# Enabling Robust C2 Systems through Evolvable Human-In-The-Loop Data Fusion

Sam Mahoney, Jonathan Pfautz, Ted Fichtl, Sean Guarino, Eric Carlson
Charles River Analytics Inc.

Gerald Powell
U.S. Army RDECOM CERDEC I2WD

Emilie Roth
Roth Cognitive Engineering

14th ICCRTS

June 2009

Roth Cognitive Engineering

charles river analytics

# Data Fusion is…

- Data fusion systems combine, correlate, and aggregate heterogeneous and distributed sources of information with the goal of providing needed information (Waltz & Llinas, 1990)
- For detection, tracking, classification, and identification
  - Across 10 seismic sensors, there's enough evidence to detect a passing vehicle
  - At time t and t+1, point (x1,y1,z1) is the same entity as point (x2,y2,z2), form a vector - this entity is traveling at 40mph NE
- How?
  - Computational methods galore!

# Data Fusion and Ontologies

- Computational methods typically require fixed descriptions of the world – "ontologies"
  - A definition of a specification
- Examples:
  - Weather = rain, sleet, hail, snow, cloudy, clear
- Ontologies are used as data structures within fusion systems and to guide inferences
  - If sensor X reports "wet" then report "rain"
- Fusion ontologies are typically designed from sensor capabilities
  - And often early in system design

… leading to problems in adapting to change

# Goals

- The research is being performed as part of an Army program focused on developing next-generation fusion methods that:
  - Enable data fusions systems that will be knowledge-intensive
  - Respond to a changing battlefield environment:
    - New threat doctrine
    - Varying tactics, techniques, and procedures (TTPs)
    - Equipment or weapon changes by the threat
    - Man-made and natural terrain features)
- A key goal of the program is to develop practical, operational systems
- This includes *evolvable support* (Roth et al., 2006) for data fusion systems
- How do we design and build an evolvable data fusion system?
  - With a human in-the-loop?
  - To evaluate different course of action (COAs)?

# Approach: Cognitive Systems Engineering

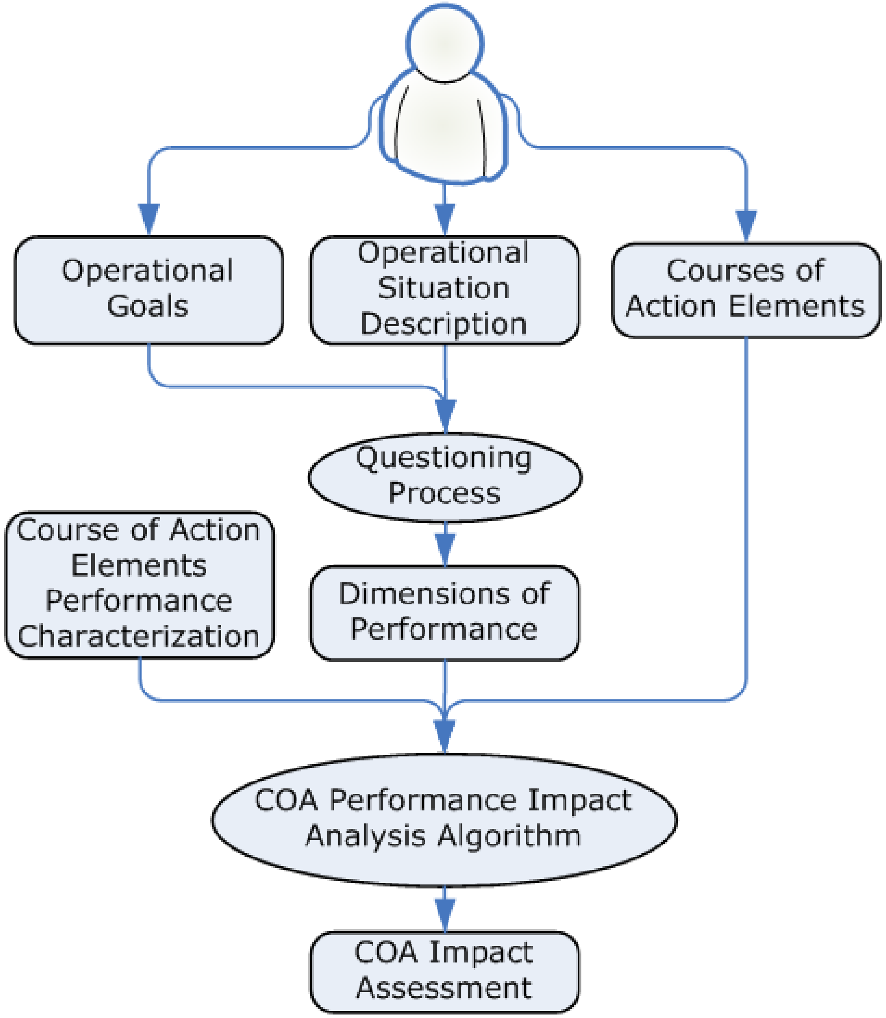
- Performed initial Cognitive Task Analysis
  - Interviews with 3 primary Army Intelligence SMEs
  - Visits to military installations - 40+ active-duty soldiers interviewed
  - Rough estimate of interviewee-hours: >750
- Identify functions performed by the analyst and the data fusion system
  - E.g., monitoring, diagnosing/assessing, deciding, planning, communicating
  - Understand the "as is" or current process vs. prescribed/doctrinal process
  - Understand the goals and constraints in the environment
- Identify sources of information and meta-information for each function
  - E.g., pedigree, confidence, rigor
- Define the complexities of the problem domain from an operator's perspective
  - E.g., time pressure, lack of information, information overload, uncertainty
- Study existing data fusion processes and how they currently account for evolution
- *Provides the basis for understanding how operators need to interact with and reason about the data fusion process to perform optimally*

