

## Multi-Agent Situation Management for Supporting Large-Scale Disaster Relief Operations

John F. BUFORD, Gabriel JAKOBSON and Lundy LEWIS

**Abstract-** Natural and human-made disasters create unparalleled challenges to Disaster Situation Management. We describe two important information technology solutions meeting these challenges: distributed situation-driven disaster relief operations management and a multi-agent architecture scalable to large numbers of interacting agent platforms. To address limitations of event-driven cognitive processing, we extend the Belief-Desire-Intention multi-agent systems model with the capability of situation awareness. We describe how the key functions of event collection, situation identification, and situation assessment are implemented in a multi-agent systems architecture suitable to the characteristics of large-scale disaster recovery. To meet the requirements of large-scale agent systems, we propose an integrated agent platform with a peer-to-peer overlay which combines the semantic discovery mechanisms of agent systems with scalability to thousands of nodes of peer-to-peer overlays.

**Index Terms**—Disaster relief operations, situation management, multi-agent systems, peer-to-peer systems, BDI agent model, FIPA

### 1. INTRODUCTION

Natural and human-made disasters create unparalleled challenges to the response, relief and recovery operations. Effective Disaster Situation Management (DSM) is a critical factor in meeting those challenges. DSM is a complex multi-dimensional process involving a large number of mobile inter-operating agents - teams of humans and systems.

The conventional DSM practice involves various, often loosely coupled databases, event monitoring and dispatch systems, public or private voice and data communication systems (over fixed or wireless lines), and public emergency announcement systems. During the last decade significant progress was made in the development and deployment of integrated disaster information monitoring systems, including weather, earthquake, fire, chemical, radiological and biological activity monitoring systems. Experimental computer supported cooperative work systems were developed for planning the routes and dispatching the emergency medical, fire, and police vehicles. The introduction of GPS and GIS added significant functionality to the disaster monitoring and emergency team dispatching systems. In recent years the progress in miniature low-powered sensors and sensor networks, unmanned aerial vehicles (UAVs), geo-spatial information systems, wireless broadband communications,

and new emerging solutions of cognitive information processing, situation management, distributed computing, and agent technologies has opened opportunities for new solutions for DSM.

The Multi-Agent System (MAS) [41] has been widely recognized as an effective solution in modeling large number of dynamic interacting entities due to (a) the distributed organizational framework of MAS, (b) the use of perceptual and reasoning models of mobile intelligent agents, and (c) the natural fit to model collaboration between the teams of agents. Such characteristics of MAS directly fit the requirements of DSM. Several different architectures of MAS have been proposed, including the Belief-Desire-Intention (BDI) agent architecture [3][33]. Since its introduction, the BDI model has experienced several functional advancements; however recent attention to the operation of large-scale distributed dynamic systems have revealed a weakness of the BDI model, namely the lack of an adequate capability to cope with complex operational situations.

In this paper we focus on cognitive aspects of DSM and require a cognitive-level MAS that is organized in a reactive situation-driven architecture, supports heterogeneous and varying populations of agents, and scales to many thousands of interacting agent systems, where each system might have many agents. These results draw on our previous work in developing the situation management paradigm [19][20], applying situation management to homeland security [5], exploring various facets of situation management in disaster relief operations [6], defining the BDI-SM model [4], defining the general coupling of agent systems with service-oriented peer-to-peer architecture [7], and developing efficient mechanisms for low-latency large-scale peer-to-peer overlays [8].

The rest of the paper is organized as follows. Section 2 describes an overall model for DSM applied to medical relief operations. Section 3 explains the situation management paradigm and its relation to disaster recovery operations. Decomposition of the DSM model into a multi-agent system is described in Section 4. Sections 5 presents the agent model using the Belief-Desire-Intention paradigm and a sample medical relief ontology for the BDI agents. Section 6 describes the BDI-SM realization in several BDI agent languages. The use of the BDI-SM agent system in a large-scale peer-to-peer overlay network is described in Section 7. The last section concludes the paper.

Manuscript received September 24, 2006; revised December 18, 2006. This paper is extended from "Situation-Aware Multi-Agent System for Disaster Relief Operations Management" published at 3<sup>rd</sup> International Conference on Information Systems for Crisis Response and Management, Newark, NJ, USA, May, 2006.

J. Buford is with Altusys Corp, Princeton, NJ 08540 (email: buford@altusystems.com). G. Jakobson is with Altusys Corp, Princeton, NJ 08540 (email: jakobson@altusystems.com). L. Lewis is with Southern New Hampshire University (email: l.lewis@snhu.edu).

## 2. CASE STUDY: DISASTER MEDICAL RELIEF OPERATIONS MANAGEMENT

### 2.1 Understanding Medical Relief Operations

A critical element of disaster recovery is medical relief to provide treatment and support to those injured due to the disaster or whose previously sustained conditions or vulnerability (e.g., elderly or displaced patients) place them in medical jeopardy due to the disaster. Medical relief operations include field mobile ambulatory aid, evacuation processes, emergency hospital operations coordination, and logistics support for medical supplies and equipment.

The scale of the disaster determines the number of participating relief organizations, the extent of use of extra-regional medical teams, and the overall complexity of the coordination and communication. In the case of recovery involving international teams, language and equipment differences compound the communication problem. The scope of the disaster may frequently make local medical facilities inoperative, and it may place relief teams in hardship conditions or at risk due to infrastructure damage and unknown conditions, limited food and water supplies, societal breakdown, and lack of law enforcement support.

At the information level, there is a significant distribution of data across teams of people, systems, information sources, and environments, and the ongoing

data collection and changing state makes the overall picture very dynamic. Further, there is a strong benefit to the overall effort if different teams can share relevant information. For example, in order to perform effective provisioning of field medical services, the mobile ambulatory teams need to develop a common understanding of the medical situation on the ground, share road and access information, and coordinate medical relief and evacuation operations.

Specific tasks that need to be supported in an automated DSM system include:

- Overall planning of the medical recovery effort including sizing, personnel, equipment and supply requisition
- Dispatching, scheduling, and routing of mobile ambulatory and other emergency vehicles
- Evacuation of victims
- Prioritization of relief operations
- Maintenance and well-being of relief personnel
- Resource allocation
- Coordination and communication between medical teams and to other relief operations

DSM (Figure 1) constructs a real-time constantly refreshed Situation Model from which relief operations can be planned and updated. The Situation Model contains a knowledge-level view of the disaster from the medical

