Modeling Current Events

Christopher Landauer, Kirstie L. Bellman
Aerospace Integration Science Center
The Aerospace Corporation, Mail Stop M6/214
P. O. Box 92957, Los Angeles, California 90009-2957, USA
cal@aero.org, +1 (310) 336-1361, bellman@aero.org, +1 (310) 336-2191

Abstract

Enormous amounts of information are produced every day, all over the world, but very little of it is true. In this paper, we describe the modeling component of a current events analysis system that combines statistical and Knowledge-Based approaches to natural language analysis with new modeling and architecting techniques to produce a Knowledge-Structured Virtual Subject World.

This Multi-User Virtual Environment (MUVE) allows groups of subject- or area- experts to collect and analyze relevant information, for visualization, exploration, consistency checking, and discovery. It supports the analyst's expertise and experience by emphasizing relevant information and filtering out irrelevant information. The system makes essential use of Computational Reflection, which is the ability of a system to analyze and adjust its own behavior. We briefly describe the architecture of the system, show how reflection leads to flexibility of operation, and concentrate on the modeling techniques that we use to collect, retain, and examine information.

Keywords: Conceptual Categories, Empirical Natural Language Analysis, Integration Infrastructure, Modeling Current Events, Virtual Worlds

1. Introduction

Modeling current events has always been a losing proposition. There is never enough time to locate and correlate all relevant information, since completeness requires collecting enormous amounts of information only subsequently found to be irrelevant, and yet anything that is missed leaves the potential for great embarrassment, loss, or even tragedy.

The open source opportunity is the enormous amount of material that is broadcast, published, or otherwise made public, around the world, every day. The problem is that the multiplicity of languages, formats, and the sheer volume of such data precludes any comprehensive coverage by any means, and we are reduced to sampling a tiny fraction of what is available.

In this paper, we describe some techniques that can be applied to this problem. Leaving aside the difficult problem of language translation, and the easy but tediously time-consuming problem of format conversions, we show that certain old and new computational methods can be used to make the information extraction problem more tractable, and extend the coverage to include many sources in a relatively cost-effective manner.

As part of a research project aimed at current events analysis, we describe a system that uses open source information to maintain a Knowledge-Structured Virtual Subject World (not just for visualization, but also for travel and manipulation, exploration and discovery), based on combining the advantages of the Web with those of a Multi-User Virtual Environment (MUVE). In this environment, systematic analyses run continuously, examining correlations and building hypotheses that can be studied and either discarded of deferred (if they are not sufficiently strong) or turned into new information structures (if they can be verified or otherwise sufficiently supported). In this environment, we can not only construct and provide information, but it can be assessed by subject-area experts, and content certification can be organized (when was the last time you believed something you read on the World-Wide Web?).

The particular example we use in this paper is a current events analysis system for crisis detection and management. The users of the system have to examine models of situations, identify potential problems and collect more information about them, and sound the alarm if dangerous events are imminent.

1.1. Focus

The context of this work is the study of the impact of new technologies, and the modeling required to make such studies more reliable and credible. We primarily consider support and integration technologies for crisis prepara-
tions and operations in crisis intervention scenarios. Other choices could be made, but this one is in an area that well illustrates our techniques.

The activities that we have to consider as designers of such a system occur on four different time scales:

1. long-term (years to months): historical situation;
2. near-term (months to days): political and economic situation (and for post-crisis assessment);
3. short-term (days to hours): crisis situation (for planning); and
4. immediate-term (hours to seconds): crisis situation (during execution).

We will generally ignore the last of these for the purposes of this paper, but it does become important in understanding and designing an appropriate crisis intervention system.

We assume the existence of a situation management team, which is an organization with responsibility for rapid response and situation management in crisis situations. We describe a rapid response system that supports all of the activities of this organization, so it is really a readiness, detection, and response system. We describe the four parts of the rapid response system (which correspond to four parts of the situation management team, and also to the four time scales above):

1. a surveillance entity, which has the responsibility for “watching the world”, and identifying potential crises;
2. a tracking entity, which has the responsibility for tracking potential crises, and estimating their severity and likelihood;
3. a planning and decision entity, which has the responsibility for determining a response to a crisis, and managing the situation during the crisis; and
4. a response entity, which has the responsibility for executing the response.

