

---

# The Simalytic Modeling Technique As Applied to Capacity Planning in a Multi- Platform Enterprise

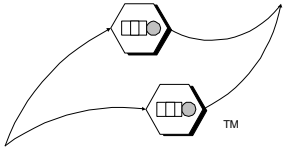
Tim R. Norton

Candidate, Doctor of Computer Science

Colorado Technical University

Colorado Springs, CO

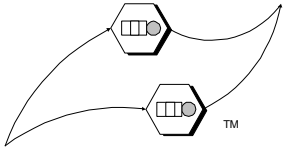
<http://www.simalytic.com>



# Overview

---

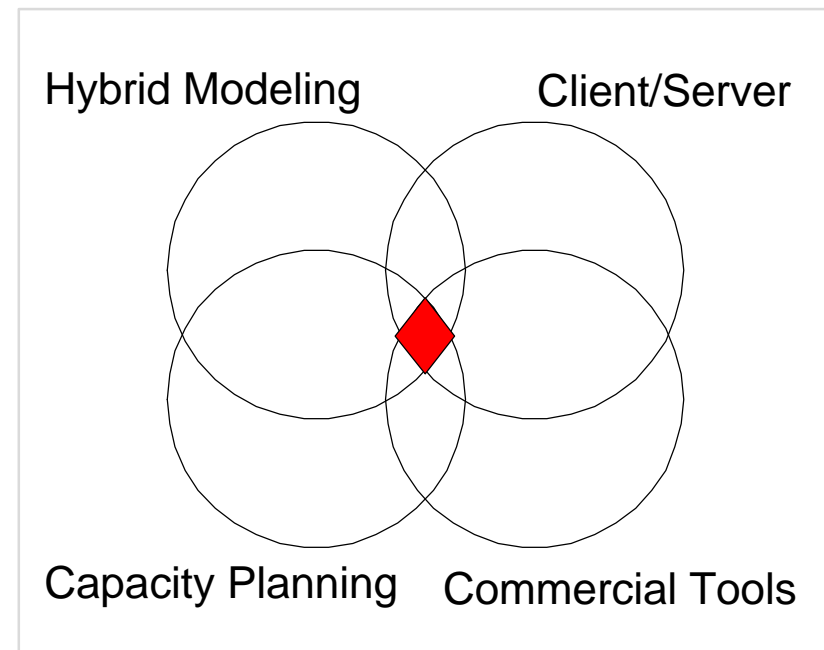
- ◆ Introduction
- ◆ Background
- ◆ Research
- ◆ Foundation
- ◆ Response Time Comparison
- ◆ Conclusion

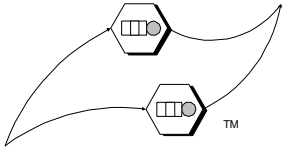


# Introduction

---

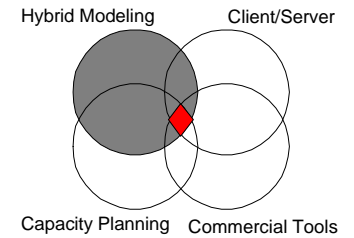
- ◆ Model Application Requirements at Enterprise Level
- ◆ Research Intersects:
  - Hybrid Modeling
  - Client / Server
  - Capacity Planning
  - Commercial Tools



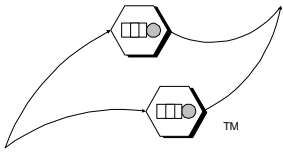


# Hybrid Modeling

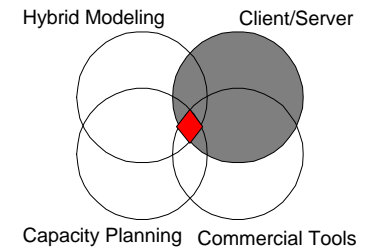
---



- ◆ Combination of Techniques
  - Simulation
  - Analytic Queuing Theory
- ◆ Needed Enterprise Modeling Technique
  - Bridge Across These Techniques
  - Construction of an Enterprise Level Application Model
  - Takes Advantage of Existing Models and Tools

