In *Proceedings of the Technology Demonstration Session of the 14th International Conference on Knowledge Engineering and Knowledge Management*, EKAW 2004, 5-8th October 2004 - Whittlebury Hall, Northamptonshire, UK.

A Learning Agent Shell for Building Knowledge-Based Agents

Gheorghe Tecuci, Mihai Boicu, Dorin Marcu,
Bogdan Stanescu, Cristina Boicu, Marcel Barbulescu

Learning Agents Center, Department of Computer Science
MSN 4A5, George Mason University, Fairfax, VA 22030, USA
Tel: 1 703 993 {1722, 1591, 1535, 1535, 4669, 1535}

{tecuci, mboicu, dmarcu, bstanesc, ccascava, mbarbule}@gmu.edu

ABSTRACT

This paper presents Disciple-RKF, a learning agent shell that can be used by subject matter experts, with limited assistance from knowledge engineers, to develop knowledge-based agents incorporating their expertise.

Keywords

Knowledge acquisition, learning agent shell, tools for subject matter experts, knowledge engineering, ontology, rules.

1. DISCIPLE-RKF LEARNING AGENT SHELL

Disciple-RKF is a learning agent shell that represents the most recent implementation of Disciple, an evolving theory and methodology for the development of knowledge bases and agents, by subject matter experts, with limited assistance from knowledge engineers [1, 2, 3]. A main goal of the Disciple approach is to overcome the knowledge acquisition bottleneck in the development of knowledge-based systems.

The knowledge-based agents developed with Disciple-RKF use task-reduction as the main problem solving paradigm. In this paradigm, a problem solving task is successively reduced to simpler tasks, the solutions of the simplest tasks are found, and these solutions are successively composed into the solution of the initial task. The knowledge base of an agent is structured into an object ontology that represents the objects from an application domain, and a set of task reduction and solution composition rules expressed with these objects. The Disciple-RKF shell is as a general problem solving and learning agent with no specific knowledge in its knowledge base. To develop an agent for a specific application domain, one needs to develop its knowledge base by using the various modules of Disciple-RKF. In essence, this process consists of developing the ontology for the specific application domain and of teaching the agent how to perform various tasks, in a way that resembles how one would teach a human apprentice. In the last four years, successive versions of Disciple-RKF have been used to develop intelligent agents for center of gravity analysis that are used in several courses at the US Army War College.

The next section presents an overview of the Disciple-RKF agent development methodology.

2. AGENT DEVELOPMENT METHODOLOGY

The Disciple approach covers all the phases of agent development and use. First, a knowledge engineer works with a subject matter expert to develop an ontology for the application domain. They use the ontology import module (to extract relevant ontology elements from existing knowledge repositories) as well as the various ontology editors and browsers of Disciple-RKF. Fig. 1 shows the interfaces of three of Disciple's ontology browsers: at top is the association browser (which displays an object and its relationships with other objects), at bottom left is the tree browser (which displays the hierarchical relationships between the objects in a tree structure), and at bottom right is the graphical browser (which displays the hierarchical relationships between the objects in a graph structure). The result of this knowledge base development phase is an object ontology which is complete enough to be used as a generalization hierarchy for learning, allowing the expert to train the Disciple agent how to solve problems, with limited assistance from a knowledge engineer.

