

UDI Network Drivers

Network Interface Metalanguage

<http://www.sco.com/forum1999/conference/developfast/F13>

Barry Feild

SCO Server Products Group

E-mail: barryf@sco.com



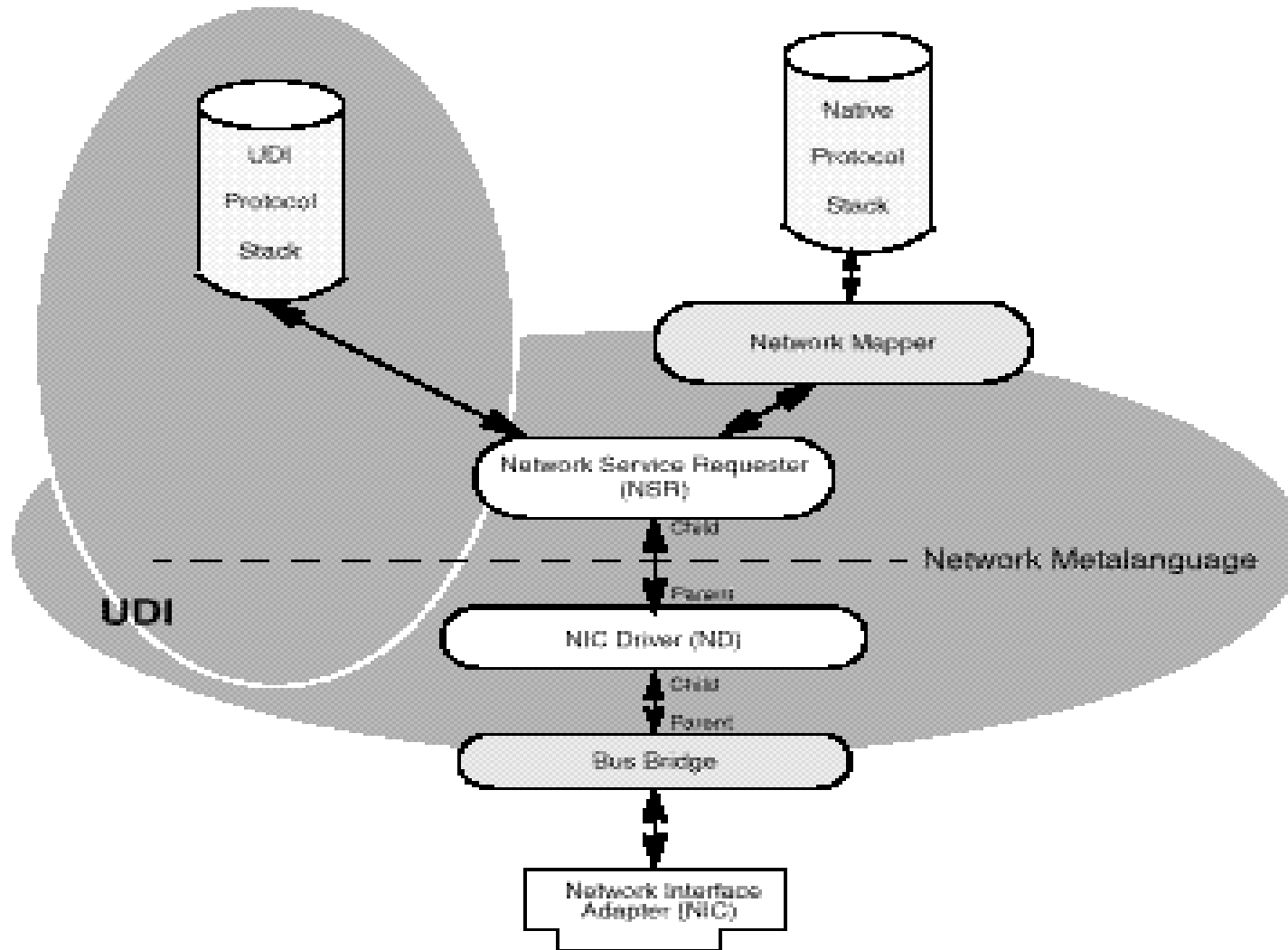
Agenda

- **SCO UDI Network Driver Architecture**
- **Network Interface Metalanguage Features**
- **Network Driver Initialization**
- **Network Driver Operations**
- **Functional Comparison of MDI and UDI**