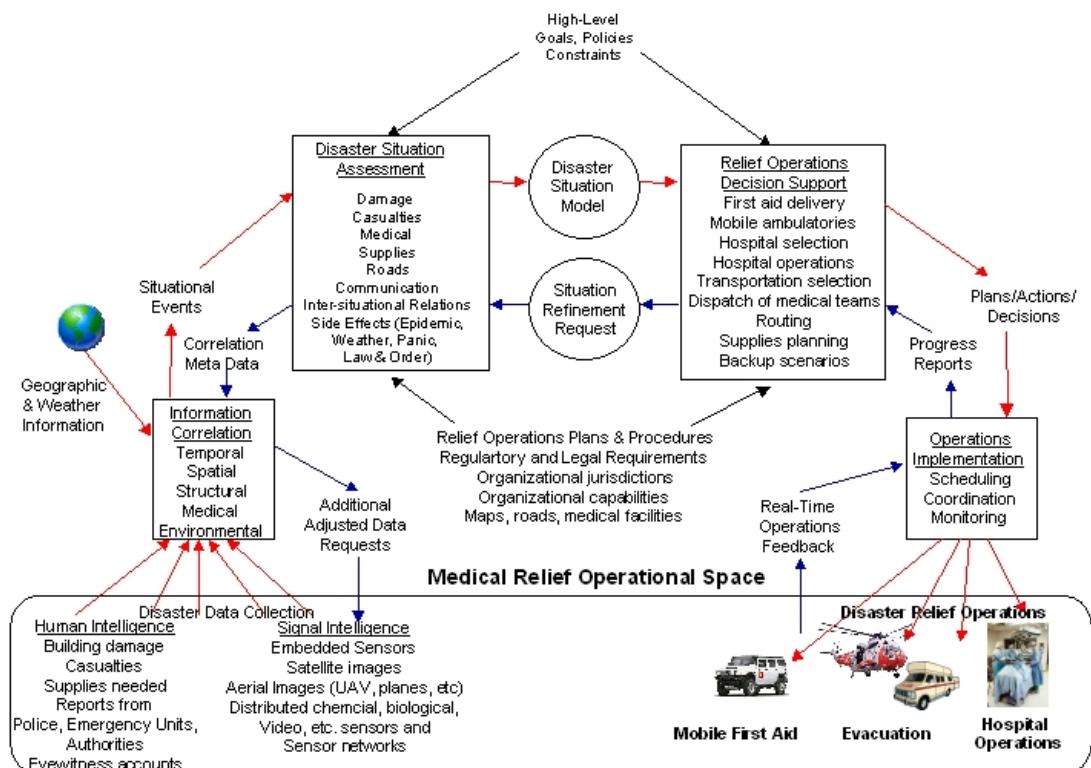


Fig. 1. Closed-loop disaster medical relief operations management using DSM

relief perspective, using an ontology specifically designed for that domain. The model is created and updated by a constant flow of events and reports collected from the operational space. These events include both human intelligence and signal intelligence. Because of the large amount of raw data being collected, the event stream needs to be processed and correlated to produce “situational events”, i.e., events at the domain level. This reduction and inference step is performed by an information correlation stage. We describe later in the paper how the information correlation function and the situation assessment function can be distributed in MAS architecture.

Integrated with the real-time Situation Model are decision support systems (DSS) for medical relief operations. The DSSs rely on the Situation Model and operations staff oversight to manage the scheduling, dispatching, routing, deployment, coordination and reporting tasks. A chain of distributed communication and control systems leads from the DSSs to the medical personnel in the field to direct the execution of these tasks.

Medical relief organizations have the responsibility and expertise to prepare for disaster recovery in many different scenarios, which includes defining goals and policies, enforcing legal and regulatory requirements, and specifying deployment plans. These goals, policies, requirements and plans are incorporated into the DSM knowledge base and are used by the situation awareness function and DSS to ensure plans, actions, and priorities are formed consistently.

## 2.2 Control Path from Relief Actions to Refined Data Collection

DSM (Figure 1) adapts to requests and feedback from medical relief personnel and DSS, as indicated in the reverse path. These requests can lead to refinement of the Situation Model, meta-level guidance to the information correlation function, and a focus on specific sensor data.

As an example, consider that a particular assessment of a situation will likely contain missing or uncertain information, although enough information is present to maintain a sufficient confidence level in the assessment. The missing information may be uncovered by the DSSs or Operations Implementation function, as shown in Figure 1. A message may be sent back to the Correlation function indicating a high-priority or crucial need for the missing information, upon which the Correlation function takes action to obtain or verify it. If verification is not forthcoming, the situation assessment remains as is and the request remains outstanding. If verification is indeed forthcoming, the situation assessment may change and the results may affect the Operations Implementation function.

## 2.3 An Emergency Situation Recognition Scenario

Assume that an earthquake of magnitude of 7.2 in Richter scale happened with an epicenter of 2.5 miles from a major industrial city, causing significant damage to roads,

bridges, power stations, buildings, water supply lines and other vital infrastructure elements. In addition, ammonia escaped from a refrigeration plant forming a toxic cloud over a densely populated area in the city. Two medical emergency vehicles (MEV) were dispatched to a certain disaster area using alternative routes, a direct route although through the contaminated area, and a detour route with moderately damaged but still relatively safe road conditions.

An emergency situation during this medical relief and rescue operation could be recognized using the following situation recognition rule, which is described in a language similar to CLIPS [10]. Suppose an operations coordination event message of type A was issued at time t1 from the first MEV ?mev1, but during the following 10-minute interval an expected event of type B was not issued by the second MEV ?mev2. The events to be correlated, then, are A and not-B. Note that not-B is treated formally as an event. An additional constraint is that MEV ?mev1 and ?mev2 belong to a team. This constraint is expressed by a grouping object GROUP with identified group type and parameters. The time constraint between events A and not-B is implemented using a temporal relation AFTER.

SituationRecognitionRuleName:

**EXPECTED-EVENT-RULE**

Conditions:

MSG: EVENT-TYPE-A ?msg1  
TIME ?t1  
VEHICLE: VEHICLE-TYPE-MEV ?mev1  
Not MSG: EVENT-TYPE-B ?msg2  
TIME ?t2  
VEHICLE: VEHICLE-TYPE-MEV ?mev2

GROUP: GROUP-TYPE-MEV ?mev1 ?mev2

AFTER: ?t1 ?t2 600

Actions:

AssertSituation:

LOST-MEV-CONTACT-SITUATION  
VEHICLE1 ?mev1  
VEHICLE2 ?mev2  
EVENT1 ?msg1  
EVENT2 ?msg2

If the conditions of the rule EXPECTED-EVENT-RULE are true, then the situation LOST-MEV-GROUP-CONTACT-SITUATION is asserted into the event correlation /situation recognition process memory.

Below is given a relatively simple association between a situation and a possible action plan, where the situation LOST-MEV-GROUP-CONTACT-SITUATION has an embedded action, which invokes plan SEND-EMERGENCY-HELICOPTER.

SituationName LOST-MEV-CONTACT-SITUATION

SituationClass MEV-SITUATION

Parameters

VEHICLE1  
VEHICLE2  
EVENT1  
EVENT2

Actions  
PLAN SEND-EMERGENCY-HELICOPTER

### 3. SITUATION MANAGEMENT AND MODELING

Situation Management (SM) is defined [19][20] as a synergistic goal-directed process of (a) sensing and information collection, (b) perceiving and recognizing situations, (c) analyzing past situations and predicting future situations, and (d) reasoning, planning and implementing actions so that a desired goal situation is reached within some pre-defined constraints (Fig. 2).

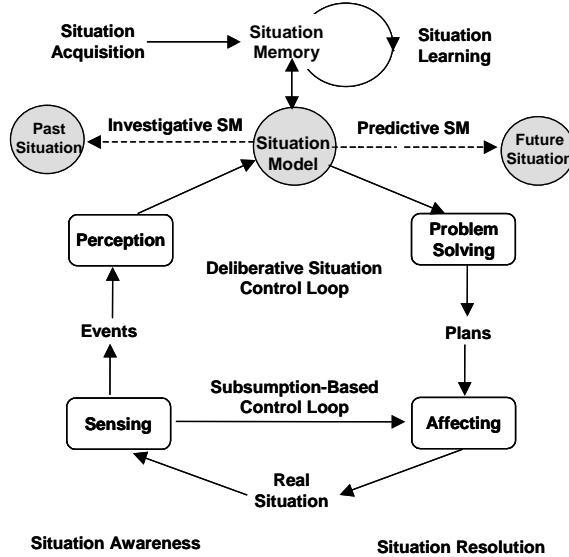


Fig. 2. Generalized view of Situation Management [20]

Depending on the specific situation on the ground, one can distinguish between three types of SM: diagnostic, control and predictive. It is useful to see these types mapped to the time axis. The diagnostic type of SM is a retrospective task of determining why a certain situation happened. The control type of SM aims to change or maintain the current situation, while the predictive SM plans to project possible future situations, e.g., threat situations.

