



SUPPORTED BY RED HAT

LOG IN

SIGN UP

478

90

34

## Main menu

[Articles](#) [Resources](#) [About](#) [Community](#)[The Open Org](#)

# 3 open source projects for modern COBOL development

13 Oct 2015 | [Joshua Allen Holm \(/users/holmja\)](/users/holmja) | 196 | [13 comments](#)

*Image by : opensource.com*

An iconic figure of the early history of computing, [Grace Hopper](https://en.wikipedia.org/wiki/Grace_Hopper) ([https://en.wikipedia.org/wiki/Grace\\_Hopper](https://en.wikipedia.org/wiki/Grace_Hopper)) is the grandmother of the [COBOL](https://en.wikipedia.org/wiki/COBOL) programming language (<https://en.wikipedia.org/wiki/COBOL>). Of her many claims to fame, she invented the first compiler and helped spread the adoption of machine-independent programming languages. Today her legacy lives on in many ways,

including the [Grace Hopper Celebration of Women in Computing](http://gracehopper.anitaborg.org/) (<http://gracehopper.anitaborg.org/>).

While Hopper contributions to computing are remembered, celebrated, and built upon by her successors, COBOL itself is often dismissed as a relic of earlier era of computing. To a certain extent, that is true. Most of the COBOL being written today is for maintaining legacy code, not starting new projects. However, the language is still being updated with [COBOL 2014](http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=51416) ([http://www.iso.org/iso/iso\\_catalogue/catalogue\\_tc/catalogue\\_detail.htm?csnumber=51416](http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=51416)) being the most recent standard for the language, and there are still plenty of opportunities to [apply for jobs that require COBOL experience](http://www.indeed.com/q-cobol-jobs.html) (<http://www.indeed.com/q-cobol-jobs.html>).

Thankfully, using COBOL in modern times does not require tracking down legacy hardware. There are several tools available that make using COBOL on modern computers as straightforward as using other programming languages. Below, I take a look at three open source projects that help you code, compile, and use COBOL on a Mac, Linux, or Windows computer. So grab a book on COBOL programming, (if you need one, [Beginning COBOL for Programmers](http://www.apress.com/9781430262534) (<http://www.apress.com/9781430262534>) by Michael Coughlan is an excellent choice), and start exploring the world of COBOL. No punch cards required.

## GnuCOBOL

[GnuCOBOL](http://sourceforge.net/projects/open-cobol/) (<http://sourceforge.net/projects/open-cobol/>) (formerly known as OpenCOBOL) is a modern, open source, COBOL compiler. It works by translating COBOL code into C and compiling the code using GCC. While the project does not claim standards compliance, it passes most of the tests in the COBOL 85 test suite from the National Institute of Standards and Technology. Other compilers might be more standards compliant or contain the same quirks as their historical antecedents, but GnuCOBOL is the compiler used by the other two projects I cover below.

To install GnuCOBOL on Fedora:

```
sudo dnf install open-cobol
```

Installing under most other Linux distributions and other UNIX-style systems is just as simple. For Debian-based distributions, just replace "dnf" with "apt-get". Mac users can use homebrew to install GnuCOBOL. Installing on Windows is a little more

complicated, but [instructions are available](http://opencobol.add1tocobol.com/oc_gettingstarted_windows.html) ([http://opencobol.add1tocobol.com/oc\\_gettingstarted\\_windows.html](http://opencobol.add1tocobol.com/oc_gettingstarted_windows.html)) for those interested in trying GnuCOBOL on Windows.

To compile source code in a file name 'PROGRAM.cob' into an executable file run:

