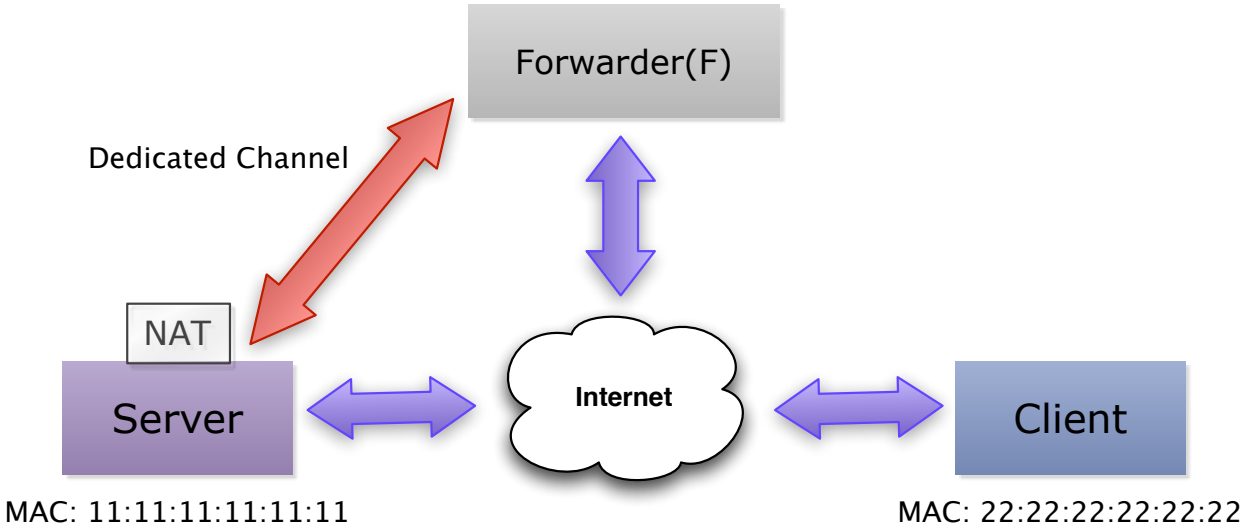
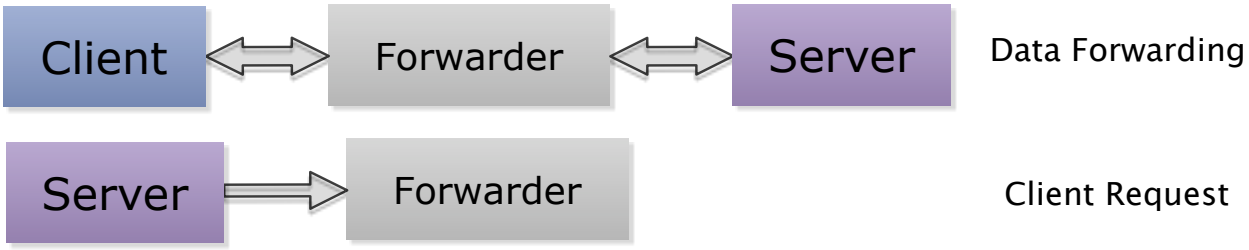
Network Diagram



Data Flows and Convention



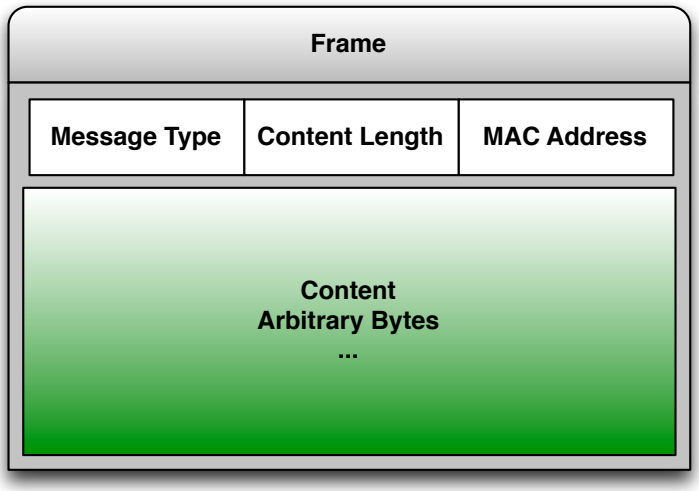
Desired Specifications

- Support TCP
- Minimize additional overhead to TCP
- Be Computationally minimalistic
- Simple to implement

