

# Resource Management IV

---

- ❑ Lecture based in part of two further survey papers:
- ❑ A Taxonomy of Workflow Management Systems for Grid Computing
  - Yu and Buyya
  - Journal of Grid Computing
  - Vol 3. pp 171–200 (2006)
- ❑ A Survey of Bargaining Models for Grid Resource Allocation
  - Sim
  - ACM SIGecom Exchanges
  - Vol 5, #5, pp22–32 (2005)
- ❑ Strongly encouraged to gain detail by skimming through the first and studying some of the listed systems, and by familiarising yourself with the nature of the problems and outline solutions in the second

GC5:1.14 – p.1/12

# Workflow Mgmt: Information, Scheduling, FT

---

- ❑ The systems exploit static, historical and dynamic information (see fig 10, c.f. the information considered in the taxonomy in lecture 11)
- ❑ Scheduling also can be structured (again cf lecture 11 taxonomy) into:
  - architecture (centralized, hierarchical, decentralized)
  - decision making (local v global)
  - planning (static: user vs simulation; and dynamic: prediction vs JIT)
  - strategy (performance, market or trust driven)
  - performance estimation technique (various forms)
- ❑ A rich decomposition of views of Fault Tolerance is also possible covering task level and workflow level issues

GC5:1.14 – p.3/12

# Workflow Mgmt Systems: Architecture and Design

---

- ❑ Architecture laid out neatly in figure 1 of Taxonomy
- ❑ Various facets of the system form the taxonomy, e.g. Workflow Design
  - Structure, Model/Specification, Composition, QoS Constraints
  - Structure further separates: DAG (Sequence, Parelel, Choice) and non-DAG (as DAG plus Iteration)
  - Model/Specification can be abstract or concrete
  - Composition can be automatic or user-directed (language based, e.g. markup, or graph-based, e.g. UML or PetriNet)
  - Constraints could be: time, cost, fidelity, reliability, security

GC

# Survey - a wide range of projects

---

- ❑ The survey covers over a dozen projects, including:
- ❑ ICENI, which builds on Globus and is a non-DAG abstract user-directed system
- ❑ GridFlow, in contrast, is a DAG-based system built on the PACE toolkit for performance monitoring and the Titan scheduler
- ❑ one is centralized, the other hierarchical (which suits Titan)
- ❑ Explore the tables of projects and read some of the summaries, to get a sense of the diversity of approach
- ❑ A key feature that distinguishes many of these tools, and which arises in other Grid settings, is the underlying economic approach to resource management.

GC

# Basic Economics

- ❑ **cost** and **price** are not the same
  - as a producer, the cost is a measure of what it costs me to produce and sell something, which will be the sum of the prices I pay for components, plus my additional overheads etc
  - the price is what I charge a consumer for that good
- ❑ marginal cost is the cost of producing one more of the good
- ❑ true cost includes the amortised cost of the equipment, including the cost of capital
- ❑ cost will vary depending on the volume of goods I produce
- ❑ if demand rises, I may produce and sell more goods
  - cost may go down if I spread my equipment costs more thinly
  - cost may go up if I increase demand for a scarce input
- ❑ Many projects (e.g. [lesc.ic.ac.uk/markets](http://lesc.ic.ac.uk/markets)) aim to produce a basic market economy for the Grid, enabling resellers, futures etc
- ❑ How does this apply to computer resources?

GC5.1.14 – p.5/12

# Congestion Charging

- ❑ Idea of congestion charging is to allow for the externalities
- ❑ Don't normally pay, because cost is negligible
- ❑ If you contribute to congestion, then you pay
  - congestion or shadow prices
- ❑ Simple example:
  - Consider a network that can cope with 500 emails per second
  - if 499 emails are sent, they are all free
  - if 501 are sent, a cost is incurred for ALL of them
- ❑ Goal is to set the prices to ensure a social optimum
- ❑ Basically, try to raise just enough money to expand the resources to cope with the excess demand, or reduce demand to eliminate congestion
- ❑ Unfortunately, there are latencies in this :-)

