Data Sharing Middleware Prototype (DSMP) for Information Dissemination Among Heterogeneous Sources

Quarterly Review Meeting, May 1, 2008

Hairong Qi (PI), University of Tennessee
Xiaorui Wang (co-PI), Seddik Djouadi (co-PI), UT
Oak Ridge National Laboratory*
Oracle Corporation*
Microsoft Research*
Rutherford Appleton Laboratory, UK*

* Oracle, Microsoft Research, and ORNL commitments for in-kind support (consulting and research software)
Contact Information

• Academia
  – Hairong Qi, 865-974-8527, hqi@utk.edu, 1508 Middle Dr., 319 Ferris Hall, EECS Department, University of Tennessee, Knoxville, TN 37996
  – Xiaorui Wang, 865-974-0627, xwang@ece.utk.edu, 421 Ferris Hall, UT
  – Seddik Djouadi, 865-974-5447, djouadi@ece.utk.edu, 307 Ferris Hall, UT
  – Raghul Gunasekaran, 865-385-5857, raghul@utk.edu, 536 SERF, UT
  – Ming Chen, Ying Sun, Samir Sahyoun, UT Graduate Students

• Research Laboratories
  – Frank DeNap, 865-576-8786, denapfa@ornl.gov, Oak Ridge National Laboratory, PO Box 2008, MS6085, Oak Ridge, TN 37831
  – Mallikarjun Shankar, 865-574-2704, shankarm@ornl.gov, Oak Ridge National Laboratory, PO Box 2008, MS6085, Oak Ridge, TN 37831
  – Steve Fisher, RAL, s.m.fisher@rl.ac.uk, Rutherford Appleton Laboratory (RAL), UK

• Industry, Private sectors
  – Dieter Gawlick, Ronny Fehling, Aravind Yalamanchi, 650-560-8706, {dieter.gawlick, ronny.fehling, aravind.yalamanchi}@oracle.com, Oracle Corporation
  – Vijay Dialani, Microsoft Research Center
Collaborative Team

- Academia
  - University of Tennessee
  - Vanderbilt
- Research Laboratory
  - ORNL (Oak Ridge National Laboratory, US)
  - RAL (Rutherford Appleton Laboratory, UK)
- Industry
  - Microsoft
  - Oracle
UT Participating Members

- Research associate
  - Raghul Gunasekarn

- Students
  - Ming Chen (Ph.D.)
  - Ying Sun (Ph.D.)
  - Samir Sahyoun (Ph.D.)
  - Ben Taylor (M.S.)

- Faculty
  - Hairong Qi: Collaborative information processing in distributed environment
  - Xiaorui (Ray) Wang: Middleware development
  - Seddik Djouadi: Optimal control
Project Description

• The objective of this project is to develop a data sharing middleware that is able to handle multiple distributed data sources and dynamically changing items, and to assist in real-time information dissemination across multiple agencies for homeland security purposes.

• The ultimate target scenarios are first responders and consequence response at the urban area of Memphis (e.g., Shelby County) with stakeholders including the Fire Department, Weather Services, the E911 Operations Center, etc.
Tasks

• Task 1: The design, development, and evaluation of a data sharing middleware prototype based on the publish-subscribe paradigm for wide-area information dissemination.
  – Task 1.1: Background…
  – Task 1.2: Implementation…
  – Task 1.3: Performance analysis…
• Task 2: The development of three application scenarios to demonstrate the effectiveness of the data sharing middleware.
  – Task 2.1: Event correlation…
  – Task 2.2: Mobile and control aspects …
  – Task 2.3: Fault tolerant collaborative event analysis…
Phased Development

- **Phase 1** – simplest scenario with a known data vocabulary and a trivial subscription
  - 1 known data vocabulary
  - 1 Publisher service added to the Registry
  - 1 Consumer service added to the Registry
  - A simple subscription with just data constraints and no property constraints.
  - The WSN Notify interface is not implemented, but instead each software development group is free to use whatever in-house “notify” mechanism that they are most comfortable with (e.g. JMS).
  - Very simple data is published by the publisher.

- **Phase 2**
  - 2 Publisher services
  - 2 Consumer services
  - The addition of Property vocabularies and instances of these properties for the publishers and consumers.
  - A subscription which now contains some simple property constraints as well as data constraints

- **Phase 3**
  - Multiple data vocabularies.
  - Multiple Publisher services added to the Registry.
  - Multiple Consumer services added to the Registry.
  - Multiple Subscriber services added to the Registry.
  - More complex subscriptions with data constraints and property constraints.
  - More complex data published by the publishers.

- **Phase 4**
  - A standard notification interface such as WSN Notify.
## Schedule

<table>
<thead>
<tr>
<th>Tasks</th>
<th>06/07 – 05/08</th>
<th>06/08 – 05/09</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1 2 3 4</td>
<td>1 2 3 4</td>
</tr>
<tr>
<td>Task 1.1 – literature survey and document study</td>
<td>×</td>
<td></td>
</tr>
<tr>
<td>Task 1.2 – prototype design and development</td>
<td>× × × ×</td>
<td>× × × × × ×</td>
</tr>
<tr>
<td>Task 1.3 – middleware performance evaluation and refinement</td>
<td>×</td>
<td>× × × ×</td>
</tr>
<tr>
<td>Task 2.1 – application scenario 1 – establishing data correlation</td>
<td>× × ×</td>
<td></td>
</tr>
<tr>
<td>Task 2.2 – application scenario 2 – dynamic mobile platform itinerary determination</td>
<td>×</td>
<td>× × ×</td>
</tr>
<tr>
<td>Task 2.3 – application scenario 3 – collaborative event analysis</td>
<td>× ×</td>
<td></td>
</tr>
</tbody>
</table>
Achievements