Construction and constant refinement of the disaster situation model is one of the prerequisites of successful DSM. While modeling situations, we assume that there exists a world that can be sensed, perceived, reasoned about and affected. The world is populated with entities that are engaged in different class, structural, causal, spatial, temporal and other relations. The entities possess distinguishable attributes with certain attribute values. Some entities are active, they change their state in space and time, and some of the entities can interact with other entities forming entity systems. Importantly, entities are time-dependent objects with their time of creation  $t$ , lifespan  $\delta=(t, t')$ , and time of elimination  $t'$ . Consequently, the attribute value of an entity is defined only during the existence of the entity.

A world situation is an aggregated state of entities and inter-entity relations of the world, which are observable at a specific time period. A world situation model is a representation of situations happening in the world.

Formally a situation is defined as follows.

Let  $e$  be an entity  $e \in E_i$ , where  $E_i \subseteq U$  is a subclass of all entities of the universe  $U$ . Each entity  $e$  is represented by its arguments  $\{a_1, a_2, \dots, a_p\}$  with a corresponding set of argument value domains  $\{v_1, v_2, \dots, v_p\}$ . Among all entity arguments we will define a subset called situational arguments. The semantics of situational arguments is declared or computed depending on the particular operational context of the application.

Importantly, entities are time-dependent objects with their time of creation  $t$ , lifespan  $\delta=(t, t')$ , and time of elimination  $t'$ . Consequently, the attribute value of an entity is defined only during the existence of the entity, i.e.  $a_i(t), t \in \delta$ . Now, the situation  $S_e$  on entity  $e$  during time  $d$  is defined in the following inductive way:

$$S_e(d) = \{ \langle a_1(t), a_2(t), \dots, a_p(t) \rangle \in v_1 \times v_2 \times \dots \times v_p / \forall (t, t') \in d \\ [\langle a_1(t), a_2(t), \dots, a_p(t) \rangle = \langle a_1(t'), a_2(t'), \dots, a_p(t') \rangle] \},$$

where  $d \subseteq \delta$  is the duration of the situation. We call  $B=\{e\}$  the base of the situation.

Complex situations could be constructed from other situations using set-theoretical union and inter-section operations, as follows:

If  $S_{B_1}(d_1)$  and  $S_{B_2}(d_2)$  are two situations, where  $B_1, B_2 \subseteq U$  and  $d_1, d_2$  are subsets of common life-spans of all entities in  $B_1, B_2$ , correspondingly, then,

$$S_B(d) = S_{B_1}(d_1) \cup S_{B_2}(d_2) \text{ and } S'_{B'}(d) = S_{B_1}(d_1) \cap S_{B_2}(d_2)$$

are situations, where  $d = d_1 \cap d_2$ , and correspondingly

$$B = B_1 \cup B_2 \text{ and } B' = B_1 \cap B_2.$$

In defining the notion of a situation, let us provide some explanations. First, we intentionally omitted to mention relations  $R_j \subseteq E_1 \times \dots \times E_m = \{(e_1, \dots, e_m) / e_1 \in E_1, \dots, e_m \in E_m\}$ , since we will handle a relationship and corresponding entities as a higher-level entity with the use of an aggregation operation. Second, while considering situations, we look on a subset of all entities, called active entities. Like situational arguments, the semantics of active entities is declared or computed depending on the specific operational context. Due to our notions of active entities and situational arguments, multiple different situations can be defined on the same set of entities.

Further discussion of the formal model of our definition of situations is described elsewhere [20].

Different situation definitions can be found in the literature. For example, McCarthy's situation calculus [28] uses a state-based approach, where a situation is considered a snapshot of a complete world state at a particular time. Pirri and Reiter [31] defined a situation as a sequence of actions enabling calculation of the current state knowing the initial state and the sequence of actions transforming the initial state. Reiter's definition was introduced with planning tasks in mind, e.g., robot actions planning. Yet another definition uses fluents [1], i.e., situation-dependent functions to define the situations. Our

definition of a situation is closer to McCarthy's definition but assumes active entities and situational arguments as the primary constructs.

The functions of situation awareness considered as a part of the overall SM process (Figure 2), have been studied by several authors, most notably by M. Endsley [13], who defined the basic concepts of situation awareness: perception, comprehension and projection.

#### 4. MULTI-AGENT SYSTEMS APPROACH TO DSM

##### 4.1 Basic Principles of the Approach

We see situation management (Fig. 2) as a closed-loop process (Fig. 3), where primary information is sensed and collected from the managed operations space (the World), then analyzed, aggregated, and correlated in order to provide all required inputs for the situation recognition process. During the next step the reasoning processes are performed to select predefined plans or automatically generate them from the specifications embedded in the situations. Finally the actions are performed to affect the World. As the World gets affected, new information about the World is sensed and the process is repeated.

The reasoning and situation recognition steps in the SM control loop are performed by agents of the MAS. The principle components of our MAS approach to DSM then are (Fig. 3):

- General Situation Management Control Loop
- Distribution and Specialization of Agents
- Agent Teams and Control
- BDI-SM Agents

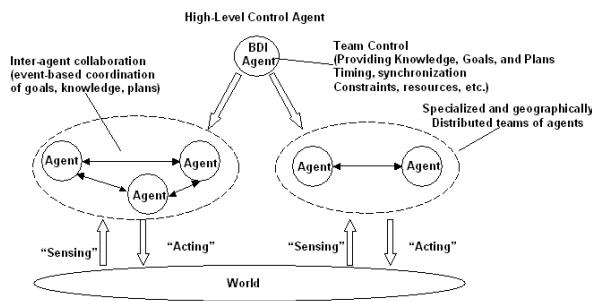


Fig. 3. MAS approach to DSM control

The distribution and specialization of agents in the DSM environment involves mapping the physical agents (i.e., vehicles, robots, human teams, medical equipment) into the abstract framework of the MAS. This task involves considerations including agent granularity, relative autonomy of physical agents, information sharing, network capacity, energy consumption, and security. The structure of the DSM operations fits the MAS organization, where distributed agent teams (communities) having peer-to-peer decentralized internal communication among the agents are coordinated externally by a higher-level agents (Fig. 3).

In this paper we introduce the BDI-SM agent model where the paradigm of classical plan invocation by a single event [41] is replaced by an enhanced model, where plans

are invoked by situations. As illustrated in Figure 1, the DSM domain is characterized by undertaking actions that in most cases require simultaneous analysis of multiple events. The introduction of the paradigm of plan invocation by a situation has specific importance to the DSM domain, since it enables initiation of plans not on a basis of a single event, but taking account the patterns of multiple events. Such event patterns will be recognized due to the use of event correlation.

##### 4.2 Abstract BDI Agent

The Belief-Desire-Intension (BDI) model was conceived as a relatively simple rational model of human cognition [3]. It operates with three main mental attitudes (Fig. 4): beliefs, desires and intentions, assuming that human cognitive behaviour is motivated by achieving desires (goals) via intentions providing the truthfulness of the beliefs.

