# *(...aside)*

comments on and
implications of
a quasi-framework

David A. Black
RubyConf 2003

# *(...aside)*

- Goals
  - Allow students to give anonymous feedback to professors

# *(...aside)*

- Goals
  - Allow students to give anonymous feedback to professors
  - Allow professors to respond, without knowing who the student is

# *(...aside)*

- Goals
  - Allow students to give anonymous feedback to professors
  - Allow professors to respond, without knowing who the student is
  - Support arbitrarily long exchanges in this “one-way mirror” fashion

# *(...aside)*

- Goals
  - Allow students to give anonymous feedback to professors
  - Allow professors to respond, without knowing who the student is
  - Support arbitrarily long exchanges in this "one-way mirror" fashion
  - Provide David with an actual Ruby project to work on

# Design of *(...aside)*

- "Scenarios"
  - Password given
  - Session ID given
    - session file exists (or not)
  - No authorization info
- Take action based on legitimacy of scenario

# Implementation of *(...aside)*

- CGI/RubLog::Template
- YAML storage of session info and messages
- Fairly abstract “scenario”-based callback structure

# *(...aside)*: program or framework?

- Reinventing the wheel vs. just writing code

# *(...aside)*: program or framework?

- Reinventing the wheel vs. just writing code
- How much to encapsulate and abstract...
  - for the program's sake?
  - for other programmers' sake?

# *(...aside)*: program or framework?

- Reinventing the wheel vs. just writing code
- How much to encapsulate and abstract...
  - for the program's sake?
  - for other programmers' sake?
- Is there a way to weigh the startup costs of code reuse (installing, learning) up front?

# *(...aside)*: program or framework?

- Reinventing the wheel vs. just writing code
- How much to encapsulate and abstract...
  - for the program's sake?
  - for other programmers' sake?
- Is there a way to weigh the startup costs of code reuse (installing, learning) up front?
- Is there a way to weight the benefits of code reuse in advance?