

Resource Allocation

- Last Time: Resource Management Models #3
 - » Slice or Proportional Local Scheduling
 - Timesharing to Proportional Share
 - Gang Scheduling: Combining Batch and Timesharing
 - Slices: Distributed Virtual Machine Monitors and Networks
- Today
 - » Allocation by Economic Means
 - » Project Discussions
- Reminders/Announcements
 - » Project Presentations will happen on May 31st and June 2nd
 - » Sign up for presentations next class (Thursday)

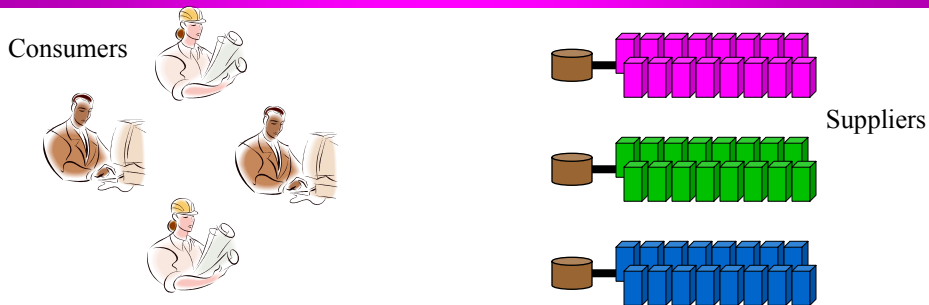
CSE225 – Lecture #12

Today's Readings

- T. Kelly, Generalized Knapsack Solvers for Multi-Unit Combinatorial Auctions: Analysis and Application to Computational Resource Allocation, HPLabs Technical Report #21, 2004.
- Wolski, Plank, Brevik, Bryan. Analyzing Market-based Resource Allocation Strategies for the Computational Grid, International Journal of High-performance Computing Applications, Volume 15, Number 3, pp. 258--281, Fall 2001.
- Auyoung, Chun, Snoeren, Vahdat. Bellagio: Resource Allocation in Federated Distributed Computing Infrastructures, OASIS '04, Boston, MA, October 2004.

CSE225 – Lecture #12

Resource Allocation Problem



- Applications/Consumers:
 - » Have work to do and needs to be met
- Resource Schedulers/Suppliers:
 - » Have resources and dictate access to them

CSE225 – Lecture #12

Allocation ?= Scheduling

al·lo·cate: to apportion for a specific purpose or to particular persons or things : **DISTRIBUTE** <allocate tasks among human and automated components>

sched·ule: to appoint, assign, or designate for a fixed time

- Allocation is about quantity or usage
 - » 100 hours of CPU time
 - » 10 Gigabytes of disk space
 - » 10 Terabytes of network use
- Scheduling is about precise mapping
 - » Use the CPU from 0800, May 7, 2004 – 0900 May 7, 2004
 - » Use 16 cluster nodes from time XX to time YY
 - » Use 100Mbps NOW for 10 minutes

CSE225 – Lecture #12

Why is Allocation Important?

- Determines if you can get enough resources to do what you want.. But not when
 - » 100 CPU years => complete a particular computation
 - » 10 Terabytes of disk space => store a particular data set
 - » 10TB of network bandwidth => move a particular data set
- Examples of other forms of allocation?
- Overallocation? And why?
- Underallocation? And why?

CSE225 – Lecture #12

Why is Scheduling Important?

- Determines how allocations are realized (consumed or supplied)
 - » Who gets to use the resource now?
 - Turnaround, interactivity
 - » Can you coordinate resource use?
 - » What is the resource utilization?
- Other things?
- Overscheduling?

CSE225 – Lecture #12

Resource Allocation in a Local Environment

- Local PC cluster of 64 nodes
 - » Total capacity = $64 * 8760 = 560,640$ CPU Hours
 - » Allocation:
 - 200,000 Computational Chemistry (GAMESS)
 - 200,000 Network/Grid Simulation
 - 150,000 Graphics/Rendering
 - 6400 Hours to Maintenance
 - Remainder is unallocated
- So, post the allocations, meter use, and we're done, right?

