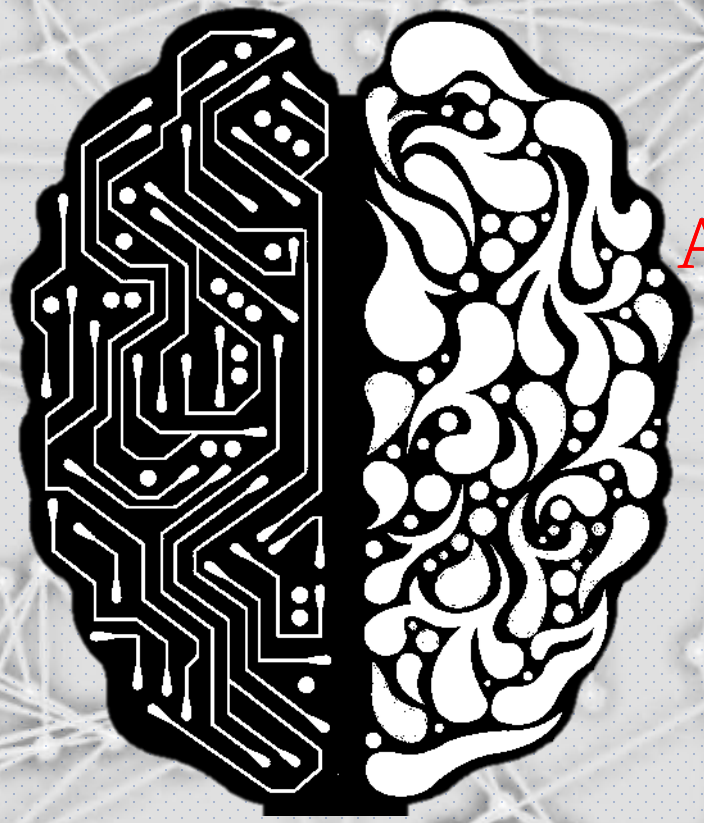


Multi-Agent Systems

Lecture III

- Dr. Nestor Velasco Bermeo,

- Researcher CONSUS (Crop OptimisatioN through Sensing, Understanding & viSualisation),
- School of Computer Science
- University College Dublin (UCD)

Lecture III Learning Objectives

- ❑ To understand the elements and principles of Agent Coordination.
- ❑ To understand the principles of Cooperative Problem Solving.
- ❑ To understand the common forms of coordination.
- ❑ Review the formalization of coordination.
- ❑ Identify the elements of the Contract Net Protocol

