Representing Complex Events Simply

by

Michael Lebowitz

Department of Computer Science

Columbia University[1]

New York, NY 10027

[1]Much of the research described here was carried out at Yale University, supported in part by the Advanced Research Projects Agency of the Department of Defense and monitored by the Office of Naval Research under contract N00014-75-C-1111.

Representing Complex Events Simply

by

Michael Lebowitz — Columbia University[1]

Department of Computer Science, 406 Mudd Building

New York, NY 10027

ABSTRACT

Complex events can often be treated as single units for purposes of cognitive processing. This paper presents a scheme for representing events that was used in the creation of the program IPP (the Integrated Partial Parser). This scheme, in effect, consists of rules for the creation of a set of primitive-like elements for a given domain. The specific structures needed to represent events in one domain, news stories about international terrorism, are presented.

1. Introduction

Human cognitive processing includes a remarkable ability to deal with complex concepts as single units. For example, we can think of physical entities such as automobiles, the United States or a personal computer as units, despite their complex nature. The same is true for actions such as writing a paper, baking a cake or hijacking a jetliner.

The ability to deal with complex concepts simply seems to encompass memory and language processing as well as general cognitive processing. Taking this fact into account is important in the development of powerful computer language understanding programs. It allows information to be stored at an appropriate level of detail. In particular, during the development of the Integrated Partial Parser (IPP) [4], a program designed to read, remember

---

[1]Much of the research described here was carried out at Yale University, supported in part by the Advanced Research Projects Agency of the Department of Defense and monitored by the Office of Naval Research under contract N00014-75-C-1111.

and make generalizations about large numbers of news stories, it was determined that a high level of representation for events should be primary. That is, in IPP stories are understood directly in terms of high-level structures and stored in memory in that same form.

The strategy of understanding directly in terms of high-level units required that a careful determination of the nature of the structures to be used for representation be made. The representation scheme derived will be the subject of this paper. I will also show why representation of the sort used in IPP is both useful and necessary, as well as point out the connection between this scheme and strictly primitive-based methods such as that of Schank [6]. Since the domain for IPP is news stories about international terrorism, most of the examples used here will be from that area.

The basic idea behind the representation scheme used for IPP is that a set of primitive-like structures are used to describe event in any particular domain of understanding. There are two types of structures used in IPP. These are Action Units and Simple MOPs (S-MOPs). The contention here is that while the specific Simple MOPs and Action Units used in a domain while be specific to that domain, the types are general. That is, a human understander will develop a set of Action Units and S-MOPs for each domain of interest.

Action Units (AUs) are used to describe concrete events, such as shootings, people being wounded, and hostages being released. They serve as modular units in the makeup of S-MOPs. AUs are basically packages of actions that can be remembered and talked about as distinct units.

Simple MOPs, or S-MOPs, describe more abstract situations, such as

extortions and attacks, that can be used to explain widely varying physical actions. An S-MOP describing extortion can, for example, capture the many similarities among kidnappings, hijackings, and building takeovers.

It is important for the success of cognitive processing that we are able to use appropriate levels of representation to describe events. If we represent an event such as a hijacking as a complex series of basic actions by the actors, we will have an unmanageable quantity of information to record in memory and to consider for potential generalizations and other inferences. On the other hand, if we think of this event as just another terrorist action, we will not have enough information available to recall the event appropriately or to process it properly.

It turns out that in IPP, these same representation units have an important function parsing. They act as the source of most of the top-down predictions that IPP uses in understanding. The details of such uses of S-MOPs and Action Units can be found in [4].

2. A Representation Scheme for Events — Overview

The representation scheme used in IPP makes use of Action Units to represent stereotypical events, and S-MOPs to capture stereotypical causal connections among S-MOPs. To see the rationale for using this level of representation, consider following news story.[2]

S1 - UPI, 6 May 80, Portugal

A teen-age gunman hijacked a Portuguese Boeing 727 airliner carrying

---

[2]All the stories used as examples in this paper are actual, unedited news of the sort processed by IPP

83 passengers and nine crewmen Tuesday, forcing it to fly to Madrid where he demanded a $10 million ransom payment, police said.

While there might be cases in which for story understanding purposes we might want to represent S1 as a collection of low-level actions — the gunman coming aboard the plane, telling the pilot where to fly, and so forth — normally this is not necessary. This story is best described as an instance of a larger stereotypical event, a hijacking.

Most readers already have a pre-packaged collection of knowledge about what happens during a hijacking — an abstract picture, of sorts — that can be used to understood this story. If we merely note that a story described a hijacking along with any unique details, then we can reconstruct any of the low-level actions later, if needed. This avoids recording the same details over and over for many different event.

In a real sense, recognizing that S1 is an instance of this pre-existing description of a hijacking, and remembering it as such an event, is what understanding such stories is all about.

Figure 1, below, presents this intuitive representation of the main events in S1. It is part of the representation that is built by IPP for this story, using just Action Units. The significant factor, for the moment, is the level of detail.

Roughly speaking, this representation says that there was a hijacking of a Portuguese airliner, and there was a ransom demand for $10 million. These two events correspond to the main actions that most readers would feel took place in this story. Each of these events can be treated as a unit in

```
$HIJACK
    ACTOR      = *gunman*
    PLANE      = *Portuguese 727*
    PASSENGERS = *83 passengers*
GS-GET-RANSOM
    ACTOR      = *gunman*
    AMOUNT     = *$10 million*
```

Figure 1: Higher-level representation of S1

understanding or memory, in the same fashion as Picture Producers.

The units of representation at this event-oriented level, such as $HIJACK and GS-GET-RANSOM in Figure 1, are Action Units.

Action Units describe clusters of events in memory organized around a principal goal, plan or action. AUs are collections of conceptual structures such as primitive ACTs [6], goals [8], political ACTs [9], and other Action Units. As shown abstractly in Figure 2, internally Action Units are packages of other elements. However, the main advantage in using this level of representation is that frequently all this detail can be ignored, in the same way that we treat Picture Producers as units, ignoring their internal structure. Within IPP's terrorism domain, Action Units are needed for events such as "shooting", "killing" and "kidnapping" as well as "hijacking" and "get ransom" that we saw before.

