

Performance Evaluation with Java Modelling Tools

Giuliano Casale

Department of Computing
Imperial College London, UK

Joint work with:

Giuseppe Serazzi (Politecnico di Milano, Italy)
Lulai Zhu (Imperial College London, UK)

Outline

- Introduction
- Activity 1: getting started
- Activity 2: load balancing
- Activity 3: parameter sweeping
- Activity 4: capacity constraints
- Activity 5: workflows & fork-join

- Please download the latest JMT (v1.0.2) here:
 - <http://jmt.sf.net/Download.html>

Introduction

Java Modelling Tools

- Simulation and analysis of queueing networks.
- Project started in 2002 at Politecnico di Milano, since 2010 co-developed at Imperial.
- JMT is open source: GPL v2,
 - Medium-size project: ~1,000 classes
 - JAR, source code and maven build files (*pom.xml*)
<http://jmt.sourceforge.net/Download.html>
- Good diffusion (59k downloads, mostly from the US)
- Community interaction mainly through
 - Bug reports
 - Feature requests
 - Templates

Supported models

- Queueing Systems
- Queueing Networks (QN)
 - Product-form
 - Extended (fork/join, blocking, priorities, ...)
- Petri Nets (PN)
 - Stochastic Petri Nets (SPN)
 - Generalized Stochastic Petri Nets (GSPN)
 - Coloured Petri Nets (CPN)
- Queueing Petri Nets (QPNs)

Who uses JMT?

- JMT is for PE practice, teaching, and research
- Several university courses worldwide (tell us!)
- Supporting materials available on website

The screenshot shows the homepage of the Java Modelling Tools (JMT) website. The header features the Politecnico di Milano logo and the text "DEIB - Politecnico di Milano - Italy" and "Project Coordinator: G.Serazzi". Below the header, the title "Java Modelling Tools" is prominently displayed. A blue navigation bar at the top right includes the "Documentation" link. The main content area is divided into several sections: "Main Menu" (with links to Introduction, Download JMT, Requirements, JSIMgraph, JSIMwiz, JMVA, JABA, JWAT, JMCH, Documentation, and License), "Links" (with links to Discussion Forums, Report a Bug, Request a New Feature, and Sourceforge Page), "Manuals & Books" (listing the "Users manual" and a book by G.Serazzi), "Tutorials" (listing a paper by G.Casale and G.Serazzi), and "Papers" (listing two academic papers by M.Bertoli, G.Casale, and G.Serazzi).

Java Modelling Tools

Documentation

Manuals & Books

- Java Modelling Tools - *Users manual*
v.0.9.1, 191 pp., Oct. 22th 2013 ([Manual](#))
- G.Serazzi Ed.
Performance Evaluation Modelling with JMT: learning by examples
Politecnico di Milano - DEI, TR 2008.09, 366pp., June 2008 ([Book](#))

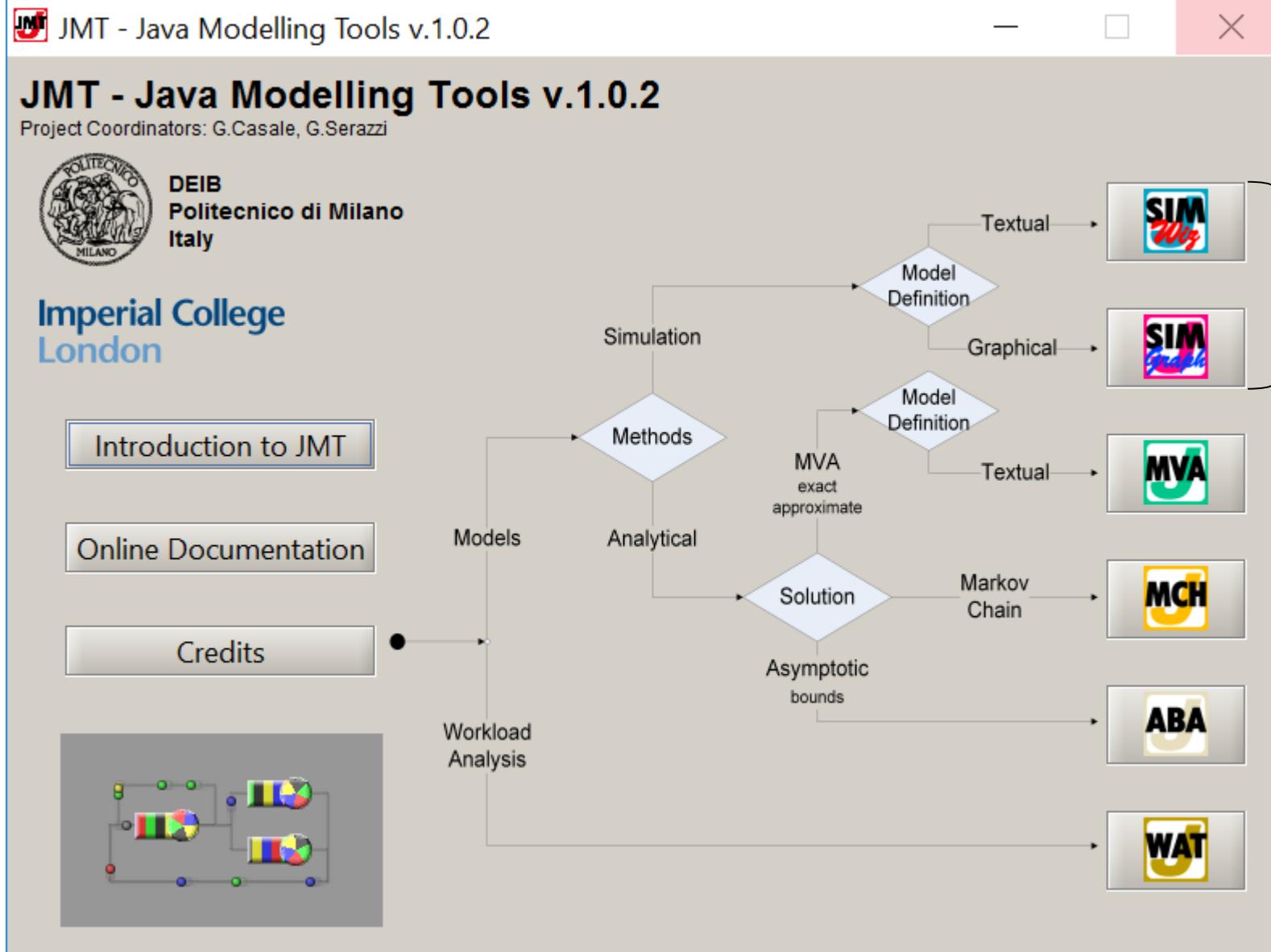
Tutorials

- G.Casale, G.Serazzi.
Quantitative System Evaluation with Java Modelling Tools.
2nd ACM/SPEC International Conference on Performance Engineering (ICPE), March 2011, Karlsruhe, German

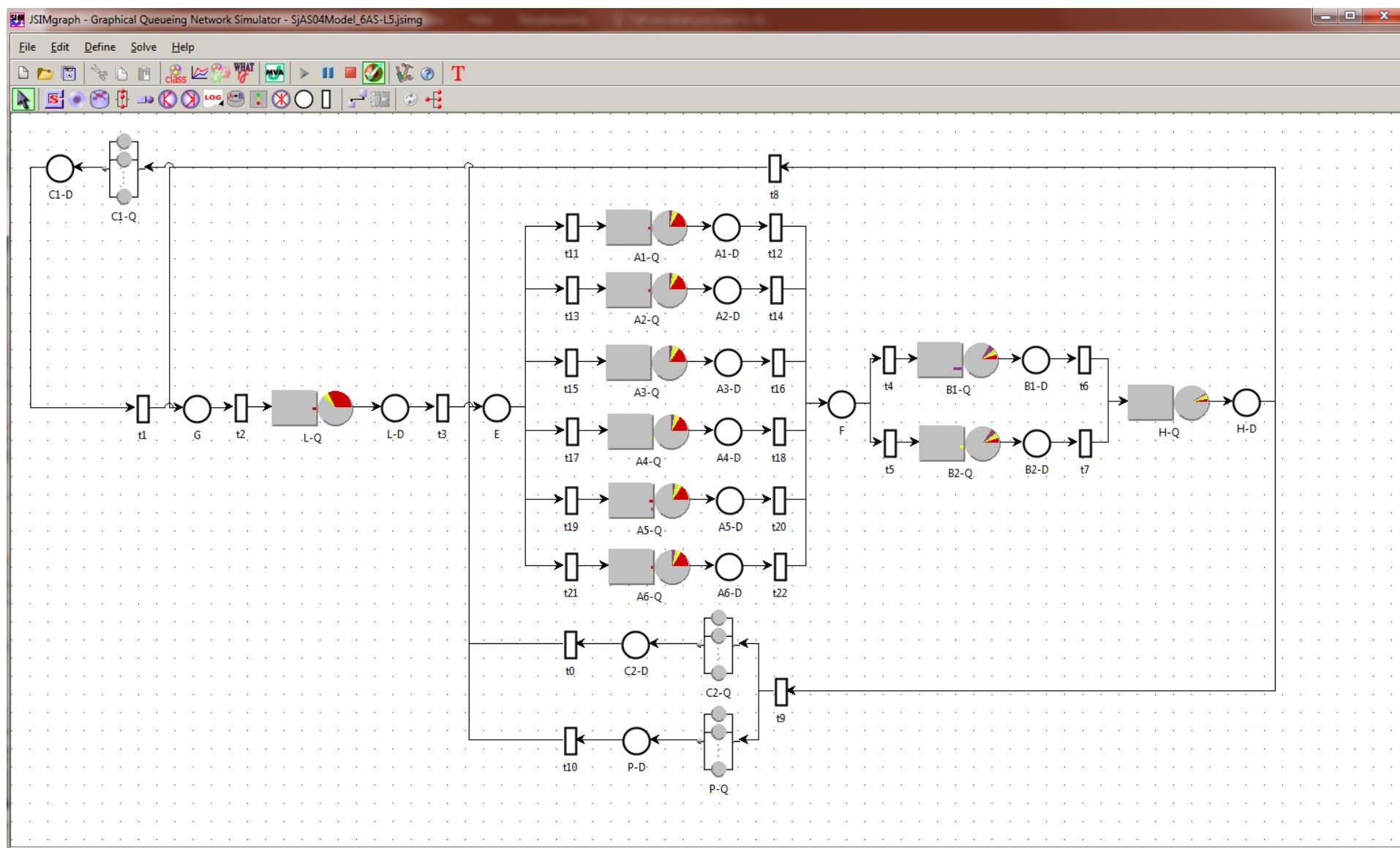
Papers

- M.Bertoli, G.Casale, G.Serazzi.
JMT: performance engineering tools for system modeling.
ACM SIGMETRICS Performance Evaluation Review, Volume 36 Issue 4, New York, US, March 2009, 10-15, ACM
- M.Bertoli, G.Casale, G.Serazzi.
User-Friendly Approach to Capacity Planning Studies with Java Modelling Tools.
Int.I ICST Conf. on Simulation Tools and Techniques, SIMUTools 2009, Rome, Italy, 2009, ACM press. ([Article](#))

JMT Start Screen



JSIMgraph: QN & PN simulation



Templates



Add/See Available Templates

Site: **JMT Off**

- JMT Official Site**
 - Intranet**
 - Three Tier Intranet**
 - Parallel computation**
 - Parallel**
 - System Components**
 - Ceph**
 - RAID**

This template generates an intranet with (for the data management). The number of service stations (that n If the classes of workload have already A limit to the number of customers in may change all the parameters as need

3tier_Intranet

The routing probabilities are the same for all stations of each tier

Model layout

Presentation tier

Name of the service centers **presServer**

Number of service centers **1**

Number of processors per service center **1**

Mean service time per class (same for all stations):

Class1	1
--------	---

Application tier

Name of the service centers **appServer**

Number of service centers **1**

Number of processors per service center **1**

Mean service time per class (same for all stations):

Class1	1
--------	---

Data tier

Name of the service centers **storageServer**

Number of service centers **1**

Number of processors per service center **1**

Mean service time per class (same for all stations):

Class1	1
--------	---

Finite Capacity Region

Max number of customers in the Intranet (-1 = infinite)

-1

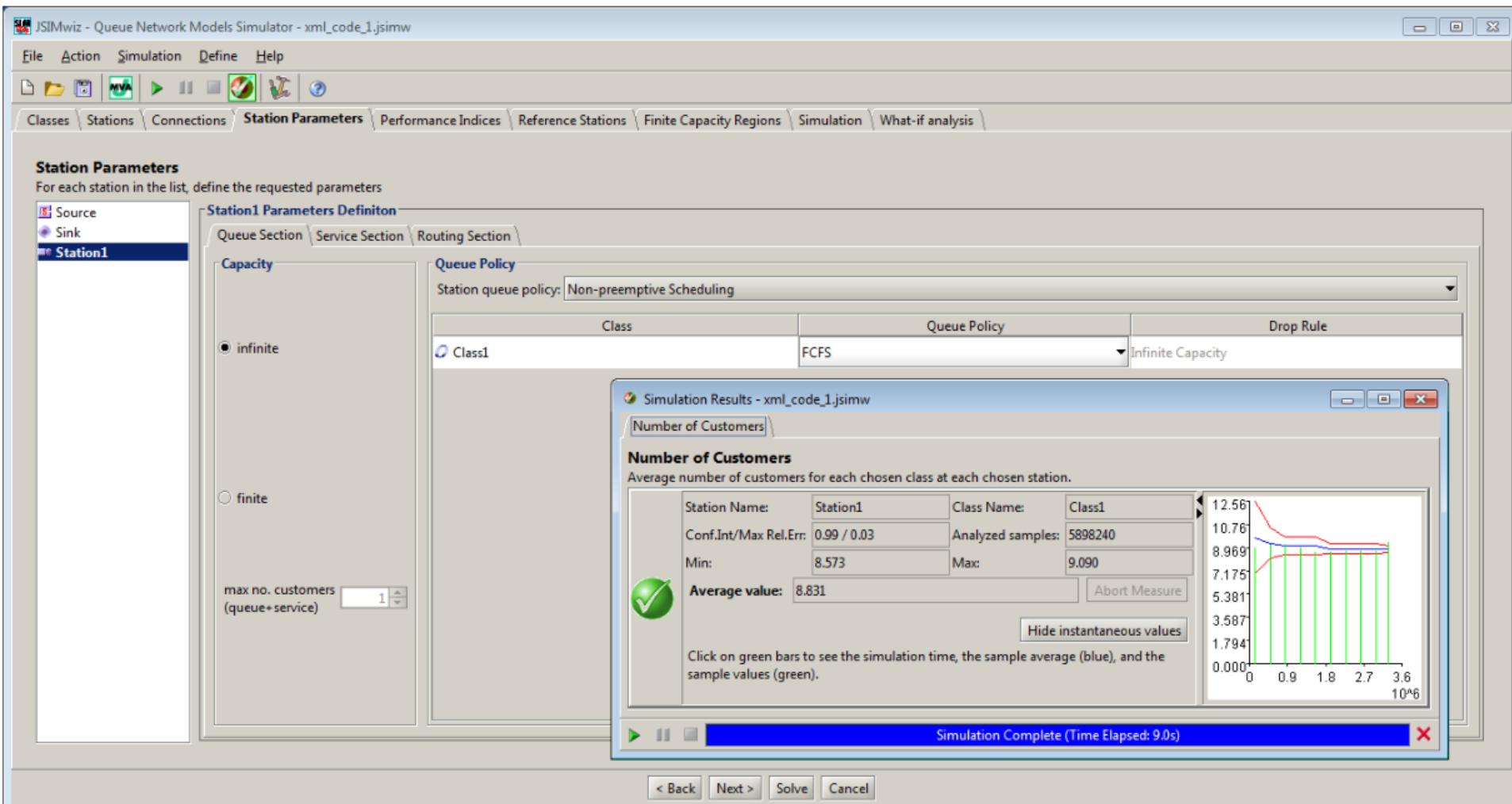
Add/Use templates

Add/See available templates

Template	Version
Parallel	1.06
Three Tier Intranet	1.09

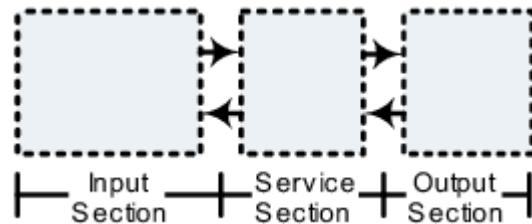
Insert

JSIMwiz: wizard-based user interface

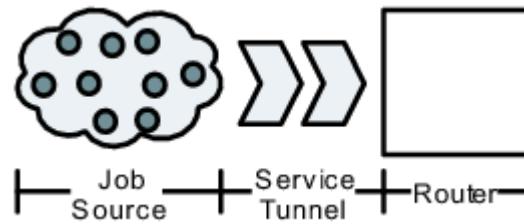


JSIMengine: discrete-event simulator

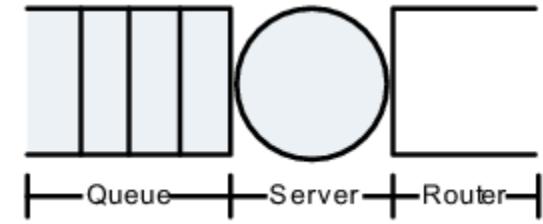
- Simulation components defined by 3 sections



component sections

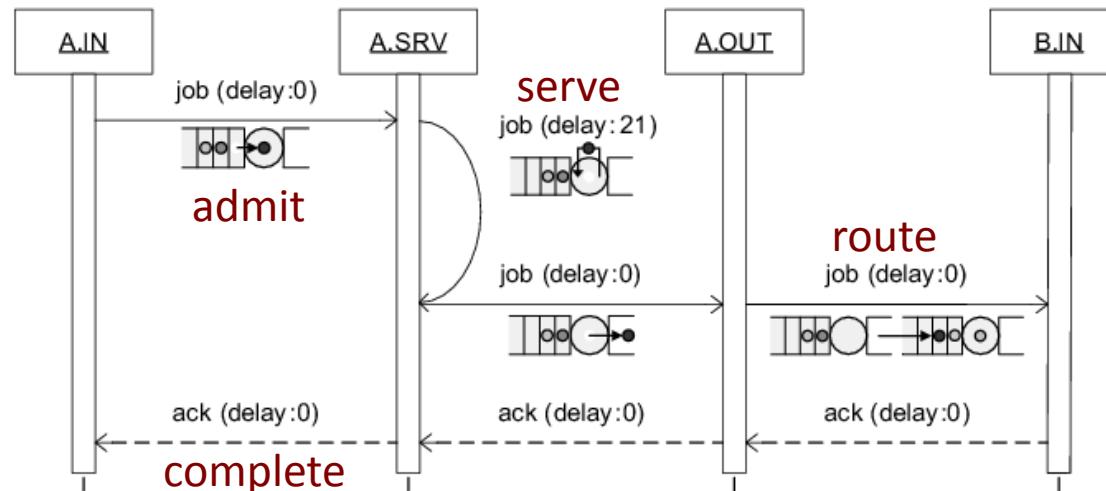


external arrivals
(open class)

