



Task Modeling for Collaborative Environments

Maik Wurdel
PhD Candidate
maik.wurdel@uni-rostock.de

Software Engineering Group
GRK MuSAMA
University of Rostock

1st Jointed Workshop on Interactive Systems
Putbus - 2008-06-27





Introduction

Existing Approaches

CTML

❖ Preconditions & Effects

❖ Task Life Cycle

❖ Simulation

Tool Support

Conclusion & Future Work

Outline

- Introduction
- Existing Approaches
- The Collaborative Task Modeling Language
 - Rationale
 - Task Life Cycle
 - Preconditions & Effects
- Tool Support
- Conclusion & Future Work





Introduction

Existing Approaches

CTML

❖ Preconditions & Effects

❖ Task Life Cycle

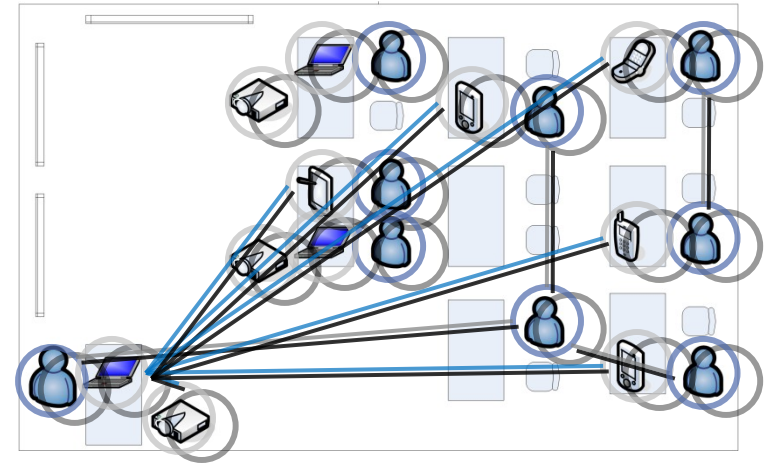
❖ Simulation

Tool Support

Conclusion & Future Work

Collaborative Environments

- Actors
 - Fulfilling roles
 - Performing tasks
- Devices
 - Personal
 - Stationary
- Cooperation
- Dependencies
 - Actors & devices



Enormous Complexity





UIs for Collaborative Environments

Introduction

Existing Approaches

CTML

❖ Preconditions & Effects

❖ Task Life Cycle

❖ Simulation

Tool Support

Conclusion & Future Work

- Hand-crafted vs. model-based
- Model-based UI development



- Task as building block
 - Understanding collaboration
 - Combining individual with collaborative task performance
 - Starting point for MB-UI development





Introduction

Existing Approaches

CTML

- ❖ Preconditions & Effects
- ❖ Task Life Cycle
- ❖ Simulation

Tool Support

Conclusion & Future Work

Existing Task Modeling Languages

- No current notation fulfills our requirements
- But..
- Cooperation
 - Cooperative CTT
- Effect of task execution
 - TaoSpec
- Life cycle of tasks
 - State Chart as Life Cycle
 - Executable Task Model
 - WebTask Model





Introduction

Existing Approaches

CTML

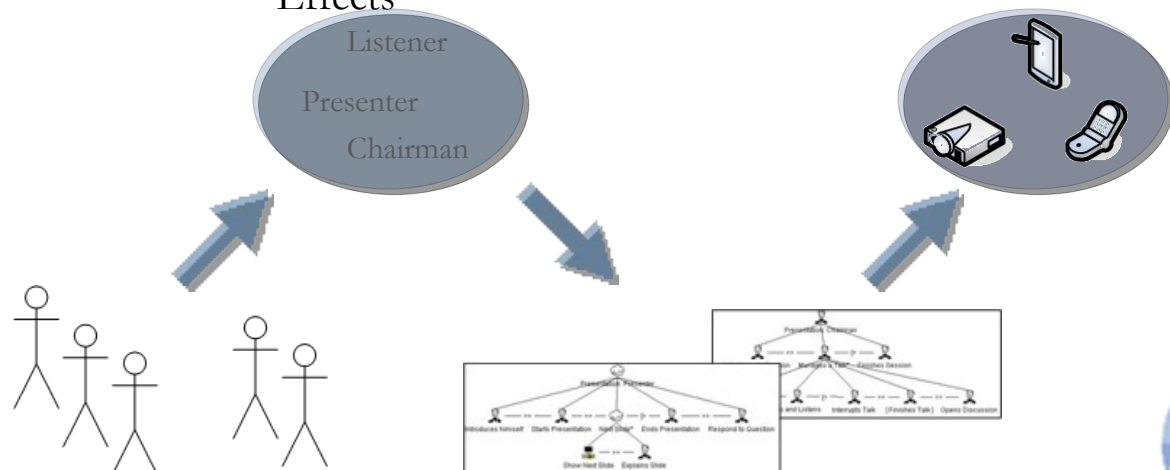
- ❖ Preconditions & Effects
- ❖ Task Life Cycle
- ❖ Simulation