Core Architecture

- Standardize on a single structure, “frames”, which are highly flexible
- Standard Header for simplified implementation
- Although “MAC” addresses are used, the field is opaque, meaning it has no physical or intrinsic value. It must be alphanumeric, 12 digits, and is checked for uniqueness.

Header Field	Description
Message Type	Changes the Server and Forwarder processing of the frame
Content Length	The byte size of the Content in the frame.
MAC Address	The destination MAC address
Notes:	Written as a string, and separated by a colon.



- The first frame is refered to as the “Initialization Frame”
  - When a TCP connection is opened, the client or server is expected to send an Initialization Frame. If the Server is connecting, an INIT_SERVER frame should be sent. If a client connects, an INIT_CLIENT frame should be sent.
  - The Forwarder is expected to immediately respond with a frame to indicate the status, or any errors that have occurred. This is of type INIT_STATUS and contains a string message.

Forwarder Message	Meaning
NO_SERVER	The desired destination server could not be found, or is not connected to the forwarder.
BSY_SERVER	The desired destination server could not be connected to because the connection limit has been exceeded.
CONFIRMED	The connection was successfully established.

- After the Client has initialized the connection, the client should treat the connection as a normal TCP connection. The forwarder will be responsible for packaging the data into Frames and routing the data to the appropriate server.
- The Forwarder needs to always send data to the Server in the form of Frames, likewise the Forwarder can always expect to recieve frames from the Server.
- As stated, the Server's replies must be packaged into frames as well.

