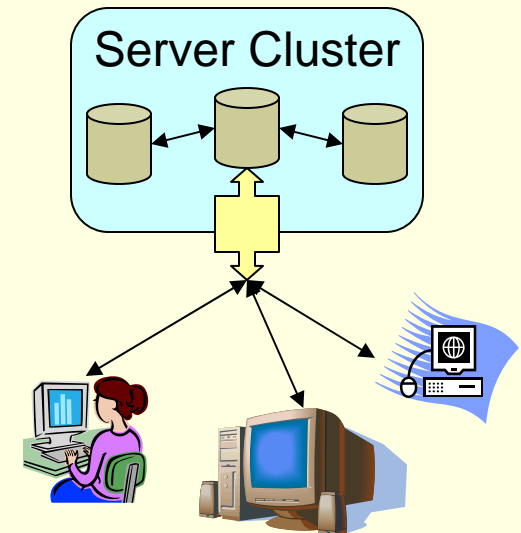

A Formal Semantics
for the
Sky.NET Simulation Engine

Timothy Bussmann
Rice University

What is Sky.NET?

- Large simulation platform
- Abstract Entities and State Transitions
- Client-Server Architecture
 - Server cluster run simulation continuously
 - Multiple clients provide input and view
- Flexibility
 - Extend and Modify running simulation
 - Conformance
- Flexibility by formalizing simulation semantics
 - Decompose simulation into Entities, Properties, Behaviors, and Rules



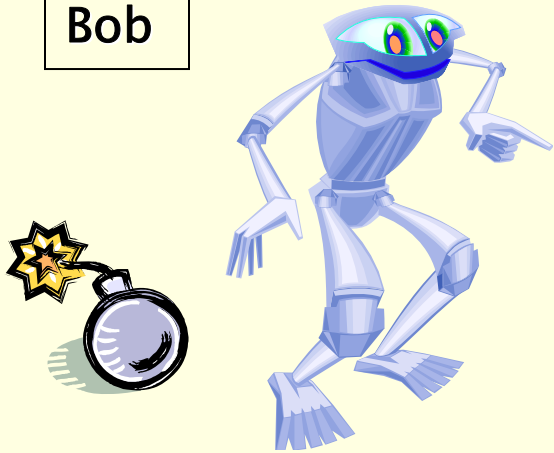
What is an Entity?

- A discrete object (can be thought of as a noun)
- Examples:
- Entity Properties:
 - Recursive data representation
 - Attribute values and Inter-Entity relationships
- Entity Behaviors (can be thought of as a verb):
 - Manipulate properties
 - Interact with other Entities

How do we formally describe entities?

What is an Entity?

