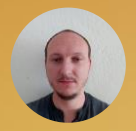
Microsoft

Linaro Connect
MADRID 2024 | MAY 14-17 2024

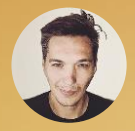
A Journey to Git for Windows on AArch64

GNU Toolchain
MinGW, Cygwin and MSYS2
GitHub CI
Community Contributions

Engineering group



Radek Barton
@Blackhex



Evgeny Karpov
@eukarpov

GitHub

MinGWW64

The Classic Way to Port an Application to Another Platform or Architecture

External components or libraries that a software application needs to function properly.

Architecture-specific code, that is written for a specific hardware architecture and may not run correctly on others.

Utilities that assist in the creation, debugging, maintenance.

A Simple Way to Port a GNU App to Windows (MinGW) if Possible

MinGW (Minimalist GNU for Windows) – tools to create Windows application.

Code Adaptation, modifying an application's source code so it can run in a different environment.

Handling dependencies, porting the external libraries or components.

Cross-compilation, compiling code on one platform (the host) to run on a different platform (the target).

Compilation on Windows

Cross-compilation on Linux for Windows

POSIX Support: Cygwin, MSYS2 Packages (bash)

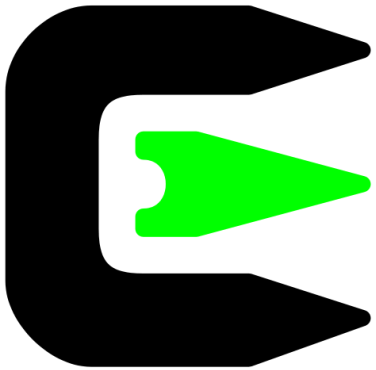
POSIX (Portable Operating System Interface)

Large collection of GNU tools on Windows.

Unix-like environment.

Easier to port applications that rely on POSIX APIs.

POSIX Experience



Git for Windows Dependencies

MSYS2, Cygwin, and MinGW are not present on Windows Arm64.

The tools do not provide utilities for building the AArch64 MinGW target.

GCC requires a new target to be implemented in order to build all packages for the new architecture.

Adding a New aarch64-w64-mingw32 Target to the GNU Toolchain

Introduce new aarch64-w64-mingw32 target.

Binutils - A collection of binary tools, the GNU linker and assembler, for creating executable programs.

GCC - The GNU Compiler Collection that supports various programming languages, including C, C++.

binutils

MinGWW64

aarch64-w64-mingw32 Target

aarch64 - ARMv8 architecture
w64 - Windows 64-bit
mingw32 – MinGW, which originally has been developed for 32-bit and later extended for 64-bit.

