

Curso de Introdução Prática ao Simulador de Redes NS2

Instrutor

Eduardo da Silva

Monitores

Elisa Mannes

Fernando H. Gielow

Urlan S. de Barros

Coordenador

Prof. Aldri L. dos Santos



Outubro de 2009



Roteiro

- Conceitos de simulação
- O simulador Network Simulator
- Um exemplo simples de comunicação
- Um exemplo mais avançado com roteadores
- Exercícios

O que é simulação?

- “Uma gama variada de métodos e aplicações que **reproduzem** o comportamento de sistemas reais, usualmente utilizando-se de **ferramentas computacionais**” (*Kelton et al., 1998*)
- “Processo de elaboração de um **modelo** de um **sistema** real (ou hipotético) e a **condução de experimentos** com a finalidade de **entender** o comportamento de um sistema ou **avaliar** sua operação” (*Shannon, 1975*)

Princípio básico da simulação

- Analistas constroem **modelos do sistema** de interesse
- **Escrevem programas** desses modelos
- Utilizam um computador para inicializar o comportamento do sistema e submetê-lo a diversas políticas operacionais

“A melhor política deve ser selecionada” (*Pidd, 2000*)

Terminologia básica

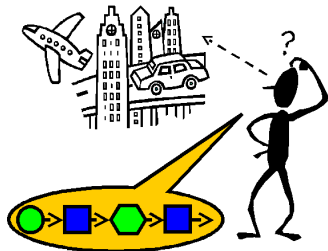
■ Sistema → agrupamento de partes

- Operam juntas
- Possuem um objetivo em comum

■ Modelo

- Representação das relações dos componentes de um sistema
- Abstração → tende a se aproximar do verdadeiro comportamento do sistema

Sistema



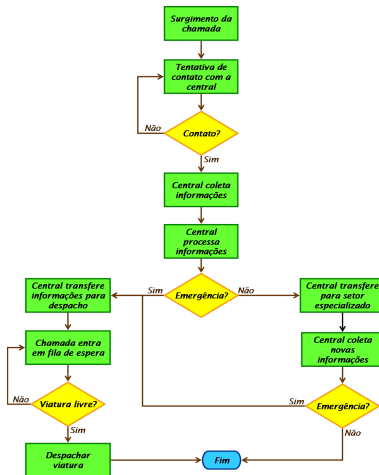
Modelo = representação

Tipos de modelos

- Modelos Simbólicos
- Modelos Analíticos
- Modelos de Simulação

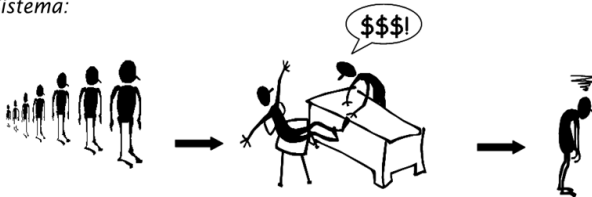
Modelo simbólico

- Símbolos gráficos (fluxogramas, DFD, Layouts etc.)
- Muito utilizado para comunicação e documentação
- Limitações:
 - Estáticos
 - Não fornece elementos quantitativos
 - Não entra no detalhe do sistema



Modelo simbólico: teoria de filas

Sistema:



Modelo Simbólico:



Simbologia:



Modelo analítico

- Forte modelagem matemática (Programação Linear, **Teoria de Filas**, etc.)
- Limitações:
 - Modelos, na grande maioria, estáticos
 - **Complexidade** → pode impossibilitar a busca de soluções analíticas diretas
- Vantagens:
 - Solução **exata**, rápida e, às vezes, **ótima**

Modelo de simulação

- Captura o comportamento do **sistema real**
- Permite a análise pela pergunta: **"E se...?"**
- Capaz de representar **sistemas complexos** de natureza **dinâmica e aleatória**
- **Limitações:**
 - Podem ser de construção **difícil**
 - **Não** há garantia do **ótimo**

