THAYER SCHOOL OF ENGINEERING AT DARTMOUTH

Jacob Jurmain

Adversary Mission Characterization

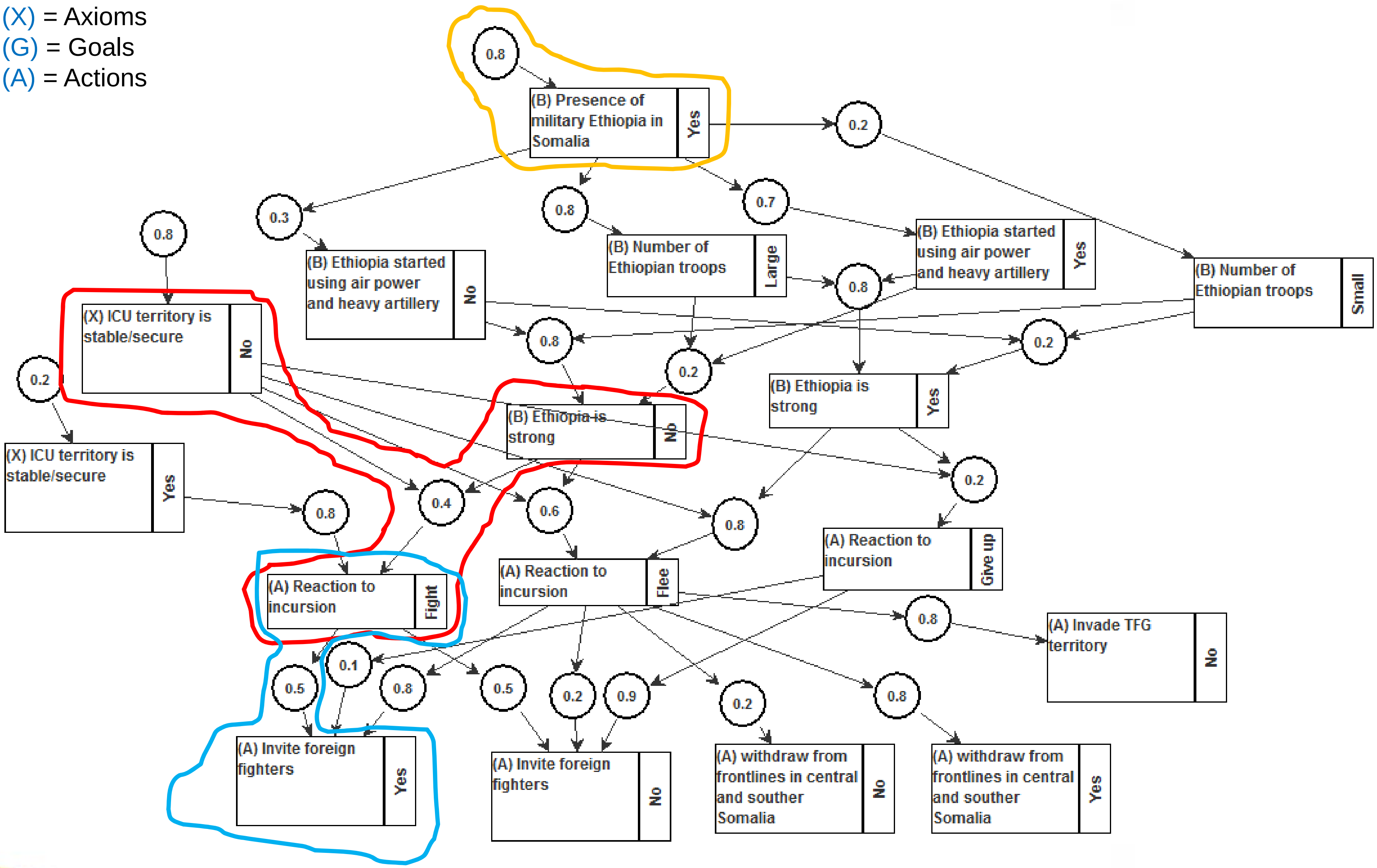
- Adversary Intent
- Bayesian Knowledge Bases
- Bayesian Knowledge-driven Ontologies

Adversary Intent

- Adversary mission characterization equates to “intent”.
  - For maximum insight, know the adversary’s perspective, motivation, and rationale, not just their methods.
    - Beliefs, axioms, goals, actions

- Model intent with Bayesian Knowledge Bases.
  - Focus on uncertainty and incompleteness – work with however much you know, even if that’s not much.
  - Focus on explanations – all inferences can be backtracked and completely explained.
  - Powerful features – capture cyclic knowledge and fuse multiple sources of knowledge, even when they conflict.

