Sense and Respond Systems for Crisis Management: The Event Web

Mani Chandy
Computer Science
California Institute of Technology

mani@cs.caltech.edu

Infospheres Group: Dan Zimmerman, Brian Aydemir, Elliott Karpilovsky, Andre Khorlin, Lu Tian, Jonathan White
iSpheres Corp.: Eric Bax, Greg Billock, Lee Rosen, Ian Swett
Outline

- Requirements for crisis management.
- Architecture of the Event Web
- Theory
- Prototype.
- Dagstuhl Workshop on Crisis Management: Feb 2004

- Work at Caltech and at iSpheres Corporation
- http://www.ispheres.com
The Hypotheses

1. Information technology for crisis management in different application spaces (financial, healthcare, natural disasters, terrorism) have a common set of problems.
The Hypotheses

1. Information technology for crisis management in different application spaces (financial, healthcare, natural disasters, terrorism) have a common set of problems.

2. A layer can be abstracted that represents these common problems.
The Hypotheses

1. Information technology for crisis management in different application spaces (financial, healthcare, natural disasters, terrorism) have a common set of problems.

2. A layer can be abstracted that represents these common problems.

3. **Computer science theory and implementation practices can be applied to this layer.**
The Hypotheses

1. Information technology for crisis management in different application spaces (financial, healthcare, natural disasters, terrorism) have a common set of problems.
2. A layer can be abstracted that represents these common problems.
3. Computer science theory and implementation practices can be applied to this layer.
4. **Software platforms for this abstraction can be tailored quickly to map to different crisis situations.**
We Cannot Plan for Every Contingency

•Crisis unfolds in unexpected ways.

Requirement: Configure sense and respond platform when crisis strikes; continuously reconfigure platform as crisis evolves.
Example

• Pilot radios from a plane that a passenger seems to have SARS symptoms.
• The plane crashes.
Dynamic Coalitions of Multiple Groups Manage Crises

- Groups that manage crises are changing collections of institutions and individuals.
- The public, connected through the Internet and wireless, plays an increasingly important role.

Requirement: Flexible dissemination network without central data but with dynamic security policies.
The response of an organization often depends on the history of a crisis, and not merely on its instantaneous state. Location plays an important role.

Relevance: respond to state trajectories and not merely to current state.
Example

- **Time:** Moving-point average of typhoid cases for 2 day window exceeds moving-point average over 1-month window, and, supply of antibiotics below threshold $T$.

- **Location – Geographical information systems:** Alert all vehicles moving towards a bridge.
Crisis Managers Need Attention Multipliers

Sense and Respond Systems for Crisis Management
3 November 2003, Agent Based Modeling and Simulation

Sense
Detect events across extended environment in real-time

Respond
Invoke distributed services in real-time

Analyze
Aggregate events across multiple sources

Friendly forces in area. I am under attack
Architecture

Source of raw data: e.g., sensor

Event processor:

Influence environment: Actuators

Event stream directory
Architecture

Event dissemination

Navy

Nassau County Hospital

TWA

CIA

Nassau County Police

Travel agent

John Q. Public

Event stream directory
Central idea of architecture of Event Web

- Compositional.
- Extensible component library.
- Components can be added, deleted, modified during a crisis.
- Components are loosely coupled.

- Communication between components: Events.
Outline

- Requirements for crisis management.
- Architecture of the Event Web
- Theory
- Prototype.
- Dagstuhl workshop on Crisis Management
Theory (time permitting)

Semantics of events

- Error analysis
- Abstraction:
  - The programming paradigm: when-then rules.
- Stepwise refinement
  - Global when-then rules to sets of sensors, actuators, and event processors with local when-then rules.
- Event notations
- Delta dataflow algorithm and chip design
- Incremental computation to process event streams.
- Optimization: intelligent sensors, capacity limitations,...
Types of events

An agent B generates an event to change other agents’ estimates of the state of the environment.

Temperature in Minneapolis

Model used by agent B of daily variation on Nov. 3
Events that changes estimates

An agent generates an event to change other agents’ estimates of the state of the environment.
Events that change estimates

Agent generates an event to change “knowledge” or “estimates” of other agents.
Events that change estimates

In this case, an event is encoded as a timestamp and a bit indicating whether the band was crossed above or below.
Types of events: periodic events

An agent samples values according to a specified sampling strategy.

