12TH ICCRTS

"Adapting C2 to the 21st Century"

Title:
Creating and Capturing Expertise in Mixed-Initiative Planning

Track 4: Cognitive and Social Issues
Track 1: C2 Concepts, Theory, and Policy
Track 8: C2 Technologies and Systems

Authors
Anthony J. Ford
Joseph A. Carozzoni

Point of Contact:
Anthony J. Ford
Air Force Research Laboratory Information Directorate
Systems & Information Interoperability Branch (IFSE)
525 Brooks Road, Rome, NY 13441-4505
315-330-2610
Anthony.Ford @rl.af.mil

Abstract

Compared with prior engagements, commanders today are exposed to a battlespace that is more dynamic and less predictable. With increasing frequency, commanders are confronted with an array of problems whose solution requires knowledge beyond their military training. In these novel situations, decision makers often rely on their past experiences incorporating a process best described in research as analogy-based reasoning and/or recognition primed decision making. While the relevancy of the experience is based on the individual, a key goal would be to capture and exchange relevant experiences between individual decision makers. The shocking events of 11 September 2001 may have been less shocking to anyone serving in the Pacific theater of operations toward the end of World War II and experienced Kamikaze warfare. This paper describes work in progress at the USAF Research Laboratory Information Directorate to capture, develop, and provide experience to commanders during mixed-initiative planning. The objective of this work is to provide a rich database of experiences for the commander to compare to the current situation. The research described in this paper is aimed at developing a computational representation for episodic models, and reasoning on those models for retrieval and experience extraction.

Introduction

Experience is the accumulation of knowledge or skill that results from direct participation in or observation of an event or activity. Experience generally refers to procedural knowledge (e.g., knowing how best to perform a task), rather than propositional knowledge (e.g., true/false facts). An expert is considered a person with considerable experience in a particular subject. Activities where the same and/or similar problems occur lend them self well from the use of experience. While most of our experience accumulates over a period of time, a single specific momentary event can result in experience.

Experience is a key ingredient for success, especially in complex situations such as warfare. A simple but critical example is well documented about the survivability of fighter pilots in the pacific campaign during World War II. Those fighter pilots frequently found themselves alone in that vast battlespace against enemy fighter pilots. The first engagement for those pilots had the highest loss of plane and/or pilot. On the other hand, those with the most experience in dog fights had the greatest chance of returning.

For the “art of warfare”, what is the useful life-span of experience? Our knowledge of military strategy is both cumulative and continually changing. The writings of Sun Tzu six centuries B.C. remain relevant today even though the technology has changed, and in many cases only the appearance of an event has changed (e.g., horses/mules being replaced by motor vehicles). This paper discusses how past experience can be used to better our understanding of the current situation to the point that we can develop an intuition about the future. A major goal of Experience-Based Reasoning (EBR) is leveraging the past in order to adapt past successes to current problems, while avoiding prior failures. Applying this sort of reasoning to war planning is an exciting opportunity to tap into a powerful human process and augment that process with the power of technology. By using information technology for the storage, retrieval, and application of experience, we can greatly enhance decision making. The challenge becomes knowing how to store experiences in a computationally reasonable way, and how to match those experiences to new situations as they arise

As with other evolutionary fields, the practice of warfare requires continuous learning and sustained improvement. Accumulation of knowledge through experience is critical to becoming an expert. All experience is potentially reusable, and in a mixed-initiative environment the key is how to represent it. Experience is normally reusable in the same domain from which it was initially derived. However, reusing experience in other domains is dependent on an understanding of the similarities and differences between the domains.

Experience in Mixed-Initiative Planning

In a Mixed-Initiative Planning System (MIPS), humans and machines act as a team to develop and manage plans [1]. Properly designed, the human contributes to the plan those

details that humans are best at (e.g., pattern recognition), and machines like wise (e.g., number crunching). MIPS are frequently generalized and categorized under Multi-Agent Planning Systems (MAPS) in involving many agents that can be either human or machine. A key issue in such systems is the dialog between the human and machine. In this research, the collaboration between the human and machine focuses on a rich database of experiences for the commander, in cooperation with the machine to compare to the current situation.

In peacetime, deliberate planning is used to evaluate hypothetical future situations the United States may be required to respond to politically and/or militarily. Because these scenarios are considered so critical, the plans must be in place one to two years before they are predicted to occur. Conversely, crisis action planning is used for specific situations as they occur, in either a response and/or shaping manner. The nature of crisis action planning reduces the time available for planning to a few days or even hours. A common characteristic with both planning processes are the dependence upon experienced planners to produce the best possible plans. A key difference is in the first, a planner has time to "walk down the hall" and collaborate with as many planners as possible, while in the second a commander may have only minutes with staff in the immediate vicinity or possibly seconds by themselves. In either case, the quality of the plan rests with the experience and knowledge of those doing the planning.

The U.S. military is well into a transformation process. While force-on-force conflicts cannot be permanently ruled out, we need to improve our capability for asymmetrical warfare. Asymmetrical warfare introduces a dramatic increase in the OPS-TEMPO and has driven the Observe, Orient, Decide and Act cycle (i.e., Boyd's OODA loop [2]) toward progressively shorter timelines. In today's fast paced conflicts, finding solutions to novel problems with little information in a crisis response is a frequent occurrence. Given the great advances in technology, it is hard to imagine looking into the past to find solutions for a conflict that to a commander appears so novel and unique. Regardless, insights can be gained by having a deep knowledge of the past. Consider a mixed-initiative planning environment where new solutions (or possible pitfalls) are found to these novel problems, an Experienced-Based Reasoning (EBR) system would store those experiences for future use. In a way, this experience base becomes a commander's "virtual staff" that expands their current toolbox of solutions. Using EBR as a conflict continues allows decision makers to gain faster insight into potential approaches to a problem, giving them the ability to make progressively faster and more effective decisions.

The Distributed Episodic Exploratory Planning (DEEP) project is a mixed-initiative decision support system that utilizes past experiences, encoded into a Case-Based Reasoning System (eventually to be supplemented with an episodic memory), to suggest courses of action for new situations. It is being implemented as a distributed multi-agent system, using agents to maintain and exploit the experiences of individual commanders as well as to transform suggested past plans into potential solutions for new problems. The agents interact through a common knowledge repository (a blackboard in the initial architecture).

The system is mixed-initiative in the sense that a commander, through its agent, can view and modify the contents of the shared repository as needed. The blackboard architecture involves specialists working cooperatively to solve a problem and is well suited for dealing with ill-defined, complex situations such as warfare. The blackboard supports non-deterministic, opportunistic reasoning and involves metaphorical specialists learning their expertise in vastly different situations. The initial blackboard architecture maintains:

- The current world state and status of battlespace
- Commanders' mission intent and operational objectives
- Candidate plans for addressing the objectives
- Plans selected for execution

Capturing Experience

The primary challenge of the DEEP project is translating the experiences collected from good (or potentially bad) command decisions into a form that is understandable by a computer and amenable for use in mixed initiative planning. The key is representing knowledge in a form that facilitates inferencing (i.e., drawing conclusions from knowledge).