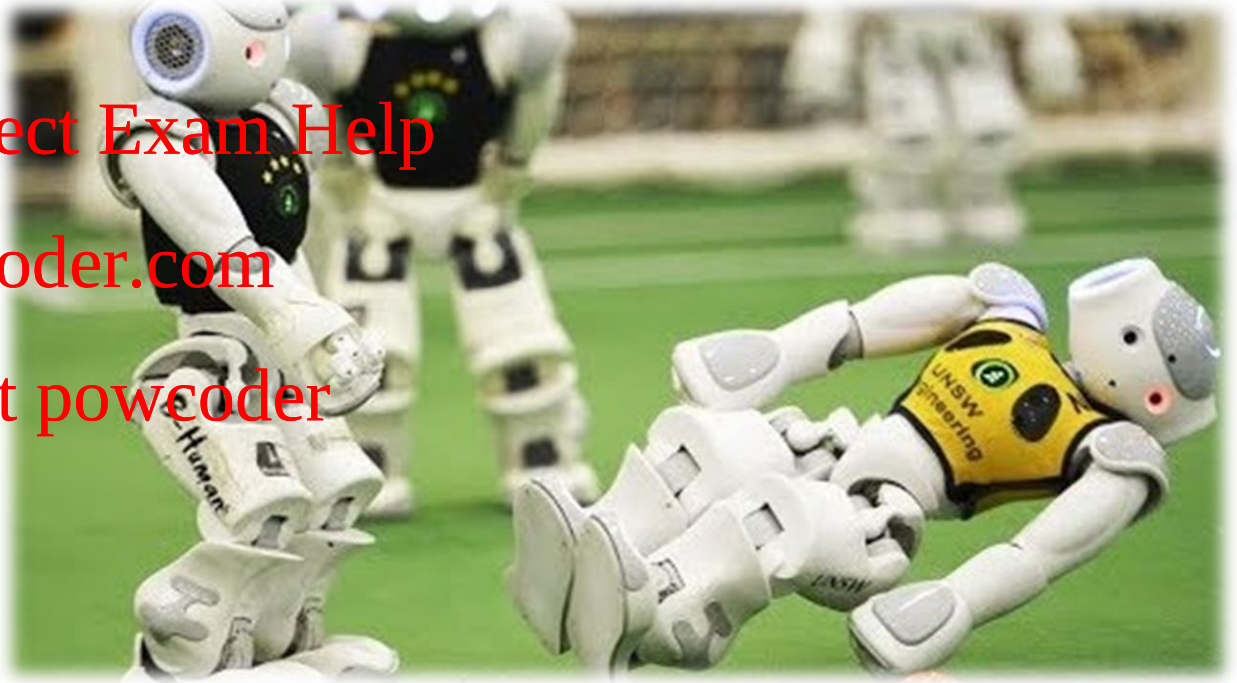
Coordination

"The process by which an agent reasons about its local actions and the (anticipated) actions of others to try and ensure that the community acts in a coherent manner."

Nick Jennings,1996

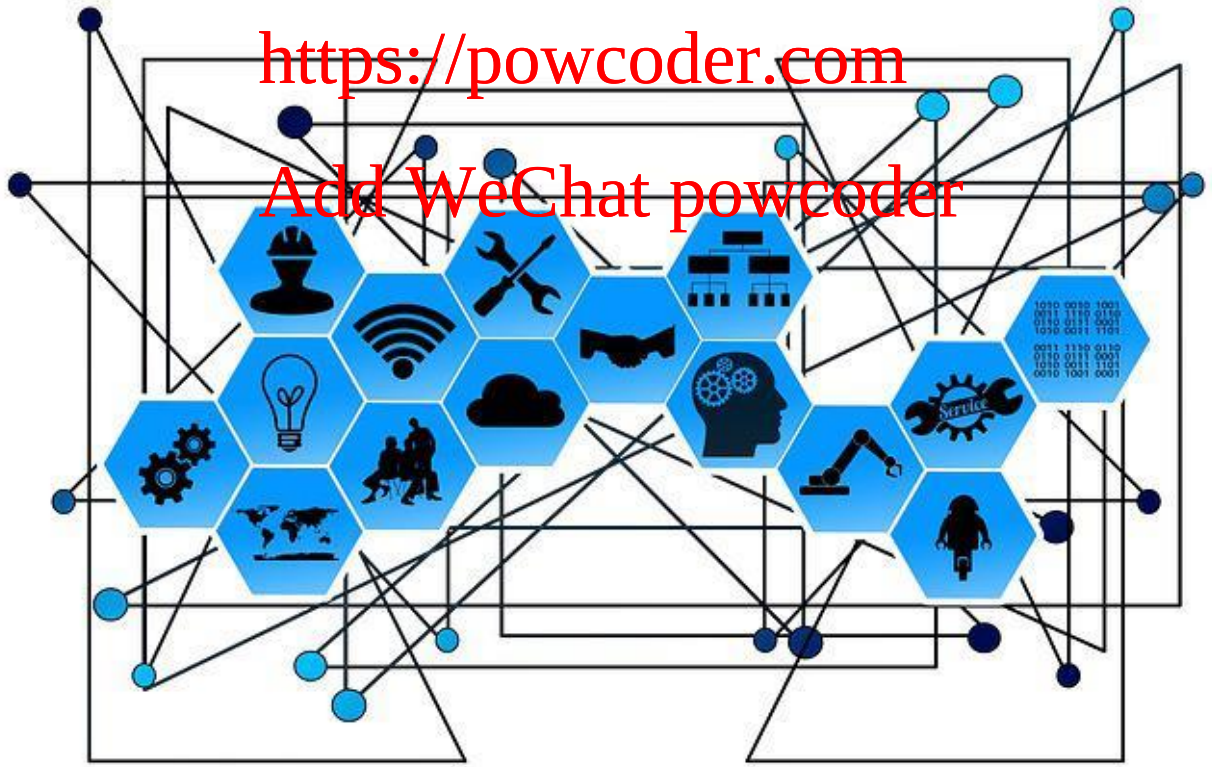
Why Coordinate?