Tool Support

Conclusion & Future Work

The Collaborative Task Modeling Language (CTML)

- Actors, Roles, Tasks, Environment
 - Actors fulfilling roles
 - Roles are associated with task models
 - Task execution changes system state
- Task enriched with
 - Preconditions
 - Effects





Introduction

Existing Approaches

CTML

❖ **Preconditions & Effects**

❖ Task Life Cycle

❖ Simulation

Tool Support

Conclusion & Future Work

CTML – Preconditions & Effects

- Preconditions & effects are used to model cooperation and the relation of tasks to the system state
- Precondition
 - State of system required for starting a task
- Effect
 - State change after performing a task

When to evaluate preconditions?

When to apply effects?

Need for a precise life cycle of tasks





Introduction

Existing Approaches

CTML

❖ Preconditions & Effects

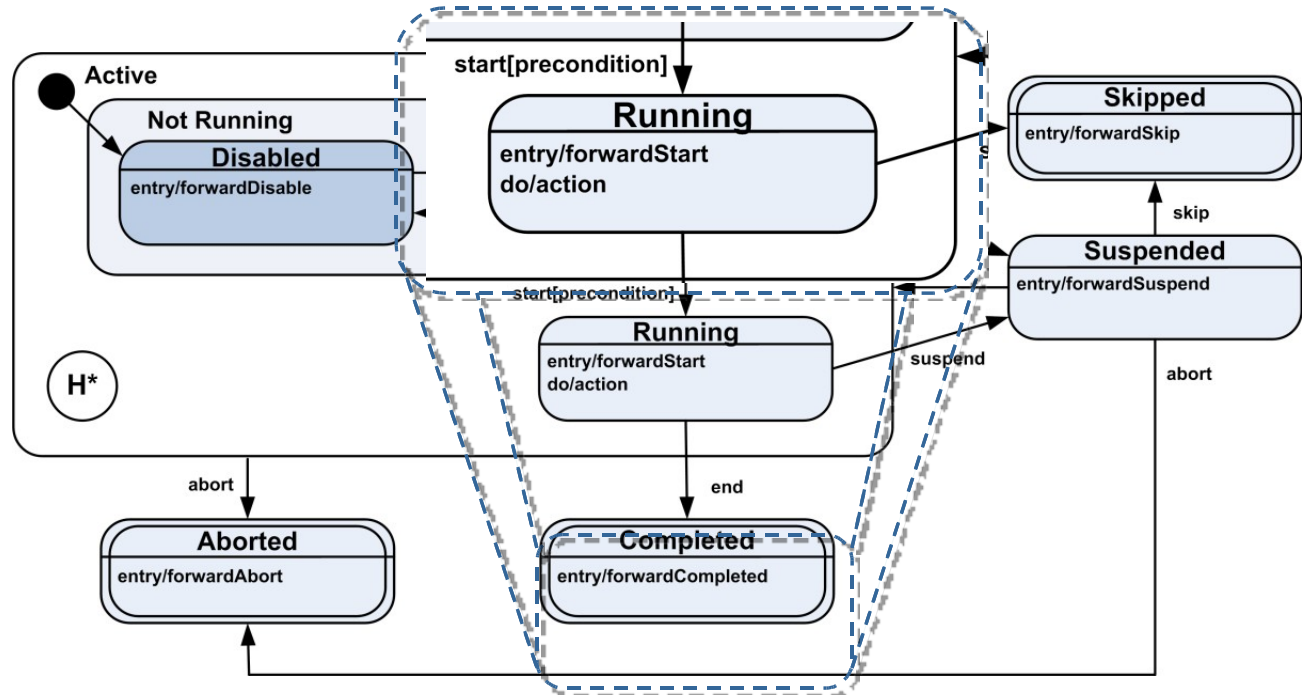
❖ **Task Life Cycle**

❖ Simulation

Tool Support

Conclusion & Future Work

Task Life Cycle





Introduction

Existing Approaches

CTML

❖ Preconditions & Effects

❖ **Task Life Cycle**

❖ Simulation

Tool Support

Conclusion & Future Work

Precondition

#	Role	Task	Precondition
(1.)	<i>Presenter</i>	<i>StartsPresentation</i>	<i>Chairman.oneInstance.AnnouncesTalk.completed</i>
(2.)	<i>Listener</i>	<i>AsksQuestion</i>	<i>Chairman.oneInstance.OpensDiscussion.completed</i>
