

Simalytic Modeling: A Hybrid Technique for Client/Server Capacity Planning

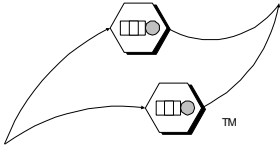
Tim R. Norton

Candidate, Doctor of Computer Science

Colorado Technical University

Colorado Springs, CO

<http://www.simalytic.com>



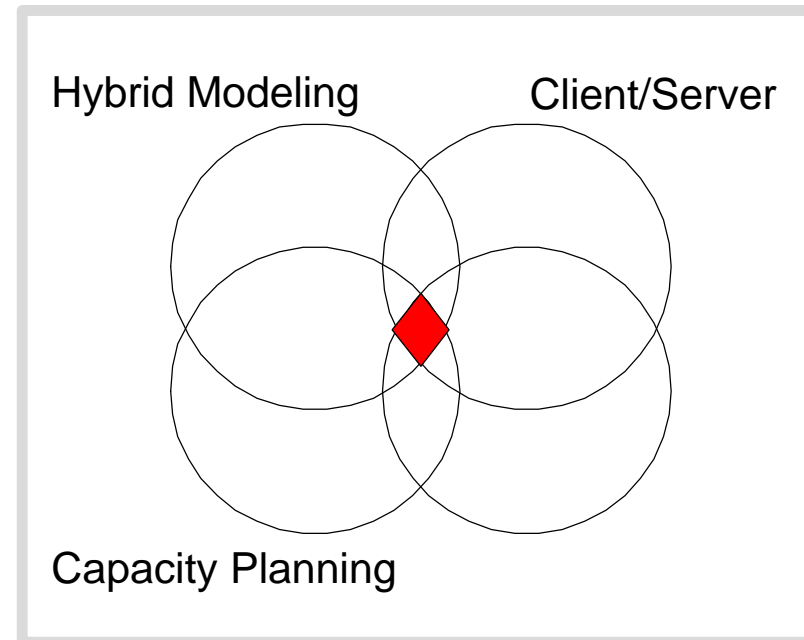
Introduction

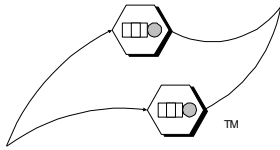
◆ Goal:

- Model Computer Application Requirements at the Enterprise Level

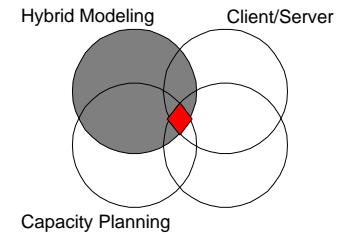
◆ Research Intersects:

- Hybrid Modeling
- Client / Server
- Capacity Planning
- Commercial Tools

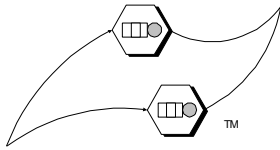




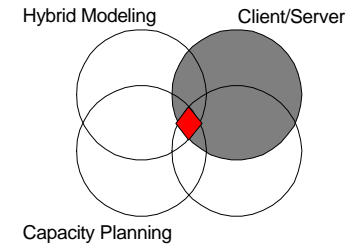
Hybrid Modeling



- ◆ Combination of Techniques
 - Simulation
 - Analytic Queuing Theory
- ◆ Advantages of a Hybrid Approach
 - Uses Strengths of Each Technique
 - Spiral Construction of Application Models
 - Reuse of Existing Models and Tools
- ◆ Transform Function Bridges Techniques