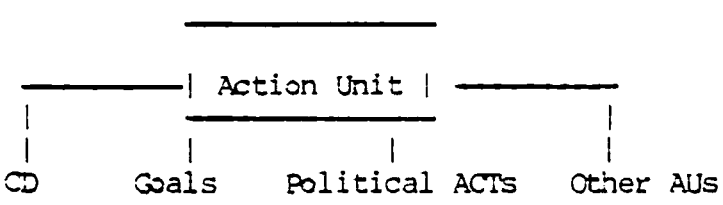

Figure 2: Action Unit schematic

Action Units are related to one another by the abstract stereotypical

situations in which they occur, represented with S-MOPs. The S-MOP S-EXTORT provides the final element of the representation of the hijacking story above. As shown in Figure 3, this representation describes an instance of the S-EXTORT, with the method a hijacking, the demand $10 million ransom, and an implicit threat (inferred from the S-MOP), that the passengers, in their role as hostage, might be killed.

```
S-EXTORT
    ACTOR     = *gunman*
    HOSTAGES  = *passengers*
    METHOD    = $HIJACK
                    ACTOR      = *gunman*
                    PLANE      = *Portuguese 727*
                    PASSENGERS = *passengers*
    DEMANDS   = GS-GET-RANSOM
                    ACTOR      = *gunman*
                    AMOUNT     = *$10 million*
    THREAT    = CAUSE-DEATH
                    ACTOR      = *gunman*
                    VICTIM     = *passengers*
```

Figure 3: Higher-level representation of S1

A representation of this sort is adequate for many understanding tasks. Figure 4 illustrates several different sorts of responses a person might give after having read S1. The first three questions are representative of the kind of question another interested person might ask, Q4 demonstrates the ability to recall the entire event, and Q5 how one story can remind a reader of other events.

The questions in Figure 4 are typical of those a reader of S1 would normally have to deal with. For most understanding tasks for stories of this sort, we seem only to make use of the basic events that occurred, taken as units, i.e., Action Units, and the fillers of the roles of the memory

Story Q/A - Q1: What did the hijackers demand?
A1: Ten million dollars.

Q2: What kind of plane was hijacked?
A2: A Portuguese 727.

Q3: What did the hijackers threaten?
A3: To kill the passengers.

Memory - Q4: What happened in Madrid today?
A4: A jet was hijacked for $10 million.

Reminding - Q5: What does this story remind you of?
A5: A hijacking of a 727 in Lisbon and an embassy takeover where they demanded a lot of money.

Figure 4: Responses from Figure 3

structures. The general information provided by the S-MOP that there was an extortion, and by the AUs that specific events took place, is enough to produce all of the responses in Figure 4.

3. Concrete Events — Action Units

The first area we must consider in a scheme for representing events is how the actual, concrete events that take place should be represented. In the same way that Schank [6] suggests that people and objects can often be represented as units (known as Picture Producers), we can often treat complex concrete events as packages. In this section I will describe the structures that serve this purpose in IPP, Action Units, by specifying their structure and scope, including a classification of AUs.

The basic definition of an Action Unit is stated in Figure 5.

It is important to notice in this definition that Action Units are ultimately based in primitives such as the Conceptual Dependency primitive

An Action Unit is a collection of primitive objects (CD ACTs, plans or goals, for example) or other Action Units, connected together by meaningful relations (causal, temporal and goal links being some examples) that can be treated as a representational unit.

Figure 5: Definition of an Action Unit

ACTs. Through AUs all the information that is available from a strictly primitive representation may still be accessed. However, the advantage of the Action Unit level of representation is to allow more efficient storage and processing of this information and, most importantly, the ability to ignore the detail when desired.

3.1. The structure of Action Units

When treated as units in understanding, Action Units have a relatively simple structure, somewhat similar to Conceptual Dependency's primitive ACTs. In this way their usage is also similar to what Schank and Abelson [8] refer to as "fleeting reference" in the case of scripts — only the presence of the whole structure is relevant.

Each Action Unit has a set of roles used to specify the various characters and props that take part in the event, as do the primitive ACTs. In each case, the roles are used to explain the actions of the various Picture Producers in a story.

A basic part of each Action Unit is also the inferences that can be made from its instantiation. These inferences indirectly define the semantics of the action that the unit is representing. Again, CD ACTs have similar sets of inferences associated with them. However, the inferences organized under an Action Unit are more appropriate to that level of representation. So rather than having the inference rule that passengers will be located at a city their

plane is hijacked to, we will have rules such as, the likely destination country for a hijacking is Cuba or Libya.

In their most common form of usage this is all there is to an Action Unit — a set of roles that specify the Picture Producers that take part in it, and a set of inferences that define the semantic meaning of the AU.

The similarities between Action Units and primitive ACTs. are illustrated in Figure 6.

Action Units		Conceptual Dependency	
Structure:			
$HIJACK	Roles: ACTOR	PTRANS	Roles: ACTOR
	PASSENGERS		OBJECT
	VEHICLE		RECIPIENT
	FROM		ORIGIN
	TO		INSTRUMENT
Inferences:			
VEHICLE slot probably filled by an airplane		ACTOR wants OBJECT to be at RECIPIENT	
TO slot may be Cuba, Libya ...		OBJECT is no longer at ORIGIN	
ACTOR will probably release hostages eventually		ACTOR may be ATRANSing OBJECT to RECIPIENT	
Encompasses domain.			
AU domains limited, interest based, e.g., terrorist attacks		CD can represent all physical actions	

Figure 6: Action Units vs Conceptual Dependency

Despite all the similarity in structure between Action Units and

primitive ACTs, one basic difference must be kept in mind. The CD primitive ACTs provide a canonical description of any event within their domain, which is the extremely broad one of physical actions. For all practical purposes, they cannot be further broken down.

Action Units, on the other hand, are domain-dependent and have an internal structure that can take one of several forms (to be discussed shortly). However, these structures need be accessed only rarely. If we can show that Action Units can be selected for a given domain in a principled fashion, as I will do in the next section, then they can perform the same representation function for that domain that primitive ACTs do for physical actions.

