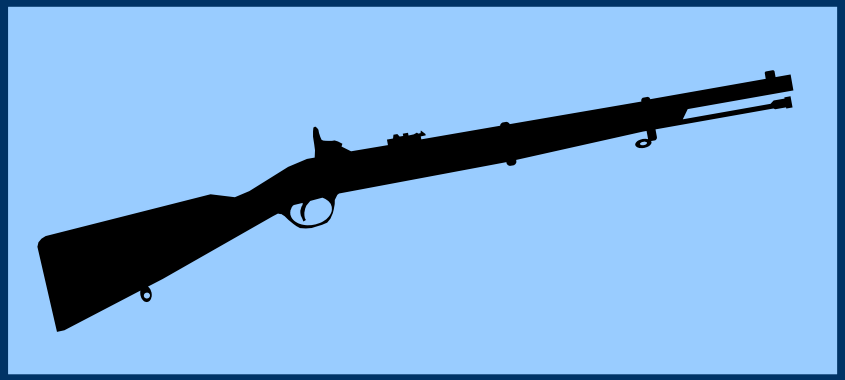
Fast, Free and Fun: Lightweight Simulation Modelling with Python

Dr John D Salt
Musketoon Limited



musketoonltd@gmail.com

Contents

- Prologue: Simulation Languages to 1990
- Free Open Source Software and Python
- Experiences with Python
- Need for a Network Editing Framework
- Inspiration
- Vision
- Disadvantages and Blockers
- Conclusion

Simulation Programming Languages to 1990

- GPSS
  - IBM, c. 1961
- SIMSCRIPT
  - RAND Corporation, c. 1965
  - CACI
- Simula
  - Norwegian NCC, 1968
- MODSIM
  - DoD/CACI, c. 1989

Free Open Source Software and Python

- Python offers:
  - Full object-orientation with multiple inheritance
  - Strict but dynamic typing
  - Easy object persistence with pickle()
  - The best available PRNG
- NumPy adds:
  - A variety of probability distributions
- SimPy adds:
  - Co-routines for process-based simulation
  - Resources and stats collection queues
- Several graphics packages to choose from

Experiences with Python

- BFT – Blue Force Tracker traffic
- ANTFARM – Cluster head election
- BG IERs – Really about Falcon
- PWAS – Really base protection surveillance
- WAPITI 1 and 2 – Tactical Internet topology
- RADIX – Role-Activity Diagrams
- PNs 1 and 2 – Petri Nets
- GAWM – Geoint analysts' workload sharing

Need for a Graphical Network Editor

- Directed graphs can represent many things
  - Communications nets
  - Command structures
  - Work flows
  - Equipment associations
- Generic Arc-Node Editing Framework:
  - GANEF – MODSIM + SIMGRAPHICS
  - GANEF 2 – Java + DEC EZGraphics
  - GANEF 3 – Python + TkInter + TkZinc
  - GANEF 4 – Python + PyQt4

Choice of Graphics Package

- TkInter comes with the standard Cpython release
  - It offers bindings to Tcl/Tk
  - TkZinc is an improved third-party graphical canvas
- PyQt was a Nokia product, now from Riverside
  - It offers bindings to C++
  - PySide is intended to offer similar functionality under a more permissive licence (LGPL)
- As neither is pure Python, there are problems with pickle()

The Inspiration

- RAD/JAD/XP/Agile/Evolutionary development
  - Stress on user participation
- The Cathedral and the Bazaar
  - Eric Raymond, elements of hacker culture
- Distillations
  - Project Albert
- Scissions
  - Ray Paul
- Modes of Practice
  - Stewart Robinson

The Vision

- Brutal Simplification
  - St. Exupéry's idea of perfection
- Complete openness
- Putting the simulation into the user's hands
- Graphical interactive construction and execution
  - "If it's not interactive, it's not fun"
  - The best V&V method I know
- Executable diagrams
  - Architecture is useless if it just sits there

9

Disadvantages and Blockers

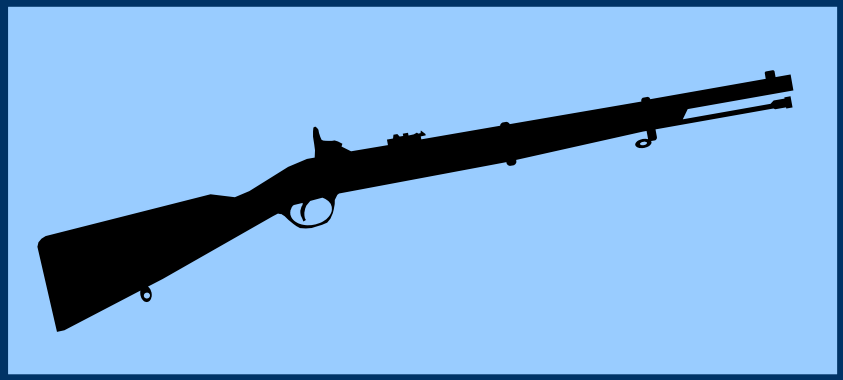
- Simulation is unpopular
  - If not in decline, at least under-appreciated
- "Need to know" rather than "Duty to share"
  - especially in defence
- FUD about FOSS
  - "If it's free, it must be worth what I paid for it"
- "No point re-inventing the wheel"
  - learning value of model construction not seen
- Organisational drag
- Not wanting to get one's hands dirty

Conclusion

- Technical obstacles can all be overcome
- Political/organisational/attitudinal obstacles are harder
- Is anyone else doing this?

Questions?

Dr John D Salt
Musketoon Limited



musketoonltd@gmail.com