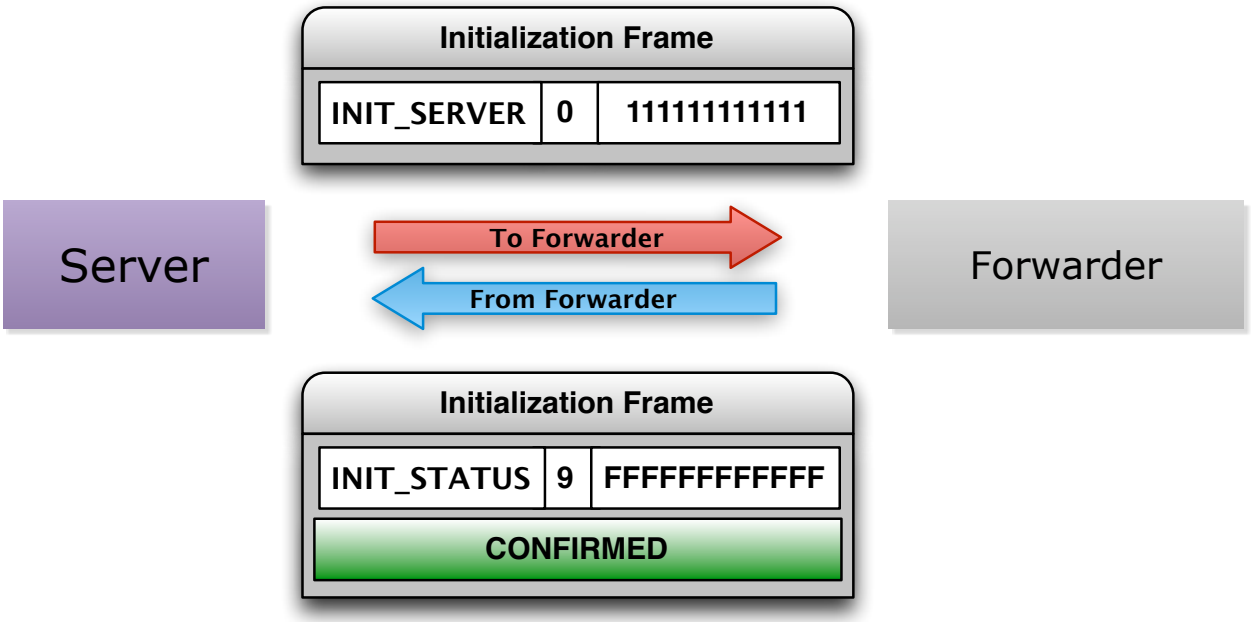
Message Type Header Field

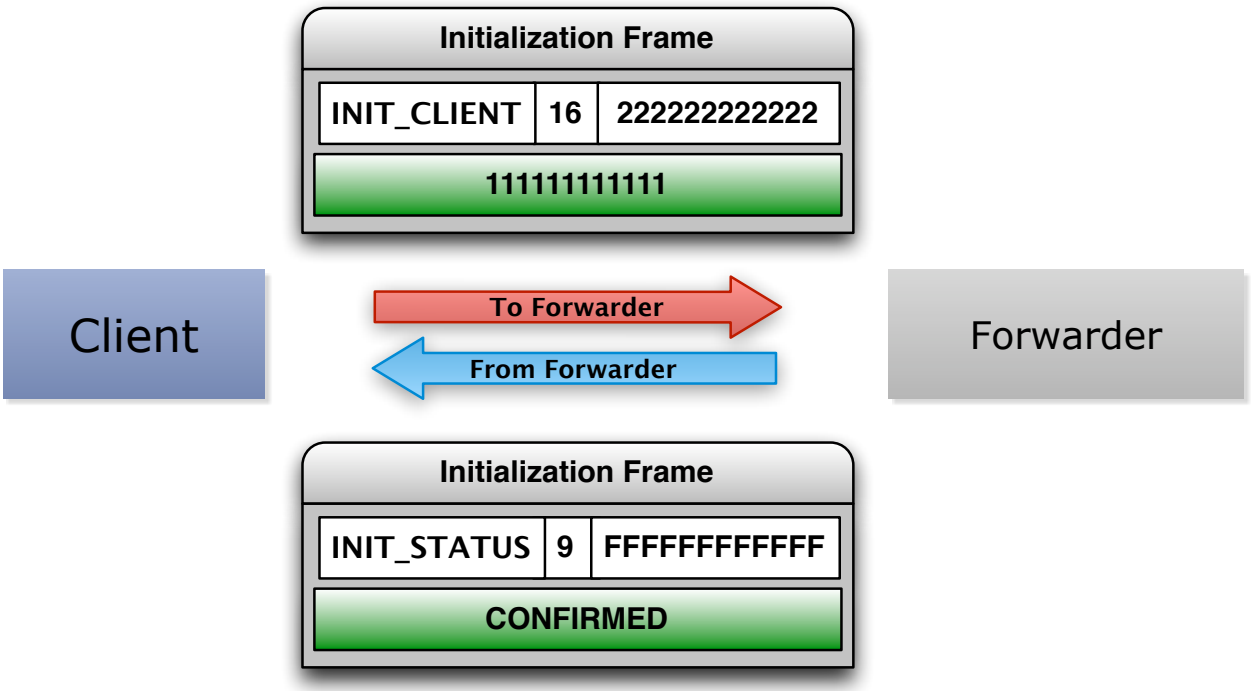
Value	Message	Meaning
0	DATA_FORWARD	Implies that the Frame is a standard data frame that should be processed or forwarded.
1	CONN_TERM	Implies that Conent Length is 0, and that the forwarder should terminate the connection to the specified MAC address
2	CONN_BUF_SIZE	Implies that the content is the Buffer Size, the MAC is the client that should be affected, and the content length should be set appropriately.
3	INIT_SERVER	Implies that the Machine connecting to the Forwarder will act like a server, and that the socket should be multiplexed. The Content Length is set properly, and the MAC is that of the Server. The content should tell the forwarder the servers default buffer size. -1 is used to signify that buffering limits should not be in effect.
4	INIT_CLIENT	Implies that the Machine connecting to the Forwarder will act like a client, and that the socket should be treated like a normal TCP socket. The Content Length is 16, the MAC is that of the Client, and the content should contain the servers MAC address.
5	INIT_STATUS	Contains the status of the last Initialization request. The content length is properly set to contain the forwarders message. MAC address is not applicable, but should be set to forwarder default of FF:FF:FF:FF:FF:FF.