(B) = Beliefs
(X) = Axioms
(G) = Goals
(A) = Actions

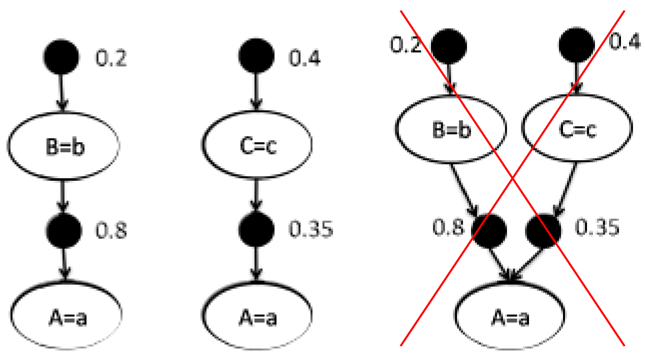


The Basic BKB Analyses

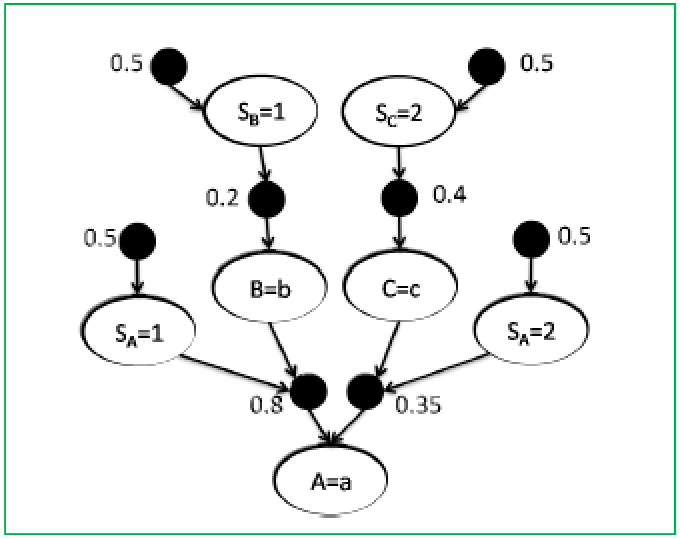
- Belief updating: What is the probability of a particular variable state assignment?
  - Ex: Given everything we know about this adversary, what are the probabilities they will attack with method X, Y, or Z?
- Belief revision: What is the most probable "state of the world"?
  - Ex: How likely are some possible explanations for the events we've just observed?
- Contribution analysis: How much did one variable state assignment appear as a cause of another?
  - Ex: How much did one source of motivation contribute to the adversary's actions, vs. another source?

Challenge of Conflicting Information

- Naïve union of rules violates mutual exclusion.
- Solution: Create a probability distribution over the sources.