Técnicas de simulação

■ Simulação não computacional

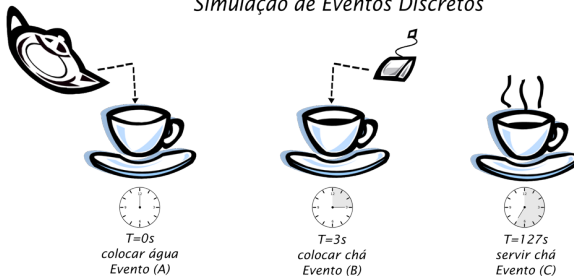
- Protótipo em túnel de vento
- Simulação de acontecimentos

■ Simulação computacional

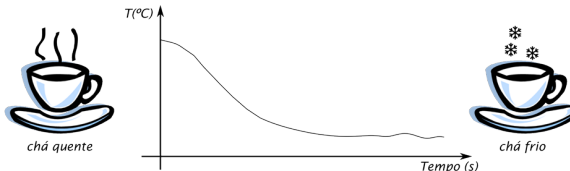
- Simulação estática ou de **Monte Carlo**
- Simulação de sistemas contínuos
- **Simulação de eventos discretos**

Simulação de eventos discretos

Simulação de Eventos Discretos





Simulação Contínua



Por que simular?

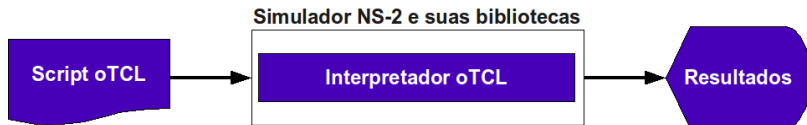
- Analisar um **novo sistema** antes de sua implantação
- **Melhorar a operação** de um sistema já existente
- Compreender melhor **o funcionamento** de um sistema
- **Confrontar** resultados
- Medir **eficiências**

Quando simular?

Problema	Ferramentas	Resultados
<p><i>Maior</i> Complexidade Dinâmica Aleatoriedade</p> 	<p><i>Simulação</i></p> <p>Planilhas</p> <p>Calculadora</p> <p>Lápis e Papel</p> <p>Intuição</p>	<p><i>Maior</i> Esforço Qualidade</p> 

O Network Simulator 2

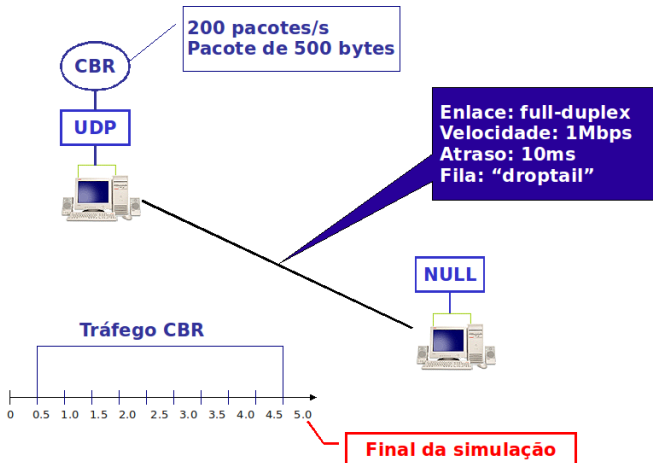
- Simulador de eventos discretos para redes
- Código aberto e livre
 - Disponível em www.isi.edu/nsnam/ns/
- Orientado a objetos
 - Escrito em C++/oTCL
- Um interpretador de scripts oTCL



Instalando o NS-2

- Usaremos a versão atual (ns-2.34)
- Instalação “all-in-one”
- Disponível em:
 - http://downloads.sourceforge.net/project/nsnam/allinone/ns-allinone-2.34/ns-allinone-2.34.tar.gz?use_mirror=ufpr
 - <http://www.nr2.ufpr.br/downloads/ns-allinone-2.34.tar.gz>
- Descompactar e executar “./install”
- Colocar diretórios do *ns* e *nam* no PATH

