A uniform framework for describing and analyzing the modern battlefield[*]

FEASIBILITY STUDY, July 2011

Keith Devlin
Stanford University

Abstract

This paper presents a preliminary report of a feasibility study of a project to develop a uniform descriptive framework that:

1. captures all of the complexity of the modern battlefield;
2. is capable of sufficient precision to allow for cross-domain communication that meets the needs of all parties;
3. allows for clarification and disambiguation;
4. is sufficiently natural and intuitive for rapid mastery by all parties.

Our framework is based on the adoption of an information-flow view of the battlefield, using situation theory.

Introduction

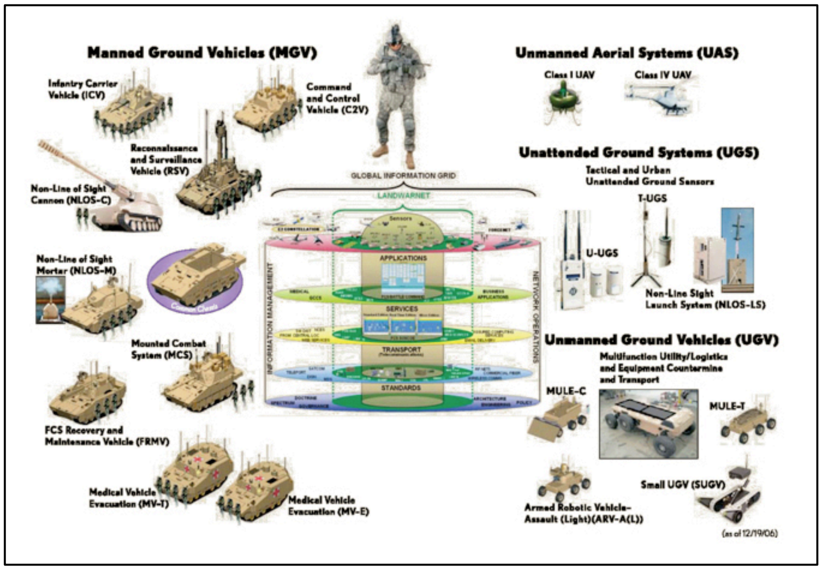


Figure 1. The complexity of the modern battlefield

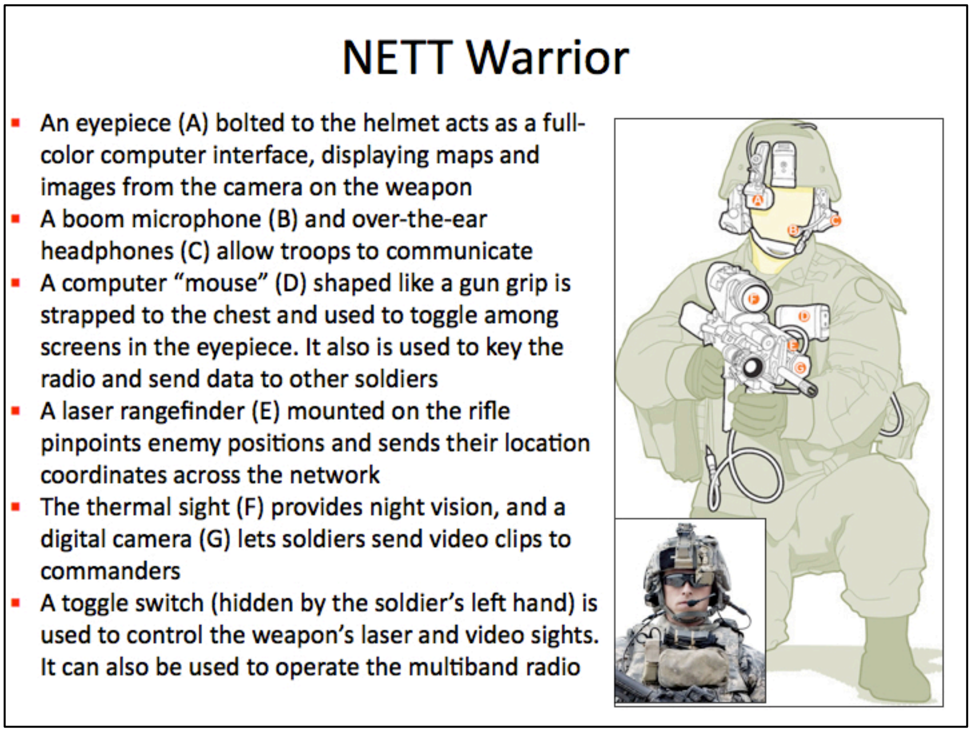
The modern battlefield ranges from vast stretches of unpopulated open desert, to tall mountain ranges, to densely populated cities. Present-day combat operations

---

[*] This material is based upon work supported by, or in part by, the U.S. Army Research Laboratory and the U.S. Army Research Office under contract/grant number W911NF1110052.

involve a variety of weapons, some carried by troops, others on vehicles of various kinds, both ground and airborne. Some weapons are operated on site by one or more personnel, others function automatically, and still others are controlled remotely by human operators who may be thousands of miles removed from the battlefield. Sensors gather information; they can be static or mobile, ground based or airborne. Cameras borne by personnel, ground vehicles, aircraft, or satellites provide visual information. (See Figure 1.) Battlefield operatives are often trained to high degrees of specialization that involve the use of function-specific, technical terminologies.

A particular layer of complexity is the information flow. While information has always played a role in military activities, in battles of the past the actual combat was largely separate from the gathering and distribution of information and the associated planning and decision making. In today's world, however, the information flow occurs in real time and is an integral part of almost every aspect of military activity. This is perhaps best iconized by the Nett Warrior (see Figure 2), the new ground soldier system announced in June 2010, and named in honor of World War II Medal of Honor recipient Col. Robert B. Nett.