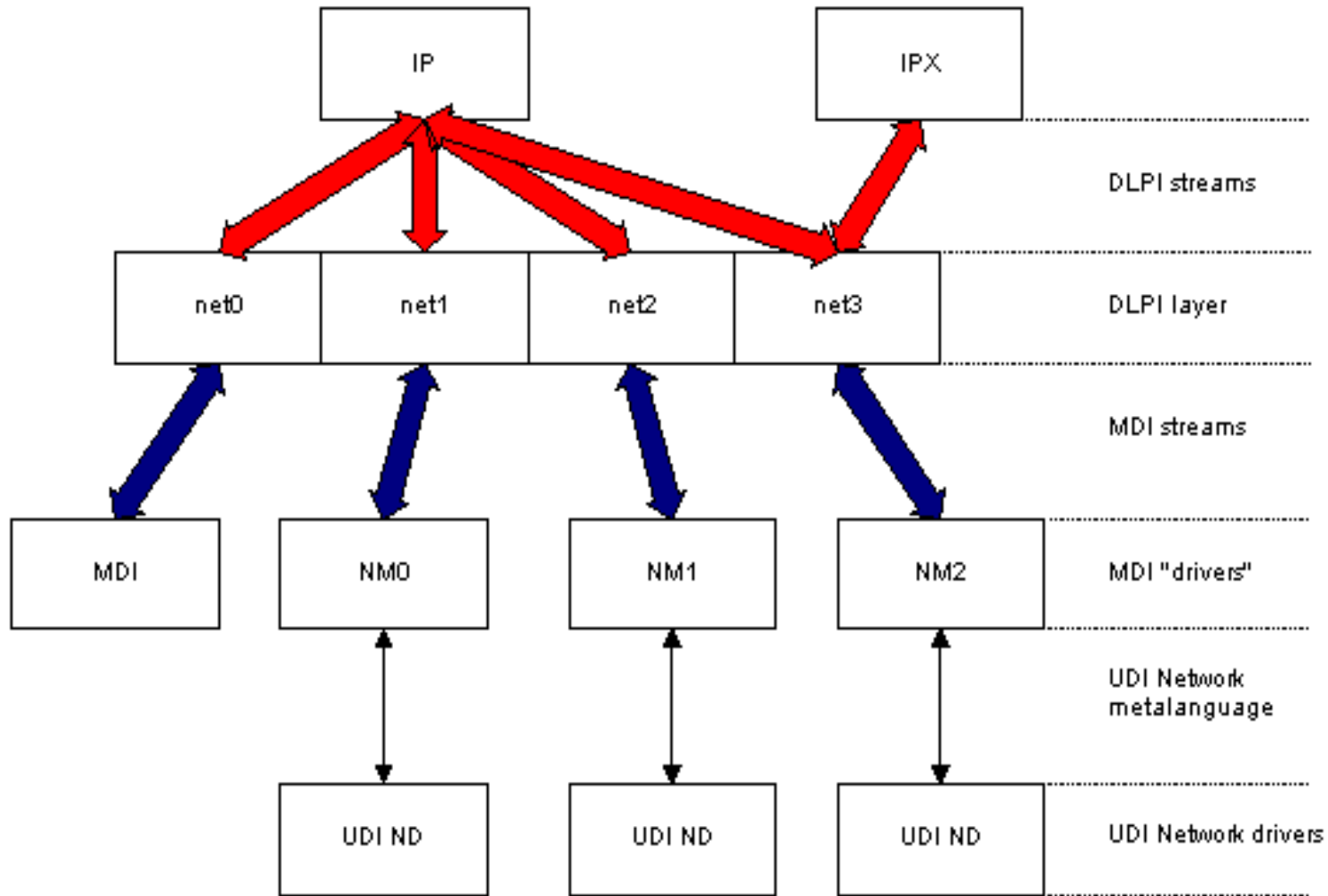
UDI Network Driver Architecture

UDI Networking Environment



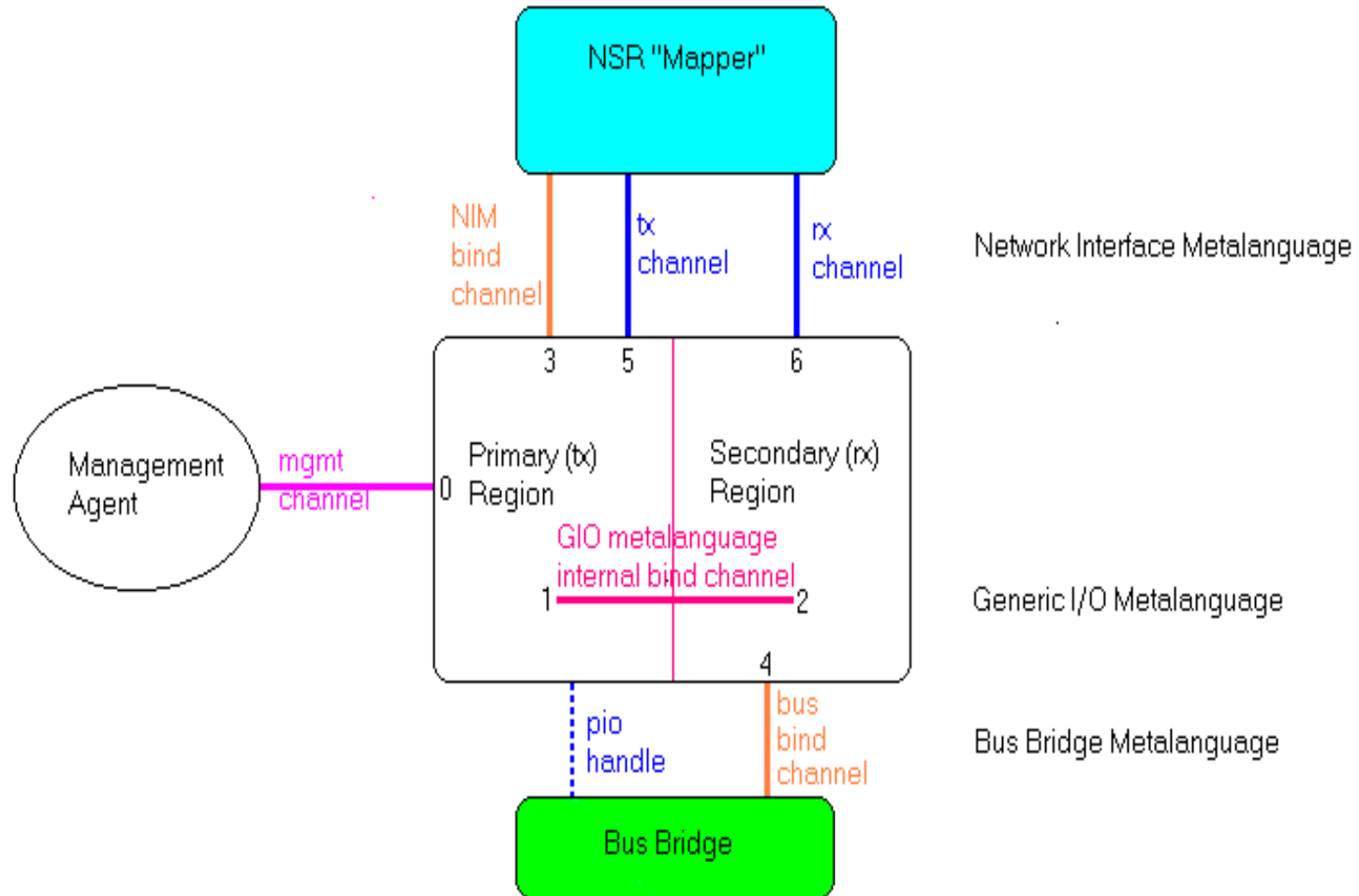
SCO Network Driver Architecture

Initial Release



SCO Network Driver Architecture

shrk UDI Driver



Network Interface Metalanguage

UDI and Native Network Driver Comparison

- **UDI Advantages**

- Common programming model
- Portability
- Binary compatibility
- Implicit MP synchronization
- Simple ND interface

- **UDI Challenges**

- New technology



Network Interface Metalanguage

Network Driver/Network Service Requestor Interface

- **NIM: a universal set of connectionless network-related functions**
- **ND's primary task is send/receive data**
- **Single NSR-ND Interface active at any given time; separate channels are defined for:**
 - control/status
 - transmit data
 - receive data



Network Interface Metalanguage

Network Service Requestor Responsibilities

- **Builds datalink headers for transmit packets**
- **Parse datalink headers on receive packets; NSR may perform SAP de-multiplexing**
- **Supports various MAC address lengths up to 20 bytes; not hard coded at a fixed size (such as 6 bytes)**



Network Interface Metalanguage

NSR-ND Channels

- **Control Channel**
 - Bind/Unbind ND and NSR
 - Network Driver control operations (link state, MAC address registration, multicasting, statistics, etc.)
- **Receive Data Channel**
- **Transmit Data Channel**



Network Interface Metalanguage

Flow Control

- **Transmit Flow Control**

- Tx control blocks owned by ND, loaned to NSR
- ND returns control blocks to NSR on tx complete
- NSR never allocates transmit blocks

- **Receive Flow Control**

- Rx control blocks owned by NSR, filled by ND
- ND never allocates receive buffers
- Rx buffers may be recycled by NSR



Network Interface Metalanguage

Hardware Checksum Offloads

- **Buffer tags used to support CRC offload**
 - `udi_buf_tag_compute()`
 - `udi_buf_tag_apply()`
- **Transmit checksum capability handled separately from Receive checksum capability**
- **Buffer tags may be used for arbitrary per-buffer data**