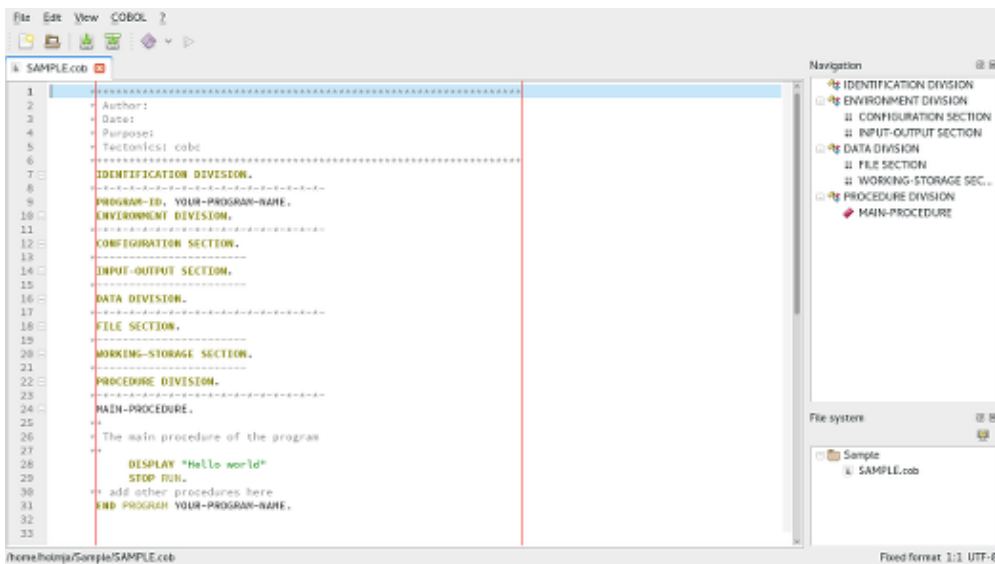
```
cobc -x PROGRAM.cob
```

There are many other compilation options, which are listed when running "cobc -h". The [GnuCOBOL FAQ](http://opencobol.add1tocobol.com/gnucobol/) (<http://opencobol.add1tocobol.com/gnucobol/>) is very extensive and provides a large amount of information about using GnuCOBOL.

## OpenCobolIDE

You can write COBOL using the text editor of your choice, but integrated development environments make life easier. [OpenCobolIDE](https://github.com/OpenCobolIDE/OpenCobolIDE) (<https://github.com/OpenCobolIDE/OpenCobolIDE>) is a specialized IDE designed for working with COBOL. It provides code completion, code folding, syntax highlighting, and a navigation tree, all of which aid in working with COBOL code. It also has a wide variety of color schemes, both light and dark, so most users should be able to find a theme they are comfortable with using.

OpenCobolIDE is a well designed program and is very nice to work with. It is, as noted above, a very specialized IDE that only work with one programming language, but it works with that language quite well. The interface is simple and should not require too much effort to get used to, but should a user need help, the [OpenCobolIDE documentation](http://opencobolide.readthedocs.org/en/latest/) (<http://opencobolide.readthedocs.org/en/latest/>) provides a good overview of the IDE.



To install OpenCobolIDE on Fedora you will need to install a few prerequisites first. Install them by running:

```
sudo dnf install open-cobol python3-qt5
```

Then install OpenCobolIDE by running:

```
sudo pip3 install OpenCobolIDE
```

Installation for other distributions is similar (just replace "dnf" with the appropriate command and install your distribution's equivalent of the open-cobol and python3-qt5 packages). Some distributions do have OpenCobolIDE available as a package in their repositories, so check to see if it is available before install via "pip". Windows and Mac users can [download OpenCobolIDE installers](https://launchpad.net/cobcide/+download) (<https://launchpad.net/cobcide/+download>) and use those instead of using "pip".

## COBOL bridge for Node.js

Maybe you would like to combine some COBOL code with a more modern development technology? If that is the case, you are in luck. [COBOL bridge for Node.js](https://github.com/IonicaBizau/node-cobol) (<https://github.com/IonicaBizau/node-cobol>) allows you to use COBOL code in Node.js projects.

Node-cobol requires GnuCOBOL, so install that using the instructions above and then run:

```
npm install cobol
```

According to the developer, the project is production ready, but honestly it is such an oddity that it is unlikely that it is actually being (or will ever be) used in production. Maybe some enterprising reader will come up with a practical use for Node-cobol and share the story of their project with us?

Topics: [Programming \(/tags/programming\)](/tags/programming/)

---



[\(/users/holmja\)](/users/holmja/)

## About the author

**Joshua Allen Holm** - Joshua Allen Holm is a Grand Rapids, Michigan, based writer and technology consultant. As one of Opensource.com's community moderators, he writes articles about open access, open education, and open source software. You can find Joshua on [GitHub \(https://github.com/holmja\)](https://github.com/holmja) and [Zotero \(https://zotero.org/holmja\)](https://zotero.org/holmja), or email him at: [holmja@opensource.com](mailto:holmja@opensource.com) (<mailto:holmja@opensource.com>).

- [More about me \(/users/holmja\)](/users/holmja/)
- [Learn how you can contribute \(/participate\)](/participate/)

## Recommended reading



### [3 open source Python GUI frameworks](/resources/python/gui-frameworks?utm_campaign=intrel)

[\(/resources/python/gui-frameworks?utm\\_campaign=intrel\)](/resources/python/gui-frameworks?utm_campaign=intrel)



### [How to create reproducible build environments with Rebuild](/article/17/8/rebuild?utm_campaign=intrel)

[\(/article/17/8/rebuild?utm\\_campaign=intrel\)](/article/17/8/rebuild?utm_campaign=intrel)



### [Create versatile visualizations with D3 and Angular](/article/17/8/d3-angular?utm_campaign=intrel)

[\(/article/17/8/d3-angular?utm\\_campaign=intrel\)](/article/17/8/d3-angular?utm_campaign=intrel)



### [What was your first](#)



### [Top 3 open source](#)



### [Managing tabular data in](#)

[programming language?](#)[\(/article/17/8/first-programming-language-poll?utm\\_campaign=intrel\)](/article/17/8/first-programming-language-poll?utm_campaign=intrel)[Python IDEs](#)[\(/resources/python/ides?utm\\_campaign=intrel\)](/resources/python/ides?utm_campaign=intrel)[Groovy using Tika](#)[\(/article/17/8/tika-groovy?utm\\_campaign=intrel\)](/article/17/8/tika-groovy?utm_campaign=intrel)

## 13 Comments



Thane Hubbell on 16 Oct 2015

1

This was a very fair article covering COBOL Open Source initiatives. Thank you.



