Grail

An Extensible Interface For Browsing The Web And More

Guido van Rossum
Corporation for National Research Initiatives
November 1995

Grail Is A Mult-Faceted Program

- Complete internet browser
- Supports downloadable Python applets
- Extensible (by the user)

- Can evolve to a general application framework

- Uses proven components (Python, Tk)

- Portable (once Tk is ported to Mac and Windows)

Standard Browser Features

- Full HTML 2.0 support (well, nearly full...):
  - <FORM ...>, <ISINDEX>, <IMG [ISMAP] ...>, ...
- The usual collection of protocols:
  - HTTP, FTP, FILE, MAILTO, ...
- All the usual commands:
  - Open, Print, Save As, View Source, Search, Bookmarks, ...
- And: CSTR architecture compatible (handles Handles)

Downloadable Python Applets

- Like Java, Grail applet code:
  - is executable code downloaded from the server
  - runs in restricted mode (to avoid security breaches)
  - can display graphics as part of (HTML) page
  - is useful to enhance interactivity of WWW pages

- But:
  - Python is a higher-level programming language
  - Python is parsed and compiled (to byte-code) in the browser
  - Tk is a higher-level GUI toolkit
    - » No redraw code necessary

Extensibility

User can add extensions (dynamically loadable):

- New protocol modules
- New file types
- New HTML tags
- New GUI elements (not yet in current version)

Advantages:

- Easier to add support for new protocols etc.
- Encourages / allows researchers to experiment

Security Implementation

- Applets run in Python's "restricted execution mode"

- Standard "import" replaced
- Some standard built-in functions replaced
- Some built-in modules disabled, others crippled
- Backdoors into implementation closed

- Controlling code is all written in Python
  - security features can be customized easily

Security Ideas

- Run applets for each page in separate "cell"
- Applets need "personal space" in file system
  - Plus: file selection dialog for controlled access to other files
- Applet loading selectively enabled (e.g. per site)
  - Better: have applets "signed" by trusted authorities
    - » But: export problems for sufficiently strong crypto
- Restrict network I/O to applet's home site
  - Plus loading URLs through Grail's standard library
- Set limits on all resources used by applets
  - This includes number of URLs loaded, for instance

A General Application Framework?

- Grail can become the next Emacs, but better:
  - Not tied to Unix
  - Real GUI support
  - Built-in network I/O instead of through subprocesses
  - State-of-the-art extension language

- Needed:
  - Larger user base
  - Larger developer's base
  - Larger code base
  - Public image perceived as free and vendor-neutral