  
[0c] JUMPDEST  
[0d] PUSH1 0x01  
[0f] JUMPDEST
```



Risoluzione orphan jumps



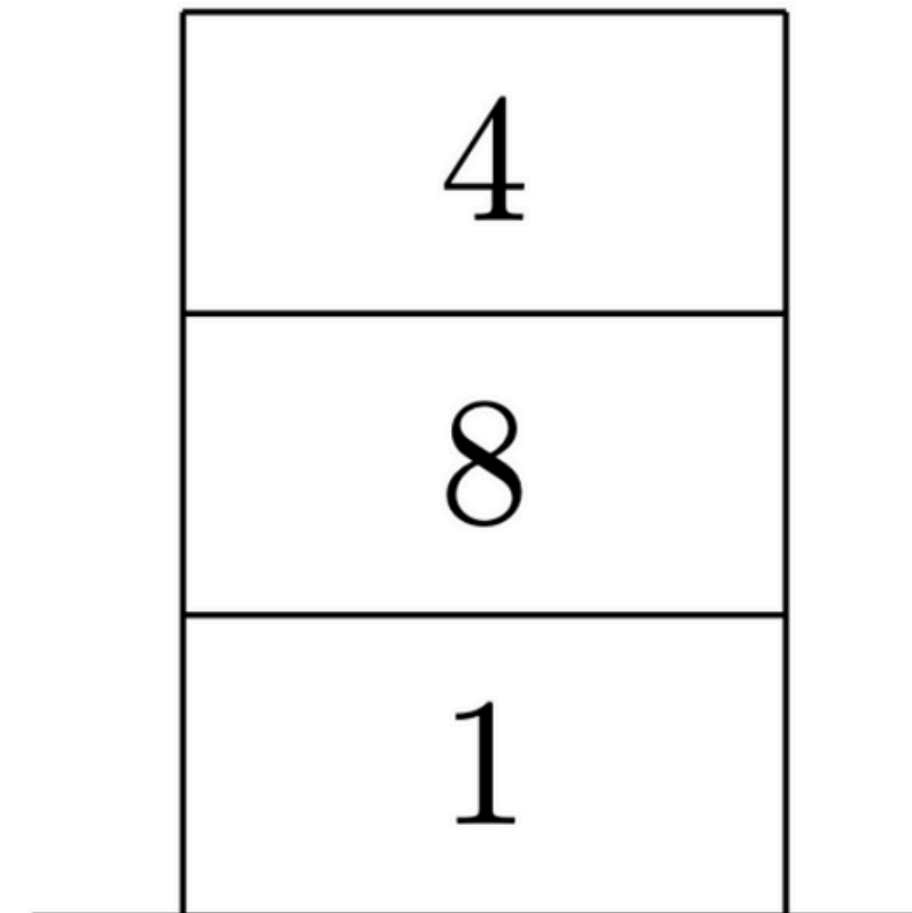
```
[00] PUSH1 0x05
[02] PUSH1 0x05
[04] EQ
[05] PUSH1 0x08
[07] PUSH1 0x04
[09] ADD
[0a] JUMPI // orphan jump
[0b] INVALID
[0c] JUMPDEST
[0d] PUSH1 0x01
[0f] JUMPDEST
```



Risoluzione orphan jumps



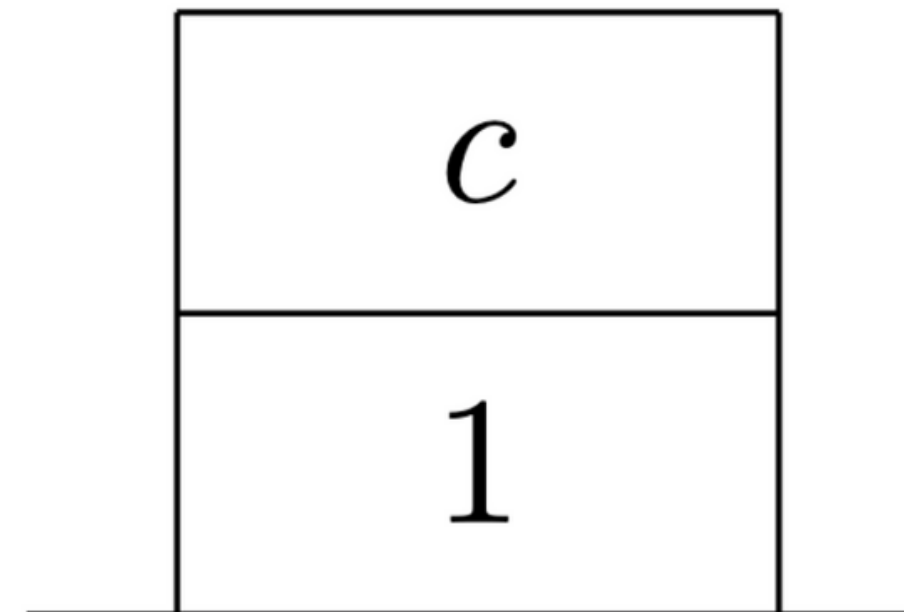
```
[00] PUSH1 0x05
[02] PUSH1 0x05
[04] EQ
[05] PUSH1 0x08
[07] PUSH1 0x04
[09] ADD
[0a] JUMPI // orphan jump
[0b] INVALID
[0c] JUMPDEST
[0d] PUSH1 0x01
[0f] JUMPDEST
```



Risoluzione orphan jumps



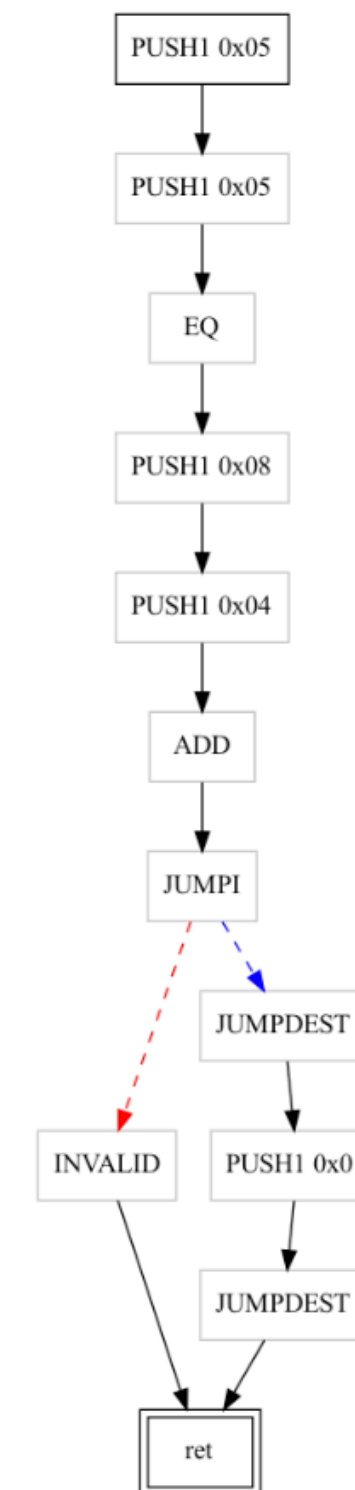
```
[00] PUSH1 0x05
[02] PUSH1 0x05
[04] EQ
[05] PUSH1 0x08
[07] PUSH1 0x04
[09] ADD
[0a] JUMPI // orphan jump
[0b] INVALID
[0c] JUMPDEST
[0d] PUSH1 0x01
[0f] JUMPDEST
```



Risoluzione orphan jumps



```
[00] PUSH1 0x05
[02] PUSH1 0x05
[04] EQ
[05] PUSH1 0x08
[07] PUSH1 0x04
[09] ADD
[0a] JUMPI // orphan jump
[0b] INVALID
[0c] JUMPDEST
[0d] PUSH1 0x01
[0f] JUMPDEST
```



Benchmark su 5000 smart contract da *Etherscan*¹

¹ <https://etherscan.io/>

Valutazione sperimentale



Dimensione StackSet	Jump Risolte	Jump Unsound	Jump Irragg.	Maybe Unsound	Definitely Fake	Maybe Fake	% Jump Solved	Time (sec)