Um exemplo simples - Cenário



Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns trace-all $f
```

```
proc finish {} {
```

```
    global ns nf f
```

```
    $ns flush-trace
```

```
    close $nf
```

```
    close $f
```

```
    exec nam out.nam &
```

```
    exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
    1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Um exemplo simples

Gera instância do simulador

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
$ns namtrace-all $nf
set f [open out.tr w]
$ns trace-all $f
```

```
proc finish {} {
    global ns nf f
    $ns flush-trace
    close $nf
    close $f
    exec nam out.nam &
    exit 0
}
```

```
set n0 [$ns node]
set n1 [$ns node]
$ns duplex-link $n0 $n1 \
1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
$ns at 4.5 "$cbr0_stop"
$ns at 5.0 "finish"
```

```
$ns run
```

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns trace-all $f
```

```
proc finish {} {
```

```
    global ns nf f
```

```
    $ns flush-trace
```

```
    close $nf
```

```
    close $f
```

```
    exec nam out.nam &
```

```
    exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
    1Mb 10ms DropTail
```

Opções para arquivos de registro

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns trace-all $f
```

```
proc finish {} {  
  global ns nf f  
  $ns flush-trace  
  close $nf  
  close $f  
  exec nam out.nam &  
  exit 0  
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1\  
  1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

Procedimento de finalização

```
$cbr0 . interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns
```

Agendamento da "finalização"

```
proc finish {} {
```

```
  global ns nf f
```

```
  $ns flush-trace
```

```
  close $nf
```

```
  close $f
```

```
  exec nam out.nam &
```

```
  exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
  1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns trace-all $f
```

```
proc finish {} {
```

```
  global ns nf f
```

```
  $ns flush-trace
```

```
  close $nf
```

```
  close $f
```

```
  exec nam out.nam &
```

```
  exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
  1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns Criação dos nós $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns trace-all $f
```

```
proc finish {} {
```

```
    global ns nf f
```

```
    $ns flush-trace
```

```
    close $nf
```

```
    close $f
```

```
    exec nam out.nam &
```

```
    exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1
```

```
    1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

Criação do enlace full-duplex

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```


Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns tra
```

Criação de um agente

```
proc finish {} {
```

```
  global ns nf f
```

```
  $ns flush-trace
```

```
  close $nf
```

```
  close $f
```

```
  exec nam out.nam &
```

```
  exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
  1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns tr Criação de outro agente
```

```
proc finish {} {
```

```
  global ns nf f
```

```
  $ns flush-trace
```

```
  close $nf
```

```
  close $f
```

```
  exec nam out.nam &
```

```
  exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
  1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns trace-all $f
```

```
proc finish {} {
```

```
    global ns nf f
```

```
    $ns flush-trace
```

```
    close $nf
```

```
    close $f
```

```
    exec nam out.nam &
```

```
    exit 0
```

```
}
```

```
set n0 [new Agent/UDP]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Conectando dois agentes

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns trace-all $f
```

```
proc finish {} {
```

```
  global ns nf f
```

```
  $ns flush-trace
```

```
  close $f
```

```
  close $f
```

```
  exec nam out.nam &
```

```
  exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
  1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Configurando a aplicação

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns trace-all $f
```

```
proc finish {} {
```

```
    global ns nf f
```

```
    $ns flush-trace
```

```
    close $nf
```

```
    close $f
```

```
    exec nam out.nam &
```

```
    exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