Network Driver Initialization

Static Driver Properties

- **Static Driver Properties**
 - Region Declarations
 - Parent/Child/Internal bind_ops Declarations
 - Custom Declarations
 - » Adapter-specific configuration parameters
 - » ND reads these attributes on link-enable requests using `udi_instance_attr_get()`



Network Driver Initialization

Network Driver Channel Ops Vector Registration

- **Control channel ops - udi_nd_ctrl_ops_t**
 - channel events
 - bind/unbind/enable/disable
 - control requests (MAC addrs, multicasting, stats)
- **Transmit channel ops - udi_nd_tx_ops_t**
 - normal/expedited transmit requests
- **Receive channel ops - udi_nd_rx_ops_t**
 - NSR gives receive control blocks/buffers to ND.



Network Driver Initialization

UDI Control Block Registration

- **Control blocks (and Ops vectors) registered in udi_init_info structures:**
 - udi_cb_init_t
 - udi_ops_init_t
- **All channel operations are done with appropriate ctrl or transfer control block**
- **All types of control blocks may be allocated internally with udi_cb_alloc()**



Network Driver Initialization

Network Driver Instantiation

- **Parent driver (Bus Bridge) instantiated, mgmt channel established; MA sends enumeration request to parent**
- **MA creates ND instance (primary and secondary regions, mgmt channel, internal bind channels) that corresponds to parent enumeration response info**
- **MA issues udi_usage_ind() on mgmt channel - first operation on newly instantiated driver instance**



Network Driver Initialization

Network Driver Instantiation (continued)

- **MA begins child/parent bind sequence; issues UDI_CHANNEL_BOUND on child (ND) end of ND-Bus bind channel**
 - may be in primary region, static or dynamic secondary region
- **ND performs internal initialization, reads instance attributes, binds to parent with udi_bus_bind_req()**



Network Driver Initialization

Network Driver Instantiation (continued)

- **Parent (Bus Bridge) processes ND's bind request, issues `udi_bus_bind_ack()`**
- **ND completes initialization, issues `udi_channel_event_complete()` for the parent bind operation to the MA**
- **Process repeated for ND (parent) and NSR (child) using NIM bind operations**



Network Driver Initialization

Network Driver Enumeration Attributes

- **Network Driver Enumeration Attributes**
 - if_num: port instance number (32bit unsigned)
 - if_media: media type (macro)
 - identifier: media type (string)
 - address_locator: port instance number (string)
 - physical_locator: interface MAC address (string)



Network Driver Initialization

Network Interface Metalanguage Bind Operation

- **MA creates NSR-ND initial bind channel (control channel) using Network Interface Metalanguage ctrl_ops role**
 - Network Interface Metalanguage control channel is now established
- **NSR issues ND udi_nd_bind_req() on NSR bind channel**
 - ND propagates constraints (memory requirements for DMA, etc.) to NSR with udi_constraints_propagate()