The model deals with the sampling strategy.

Temperature in Minneapolis

Time
Errors due to delay

Receiving agents’ estimation of state when no signal is received.
Errors due to delay

Receiving agents’ estimation of state when no signal is received.
Errors

- Errors due to:
  - loss of events
  - delays
  - clock drift
  - erroneous monitoring of environment
  - erroneous computation
WHEN $f(\text{global\_state})$ THEN generate event

- World is a multidimensional system with a huge number of dimensions.
- Information about different dimensions represented by discrete events generated at different times.

Generate event when system enters this space

Danger point

- $t = 3$
- $t = 5$
- $t = 7$
- $t = 9$
The Event Web implements When-Then Sense and Respond Systems for Crisis Management.

Source of raw data, e.g., sensor

Event Dissemination Network

Sets of When-Then Rules

Event processors

Actuators
Stepwise Refinement

- Crisis manager specifies global when-then rules.
- Stepwise refinement to:
  - Sensors that generate raw events for the “when” predicate.
  - Event processors that execute local when-then rules.
  - Actuators that implement the final “then” action.
  - So that collectively the distributed system implements the global when-then rules.
Delta dataflow for executing when-then

Rajit Manohar, Cornell: building asynchronous VLSI chip for delta dataflow

Moving – point average

At Least 2

Avg > Threshold

2-min avg > 60 min avg

f
Notations for When – Then Rules

- Alternatives

- Same as query for a database: a rule is a persistent query.
  - SELECT s WHERE b

- General structure dealing with histories, geography, action sequences (e.g., time series, location sensitive, workflow representation)
Incremental Compilation: Map Tree to Existing Node Structure on Some Machine

T > 0

P > 1
Incremental Computation

\[ f(x_1, \ldots, x_k, \ldots x_n) \]

Change in value \( x_k \)
Incremental Computation

New value: $f(x_1, \ldots, x_k', \ldots, x_n)$

Change in value $x_k'$
New value: \( f(x_1, \ldots, x_k', \ldots x_n) \)

Ideally, cost of computing \( f(x_1, \ldots, x_k', \ldots x_n) \) from \( f(x_1, \ldots, x_k, \ldots x_n) \) should be independent of \( n \).

Examples:
- Threshold functions
- Moving-point computations
- Boolean operators

Change in value \( x_k' \)
Outline

- Requirements for crisis management.
- Architecture of the Event Web
- Theory
  - Prototype.
- Dagstuhl workshop on Crisis Management
Prototype and product

- Goal: build a platform for implementing when-then rules.
- Show how global when-then rules can be refined automatically, or systematically by engineers, to obtain implementations.
Generic asynchronous node in Event Web

Sense and Respond Systems for Crisis Management
3 November 2003, Agent Based Modeling and Simulation

Message-triggered program

Timer

From event dissemination network

Event inboxes

To event dissemination network
Prototype and product

- **Event schema:**
  - Caltech Infospheres prototype: XML document
  - iSpheres Corp product: Serialized Java object.

- **Event dissemination**
  - Infospheres: explicit subscription to event generator by event consumer
  - iSpheres prototype: subscription deduced from high-level notation.

- **Event notation**
  - Infospheres: only provides compositional structure
  - iSpheres: explicit notation for parametric and time series analysis.
Processing an Incoming Message

Incoming message

- Current state
  (XML doc)
Processing an Incoming Message

1. Incoming message
2. Choose XSLT transformation based on incoming message and current state
3. Current state (XML doc)
4. New state (XML doc)
Processing an Incoming Message

Incoming message

Current state
(XML doc)

Choose XSLT transformation based on incoming message and current state

Sequence of outgoing Messages (XML docs)

New state
(XML doc)
Architecture of event processor

Incoming events

Data structures shared by event queues and when-then rules engine

Rules execution
Related Work

- Events: David Luckham, iSpheres patent apps
- Content-based subscription: Gryphon, Sienna
- Real Time Business Intelligence
- Real Time Business Rules
- Active databases
- Gartner: Roy Schulte on Event-Driven Architecture

- Download the iSpheres Sense and Respond platform from http://www.ispheres.com
Outline

- Requirements for crisis management.
- Architecture of the Event Web
- Theory
- Prototype.
- **Dagstuhl workshop on Crisis Management**

- Work at Caltech and at iSpheres Corporation
- Download product: http://www.ispheres.com