Software Agents and Multi-Agent Systems

Keith S. Decker
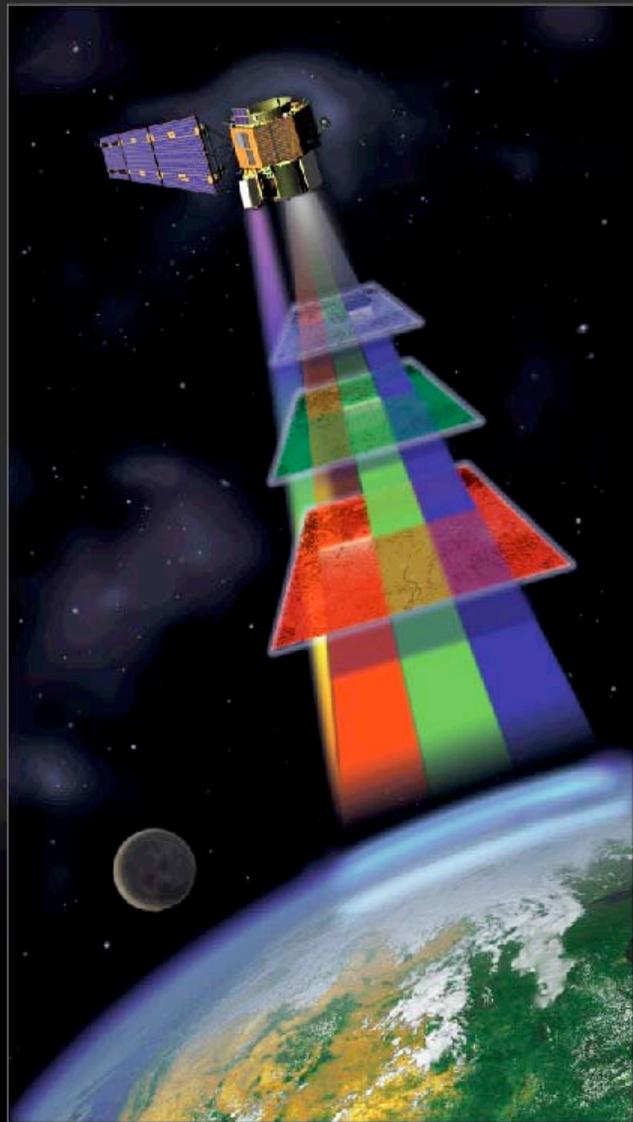Department of Computer Science
University of Delaware

What is a Software Agent?

