**ADS 2005**

**San Diego, CA April 3-7, 2005**

# Discrete-Event Multimodels and their Agent-supported Activation and Update

Levent Yilmaz
M&SNet: Auburn M&S Laboratory
Computer Science & Software Engineering
Auburn University, Auburn, AL 36849
http://www.eng.auburn.edu/~yilmaz

Tuncer I. Ören
M&SNet: Ottawa Center of the MISS
University of Ottawa
Ottawa, ON, Canada
http://www.site.uottawa.ca/~oren

# Goals

- to expand our horizons in multimodel formalism by exploring possible types of multimodels in terms of a comprehensive taxonomy.
- to present runtime dynamic model updating with multimodels as a potential strategy in dealing with uncertainty and multi-stage phenomena.
- to lay out the conceptual foundations for the realization of a specific subset of multimodels (relevant to conflict simulation) using the **agent paradigm.**

# Plan

- Motivation
- Background on Multimodel Formalism
- Expanding Our Horizon: A Taxonomy of Multimodels
- Dynamic Model Update within Multimodels
  - Why is Model Update Important?
  - Requirements for Dynamic Model Update
- A Macro Architecture for Multimodels – Extending DEVS
- Agent-support for Submodel Update and Activation
- Conclusions

# Motivation: Dynamic Model and Simulation Update for Extensible Simulations

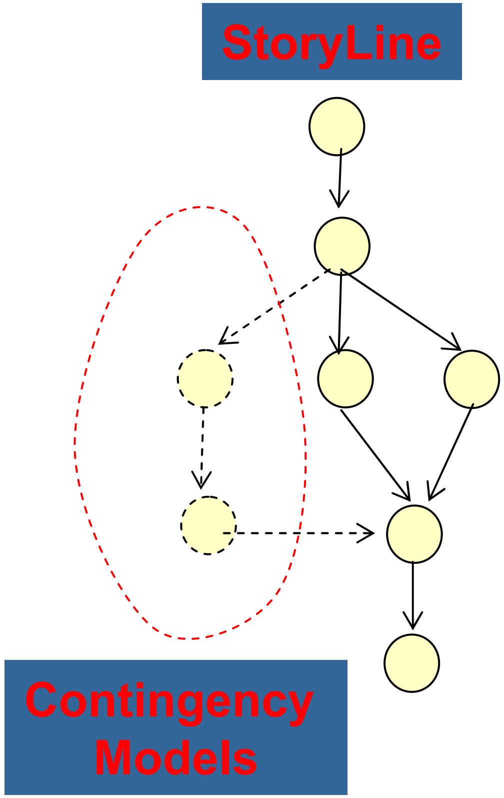
- For most realistic phenomena, the nature of the problem changes as the simulation unfolds.

- Our knowledge about the problem being studied may not be captured by any single model or experiment. Instead, the available knowledge is viewed as being contained in the collection of ensemble of models that are plausible given what is known and what is learned during experimentation.

- Dealing with uncertainty is paramount to analyzing complex evolving phenomena. Adaptivity in simulations and scenarios is necessary to deal with emergent conditions in a flexible manner.

# Example: Adaptive Experience Management in Strategic Leadership Training with Contingency Models

**Scenario:** An inspection team under the command of the team of participants is at a weapons storage site in a fictional city. The inspection team discovers that weapons from the site are missing and that a hostile crowd is forming around them. As the inspection team radios for help, the members of the command staff must prepare and launch a rescue operation. Evidence begins to mount that the weapons were stolen by paramilitary troops who are motivating the hostile crowd. As additional paramilitary troops stream into the town, the command staff must overcome a series of obstacles in order to rescue the inspection team without incident or injury – A scenario from (Gordon and Iuppa 2003).