CSE225 – Lecture #12

Possible Resource Allocation Scenarios

- Shared Data Center (Utility Data Center), Shared Compute Facility
- Individual Users in a shared facility
- Projects (or PI's) in a set of shared facilities
- Departments in a Company
- Companies in a Pool of Resources

CSE225 – Lecture #12

Resource Allocation in the NPACI and Alliance

IBM Power4, DataStar

1088-cpu (SDSC)

IBM SP, Blue Horizon

(SDSC)

IBM Power4, Longhorn

224-cpu (U. Texas)

Cray-Dell PowerEdge Xeon, Lonestar

600-cpu (U. Texas)

AMD Cluster, Hypnos

256-cpu (U. Michigan)

AMD Cluster, Nyx

200-cpu (U. Michigan)

AMD Cluster, Morpheus

134-cpu (U. Michigan)

IBM SP, Nighthawk

28-cpu (U. Michigan)

- Allocation in a Federated Environment
 - » Three Sites
 - » A Dozen resources
- + Alliance Resources
 - » Three sites
 - » A Dozen resources

CSE225 – Lecture #12

NRAC: National Resources Allocations Committee

- Central Authority which Reviews Scientific Proposals for Use of Resources -- comes from Supercomputing History
- Allocates in chunks >200,000 SU's (~ 22 CPU years)
- Proposals indicate
 - » Desired resources and quantity of each
 - » Reason those resources are appropriate and needed
 - » Why no other resources are available
 - » Scientific justification
- Awards include
 - » Quantity of resources and sites
 - » Pattern indicates that resources are NOT viewed as interchangeable by users
- Reflect an Over-allocation of resources

CSE225 – Lecture #12

Outcomes

- Not all resources are utilized (by platform, by time)
- Allocation reflects scientific priorities and agenda
- Allocation reflects support for computations which need parallel, tightly-coupled resources
- Batch queues on systems are long
 - » Allocations are not time constrained
 - » Allocations are an overcommitment
 - » Submitting doesn't cost, only executing

CSE225 – Lecture #12

Does this model work for a Grid?

- Central allocation: no, who could it be
- Shared goals: no, what could they be
- Shared single notion of capacity: SU, what could it be?
 - » Storage and Network bandwidth
- System exists because of a “single-payer” base.

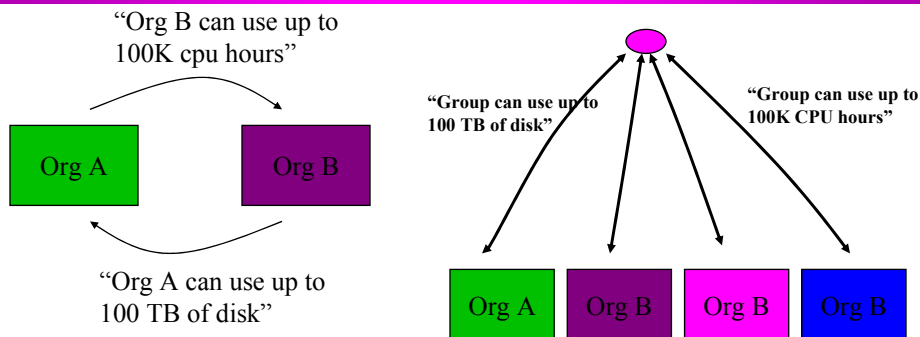
CSE225 – Lecture #12

Allocation in a Federated Environment

- Federation: independent organizations
 - » Independent values
 - » Independent resources
 - » Independent needs
 - » Independent agendas
- No trusted or controlling “parent”
 - » Bi-lateral relationships
 - » Multi-lateral relationships
 - » Transitive relationships
- Sound familiar?
- Internet? Peering, route export
- Marketplace? Contracts, consortia, reselling/transfer