- Prevent anarchy or chaos.
- Dependencies between agents' actions.
- Need to meet global constraints.
- Achieve agents' goals
- No individual has sufficient competence, resources or information to solve the entire problem.
- Efficiency



Coordination Elements

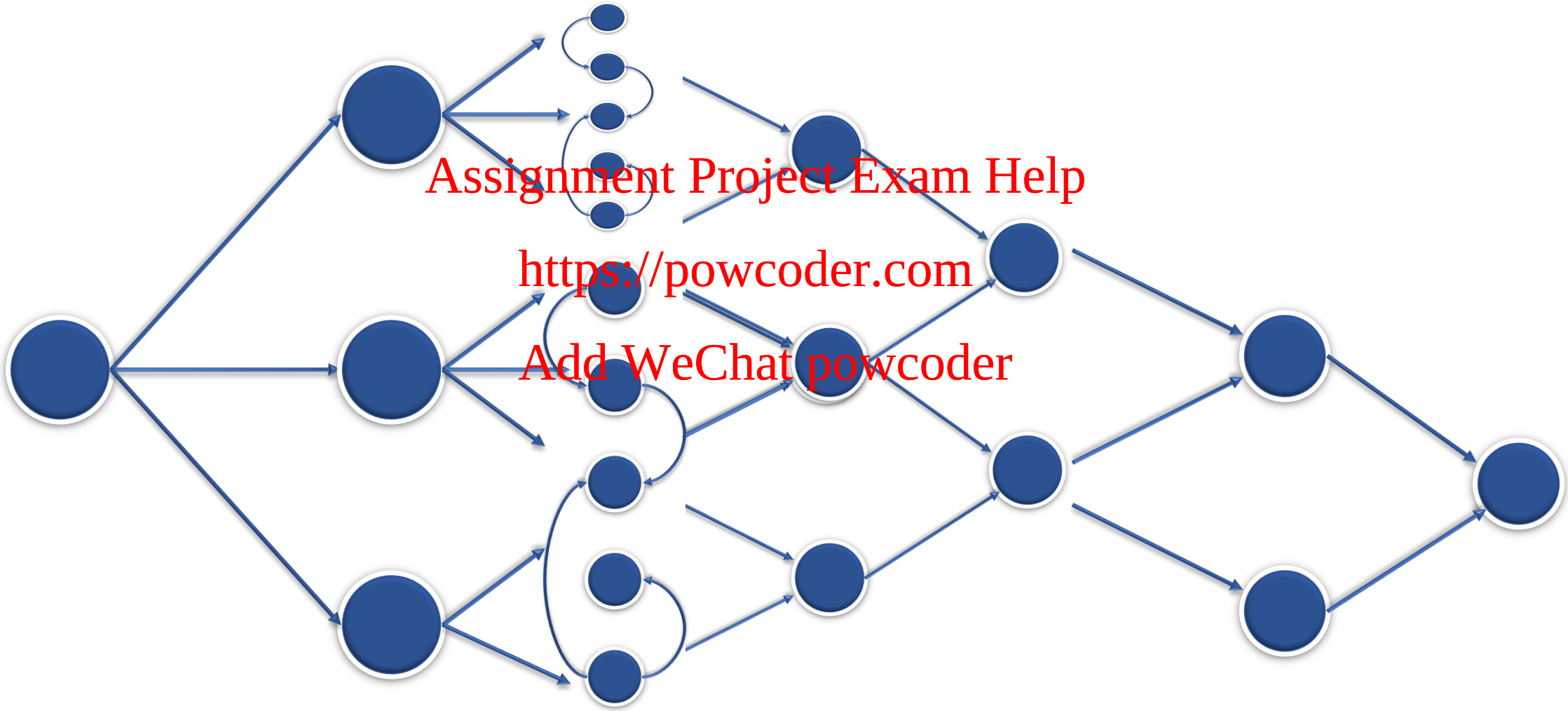
- The act of coordinating;
- Achieving proper order or a working relationship;
- Harmonious interaction as a function or part thereof;
- Autonomy;



