GPUnit

Daniel Bagnell
Jason Economou
Rajkumar Jayachandran
Tim McJilton
Gabriel Schwartz
Andrew Sherman

Advisor: Prof. Jeremy Johnson
Stakeholders: Prof. Steve McMillan
Alfred Whitehead
The Leiden Observatory

January 25, 2011

Overview

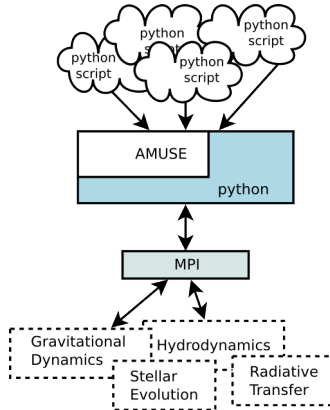# Astrophysical Multipurpose Software Environment (AMUSE)

Figure: AMUSE Architecture

State of AMUSE

- Currently used by researchers to run large-scale simulations.
- Scripts, diagnostics, logging are all written by hand.
- AMUSE API/programming knowledge is required to create experiments.

AMUSE Introduction
Purpose
Components and Features
User Interface
Final Goals

Purpose of GPUnit
Target Audiences

Purpose of GPUnit

- Ease the use of AMUSE
- Create/Design/Modify experiments
- Select, configure, swap out modules and initial conditions

AMUSE Introduction
Purpose
Components and Features
User Interface
Final Goals

Purpose of GPUnit
Target Audiences

Target Audiences

- ▶ Physics Students
- ▶ Observational Astrophysicists
- ▶ Theoretical Astrophysicists

AMUSE Introduction
Purpose
Components and Features
User Interface
Final Goals

Experiment Management
Diagnostics
Command Line Interface
Parallelism and Distribution

Experiment Management

- ▶ Experiments include a specification for:
  - ▶ Initial Conditions, Variables
  - ▶ Module Selection
  - ▶ Logging
  - ▶ Diagnostics
- ▶ Allows experiments to be repeated and shared.

AMUSE Introduction
Purpose
Components and Features
User Interface
Final Goals

Experiment Management
Diagnostics
Command Line Interface
Parallelism and Distribution

Diagnostics

- Diagnostics read the state of the system while it is computing, and output selected metrics.
- Features include:
  - Configurable output data formats
  - Sensible default diagnostics provided
  - An interface to create custom diagnostics
- Diagnostics can include a visual display of the system state.

AMUSE Introduction
Purpose
Components and Features
User Interface
Final Goals

Experiment Management
Diagnostics
Command Line Interface
Parallelism and Distribution

Command Line Interface

- The command line interface gives the user ability to:
  - Make small changes to the experiment before running
  - Run headless experiments
  - Control experiment distribution across a cluster
  - Easily repeat experiments

AMUSE Introduction
Purpose
Components and Features
User Interface
Final Goals

Experiment Management
Diagnostics
Command Line Interface
Parallelism and Distribution

Parallelism and Distribution

- ▶ Modules are independent, can run at the same time.
- ▶ No need to share data until the end of a time interval.
- ▶ Some exceptions if a special event occurs
- ▶ User interaction can be required.
- ▶ GUI will provide a display of all nodes on the cluster.
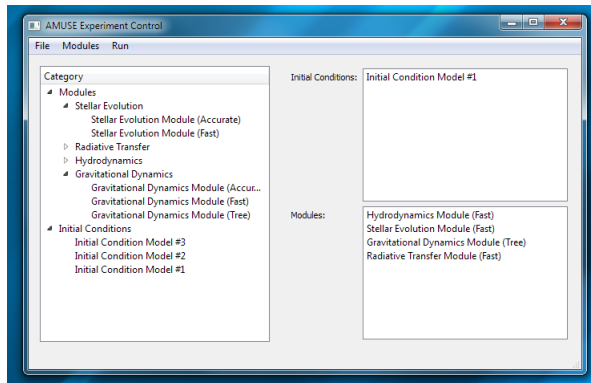- ▶ Command line provides a list of free nodes if desired.

AMUSE Introduction
Purpose
Components and Features
User Interface
Final Goals

Cluster Control
Cluster Control
Module Specification

Experiment Editor

Figure: Experiment Editor

AMUSE Introduction
Purpose
Components and Features
User Interface
Final Goals

Cluster Control
Cluster Control
Module Specification

Cluster Control

Figure: Cluster View

AMUSE Introduction
Purpose
Components and Features
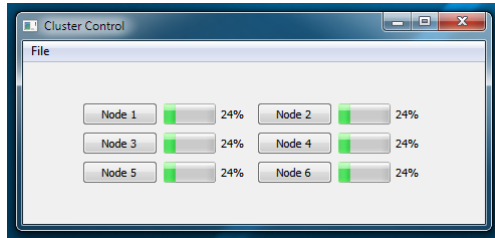**User Interface**
Final Goals

Cluster Control
Cluster Control
Module Specification

Module Specification
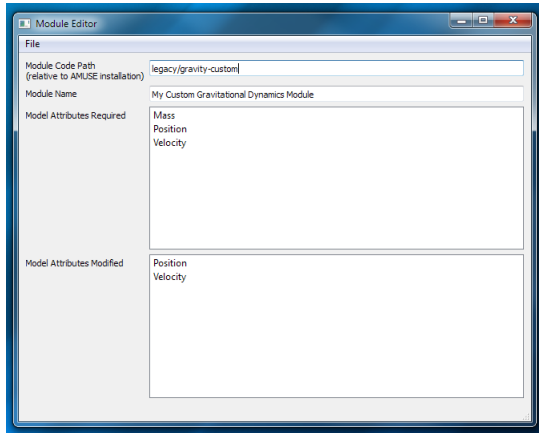

Figure: Module Editor

Final Goals

- Integration into AMUSE
- Users can download AMUSE and create/run experiments right away.
- Open AMUSE up for use by a non-programmer audience.