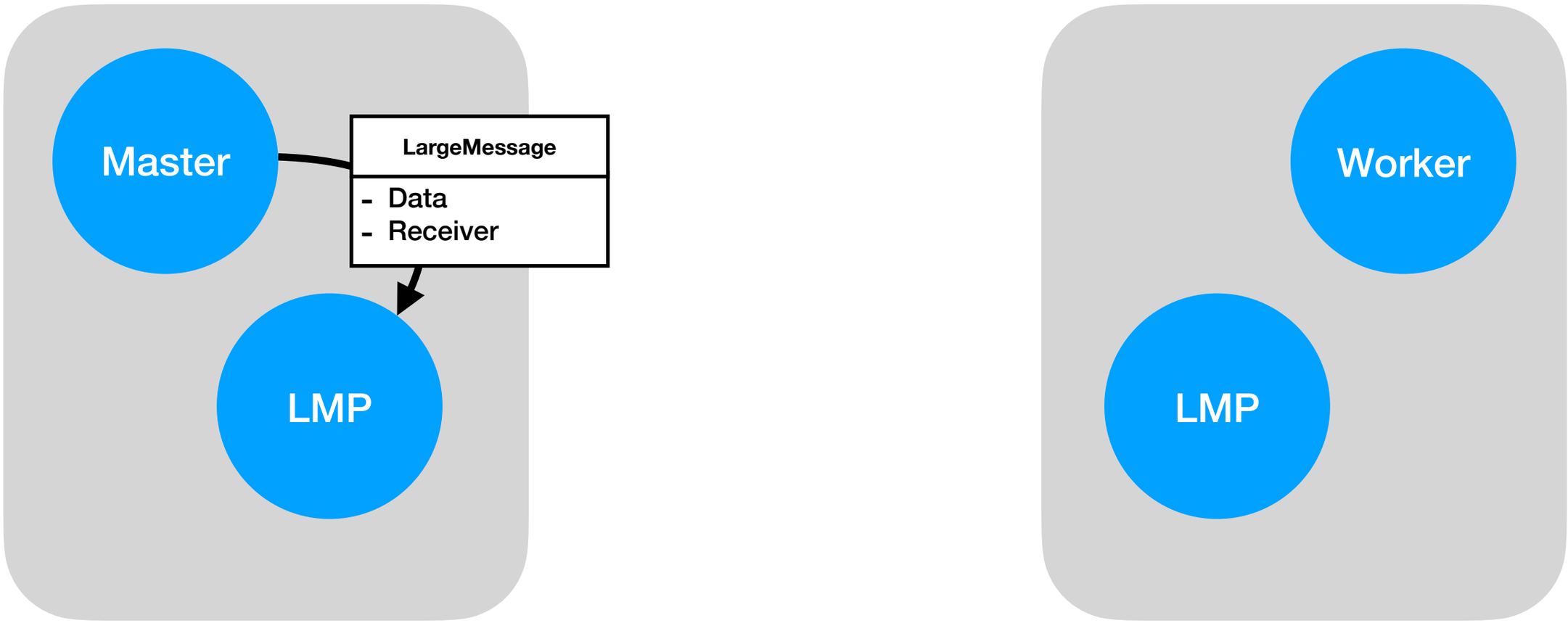
Large Message Proxy

Assignment 2 / Distributed Data Management

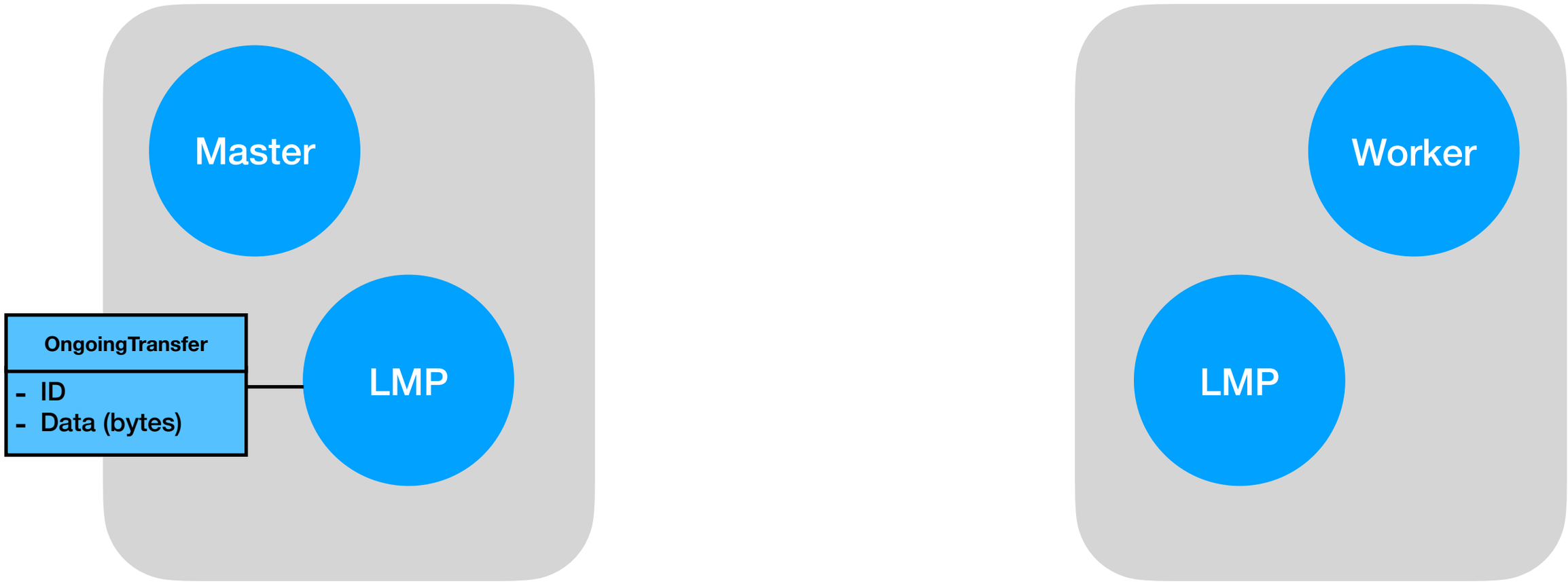
Kenneth Schröder / Jan Siebert

Large Message Proxy



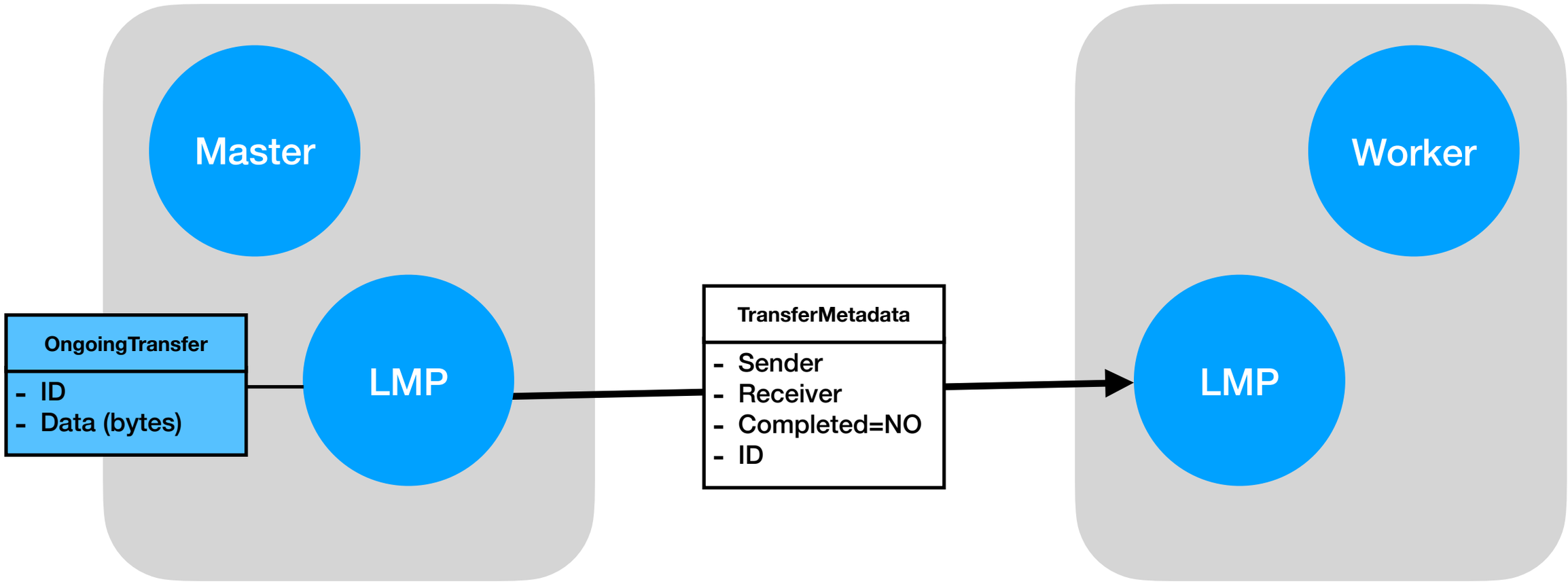
- The Master sends a big object to the “LargeMessageProxy" (LMP) in its System. This is easy, because “Data" is just a pointer here.

Large Message Proxy



- LMP serializes the data and keeps it inside an “OngoingTransfer”-Object. An unique ID is assigned to this transfer.

