

# VERB Web App / Framework

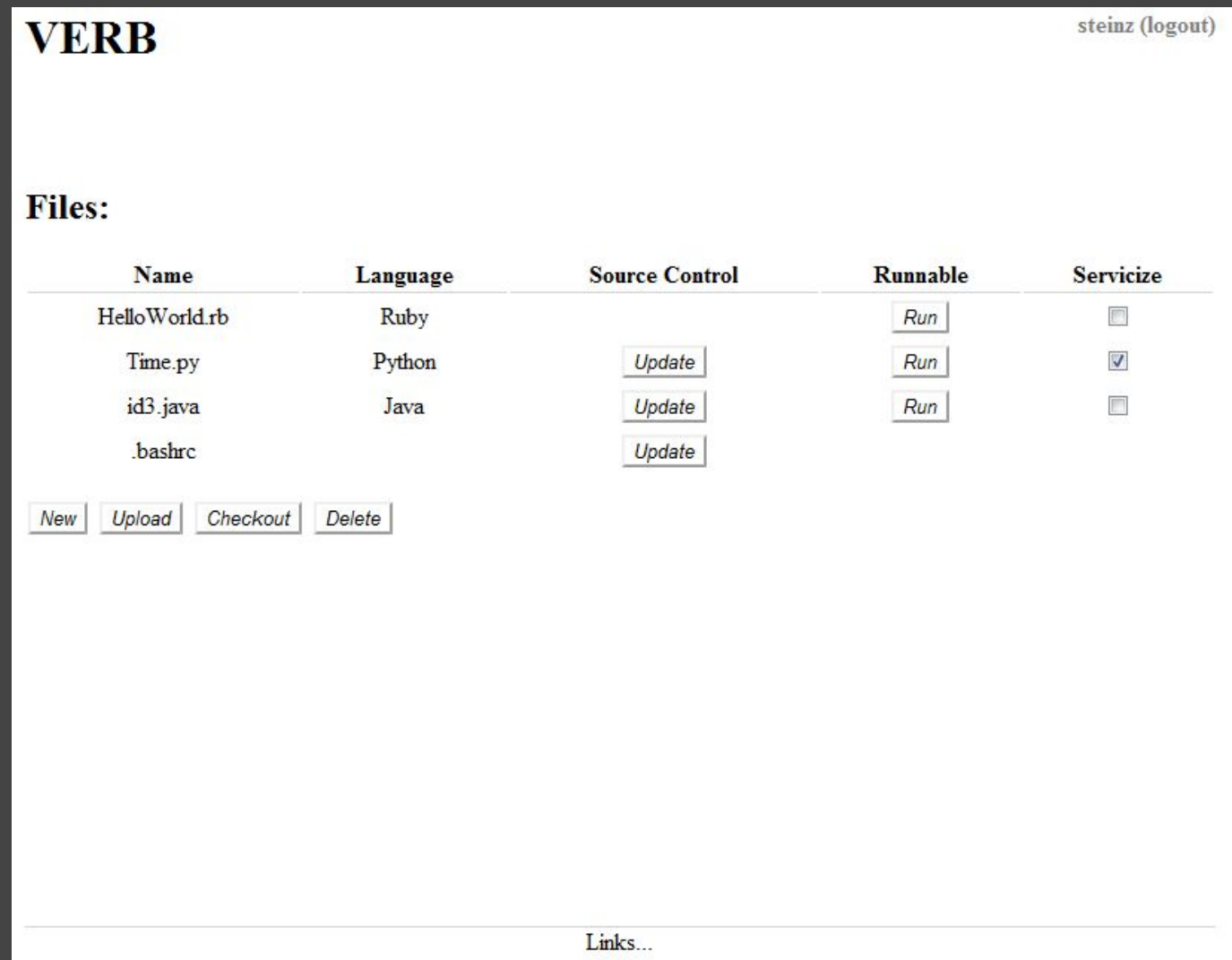
View, Edit, and Run (your code) in the Browser

An IDE  
Companion

- Remote
- Mobile
- Lightweight
- Quick
- Fully-Featured  
Text Editor

Possible Features

- SCM Integration
- Sharing
- ...