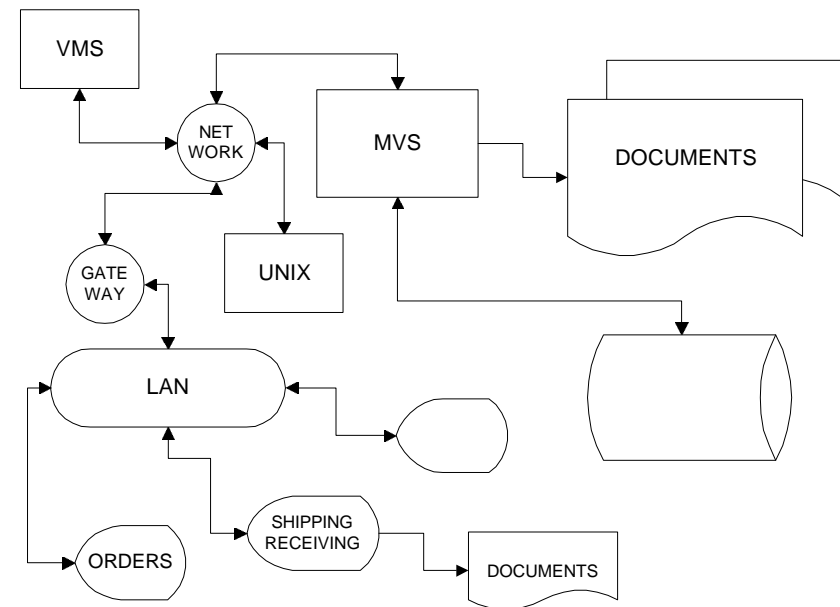


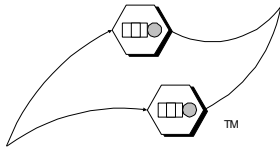
Client/Server



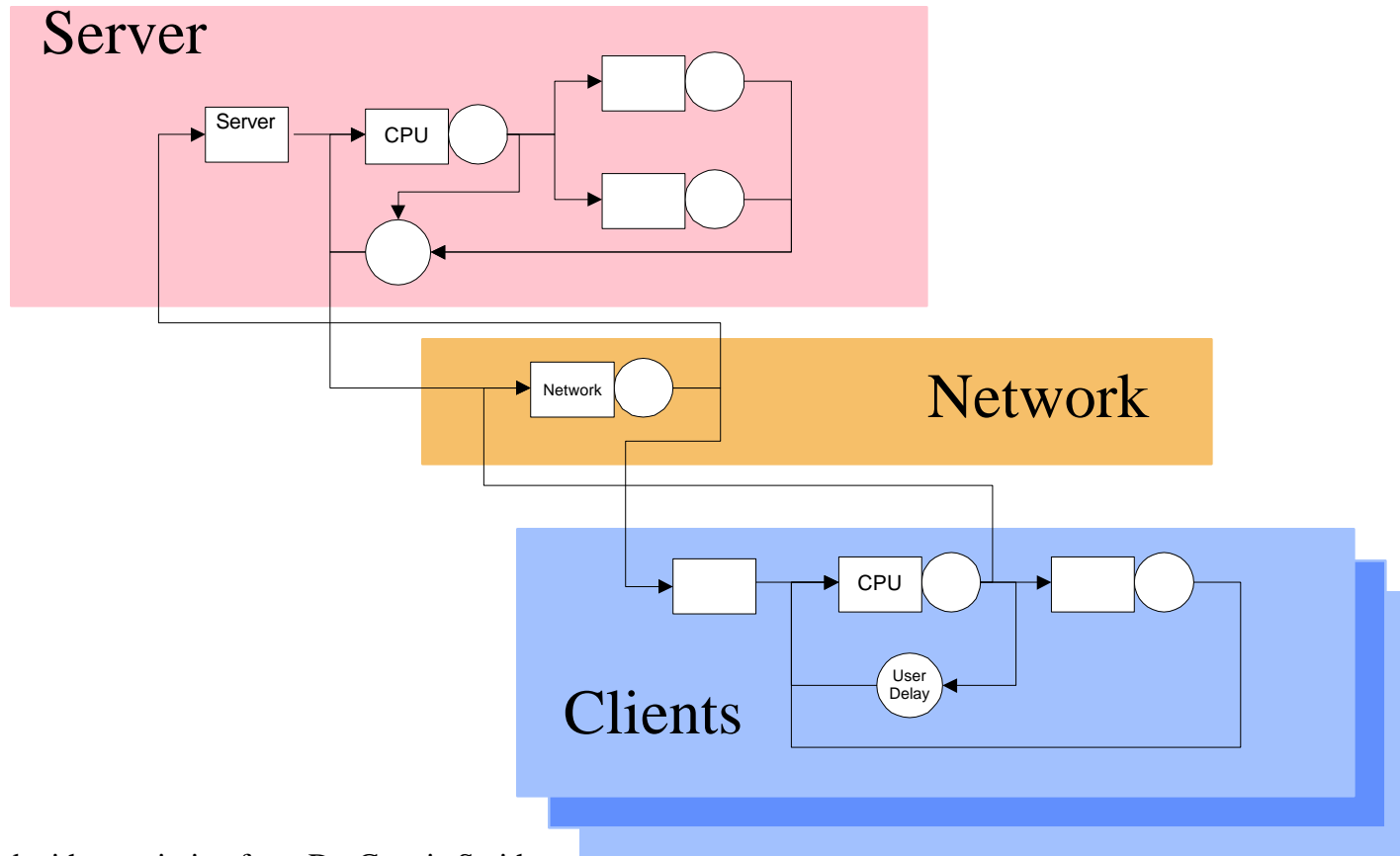
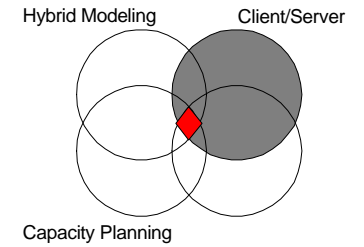
◆ Client/Server Applications