As above, we ignore the response entity to save space. This system includes many kinds of collaboration, modeling, analysis, and prediction tools, and requires different kinds of infrastructure support, so it makes a good test case for many kinds of new technologies. It is a Constructed Complex System, which is a distributed, heterogeneous system managed or operated by computing systems [12].

1.2. Modeling Approach

We use modeling as a cost-effective way of studying complex phenomena. Of course, that means that we believe that our modeling methods are relatively inexpensive to use, in addition to providing much more analytically justified conclusions about complex phenomena. We have shown, in earlier work on conceptual design of space systems [1] [2] [19], that very powerful and flexible modeling systems need not be overly expensive, by making the infrastructure that underlies them flexible enough to allow many different kinds of modeling methods and tools to work together.

The level of complexity of such a system requires careful attention to the scope and scale of our modeling projects: we cannot use just one model, into which we try to fit all of the important considerations (we believe that this is not possible anyway [6]).

Instead, we need to use a collection of interacting models, at different levels of abstraction and even different meta-levels, with analytical tools to connect their effects. At this early stage in the descriptions, we will not try to explain all of the abstractions that go into these models; we save that for discussion of the specific models we have built. We merely note that there will be several different modeling styles used, which means that we will need some strong model integration techniques, as we describe later on.

New technologies often accomplish a purpose in a very different way from what is expected of the older technology solutions. Appropriate abstraction is required to allow any assessment system to back out of unnecessary design decisions, which involves replacing proposed solutions with their underlying problems, context and applicability criteria. To this end, our descriptive models include networks of cooperating and conflicting purposes, and methods for analyzing them. Remember, we are trying to model, among other things, the processes of technology evaluation and integration, and not just the technologies themselves.

This replacement of solutions with the problems they address is a cornerstone of our modeling process, since it allows us to identify implicit assumptions in the current solutions, and determine whether or not they are necessary constraints. Much of the modeling work is concerned with problem decomposition, and identifying resources that can be applied or specifying resources that need to be constructed.

This replacement is an application of our Problem Posing Interpretation of programming and modeling languages [10]: ALL modeling and programming languages can be interpreted as defining structured problems, to which available resources may be applied. This change of interpretation gives us tremendous leeway on how we attach applicable resources to solve those problems [9].

This project also requires us to change the nature of our models, to include not just the things, i.e., the systems, components, actions, and events, but also the purposes, success and failure criteria, organizational interactions, and information pathways. For a Constructed Complex System that is inserted into a complex environment, the organizational
issues are often as important as the technical ones, and many otherwise viable technologies are precluded or discarded by organizational failures and mismatches.

We must also model the lifecycles of other relevant processes: technology system development, readiness, and situation detection, identification, response, and intervention, since that information will help a rapid response system to predict its own capabilities and when they will become available, which is useful for long-term strategic planning.

In general, we use the term “situation” to refer to partially elaborated problem statements, with some historical background, and the term “scenario” to refer to partially elaborated combinations of problem statements, event descriptions, contingencies, and partial solutions, generally organized into a narrative style.

We will primarily use an entity action perspective, describing the entities and relationships that define static structure (what is a part of what, for components and organizations, what is contained in what, for places and people, etc.), and the events and activities that define dynamic processes (what events come before what others, which activities may overlap, etc.). These descriptions allow different kinds of analyses.

1.3. Theoretical Overview

In this subsection, we give a short overview of our own theoretical work, which we will describe as it pertains to this problem. We list many references, since the technical results do not seem to be widely known. In the next Section, we describe these applicable technologies in more detail. Here we only introduce them and show how they arose.

Our organization AISC, by its very name (Aerospace Integration Science Center), is attempting to invent what we want to call “Integration Science” [4] [5], since we believe that there should be one (and that there is not one yet). There are many organizations that are excellent at integration processes, and many skilled and even talented individuals, who succeed in integration because of what we have called “heroic engineering”.

