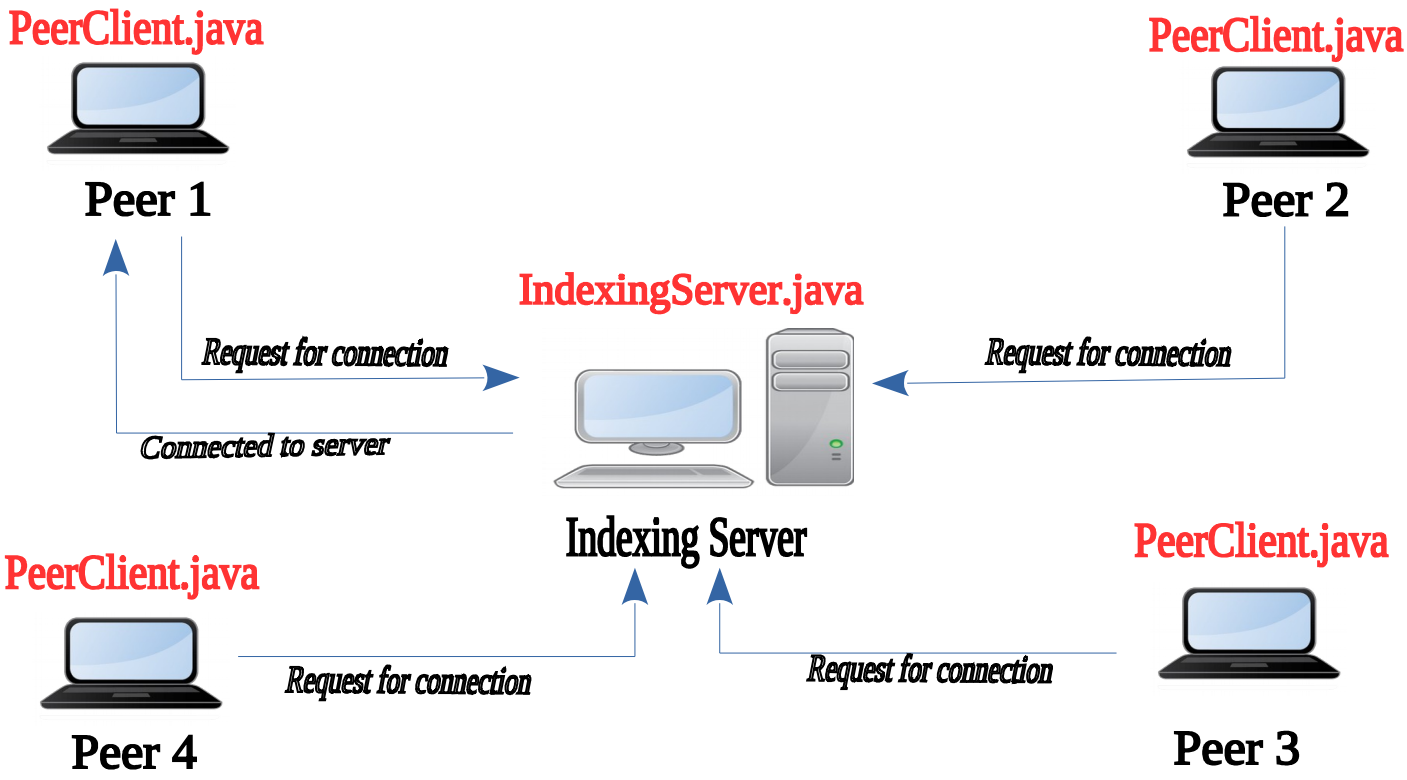
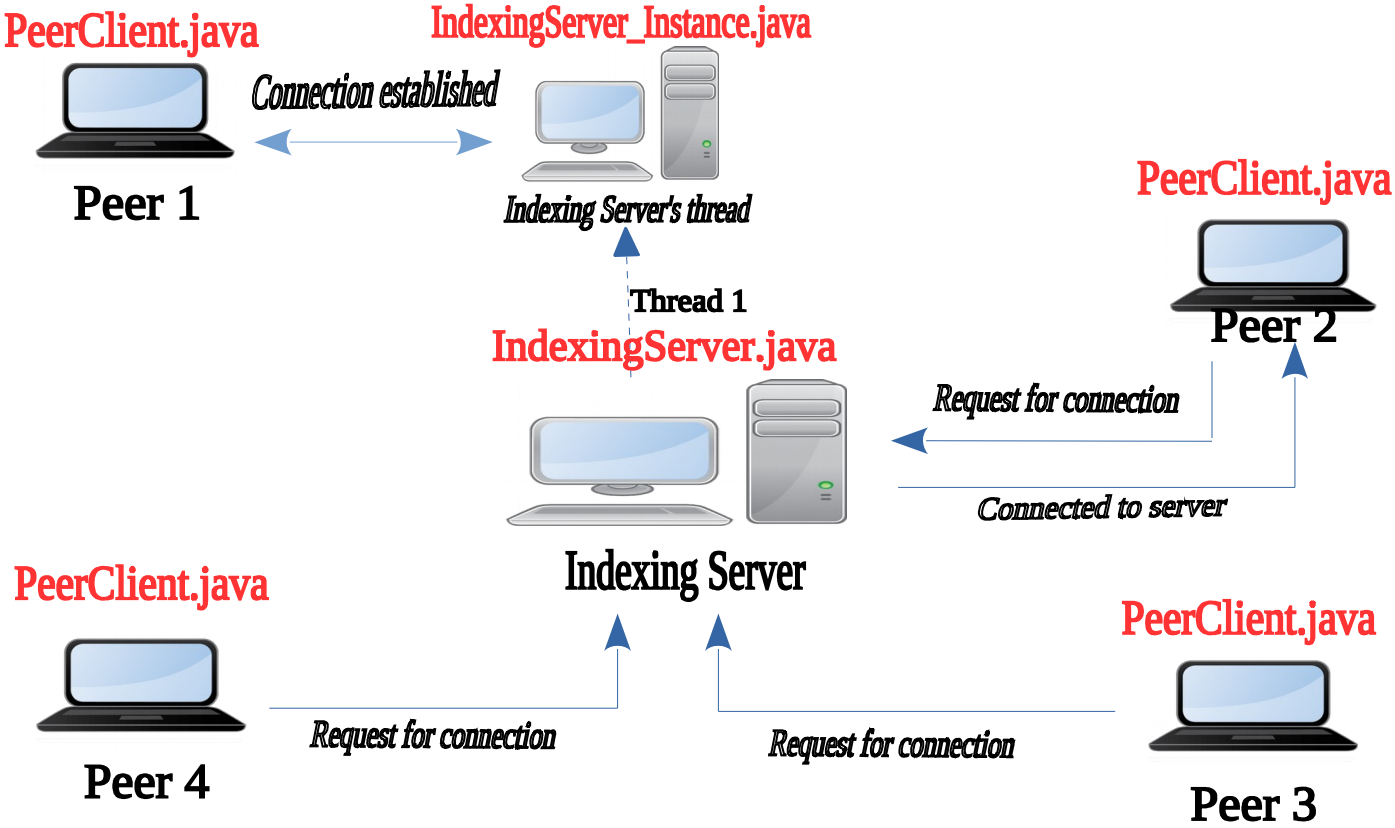
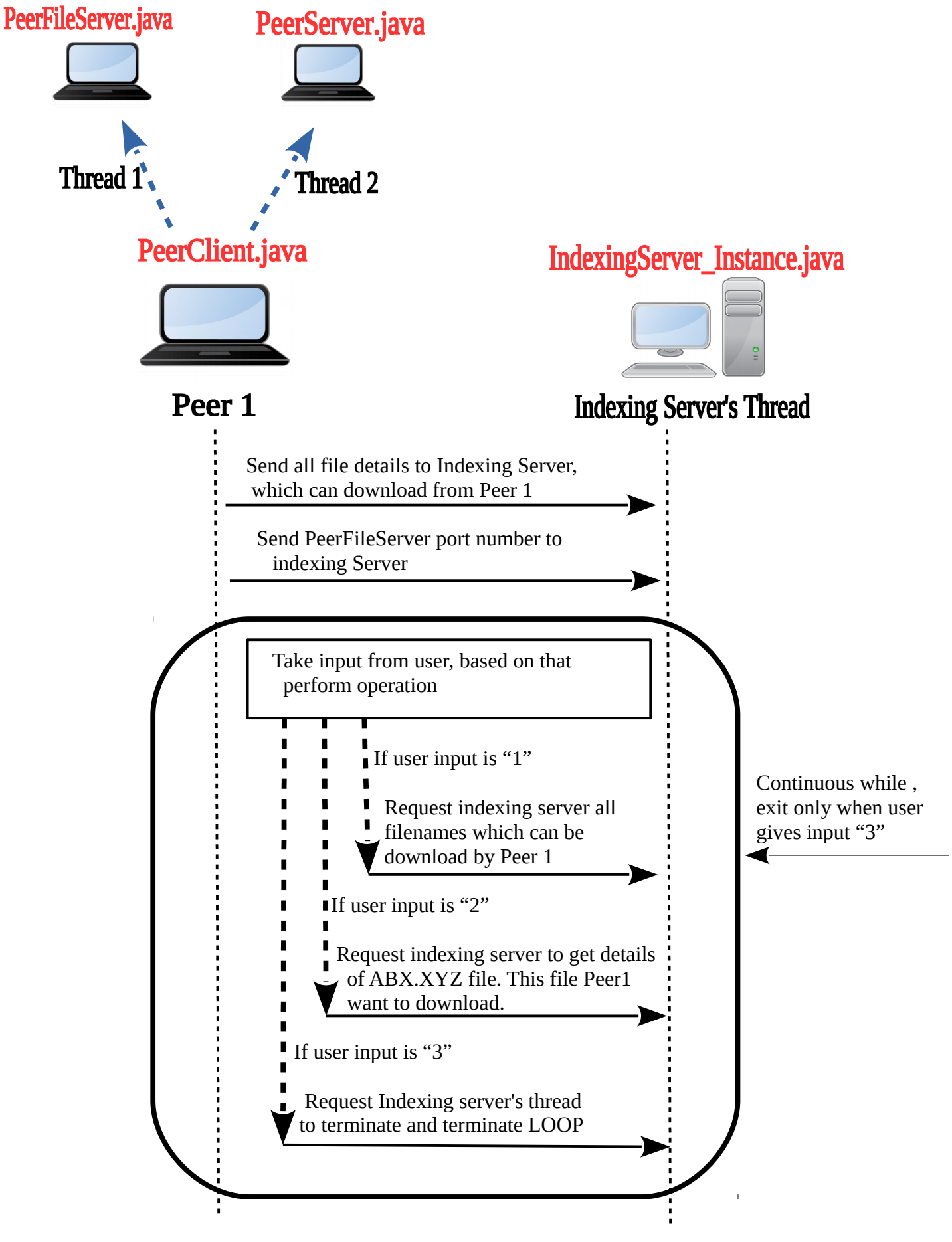
Step 1 : Four Peers try to connect with Indexing Server.
At first, indexing server connects to Peer 1.



Step 2: Indexing Server creates **thread** to provide service to Peer,
and resume back to take request from Peer 2.

