



(and related projects)

Sorry for using Keynote!

It's not open source :(

✕ Brian Tanner, Adam White
University of Alberta

About Me

- Ph.D student at the University of Alberta
- Work in reinforcement learning
- Not expert in supervised machine learning
- Enjoy creating useful software
- Passion for really good experiments
 - Drives me to find ways to make other people do good experiments

Sharing Code

- Many research groups all over the world
- All implementing different algorithms, creating different problems and data sets
- We should share!
 - Reduce duplicating code over and over
 - Reproduce results
 - Squash bugs with long-term code vetting

Why it's hard

- Everyone prefers a different platform
 - Linux, Unix, Mac, Windows, etc
- Everyone prefers a different language
 - C/C++, Java, Python, Lisp, Matlab, etc
- Everyone creates their own software framework with different conventions
- How could this possibly work?

Shared Protocol

- There has been success in supervised machine learning!
 - Although after this morning's discussion I think less than I thought
- Agree on a standardized language for input and output
 - Write import/export modules between your code and the standard language
 - Publish “good” code

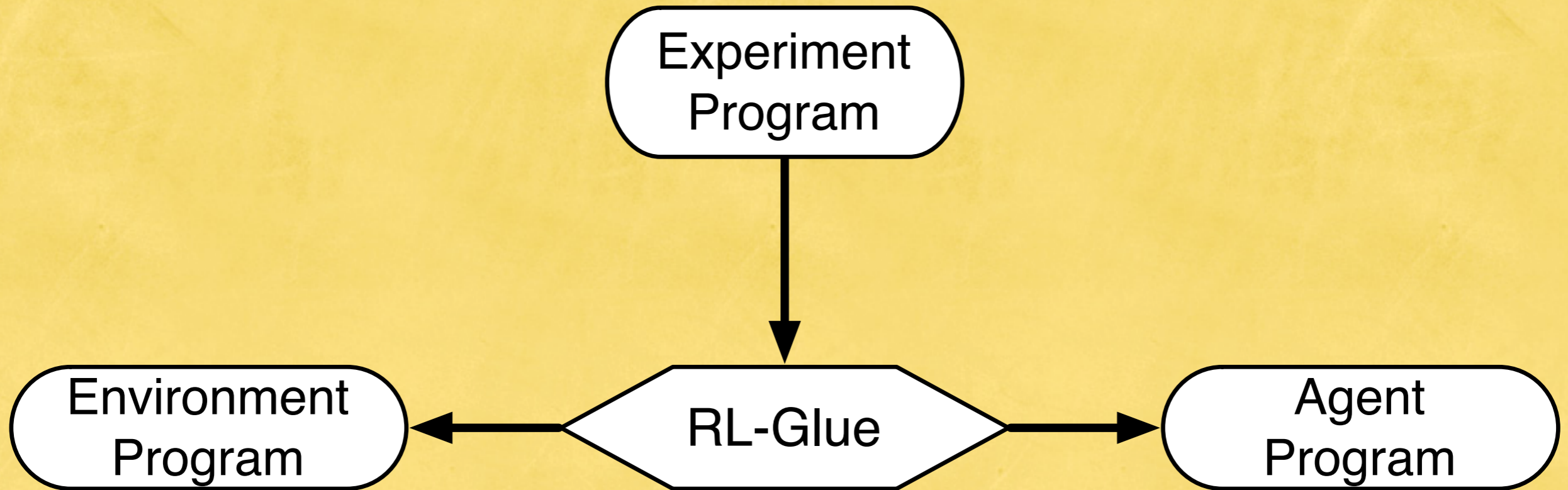
Online Reinforcement Learning

- No data sets, input is not IID
- Environments and agents are active
 - Environments create stochastic trajectories over time depending on actions chosen by the agent
 - Agents choose actions in order to learn ways to maximize a reward signal
- Environment and agents are **programs**