To represent a commander's experience of a planning decision, several details about the plan must be captured. First, the situation (i.e., the battlespace) must be known. Second, the constraints such as rules of engagement and resources are addressed. Third, we must understand what the objectives of the commander (i.e., commander's intent). Lastly, it is also important to denote what was assumed by the commander at the time of planning. Assumptions drive decision making as they denote the commander's stance towards the adversary's intent, the adversary's possible actions, available intelligence information, and the proposed efficacy of the actions the commander undertakes. Figure 1 shows the overall areas of information that need to be captured to represent the context that leads to a plan.

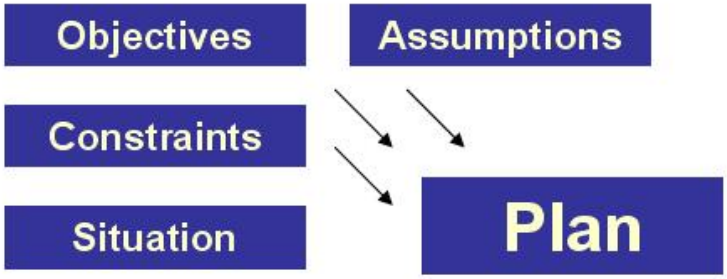


Figure 1: Context of a Plan

Once the plan is formed, the actual execution of the plan and the outcome denotes the actual experience itself. The first information that must be captured is the actual actions the commander's force undertook in accordance with their plan. Second, we need to know what events took place in reaction to the blue force's actions. Additionally, even though our own actions were planned to have certain effects, often times there are unexpected consequences. Third, the actual effect on the situation resulting from our actions also needs to be captured.

Now that we have captured context, plan, and actual events of the experience, we need to be able to characterize the outcome of the experience. Several metrics could be used to explain the overall outcome of a case. These include the objectives that were met and failed, the assumptions that were true and false, the costs incurred by all factions involved, and any plans and actions that cannot be undertaken due to the selection of this plan (opportunity cost). Figure 2 shows these additional elements in an experience. The elements in both Plan and Outcome yield a realistic flow of an experience from beginning to end.

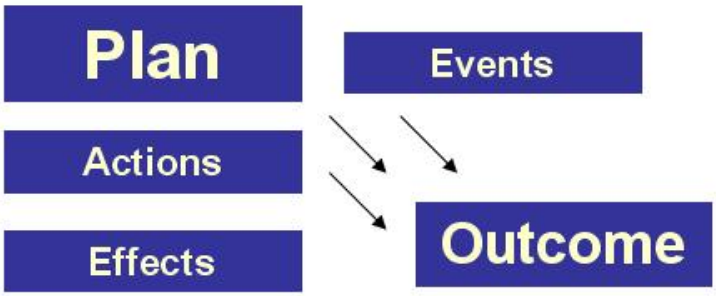


Figure 2: Context of Outcome

In the current DEEP architecture, an episode is comprised of several concepts (*objectives, assumptions, situation, constraints, plan, actions, effects, events, and outcome*). What sort of data can be collected in each of these areas, and how can that data be stored in a meaningful way that allows analogies to be formed for episodic reasoning? Each of these concepts will briefly be addressed in the context of deciphering a decision in a computational plan representation.

Objectives

There are two general approaches for understanding a decision based on what the commander was trying to accomplish. One is to allow the commander to articulate the ideal world end state as a *vision*. One way to accomplished this is by storing the world state adjusted to show what the plan was supposed to accomplish. This would require the same level of detail as the world state model – a significant amount of detail.

The other approach would be to allow the commander to enunciate objectives in a more direct, almost declarative way with statements such as “establish air superiority.” The statements denote a characterization of the end state, but not how to achieve it. As such, there is not just one specific instance of the world where air superiority is established and there are many ways to approach that end state and many methods by which to achieve it. Any end state, regardless of the details, what satisfies the objective is acceptable. This approach has more flexibility for the decision maker, since several various world states can satisfy the objectives. The computational challenge for this approach is how do we know that ‘establish air superiority’ has been attained?

Assumptions

Representing assumptions requires three basic concepts. First, what each assumption is about. Second, what the assumption is. Third, the level of confidence in the assumption.

There are multiple ways to represent assumptions such as plain text or attribute-value tuple. The goal is for the representation to support the ability to draw analogies.

Of the three concepts of an assumption, eliciting the level of confidence in an assumption provides the greatest challenge. It requires a value indicating the degree of confidence which in many cases is simply a "gut feeling." Factors involved in defining confidence may be factors a commander is not consciously aware of. Additionally, the degree of confidence could be a qualitative statement (e.g., "I'm pretty sure") or a quantitative one (e.g., sure to a degree of 0.98).

Situation

Several things within a situation, or world state, can be easily quantified. What is present and in what quantity, the temperature outside, the altitude of terrain, etc. Several challenges befall us, however, when we start seeking to model less easily visible and concrete things. Social dynamics, public opinion, cultural factors, etc., can present considerable challenges. Several opportunities exist for analogies, especially since simple similarity metrics can work well for the easily quantifiable aspects of a situation. A more thorough structural analysis could be performed for other types of information, such as social networks.

Constraints

Capturing the factors that impose limitations on a plan is closely related to two important concepts. First, *how* you model the situation is important because it defines the 'language' that you can speak in terms of constraints. The same methodology used to capture the situation defines the vocabulary with which you can refer to constraints. For example, if you do not capture the fact that there are such things as fuel or jeeps within your situation model, then you cannot impose the constraint that jeeps require fuel to operate. In this example, we see that two items (jeep, fuel) are linked together by some relationship (requires). Often, constraints can be constructed as a form of logic where items are related to each other in formal manner.

The second concept is an idea of the planner's *objectives*. It would be a monumental undertaking to capture every explicit constraint in existence, even for a very simple situation. These constraints could be physical (limitations on hardware employed, etc.) or nonphysical (policy, such as rules of engagement). How detailed and at what scope of concern these constraints are captured depend upon both the details available, and under what context the constraints are being captured.

Plan

From an abstract view, a plan is simply a collection of objectives and the actions that will be undertaken to satisfy those objectives. However, other information provides more detail to a plan. Cost, for example, is an important factor when choosing one plan over

another. Second order effects, beyond those that satisfy the objectives, are another example of additional plan details.

Actions

Action includes information on who is performing the action, what resources they are using, where they are located, when the action is carried out, and what effect the action has. While capturing an action, you can ask all the basic questions: who, what, when, where, why, and how.

Effects

Effects are potentially the most challenging part of a plan to capture for two important reasons. First, anything changing in the current situation needs to be flagged and expressed. Second, we need to have some idea on how long, in terms of time, we should pay attention to an action to see what effects it has. Determining how far out the causal chain goes for a single action has challenged both science and philosophy for many years. The initial effort in this project will focus on first order (i.e., single) effects.

To address the concept of causality over time, we can allow that detail of capturing an experience up to a human and what they believe about their situation. Because a plan contains the expected effects of an action, the person formulating the plan is using their own judgment to predict what those effects might be. If the planner has an idea of wide-spanning effects for his or her plan over huge spans of time, then we need to have the robustness to capture that kind of predictive content. Conversely, if a planner is only willing to predict single, direct effects from his or her own actions, we need to be able to capture that too.

Events

Capturing what actually happens during the implementation of a plan is an exercise in not only observation, but also good judgment. In many ways, events are a lot like actions. However, the key difference between them is that the events we observe are performed by other people with their own, unknown agendas. Essentially, we can ask all of the same questions we can with an action, except for 'why'. Because we cannot know the exact motives of another person, the reason for their action is always an assumption. Also, we have to make an assumption about the causality of our own actions. In other words, because no plan survives first contact with the enemy, we have to be prepared to observe the results of our actions and properly attribute their effects on the environment.

