

Nirbheek Chauhan

(nirbheek on Twitter/IRC)

GStreamer contributor since 2014

<http://nirbheek.in>
nirbheek@centricular.com
nirbheek.chauhan@gmail.com



~~Nirbheek Chauhan
(nirbhee on Twitter/H/C)
GStreamer contributor since 2014~~

<http://nirbheek.in>
nirbheek@centricular.com
nirbheek.chauhan@gmail.com



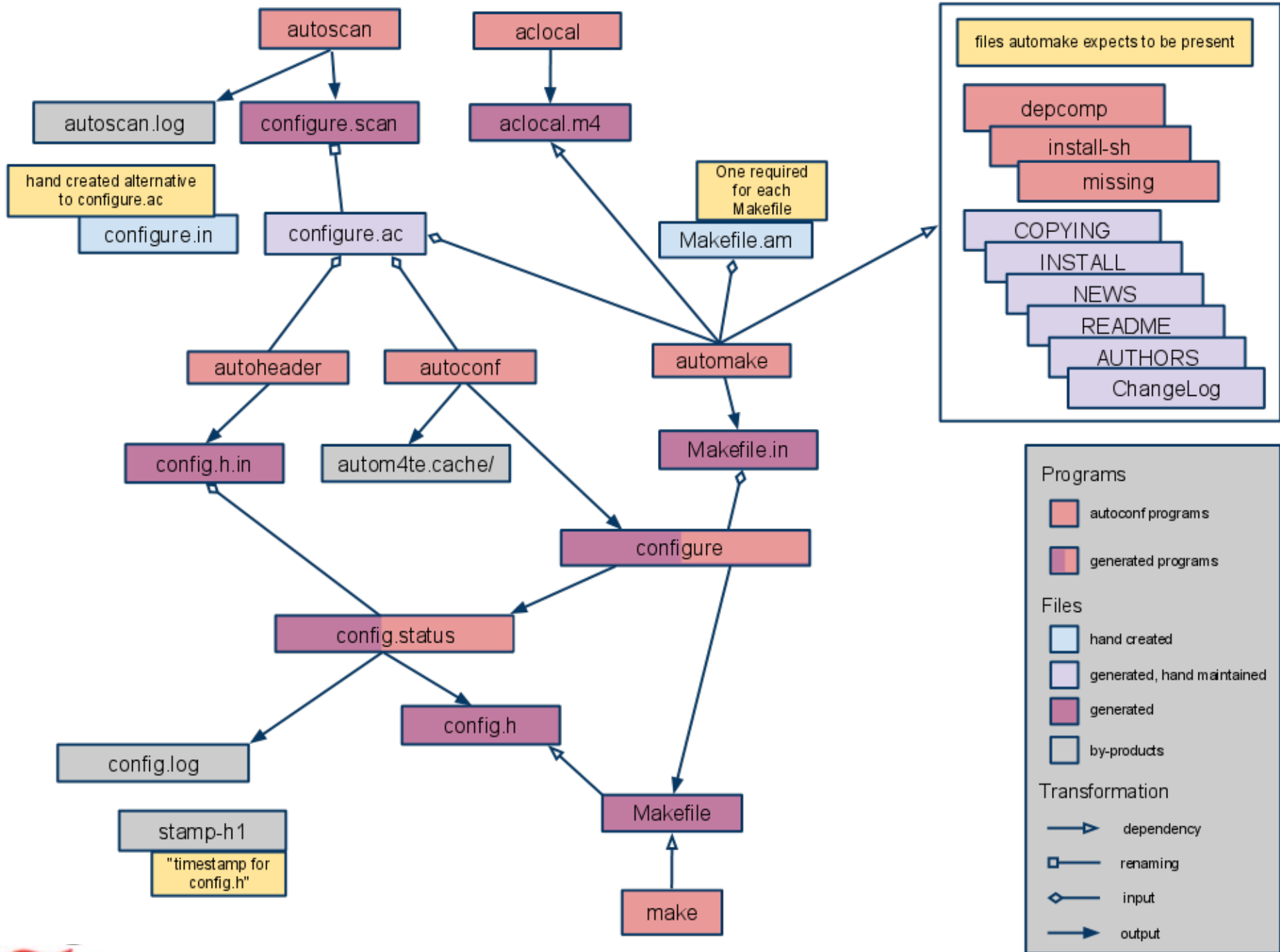
GStreamer development, Windows
Faster builds, all platforms
Meson build system



40 years of build systems

GNU Make, GNU Autotools, Waf, SCons, CMake, Gyp

... the list goes on



(PS: That graph is incomplete)

libtool, config.lt, config.guess, compile

sed, tr, awk, grep, install, coreutils

Why do we still use Autotools?