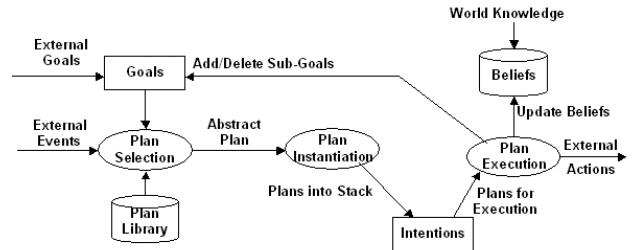


Fig. 4. Abstract BDI agent architecture

As applied to agents, the BDI model received concrete interpretation and a first order logic based formalization in [33]. Among many BDI agent models, the dMARS formalism serves as a well-recognized reference model for BDI agents [10]. Since we use the dMARS framework as a starting point to our approach on situation-aware BDI agents, we will informally sketch the basic notions of dMARS. A BDI agent is built upon the notions of beliefs, desires, events, plans and intensions.

Beliefs are the knowledge about the World that the agent possesses and believes to be true. Beliefs could be specifications of the World entities, their attributes, relations between entities, and states of the entities, relations. In many cases, the agent's beliefs include the knowledge about other agents as well models of itself. Desires are the agent's motivations for actions.

Plans are operational specifications for an agent's actions. An agent's plan is invoked by a trigger event (acquisition of a new belief, removal of a belief, receipt of a message, acquisition of a new goal). When invoking a plan, an agent tests whether the plan invocation pre-conditions are met and tests run-time conditions during the plan execution. The actions in the plan are organized into an action control structure, which in dMARS is a tree-like action flow. Actions could be external ones, essentially procedure calls, or internal ones of adding and removing beliefs. Abstract plans are stored in the agent's plan library.

During agent operations certain abstract plans are selected from the library and instantiated depending on variable bindings, substitutions and unifications.

An agent's intention is understood as a sequence of instantiated plans that an agent is committed to execute. Always while responding to a triggering external event, an agent is invoking a plan from the plan library, instantiating it and pushing into a newly created stack of intentions. In addition, when the agent responds to an internal triggering event, i.e., an event created by an internal action of some previous plan instance, then the new plan instance is pushed into the stack of the previous plan that caused the invocation of the new plan instance. An abstract architecture of a BDI agent is presented in Figure 4.

### 4.3 Abstract Situation-Aware BDI Agent

The current BDI models have a simple plan invocation model, where either the plan is triggered by a single event or by a single goal. Preference between these two invocation methods leads to event or goal-directed planning of actions. While the single goal directed planning satisfies the majority of the application needs, the single event directed planning does not. In the majority of cases of disaster operations planning, battlefield management, and security applications, decisions are made not on the basis of a single event, but rather correlating multiple events into a complex event and mapping it to a situation happening in the operational space. Central to our approach to extending the capabilities of the BDI agent is the introduction of situation management (BDI-SM). According to this approach, a plan is invoked by a situation rather than by a single event (Fig. 5).

The external events received by the BDI-SM agent and the events generated by the agent itself while executing the plans are correlated into compound high-level events called synthetic events. The real-time event correlation process [19] takes into account temporal, causal, spatial, and other domain-specific relations between the multiple events as well constraints existing between the information sources producing the events. The event correlation process could be an iterative multi-stage process, where some synthetic events could be used for building more complex synthetic events.

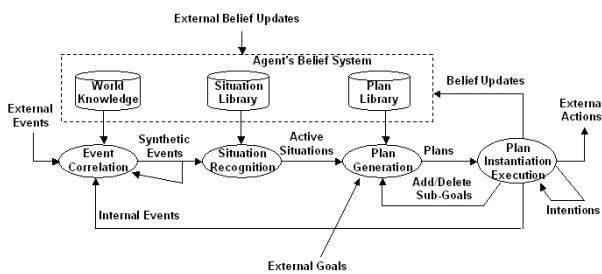


Fig. 5. BDI-SM agent

The synthetic events serve as a basis for recognizing situations taking place in the world. They are used in the

triggering patterns of abstract situations while invoking them from the Situation Library. The multiple invoked situations are instantiated and are combined into an overall situational model of the world.

The situations contain either references to the plans that will be invoked by triggering conditions specified in the situations, or contain specifications for reasoning and generating plans. The steps of plan instantiation and execution are similar to those performed in the dMAS BDI model.

A different extension of the abstract BDI architecture towards situation awareness and pro-active behavior is described in [37]. The approach is based on evidence theory and the introduction of a meta-level action control mechanism in the agent's interpretation cycle. The authors consider situations as collections of beliefs with predefined level of evidence; however the construction of situations is handled outside the agent's interpretation cycle and is not discussed.

BDI agents are used with a team of human agents (operators) to resolve joint human-machine situation awareness capability in [40]. The proposed solution is based on the JACK BDI agent implementation [16].

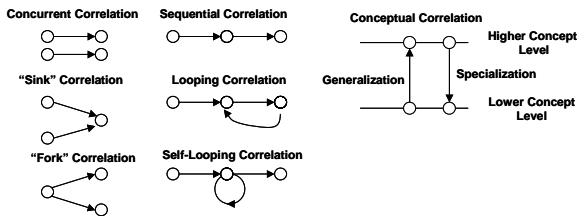


Fig. 6. Correlation interconnection types

### 4.4 Event Correlation Process in BDI-SM Agents

Event correlation is considered to be one of the key technologies in recognizing complex multi-source events. We are using event correlation as a major tool leading to situation recognition. As it will be shown later, the importance of the event correlation process influenced us to introduce a special type of event correlation agent.

The task of event correlation can be defined as a conceptual interpretation procedure in the sense that a new meaning is assigned to a set of events that happen within a predefined time interval [21]. The conceptual interpretation procedure could stretch from a trivial task of event filtering to perception of complex situational patterns occurring in the World. The act of recognition of a new situation by the correlation procedure could be formally handled as a synthetic event, and as such, it is a subject for further correlation.

The process of building correlations from correlations allows the formation of a complex fabric of multiple interconnected correlation processes, suitable for the paradigm of integrated distributed cognition and collective behavior that is proposed in this paper. In the "correlation fabric" we will consider several basic connections between correlation processes as shown in Figure 6. Intermixing between

different correlation connections creates a flexible and scalable environment for complex situation modeling and awareness solutions.

While recognizing situations happening in the DSM World, the event/situation correlation process should take account of a variety of different conditions and relations occurring between the static, dynamic and mobile entities of the World. In order to match the complexity and richness of the World from the event/situation correlation perspective, we introduce the following dimensions to the correlation process:

- Spatial Dimension – geo-spatial, topological and structural relations;
- Temporal Dimension – temporal relations, time-dependent evolution and life-cycle of the entities;
- Conceptual Dimension – concept, concept classes, and concept ontologies;
- Computational Dimension – event filtering, suppression, escalation, and counting, and callable functions;
- Domain Dimension – entities, relations (including causal and action-oriented relations), constraints, policies, first principles, etc., specific to the domain;
- Logical Dimension – Predicate calculus expressions over the terms formed from the statements using the elements of the previous dimensions.

The temporal model-based event correlation technology used in this work has been developed and implemented for managing complex telecommunication networks. More details about the technology can be found in [21].

#### 4.5 Situation Recognition and Prediction

Our approach to DSM situation recognition and prediction is to use a specific model of analogy-based reasoning called case-based reasoning (CBR), where a case is a template for some generic situation [25][29]. The formation of the library of standard case templates for representing the typical generic situations allows (a) construction of specific DSM models by selecting the appropriate case templates, (b) modifying and instantiating the selected cases with concrete parameter values, and (c) combining the instantiated cases into overall case representation of the situation. The CBR research community has developed effective solutions for tasks (a)-(c). Further, the CBR approach allows learning from experience and adapting more-or-less standard situations to accommodate the nuances of current situations.