CSE225 – Lecture #12

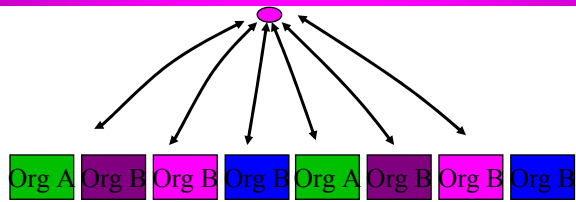
Possible Models for Giving Allocations in Grids



- Bilateral resource allocation
- Multi-lateral resource allocation
- But use is nearly always asymmetric!
- How to give and get and do so fairly? What's in it for me!

CSE225 – Lecture #12

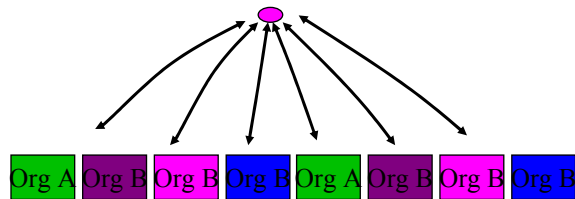
Centralized or “Marketplace” Approaches



- If we can bring the requests together, we can do allocation globally
- Centralized approach can establish a fair price for the quantity

CSE225 – Lecture #12

Centralized Allocation based on Multi-Unit Combinatorial Auction



- Set of R resource types, N_r units of each
- Set of T agents (customers) each of which submits a list of
 - » Resource bundles: things it could make use of (value for each type of resource)
 - » Utility value for each bundle All clients submit list of resource bundles and values
- Periodic Auction Allocates Bundles to Customers
- Always an excess of resources – no real competition on price

CSE225 – Lecture #12

Motivation for Economic Frameworks

- Frameworks which don't require a trusted 3rd party
- Frameworks which don't have artificial rules which limit self-interested behavior
- Open system assumptions
- And...
 - » Resource usage is not free (fixed and variable costs)
 - » Resource owners must be compensated to encourage federation
 - » Dynamics of behavior are complex and non-linear – recourse to body of economic knowledge???

CSE225 – Lecture #12

What is a Solution?

- Each Agent gets one bundle
- Not all resources get used
- Optimization = Maximize the aggregate utility over all the Agents
- Perfect model for a resource provider – in a utility
 - » Existing resources
 - » Best assignment to customers for delivering value
 - » Price is another matter ☺

CSE225 – Lecture #12

MDMCK Results

- MDMCK = multi-dimensional multi-choice knapsack problem
- So how hard is this?
 - » NP-hard in general
 - » If # of resource types is small
 - » Algorithms Linear in
 - Number of agents
 - Length of bids
 - # of units of each resource

[Kelly, Hewlett-Packard]

CSE225 – Lecture #12

Possible Applications

- Periodic Allocations for a Data Center
 - » Months, Weeks, Days, Hours
- Periodic Allocations of Grid resources
 - » In a group or market? View value as the price/bid
 - » Are the resources interchangeable – network and resource location costs
- This is an extremely efficient allocation mechanism.

