Working Together

Working Together

- Why and how do agents work together?
- Important to make a distinction between:
  - *benevolent agents*
  - *self-interested agents*

Benevolent Agents

- If we “own” the whole system, we can design agents to help each other whenever asked
- In this case, we can assume agents are *benevolent*: our best interest is their best interest
- Problem-solving in benevolent systems is *cooperative distributed problem solving* (CDPS)
- *Benevolence simplifies the system design task enormously!*

Self-Interested Agents

- If agents represent individuals or organizations, (the more general case), then we cannot make the benevolence assumption
- Agents will be assumed to act to further their own interests, possibly at expense of others
- Potential for *conflict*
- May complicate the design task enormously

Task Sharing and Result Sharing

- Two main modes of cooperative problem solving:
  - *task sharing*:
    components of a task are distributed to component agents
  - *result sharing*:
    information (partial results, etc.) is distributed

The Contract Net

- A well known task-sharing protocol for *task allocation* is the *contract net*:
  1. Recognition
  2. Announcement
  3. Bidding
  4. Awarding
  5. Expediting

Recognition

- In this stage, an agent recognizes it has a problem it wants help with.
  Agent has a goal, and either…
  - realizes it cannot achieve the goal in isolation — does not have capability
  - realizes it would prefer not to achieve the goal in isolation (typically because of solution quality, deadline, etc.)

Announcement

- In this stage, the agent with the task sends out an *announcement* of the task which includes a *specification* of the task to be achieved
- Specification must encode:
  - description of task itself (maybe executable)
  - any constraints (e.g., deadlines, quality constraints)
  - meta-task information (e.g., “bids must be submitted by…”)
- The announcement is then *broadcast*

Bidding

- Agents that receive the announcement decide for themselves whether they wish to *bid* for the task
- Factors:
  - agent must decide whether it is capable of expediting task
  - agent must determine quality constraints & price information (if relevant)
- If they do choose to bid, then they submit a *tender*

Awarding & Expediting

- Agent that sent task announcement must choose between bids & decide who to “award the contract” to
- The result of this process is communicated to agents that submitted a bid
- The successful *contractor* then expedites the task
- May involve generating further manager-contractor relationships: *sub-contracting*

Issues for Implementing Contract Net

- How to…
  - …specify *tasks*?
  - …specify *quality of service*?
  - …select between competing offers?
  - …differentiate between offers based on multiple criteria?

The Contract Net

- An approach to *distributed problem solving*, focusing on task distribution
- Task distribution viewed as a kind of contract negotiation
- “Protocol” specifies *content* of communication, not just form
- Two-way transfer of information is natural extension of transfer of control mechanisms

Cooperative Distributed Problem Solving (CDPS)

- Neither global control nor global data storage — no agent has sufficient information to solve entire problem
- Control and data are distributed

CDPS System Characteristics and Consequences

- Communication is slower than computation
  - loose coupling
  - efficient protocol
  - modular problems
  - problems with large grain size

More CDPS System Characteristics and Consequences

- Any unique node is a potential bottleneck
  - distribute data
  - distribute control
  - organized behavior is hard to guarantee (since no one node has complete picture)

