

# **wwdocshare: A middle-weight collaboration tool**

Ian Brown and Jason Rohrer

<http://www.sourceforge.net/projects/wwdocshare>

## Requirements Outline:

- Existing tool spectrum
- wwdocshare requirements
- Interfaces

# Existing tool spectrum

- Light-weight:

**Example:** emailing files back and forth

---

**Pro:** No account maintenance, no software installed, works anywhere.

---

**Con:** no collaboration management

- Heavy-weight:

**Example:** CVS and other repository-based systems

---

**Pro:** strict collaboration management

---

**Con:** account maintenance, software to install

- Middle-weight?

# wwdocshare high-level requirements

- No account creation
- Partial collaboration management
- Works across institution boundaries
- Works with multi-file projects

# wwdocshare detailed requirements

- Each author keeps local repository mirror
- No central repository
- Main operation: update local files to global latest version
- Web-based user interface using CGI
- Repository-to-repository interface through CGI
- CGI-based peer-to-peer (a first?)

# Design decisions

Why web-based?

- Leverage existing web servers
- Target users have web space

Why CGI?

- No reliable way to get file list from a web server  
⇒ Need app-to-app communication
- Use existing protocol (HTTP) for app-to-app communication

# wwdocshare interfaces

Example:

```
http://www.cse.ucsc.edu/~ian/cgi-bin/wds.pl?fetch=authorList
```

Public interfaces (script-to-script):

- `fetch=authorList`
- `fetch=fileList`
- `query=alive`

# wwdocshare interfaces

Private interfaces (user-to-script):

- `setupDirectory=<PATH>&externalURL=<URL>`
- `addAuthor=<URL>`
- `action=update`

## wds.pl interface

### wds.pl

```
+fetch=fileList()  
+fetch=authorList()  
+query=alive()  
-setupDirectory=<PATH>&externalPath=<URL>()  
-addAuthor=<URL>()  
-action=update()
```



# wds.pl interface usage