Large Message Proxy

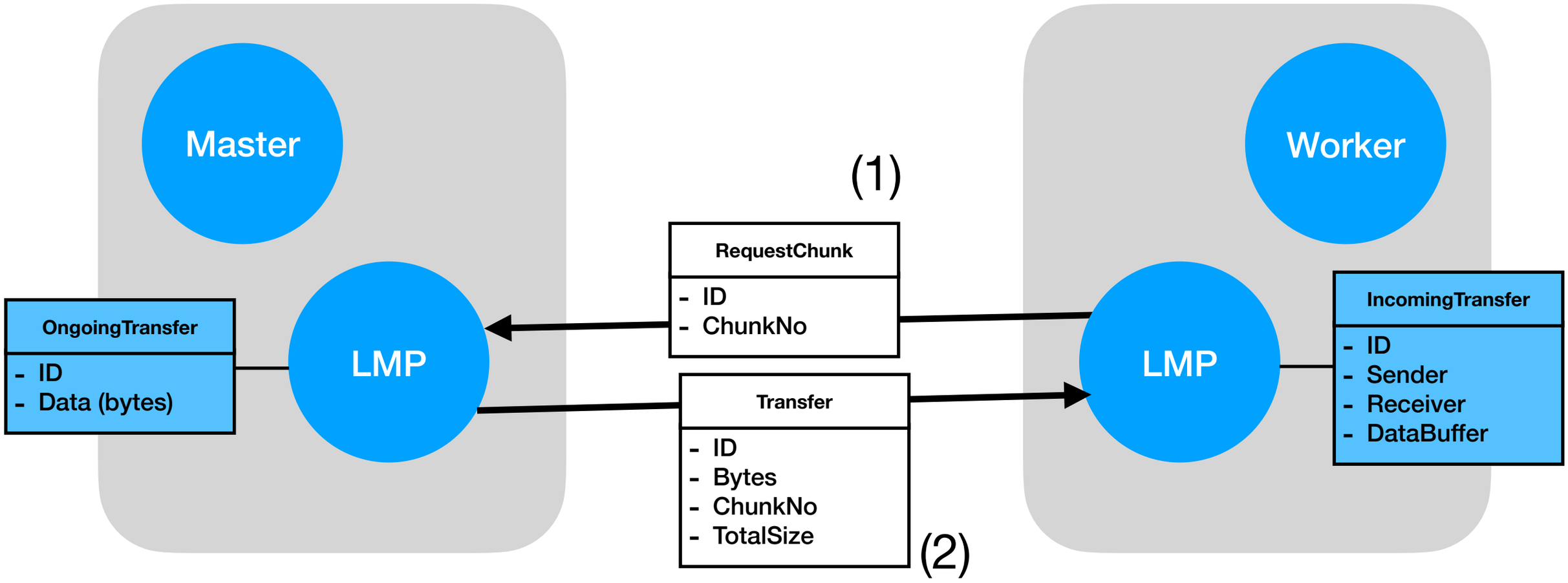


- The LMP looks up the receiving LMP and sends a “TransferMetadata”-Message, which tells the receiving LMP that a new message with its unique