# Cognitive Analysis and Initial Development: Human-in-the-Loop Data Fusion

- Examined over 200 objective questions that soldiers may need to address with the fusion system
  - E.g., What is the most effective COA when facing a unit employing SA-6 surface-to-air capabilities?
- Interviews revealed categories of factors most important to answering these questions
- Developed an interrogative interface that targets these factors
  - What is your primary objective?
  - Characterize the terrain where your objective is located?
- Developed an initial set of answers to questions
- Related answers to ontology employed in the fusion algorithm
- The user *guides* the data fusion with these answers!

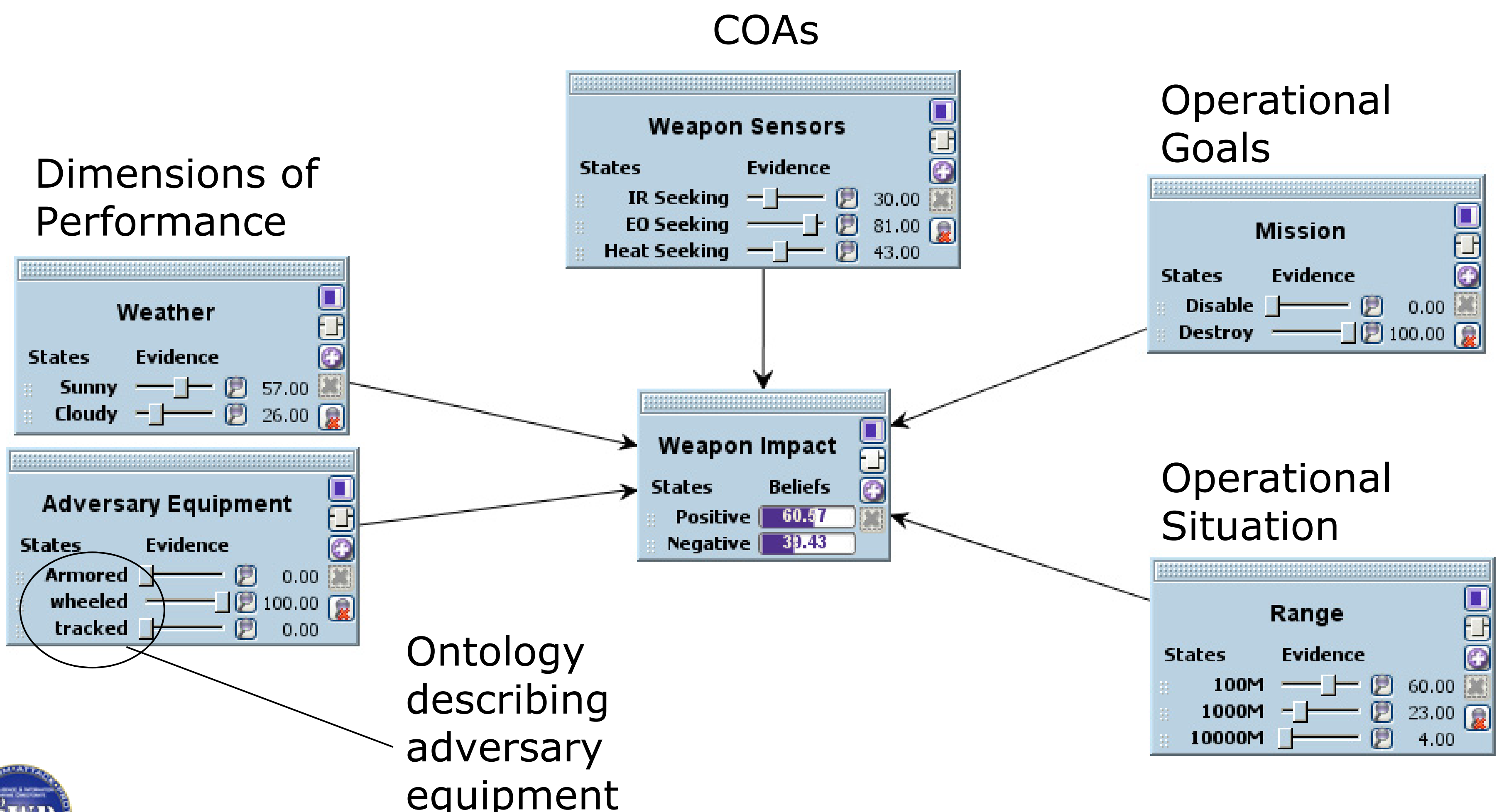
# Initial Prototype:
# Support for Human-in-the-Loop Data Fusion