3.2. Delimiting Action Units

One of the great advantages of a primitive representation scheme is that it completely delimits what can or cannot be a unit of the representation. Conceptual Dependency representations of actions must be made up of the 11 primitive ACTs, interconnected causally in well-defined ways. This limitation provides a great deal of control, in that we package information such as inference rules only once — with the ACT — rather than for each situation that includes that ACT. This is guaranteed to be sufficient, since the representation of any event that includes the action described by the ACT must include the ACT — there is no other way to represent that action.

What I will show here is that Action Units can achieve the same advantage as domain-specific representation units. Within a domain there are only a limited number of Action Units that need be used to describe events.

Furthermore, all the allowable Action Units represent one of a small number of event types. Since they are strictly limited as to the form of events they can describe, Action Units can be reasonably identified from among all the complex events that occur.

The Action Units used by IPP can be broken down into four classes: scripts, general scripts, physical states, and goal states, to be described in detail shortly. Figure 7 summarizes the Action Units of of each class used by IPP in the terrorism domain. Notice that this rather small set of AUs is sufficient to represent the events in many hundreds of terrorism stories.

Scripts

$AMBUSH	$EXPLODE-BOMB	$HIJACK
$KIDNAP	$MASS-SHOOT	$SHOOT
$TAKEOVER		

General Scripts

G$-SIEGE	G$-NEGOTIATE	G$-ASSAULT
G$-ESCAPE-HOSTAGES	G$-SHOOTOUT	

Physical States

CAUSE-DAMAGE	CAUSE-DEATH	CAUSE-WOUND

Goal States

GS-CAPTURE-TERRORIST	GS-ESCAPE-TERRORIST	GS-GET-RANSOM
GS-RELEASE-HOSTAGES		

Figure 7: IPP Action Units

3.2.1. Scripts

The first kind of Action Unit describes a stereotypical, temporally sequenced set of events. These are simply scripts, as described in [8]. Examples of these in the terrorism domain include building takeovers, and shooting. The scripts are usually treated as indivisible units, not accessing

the internal sequence of events.

It is important to understand that Action Units represent only concrete events. So $HIJACK, for instance, represents a hijacker getting on a plane, holding a gun at someone's head and demanding the plane be taken somewhere. The more abstract elements of the event, such as it being a kind of extortion, are represented at other levels, to be discussed below.

Thus the events represented by a typical script AU will look something like Figure 8. It is by noticing stereotypical sequences such as this that we can identify scripts in the first place.

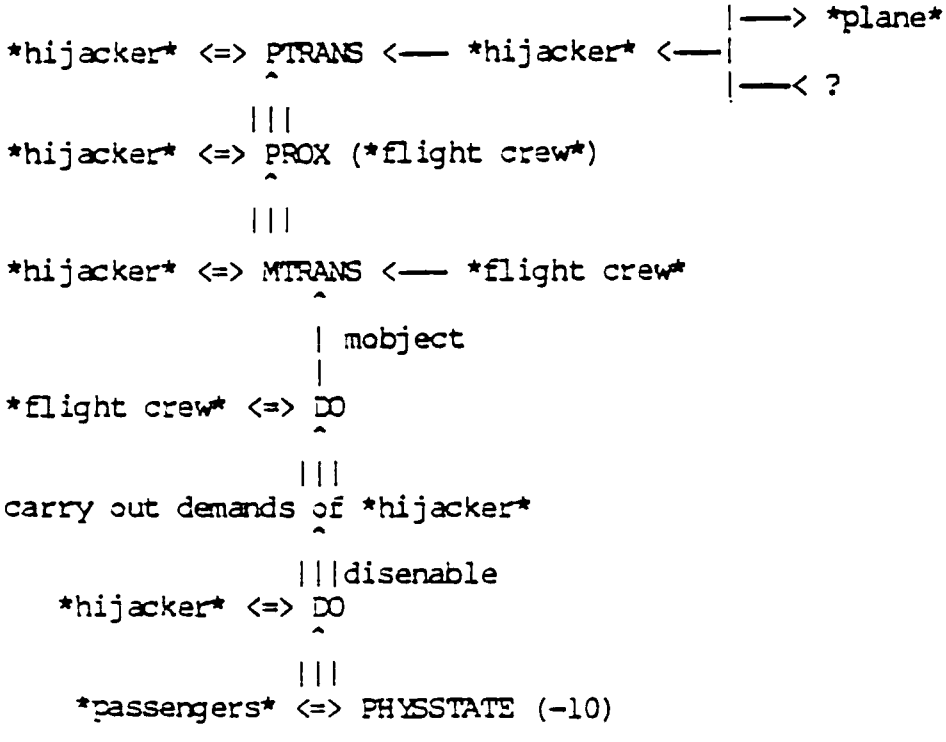

Figure 8: $HIJACK

Simple scripts are not adequate to describe all the stereotypical events we are familiar with. The next three categories of AUs illustrate this, indicating situations that are stereotypical, but fail to meet some of the

requirements of scripts.

3.2.2. General scripts

General scripts represent complex collections of events that are all in service of a single immediate goal. There are a number of events that we know occur as part of the AU, but they are not clearly sequenced or causally related. For example, the general script G$-NEGOTIATE (representing talks between terrorists and authorities) consists of many episodes of message passing, consulting with superiors, formulating replies, and so forth. These are all in service of the goal of reaching an agreement to terminate the terrorist action, but it is not necessarily clear how the pieces relate to each other.

It is only the attempt to achieve the goal that we are normally concerned with. For G$-NEGOTIATE it is important that both sides are trying to reach an agreement. With G$-SIEGE it is relevant that the authorities are trying to prevent the terrorists from getting away, but the details are rarely important. In such cases it is the goal we pay attention to.

The internal structure of a typical general script, G$-NEGOTIATE, is shown in Figure 9. It also illustrates that the basic rule for identifying general scripts is to find a number of actions that frequently occur in service of the same goal.

3.2.3. Physical states

Physical state AUs represent the final condition of people and objects after an event. They are small packages of CDs describing including a physical state and perhaps the immediate cause of the state.