Binutils

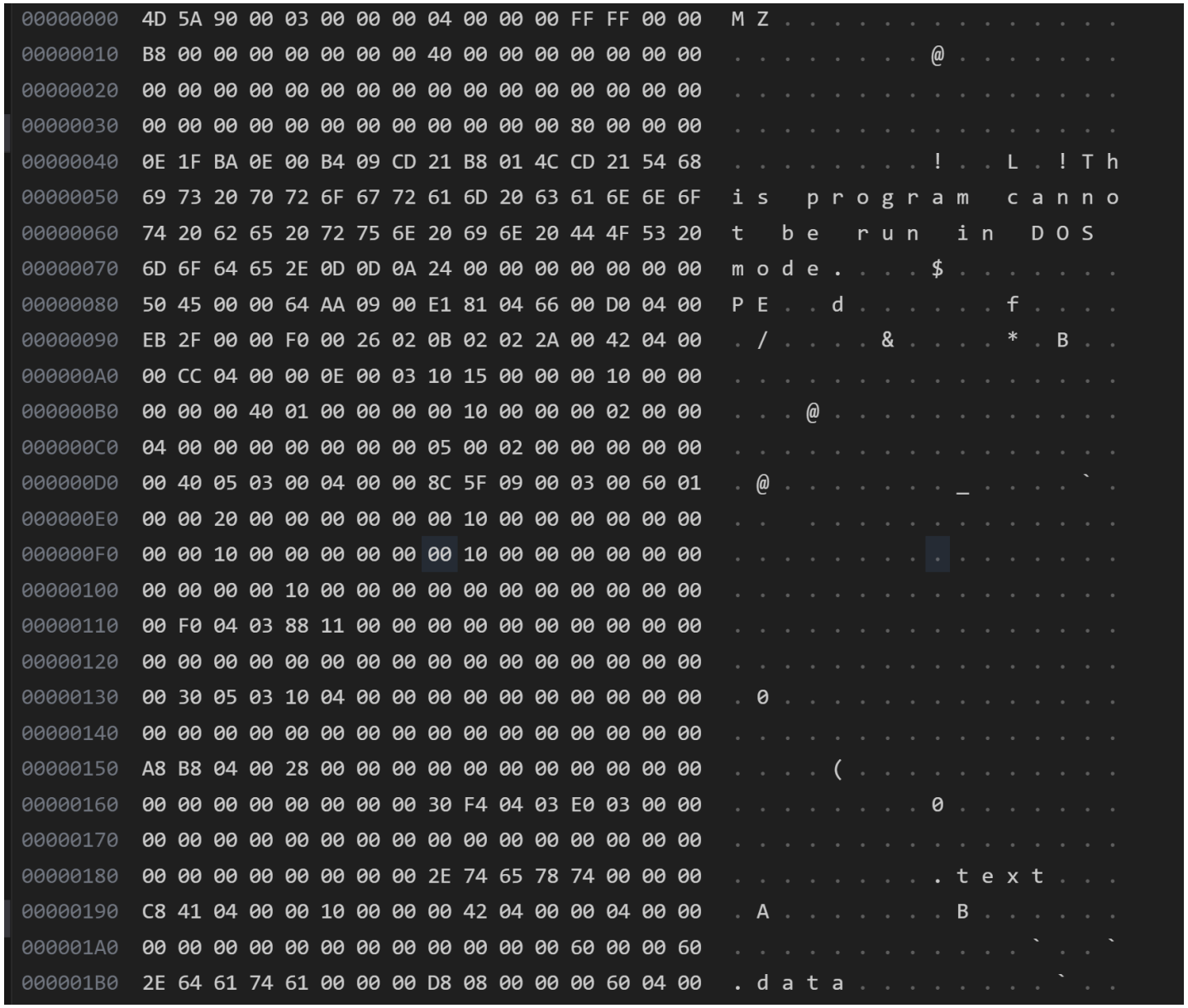
Relocations for AArch64.

Reverse Engineering of the COFF Format.

Resolving linking issues.

Community Support.

The First Contribution to the GNU Toolchain.



GCC Prototyping, Proof of Concept for C and C++.

DLL import/export support.

SEH – Structure Exception Handling implementation in binutils and GCC.

OpenBLAS, **OpenSSL**, **FFmpeg** and **libjpeg-turbo** 4 packages are supported and fully tested.

Time to start contributing upstream.

Reducing the Scope of the First Series to the C Language.

GitHub Actions as an Effective Way to Build Powerful CI

CI has proved very useful for the porting.

Working mostly on default GitHub runners.

Self-hosted Runners in Azure to Validate Arm64 Tests.

Building 6 targets.

Testing 4 packages and executing internal test.

Daily rebasing.

Validate Patch Series.

Collaboration with Linaro and Review of Patch Series.

Linaro extensive knowledge and experience in contributing for GNU Toolchain, 10+ years. Strong credibility in GNU Toolchain community.

Helpful recommendation for organizing patch series and valuable reviews.

Assisting with patch series testing.

3 internal patch series reviews.

6 patch series reviews with Linaro. One review takes usually 1w.

12 reviews in total before the patch series has been approved for merging.

The new target will be supported, tested, added to CI, and maintained by Linaro.