- Identified first responder as the ultimate use case scenario for DHS interest
- Finished phase II of the prototype development of INFOD
- Finished simulation of collaborative event analysis
- Proposed a new plume model with promising simulation results
- Visited SCFC and discussed potential integration scenario
- New issues identified
  - Real-time guarantee of system performance
  - Security issues of INFOD
  - Collaboration with Vanderbilt on security issues and the usage of generic modeling environment (GME)
- Papers submitted
  - “Control-based real-time metadata matching for information dissemination”, to be submitted to *14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications*
- Presentations at two OGF meeting
- Semi-annual review in Sept. 2007
What’s Ahead?

• Semi-annual review meeting in Sept. 2008
  – Demonstration of the first responder user case
  – Demonstration of dynamic target classification
Project Content

- INFOD development and demo
  - Raghul
- Control-based real-time metadata matching for information dissemination
  - Ming Chen
- Plume modeling
  - Samir
- Dynamic target classification
  - Ying
INFOD Development and Demo

Raghul Gunasekaran
Traditional Information Dissemination models

- Push and/or Pull model
- Repository – Data center, processing center (alerting system)
- Extending the system – s/w development and time?
- Semantics for description and communication (alerting)?
- Defining subscription or rules?
The INFOD model

- Establishes a framework for information flow
- Subscriber defines subscriptions establishing information pipelines
- Information directed from publishers to consumers
- INFOD is not a data (event) repository
- What if I want to have a data (event) repository or alerting system?
Sharing information among a group with common interest

Example: Yahoo or Google groups
- One or more users create a group defining the objective and act as moderators
- Users with similar interest join the group
- New posts are received at user inbox
- Security: Moderator controls user access

• Advantages
  - Easy to establish (and also easy to tear down)
  - Broadcast or Unicast messages

• Disadvantages
  - Users cannot specify constraints – just subscribe to a logical data channel
  - Users can spam the group
Example Scenario: Tornado relief campaign

- **Need:** Monitor activities in the affected region for a short time period.
- Track first responders in the region – ambulances, police, fire service, doctors, etc …
- First responder needs to be aware of the resources available and contact information

**Requirements**

- A functional system instantly available that would cater to the current application
- Instant setup by domain experts
Example Scenario: Tornado relief campaign

Managed by UT-Battelle for the U.S. Department of Energy – Supporting the Department of Homeland Security
Example Scenario: Tornado relief campaign

Information dissemination in 4 steps

1. Register community property and data vocabularies in the INFOD registry

2. Define subscriptions binding entities, defining events and which entity needs to be alerted on which other entities presence.

3. Entities register to the INFOD registry
   - Create entries
   - Create and update property instances

4. Notification message sent to matched entities.
The INFOD Model

- Extensibility

Example: A community with medical services and Police. The fire service joins the service.

- Accept existing vocabulary definitions
- Or Register custom vocabularies

However in the second case the subscriptions would need to be defined in terms of specific community vocabularies and the consumer receives messages in publishers language.
The INFOD Model

Plug ‘n’ Play
The INFOD Model
Right Information to the Right Entity

Evacuate People
Divert Traffic
Evacuate @ Location:X
Location: X

Geographic distance might not be the shortest distance
Subscription defines the event of interest at the publisher, the message to be generated in response to an event and helps identify consumers dynamically.

Structured Information Model, user communities are identified by property and data vocabularies.

Entries characterize real world entities and define constraints identifying other entities of interest.

Also, data source entry details on the publishers information - associated to a data vocabulary.

Vocabulary instances characterize entities and constraints are evaluated on instances created.

INFOD Resources

Subscribers and Consumers are modeled as independent entities. Subscriptions are created by a subscribers, conforming to a subset of consumer.
First Responder Use Case Scenario

- APD 2000 Chemical Sensor 2
- State Control Center
  - INFOD Registry
- Weather Station
- Alerting System
  - Plume Analysis
- Subscriber/Consumer/Publisher
- E911 Center
- Subscriber/Consumer

INFOD Message Exchange

Information Notification
**Property Vocabulary**

### Sensor Property Vocabulary

<table>
<thead>
<tr>
<th>Property Vocabulary Predicates</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sensor Identification</td>
<td>Unique id for sensor</td>
</tr>
<tr>
<td>Sensor Description</td>
<td>Details on the sensor</td>
</tr>
<tr>
<td>Location</td>
<td>Physical location of sensor</td>
</tr>
<tr>
<td>Contact Information</td>
<td>Person to be contacted</td>
</tr>
<tr>
<td>Associated Organization</td>
<td>The owner of the sensor</td>
</tr>
</tbody>
</table>

### Community User Property Vocabulary

<table>
<thead>
<tr>
<th>Property Vocabulary Predicates</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identification</td>
<td>Identifier for the Organization</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the Organization</td>
</tr>
<tr>
<td>Description</td>
<td>Text Description</td>
</tr>
<tr>
<td>Location</td>
<td>Physical location information</td>
</tr>
<tr>
<td>Contact Information</td>
<td>Person to me contacted</td>
</tr>
</tbody>
</table>

Vocabulary predicates are an abstraction from NIEM (National Information Exchange Model).
# Data Vocabulary