The screenshot shows the VERB web application interface. At the top left is the logo "VERB" and at the top right is the user name "steinz (logout)". Below the header is a section titled "Files:" containing a table with the following columns: Name, Language, Source Control, Runnable, and Serviceize. The table lists four files: HelloWorld.rb (Ruby), Time.py (Python), id3.java (Java), and .bashrc. Each file has a "Run" button in the Runnable column and a checkbox in the Serviceize column. The Python file has a checked checkbox. Below the table are buttons for "New", "Upload", "Checkout", and "Delete". At the bottom right, there is a "Links..." link.

Name	Language	Source Control	Runnable	Serviceize
HelloWorld.rb	Ruby		<a href="#">Run</a>	<input type="checkbox"/>
Time.py	Python	<a href="#">Update</a>	<a href="#">Run</a>	<input checked="" type="checkbox"/>
id3.java	Java	<a href="#">Update</a>	<a href="#">Run</a>	<input type="checkbox"/>
.bashrc		<a href="#">Update</a>		

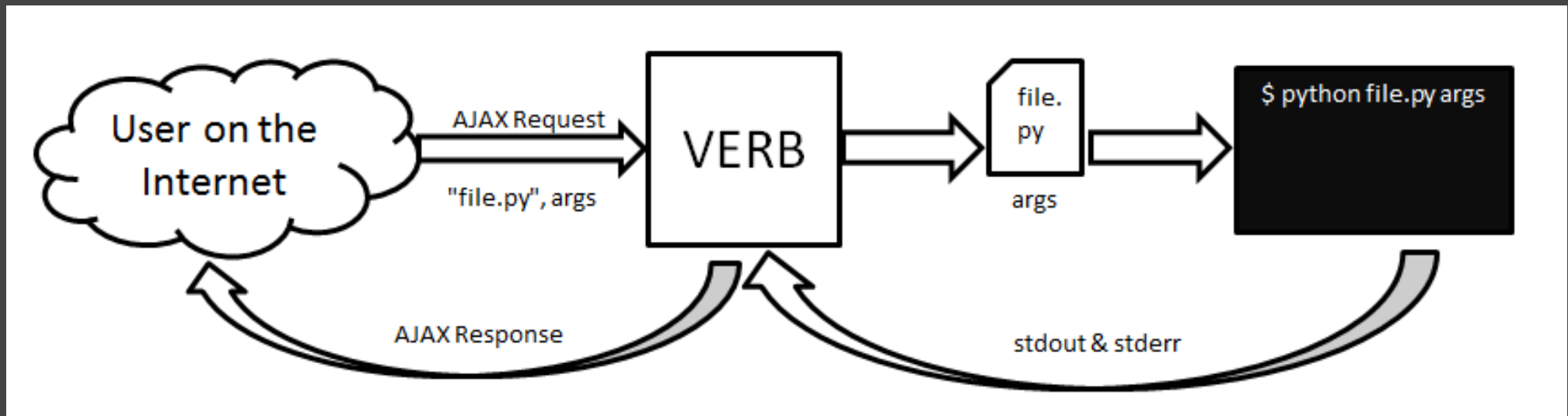
[New](#) [Upload](#) [Checkout](#) [Delete](#)

[Links...](#)

# Run -- Architecture

- Run on demand with given args from VERB webapp  
run Time.py --zone=9
- Wrap program in a service callable by URL (AJAX)  
www.steinz.verb.com/Time.py&zone=-9

Web Servicizer:



# Architecture

- LAMP / RoR / Your favorite webstack
- YUI, HAML, SASS
- Fully-featured open source (JavaScript) text editor (Ace)

## Risks / Challenges

Security

(Trust users? Sandboxing in .NET?)

Load

(CPU-intensive / buggy programs)

Long-running/always-on programs, web servers, GUI projects..?

Start with low hanging fruit

Always more advanced workflows and toolchains to integrate