We originally started this work as we developed a design support system for conceptual design of space systems, called VEHICLES” [1] [2] [19]. It combined hundreds of models and model analysis tools in an environment that was flexible enough to allow a designer to start anywhere in a design: from a new technology to a particular space mission. It made no assumptions about how the design process was supposed to work, and did not try to enforce consistency until the designer requested it. That flexibility alone allowed designers to explore complex design spaces in a way that was not previously available to them, searching for design drivers or regions of interest, and often led to radical changes in designs.

Out of that grew our main system engineering work, a Knowledge-Based Integration Infrastructure called “Wrapping” [7] [9] [12], for which we’ve written several dozen papers and several invited book chapters over the last twelve years. Its relevance to this application area is that large scale, heterogeneous and distributed system integration is not as hard as one might think [14].

We also began to study the use of Virtual Worlds for integration of humans and computing systems [8] [3] [11] [16], since they provide a very simple common semantics, and a conveniently unrestrictive interaction protocol. These studies arose from our work on Multi-User Virtual Environments (MUVEs) and other collaboration support tools [20] [13], which also led to our founding and chairing the annual Virtual Worlds and Simulation Conference (the URL for the conference is http: //www.cs.umd.edu /<tilde>cal /VWsim /VWsim-schedule. html).

Our work in VEHICLES and then Wrappings provides a new approach to software and system engineering [22], and new architectural patterns based on explicit meta-knowledge [12]. We believe that these complex infrastructures will be necessary, since the Constructed Complex Systems we are building and designing already have much too much functionality to allow direct human control. Therefore, a sufficiently complex system will need to be much more autonomous than they are now, and we have shown how our Wrapping approach to infrastructure leads to a new architecture for autonomous systems [12].

These systems will also need to have a great deal of flexibility in their modeling mechanisms, so that the notations and languages used do not preclude some design decisions [23]. We have developed a Wrapping Expression notation called wrex [9], that allows a flexible kind of generic programming, relying on a Wrapping infrastructure to provide the resources.

Similarly, Constructed Complex Systems will also need to have a much greater amount of their own structure available for examination, so that it can be studied, understood, and modified. To this end, Constructed Complex Systems will need a strong form of Computational Reflection [21] [9], which allows access to system components under the management of the system itself.

Finally, as with any large system engineering process, developers will need to be able to study these systems far in advance of completion, and we have shown that Wrappings facilitate Verification and Validation studies [15].

2. Technology Description

We have developed several methods that apply to this problem. The knowledge representation structures are of two kinds: those that hold conceptual information, using
a data structure that we call Conceptual Categories [17] [18], and those that hold more concrete information or active models, which are all constructed in Virtual Worlds [8] [16]. The infrastructure that holds it all together is provided by Wrappings [7] [9], our Knowledge-Based, Computationally Reflective, active integration infrastructure. We describe each of these technologies in this Section.

2.1. Virtual Worlds

Virtual Worlds (VWs) are computer-mediated environments in which humans and computational resources share the same access and interfaces [4]. They allow a much richer set of interaction styles for users and their tools than the usual kinds of user interface, since they use an underlying metaphor of spatial organization that supports shared presence [20]. Even when the metaphor is entirely expressed in text, the sense of place and presence can be very strong [16]. These VWs implement a notion of shared virtual space that we create, in which we interact with each other and with our computing tools, together, instead of having all tool use and interaction mediated through tools used individually and separately [8] [3].

Virtual Worlds afford a unique opportunity to integrate humans and tools in the same (simulated) world, and to make explicitly visible all the information and relationships, assumptions and confusions, which currently are only implicitly known (if at all). All of the people who design and analyze current events models in a particular application area in an organization can be interacting inside this environment, that is, the VW contains a model of a world in which the human users can move around, and yet with each other for discussions or even brainstorming, exercise any or all of their ordinary computer tools, perform speculative scenario simulations together or separately, and generally access all the available information in its own appropriate context. In fact, this attachment of information to “virtual places” supports one of the most important analytical processes that humans perform: that of noticing a relationship that was not previously apparent.