- Different Systems
- Different User Interfaces
- Real-Time Interactions
- Inter-Dependent Workloads

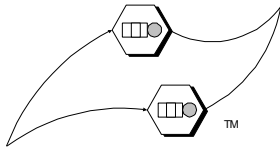




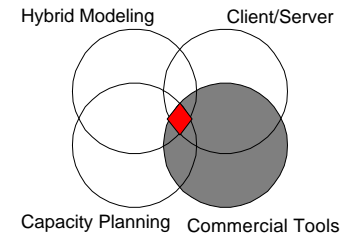
An Enterprise Model



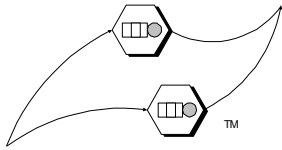
Reproduced with permission from Dr. Connie Smith,
Performance Engineering Services.



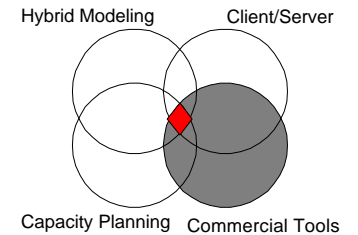
Modeling Tools



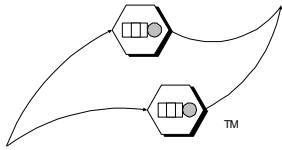
- ◆ Platform-Centric Tools
 - Narrow Focus
 - Tend to be Analytic Based
- ◆ General Purpose Tools
 - Broad Focus
 - Tend to be Simulation Based
- ◆ Address Different Problem Sets



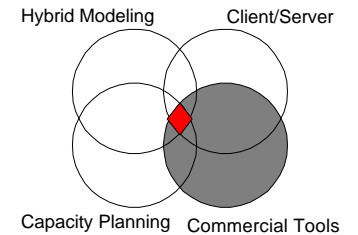
Platform-centric



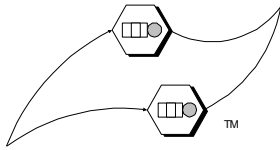
- ◆ Single platform at a time
- ◆ Detailed information about the platform
- ◆ Easier to build
- ◆ Only Environments Built Into the Tool
- ◆ Data Collected from Running Systems
- ◆ Generally Analytic or Queuing Theory



