What's coming in
LuaRocks 3

Hisham Muhammad
h@hisham.hm

Lua Workshop 2016
San Francisco, CA

What is LuaRocks

Package manager for Lua modules

- written in Lua (.lua files)
- or binary modules (.so/.dll files)

Compiles, installs, downloads, uploads

- specification format (.rockspec)
- distribution format (.rock)
- optional built-in build tool

What does it not do

Install and manage non-Lua dependencies

- e.g. It won't install OpenSSL or MySQL for you

Install and manage your versions of Lua itself

(The Windows package does include Lua though)

- Try a Lua version manager
  - Hererocks (in Python: Unix+Windows)
  - luaver (in Bash: Unix)

What happened since #luawshop14

Slow and steady development

- LuaRocks 2.2.1, 2.2.2, 2.3.0, 2.4.0, 2.4.1

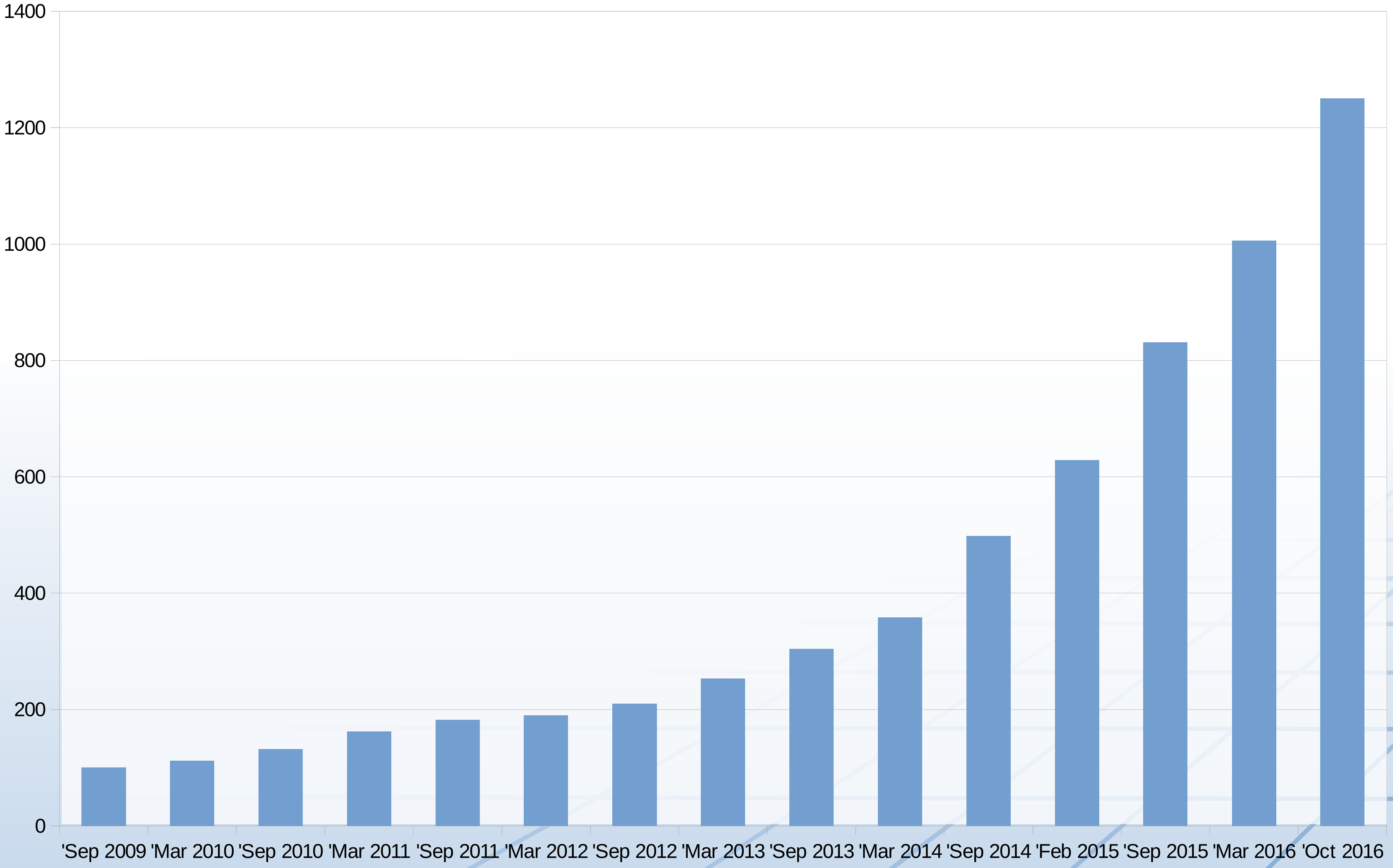
No big features

- Reorganization of the Windows setup
- hg://, git+ssh://
- Fixes under the hood, cleanups

New test suite

- Google Summer of Code project

...luarocks.org happened!



Slow development

Slow development

Because it works?

Slow development

Because it works?

Lack of time/funding

Slow development

Because it works?

Lack of time/funding

Constrained by compatibility

Compatibility: rockspec format

- Frozen since 1.0 (2008)
- Limitations of the format are well-known
  - `builtin` build mode compiles only C89, can't pass custom compiler flags
  - Can't use platform-specific detection for dependencies (pkg-config, etc.)
  - No separation between build/runtime dependencies
  - etc.

Our challenge:
making rockspecs more powerful

One approach: add-ons system

Make the rockspec format extensible by users

- Rockspecs could depend on add-ons
- Add-ons would add new fields, hooks, etc.

This requires getting the extension points right

- API design is hard, extension API design moreso

One approach: add-ons system

Concerns:

1. How to avoid breaking people's add-ons
2. Prevent add-ons from peeking into internals

This avenue was explored in GSoC 2015

- Ended up not moving forward
- Not a bad idea per se, though

Another approach: just unfreeze the rockspec format

```
rockspec_format = "3.0"
```

Another approach: just unfreeze the rockspec format