2.2. Conceptual Categories

Conceptual categories model our notion of categories of concepts [17] [18]. They generalize sets in four directions: our collective objects have (1) indefinite boundaries; (2) indefinite elements; they allow (3) leakage from context; and the elements have (4) multiplicity of structure.

The first of these extensions is familiar from fuzzy sets and other models of uncertainty. The boundaries of an ordinary set are rigid, which does not reflect what we know about human categorization. The second extension is desired because we frequently need to consider entities that do not have a completely known structure, and yet we do know something about them. The context is allowed to leak into the interpretation of the objects because we frequently change the nature of the elements under consideration simply by changing the context of analysis. Finally, there is a notion of multiplicity of structures that correspond respectively to considering the same object or class from different points of view, in different contexts. Since there is NO one model that suffices for modeling complex phenomena [6], this aspect allows us to use different models of the same phenomena for different purposes.

This notion of viewpoint allows us to model the modeling decisions explicitly, and to keep track of the modeling simplifications so we can relate them to each other and to the processes that create, change and use them.

2.3. Wrappings

Wrappings are our approach to the development, integration, and management of heterogeneous computing systems [4] [12], based on two kinds of software entities: Wrapping Knowledge Bases (WKBs) and Problem Managers (PMs). The WKBs contain explicit, machine-processable, qualitative information (called Wrappings) about the system components, architecture, and all the computational or information processing elements (called resources) in the system: not just how to use them, but also whether and when and why and in what kinds of combinations they should or can be used. The PMs are algorithms that use the Wrapping descriptions to determine which resources to use and how to combine them to apply to problems.

The advantages of our knowledge-based integration technology are (1) a simplifying uniformity of description, using the meta-knowledge organized into Wrapping Knowledge Bases, and (2) a corresponding simplifying uniformity of processing that meta-knowledge using algorithms called Problem Managers, which are active integration processes that use the meta-knowledge to organize the system’s computational resources in response to problems posed to it by users (who can be either computing systems or humans). We have shown its wide applicability in software and system development [7] (and references therein). In particular, since the entire process is recursive, Wrappings provide a general way to allow specialized methods to participate.

The Wrapping theory has four basic features:

1. ALL parts of a system architecture are resources, including programs, data, user interfaces, architecture and interconnection models, and everything else;
2. ALL activities in the system are problem study, (i.e., all activities apply a resource to a posed problem), including user interactions, information requests and announcements within the system, service or processing.
3. Wrapping Knowledge Bases contain Wrappings, which are explicit machine-processable descriptions of all of the resources and how they can be applied to problems to support what we have called the Intelligent User Support (IUS) functions:

(a) Selection (which resources to apply to a problem),
(b) Assembly (how to let them work together),
(c) Integration (when and why they should work together),
(d) Adaptation (how to adjust them to work on the problem), and
(e) Explanation (why certain resources were or will be used);

4. Problem Managers (PMs), including the Study Managers (SMs) and the Coordination Manager (CM), are algorithms that use the Wrapping descriptions to collect and select resources to apply to problems. The PMs are also resources, and they are also Wrapped.

The Wrapping information and processes form expert interfaces to all of the different ways to use the resources in a heterogeneous system that are known to the system. The most important conceptual simplifications that the Wrapping approach brings to integration are the uniformities of the first two features: the uniformity of treating everything in the system as resources, and the uniformity of treating everything that happens in the system as problem study. The most important algorithmic simplification is the reflection provided by treating the PMs as resources themselves: we explicitly make the entire system reflective by considering these programs that process the Wrappings to be resources also, and Wrapping them, so that all of our integration support processes apply to themselves, too. It is this ability of the system to analyze and modify its own behavior that provides the power and flexibility of resource use.

3. Problem Context

In this Section, we describe the details of the problem context in which our current events modeling is to occur. We assume an organization responsible for rapid response and situation management of a crisis somewhere in the world. The situation management team, using the rapid response system that we describe, has the responsibility to identify, assess, and respond to such crises. Because there can be many different locations, causes, risks, and kinds of crisis, there will be different treaties, laws, charters, and other constraints that apply, and therefore many different organizations that are involved.

