Certified Modeling and Simulation Professional Examination Preparation

Mikel D. Petty, Ph.D.
University of Alabama in Huntsville
256-824-6140 pettym@uah.edu

THE UNIVERSITY OF ALABAMA IN HUNTSVILLE
Acknowledgements

- **Sponsors**
  - National Training and Simulation Association
  - The Boeing Company

- **Guidance**
  - Modeling & Simulation Professional Certification Commission (exam structure)
  - SimSummit (body of knowledge outline)

- **Selected contributors** (alphabetical order)
  - Wesley N. Colley, Ph.D. (questions)
  - David C. Gross, Ph.D. (topics, sources, and slides)
  - Gregory S. Reed, Ph.D. (software)
  - John A. Sokolowski, Ph.D. (question review)
  - William F. Waite (topics and sources)
Instructor

- **Mikel D. Petty, Ph.D.**
  - Associate Professor, Computer Science
  - Senior Scientist for M&S, Info Tech and Sys Center

- **Education**
  - Ph.D. Computer Science, UCF 1997
  - M.S. Computer Science, UCF 1988
  - B.S. Computer Science, CSUS 1980

- **Career summary**
  - M&S R&D: UCF, ODU, UAH; 1990-present
  - IT: CSUS, UTEP, GM, UCF; 1980-1990

- **Research**
  - Modeling and simulation, esp. distributed simulation and V&V
  - > 195 publications
  - > $16 million total research funding awarded
Part 1: Introduction, structure, and logistics

- Introduction
  - Motivation
  - Certification types

- Structure
  - Topics and subtopics
  - Questions and instances

- Logistics
  - Requirements, preparation, and examination
  - Recertification

Placed first for people who must leave early; will be covered completely.
Part 2: Content survey

• Definitions and concepts
  ▪ M&S terms and attributes
  ▪ M&S categories

• Modeling methods
  ▪ Survey of modeling methods
  ▪ Discrete event simulation
  ▪ Monte Carlo simulation

• Special topics
  ▪ Verification, validation, and accreditation
  ▪ Distributed simulation

Intentionally too long; will be covered as time permits.
Part 1: Introduction, structure, and logistics

- Introduction
  - Motivation
  - Certification types

- Structure
  - Topics and subtopics
  - Questions and instances

- Logistics
  - Requirements, preparation, and examination
  - Recertification
The emergence of M&S as a distinct discipline

“Science used to be composed of two endeavors, theory and experiment. Now it has a third component: computer simulation, which links the other two.”

[Colwell, 1999] [Colwell, 2000]

Dr. Rita Colwell
Director, National Science Foundation
April 29 1999
CMSP certification background

• Professional certification, in general
  ▪ Indicator of individual expertise and competence
  ▪ Milestone in maturation of professional discipline
  ▪ Exists for law, medicine, project mgt, finance, …

• Professional certification, for M&S professionals
  ▪ CMSP “version 1” 2001, “version 2” 2010
  ▪ Approximately 400 people designated as CMSPs
CMSP organizing principles

- Overseen by M&SPCC
- Quality
  - Traceability to requirements set by M&SPCC
  - Exam questions based on credible sources, periodically updated and improved
- Transparency
  - Open process, clear values
  - Publicly available program data
  - Evidence of compliance with requirements
- Confidence; CMSPs are knowledgeable
- Ethics; ethical use of M&S part of certification
Benefits of CMSP certification for individuals

- **CMSP as credential**
  - **Peers**: community-recognized certification of M&S expertise
  - **Customers**: qualifications discriminator for competitive proposals
  - **Employer**: professional certification for career advancement

- **CMSP as learning experience**
  - Preparation for and completion of CMSP exam enhances M&S knowledge
  - Recertification requirements motivate continuous M&S learning
Part 1: Introduction, structure, and logistics

- Introduction
  - Motivation
  - Certification types

- Structure
  - Topics and subtopics
  - Questions and instances

- Logistics
  - Requirements, preparation, and examination
  - Recertification
Certification types: User/Manager

1. Employ and explain key terms, definitions, and concepts in modeling and simulation.
2. Apply important principles of modeling and simulation practice, including simulation ethics, business considerations, and related communities of practice.
3. Understand and work effectively within typical and important uses of modeling and simulation, including application areas and domains of use.
4. Identify, assess, and select relevant simulation technologies, including modeling paradigms and implementation architectures, for a specific application.
5. Determine whether the use of simulation is, or is not, appropriate for a specific application.
6. Plan, initialize, and execute simulation runs or trials to satisfy project requirements.
7. Analyze, interpret, and apply the results of simulation runs in the context of an application.
8. Manage aspects of projects involving the use or development of simulation models and systems.
Certification types: Developer/Technical

1. Employ and explain key terms, definitions, and concepts in modeling and simulation.
2. Apply important principles of modeling and simulation practice, including simulation ethics, business considerations, and related communities of practice.
3. Understand and work effectively within typical and important uses of modeling and simulation, including application areas and domains of use.
4. Design and develop simulation models of various types, including mathematical, logical, structural, and conceptual.
5. Identify the underlying mathematical issues associated with many simulation models, including numerical evaluation algorithms, digital discretization, and numerical precision.
6. Implement simulation models as executable software and verify those implementations.
7. Validate simulation models using suitable methods and assess the suitability of a model for a specific application.
8. Design and implement technical infrastructures needed to support simulation systems.
Part 1: Introduction, structure, and logistics

- Introduction
  - Motivation
  - Certification types

- Structure
  - Topics and subtopics
  - Questions and instances

- Logistics
  - Requirements, preparation, and examination
  - Recertification
Exam topics and subtopics

- **Intent:** coverage of M&S body of knowledge
- **Structure:** 8 topics, 54 subtopics
- **Content**
  - Initially based on SimSummit M&S BoK Index
  - Revised per expert recommendations
  - Revised per source availability and topic testability

[Image: http://www.sim-summit.org/]
**Topic 1: Concepts and context**

Essential terminology, foundational concepts, community consensus categorizations, and overarching modeling paradigms; history of the development and use of M&S.

1. Concepts and context
   1.1 Fundamental terms and concepts
   1.2 Categories and paradigms
   1.3 History of M&S

\[ h(t) = -16t^2 + vt + s \]

<table>
<thead>
<tr>
<th>( t )</th>
<th>( h(t) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1000</td>
</tr>
<tr>
<td>1</td>
<td>1084</td>
</tr>
<tr>
<td>2</td>
<td>1136</td>
</tr>
<tr>
<td>3</td>
<td>1156</td>
</tr>
<tr>
<td>4</td>
<td>1144</td>
</tr>
<tr>
<td>5</td>
<td>1100</td>
</tr>
<tr>
<td>6</td>
<td>1024</td>
</tr>
<tr>
<td>7</td>
<td>916</td>
</tr>
<tr>
<td>8</td>
<td>776</td>
</tr>
<tr>
<td>9</td>
<td>604</td>
</tr>
<tr>
<td>10</td>
<td>400</td>
</tr>
<tr>
<td>11</td>
<td>164</td>
</tr>
</tbody>
</table>
Topic 2: Applications of M&S

Important and cross-cutting M&S application types; modeling methods and organizing principles for each.

2. Applications of M&S
   2.1 Training
   2.2 Analysis
   2.3 Experimentation
   2.4 Acquisition
   2.5 Engineering
   2.6 Test and evaluation
**Topic 3: Domains of use of M&S**

Domains in which M&S has found wide use; key modeling methods and applications for each.

<table>
<thead>
<tr>
<th>3. Domains of use of M&amp;S</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Combat and military</td>
</tr>
<tr>
<td>3.2 Aerospace</td>
</tr>
<tr>
<td>3.3 Medicine and health care</td>
</tr>
<tr>
<td>3.4 Manufacturing and material handling</td>
</tr>
<tr>
<td>3.5 Logistics and supply chain</td>
</tr>
<tr>
<td>3.6 Transportation</td>
</tr>
<tr>
<td>3.7 Computer and communications systems</td>
</tr>
<tr>
<td>3.8 Environment and ecology</td>
</tr>
<tr>
<td>3.9 Business</td>
</tr>
<tr>
<td>3.10 Social science</td>
</tr>
<tr>
<td>3.11 Energy</td>
</tr>
<tr>
<td>3.12 Other domains of use</td>
</tr>
</tbody>
</table>
Topic 4: Modeling methods

Technical aspects of widely used modeling methods; characteristics and suitable applications for each.

4. Modeling methods
   4.1 Stochastic modeling
   4.2 Physics-based modeling
   4.3 Structural modeling
   4.4 Finite element modeling
   and computational fluid dynamics
   4.5 Monte Carlo simulation
   4.6 Discrete event simulation
   4.7 Continuous simulation
   4.8 Human behavior modeling
   4.9 Multi-resolution simulation
   4.10 Other modeling methods
## Topic 5: Simulation implementation

Engineering principles and practices for developing and validating M&S systems; M&S standards; special models.

<table>
<thead>
<tr>
<th>5. Simulation implementation</th>
<th>5.1 Modeling and simulation life-cycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.2 Modeling and simulation standards</td>
<td></td>
</tr>
<tr>
<td>5.3 Development processes</td>
<td></td>
</tr>
<tr>
<td>5.4 Conceptual modeling</td>
<td></td>
</tr>
<tr>
<td>5.5 Specialized languages</td>
<td></td>
</tr>
<tr>
<td>5.6 Verification, validation, and accreditation</td>
<td></td>
</tr>
<tr>
<td>5.7 Distributed simulation and interoperability</td>
<td></td>
</tr>
<tr>
<td>5.7 Virtual environments and virtual reality</td>
<td></td>
</tr>
<tr>
<td>5.8 Human-computer interaction</td>
<td></td>
</tr>
<tr>
<td>5.9 Semi-automated forces</td>
<td></td>
</tr>
<tr>
<td>5.10 Stimulation</td>
<td></td>
</tr>
</tbody>
</table>
Topic 6: Supporting tools, techniques, and resources

Technical infrastructures, M&S resources, and organizations supporting the development and use of M&S.

6. Supporting tools, techniques, and resources
   6.1 Major simulation infrastructures
   6.2 M&S resource repositories
   6.3 M&S organizations
Topic 7: Business and management of M&S

Business of M&S and M&S as a business; professional conduct for M&S practitioners; M&S workforce.