NETT Warrior

- An eyepiece (A) bolted to the helmet acts as a full-color computer interface, displaying maps and images from the camera on the weapon
- A boom microphone (B) and over-the-ear headphones (C) allow troops to communicate
- A computer "mouse" (D) shaped like a gun grip is strapped to the chest and used to toggle among screens in the eyepiece. It also is used to key the radio and send data to other soldiers
- A laser rangefinder (E) mounted on the rifle pinpoints enemy positions and sends their location coordinates across the network
- The thermal sight (F) provides night vision, and a digital camera (G) lets soldiers send video clips to commanders
- A toggle switch (hidden by the soldier's left hand) is used to control the weapon's laser and video sights. It can also be used to operate the multiband radio

Figure 2. Nett Warrior, the Army's new ground solider system, announced in 2010.

This complexity presents a challenge not only for those directly involved in combat, but to those whose task is to plan, to analyze, or to study military operations.

The challenge is particularly acute for a unit such as SLAD, which has to take a comprehensive, birds-eye view of combat operations, both actual and simulated, factual or hypothetical. Actions and behaviors should ideally to be described with precision in a manner that is comprehensible to the different domain experts and to modeling and simulation software developers. Failure to communicate adequately can lead to problems and delays. M&S personnel report that it can take considerable time and effort to understand a customer's real need, and misunderstandings are common as different groups "talk past one another."

This paper presents a preliminary report of a project that seeks to develop a uniform descriptive framework that:

1. captures all of the complexity of the modern battlefield;
2. is capable of sufficient precision to allow for cross-domain communication that meets the needs of all parties;
3. allows for clarification and disambiguation;
4. is sufficiently natural and intuitive for rapid mastery by all parties.

In particular, with reference to condition 4 above, the first level of formality the framework provides is of a stylized natural language. Thereafter, the formality can be increased one degree at a time. There is in principle no limit to the degree of formality adopted – it can go all the way to complete mathematical formality expressed in formal logic. Of course, for any real-world domain, expression of all activities at the level of formal mathematics becomes impossibly cumbersome. (Due to the phenomenon known as combinatorial explosion). Rather, successful use of the framework will involve an analytic technique known as *zooming*, described (using two real-world examples) in Devlin & Rosenberg 1996.

Use of our framework entails an information-flow view of all aspects of the battlefield. Experts in some domains may find this approach unusual, but that will be due largely, we believe, to the ***explicit*** role information flow plays in our framework. For in reality, military planning and command has always been about information flow. Our approach merely takes that to a further level, adopting a formalized, abstract, information-flow framework as a ***lingua franca*** for analysis and planning. As a follow-on project, we recommend the design and development of a suite of software tools that implement our abstract framework in usable tools (with natural interfaces that build on existing expertise and competencies) for analysis and planning.

Situation theory

Our framework is a novel use of situation theory, a mathematical theory of information developed in the early 1980s and subsequently. Because it builds on everyday intuitions about information and information flow, it can be used to increase the precision of analyses at various levels of granularity. For example, in our book *Infosense*, we use situation theory as a basis for what is essentially a layperson's level analysis of reasoning and decision making in a number of domains, including military campaigns, air-traffic control, and business management. At the other end of the formality spectrum, in our joint book *Language at Work* (1996), Rosenberg and I used the theory as the basis for a mathematically-precise, in-depth analysis of a problem in the management of field repair work in the computer industry.

Situation theory has been applied successfully in natural language processing [3], [5], [11], information systems analysis and design [4], [6], [9], [13], development of ontologies [15], business process analysis [6], [9], [10], manufacturing process modeling [12], mathematical deduction [1], [2], context-influenced reasoning [6], [7], [8], [9], [10], [14], and multi-agent reasoning [7], [8]. A recent study of its potential use in defense intelligence analysis (ARDA's NIMD Project) [8] produced positive results, and we are engaged in an ongoing continuation of that work focused on the design of a reasoning-support system for the analysis of video intelligence data.

Situation theory begins with the questions, what is information and how does it arise? How is it possible for something in the world, say a book or a magnetic disk, to store, or represent information, and what does that mean? Providing an answer to those questions can fill an entire book (your current author has written two of them), but for our present purposes we can work with the intuitive concept of information that we use every day. The representation or storage of information then depends upon certain rules, regularities, protocols, etc. For example, to acquire information from the words and sentences of English, you have to understand English – you need to know the meanings of the English words and you need a working knowledge of the rules of English grammar. In addition, in the case of written English, you need to know how to read – you need to know the conventions whereby certain sequences of symbols denote certain words.

In general, anything can be used to store information. All it takes to store information by means of some object – or more generally a configuration of objects – is a convention that such a configuration represents that information.

In the late 1970s, two Stanford University professors, Jon Barwise and John Perry, started to develop a mathematical framework to make these ideas precise. The result was Situation Theory, initially described in their book *Situations and Attitudes*, (Barwise and Perry 1983) with a more developed version of the theory subsequently presented by us in our book *Logic and Information* (Devlin 1991).

Situation theory takes its name from the mathematical device introduced in order to take account of context. A situation can be thought of as a limited part of reality. Such parts may have spatio-temporal extent, or they may be more abstract, such as fictional worlds, contexts of utterance, problem domains, mathematical structures, databases, etc.

The basic ontology of situation theory consists of entities that a finite, cognitive agent individuates and/or discriminates as it makes its way in the world: spatial locations, temporal locations, individuals, finitary relations, situations, types, and a number of other, higher-order entities.

The objects in this ontology include the following:

individuals – objects such as tables, chairs, tetrahedra, people, hands, fingers, etc. that the agent either individuates or at least discriminates (by its behavior) as single, essentially unitary items;
relations – uniformities individuated or discriminated by the agent that hold of, or link together specific numbers of, certain other uniformities;
spatial *locations* – these are not necessarily like the points of mathematical spaces (though they may be so), but can have spatial extension;
temporal locations – as with spatial locations, temporal locations may be either points in time or regions of time;
situations – structured parts of the world (concrete or abstract) discriminated by (or perhaps individuated by) the agent;
types – higher-order uniformities discriminated (and possibly individuated) by the agent;
parameters – indeterminates that range over objects of the various types.

