# Send Data

Session: ss

DataSession: dss

DataSessionClient: dssClient

1. ***PresencePushTask*** send a StatusUpdate signal to datatransferClient whenever the node received a roster update from sever.
2. If the remote node which currently node would send data to is online, datatransferClient will create a new dss object and send data in order to the remote node from this dss.
3. In dss creation, a new session will also be created. Dss and the new created ss will reference each other.
4. datatransferClient will receive dss creation singal after new dss is created , and connect dss state signal to its handler.
5. datatransferClient call send method of dss for sending out data.
   1. Dss create a new local http server object and a session description object.
   2. Dss call ***initiate*** method of its session object(Initiate(jid.Str(), NULL, desc):
      1. In initiate, ss first create at least one ***Transport*** object (but only one in current libjingle implementation).
      2. ss then generate xml message to send out. In order to generate xml, ss callback dssClient to translate the sessiondescription into xml (which is application specified).
      3. The xml message is send by a OutgoingMessage signal to SessionManager (which connect SigalOutgoingMessage to its method OnOutgoingMessage() in session create).
   3. On remote peer:
      1. SessionManager will create a new Session object when it got a request. A STATE\_RECEIVEDINITIATE message will dispatch to dss.
      2. Dss cached transfer data info, and notify datatransferClient asynchronously. datatransferClient accept the connection request by calling Accept method of dss.
      3. Dss creates an HTTP GET query for the item described with the HttpClient object. Finally sends the HTTP request, as well as a QN\_SHARE\_CHANNEL xmpp stanza.
   4. On initiator peer:
      1. When initiator received QN\_SHARE\_CHANNEL message, it will create a PseudoTcpChannel which connect to the remote computer. It calls HttpServer::Respond with the Transaction to start sending data back to server.
   5. On remote peer:
      1. When all the data is transferred, HttpClientComplete signal will dispatched to dss.