Network Driver Initialization

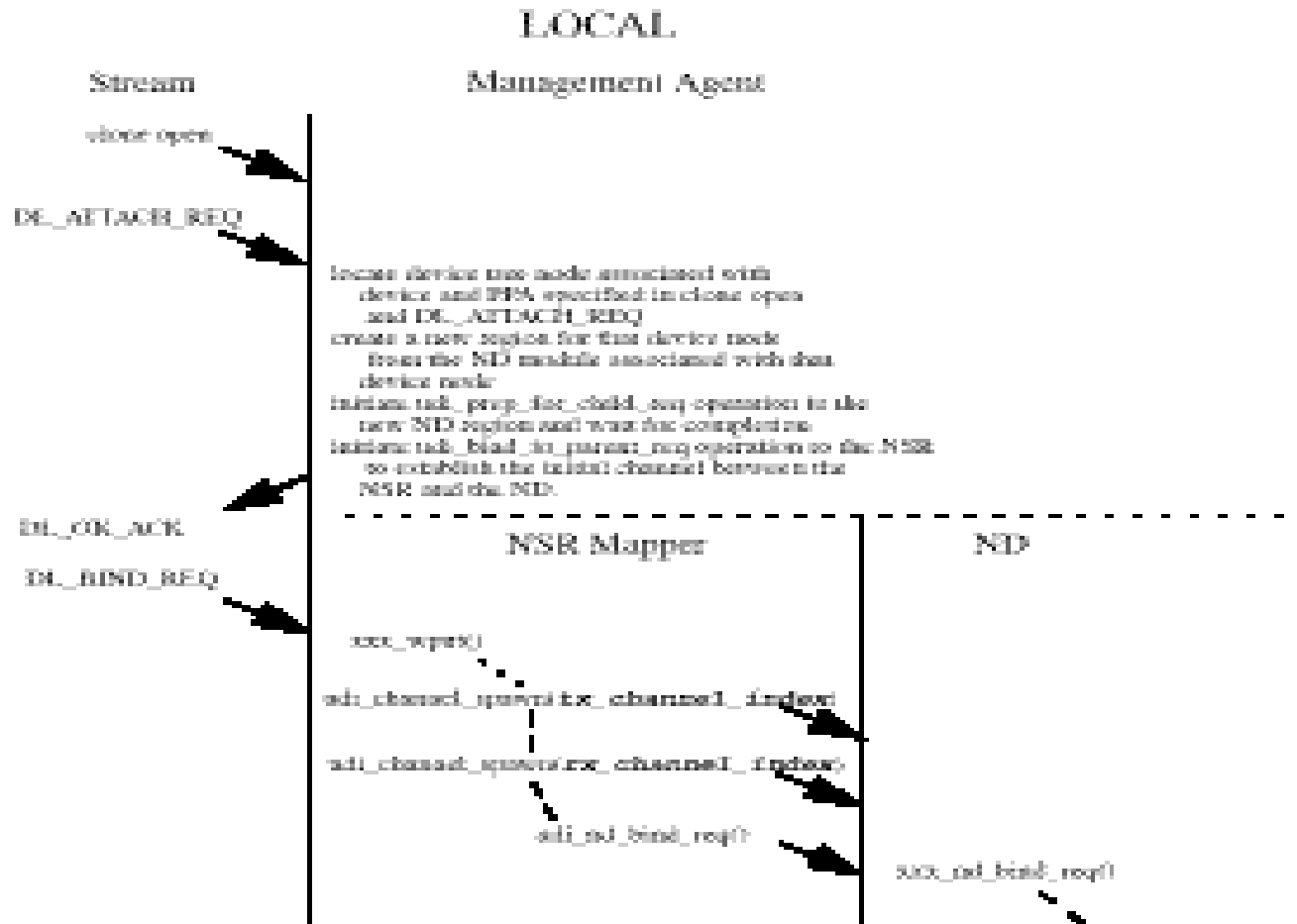
Network Interface Metalanguage Bind Operation (cont)

- **ND actions on udi_nd_bind_req() (cont)**
 - ND spawns and anchors transmit channel, using transmit channel ops vector index
 - ND spawns and anchors receive channel, using receive channel ops vector index
- **ND acks bind with udi_nsr_bind_ack()**



Network Driver Initialization

Connectionless Network Bind Operation



Network Driver Initialization

Network Interface Metalanguage Unbind Operation

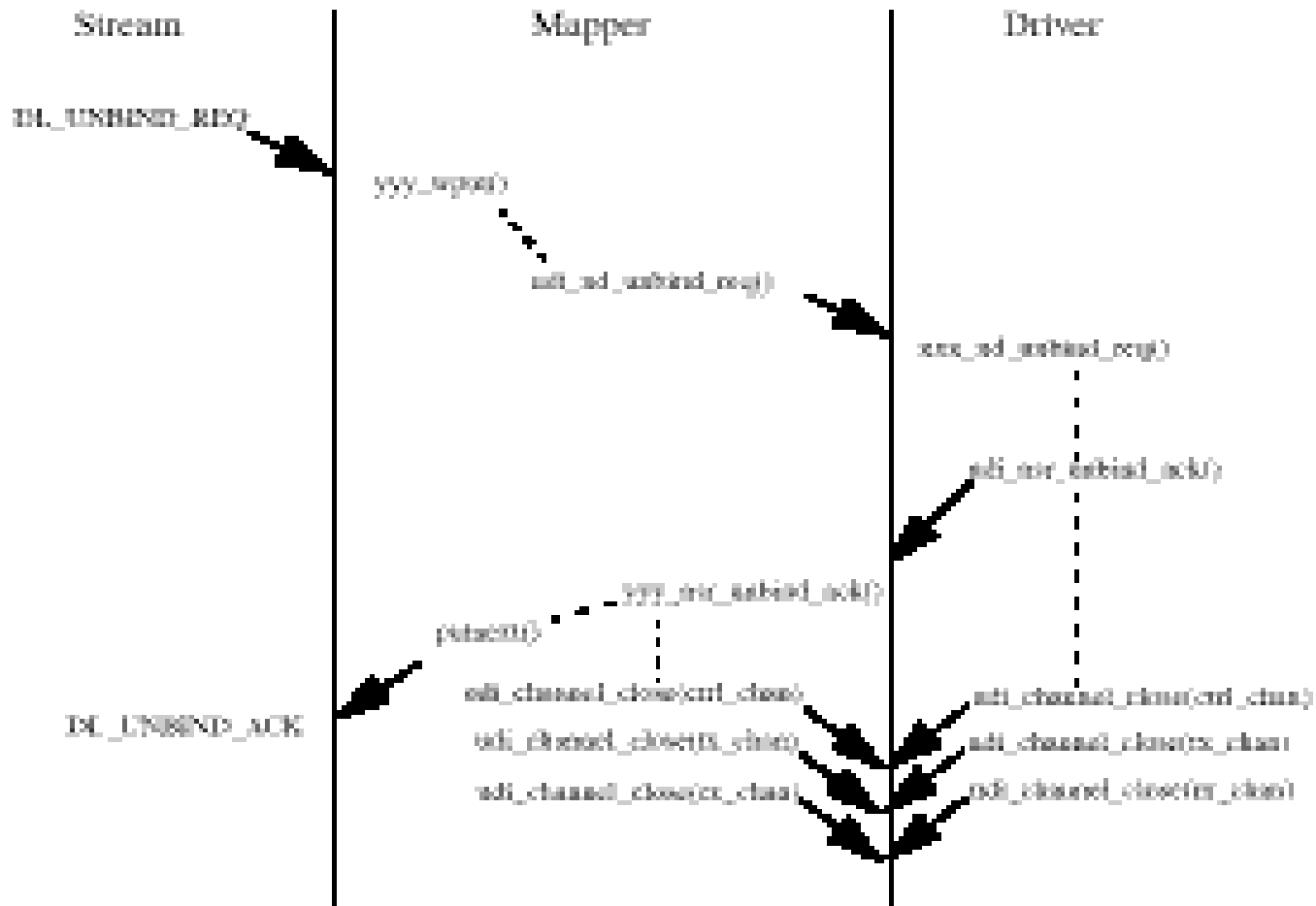
- **Unbind used to detach channels and release resources established during Network Bind operation**
- **ND acks with `udi_nsr_unbind_ack()`**
- **ND closes control and both data channels with `udi_channel_close()`**