We therefore assume that the situation management team and the rapid response system support temporary assignments of individual experts or other representatives from different organizations and possibly different countries. The team must create a virtual organization for each crisis, whose task is to manage the team’s response to that crisis. In addition, since the situation management team has some long-term responsibilities also, there will be some permanent staff and facilities, and some of the temporary assignments can be fairly long-term.

The situation management team is described in terms of its four parts: the surveillance entity, the tracking entity, the planning and decision entity, and the response entity (but we are generally ignoring the response entity in this paper).

The surveillance entity has the responsibility for “watching the world”, and identifying potential crises. This analysis is currently performed by many organizations, for many different purposes, and this part of the situation management team is primarily liaison to those organizations, and modelers of the information that is obtained from them. It is permanently a part of the team.

The tracking entity has the responsibility for tracking potential crises, having been queued by the surveillance entity, and estimating their severity and likelihood. There will also be liaison elements in this entity, and the models that support a large number of contingency planning scenarios. It is permanently a part of the team.

The planning and decision entity has the responsibility for determining a response to an identified crisis, and managing the situation during the crisis. This entity is created anew for each crisis, from domain experts, current events experts, and operations experts. It uses models in many parts of its work: to make predictions of reactions to its possible responses, to study alternative responses, to support rehearsals, to monitor execution, and to explore possible longer-term repercussions from the response. It is always a part of the team, but it has a lifetime only through after action assessment, and the corresponding adjustment in the models. Some planning and decision entities are set up but not activated, since they are expected to be needed, if at all, with a very short time frame. This is one form of readiness. We expect that strategic planners in the situation management team or elsewhere will decide which of these entities are defined but not invoked, by deciding what problems are of sufficient importance and time criticality.

3.1. Time Scales for Activities

One of the interesting things about this problem is that there are activities that occur on many time scales simultaneously, and they interact in ways that mean we must consider all of them together.

There are ongoing activities: information gathering for
history and background, event detection, and model building.

Information must be gathered about the history and background of many people, organizations, and other entities. We expect enormous amounts of open source material to be available and informative, and of course, there are also other sources. These activities include political situation assessment, in which we describe stakeholders and their interests, divided up in two different ways: us, them, or others; people, organizations, or governments; and economic situation assessment, in which we describe winners and losers, trends and forecasts.

Event detection is about noticing coincidences of people or places of interest, or interesting changes of situation (of course, interesting here means interesting to some analyst). Some of these events are indications and warnings of important activities, others are trigger events that are specifically asked for by other entities.

Model building is about constructing world models for hypothesis examination and analysis, and simple scenario generation.

These are long-term activities, with a scale of years to months. These are the responsibility of the surveillance entity, which passes identified events on to the tracking entity.

Then there are alert activities: potential situation identification and assessment, information gathering and knowledge presentation, and planning and decision making, which includes exploring possible scenarios and other relevant hypotheses. These are near-term activities, with a scale of months to days. These begin as the responsibility of the tracking entity, and are passed on to the planning and decision entity.

Then there are crisis activities: situation identification and assessment, information gathering and knowledge presentation, planning and decision making (exploring possible responses), and rehearsal and other preparation. These are short-term activities, with a scale of days to hours. These are the responsibility of the planning and decision entity, which then interacts with the response entity as decisions are made.

Then there are the crisis response activities, which are immediate-term activities, with a scale of hours to seconds. These are the responsibility of the response entity. The planning and decision entity will monitor and collect feedback during the response execution, and provide any necessary physical details that are available (materials, routes, etc.).

Then there are after-crisis activities: assessment and feedback, which are used to improve the models and the processes. These are near-term activities, with a scale of months to days. These are the responsibility of the planning and decision entity.