7. Business and management of M&S
   7.1 Ethics and principles for M&S practitioners
   7.2 Management of M&S projects and processes
   7.3 M&S workforce development *
   7.4 M&S business practice and economics *
   7.5 M&S industrial development *

* Sources sought for these subtopics.
Topic 8: Related communities of practice and disciplines

Non-M&S topics with which M&S professionals should have some familiarity.

8. Related communities of practice and disciplines
   8.1 Statistics and probability
   8.2 Mathematics
   8.3 Software engineering and development
   8.4 Systems science and engineering

\[
R = 2.59 \times \sqrt{\sigma \times \left( \log^{-1} \left( \frac{ERP_t}{10} \right) \log^{-1} \left( \frac{G_r}{10} \right) \log^{-1} \left( \frac{MDS_r}{10} \right) \right) \log^{-1} \left( \frac{FEL_r}{10} \right) F_t^2}
\]
Part 1: Introduction, structure, and logistics

- **Introduction**
  - Motivation
  - Certification types

- **Structure**
  - Topics and subtopics
  - Questions and instances

- **Logistics**
  - Requirements, preparation, and examination
  - Recertification
Question counts and sources

- **Counts**
  - Total: ~2000 new questions
  - Per subtopic: min ≥ 20, mean ~40, max > 100
- **Sources**
  - Each question directly based on specific source
  - Sources: published, peer-reviewed, publicly available
  - Yes: journal papers, conference papers, books
  - No: briefings, unpublished reports, Wikipedia
  - > 175 different sources
  - Source list available
Question format and metadata

- **Format**
  - ~75% multiple choice (one correct, three incorrect)
  - ~25% True–False
  - Diagrams, images, mathematical formulas used

- **Question metadata**
  - Question and answers
  - Unique question number
  - Source citation, including page number
  - Author
  - Subtopic
  - Certification type (User/Mgr, Dev/Tech, Core)
  - Difficulty (1–5)
Example question, with metadata

Question number 8.545
Question Which of the following is not a use of simulation?
Correct answer Justify decisions already made based on other criteria
Incorrect answer Describe and analyze the behavior of a system
Incorrect answer Ask and answer “what if” questions about a system
Incorrect answer Help in designing new systems
Type Core
Difficulty 2 (Easy)
Topic 1.1 Fundamental terms and concepts
Page number 3
Question author M. Petty
# Example question, with metadata

<table>
<thead>
<tr>
<th>Question number</th>
<th>6.20</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Question</strong></td>
<td>In simulating a physical system governed by partial differential equations, __________ can be used to facilitate estimation of derivatives.</td>
</tr>
<tr>
<td><strong>Correct answer</strong></td>
<td>Fourier analysis</td>
</tr>
<tr>
<td><strong>Incorrect answer</strong></td>
<td>The Graham-Schmidt process</td>
</tr>
<tr>
<td><strong>Incorrect answer</strong></td>
<td>The downhill-simplex method</td>
</tr>
<tr>
<td><strong>Incorrect answer</strong></td>
<td>Gauss-Jordan elimination</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>Developer/Technical</td>
</tr>
<tr>
<td><strong>Difficulty</strong></td>
<td>5 (Very difficult)</td>
</tr>
<tr>
<td><strong>Topic</strong></td>
<td>4.2 Physics-based modeling</td>
</tr>
<tr>
<td><strong>Page number</strong></td>
<td>530</td>
</tr>
<tr>
<td><strong>Question author</strong></td>
<td>W. Colley</td>
</tr>
</tbody>
</table>
Example question, with metadata

Question number 9.78

Question Which of the following terms best describes use of models and simulation by the military, for the purposes of obtaining insight into the cost and performance of military equipment?

Correct answer Requirements and acquisition

Incorrect answer Exploration of advanced technologies and concepts

Incorrect answer Training

Incorrect answer Geo-navigation

Type User/Manager

Difficulty 3 (Moderate)

Topic 3.1 Combat and military

Source R. D. Smith, Military Simulations & Serious Games, Modelbenders Press, Orlando FL, 2009.

Page number 38

Question author S. Barbosa
### Example question, with metadata

<table>
<thead>
<tr>
<th>Question number</th>
<th>8.546</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Question</strong></td>
<td>True or False: Only systems that actually exist, as opposed to those that have been planned or designed but not implemented, can be simulated.</td>
</tr>
<tr>
<td><strong>Correct answer</strong></td>
<td>False</td>
</tr>
<tr>
<td><strong>Incorrect answer</strong></td>
<td>True</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>Core</td>
</tr>
<tr>
<td><strong>Difficulty</strong></td>
<td>2 (Easy)</td>
</tr>
<tr>
<td><strong>Topic</strong></td>
<td>1.1 Fundamental terms and concepts</td>
</tr>
<tr>
<td><strong>Page number</strong></td>
<td>4</td>
</tr>
<tr>
<td><strong>Question author</strong></td>
<td>M. Petty</td>
</tr>
</tbody>
</table>
### Example question, with metadata

<table>
<thead>
<tr>
<th>Question number</th>
<th>9.65</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Question</strong></td>
<td>Which of the following terms best describes the purpose of sensor footprint exaggeration in military simulations?</td>
</tr>
<tr>
<td><strong>Correct answer</strong></td>
<td>It ensures that detection calculations are carried out on all detectable objects between two time steps</td>
</tr>
<tr>
<td><strong>Incorrect answer</strong></td>
<td>It is used for marketing brochures</td>
</tr>
<tr>
<td><strong>Incorrect answer</strong></td>
<td>It compensates for hindrances to line-of-sight</td>
</tr>
<tr>
<td><strong>Incorrect answer</strong></td>
<td>It normalizes sensor footprints</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>Developer/Technical</td>
</tr>
<tr>
<td><strong>Difficulty</strong></td>
<td>4 (Difficult)</td>
</tr>
<tr>
<td><strong>Topic</strong></td>
<td>3.1 Combat and military</td>
</tr>
<tr>
<td><strong>Page number</strong></td>
<td>357</td>
</tr>
<tr>
<td><strong>Question author</strong></td>
<td>S. Barbosa</td>
</tr>
</tbody>
</table>
Example question, with metadata

Question number 8.10
Question Which of the following terms is best defined as “the process of determining whether an implemented model is consistent with its specification”?
Correct answer Verification
Incorrect answer Validation
Incorrect answer Accreditation
Incorrect answer Calibration
Type Core
Difficulty 2 (Easy)
Topic 5.6 Verification, validation, and accreditation
Page number 330
Question author M. Petty
Examination instances

• Examination instance generation
  ▪ Unique instance generated for each candidate
  ▪ Candidate selects certification type
  ▪ Candidate selects excluded subtopics
  ▪ Questions selected randomly within selections

• Examination instance
  ▪ 100 questions
  ▪ From selected certification type (U/M, D/T) or Core
  ▪ At least 10 questions per topic
  ▪ No questions from excluded subtopics
  ▪ Mean difficulty minimum 2.5, maximum 3.5
Part 1: Introduction, structure, and logistics

• Introduction
  ▪ Motivation
  ▪ Certification types

• Structure
  ▪ Topics and subtopics
  ▪ Questions and instances

• Logistics
  ▪ Requirements, preparation, and examination
  ▪ Recertification
Certification requirements

- Education, work experience, and examination
- Letters of recommendation (3)
- Signed ethics statement

<table>
<thead>
<tr>
<th>Education</th>
<th>Related work experience</th>
<th>CMSP exam</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.S. degree</td>
<td>8 years</td>
<td>Passing grade</td>
</tr>
<tr>
<td>B.S. degree</td>
<td>6 years</td>
<td>Passing grade</td>
</tr>
<tr>
<td>M.S. degree</td>
<td>5 years</td>
<td>Passing grade</td>
</tr>
<tr>
<td>Ph.D. degree</td>
<td>3 years</td>
<td>Passing grade</td>
</tr>
</tbody>
</table>
Preparing for the CMSP exam

- CMSP exam intentionally not “crammable”
  - Questions from a large number of sources
  - Passing requires broad knowledge and experience
- Preparing for the exam
  - Attend CMSP prep course: UCF, AEgis, UAH
  - Secure access to sources for “take home” period
  - Read sources
Key sources

- More than 175 different sources for questions …
- … but four “key” sources should be available before attempting the exam


Examination delivery process

- Candidate applies and pays fee ($250)
- Administrator creates account for candidate
- Candidate selects certification type, exclusions
- System generates examination instance
- System give candidate web access to instance
- Candidate answers 100 questions over 14 days
- Candidate submits completed examination
- System scores examination
- Administrator notifies candidate of result
Examination web site

CMSP Examination

Presented below are the questions for your exam. Unanswered questions are noted with a red bar. You may answer the exam questions in any order, navigate to any section of the exam, or log out and resume at a later time during your examination period. You may also change your answers at any time before you submit your exam. Your provided responses are saved automatically by the system. Once you have answered all questions, you may submit your exam using the Submit exam button at the bottom of the page.

40. Which of the following behavior modeling issues is more likely to be associated with models of individual, rather than group, human behavior?

