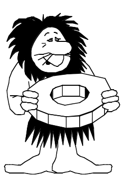
SPIN
An Extensible Operating System for Users and Applications

The SPIN Project
University of Washington

Dept. of Computer Science
and Engineering

Brian Bershad
Craig Chambers
Susan Eggers



SPIN

- Efficient application-specific specialization <u>and</u> extension of system services
- Provide safe access to physical and virtual resources
- Support multiple users simultaneously
- Predictable resource allocation
- Core set of useful services (threads, VM, etc)
- Real OS on top

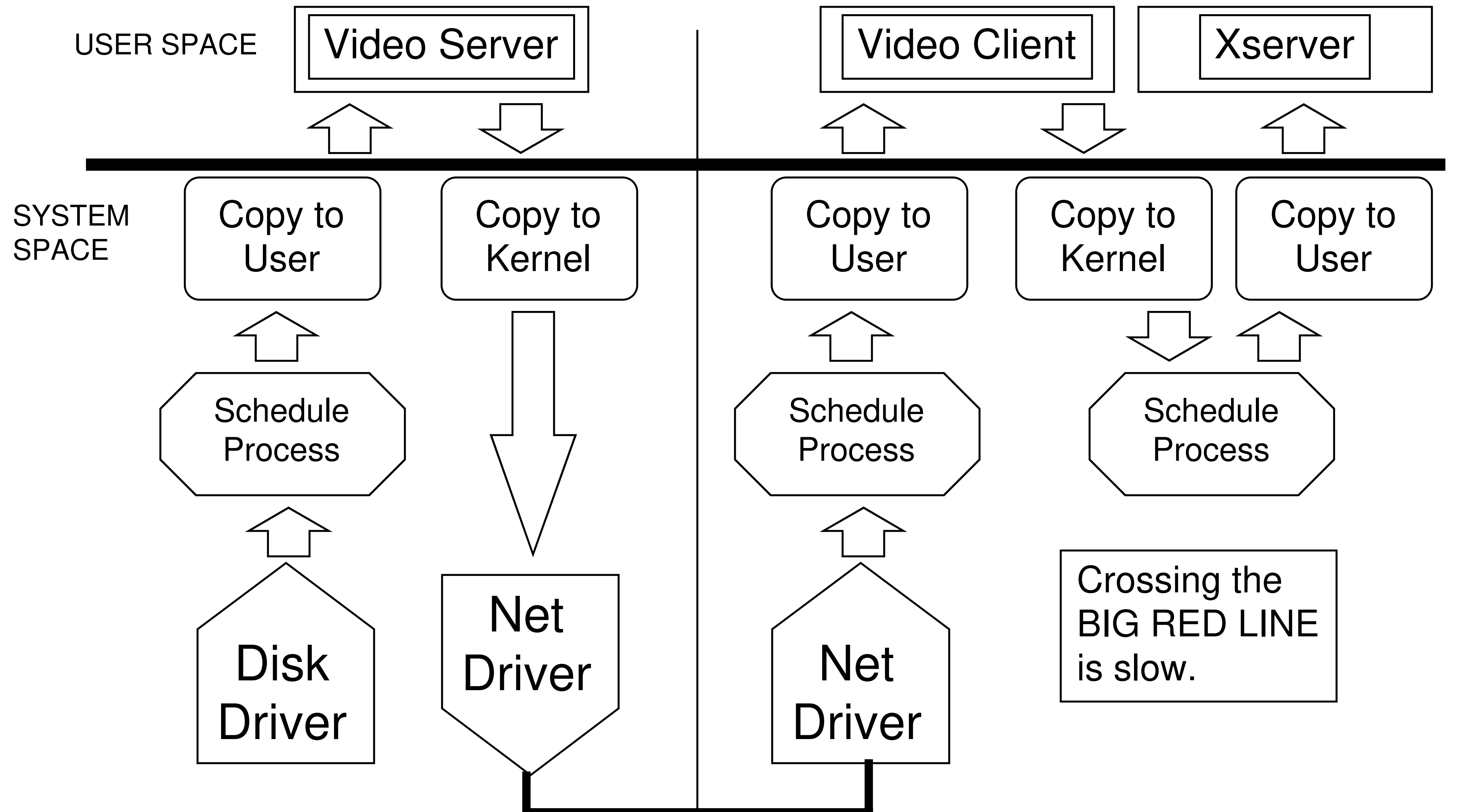
Some Leading Questions

- 100MIPs workstation but only 10 HTTP connections/second?
- 5 TPS in a Dist transaction system?
- Jerky video?
- 5 seconds to context switch on a PowerPC Mac?
- How come DOS/MAC programs do all those tricks with the file system, but your workstation doesn't?
- How come your workstation stays up but DOS doesn't seem to.