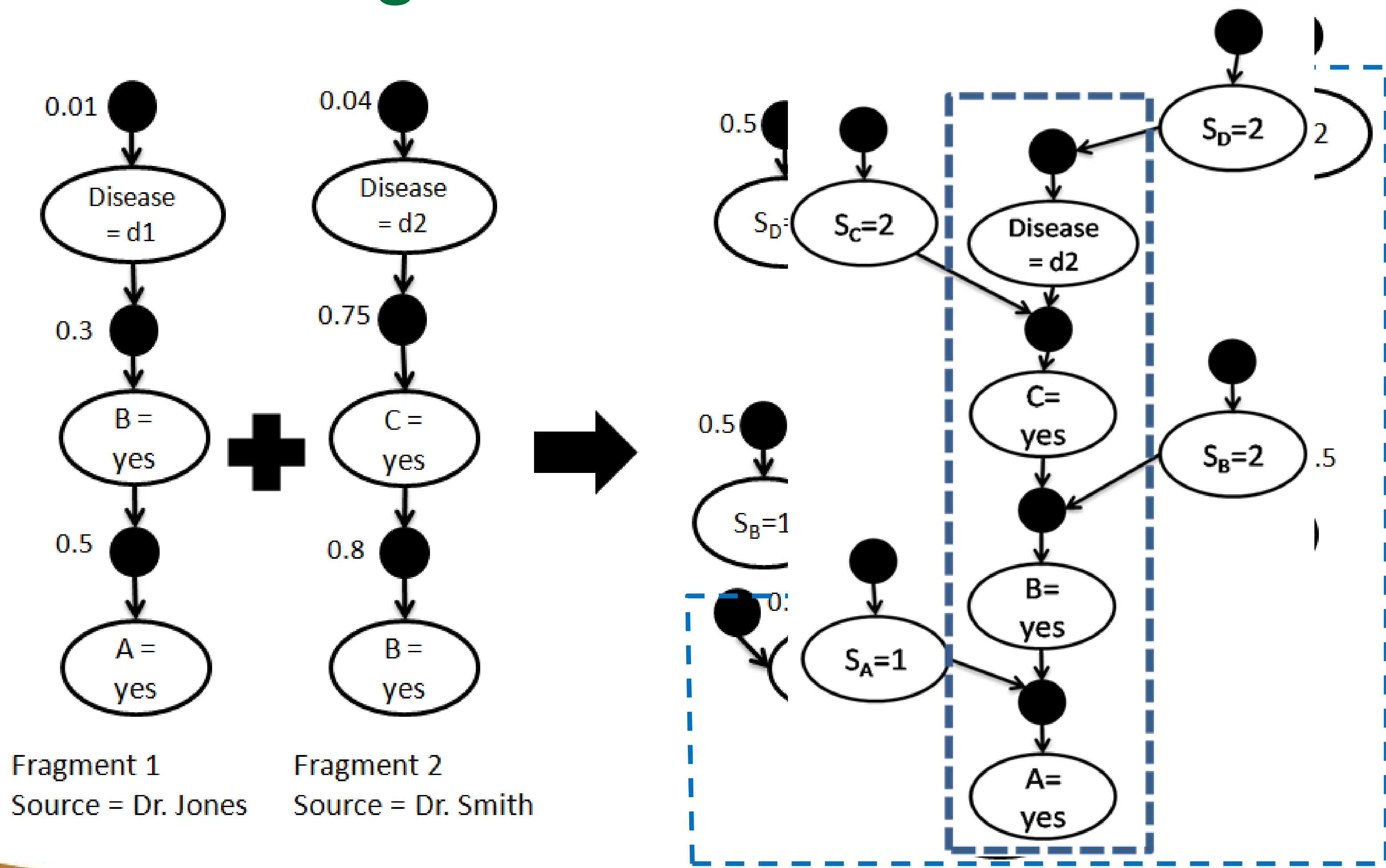
(1) (2)



Naïve union of fragments (1) and (2) puts CPRs in conflict. Invalid.

Source variables S_x prevent rules from conflicting because they give the rules mutually exclusive conditions.

New Insights From Fusion



BKB Practical Application

- For one-off analyses, manually build BKB fragments and fuse them.
  - Each fragment describes a facet of the situation.
  - Use the fused model for explanations and what-if analyses.

- For domains you plan to revisit, use our new result – Bayesian Knowledge-driven Ontologies (BKOs).
  - Starting from an initial description of the adversary, builds a broader characterization from background knowledge.
  - Uses a prebuilt library of general domain knowledge. More work up front, but analyses are easy once it's built.

BKO Theory

- Extension of BKBs to facilitate logical reasoning about probabilistic domain knowledge.
    - Automatically assemble case-specific BKBs for multiple analyses in a domain.
    - Can import ontologies and formally merge them with fusion.