- Autonomous & Persistent: The main point about agents is that they are capable of acting independently, exhibiting control over their internal state.
  - Trivial (non-interesting) agents:
    - Software thermostats, UNIX daemons (e.g. biff)
- An intelligent agent is a persistent computer system capable of flexible autonomous action in some environment.
  - "flexible" means
    - Reactive (expect change, failure)
    - Pro-active (achieve goals via multiple means)
    - Social (multi-agent systems)
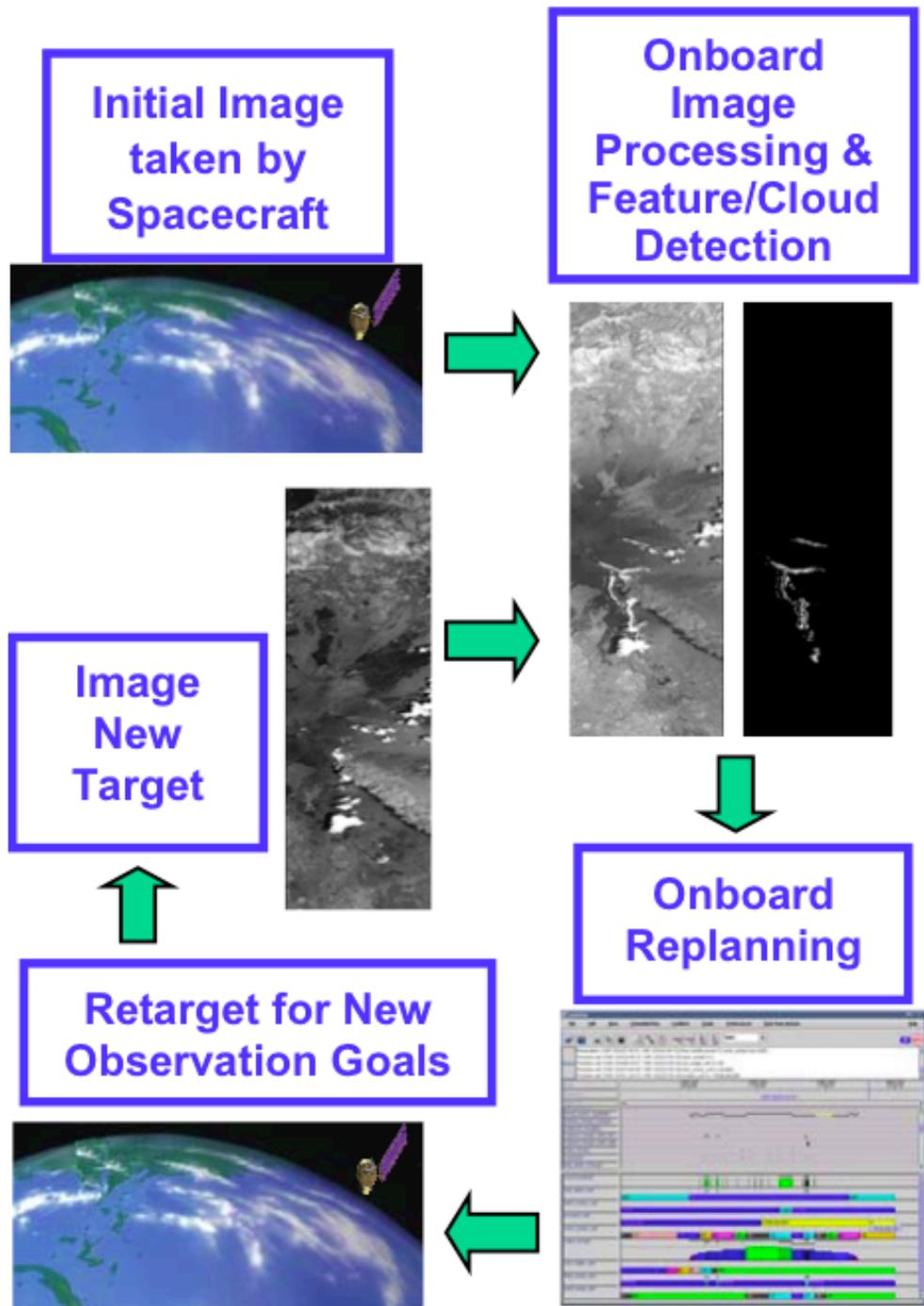
Example: Deep Space One