Appendix A: Examples

A.1. Establishing TCP Connection

- The Server is assumed to connect prior to the client, otherwise a NO_SERVER response would be sent from the Forwarder.



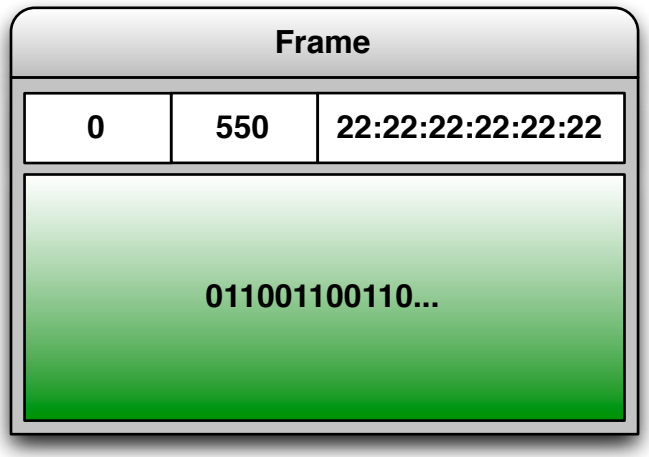


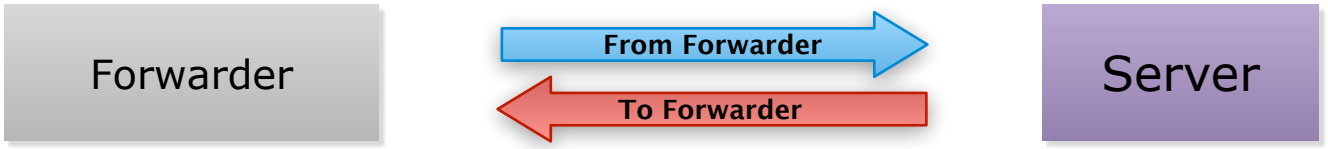
A.2. Transfer Data Client to Server

- Assumes that both Client and Server have connected to the Forwarder.

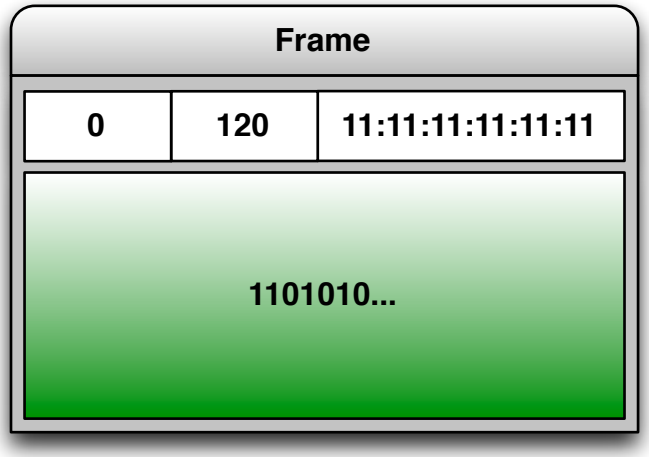
Step 1: Client sends data to Forwarder

Step 2: Forwarder packages data into a frame, and transmits it to the Server.





Step 3: Server responds. Data is packaged into a frame and sent back to the Forwarder.



Step 4: Forwarder extracts the response from the frame, and sends it to the Client.