1	1728979	1	243285	315	851	48	99.9999%	3.44
2	1728688	6	223758	333	1125	60	99.9997%	4.24
4	1727825	20	196366	421	1950	84	99.9988%	7.38
8	1726589	15	154845	482	3089	186	99.9991%	14.99
16	1727152	18	129387	520	2321	367	99.9990%	25.97
32	1728251	0	113854	479	1491	137	100%	161.65

$$k = 1, h = 64$$

Benchmark su 5000 smart contract da *Etherscan*¹ Confronto con *Ethersolve*² (94,61%)

¹ <https://etherscan.io/>

² "Enhancing ethereum smart-contracts static analysis by computing a precise control-flow graph of ethereum bytecode", Pasqua et al. [2023, *Journal of Systems and Software*]

Ottimizzazione di EVMLiSA

Ottimizzazione di EVMLiSA

Elaborazione di informazioni esterne

Ottimizzazione di EVMLiSA

Elaborazione di informazioni esterne

**Reentrancy & buffer overflow checker,
gas estimator**

Focus sulla creazione di un CFG sound

Focus sulla creazione di un CFG sound
Implementati diversi domini astratti

Focus sulla creazione di un CFG sound

Implementati diversi domini astratti

Ottenuti ottimi risultati



EVMLiSA



github.com/lisa-analyzer/evm-lisa

Sottomesso articolo al FTfJP 2024