- Recent publication: *E. Santos Jr. & J. Jurmain, "Bayesian Knowledge-driven Ontologies", Proc. IEEE SMC, Oct. 2011*
    - Core contribution – probabilistic terminological (i.e. 1st order) knowledge expression and logical reasoning.
        - Past attempts were either crippled or restricted.
    - Core insight – a fundamental connection between ontologies and probability theory.

THAYER SCHOOL OF ENGINEERING AT DARTMOUTH

Expression and Reasoning

BKOS – HOW THEY WORK

Terminological Knowledge – Old Way

- Ontologies make inferences by applying terminological knowledge to assertional knowledge.
  - No uncertainty allowed. Only T/F variable interactions.
  - Ex: We know that a specific vehicle is a car and that all cars have wheels. Therefore that specific vehicle has wheels.

$$\text{The\_Vehicle} \in \text{Car}$$
$$\text{Car} \sqsubseteq \text{has\_Wheels}$$
$$\Longrightarrow \text{The\_Vehicle has Wheels}$$

Uncertain Terminological Knowledge

- BKOs extend this to handle uncertainty.
  - More complex variable interactions allowed.
  - Ex: We're pretty sure we've been hit by a DDoS attack. How bad is the threat?

$$P(\text{attack} \in \text{DDoS}) = 0.95$$

The library says DDoS's are usually from amateur hacktivists, but sometimes are from a foreign government:

$$P(\text{x done\_by } \mathbf{some} \text{ Hacktivist} | \text{any x} \in \text{DDoS}) = 0.7$$
$$\Rightarrow P(\text{attack done\_by } \mathbf{some} \text{ Hacktivist} | \text{attack} \in \text{DDoS}) = 0.7$$

$$P(\text{x done\_by } \mathbf{some} \text{ Gov't} | \text{any x} \in \text{DDoS}) = 0.2$$
$$\Rightarrow P(\text{attack done\_by } \mathbf{some} \text{ Gov't} | \text{attack} \in \text{DDoS}) = 0.2$$

The library also says government attacks are more likely to be a real threat:

$$P(\text{x} \in \text{Threat} | \text{any x done\_by } \mathbf{some} \text{ Gov't}) = 0.6$$
$$\Rightarrow P(\text{attack} \in \text{Threat} | \text{attack done\_by } \mathbf{some} \text{ Gov't}) = 0.6$$

$$P(\text{x} \in \text{Threat} | \text{any x done\_by } \mathbf{some} \text{ Hacktivist}) = 0.1$$
$$\Rightarrow P(\text{attack} \in \text{Threat} | \text{attack done\_by } \mathbf{some} \text{ Hacktivist}) = 0.1$$

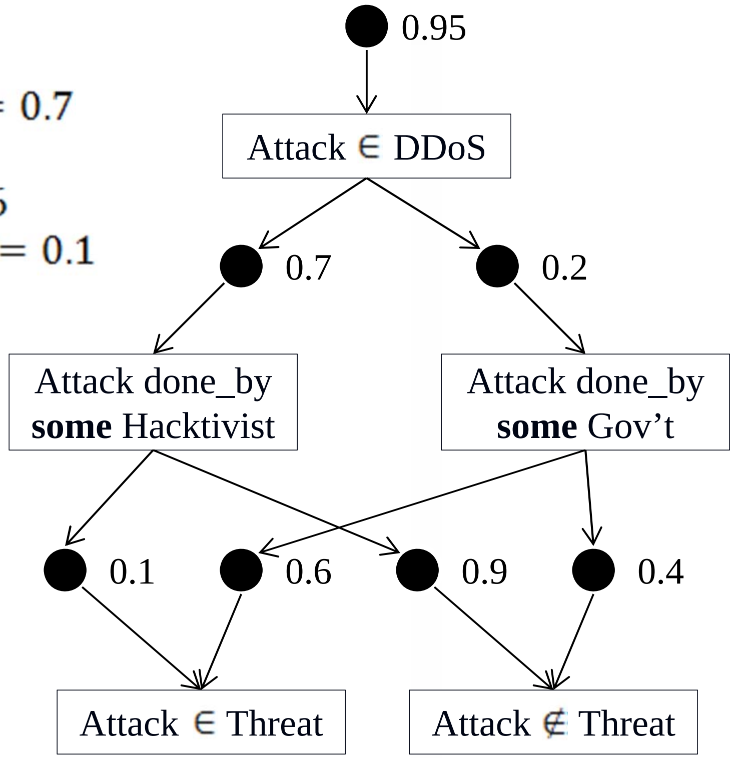
$P(\text{attack} \in \text{DDoS}) = 0.95$