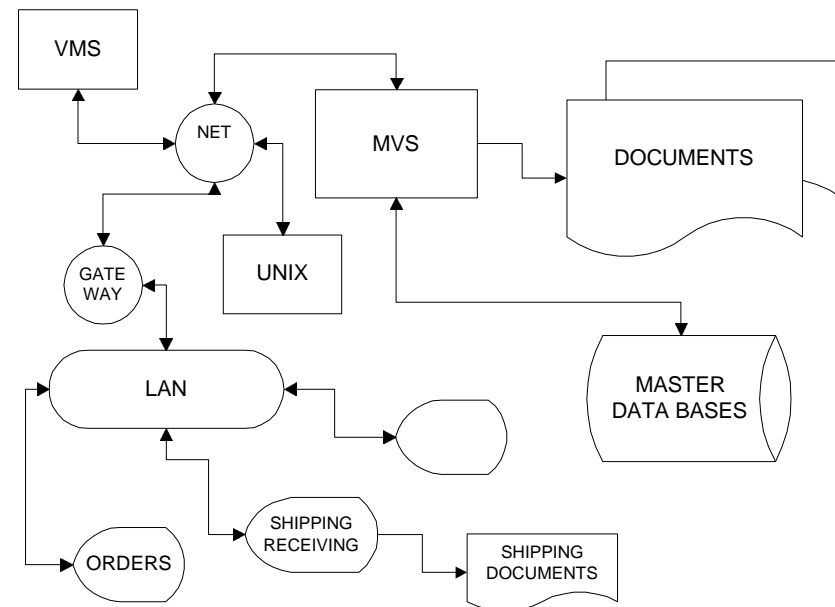


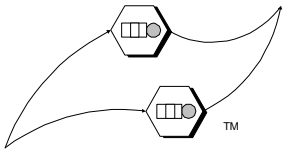
# Client/Server



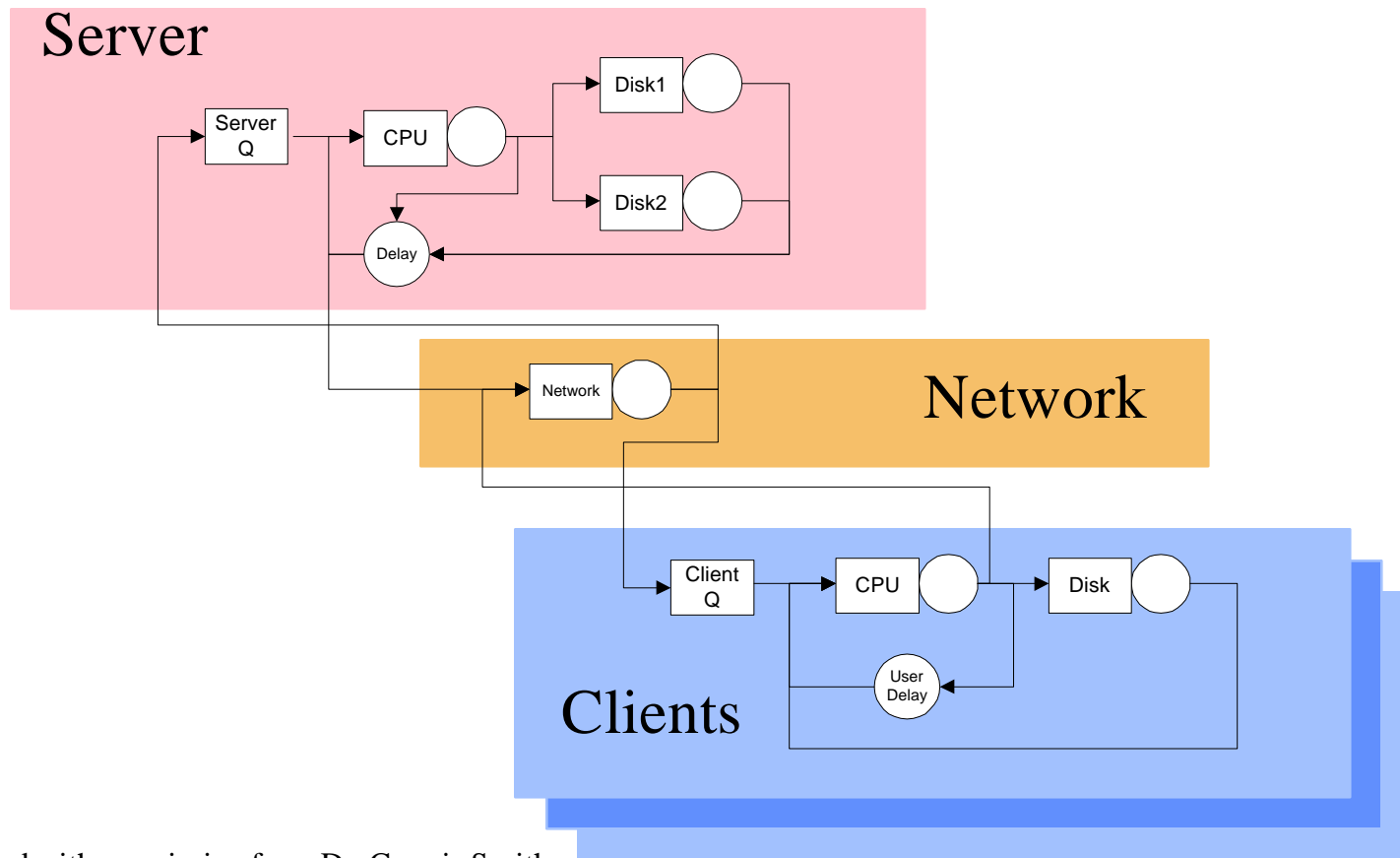
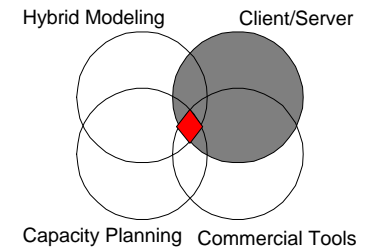
## ◆ Client/Server Applications

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

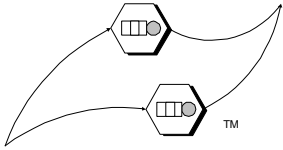




# An Enterprise Model

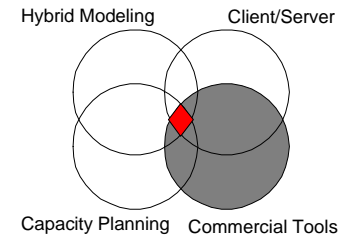


Reproduced with permission from Dr. Connie Smith,  
Performance Engineering Services.

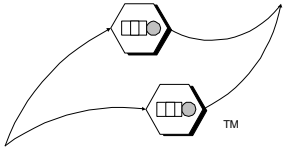


# Modeling Tools

---

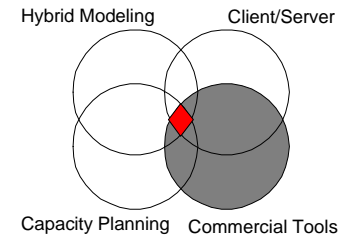


- ◆ Platform-Centric Tools
  - Tend to be Analytic Based
- ◆ General Purpose Tools
  - Tend to be Simulation Based
- ◆ Different Problem Sets



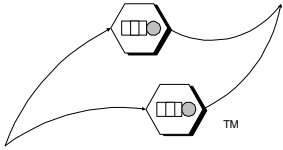
# Platform-centric

---



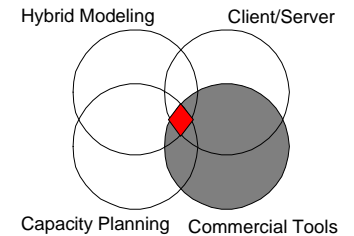
- ◆ Single platform at a time
- ◆ Detailed information about the platform
- ◆ Easier to build
- ◆ Only Environments Built Into the Tool
- ◆ Data Collected from Running Systems
- ◆ Generally Analytic or Queuing Theory