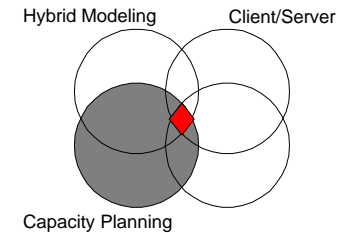
General Purpose



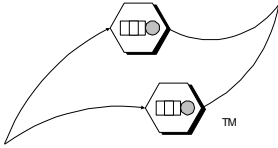
- ◆ Features to Model Anything
- ◆ No “Built-in” Platform Understanding
- ◆ Libraries of Sub-Models
- ◆ Model More Than Just Hardware
- ◆ Understand the Target System Design
- ◆ Level of Granularity = Level of Effort
- ◆ Generally Simulation Techniques



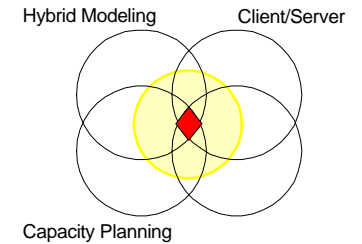
Capacity Planning



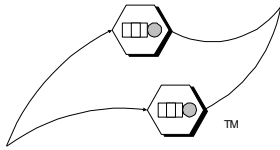
- ◆ Capacity is Measured by Business Performance Objectives
 - Decisions About Resource Requirements
 - Predicting Future Application Performance
 - Business Goals and Expectations
 - What Do We Have to Buy and When Do We Have to Buy It to Make Sure That the Business Applications Perform at the Level Required to Insure the Business Succeeds?



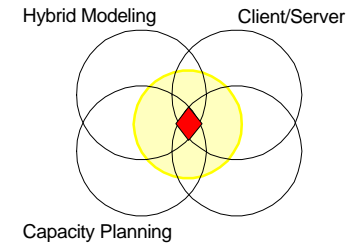
Simalytic Approach



- ◆ “Simalytic” (**Simulation/Analytic) Modeling
 - Hybrid Modeling Technique
 - General Purpose Simulation Tool for Framework
 - Analytic Modeling Tools for Nodes
 - Existing Tools
 - Predict Capacity Requirements
 - Heterogeneous Computer Systems
 - Enterprise Level Application Model**



Application Example

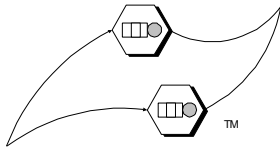


◆ Two Servers

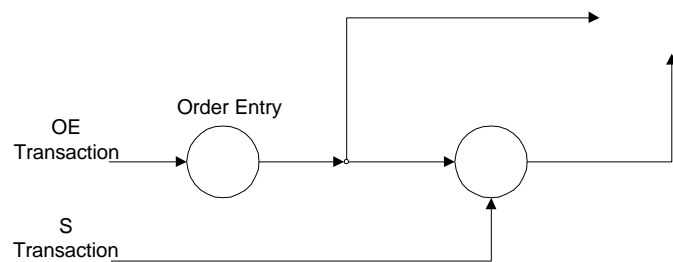
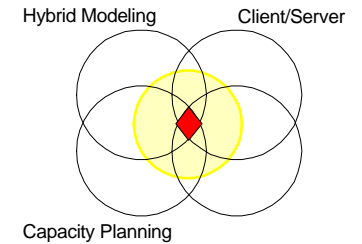
- Order Entry Server - workload of interest
- Shipping Server - also used by the Order Entry

◆ Objective of Model

- Business will grow to 4 OE arrivals per second
- Will OE server support load?
- Response Time Goals:
 - OE = 1.7 seconds
 - S = 10 seconds