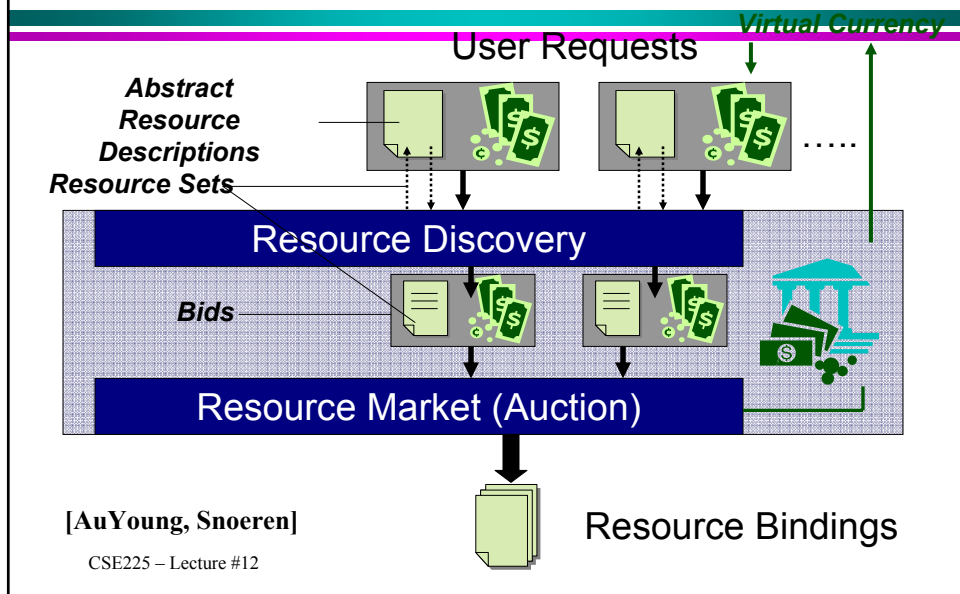
CSE225 – Lecture #12

Two Approaches

- Bellagio: Auction-Based Technique
- G-commerce: Commodity Pricing Technique (Market, but not Auction)

CSE225 – Lecture #12

Bellagio Architecture



- **Bids**
 - [quick bid](#)
 - [advanced bid](#)
 - [view bids](#)
 - [view allocations](#)
- **Bank**
 - [view account info](#)
 - [view account users](#)
 - [view currency info](#)
- **Nodes**
 - [view all nodes](#)
 - [view availability](#)
 - [view price history](#)
 - [view bind history](#)
- **Admin functions**
 - [view all bids](#)
 - [view all users](#)
 - [view site accounts](#)
 - [view failed tickets](#)

Welcome **alvina@cs.ucsd.edu!**

Next auction clearing time is: **Wed Apr 27 11:00:00 2005 GMT**

Quick Start:

1. *Submit a bid for CPU shares on a specific set of PlanetLab nodes.*

Once you have successfully submitted a bid, you will be able to see the details of [your bid](#) and when it will be cleared.

2. *Wait for the Bellagio combinatorial auction to clear (every 4 hours).*

We will place a hold on the amount of your bid in your institution's bank account. You will be able to cancel your bid before the auction is cleared in order to return this money to your bank account.

3. *If your bid wins:*

You will receive an allocation for a specified period of time. During this allocation period, your slice will be augmented with CPU shares on PlanetLab machines contained in your allocation. Your institution's bank account will be debited by the amount of your bid, with an equal portion of the bid being split among the

CSE225 – Lecture #12

Example: G-Commerce

- **A Resource Economy which achieves resource allocation...**

Requirements:

- **Grid-wide Price Stability**
 - » They don't change wildly, allowing rational scheduling and planning
- **Market Equilibrium**
 - » Price fairness
- **Application Efficiency**
 - » Applications can get things done (translate resource consumption into progress)
- **Resource Efficiency**
 - » High Utilization

CSE225 – Lecture #12

Challenging Questions

- How to establish price for exchange of goods?
- What properties for the system, participants are desirable? (see previous)
- What can we assume about the behavior of parties?
- What properties for the system, participants are achieved?

CSE225 – Lecture #12

Assumptions

- Relative worth of a resource is determined by the supply and demand for it.
 - » If it is traded, providers can choose not to participate
- Resource decisions are based on self-interest are a fundamental element of any federated system.
 - » Some decisions may not be based on self-interest, but you cannot assume that this is true in general
- How general are these assumptions?
 - » Examples
 - » Counterexamples

CSE225 – Lecture #12

Model

- Supply: slots of 10% units, sell if current price is greater than average income to date
 - » if I don't sell any, then keep dropping my threshold
 - » If sell at a high price, threshold goes up
- Demand: Initial \$G, periodically wiped out (no reward for saving)
 - » General a clump of jobs, randomly thru day, run subject to economics
 - » Run if current price is lower than funds left
 - » expend rate is greater than average paid to date