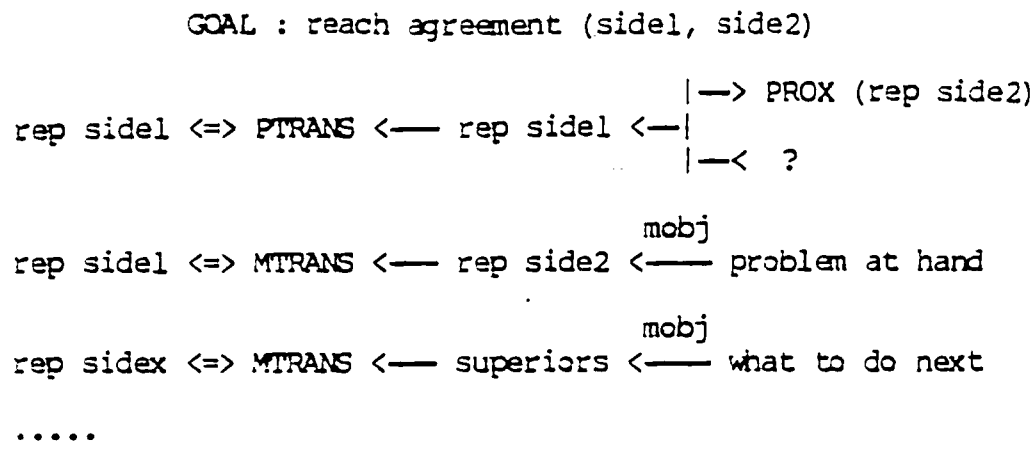

Figure 9: GS-NEGOTIATE

As an example, one of the most common physical states in the terrorism domain is CAUSE-DEATH, which is a package containing the simple CD causal shown in Figure 10. These structures are also easy to identify in a domain. We simply look for states that occur frequently with similar causes.

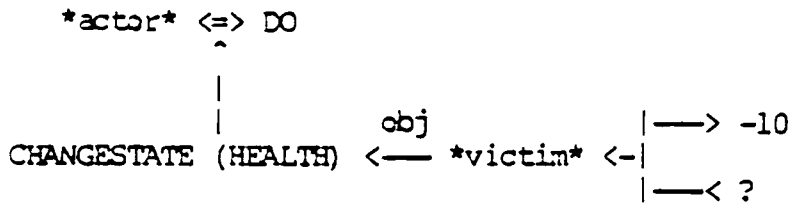

Figure 10: CAUSE-DEATH

3.2.4. Goal states

Goal states are similar to physical states, but instead of describing the condition of PPs, they describe the outcomes of goals the actors in a story may have. For instance, the payment of ransom indicates the success of a goal common in instances of extortion, and is represented by GS-GET-RANSOM. Similarly, the escape of a terrorist actor from the authorities, GS-ESCAPE-

TERRORIST, represents the success of the goal of staying free.

Goal states thus have the simple internal structure in Figure 11. They can be recognized by the frequent appearance of the same goal, with the same resolution.

Goal: *terrorist* DCONT *money*

Outcome: succeed

Figure 11: GS-GET-RANSOM

3.3. Identifying Action Units

In the above description of the classes of Action Units I mentioned rules for identifying members of each class. These rules provide a principled way of selecting Action Units to represent a domain for a computer model such as IPP, and they are presumably at the center of the way a person learns them in the first place.

The identification rule in all four cases involves looking at detailed representations of many events within the domain for frequently recurring patterns of events. These patterns then become Action Units. The four patterns we look for are listed in Figure 12.

- Sequence of actions in a specified sequence (SCRIPTS).
- Many events in service of the same goal (GENERAL SCRIPTS).
- States with the same immediate cause (PHYSICAL STATES).
- Goals with the same outcome (GOAL STATES).

Figure 12: Patterns of events that become Action Units

3.4. The role of Action Units in understanding

Action Units serve a number of purposes in the understanding process. They provide a representation for a large class of factual stories, including news stories. This level of representation is suitable for performing many of the tasks we normally associate with understanding, such as summary and question answering.

The second reason for using Action Units is to represent event in memory. Action Units are at a level that seems to correspond to the amount of information remembered for factual events. It is quite reasonable to want to use the same representation for understanding text as is used to represent the events described in that text in memory.

A third reason for using Action Units involves the organization of inference rules. While I will not go into the details here (see [4]), the Action Unit level of representation allows inference rules to be stored in an efficiently accessible, and yet not overly redundant fashion. This is largely due to the fact that many of our most useful inference rules are domain dependent, as are Action Units.

A final point about the utility of AUs in memory and parsing is that the different types of AUs can usually be treated quite similarly. Despite the fact that the four classes of AUs I have presented represent such different kinds of events, virtually all of their processing uses in IPP make no distinction as to the AU type.

This is indeed a major advantage of Action Units. By identifying four different types of conceptual events as instances of a single representation

structure, we can simplify processing, using the same processes for AUs of all classes. In effect, these units are virtually identical when "closed", and their differences only become apparent when they are "opened up".

The essence of this level of representation is that it allows quite different events to be processed and remembered without regard to the details of their structure.

4. Stereotypical Situations — S-MOPs

Action Units do not fully capture the meanings of the events described in news stories. There is also an important element of causality over and above the concrete events that take place. However, the causal connections, as well as the specific events, are often quite stereotypical, and can be represented in a compact form.

The structures used to represent the stereotypical causal relations among Action Units are known as Simple MOPs, or simply S-MOPs. S-MOPs capture similar causal relations among seemingly disparate actions. To see the kinds of relations involved, consider the story S2 about a hijacking in Portugal.

S2 - UPI, 6 May 80, Portugal

A teen-age gunman hijacked a Portuguese Boeing 727 airliner carrying 83 passengers and nine crewmen Tuesday, forcing it to fly to Madrid where he demanded a $10 million ransom payment, police said.

In S2 we know that the hijacking and ransom demand did not occur together coincidentally. We have a good idea of the causal relation between these events. While it might be possible to compute such relationships every time they are encountered, in this case, as well as many others, the relation is stereotypical enough that it seems more practical to have it pre-stored.

This causal relation can be extended to events other than just hijackings. Consider the following additional stories.

S3 - UPI, 27 February 80, Colombia

> Heavily armed leftist guerrillas shot their way into a diplomatic reception Wednesday seized the US ambassador and 44 other hostages and threatened to start "the painful task of executing them" unless the army pulled back from the diplomatic compound.
>
> The leftists who barged into the embassy residence in joggers' warmup suits with their weapons inside gym bags demanded the release of more than 300 political prisoners and a $50 million ransom.