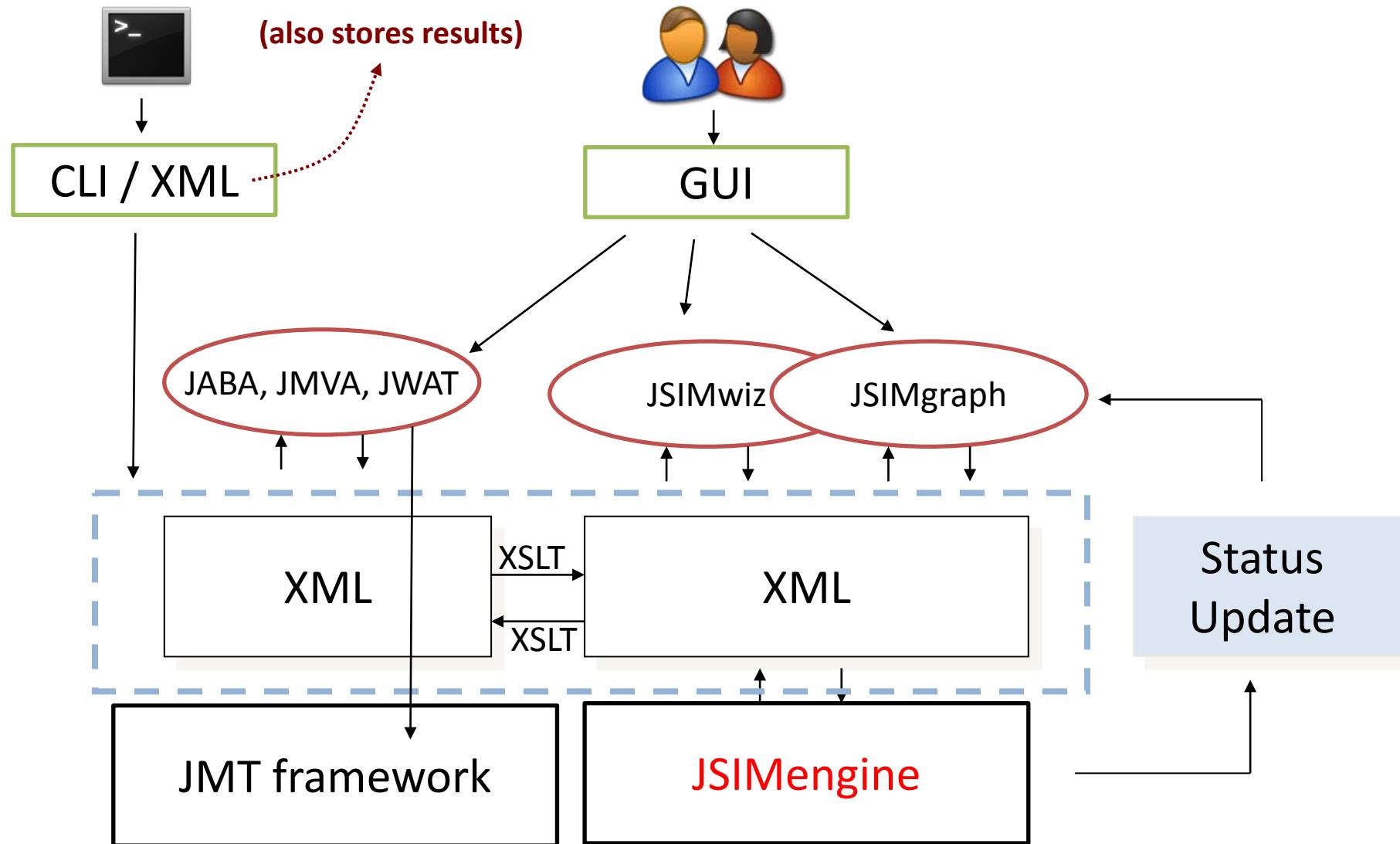


queueing station

- Discrete-event simulation of section messaging



JSIMengine: JMT architecture



JMVA: analytical solver

- Analysis of product-form queueing networks
- Several exact and approximate algorithms
 - *Exact MVA*
 - Reiser & Lavenberg $O(N^R)$
 - Load-dependent $O(N^{2R})$
 - *Approximate MVA* $O(1)$
 - Chow
 - Bard-Schweitzer
 - AQL
 - Linearizer
 - De Souza-Muntz
 - *Normalizing constant*
 - RECAL $O(N^M)$
 - CoMoM $O(N \log N)$

N : *jobs*

M : *stations*

R : *classes*

JMVA: model parameterization

 JMVA - Product form queueing network solver

File Action Help

Algorithm: MVA

Classes Stations **Service Demands** Reference Station What-if Comment

Service Demands
Input service demands of each station and class.
If the station is "Load Dependent" you can set the service demands for each number of customers by double-click on "LD Settings..." button.
Press "Service Times and Visits" button to enter service times and visits instead of service demands.

*	Class1	Class2
Storage1	20.000000	90.000000
Storage2	80.000000	30.000000
Storage3	31.000000	33.000000
ApplServer1	14.000000	20.000000
ApplServer2	23.000000	14.000000
WebServer	12.000000	7.000000

JMVA: solutions

JMVA Solutions - MVA

Synopsis | Throughput | Number of Customers | Residence Times | System Response Time | **Utilization** | System Power

Utilization
Utilization of a customer class at the selected station. The utilization of a delay station is the average number of customers in the station (it may be greater than 1)

*	Aggregate	Class1	Class2
Storage1	0.999999	0.181818	0.818181
Storage2	1.000000	0.727273	0.272727
Storage3	0.581818	0.281818	0.300000
ApplServe...	0.309091	0.127273	0.181818
ApplServe...	0.336364	0.209091	0.127273
WebServer	0.172727	0.109091	0.063636

Choice of solution algorithm

File Action Help

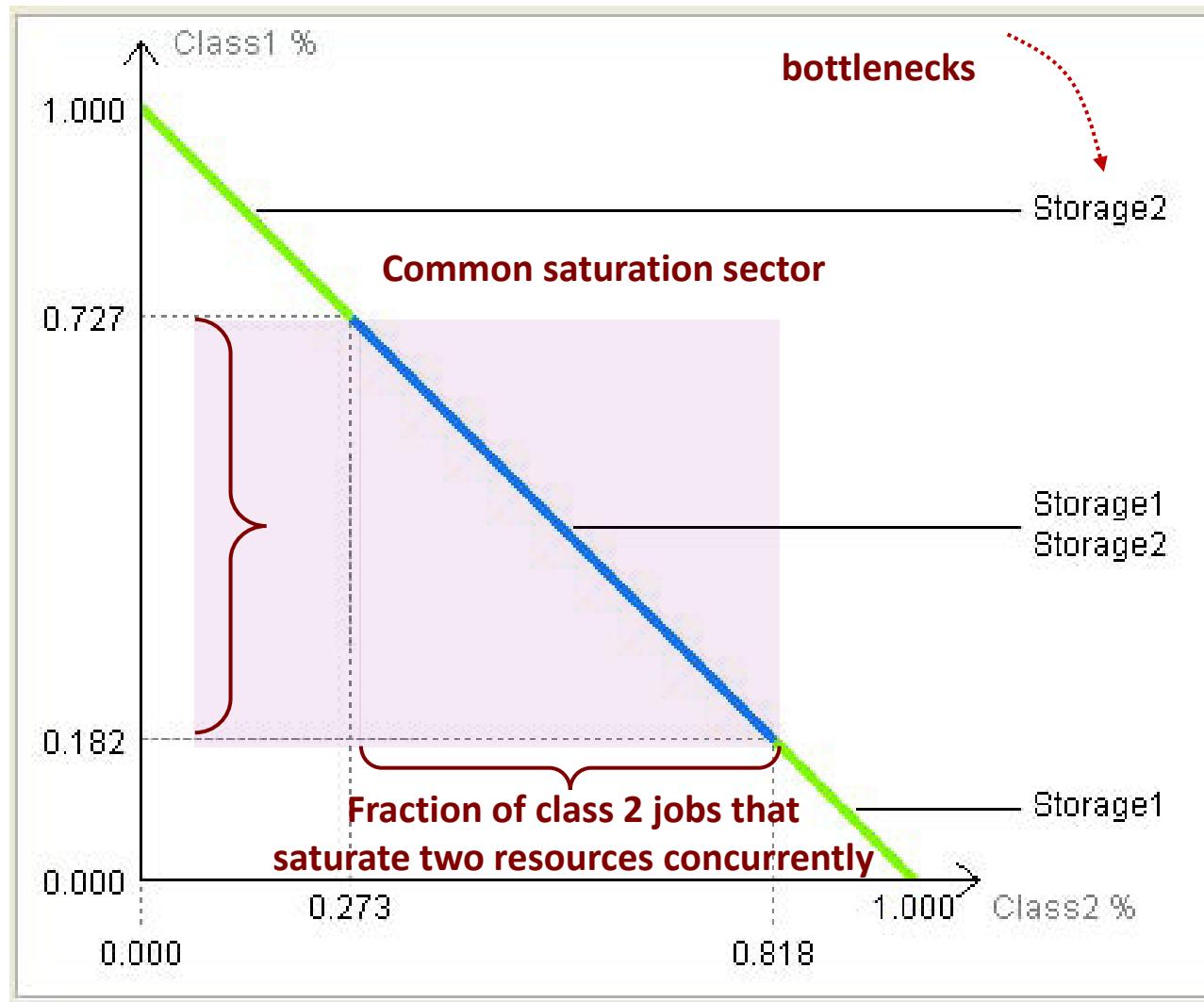
Algorithm: MVA

RECAL
MoM
CoMoM
----- Approximate -----
Chow
Bard-Schweitzer
AQL
Linearizer

Classes | Stations | Service Times

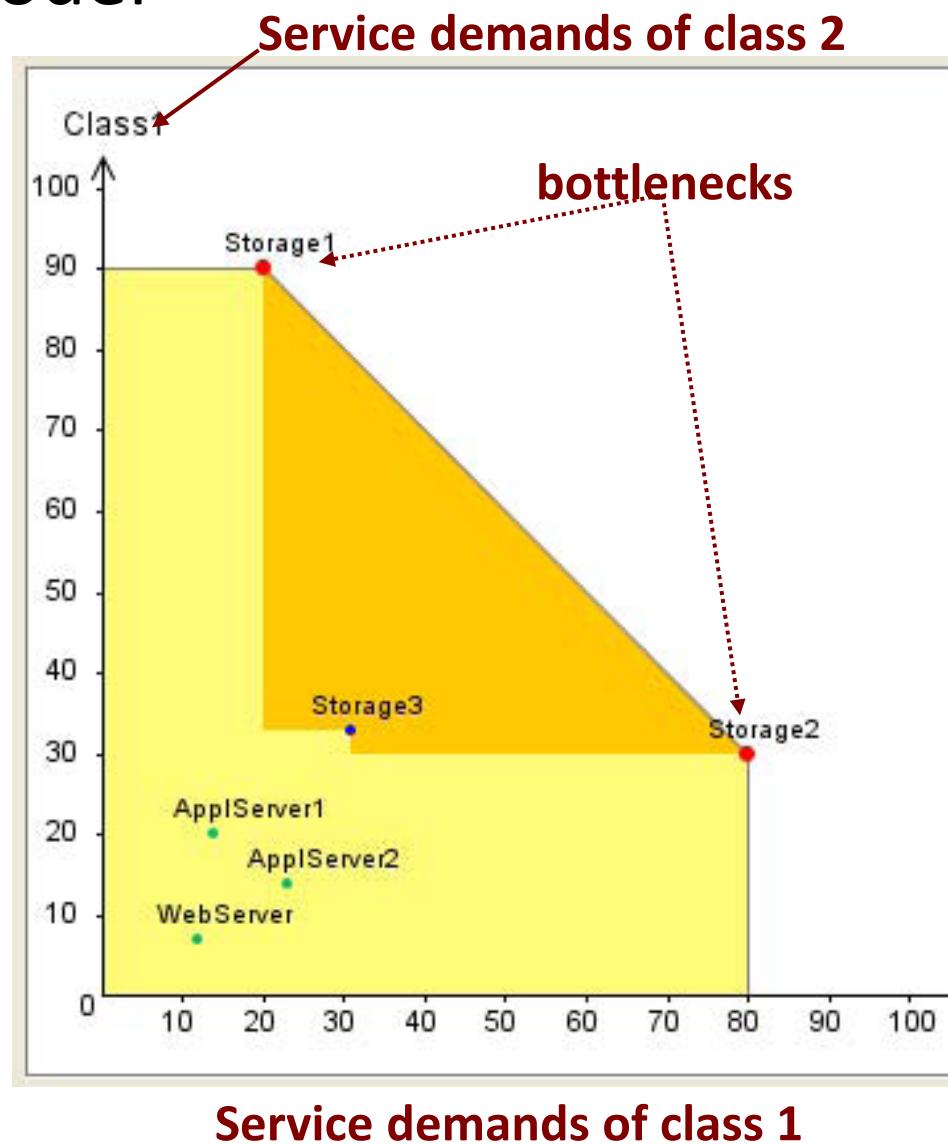
What-if analysis
Select a control parameter if you want to model your system with its values changing in time. The performance indices will be displayed in real time.

