



Out-of-band Flow Control for Reliable Multicast

00S-SIW-071

Harry Wolfson

<HarryWolfson@LL.MIT.EDU>

MIT Lincoln Laboratory

March 30, 2000



Outline

- **Introduction**
- **Time Management Issues**
- **Review of RTI 1.3 Multicast**
- **Throughput Processing Imbalance**
- **Flow Control Design in RTI 1.3**
- **Summary**



Introduction

- **RTI Reliable Multicast began with STOW Program**
 - **Design emphasis on low latency, high throughput performance**
 - **Large buffers accommodated bursty traffic**
 - **“Last resort” / Anomalous behavior**
 - **Drop messages instead of locking up**



Introduction (cont)

- **RTI 1.3 Reliable Multicast required:**
 - **100% Reliability**
 - No dropped messages
 - **Adherence to tick (min, max)**
 - Limited time permitted for processing messages
 - Receive Queues added for temporary storage



Time Management Issues

- **Flow Control has no impact on Time Managed Federations**
 - Wall clock time might slow down / pause
- **Real Time Federations (Hardware in the Loop)**
 - People responsible for planning Execution need to provide adequate communication / computational resources to match Scenario



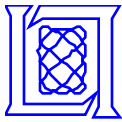
Review of RTI 1.3 Multicast

- **Multicast based on Reliable Distributor**
 - Client / Server
 - Similar to “Exploder”
- **No “production / release” quality
Reliable Multicast available during
STOW and 1.3 development**
 - Many to many multicast
 - Dynamic Join / Leave
 - Support DDM Interest Management



Review of RTI 1.3 Multicast (cont)

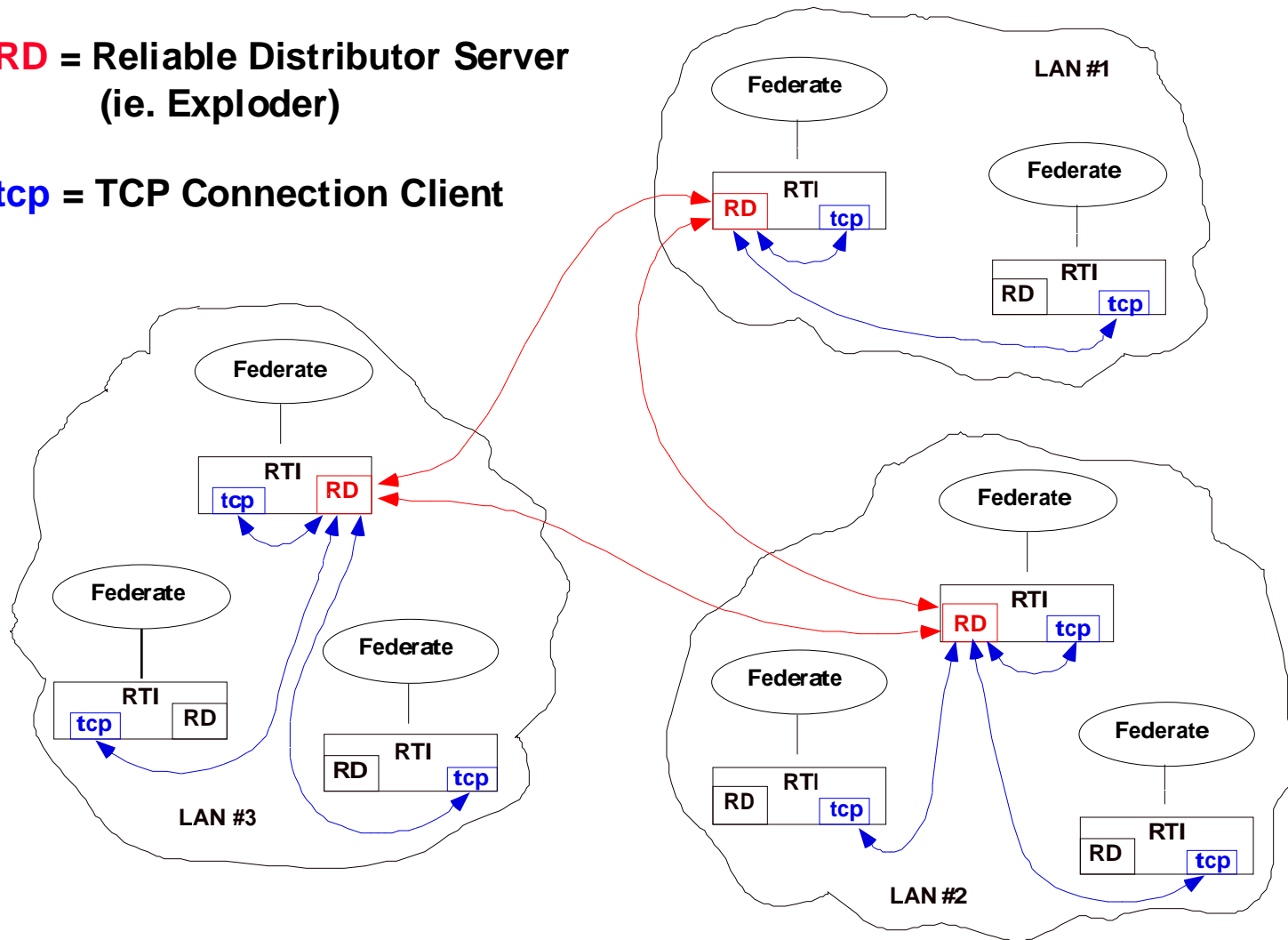
- **Built on top of TCP / IP**
 - **Sequential message delivery**
 - **Reliable point to point msg transfer**
- **TCP does *NOT* provide Application-to-Application Flow Control**
- **Robust, fault tolerant**
- **Interest Filtering (DDM) implemented at sender, exploder and receiver**