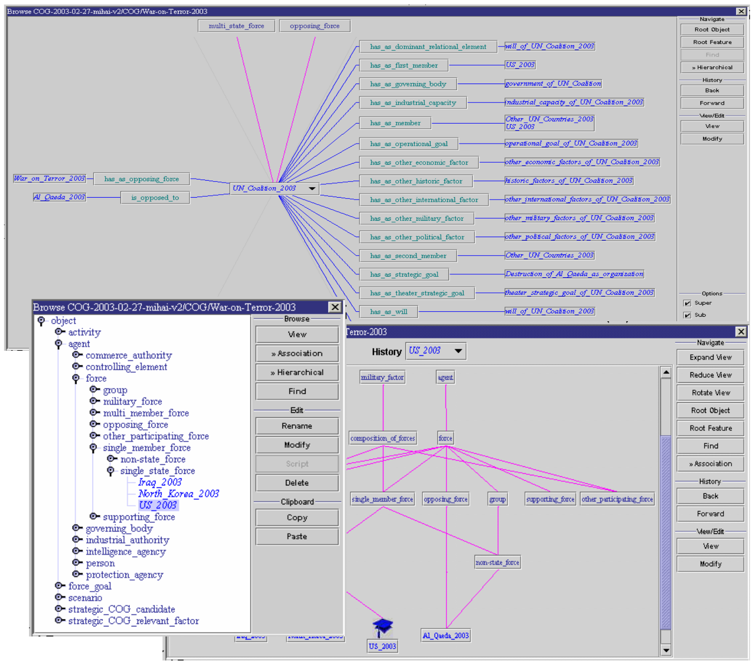
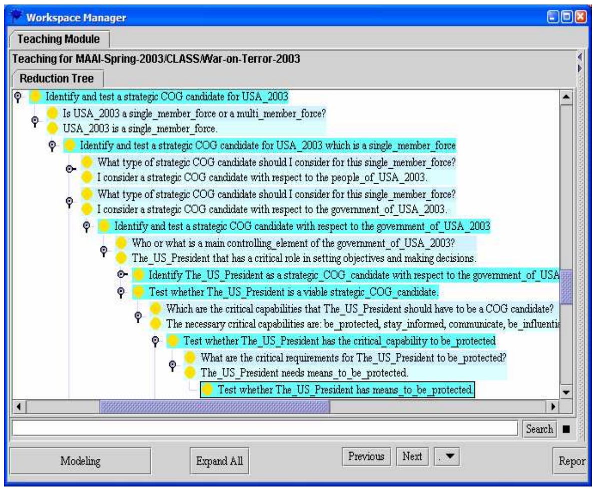
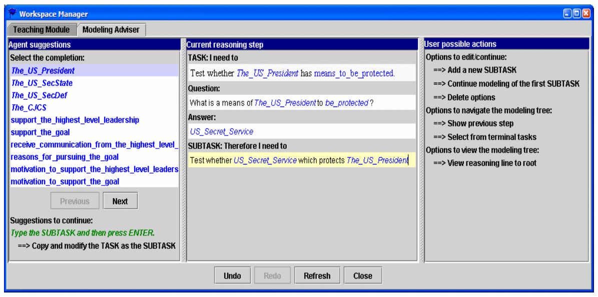


Figure 1. Three ontology browsers of Disciple-RKF

The expert formulates a specific problem solving task and shows the agent the corresponding problem solving steps, helping the agent to understand them. Each problem solving step indicated by the expert consists of a task to be reduced, a question related to that task, the answer to the question, and one or several subtasks or solutions that reduce the task. The top part of Fig. 2 shows a fragment of a reasoning tree consisting of a sequence of task reduction steps. The bottom part shows the interface of the Modeling Assistant that helps the expert to express how to reduce the task from the bottom of the reasoning tree. This assistant may suggest the question to be asked, the answer to the question, and the knowledge base elements the expert may be referring to, since the expert's input is in natural language. Each problem solving step indicated by the expert is an example from which the agent learns a general problem solving rule. First, the expert and Disciple collaborate in formalizing the tasks from the expert's examples, and Disciple learns general task patterns. Then the expert helps Disciple to find an explanation of why the task reduction step is correct and Disciple learns a general task reduction rule. The top part of Fig. 3 shows the interface of the task formalization module, with the tasks in natural language in the middle and their formalizations in the right. The bottom part of Fig. 3 shows the explanation generation and selection interface. The task reduction rule learned from the example specified in the modeling assistant (see bottom of Fig. 2) is shown in the rule viewer from Fig. 4.