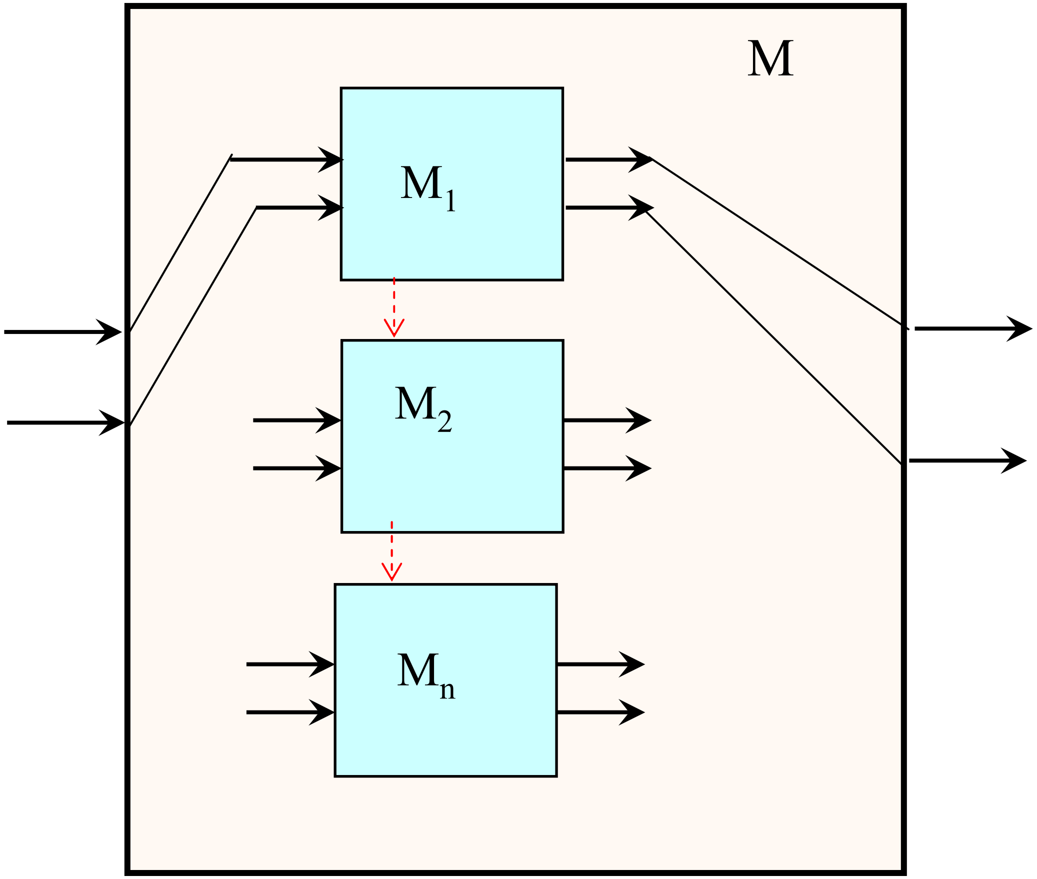

- Can we foresee all moves in a conflict? What if the original training scenario did not foresee a trainee decision that can result in civilian casualty?

# The multimodel concept is not new……….

- A multimodel is a modular model that subsumes multiple submodels that together represent the behavior of a phenomenon.
- Originally developed by Ören (1987, 1991) as a generalization of discontinuity in piecewise continuous systems.
- Influenced the development of combined simulation, which entails the integration of continuous and discrete-event simulations within the same system description.
- Multiformalism specification developed by Praehofer (1992) extended DEVS (Zeigler et al. 2000) formalism to provide a simulation environment for combined continuous/discrete-event modeling for hybrid simulation.
- Fishwick and Zeigler (1992) developed a FSA-controlled multimodel to simulate qualitative dynamics of physical systems.
- Davis and Bigelow (2002) define multiresolution as building a single model, a family of models, or both, to describe the same phenomenon at different levels of resolution in a mutually consistent way.