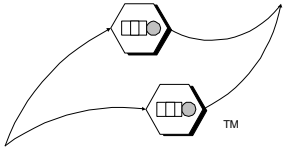


# General Purpose

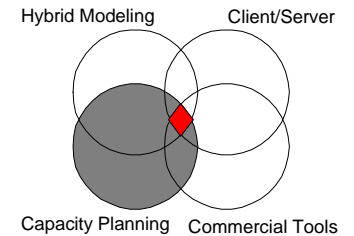
---



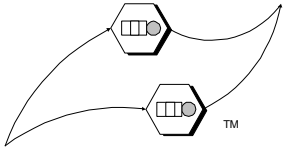
- ◆ Features to Model Anything
- ◆ No “Built-in” Platform Understanding
- ◆ Libraries of Sub-Models
- ◆ Model More Than Just Hardware
- ◆ Understand the Target System Design
- ◆ Level of Granularity
- ◆ Generally Simulation Techniques



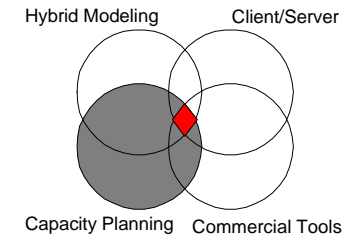
# Capacity Planning



- ◆ Capacity Measured by Business Performance Objectives
  - Decisions About Resource Requirements Based Predicting Future Application Performance Using 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?

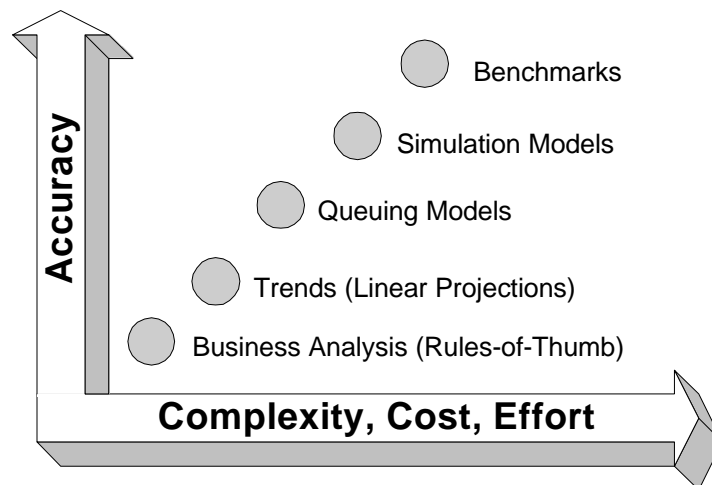


# Capacity Planning



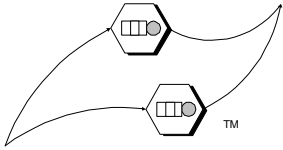
## ◆ Past

- Processor Utilization
- Overnight Batch Window

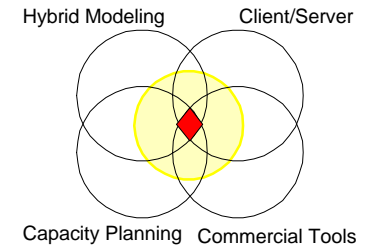


## ◆ Today

- Operating Systems
- The Platforms
- The Clients
- The Servers
- The Networks
- The Transaction Systems
- Relationships



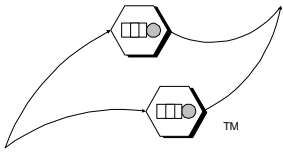
# Research Approach



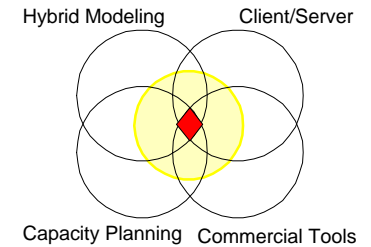
## ◆ “Simalytic” (Simulation/Analytic)