Utilizing good judgment on the part of the observer is the critical element of this portion of capturing an experience. If we do not fully understand the motives of other actors in our environment, we may be misattributing the reasoning behind their actions. Later, we might make a misstep because we have false assumptions about how another actor in the environment will respond. Also, if we cannot dissect the primary, secondary, and higher-order effects of our own actions when we carry them out, then we may make mistakes in

the future because we believe something about our actions and their results that is not actually supported by experience.

Outcome

There are both long-term and short term aspects of the outcome of a plan. Short-term aspects include the objectives that were satisfied, the resources expended, and the assumptions that panned out. However, long-term aspects of an outcome are more difficult to capture. For example, what other plans or actions cannot be taken due to the result of this plan (opportunity cost)? Beyond that, what are the second and tertiary effects of this plan? How long should I pay attention to possible effects from my plan? These are all deep concerns when trying to determine the efficacy of a plan. What seems like a good outcome now, might actually pan out into a disaster down the road.

The observation techniques utilized while observing the effects of single actions now must be expanded to encompass the whole scope of an episode of experience. Knowing what was sacrificed and gained in both the short and long term is very important for scoring the efficacy of a plan. This score of efficacy is critical in using case-based reasoning, because you do not want to repeat a bad experience in the future. The only way to avoid that is to store the success of your experience to aid in future judgment.

Developing Experience

In the prior sections, we examined how to capture the planning, execution, and results of an experience. We can now examine how to represent that information within a formalized structure that can be understood by a computer. In order to facilitate information sharing between different planning systems, DARPA conducted extensive R&D in plan representation under the ARPA Rome Laboratory Planning Initiative (ARPI).

An important product of that work was the Core Plan Representation (CPR) [5][6], an object-oriented model for expressing information common to many plan, process, and activity models. CPR was designed to model a basic level plan (i.e. information common to any plan), then be specialized in more detail for restricted domains. In the DEEP project, CPR will be extended to allow a computer to understand the episode for the sake of reasoning, as well as capturing the experience of a human planner. Appendix A is the 1996 version of CPR, and appendix B is the 1998 model of a plan. While the 1998 version of CPR is a better model of a “generic/basic plan”, we found some useful aspects to the 1996 version and thus have used aspects of both.

Adapting CPR

Moving forward chronologically, there were several intriguing elements from the initial 1996 report on CPR that are highly interesting and potentially very useful. Namely, the inclusion of Facts and Assumptions, which are subclasses of Annotation, help the planner

and computer understand the subtle nature of the source of information. Rather than simply tacking on information to an encapsulated element in planning, it can be understood more concretely that information is either derived from evidence with certainty (a Fact) or something the human planner believes is true (an Assumption). This allows the computer-based reasoning to understand how confident it can be in handling the information.

In the same token, the inclusion of Imprecision and Uncertainty in the 1996 version is very important to further articulating the confidence in information. Although the ARPI group admitted that the exact nature of modeling Imprecision and Uncertainty were wider spanning research topics than they were willing to tackle at that time, the inclusion of the classes themselves leaves the opportunity to expand upon the idea. By including a concrete representation of the doubt of the source of information (Uncertainty) and the doubt in the exactness of the information itself (Imprecision), we can carry these ideas forward by utilizing Fuzzy Logic. In other words, even though it is difficult to capture Imprecision and Uncertainty, there are paths by which a computer can understand them mathematically in terms of its internal reasoning using Fuzzy Logic [11].

The 1998 version of CPR added a super-class for all represented plan elements called PlanObject. This is a very useful addition, since the Object Oriented nature of the model lends itself well to these types of generalizations. This object allows us to create very flexible plan structures where each planning element has fields and methods in common. This also aids in implementation, since writing code once for all represented PlanObjects when they all have something in common is simply more efficient. Because the plan schema is recursive, plans can be represented at any level of planning (e.g., strategic, operational, and/or tactical). Additionally, the recursive nature of CPR allows plans to be reasoned over at the plan and plan fragment level.

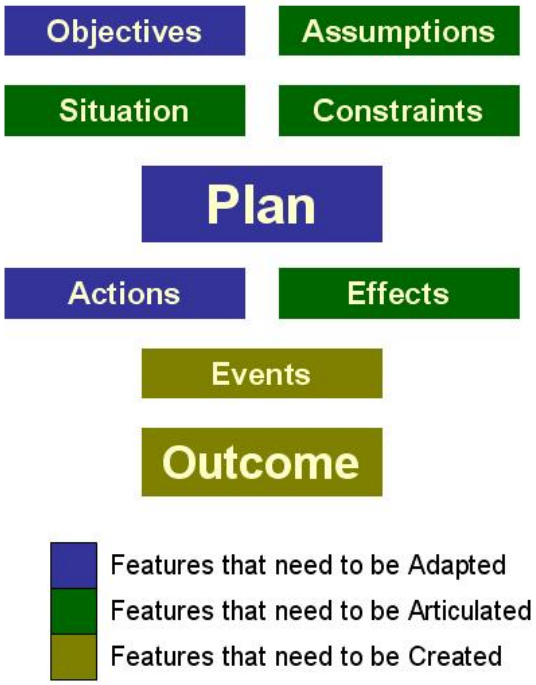


Figure 3: Adaptations to CPR

Another welcome addition from the 1998 version is SpatialSpec. It defines a location where an action takes place. However, the current structure allots only one SpatialSpec per Action. In reality however, thinking from a non-geospatial point of view, every Action can take place in a huge variety of places. Not only can an action take place on the ground, but also in cyberspace, in the realm of a political structure, on the lines of a power grid, or any other huge variety of 'places' that are defined by the context of the

action. Because of this, SpatialSpec will need to be dealt with specifically in terms of adapting the concept to fit this thinking. This will be discussed later in further detail.

An especially useful addition from the 1998 version of CPR is Role. Using Role, we can understand the exact part that is played by the action. Using this information, we can easily draw analogies for that Action by looking for Actions with similar Roles. In other words, by understanding what function an action played in the overall scheme of an operation (such as 'Fire Support' or 'Transport'), we can make a quick and efficient search in the case-base to find similar actions based on what role they accomplished. Using this approach, deeper pruning of those results is possible, because the 'first cut' of the Actions are already known to perform the role that is required in the current planning context.

A final, especially useful contribution from 1998 is the Entity class. This class allows Actors, Resources, and Roles to interrelate to each other. In other words, we can view one item from a variety of perspectives by allowing Entity to 'bridge the gap' between its various facets. While one real-world item may be considered an Actor and Resource, for example, we can model it as an Entity, and treat Actor and Resource as 'perspectives' on that single item. This is an exciting opportunity to model highly complex and subtle items without the limitations of only one of the three perspectives. This also allows us to create more perspectives with the ability to interrelate them in the overall CPR scheme.

Now that we've identified specific items within the Core Plan Representation that are particularly interesting, we can work to expand the representation to encompass an entire episode of an experience, rather than just the plan itself. As discussed before, we need to adapt CPR to represent the *objectives, assumptions, situation, constraints, plan, actions, effects, events, and outcome* of an experience. Figure 3 shows the level to which we need to change CPR to allow it to represent whole episodes. Some features need to be adjusted, meaning that they already exist and simply need to be customized. Other features need to be articulated, which means that they exist, but not to a sufficient degree of fidelity or development, thusly requiring further research and refinement to use. Other features still need to be created, meaning that they go outside the scope of the original CPR context, thusly requiring that they be made from scratch.

