SChain: A Scalable Consortium Blockchain Exploiting Intra- and Inter-Block Concurrency

Zhihao Chen, Haizhen Zhuo, Quanqing Xu, Xiaodong Qi, Chengyu Zhu, Zhao Zhang, Cheqing Jin, Aoying Zhou, Ying Yan, Hui Zhang

chenzh@stu.ecnu.edu.cn
Presented at VLDB 2021
Introduction

- Blockchain provides data integrity, traceability and immutability to tackle trust problems among mutually distrusting parties

- **Consortium blockchain** is being widely applied to support large-scale businesses in enterprise collaborations
As users and applications of blockchain proliferate, the system has to **scale** to provide more transaction processing.

1. exploit the parallelism of network, i.e **sharding**
2. enhance the capability of every **single participant**

Cross-shard txn incurs **a large number of** intra- and cross-shard communications

Scale the consortium blockchain in terms of each participant **based on trust domain**
Background

To empower the individual participant

- Fabric incorporate concurrency
  - **High abort rates** for hotspot workloads
  - Enhanced works still inherits the limitations of serial validation

- ParBlockchain and BlockchainDB parallelize the execution
  - Allow non-conflicting transactions to execute in parallel

1. **Limited** to single peer
2. **Overlook** transaction parallelism across multiple blocks
SChain Overview

- System architecture
  - **Scalable** order-execute-finalize (SOEF) paradigm
  - Hybrid trust and fault assumptions
  - Exploit **Intra-** and **Inter-Block** concurrency

---

Fig. 5: Scalable order-execute-finalize paradigm
SChain’s Intra-Block Concurrency

- Multiple executors
  - **Scalable deterministic** concurrency control
  - Early read/write **keys** acquisition for Turing-complete smart contract
  - Guarantee the **merge** of execution result is **equivalent** to the predetermined serial order

Transactions are executed **in parallel** among all executors

**concurrently** within a single executor

---

**Organization 1**

<table>
<thead>
<tr>
<th>Executor E₁</th>
<th>TXID: 1319</th>
</tr>
</thead>
<tbody>
<tr>
<td>R(Tₐ) = {{KeyA, Vₐ}, (KeyC, Vₐ)}</td>
<td></td>
</tr>
<tr>
<td>W(Tₐ) = {{KeyC, Vₐ'}}</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Executor E₂</th>
<th>TXID: 1332</th>
</tr>
</thead>
<tbody>
<tr>
<td>R(Tₐ) = (KeyA, KeyB)</td>
<td></td>
</tr>
<tr>
<td>W(Tₐ) = {{KeyA, Vₐ'}}</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Executor E₃</th>
<th>TXID: 1385</th>
</tr>
</thead>
<tbody>
<tr>
<td>R(Tₐ) = (KeyA, KeyC)</td>
<td></td>
</tr>
<tr>
<td>W(Tₐ) = {{KeyC, Vₐ'}}</td>
<td></td>
</tr>
</tbody>
</table>

---

**Fig.6: Intra-Block Concurrency (within single part.)**
SChain’s Inter-Block Concurrency

- Pipelined workflow
  - Interleave workflows for different blocks
    - no longer block-by-block quiescently
  - Explore the inter-block concurrency
    - allow txns in later blocks to be executed earlier

Non-quiescent workflow
Inter-Block concurrency

Fully-utilized resources

(async commit: keep consistent among participants by reaching consensus on checkpoint periodically)
SChain’s Scalability and Flexibility

- **Ordering**: 
  - Merely order the transactions
  - Concurrent ordering instances (obtain a global(total) order due to trust domain)

- **Execution**: 
  - Devote more executors on demand

- **Finalization**: 
  - Complexity of state partition
  - Expect to design a scalable storage

---

ECNU & ANTCHAIN © Aug. 2021
Conclusion and discussion

- We introduce **SChain**, a scalable consortium blockchain that scales transaction processing by exploiting intra- and inter-block concurrency.

Future works

- Design efficient cache maintenance to leverage data locality
- Explore the scalable state storage
THANKS!