- Collaboration
- Stimulus-response reactions
- Evolutionary dynamics
Part 1: Introduction, structure, and logistics

- Introduction
  - Motivation
  - Certification types

- Structure
  - Topics and subtopics
  - Questions and instances

- Logistics
  - Requirements, preparation, and examination
  - Recertification
CMSP recertification

- CMSP designees must recertify every 4 years
  - Recertification application and fee
  - Formal resubscription to Simulationist Code of Ethics
  - Minimum 275 recertification units

- Recertification units
  - Employment
  - Publications
  - Professional society participation
  - Education
  - Conference participation
  - Continuing education units
Part 2: Content survey

• Definitions and concepts
  ▪ M&S terms and attributes
  ▪ M&S categories

• Modeling methods
  ▪ Survey of modeling methods
  ▪ Discrete event simulation
  ▪ Monte Carlo simulation

• Special topics
  ▪ Verification, validation, and accreditation
  ▪ Distributed simulation
Concepts of model and simulation

- **Model**: representation of something else
- **Simulation**: executing a model over time

\[ R = 2.59 \times 4 \sqrt{\frac{1}{\sigma} \left( \log^{-1} \left( \frac{ERP_t}{10} \right) \log^{-1} \left( \frac{G_r}{10} \right) \log^{-1} \left( \frac{MDS_r}{10} \right) \right) \log^{-1} \left( \frac{FEL_r}{10} \right) F_t^2} \]
Definition: Model

Model. A physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process. [DOD, 2011]

To an observer $B$, an object $A^*$ is a model of an object $A$ to the extent that $B$ can use $A^*$ to answer questions that interest him about $A$. [Minsky, 1965]

- Representation of something else, often a “real-world” system
- Some aspects of the modeled system are represented in the model, others not
Example: Model

Equation describing vertical height of an object moving under gravity.

\[ h(t) = -4.9t^2 + vt + s \]

- \( h \) = height (meters)
- \( t \) = time in motion (seconds)
- \( v \) = initial velocity (meters per second, + is up)
- \( s \) = initial height (meters)

Note that at \( t = 0 \), \( h = s \), as expected.
Definition: Simulation

Simulation.  [Executing] a model over time.  [DOD, 2011]
The imitation of the operation of a real-world process or system over time.  [BanksJ, 1998]
A technique for testing, analysis, or training in which real world systems are used, or where a model reproduces real world and conceptual systems.  [DOD, 1996]

Alternative uses of the term
- A large composite model (avoid)
- Software implementation of a model (avoid)
- Application of a model to solve a problem  [Velten, 2009]
- An experiment performed on a model  [Fritzson, 2004]
- Several others  [Ören, 2009] [BanksC, 2010]
Example: Simulation
Model: \( h(t) = -4.9t^2 + vt + s \)  Data: \( v = 30, \ s = 300 \)

<table>
<thead>
<tr>
<th>( t )</th>
<th>( h(t) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>300.0</td>
</tr>
<tr>
<td>1</td>
<td>325.1</td>
</tr>
<tr>
<td>2</td>
<td>340.4</td>
</tr>
<tr>
<td>3</td>
<td>345.9</td>
</tr>
<tr>
<td>4</td>
<td>341.6</td>
</tr>
<tr>
<td>5</td>
<td>327.5</td>
</tr>
<tr>
<td>6</td>
<td>303.6</td>
</tr>
<tr>
<td>7</td>
<td>269.9</td>
</tr>
<tr>
<td>8</td>
<td>226.4</td>
</tr>
<tr>
<td>9</td>
<td>173.1</td>
</tr>
<tr>
<td>10</td>
<td>110.0</td>
</tr>
<tr>
<td>11</td>
<td>37.1</td>
</tr>
<tr>
<td>11.46</td>
<td>0</td>
</tr>
</tbody>
</table>

\( h(t) \) vs. \( t \):
- Start state: 11.46, 0
Simulation vs reality

Real-world system in start state  
Real-world system in end state  

Model in start state  
Model in end state

\[ h(t) = -4.9t^2 + vt + s \]
\[ 300 = -4.9(0)^2 + 30(0) + 300 \]
\[ h(t) = -4.9t^2 + vt + s \]
\[ 0 = -4.9(11.46)^2 + 30(11.46) + 300 \]
Definitions: Simuland and referent

Simuland. The subject of a model or simulation.
- A real-world (or notional) system of interest
- Object, phenomenon, or process to be simulated

Referent. The body of knowledge available to a modeler regarding a simuland.
- Quantitative and formal,
  e.g. physics equations for aircraft flight dynamics
- Qualitative and informal,
  e.g., pilot’s expectation of buffet before stall
Definition: Intended use

**Intended use.** The purpose for or function of a model; what the model will be used for; the problem to be addressed by the model. [DOD, 2012]

**Characteristics [Piersall, 2014]**
- “Not a capability, solution, or implementation”
- Expressed from the perspective of the user
- Should be formally specified

“Any attempt to suppress the role of the intentions of the investigator, B, leads to circular definitions or to ambiguities about essential features ...” [Minsky, 1965]

“Modeling and simulation is always goal-driven, ... , we should know the purpose of our potential model before we sit down to create it.” [Cellier, 1991]
Definitions: Conceptual and executable models

**Conceptual model** [BanksC, 2010]
- Simuland components, structure
- Aspects of simuland to model
- Implementation specifications
- Use cases
- Assumptions
- Initial model parameter values

**Executable model**
- Computer software
- Implemented conceptual model [DOD, 2011]
- AKA computational model [Waltz, 2010] [DOD, 2011]

```c
/* Height of an object moving in gravity. */
/* Initial velocity v and height s constants. */
main()
{
  float h, v = 30.0, s = 300.0;
  int t;
  for (t = 0, h = s; h >= 0.0; t++)
  {
    h = (-4.9 * t * t) + (v * t) + s;
    printf("Height at time %d = %f\n", t, h);
  }
}
```
Definition: Abstraction

Abstraction. Intentional omission of aspects of the simuland considered negligible or irrelevant.

Abstraction may be motivated by …

• Cost; reduce implementation and testing resources by omitting simuland aspects not need for application
• Execution speed; don’t compute details or effects that are negligible
• Understanding; omit aspects of simuland not understood well enough to model accurately

“Modeling and simulation aims at simplification, rather than at a useless production of complex copies of a complex reality. … The best model is the simplest one that still serves its purpose.” [Velten, 2009]
Examples: Abstraction

**Physical model**
- Represents: Appearance
- Abstracts: Size

**Physical model**
- Represents: Aerodynamics
- Omits: Ailerons

**Physical model**
- Represents: Flight
- Omits: Appearance

**Simuland P-51D Mustang**

**Visual model**
- Represents: Appearance
- Omits: Flight

**Functional model**
- Represents: Aerodynamics
- Omits: Appearance

**Aggregate model**
- Represents: Combat
- Abstracts: Pilot skill

[Han, 2012]
[Bollay, 1951]
Definition: Validity

Validity. Accuracy of model’s representation or simulation’s results. AKA fidelity.

“The degree to which a model or simulation reproduces the state and behavior of a real world object or the perception of a real world object, feature, condition, or chosen standard in a measurable or perceivable manner...” [Gross, 1999]

Validity is relative to …

- Simuland: How closely do the simulation results match the simuland’s behavior?
- Intended use: How much validity is needed for the model’s intended use?
Definition: Resolution

Resolution. The degree of detail with which the real-world is simulated. More detail is higher resolution. AKA granularity.

Examples (low to high resolution)
- Simulate fighter squadron as a whole
  - Squadron capabilities, strength represented abstractly
- Simulate individual aircraft
  - Separate model(s) for each individual aircraft
  - Aircraft capabilities represented abstractly
- Simulate sensor and weapons systems
  - Separate model(s) for each system on each aircraft
Definition: Scale

Scale. Size of the overall scenario or event the simulation represents. AKA level.

Typical scales for military simulations

- **Engineering, Component**
  - System or subsystem of a single entity
- **Engagement, Platform**
  - 1-v-1 to many-v-many
- **Mission, Battle**
  - 10s to 1000s of entities
- **Theater, Campaign**
  - 10000s of entities
Part 2: Content survey

- Definitions and concepts
  - M&S terms and attributes
  - M&S categories

- Modeling methods
  - Survey of modeling methods
  - Discrete event simulation
  - Monte Carlo simulation

- Special topics
  - Verification, validation, and accreditation
  - Distributed simulation
Different M&S categories

- **Application type**
  - Training, Analysis, Experimentation, Engineering, Acquisition, Entertainment

- **Randomness**
  - Deterministic, Stochastic

- **Time**
  - Discrete, Continuous

- **Environment**
  - Live, Virtual, Constructive

- **Modeling method**
  - Many; surveyed later

"Model space"
Definition: Training

Training. M&S used to produce learning in a user or participant.

- Realistic enough to produce useful skills or knowledge
- Safer, more forgiving of mistakes
- Encounter unusual and/or dangerous situations
Definition: Analysis

**Analysis.** M&S used to predict, test, or evaluate a real or notional system or idea.

- Used to answer questions
- Repeatability often desirable
  - Avoid confounding variability
- Careful experimental design
  - Trials planned in advance to cover cases
  - Multiple trials for statistical significance

JCATS, Wurtsmith AFB
Definition: Experimentation

Experimentation. M&S used to explore design or solution spaces, or to gain insight into an incompletely understood situation. [Ceranowicz, 1999]

[An] iterative process of collecting, developing and exploring concepts to identify and recommend the best value-added solutions for changes in the doctrine, organization, training, material, leadership and people required to achieve significant advances in future joint operational capabilities. [Helfinstine, 2001]

- Simulation used to “explore” possibilities
- Less “controlled” than analysis
  - Later trials may be determined by earlier
  - Statistical significance may not be important
Definition: Engineering

Engineering. M&S used to develop, analyze, or test an engineering design.

- Model artificial systems and components
- Models physics-based, no “behavior”
- No virtual environment or simulators
- User not expected to benefit from experience of execution
- Primary goal: informative result
**Definition: Acquisition**

**Acquisition.** M&S used to specify, design, develop, and acquire new systems.

- Simulation used to acquire systems
  - Effectiveness and selection: “Build the right thing”
  - Engineering and production: “Build the thing right”
- Goal: save lives, money, time
Definition: Entertainment

Entertainment. M&S used to entertain or amuse a user or participant.