Bob

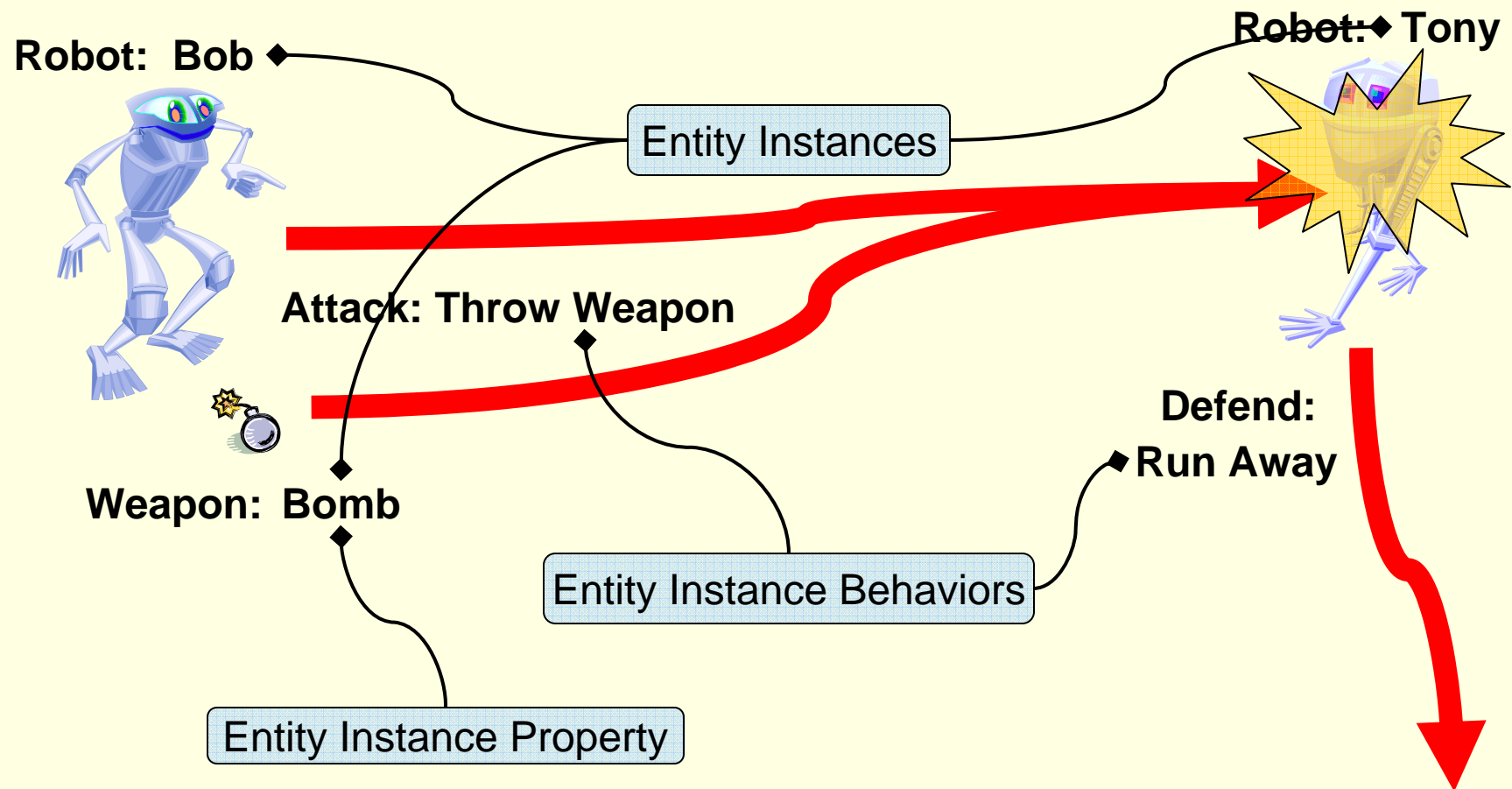


Tony



- Entities have **Properties**
 - Name
 - Weapon
- Entities have **Behaviors**
 - Attack
 - Defend

