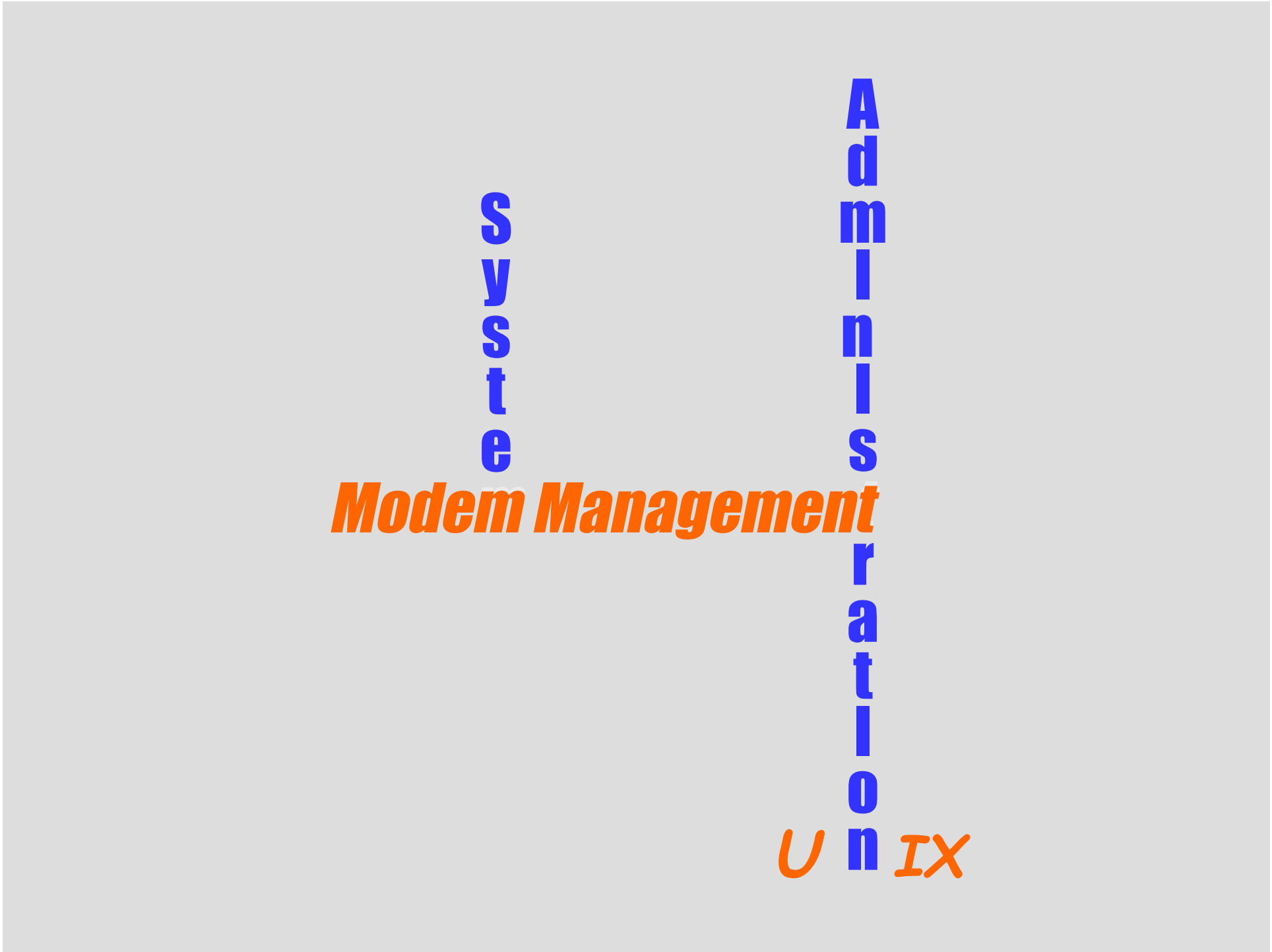

System Administration
Modem Management
UNIX

Modems

- **fundamentally similar to terminals**
  - recall ttyS*n* (cua*nn*)
    - *devices for direct connected serial terminals/modem lines*
    - *cuann devices practically obsolete*
  - we'll look at simple dial-out and dial-in configuration
    - *leave PPP for later…*
    - *also assume that physical connection is OK…*