(3.)	<i>Chairman</i>	<i>Wraps-UpSession</i>	<i>Presenter.allInstances.EndsPresentation.completed</i>

Effects

#	Role	Task	Effect
(1.)	<i>Presenter</i>	<i>EndsPresentation</i>	<i>this.presented = true</i>
(2.)	<i>Presenter</i>	<i>StartsPresentation</i>	<i>@RoomsProjector.connectedWith = this.notebook</i>





Introduction

Existing Approaches

CTML

❖ Preconditions & Effects

❖ Task Life Cycle

❖ **Simulation**

Tool Support

Conclusion & Future Work

CTML Simulation

- Each actor uses its task model according to the role
 - An “instance” of the task model is created
- Each instance task model is represented by a set of communicating task state charts
- Concurrent task simulation of all actors
 - Synchronization between actors is defined in terms of precondition & effects





Introduction

Existing Approaches

CTML

❖ Preconditions & Effects

❖ Task Life Cycle

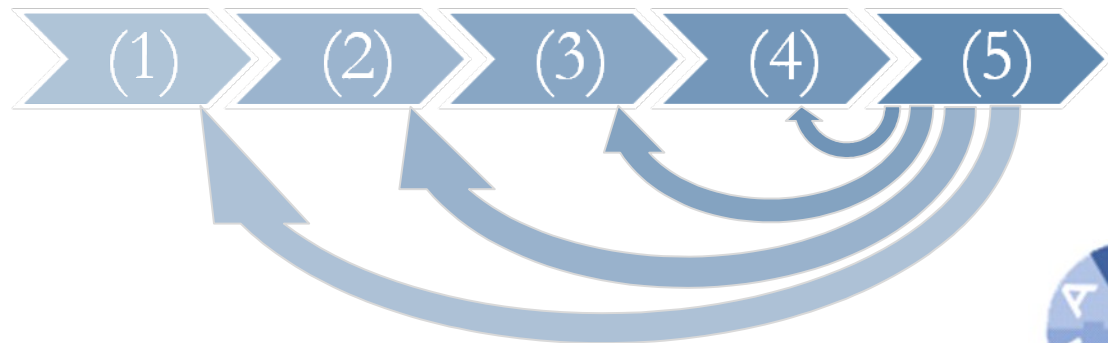
❖ Simulation

Tool Support

Conclusion & Future Work

Development Life Cycle for CTML

- (1) Definition of roles and corresponding task models
- (2) Animation and validation of these sub-specifications
- (3) Specification of actors, environment & devices
- (4) Annotation of tasks with preconditions and effects
- (5) Animation and validation of these entire specifications





