QFC

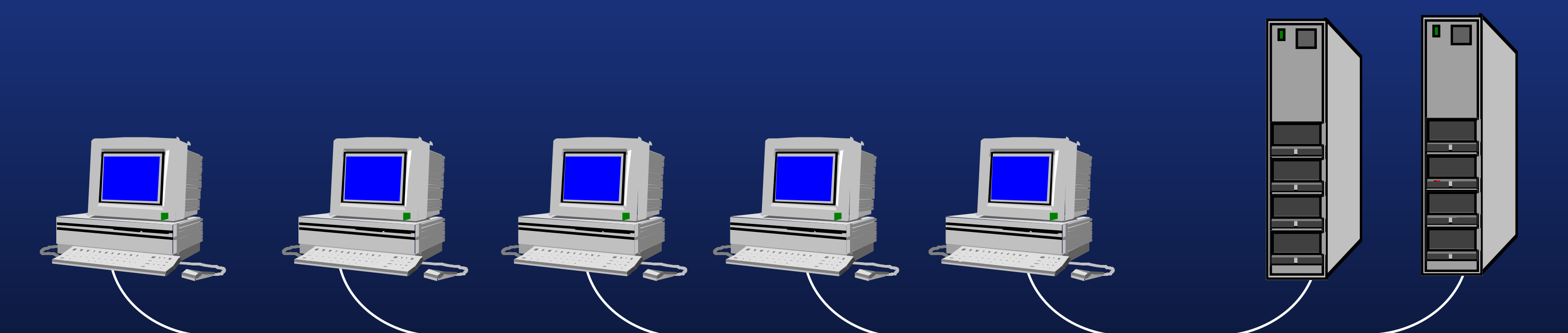
Quantum Flow Control

Enabler for ABR Service

QFC

What is ABR Service?

- Defined for LAN data applications
  - Load executable programs
  - Write and retrieve application data
  - Copy data files



LAN Applications

- Transaction oriented
  - User or applications triggers transaction
  - Applications transfer data
  - Completion is first visible result
    - Not sensitive to delay variance
    - But no tolerance for data loss

ABR Service Definition

- ABR as defined by LANE (94-0533) & IETF (RFC 1754)
  - No spec of min or max bandwidth
    - Applications ready - want infinite
    - Applications not ready - want nothing
  - As simple as possible
  - Instant access to bandwidth by applications
  - Cell Loss Rate ~$3x10^{-8}$ for 1.5K byte PDU and $3x10^{-9}$ for 18K byte PDU
  - Point-to-point & point-to-multipoint
  - Goal is no CAC rejection

ABR Service Definition

- More ABR service requirements
  - Efficient use of available bandwidth
  - Goal of 100% goodput
  - Must not impact CBR, VBR
  - Fair access to bandwidth among peers
    - Equivalent to Ethernet and Token Ring

QFC

Key to ABR Service

- Transaction time is key
  - Sending application just wants data out ASAP
  - Receiving application just wants data in ASAP
  - Switches should forward data ASAP
  - Data = cells
  - Everybody move cells ASAP
    - Without losing any

ABR Service

- Conditions required to move cells
  - Cell slots available for ABR
    - Whatever's left after CBR, VBR
    - Outside control of ABR
  - Buffers
    - Transmit cells buffered and ready to send
    - Receive buffers to accept them
    - Dependent on ABR control mechanism

ABR Service

- ABR control challenge
  - Have a buffered cell ready on transmit end and a free buffer on receive end anytime a free cell slot comes along - all along the VC path