JABA: bottleneck identification

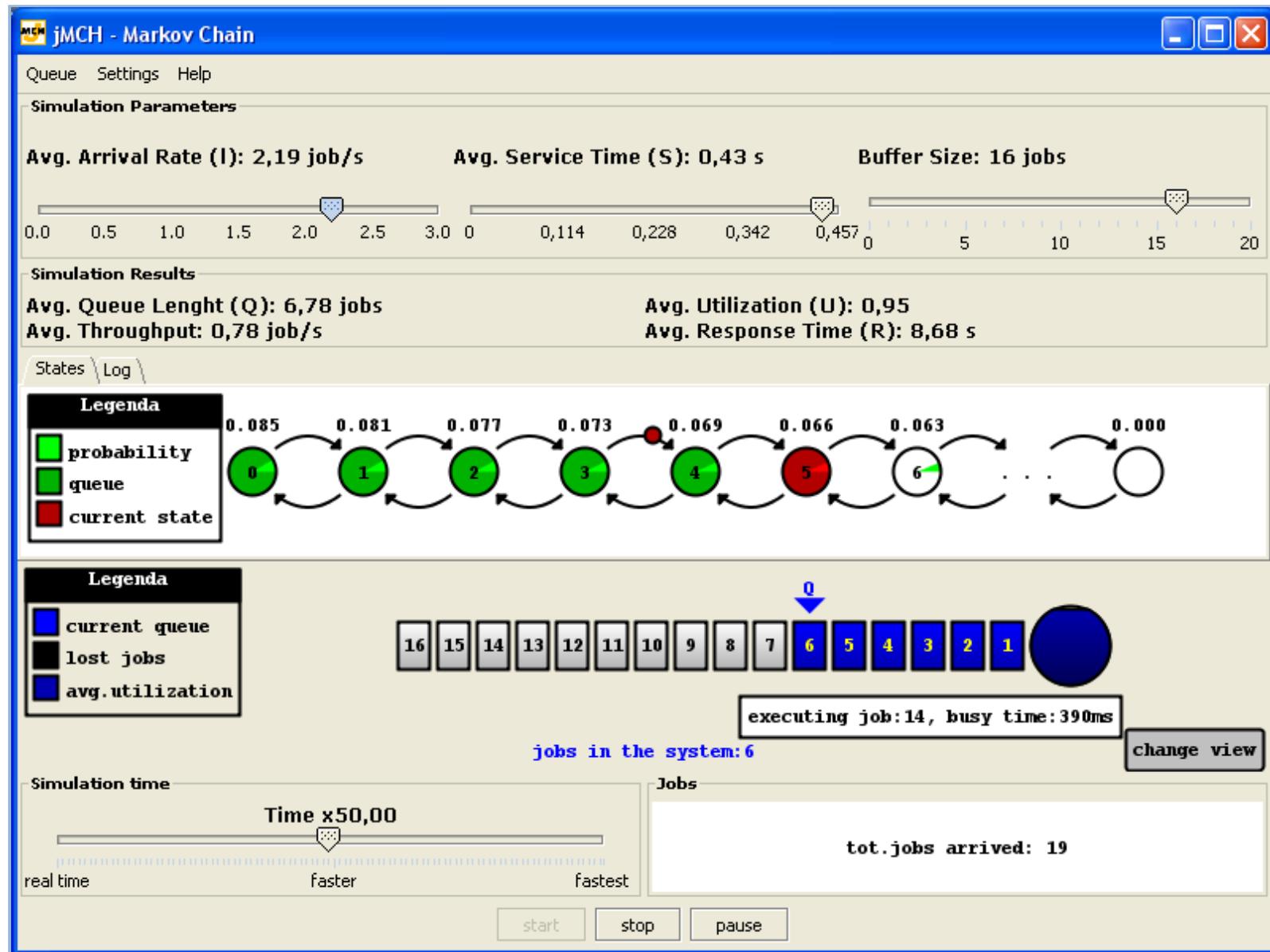


JABA: bottleneck identification

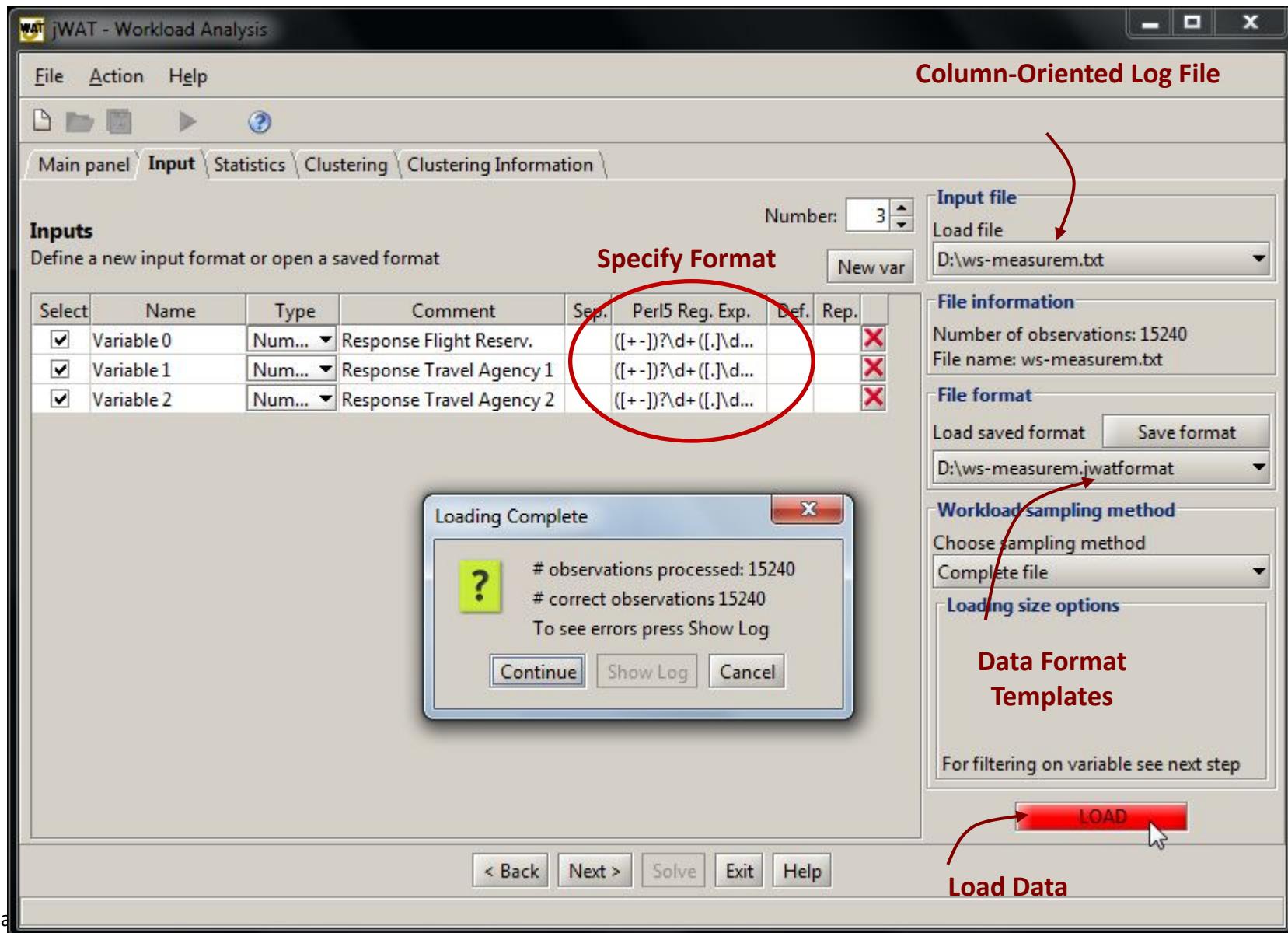
- 3-class model



JMCH: Markov chain animation



JWAT: workload characterization



JWAT: workload characterization

jWAT - Workload Analysis

File Action Help

Main panel Input Statistics Clustering Clustering Information

Univariate Sample Construction Bivariate QQ-plot Matrix Scatter plot Matrix

Scatter plots

Univariate statistics
This panel shows univariate statistics and graphs

$c = \text{std. dev.} / \text{mean}$

Variable	Mean	Variance	Std. Dev.	Coeff. of var.	Minimum	Maximum	Range	Median	Kurtosis
Variable 0	966.580E0	923.198E4	303.842E1	314.347E-2	200.000E-2	302.900E2	302.880E2	328.000E0	426.838E-1
Variable 1	215.074E1	131.959E5	363.262E1	168.900E-2	300.000E-2	653.990E2	653.960E2	143.300E1	637.504E-1
Variable 2	911.325E0	698.850E4	264.358E1	290.081E-2	100.000E-2	250.140E2	250.130E2	394.000E0	576.731E-1

Variable 1 frequencies graph

Histogram

Variables

Transformations

Frequency

Variable 1

$x10^3$

$x10^3$

Hyper-Exp (c > 1)

Graphs

Frequencies QQ-Plot

$x10^3$

$x10^4$

< Back Next > Solve Exit Help

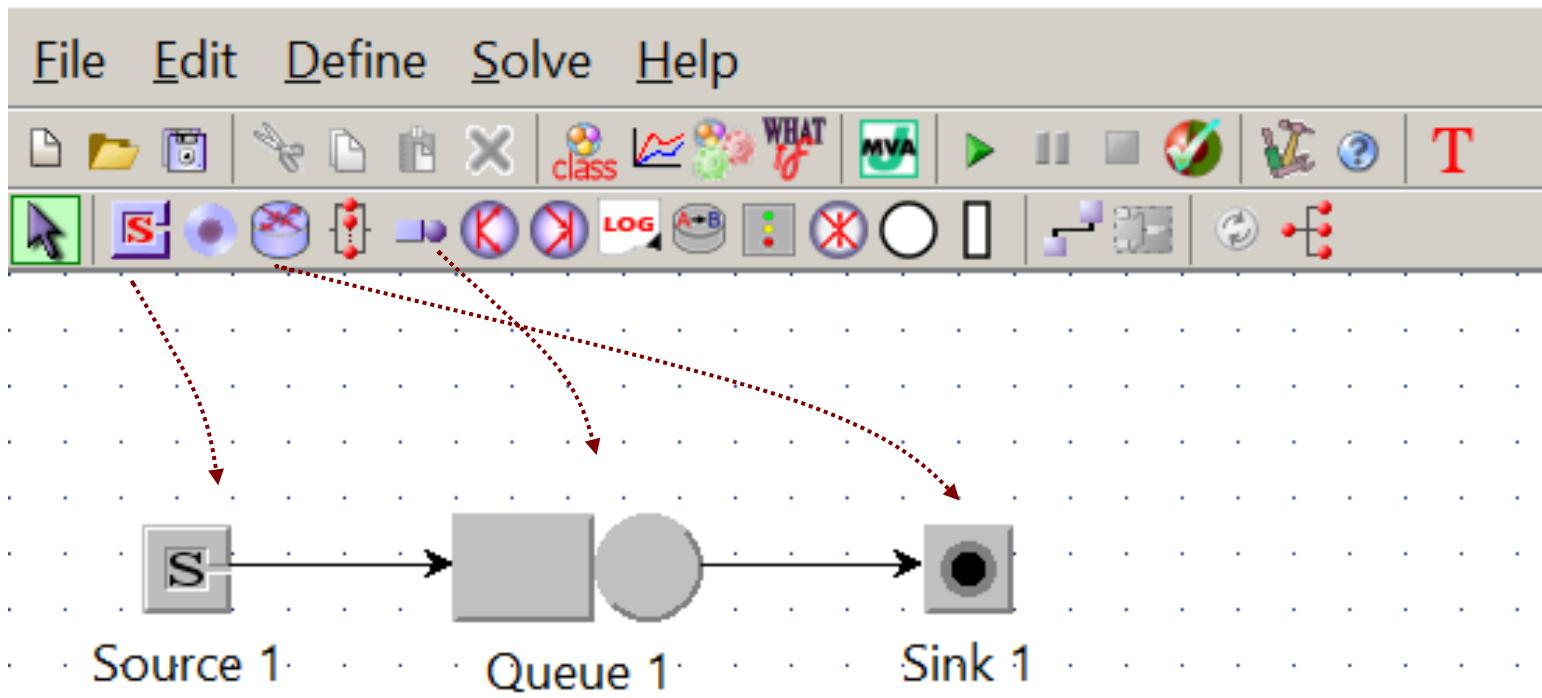
Activity 1: getting started

Hands-on activity: M/M/1

- Arrival rate: $\lambda=0.5$ job/s (Exponential)
- Service rate: $\mu=1.00$ job/s (Exponential)
- Goal: verify $\mu \Rightarrow \lambda$ property



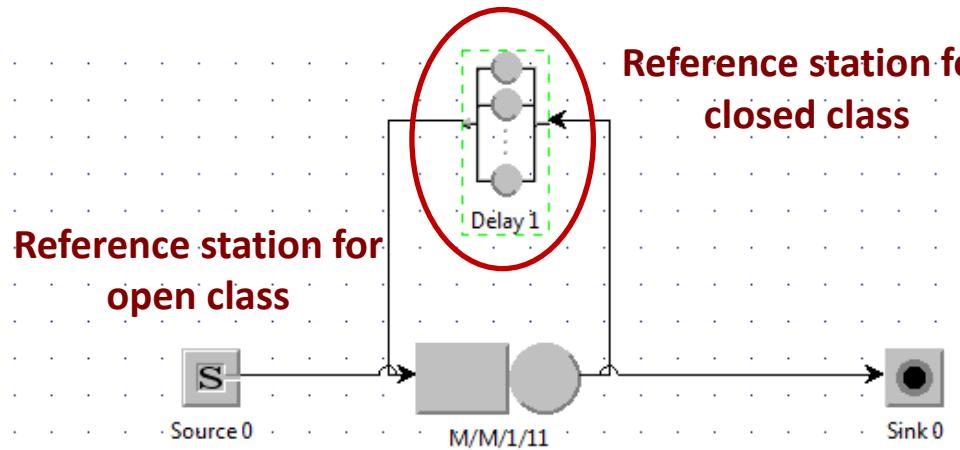
JSIMgraph - Graphical Queueing Network and Petri Net Simulator



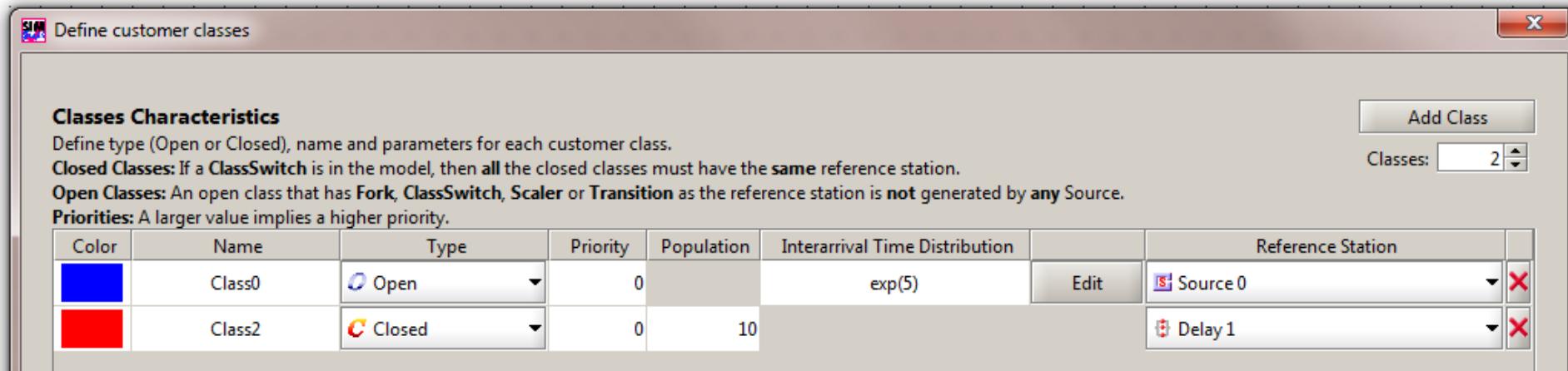
Class definition



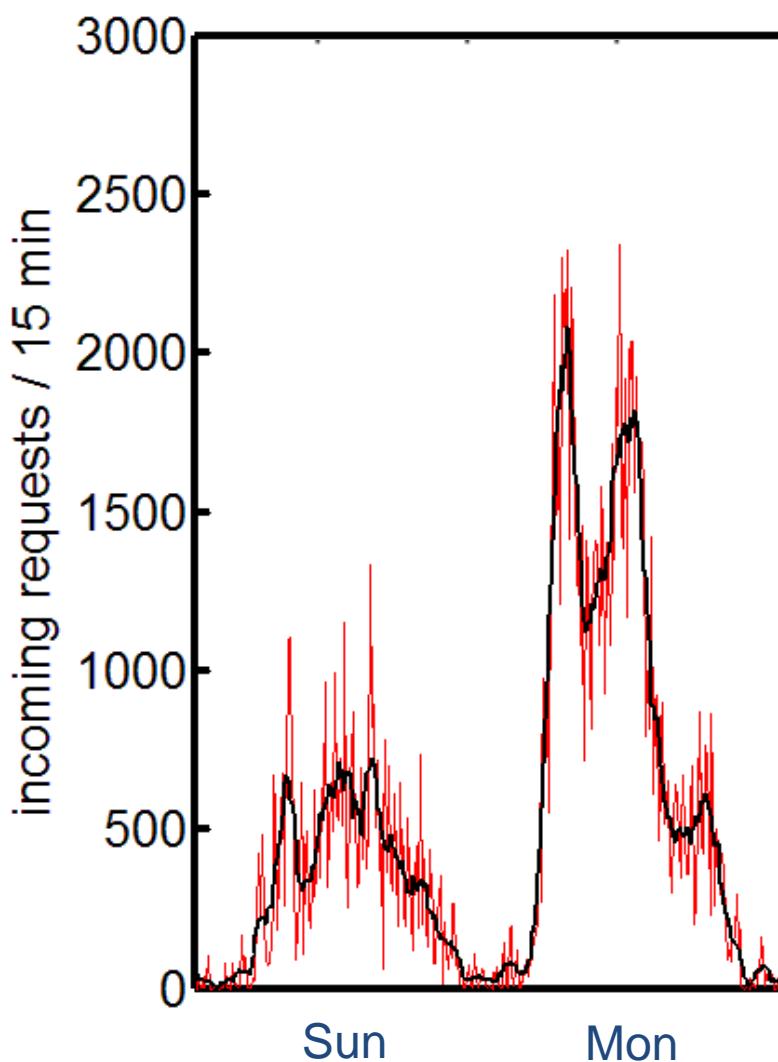
- Open, closed, and mixed workloads
- Priorities and reference stations



- Reference stations used for system metrics
- Visits at node = $\text{TPUT}_{\text{node}}/\text{TPUT}_{\text{ref}}$
- ref is the reference station



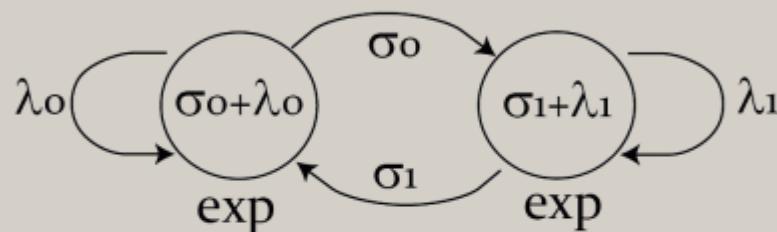
Arrival distribution



 Editing Class1 Service Time Distribution... X

Selected Distribution: Burst (MMPP2)