$P(\text{attack done\_by } \textbf{some} \text{ Hacktivist} | \text{attack} \in \text{DDoS}) = 0.7$

$P(\text{attack done\_by } \textbf{some} \text{ Gov't} | \text{attack} \in \text{DDoS}) = 0.2$

$P(\text{attack} \in \text{Threat} | \text{attack done\_by } \textbf{some} \text{ Gov't}) = 0.6$

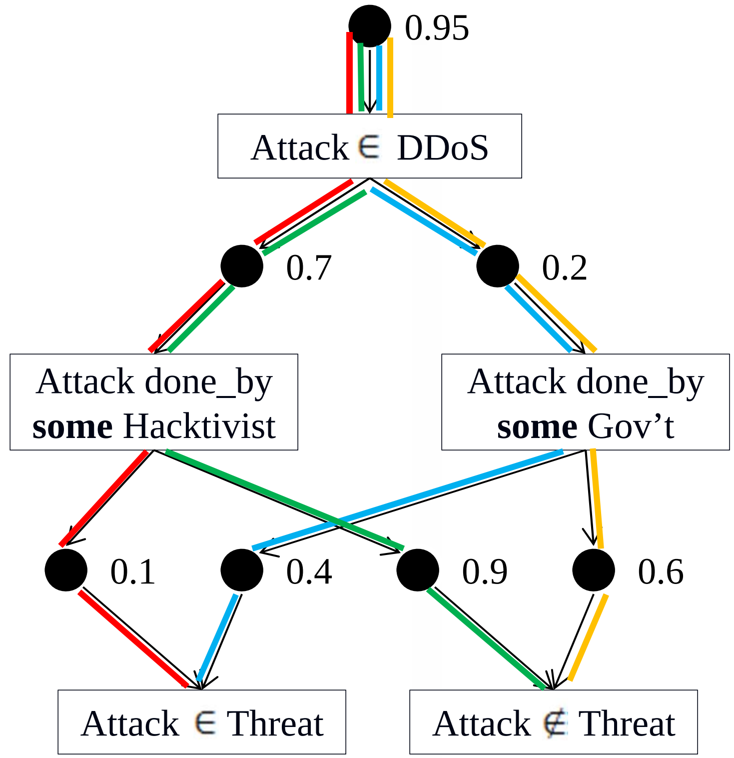
$P(\text{attack} \in \text{Threat} | \text{attack done\_by } \textbf{some} \text{ Hacktivist}) = 0.1$



- Belief revision: determine most probable state of the world.
  - P = 0.5985

- Belief updating: compute posterior probability of a single variable assignment.
  - Sum of probabilities of inferences that assignment appears in.
  - P(Attack ∈ Threat) = 0.0665 + 0.076 = 0.1425

- Contribution analysis: compute how much one random variable appears as a cause of another.
  - Sum of probabilities of inferences in which the hypothesized cause appears with the effect, divided by the effect's posterior probability from updating.
  - Contribution of "Attack done_by **some** Hacktivist" to "Attack ∈ Threat" 0.0665 / 0.1425 = 0.467