Effective ABR = Effective Buffer Management

ABR Service

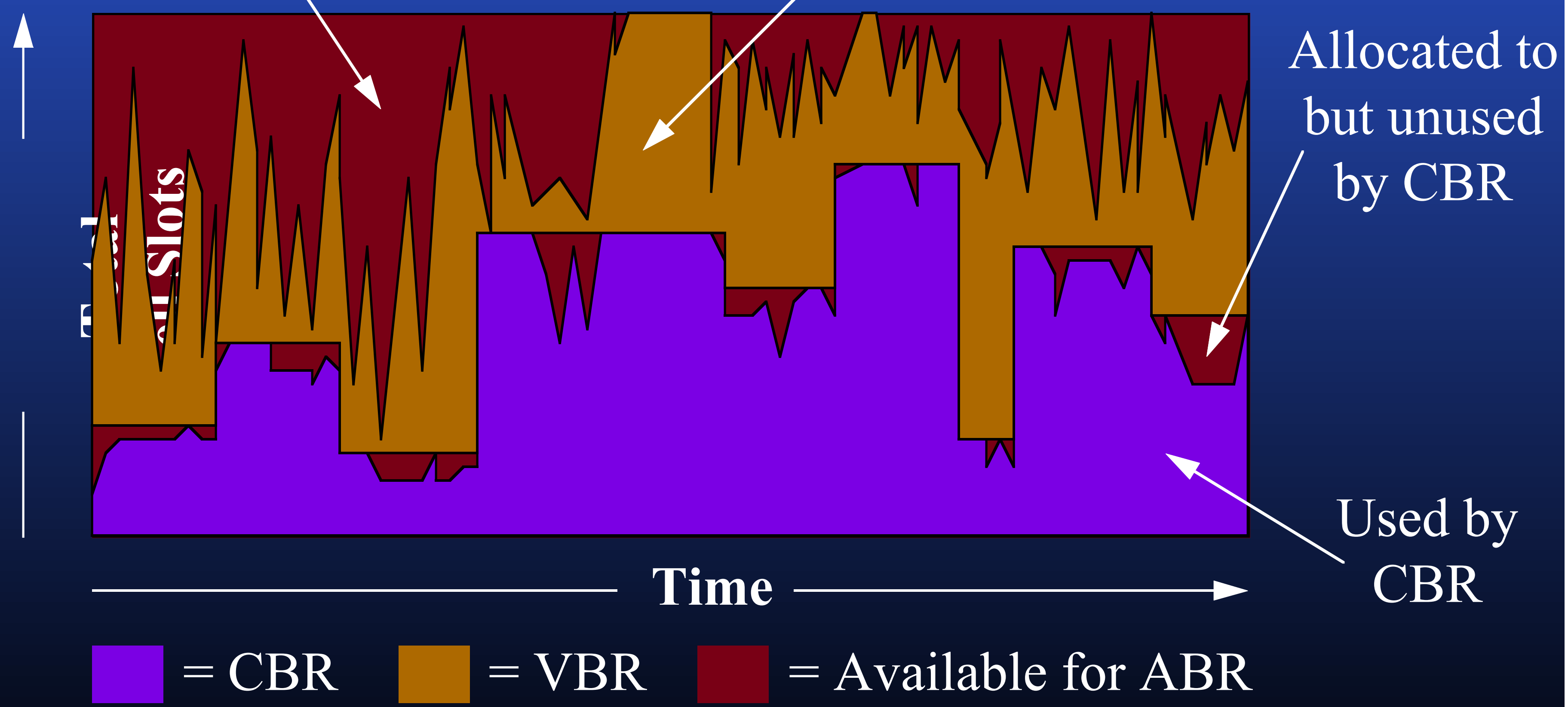
Sounds simple ... but ...

ABR Cell Slots

- Cell slot availability varies wildly
  - ABR left with whatever CBR, VBR don't use
  - CBR connections come and go
  - CBR data rates can vary too
    - Non-constant mission critical data often sent through guaranteed CBR pipe
  - VBR is expected to vary, and it does
  - Difficult to predict when ABR cell slots will appear or how many