Four Phases to Solution, as Seen in Contract Net

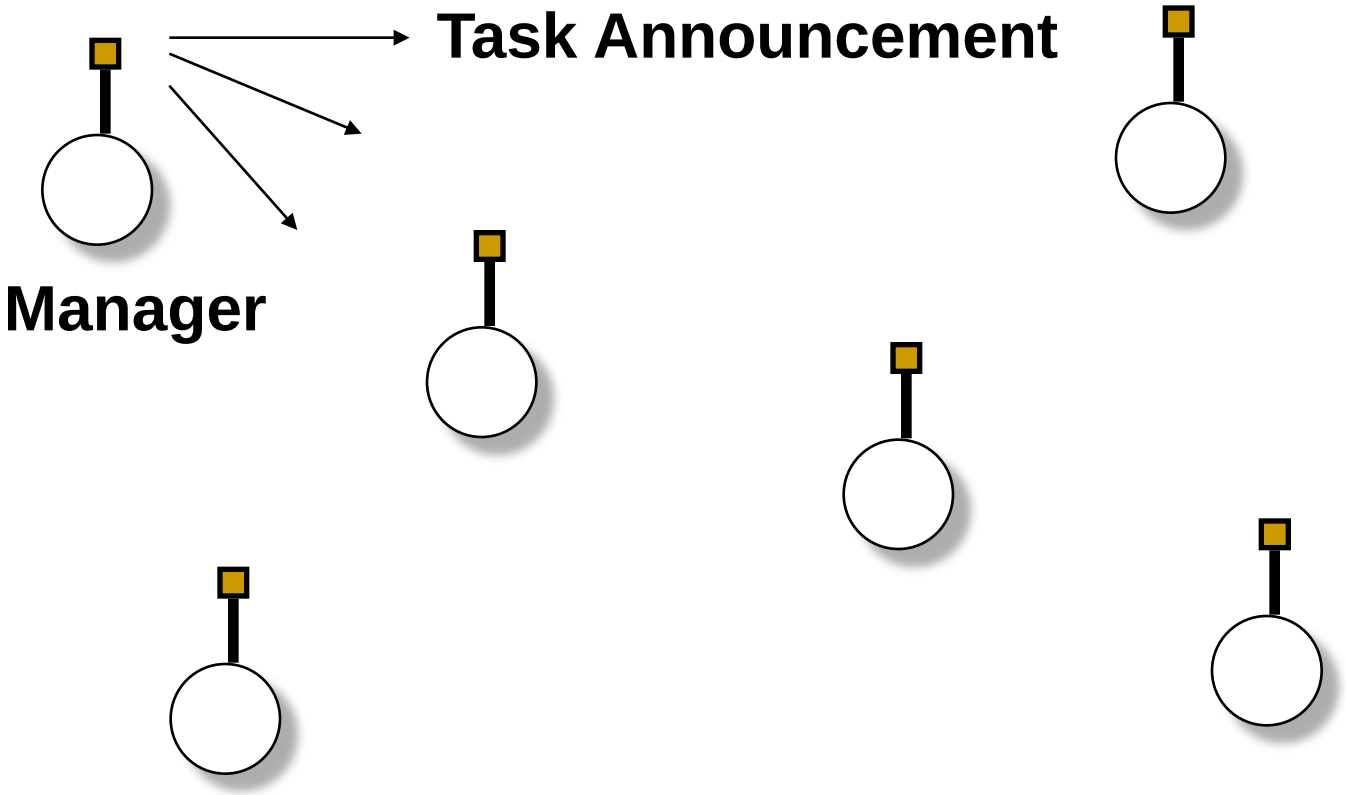
1. Problem Decomposition
2. Sub-problem distribution
3. Sub-problem solution
4. Answer synthesis

The contract net protocol deals with phase 2.