ID from "Master" to “Worker" is available.

Large Message Proxy

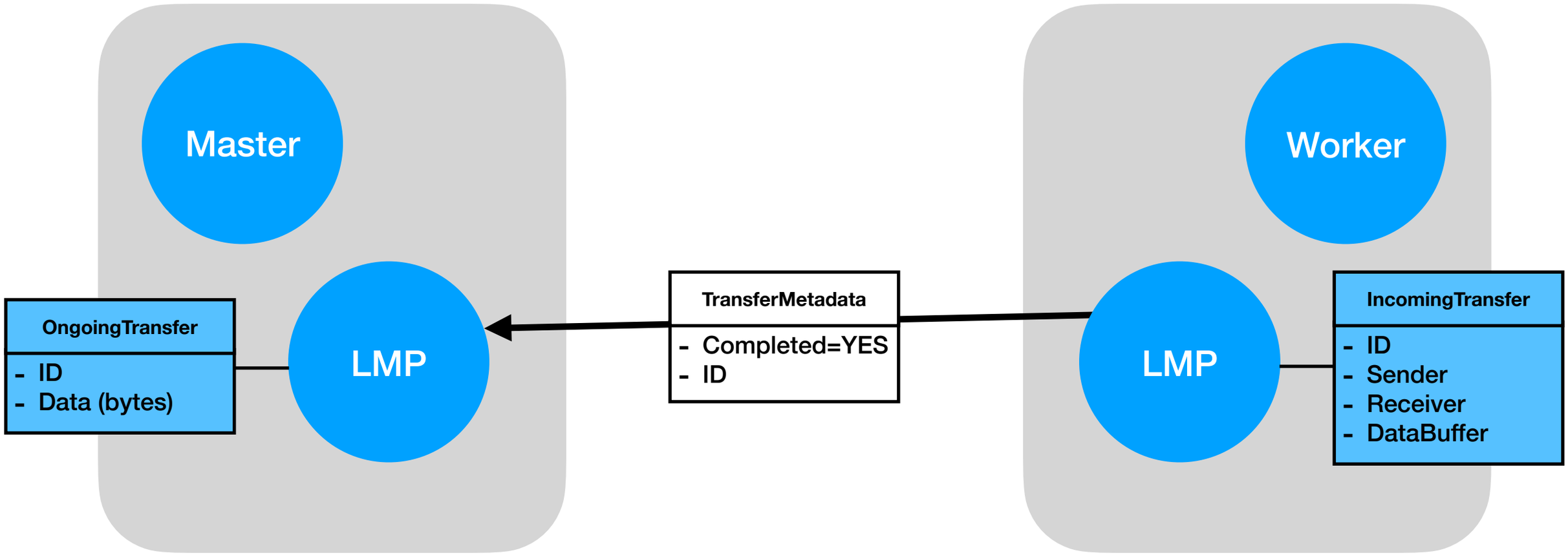
- The receiving LMP creates an “IncomingTransfer”-Object that keeps track of the current progress.