- Operational Goals
  - Mission: Evacuation operation
- Operational Situations
  - Requires support from air assets
- Courses of Action
  - Use defensive IR-guided weapons
- Dimensions of Performance
  - Weather, terrain, adversary assets
- COA Performance Impact Analysis Algorithm
  - *Will the employment of IR-guided systems be effective?*



# Initial System Description

- Assume we are interfacing with Bayesian reasoning algorithm to direct a fusion system's prioritization of targets, then…



# Adapting to Change

- Did evaluations at ~6-month intervals over 2 years
  - Terminology, TTPs changing fast!
  - **Need evolvable system!**
- But how?
  - Revisited Cognitive Task Analysis
    - *To understand the vulnerabilities to change*
  - Defined questions to reveal transient aspects of domain, e.g.,
    "What will the new doctrine do to how you define X?"
    "When you were first trained, how did you assess the impact of factor Y?"
    …
  - *Iterated* on evolvable parts of system
    - Across domain experts' and users': Areas of expertise, areas of experience, years of experience
    - Repeated interviews
    - Repeated tests and refined system designs
    - Developed corpus of examples (!)

# Revisiting our Cognitive Task Analysis and Design: Terminology Issues

- What is transient?
  - Terminology – and association with doctrine, adversary tactics
  - But not underlying meaning and implications

- Performed iterative analysis to develop abstract representation
  - Resistant to terminology change
  - E.g., "Pickup truck", "A Technical", "VBIED"
    … "a singular instance of a small, vehicle-based threat"

- Example abstractions
  - Count: singular, multiple
  - Area: point, line, defined/undefined area, abstract
    … remember, these map to data fusion methods

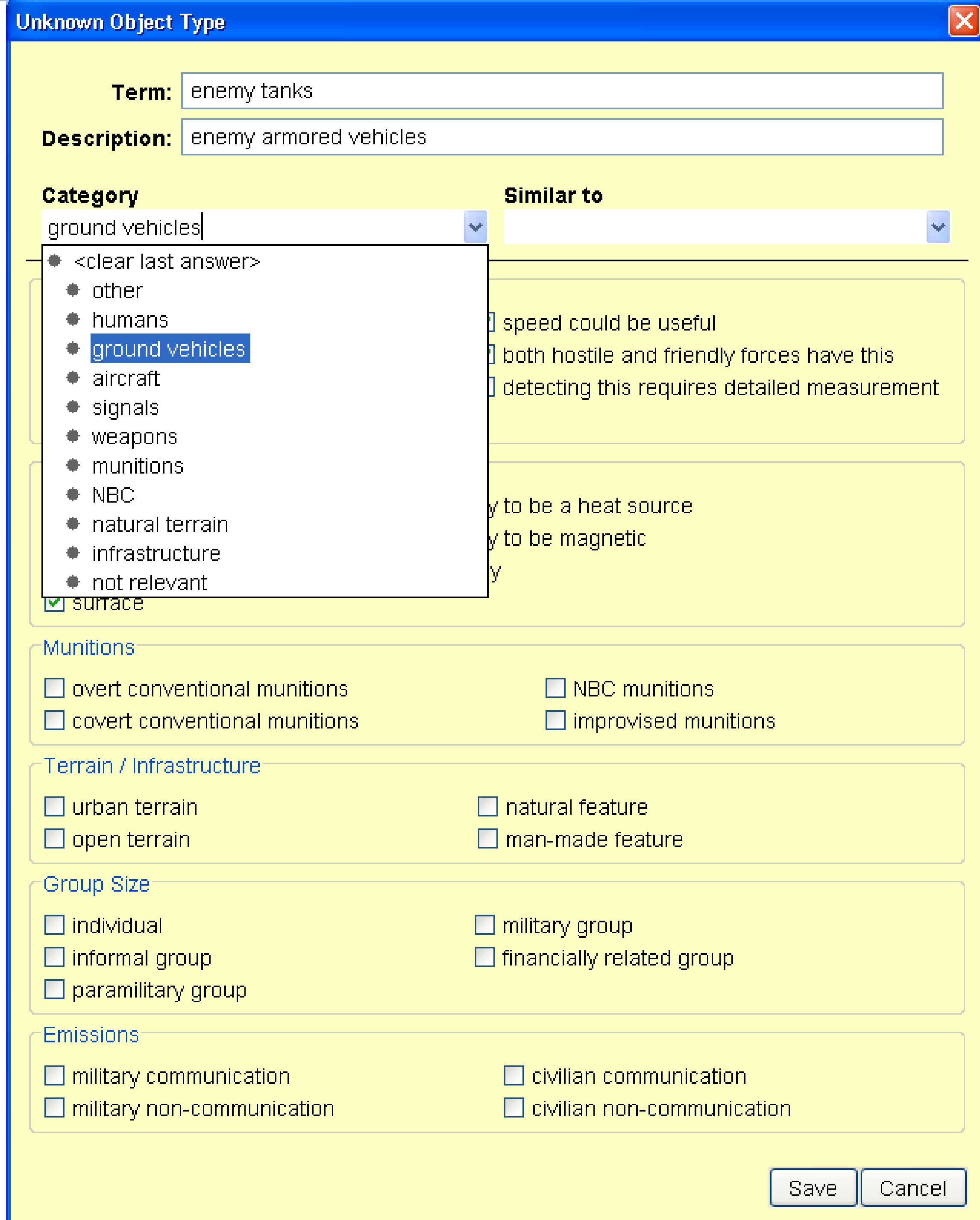
# Revisiting our Cognitive Task Analysis and Design: Terminology Issues