Review of RTI 1.3 Multicast (cont)

RD = Reliable Distributor Server
(ie. Exploder)

tcp = TCP Connection Client

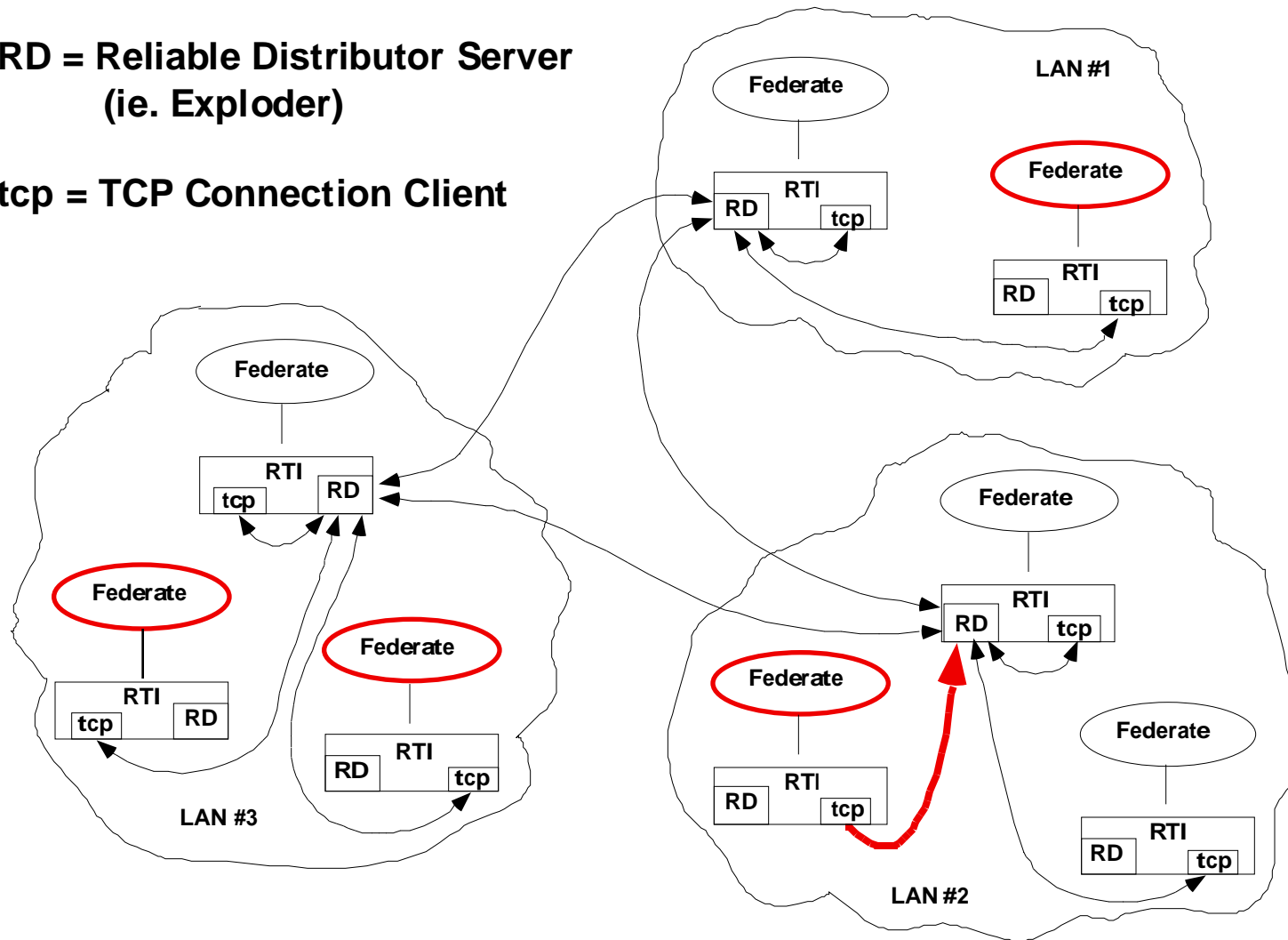




Review of RTI 1.3 Multicast (cont)

RD = Reliable Distributor Server
(ie. Exploder)