# A type of multimodel: metamorphic model - (e.g., egg, larva, pupa, butterfly)

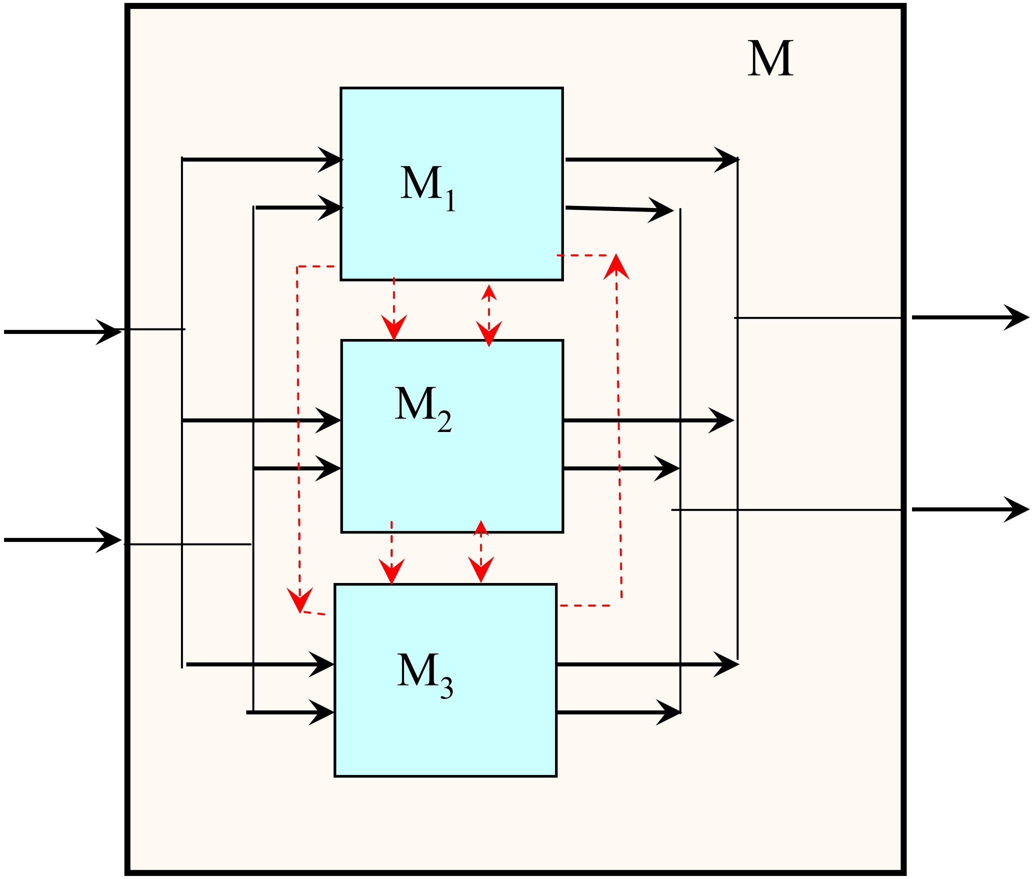
There is a predefined sequence for the alternate models.