Markov-Modulated Poisson Process:
[mmpp2($\lambda_0, \lambda_1, \sigma_0, \sigma_1$)]



$\lambda_0:$ 12

$\lambda_1:$ 0.0872

$\sigma_0:$ 0.0098

$\sigma_1:$ 0.0008

OK

Cancel

Queue section

- Non-preemptive scheduling: FCFS, LCFS, RAND, SJF, LJF, SEPT, LEPT, HOL (FCFS priority)
- Preemptive scheduling: PS, GPS, DPS

 Editing Queue 1 Properties... X

Station Name

Station Name: Queue 1

Queue 1 Parameters Definition

Queue Section Service Section Routing Section

Capacity

Infinite

Finite

Buffer size

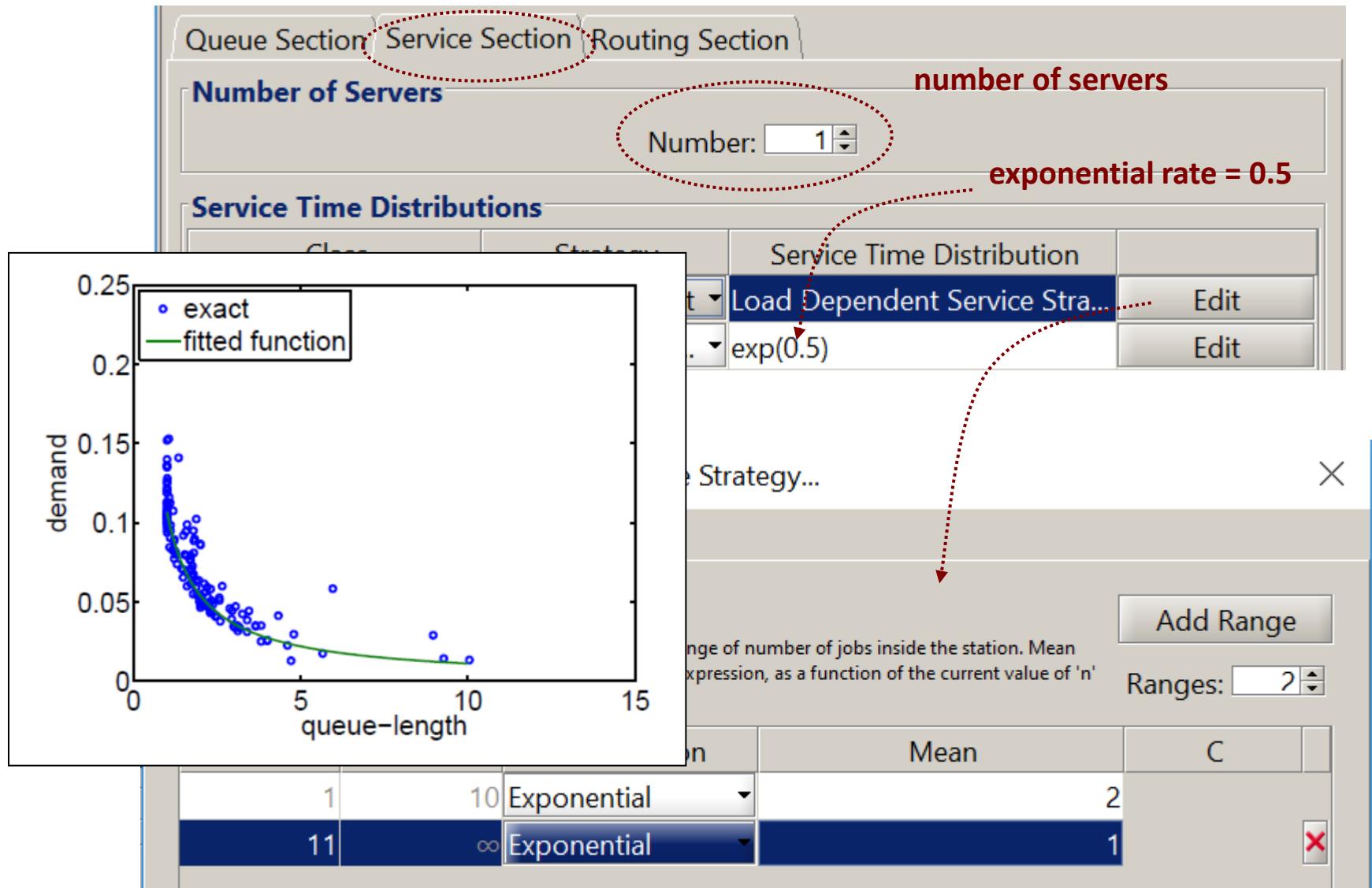
Max no. customers (queue+service)

Queue Policy

Station queue policy: Preemptive Scheduling

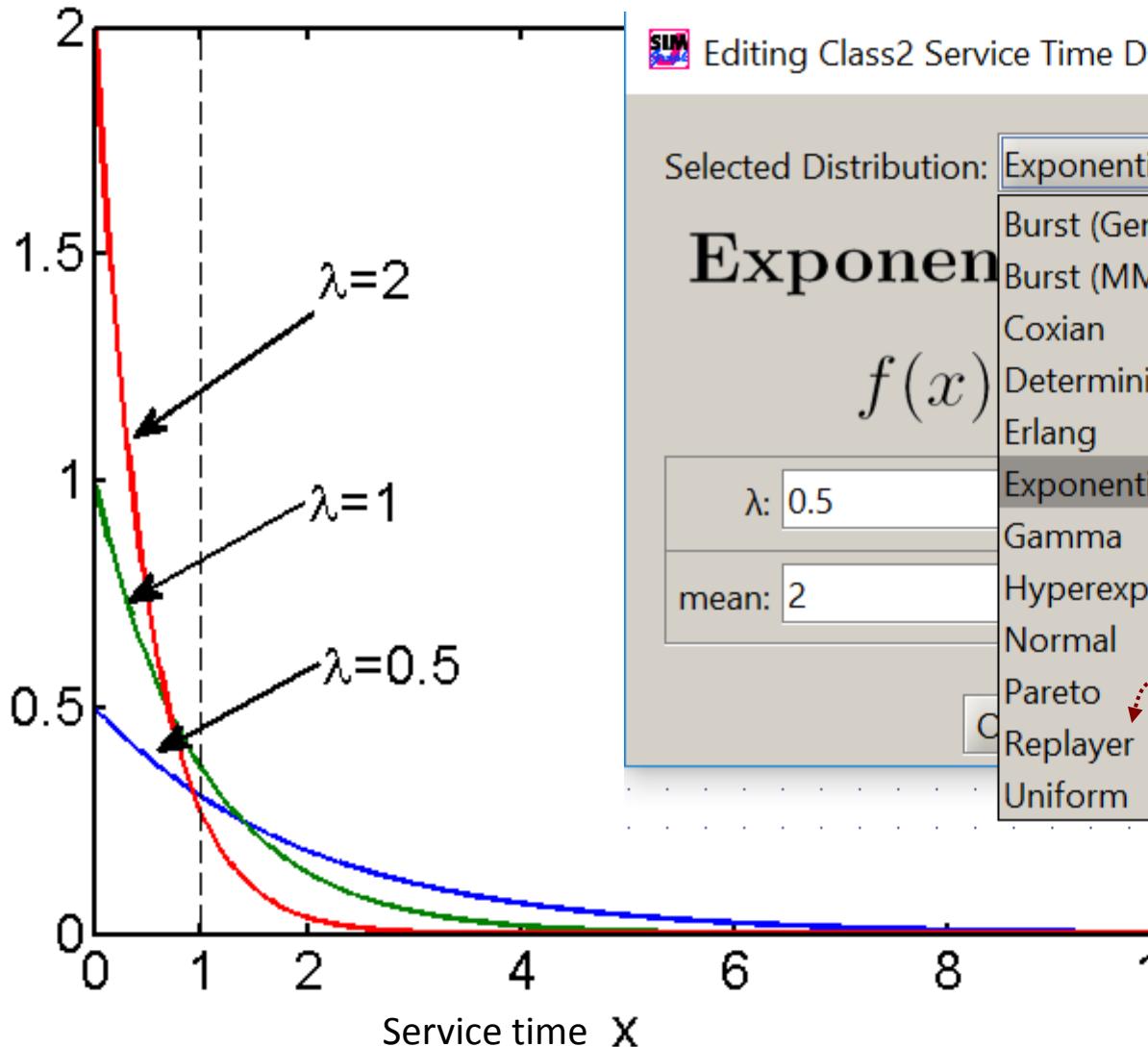
Class	Queue Policy	Drop Rule	Service Weight
Class1	PS	Infinite Capacity	--
Class2	PS	Infinite Capacity	--

Service section



Service time distribution

probability density function: $f(x) = \lambda \exp(-\lambda x)$



SIM Editing Class2 Service Time Distribution... X

Selected Distribution: Exponential

Exponen

$f(x)$

$\lambda: 0.5$

mean: 2

External trace

Burst (General)
Burst (MMPP2)
Coxian
Deterministic
Erlang
Exponential
Gamma
Hyperexponential
Normal
Pareto
Replayer
Uniform

Perf. Indices



- 19 types of performance indices
 - Utilization, residence time, response time
 - Throughput, firing rates, drop rates, ...
- Granularity: system, station, class, mode, sink

Define performance indices X

Performance Indices
Define performance indices to be collected and plotted by the simulation engine. ---Select an index---

Performance Index	Class/Mode	Station/Region	Stat.Res.	Conf.Int.	Max Rel.Err.	
Utilization	--- All Classes ...	--- Queue 1	<input type="checkbox"/>	0.99	0.03	
	--- All Classes ---					
	Class1					
	Class2					

Sim. Results



Simulation Results...

Utilization \ System Response Time \ System Throughput \ System Number of Customers \ System Power

System Number of Customers

Average customer number for each chosen class.

performance indices

confidence interval

Station Name:

-- Netwo

Class Name:

-- All

Conf.Int/Max Rel.Err:

0.95 / 0.

Analyzed samples:

15000

Min:

8.3130

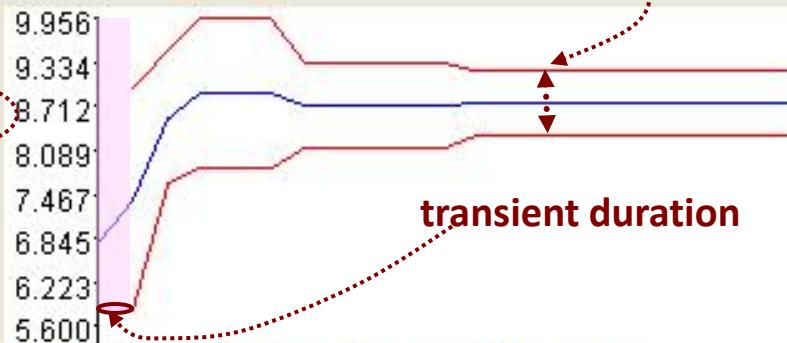
Max:

9.200

Average value:

8.7567

Abort Measure

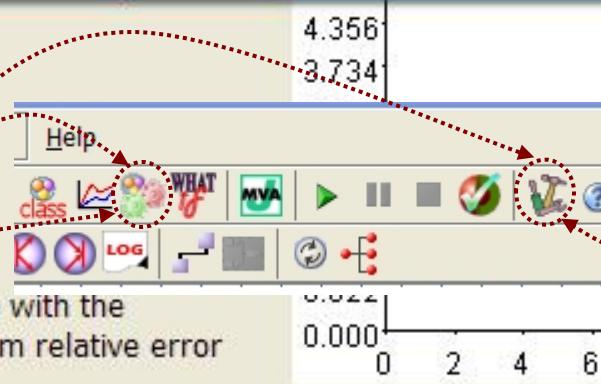


 Simulator failed to compute this measure with the specified confidence interval and maximum relative error

the number of samples needed is greater than the max allowed

actual sim. parameters

Simulator failed to compute this measure with the specified confidence interval and maximum relative error



default values of parameters

Statistical analysis

- Automated (overridable) simulation stop

$1 - \alpha \leq P[\text{relative error}] \leq \frac{\varepsilon}{1 - \varepsilon}$

confidence level maximum relative error

The image shows a screenshot of the JSIMGraph software interface. At the top, there is a menu bar with 'File', 'Edit', 'Define', 'Solve', and 'Help'. Below the menu is a toolbar with icons for 'Customer classes...', 'Performance indices...', 'Simulation parameters...', 'What-if analysis...', and 'Default parameters...'. The 'Default parameters...' icon is highlighted with a red circle and has a red arrow pointing to it from the text 'Define default value of model's parameters' at the bottom of the window. The main window is titled 'Editing Default Values...' and contains a 'Simulation parameters' section. This section includes checkboxes for 'Infinite' and 'Random', and dropdowns for 'Confidence Interval Measure (0-1)' set to 0.99, 'Max Relative Error Measure (0-1)' set to 0.03, 'Simulation seed' set to 23,000, 'Maximum duration (sec)' set to 5 with 'Infinite' checked, and 'Maximum number of samples' set to 1,000,000. A red circle highlights the 'Max Relative Error Measure (0-1)' field, and a red arrow points from this circle to the formula 'maximum relative error' in the text above. Another red circle highlights the 'Maximum duration (sec)' field, and a red arrow points from this circle to the formula 'confidence level' in the text above. The bottom right corner of the window has the number '30'.

30

JSIMGraph - Advanced queueing network design tool

File Edit Define Solve Help

Customer classes... Performance indices... Simulation parameters... What-if analysis... Default parameters...

Define default value of model's parameters

Editing Default Values...

30

Simulation parameters

Confidence Interval Measure (0-1): 0.99

Max Relative Error Measure (0-1): 0.03

Simulation seed: 23,000 Random

Maximum duration (sec): 5 Infinite

Maximum number of samples: 1,000,000

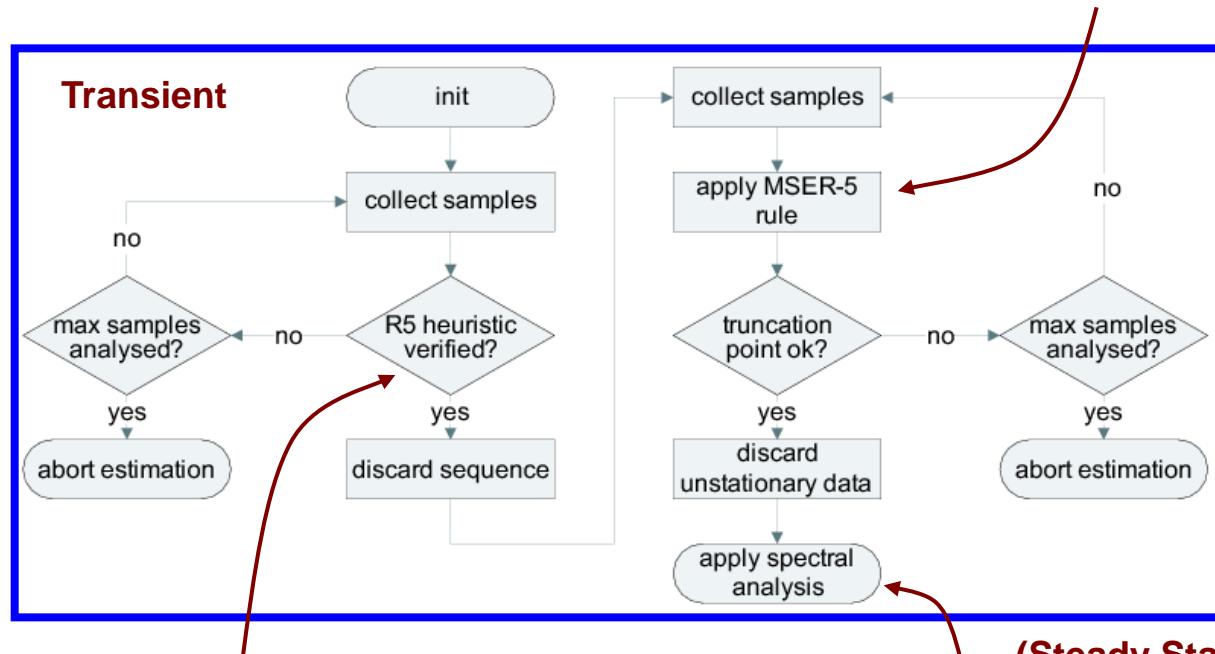
confidence level maximum relative error

traditional control parameters

Transient filtering

- Intelligent filtering of simulation data
 - R5 heuristics, spectral analysis, MSER-5 rule, ...