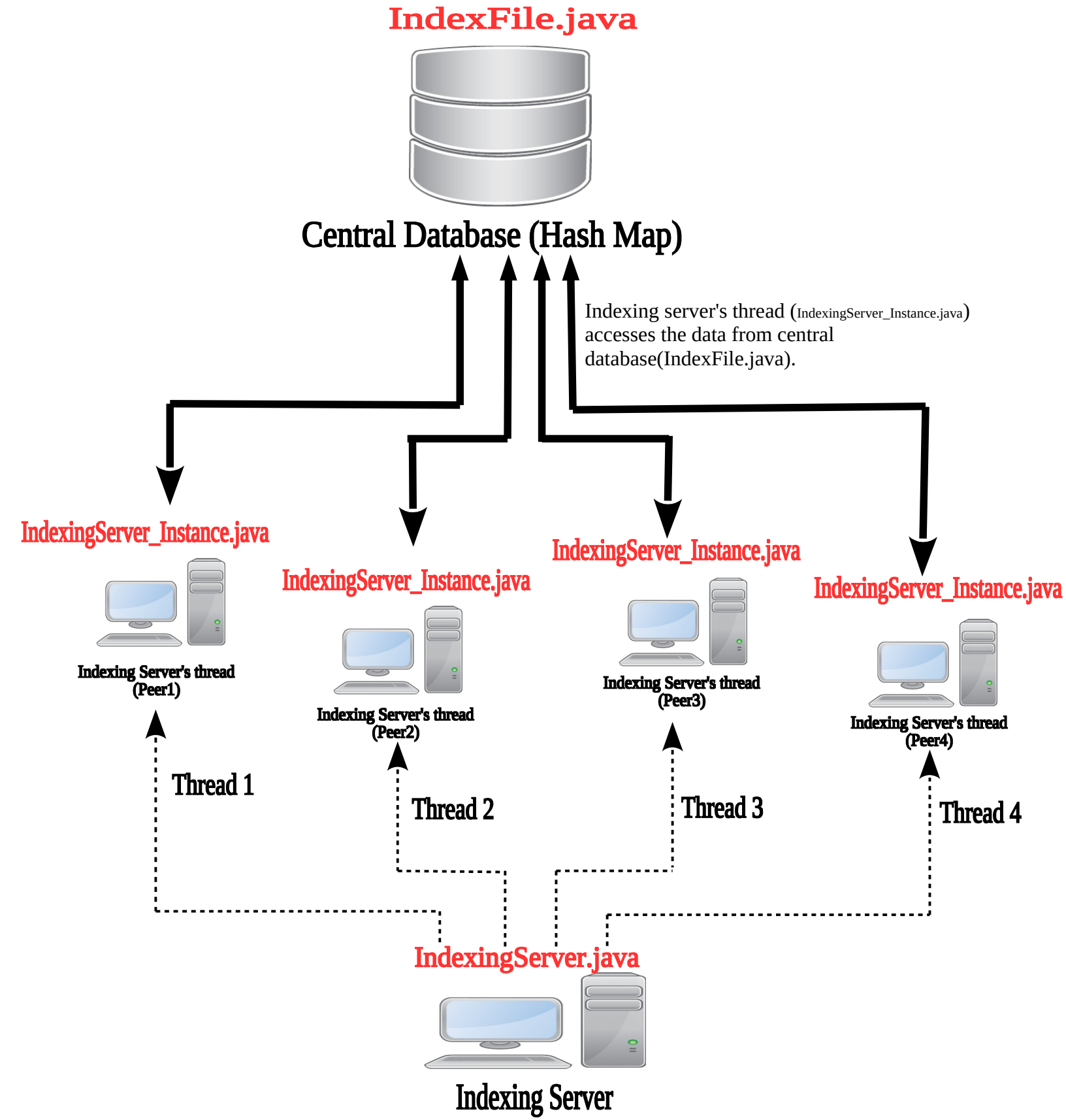


Step 3 : Details communication between Peer 1 and Indexing Server



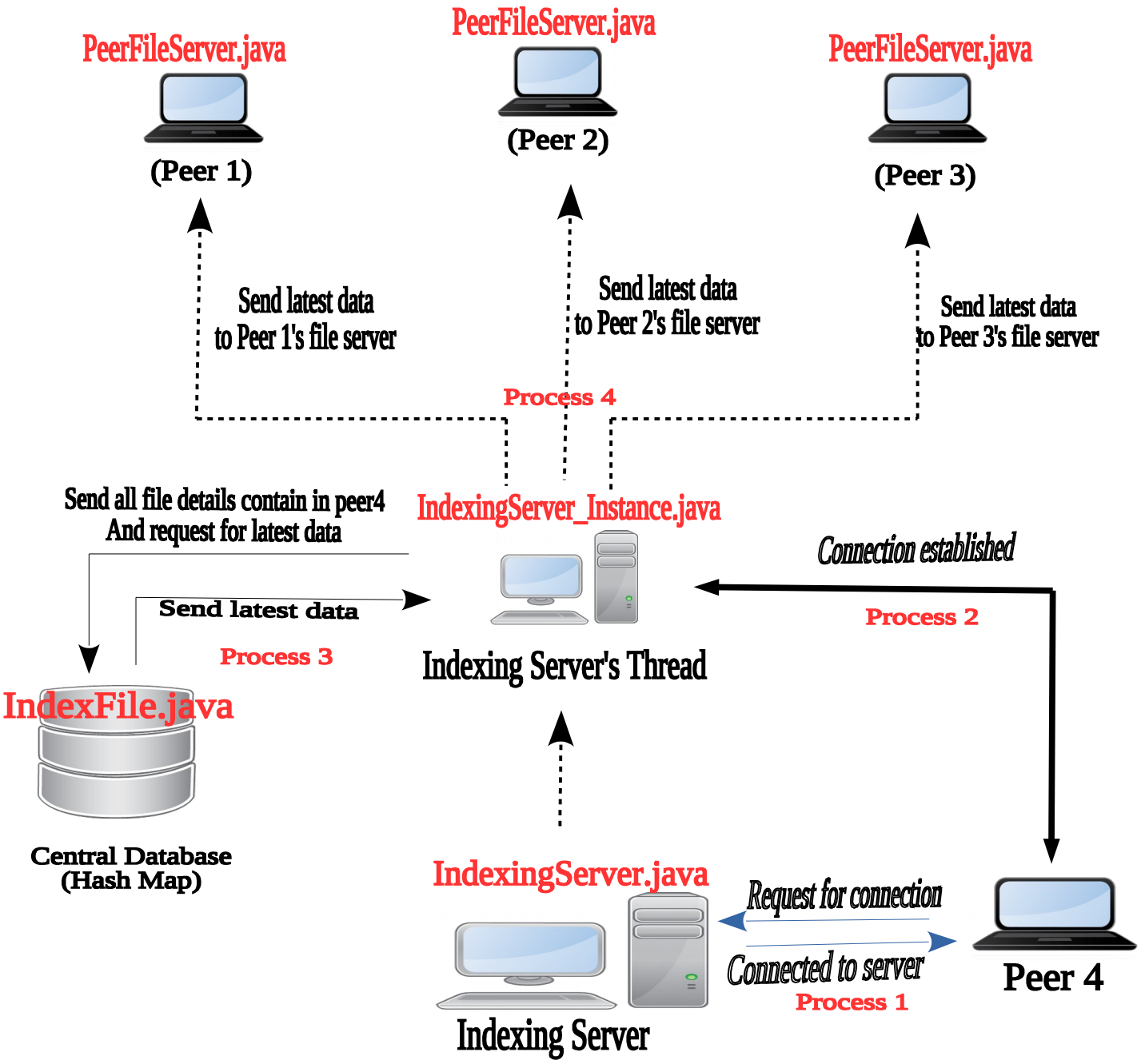
Step 4 : Details communication between Indexing Server and its Thread.

IndexFile.java has declared **HASHMAP,** which acts as a Central Database for the system.
For concurrency control used **ConcurrentHashMap** which handles multiple instances at a time.



Step 5: More about PeerFileServer.

PeerFileServer is introduced to automatically update all peers when new Peer connects to Indexing Server. Because of this, all peers gets auto-update of filenames.



Process 1: Indexing server accepts connection request from Peer4.
Process 2: Indexing server creates thread named **IndexServer_Instance.java** and established connection with Peer 4.
Process 3 : Peer 4 send all his file details to central database. And also request all latest data from DB.
Process 4: Latest data send to all peers which are connected to Indexing server. Each peer has peerFileServer which takes this data and update corresponding peer's database.