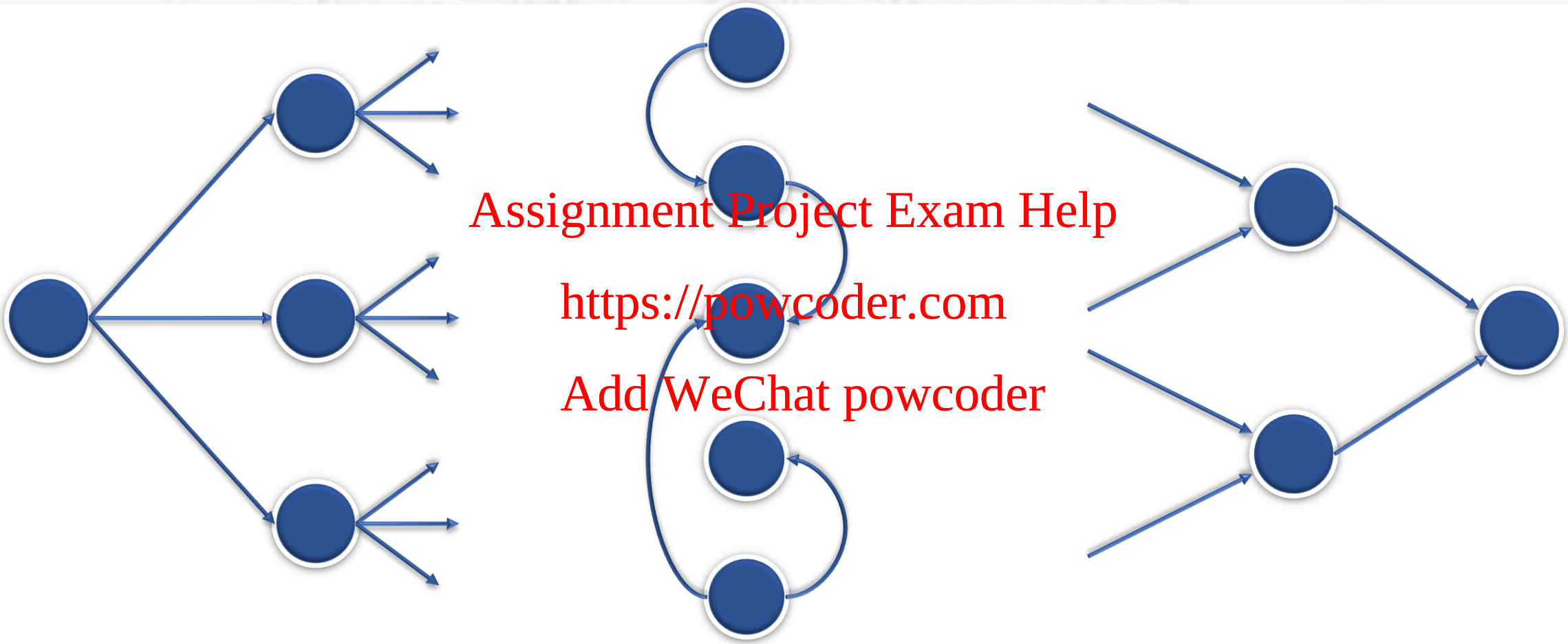
Chess vs Robocup

	Chess	RoboCup
Environment	Static	Dynamic
State Change	Turn taking	Real time
Info. accessibility	Complete	Incomplete
Sensor reading	Symbolic	Non-symbolic
Control	Central	Distributed