P = 0.0665
P = 0.076
P = 0.5985
P = 0.114

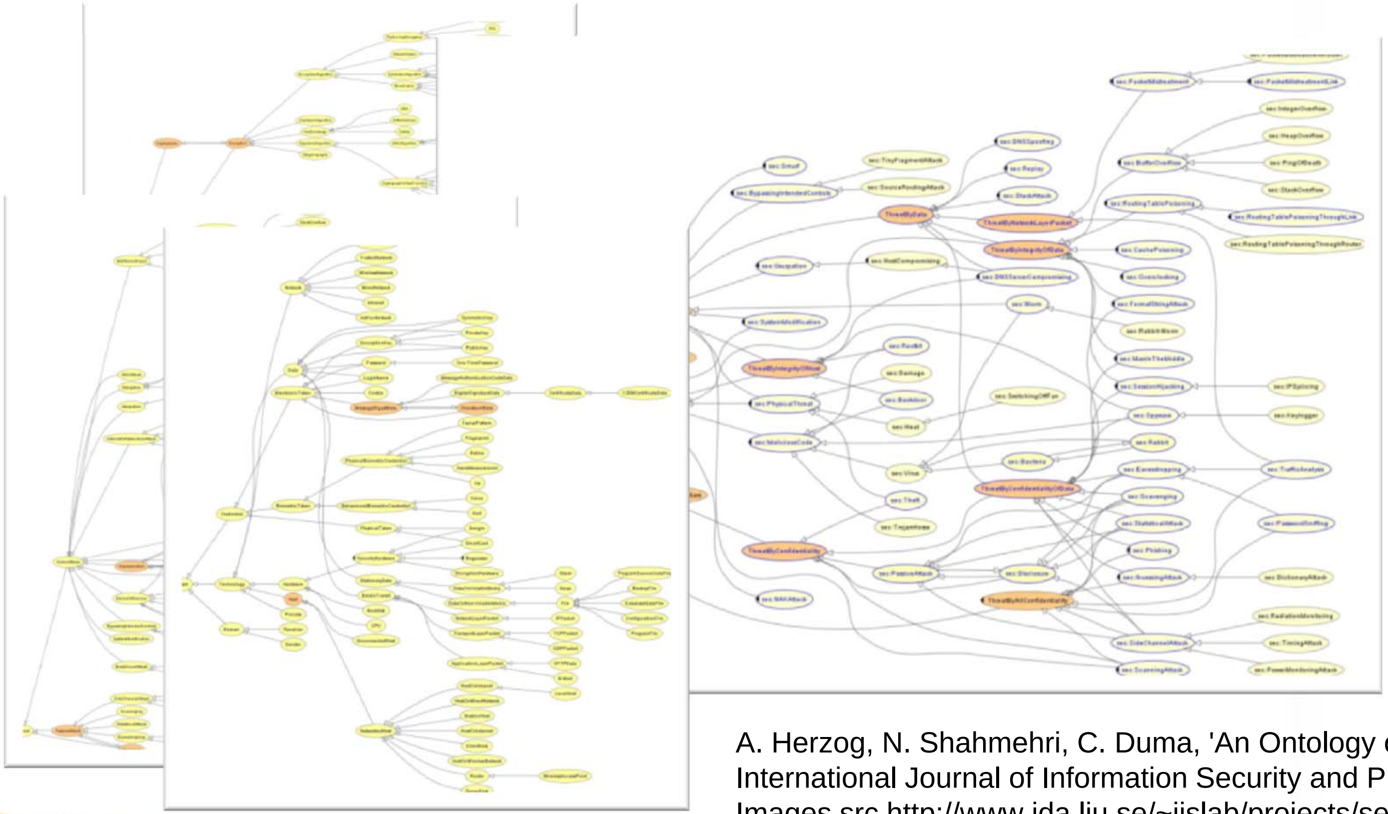
THAYER SCHOOL OF ENGINEERING AT DARTMOUTH