[Spratt, M.S. Thesis, 1998]

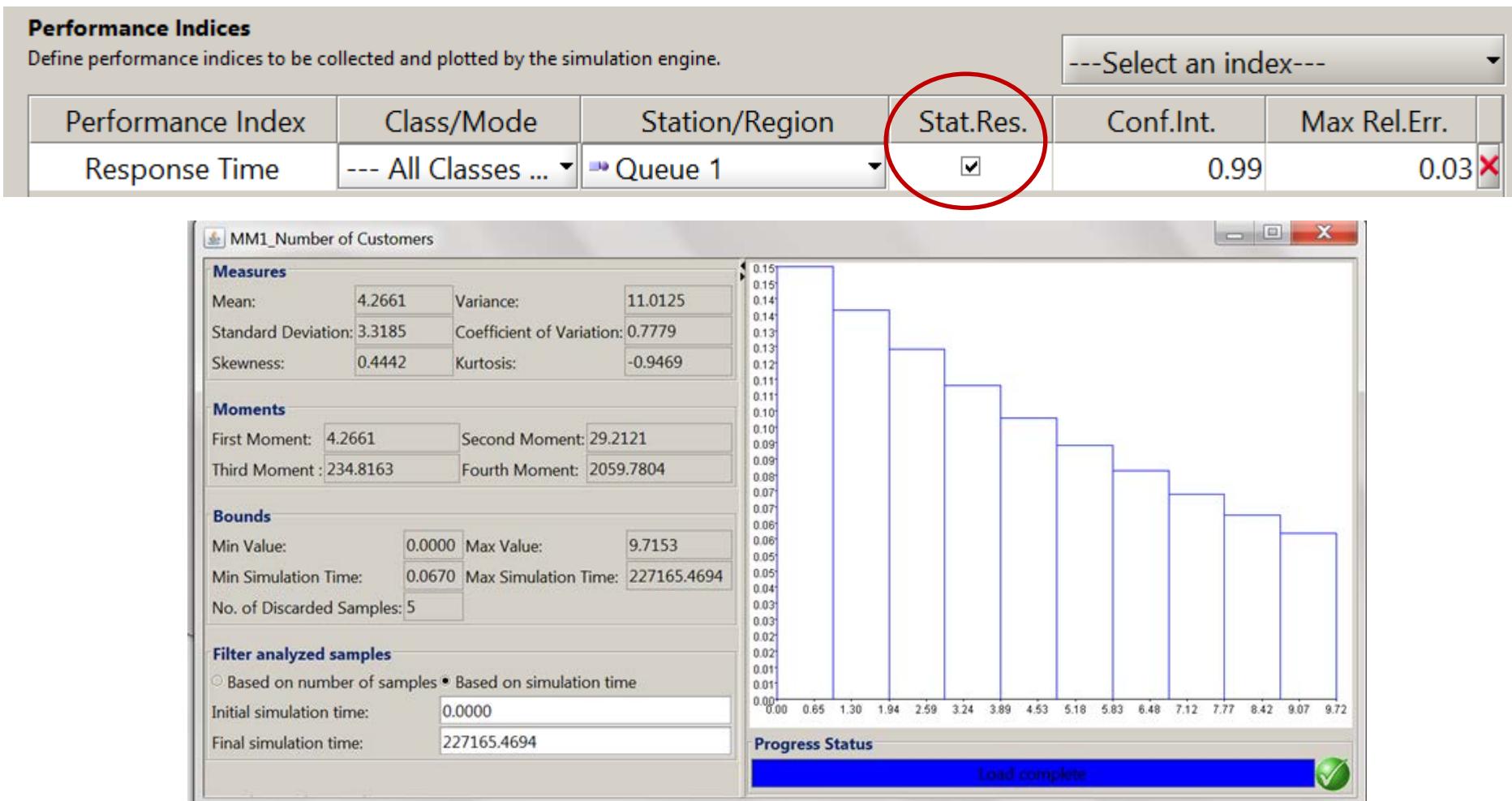


[Pawlakowski, CSUR, 1990]

[Heidelberger&Welch, CACM, 1981]

Detailed statistical results

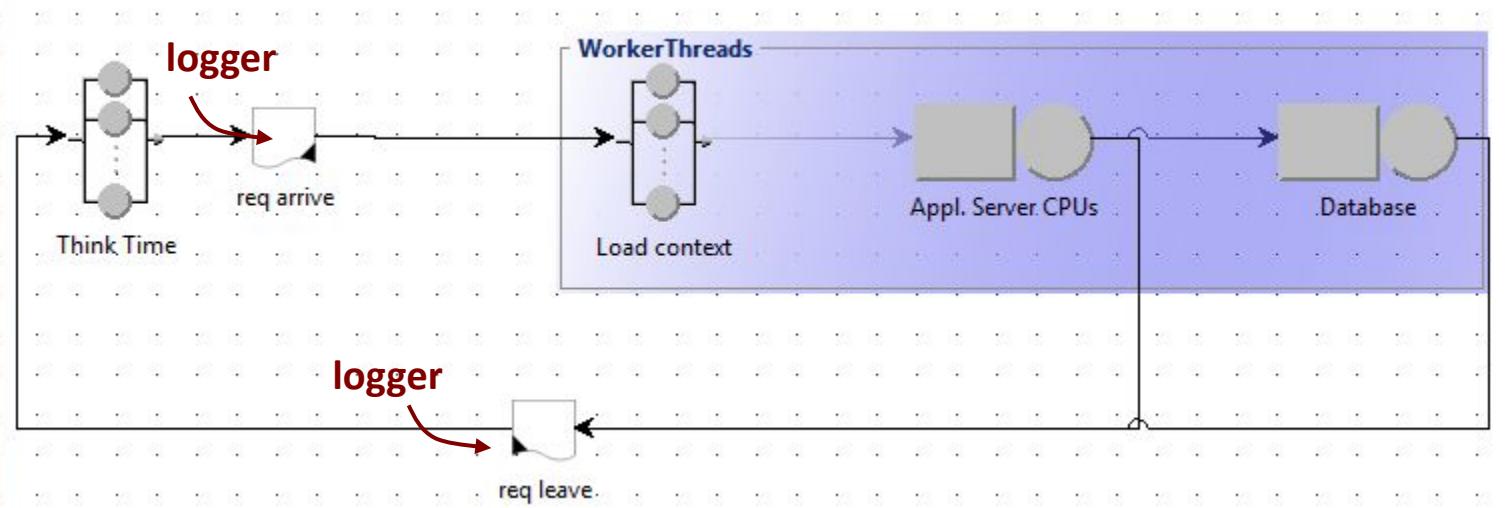
- Response time percentiles, buffer overflow probability, departure process moments, ...



Loggers



- Simulation events can be traced in CSV files



global.csv

job id (same throughout simulation)

LOGGERNAME;TIMESTAMP;JOB_ID;CLASS_ID;INTERARRIVAL_SAMECLASS;INTERARRIVAL_ANYCLASS;SIMUL_START_TIME
req arrive;0.009420010041266342;253625;Transactional workload;;
req leave;0.0217557654334812;253625;Transactional workload;;
req arrive;0.031032734664243056;253498;Transactional workload;;
req leave;0.04915995332909814;253498;Transactional workload;;
req arrive;0.07727161520772474;253542;Transactional workload;;

job class

Activity 2: Load balancing

Routing section or

- Probabilistic routing
- State-dependent routing: JSQ, SRT, LU, FS
- Load-dependent probabilistic routing

Queue Section | Service Section | Routing Section

Routing Strategies	
Class	Routing Strategy
Class1	Random
Class2	Random

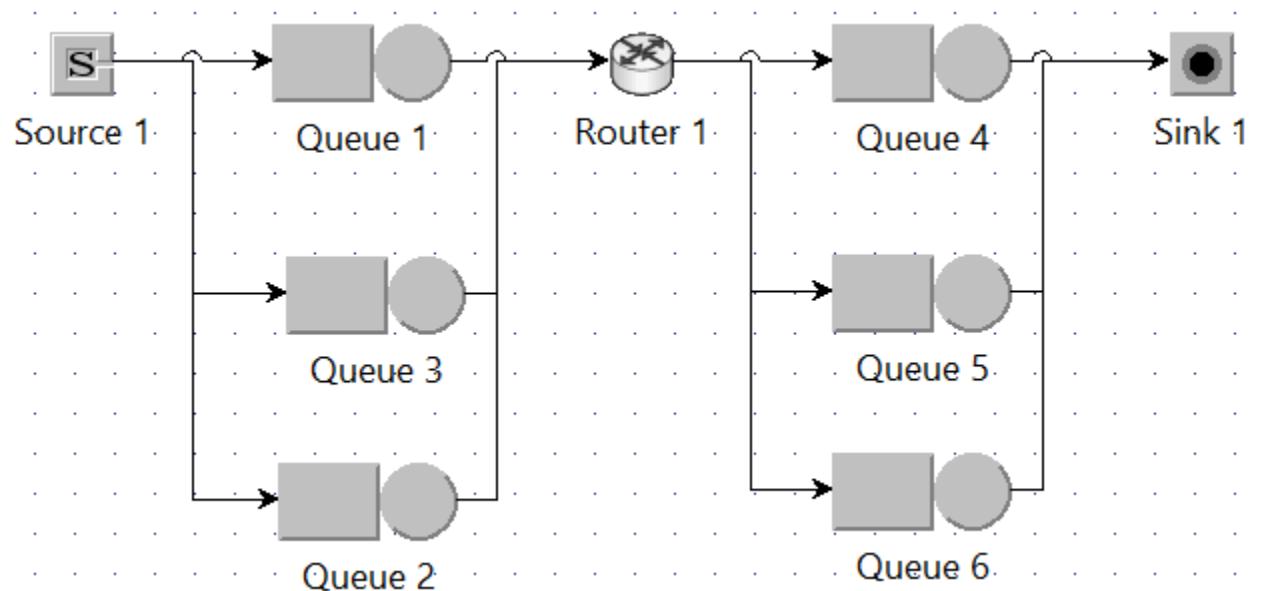
Description
Jobs are routed randomly to stations connected to the current one. All routes have the same probability to be selected.

Routing Options
No options available for this routing strategy

... router node

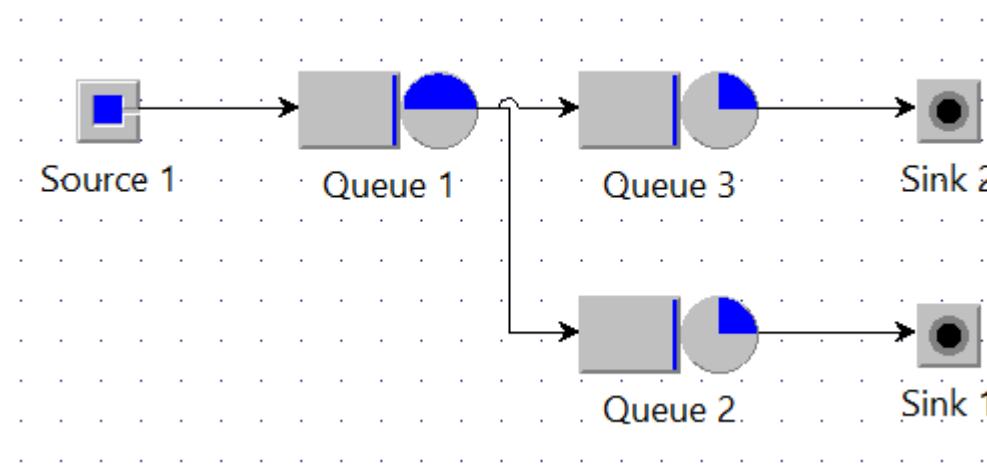


- Router node also allows to specify routing
 - Applies policy across multiple input queues
 - Same policies as routing section



Hands-on activity: load balancing

- We add two queues to the M/M/1 model.
- Goal: compare *round-robin* and *probabilistic* load-balancing



Blocking after service

Editing Server Properties...

Station Name
Station Name: Server

Server Parameters Definition

Queue Section \ Service Section \ Routing Section

Capacity

infinite

finite

max num.customers: 5

Queue Policy

Station queue policy: Non-preemptive Scheduling

Class	Queue Policy	Drop Rule
Class0	FCFS	BAS blocking

selection of the BAS policy

BAS policy:
requests are blocked in the sender station when the max capacity of the receiver is reached

station with finite capacity

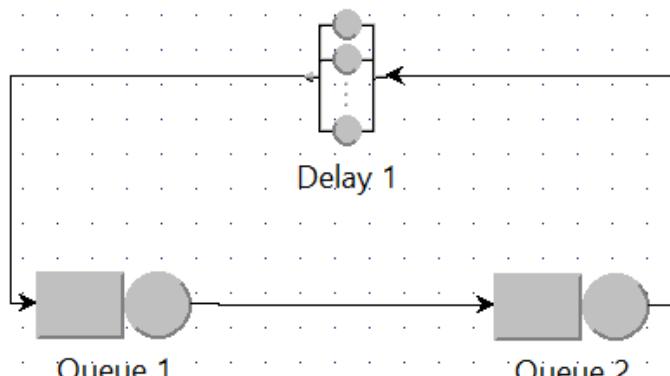
max number of requests in the station

38

Activity 3: Parameter sweeping

Hands-on activity: bottleneck switch

- Analysis of bottleneck switch
- Measure: *Number of Customers*
- Demands: *Queue 1: 10 , 5; Queue 2: 5 , 9*



Define What-if analysis parameters

What-if Analysis
Define the type of What-If analysis to be performed and modify parameter options.

WARNING:
Enabling What-If analysis will disable all statistical outputs.

Parameter selection for the control of repeated executions

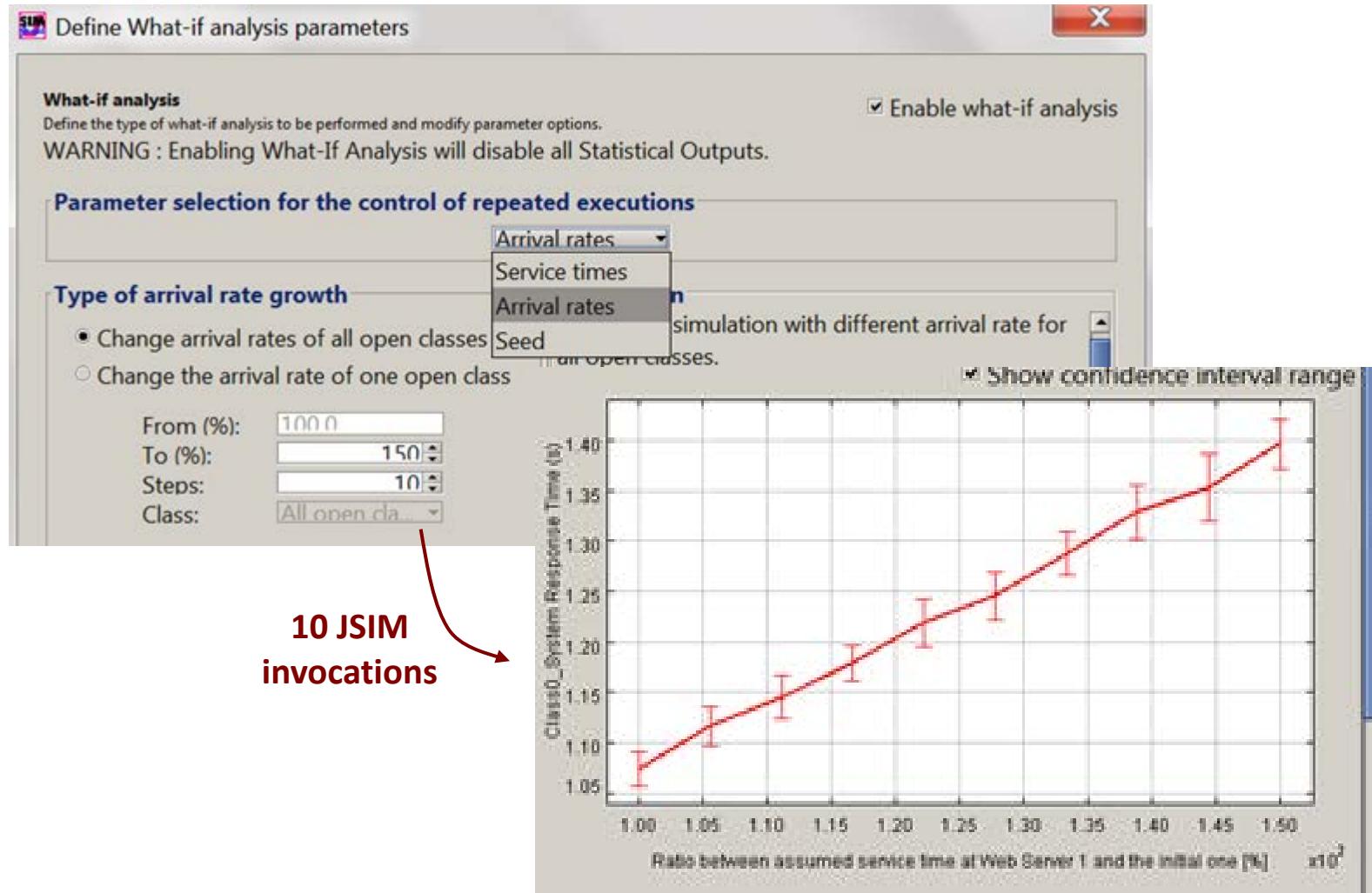
Population mix

Type of population mix	Description
Initial β : <input type="text" value="0"/>	This type of analysis is available for other open classes) and it applies
Final β : <input type="text" value="1"/>	
Steps (n. o...): <input type="text" value="11"/>	Repeat the simulation changing
Class: <input type="text" value="Class1"/>	keeping constant the total num