```
$ns at 5.0 "finish"
```

```
$ns run
```

Início e fim da transmissão

Um exemplo simples

```
set ns [new Simulator]
```

```
set nf [open out.nam w]
```

```
$ns namtrace-all $nf
```

```
set f [open out.tr w]
```

```
$ns trace-all $f
```

```
proc finish {} {
```

```
    global ns nf f
```

```
    $ns flush-trace
```

```
    close $nf
```

```
    close $f
```

```
    exec nam out.nam &
```

```
    exit 0
```

```
}
```

```
set n0 [$ns node]
```

```
set n1 [$ns node]
```

```
$ns duplex-link $n0 $n1 \
```

```
    1Mb 10ms DropTail
```

```
set udp0 [new Agent/UDP]
```

```
$ns attach-agent $n0 $udp0
```

```
set cbr0 [new Application/Traffic/CBR]
```

```
$cbr0 set packetSize_ 500
```

```
$cbr0 set interval_ 0.005
```

```
$cbr0 attach-agent $udp0
```

```
set null0 [new Agent/Null]
```

```
$ns attach-agent $n1 $null0
```

```
$ns connect $udp0 $null0
```

```
$ns at 0.5 "$cbr0_start"
```

```
$ns at 4.5 "$cbr0_stop"
```

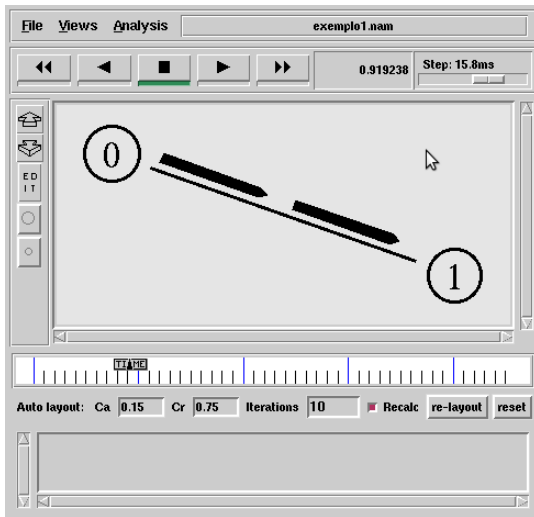
```
$ns at 5.0 "finish"
```

```
$ns run
```

Início da simulação

Resultado

via Network Animator



Resultado

via Arquivo de Trace

evt	temp.	ori.	dst.	tp pct	tam	flags	fluxo	# orig.	# dest.	# seq	ID pct
r	0.529	0	1	cbr	500	- - - - -	0	0.0	1.0	3	3

r 0.529 0 1 cbr 500 - - - - - 0 0.0 1.0 3 3

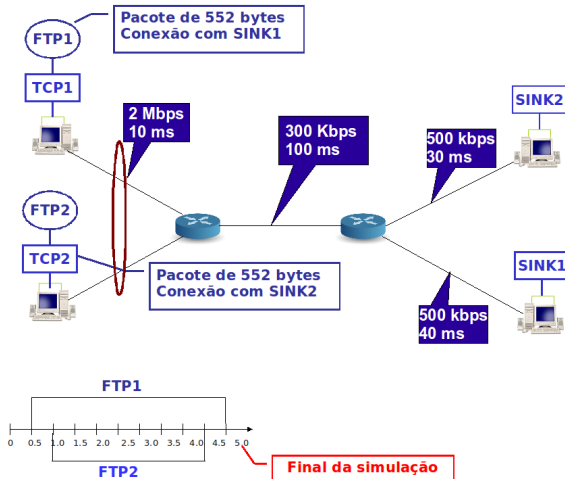
■ Eventos

- +: pacote adicionado na fila
- -: pacote saiu da fila
- r: pacote recebido
- s: pacote enviado
- d: pacote descartado
- entre outros

Vamos exercitar?

- 1 Executar o exemplo apresentado
- 2 Fazer algumas alterações
 - Aumentar/diminuir o tamanho dos pacotes
 - Aumentar/diminuir a taxa de envio de dados

Um exemplo mais complicado



Criação dos nós e enlaces

Como era