Example Model



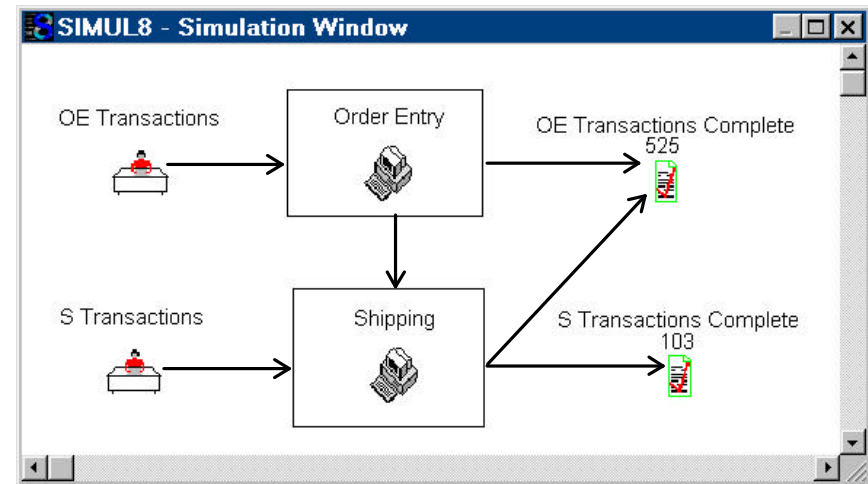
◆ Simple Two Server Model

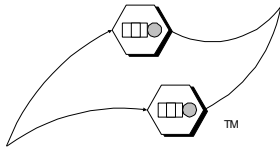
- S source constant (0.1/sec)
- OE source varied (0 to 5/sec)
- Measure both response times

◆ Simul8 Implementation

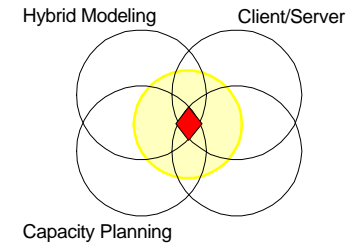
- OpenQN node models
- Visual Basic Simalytic Function

◆ Pure Simulation Model Baseline





Example Node Models



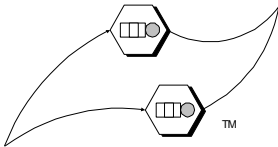
- ◆ OpenQN Results
- ◆ Implemented in Simalytic Function

	Order Entry	Shipping
Device 1: CPU	0.02	0.10
Device 2: Disk1	0.06	0.70
Device 3: Disk2	0.02	0.50
Device 4: Disk3		0.40
Device 5: Disk4		0.30
Total Service Time	0.10	2.00