The general operation of CBR as applied to DSM is (i) based on current information and a similarity metric, to retrieve generic situation templates from a situation library, (ii) instantiate and optionally adapt the generic situation template to fit the nuances of current information, (iii) execute the plan or decision in the adapted situation and observe the results, and (iv) enter the adapted situation and results back into the situation library for future use. To illustrate in the abstract, suppose a generic situation in a situation library holds a particular plan for a problem

where the plan is based on the value of a variable  $x$ . Let the retrieved situation be as follows:

```
Given Situation S with parameter x, then
Perform action A(x)
Perform action B(x)
Make decision d=C(A(x), B(x))
```

Here, A, B, and C may be functions that take numeric parameters or they may be inferences from symbolic parameters. In practice, an operator might find that the decision is inadequate because an additional parameter  $y$  appears that renders the decision ineffective. The introduction of the new parameter  $y$  may cause one to replace the function  $B(x)$  with a new function  $B(x, y)$ , thereby improving the decision. Further, parameter  $x$  in the current situation might be some new value of  $x$ , say  $x'$ . The operator can adapt the retrieved situation using CBR adaptation methods called parameterized adaptation and critic-based adaptation. Thus, the adapted situation might be as follows:

```
Given situation S with parameters x' and y, then
Perform action A(x')
If y=null then Perform action z = B(x')
Else Perform action z = B(x', y)
Make decision d=C(A(x'), z)
```

The adapted situation that is entered in the situation library will cover future problem-solving situations in which only  $x$  is available and in which both  $x$  and  $y$  are available. Also, it is expected that further experiences with situation S will enhance the knowledge required to perform tasks in future situations that are similar to S. In this way, the system's knowledge is improved with experience.

This abstract example demonstrates three features of cognitive situation-aware agents: First, it demonstrates how such agents exhibit a degree of learning with use. Once a retrieved case is adapted and improved upon, the adapted case is retained in the case library for future use. Some CBR-based agents employ *master cases* in which the totality of experiences with a problem is recorded in one case, while other CBR-based agents record individual experiences in individual cases. Both approaches provide learning capabilities in situation-aware agents.

Second, it demonstrates how alternative situations can be ranked with confidence factors based on the available information -- the situation produced when both  $x$  and  $y$  are available would be of higher confidence than a situation produced when only  $x$  is available, all else being equal. Third, it demonstrates how the system may uncover impediments or opportunities -- the situation template may be retrieved when only  $x$  is available, whereupon the system discovers an opportunity for a more refined decision if  $y$  were available, and thus generates a request accordingly.

#### 5. ONTOLOGIES FOR BDI-SM AGENTS

Situation management is a knowledge intensive process and central to this process is situation knowledge

representation, i.e., how to effectively and efficiently describe entities, relations, situations, goals, and reasoning processes related to situation modeling and reasoning. One of the most effective tools for handling knowledge is ontology. Informally, ontology is a set of concepts which define the domain entities and relations between them. Ontology can be expressed using text, graphs, structured data, or tables. Formally, ontology is often expressed using languages based on first (or higher) order predicate calculus, so that detailed, accurate, consistent, sound, and meaningful distinctions can be made among the entities, properties, and relations. The roots of ontology as a knowledge representation method can be found in semantic networks as well as in frame representation languages.

The Semantic Web community with the development of the Web Ontology Language (OWL) has made significant progress in formal ontology specification. OWL has been used with moderate success for describing situations; however the lack of explicit declarations for describing the dynamic components of situation management (events, situations, time, temporal relations, situation transitions, etc.) limits the expressiveness of OWL. Recent extensions to OWL have expanded its rule-based reasoning capabilities (SWRL – the Semantic Web Rule Language).

There are three major components to an ontology: (a) vocabulary - the set of terms of the modeled domain; (b) structure - statements formed by terms and relations between them; and (c) rules for semantic interpretation of the statements. Ontologies have multiple dimensions, depending on different application domains as well the level of abstraction of the concepts that the ontology represents. Regarding those dimensions, the following types of ontologies can be defined:

- Core ontologies – definitions of general concepts that have a wide scope of validity and usually represent the basic concepts and the first principles of the domain. Usually, core ontologies represent theoretical or abstract concepts like entity, relation, attribute, time, goal, list, and set.
- Domain ontologies – describe the concepts of a particular application domain such as medical or transportation. Domain ontologies might form complex multi-level hierarchies of component ontologies, e.g. medical ontology, clinical ontology, internal medicine ontology, cardiology ontology.
- Situation ontologies – describe typical time-dependent states of the entities and relations between them, e.g. road condition ontology, infrastructure damage situation ontology.

Figure 7 depicts three (simplified) related ontologies: Patient Ontology, Hospital Ontology and Medical Relief Operations Ontology. The Patient Ontology illustrates only a class hierarchy between the entities, the Hospital Ontology shows some complex class, structural, and domain-specific relations between the entities. For example, the entity Clinical Service contains four entities (hospital divisions: Internal Medicine, Surgery Ward, Gastroenterology and Cardiology divisions. The entity

Clinical Service is by itself a class of Service and at the same time Reception Ward and Radiology & Radiotherapy provide services to the Clinical Service.

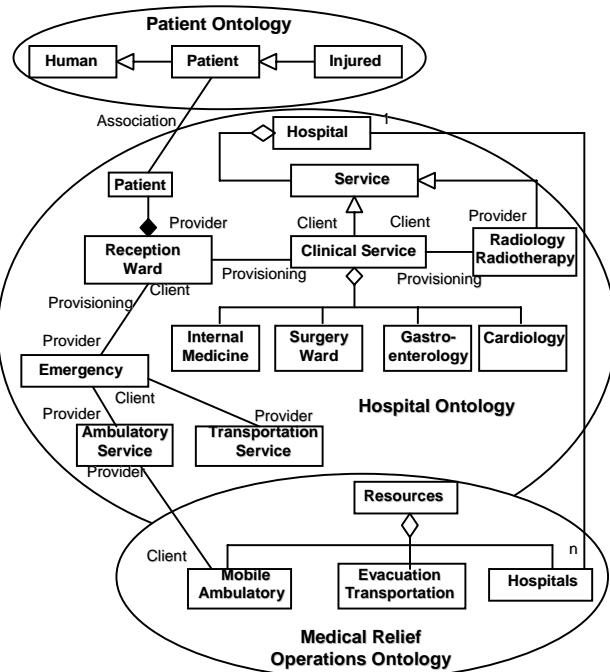


Fig. 7. Sample ontology for medical relief operations

Figure 7 underlines an important aspect of organizing knowledge: structuring of knowledge using multiple inter-related ontologies. One challenge in building multiple ontologies is semantic consistency of terms (entities).

There are extensive resources on medical taxonomies, disease classification, and medical dictionaries. Organizations such as the World Health Organization (WHO), the US National Library of Medicine, and the US Centers for Disease Control and Prevention maintain and publish these taxonomies. In addition, taxonomies and processes for disaster recovery have been developed [10][29]. Ontologies used for Situation Management and BDI-SM agents should directly follow the classification models and vocabulary used today by disaster relief personnel. Such ontologies have an extensive set of concepts and are large and complex. A detailed discussion of disaster relief ontologies is beyond the scope of this paper.

## 6. PLATFORM AND AGENT ARCHITECTURE

### 6.1 Integration considerations

Many agent platforms have supported the BDI model, including PRS [38], Open-PRS [18], UM-PRS [24], JACK [16], JAM [17], Jason [22] and Jadex [2]. The integration of situation management with a BDI agent platform should enhance scalability of the event processing capability of the platform and provide a representation match between

the belief representation used by the BDI agents and the situation representation provided by the situation manager.

