MODEL-INTEGRATED COMPUTING
ECE505

Jonathan Sprinkle

Day/Time:MWF 1:00-1:50 pm
Location: ECE 258

http://www.ece.arizona.edu/~ece505-2012F/

---

**Instructor:** Jonathan Sprinkle
ECE 456N
Email: sprinkle@ECE.Arizona.Edu (preferred contact method)

**Office Hours:**
M 10:00-11:00 am
W 11:00-12:00 pm
And by appointment. Instructor may cancel office hours without advance notice.

**Reference(s):**
- *Object-Oriented Modeling and Design with UML* 2nd ed., M. Blaha, J. Rumbaugh
- *Model-Integrated Computing*. J. Sprinkle (provided as a working manuscript)
- Academic papers available through the IEEE and ACM subscription
- Selections from online books

**Prerequisites:**
Graduate standing, or standing within the 4+1 MS program, or consent of the instructor prior to early withdrawal date. Advanced standing undergraduates are encouraged to enroll if they have already completed the undergraduate courses through ECE373. A strong command of the C++, VB, C#, or Java programming languages, or consent of instructor, is recommended. Previous knowledge of modeling technology is *not required*. Auditing requires written consent of instructor, prior to audit change date.

---

**Course Description:**

From the course catalog: This course addresses modeling, metamodeling, advanced object-oriented system design, model-integrated computing, and integrated systems. Behavioral, structural, and process modeling are among the modeling approaches to be examined. Additional topics include semantic mapping, models of computation, graph rewriting, and domain-specific modeling. In lieu of a midterm and final exam, an individual project will be performed over the course of the semester, constituting a large portion of the grade.

Model-Integrated Computing (MIC) addresses the problems of designing, creating, and evolving information systems by providing rich, domain-specific modeling environments including model analysis and model-based program synthesis tools. MIC is used to create and evolve integrated, multiple-aspect models using concepts, relations, and model composition principles routinely used in the specific field, to facilitate systems/software engineering analysis of the models, and to automatically synthesize applications from the models.

The class members will be exposed to several state of the art approaches for each type of modeling. At the conclusion of this class, the members will be able to develop a complete model-integrated computing environment for a given problem involving integrated systems. Class members will also understand why certain modeling approaches
are better suited for specific problems, and how to chose a particular modeling formalism, design or customize it, and use it for the given purpose.

The class will be lecture based. Students must have a working knowledge of C++ or Java, and must be familiar with object-oriented programming concepts. Grades will be based on homework, quizzes (scheduled, and unscheduled), in-class presentation (including forum discussions), and a substantial final project.

**Academic Topics**
The set of topics and areas covered by this course, and upon which you may be tested, include:

- software modeling and consistency with UML
- syntax and semantics of modeling languages;
- techniques of abstraction;
- metamodeling concepts, including semantic attachment, and constraint specification;
- graph rewriting, model interpretation;
- the Multi-Graph Architecture, as implemented in the Generic Modeling Environment;
- models of computation and communication;
- metamodel evolution and model migration; and
- technical topics presented by other course participants, including your classmates' technical presentations.

**Technical Topics**
Technical processes and skills of which you will gain knowledge of in this course include:

- public presentation of a research topic;
- modeling language design and implementation;
- abstraction of complex systems into manageable hierarchies of programming and modeling;
- high-level implementation details of compilers, especially C++;
- technical writing;
- proposal writing and evaluation; and
- peer review of technical discussions.

**Student Requirements**

- Students will be required to do a topical **presentation** (topics provided by instructor) approximately 30-45 minutes in length (depending on the number of students in the course).
- Each student will complete a **project** that entails developing a modeling paradigm and constructing a complete model-integrated computing environment to solve a particular engineering problem deemed acceptable by the instructor.
- There will be **in-depth homework** assignments given throughout the semester. Late assignments will not be accepted without prior approval.
- In addition to announced quizzes, **unannounced quizzes** may be given anytime during the semester. The unannounced quizzes may be given on material assigned for reading, but not yet covered in class.
- Be warned that installation of class software may require you to install a virtual machine on your system. **Not having the software installed in time to finish a homework assignment will not be accepted as an excuse.** If you anticipate this as a problem, please see the instructor well in advance of the first assignment due date.
• A significant portion of the grade for some assignments will be in the form of written documentation or formal writing (similar to an academic paper submission).

• A Homework Companion, which gives detailed submission instructions, will be provided to the students at the time of the first homework assignment. All graded work that is electronically submitted in this course must conform to the Homework Companion. Failure to follow these instructions may result in a reduction of the grade for that assignment.

• A Project Companion will be provided to students upon discussion of the project, detailing various requirements and providing a timeline for completion.