tcp = TCP Connection Client

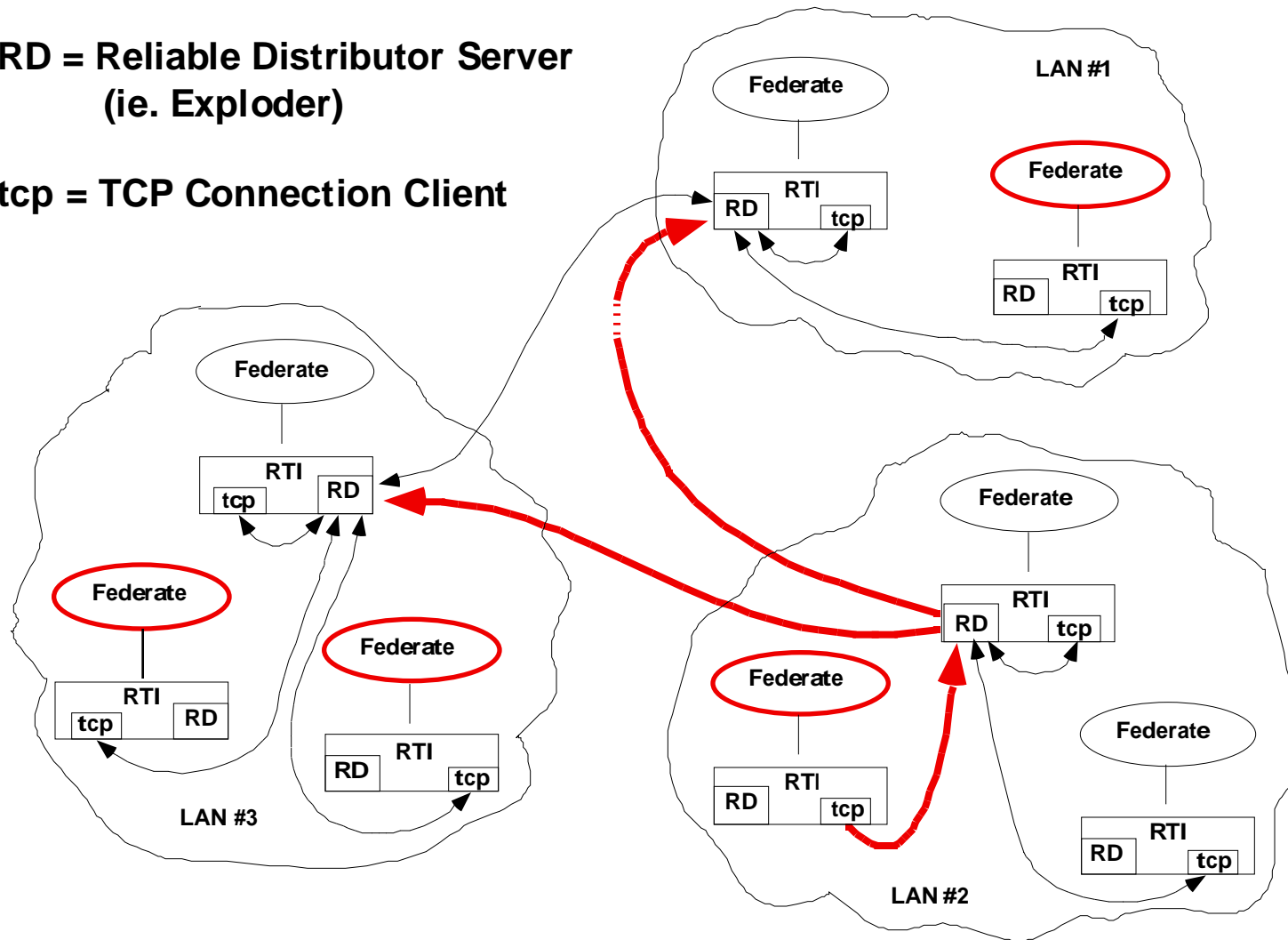


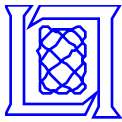


Review of RTI 1.3 Multicast (cont)

RD = Reliable Distributor Server
(ie. Exploder)

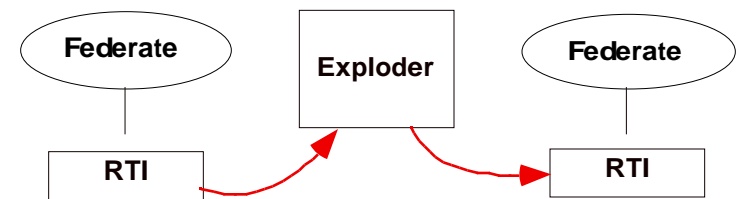
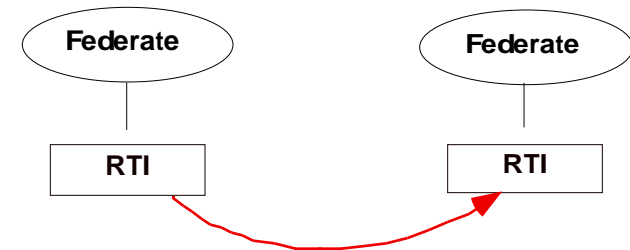
tcp = TCP Connection Client





