

Final Submission

Current work

<https://github.com/tikv/tikv/pull/8282/files/5855d8d9577b839c0ca03764bf89bee192f7f88a..ec2eb41dc82f7575546da60c2d2a9c2895a4566e>

What is done:

- Add the Put/BatchPut RPC interface, and implement tikv-server side logic
- Add the Get/BatchGet RPC interface, and implement tikv-server side logic with RocksDB iterator
- Implement Delete related RPC functionality, like the delete of RawKV.

What needs to be done:

- Build unit test code for each interface
- Redefine Version region split point, here ensure all versions of a key is located in one region
- Add config about MaxVerNum about VerKV's data, and add GC logic to implement the version recycling
- Add the GetNumWithSV/BatchGetNumWithSV RPC interface, and implement tikv-server logic. Here it needs to use the iterator of RocksDB to get all versions of the key and filter older versions of key-value to return. This interface also supports TTL function
- Implement Scan related RPC functionality. Like the scan of RawKV. It would return a range of key-value of needed versions
- Implement all above RPC API in client-go SDK <https://github.com/tikv/client-go>. Here define the key-value format and need to obtain the ts from PD TSO, then encode on clientgo SDK. For an easy test, mock the server logic in SDK.