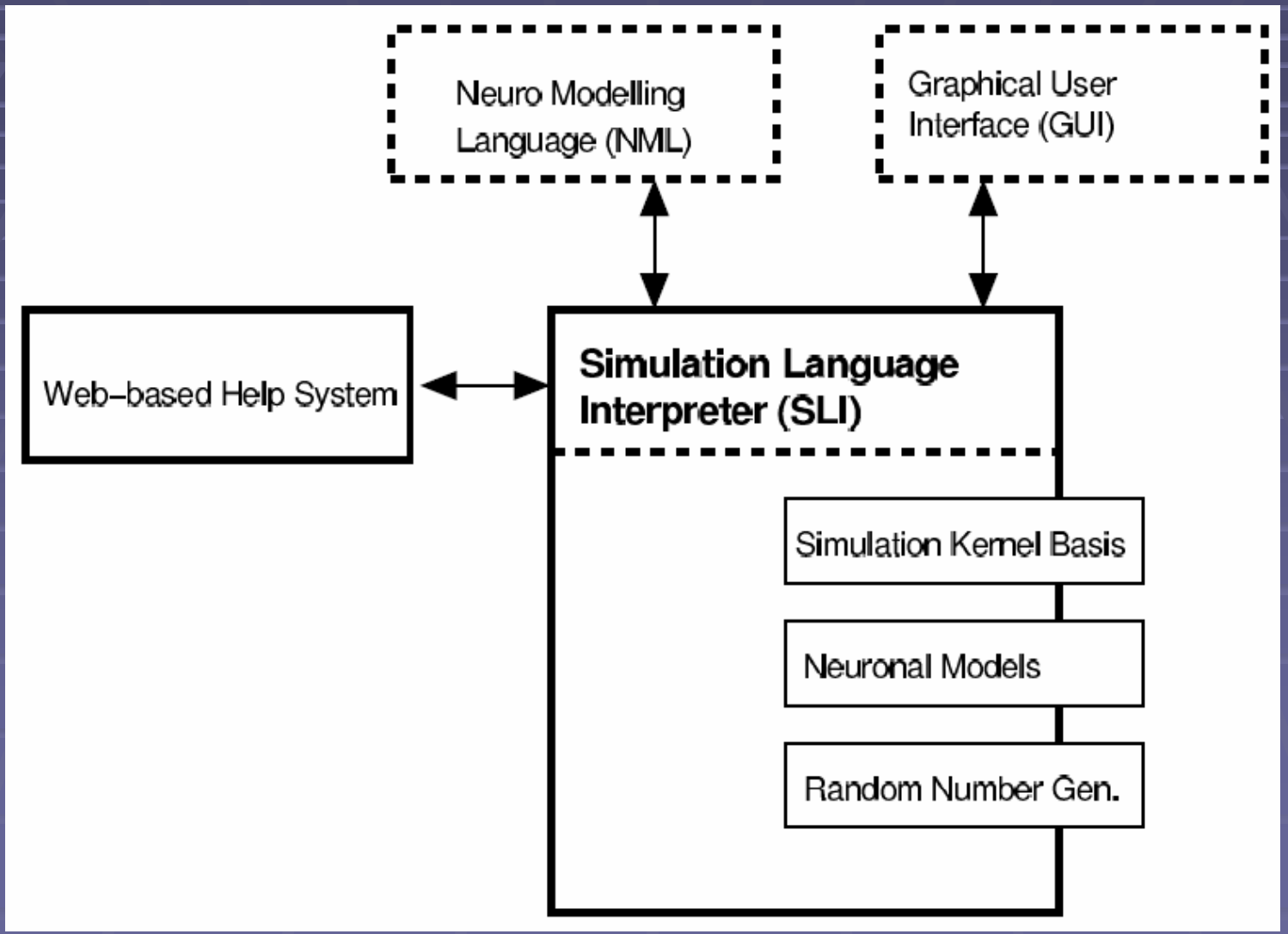
An Introduction to NEST

Amir Reza Saffari Azar Alamdari
Institute for Theoretical Computer Science,
Graz University of Technology
amir@igi.tugraz.at

Overview

- NEST: NEural Simulation Technology
- Simulation of large networks of spiking point-neuron models
- No easy way to integrate morphology or geometry of neurons
- Flexible for design of network architecture
- Changes to internal working of a neuron model (or other components) should be done at C++ level and requires recompilation