Why do we still use Autotools?

```
$ du -sh gstreamer.git/configure  
956K gstreamer.git/configure
```




“
640K ought to
be enough for
anybody.”

-- Bill Gates, founder of Microsoft, in 1981

Why have we been using Autotools?

(Not rhetorical)

Why have we been using Autotools?

It works.

Has served us well so far

Why have we been using Autotools?

Works out of the box on any Linux distro

Cross-compilation

Why have we been using Autotools?

Familiar system

gtk-doc, docbook, glib-mkenums,
gettext, gobject-introspection

Why have we been using Autotools?

Conventions

pkg-config, library conventions, etc.

We need something better.



Why do we need something better?

Technical debt

Macros that spit out MBs of shell
Cargo Cult Programming

Why do we need something better?

Ridiculously slow

Remember MBs of shell?

10-20x slower on Windows & embedded systems



Why do we need something better?

“Cross-platform” support

... for the platforms of the 1970s and 1980s

```
$ grep -A 1 -e 197 configure
# Use only awk features that work with 7th
edition Unix awk (1978).
# My, what an old awk you have, Mr. Solaris!
```

Why do we need something better?

“Cross-platform” support

Terrible experience on OS X and Windows



Why do we need something better?

Terrible experience on OS X and Windows

GNU vs BSD MinGW vs Cygwin

Why do we need something better?

Terrible experience on OS X and Windows

Can't build with native IDEs

Why do we need something better?

Crappy experience on Windows

“MinGW32 or MinGW64?” “MSYS or MSYS2?”
“m4.exe crashes for me” “make.exe hangs”

Why do we need something better?

Crappy experience on Windows

“Why does building all of GStreamer take a full day?”



Why do we need something better?

Unusable experience on Windows

“We use C++ exceptions”
SJLJ vs DW2EH vs SEH

Why do we need something better?

Unusable experience on Windows

“I want debug symbols in Visual Studio”

DWARF vs PDB

Why do we need something better?

Unusable experience on Windows

“My application uses a symbol server”

Why do we need something better?

Unusable experience on Windows

“I want to build everything with MSVC”

Why do we need something better?

Unusable experience on Windows

“I've always done development inside Visual Studio”



Why do we need something better?

Unusable experience on Windows

“Can I use GStreamer on Windows Phone?”



Why do we need something better?

OS X elements lag behind

Windows elements use APIs added in XP

Why do we need something better?

The harder it is to contribute

The fewer contributors you will have

We need something better.



Why is there no alternative?

Why is there no alternative?

Technical debt, Inertia

Autotools was “good enough”

Why is there no alternative?

Apathy, Inertia

Linux users live in a filter bubble

Why is there no alternative?

Not sexy

Difficult problem, lots of work

Well, now we have an alternative.

Meson

<http://mesonbuild.com>

Meson

Generates Ninja build files

2 years of development, 40 contributors

```
$ cd gstreamer.git/  
$ mkdir build-dir && cd build-dir  
$ meson ..  
$ ninja
```


Meson

Always builds out-of-tree

Auto-detects source code changes and re-runs itself

Meson

Why do **we** use it?

GStreamer ❤️ Meson



GStreamer ❤️ Meson

Finally a build system that does what we need



GStreamer ❤️ Meson

pkg-config for everything



GStreamer ❤️ Meson

```
gnome.mkenums()  
gnome.generate_gir()  
gnome.gtk_doc(), ...
```

GStreamer ❤️ Meson

Useful cross-platform support



GStreamer ♥ Meson

Fewer dependencies

Don't need sh, m4, grep, awk, tr, perl, coreutils,
autoconf, automake, kitchen sink for building

GStreamer ❤️ Meson

Fewer dependencies

Python 3, Meson, Ninja
+ whatever actually builds stuff



Meson ❤️ Platforms

Linux, Windows, OS X, BSD, iOS, Android

Integrated Cross-compilation

Meson ❤️ Toolchains

GCC, Clang, MinGW, Cygwin, MSVC

Ninja, MSBuild, Visual Studio, XCode

GStreamer ❤️ Meson

Everything builds faster!