Network Driver Initialization

Connectionless Network Unbind Operation

LOCAL



Network Driver Operations

Network Interface Control

- **Enable network interface link activity**
 - udi_nd_enable_req()
 - udi_nsr_enable_ack()
- **Disable network interface link activity**
 - udi_nd_disable_req() - no ack
- **Indicate link state change**
 - udi_nsr_status_ind()



Network Driver Operations

Control and Status Operations

- **Control Requests - udi_nd_ctrl_req()**
 - UDI_NET_ADD_MULTI
 - UDI_NET_DEL_MULTI
 - UDI_NET_ALLMULTI_ON
 - UDI_NET_ALLMULTI_OFF
 - UDI_NET_GET_CURR_MAC
 - UDI_NET_SET_CURR_MAC
 - UDI_NET_GET_FACT_MAC



Network Driver Operations

Control and Status Operations (continued)

- **udi_nd_ctrl_req() (continued)**
 - UDI_NET_PROMISC_ON
 - UDI_NET_PROMISC_OFF
 - UDI_NET_HW_RESET
 - UDI_NET_BAD_RXPKT
- **Control Request ack - udi_nsr_ctrl_ack()**



Network Driver Operations

Control and Status Operations (continued)

- **Status Indications - udi_nsr_status_ind()**
 - UDI_NET_LINK_UP
 - UDI_NET_LINK_DOWN
 - UDI_NET_LINK_RESET
- **Statistics**
 - udi_nd_info_req()
 - udi_nsr_info_ack()
 - udi_net_info_cb_t



Network Driver Operations

Control and Status Operations (continued)