TCP Alone is NOT Sufficient

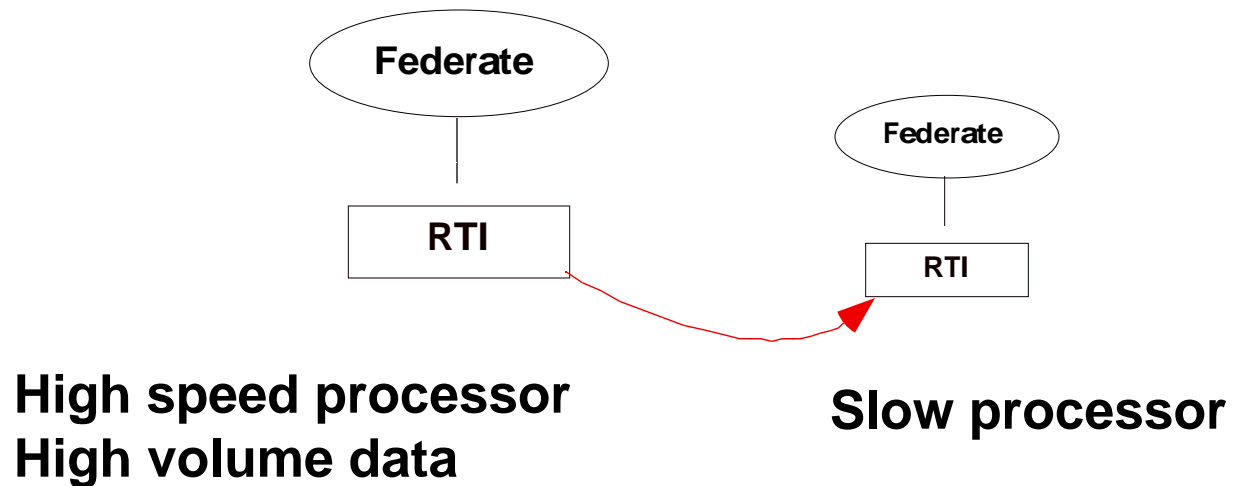
- TCP / IP does *NOT* provide true Application-to-Application Flow Control
 - Single connected pair would require “Blocking Send”
 - Exploder breaks any end-to-end flow control provided by TCP





Throughput Imbalance (a)

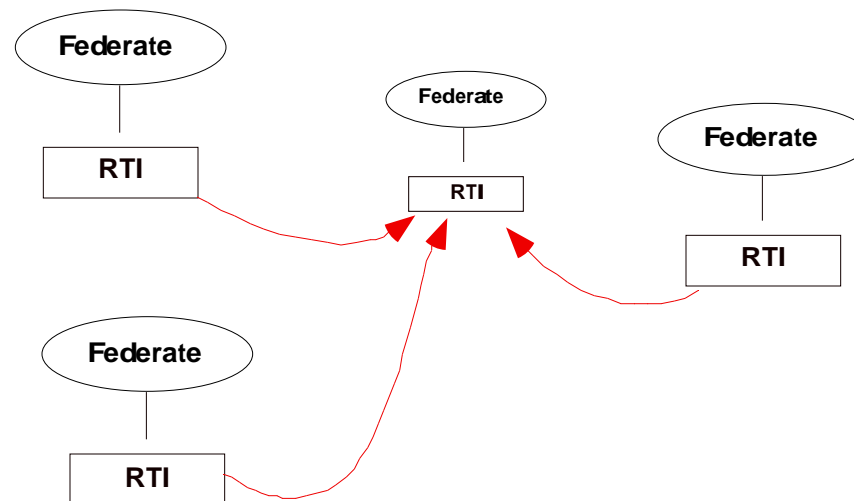
- Many Federation Scenarios can lead to Overloaded Network





Throughput Imbalance (b)