## 6.2 Multiple agent platforms

In the application domains of interest, various agent platforms will be deployed and should interoperate. As an example, in disaster recovery multiple teams from various jurisdictions and agencies will enter the disaster area with their staff, equipment, and supplies. These teams would typically have sensing and analysis equipment with semantic level functions enabled by agent platforms. However it is unlikely that all jurisdictions and agencies will use a common agent platform. Instead, there should be mechanisms by which different agent platforms interoperate.

FIPA (Foundation for Intelligent Agents) [13] provides interoperability between agent platforms and a directory mechanism by which agent platforms can discover other agent services. Important features of FIPA specifications include 1) a generic message transport by which FIPA-compliant agent platforms (AP) connect, 2) ability for nomadic agents to adapt to changing network conditions using monitor and control agents, 3) a formal agent

communication language based on a set of defined communicative acts and a method by which agents can establish a common ontology for communication. In addition, FIPA defines an experimental specification for an ontology service by which agents can share an ontology and translate between different ontologies. Note that FIPA does not currently specify ontologies for application domains.

In addition to the FIPA interoperability mechanisms, it is necessary that agents in different agent platforms be able to share situations and events. Continuing the example from disaster recovery, mobile teams in the field may have separate BDI-SM agent platforms. Each team may want to coordinate with other teams, and also communicate to control centers which provide overall direction. To meet this requirement, the situation management function can chain to other situation management instances on other platforms (Figure 8). Message interoperability between different BDI-SM agent platforms can be enabled by the FIPA mechanisms described previously. The specific types of situations and events that are propagated could be determined through negotiation between the respective sets of agents on each platform.

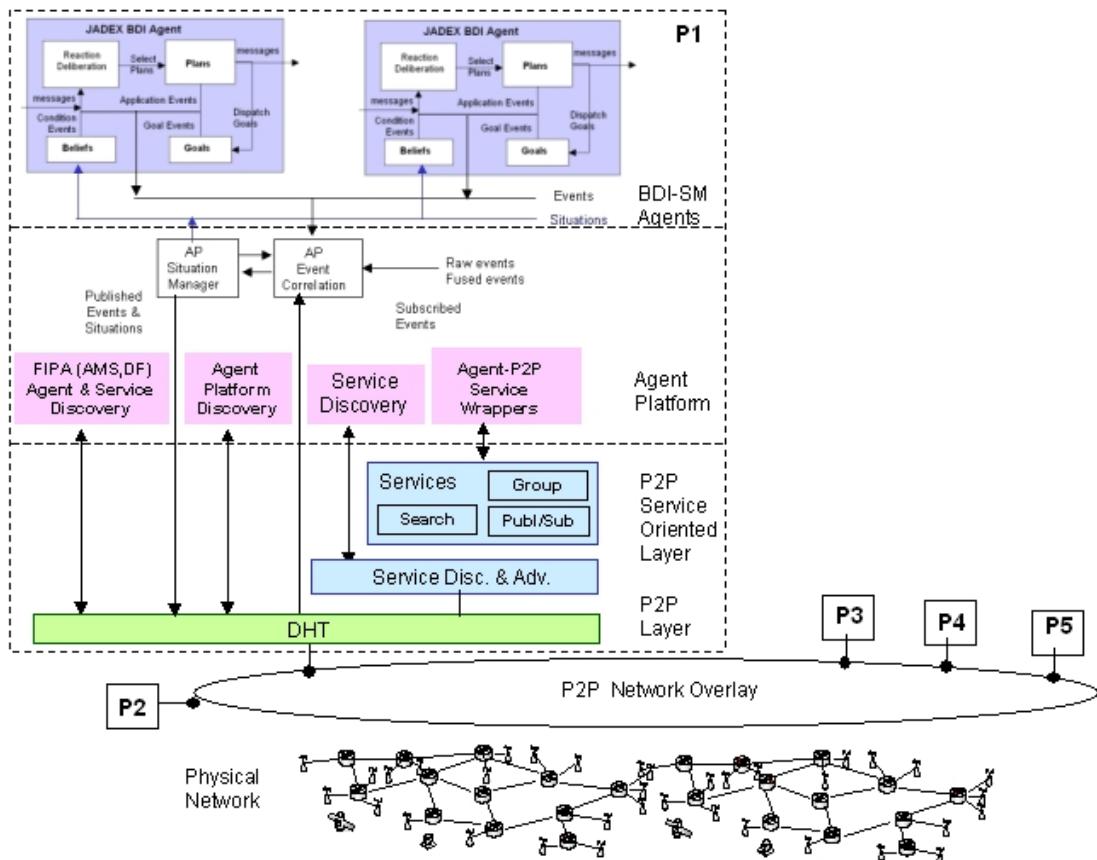


Fig. 8. BDI-SM agent platform integrated with peer-to-peer overlay

## 7. SCALABILITY THROUGH COUPLING TO LARGE-SCALE PEER-TO-PEER OVERLAYS

In large scale disaster recovery, we expect many relief personnel and their equipment to be agent-enabled. The population of relief personnel will vary widely during the relief operation due to ramp-up time, down-time, and changing needs during different stages of the relief operation. Further there will be many jurisdictions cooperating in the operations efforts, each collecting real-time events and situations and performing command and control for its jurisdiction.

These characteristics of the information distribution, equipment connectivity and scale of operation are well met by a peer-to-peer architecture. The common characteristics of peer-to-peer systems include 1) elimination of dependence on servers with their inherent scalability issues, 2) self-organization, 3) adaptability to changing population of peers, 4) heterogeneous peer populations. There are many designs for peer-to-peer overlays (see [27] for a recent survey) and several commercial deployments which scale to millions of simultaneously connected peers.

Thus there has been growing interest in coupling agent platforms with peer-to-peer overlays. In [7] we describe the coupling of FIPA-compatible agent platforms with peer-to-peer service overlay. In this design, the agent platform retains its functionality for semantic level discovery, while the peer-to-peer overlay provides keyword search, self-organization, and scalability. Here we focus on how BDI-SM agents on multiple platforms discover, subscribe to situations developing on different platforms, and share events. As shown in Figure 8, there are four layers at each peer:

- Peer-to-Peer (P2P) Layer: Connects each networked device to the P2P overlay network using a DHT-type overlay.
- P2P Service Oriented layer: Implements service discovery and advertisement mechanisms through the P2P overlay so that agents and the agent platform can advertise and discover other agent platforms and their agents.
- Agent Platform: A set of common services used by all agents on that platform, including FIPA-compliant services such as agent management and agent directories, and the situation manager and event correlation functions needed for situation management.
- BDI-SM agents: Through integration with the situation manager and event correlation function, implement the agent behavior for the disaster situation management

BDI-SM agents on a single agent platform discover and associate with each other using the agent platform directory services. The event correlation function processes incoming events from directly connected fusion layer and local event sources. Each BDI-SM agent can push events to the event correlation function. Situations are managed by the situation manager (SM) and each BDI-SM can

directly access situations maintained by that platform's situation manager.

Each agent platform is integrated with the peer-to-peer (P2P) Distributed Hash Table (DHT) and overlay routing algorithm. The DHT is a generic indexing mechanism on top of which a variety of discovery and search methods can be built. This permits the following type of search activities between different agent platforms:

- The arrival and departure of agents and agent platforms can be detected
- Agent services can be discovered by agents on other platforms
- A situation manager can advertise situation instances and situation meta information
- An event correlator can advertise event types and event history information

In addition, an agent, situation manager and event correlator can use the P2P publish/subscribe mechanism to be dynamically informed of new agents, situations, and events of interest.