- **Statistics (cont) - udi_net_info_cb_t**

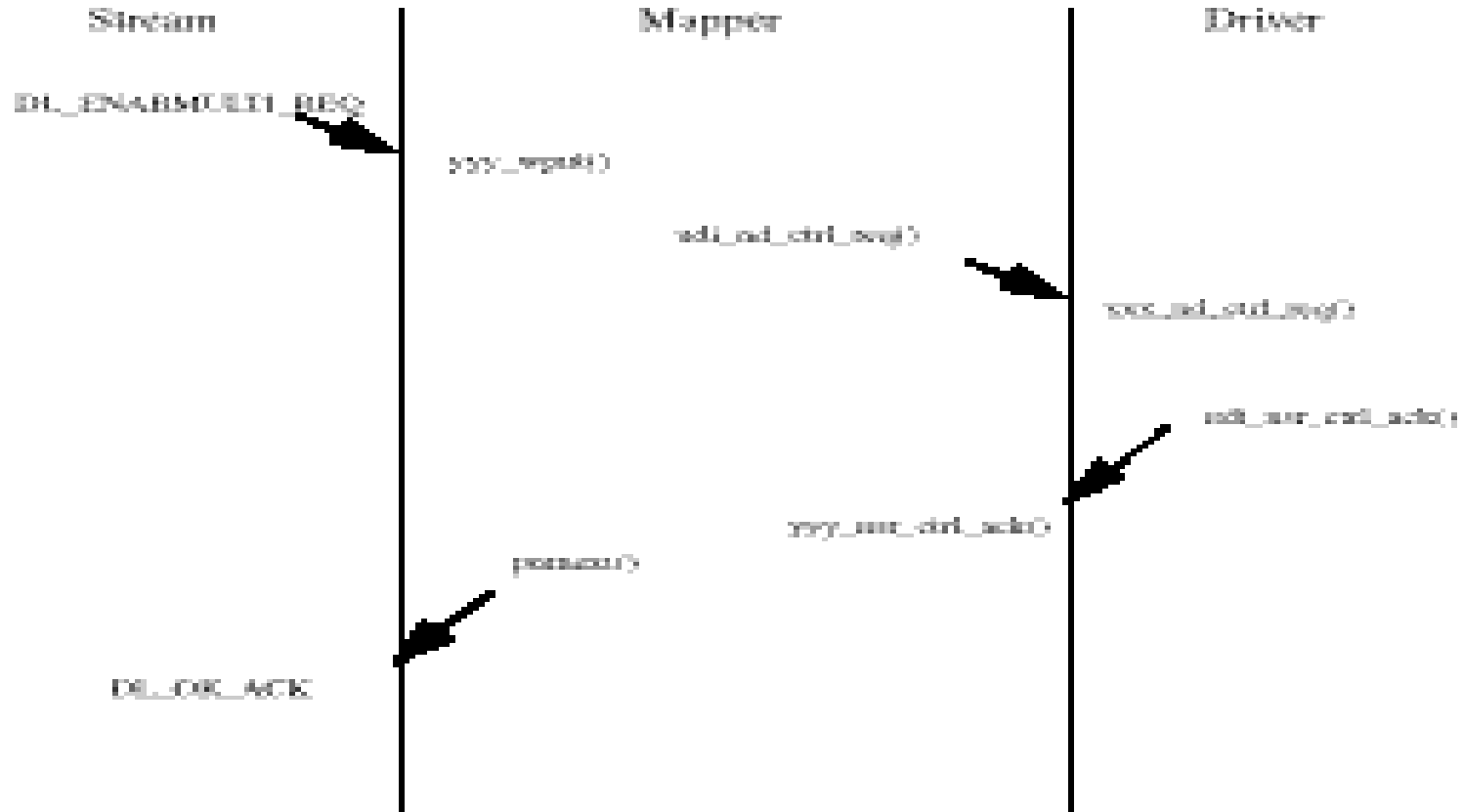
- interface_is_active
- link_is_active
- is_full_duplex
- link_mbps
- link_bps
- tx_packets
- rx_packets
- tx_errors
- rx_errors
- tx_discards
- rx_discards
- tx_underrun
- rx_overrun
- collisions



Network Driver Operations

Control Operation (enable multicast address)

LOCAL



Network Driver Operations

Data Transfer - Transmit

- **ND owns transmit buffers/control blocks**
 - udi_cb_alloc()
- **ND loans transmit control blocks to NSR**
 - udi_nsr_tx_rdy()
- **NSR sends transmit packets to ND**
 - udi_nd_tx_req()
 - udi_nd_exp_tx_req()