Introduction

Existing Approaches

CTML

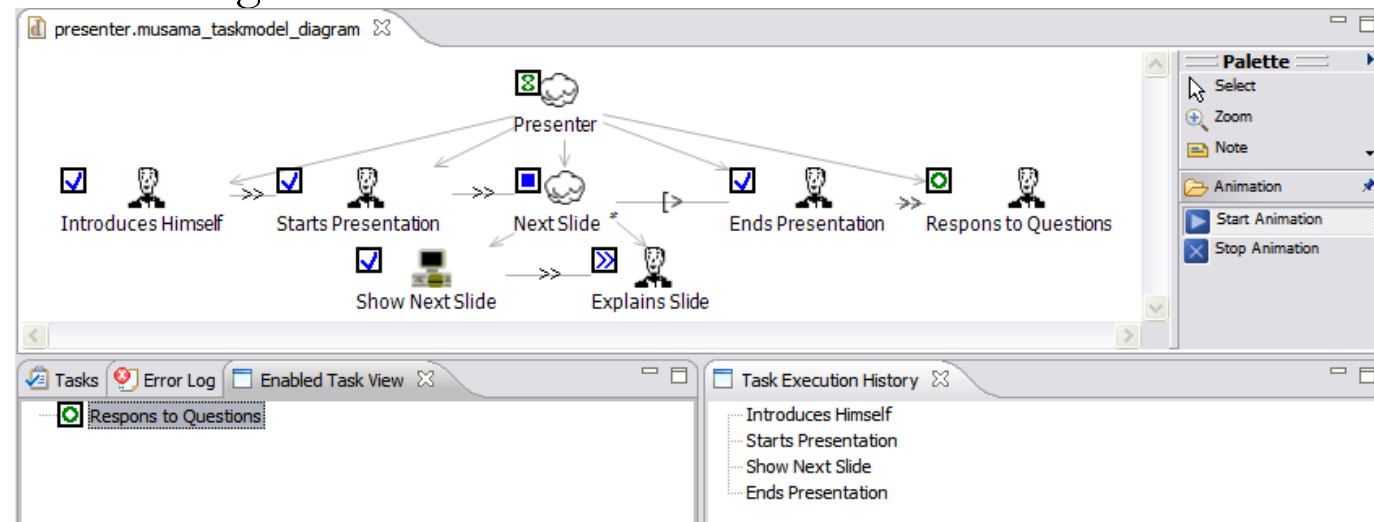
- ❖ Preconditions & Effects
- ❖ Task Life Cycle
- ❖ Simulation

Tool Support

Conclusion & Future Work

Tool Support

- Single Model Simulation





Introduction

Existing Approaches

CTML

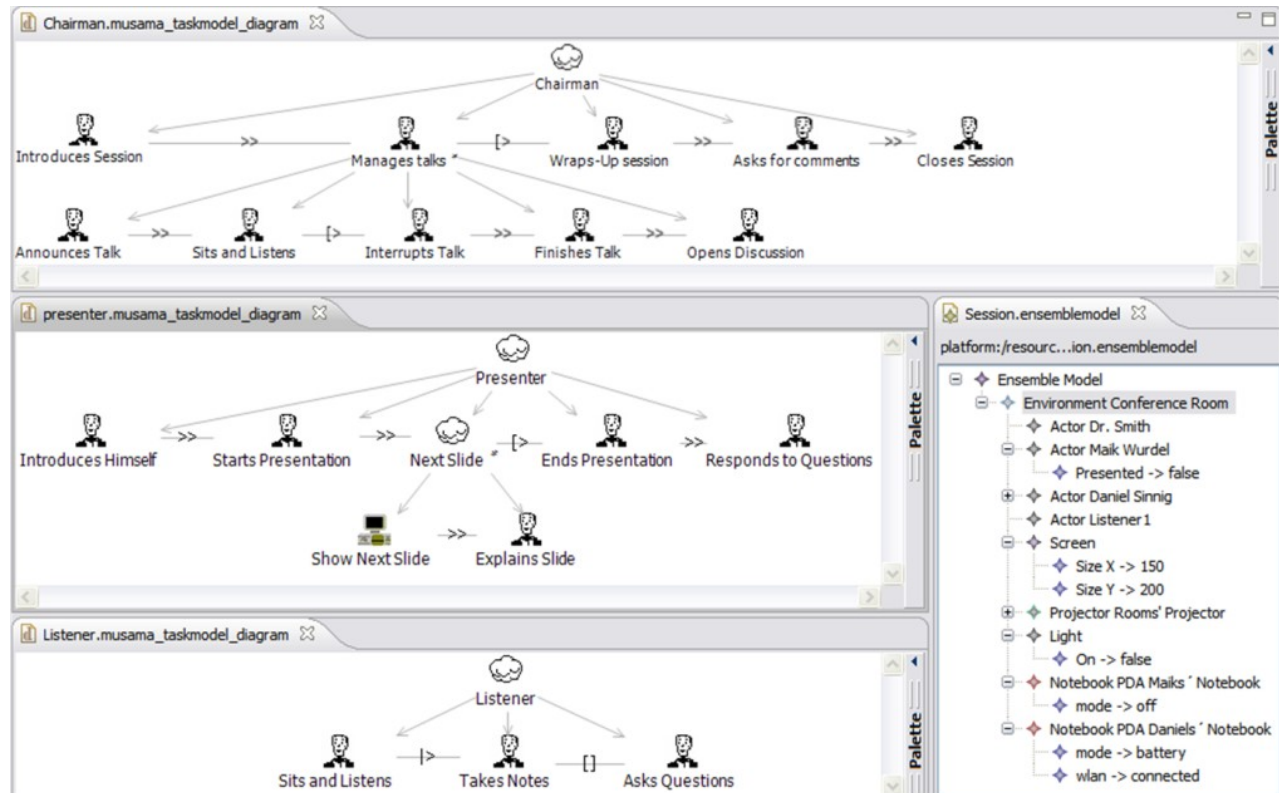
- ❖ Preconditions & Effects
- ❖ Task Life Cycle
- ❖ Simulation