- Engaging gameplay and/or goals
- Place user in situation, place, or activity he/she may not have access to in reality
- Realism secondary to user enjoyment

Examples:
- Combat Flight Simulator 3
- World of Warcraft
Definition: Continuous

Continuous model. Model where state variables can change (pseudo-)continuously over time. Typically time advances in small fixed time steps. AKA “time-stepped”. [BanksJ, 2010]

Continuous simulation. Simulation using continuous models.

\[
t = \text{start\_time} \\
\text{while } t < \text{end\_time} \\
\quad t = t + \Delta t \\
\quad \text{calculate simulation state at } t \\
\text{ endwhile}
\]
Definition: Discrete

Discrete model. Model where model state variables change only at a discrete set of points in time ("events"). AKA “event-driven”. [BanksJ, 2010]

Discrete event simulation. Simulation using discrete models and event handling.

```
t = 0
while t < end_time
    t = time of next event e
    process event e
    possibly schedule future events
endwhile
```
Definition: Live

Live. Simulation involving real people operating real systems. [DOD, 2011]

- Real systems
- Systems operated by participants
- Participant experience close to reality
- Instrumentation substitutes for simuland aspects
  - e.g., weapon firings or impacts
  - e.g., spacecraft launch
- Primary goal: useful experience
- Typical application: training
Example: Hazmat exercise

- Live training for local emergency responders
- Practice hazardous material response, cleanup
- Simulation
  - Harmless or noxious hazmat surrogate
  - Roleplayers act as victims
  - Real equipment used
Definition: Virtual

Virtual. Simulation involving real people operating simulated systems. [DOD, 2011]

- Simulators and virtual environment
- Simulators operated by participants
- Participant experience usefully realistic
- Important simuland aspects modeled
- Participants exercise motor control skills, decision skills, or communication skills [DOD, 2011]
- Primary goal: useful experience
- Typical application: training
Example: Close Combat Tactical Trainer

- Virtual simulators
  - Trainees in vehicle simulators
  - Computer generated battlefield

- Combat training
  - Trains team tactics
  - Platoon to battalion units
Definition: Constructive

Constructive. Simulation involving simulated people operating simulated systems. Real people make inputs to such simulations, but are not involved in determining the outcomes. [DOD, 2011]

- No real systems, virtual environment, or simulators
- Important simuland aspects modeled
- Operator and model control simuland behavior
- Operator not expected to benefit from use
- Primary goal: useful result
- Typical applications: analysis, experimentation
Example: Joint Theater Level Simulation

[Kang, 1998] [JFCOM, 2007]

- “Wargame” style constructive simulation
- JTLS models
  - Hexagonal terrain
  - Military units (air, ground, naval)
  - Movement, sensing, logistics: abstract mathematical models
  - Combat: Lanchester equations
- Applications
  - Training (exercise driver)
  - Analysis
Part 2: Content survey

- Definitions and concepts
  - M&S terms and attributes
  - M&S categories

- Modeling methods
  - Survey of modeling methods
  - Discrete event simulation
  - Monte Carlo simulation

- Special topics
  - Verification, validation, and accreditation
  - Distributed simulation
Definition: Physical model

Physical model. A physical object which models physical aspects of the simuland. [Haefner, 2005]

AKA surrogate.

Physical models are

- Typically used in live simulation
- Usually not a perfect replica, but "close enough"
- Often motivated by
  - Safety; use of simuland dangerous
  - Cost; use of simuland costly
  - Non-availability; simuland not available for use
Example: Harvard glass flowers

- Glass models of plants
- Made 1887–1936 by Leopold and Rudolf Blaschka
- 3000 models, 847 species
- Used in botany instruction
- Always available, regardless of season
Definition: Conceptual model

Conceptual model. Representation, in any of a variety of narrative or graphical means, of the elements, relationships, and causal relationships of a simuland and its context. Adapted from [Waltz, 2010]

- Possible forms: Text, photographs, sketches, equations, concept maps, tables, algorithms, data, UML diagrams
- Uses
  - Communicate among SMEs, modelers, users, managers, and customers
  - Define what will be represented and omitted in implemented executable model [Petty, 2010]
  - Document model assumptions
  - Support comparison of implemented model capabilities to stated requirements [Petty, 2010]
Example: Concept map

- Semi-active radar homing air-to-air missile
- Boxes = objects, arrows = interactions
- Additional components, interactions may be added
Definition: Declarative models

- **Description** [Fishwick, 1995]
  - Model represents qualitatively distinct simuland states
  - Simulation traces sequence of states over time
  - State transitions conditional (inputs or events), or stochastic (probability distributions)
  - Processes, phenomena often modeled stochastically

- **Forms**
  - Finite state automata
  - Markov chains
  - Queueing models, e.g., discrete event simulation
  - Formal models
Example: Finite state machine

- **Description**
  - Structured as directed graph; vertices are states, edges are transitions
  - Uniform or continuing activities are states
  - Changes from one activity or another are transitions
  - Transitions triggered by conditions (input, event, time)
  - Concept adapted from theoretical computer science
  - AKA finite state automata (FSA)

- **Modes of use**
  - FSM is complete model
  - FSM is organizing framework for other models
Example: Dismounted infantry ATGM FSM

- Example of FSM used to organize other models
- Each state’s action implemented as source code

[Petty, 2009a]
Definition: Formal models

- **Description**
  - Modeling notation or language defined mathematically
  - Precise, unambiguous syntax and semantics
  - Allows proofs of desirable model properties, e.g., syntactic interoperability, state reachability
  - Potentially amenable to automated analyses
  - Theoretically equivalent to programming language

- **Modes of use**
  - Use formal model as design specification for subsequent implementation as software
  - Execute formal model directly using interpreter
Example: Discrete Event System Specification
[Zeigler, 2000]

- DEVS: Well known formal modeling notation
- Research literature of example applications

The atomic DEVS model for player A of Fig. 1 is given by $\text{Player} = \langle X, Y, S, s_0, t_a, \delta_{\text{ext}}, \delta_{\text{int}}, \lambda \rangle$ such that

- $X = \{ ?\text{receive} \}$
- $Y = \{ !\text{send} \}$
- $S = \{ (d, \sigma) | d \in \{ \text{Wait, Send} \}, \sigma \in T^\infty \}$ and $s_0 = (\text{Send}, 0.1)$
- $t_a(s) = \sigma$ for all $s \in S$
- $\delta_{\text{ext}}((\text{Wait}, \sigma), t_c, ?\text{receive}) = (\text{Send}, 0.1)$
- $\delta_{\text{ext}}((\text{Send}, \sigma), t_c, ?\text{receive}) = (\text{Send}, \sigma - t_c)$
- $\delta_{\text{int}}(\text{Send}, \sigma) = (\text{Wait}, \infty)$, $\delta_{\text{int}}(\text{Wait}, \sigma) = (\text{Send}, 0.1)$
- $\lambda(\text{Send}, \sigma) = !\text{send}$, $\lambda(\text{Wait}, \sigma) = \phi$

Both Player A and Player B are deterministic DEVS models.
Definition: Functional models

• Description
  ▪ Model represents dynamic physical processes and phenomena of simuland and context
  ▪ Often consists of directionally-connected equations
  ▪ Equations based on physics, hence functional models AKA physics-based
  ▪ Often differential equations, or discretized versions
  ▪ May be directly solvable (AKA “analytic”) or may require numerical approximation (AKA “numeric”)
  ▪ Methods for latter exist, e.g., Runge-Kutta [Colley, 2010]

• Forms
  ▪ Direct implementation as source code
  ▪ Block diagrams in modeling environment
Example: Deceleration

Auto deceleration at toll booth

- Is there sufficient stopping distance in toll booth ramp?
- Model with physics equations, modeling environment

\[ F_f = m\mu g \]
\[ a_f = \mu g \]
\[ a_f = \frac{\mu}{m} \]  

Newton’s second law  
\[ a = \frac{\Delta v}{\Delta t} \]  
Acceleration  
\[ v = \frac{\Delta x}{\Delta t} \]  
Velocity
Example: Deceleration implementation

Adapted from [McKenzie, 2010]
Definition: Constraint models

• Description
  ▪ Similar to functional models
  ▪ No explicit directionality of connections, rather balancing of quantities or relationships
  ▪ Often physics-based equations or components

• Forms
  ▪ Direct implementation as source code
  ▪ Block diagrams in modeling environment
Example: Stopping distance

- How much distance is required to stop for given speed?

\[ F_f = m\mu g \]

\[ \frac{1}{2}mv^2 \]

\[ \frac{1}{2}mv^2 = F_f d \]

\[ \frac{1}{2}mv^2 - \mu mgd = 0 \]

\[ d = \frac{1}{2}v^2/\mu g \]

Friction force
Car kinetic energy
To stop, kinetic energy equals friction over distance
Algebra
Solve for distance; constraint model

Stopping distance, given speed

<table>
<thead>
<tr>
<th>Velocity (mph)</th>
<th>Velocity (m/s)</th>
<th>Stopping distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>8.9408</td>
<td>5.438</td>
</tr>
<tr>
<td>30</td>
<td>13.4112</td>
<td>12.235</td>
</tr>
<tr>
<td>40</td>
<td>17.8816</td>
<td>21.752</td>
</tr>
<tr>
<td>50</td>
<td>22.3520</td>
<td>33.907</td>
</tr>
<tr>
<td>55</td>
<td>24.5872</td>
<td>41.125</td>
</tr>
<tr>
<td>60</td>
<td>26.8224</td>
<td>48.942</td>
</tr>
<tr>
<td>70</td>
<td>31.2928</td>
<td>66.615</td>
</tr>
</tbody>
</table>

Speed, given stopping distance

<table>
<thead>
<tr>
<th>Stopping distance</th>
<th>Velocity (m/s)</th>
<th>Velocity (mph)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>12.124</td>
<td>27.121</td>
</tr>
<tr>
<td>20</td>
<td>17.146</td>
<td>38.355</td>
</tr>
<tr>
<td>30</td>
<td>21.000</td>
<td>46.976</td>
</tr>
<tr>
<td>40</td>
<td>24.249</td>
<td>54.243</td>
</tr>
<tr>
<td>50</td>
<td>27.111</td>
<td>60.645</td>
</tr>
<tr>
<td>60</td>
<td>29.698</td>
<td>66.434</td>
</tr>
<tr>
<td>70</td>
<td>32.078</td>
<td>71.757</td>
</tr>
</tbody>
</table>
Definition: Spatial models

• Description
  ▪ Simuland, or space occupied by simuland, partitioned into a large number of spatial elements
  ▪ Model state of simuland at each element, or model entities located within the elements
  ▪ Entities/elements treated in “regularized manner”, i.e., update rules for each element often identical
  ▪ State or activity of each entity/element often depends on neighboring entities/elements

• Forms
  ▪ Direct implementation as source code
  ▪ Dedicated spatial modeling tools
Example: Finite element modeling

- Simuland represented by “mesh” of nodes (elements) and connecting edges (neighboring elements)
- Nodes’ states modeled with physics equations
- State computations iterated over successive time steps
- Each time step, each node’s state calculated based on:
  - Length of time step
  - Neighbors’ states
Example: Agent based models

- Individual entities (“agents”) focus of model
- Agents may occupy, move between elements
- Agents interact with each other and environment according to behavior rules and parameters
- Complex behavior emerges from interaction
- Widely varying in details, sophistication

Sugarscape [Epstein, 1996]
Creative City [Malik, 2015]
Definition: Visual models

Visual model. A model of the appearance of an object, perhaps in different variations.

