The initial scope of this project is to build a container to remove the bottleneck between transferring unstructured data between different systems. Unison goal is to provide an API to easily transfer data between large HPC data warehouses. In order to achieve this, we need to provide and Database API to get information about the type of system including the authentication and security implementation. After this phase we need to provide a Parallel algorithm to not only create multiple connections between systems also, provide splitting and preserving the data information in metadata servers.

In our program we focused on the parallel part and assumed that database API is already implemented and ready to go, our approach is to use an external node act as a parallel manager to use Restful GET or PUT which originally requested by user and transfer data between two systems.

Scenario one:

User wants to request a file to store in local lustre system, the initial process is to define the source and destination including the name of the file, second system our API will be decide on the number of nodes to use depending on the size of the file or the number of available nodes. File will be split into multiple segments and each segment will fire a external process to be executed. Finally, multiple connection to system will be initiated and transfer process will be start.

Scenario Two:

User wants to put new file into either google cloud or any other system such as parallel system, program generates a unique identifier based on hash function and gets the active nodes lists, after this process program splits that file based on number of nodes, size, or requested number and creates same number of connection to destination ( if possible) then puts those segments individually into sets of nodes to improve the throughput and use the maximum resources.