In summary, the P2P overlay provides a generic message routing and indexing mechanism. A variety of DHT overlays have been shown to have highly scalable indexing behavior, and can maintain operation under a significant degree of churn. As argued by Buford and Burg in [7], this complements agent platform architectures which provide semantically rich discovery and communication mechanisms without large scalability and ability to operate in dynamic network environments. This is particularly important for DSM which are characterized by large scale, multi-domain operations with dynamic teams and infrastructure.

A different approach based on the Java virtual machine implementation is taken in the Cougaar system [15], which is based an open scalable architecture and has experimented with a thousand agent nodes. However, Cougaar is not FIPA compatible, and this requires all agents to be based on Cougaar.

## 8. RELATED WORK

The application of MAS technology to DSM has been the subject of a growing amount of research. An automated planner for coordinating the actions of multiple agents engaged in emergency response is described in [32]. The proposed method of inter-agent emergency response coordination is an effective solution for relatively simple coordination tasks, which could be described as a list of pre-defined actions. A MAS in which multiple agents explore sections of a damaged building with the goal of updating a topological map of the building with safe routes is discussed in [39]. The proposed approach is an interesting one, however is more suited for environments, which are static during the emergency operations. A MAS using web services is discussed in [34], in which the goal of the MAS is to detect possible emergencies and to suggest possible courses of action. The convergence of two disaster management systems is described in [42], wherein

the first system facilitates the surveillance of a remote geographical area and the second system provides a model of infectious diseases. Seen as a large-grain MAS, the two systems collaborate to infer potential outbreaks of infectious disease and appropriate response strategies. In [23], research is described towards building more robust and resilient multi-agent systems for future applications in disaster management.

Several recent experimental systems use the elements of situation awareness and information fusion for implementing the functions of disaster situation management. Llinas [26] describes a research agenda for using the methods of cognitive information fusion for building intelligent systems for man made and natural disasters management, while Scott and Rogova [35] present a concrete architecture of those systems. Both papers are focused on the tasks of emergency situation analysis with less emphasis on situation action planning and control. Smirnov et al. [36] describe an interesting solution for building a knowledge-driven evacuation operation system, where the chief result is based on introduction of an application ontology. The methods of recognition of complex situations are based on a logical constraint satisfaction algorithms. The method is an effective one when there are a relatively small number of constraints. A method which uses domain ontology and analogical Case-Based Reasoning in urban transportation situation and threat monitoring has been described by the authors in [5].

## 9. CONCLUSION

We have addressed two important challenges in developing automated disaster relief operations: distributed event-driven cognitive processing and scalability to thousands of interacting agent platforms. To address existing limitations of event-driven cognitive processing, we extended the Belief-Desire-Intention model with the capability of situation awareness. We described how the key functions of event collection, situation identification, and situation assessment are implemented in MAS architecture suitable to the characteristics of large-scale disaster recovery. To address the scalability limitations of agent systems, we described an integrated agent platform with peer-to-peer overlay which combines the semantic discovery mechanisms of agent systems with scalability to millions of nodes of peer-to-peer overlays.

This paper made the following contributions:

- We positioned the problem of disaster relief operations within the paradigm of Situation Management
  - We summarized the formal properties of Situation Management for disaster recovery
  - We described BDI-SM, our modified Belief-Desire-Intention (BDI) Situation Management (SM) agent model which achieves situation management at the agent and agent system level, that is, achieves the reactive cognitive agent model that we argue is required for DSM.
  - We described the coupling of large-scale peer-to-peer overlays with BDI-SM agent systems to address the scalability requirement of large-scale disaster relief operations.
- Although it is an important problem in DSM, the issues of how situation assessment reflects on knowledge and events about the state of its own data collection infrastructure, as might be impacted in a disaster, is not discussed in this paper (for details about reflective situation management see [9]). Further, the situation assessment function is assumed to use techniques for reasoning with incomplete information characteristic of this type of environment. These techniques permit incomplete and possibly inconsistent situations with different probabilities and event support to be maintained and changed as new information arrives.

## ACKNOWLEDGEMENT

The authors would like to thank the comments provided by the anonymous reviewers and editor, which helped the authors improve this paper significantly.

## REFERENCES

- [1] Barwise, J. *The Situation in Logic, CSLI Lecture Notes, Number 17*, Leland Stanford Junior University. 1989.
- [2] Bellifemine, F., Caire, G., Poggi, A., Rimassa, G. Jade: A Whitepaper. *Exp. v. 3, n. 3*. 2003.
- [3] Bratman, M. *Intension, Plans, and Practical Reason*. Harvard University Press. 1987.
- [4] Buford, J., Jakobson, G., Lewis, L. "Extending BDI Multi-Agent Systems with Situation Management." *The Ninth International Conference on Information Fusion*. Florence, Italy, 2006.
- [5] Buford, J., Jakobson, G., Lewis, L. "Case Study of Urban Transportation Threat Monitoring Using the AESOP Situation Manager™". *2005 IEEE Technologies for Homeland Security Conf*. Boston, MA, 2005.
- [6] Buford, J., Jakobson, G., Lewis, L., Parameswaran, N., Ray, P. "D-AESOP: A Situation-Aware BDI Agent System for Disaster Situation Management". *First Intl. Workshop on Agent Technology for Disaster Management (ATDM)* May 2006, Hakodate, Japan.
- [7] Buford, J., Burg, B. "Using FIPA Agents with Service Oriented Peer-to-Peer Middleware". *Proceedings Mobile Services and Ontologies Workshop (MOSO 2006)*. Nara, Japan, May 2006.
- [8] Buford, J., Brown, A., Kolberg, M. "Parallelizing Peer-to-Peer Overlays with Multi-Destination Routing". *IEEE Consumer Communications and Networking Conf. (CCNC 2007)*. Jan. 2007.
- [9] Buford, J., Jakobson, G., Lewis, L. "Reflective Situation Management". *Workshop on Situation Management (SIMA 2006)*. MILCOM 2006. October 2006.
- [10] Cahill, C. (Ed.) *Emergency Relief Operations*. Fordham University Press, 2003.
- [11] CLIPS 6.2. <http://www.ghg.net/clips/CLIPS.html>
- [12] d'Inverno, M., Luck, M., Georgeff, M., Kinny, D., Wooldridge, M. "The dMARS Architecture: A Specification of the Distributed Multi-Agent Reasoning System", *Journal of Autonomous Agents and Multi-Agent Systems*, 9(1-2):5-53, 2004.
- [13] Endsley M. R. "Theoretical Underpinnings on Situation Awareness: A Critical Review". in Endsley M. R. and Garland D. J., (eds), *Situation Awareness Analysis and Measurement*, pages 3-32. 2000.
- [14] FIPA. FIPA Abstract Architecture Specification. SC00001L, 2003.
- [15] Helsingr, A., Thome, M., Wright, T. "A Scalable, Distributed Multi-Agent Architecture". *Proc. of the 2004 IEEE Int'l. Conf. on Systems, Man & Cybernetics*, The Hague, Netherlands, 10-13 Oct. 2004.