The intuition behind this ontology is that in a study of the activity (both physical and cognitive) of a particular agent or species of agent, we notice that there are certain regularities or *uniformities* that the agent either individuates or else discriminates in its behavior. For instance, people individuate certain parts of reality as *objects* ("individuals" is situation theory), and their behavior can vary in a systematic way according to spatial location, time, and the nature of the immediate environment ("situation types" in the theory).

In general, we write

$$a{:}T$$

to indicate that entity a is of type T.

Information is always taken to be information *about* some situation, and is taken to be built up from discrete items known as *infons*. These are of the form

$$\langle\langle R, a_1, \ldots, a_n, i \rangle\rangle$$

where R is an n-place relation, $a_1, \ldots, a_n$ are objects appropriate for R (often including spatial and/or temporal locations), and i = 0 or 1. These may be thought of as the informational item that objects $a_1, \ldots, a_n$ do, respectively, do not, stand in the relation R, depending on whether i = 1 or 0.

Infons are items of information (hence, more precisely should be called data). They are not things that in themselves are true or false. Rather a particular item of information may be true or false about a certain part of the world (a situation).

A fundamental assumption underlying the situation-theoretic approach to information is that information is not intrinsic to any signal or to any object or configuration of objects in the world; rather information arises from interactions of agents with their environment (including interactions with other agents). The individuals, relations, types, etc. of the situation-theoretic ontology are (third-party) theorist's inventions. For an agent to carry out purposeful, rational activities, however, and even more so for two or more agents to communicate effectively, there must be a substantial agreement first between the way an agent carves up the world from one moment to another, and second between the uniformities of two communicating agents.

The objects in the ontology of situation theory are intended to be theorist's idealized representatives of the common part of the extensions of individual agent's ontologies. In consequence, the infons are theoretical constructs that enable the theorist to analyze information flow.

Situation theory provides various mechanisms for defining types that will become clear as our development proceeds. For more details, we refer to Devlin 1991, Devlin 1999, and Devlin & Rosenberg 1996.

In using situation theory to capture all aspects of the modern battlefield, we need to represent several different ontologies in situation-theoretic terms:

- **Terrain**: the features of the physical space on which the battle takes place
- **People**: missions, rules, commands, actions, capabilities and expertise
- **Weapons/equipment**: operation, capabilities, effects, accuracy, reliability
- **Vehicles**: manned and unmanned, ground and aerial, conditions for use, reliability
- **Supplies**: many different suppliers, each with its own ontology
- **ICT**: many different information and communication technologies
- **Information**: information itself is a commodity used on the battlefield, including information about people, weapons, terrain, and vehicles.

It is precisely to overcome the communication difficulties of having all of these different ontologies interact that we are advocating the adoption of a single uniform framework.

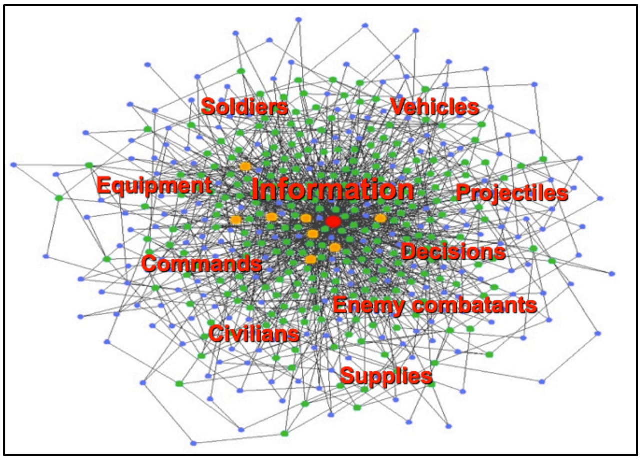


Figure 3. A dynamic-network conception of the modern battlefield

One way to understand our framework is to view the battlefield as a dynamic network in which various different kinds of entities flow and interact, as illustrated in Figure 3. To achieve a uniform description, we provide an informational specification of each action by (or involving) each entity in the field.

The uniform specification device we adopt is what we shall refer to as the *satisfaction diagram*. This is a novel use of a mathematical structure that has played a fundamental role in situation theory since the very beginning. Barwise originally introduced it to understand information flow.

The satisfaction diagram

We begin with a specific example. If we see smoke coming from a nearby forest, we can infer that there is (or very likely is) a fire. How can we do that? The answer is, of course, that there is a systematic relationship between situations where there is smoke and situations where there is a fire. We may have become aware of that relationship by personal experience, by being told it by others, or through some other means. But once we know it, we can henceforth infer the existence of a fire from the evidence of smoke. We can represent this state of affairs by the diagram in Figure 4.

Since the ideas captured in this diagram will play a fundamental and pervasive role in our entire analysis, it is worth devoting some time on this simple example.

First, note that we use ovals to denote abstract types (in this example, situation types) and rectangles to denote actual entities in the world (in this example, an actual situation).

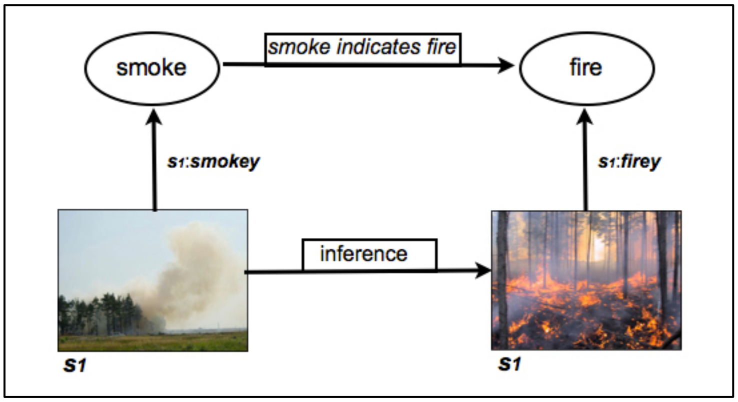


Figure 4. Satisfaction diagram for the inference of fire from smoke

The left-to-right arrow at the top represents the regularity in the world that enables cognitive agents like people (and many species of animal) to infer the existence of fire from the evidence of smoke. In situation theory, this rule is represented by a binary relation called a *constraint* that connects two *situation types*, the type of situation where there is smoke present and the type of situation where there is a fire. The upward-directed arrow on the left indicates that the situation s_1 in the bottom left is of *type* smokey. An agent aware of (or more generally attuned to) the constraint represented by the top left-to-right arrow, that can classify the situation s_1 it sees as being of type smokey, can then infer that the situation is in fact of the type firey, as represented by the right-hand up-arrow. (With this particular example, only one situation is involved, s_1, but in general the constraint will lead to a second situation of the appropriate inferred type.)