## Sensor/Alert Data Vocabulary (NIEM and CAP - Common Alerting Protocol)

<table>
<thead>
<tr>
<th>Data Vocabulary Predicates</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity</td>
<td>Description between time period</td>
</tr>
<tr>
<td>Event</td>
<td>Description at a specific time</td>
</tr>
<tr>
<td>Substance</td>
<td>Description of a chemical material</td>
</tr>
<tr>
<td>Status</td>
<td>Actual, Exercise, System, Test</td>
</tr>
<tr>
<td>Message Type</td>
<td>Alert, Update, Cancel, Ack, Error</td>
</tr>
<tr>
<td>Scope</td>
<td>Public, Restricted, Private,</td>
</tr>
<tr>
<td>Urgency</td>
<td>Immediate, Expected, Future, Past</td>
</tr>
<tr>
<td>Response Type</td>
<td>Shelter, Evacuate, Prepare, Execute, Monitor, Assess, None</td>
</tr>
<tr>
<td>Severity</td>
<td>Extreme, Severe, Moderate, Minor</td>
</tr>
<tr>
<td>Certainty</td>
<td>Very likely, Likely, Possible, Unlikely</td>
</tr>
<tr>
<td>Category</td>
<td>Geo, Met, Safety, Security, Rescue, Fire, Health, Env, Transport, Infra, CBRNE, Other</td>
</tr>
</tbody>
</table>

## Weather Sensor Data Vocabulary (DWML – Digital Weather Markup Language)

<table>
<thead>
<tr>
<th>Data Vocabulary Predicates</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Weather information at a location</td>
</tr>
<tr>
<td>Time</td>
<td>Information at a specific date &amp; time</td>
</tr>
<tr>
<td>Temperature</td>
<td>Temperature unit as an attribute</td>
</tr>
<tr>
<td>Wind Speed</td>
<td>With units as an attribute</td>
</tr>
<tr>
<td>Wind Direction</td>
<td>As an angular measure</td>
</tr>
<tr>
<td>Humidity</td>
<td>As a relative or specific value</td>
</tr>
</tbody>
</table>
The Registry

- Property Vocabulary – schema for characterizing an external entity (Publisher/Consumer/Subscriber)
- Data Vocabulary – schema for characterizing an event or information from the publisher
- Publisher – source of information
  - Publisher Entry
  - Property Vocabulary Instance
  - Data Source Entry
    - Property Vocabulary Instance
The Registry

- **Consumer**
  - Consumer Entry
  - Property Vocabulary Instance
- **Subscriber**
  - Subscriber Entry
  - Property Vocabulary Instance
  - Subscription
- An update to the Registry triggers matching between entries.
- Entities receive notification message
- Publisher sends information to consumer
Constraints

Constraints are for expressing entities and data of interest.

• Property Constraint
  – Used for matching within the INFOD registry
  – Identifies other entries of interest defining constraints on entry properties.

• Data Constraint
  – Part of the subscription; evaluated by the publisher
  – Describes an event of interest and the message to be generated by the publisher

• Dynamic Consumer constraint
  – Part of the subscription; evaluated by the publisher
  – Identifies consumers based on certain characteristics from a set of consumer identified by the INFOD registry.
First Responder Use Case Scenario

INFOD Registry

- Publisher Entry
- Subscriber Entry
- Subscription
- Consumer Entry
- Data Source Entry
- Data Vocabulary
- Property Vocabulary Instance
- Property Vocabulary

- Weather Station
- Alerting System
- Plume Analysis
- E911 Center
- First Responder
- First Responder
- First Responder
- E911 Center
- Chemical Sensor 1

- Resource – not an entry
- Entry
- Creation of resource
- Reference (EPR)
- Notification (by INFOD registry)
- Notification (by Publishers)
Implementation specifics

Publisher, Subscriber and Consumers are modeled as web services
DEMO
First Responder Use Case

Acknowledge: Ming Chen
Subscription – Data Constraint

<DataConstraint>
declare namespace $data = http://infod.firstresponder.net/AlertDataVocabulary;

let $msg1 := for $firstresponders
    where $data:AlertStatus = ‘Actual’ and
    $data:EventCategory = ‘CBRNE’
    $data:EventSeverity > ‘Moderate’
    return { $data, $data:Instruction = ‘Evacuate people in the region’ }

let $msg2 := for $firstresponders
    where $data:capAlertStatus = ‘Actual’
    $data:EventCategory = ‘Fire’
    $data:EventSeverity > ‘Moderate’
    return { $data:Substance, $data:Volume, $data:EventCategory }

let $msg3 := for $firstresponders
    where $data:capAlertStatus = ‘Actual’
    return $data:Instruction = ‘Be prepared for an emergency’

let $msg4 := for $firstresponders
    where $data:capAlertStatus = ‘Test’ and
    return $data:Instruction = ‘Just a test Message’

let msg5 := for $firstresponders
    where $data:capAlertStatus = ‘Actual’
    return <alertmessage>
    { if ( $data:Certainty eq “Very Likely”) then
        $data:Instruction = ‘Evacuate in the next 5 hours’
    if ( $data:Certainty eq “Possible”) then
        $data:Instruction = ‘No current threats in the region’
    } </alertmessage>
</DataConstraint>
Subscription – Dynamic Consumer Constraint

Data Constraints define the event and the message to be sent but whom to be sent is determined by the dynamic consumer constraints

```xml
<DynamicConsumerConstraints>
  for $firstresponders
    where $firstresponders//OrganizationSubUnitName="Police"
      return msg1
  for $firstresponders
    where $firstresponders//OrganizationSubUnitName="FireService"
      return msg2
  for $firstresponders
    where $firstresponders//OrganizationSubUnitName="RedCross"
      return msg5
</DynamicConsumerConstraints>
```
Subscriptions

A user friendly web based subscription definition based on the vocabularies.