QFC

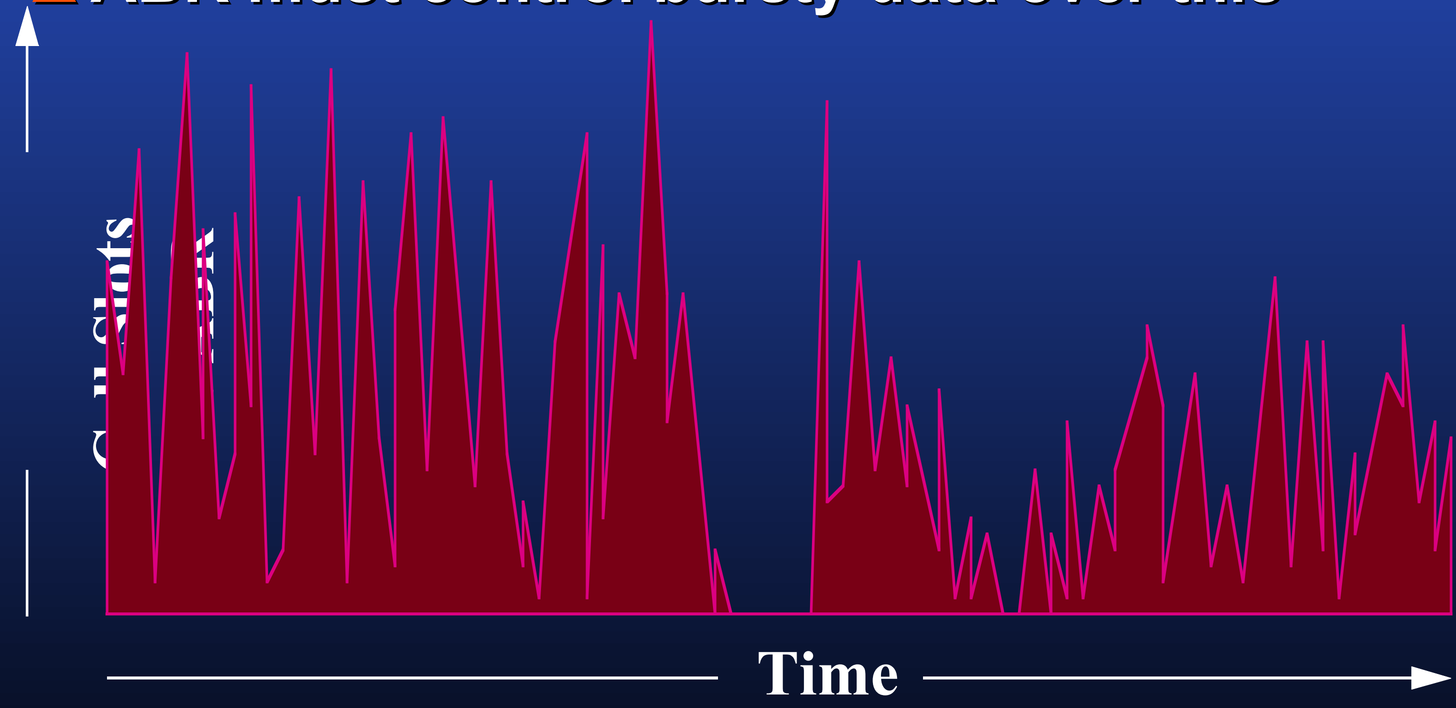
Cells Available For ABR



QFC

Cells available For ABR

- These are the cell slots, i.e. bandwidth, that ABR has to work with
- ABR must control bursty data over this



Buffer Management

- Requirement
  - When a buffer is free, move a cell into it at the earliest available link cell slot
- Required
  - The very best buffer information
  - The most effective feedback

Buffer Management

- Two kinds of buffer information
  - Buffer state
    - Total and free buffer count
    - Receiver always knows exactly
  - Cell Rate
    - How fast buffers are being filled
    - How fast buffers are being emptied
    - Average of cell or buffer counts over time
      - Integration accumulates errors over time
    - Current and accurate only under steady state

QFC

Buffer Management

- Two feedback mechanisms
  - End to end
    - Longest feedback delay
    - Best for controlling source station rate
    - Not effective for controlling across link
  - Hop by hop
    - Fastest feedback possible
    - Best for controlling across link
    - Feedback propagates back to source station

QFC

Buffer Management

- Two localizations of control
  - Per port
    - Easy to protect against buffer pool overflow
    - Inherent sharing of buffer pool
      - Optimal buffer utilization
    - Can't provide fairness
  - Per VC
    - Needed to assure fairness

Quantum Flow Control

- Combines all the best elements
  - Buffer state information
  - Hop by hop feedback
  - Both per port and per VC control

Quantum Flow Control

- **Delivers best result**
  - Instant full bandwidth
  - No congestion related cell loss
  - Minimal control overhead
  - Fairness
  - Scaleability
  - Robustness under error conditions

QFC

Quantum Flow Control

- Complete solution
  - Automatically configures
  - Minimal ATM signaling
    - Supportable with UNI 3.0, 3.1, 4.0 signaling
  - Self policing
  - Point-to-point & point-to-multipoint
  - VC and VP connections
  - Tunnel through CBR VPCs