GC5.1.14 – p.7/12

# Externalities

- ❑ Simplistic economics does not capture the total costs
- ❑ **Externalities** are the costs other people have to pay because of my resource usage, but which I don't pay directly
  - e.g. cost of treating elderly miners was borne by the NHS, not the people who bought coal
- ❑ Health externalities are easy to find, e.g. impact of passive smoking
- ❑ Global warming is another major source of examples
  - Some countries will vanish (e.g. Maldives) because of emission of greenhouse gasses by developed and developing nations
- ❑ Many externalities arise in the use of Computing Resources
  - cost of sending one ethernet packet is essentially nil
  - all of the infrastructure is paid for
  - additional electricity consumption is undetectable
  - but if it collides with another packet, both suffer

GC

# Shadow Prices — outline theory I

- ❑ Consider a single finite resource
- ❑ Consumer  $i$ 
  - attempts to maximise **utility**  $u_i(x_i)$ 
    - this is the benefit gained from consuming  $x_i$  of resource
  - is charged some cost  $C(\cdot)$  for the resource usage
  - maximise  $U_i(x_i) = u_i(x_i) - C(\cdot)$
- ❑ Social Planner
  - wishes to achieve a *socially optimal* resource allocation
  - maximise sum of consumers' utilities, minus cost of overall system load (externalities)
    - let  $y = \sum_i x_i$
    - maximise  $\sum_i u_i(x_i) - C(y)$
  - solving this, for convex  $C(y)$ , gives:
    - $u'_i(x_i) = p(y)$  with  $p(y) = C'(y)$
    - $p(y)$  is the shadow price of the load

GC

# Shadow Prices — outline theory II

- ❑ marginal benefit for consumer  $i$  matches the marginal cost to all consumers
- ❑ this is mathematically tractable, but:
  - requires knowledge of  $u_i(x_i)$
  - which is usually not explicitly known
  - especially not to an outside agent
    - lying about  $u_i(x_i)$  could give benefits!
- ❑ Decomposition:
  - consumer  $i$  is charged at a fixed rate  $t_i$ , in proportion to the amount of resource it receives
  - consumer maximises:
    - $U_i(x_i) = u_i(x_i) - t_i x_i$
  - solving this, for concave  $u_i(x_i)$ , gives:
    - $u'_i(x_i) = t_i$
  - if  $t_i = p(y)$  this is the same as the social optimum
- ❑ When prices are correct: **User optimum**  $\equiv$  **Social optimum**

GC5.L14 – p.9/12

# Pricing Multiple Resources

- ❑ Multiple independent resources can be shadow priced separately
  - but how does consumer divide funds?
- ❑ If resources are substitutable it's more complex
  - Continuous substitution is OK
  - Discrete switching of pieces of load is harder to cope with
    - in general case, think about bin-packing problems
- ❑ Economic theoreticians can produce more complex models
- ❑ Systems builders want simple mechanisms
  - e.g. for an indivisible resource:
    - *Vickrey Auction*:
      - highest bidder wins
      - price paid is second highest bid

GC5.L14 – p.11/12

# Problems with Shadow Pricing

- ❑ Assumes consumers are rational and competent
- ❑ Assumes perfect information about prices is available
- ❑ Assumes utility functions are concave/convex
- ❑ Assumes elasticity
  - too little of a resource may be pointless
  - may have step-shaped utility function
- ❑ Favours richer consumers
- ❑ Allows rich malicious user to wreak havoc
  - submit large jobs to busy machines, ignoring idle ones
  - essentially a form of denial-of-service attack
  - trying to price other users out of the system
  - since users are not omniscient, could succeed
- ❑ Only find out about congestion afterwards

GC5

# Negotiation

- ❑ Auction mechanisms, markets for computational commodities and other techniques from economics/business have been explored in the context of Grid Resource Management
- ❑ One particular aspect is negotiation. . .
- ❑ Job submission systems may wish to negotiate over price or resource combinations with multiple autonomous providers:
  - provider may not grant unconditional access to a resources
  - differences must be resolved
  - may be able to optimize use of resources/trade-off different resources (e.g. cash for CPU)
- ❑ service level agreements, for example, are usually negotiable but: people vs machine/agents
- ❑ Can we optimize utility, over multiple resources, given market dynamics
- ❑ May need to relax bargaining terms if the negotiation is proving difficult

GC5