Part of the modelling challenge for the rapid response system is to separate what is in the subject world from what is in the models. In general, the models will be prepared in advance, and they be able to extrapolate the situation through the time of intervention, including the style of intervention and the possible responses (for rehearsal, training, and planning purposes). The “known” parts of the models will “catch up” by the time of decisions, to allow real or hypothetical situation study. All of the models will allow predictions, hypothesis generation, and analysis.

An example timeline: When a trigger event occurs (a host of questions immediately arises here: who decides? how can they tell? how do they decide? a trigger event is one that instigates or should instigate a rapid response, because it signals an unusual or dangerous situation), several activities have to happen very quickly:

1. severity assessment,
2. model building and adaptation,
3. information customization,
4. crisis management team formation,
5. crisis management center construction,

and many others. These activities are performed by various entities in the situation management team, the most important of which is setting up the crisis management team: identifying the right players, scheduling their participation, gathering them together virtually, and supporting their interactions securely.

The rapid response system also needs tools and technologies that support these activities, and models to assess the effects of new ones on this same process. That is, it has models of its own processes, so it can assess the use of new technologies in improving its own behavior.

We can think of the several things that the system needs to know or to find out for (or in, or about) a crisis situation in a well-known sequence: who, what, where, when, why, and sometimes how. All of these are relevant and most of them are hard to determine.

3.2. Some Definitions

A scenario is a story that describes some collection of (presumably) relevant past events or activities, some current situations, and some possible futures. It is always described from a third-person, ubiquitous, point-of-view (which means that it is quite likely that no participant has all of the relevant information), even if it is not completely specified. That means that a scenario is a combination of partially elaborated problem statements, historical background event descriptions, contingencies, and partial solutions, generally organized into a narrative style.
One of the main differences between the hypothetical scenarios that we expect the team to use for planning, training, and rehearsal, and the “real” scenarios that the situation management team is being constructed for is the degree of completeness of the models: in the “real” scenarios, we never know everything, whereas we can specify the realistic (we hope) scenarios as if we know everything, even if none of the participants is able to determine it all.

Every scenario has a reference timeline, with a “now” point to distinguish the past events from the future possibilities, called the local time. The process of moving the reference “now” point forward is called scenario elaboration, and it is assumed that any scenario can be run forward or replayed. It is important for this purpose that the scenario “now” can vary under system control, and is not necessarily the same as the time at which the scenario is being played out (this is the same as the difference between “simulation time” and “wall clock time” in a distributed simulation).

At any given local time during the scenario, the situation describes the current state of affairs. It is a partially elaborated problem statement, with some historical background of event descriptions, generally organized to present information in support of a collection of possible courses of action for the participants (called decisions for the participants and contingencies for the others). Therefore, elaborating a scenario provides a sequence or trajectory of situations for each participant (whether or not they know all the relevant information). It is a sequence when the modeling uses discrete time, and a trajectory when it uses continuous time.

For our purposes, therefore, a scenario is like a situation setup with some actions already presumed to have been taken. Our model of a scenario includes at least the situation, with both the stakeholders (players) and the things that have happened; the context, including the geographical places and political divisions that are relevant, and the facilities in which activity is expected; the threats and their consequences; and any criteria for success or failure.

Decision justifications are often dependent on political policies and similar considerations. The associated reasons, but not the results, are beyond the scope of our models.

The situation management team cannot control all of the systems to which it needs to have interfaces. When reasoning chains go beyond the boundaries of its responsibility, we as modelers will record the source, the source information, and the source’s presumed responsibility, so that the team’s external linkages can be examined.

Some of these elements are known and some are not. Some of them cannot be known and must be inferred, and others are only important because they affect the possible courses of responsive action.

4. Virtual World Structure

In this Section, we describe the architecture of the Virtual World, the models it contains, and the mechanisms for the users to make changes, by adding, altering, combining, or deleting models.

We start with our intention to put everything in the Virtual World, all data, models, users, analyses, results, and everything else. Then the users can interact with their tools and with each other from within, so they can share information by going there. Parts of the Virtual World represent their own office areas, parts represent areas of interest in the external world, and parts represent conceptual structures derived from the input data.