Quantum Flow Control

- Compatible Solution
  - Works with ATM Forum ABR flow control
    - QFC links are transparent in ATMF ABR path
    - QFC protects ATMF ABR from cell loss
    - QFC enables faster ATMF ABR startup
    - QFC minimizes transient effects
    - QFC extends dynamic range of ATMF ABR

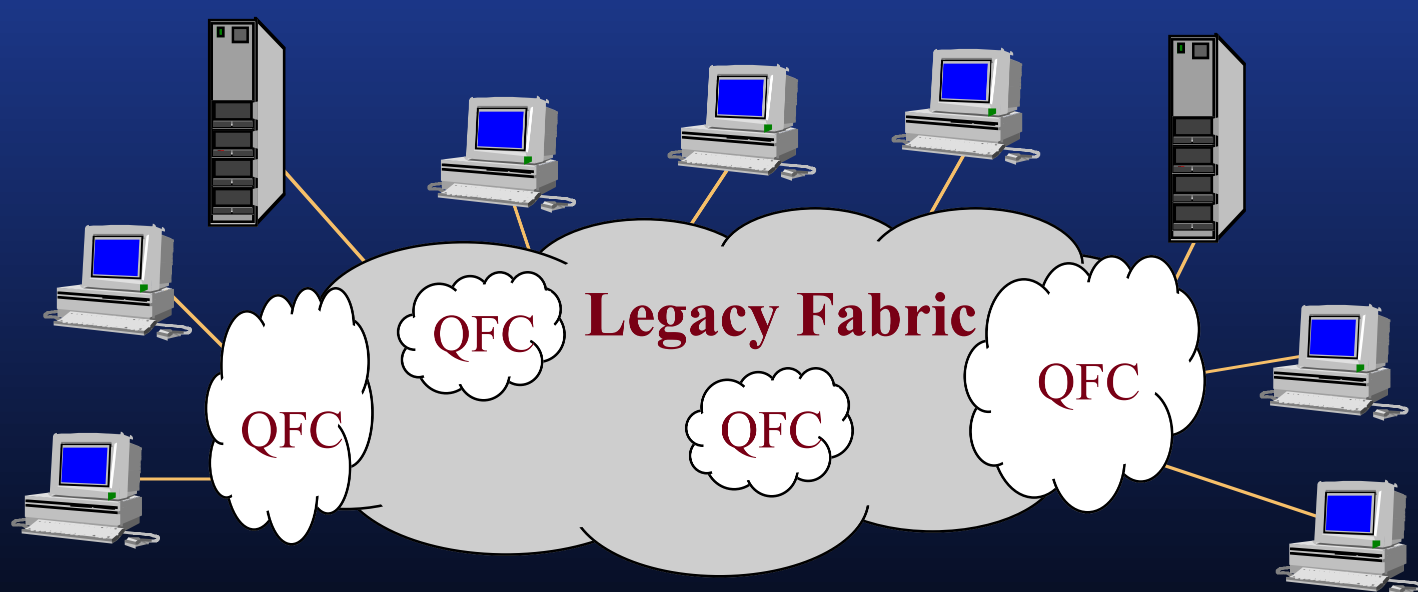
Quantum Flow Control

Deployment

QFC

Incremental Deployment

- No forklifts needed
  - Incrementally add QFC
  - Each new QFC link improves performance
    - End-to-end consistency unnecessary