Grade Policy:

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
<th>Range</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homework</td>
<td>35%</td>
<td>[90 - 100]</td>
<td>A</td>
</tr>
<tr>
<td>Project</td>
<td>35%</td>
<td>[80 - 90]</td>
<td>B</td>
</tr>
<tr>
<td>Presentation</td>
<td>10%</td>
<td>[70 - 80]</td>
<td>C</td>
</tr>
<tr>
<td>Pop Quizzes/Participation</td>
<td>10%</td>
<td>[0 - 69]</td>
<td>E</td>
</tr>
<tr>
<td>Announced Quizzes</td>
<td>10%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>In-class Participation</td>
<td>10%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Assignment of grades is done according to a “modified-contract” method. The above scale represents a minimum guarantee. However, the instructor reserves the right to “upward curve” the final grade of the entire class, or of one or more individuals whose objective performance improves as the term progresses. Work will never be curved down.

Presentations
Each student will chose relevant material for the class to read, as part of a topical presentation; note: this is distinct from the project presentation. Each student must meet during office hours with the instructor for approval of the selected material and presentation outline at least two weeks before the scheduled presentation.

• The class must receive, from the presenter, the relevant material at least 7 days prior to the scheduled presentation. Deductions will be made if the material is not distributed in advance, according to the homework companion.

• All students are expected to have read the material before the presentation.

• Each presenter will prepare a quiz on the distributed reading material, which will count as a pop quiz grade, and be graded by the presenter.

• Presentations that lack depth, or reflect a simple re-presentation of an existing body of knowledge, will receive poor marks. In this presentation, you are responsible for presenting the knowledge just as you would have it presented to you in class.

• Reminder: class participation will affect your grade.

Possible Student Presentation Topics
• Discrete Event Simulation (DEVS)
• Logical Execution Time (LET) semantics
• Tagged Signal Model
• MOF/EMOF from OMG
• Advanced topics in models of computation
• Symbolic Model Analysis
• SPIN model checker
• Linear Temporal Logic
• Mathworks Design Verifier
• Object Constraint Language (OCL)

Students are encouraged to suggest other topics, though they must be approved by the instructor. Significant experience of the student (in the opinion of the instructor) with a particular topic may prevent that student from presenting on that topic.

Important Dates:

9/10  Last day to add classes for credit from zero units
9/16  Last day to drop courses resulting in deletion of course enrollment from record
10/12 Last day to drop a class with a grade of “W” (if passing) or to change to or from audit grading; the instructor’s signature on a Change of Schedule form is required
12/5  Last day of classes and laboratory sessions

Course Outline:

The listing of weekly course lecture topics may be found on the webpage, and is subject to change without notice due to class progress. In the event of class cancellation, advance notice via email will be given, but any homework due that day will still be due unless otherwise notified via email.

Homework Companion

The homework must be submitted according to the guidelines set forth by the Homework Companion (available from the course webpage). Failure to abide by the Homework Companion may result in a failing grade, regardless of the correctness of the work.

Project Companion:

Students in the course will perform work on a course project which will be of significant value as deemed appropriate by the instructors. The course project will be submitted via the terms of the Project Companion, which will be distributed.

Attendance, Participation, and Quizzes:

Attendance is mandatory. Although the class roll may not be taken every day, pop quizzes may be given without notice. Pop quizzes may not be made up.

Class Disruptions:

Please silence your cell phone, and do not use it during the class. The use of a phone in class will adversely affect your attendance grade.
Academic Integrity:

Students are expected to do all work by themselves, except when specified by the instructor in writing. All exceptions will be plainly marked in the requirements for that exercise or project. Any violations of this policy will be dealt with to the full extent permitted by the University of Arizona, and may result in suspension or expulsion from the university, in addition to a failing grade. Please familiarize yourself with the Code of Academic Integrity if you have any questions (see http://deanofstudents.arizona.edu/codeofacademicintegrity).

Safety Instructions:

The frequent operation of a computer, such as will be required in this course, may have long-term disabling effects if you do not appropriately consider your ergonomic interaction with the computer, desk, chair, and light sources. Poorly designed work stations/practices can lead to musculoskeletal disorders, and may result in chronic pain, inability to sleep, or expensive surgery decades from today. The habits you form in your university years may well impact your future performance, and it is highly recommended that you consult the ergonomics information from the Office of Risk Management, available at http://risk.arizona.edu/healthandsafety/ergonomics.shtml

Students with Disabilities:

If you anticipate issues related to the format or requirements of this course, please meet with the instructor to discuss ways to ensure your full participation in the course. If you determine that formal, disability-related accommodations are necessary, it is very important that you be registered with the Disability Resource Center (621-3268; drc.arizona.edu) and afterward notify the instructor of your eligibility for reasonable accommodations. Only after that point can we plan how to best coordinate any accommodations.