Cooperative Problem Solving



Cooperative Problem Solving



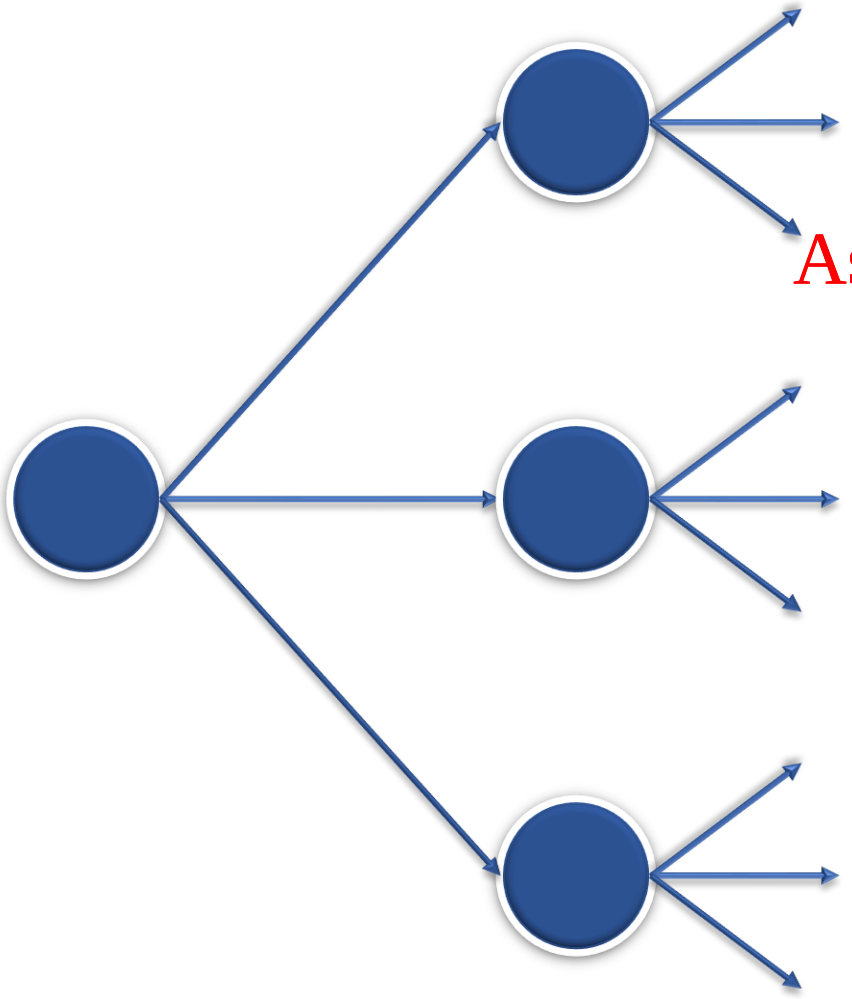
1
Problem decomposition

2
Subproblem solution

3
Answer synthesis

Cooperative Problem Solving

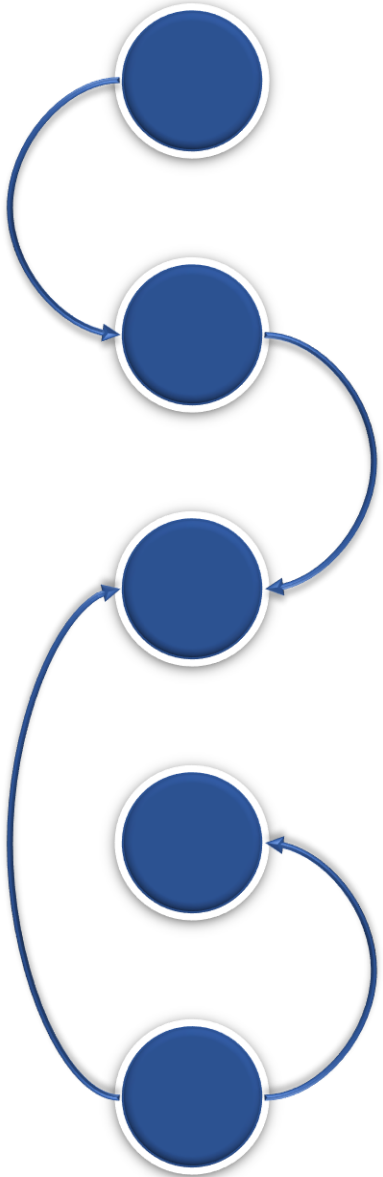
1. Problem decomposition



- How to break a problem down into a set of atomic sub-problems:
  - Identify a minimal set of **atomic operations** that can be performed by **individual agents**.
  - Repeatedly decompose the initial problem.
- Issues:
  - What is an appropriate level of granularity?