S4 - Boston Globe,. 29 April 79, Mexico

> Two men and a woman broke into the home of one of Yucatan state's richest families, kidnapped their baby daughter and are holding her for ransom, police said Friday.
>
> A police spokesman said the kidnappers of 18-month-old Affife Greige Xacur left a note demanding between $200000 and $1 million in ransom.

In both of these stories, one an embassy takeover and the other a kidnapping, there is a ransom demand. In each case, the ransom request has the same relation to the instance of terrorism as in the hijacking story — it is the demand (or part of it) that must be acceded to in order to avert the terrorists' threat.

These stories are all examples of a concept more abstract than hijacking or kidnapping — extortion. While extortion does not consist of a single known sequence of events, we do know a great deal about it. For example, we know different ways it can be carried out — hijacking, or kidnapping for instance, and possible results — the release or escape of the hostages being two. In addition, we know how these elements normally relate to each other causally, so we need not recompute it for each instance. In the cases above, since we know how the method and threat of extortion relate to the demands

(and that ransom is a common demand for any for of extortion), we can apply this information to determine the relation in a specific case, such as a hijacking for ransom.

4.1. Using S-MOPs to determine causality

The major role of S-MOPs as representational units is to organize and explain Action Units. S-MOPs are concerned with topics such as extortion and attacks on individuals instead of hijackings and shootings. One of the important parts of understanding a collection of events is determining how they fit together causally (see [7], for example). For events as complicated as those IPP is concerned with, this can be an arbitrarily complex process. To avoid performing this computation every time it arises, it is possible to take advantage of standard relations among events, and pre-store these in memory.

However, there is a storage problem involved in keeping track of standard relations among events. Even though the number of Action Units is relatively modest, the number of pairwise combinations of these units can become quite large. This number grows as the square of the number of AUs, even if we ignore the possibility of multiple interconnections.

Figure 13 illustrates what happens when we try to connect even a small number of Action Units — in this case the methods for extortion — with a few of their results. We end up with quite a few connections. Also keep in mind that each of these causal connections can be relatively complex.

This method of storing standard explanations quickly becomes unmanageable. Furthermore it fails to capture significant generalities about the Action Units. As we saw from several examples of hijackings, kidnappings

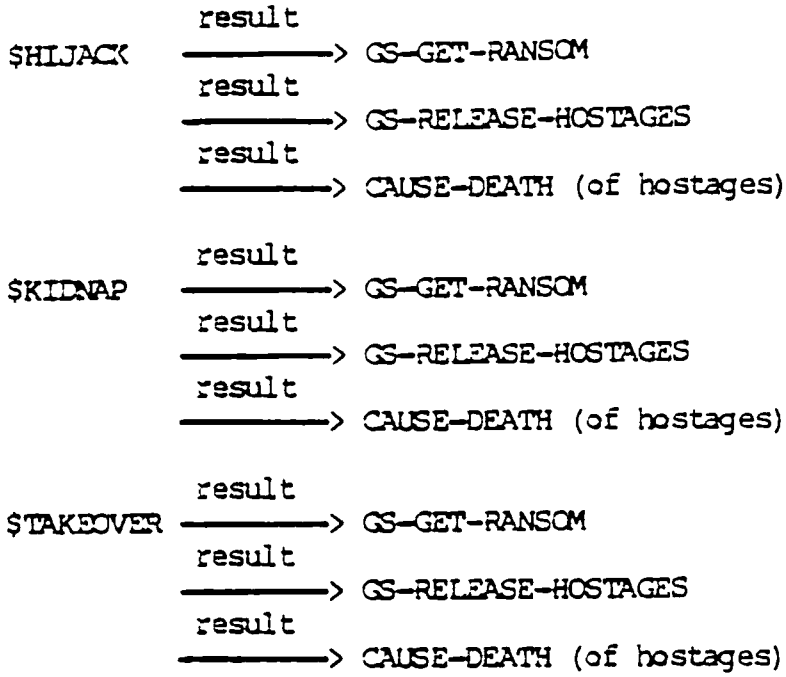

Figure 13: Pairwise explanations

and takeovers earlier in this chapter, the relation between any of the various methods for a concept such as extortion all have basically the same relation to a result like getting paid ransom. This generality should also be captured in our representation.

Representing such generalities is accomplished with S-MOPs. Action Units that commonly appear in service of an S-MOP have standard, known causal connections to that S-MOP. Then, if we desire to determine the relation between two such Action Units, we need only recognize how each relates to an appropriate S-MOP, and then use the standard relation of each to that S-MOP.

As an example, we know the following about extortion. Hijacking is an instance of extortion; asking for ransom is a demand of extortion; and giving in to the demand is the way to avoid harm to the hostages in an extortion. From this we can easily compute that giving a hijacker his ransom is the way to prevent the passengers of the plane (who are the hostages) from being hurt.

Arranging the same information seen in Figure 13 using an extortion S-MOP results in the organization in Figure 14.

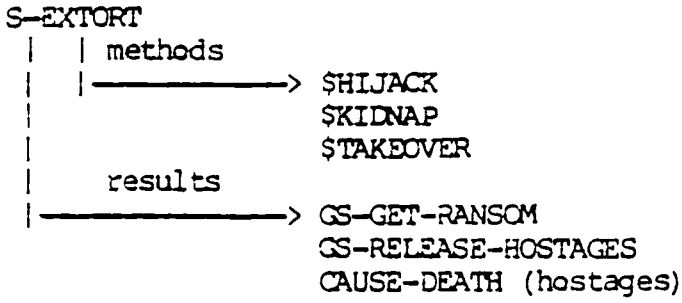

Figure 14: S-MOP explanation of AUs

The organization in Figure 14 is clearly a more efficient one than Figure 13. In effect we treat the recognition of the S-MOP as an explanation for instances the various component AUs. As we add new Action Units to an S-MOP, the size of the representation will only grow about linearly with the number of AUs.

An additional benefit of this representation is that by capturing the generality in situations such as extortion in a representation unit, we are able to incorporate new information more easily. So if we discover, for example, a new result for hijacking — the victims become emotionally scarred, perhaps — then we can assume immediately that it may also apply to kidnapping and takeover victims. In general, organizing information at its maximum level of generality provides that most efficient method of storage, and allows application to the widest range of situations.