Web App Example

work in progress……..
Matching Entities

- In generic pub/sub models – subscription binds publishers to consumers of information.
- In INFOD, apart from the subscription, every entity (publisher, data source, consumer and subscriber) can define constraints that grant or limit information flow.
Matching Entities

• Increased complexity – multi constraint matching
• Worst Case response
  – 100 publishers, 20 consumers, 10 subscriptions – response time is 4.5 seconds
• Work in progress…
  – Improve the matching algorithm – reduce response time
  – Oracle – Expression Filters – a tool for evaluating XPATH expressions against an XML doc – an inbuilt feature in Oracle database
  – Work in grouping constraints to limit the number of evaluations
  – Pre compute and use materialized views (previous match results) in determining new associations based on the updates in the registry.
INFOD - Security Considerations

• Authentication
  – Users need to establish identity with the registry
  – Publishers need to authenticate themselves to the consumer
  – Web Service Security Specification
    • Establish Tokens for users
    • Authenticate users
    • Encrypt messages

• Authorization
  – Access Control policies
  – Limit the operational access and resource access of users with the registry
  – Limit publisher and consumer mapping
INFOD - Security Considerations

- RBAC – Role Based Access Control policy
- XACML (XML Access Control Markup Language) standards
- INFOD users are associated with Roles
- Roles determine
  - Create/replace/drop privileges within the registry
  - What information a user can access from the registry
  - Does not limit the association between entities
INFOD - Security Considerations

Security constraints limiting or granting data exchange between entities matched through subscription

P1
P2
C1
C2
C3
C4
C5
Subscription Evaluation

C1
C2
C3
C4
C5
C6

P1
P2
C1
C2
C3
C4
C5
C6
Security constraint
- GME is a meta-programmable toolkit for creating domain-specific modeling environments.
- GME models take the form of graphical, multi-aspect, attributed entity-relationship diagrams.
- Their syntax is defined by the meta-models specified in a UML class diagram-based notation.
- The dynamic semantics are applied by the model interpreters, i.e. by the process of translating the models to source code, configuration files, database schema or any other artifact the given application domain calls for.
• Two Aspect: Registration, Data
• Basic Entities: Registry, Publishers/Consumers
• Relationships: Subscriptions, Registration, Data Communication and non-INFOD Communication
Architectural Model / Vocabularies

- Inside of basic Entities
- Data Sources/Consumers
- PropertyVocabulary
- Instance
- DataVocabulary
- References
XML Schemas for Vocabularies

- Typical standards used to describe vocabularies in INFOD
  - NIEM (http://niem.gov/niem)
  - WSA addressing (http://www.w3.org/2005/08/addressing)
  - Weather (http://www.weather.gov/forecasts/xml)

- Application specific Schemas in Vocabularies
  - May extend/include other Schemas

- Usage difficulties
  - Complex
  - Visually not appealing
  - Reference following is almost impossible
Workflow

1. Import Language Libraries (Schemas)
2. Define Vocabularies using Libraries and extensions
3. Create the Entities and Connections in the Architecture
4. Associate Entities with Vocabularies both Data and Property
   - Generate Registration Messages
   - Generate API code
Control-based Real-Time Metadata Matching for Information Dissemination

Ming Chen, Xiaorui Wang, Raghul Gunasekaran, Hairong Qi,

Mallikarjun Shankar

University of Tennessee

Oak Ridge National Lab
Motivations

- Hundreds even thousands of subscriptions are registered;
- Different subscriptions have different priorities;
- Updates arrive with unpredictable intervals;
- **Valuable Information at the Right Time (VIRT).**
Motivations (cont’d-2)

Challenges:

- The execution time of reevaluating a subscription may vary significantly;

- Reevaluating all subscriptions may cause severe system overload and unacceptable long delays;

- We can NOT accurately calculate the time reevaluations take to find matching results;

- We can NOT trigger all subscription reevaluations upon each update;
Motivations (cont’d-3)

• Solutions
  A control-based architecture is designed.

• Goals
  – Primary Goal
    Average response time of subscription reevaluation meets with the soft deadline.
  – Secondary Goal
    Maximize the number of low-priority subscription reevaluation upon each update.
Outline

• Control-based Architecture;
• Controller Design;
• Performance Analysis;
• Experiments
• Related work
• Conclusions
Control-based Architecture

Why control-based?

- **Standard approaches** to choose the control parameters so that exhaustive tuning and testing are avoided;

- **Theoretical guaranteed control performance** such as accuracy, stability, settling time and overshoot;

- **Quantitative control analysis** when the system is suffering unpredictable workload variations.
Control-based Architecture (cont’d)

$R_{\text{ref}}$ Reference for average response time;

$r(k)$ Measured average response time in the $k$-th control period;

$n(k)$ Job budget in the $k$-th control period;
Control-based Architecture (cont’d)

Reference for average response time;

Measured average response time in the k-th control period;

Job budget in the k-th control period;
Control-based Architecture (cont’d)

$R_{ref}$ Reference for average response time;

$r(k)$ Measured average response time in the k-th control period;

$n(k)$ Job budget in the k-th control period;
Control-based Architecture (cont’d)

$R_{\text{ref}}$ Reference for average response time;

$r(k)$ Measured average response time in the $k$-th control period;

$n(k)$ Job budget in the $k$-th control period;
Control-based Architecture (cont’d)

$R_{ref}$ Reference for average response time;

$r(k)$ Measured average response time in the k-th control period;

$n(k)$ Job budget in the k-th control period;
Controller Design

- System modeling
  - Job budgets mapped from white noise
  - Open-loop system
  - Average response time
  - System identification
  - $r(k) = b_1 n(k - 1)$

- Model verification

Response time (s) vs. Time (s) graph with Actual output, Model 1, and Model 4.
Controller Design (cont’d)

- **Root-Locus design**
  - Stability;
  - Zero steady state error;
  - Shortest settling time.