  - How to decompose the problem? Recursively?
  - Who should decompose the problem? (user, agents)
  - Who should solve the sub problems?

Cooperative Problem Solving

2. Subprobem solution

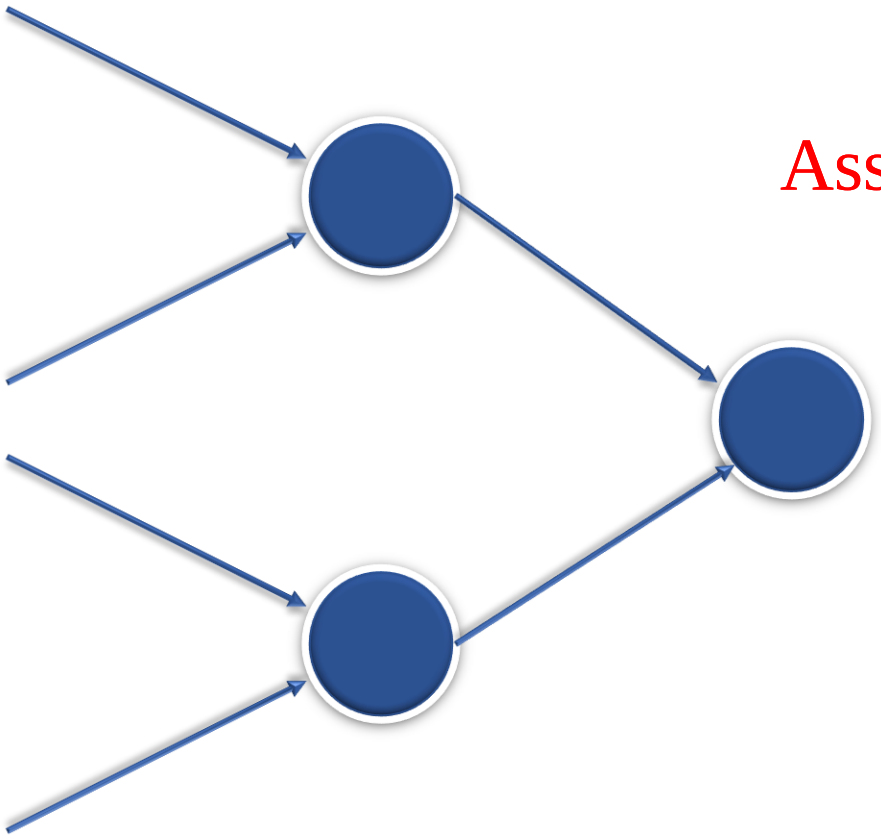


- Solving individual sub-problems.
- Sub-problems are allocated in the previous phase.
- Sub-problems may have dependencies.
- Agents assigned sub-problems may need to share information.

- Issues:
  - How to get help?
  - Who to share information with others?
  - What information to share?
  - Static / Dynamic decomposition.