CSE225 – Lecture #12

Transactions

- Durations are ignored.
- Prices for resources are fixed at time a job begins, and run.
- Jobs are fixed length at the time they are submitted.
- Assumptions:
 - » Resources are slowly varying – don't need to focus on durations
 - » Jobs don't vary their duration based on the cost (non-adaptive)
 - » Consumers/Producers honor their commitments

CSE225 – Lecture #12

Pricing

- Commodity markets: set a price for the commodity at each point in time. Strong proofs of existence of an equilibria, but not constructive.
- Smale's method
 - » Global hill climbing in a multi-variate space
 - » Poll incremental changes in consumers and producers (approximate)
- First Bank of G
 - » Use tatonnement – raise/lower on each variable to reach equilibrium
 - » Use historical polynomial fit to estimate excess demand to fine time
- Auctions
 - » Sealed-bid, second price auction
 - » All sorts of regularity problems (1000's of these all the time??)

CSE225 – Lecture #12

Simulation Parameters

- 100 cpus, 100 disks
- 2-10 slots/cpu
- Jobs 1-60 time units, 1440 time units/day
- 1-100 jobs submitted at daybreak
- 0.1 new job probability
- ...

- => Yes, this is a lot of assumptions...

CSE225 – Lecture #12

Prices for Under-Demand

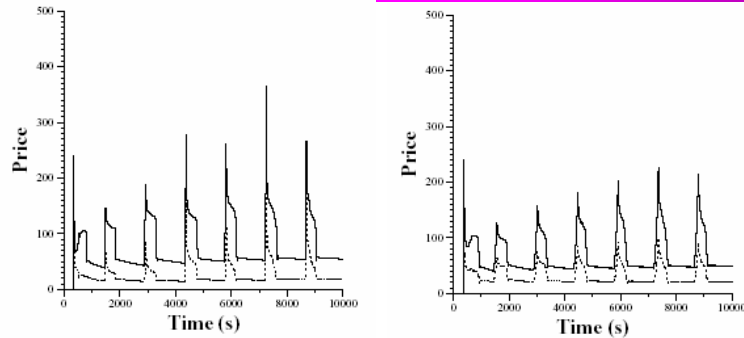
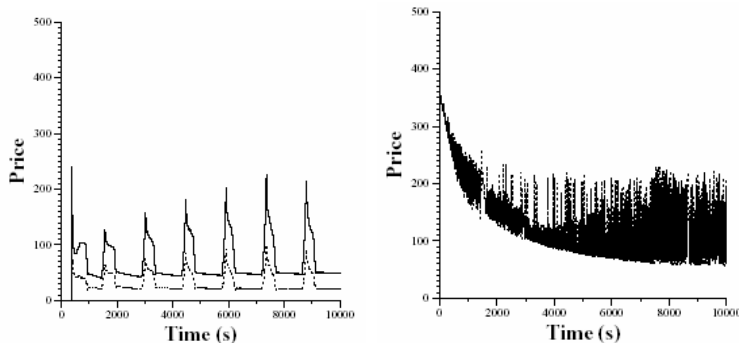


Figure 4. Sample prices for the under-demand scenario.

- Burst increase at start of each day; works off backlog, rapid return to lower price
- Methods work similarly well, Bank of G has scalability advantages

CSE225 – Lecture #12

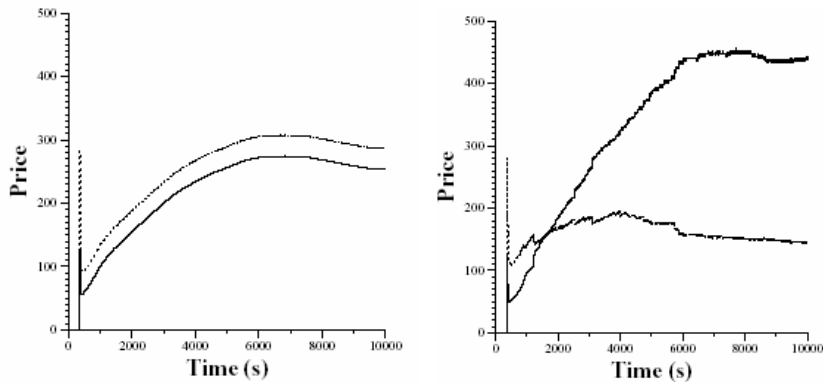
Prices for Under-Demand



- Averages shown in the Auction pricing
- Auction pricing produce VERY different results
- Higher in absolute terms; unstable across consumers