Notice that the vertical arrows connect two very different domains. The top half of the diagram represents an abstraction, a constraint, in this example a behavior-guiding principle in the cognitive apparatus of a rational agent. The bottom half is firmly rooted in physical reality, namely an actual part of the world in which there is a fire giving off smoke. The agent can see the smoke but not the fire. The agent is able to *infer* the existence of the fire — to see in its mind's eye what it would see in reality were it closer to s_1 — by virtue of the typing represented by the left-hand up-arrow. The diagram captures the fact that cognitive agents make inferences by particularizing from general abstractions (constraints that link types) to actual situations.

Figure 5 shows the general case in which an agent is able to make an inference based on sensory input.

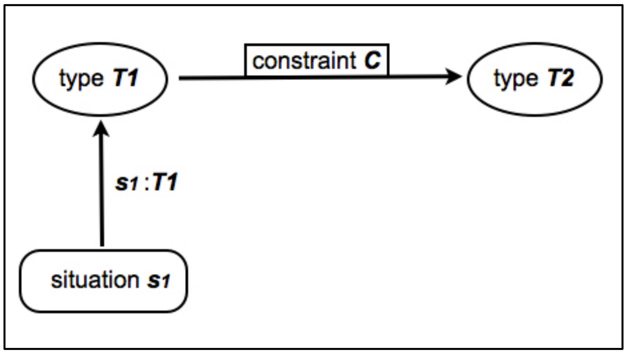


Figure 5. Satisfaction diagram, Stage 1

The agent recognizes that the situation *s*1 is of type *T*1, and is aware of, or attuned to, the constraint *C* that links type *T*1 to type *T*2. The agent is then able to infer the existence of a situation *s*2 of type *T*2, as represented in Figure 6.

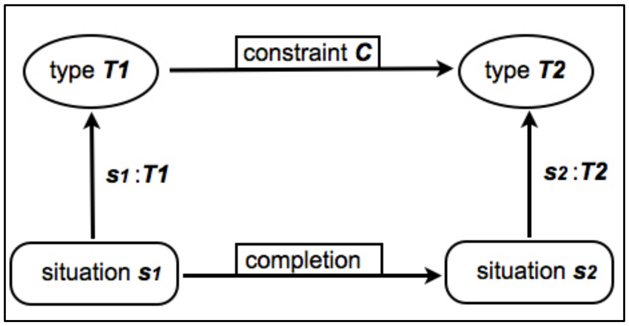


Figure 6. Satisfaction diagram, Stage 2

The connection between the situation *s*1 and the situation *s*2 represented by the lower left-to-right arrow, what we call the *completion* of the diagram, results from the agent "pulling down" from the abstract constraint linking two types to an actual inference from one real-world situation to another.

Constraints thus (can) represent general, cognitive templates that agents can draw upon to make inferences in actual situations. The satisfaction diagram captures this process of particularizing from abstract constraints to concrete inferences. We refer to the process of completing an inference in this fashion as *satisfaction* of the diagram; see Figure 7.

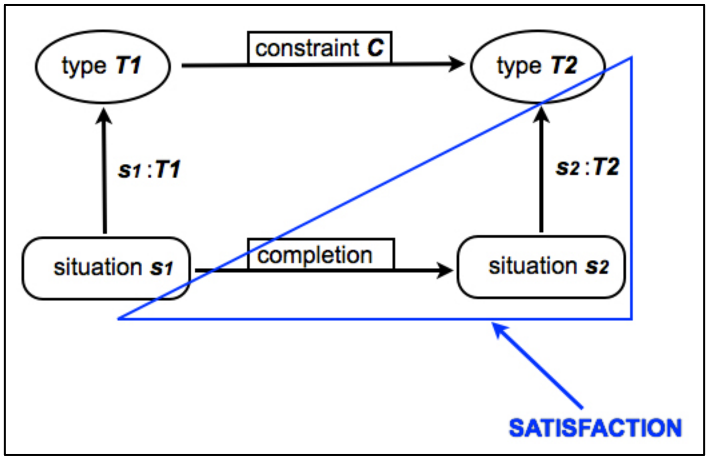


Figure 7. Satisfaction diagram, showing the satisfaction step.

The satisfaction diagram applies to other kinds of scenario besides inference; following commands, for example. Figure 8 captures what occurs when a motorist obeys the rule that when a traffic light is red, she or he must stop the car.

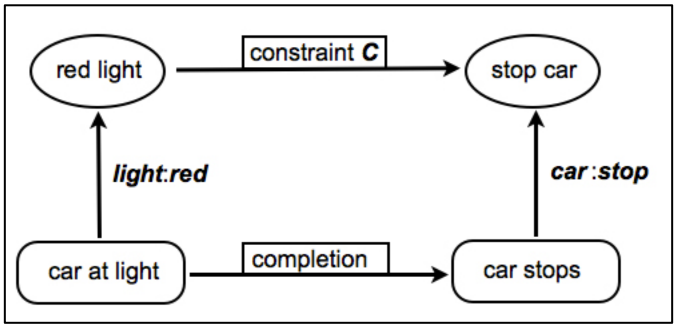


Figure 8. Following commands. The traffic-light example.