Cooperative Problem Solving

3. Answer Synthesis

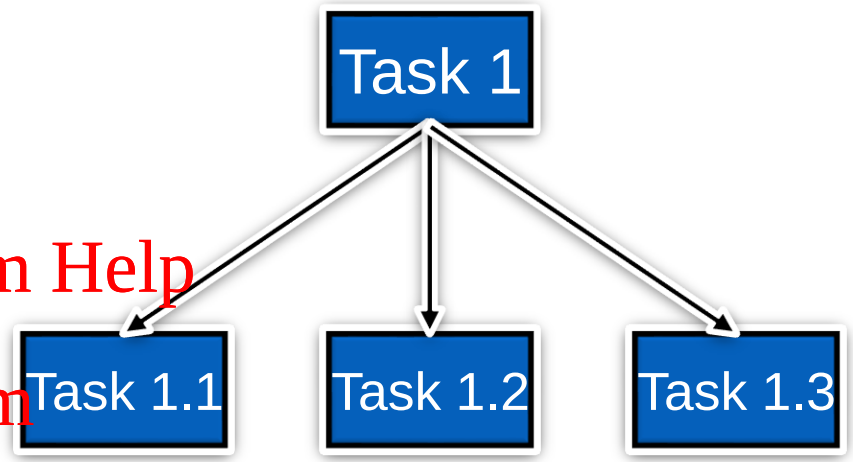


- Combining the results.
- Solutions to atomic sub-problems are combined to provide solutions to higher-level sub-problems.
- Partial solutions may be assembled during this process.

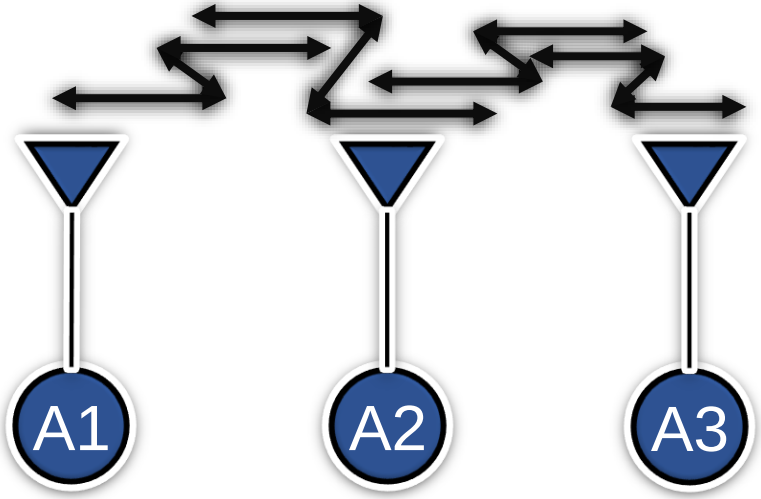
- Issues:
  - Who does the solution synthesis?
  - How are the sub-problems combined?
  - Where are the sub-problems combined?

How to Coordinate?

- There are two common forms of coordination:
- **Task sharing:**
  - When a problem is decomposed into subproblems and allocated to different agents.



- **Result sharing:**
  - When agents share information relevant to their subproblems.