- Many Federation Scenarios can lead to Overloaded Network

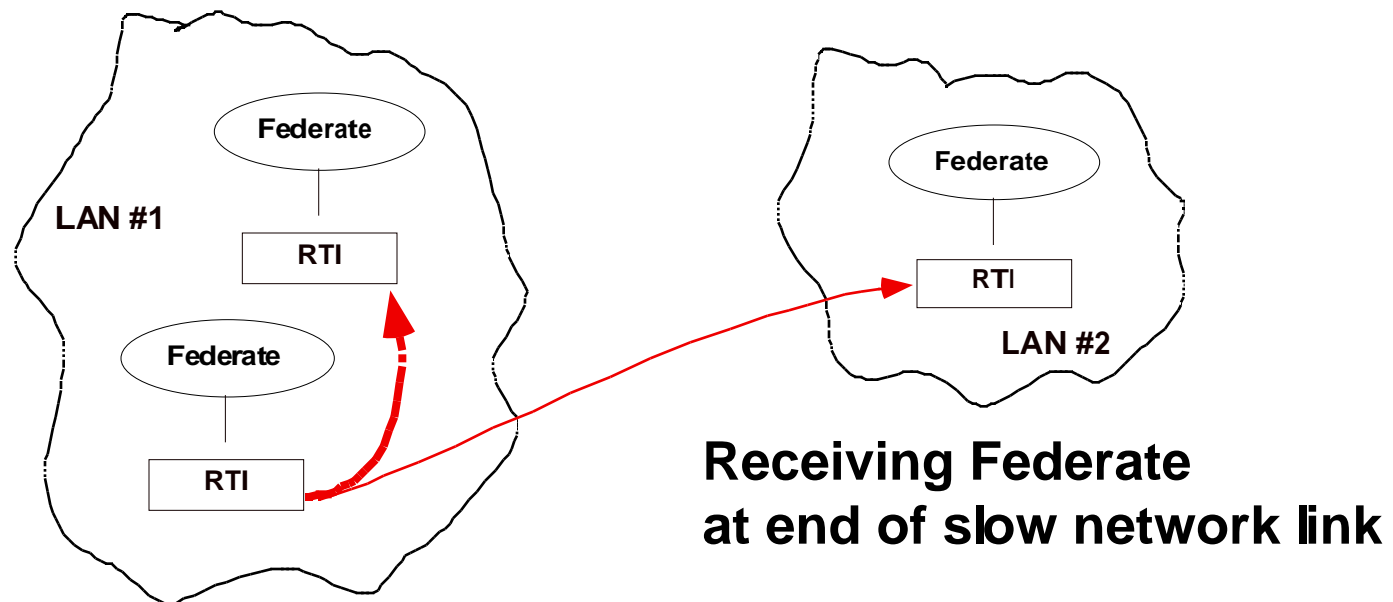


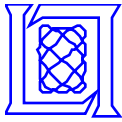
Federate that receives data from numerous high volume Federates



Throughput Imbalance (c)

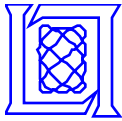
- Many Federation Scenarios can lead to Overloaded Network





Throughput Imbalance (d)

- **Overloaded Execution typical scenario without “Application to Application” Flow Control**
 - Receiver can’t process incoming msgs
 - Buffers in LRCs and kernel begin to fill up
 - Federates, or entire Execution, slows to crawl as Federates try to clear buffers
 - Deadlock / deadly embrace



Flow Control Design in RTI 1.3

- **Regulate message throughput level**
 - Prevent Federates from sending new data based on RTI internal state
 - Squelch / ClearToSend
 - LRC grabs control of tick from Federate
 - Hysteresis in system prevents “thrashing”
- **Out of Band handshake protocol**
 - Full TCP buffers do not impede protocol



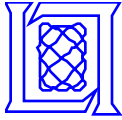
Monitor Internal Queues

- **Each Federate's LRC has two message queues**
 - **Receive Queue stores messages:**
 - After tick(min,max) expires
 - During save / restore
 - **Send Queue stores messages:**
 - After tick(min,max) expires
 - As remote LRC's buffers fill across Execution



Monitor Internal Queues (cont)

- **Reliable Distributor message queues**
 - One Send Queue for each Federate Client
 - One Send Queue for each remote Reliable Distributor
 - Stores messages if clients' buffers full
- **Squelch Mode activated when any Queue exceeds threshold**