In the next subsection, we describe the current events semantics in which we are interested (that is, the domain-specific terminology that is used to describe the current events), then we describe the architecture of the system and then the knowledge management. Finally, we describe some techniques for automatically incorporating information into the Virtual World.

4.1. Current Event Semantics

The basic purpose of a current events analysis system is to recognize facts, such as locations of people, places, facilities, and equipment, and movements of people, organizations, equipment, and money, relationships, such as co-occurrence and communication, and events, such as changes in the above facts, reports of unexpected relationships, or other observations. There are also some more difficult elements that are important to model, such as the function, purpose, or cause of an event. For this reason, we rely on human operators to provide a priori structures and relationships to the system, and to interpret the empirical structures that the system derives. These inputs include offering problem statements for operations, and abstracting the changes made by and the reasons for an event. We described above the four relevant time scales, and the corresponding parts of the situation management team.

We describe the first three of these entities in more detail. The surveillance entity is a permanent entity, with some rotating members (possibly temporary assignments of individual experts from different organizations and possibly different countries). It has the responsibility for “watching the world”, gathering history and background, and identifying potential crises. Its members include external liaison and information modelers. It is located primarily in the situation management center (which can be distributed, but has a physical center somewhere, called the “core”).

The tracking entity is also a permanent entity, with some rotating members. It has the responsibility for tracking potential crises, estimating their severity and likelihood, and creating the planning and decision entity for each crisis (and for assessing the situation management team processes). It
contains some liaison and subject- or area- experts, and it is also located primarily in the situation management center.

The planning and decision entity is not permanent. There is one instance of this entity per crisis, which is created anew for each crisis, from domain experts, current events experts, and operations experts (but it can have some permanent members). It has the responsibility for determining a response to a crisis, managing the situation during the crisis, and performing the post-crisis assessment. Some of these entities are set up and not activated, to allow very rapid activation for some potential crises (strategic policy determines which ones). It is located primarily in the crisis management center, which is distributed, but is expected to be partly in the core of the situation management center.

We also noted that the activities that this system must support fall into the same set of time scales, from long-term to immediate. The surveillance activities are intended to gather history and background. They include information gathering and knowledge presentation for historical background (we expect enormous amounts of open source material to be available and informative, and of course, there are also other sources from whom information can be requested).

The information of interest is geographical information, including map validation and / or production, ethnic distribution and other demographic information; climate information, including weather patterns (and predictions or observations of unusual patterns); historical background (significant individuals and groups, historical events, cultural development, technology development); political situation assessment (describing stakeholders and their interests, divided up in at least two different ways: us, them, or others; people, organizations, or governments), including ethnic background (who hates whom, where they are, who leads them, how they are moving and have moved in the past); economic situation assessment (describing winners and losers, trends and forecasts, trade relationships and dependencies); legal situation assessment (including applicable treaties, laws, policies).

Other activities include model building (generic hypothesis examination and analysis, simple scenario generation), event detection (some of these events are indications and warnings of important activities, others are “trigger events” that are specifically identified and asked for by other entities, and others may just be surprises, though we hope not very many). This entity is expected to pass potential crises to tracking entity, along with their background and context.

Similarly, the tracking activities are intended to monitor potential trouble spots, identify crisis situations, and create the planning and decision entity for each crisis (and assess the situation management team processes). They include information gathering and knowledge presentation for current events (also, specific information can be requested from other sources).

The information of interest is more detailed geographical information, terrain and construction maps, weather information, and technology capabilities; political and economic situation (ethnic sensitivities and conflicts, important individuals and groups); and legal situation.

Other activities include model building (specific hypothesis examination and analysis, generation of scenario refinements), which means model adaptation to particular circumstances, information customization, and model integration; planning and decision making (exploring possible scenarios and other relevant hypotheses); potential situation identification and severity assessment; and creation of the planning and decision entity (the crisis management team): this activity must identify the right players, schedule their participation, gather them together virtually, and support their interactions securely. It must pass crises to the planning and decision entity, and receive crisis feedback from the planning and decision entity (process problems and change suggestions).