New Classes in CPR

Given this more specific task of adapting CPR to the context of capturing experience, we can explore new and adapted classes within the CPR framework. In this section we will discuss those new classes and the thought process that went into their creation.

To represent more than just the plan involved in a military operation, CPR needed to be adapted to show the whole story: from the planning, to the action, to the results. This would allow us to model a whole experience; one that can be stored as an episode for the purposes of Episodic Reasoning.

Event

One of the major drawbacks of the Core Plan Representation was its lack of support for representing red force actions. In other words, although it had all sort of things in mind for planning from your own perspective, it had no support for tracking what you believed an adversary might do, or what they actually did. To alleviate this, we took an approach that used similar classes for modeling assumptions about the adversary, and actual adversary actions.

Under this approach, a new class very similar to Plan was formulated, called Event. Event is used to hold information about what actions some other group undertook, without direct knowledge of their intentions or world state. In this sense, it is similar to Plan, but trimmed down to accommodate the lack of knowledge of the observed group's subjective universe. Rather than store the Objective of another person directly, we can store that objective in terms of an Assumption. That Assumption would contain the Objective object, rather than the Objective object being stored within the Event itself.

Assumption

Using this same approach, we can adapt the Assumption class to contain any other CPR entity that encapsulates an entity or concept that is only assumed to exist. Therefore, we can capture intelligence information, assumed adversaries with unknown identities, resources that are only assumed to exist at certain locations, and other pieces of ambiguous information that is important to planning. These pieces of assumed information are very important to the planning process, because they help capture the context of a planner's judgment at the time of planning. Taking a look not only at assumptions that store straight information (such as text), but also assumptions about entire entities in the environment can give us a glimpse of the complex thought process that goes into planning.

Outcome

Keeping track of the success or failure of an experience is a tricky thing. There are many important things to consider. Not only are the goals the endeavor was undertaken with important, but also the costs of following that particular plan. It is not only a matter of success, but also of pragmatic impetus. In terms of comparing one plan to another, shear accomplishment of goals leaves no basis to discern. In order to track the many and varied possible costs of a plan, a second new class (contained within Outcome) can be created called simply *Cost*.

However, tracking the objectives achieved and cost incurred is not the entire outcome either. Forcing the adversary to spend their resources is also an important factor in the success of a military operation. This is especially so in terms of Fourth Generation Warfare, where affecting the will of the adversary can be strongly influenced by the effect of their own costs. Knowing the red line past which an enemy will not spend any more to oppose you is vital to developing plans and strategies over time. For these

reasons, not only the commanders' own costs, but also the cost of their adversaries should be tracked.

In the same grain of responding to important measures of outcome in military planning, another key area of this type of planning is the effective use of good intelligence to produce valid assumptions about the situation. For example, if a patrol tells you that they spotted ten enemy troops at the top of the next hill, your planning should reflect this observation. In fact, this observation could be an indicator of something else not yet seen. For example, seeing ten troops might indicate that there is an enemy base nearby, or that there are one hundred more troops waiting to march once they know the way is clear, or that there is a convoy out of gas ahead with ten hitchhikers looking for a gas station, etc. One piece of information could spawn thousands of assumptions. It is up to a good commander to make the best assumptions possible given the situation. At the very least, in this example, there are ten enemy troops on or near the next hill. This piece of information already can influence planning, without a terribly wide leap of faith. Either present more force to the hill to prepare for an enemy force, or avoid that hill all together. At least you know that the hill is not empty, if nothing else. Keeping track of a commander's assumptions and the degree to which they were actually true is a good way to capture one of the essential skills in good planning. Since CPR already includes the *Assumptions* class, simply maintaining a collection of them within the *Outcome* class, along with corresponding truth values, is adequate to take the truthfulness of assumptions into account.

Another important piece of the planning puzzle is to know what you *could have* done if you had not undertaken this particular plan. Some opportunities are particular, and don't come knocking again. Time constraints, resources limitations, etc. leave only a certain window of opportunity for many plans. This opportunity cost should be taken into account when recording an experience. Squandering useful opportunities can have a huge impact on the success of an overall campaign. To keep track of these costs, we can use the existing CPR objects of *Plan* and *Action* to keep track of full plans or single actions that cannot be undertaken because of the events unfolding in this experience. Of course, many of these cannot simply be known at the outset, so these should only be stored if they were already prepared plans that are now impossible, or obviously actions that were noted to be operationally advantageous that are now impossible.

With these four main areas (Objectives, Costs, Assumptions, and Opportunity Cost) in play, we can now begin to develop the *Outcome* class, which will extend *PlanObject*. This is shown below in Figure 4.

Outcome
ObjectivesMet : Objective ObjectivesFailed : Objective FriendlyCost : Cost AdversaryCost : Cost AssumptionsTrue : Assumption AssumptionsFalse : Assumption

OpportunityCostP : Plan OpportunityCostA : Action

Figure 4: Outcome

Using this class as a single point of collection for various types of information about the success or failure of a military plan allows objective scoring mechanisms to examine the case from their own perspective. For example, if a member of the Red Cross was looking at a plan, he or she may gauge the success of the experience as a function of casualties and medical supplies used. From a different perspective, an intelligence collector may gauge the success of a plan based on what information sources have become unavailable because of the plan compared to how many new information sources were created/revealed. The success of a plan is all a matter of perspective; however each perspective requires similar kinds of information with which to establish the context.

Going beyond simply storing information, we can apply some forms of objective scoring. If someone were looking at this plan not from a specific point of view, but rather from a broad and general sense, they may want to see some overall measures of this experience's success. To accomplish this, we can look at how important each element was to the commander's vision of the plan. For example, if certain objectives were more important than others, they will simply have a higher weight. If certain assumptions are more critical and certain than others, they could have a higher weight. Then, when determining the overall, general success of the experience, we can simply use those weights to determine the ratio between what elements were successful and not successful. In other words, we can take a simple ratio to see which of the most important objectives were met. We can apply the same sort of reasoning to costs. Applying a weight to resources can allow us to judge by simple ratio the comparative cost between commander and adversary. Constructing ratios based on importance gives a way to examine the success of the plan in terms of what the commander doing the planning deemed important. These ratios can provide a quantitative way to broadly look at the success of a plan. This can be used to scan through an experience base quickly to 'weed out' certain plans, and focus attention on others based on success in one or more areas. You can look at highly effective plans, in terms of accomplishing goals. Also you could seek out plans with good use of judgment and intelligence information with a high rate of true assumptions. You can look for plans with a cost ratio that is very high; plans that are exceptionally costly to the adversary. Or you could look for plans that leave your options the most open for future planning. Feasibly, using these weighted scores you could aggregate them to find the best overall plan (by the numbers).

Cost

Being able to tell how much a plan costs is an important factor in plan selection. However, the current version of CPR has no way to take this into account. We must first understand what cost is, in this context. For an operational definition, we are defining Cost as a resource that was expended to enable your own actions. Cost differs from simple negative effects in that it implies expenditure in something that you already own and control (in some way). In this context, costs can be expected to occur, or could arise

unexpectedly, but all pertain to resources that are under your command. In a sense, cost is an element of control, because the resources expended to enable your actions can be perceived as a management of risk. Being able to perform a cost-benefit analysis gives a potential plan further context. Representing those costs in our plan representation further enables reasoning upon plans and cases. However, this challenge is two pronged. First, we must be able to represent costs themselves. Next, we must be able to differentiate between expected costs and real costs.