Earth Observing -1

A Clearer Picture of Our Changing Planet

Initial Image taken by Spacecraft

Onboard Image Processing & Feature/Cloud Detection

Image New Target

Onboard Replanning

Retarget for New Observation Goals

Multi-Agent Systems

- Natural problem distributions

- Concurrent speedups

- Increased reliability/robustness

- Bounded/Limited rationality

- Human organizational contexts

Hospital Scheduling
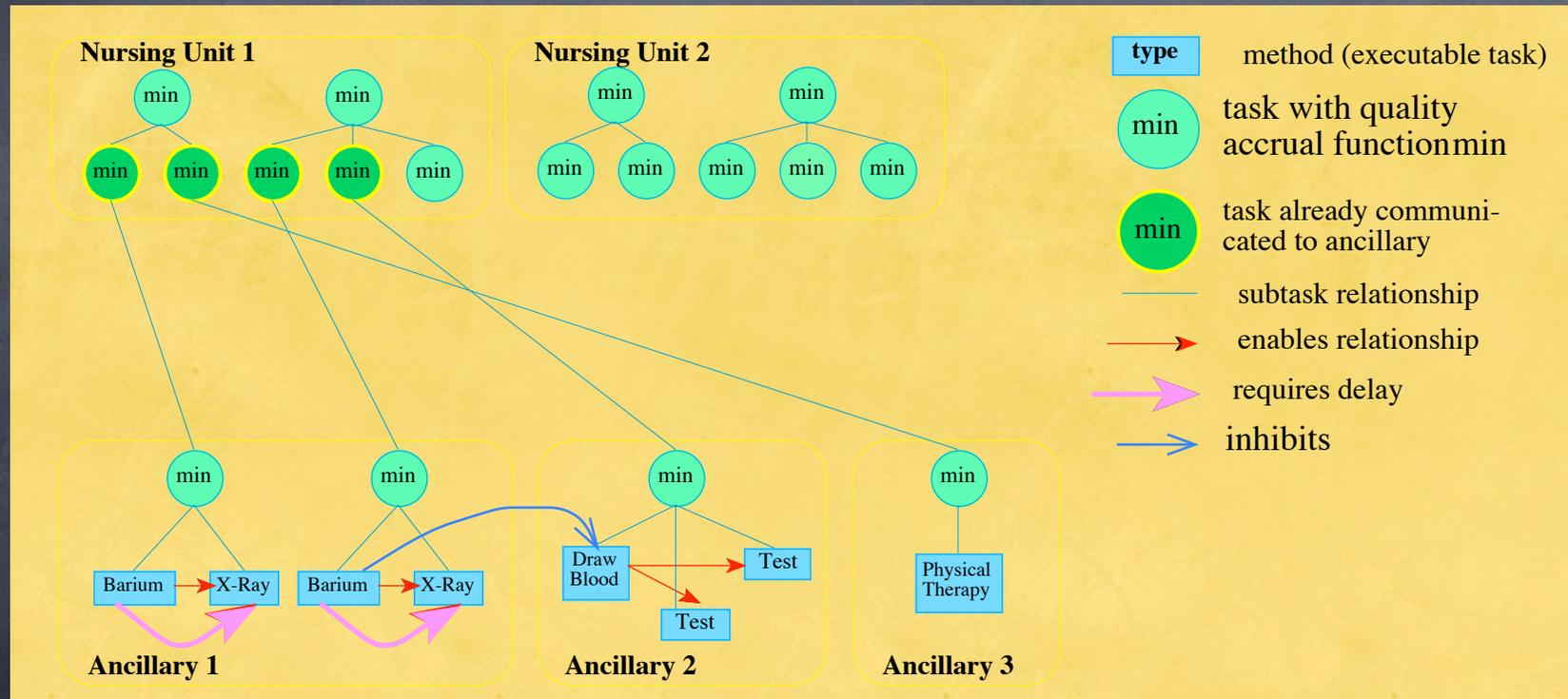
Distributed Computing vs. Distributed AI Viewpoints

- Distributed Computing
  - Tightly coupled, parallelization, centralized control
  - [Distributed OS] Independent processes, load balancing
  - Total database consistency
- Distributed AI
  - Loose coupling, distributed control
  - Interdependent processes
  - "Functionally Accurate" (often inconsistent)

Key Drivers for Agents [Jennings]

- Open Systems
  - Entities not known in advance and can freely enter and leave system at run time (e.g. Internet)
- Complex Distributed Systems
  - Industrial-strength software is difficult to build, even with modern software engineering advances
  - Agent concept is a new abstraction for system builders
- Ubiquitous systems
  - Presently, too much onus on user, not computer
  - Make it more of an equal partnership
  - Machine should not just be a dumb receptor of tasks
    - "future of computing will be 100% driven by delegating to, rather than manipulating, computers" (Negroponte 1995)

Designing Intelligent Agents & Organizations that:

- Operate in environments with uncertainty, deadlines
- Have multiple, possibly +/- interacting goals/ objectives
- Need to satisfice, not optimize
  - produce results that vary in quality depending on time pressure
- Interact with other agents
  - non-independent subproblems
  - partially overlapping goals/objectives

