



Task 2: Large Message Proxy

Message Types

```
LargeMessage<T>{  
    T message;  
    ActorRef receiver;  
}
```

original message

receiver of original message

```
AckMessage {  
    String messageId  
    int chunkOffset  
}
```

identifier for original message (UUID)

index of message chunk in original message

```
ByteMessage<T>{  
    T bytes;  
    ActorRef sender;  
    ActorRef receiver;  
    int messageLength;  
    String messageId;  
    int chunkOffset;  
}
```

serialized chunk of original message

sender of original message

receiver of original message

total length of original message

identifier for original message (UUID)

index of message chunk in original message

Solution Description



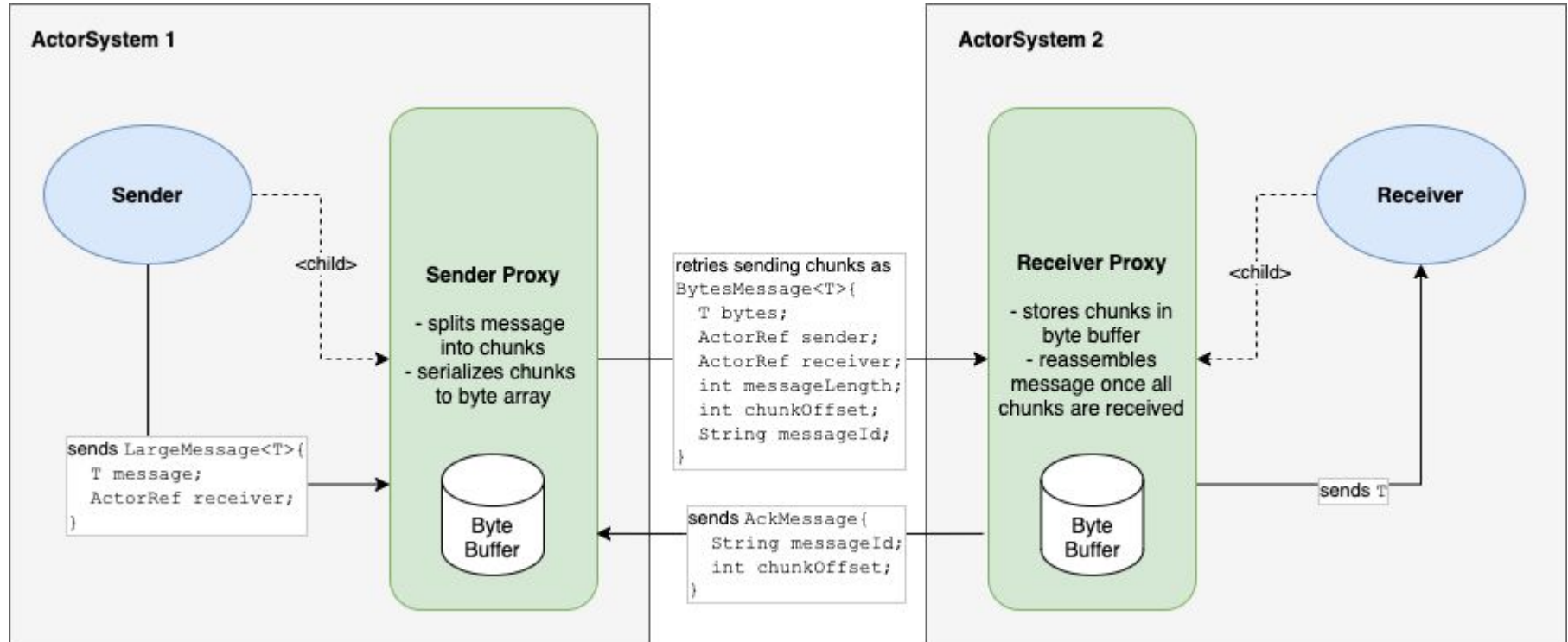
Sender Proxy

- gets `LargeMessage` from Sender
- 1) splits the message into smaller chunks of fixed size
 - `largeMessageChunkSize` configurable in `Configuration`
 - serialized to byte array with `KryoPool`
- 2) separately sends each chunk as `ByteMessage` to Receiver Proxy with retries and stores them in `ByteBuffer` until receive is acknowledged
- 3) on receiving acknowledgement cancels retries and deletes message chunk for `ByteBuffer`

Receiver Proxy

- gets all `ByteMessage` chunks from Sender Proxy
- 1) stores the message chunks in `ByteBuffer` and sends acknowledgement to sender proxy
- 2) checks if all chunks are present
 - if yes:
 - reassembles message's content
 - deserializes it with `KryoPool`
 - sends it to Receiver
 - deletes message chunks from `ByteBuffer`
 - if no: waits for more chunks

Solution Visualization



Critical Reflection of the Solution



- To add more fault tolerance and reliability:
 - Detection of missing messages in receiver proxy
 - Receiver proxy pulling messages instead of sender proxy immediately sending all chunks and retrying them
- The solution is currently pretty slow which causes the Test to fail, so we disabled it
 - WelcomeMessage for workers in master system ~3000-4000 ms
 - WelcomeMessage for 1 worker in worker system ~10 000 ms
 - WelcomeMessage for 2 workers in worker system ~25 000 ms
- We get an error we couldn't solve (the WelcomeMessages arrive anyways after a while):
 - Message [de.hpi.ddm.actors.LargeMessageProxy\$BytesMessage] wrapped in [akka.actor.ActorSelectionMessage] from Actor[akka://ddm/user/master/largeMessageProxy#1969879601] to Actor[akka://ddm@192.168.2.115:7879/] was dropped. Due to overflow of send queue, size [3072]. [1] dead letters encountered. This logging can be turned off or adjusted with configuration settings 'akka.log-dead-letters' and 'akka.log-dead-letters-during-shutdown'.