Tool Support

Conclusion & Future Work

Tool Support

- CTML Editor





Introduction

Existing Approaches

CTML

- ❖ Preconditions & Effects
- ❖ Task Life Cycle
- ❖ Simulation

Tool Support

Conclusion & Future Work

Tool Support

The screenshot displays a task modeling tool interface with four user-specific task lists and a global task list on the right. Each task is represented by a colored icon (green, blue, or red) and a text label. The 'Show Next Slide' task is highlighted in yellow in the global list.

- Dr. Smith:**
 - >> Chairman
 - Introduces Session
 - >> Manages talks
 - AnnouncesTalk
 - Sits and Listens
 - Interrupts Talk
 - Finishes Talk
 - Opens Discussion
 - Wraps-Up session (Presenter.allInstances.EndsF
 - Asks for comments
 - Closes Session

- Maik Wurdel:**
- >> Presenter
 - Introduces Himself
 - Configure Equipment
 - Starts Presentation (Chairman.oneInstance.Announ
 - >> Next Slide
 - Show Next Slide
 - Explains Slide
 - Ends Presentation
 - Responds to Questions
- Daniel Sinnig:**
- >> Presenter
 - Introduces Himself
 - Configure Equipment
 - Starts Presentation (Chairman.oneInstance.Announ
 - >> Next Slide
 - Ends Presentation
 - Responds to Questions
- Listener1:**
- > Listener
 - Sits and Listens
 - Takes Notes
 - Asks Questions (Chairman.oneInstance.OpensDiscu

Global Task List (Dr. Smith):

- tasks
 - Introduces Session
 - AnnouncesTalk
 - Sits and Listens
 - Interrupts Talk
 - Finishes Talk
 - Opens Discussion
 - Wraps-Up session
 - Asks for comments
 - Closes Session
- attributes
- Maik Wurdel
 - tasks
 - Introduces Himself
 - Configure Equipment
 - Starts Presentation
 - Show Next Slide
 - Explains Slide
 - Ends Presentation
 - Responds to Questions
 - attributes
- id: id1200504938008854





Introduction

Existing Approaches

CTML

❖ Preconditions & Effects

❖ Task Life Cycle

❖ Simulation

Tool Support

Conclusion & Future Work

Conclusion

- **Main Motivation:** Vast complexity of collaborative environments in terms of involved actors, tasks, and devices.
- Need for a holistic task-based specification language
- **Solution:** Collaborative Task Modeling Language
 - Formal syntax and semantics
 - Superior expressiveness
- Tool Support: The CTML Editor and Simulator





Introduction

Existing Approaches

CTML

❖ Preconditions & Effects

❖ Task Life Cycle

❖ Simulation

Tool Support

Conclusion & Future Work

Future Work

- Definition of a refinement relation:
 - Already introduced for single task models
- Domain modeling
- Physical context of task execution

Ultimate Goal:

- Definition of Development Methodology for UIs for Collaborative Environments





Thank You for Your Attention,
Discussion and Feedback.

Task Modeling for Collaborative
Environments

Maik Wurdel
Software Engineering Group
University of Rostock
maik.wurdel@uni-rostock.de