# Another type of multimodel: multiaspect model - (e.g., ice, water, vapor)

More than one alternate model can exist at the same time

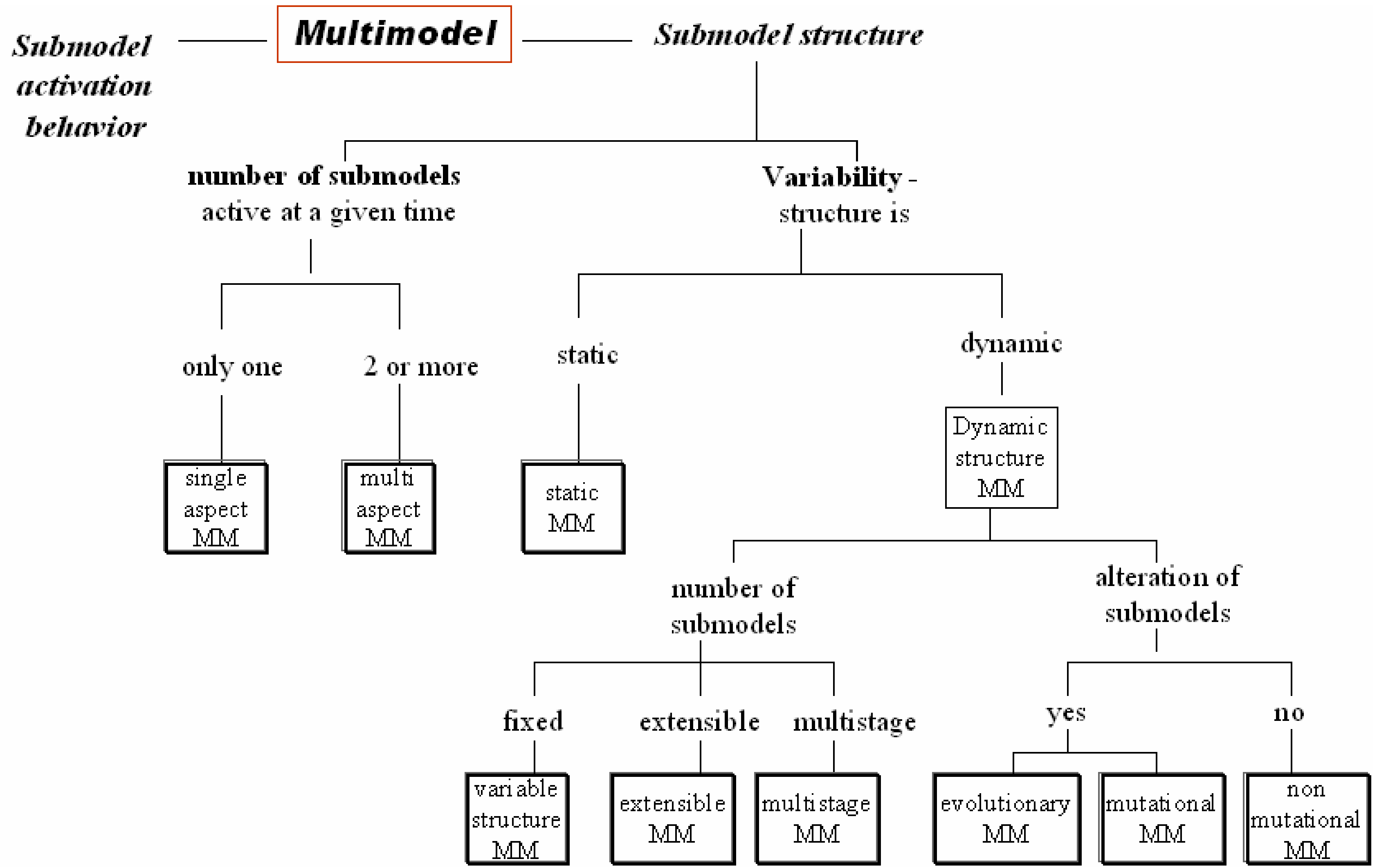
with possible flows of entities (e.g., mass) between submodels



