# Middleware Support for Disaster Response Infrastructure

Jun Suzuki and Tatsuya Suda
jsuzuki@ics.uci.edu
www.ics.uci.edu/~jsuzuki/
netresearch.ics.uci.edu/bionet/
Dept. of Information and Computer Science
University of California, Irvine

## Our Assumptions on the Disaster Infrastructure

- Ad-hoc net spontaneously established in a disaster area to evacuate victims and aid emergency response crews.
- Various devises participate in the disaster ad-hoc nets.
  - Victims carry their own devices.
  - Emergency response crews carry and/or wear devises.
  - Emergency vehicles (e.g. fire truck, ambulance) carry devices.
  - Sensors are densely scattered (e.g. scattered from helicopters).

# Our Assumptions on the System Characteristics

- Large scale with a number of
  - people/organizations
  - devices
  - software objects
    - Objects represent devises, execute devise-specific functions (e.g. temperature sensing), or carry information (e.g. map, a building's floor plan and air contamination).
- Heterogeneous
  - processing, memory and networking capabilities of devises
  - functionalities of software objects
- Dynamic
  - changing network connectivity, density, and traffic
    - Connectivity and density change due to movement of users/devises and additional deployment of devises and software objects.
    - Traffic changes depending on the rescue operation stages
      - e.g. The traffic among temperature sensors increases while fire occurs.
  - intermittent availability of devises and software objects

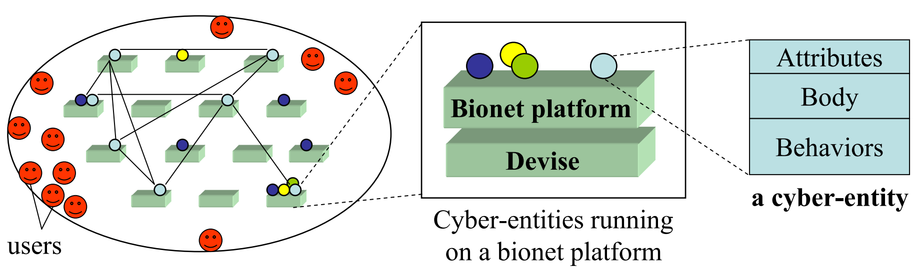
# Research Goals

- To design an application architecture which
  - meets key requirements of applications running on disaster response networks (i.e. large-scale, heterogeneous and dynamic networks).
  - diminishes the maintenance/administration burden of disaster response network applications.

# A New Application Architecture

- Key requirements in disaster response network applications
  - *scalability* in terms of # of objects/devises/users,
  - *adaptability* to dynamic changes in network conditions
  - *availability/survivability* from failures
  - *simplicity* to develop and maintain.
- The Bio-Networking Architecture
  - applies biological concepts and mechanisms to network application design
    - Biological systems already have above required characteristics

# Cyber-Entity (CE)




- Biological individual = Cyber-entity (CE)  (objects)
  - Abstraction of system components (e.g., victim, rescuer, service, etc.)
  - provides service (e.g. temperature sensing, providing information such as building's floor plan).
  - autonomous with simple behaviors
    - migration, replication, reproduction, death, energy exchange, relationship establishment, discovery
- Application
  - constructed from a collection of interacting cyber-entities

# Biological Concepts Applied

- Emergence
  - Useful group behavior (e.g. adaptability and survivability) emerges from autonomous local interaction of individuals with simple behaviors.
- Lifecycle
  - energy gain/consumption/exchange
    - CE gains energy in exchange for providing its service.
    - It expends energy for using resources (e.g. CPU and memory) and performing behaviors (e.g. migration and replication)
- Adaptation and evolution
  - CEs evolve by generating behavioral diversity and executing natural selection.
  - replication (with mutation), reproduction (with mutation and/or crossover) of CEs
- Decentralized system organization
  - to increase scalability and robustness
  - e.g., decentralized discovery
    - Each CE keeps relationships with others. Discovery is performed based on CE's unique ID and attributes through relationships in a peer-to-peer manner.

# Application Scenario 1: Wildfire

- Disposable sensors are scattered over an affected area
  - e.g. temperature, wind force, oxygen, smoke sensing
  - Some of them are broken if they fall into a fire.
  - The CEs within sensors do their sensing tasks and maintain relationships with each other.
- Each fire fighter has devises (e.g. info pad, sensors).
  - The CEs within the devises may
    - direct the fire fighter to a place to extinguish a flame, even when visibility is not good, by interacting with scattered sensor CEs.
      - The CEs may suggest a safer (i.e. lower temperature, less air contaminant) route to the place from multiple options.
    - display the current positions of the fire fighter and other fighters by interacting with other fighters' CEs and the CEs that provide map information.
    - display the current area affected by fire(s) by interacting with sensor CEs.
    - sense what is happening nearby (e.g. approaching blaze) by interacting with neighboring sensor CEs, and alert the crew that.

## Application Scenario 2: Building Collapse

- The CEs within victims’ devises may
  - find rescuers through passing advertisement (e.g. “I’m here” beacon) or asking its relationship partners (they will ask their partners in turn).
  - provide an evacuation path to the victim by interacting with sensor CEs.
  - obtain the first aid treatment information for injured victims by discovering and inquiring the CEs that provides the information.
- The CEs within rescuers’ devises may
  - locate victims, represented by CEs, through passing advertisement or asking its relationship partners (they will ask their partners in turn).
  - display a street map or building floor plans depending on the rescuer’s current position.
  - examine what is happing near the rescuer (e.g. gas leaking and approaching blaze) by discovering and inquiring nearby sensor CEs.
- A CE that provides any information may
  - adjust its population through replication, reproduction and natural selection (energy exchange) depending on the demand;
  - adjust its location through migration (e.g. toward users) and resource sensing (e.g. more CEs on the devises that provide more resources).

## Current Status and Future Work

- Current status
  - Design and implementation of a platform software
    - OMG standardization (Super Distributed Objects group)
  - Distributed (i.e. peer-to-peer) discovery
  - Adaptation and evolution
  - Service interface description language
  - Mathematical stability analysis
- Future work
  - Deployment and empirical study
  - Reconfigurable middleware