Workspace Manager
Teaching Module
Teaching for MAAI-Spring-2003/CLASS/War-on-Terror-2003
Reduction Tree

Identify and test a strategic COG candidate for USA_2003
Is USA_2003 a single_member_force or a multi_member_force?
USA_2003 is a single_member_force.
Identify and test a strategic COG candidate for USA_2003 which is a single_member_force
What type of strategic COG candidate should I consider for this single_member_force?
I consider a strategic COG candidate with respect to the people_of_USA_2003.
What type of strategic COG candidate should I consider for this single_member_force?
I consider a strategic COG candidate with respect to the government_of_USA_2003.
Identify and test a strategic COG candidate with respect to the government_of_USA_2003
Who or what is a main controlling_element of the government_of_USA_2003?
The_US_President that has a critical role in setting objectives and making decisions.
Identify The_US_President as a strategic_COG_candidate with respect to the government_of_USA
Test whether The_US_President is a viable strategic_COG_candidate.
Which are the critical capabilities that The_US_President should have to be a COG candidate?
The necessary critical capabilities are: be_protected, stay_informed, communicate, be_influenti
Test whether The_US_President has the critical_capability to be_protected
What are the critical requirements for The_US_President to be_protected?
The_US_President needs means_to_be_protected.
Test whether The_US_President has means_to_be_protected.

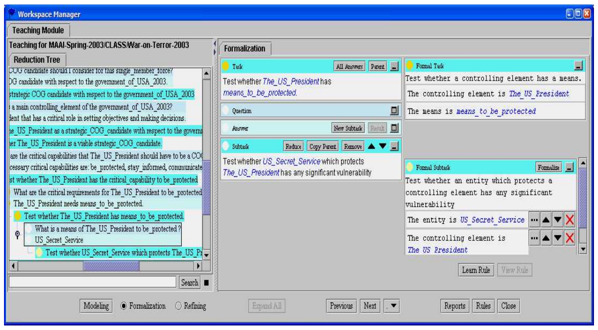

Workspace Manager
Teaching Module | Modeling Adviser

Agent suggestions
Select the completion:
The_US_President
The_US_SecState
The_US_SecDef
The_CJCS
support_the_highest_level_leadership
support_the_goal
receive_communication_from_the_highest_level_
reasons_for_pursuing_the_goal
motivation_to_support_the_highest_level_leaders
motivation_to_support_the_goal
Suggestions to continue:
Type the SUBTASK and then press ENTER.
==> Copy and modify the TASK as the SUBTASK

Current reasoning step
TASK: I need to
Test whether The_US_President has means_to_be_protected.
Question:
What is a means of The_US_President to be_protected ?
Answer:
US_Secret_Service
SUBTASK: Therefore I need to
Test whether US_Secret_Service which protects The_US_President

User possible actions
Options to edit/continue:
==> Add a new SUBTASK
==> Continue modeling of the first SUBTASK
==> Delete options
Options to navigate the modeling tree:
==> Show previous step
==> Select from terminal tasks
Options to view the modeling tree:
==> View reasoning line to root

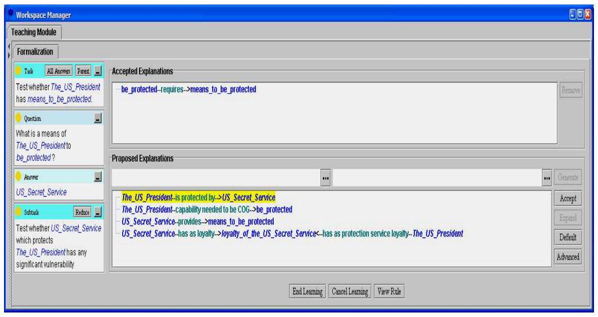
Figure 2. Reasoning tree and modeling assistant interfaces


Workspace Manager
Teaching Module
Teaching for MAAI-Spring-2003/CLASS/War-on-Terror-2003
Reduction Tree
COG candidate should I consider for this single_member_force?
OG candidate with respect to the government_of_USA_2003.
strategic COG candidate with respect to the government_of_USA_2003
a main controlling_element of the government_of_USA_2003?
lent that has a critical role in setting objectives and making decisions.
he_US_President as a strategic_COG_candidate with respect to the governm
er The_US_President is a viable strategic_COG_candidate.
are the critical capabilities that The_US_President should have to be a CO
cessary critical capabilities are: be_protected, stay_informed, communicate
st whether The_US_President has the critical_capability to be_protected
What are the critical requirements for The_US_President to be_protected
The_US_President needs means_to_be_protected.
Test whether The_US_President has means_to_be_protected.
What is a means of The_US_President to be_protected ?
US_Secret_Service
Test whether US_Secret_Service which protects The_US_Pr