Structure of NEST



SLI

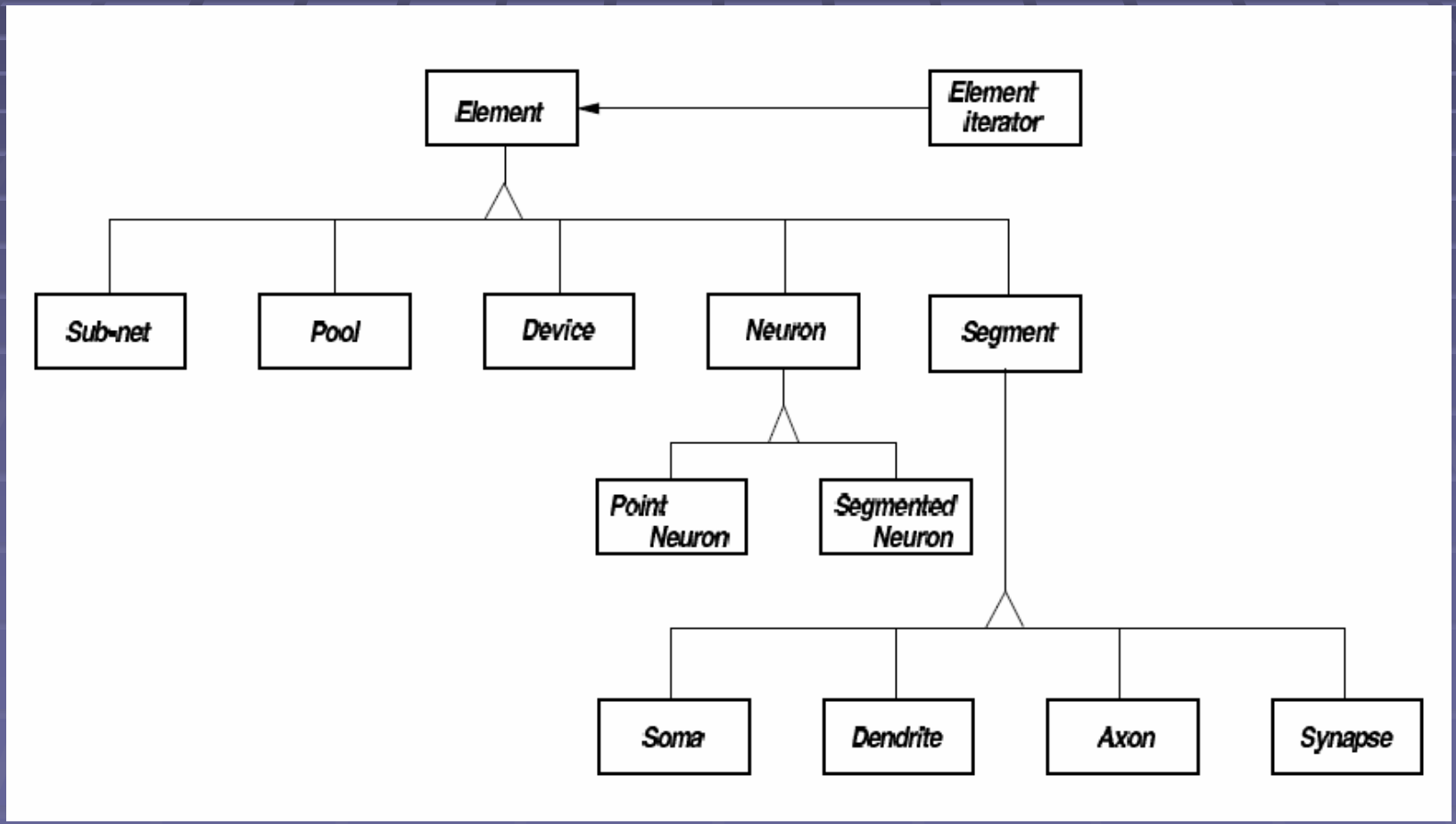
- NEST is written in OOP style in C++
- SLI: Simulation Language Interpreter
- SLI provides primary user interface
- SLI also provides a set of basic data structures used to exchange data between modules
- SLI is a simple stack machine, optimized for fast processing of higher-level commands and objects

Simulation Kernel

- Simulation kernel: definition and simulation of large structured networks of biologically realistic neural networks
- Elements: objects like neurons, synapses, columnar circuits, whole cortical area, and devices
- Elements are hard-coded in C++, but configurations are dynamic at SLI level

Network Elements

- Element hierarchy



Configuration Interface for Elements

- The element base class defines the minimal interface that each element has to implement
- It is not trivial to define a generic interface for all neurons which can be used by interpreter to access an element‘s internal parameters
- The solution used is keyword parameters, using SET and GET commands

Simulation Driver

- Tasks:
  - Administration of network structure
  - Administration of network elements
  - Organizing the temporal update of each element
  - Communicating information between elements

Update and Communication

- Element update: fixed simulation step size
- Common interface for update
- Spike events: during update a neuron model combines all incoming spikes and determines its output
- When spike is generated, it is simultaneously sent to all target neurons
- The dynamics of the synapses are placed on the post-synaptic side

Time Driven vs Event Driven

- Event driven: useful where communication is cheaper than evaluation of each element
- Time driven: useful where evaluation of each element is cheaper than communication
- Hybrid: updating neurons with every time step, synapses with every spike

Comparision

	NEST	CSIM
Major Goal	Large Scale SNN	Large Scale SNN
Core	C++	C++
Interface	SLI	MATLAB
Neuron Models	I&F models with Different PSPs	Most of Spiking Neuron Models and also Sigmoidal Neurons
Synapse Models	None	Dynamic and Static Models
Synaptic Plasticity	None	STDP and Other Learning Mech.

Comparision

	NEST	CSIM
Visualization Tools	None	Provided Through MATLAB
Higher Level Functions	None	NMC and Learning
Simulation Strategy	Hybrid	Hybrid
Parallelization	Yes, but not available for public	None
Documentation	Good	Not so much
Download	Indirect (GPL Based)	Direct (GPL)

References

- Markus Diesmann, Marc-Oliver Gewaltig, "**NEST: An Environment for Neural Systems Simulations**", In Theo Plesser, editor, *Forschung und wissenschaftliches Rechnen, Beiträge zum Heinz-Billing-Preis 2001, GWDG Bericht Nr.58*. Ges. für Wiss. Datenverarbeitung, Göttingen, 2003

- "**NEST Helpdesk**", http://www.nest-initiative.uni-freiburg.de/helpdesk/synod2/doc/

- Marc-Oliver Gewaltig, "**NEST: The Impatient Modeller's Guide**", 2004

- Abigail Morrison, Carsten Mehring, Theo Geisel, Ad Aertsen, Markus Diesmann, "**Advancing the boundaries of high connectivity network simulation with distributed computing**", *Neural Computation*, 2005 (in press)

- "**FACETS Discussion Forum**", http://groups.yahoo.com/group/facetsproject/

Some Examples

- A very short introduction to SLI
- Simulating an integrate-and-fire neuron
- Simulating a network of spiking neurons