The bottom left refers to an actual car coming to a traffic light. The driver recognizes that the light is red, indicated by the left up-arrow that shows the automobile situation (i.e., the situation comprising the car and its immediate environment, which contains the traffic light) is of the type when the light it red. The driver knows the rules of the road, so in particular she knows the constraint *C*, and knows she is obliged to follow those rules, and so she drives in accordance with constraint *C*. She therefore acts in order to bring the automobile situation to a state that is of the type referred to by the right-hand up-arrow. Namely, she stops the car.

In this example, the bottom-left situation is the one where the car is approaching the traffic-light, the bottom right is the one when and immediately after the car reaches the light. The two situations are successive time-slices of the same spatial region. This is typical of how constraints operate when they function as commands or instructions. The agent acts to ensure compliance, i.e., to satisfy the diagram.

Although our two examples are extremely simple ones, the very fact that the same information-flow structure fully captures both inference and obeying commands should indicate that the satisfaction diagram is a powerful concept. Indeed, during the thirty-years since situation theory was first introduced, it has been used successfully in many different application domains. What is unique about our present work is that we propose using it not to support an analysis of one particular phenomenon, rather to provide a unifying framework to study a complex domain, namely the modern battlefield, where, as shown in Figure 3, entities in many different ontologies interact.

Applying the satisfaction diagram

As a mathematical structure, the satisfaction diagram is extremely simple, but its simplicity hides a lot of power and depth. The same is true of the commutative diagram of category theory, on which it was originally based. The complexity inherent in the modern battlefield is captured by the way satisfaction diagrams nest and fit together into a complex network. Their role in a battlefield analysis (or plan) is somewhat reminiscent of *Lego* blocks. Although the basic *Lego* blocks are all identical, by fitting them together appropriately, objects of considerable complexity can be created. In this way, the satisfaction diagram provides a powerful *lingua franca* for discussion, analysis, and planning of modern military activities.

By way of an example, we'll begin with a well known example of a recent, highly successful military operation, the killing of Osama bin Laden. It was initiated by a Presidential directive, so at the first level of analysis, its execution is captured by Figure 9.

According to public pronouncements, the directive was to capture or kill the al Queda leader, so its goal could have been met in either or two ways. At an initial level of analysis, we can combine both goals into a single one "capture or kill." The operatives involved had to select the specific action in real-time, based on conditions on the ground at the time. That part of the operation can also be represented by a satisfaction diagram, which is implicitly nested in the one shown.

[IMPLEMENTATION NOTE: Use of our framework by military planners will require the provision of a software system for real-time, interactive reasoning with situation diagrams. Designing such a system would be a significant project in itself, but since the basic, "*Lego*-block" constituent is the same for all stages of the analysis, we are confident that current technology readily supports the construction of such a system. In particular, the tough-screen interface controls offered by Apple's *iOS* seem ideal for that purpose. Although the satisfaction diagram is simple and intuitive, we do not rule out adoption of a different on-screen presentation of the same information for an interactive digital reasoning tool. That would surely be part of any subsequent R&D project. But the back-end reasoning engine could, and we think should, be designed around that one component — for the simple but powerful reason that the satisfaction diagram has proved itself highly useful in a range of applications.]

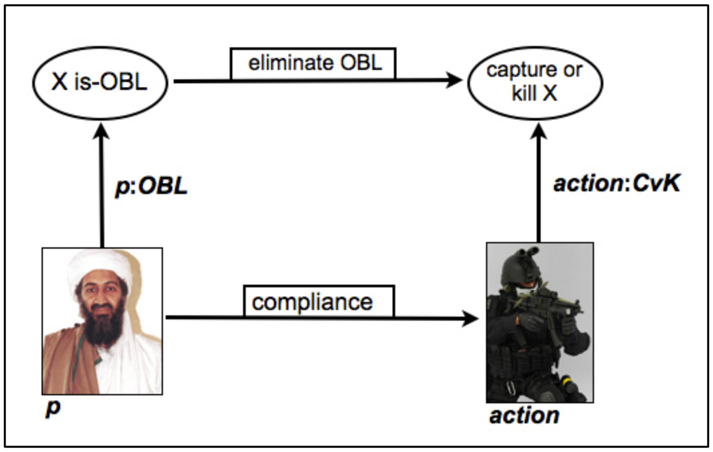


Figure 9. Successful completion of a Presidential directive.

Completion of the OBL mission (i.e., action by the operatives that ensured satisfaction of the diagram) depended on the original typing: the operatives had to decide that the individual before them was indeed Osama bin Laden, based on visual identification. Success of the operation hinged upon that initial typing being correct. Application of a satisfaction diagram assumes correct typing. This does not mean, however, that the framework breaks down in the presence of incorrect information or false typing. Indeed, the possibility of error can be built in to the diagram, should that be important.

For example, following the successful completion of the OBL mission, the White House announced that the decision to launch the operation in the first place depended on identification of the right location with a probability of at least 0.6 that OBL was there (at the time). This can be represented by Figure 10.

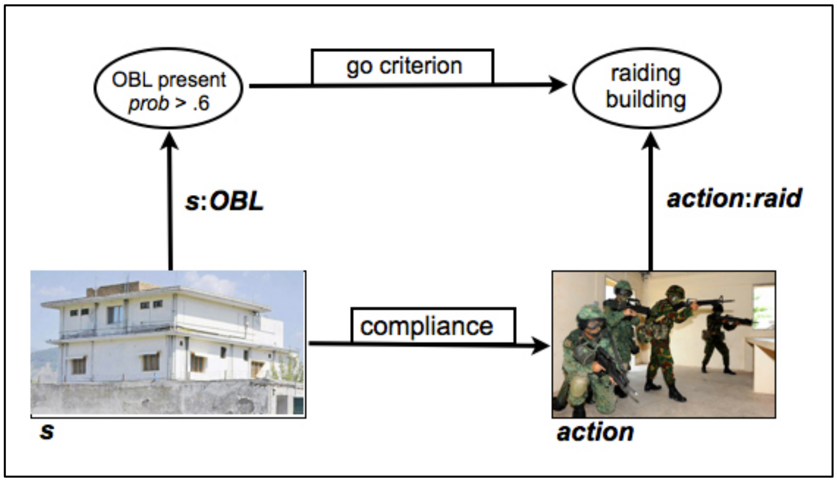


Figure 10. Decision to launch attack based on probabilistic information.