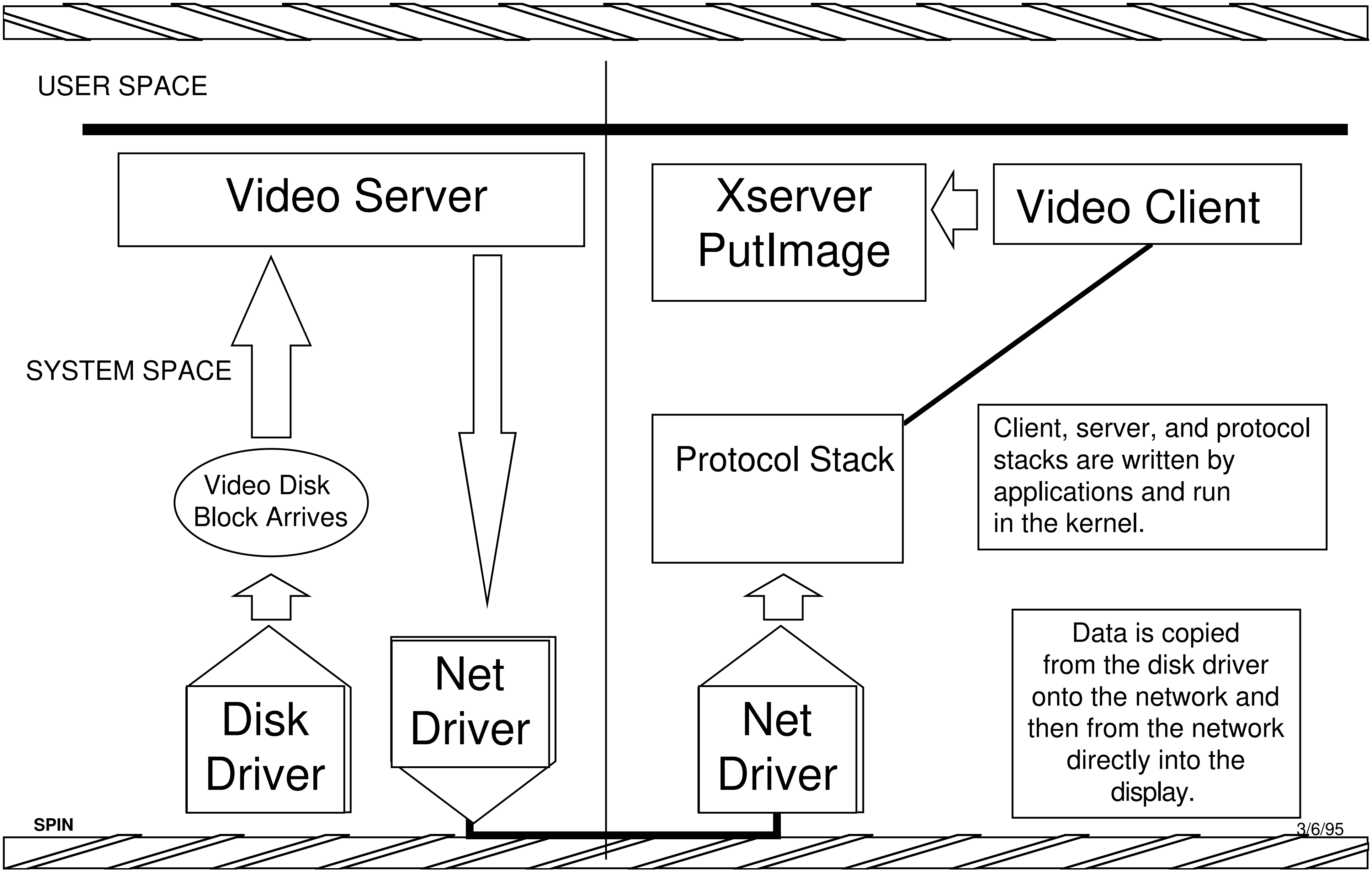
The Problem

- Diverse application requirements.
  - Database and TPS
  - Parallel and scientific
  - Multimedia/Realtime
  - Secure HTTP Gateways and Redirectors
  - GroupWare
  - Emulators
- Limited operating system services
  - files
  - network
  - scheduler
  - V M

How Networked Video Works Now



The Way It Ought to Be



Specializing Operating Systems is Hard Work

- Complexity
  - Most people don't know how.
- Prohibition
  - Most environments don't permit it.
- Barriers
  - Applications aren't allowed (or shouldn't).
- Fear
  - correctness, safety, performance
- Economics
  - if it's so painful, it ought to be general, but...