Visual models
- Often based on polygons (shape) and textures (colors applied to polygons)
- May or may not be specific to a particular image generator
- Realism of image not related to validity of underlying physics or behavior
Example: Aircraft visual models

Personal computer flight simulation game
Microsoft® CFS 3, 1920x1200 pixels

P-38J
Note polygonalization in model

P-38L
Same polygons, different textures
Definition: Data-based models

Data-based model. A model based on data (rather than equations, blocks, or logic) that describe the represented aspects of the simuland.

AKA empirical [Waltz, 2010], phenomenological [Velten, 2009], data-driven [Velten, 2009], statistical [Velten, 2009]

- Model not directly based on physics equations
- Data collected (or generated) in advance
- Examples: probability distribution models, linear regression models
**Example: Atmospheric CO$_2$**

Model based on observational data

<table>
<thead>
<tr>
<th>Year</th>
<th>CO2 ppm</th>
</tr>
</thead>
<tbody>
<tr>
<td>1980</td>
<td>338.7</td>
</tr>
<tr>
<td>1982</td>
<td>341.1</td>
</tr>
<tr>
<td>1984</td>
<td>344.4</td>
</tr>
<tr>
<td>1986</td>
<td>347.2</td>
</tr>
<tr>
<td>1988</td>
<td>351.5</td>
</tr>
<tr>
<td>1990</td>
<td>354.2</td>
</tr>
<tr>
<td>1992</td>
<td>361.4</td>
</tr>
<tr>
<td>1994</td>
<td>358.9</td>
</tr>
<tr>
<td>1996</td>
<td>362.6</td>
</tr>
<tr>
<td>1998</td>
<td>366.6</td>
</tr>
<tr>
<td>2000</td>
<td>369.4</td>
</tr>
<tr>
<td>2002</td>
<td>372.9</td>
</tr>
</tbody>
</table>

\[
\sum x = 23,892 \\
\sum y = 4,263.9 \\
\sum x^2 = 47,569,544 \\
\sum y^2 = 1,516,451.25 \\
\sum xy = 8,490,312.6 \\
\bar{x} = 1991 \\
\bar{y} = 355.33
\]

Slope \( m = \frac{n \sum xy - (\sum x)(\sum y)}{n \sum x^2 - (\sum x)^2} = \frac{12 \cdot 8,490,312.6 - 23,892 \cdot 4,363.9}{12 \cdot 47,569,544 - 23,892^2} \approx 1.5519 \]

Intercept \( b = \bar{y} - m\bar{x} = 255.33 - 1.5519 \cdot 1991 = -2,734.55 \]

**Model** \( C = 1.5519t - 2734.55 \)

[Stewart, 2008]
[Brase, 2009]
Definition: Aggregate models

**Aggregate model.** A model that represents a large number of small objects and actions in a combined, or aggregate, way.

Aggregate models
- Often used in constructive simulation
- Generally not directly physics-based
  - Abstraction of many physics-based interactions
Example: Lanchester equations

Differential equations for attacker and defender force attrition with respect to time; widely used for military operations research. \[\text{[Lanchester, 1956]}\] \[\text{[Davis, 1995]}\]

\[
\frac{dA}{dt} = -K_d A^r D^s \quad \frac{dD}{dt} = -K_a D^t A^u
\]

- \(A\): Attacker strength (abstract, aggregate value)
- \(D\): Defender strength (abstract, aggregate value)
- \(K_a, K_d\): Lethality (\(K_a\) attacker, \(K_d\) defender)
- \(r, s, t, u\): Free parameters, time independent
Part 2: Content survey

● Definitions and concepts
  ▪ M&S terms and attributes
  ▪ M&S categories

● Modeling methods
  ▪ Survey of modeling methods
  ▪ **Discrete event simulation**
  ▪ Monte Carlo simulation

● Special topics
  ▪ Verification, validation, and accreditation
  ▪ Distributed simulation
DES: Customers, queues, servers, and events
Example: Simple queueing system simulation
Event-driven, discrete, probability-based

- Initial condition
- $t = 0$, Customer 1 arrives, begins service
- $t = 3$, Customer 2 arrives, enters queue
- $t = 4$, Customer 3 arrives, enters queue
- $t = 7$, Customer 1 departs, Customer 2 begins service
- $t = 9$, Customer 4 arrives, enters queue

- What was the maximum queue length?
- What was the average queue length?
- What the average customer wait for service?
- What was the average customer service time?
Scope of DES

- Simulands representable as a queuing system
- Queueing system
  - Characterized by customers, queues, and servers
  - State changes discretely at events
  - Customer attributes may affect events or service times

<table>
<thead>
<tr>
<th>Simuland</th>
<th>Customers</th>
<th>Attributes</th>
<th>Servers</th>
<th>Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bank lobby</td>
<td>Customers</td>
<td>Account balance</td>
<td>Teller ATM</td>
<td>Arrival, Departure</td>
</tr>
<tr>
<td>Subway</td>
<td>Riders</td>
<td>Origin, Destination</td>
<td>Subway car</td>
<td>Arrival at station, Arrival at destination</td>
</tr>
<tr>
<td>Assembly line</td>
<td>Assemblies</td>
<td>Speed, Breakdown rate</td>
<td>Welding robot, Installation worker</td>
<td>Breakdown</td>
</tr>
<tr>
<td>Comm network</td>
<td>Messages</td>
<td>Length, Destination</td>
<td>Router, Switch</td>
<td>Arrival at destination</td>
</tr>
<tr>
<td>Field hospital</td>
<td>Wounded</td>
<td>Wound type, Blood pressure</td>
<td>Surgeon, Operating room</td>
<td>Arrival at hospital, Begin treatment</td>
</tr>
</tbody>
</table>
Basic logic of DES

- **Model status**
  - Current simulation time $\text{CLOCK} = t_0$
  - Future Event List (FEL), with next event $(e_1, t_1)$

- **Event-driven time advance algorithm**
  - Remove next event $(e_1, t_1)$ from FEL
  - Advance (set) $\text{CLOCK}$ to $t_1$
  - Process event $e_1$ per rules for event type:
    - update system state and possibly schedule future events by inserting events into FEL
  - Repeat

Time advances from event time $t_i$ to next event time $t_j$
without simulating time in between the events.
Events in a single queue, single server system

Arrival event

Departure event
Event logic: Arrival

1. Arrival event occurs at $CLOCK = t$

2. **Is $LS(t) = 1$?**
   - **No**
     - Set $LS(t) = 1$
   - **Yes**
     - Increase $LQ(t)$ by 1

3. **Server busy?**
   - **No**
     - Generate service time $s^a$; schedule new departure event at time $t + s^a$
   - **Yes**
     - Start service

4. Schedule next departure event

5. Collect statistics

6. Return control to time-advance routine to continue simulation

Start service

Enter queue

Schedule next departure event

Schedule next arrival event
Event logic: Departure

Server idle

Step 3: Set $L_S(t) = 0$

Customer in queue?

Step 3: If $L_Q(t) > 0$, reduce $L_Q(t)$ by 1.

Step 4: Generate service time $s^*$; schedule new departure event at time $t + s^*$.

Step 5: Collect statistics.

Remove customer from queue

Start service

Schedule next departure event

Return control to time-advance routine to continue simulation.
Modeling multistep processes
Randomness and random variates

- Randomness in discrete event simulation
  - Randomness used extensively in DES
  - DES randomness imitates uncertainty in real life
  - Represents system aspects not otherwise modeled, individually unpredictable but follow a pattern
  - e.g., system events (interarrival times)
  - e.g., system activities (service times)

- Random variates
  - Random values for quantities of interest
  - Generated per probability distributions that model phenomenon or process
Exponential distribution

Probability density function (pdf)

\[
f(x) = \begin{cases} 
\lambda e^{-\lambda x} & x \geq 0 \\
0 & \text{otherwise}
\end{cases}
\]

Cumulative distribution function (cdf)

\[
F(x) = \begin{cases} 
0 & x < 0 \\
\int_{0}^{x} \lambda e^{-\lambda t} \, dt = 1 - e^{-\lambda x} & x \geq 0
\end{cases}
\]

- Larger values increasingly less probable
- Parameter $\lambda$ is rate (events per time unit), $1/\lambda$ is mean interevent time
Normal distribution

Probability density function

\[
f(x) = \frac{1}{\sigma \sqrt{2\pi}} \exp\left[-\frac{1}{2} \left( \frac{x - \mu}{\sigma} \right)^2 \right]
\]

Cumulative distribution function

\[
F(x) = \int_{-\infty}^{x} \frac{1}{\sigma \sqrt{2\pi}} \exp\left[-\frac{1}{2} \left( \frac{t - \mu}{\sigma} \right)^2 \right] dt
\]

- Values clustered symmetrically around mean
- Parameters mean \( \mu \) is center, variance \( \sigma^2 \) is measure of dispersion
## Commonly used probability distributions

<table>
<thead>
<tr>
<th>Type of simuland</th>
<th>Phenomenon or process</th>
<th>Distribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>General queueing systems</td>
<td>Interarrival times</td>
<td>Exponential</td>
</tr>
<tr>
<td></td>
<td>Service times</td>
<td>Normal</td>
</tr>
<tr>
<td>Inventory and supply chain</td>
<td>Demand</td>
<td>Poisson</td>
</tr>
<tr>
<td></td>
<td>Time between demands</td>
<td>Poisson</td>
</tr>
<tr>
<td></td>
<td>Lead time</td>
<td>Gamma</td>
</tr>
<tr>
<td>Reliability and maintainability</td>
<td>Time to failure</td>
<td>Weibull</td>
</tr>
<tr>
<td>Limited data</td>
<td>Varies</td>
<td>Triangular</td>
</tr>
<tr>
<td>Varies</td>
<td>No suitable theoretical</td>
<td>Empirical</td>
</tr>
</tbody>
</table>
Part 2: Content survey

- Definitions and concepts
  - M&S terms and attributes
  - M&S categories

- Modeling methods
  - Survey of modeling methods
  - Discrete event simulation
    - Monte Carlo simulation

- Special topics
  - Verification, validation, and accreditation
  - Distributed simulation
Two definitions of Monte Carlo Simulation

**Definition 1**

- **Stochastically varying initial conditions**
  - Probability distributions used to model variability in initial conditions
- **Deterministic simulation**
  - Often time-stepped physics-based model; time represented explicitly
- **Stochastically varying results**
  - Multiple runs with run-to-run variability in results; analyzed statistically

**Definition 2**

- **Fixed initial conditions**
  - Specific known or given initial conditions
- **Stochastic simulation**
  - Probability distributions used to model variability in simuland processes; often no explicit representation of time
- **Stochastically varying results**
  - Multiple runs with run-to-run variability in results; analyzed statistically
Monte Carlo 1 procedure

Defining a Monte Carlo model
1 Identify a set of random variables that specify the initial condition.
2 Select probability distribution and parameters for each.
3 Develop deterministic model to calculate results from a set of inputs.

