

Progress Presentation

- Distributed Sorting -

: Team Blue

Team Info

Surgeon



How we communicated

- Individual task updates on weekly reports
- Weekly Update + Discussion via KakaoTalk

Technical Details

Programming Environment



Windows



IntelliJ IDEA

Language & SDK



Scala 2.13



JDK 17

AI Assistance



Chat GPT

Documentation



Notion

Communication



KakaoTalk



Github

Milestones

Local Sort & Connection

- Generate input data
- Worker can sort gensort data
- Worker sends request to Master
- Master register workers

85%

Sampling & Partitioning

- Worker sends sample keys
- Master distributes partition
- Worker receives partition plan
- Worker labels data by partition

Upcoming

Merge & Fault Tolerance

- Worker sends/receives partitions to/from other workers
- Worker merges assigned partitions
- Worker writes final sorted blocks
- Master verifies merge completion
- Worker saves checkpoint regularly
- Master detects corruption
- New worker reloads from last checkpoint

Pending

Weekly Review

Week 1

- Create Team communication format
- Study about project

Week 2 - 3

- **Build project structure**
- **Appoint Surgeon**
- Research for potential libraries we could use
- **Divide roles(Master, Worker, Integration)**
- Create Notion workspace for documentation

Week 4

- **Finalize Project structure**
- Design Partition and Sampling Workflow
- Basic connection test using gRPC