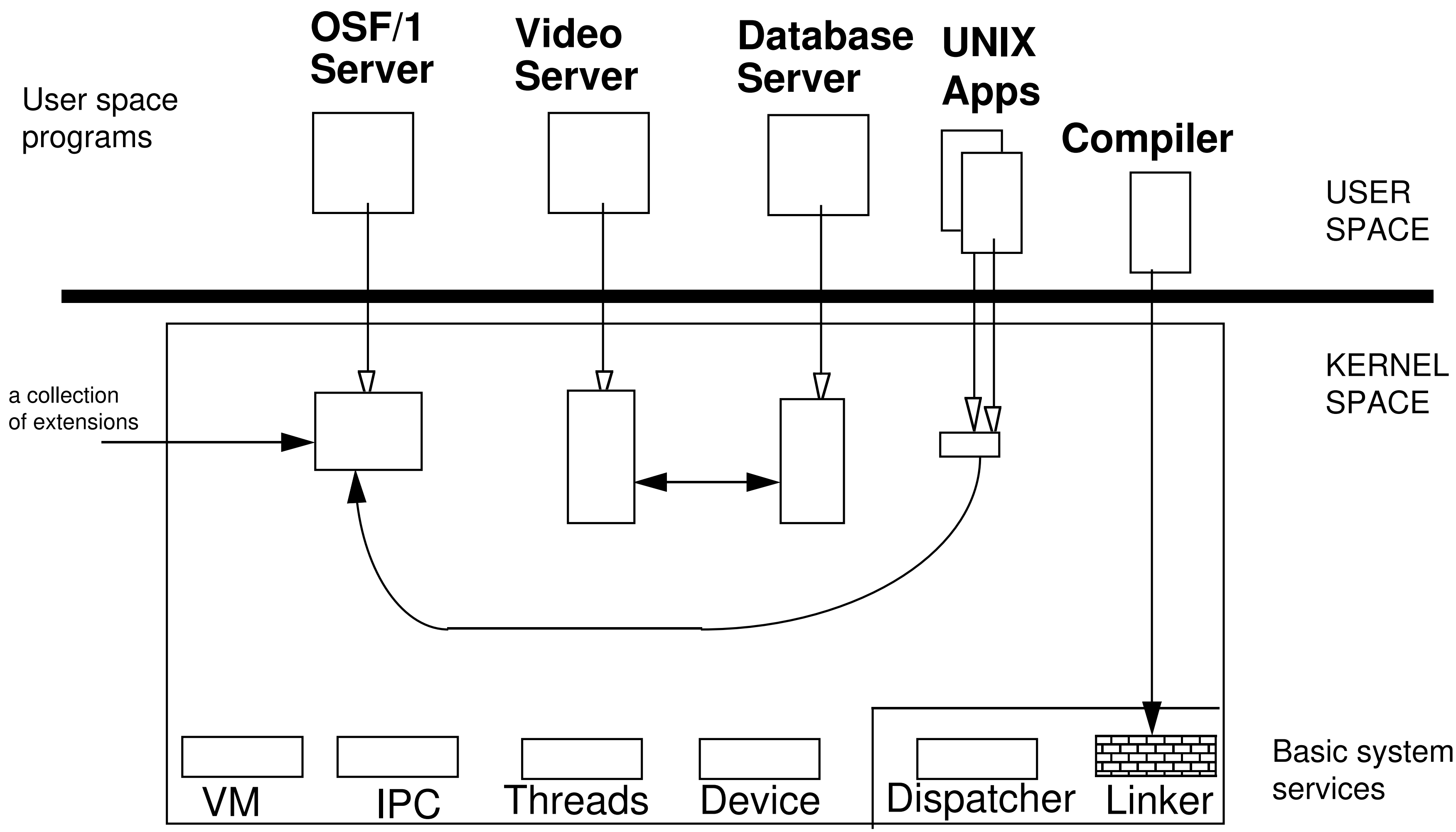
Some examples of application-specific services

- Filing and buffer cache management
  - multimedia, mail, secure anything
- Protocol processing
  - Network splice
  - Active messages
  - Integrated DSM
- Scheduling and thread management
  - Optimistic synchronization
  - Asynch I/O, Multi-level schedulers
- Virtual memory
  - Mach, Database,Cao&Li, Hipec, V++, TLB-prefetch, Network resource reservation, real-time, cache management

Application-specific services with SPIN

- Extensions are linked into a running kernel.
    - inexpensive sharing
- Modular, pointer-safe extension language
    - no bad access
- Logical protection domains within the kernel isolate “semantic” failures.
    - inexpensive context switch
- Exposed kernel interfaces implemented defensively.
    - cannot rely on something you cannot trust.