Formalization
Task
Test whether The_US_President has means_to_be_protected.
Question
Answer
Subtask
Test whether US_Secret_Service which protects The_US_President has any significant vulnerability

Formal Task
Test whether a controlling element has a means.
The controlling element is The_US_President
The means is means_to_be_protected

Formal Subtask
Test whether an entity which protects a controlling element has any significant vulnerability
The entity is US_Secret_Service
The controlling element is The_US_President


Workspace Manager
Teaching Module
Formalization
Task
Test whether The_US_President has means_to_be_protected.
Question
What is a means of The_US_President to be_protected ?
Answer
US_Secret_Service
Subtask
Test whether US_Secret_Service which protects The_US_President has any significant vulnerability

Accepted Explanations
be_protected--requires-->means_to_be_protected

Proposed Explanations
The_US_President--is protected by-->US_Secret_Service
The_US_President--capability needed to be COG-->be_protected
US_Secret_Service--provides-->means_to_be_protected
US_Secret_Service--has as loyalty-->loyalty_of_the_US_Secret_Service<--has as protection service loyalty--The_US_President

Figure 3. Task formalization and explanation

As Disciple learns new rules from the expert, the interaction between the expert and Disciple evolves from a teacher-student interaction, toward an interaction where both collaborate in solving a problem. During this mixed-initiative problem solving phase, Disciple learns not only from the contributions of the expert, but also from its own successful or unsuccessful problem solving attempts, which lead to the refinement of the learned rules. At the same time, Disciple extends the object ontology with new objects and features.

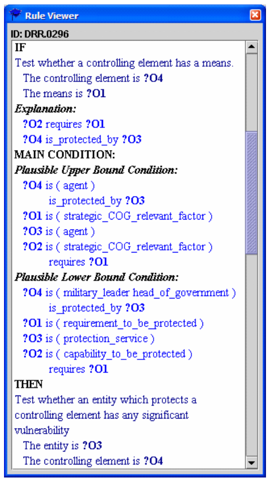

Rule Viewer
ID: DRR.0296
IF
Test whether a controlling element has a means.
The controlling element is ?O4
The means is ?O1
Explanation:
?O2 requires ?O1
?O4 is_protected_by ?O3
MAIN CONDITION:
Plausible Upper Bound Condition:
?O4 is (agent)
is_protected_by ?O3
?O1 is (strategic_COG_relevant_factor)
?O3 is (agent)
?O2 is (strategic_COG_relevant_factor)
requires ?O1
Plausible Lower Bound Condition:
?O4 is (military_leader head_of_government)
is_protected_by ?O3
?O1 is (requirement_to_be_protected)
?O3 is (protection_service)
?O2 is (capability_to_be_protected)
requires ?O1
THEN
Test whether an entity which protects a controlling element has any significant vulnerability
The entity is ?O3
The controlling element is ?O4

Figure 4. Learned rule

Copies of Disciple agents may also be trained in parallel by different experts. In this case the individual knowledge bases have to be merged into an integrated knowledge base, as discussed in [2]. The agent's knowledge base can also be exported to be used in the development of new agents.

3. ACKNOWLEDGEMENTS

This work was supported by the DARPA, AFRL, AFMC, USAF agreement F30602-00-2-0546, by the AFOSR grant F49620-00-1-0072 and by the US Army War College.

4. REFERENCES

[1] Tecuci, G.:*Building Intelligent Agents: An Apprenticeship Multistrategy Learning Theory, Methodology, Tool and Case Studies.* Academic Press, London, 1998.

[2] Tecuci, G., Boicu, M., Marcu, D., Stanescu, B., Boicu, C., Barbulescu, M., Parallel Knowledge Base Development by Subject Matter Experts, in *Proc. 14th International Conference on Knowledge Engineering and Knowledge Management,* EKAW 2004, 5-8th Oct 04 - Northamptonshire, UK, Springer-Verlag, 2004.

[3] Disciple-RKF demo at http://lac.gmu.edu/RKF/cog /DISCIPLE-COG%20DEMO.pdf