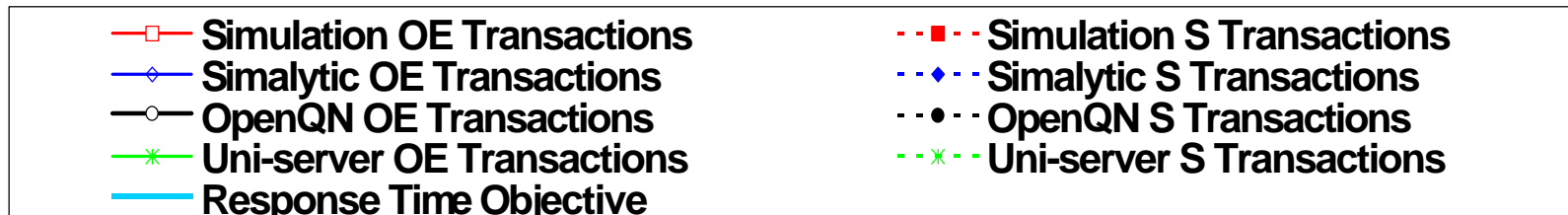
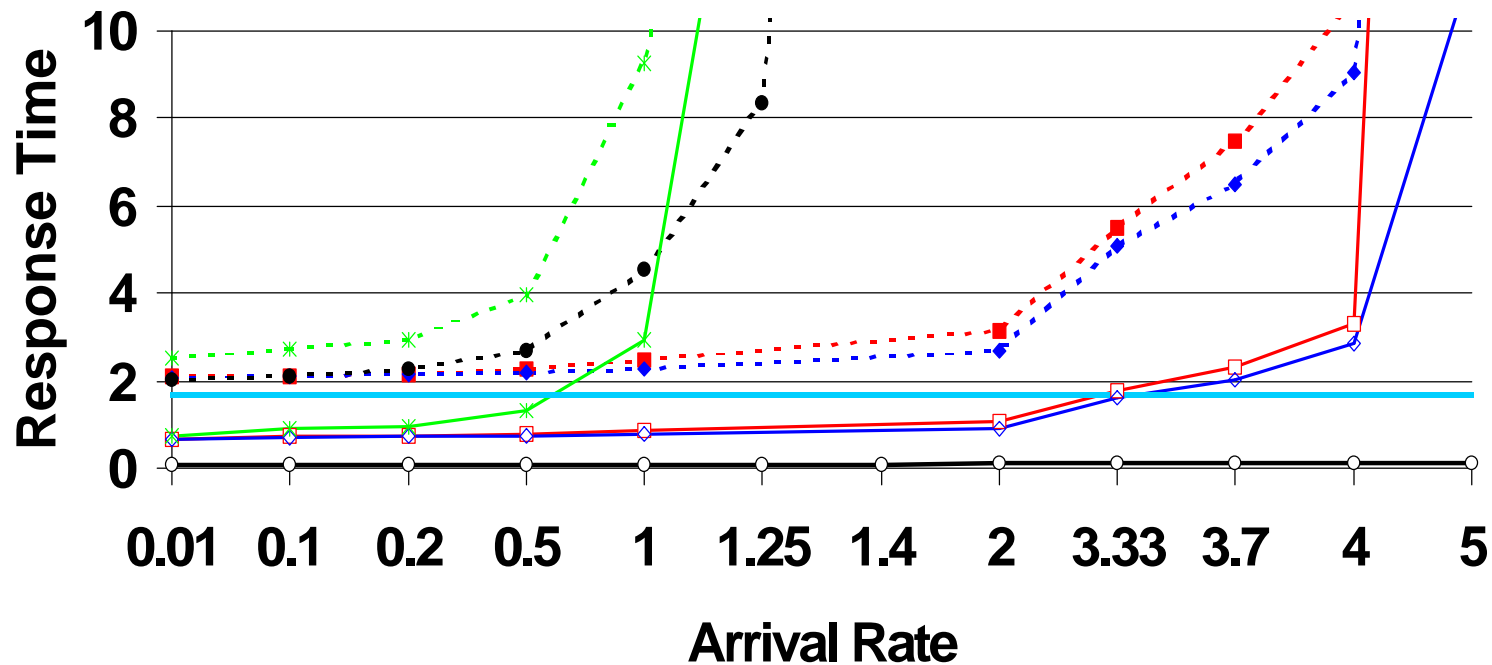
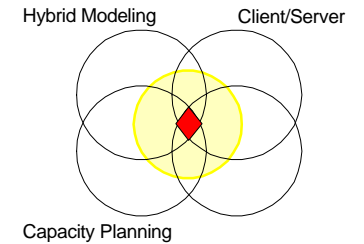
Example Device Service Times

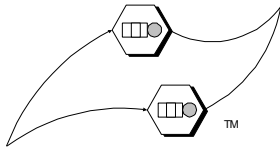
Arrival Rates	Response Order Entry	Times Shipping
0.01	0.10	2.01
0.50		2.70
1.00	0.10	4.54
1.10		5.43
1.20		6.98
1.25		8.33
1.30		10.65
1.35		15.76
1.40		38.21
1.42		119.96
2.00	0.11	
10.00	0.20	
15.00	0.66	
15.75	1.15	
16.00	1.56	
16.25	2.46	
16.50	6.06	