- **Proportional-Integral controller**

  \[ F(z) = \frac{K_1(z - K_2)}{z - 1} \]

- **Feedback loop transfer function**

  \[ G(z) = \frac{F(z)P(z)}{1 + F(z)P(z)} = \frac{1}{z} \]
Performance Analysis

- Job budgets Vs Average response time

\[ r(k) = b_1 n(k - 1) \]

The slope

- Causes for variation
  - Different attributes and constraints involved;
  - Variation of the number of consumers and publishers at runtime;
  - Different Hardware and software configurations of systems

\[ r(k) = b_1 n(k - 1) \]

Model variation
Performance Analysis (cont’d)

- **Stability**
  
  \[ 0 < g < 2 \]

- **Steady state error**
  
  \[
  \lim_{z \to 1} (z - 1)G(z)R_{\text{ref}}(z) = \lim_{z \to 1} \frac{gz}{z - (1 - g)} \]

- **Settling time**
  
  \[
  k \geq \frac{\ln 0.05}{\ln |1 - g|} \]
Experiments

- Test-bed setup
  - INFOD registry is implemented on Server 1;
  - Clients are implemented on Server 2.
Experiments (cont’d-2)  

PI Vs OPEN

- OPEN only works well when we have accurate knowledge of the average execution time of subscription reevaluation;

- PI controller can guarantee the average response time converges to the set point in spite of variations of average execution time.
Experiments (cont’d-3)  

**PI Vs Ad Hoc**

- **Control accuracy**
  - Ad Hoc needs to be tuned for different sets of subscriptions and working environments;
  - PI controller can provide response time guarantees in spite of variations.

- **System quality of service**
  - PI can reevaluate more low-priority subscriptions.
Experiments (cont’d-4) Control Performance with Variations

- Stability
  - As $g$ approaches to 2, the deviation starts to increase noticeably;
  - As $g > 2$, the controller becomes unstable.

- Settling time
  - The experimental results are very close to the theoretical results;
  - PI controller features quantitative performance analysis.
Related Work

• Real-time information dissemination systems
  – Real-time broadcasting in wireless networks;
  – Asymmetric data links;

• Real-time databases
  – Periodic updates and aperiodic user transactions;
  – Trade off immediate updating to guarantee time constraints for user transactions;

• Control theoretic approaches
  – Real-time scheduling algorithms;
  – Resource management in distributed computing;
  – Web servers, power management.
Conclusions

- A feedback controller is designed and analyzed;
- The average response time converges to the set point in spite of workload variations;
- Better control accuracy and system quality of service than two baselines;
- Theoretical analysis with extensive experiments on a physical test-bed verifies our conclusions.
INFO-D Application Scenario

III

Chemical Plume Tracking

Seddik M. Djouadi
Samir S. Sahyoun
Problem

ABC Chemicals

Fire Station

Evacuation Route

Service Providers

Hospital

Weather Station

Fire Station

County Office

Service Providers

Police

Managed by UT-Battelle for the U.S. Department of Energy – Supporting the Department of Homeland Security
Contents

• Types of Plumes
• Mathematical Modeling of Plumes
• Source Localization
• Boundary Tracking and Prediction
• Mobile Sensors and Position control
Types of Plumes

- Buoyant Plumes
- Dense Gas Plumes
- Passive Plumes
Buoyant Plumes

• Lighter than air

• Lower density than air, because of either lower molecular weight or the emission happens at higher temperature.

• Example: a plume of methane gas at the surrounding air temperature, methane has lower molecular weight than the surrounding air.
Dense Gas Plumes

• Heavier than air.

• Higher density than air, because of either heavier molecular weight or it happens at much lower temperature than the surrounding air.

• Example: a plume of carbon dioxide at the surrounding temperature.
Passive Plumes

• Passive plumes have the same weight as air, they move according to the wind direction and velocity.

• Example: a plume of carbon dioxide at higher temperature.
Mathematical Model

- The concentration \( c \) at any point \( (x, y, z) \) in the plume at time \( t \) is the solution of the following PDE:

\[
\frac{\partial c}{\partial t} = \frac{\partial}{\partial y} D_y \frac{\partial c}{\partial y} + \frac{\partial}{\partial z} D_z \frac{\partial c}{\partial z} - v \frac{\partial c}{\partial x}
\]

- \( c \): concentration (kg/m³)
- \( v \): wind speed (m/s)
- \( D_y, D_z \): eddy diffusivities (m²/sec)
Example: Steady state Gaussian Plume
Example: Steady state Gaussian Plume

\[ c(x, y, z) = \frac{Q}{4\pi x \sqrt{D_y D_z}} \exp \left( -\frac{vy^2}{4xD_y} \right) \left[ \exp \left( -\frac{v(z - H)^2}{4xD_z} \right) + \exp \left( -\frac{v(z + H)^2}{4xD_z} \right) \right] \]