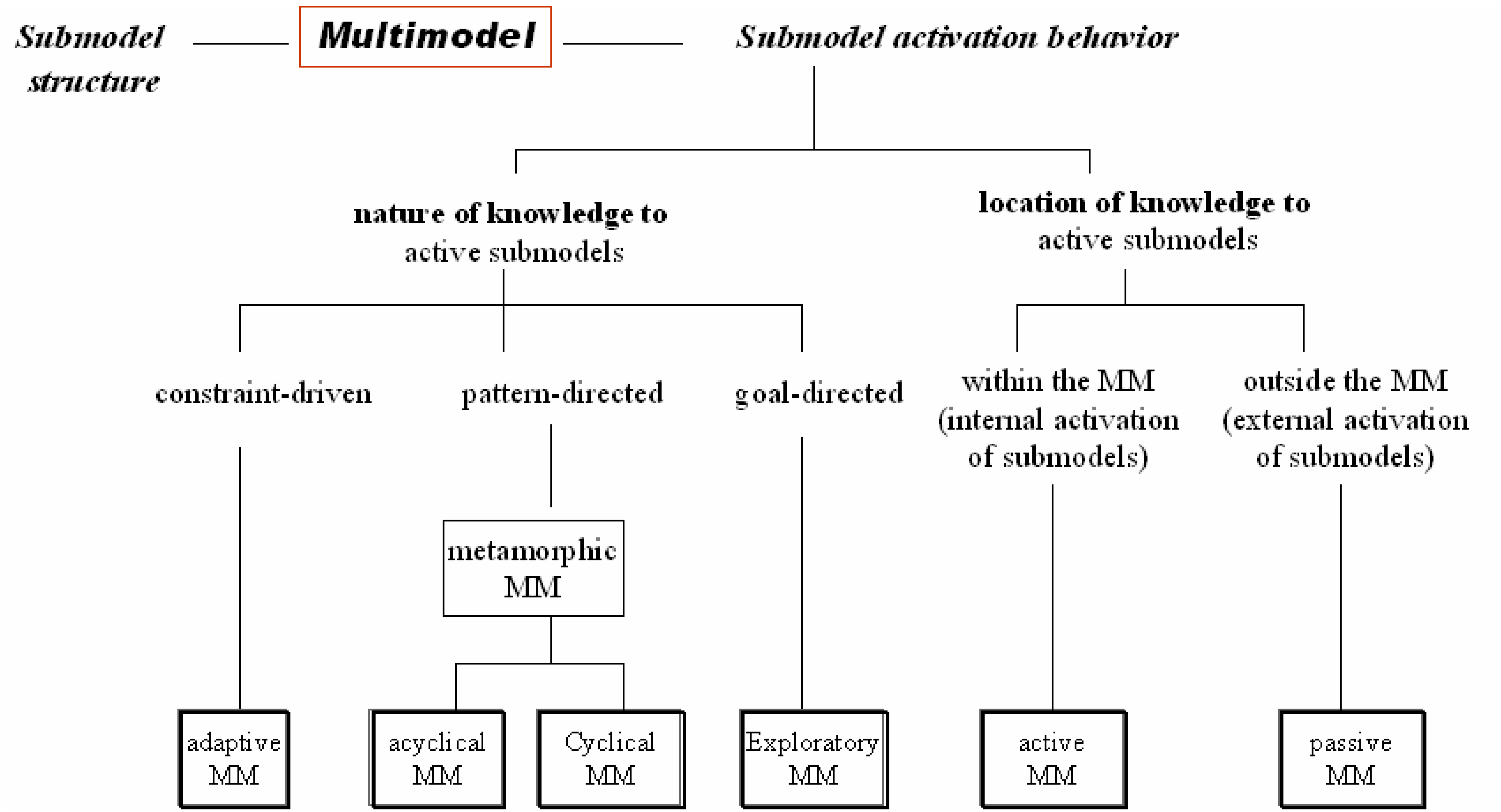
# Plan

- Motivation
- Background on Multimodel Formalism
- Expanding Our Horizon: A Taxonomy of Multimodels
- Dynamic Model Update within Multimodels
  - Why is Model Update Important?
  - Requirements for Dynamic Model Update
- A Macro Architecture for Multimodels – Extending DEVS
- Agent-support for Submodel Update and Activation
- Conclusions

# Taxonomy of Multimodels (Submodel Structure)



# Taxonomy of Multimodels (Submodel Activation Behavior)



- **Why and When Dynamic Model Update is Needed?**

Changing Scenarios: In most realistic social phenomena, the nature of the problem changes as the simulation unfolds.

Ensembles of Models: Our knowledge about the problem (i.e., conflict) being studied may not be captured by any single model or experiment (Bankes 1999)

Uncertainty: Adaptivity in simulations and scenarios is necessary to deal with emergent conditions for evolving systems in a flexible manner.

Exploration: As simulations of complex phenomena are used to aid intuition, dynamic run-time simulation composition will help identify strategies that are flexible and adaptive.

# Requirements for Model Updating

| Requirement | Objective |
| --- | --- |
| **Discovery and Location** | Model (re)placement must be initiated, either internally by the federation or externally. New models must be discovered, located, and activated (pulled) for inclusion in the federation. |
| **Integrity** | |
| **Instantiation** | |
| **State (re)construction** | |
| **Rebinding** | |

# Requirements for Model and Simulation Updating

| Requirement | Objective |
| --- | --- |
| **Discovery and Location** | |
| **Integrity** | The consistency of models undergoing (re)placement needs to be preserved. The event scheduling and simulation protocol need to be coordinated to facilitate interleaving of model replacement (update) tasks with the simulation events. |
| **Instantiation** | |
| **State (re)construction** | |
| **Rebinding** | |

# Requirements for Model and Simulation Updating

| Requirement | Objective |
|---|---|
| **Discovery and Location** | |
| **Integrity** | |
| **Instantiation** | The new model must be dynamically loaded and linked into the run-time environment |
| **State (re)construction** | |
| **Rebinding** | |

# Requirements for Model and Simulation Updating

| Requirement | Objective |
| --- | --- |
| **Discovery and Location** | |
| **Integrity** | |
| **Instantiation** | |
| **State (re)construction** | The state of the simulation must be constructed or at least continue from a specific state after an update operation. |
| **Rebinding** | |