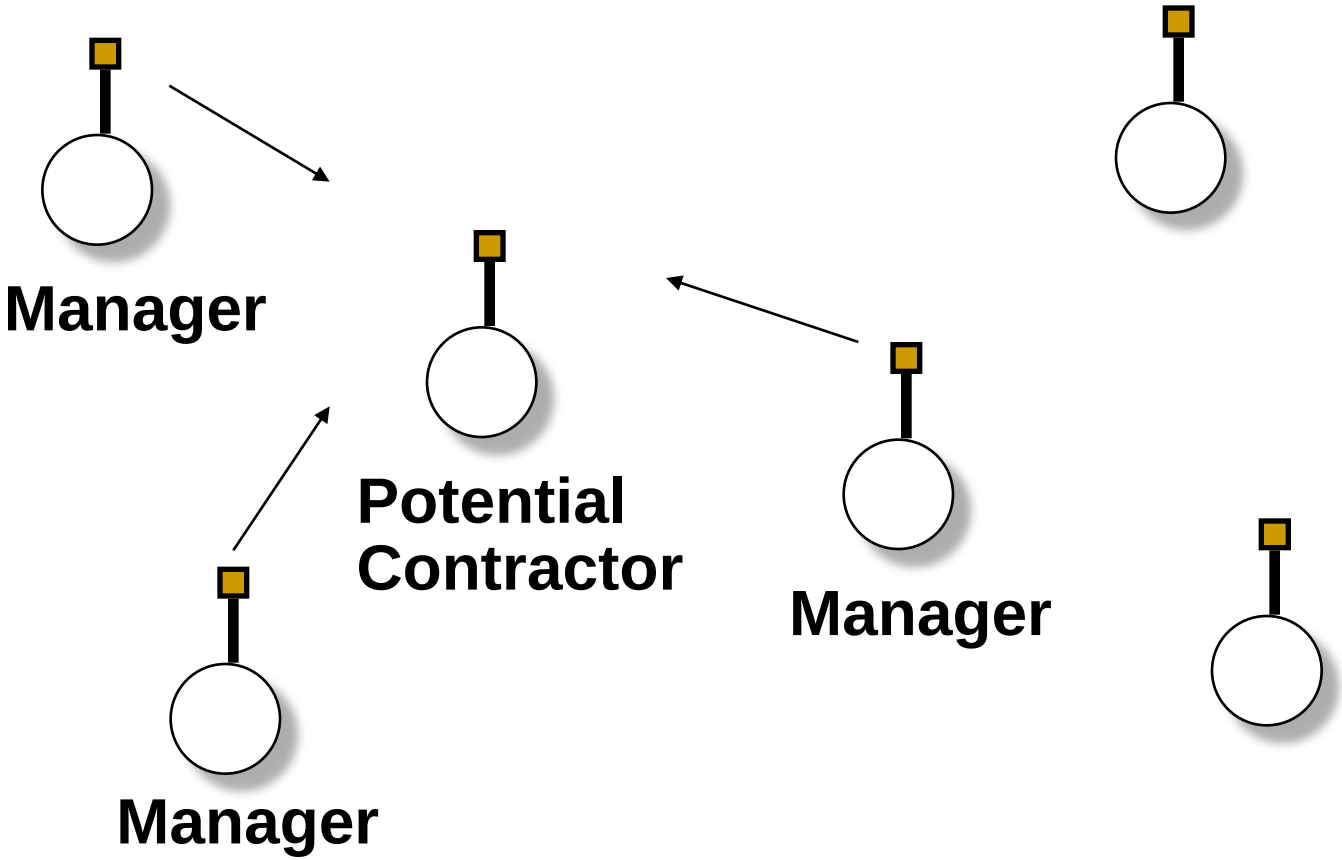
Contract Net

- The collection of nodes is the “contract net”
- Each node on the network can, at different times or for different tasks, be a manager or a contractor
- When a node gets a composite task (or for any reason can’t solve its present task), it breaks it into subtasks (if possible) and announces them (acting as a manager), receives bids from potential contractors, then awards the job (example domain: network resource management, printers, …)