- [PATCH v1 00/13] Add aarch64-w64-mingw32 target *Evgeny Karpov*
  - [PATCH v1 01/13] Introduce aarch64-w64-mingw32 target *Evgeny Karpov*
    - [PATCH v1 01/13] Introduce aarch64-w64-mingw32 target *Andrew Pinski*
      - [PATCH v1 01/13] Introduce aarch64-w64-mingw32 target *Evgeny Karpov*
      - [PATCH v1 01/13] Introduce aarch64-w64-mingw32 target *Fangrui Song*
    - [PATCH v1 01/13] Introduce aarch64-w64-mingw32 target *Richard Earnshaw (lists)*
  - [PATCH v1 00/13] Add aarch64-w64-mingw32 target *Maxim Kuvyrkov*
    - [PATCH v1 00/13] Add aarch64-w64-mingw32 target *Evgeny Karpov*
  - [PATCH v1 02/13] aarch64: The aarch64-w64-mingw32 target implements *Evgeny Karpov*
    - [PATCH v1 02/13] aarch64: The aarch64-w64-mingw32 target implements *Richard Earnshaw (lists)*
      - [PATCH v1 02/13] aarch64: The aarch64-w64-mingw32 target implements *Evgeny Karpov*
      - [PATCH v1 02/13] aarch64: The aarch64-w64-mingw32 target implements *Richard Sandiford*
      - [PATCH v1 02/13] aarch64: The aarch64-w64-mingw32 target implements *Andrew Pinski*
      - [PATCH v1 02/13] aarch64: The aarch64-w64-mingw32 target implements *Mark Harmstone*
      - [PATCH v1 02/13] aarch64: The aarch64-w64-mingw32 target implements *Martin Storsjö*
      - [PATCH v1 02/13] aarch64: The aarch64-w64-mingw32 target implements *Evgeny Karpov*
      - [PATCH v1 02/13] aarch64: The aarch64-w64-mingw32 target implements *Evgeny Karpov*
  - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Evgeny Karpov*
    - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Richard Earnshaw (lists)*
      - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Andrew Pinski*
      - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Iain Sandoe*
      - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Jacek Caban*
      - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Evgeny Karpov*
      - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Richard Sandiford*
      - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Evgeny Karpov*
      - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Evgeny Karpov*
    - [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI *Richard Earnshaw (lists)*
      - [EXTERNAL] Re: [PATCH v1 03/13] aarch64: Mark x18 register as a fixed register for MS ABI
  - [PATCH v1 04/13] aarch64: Add aarch64-w64-mingw32 COFF *Evgeny Karpov*
    - [PATCH v1 04/13] aarch64: Add aarch64-w64-mingw32 COFF *Richard Sandiford*
      - [PATCH v1 04/13] aarch64: Add aarch64-w64-mingw32 COFF *Evgeny Karpov*
  - [PATCH v1 05/13] Reuse MinGW from i386 for AArch64 *Evgeny Karpov*
    - [PATCH v1 05/13] Reuse MinGW from i386 for AArch64 *Andrew Pinski*
      - [PATCH v1 05/13] Reuse MinGW from i386 for AArch64 *Evgeny Karpov*

Timeline Estimates for the Contribution Upstream

May 2024: Introduce new aarch64-w64-mingw32 target. Build a hello-world application.

May 2024: Add DLL import/export support. Build for OpenSSL, OpenBLAS, FFmpeg and libjpeg-turbo. Fully tested.

June 2024: Enabling debugging information, call stack support in debugger and optimization fixes in GCC.

Sep – Dec 2024: Resolving issues with unit testing for the new architecture.

2024: C++ support without SEH.

Potential contribution
2024: SEH implementation for C++.

Why is Port of Cygwin Needed?

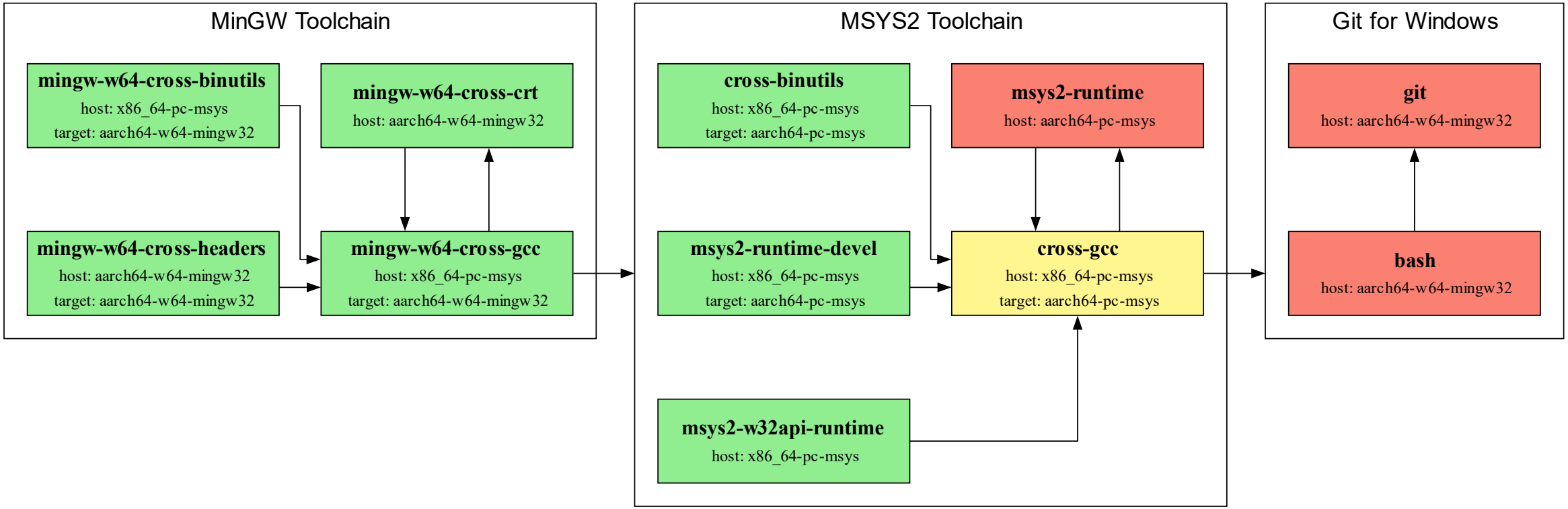
Git for Windows is a MSYS2 distribution with more than 230 packages.