A Practical BKO System

BKOS – HOW TO USE THEM

Step 1: Domain Taxonomy

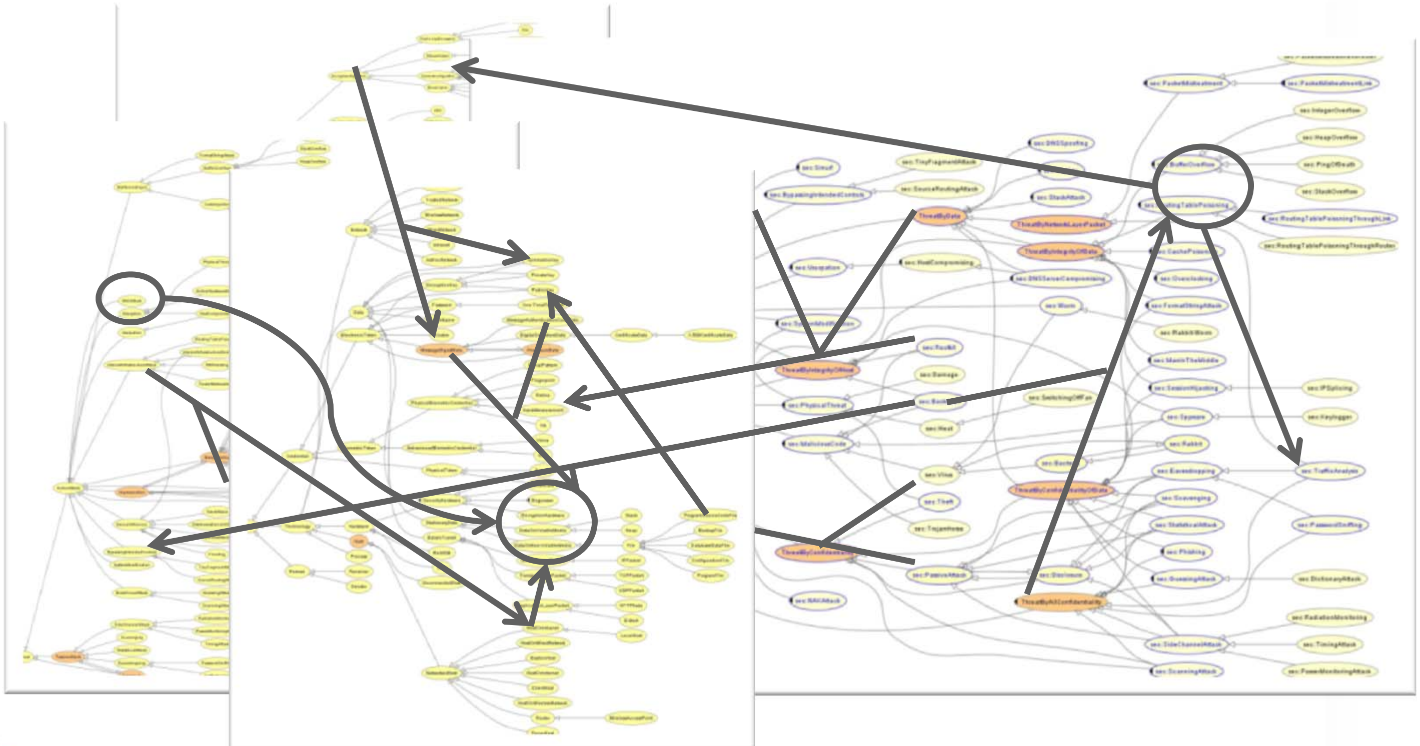
- Find or build ontolog(ies) laying out the key concepts of the domain. This will be the library's skeleton.



A. Herzog, N. Shahmehri, C. Duma, 'An Ontology of Information Security', International Journal of Information Security and Privacy, 1(4):1-23, 2007. Images src http://www.ida.liu.se/~iislab/projects/secont/