- In our data fusion system, support definition of new or missing terms
- Users can:
  - Drill down to find explanation of specific terms in abstraction
  - Using an existing term as a basis – define by analogy
  - Define the term against the abstraction
  - Create categories of terms with properties and inheritance
- E.g.,

  New term: "Foo"

  A type of "a singular instance of a small, vehicle-based threat"

  But, using large vehicles…


Unknown Object Type
Term: enemy tanks
Description: enemy armored vehicles
Category
ground vehicles
Similar to
<clear last answer>
other
humans
ground vehicles
aircraft
signals
weapons
munitions
NBC
natural terrain
infrastructure
not relevant
speed could be useful
both hostile and friendly forces have this
detecting this requires detailed measurement
to be a heat source
to be magnetic
surface
Munitions
overt conventional munitions
covert conventional munitions
NBC munitions
improvised munitions
Terrain / Infrastructure
urban terrain
open terrain
natural feature
man-made feature
Group Size
individual
informal group
paramilitary group
military group
financially related group
Emissions
military communication
military non-communication
civilian communication
civilian non-communication
Save
Cancel

# Revisiting our Cognitive Task Analysis and Design: Uncertainty in Terminology

- Found that abstraction is inherently higher-level and vague
  - Need well framed terms and/or explanations
  - Need ability to say *"I don't know"* in the face of an unanticipated case not well supported by abstraction
- This uncertainty needs to be okay in underlying system!
- Users can:
  - Simply express "unknown" as response
    - Underlying formalism must still respond given known definitions
  - Annotate their definition
    - "Not sure if this fits this category or not"

# Revisiting our Cognitive Task Analysis and Design: Ownership and Authority

- Who owns the evolvability?
- In our case, experts expressed desire for:
  - Maintain individual adaptability
  - Authority for incorporating terms into shared, group-level system
  - Authority at a specific echelon level (e.g., Bn)
  - In other words, facilitate existing organizational methods for collaboration
- Design implication: Create both individual and shared corpus of terminology and definitions
- Future work:
  - Observe individual and group ontologies, use as data for refining abstraction

# Conclusions: Developing an Evolvable Systems

- Cannot take off-the-shelf CTA approach for evolvable systems
  - Ability to evolve appears proportional to on-going analysis effort!
  - Iteration really, really needs to happen
  - Domain experts' length and variation of experience is critical
  - Focus on *transient* elements of the domain
  - Higher effort in interview question design and analysis of example
    - What parts of your answer were different two years ago?
- System engineering for evolvability requires more design savvy and ingenuity… and, potentially, cost
  - Fortunately, engineers are encultured to think about extensibility
    Though typically w.r.t to systems, not users
  - And lifecycle cost assessment is harder to do
- Evolvable systems can provide feedback to design processes

# Questions?


Corresponding Author:

Samuel Mahoney
Charles River Analytics Inc.
625 Mt. Auburn Street
Cambridge, MA 02138

(617) 491-3474 x540
smahoney@cra.com
http://www.cra.com