In order to represent costs, we must first examine the variety of things that could be expended in the course of a plan. Then, we must have the ability to understand the level of fidelity with which we can model those various factors in regards to the level of fidelity of the overall plan. For example, although employing a tank in battle requires fuel, ammunition, and other supplies, those materiel items were all at some point purchased. From that abstract perspective, you can express that cost in simple dollars and cents. However, from a more concrete perspective, if the tank needs to be deployed rapidly, and purchasing that fuel would take more time that you have, then you need to rely on the materiel *on hand* to accomplish a task. From that angle, it is not the dollars you spend that counts, but rather the value of the actual materiel in terms of its limited supply. The logistical ability expended to attain supplies in that case is worth more than the actual cost of the materiel itself in a vacuum. Depending upon the situation, there is a difference between the simple dollar cost and the *value* of the materiel used in a plan.

Another cost that could be incurred could be expressed in terms of political capital. Sometimes when you have to call in a favor, you might find your relationship with that person strained. You obviously cannot put a dollar amount on someone's trust, respect, or friendship. Obviously, we must examine methods of qualifying and quantifying cost that go beyond simple monetary exchanges, even if those exchanges are weighted into various levels of situational importance.

Location

In order to model concepts of location that are not strictly geo-spatial, we had to explore what sort of construct could be used to represent SpatialSpec in CPR. As you remember, each Action can contain only one SpatialSpec. Therefore, we are left with two options. First, we could store more data inside of SpatialSpec as to leave a place for the many and varied aspects of location that we could conceive. Second, we could alleviate the restriction in the Action class by allowing it to store any number of instances of information to denote locations. Because it is not known how many different aspects of a location might be modeled a priori, we decided that each class should store only one type. This makes each instance highly cohesive, since it only stores data on one aspect of location. Therefore, we lifted the restriction on the Action class, allowing it to store any number of SpatialSpec. Also, because of the changed nature of the SpatialSpec, to allow several different aspects of locations to be modeled, we also renamed the class Location.

Providing Experience by Reasoning over an Experience Base

An important area of research in artificial intelligence is the field of understanding and forming analogies. In a sense, our power to create relationships between symbols, experiences, and situations is at the root of our ability to apply that experience to our everyday life. We would not know what experiences were valid to draw upon if we did not know how to relate the past to the present. In the same token, we would not know how to adapt those situations to the present context if it were not for that same power. Thus, if we are to truly accomplish mixed-initiative planning, we need to understand how to enable a computer to draw analogies.

There are several approaches we can take. If we think for a moment, we can easily conceive of why two items, say apples and oranges are actually similar. There are several ways to approach that similarity, most notable three.

First, apples and oranges share some similar physical characteristics. They are both sweet, both about the same size, and both have an outer skin. We can examine those characteristics through measurement and observation. In this same sense, if we are trying to draw an analogy in war planning, we can say that a WWII-era Sherman tank and a modern M1A Abrahams tank are similar. Both have guns, both have treads, and both have crews of more than one person. They share these characteristics, which we can determine through an examination of the two items. This of course takes not only time, but also the ability to measure those characteristics. All of this is under the assumption that the two sets of characteristics can even be compared.

A second approach to determining if apples and oranges are similar is examining them not by looking at their attributes, but how they are organized and classified in various taxonomies. In this example, apples and oranges can both be classified as ‘fruits’, so in this sense they are similar. They can also both be classified as member of the plant kingdom, although they are from different Families and Genus’, from the perspective of botanical nomenclature. In this sense, the two are similar, but not as similar as they could be. From this perspective, knowing the taxonomical structure of how we classify plants and animals gives us the ability to articulate the level to which two things are similar. In this context, cats and dogs would be more similar than cats and bamboo plants. Because cats and dogs share a common classification that is at a deeper level of the taxonomy, we can consider them more similar. The asset of this approach is that we do not need to be able to measure to compare the attributes of the items to be matched, as long as they are already classified in some taxonomical structure. This is also the drawback of this approach; items must already be classified, or else you must find a way to classify them yourself, which would require an examination of its attributes (which returns you to the first approach). A military planning example of this would be determining that our Sherman and Abrahams tanks are again similar, because they are both classified as Main Battle Tanks. In this sense, the Sherman and the Abrahams are more similar to each other than the Abrahams and the F-16 are, because the F-16 is a fighter jet and not a Main Battle Tank.

However, following these two approaches blindly may not be enough to adequately make analogies. A fighter jet and a main battle tank actually do share some things in common, even if they are classified differently. Both are able to lay destruction upon targets. Both are capable of being a show of force. Both are capable of protecting soldiers (and airmen) from harm. Even though both items may go about these capabilities differently, they can in a sense be interchanged for each other in the proper situation. Examining not just the physical characteristics of an item is not enough for drawing analogies. Especially in military planning, where using available resources to utilize a capability is supremely important. It is not as important that you have two similar vehicles, what is important is that the mission is accomplished effectively.

Pulling away from an equipment-centric point of view, even two different approaches can be analogous to each other if they accomplish the same effect. For example, a show of force flight, a pamphlet drop, and a bombing run might actually all accomplish the same goal. Knowing this allows the commander to plan in such a way that optimizes costs, benefits, and risk to make more effective decisions.

We can, then, examine analogies not simply as the similarity between the objects involved, but in terms of capabilities and effects. From this perspective, analogies are drawn based on what you want to accomplish, not just on the attributes that you can examine. Going deeper, we can think of those capabilities and effects as relationships between the actor and the affected. If we know the relationship we desire, and the target object, we know what sorts of capabilities are required to satisfy that relationship. From this, we can draw analogies from our past experiences to see what exact items have satisfied this relationship in the past.

Similarity Metrics

There are a variety of algorithms and approaches that can allow us to examine the experience we have captured for the purposes of experience-based reasoning. The approaches range from straightforward comparisons of values to deep, complex examination of relationships and structure. A variety of approaches should be examined to ensure the best possible choices for drawing analogies based on military planning situations.

K-Nearest Neighbor (kNN)

This approach is among the most popular choices for defining similarity in case-based reasoning. The K Nearest Neighbor algorithm works by analyzing a set of features defined by the implementer. Those features describe each case. After being analyzed, the distance between each case, based on those features, is known. By using K, a constant, the algorithm returns that number of cases which have the shortest distance from the target case [12]. In order to use K-Nearest Neighbor (kNN), the implementer needs three things. First, the implanter needs a set of features in each case, which are facets of the experience that can be observed and measured. Next, the implementer needs

a method by which those features can be compared between cases to determine how similar they are to each other. Last, a reasonable value for K must be chosen. The constant K denotes how many of the closest cases will be chosen for a target situation.

The kNN approach is actually deceptively simple; it basically comes down to comparing values that you have deemed important within the experience. In terms of military planning, this approach is best suited for easily measured, detail-oriented facets of a situation. Locations, resources, actors, times, etc., are all items within a planning context that can be observed and measured with many quantifiable attributes. The latitude and longitude of a target, the amount of fuel you have, the number of pilots in the barracks… all of these things can be measured and compared to each other in different situations.

