A Brief History of Pavlov

1. Pavlov's History

Pavlov has been in private development for 4 years. Bored with writing and flipping flash cards for the United States Coast Guard's Captain's license, I wrote a Perl Script to do it and hacked in statistical choice of questions and question history logging.

After a hard disk crash in 2000, I (grudgingly) reimplemented it in Java.

Noticing how successful HotOrNot.com was at keeping users (i.e. me and my neighbor Josh) clicking their buttons, the idea occured to me to have my program show pictures of bikini-babes when the user got correct answers. This proved extremely successful with my Navy shipmates who borrowed Pavlov to study for advancement examinations and Enlisted Surface Warfare boards.

As the program grew in complexity, I went through several stages of refactoring and API refinement. Writing plugin support for "question selection strategies" and "feedback pluglets" allowed me to simultaneously decrease the core source code complexity and increase its feature set. Now, it's not "bikini-babe-centric," anyone who can write a normal applet can use his imagination and write a feedback pluglet to provide positive and negative support based on a wide variety of historical data that's stored about the user.

> **Note:**
> With Pavlov 1.1X, you don't need to be able to write a Java program to create a pluglet. Now you can do it using HTML and some minor programmatic content using the Velocity Template Language.

Question selection strategies, similarly, are only a matter of a programmer using his imagination, writing a simple program, and dropping the class or JAR file into the `pluglets/strategy/` directory.

The core source is hovering around 130 Java classes and 14,000 lines of code. Heavily used technologies include XML, Swing, and runtime class loading. It has been the target of several bouts of intense refactoring. Abstraction and design patterns are the rule rather than the exception throughout the code.

> **Note:**
> With 1.1B1 it's more like 100 classes and 16,000 lines of code.

Pavlov's evolution has resulted in several packages suitable for general use that are available under the GNU General Public License (GPL).

- `pavlov.randommedia` is a (more-or-less) polished API for picking random or sequenced media objects (images/sound/html/etc) from directory trees.
- `net.sourceforge.steelme` (formerly `pavlov.themes`) is a polished API for user customizable GUI color and font themes.
- `pavlov.pluglets` would be very easy to use in any other Swing program to provide pluggable tools.
- `pavlov.startup` provides slick & GPL'd startup windows for Java applications.

My main goals in sourceforging this project are:

1. make the Pavlov program and API's available to users and programmer-users,
2. get suggestions/assistance on problems remaining in the code and/or architecture,
3. provide a home for pluglet developers to create and distribute feedback & question strategy pluglets,
4. to provide a home for the collection and distribution of content: Library files of questions, and,
5. to encourage different implementations for abstract classes, i.e. DBMS storage of questions and statistics, a servlet Abstract UIFactory implementation, PDA-friendly implementations, etc.

If you've read this far, you are either a very bored person or a potential contributor. If you'd like to take part of Pavlov and run with it, whether it be part of the core, pluglets, writing a "book" of questions, documentation, a study on Pavlov's efficacy in the classroom, or whatever, there's plenty of room in the project for you. Any contributor who would like to get his or her name mentioned has but to ask.