In IPP, the same S-MOPs that are used to represent causal connections among events also serve a crucial role in memory organization, allowing events

to be stored in terms of their stereotypical representation. In this paper, however, I will concentrate on the role of S-MOPs in fully representing events at a high level.

However useful individual S-MOPs were, they would not be valuable in understanding if there were a large number of them relevant to each domain. However this is not the case. For a domain as large as international terrorism, there seem to be only a small number of abstract stereotypical situations, that need be represented by S-MOPs. IPP uses three — S-EXTORT, which I have discussed above, S-ATTACK-PERSON, for direct attacks on individuals, and S-DESTRUCTIVE-ATTACK, for indiscriminant attacks, such as most bombings.

The basic rule is that a group of Action Units can serve as interchangeable modules for only a small number of S-MOPs, all of which have somewhat related immediate goals. Each collection of Action Units and related S-MOP forms what we commonly think of as a domain.

4.2. S-MOP / Action Unit relations

For the same reasons that we do not want to allow arbitrarily constructed Action Units, a scheme of relating AUs to S-MOPs must limit the various ways the structures can relate. If we end up with as many different relations as we have AUs and S-MOPs we will not have gained very much. However, this is not the case. It turns out that the relations of Action Units to S-MOPs fall into three groups — methods, results and scenes.

Methods indicate how an S-MOP is carried out. Every instantiation of an S-MOP will include a method, usually explicitly mentioned, but occasionally

inferred. Typical methods for the S-MOP S-EXTORT are scripts such as $HIJACK and $KIDNAP.

Results describe the various outcomes of an S-MOP. For S-EXTORT some possible results are GS-RELEASE-HOSTAGES and GS-GET-RANSOM. It is the results that describe the final situation resulting from an S-MOP.

The final class of S-MOP related AUs is made up of scenes. These are events that often occur as some part of the causal understanding of the S-MOP, but are not crucial to understanding either the initiation of the S-MOP or its final outcome. In contrast with methods, it is quite possible for an event described by an S-MOP to take place without any scenes occurring. Again looking at S-EXTORT, some typical scenes are G$-NEGOTIATE and G$-SIEGE.

While the internal structure of an S-MOP contains the specific causal relations among all the potentially related S-MOPs, this broad classification provides an acceptable explanation of events in most cases. For instance, while a reader could, if asked, explain in detail the connection between negotiations and the release of hostages in an embassy take over, it is not necessary to make this explicit in most understanding situations. In addition, even this complex relation need only be stored once, in S-EXTORT, rather than for every different method of extortion.

An S-MOP also contains information about how the roles in the S-MOP relate to roles in various Action Units. So, for example, S-EXTORT indicates that the HOSTAGES in an extortion are the are the PASSENGERS of $HIJACK, but the VICTIM of $KIDNAP. This information is extremely valuable in making simple inferences about who did what during story understanding.

Figure 15 illustrates the different connections between an S-MOP and AUs by listing all the AUs associated with the S-EXTORT S-MOP, along with the relations among roles.

	S-EXTORT AU	Role in S-EXTORT	Role in AU
methods:	$KIDNAP	ACTOR	ACTOR
		HOSTAGES	VICTIM
	$HIJACK	ACTOR	ACTOR
		HOSTAGES	PASSENGERS
	$TAKEOVER	ACTOR	ACTOR
		HOSTAGES	HOSTAGES
results:	GS-GET-RANSOM	ACTOR	ACTOR
	GS-CAPTURE-TERRORIST	ACTOR	OBJECT
	GS-RELEASE-HOSTAGES	ACTOR	ACTOR
		HOSTAGES	OBJECT
	GS-ESCAPE-TERRORIST	ACTOR	ACTOR
	CAUSE-DEATH	ACTOR	ACTOR
		HOSTAGES	VICTIM
scenes:	GS-SHOOTOUT	ACTOR	ACTOR1
	GS-NEGOTIATE	ACTOR	OBJECT
	GS-SIEGE	ACTOR	OBJECT
	GS-ASSAULT	ACTOR	OBJECT
	GS-ESCAPE-HOSTAGES	HOSTAGES	ACTOR
		ACTOR	ESCAPE-FROM

Figure 15: S-EXTORT related AUs

5. Examples from Another Domain

So far in this paper, all the examples I have used have come from the domain that IPP was originally designed to handle — international terrorism. In order to illustrate the generality of the level of representation described here, I have selected examples from another domain, and will describe here how they would be represented using AUs and S-MOPs. The alternate domain I have selected includes stories about labor disputes — i.e., strikes.

The categorization of Action Units I have presented allows the selection

of relevant AUs for a new domain. In fact AUs from this classification cover the range of actions for this new domain as well as they do for terrorism.

Story S5 requires many of the AUs that recur throughout this domain.

S5 - UPI, 14 July 80, Massachusetts

Some 1,200 western Massachusetts carpenters, who returned to work Monday ending a 5-week strike, will vote this week on a new contract.

The carpenters had walked off their jobs at construction sites totaling about $400 million worth of work throughout the Connecticut River Valley.

Last week, a tentative agreement on the new pact was reached between representatives of Carpenters Union Local 32 of Springfield, Local 540 of Holyoke, Local 402 of Northampton and the Construction Industry Association of Western Massachusetts.

Union negotiators, who would not reveal the pact's contents, have urged members to accept it and were optimistic Monday the contract would be ratified.

Taking S5 as a example typical of this domain, we can look for Action Units using the criteria established in Section 3.2 for each category.

Scripts — To find scripts we look for stereotypical sequences of events. For this domain we find walking out of a job, marching in picket lines, and taking ratification votes.

General Scripts — These are larger collections of events in service of a single goal that have less internal structure, but can still be thought of as units. Here we have examples such as negotiations and campaigns for ratification.

Physical States — The labor dispute domain is not rich in physical states, as the physical state of the world does not change very much due to

strikes. However, there are a few such AUs, including perhaps the poverty of the strikers after a long strike and the destruction of work facilities that sometimes occurs during disputes.

Goal States — Since labor disputes are fundamentally conflicts between the goals of the strikers and the goals of the struck company, it is not surprising that this domain contains a number of goal states. In S5 we see examples such as getting a tentative agreement, the ratification of a contract, and the strikers returning to work.