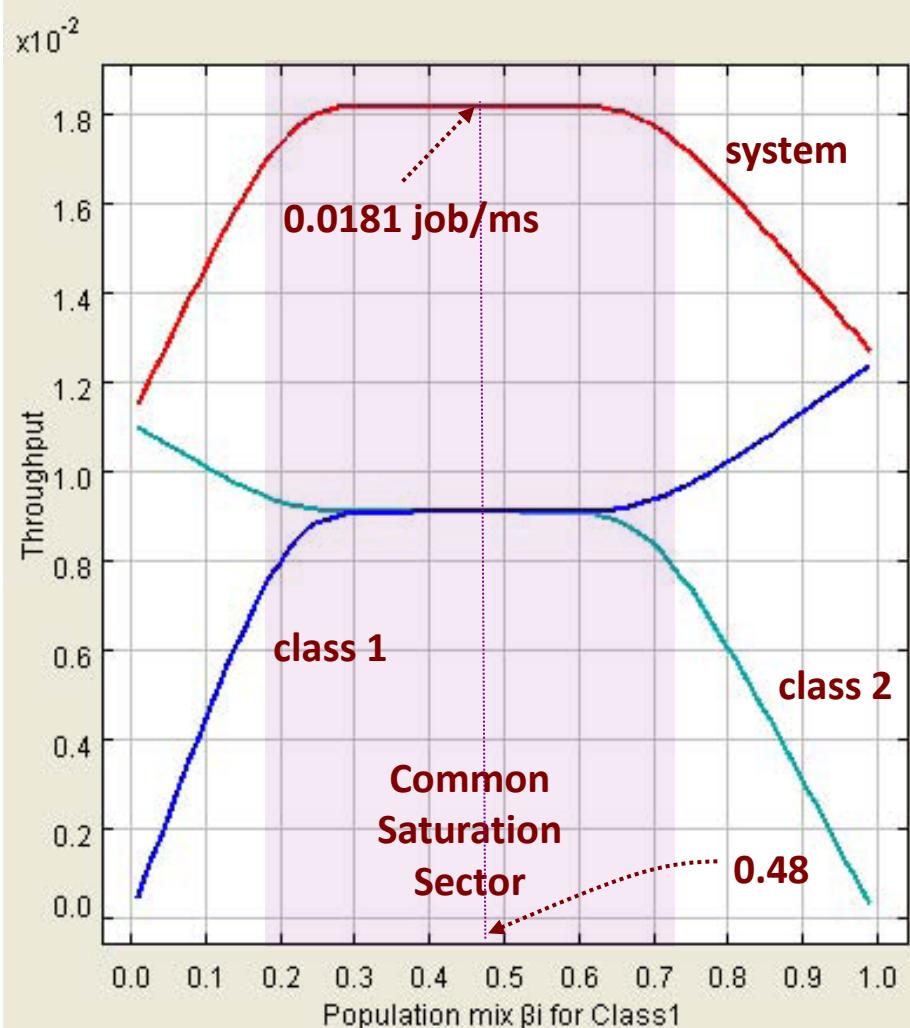
What-If analysis



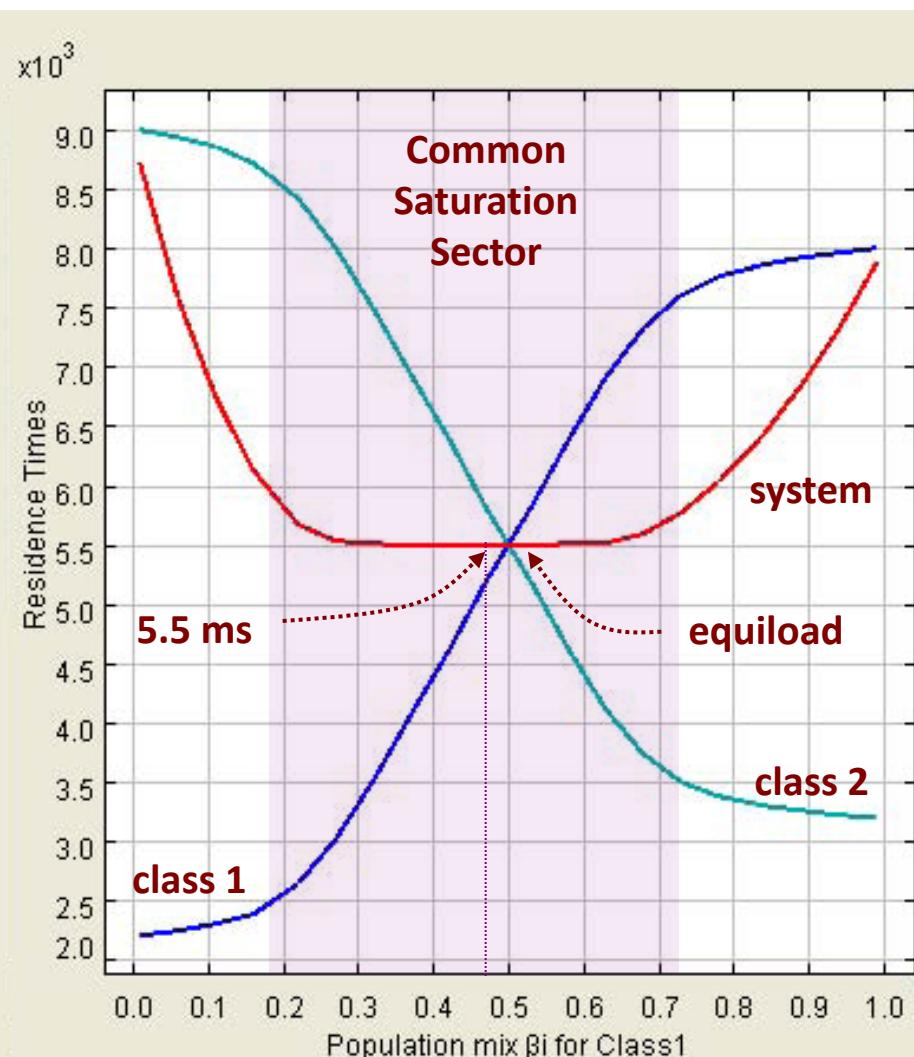
- Perform repeated executions automatically



JMVA: What-If



Throughput



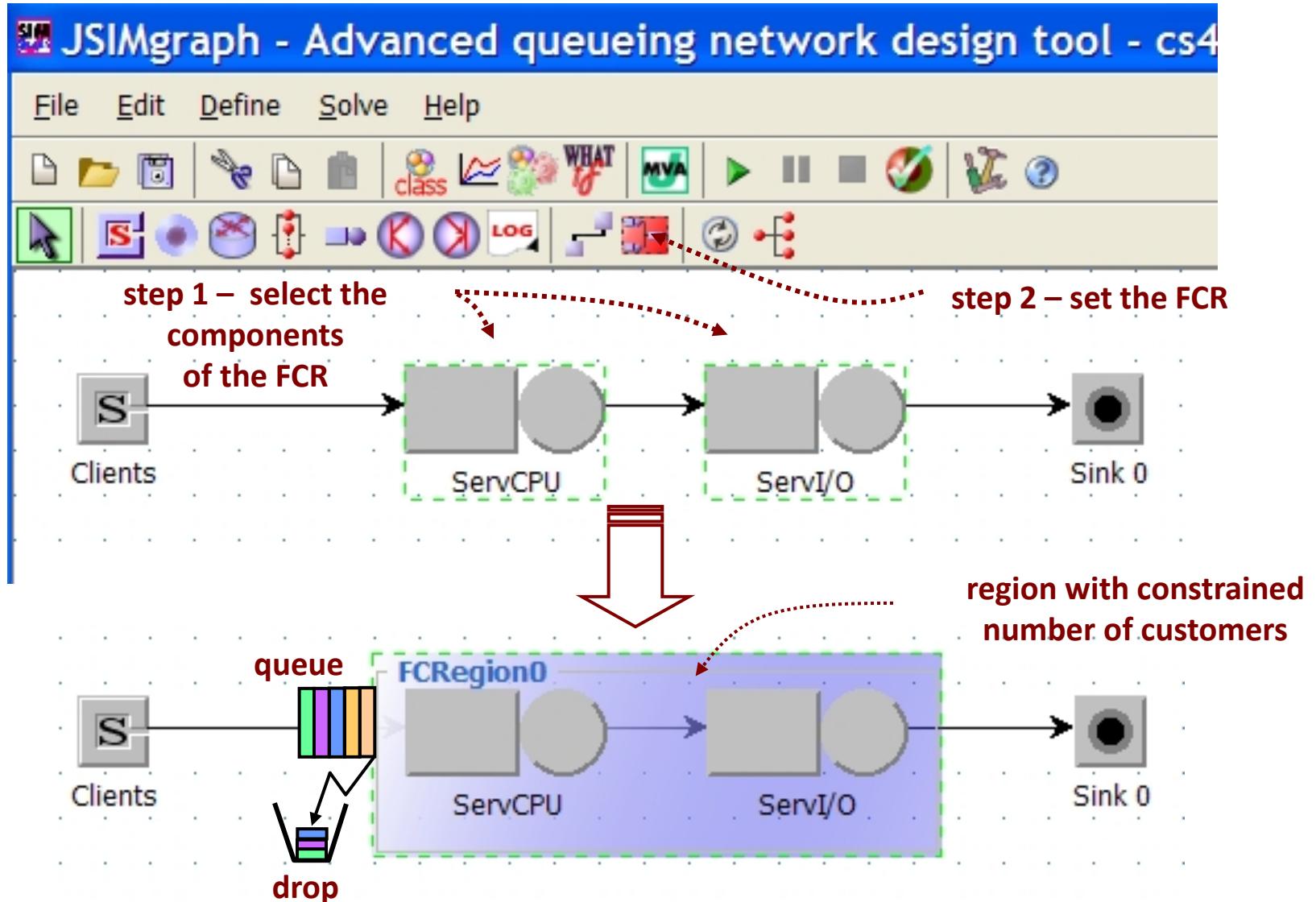
Response times

Activity 4: Capacity constraints

FCR definition



- Thread limits via *finite capacity regions* (FCRs)



FCR parameters

- Capacity constraints: total, per-class, per-group
- Memory constraints: jobs have sizes and must fit

SIM Editing FCRegion1 Properties...

Global Properties of FCRegion1

Region Name:	FCRegion1	Global job capacity
Region Capacity:		10 <input type="button" value=""/>
Region Memory Size:		15 <input type="button" value=""/>
Region Groups:		<input type="checkbox"/> Enable

Specific Properties of FCRegion1

Memory Capacity **Job Size**

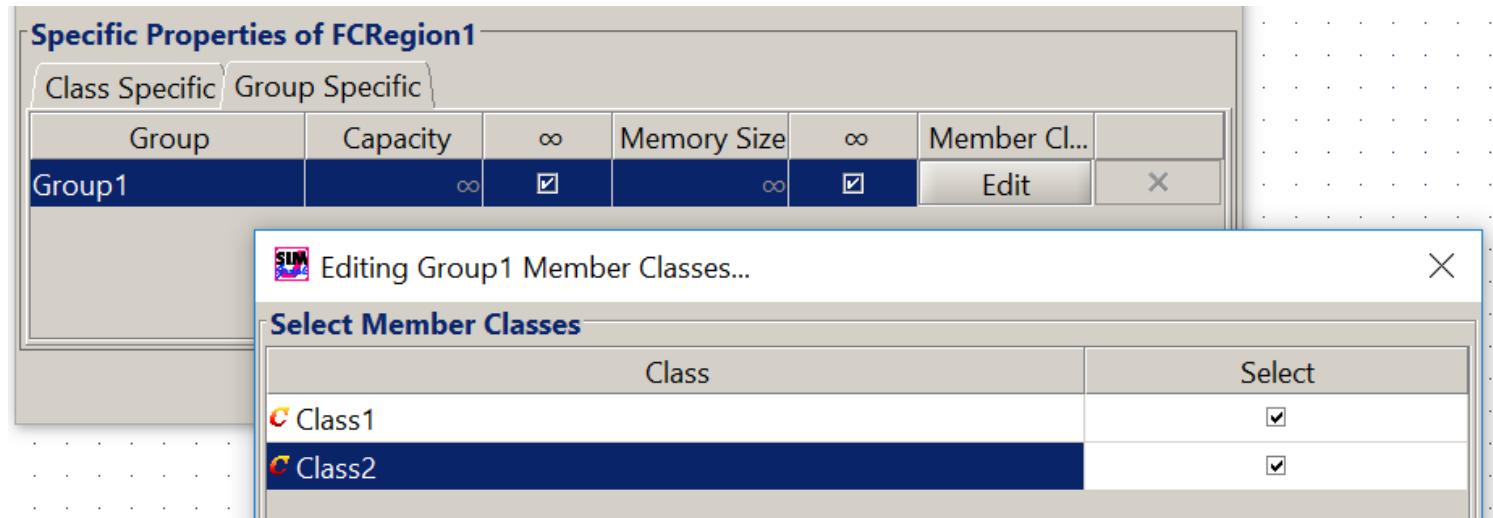
Class	Capacity	∞	Memory Size	∞	Drop	Weight	Size
Class1	2	<input type="checkbox"/>	3	<input type="checkbox"/>	true <input type="button" value=""/>	1	1
Class2	4	<input type="checkbox"/>	5	<input type="checkbox"/>	true <input type="button" value=""/>	1	2
Class3	6	<input type="checkbox"/>	7	<input type="checkbox"/>	true <input type="button" value=""/>	1	1

max number of requests per class in the FCR

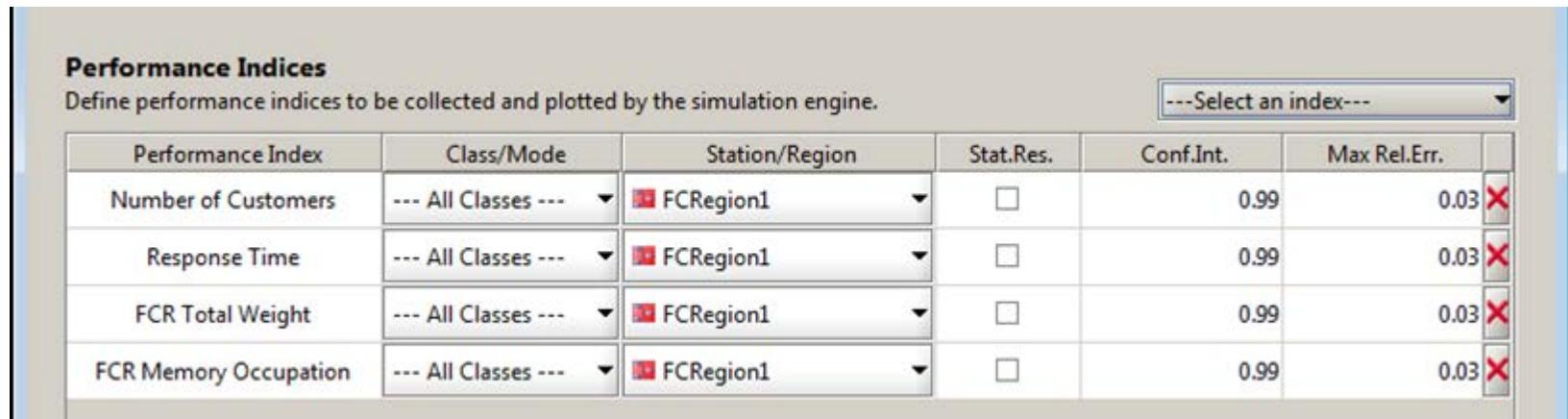
drop the requests when the region capacity is saturated

FCR groups and indices

- Group-specific constraints (*i.e.*, for subset of classes)

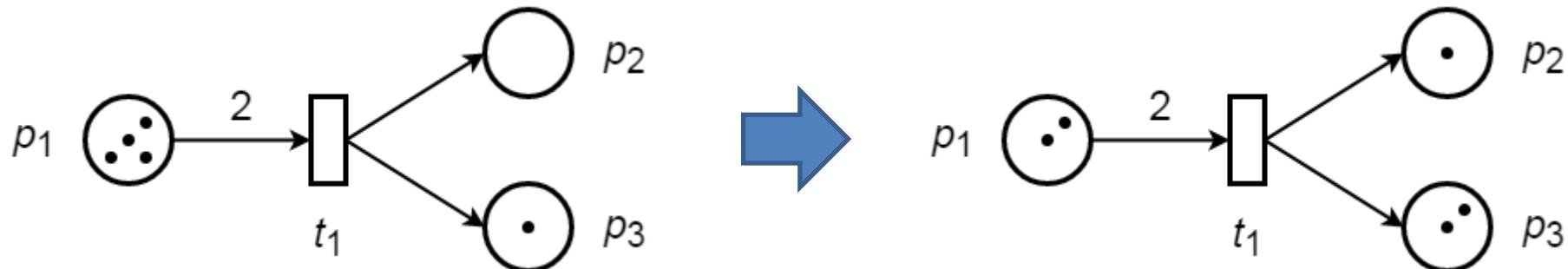


- Dedicated performance indices

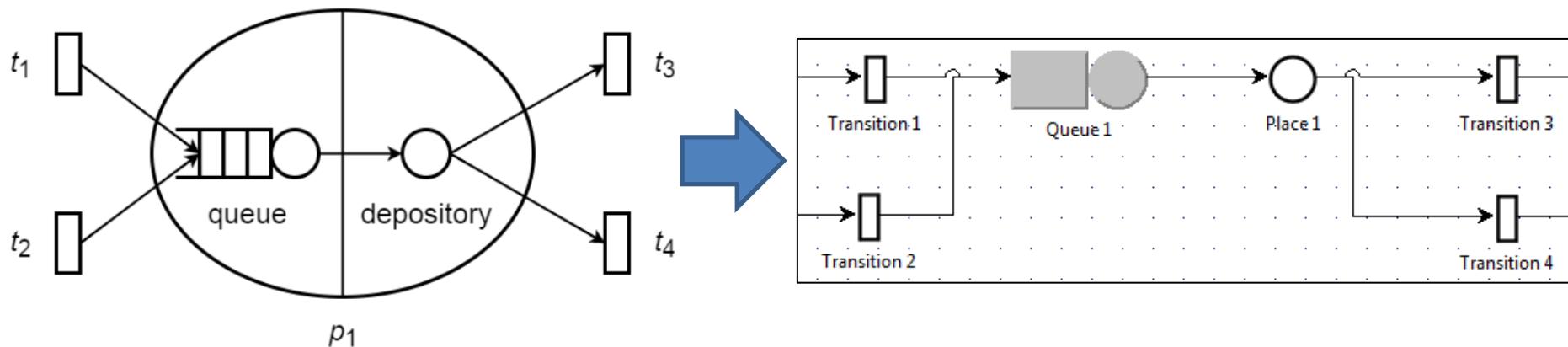


Support for PN elements

- Places and transitions



- Queueing Petri nets



PN sections & modes

- JMT design paradigm extends to PN elements
- Mode: a rule to activate and fire a transition