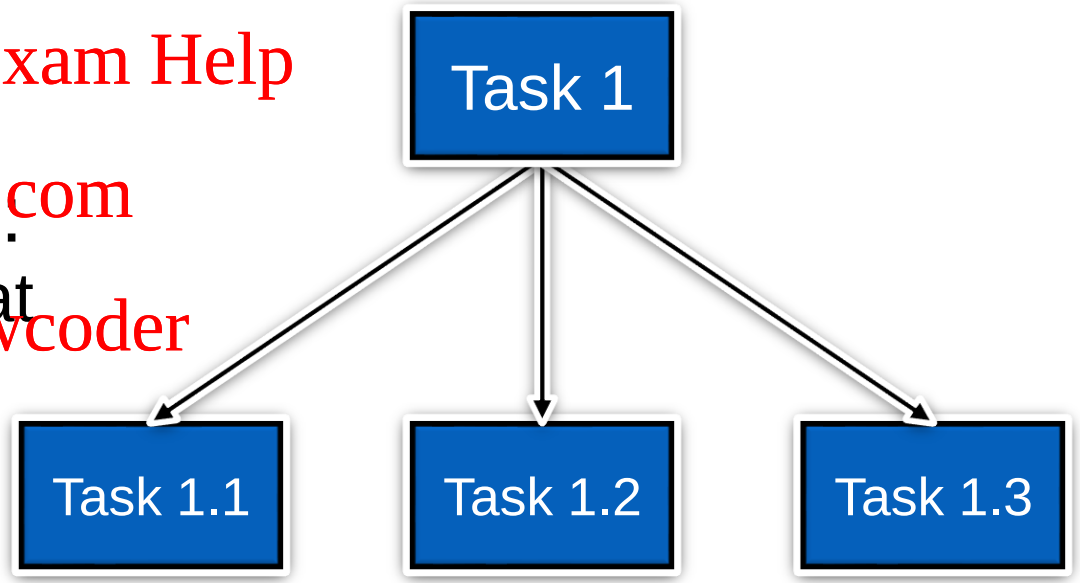
Task Sharing

The agent decomposes the task into a set of sub-tasks that are assigned to agents:

- The tasks are assigned dynamically at run-time based on agents capabilities.

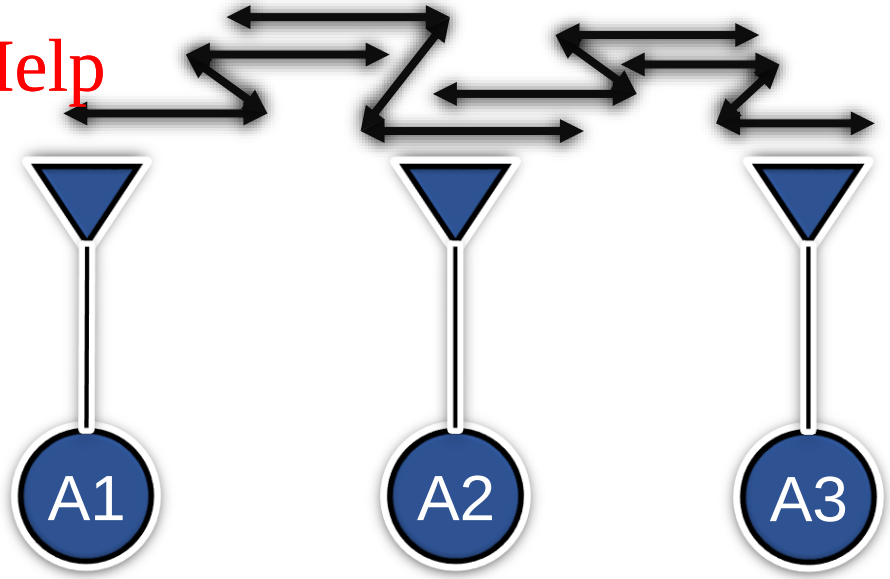
The task allocation process can be done:

- through a central coordinator (manager) that is authorised to assign tasks to worker agents.
- through some form of coordination mechanism that allows peers to reach agreement as to who will do what (e.g. negotiation, distributed planning, ...).



Result Sharing

- Problem solving proceeds by agents cooperatively exchanging information as the solution is developed.
  - The set of tasks are pre-assigned at design time
- Results may be shared:
  - **proactively** - one agent sends another agent some information because it believes that the other will be interested in it.
  - r**eactively** – an agent sends information to another in response to a request.



Result Sharing

- A Result Sharing system's performance can take advantage of:

- **Confidence**: Higher confidence in a result if multiple agents independently derive it.
- **Completeness**: Sharing ***local views*** can help to achieve a better ***global view***.
- **Precision**: Individual agents can refine their result based upon the results of the other agents.
- **Timeliness**: If the agents work together to formulate the solution in parallel, the result can be attained quicker.