Network Driver Operations

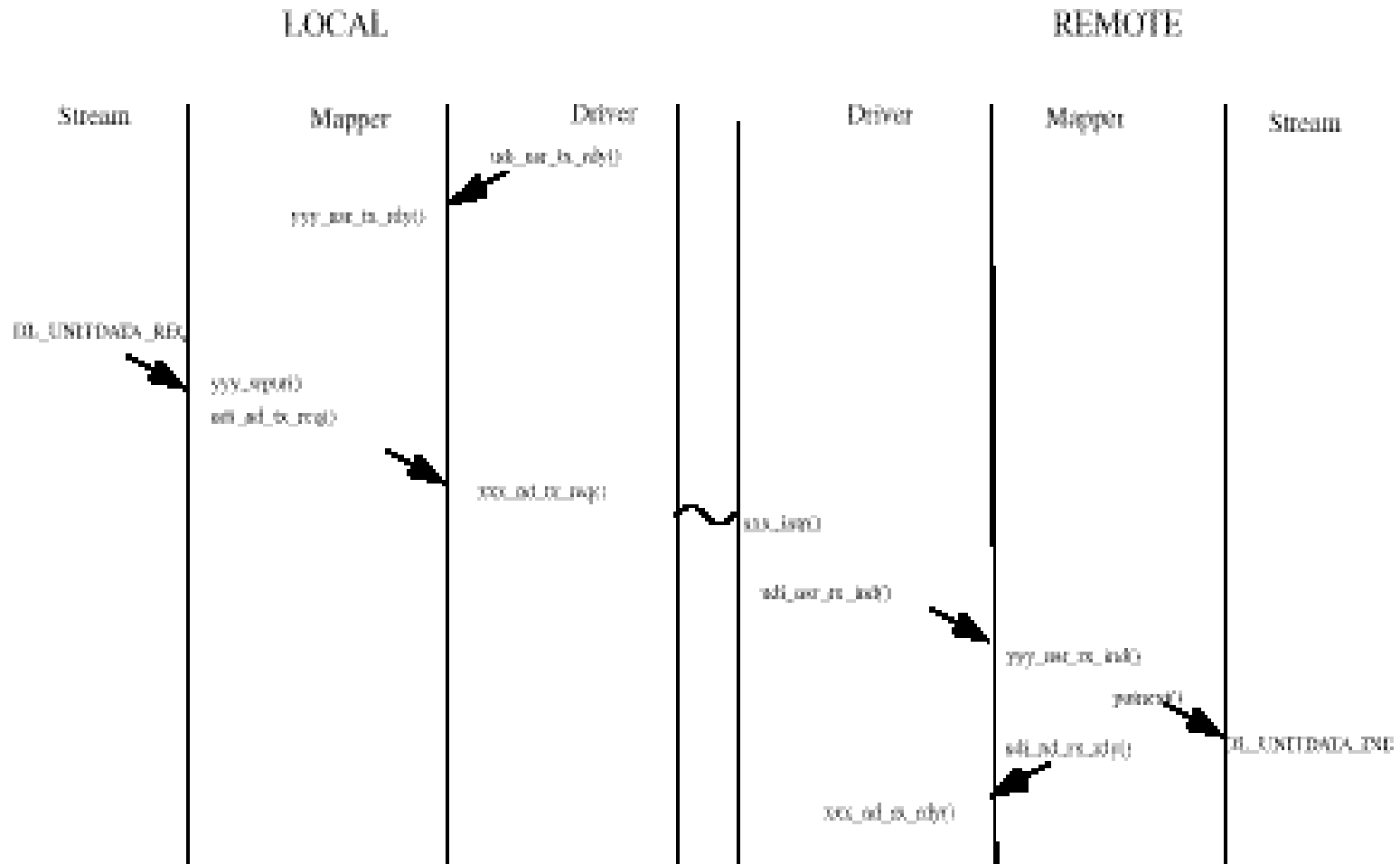
Data Transfer - Receive

- **NSR owns receive control blocks**
 - udi_cb_alloc()
- **NSR loans receive control blocks to ND**
 - udi_nd_rx_rdy()
- **ND sends receive packets to NSR**
 - udi_nsr_rx_ind()
 - udi_nsr_exp_rx_ind()



Network Driver Operations

Connectionless Data Transfer Operations



MDI/UDI Functional Comparison

MDI Primitives

- **MDI Primitives**

- MAC_BIND_REQ
- MAC_OK_ACK
- MAC_INFO_REQ
- MAC_INFO_ACK
- MAC_ERROR_ACK
- MAC_HWFFAIL_IND
- MAC_HWSUSPEND_IND
- MAC_HWRESUME_IND

- **UDI Equivalent**

- udi_nd_bind_req()
- udi_nd_bind_ack()
- udi_nd_info_req()
- udi_nsr_info_rsp()
- udi_nsr_bind_res error
- UDI_NET_HW_RESET
- UDI_DMGMGT_SUSPEND
- UDI_DMGMGT_RESUME



MDI/UDI Functional Comparison

MDI ioctl

- **MDI ioctl**

- MACIOC_SETALLMCA
- MACIOC_DELALLMCA
- MACIOC_SETMCA
- MACIOC_DELMCA
- MACIOC_GETADDR
- MACIOC_GETRADDR
- MACIOC_SETADDR
- MACIOC_GETSTAT
- MACIOC_PROMISC

- **UDI Equivalent ctrl_ops**

- UDI_NET_ALLMULTI_ON
- UDI_NET_ALLMULTI_OFF
- UDI_NET_ADD_MULTI
- UDI_NET_DEL_MULTI
- UDI_NET_GET_CURR_MAC
- UDI_NET_GET_FACT_MAC
- UDI_NET_SET_CURR_MAC
- udi_nd_info_req()
- UDI_NET_PROMISC_ON/OFF



MDI/UDI Functional Comparison

MDI ioctl (continued)

- **MDI ioctls with no UDI equivalent**
 - MACIOC_SETSTAT
 - MACIOC_CLRSTAT