```
set n0 [$ns node]
set n1 [$ns node]
$ns duplex-link $n0 $n1 1Mb 10ms DropTail
```

Como fica

```
set n0 [$ns node]
set n1 [$ns node]
set n2 [$ns node]
set n3 [$ns node]
set r0 [$ns node]
set r1 [$ns node]

$ns duplex-link $n0 $r0 2Mb 10ms DropTail
$ns duplex-link $n1 $r0 2Mb 10ms DropTail
$ns duplex-link $n2 $r1 500Kb 30ms DropTail
$ns duplex-link $n3 $r1 500Kb 40ms DropTail
$ns duplex-link $r0 $r1 300Kb 100ms DropTail
```

Criação dos agentes de transporte

Como era

```
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
set null0 [new Agent/Null]
$ns attach-agent $n1 $null0

$ns connect $udp0 $null0
```

Como fica

```
set tcp0 [new Agent/TCP]
$tcp0 set class_ 2
$tcp0 set packetSize 552
$ns attach-agent $n1 $tcp0
set sink0 [new Agent/TCPSink]
$ns attach-agent $n2 $sink0

$ns connect $tcp0 $sink0
```

Criação das aplicações

Como era

```
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 0.005
$cbr0 attach-agent $udp0

$ns at 0.5 "$cbr0_start"
$ns at 4.5 "$cbr0_stop"
```

Como fica

```
set ftp0 [new Application/FTP]
$ftp0 attach-agent $tcp0

$ns at 0.5 "$ftp0_start"
$ns at 4.5 "$ftp0_stop"
```

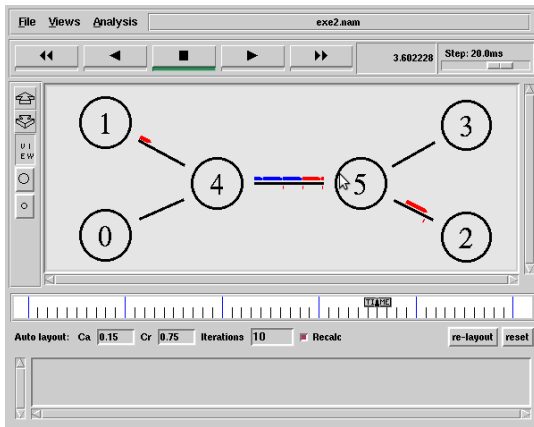
Definindo cores para os fluxos

Adicionar ao código

```
$ns color 1 Blue  
$ns color 2 Red
```

Resultado

via Network Animator



Exercícios 1

- 1 Introduzir congestionamento no exemplo anterior, limitando o tamanho máximo da fila entre os roteadores em 15 pacotes
- 2 Simular uma rede com 10 nós ligados em anel, usando um protocolo de roteamento dinâmico
 - Criar duas fontes de tráfego CBR. Uma entre 1 e 4 e outra entre 2 e 5
 - Simule um queda do enlace entre os nós 2 e 3 entre os tempos 3.0 e 5.0
 - Use o comando “\$ns rtproto DV” para o roteamento dinâmico

Exercícios 2

- ❶ Criar um cenário com as seguintes características:
 - Um servidor de FTP conectado a um roteador R1 a 100Mbps e atraso de 8ms
 - 5 clientes conectados a um roteador R2 a 16Mbps e atraso de 25ms
 - Roteador R1 e R2 interligados a 56Mbps e atraso de 12ms
 - Todos os cliente solicitam dados do servidor entre os tempos 0.5 e 4.5
- ❷ Para melhorar o desempenho do cliente 1, responda:
 - Ele deveria aumentar o seu enlace para 25Mbps ou 50Mbps?
 - Seria melhor investir num enlace *Fast-Path* que reduzisse o atraso para 15ms?
 - Poderia ser usada uma disciplina de enfileiramento para influenciar a decisão do usuário?

OBRIGADO!