From this analysis we see that it is possible to represent all of the events in this story with units of the same types that proved useful for terrorism.

To look at this sample domain a bit more, S6 is another story about a labor dispute.

S6 - New York Times, 11 July 80, New Hampshire

> Ironworkers at the Seabrook nuclear plant walked off the job today, joining other union members in a statewide job action that began last week.
>
> Contract negotiations, which began in May, broke off July 2 between the ironworkers union and six members of Associated General Contractors.
>
> ...
>
> He [a spokesman] said the ironworkers, who earn $13.42 an hour, were demanding a wage increase of $3.45 an hour.
>
> He said the six contractors offered an increase of 65 cents an hour and other concessions.

S6 repeats some of the same Action Units seen in S5, such as the walkout script, and adds several more. In particular we see more goal states — wage

increases, and "other concessions" — and general scripts like breaking off negotiations and making contract offers.

So from these two examples, as well as a number of others that were examined, it appears as if the labor dispute domain can be covered by a set of Action Units like those in Figure 16.

Scripts	
$WALKOUT	$PICKET
$RATIFICATION-VOTE	$SLOWDOWN
General Scripts	
G$-NEGOTIATE	G$-OFFER
G$-RATIFICATION-CAMPAIGN	G$-BREAK-OFF-NEGOTIATIONS
Physical States	
CAUSE-DAMAGE	REDUCE-INCOME (due to strike)
Goal States	
GS-RETURN-TO-WORK	GS-WAGE-CONCESSIONS
GS-BENEFITS-CONCESSIONS	GS-LOWER-PROFITS

Figure 16: Labor dispute Action Units

Being able to successfully represent the major events in a domain that is as far afield from terrorism as labor disputes indicates the validity of Action Units. Exactly the same rules for identifying Action Units that work for terrorism work in this new domain. This kind of test indicates that Action Units provide a useful level of domain-specific representation.

In addition to specifying the Action Units for this new domain, it is also necessary to look for the stereotypical patterns behind their occurrences — i.e., the relevant S-MOPs. There seems to be just one S-MOP — a job action S-MOP — that can be used to explain these Action Units in the same way S-EXTORT, S-ATTACK-PERSON and S-DESTRUCTIVE-ATTACK are used for terrorism

stories.

This job action S-MOP looks structurally very much like the ones used for terrorism. It has a set of roles — the sides involved, the demands of the strikers, and the final contract — as well as related AUs in all three relation classes. There are methods — such as a walkout — results — the workers either do or don't return to work, for instance — and scenes — such as negotiations. The total structure of S-JOB-ACTION is shown in Figure 17.

```
S-JOB-ACTION roles: ORGANIZATION  [government or company]
                    LABOR-GROUP   [union]
                    DEMANDS

                    NEW-CONTRACT

Methods
   $WALKOUT
   $SLOWDOWN

Scenes
   $PICKET
   $RATIFICATION-VOTE
   GS-NEGOTIATE
   GS-OFFER
   GS-RATIFICATION-CAMPAIGN
   GS-BREAK-OFF-NEGOTIATIONS

Results
   CAUSE-DAMAGE
   REDUCE-INCOME (due to strike)
   GS-RETURN-TO-WORK
   GS-TENTATIVE-AGREEMENT
   GS-WAGE-CONCESSIONS
   GS-BENEFITS-CONCESSIONS
   GS-LOWER-PROFITS
```

Figure 17: S-JOB-ACTION S-MOP

I will conclude my look at the labor dispute domain by simply presenting the final representation for one of the stories I used above. Recall that S5 involved a number of the aspects of a typical labor dispute. An appropriate representation for S5 is shown in Figure 18.

```
S-JOB-ACTION
        EMPLOYER = construction industry of W. Mass
        UNION    = carpenters
 Methods
   $WALKOUT  ACTOR = carpenters

 Scenes
   G$-NEGOTIATE   SIDE1 = construction industry
                  SIDE2 = carpenters
   G$-RATIFICATION-CAMPAIGN  ACTOR = union negotiators

 Results
   GS-RETURN-TO-WORK  ACTOR = carpenters
                      PLACE = work sites
   GS-TENTATIVE-AGREEMENT   SIDE1    = construction industry
                            SIDE2    = carpenters
                            CONTENTS = ?
```

Figure 18: Representation of S5

6. Psychological Validity

In this paper, I have proposed a representation scheme that matches some intuitive ideas as to how events should be represented at a high-level. In [4], this scheme was shown to have considerable processing advantages. However, as this is intended to be a psychological model, it would also be useful to have further evidence that this is actually the way people remember events.

In an attempt to acquire such evidence informally, I asked a number of people to summarize a well-known news event from the past, similar in nature to those that I have been studying, that I expected most of them to be familiar with. In particular, I asked people to recall what terrorist actions took place at the Munich Olympics in 1972. Three typical responses are shown below (emphasis added).

1 — A half-dozen PLO gunmen, affiliated I think with the Black September faction of the PLO, invaded the dormitories occupied by Israeli athletes and took some (all?) of the athletes hostage. When German police stormed the dormitories, the Israeli athletes, who were being held hostage, were killed. Israeli recriminations included the question whether or not the Germans would have stormed the dormitories, if the hostages had been non-Jewish.

2 — A group of terrorists (I don't know the political sympathies) occupied the Olympics dormitories and took hostages. They held the hostages for a short time and eventually demanded to go the airport. At the airport they were attacked by police (they had gotten as far as into a helicopter) and some of the players and perhaps all of the terrorists were killed.

3 — Palestinian terrorists took the Israeli compound in the Olympic Village, and held it all day while Jim McKay provided commentary. The terrorists demanded a plane out of Germany. When they arrived at the airport with their hostages, there was a shootout (begun, I assume, by the West German government), that ended in the death of all the hostages and terrorists.

Figure 19: 1972 Olympic protocols

Protocols of this sort seem most useful in determining the detail in the information that people remember about an event. We certainly cannot expect such protocols to match directly with the subjects' internal memory representations. However, they clearly must have available all the information they provide, and in all likelihood, they do not have much more specific memories, or they would provide some of it during recall.