Place Station

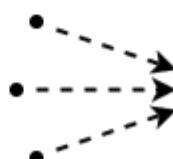


Store Section

Tunnel Section

Link Section

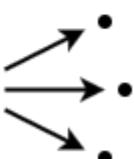
Transition Station



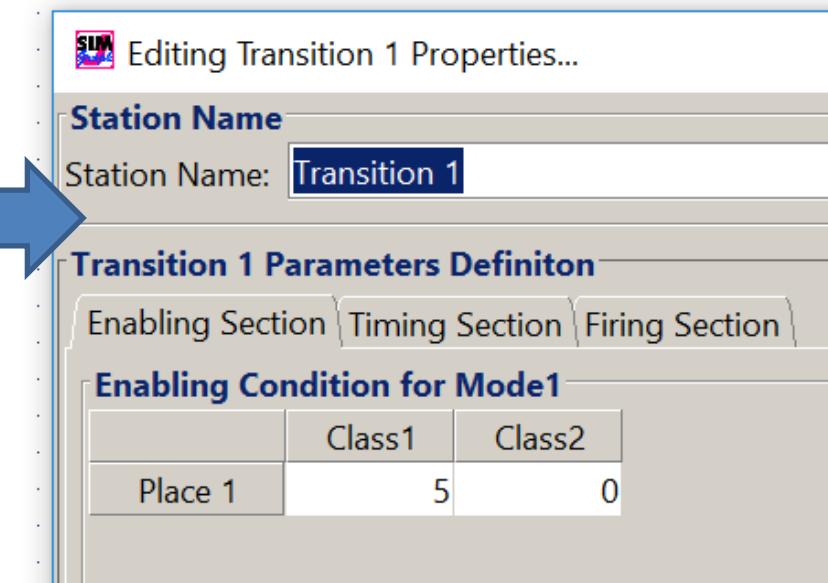
Enabling Section



Timing Section

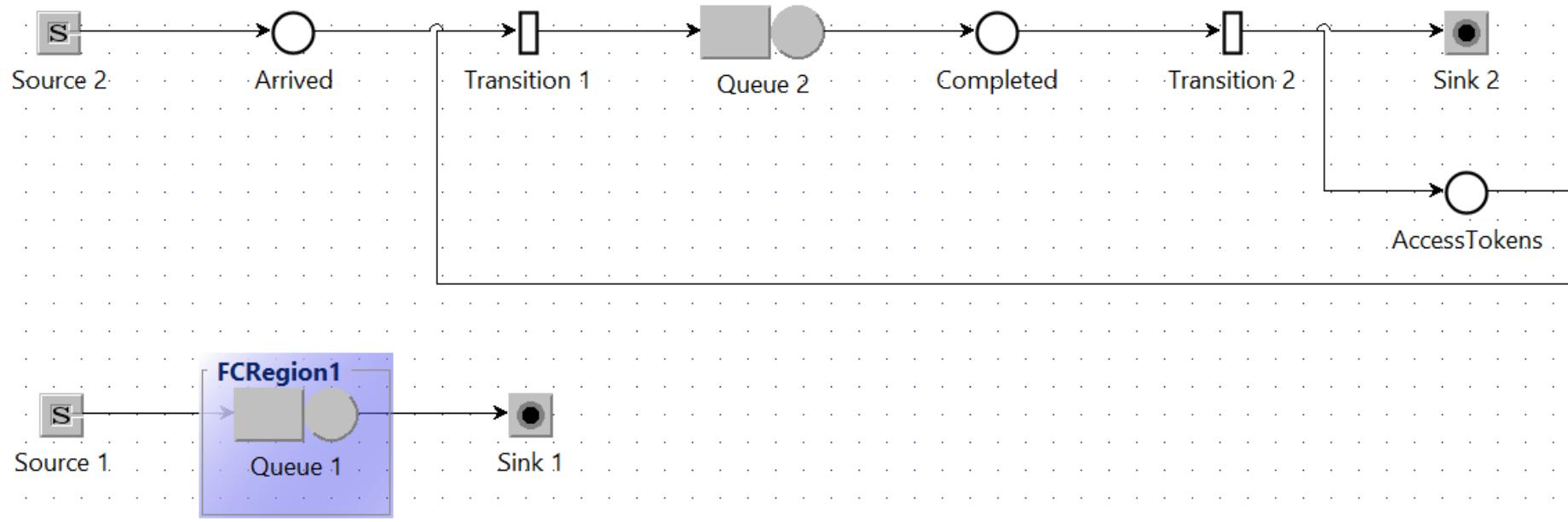


Firing Section



Hand-on activity: FCRs vs QPNs

- Arrival rate: $\lambda=0.99$ job/s
- Service rate: $\mu=1.00$ job/s
- Goal: restrict max 1 job inside queue

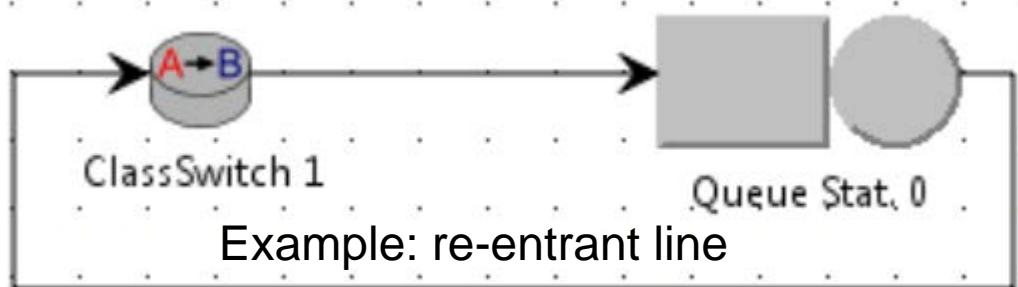


Activity 5: Workflows & fork-join

Class-switching



- A job can change its class during the simulation
 - Workflows, re-entrant lines, track path-wise perf., ...



Editing ClassSwitch 1 Properties...

Station Name

Station Name: ClassSwitch 1

ClassSwitch 1 Parameters Definition

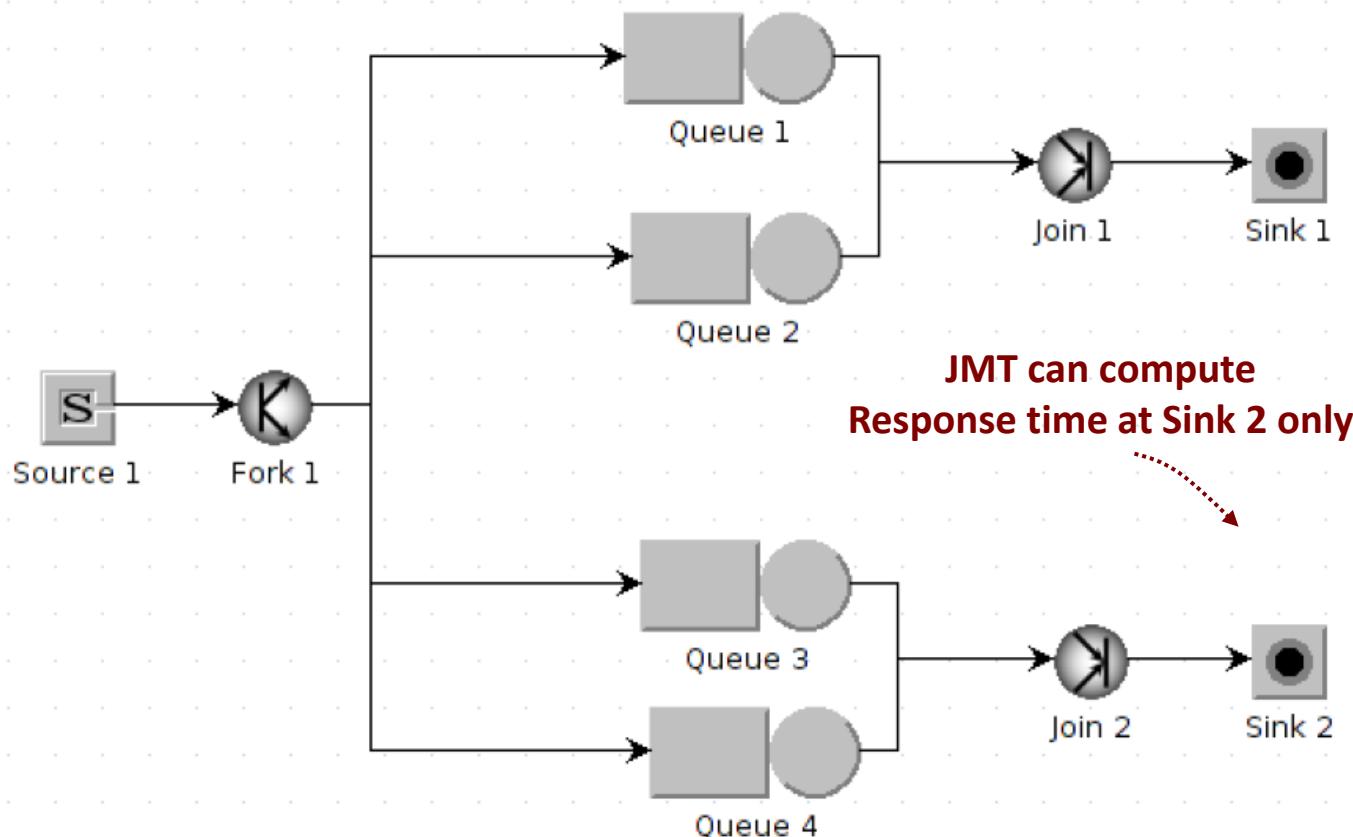
Class Switch Matrix Routing Section

CS Strategies

*	Class0	Class1
Class0	0.6 (60%)	0.4 (40%)
Class1	0.8 (80%)	0.2 (20%)

Fork-Join elements

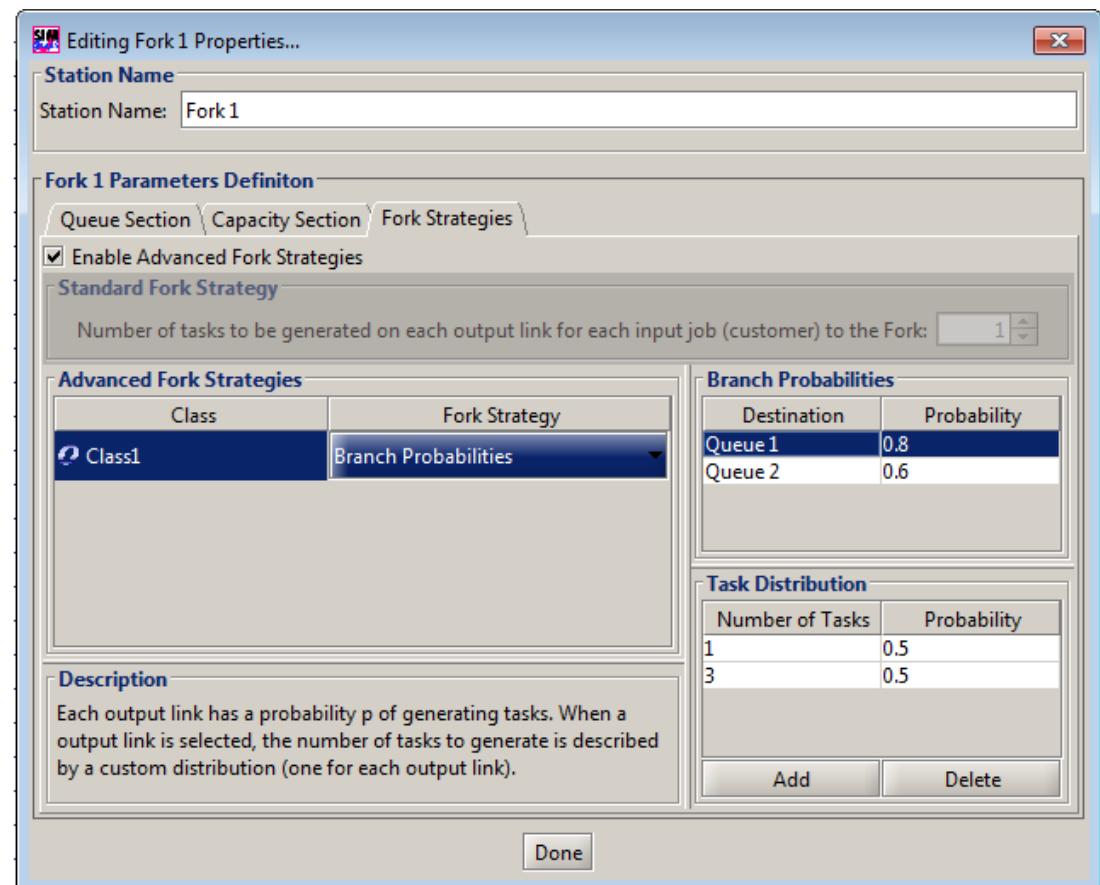
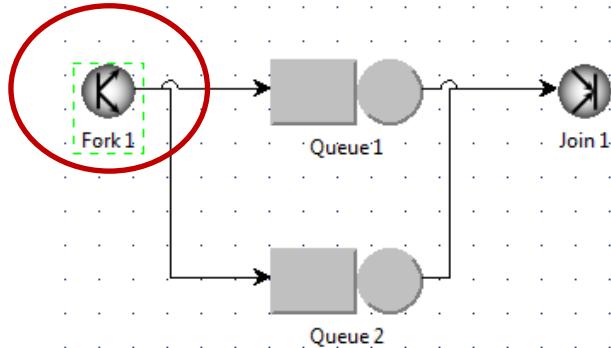
- Jobs split into *tasks* carrying *id* of the parent job
- Support for:
 - nested fork-joins
 - multiple join points
 - finite capacity between fork-join
 - advanced policies (e.g., quorum)