Application of the satisfaction diagram in Figure 10 will be preceded by an initial filtering process that restricts attention to locations of a type that could hide a wealthy fugitive. In particular, selection of the house shown (the one that, as events transpired, did house OBL) will come after the candidate situation has passed the filter shown in Figure 11.

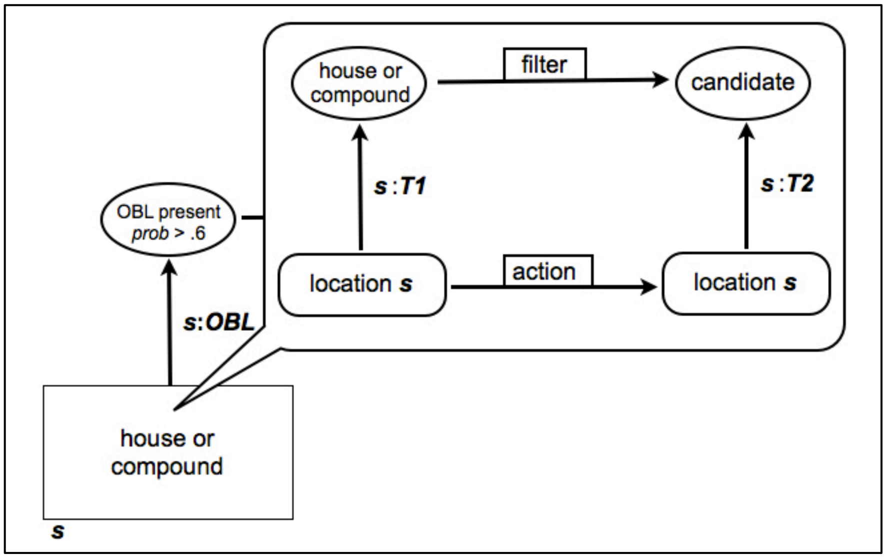


Figure 11. Selection of possible target locations.

[IMPLEMENTATION NOTE: In general, we envisage every satisfaction diagram allowing for expansion on any of its entries, with each expansion yielding one or more constituent or prior satisfaction diagrams. Thus the comparison with *Lego* blocks in not perfect. In our framework, each *Lego* block can be broken down into component blocks, each identical in structure to the parent block.]

Figure 12 illustrates another application of the satisfaction diagram, this time the successful operation of a precision-targeted mortar.

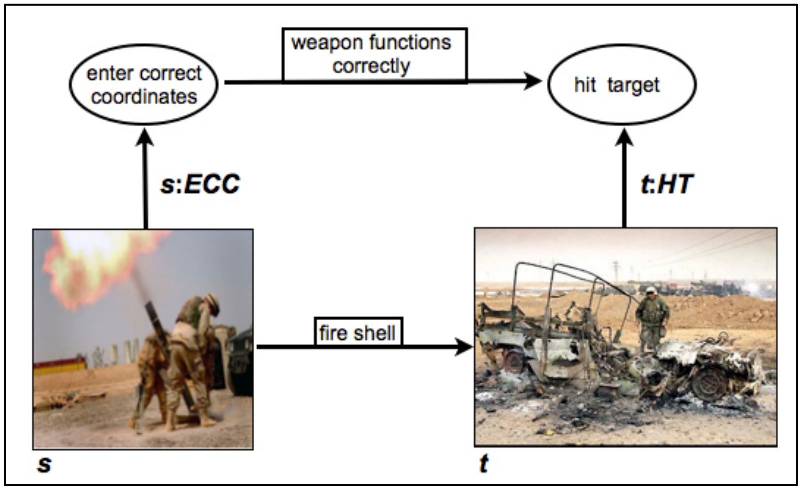


Figure 12. Weapon operation: the 120mm HE APMI is use.

Killing a high-value target in Iraq

Figure 13 shows a traditional aerial terrain view of a hypothetical mission to locate and kill a HVT in Baghdad. The representation is a familiar one, indeed a classic “overview.” But that familiarity hides a great deal of complexity, since it is a highly rich representation that includes many different kinds of entities. Different domain experts will understand the diagram in different ways. For instance, while M&S analysts may find this representation the most suited to their needs, ground personnel are likely to zoom down to a street-level view to provide the understanding they require.

The framework we are advocating is in no way intended to replace these traditional views of the battlefield. Rather, when implemented in a suitably designed software system, our framework is intended to enhance understanding and facilitate communication among different domain experts, by providing a uniform description that is neutral to all the domain-expert perspectives.

[IMPLEMENTATION NOTE: We envisage the reasoning support system view – which directly interfaces the back-end satisfaction-diagram reasoning system – being provided in conjunction with the view each domain expert is familiar with

Figure 13. Mission: Locate and kill a HVT in Baghdad. Traditional view.