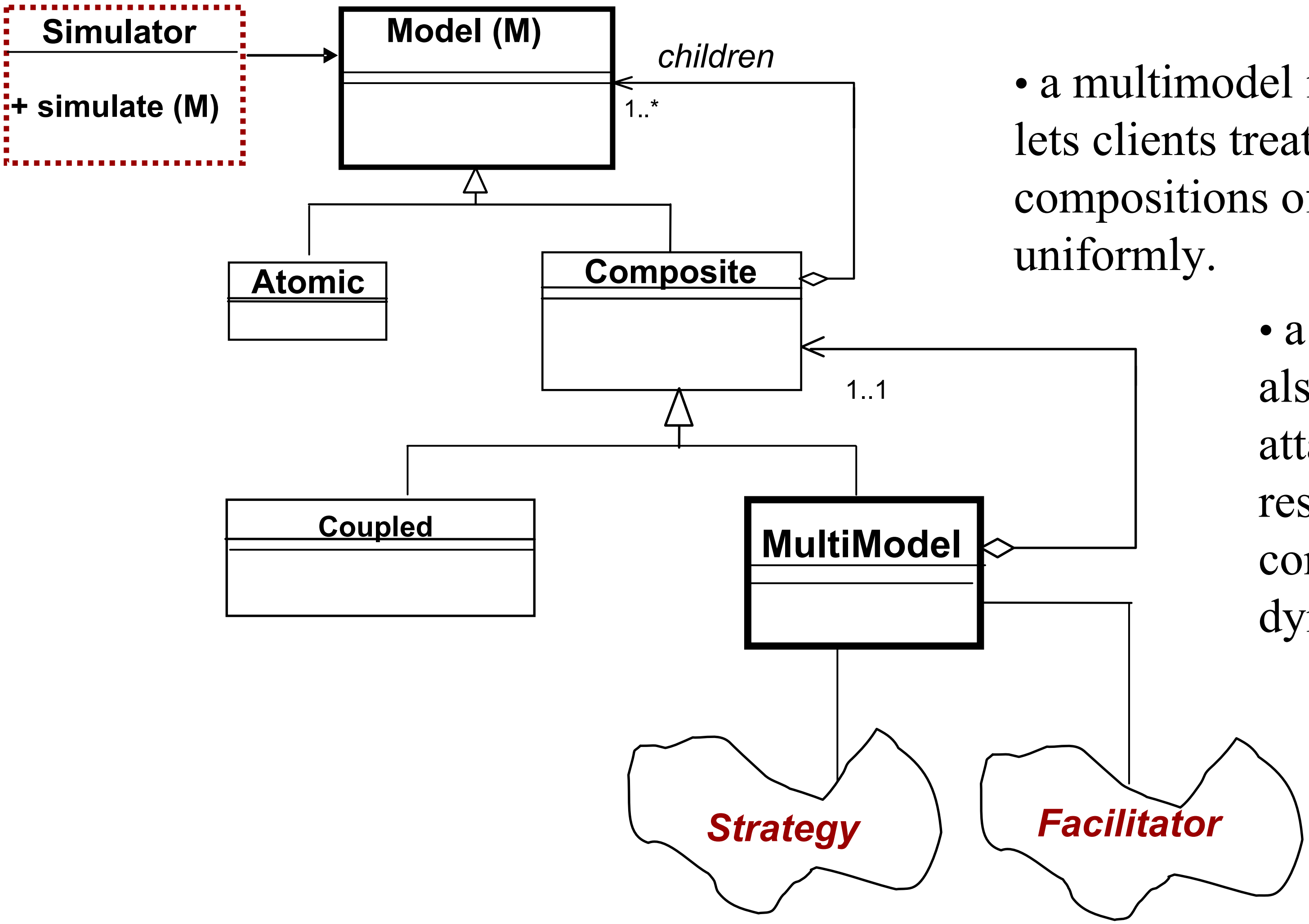
# Requirements for Model and Simulation Updating

| Requirement | Objective |
|---|---|
| **Discovery and Location** | |
| **Integrity** | |
| **Instantiation** | |
| **State (re)construction** | |
| **Rebinding** | Once a model is loaded and linked to the run-time environment, the simulator needs to be bound to the new model. This requires novel model and simulator decoupling strategies that avoid persistent or permanent connections. |