Advanced fork



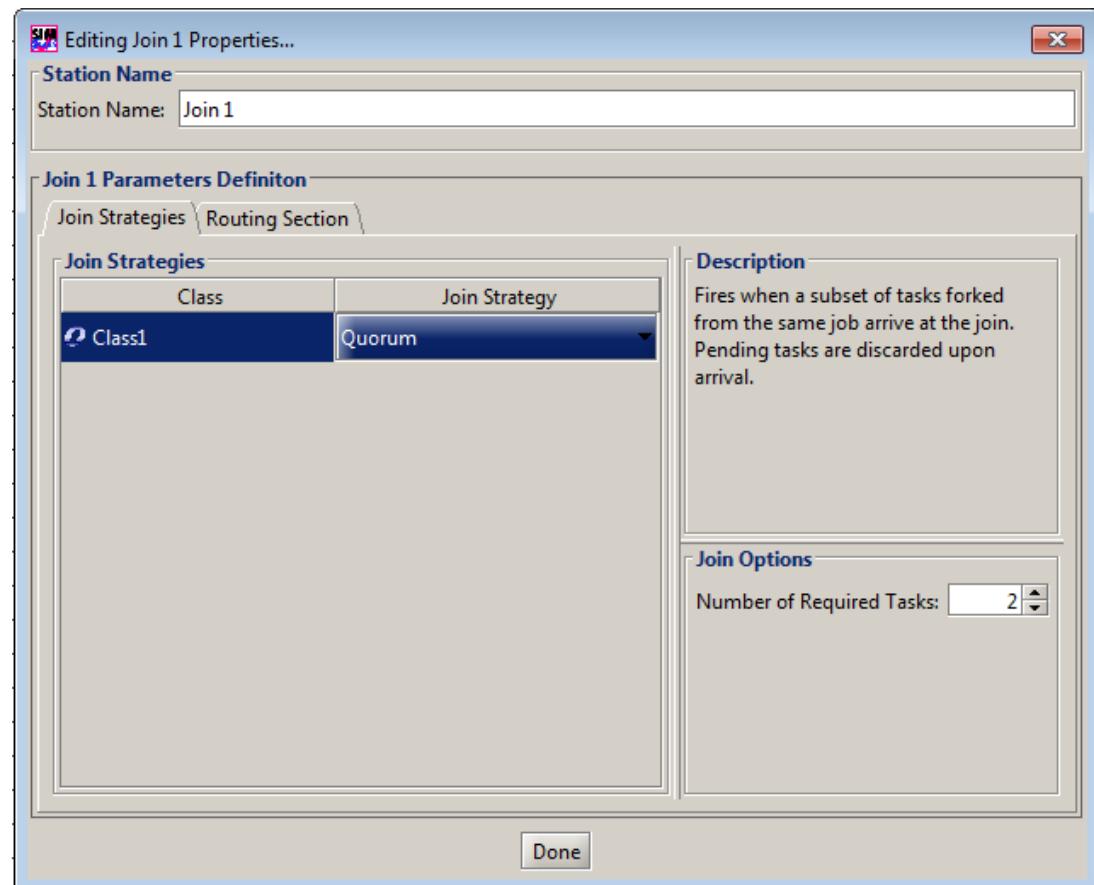
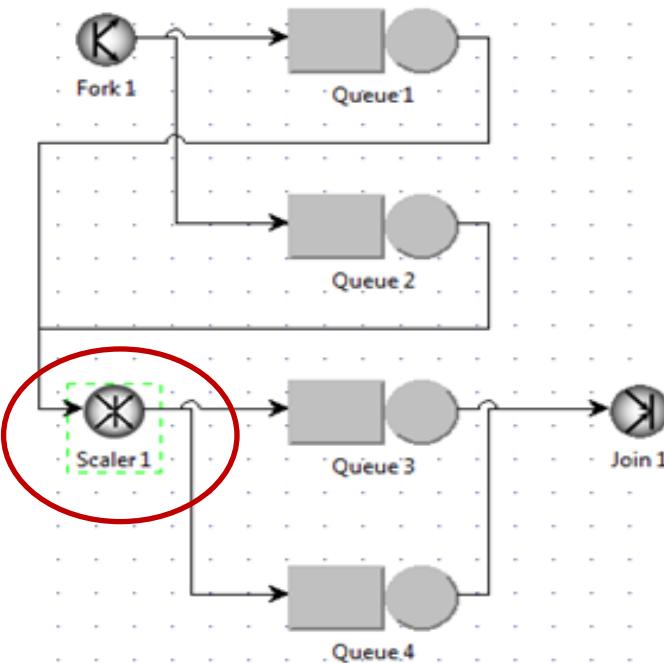
- Branch prob.: randomize no. tasks and output links
- Random subset: choose n-out-of-k output links
- Class Switch: assign new class to forked tasks



Advanced join



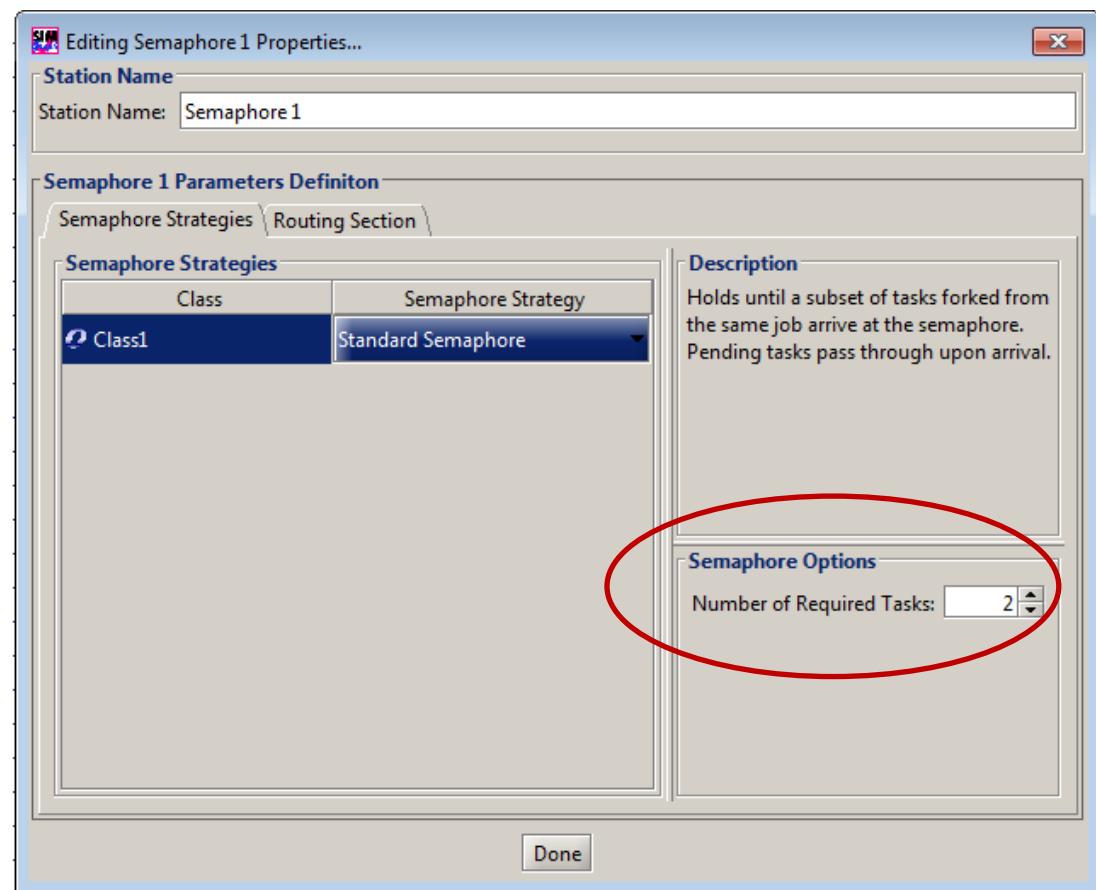
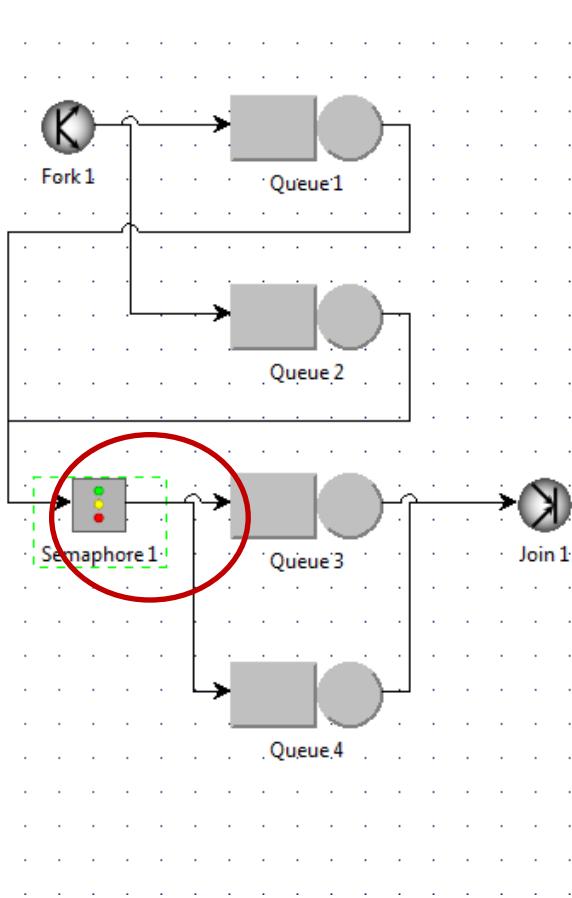
- Quorum: wait a subset of tasks (of the same job)
- Guard: like quorum but requires given class mix
- Scaler: join then fork again



Semaphore

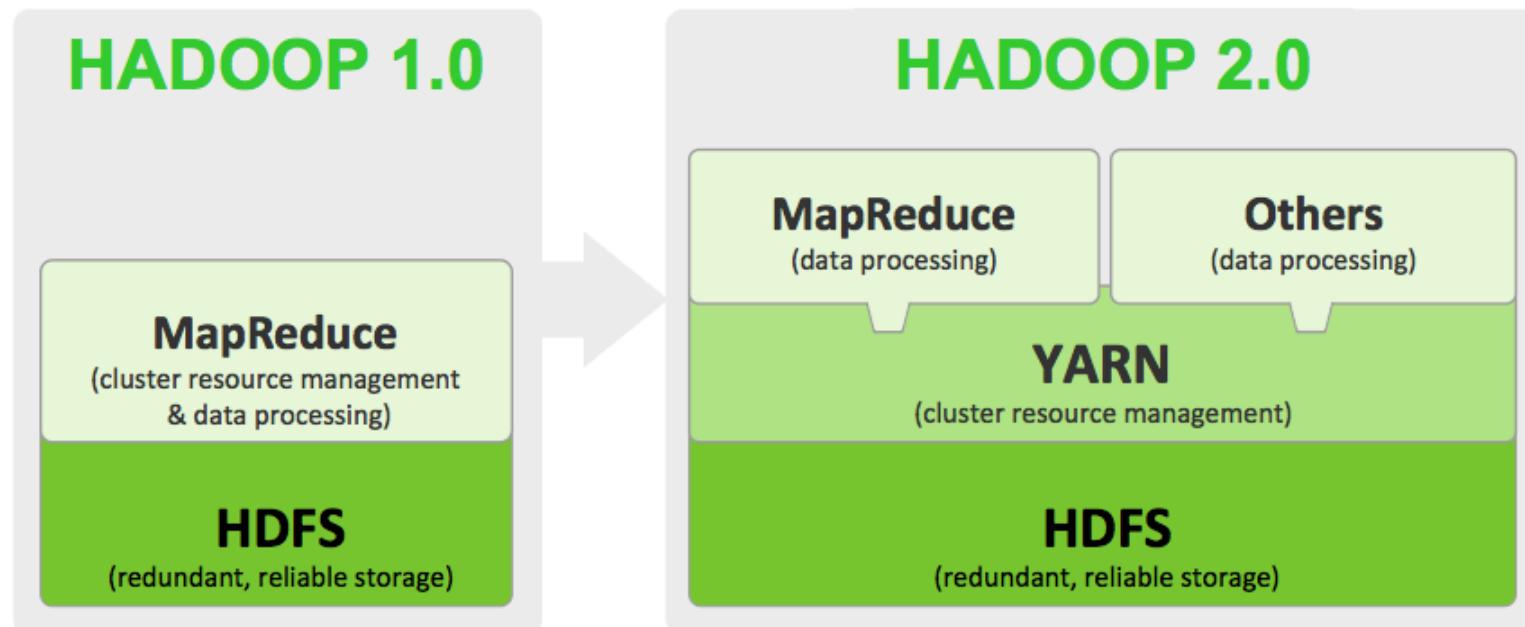


- Block first N tasks forked from the same job
- Upon arrival of the Nth, unblock and let the other pass



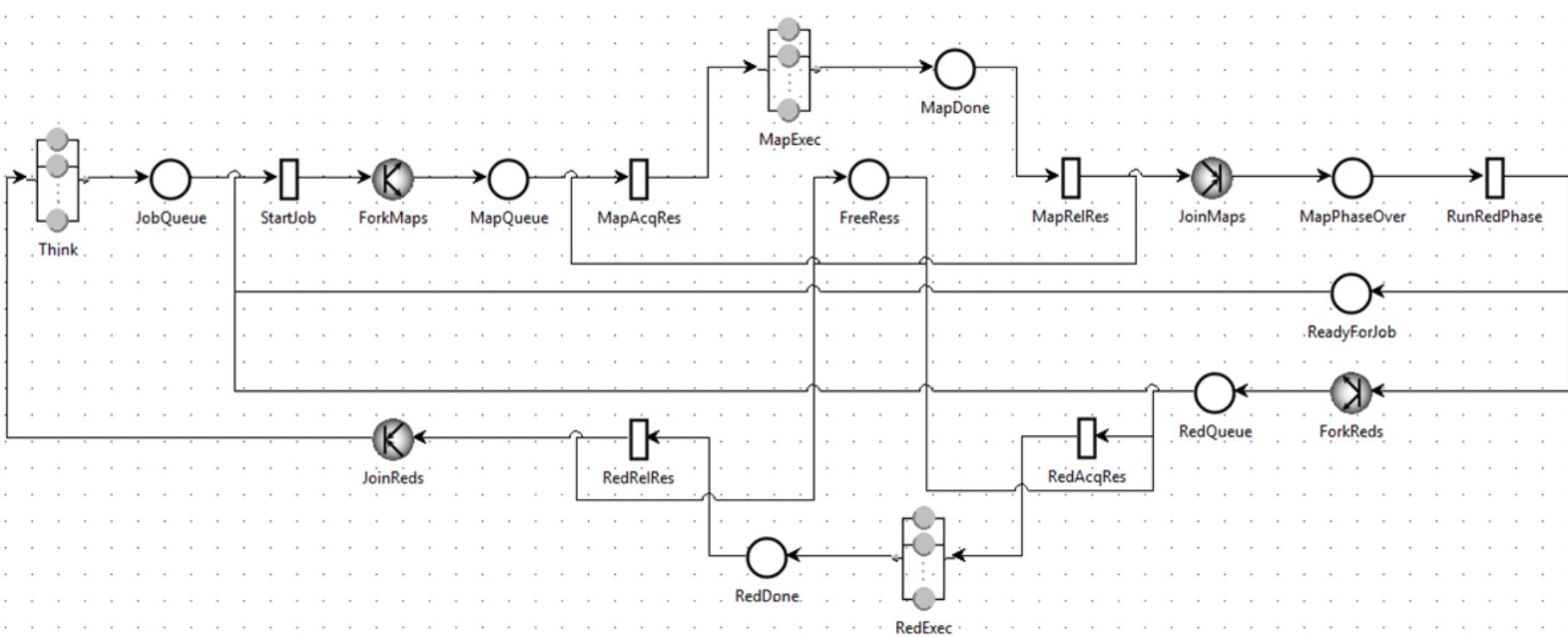
Case Study: YARN Capacity Scheduler

- YARN — Yet Another Resource Negotiator



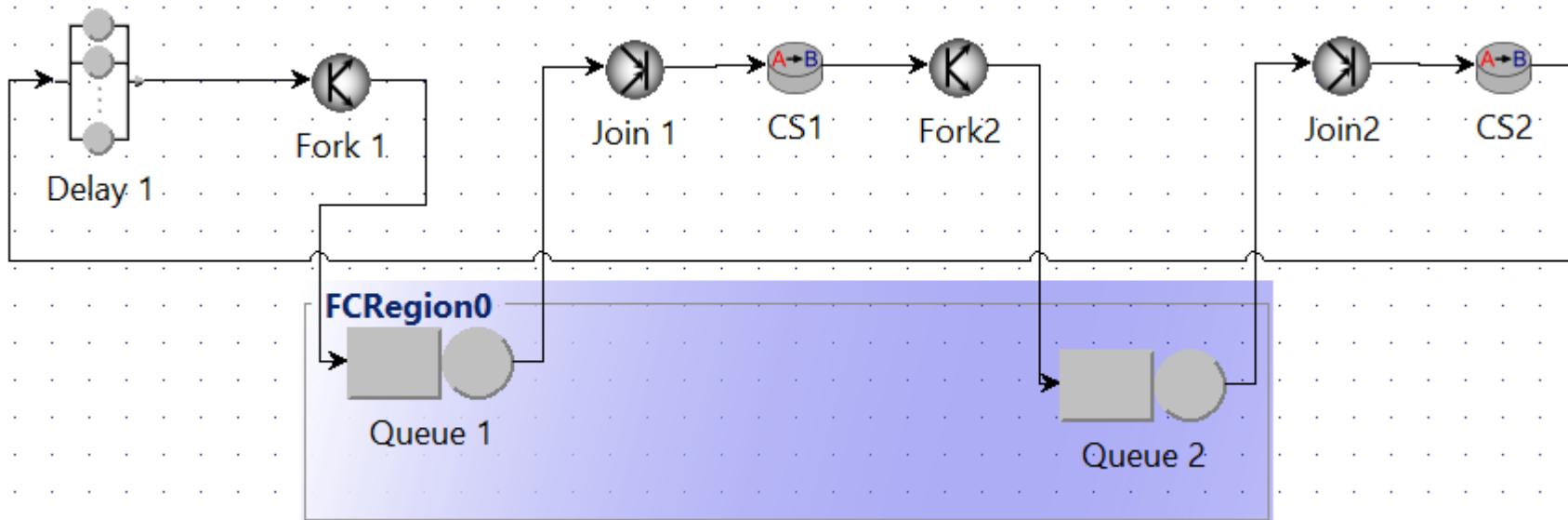
Case Study: YARN Capacity Scheduler

- Detailed model using QPN
 - Nested FCRs (JobQueue, MapQueue, RedQueue)
 - 14.13% error in trace-driven simulation [D. Ardagna *et al.*, ICA3PP'16]



Case Study: YARN Capacity Scheduler

- Simplified model using QN
 - Class switching between Map tasks and Reduce tasks



Conclusion

Coming Soon (>= version 1.0.3)

- Customer impatience
- Ability to parallelize JMT on multiple cores
 - Collect samples or run what-ifs in parallel
 - Internal simulation remains single-threaded
- New load-balancing policies
 - Power of k choices
 - SITA
 - ...
- TreeMVA in JMVA for sparse networks

