

# Problem Statement

---

Quest for understanding of private blockchain performance

- Design a general benchmark framework to find out to what extent can blockchain handle data processing workload.

# Problem Statement

---

Quest for understanding of private blockchain performance

- Design a general benchmark framework to find out to what extent can blockchain handle data processing workload.

Our framework will:

- Help blockchain application developers to assess blockchain's potentials in meeting the application needs.
- Help blockchain platform developers to identify and improve on the performance bottlenecks.

# Related Works

---

- TPC benchmark series
  - End-to-end macro-benchmarks
  - Focus on relational data model
- Yahoo! Cloud Serving Benchmark (YCSB)
  - For NoSQL data storage
  - To evaluate performance and scalability
- GridMix, PigMix, TeraSort/GraySort, etc.
  - Benchmark for MapReduce-like systems
- BigBench
  - Industry standard end-to-end benchmark
  - For big data processing systems

**No benchmark for private blockchains at the moment**

# Outline

---

- Introduction
  - Backgrounds
  - Problem Statement
  - Related Works
- **BlockBench Framework**
  - System Design
  - Implementation
- Performance Benchmark
  - Macro Benchmarks
  - Micro Benchmarks
- Discussion
- Conclusion

# Challenges

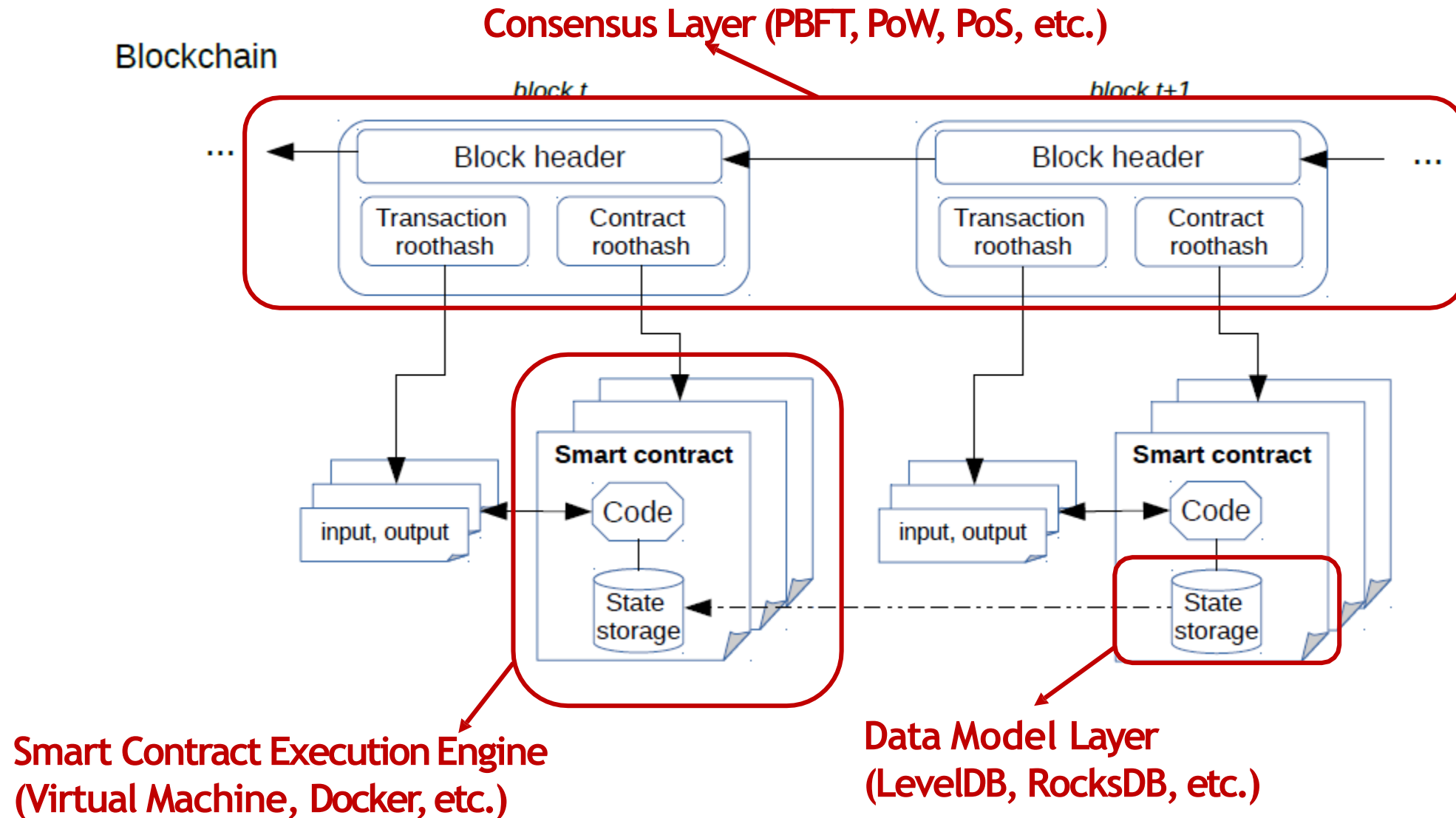
---

- Three main challenges

**Challenge 1:** a blockchain system comprises many parts, we observe that a wide variety of design choices are made among different platforms at almost every single detail.

**Approach:** We extract the common modules of blockchain platform, and divide the blockchain architecture into three modular layers and focus our study on them: the consensus layer, the data model layer and smart-contract execution layer.