OpenQN Example Results

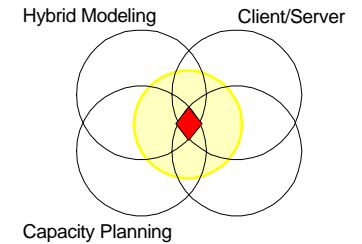


Example Results

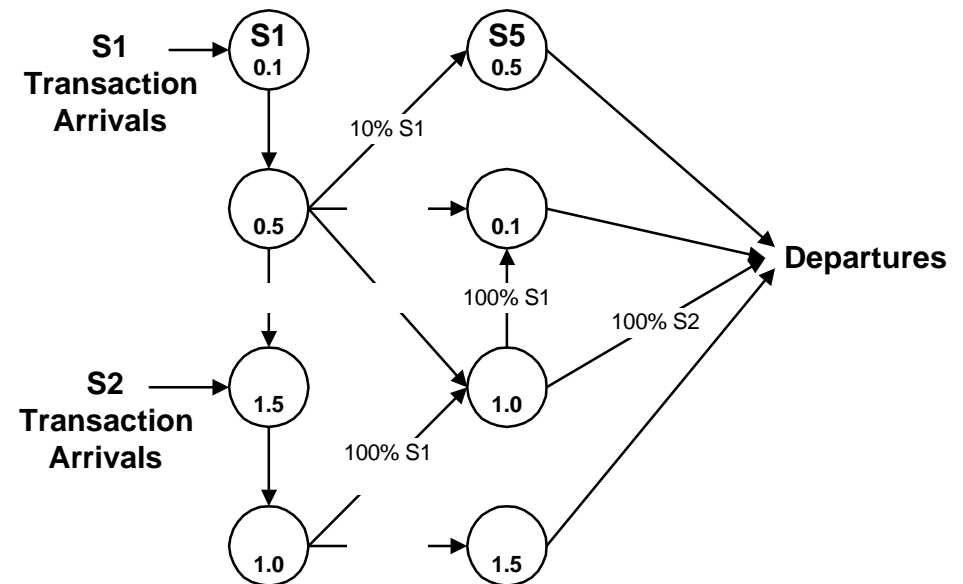


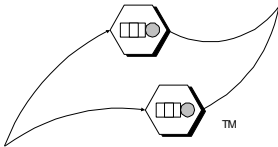


8 Server Example Model

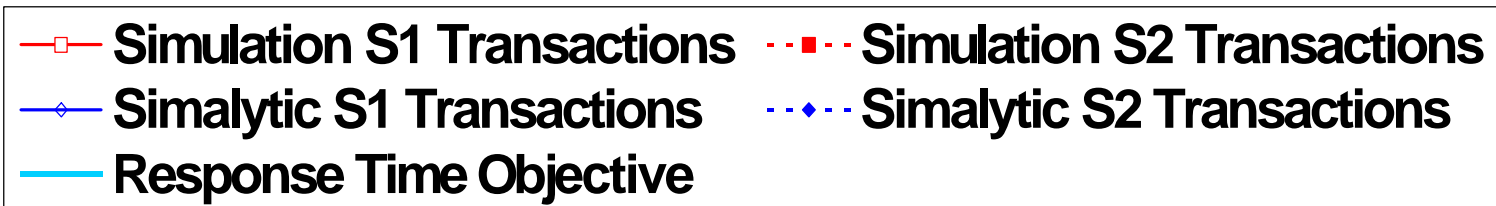
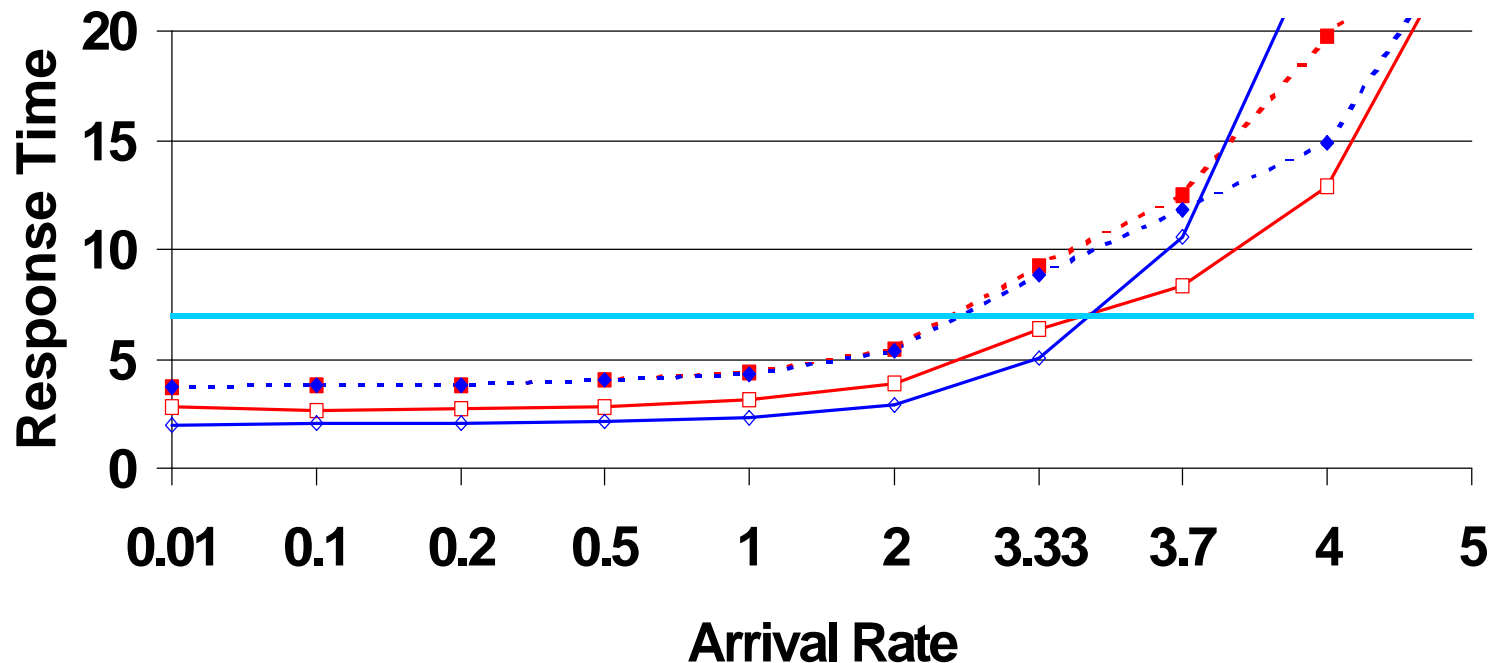
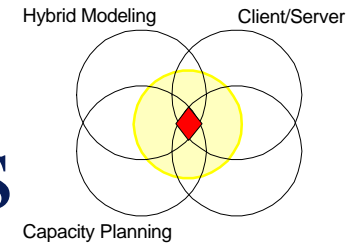


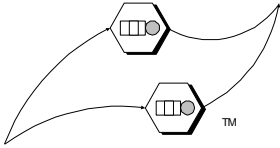
- ◆ Complex Application
 - Eight Servers
 - Two Transaction Types
 - Routing
 - Series
- ◆ Simul8 Framework
- ◆ OpenQN Node Models





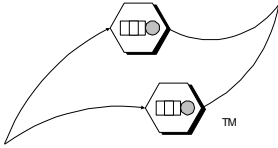
8 Server Example Results





Conclusion

- ◆ Combination of Techniques
 - Simulation and Analytic
 - Platform-Centric and General Purpose
- ◆ Predict Future Performance
- ◆ Client/Server Applications
- ◆ Reduce Time and Effort
- ◆ Focus on Business Results



Questions

?

Simul8:

Visual Thinking International Limited, 141 St James Rd., Glasgow, UK G4 0LT

<http://www.colloquium.co.uk/www/vti/>

OpenQN:

Menascé, D., V. Almeida, and L. Dowdy. 1994. Capacity Planning and Performance Modeling: from mainframes to client-server systems. Englewood Cliffs, New Jersey: Prentice Hall.

<http://cne.gmu.edu/modules/workflow/DOCUMENTS/Menasce-Abstract1.html>

Simalytic Modeling:

<http://www.simalytic.com>

tim.norton@simalytic.com