The major events mentioned in the summaries are in boldface in Figure 19. These descriptions correspond very well with the set of Action Units I have presented for terrorism in this chapter. The four kinds of Action Units presented cover the range of actions that people use to describe events. This can be seen if we compare the above protocols with a possible representation of the actual events in Munich in 1972, shown in Figure 20.

```
S-EXTORT ACTOR     terrorists, org Black September
         HOSTAGES  Israeli athletes
         PLACE     Olympic village, Munich
         DEMANDS   Release of political prisoners in Israel

         THREAT
           CAUSE-DEATH OBJECT Israeli athletes

         METHODS
           $TAKEOVER    ACTOR     Terrorists
                        HOSTAGES  Israeli athletes
                        TARGET    Israeli dormitory

         SCENES
           G$-NEGOTIATE ACTOR1    German police
                        ACTOR2    Terrorists
                        PLACE     Airport

           G$-SIEGE     ACTOR     German police
                        PLACE     Israeli dormitory

           G$-SHOOTOUT  ACTOR1    German police
                        ACTOR2    Terrorists
                        PLACE     Airport

         RESULTS
           CAUSE-DEATH          OBJECT Israeli athletes

           CAUSE-DEATH          OBJECT terrorists (some)

           GS-CAPTURE-ACTOR     OBJECT terrorists (some)
```

Figure 20: Munich 1972 in IPP terms

Most of the events that people described in their protocols appear in this representation, at about the save level of description that people used. This gives credence to the idea that these people remembered just about as must detail as is captured by an Action Unit representation.

Figure 21 lists a number of Action Units that appear in the protocols, along with the various words used to describe them.

Obviously the fact people describe events in Action Unit sized chunks

EVENT: $TAKEOVER
WORDS: attacked, held captive, took ... hostage, invaded
TYPE: Script

EVENT: $ASSAULT or GS-SHOOTOUT (there was some confusion among subjects)
WORDS: ambushed, attacked, raid, stormed
TYPE: General script

EVENT: CAUSE-DEATH (of hostages or terrorists)
WORDS: killed, death
TYPE: Physical state

EVENT: demands role of S-EXTORT
WORDS: asked for, demands, demanded
TYPE: Goal state

EVENT: GS-ESCAPE-TERRORIST
WORDS: escaped, get away
TYPE: Goal state

EVENT: GS-NEGOTIATE
WORDS: negotiated, settlement
TYPE: General script

EVENT: $SHOOT
WORDS: shot
TYPE: Script

EVENT: GS-SHOOTOUT
WORDS: shootout, battle
TYPE: General script

Figure 21: Memory units illustrated in Olympic protocols

does not prove that that is how the events are actually represented. However, these protocols, along with the success of IPP, do suggest that we are on the right track. Further verification would require much more highly controlled experimentation.

7. Comparison with Other Systems of Representation

The suggestion made in this chapter that understanding involves several different types of representation is not a unique one. [8] discuss the need for scripts, including scripts that may not be opened (also see [3]), goals

(also [10]) and themes. An entirely new system of primitives was developed for use in describing political and other situations involving authority ([9, 1]). All of these forms of representation, including Action Units and S-MOPs are instances of frames, as introduced by Minsky [5, 2].

However there is an important differences between these sorts of systems of primitives and the Action Units and S-MOPs described here. This involves the contrast between levels of description and domain of description.

Scripts, goals and themes are all necessary for understanding actions, but are all different in nature and describe different aspects of the world. Similarly, Schank and Carbonell's political primitives describe authority relationships that cannot be represented in CD.

Action Units, on the other hand, are units that may include information at multiple levels, i.e., CDs, scripts and goals, but represent a cluster of ideas that are functionally related and can often be treated as a single item. They are somewhat idiosyncratic and highly dependent upon the domain being considered. The recognition of a domain results in events being understood in terms of the memory units that have been developed over time for that domain.

So Action Units, rather than describing events at a single level, include knowledge at several different levels combined into packages that can be treated as single units. This allows access to CD, plan and goal level information when needed, but enables all of that to be avoided when it is not required for understanding.

8. Conclusion

In this paper I have shown that events normally need not be represented in terms of the lowest level actions that make them up. I have presented a type of representation unit known as an Action Unit that describes larger collections of actions. These units must have one of four specific internal structures, which allows them to be identified for a given domain. I have completed the description of another type of unit, S-MOPs, that describe more abstract situations, and can be used to explain Action Units.

Importantly, I am not saying here that the idea of a primitive representation is not a valid one. In fact a language independent conceptual representation is the basis of an Action Unit representation. Rather this idea is an extension of the idea of a primitive, and indicates that people develop packages of primitives for various domains of interest and levels of understanding. These constructs retain all the power of primitive decomposition, and yet provide considerable improvements in terms of memory and processing efficiency.

REFERENCES

1. Carbonell, J. G. Jr. Subjective understanding: Computer models of belief systems. Tech. Rept. 150, Yale University Department of Computer Science, 1979.

2. Charniak, E. On the use of framed knowledge in language comprehension. Tech. Rept. 137, Yale University Department of Computer Science, 1978.

3. Cullingford, R. Script application: computer understanding of newspaper stories. Tech. Rept. 116, Yale University Department of Computer Science, 1978.

4. Lebowitz, M. Generalization and memory in an integrated understanding system. Tech. Rept. 186, Yale University Department of Computer Science, 1980. PhD Thesis

5. Minsky, M. A framework for representing knowledge. In P. H. Winston, Ed., The Psychology of Computer Vision, McGraw-Hill, New York, 1975.

6. Schank, R. C. "Conceptual Dependency: A theory of natural language understanding." Cognitive Psychology 3, 4 (1972), 532 - 631.

7. Schank, R. C. The structure of episodes in memory. In D. Bobrow and A. Collins, Ed., Representation and Understanding: Studies in Cognitive Science, Academic Press, New York, 1975.

8. Schank, R. C. and Abelson, R. P.. Scripts, Plans, Goals and Understanding. Lawrence Erlbaum Associates, Hillsdale, New Jersey, 1977.

9. Schank, R. C. and Carbonell, J. C. Re: The Gettysburg Address: Representing social and political acts. In Findler, N., Ed., Associative Networks: Representation and Use of Knowledge by Computers, Academic Press, New York, 1979.

10. Wilensky, R. Understanding goal-based stories. Tech. Rept. 140, Yale University Department of Computer Science, 1978.