Research Agenda

- Representing and reasoning about these environmental features
  - Distributed Planning & Scheduling (TÆMS) [Gang]
  - Multi-agent Coordination (GPGP) [Wei]
- Software agent architectures and organizations that embody these solutions, that adapt in dynamic environments
  - DECAF (earlier, RETISINA)
- Information Gathering Systems based on agent models
  - Bioinformatics [Gang, Kay, Li, Sachin, Morgan]
  - Text integration [Terry]
- Understanding human organizational models computationally
  - Economically-oriented Organizational Behavior [Foster]
  - Organization Formation [Sachin]
  - Organizational Policy interaction

Coordinating Computational Actions

- Primary difficulties in CHOOSING and TEMPORALLY ORDERING actions
  - incomplete view of the problem
  - dynamically changing situation
  - uncertainty in the outcomes of actions
- Overcome difficulties with Coordination Mechanisms
  - schedules, plans, timelines, appointments, commitments
  - laws, rules, social behavioral norms
  - organizations, roles, negotiated order
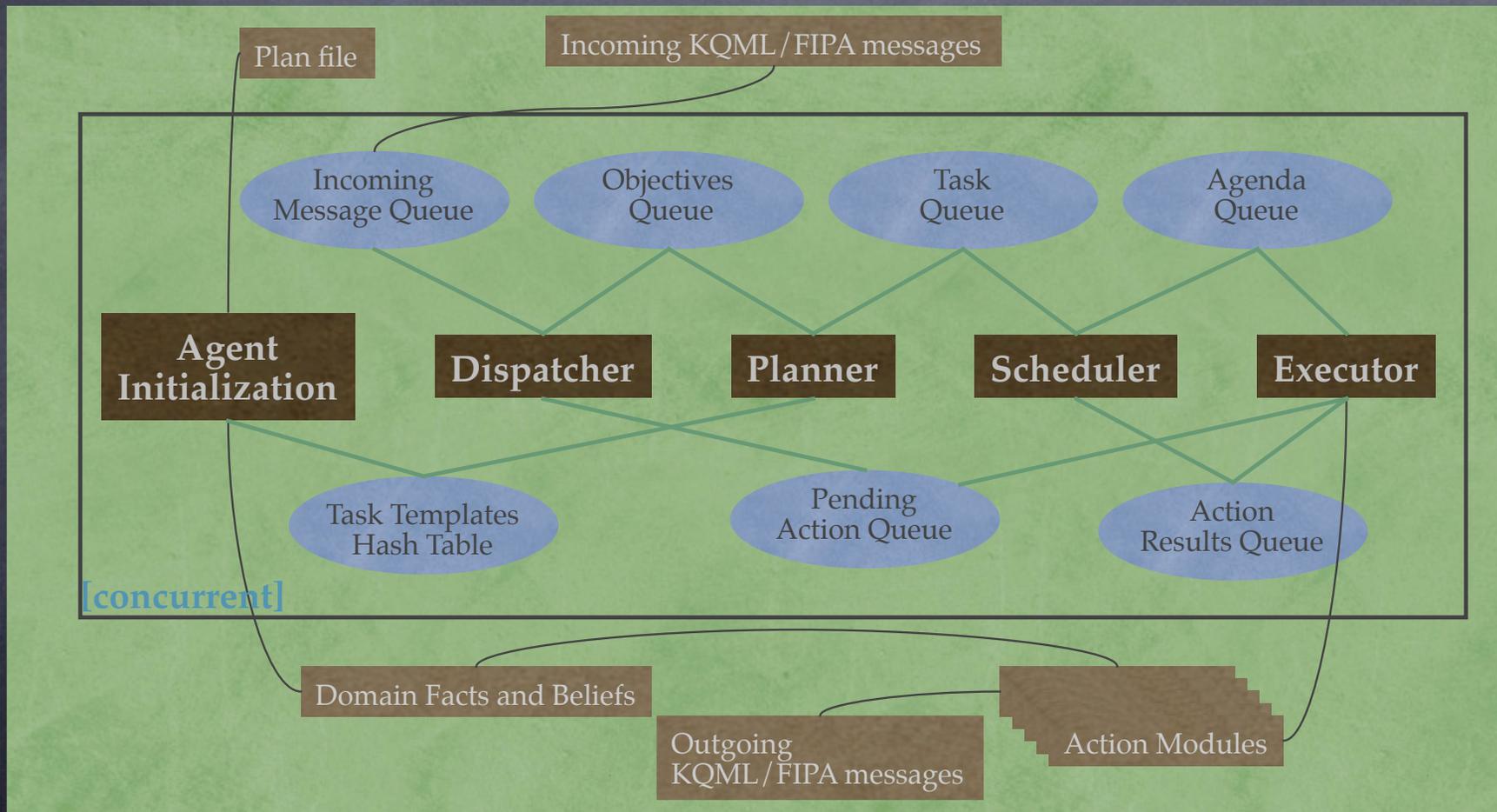
TÆMS Task Structure Representation