Executing a Monte Carlo simulation
1 Repeat for each of $n$ trials:
   1.1 Randomly generate random variate for each input.
   1.2 Calculate trial outcome with deterministic model.
   1.3 Record trial outcome.
2 Statistically analyze the results.
Monte Carlo 2 procedure

Defining a Monte Carlo model
1 Identify a set of input variables that specify the initial condition.
2 Select specific values for each.
3 Develop stochastic model based on probability distributions to calculate results from a set of inputs.

Executing a Monte Carlo simulation
1 Repeat for each of $n$ trials:
   1.1 Initialize model with selected input values.
   1.2 Calculate trial outcome with stochastic model.
   1.3 Record trial outcome.
2 Statistically analyze the results.
Example (MC1): Missile impacts [Zhang, 2008]

- **Application**
  - Deterministic 6DOF model of missile trajectory
  - Used to calculate impact point given initial conditions
  - Measure $x$ and $y$ error w.r.t. aiming point
  - Compare model and live test $x$ and $y$ error variances
  - Two ranges: 60 Km and 100 Km
  - 6 live tests, 800 Monte Carlo model trials each range

- **Monte Carlo analysis**
  - For each trial, generate initial conditions from probability distributions
  - Calculate impact point using model
  - Repeat for 800 trials
  - Compare variances
Missile trajectory initial conditions [Zhang, 2008]

- Source does not identify initial conditions
- Conjectured initial condition variables
  - Launch elevation
  - Launch azimuth
  - Engine thrust while burning
  - Engine burn time
- Setting initial condition variables
  - Appropriate probability distribution selected for each
  - Random variate for each variable for each trial
Missile trajectory model

- Physics based
- Organized into modules: velocity, rotation, atmospheric conditions, aerodynamics, thrust
- Implemented in MATLAB Simulink

**Velocity module equations**

\[
\frac{m}{dt} \frac{dV}{dt} = P \cos \alpha \cos \beta - X - mg \sin \theta
\]

\[
V \frac{d\theta}{dt} = p(\sin \alpha \cos \gamma_v + \cos \alpha \sin \beta \sin \gamma_v) + Y \cos \gamma_v - Z \sin \gamma_v - mg \cos \theta
\]

\[-mV \cos \theta \frac{d\phi_v}{dt} = P(\sin \alpha \sin \gamma_v - \cos \alpha \sin \beta \sin \gamma_v) + Y \sin \gamma_v + Z \cos \gamma_v\]

**Velocity module block diagram**
Impact data

<table>
<thead>
<tr>
<th>Trial</th>
<th>x error s</th>
<th>y error s</th>
<th>n</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>526.62</td>
<td>85.91</td>
<td>800</td>
</tr>
<tr>
<td>Simuland</td>
<td>566.66</td>
<td>89.77</td>
<td>6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Trial</th>
<th>x error s</th>
<th>y error s</th>
<th>n</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>921.39</td>
<td>111.25</td>
<td>800</td>
</tr>
<tr>
<td>Simuland</td>
<td>980.52</td>
<td>120.68</td>
<td>6</td>
</tr>
</tbody>
</table>
Example (MC2): Attrition combat [Strickland, 2011]

- **Application**
  - Force-on-force attrition
  - Direct fire only
  - No maneuver

- **Modeling**
  - Markov chain Lanchester attrition model
  - Fixed initial conditions, stochastic model → MC2

- **Notation**
  - Blue strength $B$, Red strength $R$
  - Blue lethality (vs Red) $K_B$, Red lethality (vs Blue) $K_R$
  - Blue force attrition rate $A_B(B, R) = K_R \cdot \min(B, R)$
  - Red force attrition rate $A_R(B, R) = K_B \cdot \min(B, R)$
Model details

- Model assumptions
  - No more than one shooter may engage a target at a time
  - Time between Blue casualties exponentially distributed with rate $\lambda_B = A_B(B, R) = K_R \cdot \min(B, R)$
  - Time between Red casualties exponentially distributed with rate $\lambda_A = A_R(B, R) = K_B \cdot \min(B, R)$
  - Battle ends when either side reaches break strength

- Simulation
  - Stochastically generate times of first Blue, Red casualties
  - Identify earlier casualty time and decrement side’s strength
  - Compare decremented side’s strength to break strength
  - Generate time of next casualty for decremented side
  - Repeat
# Results, single trial

<table>
<thead>
<tr>
<th>Step</th>
<th>Time</th>
<th>Blue strength</th>
<th>Red strength</th>
<th>Next Blue casualty</th>
<th>Next Red casualty</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>20</td>
<td>30</td>
<td>1.34</td>
<td>3.99</td>
<td>Continue</td>
</tr>
<tr>
<td>1</td>
<td>1.34</td>
<td>19</td>
<td>30</td>
<td>8.50</td>
<td>3.99</td>
<td>Continue</td>
</tr>
<tr>
<td>2</td>
<td>3.99</td>
<td>19</td>
<td>29</td>
<td>8.50</td>
<td>8.64</td>
<td>Continue</td>
</tr>
<tr>
<td>3</td>
<td>8.50</td>
<td>18</td>
<td>29</td>
<td>17.29</td>
<td>8.64</td>
<td>Continue</td>
</tr>
<tr>
<td>4</td>
<td>8.64</td>
<td>18</td>
<td>28</td>
<td>17.29</td>
<td>16.62</td>
<td>Continue</td>
</tr>
<tr>
<td>5</td>
<td>16.62</td>
<td>18</td>
<td>27</td>
<td>17.29</td>
<td>20.25</td>
<td>Continue</td>
</tr>
<tr>
<td>6</td>
<td>17.29</td>
<td>17</td>
<td>27</td>
<td>17.84</td>
<td>20.25</td>
<td>Continue</td>
</tr>
<tr>
<td>7</td>
<td>17.84</td>
<td>16</td>
<td>27</td>
<td>26.48</td>
<td>20.25</td>
<td>Continue</td>
</tr>
<tr>
<td>8</td>
<td>20.25</td>
<td>16</td>
<td>26</td>
<td>26.48</td>
<td>28.69</td>
<td>Continue</td>
</tr>
<tr>
<td>9</td>
<td>26.48</td>
<td>15</td>
<td>26</td>
<td>53.28</td>
<td>28.69</td>
<td>Continue</td>
</tr>
<tr>
<td>10</td>
<td>28.60</td>
<td>15</td>
<td>25</td>
<td>53.28</td>
<td>38.51</td>
<td>Continue</td>
</tr>
<tr>
<td>11</td>
<td>38.51</td>
<td>15</td>
<td>24</td>
<td>53.28</td>
<td>41.59</td>
<td>Continue</td>
</tr>
<tr>
<td>12</td>
<td>41.59</td>
<td>15</td>
<td>23</td>
<td>53.28</td>
<td>54.85</td>
<td>Continue</td>
</tr>
<tr>
<td>13</td>
<td>53.28</td>
<td>14</td>
<td>23</td>
<td>60.21</td>
<td>54.85</td>
<td>Continue</td>
</tr>
<tr>
<td>14</td>
<td>54.85</td>
<td>14</td>
<td>22</td>
<td>60.21</td>
<td>63.33</td>
<td>Continue</td>
</tr>
<tr>
<td>15</td>
<td>60.21</td>
<td>13</td>
<td>22</td>
<td>68.88</td>
<td>63.33</td>
<td>Continue</td>
</tr>
<tr>
<td>16</td>
<td>63.33</td>
<td>13</td>
<td>21</td>
<td>68.88</td>
<td>70.33</td>
<td>Continue</td>
</tr>
<tr>
<td>17</td>
<td>68.88</td>
<td>12</td>
<td>21</td>
<td>84.04</td>
<td>70.33</td>
<td>Continue</td>
</tr>
<tr>
<td>18</td>
<td>70.33</td>
<td>12</td>
<td>20</td>
<td>84.04</td>
<td>91.41</td>
<td>Continue</td>
</tr>
<tr>
<td>19</td>
<td>84.04</td>
<td>11</td>
<td>20</td>
<td>88.73</td>
<td>91.41</td>
<td>Continue</td>
</tr>
<tr>
<td>20</td>
<td>88.73</td>
<td>10</td>
<td>20</td>
<td>98.63</td>
<td>91.41</td>
<td>Red</td>
</tr>
</tbody>
</table>
### Results, multiple trials

<table>
<thead>
<tr>
<th></th>
<th>Trials</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Blue initial strength</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>Red initial strength</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>Blue break strength</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Red break strength</td>
<td>15</td>
<td>15</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td><strong>Blue lethality</strong></td>
<td><strong>0.008</strong></td>
<td><strong>0.010</strong></td>
<td><strong>0.012</strong></td>
<td></td>
</tr>
<tr>
<td>Red lethality</td>
<td>0.006</td>
<td>0.006</td>
<td>0.006</td>
<td></td>
</tr>
<tr>
<td>Blue wins</td>
<td>34</td>
<td>52</td>
<td>76</td>
<td></td>
</tr>
<tr>
<td>Red wins</td>
<td>66</td>
<td>48</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>Blue mean losses</td>
<td>8.88</td>
<td>8.08</td>
<td>7.14</td>
<td></td>
</tr>
<tr>
<td>Red mean losses</td>
<td>11.01</td>
<td>12.85</td>
<td>14.06</td>
<td></td>
</tr>
</tbody>
</table>
Part 2: Content survey