QFC

Protect UBR Investment

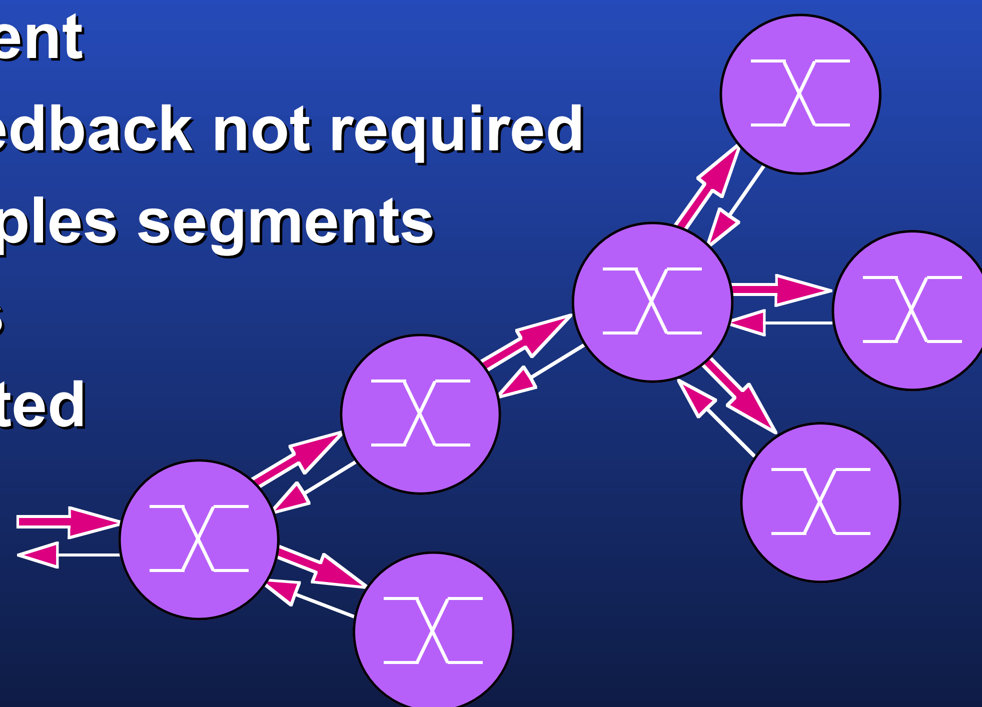
- QFC fabric core
  - Creates high performance, zero loss core
    - End-to-end QFC not required
  - UBR links mostly retain their original load
    - UBR still functions as before
    - UBR investment protected

ATMF ABR Over QFC

- QFC links are transparent to ATMF ABR
  - QFC switches pass ATMF RM cells
    - Can mark per VC under downstream congestion
  - QFC destinations turn around ATMF RM cells
- QFC links are well behaved
  - Handle transient conditions without cell loss
  - Drain as quickly as downstream congestion allows

Point-to-multipoint

- QFC makes multipoint straightforward
  - Each link is independent
  - Multipoint-to-point feedback not required
  - Switch scheduler couples segments
  - Zero loss across links
  - All topologies supported
    - Point-to-point
    - Point-to-Multipoint
    - Multipoint-to-point
    - Multipoint-to-multipoint
  - No impact as leaves added



QFC

Quantum Flow Control

Technical Overview

Full Spec Available

- Complete spec available to public
  - ftp://nexen.com/pub/qfc/
  - ftp://ftp.qfc.org
  - http://www.qfc.org

Link Definition

- For control, links are unidirectional
  - Each direction treated as separate data stream
  - Each direction controlled independently
- Each direction has a transmit end and a receive end
  - Data from transmit end to receive end
  - Feedback from receive end to transmit end

QFC

Link Definition

QFC

Counters & Variables

- Transmit end (for link and per VC)
  - Cells transmitted
  - Cells forwarded by receiver, per feedback
  - Maximum allowable transmitted & unforwarded
- Receive end (for link and per VC)
  - Cells received
  - Cells forwarded
  - Minimum cells before sending feedback
- Buffers Committed = Cells Transmitted minus Cells Forwarded

Counters & Variables

- Link counters protect link receive buffers from overflow
- Per VC counters provide fairness and prevent blocked VCs from impacting unblocked VCs

Operational Concept

- Use every available cell slot while committed buffers less than maximum
  - No transmit pacing
  - Never overrun receive buffers
- Return free buffer information faster than buffers are used
  - No performance loss from control mechanism