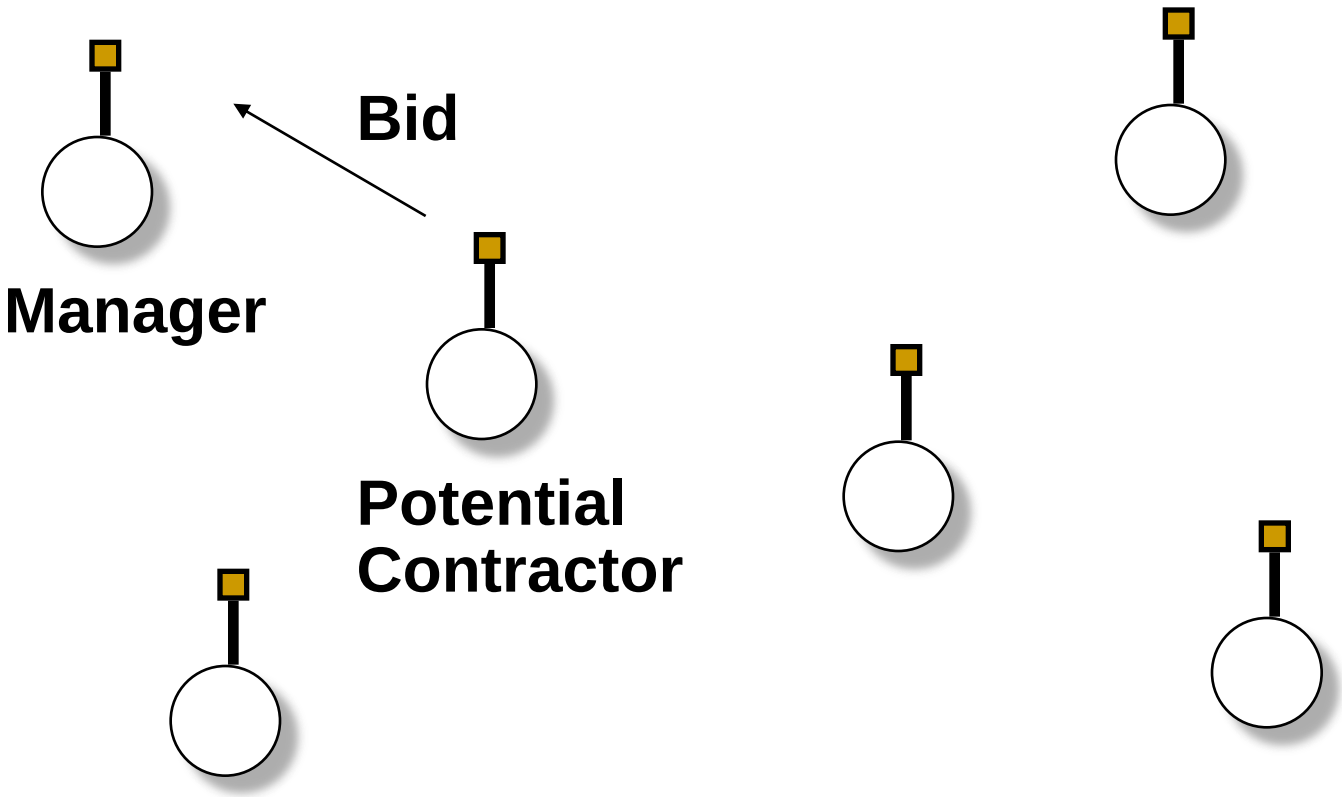
Node Issues Task Announcement



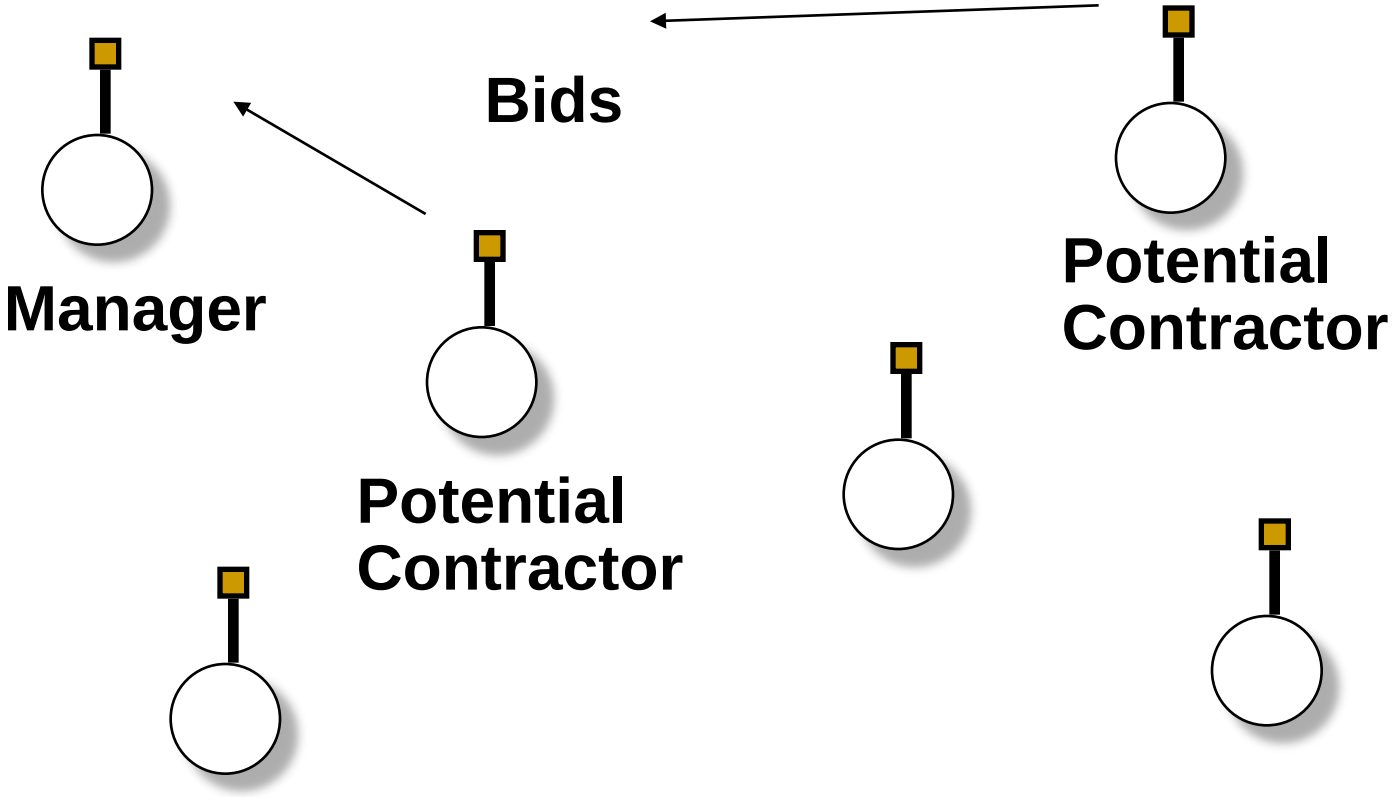
Idle Node Listening to Task Announcements