Agents/Environments As Programs

- Active nature of agents and environments requires a different approach
- Cannot import/export passive data
- We import/export a functional interface
- Provide infrastructure to the reinforcement learning community that connects these interfaces together
- We call it “RL-Glue”

The Picture



- Each of agent/environment/experiment is a program
- RL-Glue connects them via inter-process communication

Design Requirements

- Low barrier to be RL-Glue compatible
 - Absolutely minimal interface
 - Absolutely minimal dependencies
- Cross-platform compatibility
- Inter-language compatibility
- All this while maintaining flexibility

Environment Interface

```
task_description    <== env_init()
observation         <== env_start()
observation, reward <== env_step(action)
                   env_cleanup()
return_message     <== env_message(message);
```


RL-Glue Socket Server

- **rl_glue** is a TCP/IP socket server
 - written in C (not C++)
 - no external library dependencies
 - compiles on (m)any platform(s)
- Agents, environments, experiment programs all connect to **rl_glue**

Connecting to rl_glue

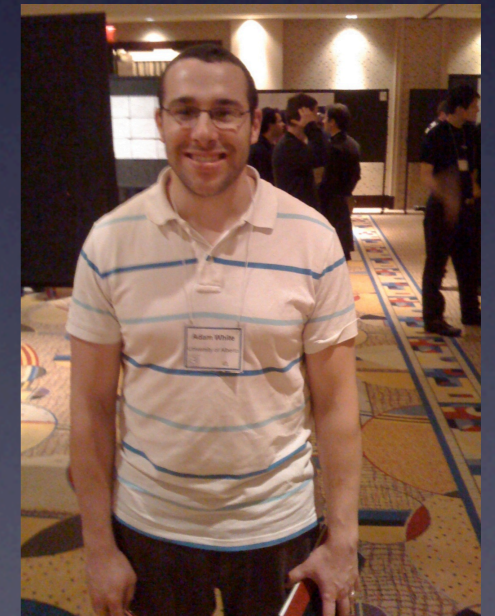
- Each supported language has a specific implementation of the simple interface
- Users write their agents/environments to that interface
- We (developers) write a socket “codec” that converts from that language to a neutral data format for the network
- Codec links/wraps user code and makes it work with rl_glue server

Cross Language

- Codecs have been written and tested for:
 - C/C++
 - Java
 - Python
 - Matlab
 - Lisp

Brief History

- Rich Sutton (my supervisor)
 - Advocating RL-Glue-like ideas for years
- Adam White (another student), formalized and created RL-Glue 1.0 for his Masters thesis



Current Project Details

- We are in a release candidate phase with RL-Glue 3.0
 - Final release planned for January 31/2009
- Released under the Apache 2 license
- It is a Google Code project
- Homepage <http://glue.rl-community.org>

RL-Library

- Incentive for users to create agents/ environments compatible with RL-Glue
- Open source collection of agents and environments from the RL literature
 - Easily build on the work of others
 - Reduce programming burden

RL-Viz

- Layered protocol through the messaging system of RL-Glue
- Allows for fancy features
 - dynamic loading of agents/environment
 - visualization
 - run-time configuration

Some Success Stories

- RL-Competition is an annual event that is gaining momentum in the RL-Community
- Our project has encouraged people to share projects that were not previously published
 - Helicopter hovering simulator
 - Simulated Octopus Arm
 - CritterBot robotic platform \Leftarrow real robot!
 - Atari games via emulator



More Success

- RL-Glue has opened a dialogue in the community
 - exposing the differing assumptions and methodologies in the community
 - iterative feedback allows us to make RL-Glue more useful and practically applicable

Demo

Looking Forward...

- RL-Logbook
 - Large scale database of reinforcement learning results
 - All results involve open source agents and environments
 - Computation can be distributed to people in the community with idle CPU cycles or to on-demand grid computing resources like Amazon Compute Cloud
 - Come see my poster at parallel workshop tomorrow!