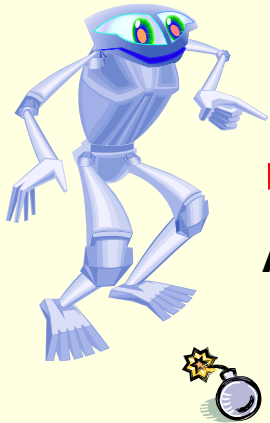
Simulation Entity-Entity Interactions



Use Type System to Generalize Entities

Simulation Entity-Entity Interactions

Robot: Bob

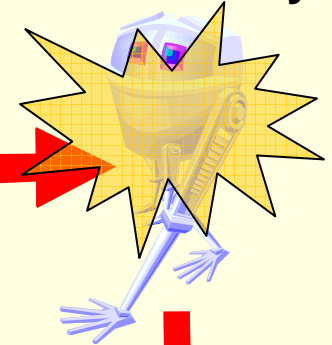


Attack: Throw Weapon

Weapon: Bomb

Rule governing attacks

Robot: Tony



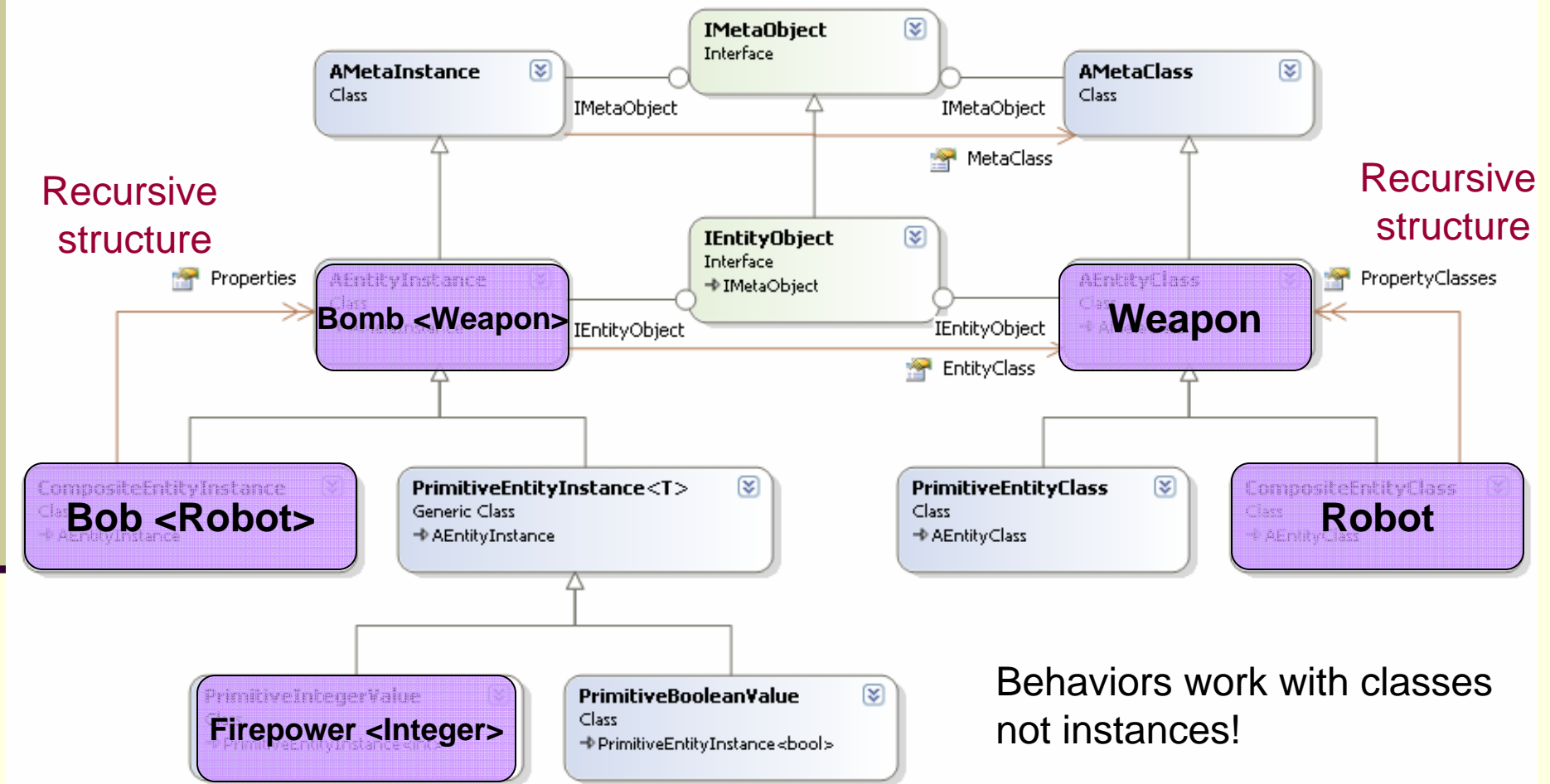
Defend:
Run Away

Rule governing movement

Use Type System to Generalize Entities

*Allows generalized behaviors and rules to be defined
→ greater flexibility and extensibility*