The pros and cons of this approach are also fairly evident. A major benefit of this approach is that there is no requirement for classifying situations or classes or objects. Because kNN only cares about the measurable features, these classifications will emerge naturally. In other words, you do not need to know how to tell a computer what an airport is, because it will eventually notice that all locations with long runways, several airplanes, extra fuel, and air traffic control towers are all fairly similar. It will 'discover' airports, even though the computer would never have to be consciously aware of their existence. A major drawback of this approach, however, rests in the fact that the kNN approach is totally blind to such high-level concepts as airports. In does not understand the *relationships* between data, or how those relationships can give information a context, with that context leading to higher-order understanding of an experience. In other words, even though kNN is interested 'strictly in the facts', it is blind to the higher-order relationships and implications of those facts that could lead a mixed-initiative planning system to a deep understanding of a situation.

Semantic Similarity

If kNN leaves us high and dry on understanding the meaning of information, then perhaps we can examine that very meaning as the means by which we can draw analogies. Semantic similarity attempts to address that very detail [13]. By using ontologies and taxonomies, this form of similarity attempts to examine the difference between two objects in terms of those objects' semantic meaning. In other words, rather than examine the specific details of two objects, semantic similarity examines the meaning stored within the computer's vocabulary of meanings.

The advantage of this approach is that a mixed-initiative planning system could understand semantic distinctions (such as the existence of airports, how similar they are to train stations, and how dissimilar they are to toaster ovens) that usually are taken for granted in everyday thought and conversation. This would help a computer reason 'on par' with the more context-sensitive and meaning-driven human reasoner. However, while this form of similarity leads us to a deeper understanding of the similarity between objects, it leaves us high and dry in understanding the relationships between them. This same drawback was seen with the kNN approach; that this form of similarity is blind to context, even though this approach takes meaning into account.

What we require, then, is a form of analogy that does not only rely on the understanding of the similarity between objects, but also takes into account the relationships between them.

Structural Mapping

Important work in this style of analogy building has been ongoing. One interesting algorithm is the Structural Mapping Algorithm (SMA), invented by Diedre Gentner and Ken Forbus[15]. This algorithm relies on forming an analogy not just on the objects involved, but rather on their relationship to each other. In other words, SMA utilizes the structure of a representation, rather than simply the attributes of the constituent parts.

More specifically, SMA seeks to make analogies that adhere to the *systematicity principle*, which stipulates that matches are preferred when they involve higher-order relations rather than several lower-order facts. In other words, SMA matches are based not only on attribute matches (like similarity), but more importantly on true analogy, where the matching elements fit into a system.

However, this approach is not without its cost. This kind of keep analysis of a situation can be computationally taxing. In order to be thorough, perhaps thousands of tiny and otherwise implied relationships might need to be examined. A good approach, then, should be able to blend the best of both worlds. This blend should combine the computational speed of similarity between objects, and the analogical thoroughness of similarity using systematicity.

Many are Called, Few are Chosen

An interesting approach to drawing analogies in an efficient and interesting way is the Many are Called but Few are Chosen (MAC/FAC) approach designed by Gentner and Forbus from Northwestern University [4]. It combines a speedy similarity based approach with a deeper structural matching approach to optimize analogy drawing. This approach is interesting in two ways. First, it can make episodic reasoning more efficient in terms of matching and retrieval. Second, the approach itself can help a commander understand their priorities in planning. Let us first briefly describe the approach itself, and then this important second benefit.

In the MAC stage, each case is assigned a vector of numbers. This vector contains the number of occurrences of a given description of *functors* (predicates, functions, connectives) contained within the case. In other words, if a description contained *[DESTROYS, BIG]*, then each case would be assigned a vector that simply counted the number of occurrences of the function *DESTROYS* and the predicate *BIG* in the case. This is done regardless of the structure of the case; it is simply a count. Then, the MAC chooser takes a look at the vectors compared with the description and selects the best match and everything within 10% of that. These cases are passed onto the FAC stage to examine their structure.

In the FAC stage, the pruned case-base is subjected to structural matching with the current situation, as with the traditional Structural Matching approach, using similarity based matching. The combination of the two approaches allows the otherwise thorough and possibly costly Structural Matching algorithm to operate in a concise and focused way; eliminating wasteful work. However, the MAC/FAC approach operates on the assumption that a 'first-pass' with a desired description will eliminate cases with little value to the case to be matched. Bear in mind that if this assumption does not hold, then there is a possibility of prematurely eliminating useful results. However, Gentner and Forbus' experiments with the algorithm suggest that it is well-founded.

Using this approach, a commander could become involved in the retrieval of relevant cases by examining their priorities in a situation to help prune the search during the MAC phase. In other words, the MAC stage could help a commander set the 'red line' that their situation affords them. The number of troops, the available resources, and other constraints place real limits on the options available to a commander. By using the MAC stage, the commander can critically examine these elements of their situation, and only deeply examine cases that present options that fit within the realm of the possible.

Taking this approach one step further, we could approach planning from the perspective of examining constraints. In other words, given a problem that needs to be solved, and a set of resources with which to do it, how do I satisfy this purpose with the employment of those resources? This adds a third element, beyond object similarity and structural similarity, which is the purpose of a plan. We can use the important guide of utility to help us retrieve cases; asking not only are these two situations 'the same', but also asking if they solve the same problem.

Multi-Constraint Theory

Taking a merged approach of similarity between objects, structures, and purpose is being researched under an approach called multi-constraint theory [10]. This approach allows for the purpose of the analogy to be taken into account, not only the similarity of its constituent parts.

This is especially important in the context of analogies for military planning because purpose drives every planning situation. In other words, it is not very important in a vacuum that two things are similar, or even that two sets of relationships are similar. Without the guiding context of a purpose, two plans might look very much the same at face value, but accomplish two very different goals. This awareness of a commander's goal guides our continued development of both our experience representation and similarity metrics for our mixed-initiative planning system. As many thinkers have postulated: computers are not stakeholders. Because they do not understand purpose and are not invested in accomplishing any specific goal (outside of completing instructions), keeping the purpose of a plan in mind is important to making a mixed-initiative planning system not only accurate, but also trustworthy.

Summary

In military operations, a key objective of the command and control process is making sound and timely decisions. The process of decision-making can be broadly partitioned into two types, analytical (e.g., deliberate planning where time and information are sufficient) and intuitive (e.g., crisis action planning where time is critical and information is lacking). DEEP can support both decision making processes, but is especially well suited to the later and can be best described as Recognition-Primed Decision Making [14]. The primary means of experience-based reasoning developed the first year is Case-Based Reasoning (CBR). These cases contain the context of a problem, the solution employed, and the results of that solution.

CBR is a method by which we can leverage prior experience as a guide to possible solutions to a problem. However, the foundation of good CBR is a solid case-base. Formalizing how an experience is captured, stored, and reasoned upon is the brick and mortar upon which a potentially powerful decision aide could be built. By exploring and adapting existing plan representation work, we can formalize a case structure for military planning. Then, by employing analogy-based matching algorithms, we can begin to understand how to make insightful connections between the past and the present, in order to afford a commander an intuition about the future.

