Workflow Simulation in a Virtual World

Keith Perkins

Masters Thesis Defense Presentation
3 May 2011

Thesis Committee
Dr. Craig Thompson (adviser)
Dr. Brajendra Panda
Dr. Gordon Beavers
Outline

- Introduction
  - Background
  - Building a Workflow Simulation System
  - Evaluation
  - Conclusions and Future Work
Context

- In everyday life, humans engage in hundreds of rote activities, from brushing their teeth to driving home to performing heart operations.
- We call these activities workflows by which we mean coordinated collections of tasks performed by actors and involving props.
- This thesis explores using 3D virtual worlds to model workflows.
Problem

- Currently we do not have computational models of everyday workflow activities – we don’t have ways to represent, simulate (execute), recognize, catalog, combine or optimize such workflows.

- Currently 3D virtual worlds like Second Life do not provide support for workflows.

- If computers could process workflows, they might help humans in many areas: computers providing help in workflow monitoring and training with applications in supply chains, healthcare, and entertainment; eventually, robots and smart objects interacting with humans.
Objective

- Develop a representation for workflows
- Develop a simulation system for executing workflows in a 3D virtual world
- Show that the system can be used to develop and test workflows.
Outline

- Introduction
- Background
  - Building a Workflow Simulation System
  - Evaluation
  - Conclusions and Future Work
Virtual Worlds

Definition: A computer generated, controlled, and rendered environment supporting multiple simultaneous users

- Evolutionary intersection of Virtual Reality and Multi User Dungeons (MUDs)
- Closely related to Massively Multiplayer Online Games (MMOGs)
- Common elements: computer-rendered 3D view, physics engines, players have in-world representations and interact with each other
Second Life

- Commercial product of Linden Lab since 2003
- 3D environment geared to social interaction and user content creation
- “Land” consists of 1000s of 256m x 256m regions
- Virtual economy with currency, enforced intellectual property rules
- Objects can be created and scripted to have actions and reactions (events)
- Client/Server Architecture – users access Second Life servers via a “client viewer” program
Everything is Alive (EiA)

- EiA is a project at University of Arkansas aimed at studying pervasive computing, Internet of Things, and a semantic world where every object has identity, meta data and can communicate with humans and other objects.
- In 2007 the EiA project began leasing the University of Arkansas island region in SL:
  - Readily available via the Internet
  - Allows content creation and scripting
  - Low cost of entry for users
  - High availability of useful information
DES and Workflow

- Discrete Event Systems – Event-driven, asynchronous, sequential relations, concurrency, conflict, mutual exclusion, deadlock, non-determinism

- Workflow Management Systems – Control system manages workflow execution by interpreting the model and directing the tasks

- Workflow Modeling – Specification language and modeling tools
Background – Petri Nets

- Created by Carl Petri (1939, 1962)
- Useful abstraction for representing parallel and distributed computing and workflows
- Based on a formal mathematical definition and have been extensively researched algorithmically and for questions of computational complexity and decidability.
Background – Petri Nets

- A basic Petri net is a directed, bipartite graph
- Can be defined as a tuple \((P, T, I, O)\) where:
  - \(P\) is a finite set of place nodes
  - \(T\) is a finite set of transition nodes
  - \(I\) is a finite set of arcs from places to transitions
  - \(O\) is a finite set of arcs from transitions to places
  - \(P \cap T = \emptyset\) and \(P \cup T \neq \emptyset\)
  - \(I \cup O \neq \emptyset\)
  - Each transition node in \(T\) has a set of input places and a (possibly empty) set of output places

Petri net Example
Background – Petri Nets

- A marked Petri net includes an mapping of places to non-negative integers which represents the number of tokens initially present in each place.
- The initial marking and transition rules define the actions of the Petri net.
- A transition is enabled and can fire if every node in its set of input places contains a token.
- Upon firing, a token is removed from each of its input places and a token is placed in each of its output places.
- When more than one transition is enabled, a non-deterministic choice decides which one fires.
- When two or more enabled transitions share an input place they are in conflict.
- A Petri net without any enabled transitions is considered dead.
- Generalized Petri nets, Prioritized Petri nets, Extended Petri nets. Prioritized and Extended Petri nets have computational power of a Turing machine.
Other Related Work