MSYS2 contains both **MinGW** and **Cygwin** packages.

MSYS2 runtime (msys-2.0.dll) is friendly fork of **Cygwin** (cygwin1.dll).

MinGW packages are already ported using LLVM (CLANGARM64 environment).

Enabled MSYS2 Packages



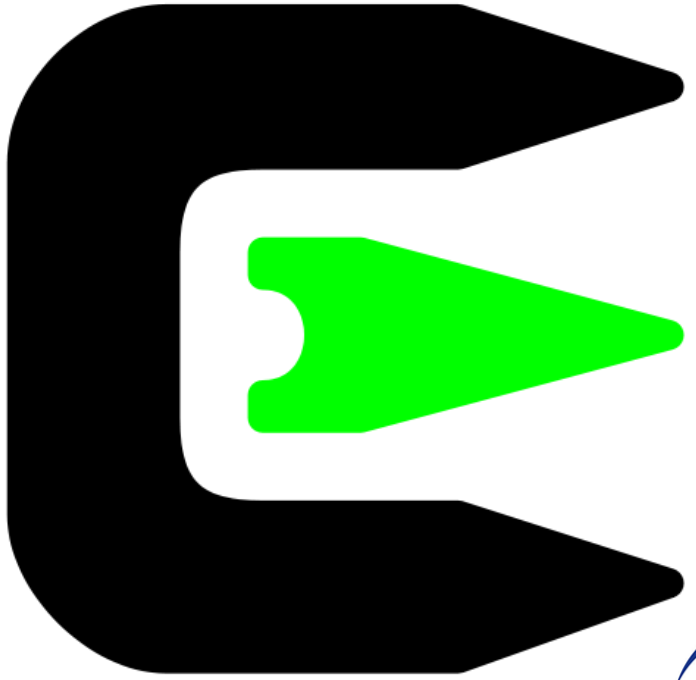
AArch64 Cygwin/MSYS2 Port

Yet another targets need to be added to GCC and binutils:

- aarch64-pc-cygwin
- aarch64-pc-msys

Challenges with porting certain features properly:

- Fork vs. Windows memory model.
- Deprecated APIs not supported on Windows Arm64.
- POSIX threads, signals and traps, and stack manipulation.
- cygwin1.dll is a hybrid DLL, functions autoloading.



How to Try Our MinGW Toolchain in MSYS2?

- Add to `/etc/pacman.conf`:

```
[woarm64]
Server = https://windows-on-arm-experiments.github.io/msys2-woarm64-build/$arch
SigLevel = Optional
```

- Then update the database and install the toolchain:

```
pacman -Sy
pacman -S mingw-w64-cross-gcc
```

- Enjoy and report issues to https://github.com/Windows-on-ARM-Experiments/msys2-woarm64-build/issues

Q & A

An active experimental repo with C++ support for AArch64 on Windows.

https://github.com/Windows-on-ARM-Experiments/mingw-woarm64-build

https://github.com/Windows-on-ARM-Experiments/msys2-woarm64-build

Please report issues on that repo if you found something.

Thank you