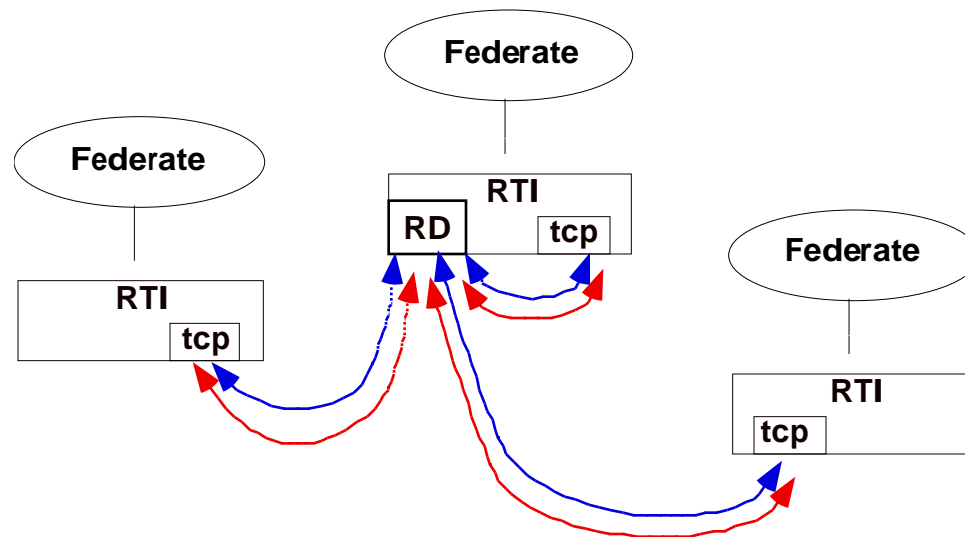
Out-of-Band Messaging Link

- **Squelch / ClearToSend internal state communicated between RTI pairs**
 - **Federate to Reliable Distributor**
 - **Reliable Distributor to Reliable Distributor**
- **UDP “point to point” link between pairs**
 - **Best Effort**
 - **Not subject to TCP congestion**
 - **Heartbeat Status (fail-safe time out)**



Out-of-Band Link (cont)

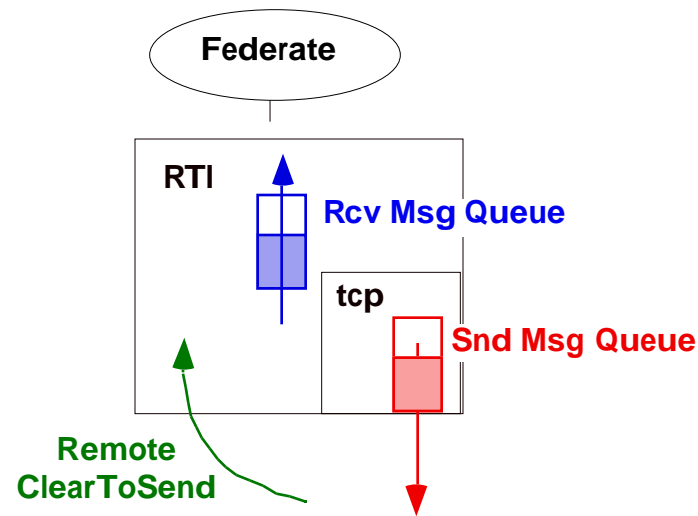
- **UDP link in parallel to every TCP connection**





