< Shafe / Leonard
< Meeting / Annunciation
< Cheryl / Experienced
< Sound / RC
< Group Project
< Most Present (No Show = Me)

Class: 2
Date: 11/19

Fall 2012
CSE 531: Distributed and Multiprocessor Operating Systems
\text{Purification} \quad \rightarrow \quad \text{part} \quad \rightarrow \quad 2

\rightarrow \text{local vs distributed}

\rightarrow \text{TCP - UP} \quad \rightarrow \text{Local VS distributed}
sent/reco -> use alphabet, from/to any

( )

port <- IP addr, local port #

\[
\text{first case}
\]

S

\[
\text{second case}
\]

\[
\text{second message}
\]

port -> numbers -> beginning message
San r to a move figure media?

? (sec. [p#] and)

3.

Check f. ir - bca - 3

3. In (p, msg)

< # p, p'
returns

Send msg to server

block

recv net send (msg, msg)

recv net recv (msg)

recv parameter

recv

make
A process sends a message to another process. The message is received by the process and acknowledged.

\[ \text{recv}(p_x) \rightarrow \text{send}(p_y) \]

\[ \text{recv}(p_y) \rightarrow \text{send}(p_x) \]

\[ \text{recv}(p_z) \rightarrow \text{send}(p_w) \]
return (test)
if (msg != empty)
    copy msg to test

newsock (port, IP, msg)
rm sock

(r(m, msg))