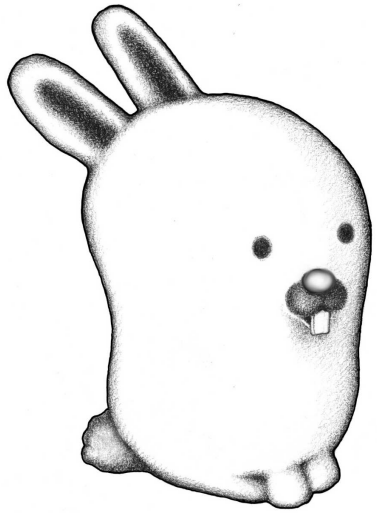
Connection between Go and Plan 9

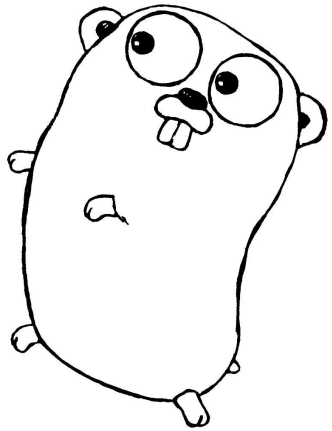
Jonas Amoson
jonas.amoson@home.se

Vilnius Golang Meetup 25 January 2017

Uber HQ Vilnius, Lithuania

Look how similar their respective mascots are!





Glenda & Gopher

Outline

- **Intro to Plan 9**
- **Connections between Plan 9 and Go**
- **The port of Go to Plan 9**
- **Installing Go on Plan9 on Rasberry Pi**

Plan 9 Overview

- General-purpose operating system
- Developed as a successor to Unix
  - By Ken Thompson, Rob Pike, Russ Cox, et al.
- Research oriented – for programmers
- Open Source since 2002

Resources as files or file trees

- **Like Unix: data files and special files**
  - **Easy shell scripting (only 30 syscalls needed)**
- **Network is `/net`**
- **Directories of special files**
  - **Kernel or user space file servers**

Private name-spaces

- **Unix: processes share most file system**
  - **Special files in `/proc` and `/dev` are private**
- **Plan 9: each process has its own file tree**
  - **Similar to running in a chroot environment**
  - **Binds directories and file servers in the file tree**
  - **Sub-processes inherits file tree of parent**

Distributed operating system

- **All files accessed through network file system**
- **Not important if resource is local or remote**
- **Sharing by message passing**
- **Ex: running *rio(1)* remotely**
  - **Import `/dev/cons /dev/mouse /dev/draw`**
  - **No need for special network protocol**

Union mounts

- **A directory can be a view of multiple directories**
- **Example: `/bin` contains files from**
  - **`/386/bin`**
  - **`/rc/bin`**
  - **`$home/bin/rc`**
  - **`$home/bin/386`**
- **`bind -a /another/bin /bin`**

Graphical User Interface

- **Plan 9 is very textual**
- **Copy and paste text with the mouse**
- **Not many applications with their own GUI:s**

But...

...it is also very GUI:ish

- **Almost no keyboard shortcuts**
- **An app can use the window it is started in**
- **Possible to edit text in window and resend**
- **Mouse with three buttons recommended**

Plan 9 and Go

- **Rob Pike, Ken Thompson, Russ Cox**
- **Tool chain carry over (Ken C and Assembler)**
- **Alef & thread library**
  - **“Goroutines”**
  - **Channels**
  - **Share by communication**
- **Easy cross compilation**
- **Everything is UTF-8**

Ken C

- **Not 100% ANSI compliant**
- **Sane optimisations (compare gcc)**
- **Include files never include other include files**
- **Static instead of dynamic linking**
- **Unicode symbols (Runes)**
- **Plan 9's standard library influenced Go's**
- **Very similar assembly output as Go**

Coding Style

- **Short variable names**
- **Tabs for indention, instead of spaces**
- **Avoid complexity**

What Go brings to Plan 9

- **A modern safe systems programming language**
  - **There is (essentially) no C++ / Java**
- **Rich standard library**
- **Even greater flora of third party libraries**
- **Some "advertisement"**

Installing Plan 9

- **PC / Virtual machine (Qemu recommended)**

  http://plan9.bell-labs.com/plan9/download/plan9.iso.bz2

- **Raspberry Pi**

  http://plan9.bell-labs.com/sources/contrib/miller/9pi.img.gz

- **Other distributions**

  http://9atom.org

  http://9front.org

The port of Go to Plan 9

- **The first attempts at compiling est. around 2010**
- **Implementation of** *nsec(1)* **2014**
- **Architectures:** `386`, `amd64`, `arm`

Options for Go on Plan 9

- **Install Go 1.4 on `plan9/386` and bootstrap newer go-version of the compiler.**
- **Crosscompile from other supported OS, e.g. Linux**
- **Download binaries (not officially provided)**

Crosscompiling from Linux

- On Linux
  - Cross compile Go 1.7 (or newer)
  - Transfer `go-plan9-xxx-bootstrap.tbz`
- On Plan 9
  - Unpack the bootstrap archive
  - Set `GOROOT` and `GOPATH` environment variables

Crosscompiling from linux/amd64 to plan9/arm

Go 1.7.1 is installed in `/usr/local/go`

```
$ cd /usr/local/go/src
$ export GOROOT_BOOTSTRAP=/usr/local/go
$ GOOS=plan9 GOARCH=arm ./bootstrap.bash
```

⇒ `/usr/local/go-plan9-arm-bootstrap.tbz`

The file is moved to an ftp directory

Transfer the compiler to Plan 9

```
% ftpfs 192.168.1.109
220 (vsFTPd 3.0.2)
!Adding key: proto=pass
server=192.168.1.109 service=ftp
user[glenda]: jonas
password:
!
331 Please specify the password.

230 Login successful.
215 UNIX Type: L8
257 "/home/jonas"
% cp /n/ftp/go-plan9-arm-bootstrap.tbz .
% tar xf go-plan9-arm-bootstrap.tbz
% mv go-plan9-arm-bootstrap go-arm
```

Setting the enviroment variables

Adding the following lines to `$home/lib/profile`

```
GOPATH=$home/go
GOROOT=$home/go-arm
bind -a $home/go-arm/bin /bin
```

So does it work?