- \( c \): concentration (kg/m³)
- \( v \): wind speed (m/s)
- \( Q \): Mass emission rate (kg/sec)
- \( H \): Stack height (m)
- \( D_y, D_z \): eddy diffusivities (m²/s)
Source Localization

\[ z_{i,t} = \frac{Q}{r_i^\alpha} + w_{i,t} \]

where \( i = 1, \ldots, N; \ t = 1, \ldots, M \)

Measurement \( z_{i,t} \) is the \( t^{th} \) sample at sensor \( i \) and \( r_i \) is the radial distance from the source

\[ r_i = \sqrt{(x_i - x_s)^2 + (y_i - y_s)^2} \]
Cost Function to be Minimized

Given measurements $\bar{z}_i$ of concentrations at sensor $i$, $i = 1, \ldots, N$ where $N$ is the number of sensors.

Determine the source of the plume $(\bar{x}_s, \bar{y}_s)$ that minimizes the cost function:

$$J = \sum_{i=1}^{N} \frac{Q}{[(\bar{x}_s - x_i)^2 + (\bar{y}_s - y_i)^2] - \bar{z}_i}^2$$
Results with Different Signal to Noise Ratios (S/N)
Effect of increasing the number of sensors
Location Prediction

- The time-variant 2D Gaussian plume equation is given by:

\[
c(x, y, t) = \frac{Q}{2\pi t \sqrt{D_x D_y}} \exp \left( -\frac{1}{4t} \left( \frac{(x - vt)^2}{D_x} + \frac{y^2}{D_y} \right) \right)
\]

- \(c\): concentration (kg/m³)
- \(\nu\): wind speed (m/sec)
- \(Q\): Mass emission rate (kg/sec)
- \(D_y, D_z\): eddy diffusivities (m²/sec)
Location Prediction
Process Block Diagram
Model for Location Estimation and Prediction

• Given measurements of the concentration, a Least squares technique is used to estimate the parameters of the state space model:

\[
\begin{align*}
x_{(k+1)T} &= A_{kT}x_{kT} + w_{kT} \\
y_{kT} &= C_{kT}x_{kT} + v_{kT}
\end{align*}
\]

where \( x = \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix} \) is the state vector,

\((x, y)\) represents the plume location,

\((\dot{x}, \dot{y})\) represents the plume velocity,

\( y \) represents the measurements,

\( w, v \) noises.
Model Parameter Estimation

\[
\begin{bmatrix}
X_{(k+1)T} \\
Y_{kT}
\end{bmatrix} =
\begin{bmatrix}
A_k \\
C_k
\end{bmatrix}
\begin{bmatrix}
x_k \\
v_k
\end{bmatrix} +
\begin{bmatrix}
w_{kT} \\
v_{kT}
\end{bmatrix}
\]

\[
Y_{kT} =
\begin{bmatrix}
X_{(k+1)T} \\
Y_{kT}
\end{bmatrix},
\quad
\Theta_{kT} =
\begin{bmatrix}
A_k \\
C_k
\end{bmatrix},
\quad
E_{kT} =
\begin{bmatrix}
w_k \\
v_k
\end{bmatrix}
\]

\[
Y_{kT} = \Theta_{kT} + E_{kT}
\]

- We use Kalman filter to estimate and predict the states, after estimating the model parameters using Least Square techniques.
Estimation and prediction

• For the least squares problem, find $\theta$ such that the prediction error:

$$\frac{1}{N} \sum_{k=1}^{N} \| Y_{kT} - \Theta_{kT} + x_{kT} \|^2$$

is minimized.
Kalman Filter Equations

\[ x_{(k+1)T} = A_{kT} x_{kT} + w_{kT} \]

\[ y_{kT} = C_{kT} x_{kT} + v_{kT} \]

\[ E w w^T = R_1, \quad E v v^T = R_2, \quad E w v^T = R_{12} \]

\[ \hat{x}_{(k+1)T} = A_{kT} \hat{x}_{kT} + K_{kT} (y_{kT} - C_{kT} \hat{x}_{kT}) \]

\[ \hat{y}_{kT} = C_{kT} \hat{x}_{kT} \]

\[ K_{kT} = (A_{kT} \bar{P}_{kT} C_{kT}^T + R_{12}) (C_{kT} \bar{P}_{kT} C_{kT}^T + R_2)^{-1} \]

\[ \bar{P}_{kT} = A_{kT} \bar{P}_{kT} A_{kT}^T + R_1 - K_{kT} (A_{kT} \bar{P}_{kT} C_{kT}^T + R_{12})^T \]

\[ \hat{y}_{kT} = C_{kT} (q I - A_{kT} + K_{kT} C_{kT})^{-1} K_{kT} y_{kT} \]
K- Step Prediction
K=3 hours
Prediction Error

![Graph showing prediction error over time](image)
Mobile Sensors

• To track the plume, we need mobile sensors to move to the predicted plume location and collect measurements.

