1. Agents and Multiagent Systems

   a. Intelligent Agents
      i. “An agent is a computer system that is situated in some environment and that is capable of autonomous action in this environment in order to meet its design objectives” (Wooldridge, 2002)
      ii. “Intelligent agent adds reactive, proactive and social to agent.” (Wooldridge, 2002)
         1. Reactive
            a. responds to events (user, agent, system) in environment
            b. perception (sensors) and action (effectors)
         2. Proactive
            a. Persistent goals
            b. Goal-directed reasoning through if-then rules
         3. Social
            a. Shared knowledge representation between agents (KIF)
            b. Shared communication between agents (KQML)
            c. Rules for negotiation and team-work
   iii. Other Attributes
         1. Robust by being flexible
         2. Rational – problem solving, reasoning, planning
         3. Capable of learning

   b. EXAMPLE: RoboCup soccer <www.robocup.org>
      i. Goal: By 2050 develop a team of fully autonomous humanoid robots that can win against human world champion team.
      ii. Technologies: design principles of autonomous agents, multi-agent collaboration, strategy acquisition, real-time reasoning, robotics, and sensor-fusion. RoboCup is a task for a team of multiple fast-moving robots under a dynamic environment.
      iii. ISIS team at USC
            1. Planning occurs jointly between agents: mutual goal
            2. Explicitly models teamwork
            3. Communication includes requesting joint goals & confirmation
iv. CMUnited at CMU
   1. Agents are completely autonomous
   2. Agent trained (learning through neural nets, etc) to position itself to
      intercept ball plus set plays. Expected behavior of opponents is
      modeled probabilistically as hidden Markov models

   c. Intelligent Agent Communication
      i. Blackboard
         1. Shared data between agents (global)
         2. Tightly couples agents and application for modularity
      ii. Common Language
          1. Knowledge Query and Manipulation Language (KQML)
          2. Allows independent agents to communicate
             a. Types of language acts: performatives (messages),
                representatives (state facts), commissives (promise or threat)
             b. Uses ontologies to interpret meaning
             c. Requires a KQML interpreter (or matchmaker) to enhance
                communication between multiple agents.
      iii. Negotiation protocols
          1. Needed for e-commerce applications
          2. Contract Net protocol
             a. Manager agent calls bids to complete a task (announcement)
             b. Contractor agents evaluate announcement and optionally
                respond with bids
             c. Manager agent selects one of the bids
             d. Contractor agent performs task and returns results
          3. Auction
             a. Seller initiates auction and monitors the process while buyers
                respond with offer to auctioneer
             b. Strict rules on selection of buyer and behavior of auctioneer
          4. Bargaining
             a. Most complicated form of negotiation
             b. Proposals & counter-proposals offered back and forth until
                both parties agree

   d. Intelligent Agent Standards
      i. BDI (Belief, Desire, Intention) model
         1. Agents have mental states:
            a. beliefs about other agents and state of the world
            b. desires and plans for taking action
      ii. Foundation for Intelligent Physical Agents FIPA
         1. Computer and telecommunication companies
         2. Agent-level issues: Agent platform
            a. Agent + agent management system (controls life cycle) +
               directory facilitator (yellow-page lookup service) + message
               transport systems (inter-agent communication)
            b. Several agent platforms communicate through the message
               transport system
iii. **Object Management Group (OMG) Agent Working Group**
   1. Standards body that created CORBA
   2. Object-level interoperability and management

2. **IA Framework**

   a. **General Software Engineering issues**
      i. Integration of agents with traditional information technology
         1. Database access, communications, etc.
         2. Common programming language like C++ and JAVA
      ii. **Use IA standards**
      iii. Multi-threaded systems give concurrency
      iv. **Define agents by roles: can change roles**

   b. **Functional specs for Java IA application**
      i. Easy to add IA to existing application (Java)
         1. Agent runs in separate thread
         2. Use events to communicate between application and agent (rather than direct invocations of methods)
      ii. Compose agents out of other Java components and other agents
         1. Java.beans package
         2. Reuse agents, specialize
      iii. Event processing
         1. JDK Delegation Event Model
            a. Sources and listeners
            b. Agent is an EventListener
      iv. Domain knowledge represented as If-then rules, forward-backward chaining
      v. Agents do classification, clustering and prediction (learning algorithms)
      vi. Multi-agents supported by KQML message protocol
      vii. Agents are persistent
         1. save state in a file and reload later
         2. java.io package supports Serializable for saving and loading Java objects
c. **CIAgent Architecture**

i. 

![CIAGent UML class diagram](image)

*Figure 7.2* The CIAGent UML class diagram.

ii. CIAGentTimer and EventQueue handle asynchronous events

1. Creates autonomous agent
2. CIAGentEvents allow communication between agents

iii. Modify If-then rule processing

1. Adds ability to do procedures in addition to basic rule-processing
2. allow external sensor input to antecedent of rule
3. trigger actions (other than add fact to database)