CSE225 – Lecture #12

Prices for Over-Demand



- Fig. 10.10: Different prices for CPU and disk
over time
dot: CPU
solid: disk
Claim: these are multiple-equilibria for the system
are SG.

CSE225 – Lecture #12

Allocation and Scheduling

- Centralized allocation is analogous to global scheduling
 - » Know all the value functions, optimize use of the resources
 - » However, scheme has no specific notion of time value or of changing value
 - » All values and bundles synchronized in rounds
- Example: I need 200 procs, high value, but then I finish my job => Low value
 - » No easy way to change the value
 - » No way to reflect until next round

CSE225 – Lecture #12

Discussion

- What insights do these models / simulations bring to the idea of using economic mechanisms for resource allocation?
- What type of attention/information do you need to do well in this type of environment?
- How robust are the results? Which of the assumptions seem strongest? Weakest?
- Can these approaches be used for allocation and mapping? Remapping?
- Can we extend this idea to Federated environments?
 - » Can we apply it directly?

CSE225 – Lecture #12

Summary

- Federated Resource Allocation
 - » Current NPACI/Alliance/NRAC Model
 - » Underlying batch scheduling to maximize resource efficiency
- Grid techniques for No trusted 3rd parties
 - » Bilateral and multi-lateral approaches
 - » Marketplace approaches
- G-commerce uses Commodity markets to address this issue
 - » Appears to work better than auctions – traditional approach for stability
 - » Smale's method, First Bank of G methods
- A Centralized Allocation approach and its limitations

CSE225 – Lecture #12

Supply (Resource Managers)

- CPU – sells units of computation in slots (10% units)
 - » Heterogeneous speeds can be handled this way
- CPU – sells all available slots if the current price is greater than the average income to date
 - » If $\text{current_price} > \text{mean_price}$
 - » Where $\text{mean_price} = \text{myrevenue}/\text{time}/\text{myslots}$
- => if I don't sell any, then keep dropping my threshold
 - » More aggressive with no income
- => if I do sell at a high price, threshold goes up
 - » Less aggressive, having made more money

Demand (Consumers)

- Given an initial \$G; periodically wiped out (refresh)
 - » Analogy to the NPACI Approach, Planetlab?
 - » No reward for saving (or ability to do so)
- Jobs specific a size (amount of computing cycles) and a duration – runtime
- Generate a “clump” of jobs each morning
- Generate jobs randomly throughout the day
- Run them subject to economics...

CSE225 – Lecture #12

Demand (consumers), continued

Calculates

- Cost for jobs run so far if at current price / time so far
$$\frac{\text{Work done so far} * \text{current price}}{\text{Current_Time_in_Period}} \quad \text{Average_rate}$$
- Spending rate to expend budget by end of period
$$\frac{\text{Funds_remaining}}{\text{Time_remaining_in_period}} \quad \text{Capable_rate}$$
- If Capable_rate > Avg_rate then generate work
 - \$/time
 - If have lots of funds, spend out by end of period, do more
 - If either got lots of work done or current price looks high, do less

CSE225 – Lecture #12

So, how does this all work?

- Haven't said how prices are set! There is a market-maker

Summary

- All entities act in their self-interest (except the market-maker)
- Producers consider long-term profit and past performance when deciding to sell or not
- Consumers are given periodic budget replenishment and spend opportunitically
- Consumers introduce workloads in bulk at beginning of day, and randomly throughout the day (enter system subject to economics)