# Challenges



# Challenges

---

- Three main challenges

**Challenge 2:** there are many different choices of platforms, but not all of them have reached a mature design, implementation and an established user base.

**Approach:** We start designing BlockBench based on three most mature platforms which support smart-contract functionality, namely **Hyperledger Fabric**, **Ethereum** and **Parity**, and the framework is general to support future platforms.



ethereum



# Challenges

---

- Three main challenges

**Challenge 3:** There is lack of a database-oriented workloads for blockchain.

**Approach:** We treat blockchain as a key-value storage coupled with an engine which can realize both **transactional** and **analytical** functionality via smart contracts.

We design and run both transaction and analytics workloads in our benchmark framework.

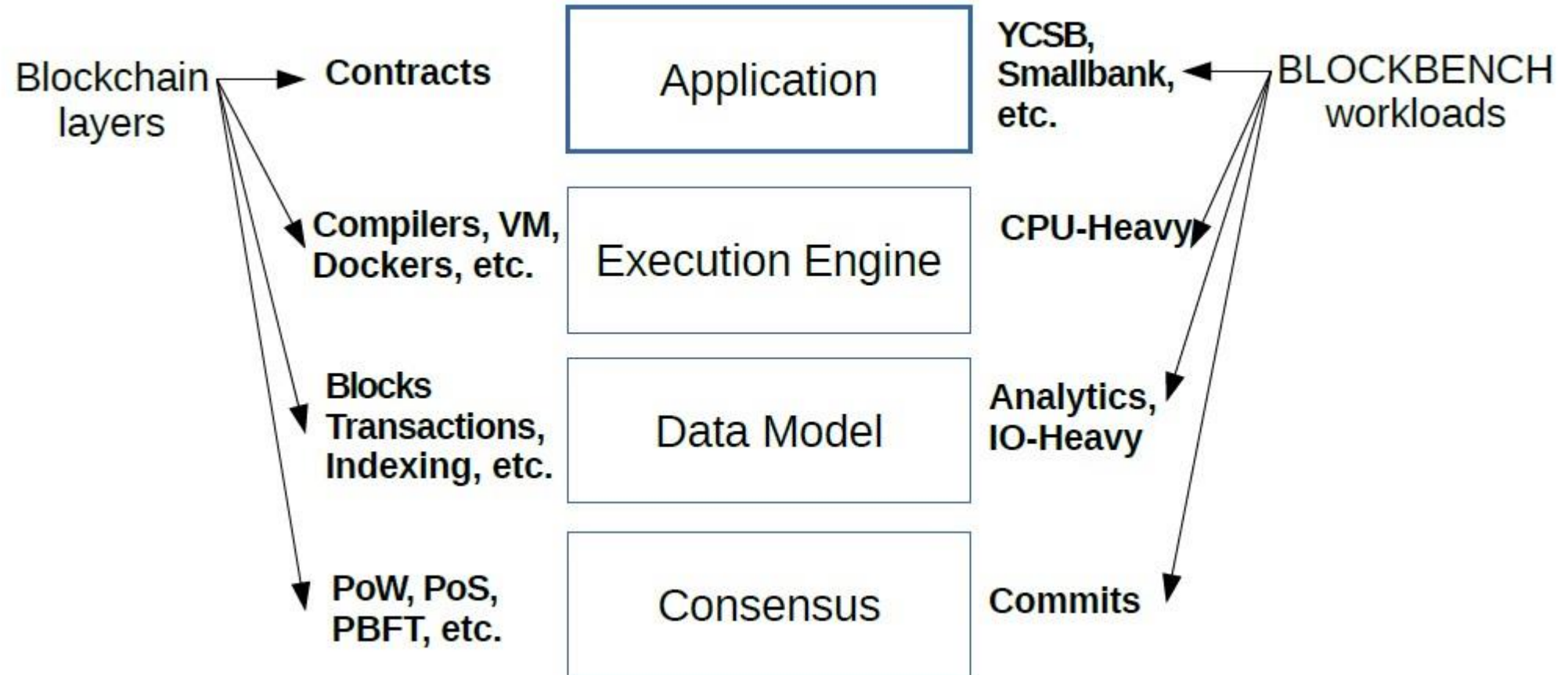


# Workloads

	Smart contracts	Description	
Macro-Benchmarks	YCSB	Key-value store	Storage-oriented
	Smallbank	OLTP workload	
	EtherId	Name registrar contract	
	Doubler	Ponzi scheme	Application-oriented
	WavesPresale	Crowd sale	
Micro-Benchmarks	VersionKVStore	Keep state's versions (Hyperledger only)	Data model
	IOHeavy	Read and write a lot of data	
	CPUHeavy	Sort a large array	→ Execution engine
	DoNothing	Simple contract, do nothing	→ Consensus layer

# Framework Design

---



# Workloads

	Smart contracts	Description	
Macro-Benchmarks	YCSB	Key-value store	Storage-oriented
	Smallbank	OLTP workload	
	EtherId	Name registrar contract	
	Doubler	Ponzi scheme	Application-oriented
	WavesPresale	Crowd sale	
Micro-Benchmarks	VersionKVStore	Keep state's versions (Hyperledger only)	Data model
	IOHeavy	Read and write a lot of data	
	CPUHeavy	Sort a large array	→ Execution engine
	DoNothing	Simple contract, do nothing	→ Consensus layer