Future Work

The objective of the first year of the DEEP project was quickly developing a “breadboard” as an initial capability to conduct experiments on experience-based mixed-initiative planning. The approach taken was the modest integration of CBR for experience, and a blackboard for mixed-initiative planning. Beginning the second year of the effort 2008-2009, the CBR will be either replace or supplemented with a more powerful episodic memory framework and a more semantic analogical reasoning engine. The development of a powerful structured analogy capability is possibly the most challenging aspect of the DEEP project. The traditional AI blackboard we built using open source tools in Java will be upgraded by laying the blackboard services onto an Oracle database and using object-relational mapping tools. The rational for using Oracle is based on issue dealing with building an operational prototype for testing in a realistic environment (e.g., JFEX 2010).

Other possible research activities include formalizing the measurement of trust and confidence in information for making assumptions, determining the right level of fidelity in information capturing to ensure good case matches, determining the best way to capture a world state to provide context, and methods by which to capture cases from written (or lived) historic war stories. Another extension could be into hybrid case reasoning systems.

References

[1] Burstein, Mark H. and McDermott, Drew V. (1994) “Issues in the Development of Human-Computer Mixed-Initiative Planning Systems.” Proceedings of the DARPA/RL Planning Initiative, February 1994, Chapter 17.

[2] Cowan, Major Jeffrey L.. "From Air Force Fighter Pilot to Marine Corps Warfighting: Colonel John Boyd, His Theories on War, and their Unexpected Legacy." Defense and the National Interest. 2000. United States Marine Corps Command and Staff College. 4 Apr 2007 <http://www.d-n-i.net/fcs/boyd_thesis.htm>.

[3] Forbus, Ken, Dedre Gentner, and Brian Falkenhainer. "The Structure-Mapping Engine: Algorithm and Examples." Artificial Intelligence 41(1989): 1-63.

[4] Forbus, K., Gentner, D. & Law, K. 1994b MAC/FAC: A model of similarity-based retrieval. *Cognitive Science* **19**(2), 141-205.

[5] Pease, Adam R. “Object Model Working Group Core Plan Representation” Armstrong Laboratory Technical Report. September 1996

[6] Pease, Adam R. "Core Plan Representation - Request for Comment Version 4." Core Plan Representation. 06 NOV 1998. Teknowledge. 5 Feb 2007 <http://projects.teknowledge.com/CPR2/Reports/CPR-RFC4/>.

[7] Pease, Adam. "Object Model Focus Group: Core Plan Representation." Computer and Information Science Papers CiteSeer Publications Research Index. 06 NOV 1998. CiteSeer. 12 Feb 2007 <http://citeseer.ist.psu.edu/cache/papers/cs/5542/http:zSzzSzwww.teknowledge.comzSzCPR2zSzReportszSzCPR-RFC4zSzcpr4.pdf/core-plan-representation.pdf>.

[8] Pease, Adam, and Capt Todd M Carrico. Object Model Working Group Core Plan Representation. Dayton OH: United States Air Force Armstrong Laboratory, 1996.

[9] De Mantaras, Ramon Lopez et al.. "Retrieval, reuse, revision, and retention in case-based reasoning."The Knowledge Engineering Review 00(2005): 1-32.

[10] Thagard, Paul, and Karsten Verbeurgt. "Coherence as Constraint Satisfaction." Cognitive Science 22(1998): 1-24.

[11] Klir, George, and Bo Yuan. Fuzzy Sets and Fuzzy Logic: Theory and Applications. 3rd. Upper Saddle River, NJ: Prentice Hall PTR, 1995.

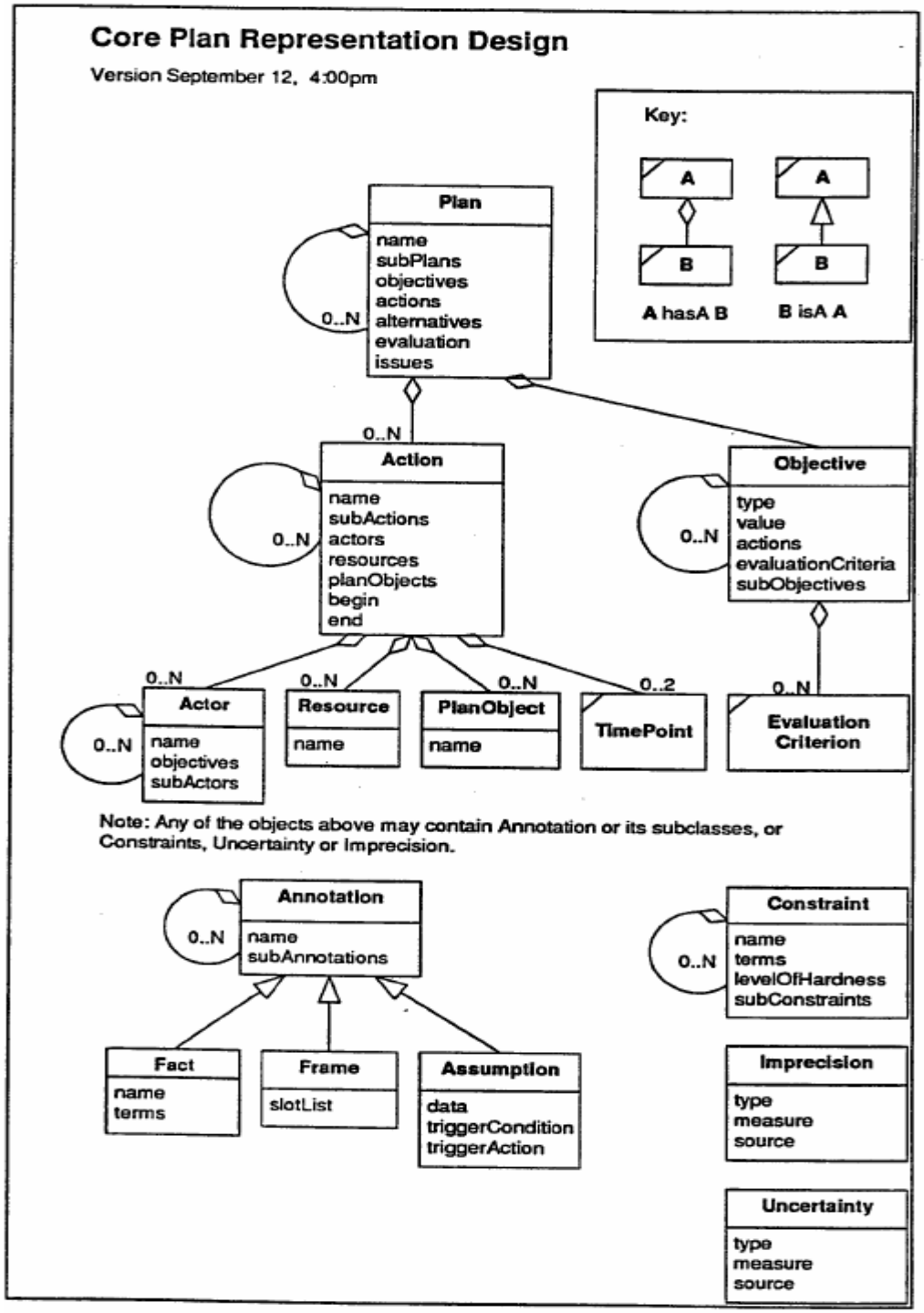
[12] Fix, Evelyn , and L. Hodges, Jr.. "Discriminatory Analysis. Nonparametric Discrimination: Consistency Properties ." International Statistical Review / Revue Internationale de Statistique 57(1989): 238-247.