Rockspec format version check is already there

- It assumes "1.0" by default

Rockspec type-checker checks each entry against the minimum version it is supported

- i.e. we can make field "foo" valid only for rockspecs marked version "3.1" and up
- can't use a new feature and forget to mark the rockspec format accordingly

Another approach:
just unfreeze the rockspec format

Simpler, bolder approach

Concerns:

1. Don't break existing rockspecs
2. Feature bloat
3. Rockspecs may require upgrading LuaRocks

Criteria for additions to the format

Backward compatibility

- old rockspecs must keep working
- Forward compatibility is hard because of strict type-checking in LuaRocks

Goals:

- Add functionality needed by real modules
- Make it easier/shorter to write rockspecs
- Proper solutions for things that currently demand workarounds

Upgrading LuaRocks

Distros are slow to upgrade LuaRocks

`luarocks install luarocks` must work!

Difficulties:

- LuaRocks build configuration is very flexible (many moving parts)
- It still doesn't work on Windows

If it works 100% well, can we auto-upgrade LuaRocks on demand?

Possible additions to the format

- **`builtin`** by default
  - src/*.lua or *.lua when **`build.modules`** is absent
  - propagate **`external_dependencies`** automatically
  - add a **`language`** field (C11, C++)
- **`build_dependencies`** (for testing tools, etc.)
- virtual dependencies (**`provides`**)
- optional dependencies
- allow platform overrides in more places
- installing arbitrary assets (a design challenge)

In closing

LuaRocks 3.0 will bump the rockspec format

Not all changes will come at once

- We need to set the stage to make changes painless

You're welcome to propose features in the `luarocks-3` branch at GitHub

New test suite gives more confidence for future changes

Thank you!

https://luarocks.org
github.com/keplerproject/luarocks
gitter.im/keplerproject/luarocks
@luarocksorg

http://hisham.hm
h@hisham.hm
@hisham_hm

About these slides

- Feel free to share this presentation
  and to use parts of it in your own material
- Licensed under the Creative Commons CC BY 4.0:
  - https://creativecommons.org/licenses/by/4.0/