- Definitions and concepts
  - M&S terms and attributes
  - M&S categories
- Modeling methods
  - Survey of modeling methods
  - Discrete event simulation
  - Monte Carlo simulation
- Special topics
  - Verification, validation, and accreditation
  - Distributed simulation
Definition

Verification. The process of determining that a model implementation and its associated data accurately represents the developer’s conceptual description and specifications. [DOD, 2009]

- Transformational accuracy [Balci, 2002]
  - Transform specifications to code
- Software engineering quality
  - Software engineering methods apply
Definition

Validation. Determining the degree to which a model or simulation and its associated data are an accurate representation of the real world from the perspective of the intended uses of the model. [DOD, 2009]

Comparing model results with experimental data. [Velten, 2009]

- Representational accuracy [Balci, 2002]
  - Recreate simuland with results
- Modeling quality
  - Special validation methods needed
Definition

**Accreditation.** Official certification [by a responsible authority] that a model or simulation is acceptable for use for a specific purpose. [DOD, 2009]

- Official usability for specific purpose or function
  - Management decision, not technical process
  - Not a blanket or general-purpose approval
- Accrediting (or accreditation) authority
  - Agency or person responsible for use of model
  - Normally not model developer
VV&A comparisons

- Requirements
- Simuland
- Results
- Conceptual model
- Executable model

- Accreditation
- Modeling
- Validation
- Simulation
- Implementation
- Transformation
- Comparison

References:
[Petty, 2009b]
[Petty, 2010]
## VV&A errors and risks

[Balci, 1981] [Balci, 1985] [Balci, 1998]

<table>
<thead>
<tr>
<th>Model used</th>
<th>Model valid</th>
<th>Model not valid</th>
<th>Model not relevant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model used</td>
<td>Correct</td>
<td><strong>Type II error</strong>&lt;br&gt;Use of invalid model;&lt;br&gt;Incorrect V&amp;V;&lt;br&gt;Model user’s risk;&lt;strong&gt;More&lt;/strong&gt; serious error</td>
<td><strong>Type III error</strong>&lt;br&gt;Use of irrelevant model;&lt;br&gt;Accreditation mistake;&lt;br&gt;Accreditor’s risk;&lt;strong&gt;More&lt;/strong&gt; serious error</td>
</tr>
<tr>
<td>Model not used</td>
<td><strong>Type I error</strong>&lt;br&gt;Non-use of valid model;&lt;br&gt;Insufficient V&amp;V;&lt;br&gt;Model builder’s risk;&lt;strong&gt;Less&lt;/strong&gt; serious error</td>
<td>Correct</td>
<td>Correct</td>
</tr>
</tbody>
</table>
## VV&A error examples

<table>
<thead>
<tr>
<th>Model</th>
<th>Reality</th>
<th>Error</th>
</tr>
</thead>
</table>
| ![Model Image](image1.png) | ![Reality Image](image2.png) | **Type I**  
Japanese wargames  
and the Battle of Midway  
[Fuchida, 1955]  
[Barker, 1971] |
| \( \Pr[T_A < 1, T_B < 1] = \phi_2(\phi^{-1}(F_A(1)), \phi^{-1}(F_B(1)), \gamma) \) | ![Graph Image](image3.png) | **Type II**  
Gaussian copula and the  
2008 Financial Crisis  
[Salmon, 2009] |
| ![Model Image](image4.png) | ![Reality Image](image5.png) | **Type III**  
SIMNET, *Primetime Live*,  
and the 1991 Gulf War |
## V&V methods

- Different purposes, advantages, suitable models

<table>
<thead>
<tr>
<th>Informal</th>
<th>Static</th>
<th>Dynamic</th>
<th>Formal</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Audit</td>
<td>- Cause-Effect Graphing</td>
<td>- Acceptance Testing</td>
<td>- Induction</td>
</tr>
<tr>
<td>- Desk checking</td>
<td>- Control Analysis</td>
<td>- Alpha Testing</td>
<td>- Inductive Assertions</td>
</tr>
<tr>
<td>- Documentation Checking</td>
<td>- Data Analysis</td>
<td>- Assertion Checking</td>
<td>- Inference</td>
</tr>
<tr>
<td>- Face validation</td>
<td>- Fault/Failure Analysis</td>
<td>- Beta Testing</td>
<td>- Logical Deduction</td>
</tr>
<tr>
<td>- Inspections</td>
<td>- Interface Analysis</td>
<td>- Bottom-up Testing</td>
<td>- Lambda Calculus</td>
</tr>
<tr>
<td>- Reviews</td>
<td>- Semantic Analysis</td>
<td>- Comparison Testing</td>
<td>- Predicate Calculus</td>
</tr>
<tr>
<td>- Turing test</td>
<td>- Structural Analysis</td>
<td>- Statistical Techniques</td>
<td>- Predicate Transformation</td>
</tr>
<tr>
<td>- Walkthroughs</td>
<td>- Symbolic Evaluation</td>
<td>- Structural Testing</td>
<td>- Proof of Correctness</td>
</tr>
<tr>
<td></td>
<td>- Syntax Analysis</td>
<td>- Submodel/Module Testing</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Informal V&V methods

- **Characteristics**
  - Methods that rely heavily on Subject Matter Expert (SME) expertise and evaluation
  - More often qualitative and subjective
  - More often performed by SMEs
  - Effectiveness depends on SME qualifications [Balci, 2002]
  - Useful when modeling notional system [Balci, 2002]

- **Example informal V&V methods**
  - Inspection
  - Face validation
  - Turing test
Face validation (validation)

- SMEs, modelers, and users observe model execution and/or examine results
- Compare results to simuland behavior, as understood by SMEs
- Assessment
  - Model validity evaluated subjectively
  - Based on expertise, estimates, and intuition
- Comments
  - Frequently used because of simplicity
  - Often used when user interaction important
  - Clearly better than no validation
Face validation example: JOFT

- Joint Operations Feasibility Tool [Belfore, 2004]
  - Assess deployment transportation feasibility
  - Assess logistical sustainment feasibility

- Validation process
  - Special scenarios exercise full range of capabilities
  - 20 SMEs with extensive experience evaluated model
  - Assessments elicited via written questionnaires

- Process structure addressed face validation limits
Static V&V methods

• Characteristics
  ▪ Methods based on artifact characteristics that can be determined without running a simulation
  ▪ Often involve analysis of executable model code
  ▪ May be supported by automated tools or manual notations or diagrams
  ▪ More often performed by technical experts

• Example static V&V methods
  ▪ Data analysis
  ▪ Cause-effect graphing
Cause-effect graphing (validation)

- Compare causes and effects in simuland to those in conceptual model
  - Cause: event or condition
  - Effect: state change triggered by cause
- Compare simuland to conceptual model
- Identify missing, extraneous, and inconsistent cause-effect relationships
Dynamic V&V methods

• Characteristics
  ▪ Methods that involve running the executable model and assessing the results
  ▪ May compare results with simuland or other models
  ▪ More often quantitative and objective
  ▪ More often performed by technical experts

• Example dynamic V&V methods
  ▪ Execution tracing
  ▪ Sensitivity analysis
  ▪ Predictive validation
  ▪ Comparison testing
  ▪ Statistical methods
Comparison testing (verification or validation)

- Run simulations of simuland (and scenario) using two different models, compare results
- Compare results to results
- Differences between results signal problems
- Comments
  - If differences, which model has problems?
  - If one model assumed valid, validation method
  - If neither model assumed valid, verification method
Comparison testing example: Radio propagation model  [Filiposka, 2011]

- Durkin’s radio propagation model
  - Estimates radio coverage area of a transmitter
  - Models attenuation caused by diffraction
  - Considers shadowing caused by terrain
  - Predicts transmission loss using path geometry

- Validated using comparison testing
  - Durkin’s model compared to freely available Longley-Rice Irregular Terrain Model
  - Estimated radio coverage areas compared
Longley-Rice

Durkin’s

Coverage comparison
Green  = Both
Yellow = Longley-Rice only
Red    = Durkin’s only

[Filiposka, 2011]
Statistical methods (validation)

- Compare model results to simuland observations using statistical methods
  - Various statistical methods: regression analysis, analysis of variance, confidence intervals, hypothesis tests, others [Balci, 1998] [Petty, 2010]
  - May be used in combination with other methods
- Compare results to simuland
- Comments
  - Each statistical method defines statistic or metric of “closeness” or similarity; measure of validity
  - Generally underutilized
  - Selecting method requires knowledge of assumptions
# Example applications of statistical methods

<table>
<thead>
<tr>
<th>Model(s)</th>
<th>Statistical method</th>
<th>Reason for selection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spacecraft propulsion system sizing</td>
<td>Linear regression</td>
<td>Paired data, simuland–model</td>
</tr>
<tr>
<td>Medical clinic waiting</td>
<td>Confidence intervals</td>
<td>Single simuland observation, multiple model runs</td>
</tr>
<tr>
<td>Seaport loading/unloading</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Historical tank battle</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bombing accuracy</td>
<td>Confidence intervals with error tolerance</td>
<td>Single simuland observation, multiple model runs, error tolerance available</td>
</tr>
<tr>
<td>Waiting line</td>
<td>Hypothesis test for equivalence of means</td>
<td>Multiple simuland observations, multiple model runs</td>
</tr>
<tr>
<td>Ground combat</td>
<td>Hypothesis test for equivalence of means</td>
<td>Multiple simuland observations, multiple model runs, equality not assumed</td>
</tr>
<tr>
<td>Commander decision making</td>
<td>Hypothesis test for equivalence of variances</td>
<td>Multiple simuland observations, multiple model runs</td>
</tr>
<tr>
<td>Missile impact accuracy</td>
<td>Hypothesis test for equivalence of variances</td>
<td>Multiple simuland observations, multiple model runs</td>
</tr>
<tr>
<td>Wastewater treatment facility</td>
<td>Confidence interval for difference of two means</td>
<td>Multiple simuland observations, multiple model runs, variances not equal</td>
</tr>
<tr>
<td>Sewage collection site</td>
<td>Hypothesis test for equivalence of variances</td>
<td>Multiple simuland observations, multiple model runs</td>
</tr>
<tr>
<td></td>
<td>Hypothesis test for equivalence of means</td>
<td>Multiple simuland observations, multiple model runs, variances equal</td>
</tr>
</tbody>
</table>
Formal V&V methods