• The sensor dynamics are represented by

\[ M_1 \ddot{Y}_1(t) + (B + B_1) \dot{Y}_1(t) + (K + K_1)Y_1(t) - B\dot{Y}_2(t) - KY_2(t) - B\dot{Y}_3(t) - KY_3(t) = u_1(t) \]

\[ M_3 \ddot{Y}_2(t) + (B + B_2) \dot{Y}_2(t) + (K + K_2)Y_2(t) - B\dot{Y}_1(t) - KY_1(t) - B\dot{Y}_3(t) - KY_3(t) = u_2(t) \]

\[ M_3 \ddot{Y}_3(t) + (B + B_3) \dot{Y}_3(t) + (K + K_3)Y_3(t) - B\dot{Y}_1(t) - KY_1(t) - B\dot{Y}_2(t) - KY_2(t) = u_3(t) \]
Sensor Tracking

- Sensor system dynamics is governed by the following differential equations:

\[
M_1 \ddot{Y}_1(t) + (B + B_1)\dot{Y}_1(t) + (K + K_1)Y_1(t) - BY_2(t) - KY_2(t) - BY_3(t) - KY_3(t) = u_1(t) \\
M_3 \ddot{Y}_2(t) + (B + B_2)\dot{Y}_2(t) + (K + K_2)Y_2(t) - BY_1(t) - KY_1(t) - BY_3(t) - KY_3(t) = u_2(t) \\
M_3 \ddot{Y}_3(t) + (B + B_3)\dot{Y}_3(t) + (K + K_3)Y_3(t) - BY_1(t) - KY_1(t) - BY_2(t) - KY_2(t) = u_3(t)
\]

- Linear Quadratic Regulator (LQR) is used to control position destinations.
LQR Controller

- For a discrete-time linear system described by

\[ x_{(k+1)} = Ax_k + Bu_k \]

- the optimal control sequence minimizing the performance index is given by

\[ J = \sum_{k=0}^{\infty} x_k^T Q x_k \]

\[ u_k = -F x_k \]

and \( P \) is the solution to the discrete time algebraic Riccati equation

\[ P = Q + A^T(P - PB(R + B^TPB)^{-1}B^TP)A \]
Complete Process Demonstration

- Video file
Application Scenario – Dynamic Target Classification

Ying Sun
Dynamic Target Classification

- **Main idea**: dynamically select the optimal combination of features and classifiers based on the “probability” that the target to be classified might belong to a certain category.

- **Example**:
  - First Sensor detect the target using full set of features
  - The rest selected sensors using optimal combination of features and classifiers
  - First Sensor detect the target using full set of features
Dynamic Target Classification (cont’d)

• Content
  – **Hypothesis**: for each type of target, there exists an optimal set of features in conjunction with a specific classifier, which can yield the best performance in terms of classification accuracy using least amount of computation, measured by the number of features used.
  – **Dynamic Target classification**: base the selection of optimal combination of features and classifiers on the “potential” appearance of a certain target

• Outline
  – Feature selection and Classification
  – Optimal combination selection rule
  – Validation
Feature selection

Data

Normalization

Fast Fourier Transform
- Shape Statistics (4)
- Amplitude Statistics (4)
- Peak selection (6)

Wavelet Transform
- Average (4)
- Standard deviation (4)

Spectrogram Analysis
- Energy
- Spectrogram peaks selection (6)

Feature Vectors (32 in total, 7 groups)
- FFT1
- FFT2
- FFT3
- WT1
- WT2
- WT3
- Spec
Classification

• **k-Nearest Neighbor (kNN)**
  assigns the unknown sample to the class $i$ if more samples belong to class $i$ within the $k$ nearest neighbors of the unknown.

• **Maximum Posterior Probability (MPP)**
  Discriminant Function
  For a given $x$, if $P(w_1 \mid x) > P(w_2 \mid x)$

  Then $x$ belongs to class 1, otherwise, $x$ belongs to class 2

$k=5$

Unknown belongs to class 1
Optimal Combination Selection Rule

• Feature space:
  \[ x = [\text{fft1, fft2, fft3, wt1, wt2, wt3, spec}] \]
  \[ 2^{32} - 1 \rightarrow 2^7 - 1 \]
  \[ 4.295 \times 10^9 \rightarrow 127 \]
  exhaustive search on the combination space.

• Target Type:
  AAV (Assault Amphibian Vehicle), DW (Dragon Wagon), HMMWV (High Mobility Multipurpose Wheeled Vehicle)

• Cost Function:

\[
F_{\text{min}}(x, y) = \frac{f_1(x, y)}{f_2(x, y)} \quad \left\{ \begin{array}{l}
  f_2 > 0.8 \quad \text{AAV} \\
  f_2 > 0.7 \quad \text{DW, HMMWV}
\end{array} \right.
\]
Combination Results

Table 1: Optimal combination results validated on SITEX00 and SITEX02 data sets.

<table>
<thead>
<tr>
<th>Type</th>
<th>Met</th>
<th>Optimal</th>
<th>Thr</th>
<th>Accu</th>
<th>Time</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAV</td>
<td>MPP</td>
<td>FFT1</td>
<td>0.8</td>
<td>0.8204</td>
<td>1.1367</td>
<td>1.3856</td>
</tr>
<tr>
<td>DW</td>
<td>MPP</td>
<td>WT2, WT3</td>
<td>0.7</td>
<td>0.7463</td>
<td>0.7850</td>
<td>1.0518</td>
</tr>
<tr>
<td>HMMWV</td>
<td>MPP</td>
<td>WL3</td>
<td>0.7</td>
<td>0.7325</td>
<td>0.6288</td>
<td>0.8584</td>
</tr>
</tbody>
</table>

Note: Met–Method, Optimal–Optimal feature set, Thr–Threshold, Accu–Accuracy.
Application Scenario

- Target enters the region
- First node (A) activate, sense, classify
- A send request to all its neighbors, select the best one
- Hands off result to B
- B looks up table, select the optimal combination, make a new decision
- \( r_{\text{pre}} = r_{\text{new}} \), go to step 3.