### Enterprise Modeling

- Hybrid Modeling Technique
- General Purpose Simulation Tool Framework
- Analytic Modeling Tool Nodes
- Existing Tools
- Predict Capacity Requirements
- Heterogeneous Computer Systems
- Enterprise Level Application Model



# Foundation



◆ Simulation Response Time Formula  $T = \frac{\sum_{i=1}^{n_t} T_i}{n_t}$

● Transform Function Using Queuing Theory Formula  $T = \frac{S}{1 - IS}$

●  $f(I_i)$  Replaces  $T_i$  Server Time

- Where  $T$  is the time,  $i$  is the Iteration and  $I_i$  is Based on Interarrival Times

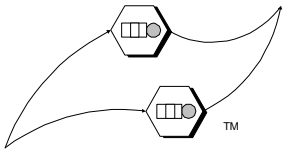
$$T = \frac{\sum_{i=1}^{n_t} f(I_i)}{n_t}$$

where  $I$  = arrivals per second as:

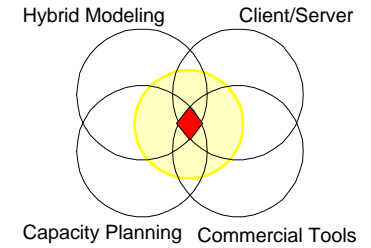
$$I_i = \frac{b}{c_i - c_{i-1}}$$

where  $c$  = simulation clock value

and  $b$  = simulation clock ticks per second

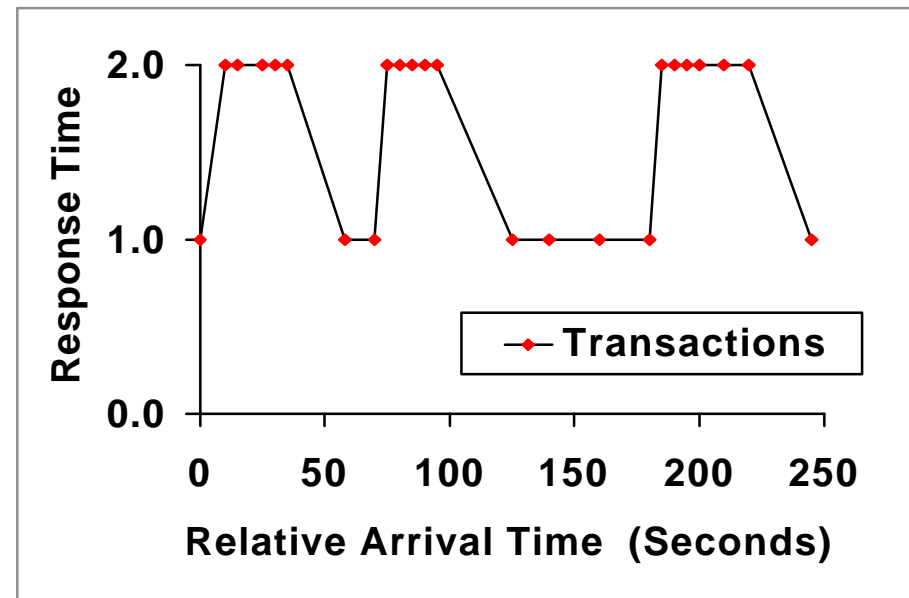
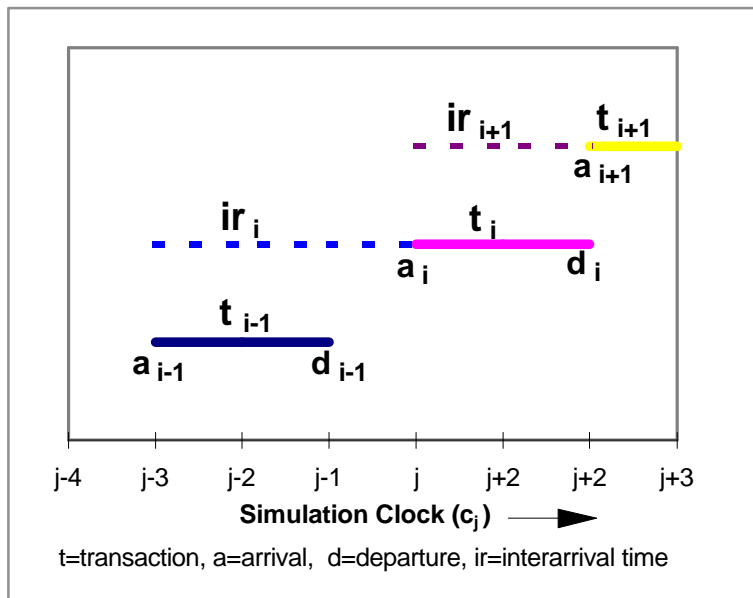