• Characteristics
  ▪ Methods based on formal mathematical proofs of program correctness
  ▪ Quantitative (or logical) and objective
  ▪ Performed by technical experts
  ▪ Difficult to apply in practice [Balci, 1998]

• Example formal V&V methods
  ▪ Inductive assertions
  ▪ Predicate calculus
Predicate calculus (validation)

- Logically analyze conceptual model
  - Predicate calculus is a formal logic system
  - Create, manipulate, and prove statements
  - Simuland, conceptual model described in pred calc
  - Prove properties of both to show logical consistence
- Compare conceptual model to simuland
- Quite difficult to apply to non-trivial problems

\[
(\forall x) [D(x) \rightarrow (\forall y)(R(y) \rightarrow C(x, y))] \\
(\exists x) [D(x) \land (\forall y)(R(y) \rightarrow C(x, y))] \\
(\forall y) [R(y) \rightarrow (\forall x)(C(x, y) \rightarrow D(x))] \\
(\forall x)(\forall y) [R(y) \land C(x, y) \rightarrow D(x)]
\]

Last two: “Only dogs chase rabbits.” [Gersting, 2003]
Part 2: Content survey

- Definitions and concepts
  - M&S terms and attributes
  - M&S categories

- Modeling methods
  - Survey of modeling methods
  - Discrete event simulation
  - Monte Carlo simulation

- Special topics
  - Verification, validation, and accreditation
  - Distributed simulation
Example distributed simulation: America’s Army

- Recruiting and familiarization tool for U. S. Army
- Multiplayer online game, linked via Internet
- First person shooter
- 13M registered users (2014)
Definition

Distributed simulation. Multiple collaborating simulations distributed across locations, computers, and/or processes.

Distributed simulations typically
- Cooperatively simulate simuland
- Each simulates some portion of simuland
- Exchange data about simuland via network messages
Definitions

• **Interoperability;** the ability of models to meaningfully communicate in a distributed simulation

• **Composability;** the ability to combine and recombine models and model components into different complex simulations
Distributed simulation system components

- Models/Simulations (simulation nodes)
- Utilities (non-simulation support nodes)
- Network and protocol
Definition

**Distributed simulation protocol.** Network protocol designed to support a category of distributed simulation systems.

General protocol characteristics

- Definitions of
  - Data items
  - Message formats
  - Interaction sequences
- Standardized to support interoperability
Distributed simulation interoperability protocols

- **Simulator Networking SIMNET**
  - First functional distributed simulation protocol
  - Homogenous, entity-level, mostly virtual

- **Distributed Interactive Simulation DIS**
  - Expanded capabilities w.r.t. SIMNET
  - Heterogeneous, entity-level, mostly virtual

- **Aggregate Level Simulation Protocol ALSP**
  - Heterogeneous logical time constructive

- **High Level Architecture HLA**
  - General purpose, subsumes previous protocols

- **Test and Training Enabling Architecture TENA**
  - Designed with test range applications in mind
Distributed simulation protocol development

- **SIMNET**
  - Virtual; real-time; entity level; 1980s

- **DIS**
  - Virtual; real-time; entity level; 1990s

- **TENA**
  - Ranges; real-time; entity level; 2000+

- **HLA**
  - General purpose; 1995+

- **ALSP**
  - Constructive; logical-time; aggregate level; 1990s
Distributed simulation protocol: Distributed Interactive Simulation (DIS) [IEEE, 1995]

- Development history
  - Developed from SIMNET, beginning early 1990s
  - Exploited lessons learned from SIMNET

- Characteristics
  - Mounted combat (primarily)
  - Distributed, virtual, entity level, real-time
  - Heterogeneous, non-proprietary
  - Open protocol standard development process

- Used for multiple simulation systems
Basic concepts of DIS

- Simulation nodes
  - Multiple distributed simulators, simulations, utilities
  - Exchange messages via a network (LAN)

- Network messages
  - Conform to predefined standard protocol
  - Called Protocol Data Units (PDUs)
  - Transmitted broadcast (UDP/IP, TCP/IP)

- Message purposes
  - Report entity state (movement, status)
  - Mediate interactions between entities
  - Manage or control simulation execution
Main parts of DIS protocol [Loper, 1995]

- Data items to be passed
- Format of data items
  - e.g., int vs. float, value enumerations
- Grouping of data items into messages (PDUs)
- Conditions for sending PDUs
  - Specific to PDU type
- Processing to perform upon receiving PDUs
  - Specific to PDU type
- Key algorithms to be shared among nodes
  - e.g., dead reckoning
Most common DIS PDU types

- **Entity State**
  - Announce entity existence, location, movement, and appearance

- **Fire**
  - Announce that entity has fired a weapon
  - Important for rendering muzzle flashes

- **Detonation**
  - Announce that round has hit entity or terrain

- **Collision**
  - Exchanged between colliding entities
Example DIS interaction: direct fire

- Entities interact by exchanging PDUs
- Protocol defines PDU sequence for interaction
Distributed simulation protocol:
High Level Architecture [Dahmann, 1998b] [Möller, 2012]

• Architecture
  ▪ Distributed simulation systems assembled by connecting nodes via network and protocol

• Flexibility
  ▪ No fixed protocol can serve all users’ needs, nor can all future applications be anticipated
  ▪ Protocol must allow customization
  ▪ Intended to be general purpose protocol [Dahmann, 1998a]

• Separation of functionality
  ▪ Application-specific (i.e., data definition)
  ▪ General infrastructure (i.e., data transport)
HLA specifications

- **Rules** [IEEE, 2010a]
- **Object Model Template** [IEEE, 2010c]
- **Interface Specification** [IEEE, 2010b]
HLA terms

- **Federate**: individual node in distributed simulation system (simulation or utility)
- **Federation**: set of interoperating nodes
• Object Model; specification of data to be exchanged by a federation
• Run-Time Infrastructure (RTI); software that supports exchange of data in federation
• RTI service; specific capability provided by RTI
HLA Rules

- Define responsibilities and restrictions
- 10 rules total
- 5 rules each for federates and federations
HLA object models

- Define data to be sent and received in federation
- Object classes and attributes
  - Persistent objects
  - Hierarchy, single inheritance
- Interaction classes and parameters
  - Non-persistent interactions between objects
  - Hierarchy, single inheritance
- Documented per Object Model Template
- Similar, not same, as “object-oriented”
- Special object models
  - Federation Object Model (FOM); OM for federation
  - Simulation Object Model (SOM); OM for federate
HLA Interface Specification

• Purpose
  ▪ Formal definition of operations (“services”) used to exchange simulation and control information in a federation execution
  ▪ Formal specification of interface between RTI and federates, defined as a set of functions with API

• Interface Specification and the RTI
  ▪ Interface Spec; defines services and software interface to use them
  ▪ RTI; implements and executes the services
<table>
<thead>
<tr>
<th>Service Category</th>
<th>Functionality</th>
<th>Services 1516-2010</th>
</tr>
</thead>
<tbody>
<tr>
<td>Federation Management</td>
<td>Create, control, destroy federation executions</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>Join and resign federation executions</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Pause, resume, checkpoint, restart</td>
<td></td>
</tr>
<tr>
<td>Declaration Management</td>
<td>Announce intent to send or receive object and interaction information</td>
<td>12</td>
</tr>
<tr>
<td>Object Management</td>
<td>Create and delete objects</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>Send and receive object attribute updates</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Send and receive interactions</td>
<td></td>
</tr>
<tr>
<td>Ownership Management</td>
<td>Transfer ownership of object attributes between federates</td>
<td>18</td>
</tr>
<tr>
<td>Time Management</td>
<td>Control and synchronize simulation time</td>
<td>23</td>
</tr>
<tr>
<td>Data Distribution Management</td>
<td>Filter data sent between federates</td>
<td>12</td>
</tr>
<tr>
<td>Support</td>
<td>Provide infrastructure status information to federates</td>
<td>43</td>
</tr>
</tbody>
</table>
Run-Time Infrastructure (RTI)

- Not a part of the definition of HLA
- Software realization of the HLA definition
- Provides run-time support to federation
  - Transports data between federates
  - Controls federation execution
  - Manages simulation time
Logical view of a federation

- Federates send data to and receive data from RTI, via services
- RTI is intermediary between federates
Technical view of a federation

- LRC integrated into each federate
- Federate passes data to/from LRC via services; LRCs exchange data via network
- CRC handles special services

LRC = Local RTI Component  CRC = Central RTI Component
HLA standards

- **DoD 1.3**
  - Original HLA standard
  - Initial RTI and HLA software implemented in DoD 1.3
  - DoD 1.3 software no longer supported
  - No longer in use?

- **IEEE 1516-2000**
  - Developed from DoD 1.3
  - Many improvements [DMSO, 2004] [Morse, 2004]
  - Widely used
  - Federates, federations, tools, products available

- **HLA 1516-2010**
  - Developed from IEEE 1516
  - Standardized 2010 [IEEE, 2010a] [IEEE, 2010b] [IEEE, 2010c]
Conclusion
Summary

- CMSP: professional certification for M&S
- Renewed exam
  - Topical coverage of M&S Body of Knowledge
  - Questions traceable to authoritative sources
- Examination delivered via custom web system
- Content comprehensive of M&S
References


End notes

• Additional information regarding exam structure

• Additional information regarding exam content
  • Online UAH PCS courses: http://www.uah.edu/pcs/professional-development/engineering?target=index&categoryId=10416
  • Custom short courses

• Contact information
  • Mikel D. Petty, Ph.D.
  • University of Alabama in Huntsville
  • 256-824-6140 pettym@uah.edu