Control & Feedback Cells

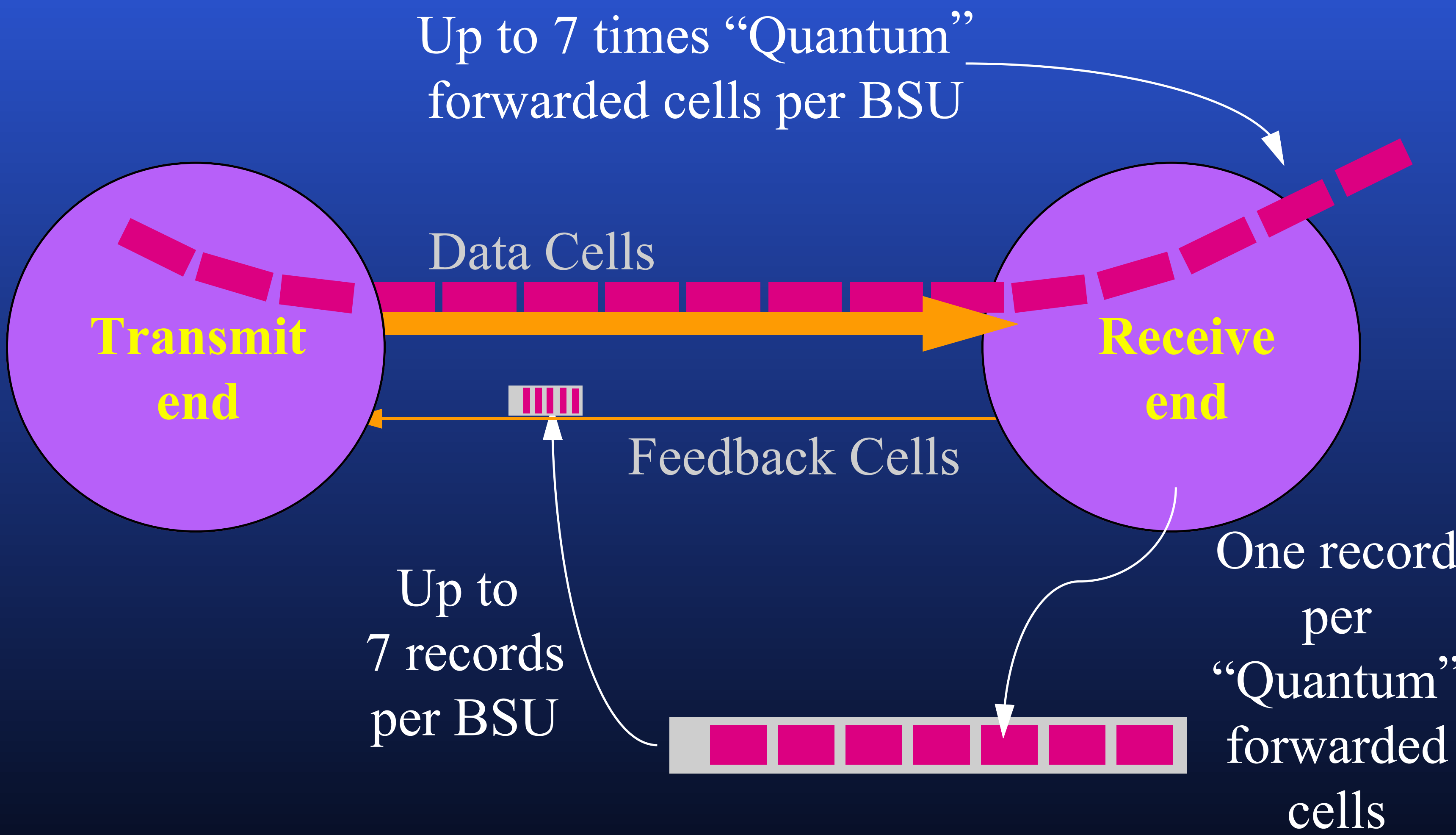
- Control cells
  - Initialization (ACP)
    - Measure link propagation time
  - Buffer State Limit (BSL)
    - Communicate max allowable in-use buffers
  - Buffer State Check (BSC)
    - Adjust for lost feedback cells
- Feedback cells
  - Buffer State Update (BSU)
    - Report forwarded cells

Feedback Quantization

- Form feedback record once per forwarded cell count “quantum”
- Pack feedback records into BSU cell
- Balance feedback overhead vs buffers
  - Pack up to 7 feedback records per BSU cell
  - Pack one record each “quantum” forwarded cells
  - Send BSU cell when fully packed
  - Allowed “quantum” values guarantee instantaneous full data rate with less than 7.5% control overhead

QFC

Feedback Cells



Buffer Sharing

- Limit on committed buffers per link protects buffer pool
  - Permits over-booking with no danger of overflow
- Limit on committed buffers per VC protects against unfair buffer usage
  - High level of sharing with no per VC performance impact