- Representing complex domains
  - worth-oriented
  - time-oriented
  - distributed
  - uncertain
- Representing quantitative change in characteristics over which agents have preferences
  - quality
  - cost
  - duration vs. deadline
- State-based semantics
- Annotation for HTN style task networks

A Vision for Multi-Agent System Engineering

DECAF: Distributed, Environment Centered Agent Framework

- Focus on programming agents, not designing internal architecture

- Programming at the multi-agent level

- Value-added architecture

- Support for persistent, flexible, robust actions

DECAF Architecture

Plan file

Incoming KQML/FIPA messages

Incoming Message Queue

Objectives Queue

Task Queue

Agenda Queue

Agent Initialization

Dispatcher

Planner

Scheduler

Executor

Task Templates Hash Table

Pending Action Queue

Action Results Queue

[concurrent]

Domain Facts and Beliefs

Outgoing KQML/FIPA messages

Action Modules

Task Structure [TÆMS]

- Multiple ways of achieving goal

  - And, Or, Sum, Xor

  - Schedule/execution time decision (not plan-time)

- Multiple outcomes can enable different downstream actions (contingencies, loops)

- Explicit representation of non-local tasks

GoFigure @ Univ of Delaware
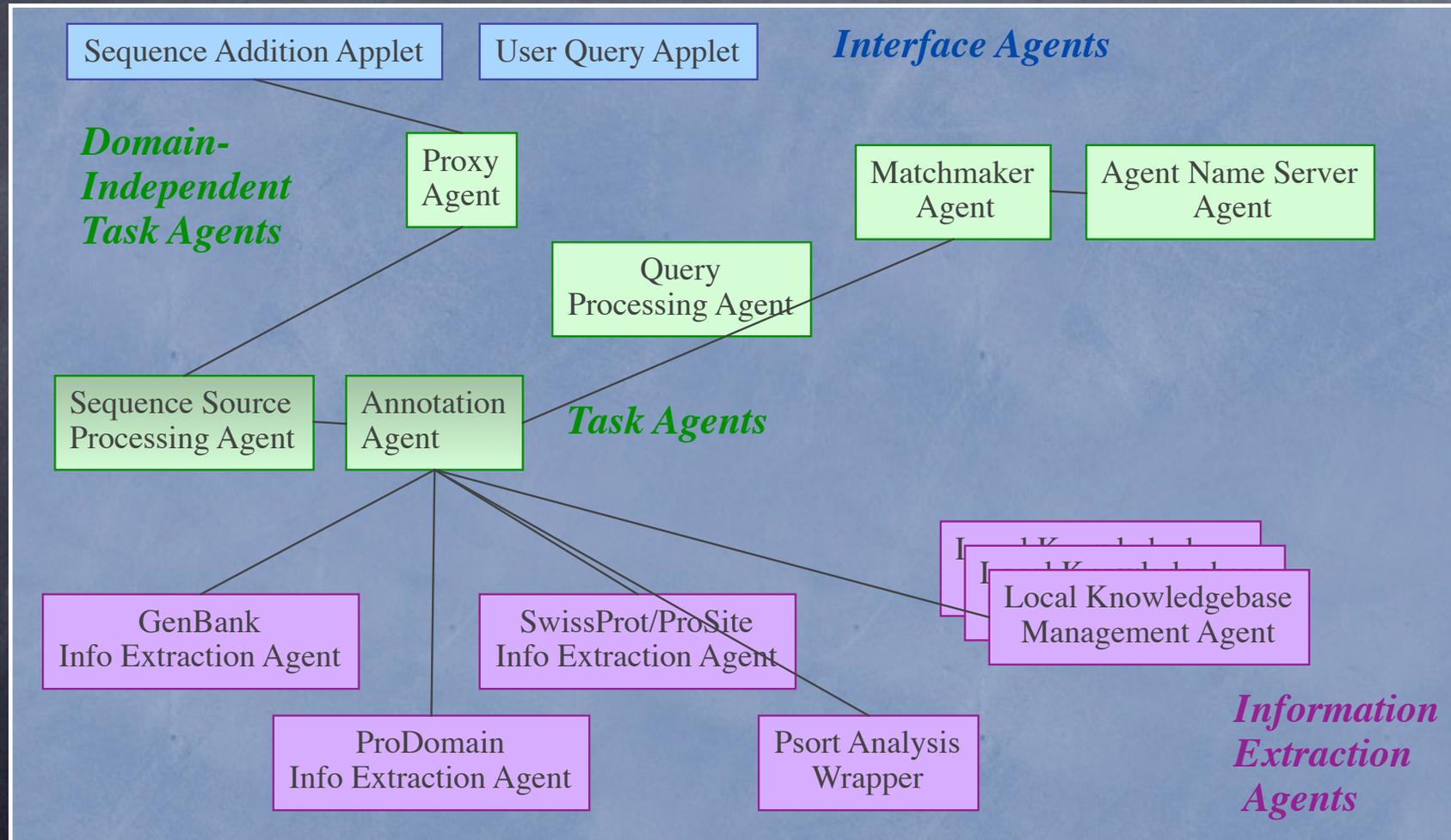
GOFIGURE!