Squelch Mode

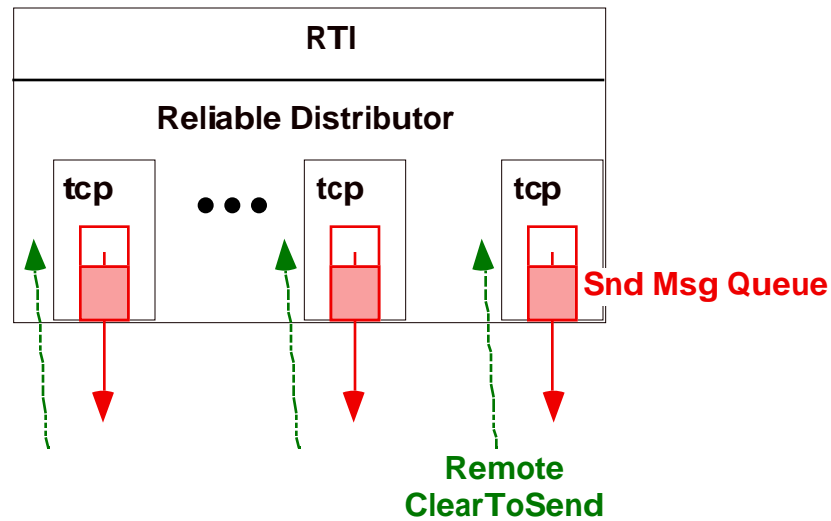
- **Federate's LRC enters Squelch Mode if:**
 - Rcv Msg Queue over threshold, or
 - Snd Msg Queue over threshold, or
 - Received Remote Squelch message
 - or: ClearToSend times out

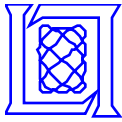




Squelch Mode (cont)

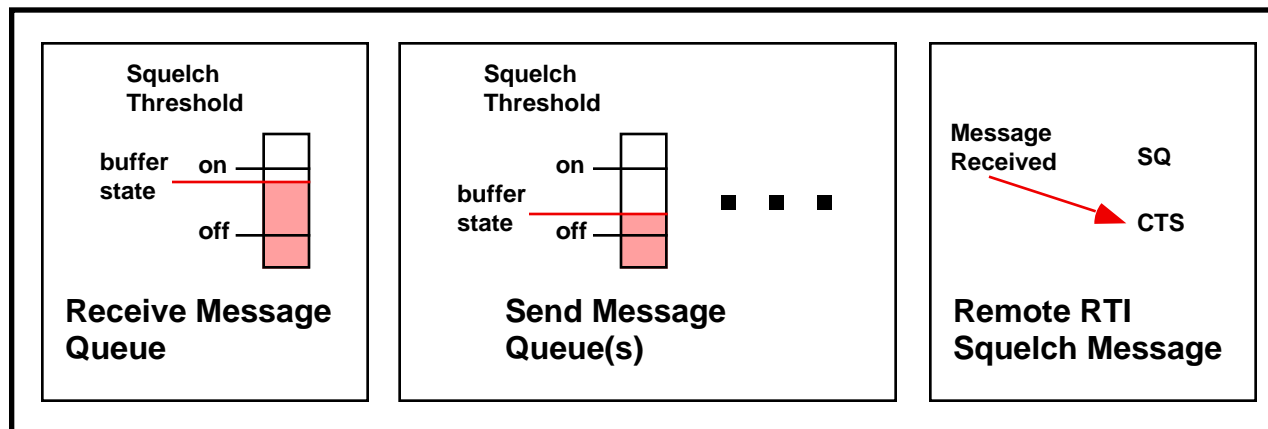
- **Reliable Distributor enters Squelch Mode if any of its Clients:**
 - Snd Msg Queue over threshold
 - Received Remote Squelch message
 - or: ClearToSend times out





Squelch Mode (cont)

- **Federate prevented from sending new messages when LRC in Squelch Mode**
 - LRC grabs control of tick until ClearToSend
- **Built in hysteresis / latency prevents “thrashing”**





Summary

- **End-to-end, “Application to Application”, coordinated, Flow Control implemented in RTI 1.3v7**
- **Prevents new messages from being sent when slower Federates can’t keep up**
 - **Allows sustainable message throughput**



Summary (cont)

- **Demonstrated system wide improvement in large Simulations**
 - **JTC, JTLS**
 - **9 - 11 Federates, ~15,000 objects**
 - **“Reduced” Load Test**
 - **5 Federates; 10,000 objects; updated every tick**
- **Time Managed Simulations not impacted**
- **Real Time requires adequate resources**