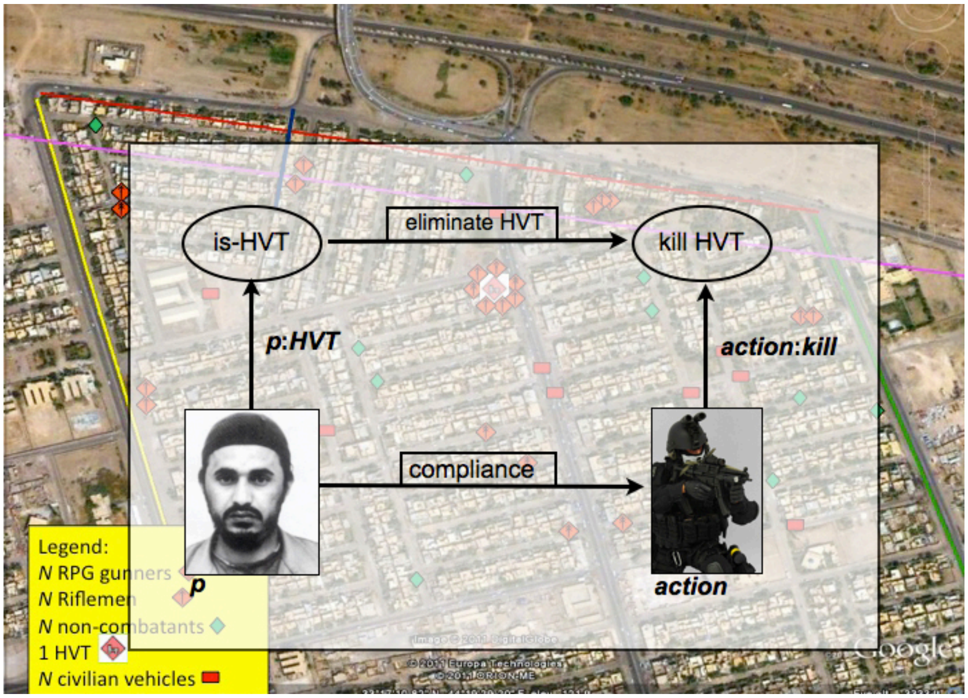


Figure 14. Mission: Locate and kill a HVT in Baghdad. Satisfaction diagram, top level view.

and preferentially selects, ideally linked so that changes made to one produce corresponding changes to the other. In this discussion we are showing the satisfaction diagram view overlaid on top of the traditional view. This view emphasizes the fact that our framework provides a lens or filter through which domain experts can view the battlefield in a domain-neutral fashion. In practice, the reasoning system may need to display several levels of the satisfaction diagram representation alongside one or more traditional views, with the human analyst or planner able to toggle between them in real time.]

Figures 14 and 15 show the first two levels of analysis using the satisfaction diagrams.

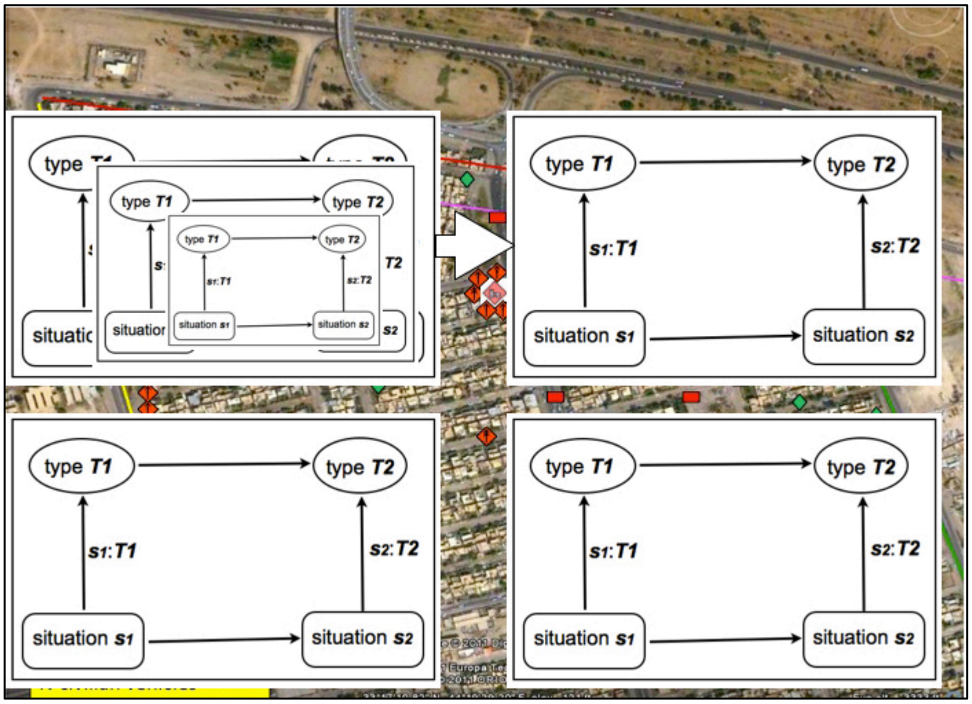


Figure 15. Mission: Locate and kill a HVT in Baghdad. Satisfaction diagram, 2nd level view.

In Figure 15, the single top-level mission statement shown in Figure 14 has been broken down into constituents, such as target identification, determination of troops required to complete the mission, backup plans, perimeter security measures, etc. At this stage, the diagrams will be nested and linked in various ways. This is just the normal planning process, with the added twist that in parallel to the familiar process there is a planning sequence in the *lingua franca* of satisfaction diagrams. Experiences with such processes in other domains suggest that this can yield more than minimize domain-domain miscommunication, it can lead to better planning within the domains themselves. Forcing planners and analysts to express their thoughts in a mathematically-