Classes and Entities



Recursive structure

Recursive structure

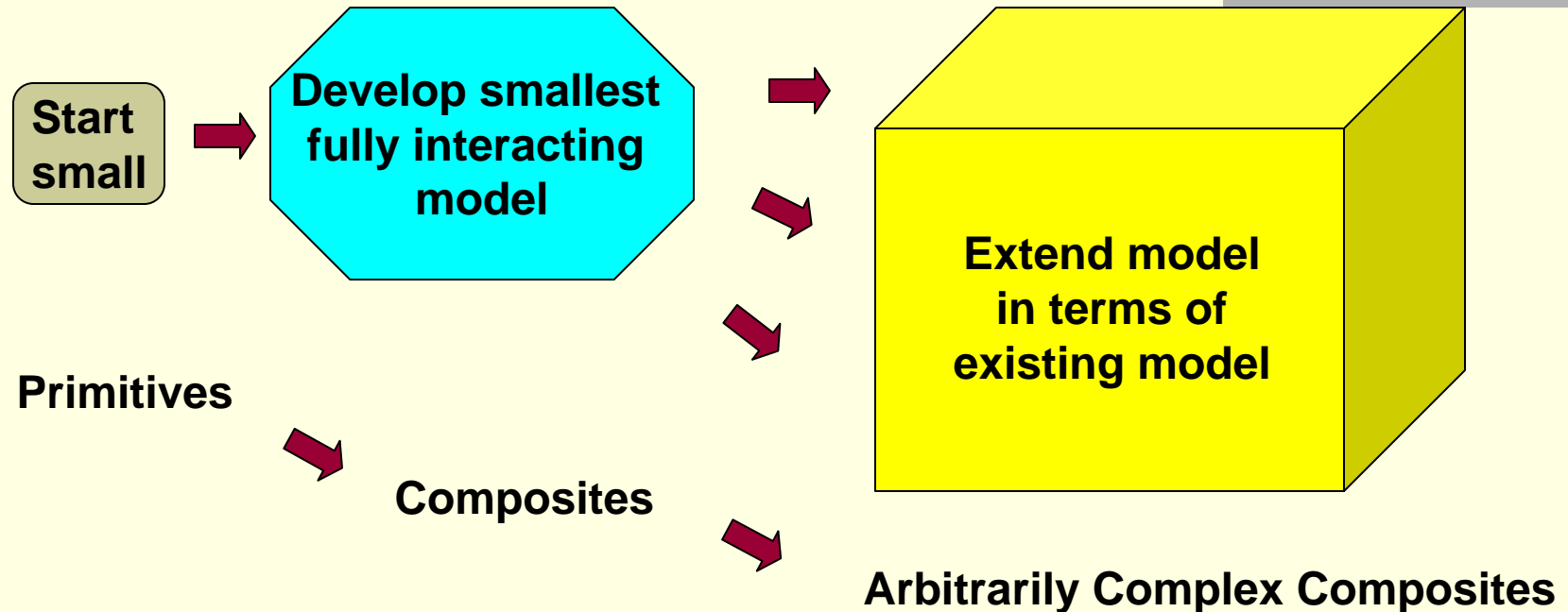
Behaviors work with classes not instances!

Primitive Entities

Simulation Flexibility

- How does a predefined Entity interact with a newly injected *arbitrary* Entity?
- Bottom-Up approach:
 - Basic Primitive Entity and Behavior Types
 - Entities then composed of primitives
 - Arbitrarily complex Entity composition

Modeling Approach: Bottom-Up



- *Existing semantics constrained and verified*
- *Only need to specify and test additions*
- *Changeable at runtime*

Another Example

■ TelecomEntity

- Properties: Capital, Infrastructure, Services
- Behaviors: Build, CollectRevenue



■ CapitalEntity

- Properties: Integer



■ InfrastructureEntity

- Properties: Location, OperatingServices
- Behaviors: OperateService, Upgrade