- [16] Howden, N., Rönnquist, R., Hodgson, A., Lucas, A. "JACK - Summary of an Agent Infrastructure". *5th International Conference on Autonomous Agents*, 2001.
- [17] Huber, M. J. "JAM: A BDI-theoretic Mobile Agent Architecture". *Proc. of the Third Intl. Conf. on Autonomous Agents (Agents'99)*. Seattle, WA. May, 1999. pp 236-243.
- [18] Ingrand, F. F., Chatila, R., Alami, R., Robert, F. "PRS: A High Level Supervision and Control Language for Autonomous Mobile Robots". *IEEE ICRA 96*.
- [19] Jakobson, G., Buford, J., Lewis, L. "Towards an Architecture for Reasoning About Complex Event-Based Dynamic Situations", *Intl. Workshop on Distributed Event based Systems (DEBS'04)*, Edinburgh, UK, 2004.
- [20] Jakobson, G., Buford, J., Lewis, L. "A Framework of Cognitive Situation Modeling and Recognition". *Workshop on Situation Management (SIMA 2006)*. MILCOM 2006. Oct. 2006.
- [21] Jakobson, G., Weissman, M. "Real-Time Telecommunication Network Management: Extending Event Correlation with Temporal Constraints". *Integrated Network Management IV*, IEEE Press, 1995.
- [22] Jason. <http://jason.sourceforge.net/>
- [23] Jennings, N., S. Ramchurn, M. Allen-Williams, R. Dash, P. Dutta, A. Rogers, Vetsikas, I. "The ALADDIN Project: Agent Technology To The Rescue". *Proceedings of the First Intl. Workshop on Agent Technology for Disaster Management*, in conjunction with the Fifth Intl. Joint Conf. on Autonomous Agents and Multi-agent Systems.
- [24] Lee, J., Huber, M., Durfee, E., Kenny, K. "UM-PRS: An Implementation of the Procedural Reasoning System for Multi-Robot Applications", in *Conference on Intelligent Robotics in Field, Factory, Service and Space*, MIT Press, pp.842-849, 1994.
- [25] Lewis, L. *Managing Computer Networks: A Case-Based Reasoning Approach*. Artech House, 1995.
- [26] Llinas, J. "Information Fusion for Natural and Man-made Disasters". In *Proc. of the 5<sup>th</sup> Intl. Conf. on Information Fusion*, Sunnyvale CA, 2002, pp. 570-574.
- [27] Lua, E. K., Crowcroft, J., Pias, M., Sharma, R., Lim, S. "A Survey and Comparison of Peer-to-Peer Overlay Network Schemes". *IEEE Communications Surveys and Tutorials*, Second Quarter, Vol. 7, No. 2, 2005.
- [28] McCarthy, J., and Hayes, P. "Some Philosophical Problems from the Standpoint of Artificial Intelligence". In D. Michie, editor, *Machine Intelligence 4*, American Elsevier, New York, NY, 1969.
- [29] Moore, T., Lakha, R. (Eds) *Tolley's Handbook of Disaster and Emergency Management, Third Edition: Principles and Practice*. Butterworth-Heinemann, 2006.
- [30] Pavón, J., Corchado E., Castillo L. F. "Development of CBR-BDI Agents: A Tourist Guide Application". in *7<sup>th</sup> European Conf. on Case-based Reasoning*, Funk P. y González Calero P. A. (Eds.) *Lecture Notes in Computer Science, Lecture Notes in Artificial Intelligence (LNAI 3155)*, Springer Verlag, 2004, pp. 547-55.
- [31] Pirri, F., Reiter, R. "Some Contributions to the Situation Calculus". *J. ACM*, 1999, 46(3): 325-364.
- [32] Potter, S., A. Tate, Wickler, G. "Using I-X Process Panels as Intelligent To-Do Lists for Agent Coordination in Emergency Response" *Proc. of the Third Intl. Conf. on Inf. Syst. for Crisis Response and Mgmt.* (Eds. B. Van de Walle and M. Turoff), 2006.
- [33] Rao, A., Georgeff, M. "BDI Agents: From Theory to Practice". *Proc. of the First Intl Conf. on Multiagent Systems (ICMAS'95)*, 1995.
- [34] Schoenharl, T., G. Madey, G. Szabo, Barabasi, A. "WIPER: A Multi-Agent System for Emergency Response". *Proc. of the Third Intl. Conf. on Inf. Systems for Crisis Response and Management* (Eds. B. Van de Walle and M. Turoff), 630 pages, 2006.
- [35] Scott, P. Rogova, G. "Crisis Management in a Data Fusion Synthetic Task Environment", *7<sup>th</sup> Conference on Multisource Information Fusion*, Stockholm, 2004.
- [36] Smirnov, A. et al. "KSNET-Approach Application to Knowledge-Driven Evacuation Operation Management", *SIMA 2005*, 2005
- [37] So R., Sonnenberg L. "Situation Awareness in Intelligent Agents: Foundation for a Theory of Proactive Agent Behavior". *Proc. of the IEE/WIC/ACM Intl. Conf. on Intelligent Agent Technology (IAT'04)*, 2004.
- [38] SRI International. Menlo Park, California. *Procedure Reasoning System User Guide*. 1992.
- [39] Tatomir, B., Rothkrantz, L., Popa, "M. Intelligent system for exploring dynamic crisis environments". *Proc. of the Third Intl. Conf. on Information Systems for Crisis Response and Management* (Eds. B. Van de Walle and M. Turoff), 2006.
- [40] Urlings P., Tweedale J., Sioutis C., Ichalkaranje N. "Intelligent Agents and Situation Awareness". *Lecture Notes in Computer Science*, Vol. 2774, Springer Berlin/Heidelberg, 2003, pp. 723-733.
- [41] Wooldridge, M. *An Introduction to Multi-Agent Systems*. John Wiley and Sons, 2002.
- [42] Yeargins, D., Noseworthy, T., Hamilton, D., Denzinger, J. *Proc. of the First Intl. Workshop on Agent Technology for Disaster Management*, in conjunction with the Fifth Intl. Joint Conf. on Autonomous Agents and Multi-agent Systems. 2006.



**John F. Buford** received the BS and MS degrees from MIT and the PhD in Computer Science from Technische Universitaet Graz, Austria. He is President of Altusys Corp. Previously he was VP of Software Development at Kada Systems, Director of Internet Technologies at Verizon, and Chief Architect-OSS at GTE Laboratories. Earlier he was tenured Associate Professor of Computer Science at the University of Massachusetts Lowell, where he also directed the Distributed Multimedia Systems Laboratory. He has authored or co-authored eighty refereed publications and the book Multimedia Systems. He is an IEEE Senior Member and co-chair of the IRTF Scalable Adaptive Multicast Research Group.



**Gabriel Jakobson** has the PhD in Computer Science from the Estonian Academy of Science. He is Chief Scientist of Altusys Corp. and previously was a research scientist with Verizon and GTE Laboratories for more than 20 years. He has authored more than 70 technical papers in the area of semantic modeling and knowledge representation, event correlation, intelligent agents, man-machine interfaces, expert systems, database management systems, logical systems design, abstract automata, and fuzzy reasoning. He has been active in organization and serving in the TPC of various conferences, including NOMS, IM, DSOM. He has been the TPC co-chair for EntNet @ SUPERCOM 2001-2004, EntNet TPC co-chair for 2002-2004, and SIMA 2005-2006. He is an IEEE Senior Member.



**Lundy Lewis** received the MS degree from RPI and the PhD from the University of Georgia. Dr. Lewis is chair of the Information Technology Dept at SNHU. Previously he was the Director of Research at Aprisma Management Technologies and Cabletron. He holds 20 US patents and has 20 patents pending. These patents include new techniques for alarm correlation, policy management, object filtering, and configuration management. He has published 3 books and more than 30 papers in the professional literature and frequently gives presentations and tutorials at IEEE/IFIP conferences.