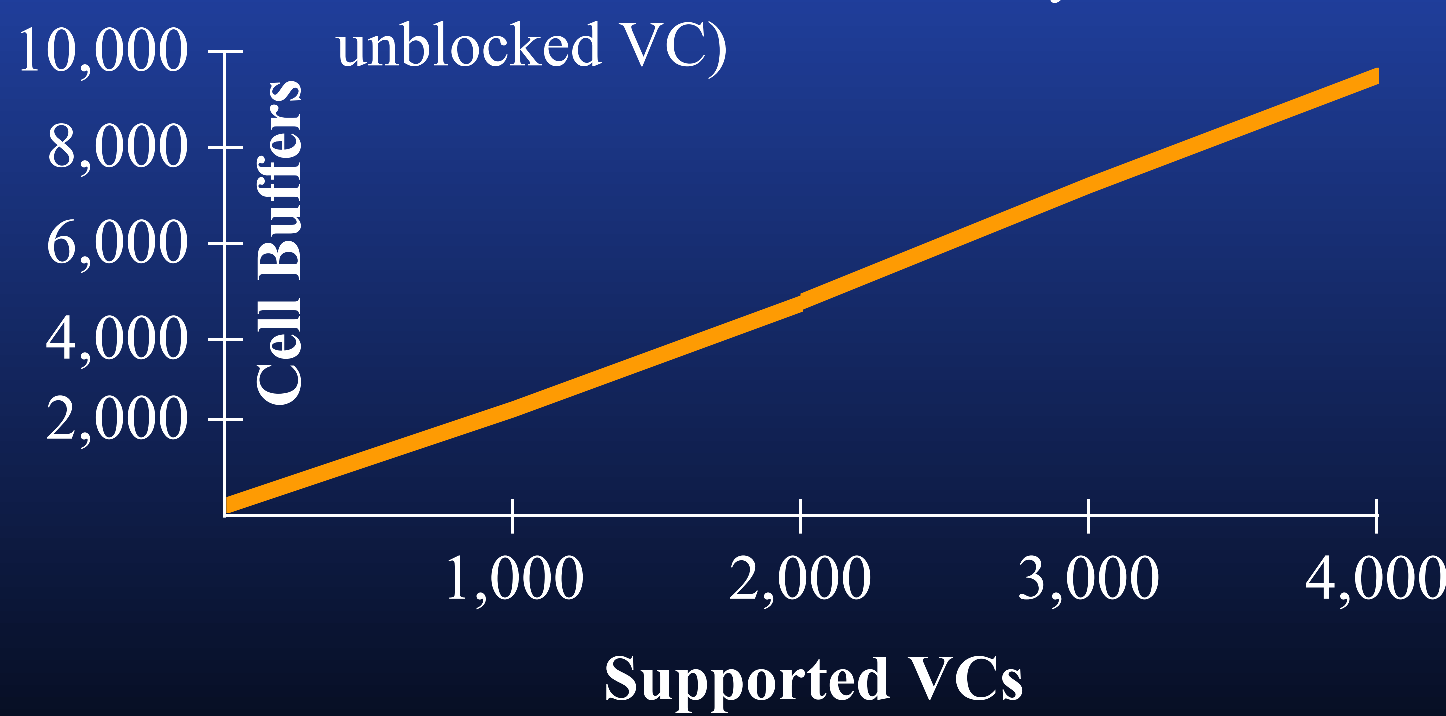
QFC

Buffer Sharing

Example:

Buffers per VC with 90% Sharing

(If less then 10% of supported VCs are blocked, full link rate will be instantly available to any unblocked VC)



Buffer Sharing

- Amount of sharing needed depends on % of VCs with downstream blocking
  - Example:

    2000 defined VCs

    90% sharing (i.e. 10 times overbooking)

    As long as less than 200 VCs are downstream blocked, any VC can get instant access to full bandwidth

QFC

Buffer Sharing

- Need for buffers is greatly diminished by buffer sharing
  - Less buffers = lower cost
- QFC very amenable to sharing
  - 90 - 95% buffer sharing typically possible with no degradation due to flow control
  - 10x - 20x buffer memory reduction
- QFC provides highest performance at low cost