- When conflict happens, the current use full set of features to classify the target, then go to step 3.
Experimental Results

Figure 3: Accuracy comparison between normal and dynamic classification for AAV, DW, and HMMWV. Solid line: dynamic classification; Dash line: normal classification. X-axis: selected nodes Index, Y-axis: accuracy

Figure 4: Response time comparison between normal and dynamic classification for AAV, DW, and HMMWV. Solid line: dynamic classification; Dash line: normal classification. X-axis: selected nodes Index, Y-axis: response time
Results (cont’d)

(a) Accuracy

<table>
<thead>
<tr>
<th></th>
<th>AAV</th>
<th>DW</th>
<th>HMMWV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full</td>
<td>81%</td>
<td>53.33%</td>
<td>74.18%</td>
</tr>
<tr>
<td>Opt</td>
<td>95.57%</td>
<td>70.57%</td>
<td>89.38%</td>
</tr>
</tbody>
</table>

(b) Time

<table>
<thead>
<tr>
<th></th>
<th>AAV</th>
<th>DW</th>
<th>HMMWV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total response time (s)</td>
<td>1.5954</td>
<td>0.9090</td>
<td>1.2274</td>
</tr>
<tr>
<td></td>
<td>0.4356</td>
<td>0.2575</td>
<td>0.266</td>
</tr>
</tbody>
</table>
Dynamic Classification Application
Cyber Attack Detection

• Benefits of dynamic classification meets requirements for cyber attack detection system;
• ‘tcpdump’ intercepts the packets—redundant features—dynamic classification performs better
Dynamic Classification Application Cyber Attack Detection (cont’d)

- Assumption:
  for each time interval, only one type of intrusion attacks the system consecutively

- Steps:
  1. For the first receiving packet, use all the input features to classify: normal or intrusion, intrusion type;
  2. Once the type is determined, use optimal combination set to classify the rests.
  3. When conflict occurs, which might means wrong classification or another type of action happens, then again use full features to classify the action;
  4. Return to step 2.
Future work

• Build the application model
• Integration with INFOD
• What kind of data can be maintained in INFOD
• Implementation: user interfaces
• Performance comparison: using INFOD v.s. without using INFOD
Collaborative Opportunities

• ORNL’s Shelby County Sensor Information Fusion Center project provides a comprehensive target application scenario for INFOD.
  – Help establish common understanding (finding matches) between homeland security response agencies
  – Facilitate the sharing of federated data for first responders and consequence responses

• All the research findings and software developments will be accessible through public domains
Shelby County Fusion Center
SNAPS+POM+NOAA+INFOD+Video Management

1. Shelby County Sheriff
SNAPS II Mobile System
- 8 chem/5 rad/5 video /1 weather sensors
- Fusion Center Portal and Viewer (Web Server; Database; GIS (Google); HPAC plume modeling)

2. Port of Memphis Sensors
- 5 chem/1 weather sensors

3. NOAA Live Regional Weather
- Plotting of Data
- Display Video Feeds
- HPAC with Live Weather Feeds
- GIS Situational Awareness (ArcView or Google Earth, Browsers...)

4. INFO-D
- Distributed Wide-Area Middleware
  - Prototype and Analysis
  - Distributed querying and top-down programming
  - Policy-based data-sharing
  - Asynchronous messaging

Contact: ORNL UT-Battelle; HT Hunter; Hunterht@ornl.gov; 865-574-6297

Managed by UT-Battelle for the U.S. Department of Energy – Supporting the Department of Homeland Security
### Project Timeline

<table>
<thead>
<tr>
<th>Tasks</th>
<th>06/07 - 05/08</th>
<th>06/08 - 05/09</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Q1</td>
<td>Q2</td>
</tr>
<tr>
<td>Task 1: Design, Development, and Evaluation of the DSMP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 1.1: Literature survey and document study</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 1.2: Prototype design and development</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 1.3: Middleware performance evaluation and refinement</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 2: Demonstration of the DSMP</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 2.1: Application scenario 1 - collaborative event analysis</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 2.2: Application scenario 2 - establishing data correlation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Task 2.3: Application scenario 3 - first responders and consequence response at Shelby County/Memphis</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- The development of the DSMP (Task 1.2) has been divided into a 4-phase implementation plan. Because of the close collaboration with Oracle, IBM, ORNL, and RAL, we are able to finish Phase II ahead of time. With the phase II prototype accomplished, the application development tasks can be carried out in parallel to the DSMP development.
Budget Information

• Project budget (June 5, 2007 - May 31, 2009): $400,000
• Spending as of March 31, 2008: $103,308
Commercialization Progress

• Since the project has just started on June 5, 2007, there has not been any progress made toward commercialization.
IP STATUS

• No disclosures have been filed so far.
Achievements

- Identified first responder as the ultimate use case scenario for DHS interest
- Finished phase II of the prototype development of INFOD
- Finished simulation of collaborative event analysis
- Proposed a new plume model with promising simulation results
- Visited SCFC and discussed potential integration scenario
- New issues identified
  - Real-time guarantee of system performance
  - Security issues of INFOD
  - Collaboration with Vanderbilt on security issues and the usage of generic modeling environment (GME)
- Papers submitted
  - “Control-based real-time metadata matching for information dissemination”, to be submitted to 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications
- Presentations at two OGF meeting
- Semi-annual review in Sept. 2007
What’s Ahead?

• Semi-annual review meeting in Sept. 2008
  – Demonstration of the first responder user case
  – Demonstration of dynamic target classification
  – Integration of SCFC and video management with INFOD