# Plan

- Motivation
- Background on Multimodel Formalism
- Expanding Our Horizon: A Taxonomy of Multimodels
- Dynamic Model Update within Multimodels
  - Why is Model Update Important?
  - Requirements for Dynamic Model Update
- A Macro Architecture for Multimodels – Extending DEVS
- Agent-support for Submodel Update and Activation
- Conclusions

# A Conceptual Basis for Multimodels

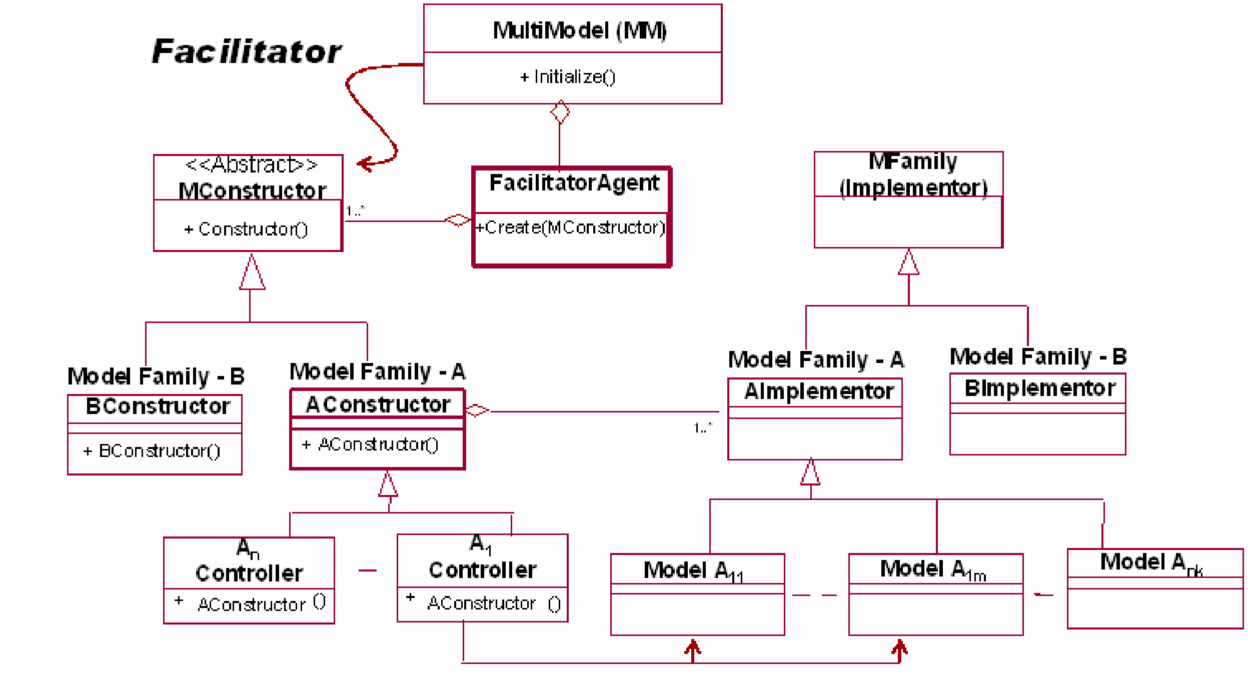



- a multimodel is a composite that lets clients treat atomic models and compositions of submodels uniformly.

- a multimodel is also a wrapper that attach additional responsibility to a composite model dynamically.

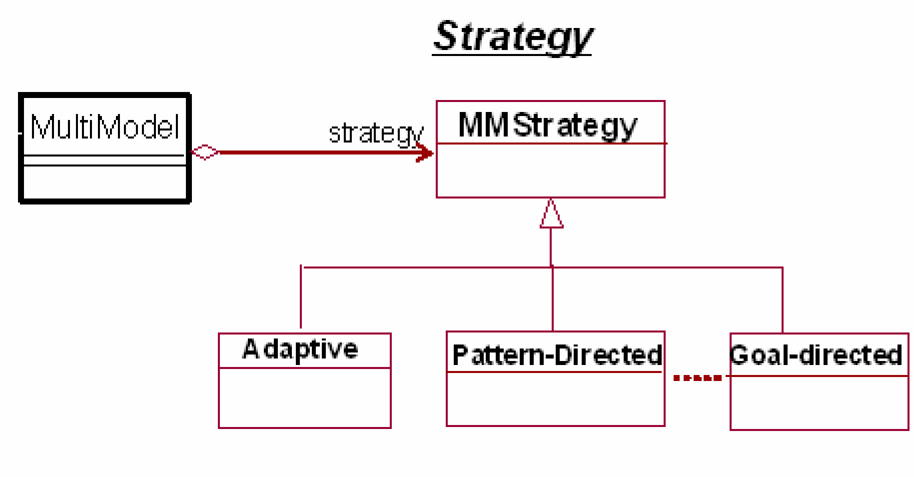
# Facilitator needs to facilitate seamless exchange of submodels…….