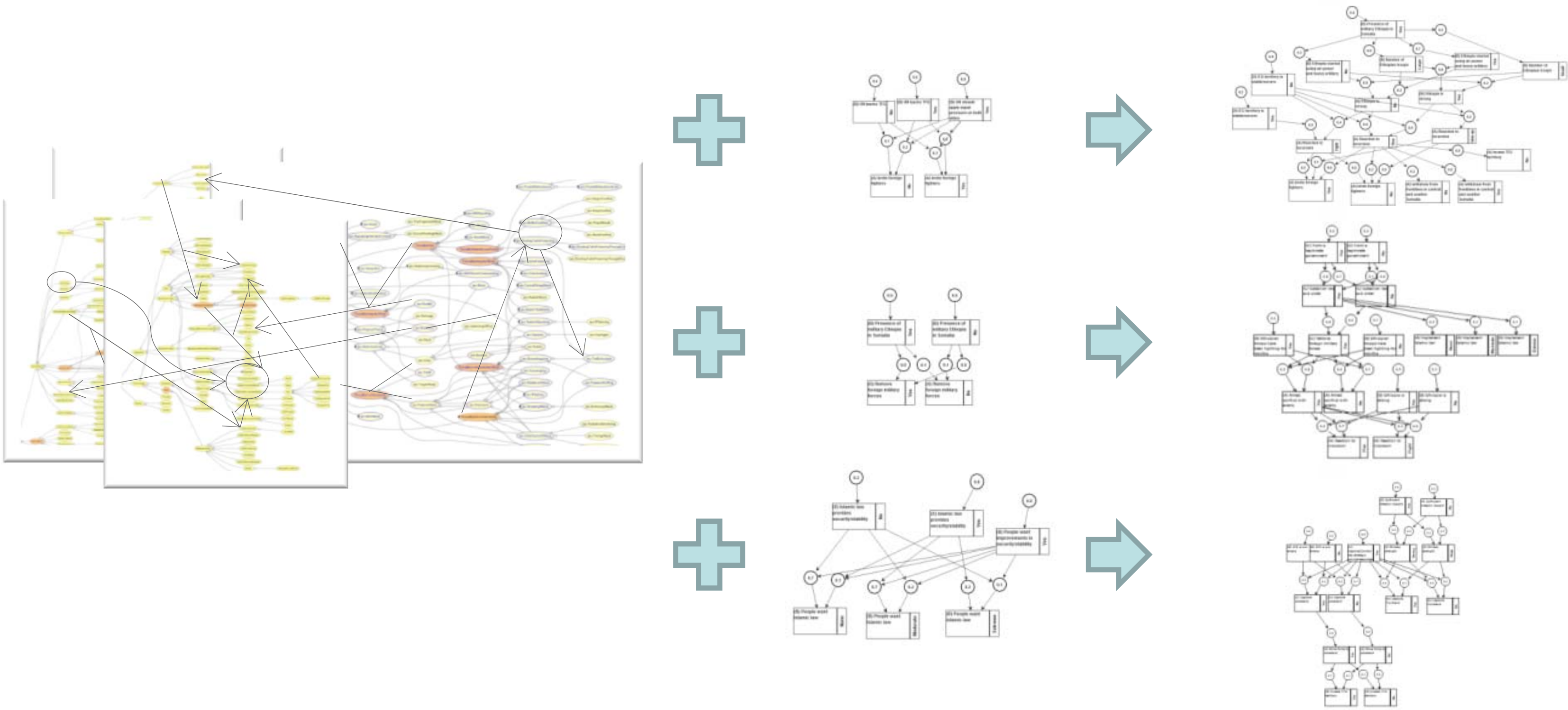
Step 2: Collect Domain Knowledge

- Fill in the model with relationships and "if-then" rules.
- Resolve conflicts between sources formally, with fusion. No more lossy, unsound "ontology merging".



Step 3: Use it to Answer Questions

- Build small case descriptions and let the library build BKBs with all its relevant knowledge. Then analyze the BKBs.



Step 4: Grow and Update

- Use the BKO to find gaps in collective knowledge. Add to it over time.
- Exchange BKOs between groups. Fuse other perspectives with your own and see new explanations of the world emerge.

Concept Application: Sysadmin's Helper

- Expert system to detect simple attacks.
  - Duplicate expert's basic threat assessment rules… and maybe some of the complex ones too.
    - BKOs are uniquely good at this.
  - Fusion facilitates pooling of expertise over time.
- Decrease human system defenders' workload
  - Goal: handle the script kiddies and let them focus on the real threats.
  - Man-on-the-loop instead of -in-the-loop.
    - Explainability: report the system's whole reasoning chain, not just its conclusions.