[13] Philip, Resnik. "Semantic Similarity in a Taxonomy: An Information-Based Measure and its Application to Problems of Ambiguity in Natural Language." Journal of Artificial Intelligence Research 11(1999): 95-130.

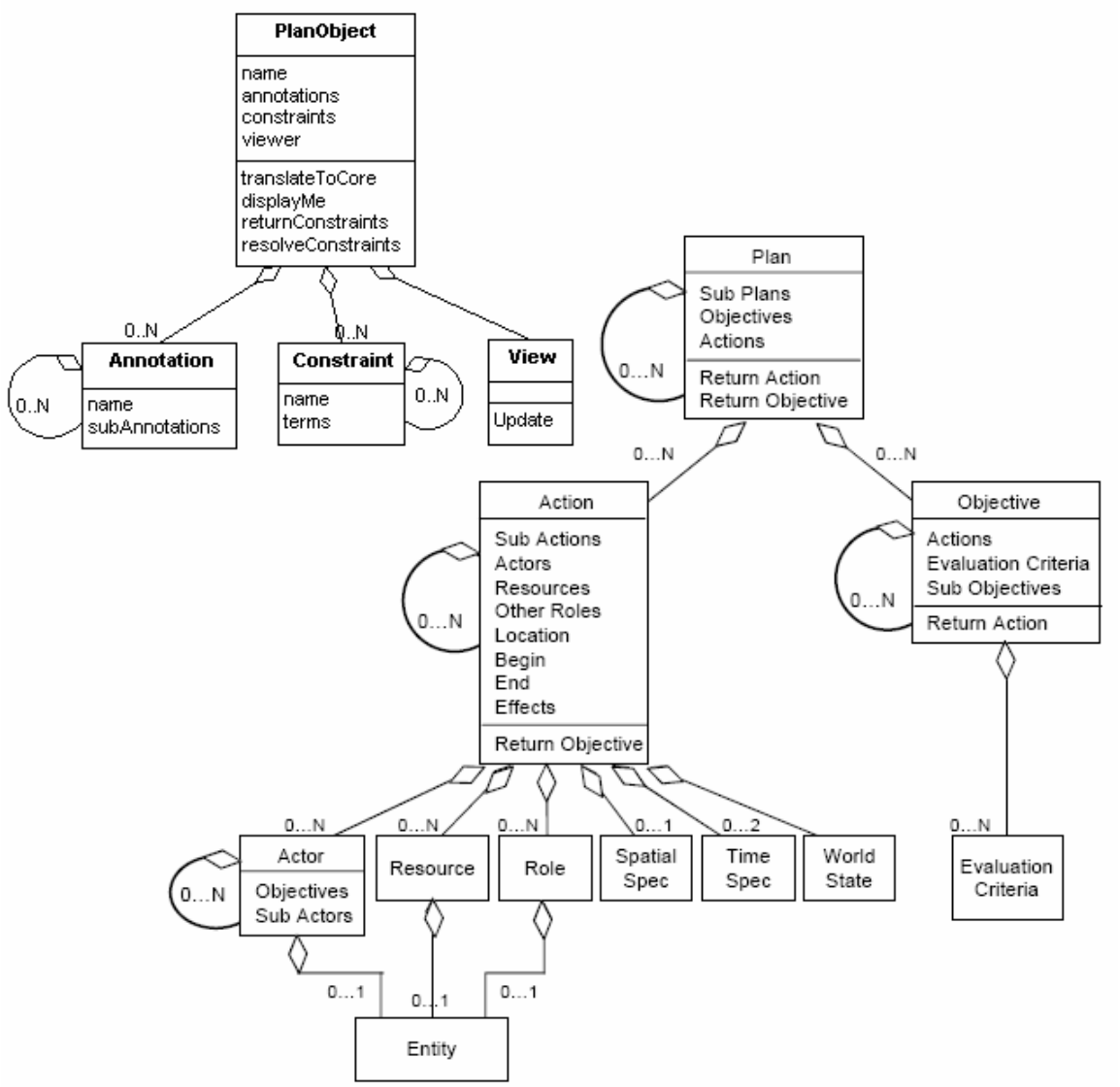
[14] Klein, Gary. “Sources of Power”, The MIT Press, 1998

[15] [Falkenhainer et al., 1990] Brian Falkenhainer, Kenneth D. Forbus, and Dedre Gentner. TheStructure mapping engine: algorithm and examples. Artificial Intelligence, 41:1-63, 1990.

Appendix A: 1996 CPR Model



Appendix B: 1998 CPR Model



Appendix C: Task-method decomposition of CBR

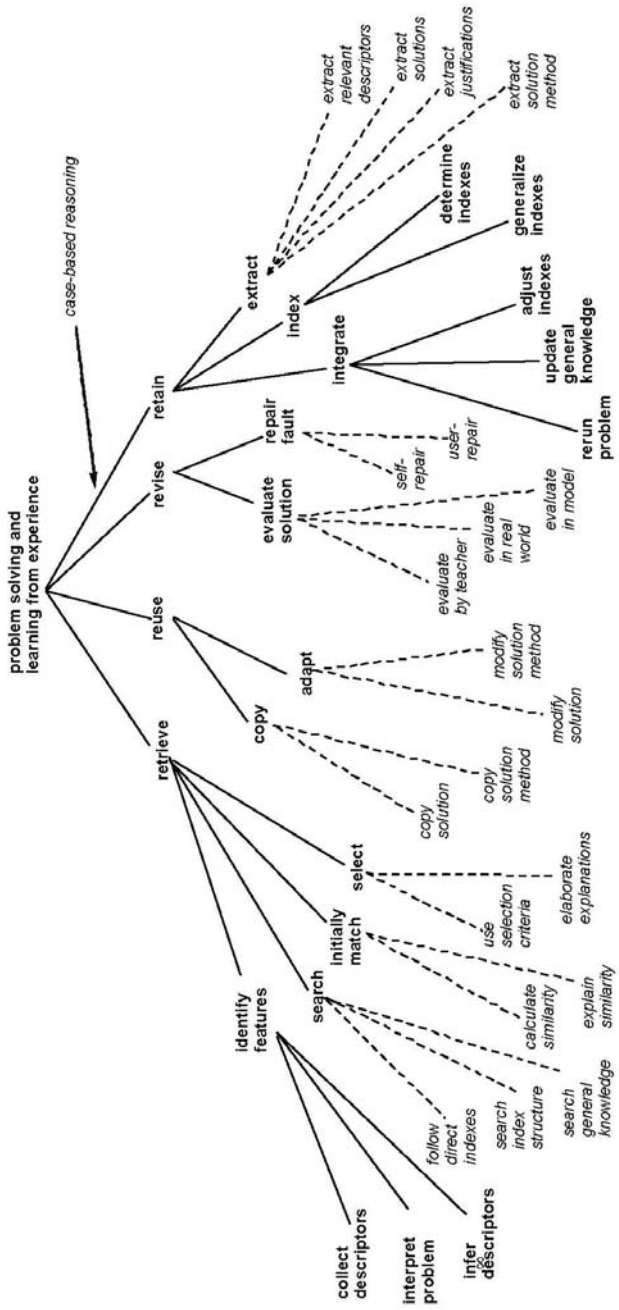


Figure 2. A task-method decomposition of CBR