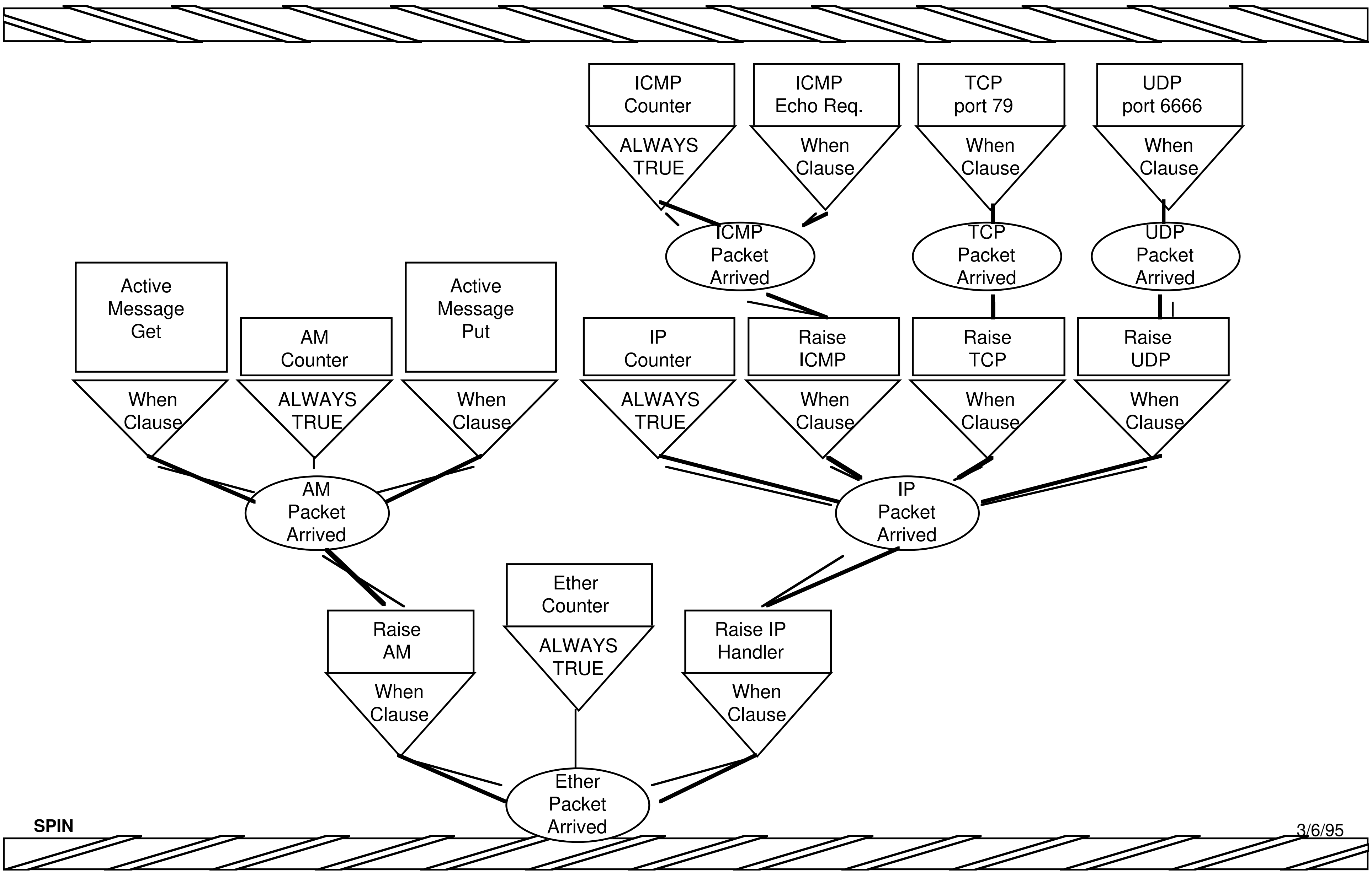
SPIN System Components



An Example: SPIN Networking

- Extensible Network Architecture
  - new protocols, application-specific optimizations
- Packets arrive and are demultiplexed through a protocol graph.
  - arbitrary processing at each node in the graph.
  - but applications can manipulate the protocol graph
- SPIN dispatcher links the nodes in the graph
  - packet filters route incoming packets to handlers
  - handlers do packet processing

SPIN Protocol Stacks



Project Status

- Basic kernel working
  - threads, VM, linker, IPC, devices, networking
- Networking infrastructure supports downloadable protocols
- Video service (client,server) functioning
- OSF/1 server development is underway
- The future
  - “other” OS emulation
  - retarget Modula-3 to use runtime code generation
    - specialization at the level of the compiler