# Forth - The Next Generation

Gerald Wodni

September 16, 2014

## Abstract

To attract the next generation of Forth programmers, new tools are needed. The Forth Net should serve as a single point of entry to get them started.

## 1 Introduction

The Forth Net[1] is in the process of being changed to a meta-repository which can host an optional git repository for each project, but can also link to other repository websites like GitHub[2]. The main features remain to provide a single point of entry for Forth-related projects, declaring dependencies between projects, and the ability to specify additional tags for each project to find similar ones or specify groups.

To make the Forth Net attractive for new programmers I investigated the Node.js community.

## 2 Related Work

Node.js[3], a platform for running JavaScript outside the browser environment is one of the fastest growing communities on the web. To find out what the next generation of Forth programmers want and need, I investigated the community to identify its main pillars.

NPM, Node Packaged Modules[4] is the main repository for sharing JavaScript source. It has a small and easy to learn interface based on a simple file in each project and the NPM program itself. To use NPM for a new project one adds a package.json file, which specifies the dependencies. This file also contains project meta data like name and author, making the project itself a valid NPM package.

GitHub has no fancy website for each project, but just displays a README file different formats, most prominent ones are MarkDown or plain text. This makes the user interface required to setup a project description website even smaller.

## 3 Flink

Copying these features is not doing justice to Forth, I wanted to emphasize Forth's unique features like the interactive compiler interface. An emulated Forth System inside the browser is not of much use for serious projects, so the system is laid-out as follows:

**Server** A web server capable of handling WebSockets[5] used as a broker between the other parties.

**Flink** An interactive browser IDE, build as responsive website running on every major browser which supports HTML5 and WebSockets.

**Uplink** A tiny implementation of the WebSocket interface, which is only necessary until the target Forth system understands the Flink-WebSocket protocol. As the protocol is a work in progress, please consult the repository for the latest command set[6].

Flink consists of an interactive console[7] and an editor[8] which can load and save source code to the project's repository. Once the programmer is logged in, and has a target system attached via uplink, the console behaves like a line-buffered Forth. To compile the code from the editor window, it is transfered a line at a time waiting for the Forth's "ok" or an error messages. If the Uplink is connected directly to a system with no Internet access (i.e. over a serial line), Flink enables this device to a rich IDE and allows inclusion of other files and even projects.

## 4 Further Steps

**Package Format** A simple format for the Forth Net which provides similar functionality like NPM's package.json . An alternative would be to parse the forth source code for "finclude ...".

**User Interface** A HTML5 user interface which simplifies API to the Forth System drastically by having a full-blown GUI on the front end, and a simple text interface to Forth.

**M2M Communication** As Flink is based on the Websocket Protocol, it also works behind most firewalls and allows for remote machine maintenance as well as indirect machine to machine communication

# References

[1] The Forth Net. URL http://theforth.net.

[2] GitHub. URL https://github.com.

[3] node.js. URL http://nodejs.org.

[4] NPM. URL https://www.npmjs.org.

[5] I. Fette and A. Melnikov. RFC6455 The Web-Socket Protocol.

[6] Uplink. URL https://github.com/GeraldWodni/uplink.

[7] jq-console. URL https://github.com/replit/jq-console.

[8] Ace cloud 9 editor. URL http://ace.c9.io/.