Meson ❤️ Speed

Made to be fast on first principles

Written in Python, so could be even faster

Meson ❤️ Speed

Meson configuration is fast

Ninja is *faster*

GStreamer ❤️ Meson

We ported GStreamer to Meson

Can't get any closer to 'enjoyment'
(when talking about build systems)



GStreamer ❤️ Meson

libffi, glib, orc, gstreamer
gst-plugins-base, gst-plugins-good
gst-plugins-bad, gst-plugins-ugly, gst-libav
gst-python, gst-devtools, gst-editing-services



GStreamer ❤️ Meson

Everything builds faster!

No forking, no shell, no recursion



Meson ♥ Speed

Autotools vs Meson — GCC on Linux 64-bit, i5×4

gstreamer.git: 32s + 92s = 124s (~2 min)
3s + 67s = 70s → **44% faster**

Meson ♥ Speed

Autotools vs Meson — GCC on Windows 64-bit, i5×4

gststreamer.git: 165s + 475s = 640s (>10 min)
5s + 55s = 60s → **1066% faster**

GStreamer ♥ Meson

Suddenly...

Building on Windows doesn't take a day!



GStreamer ❤️ Meson

Suddenly...

Building on embedded devices doesn't take 2 days!

GStreamer ❤️ Meson

New things are possible

Generate Visual Studio project files!

Visual Studio development



GStreamer ❤️ Meson

New things are possible

Debug with Visual Studio!

Full support for PDB debug symbols



GStreamer ❤️ Meson

Alternative/parallel build system for gstreamer

Available in GStreamer 1.10

More work to do on the port itself



GStreamer ❤️ Meson

Some projects have already moved to it

PiTiVi, gst-transcoder



GStreamer ❤️ Meson

Tim and Nirbheek

Linux support ✓

Windows support ✓



GStreamer ❤️ Meson

Tim, Nirbheek, and **you!**

macOS support WIP

iOS support WIP

Android support WIP



The biggest reason we use Meson

Meson ♥ GStreamer

Listens, understands, adapts

Bugs have always been welcome!



Meson

Why do **we** use it? ✓

Why should **you** use it?

Meson ❤️ Usability

Readable build file syntax

Python-inspired

```
project('myapp', 'c', version : '1.0')
```

```
mylib = library('mylib', 'the-lib.c',  
install : true,  
soversion : 1)
```

```
executable('myapp', 'the-app.c',  
install : true,  
link_with : mylib,  
c_args : ['-DVER=' + meson.project_version()])
```


...

```
cdata = configuration_data()  
cc = meson.get_compiler('c')
```

```
if cc.has_function('poll')  
    cdata.set('HAVE_POLL', 1)  
endif
```

```
cdata.set('HAVE_TIME_H', cc.has_header('time.h'))
```

```
configure_file(output : 'config.h',  
    configuration_data : cdata)
```

...

```
meson --cross-info=info.txt
```

```
[host_machine]
```

```
system = 'windows'
```

```
cpu_family = 'x86'
```

```
cpu = 'x86'
```

```
endian = 'little'
```

```
[binaries]
```

```
c = 'i686-w64-mingw32-gcc'
```

```
cpp = 'i686-w64-mingw32-cpp'
```

```
ar = 'i686-w64-mingw32-ar'
```

```
strip = 'i686-w64-mingw32-strip'
```

```
pkgconfig = 'pkg-config'
```

```
exe_wrapper = 'wine'
```

```
[properties]
```

```
c_link_args = ['-DWINVER=0x0501', '-D_WIN32_WINNT=0x0501']
```

```
has_function_stpcpy = false
```



Meson ❤ Usability

From configuration to building in one step

In Autotools terms, automake + autoconf in one go

Meson ❤️ Usability

Not Turing-complete

No macros, no function definitions, no infinite loops

Meson ♥ Usability

Explicit, Forgiving, Extensible

99% of things work OOB, 1% are easy to fit

Meson ❤️ Usability

Explicit, Forgiving, **Extensible**

Not a static platform!

Meson

We think it's the future

Try it out!



Thanks for listening!



Questions?