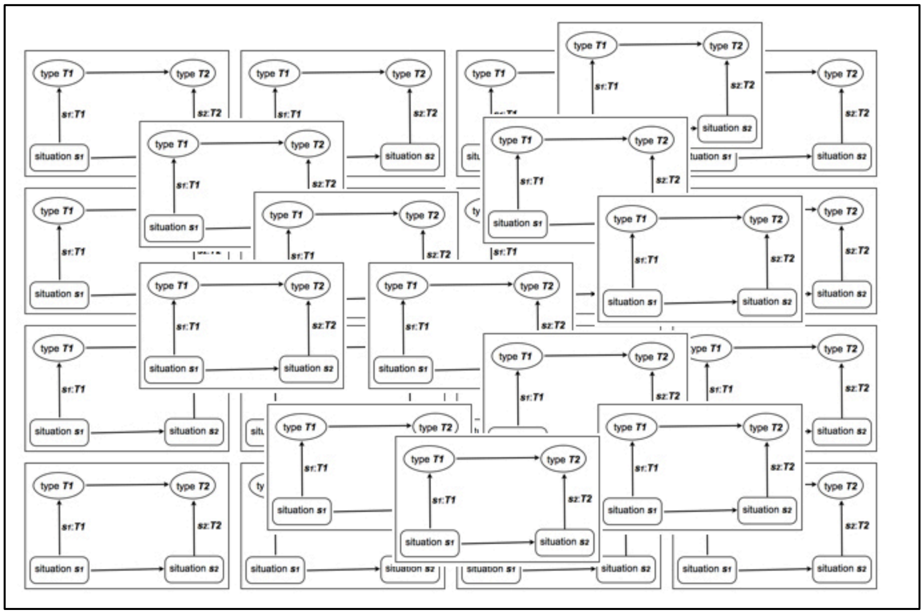
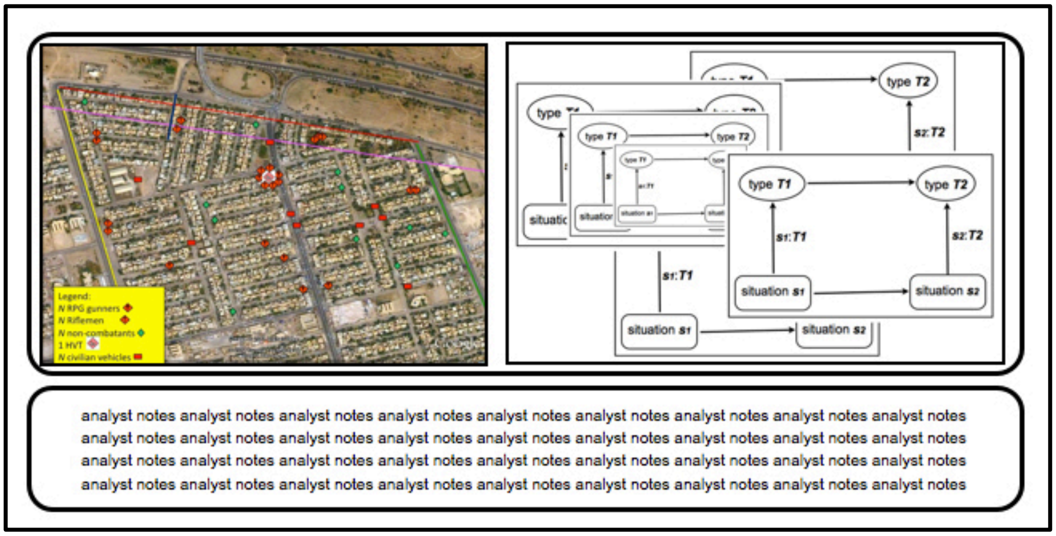


Figure 16. Mission: Locate and kill a HVT in Baghdad. Satisfaction diagram, deeper view.

based framework can highlight inconsistencies, identify missing steps, and uncover hidden assumptions.

If allowed to develop unfettered, the satisfaction-diagram representation will rapidly resemble Figure 16. Under some circumstances, this may be appropriate, but in general we envisage the reasoning being carried out using a zooming technique (as described fully in our monograph Devlin & Rosenberg 1996), where the analyst repeatedly zooms in and out of the detail for different parts of the analysis, thereby avoiding screen clutter. The reason for showing this illustration here is to emphasize that the framework can provide a complete description of the battlefield shown on the annotated map in Figure 13, using just a single, *Lego*-like element.

In practice, the analyst is more likely to be faced with a screen looking something like Figure 17, where it is possible to reason with both representations simultaneously. Notice that while different domain experts are likely to see very different things in the aerial view shown on the left of the screen, the satisfaction-diagram representation is expertise-domain neutral, and thus can play the desired role of a *lingua franca*. We would expect that the two representations are linked so that an action performed on one will change the other in the appropriate way.



analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes
analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes
analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes
analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes analyst notes

Figure 17. A possible screen display for the analyst’s reasoning tool.

Regardless of the actual interface chosen, which as always should be done to maximize ease-of-use and efficacy for the user, because it is built on the satisfaction diagram as the single conceptual element, the full power of situation theory can be drawn upon, together with the cumulative knowledge of thirty years of development and application of the theory.

Bibliography

1. Barwise, J. *The Situation in Logic*, CSLI Lecture Notes 17 (1989).
2. Barwise, J. and Etchemendy, J. *The Liar: An Essay on Truth and Circularity*, Oxford University Press (1987).
3. Barwise, J. and Perry, J. *Situations and Attitudes*, Bradford Books, MIT Press (1983).
4. Barwise, J. and Seligman, J. *Information Flow: The Logic of Distributed Systems*, Cambridge University Press (1997).
5. Devlin, K. *Logic and Information*, Cambridge University Press (1991).
6. Devlin, K. *Infosense: Turning Information into Knowledge*, W. H. Freeman (1999).
7. Devlin, K. “Modeling real reasoning”, in Giovanni Sommaruga (ed), *Formal Theories of Information*, Springer Verlag "Lecture Notes in Computer Science" (2009), pp.234-252.
8. Devlin, K. *Confronting context effects in intelligence analysis: How can mathematics help?* Unfinished preprint, 70 pages to date, http://www.stanford.edu/~kdevlin/papers.html

9. Devlin, K. and Rosenberg, D. *Language at Work: Analyzing Communication Breakdown in the Workplace to Inform Systems Design*, Stanford University: CSLI Publications and Cambridge University Press (1996).
10. Devlin, K. and Rosenberg, D. “Information in the study of human interaction”, in Johan van Benthem et al (eds), *Handbook of the Philosophy of Information*, North Holland (2008), pp.685-710.
11. Gawron, J.M. and Peters, S. *Anaphora and Quantification in Situation Semantics*, CSLI Publications (1990).
12. Grüninger, M. and Menzel, C. The Process Specification Language (PSL) Theory and Applications, *AI Magazine* Vol. 24 No. 3 (2003), pp.63-74.
13. Israel, D. and Perry, J., What is Information?, in *Information, Language and Cognition: Vancouver Studies in Cognitive Science, Vol. I*, University of British Columbia Press (1990).
14. Menzel, C. “The Objective Conception of Context and Its Logic, *Mind and Machines* Vol 9 Issue 1 (1999), pp.29–56.
15. Obrst, L. and Nichols, D. *Context and Ontologies: Contextual Indexing of Ontological Expressions*, The MITRE Corporation (2005).