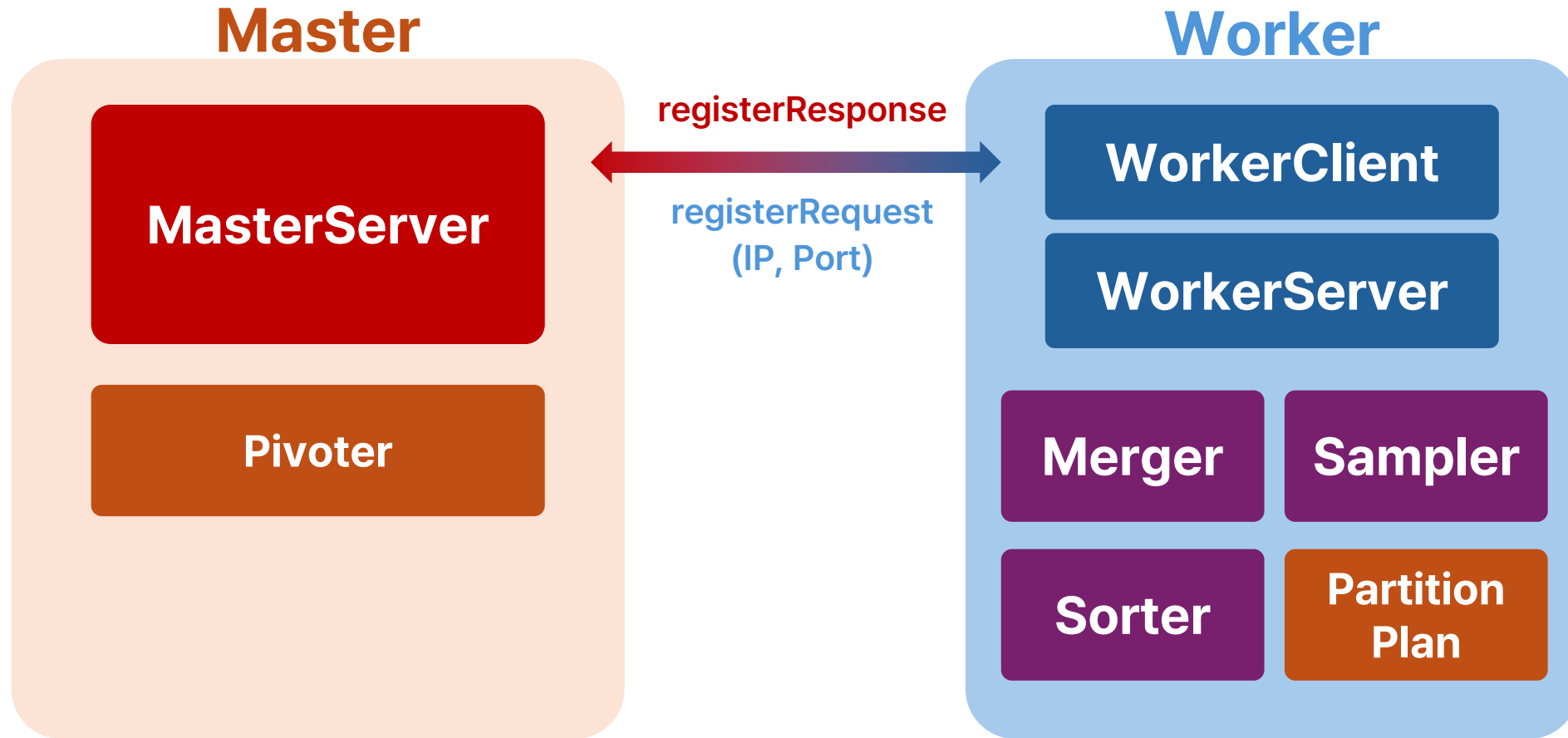
Week 5

- **Implement local sort**
- Implement Master & Worker connection

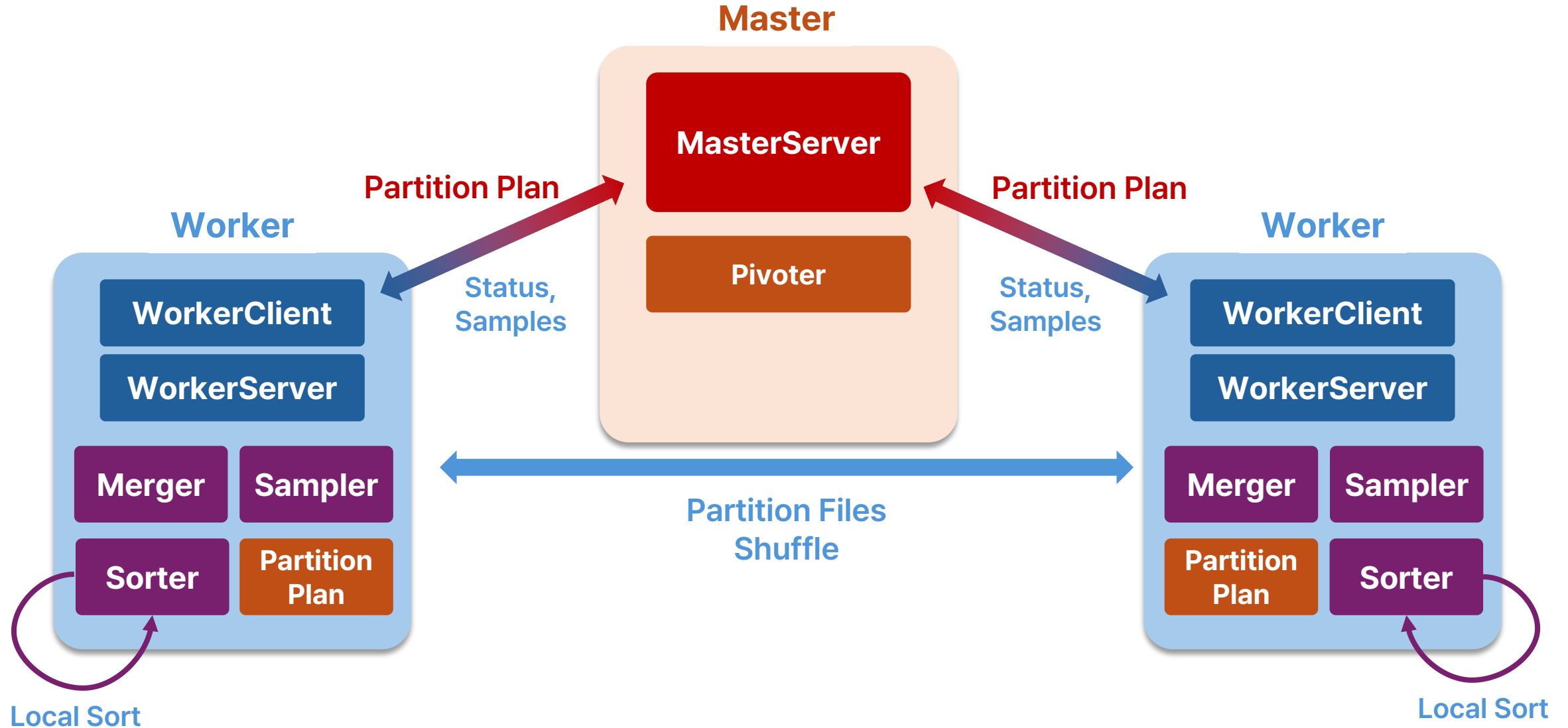
Our Principle

Simplicity
&
Correctness

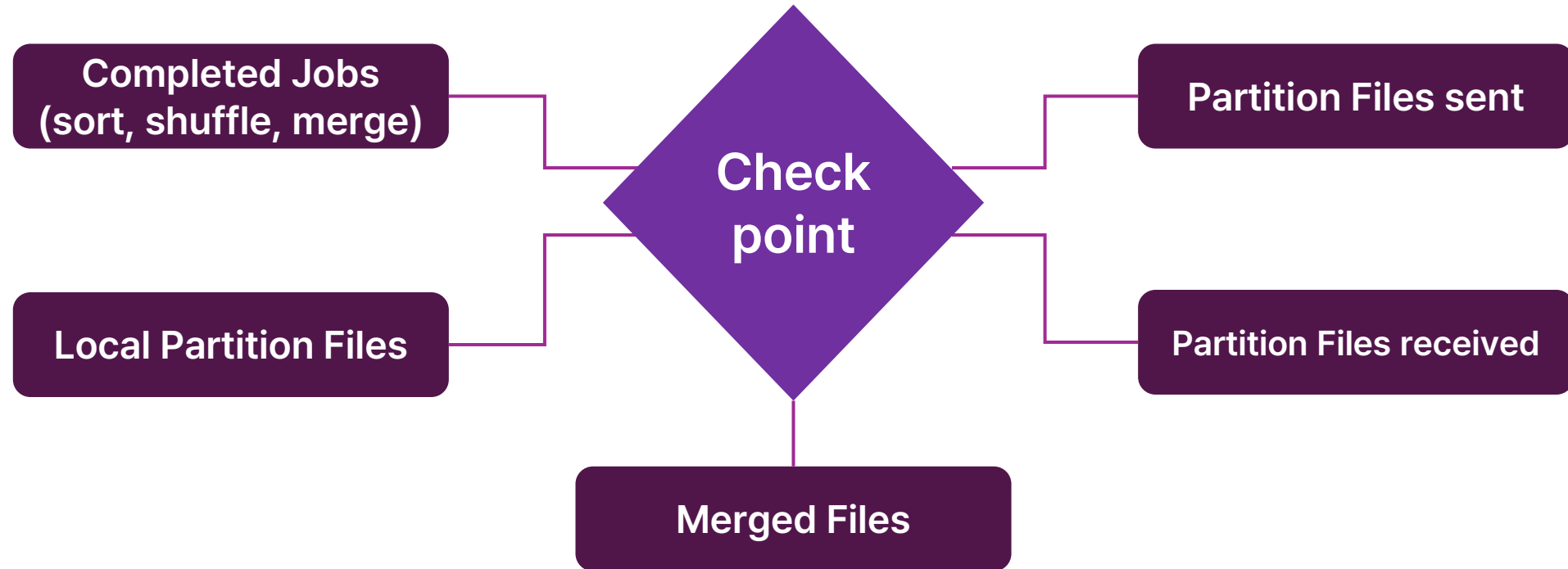
General Design



General Design



General Design – Fault Tolerance



Each Worker periodically stores this state so that after a crash, it resumes exactly from the last consistent point.

Future Tasks

Logging System

How are we going to implement a reliable logging system?

- Centralized logs, consistent format, timestamps, error tracking

Concurrency

How are we going to handle concurrency across Master and Workers?

- thread-safe shared state in Master
- parallel block sorting, concurrent shuffle, checkpoint updates

Developing with AI

PROS

Great for learning
project-related
knowledge

Helpful partner for
design discussions

Improved GitHub
workflow and
repository
management

CONS

Suggestions are not
always consistent

Sometimes gives
designs that are too
complex

A useful tool, but not a
reliable teammate