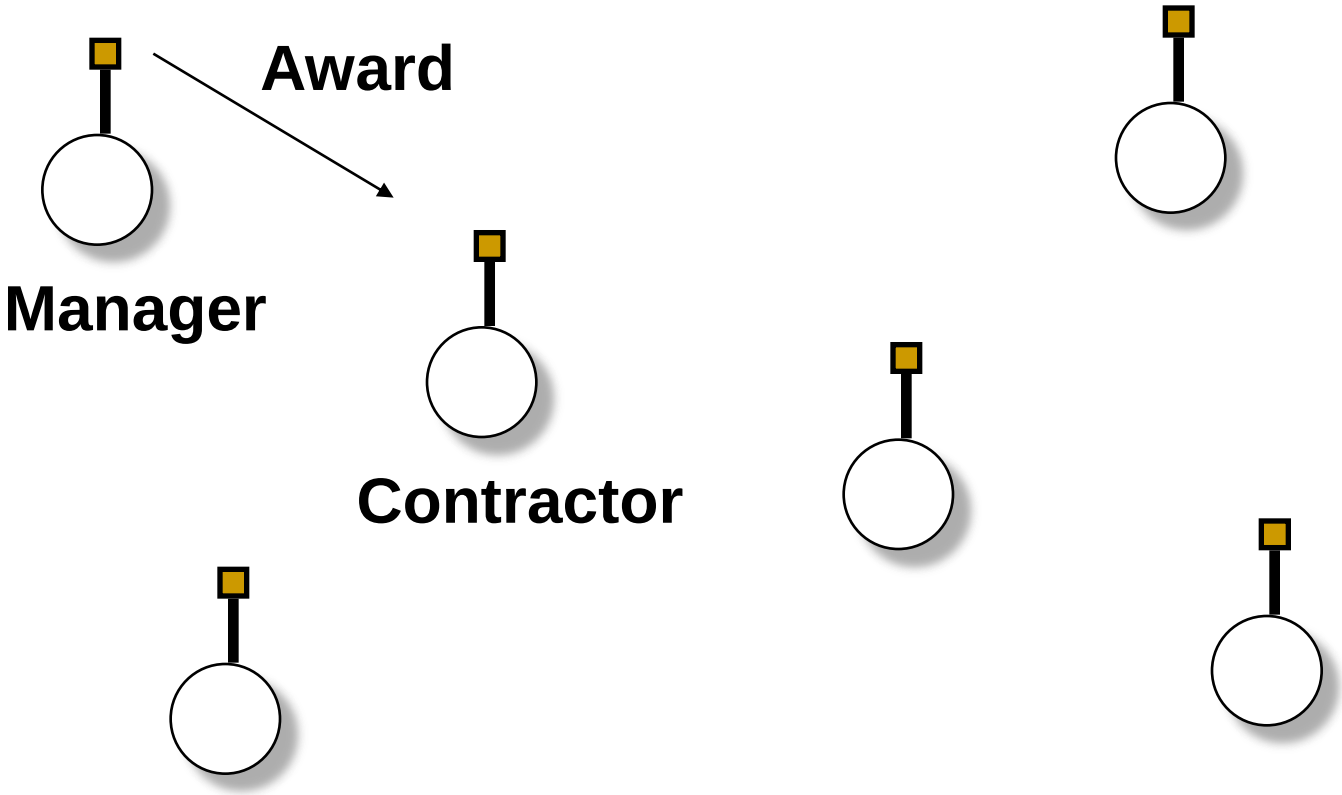
Node Submitting a Bid



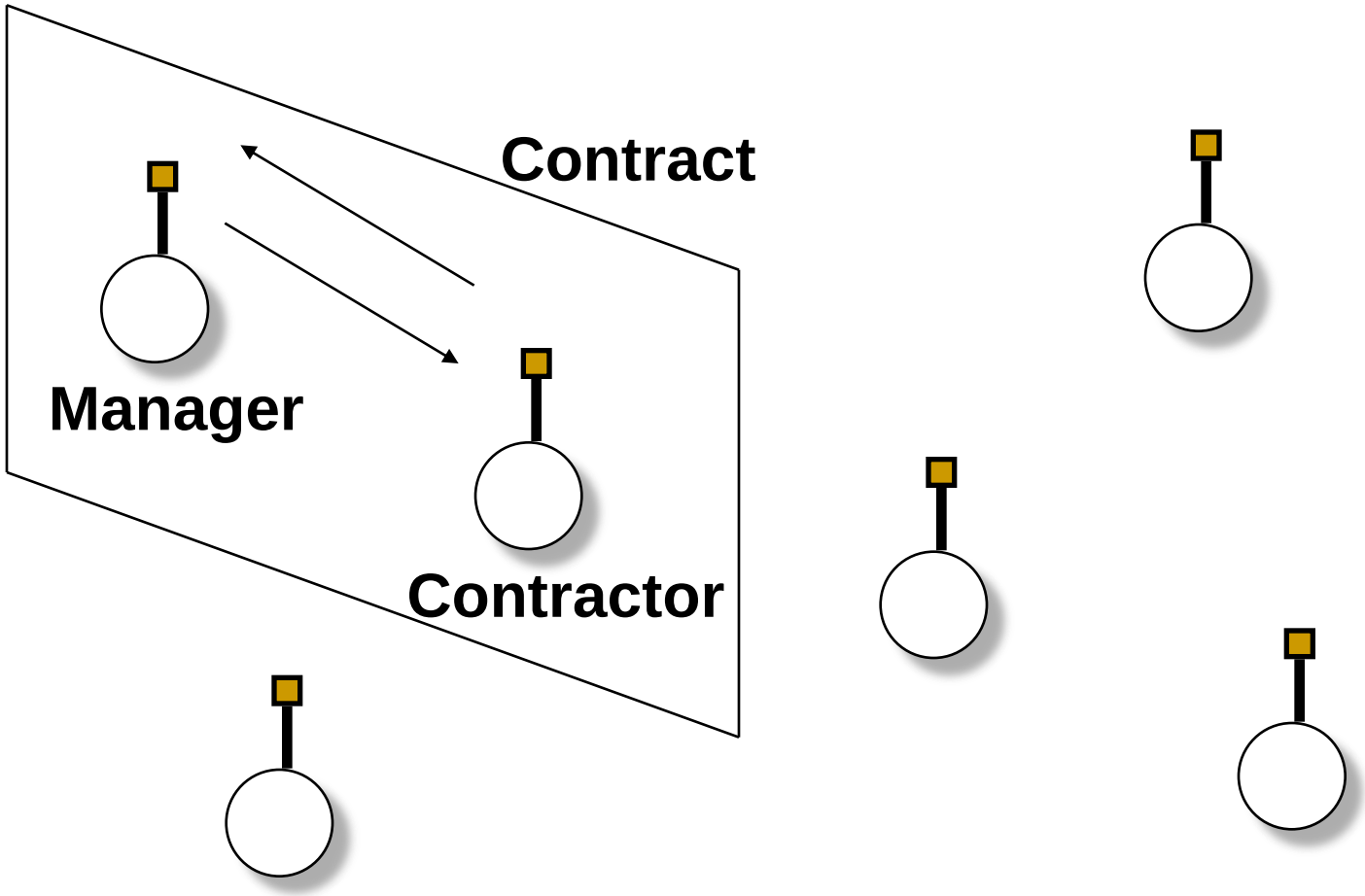
Manager listening to bids



Manager Making an Award



Contract Established



Types of Messages

- Task announcement
- Bid
- Award
- Interim report (on progress)
- Final report (including result description)
- Termination message (if manager wants to terminate contract)

Efficiency Modifications

- Focused addressing — when general broadcast isn't required
- Directed contracts — when manager already knows which node is appropriate
- Request-response mechanism — for simple transfer of information without overhead of contracting
- Node-available message — reverses initiative of negotiation process