Selected Publications

- **Kun Ren**, Thaddeus Diamond, Daniel Abadi, Alexander Thomson. *"Low-Overhead Asynchronous Checkpointing in Main-Memory Database Systems"*, **SIGMOD 2016**
- **Kun Ren**, Jose Faleiro, Daniel Abadi. *"Design Principles for Scaling Multi-core OLTP Under High Contention"*, **SIGMOD 2016**
- **Kun Ren**, Alexander Thomson, Daniel Abadi. *"VLL: A Lock Manager Redesign for Main Memory Database Systems"*, **VLDB Journal 2015**
- **Kun Ren**, Alexander Thomson, Daniel Abadi. *"An Evaluation of the Advantages and Disadvantages of Deterministic Database Systems"*, **VLDB 2014**
- **Kun Ren**, Alexander Thomson, Daniel Abadi. *"Lightweight Locking for Main Memory Database Systems"*, **VLDB 2013**

Selected Projects

- **CalvinDB Project**, Yale University, USA **2011 – Present**
  - CalvinDB is a scalable transactional database system that leverages determinism to guarantee active replication and full ACID-compliance of distributed transactions without two-phase commit.
  - CalvinDB was implemented by C/C++. Experiments show that Calvin scales near-linearly and has achieved near-world record transactional throughput on a simplified TPC-C benchmark.
  - I am the main contributor and source code available to download at https://github.com/yaledb/calvin.

- **CavlinFS Project**, Yale University, USA **2015 – Present**
  - CalvinFS leverages the idea of deterministic database system CalvinDB to build a scalable geo-replication metadata management layer for large scale distributed file systems.
  - I am the main contributor to build this distributed file system prototype(CalvinFS) from ground up. I am currently working on developing creative ideas to reduce the latency for geo-replication setting.

- **Orthrus Project**, Yale University, USA **2014 – 2016**
  - This project proposes two design principles for multi-core OLTP database systems: separation of database component functionality and advanced planning of transactions.
  - I built a database prototype system Orthrus and experiments show that these two principles alone can significantly improve the scalability on high-contention workloads, and an order of magnitude increase in throughput.

- **Checkpointing Project**, Yale University, USA **2014 – 2016**
  - This project presents CALC and pCALC, novel methods for asynchronous checkpointing that require minimum additional memory usage, no physical point of consistency and extremely low runtime overhead.
  - Experiments show that CALC and pCALC have an overhead that is 2-10X less than alternative approaches.

- **VLL Project**, Yale University, USA **2012 – 2014**
  - This project introduces very lightweight locking(VLL), a protocol for main memory database systems that avoids the costs of maintaining the data structures kept by traditional lock managers.
  - VLL co-locates lock information with the raw data instead of centralizing them. Experiments show that VLL dramatically reduces lock management overhead.