Finally, the planning and decision activities are intended to prepare for a crisis situation (for planning). They include information gathering and knowledge presentation for crisis management (the system can also request specific information from other sources). The information of interest is maps of buildings, equipment, or other essential construction; the identification of materials, escape routes, etc.; weather forecasts; possible opposition, strengths, locations, possible civilian interactions or other interference (different factions and technologies available to them, important individuals and how to recognize them); and legalities.

Other activities include model building (situation and response scenarios for training, capabilities and equipment required for response); planning and decision making (exploring possible responses); rehearsal and other preparation (response team identification, training, organizing delivery and extraction transport and execution support); and getting execution feedback from response entity for post-crisis assessment (identify unforeseen situations and missing information, process failures and planning gaps, and pass them back to the tracking entity, along with change suggestions).

4.2. Architecture

It is clear from the preceding that this is a model development system with a large empirical component. The architecture of this system can be seen in the diagram in Figure 1. It takes open source information, passes it through a number of automatic analyses, and puts the results into a document knowledge base (doc KB in the figure). The user has access to that knowledge base through the Query Processor (QP in the figure), which records all transactions in the query / response knowledge base (qr KB in the figure),
and the user can insert other open source information from the web through a Web Browser (WB in the figure). The user can also build models of situations using the Model Builder (MB in the picture), which puts models into a commonly accessible model base.

There is a background analyzer that uses all of these sources of information to identify interesting structures or define new terms (the automatic analyses and the background analyzer will be described in a subsequent paper).

Finally, the user has access to a Virtual World (VW in the architecture figure, which consists of models of concepts, automatically built by the background analyzer, and models of places and / or situations, which are built by the users in accordance with the information they receive.

Even though we have described four different kinds of user activity, they are not to be considered as separate interfaces. First, all interactions are in the Virtual World, with the users being able to communicate and collaborate with each other, and access all of their other tools from within the Virtual World. Secondly, they can enter into the models of situations, to examine them in detail and discuss the changes that should be made. Third, the Web Browser and the Query Processor can be the same also, with a simple set of commands from the user to indicate whether this is an ordinary web request, a query for the document knowledge base, or some other tool to be invoked.

4.3. Knowledge Management

Knowledge management in this system is interesting, since both human users and computer programs need to access various parts of the knowledge recorded and derived in the system. Here we argue that a simple web interface is not sufficient.

First, the web alone is not good enough when it is treated merely as a data repository, since a repository is like an information dump; material sits there and waits for users to request it (also sometimes called information “pull”), without any direct distribution of indexing terms. When users are required to specify their interests in advance, so that the servers can notify them of relevant information (also sometimes called information “push”), they still miss out on many relevant sites, since their interest profiles cannot be sufficiently precisely stated, and since those profiles are not actively promoted. Search engines are failing to keep up with the growth rate of even well-indexed sites, so only the users who are interested in some very specific and well-specified information can find it. Exploration is a hit-and-miss affair.

Active processes are needed, continually, for consistency maintenance, integration of new information, and discovery of new derived information. Current search engines can do some of this, but it means that when the amount of Web information gets big enough, most of the server time will be spent responding to indexing requests, not search requests.

Our use of conceptual categories in this complex application allows us to represent many disparate kinds of information, and to account for different viewpoints and context. We can represent categories of phrases or phrase structures, derived from the incoming messages, and we can represent a priori categories that we know are important to the problem domain.

5. Conclusions

In this paper, we have described many of the kinds of information required in a current events analysis system, and shown how our approach to Virtual Worlds, our Conceptual Categories knowledge representation, and our Wrapping integration infrastructure can be used together to provide an extremely flexible modeling environment for building responsive models of current situations.

We have shown how some information is derived and applied automatically, and how other interpretations must be defined by human users. We have shown that many different models can interact in the same environment, and that they can all interoperate at some level of abstraction.

Computational Reflection plays an essential role in the continual improvement of the system, since it contains models of its own behavior that are used to monitor and evaluate system behavior, which allows the system and its users to assess the weak points and identify potential locations for improvement.
References