Large Message Proxy



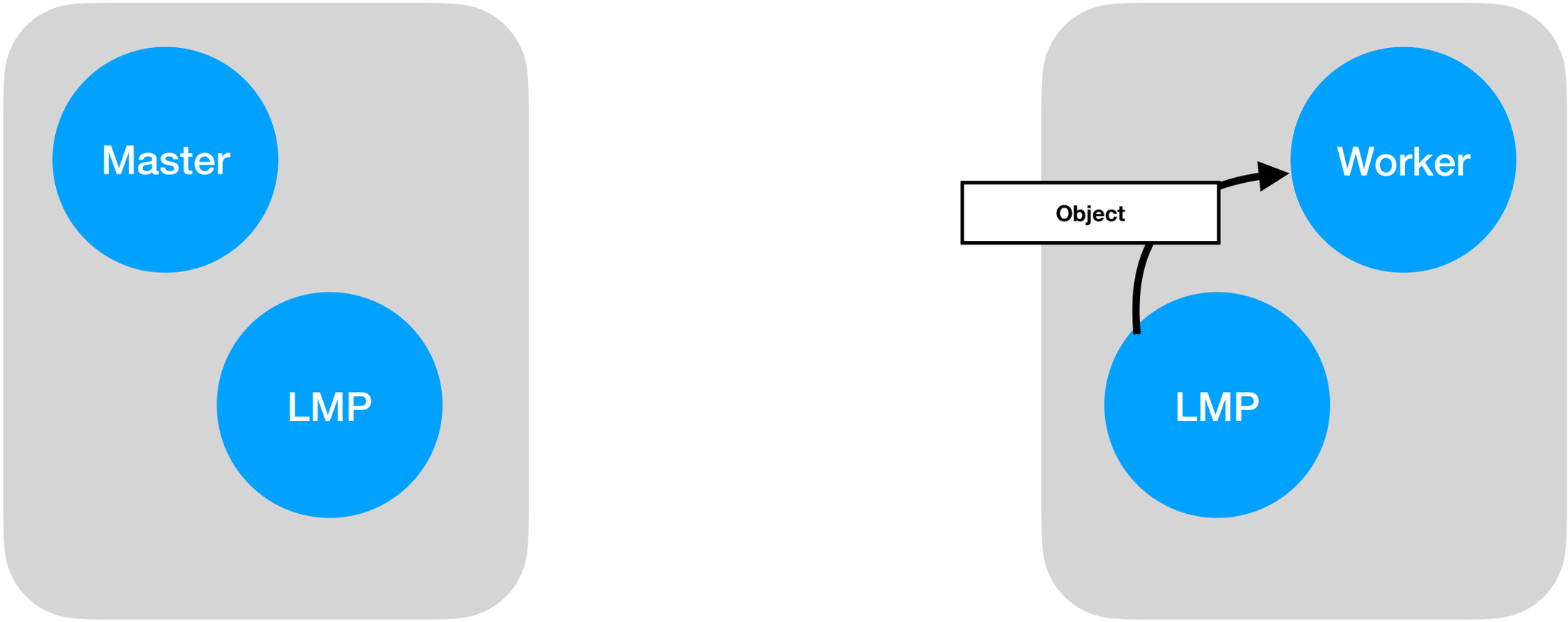
- The receiving LMP requests the next chunk (1) and receives a small amount of data of size 64K (2). This process is repeated until the receiving LMP got all the data. The “Transfer”-Object contains information about the total file size.

Large Message Proxy



- Finally a “TransferMetadata”-Message sent by the receiving LMP signals that the sending LMP can free its memory.

Large Message Proxy



- The receiving LMP deserializes the bytes, restores the object and sends a pointer to it to the “Worker".

Large Message Proxy

Summary

- This solution implements a Pull-based approach to avoid congested mailboxes.
- The unique ID allows multiple ongoing transfers in parallel between the same sender and receiver. In theory each System only needs a single LargeMessageProxy.
- Drawback: The whole message needs to stay in memory during the whole time of the transfer.