• a multimodel should be independent of how its submodels are created, composed, and represented.
• a multimodel should be configured with one of multiple families of submodels.
• a family of related submodels is designed to be used together, and we need to enforce this constraint.



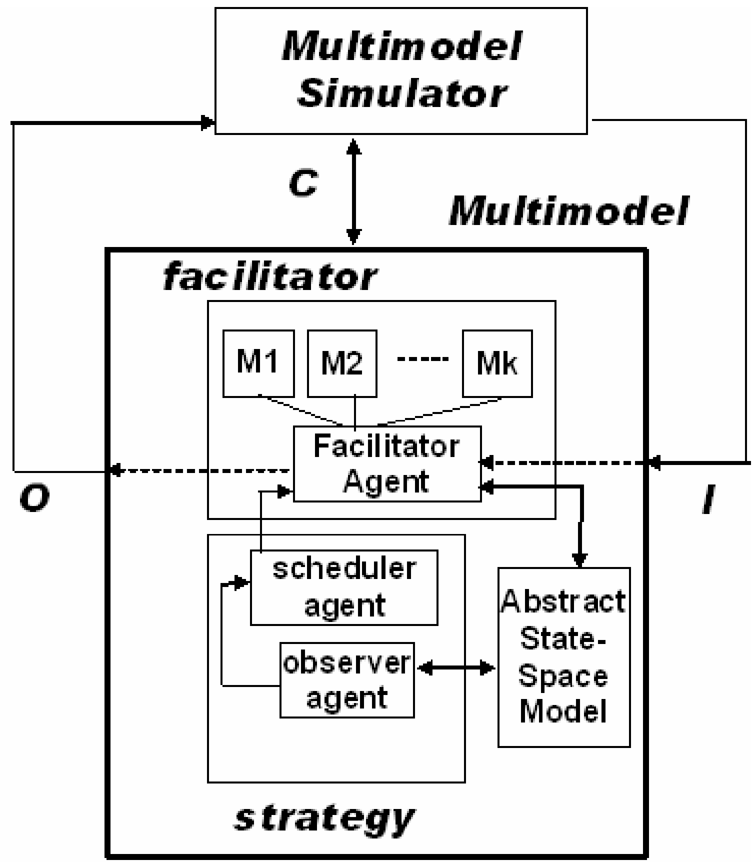
# Strategy component….

• Depending on the activation policy of submodels, multimodels are classified into various types, including constraint-driven, pattern-directed, goal-driven multimodels.
• Each one of these multimodel types requires a distinct protocol and mechanism for activating submodels.




Strategy component: Define a family of protocols, encapsulate each one, and make them interchangeable. Strategy lets the protocol vary independently from the clients that use it.

# The macroarchitecture of a multimodel….




- A scheduler agent can have planning capability that drives a simulator with alternative models by updating facilitator.

- Observer agents need to notify scheduler agent of the change
  - The number of observers may not be known. Each observer is responsible for specific states of interest in the application domain.
  - The strategy requires simulator and observer to be loosely coupled

# Conclusions

- **Multimodeling** formalism inspired the development of several methodologies, including
  - combined simulation [Preahofer 1992],
  - FSA-controlled multimodeling [Fishwick and Zeigler 1992], and
  - MOOSE [Cupert and Fishwick 1997].
- Yet, we are still scratching the surface of what is possible with the very basic dynamic model update concept.
- **Agent-based** approach presents a reasonable strategy to realize multimodel simulators.
- However, more effort is needed to identify and implement intricate details of **observing/interpreting** the simulation state, **reason** to qualify models for update, facilitate run-time model **rebinding**, and **plan** for goal-directed activation of submodels.