a functional annotation tool using Gene Ontology

Annotator
Help

GoFigure

This form allows you to query your DNA or protein sequence against the GO annotated sequences from GO Consortium members' databases. If you wish to use this facility during a course, or if you have any problems or suggestions, then please contact us at situ@cis.udel.edu.

1. Input Sequence

Please enter your DNA or protein sequence (in FASTA format) into the text box below. For more information on the FASTA format, please click here.

○ DNA sequence (blastx)
● protein sequence (blastp)

```
>gi|P51355|pir||T30549 hensin - rabbit
MGISTVLALSLLWGPALSQGQWIPYTTYHDSVSSPGAPVETTTTEVFPSGL
ELRLANGGDRCQGRMEVLY
QGSWGTVCDDGWDINDAQVVCRQLGCGMAVSAPGSARFGQGPGQIVLD
DVSCSGQEPYLWSCHHRGWLSH
NCGHQEDAGVICSDAMMMTSPPPDTWPTTVMYESTPHFPSGLELVFPSGLE
LRLANGSDRCQGRMEVLYQ
GSWGTVCDDGWDINDAQVVCRQLGCGMAVSAPGSARFGQGPGQIVLDD
VSCSGQEPYLWSCHHRGWLSHN
CGHQEDAGVICSDAVPTTTPPPDTWPTTVMYESSPVFPSGLELRLANGSDR
```

Enter or cut and paste sequence here.

2. Choose Ontologies

Select the ontologies for which you would like the GoFigure displayed.

☑ Molecular Function
☑ Biological Process
☑ Cellular Component

3. Choose Weight Coefficients

In addition to the GoFigure graph, you can specify weight coefficients for factors such as e-value and evidence code, which are input to produce the GoDel automated annotation.