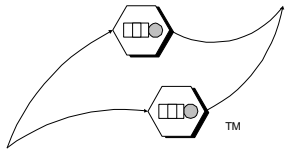
# Load Dependent



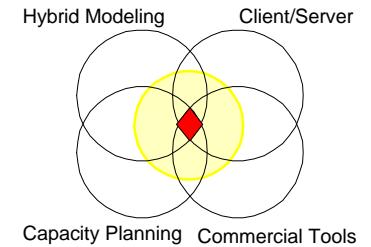
## ◆ Response Time

- Calculated Based on Interarrival Time Between Each Pair of Transactions



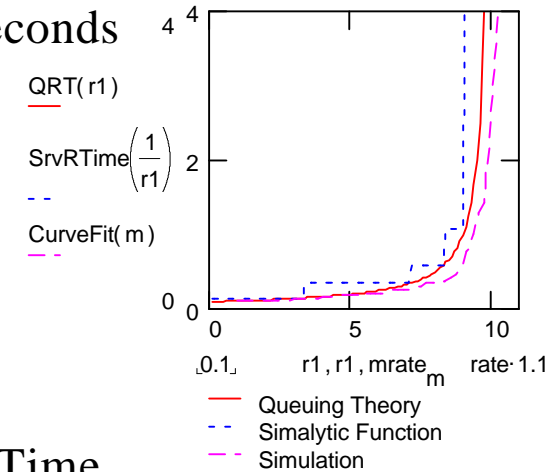


# Response Time Comparison

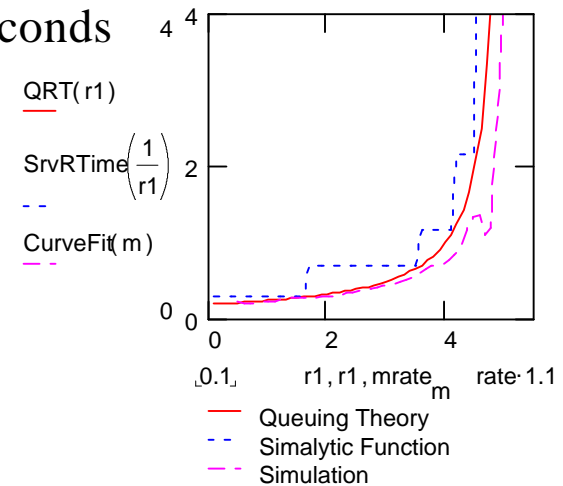


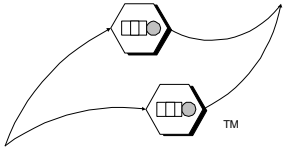
- ◆ Range of Arrival Rates for each Service Time
- ◆ Three Techniques
  - Simulation
  - Analytic Queuing Theory
  - Simalytic
- ◆ Similar Results

Service Time of .1 Seconds



Service Time of .2 Seconds



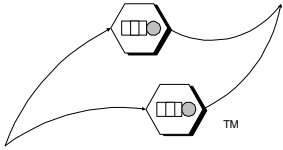


# Conclusion

---

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





---

# Questions

?