- Actor/Agent Model - Hewitt et al., Agha, Wooldridge et al., Maher et al.
- Electronic Institutions - Esteva et al, Seidel et al.
- Conceptual Modeling in 3D Virtual Worlds - Brown
Outline

- Introduction
- Background
- Building a Workflow Simulation System
- Evaluation
- Conclusions and Future Work
Workflow Simulation System

Initial Workflow Demos

- Human-controlled avatars – Supply Chain and Prescription Management
- Object-based workflows – Robot Patients and Nurses
- Avatarbot workflows – Catheterization Operation, Medical Equipment Management, Dance of the interns, Romeo and Juliet scene

Video Example
Workflow Simulation System

Architecture

- Avatarbot control program
- Leaf-level commands
- Mark 1 – monolithic, linear, hard-coded tasks
- Mark 2
  - Parallel leaf-level commands
  - Lua tasks
  - Workflow control module & Petri net interpreter
Methodology

Steps for developing a workflow simulation

1. Define workflow in English – Actors/Roles, Objects, Tasks
2. Optionally – translate English to planner representation
3. Model workflow as a Petri net where transitions represent tasks
4. Define tasks as sequences of leaf-level commands in Lua
5. Define leaf level commands and objects: create costumes, props, objects, scripts, and animations
6. Execute and debug workflow - process is iterative
Workflow Simulation System

Modeling Workflows with Petri Nets

- Generalized, extended, prioritized Petri nets used in Petri net interpreter
- However, transitions do not fire as an instantaneous atomic action
- Each workflow task is associated with a single transition and when the transition fires its task is performed

Petri net example
Workflow in English

Handout of Appendix I
Petri net for Cath Lab Operation
Outline

- Introduction
- Background
- Building a Workflow Simulation System
  - Evaluation
- Conclusions and Future Work
Comparison of Systems

Catheterization Operation Workflow
An Example

- Guided by Prolog
- List of commands with some parallelization
- Petri net concurrent tasks

[Catheterization Workflow Improvements Video](#)
Workflow Simulation System

Modeling Workflows with Petri Nets

- Tokens are removed from the transition’s input places when the task starts, and placed in the transition’s output places when it finishes.
- This change requires additional rules to help determine when a transition is enabled.
- Transitions fire when enabled, allowing asynchronous, concurrent task execution.
Benefits of Improved System

- Workflow simulations in Second Life are easier to create and test
- Workflows are gaining more detail and expressiveness
- Added capabilities support greater variation, concurrency, and real-time decision-making
Second Life as Simulation Platform

Problems Solved

- Avatarbot navigation
- Picking up and setting down objects
- Precise avatarbot positioning and coordination
- Actors moving other actors (workaround)
- Object identification
Second Life as Simulation Platform

Solvable Problems

- Avatarbot turning
- More realistic object dropping
- Multiple roles per avatarbot & role hierarchy
- True or-splits/joins
- Configurable initial Petri net markings
Second Life as Simulation Platform

Limitations of current Second Life

- Lack of fine motor skills and facial expressions
- Lack of dynamic range in visual detail
- Lack of scalability to workflows with many actors, many objects, and/or large real-world systems
- Lack of real-world physics for object-object, object-avatar, and avatar-avatar interaction
- Lack of object model import capability
Outline

- Introduction
- Background
- Building a Workflow Simulation System
- Evaluation

☐ Conclusions and Future Work
Summary

- We developed
  - A collection of workflow simulations to better understand workflow requirements.
  - A workflow simulation system that can execute workflows in the virtual world Second Life. The current version uses Lua scripting and Petri nets for asynchronous, concurrent tasks.
Potential Impact

- As mentioned, current virtual worlds do not support workflow.
- Adding workflow simulation to virtual worlds enables visualization, modeling and simulation useful for so-called serious games with applications in try-before-you-buy what-if analysis, training, and entertainment.
- Longer term, computational workflows for everyday activities may enhance human-computer interactions in real-world situations.
Future Work

- Solvable problems with current Second Life
- Hybrid avatar/avatarbot workflow simulations
- Hierarchical parameterized workflows
- Planning and re-planning in real-time
- Simulation logging for later analysis
- Workflow optimization
- Workflow as a service extension to other virtual worlds