E-value [1.0] + Evidence [0.0] = 1

4. Email Address

You must enter your email address in the box below to use this service. Email addresses in the standard form, i.e. user@udel.edu.
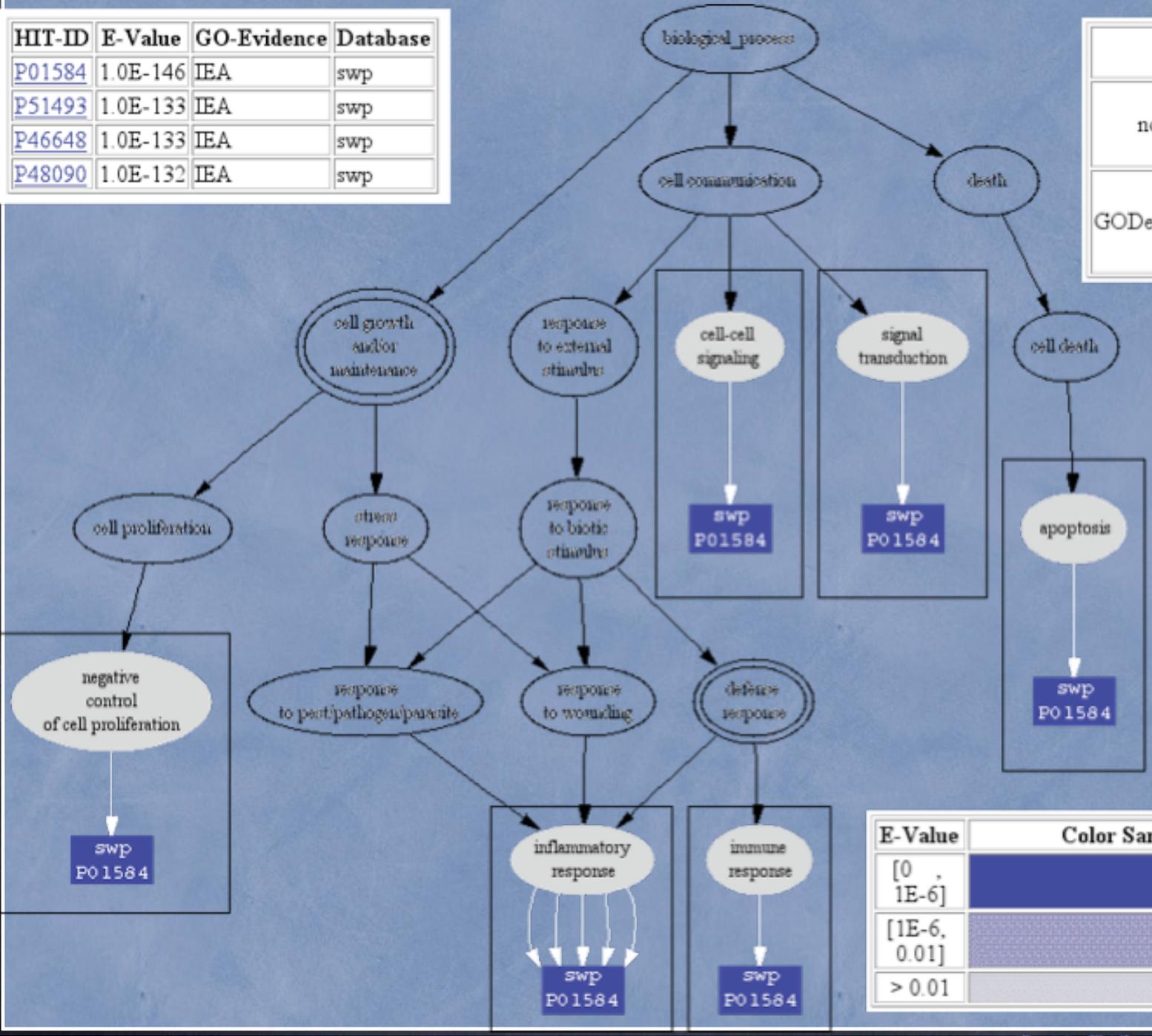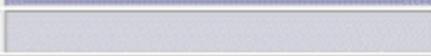
Email address >> [   ]

5. Finish!

To run your query, click the 'Submit Job' button. The 'Reset Form' button acts as you might expect!

[Submit Job] [Reset Form]

Summary

- Agent Research is fun, exciting, cutting-edge
- Still very young field
  - open, multi-disciplinary
- Our foci:
  - Coordination, planning, scheduling
  - Agent toolkits
  - Information Gathering
  - Organizational Issues