Simple Dial-Out Usage

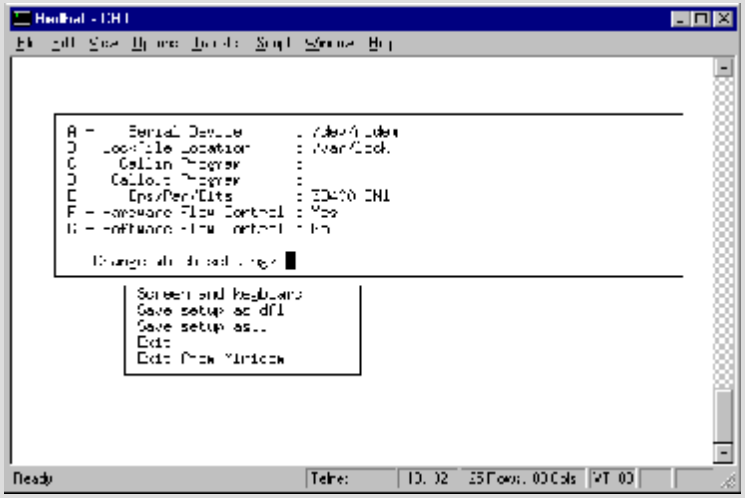
- typically via programs such as cu, tip, kermit or (for Linux) minicom
  - need to ensure that /dev/ttyS*n* device allows an arbitrary user to use the device

```
# chmod o+rw /dev/ttyS0
```

  - for users, dialing out is very similar to using DOS Telix…
    - *if using Linux minicom, root should configure /etc/minirc.dfl before any users attempt to dial-out*

```
# minicom -s
```

    - *should also ensure that authorized dial-out users are listed in /etc/minicom.users*



Dial-In Configuration

- can (and under linux, normally do) setup the same device for dial-in as well as dial-out
  - main difference: the need to have a getty running on the line to handle logins
    - *the special version called uugetty is often recommended for this*
      - almost the same as normal getty BUT it handles uucp-style device locking so that only one process should be able to use the line at any one time
  - *may* also need to set serial device characteristics
    - *typically, the getty application will do this as a result of reading /etc/gettydefs*
    - *if not:*
      - put a call to /bin/setserial in the /etc/rc.d/rc.serial file
        - to handle high speed modems (standard Unix considers >= 2400bps to be "high speed")
      - may also need to turn on hardware handshaking

```
/bin/setserial /dev/ttyS0 spd_vhi
/bin/stty crtscts < /dev/ttyS0
```

More About getty

- getty also looks for the file /etc/conf.getty.*line*, (or /etc/conf.getty if this cannot be found), and initializes *line* according to its contents

```
# uugetty configuration file for a Hayes compatible modem to allow incoming modem connections
#
# this config file sets up uugetty to use the RINGBACK feature for answering calls.

# ringback enable.  The defaults are trusted here (and are pretty sane).
# first, one to three rings, then hangup and call back within 6 and 60 seconds.
RINGBACK=YES

# line to use to do initialization.  All INIT, OFF, and WAITFOR functions
# are handled on this line.  If this line is not specified, any other
# program that wants to share the line (like kermit, uucp, seyon) will
# fail.  This line will also be checked for lockfiles.
#
# format: <line> (without the /dev/)
INITLINE=cua2

# timeout to disconnect if idle...
TIMEOUT=60

# modem initialization string... Sets the modem not to auto-answer
#
# format: <expect> <send> ... (chat sequence)
INIT="" \d+++\dAT\r OK\r\n ATH0\r OK\r\n AT\sM0\sE1\sQ0\sV1\sX4\sS0=0\r OK\r\n

# waitfor string... if this sequence of characters is received over the line, a call is detected.
WAITFOR=RING

# this line is the connect chat sequence.  This chat sequence is performed
# after the WAITFOR string is found.  The \A character automatically sets
# the baudrate to the characters that are found, so if you get the message
# CONNECT 2400, the baud rate is set to 2400 baud.
#
# format: <expect> <send> ... (chat sequence)
CONNECT="" ATA\r CONNECT\s\A

# this line sets the time to delay before sending the login banner
DELAY=1
```