LoriZ on 19 Oct 2015

1

A few years ago I posted on HalfBakery (a brainstorming/ideas/invention site) about the idea of creating a COBOL-like language for mobile computing, specifically for doing the actual coding on an actual cell phone. That post is at [http://www.halfbakery.com/idea/COBOL-like\\_20programming\\_20language\\_20for...](http://www.halfbakery.com/idea/COBOL-like_20programming_20language_20for...) ([http://www.halfbakery.com/idea/COBOL-like\\_20programming\\_20language\\_20for\\_20mobile\\_20computing](http://www.halfbakery.com/idea/COBOL-like_20programming_20language_20for_20mobile_20computing))

My rationale is that I use Swype for typing on my Android phone. When doing so, I find that words in the English language (at least those in Android's remarkably G-rated spelling dictionary) is something I can do fluidly, while punctuations and numerals are the main speed bump that makes blog commenting less tedious on a desktop. Coding in Java (the lingua franca of Android apps) on the phone the code would run on is something I can't even begin to contemplate trying. But what about COBOL? Given a mobile editor that can help me with indentation rules, typing code would be a breeze. The data structure needs would be vastly different from PICTUREs, though, but I assume COBOL must have evolved some sense of object orientedness or event drivenness or something of the like. Language syntax consisting mostly of English words would, I think, be just the ticket for making the smart phone a full citizen of the world of general-purpose computing, and something the open source community might logically take an interest in. Should I put up a "kickstarter" or something?



[Seth Kenlon \(/users/seth\)](/users/seth) on 14 Dec 2015

0

I think this is a brilliant idea. I don't see how mobiles are useful if they can't be programmed by the user, and like you, I also cannot imagine trying to code on a mobile with a punctuation-heavy language. I would love to see this idea come to fruition.

Also, I fixed the link for you.

Tod Conover on 20 Oct 2015

0





Grace Hopper did not invent COBOL or write the first compiler for it; however, she was critical in its adoption and often gets inaccurate credit, as is the case here.



[Joshua Allen Holm \(/users/holmja\)](/users/holmja) on 22 Oct 2015

1

I apologize if that sentence is confusing. Grace Hopper developed the first compiler for any computer programming language, but no, it wasn't a COBOL compiler. For that reason, I stated that she is the "Grandmother" of COBOL, because she had great influence on the people who directly developed the language. It would have been incorrect to state that she was the "Mother" of COBOL, which would have implied she invented it. Thanks for your feedback.



[DoctorKennyG \(/users/doctorkennyg\)](/users/doctorkennyg) on 30 Oct 2015

1

I believe the true Grandmothers of COBOL would be Jean Sammet and Gertrude Tierney. However, that's a nitpicky detail.

While the idea of Modern Open Source COBOL is marvelous, it's probably 15 years too late. After Y2K, the descent of COBOL resumed. I don't foresee that trajectory ever changing.

I did COBOL for 24 years, up until almost 18 months ago when I made a successful transition into .NET development. Recently I made the transition into Data Management as a SQL Server Data Warehouse DBA. Short of someone offering me a hugely lucrative gig in COBOL, I don't foresee ever going back for either a contract or a perm gig. The end is coming for COBOL, but the death of a language occurs slowly over many years. Old programming languages never die. They just fade away.

For perverse entertainment, I sometimes answer COBOL questions on Stackoverflow. Yes, it's true, people really do ask COBOL questions on SO. I've done it for a sufficiently long time that I don't even need a compiler in front of me to check my code, which is scary in and of itself.

While the idea of a COBOL Renaissance would be wonderful, it just ain't gonna happen.



[Joshua Allen Holm \(/users/holmja\)](/users/holmja) on 30 Oct 2015

0

DoctorKennyG, thank you for your comment. It is great to hear from someone who has considerable experience working with COBOL. As someone who has only worked with COBOL in the classroom and as a hobbyist, I really appreciate your insights. You are right that Jean Sammet and Gertrude Tierney deserve much credit for the creation of COBOL, but in the family tree analogy, I'd classify them as parents, not grandparents.



jmlietaer on 15 Dec 2015

0

I was a COBOL programmer for some years (1989/1995) and enjoyed the job.

I recently discovered a website that provides an online development server (virtual) and I installed OpenCobol

See: [www.koding.com](http://www.koding.com) (<http://www.koding.com>)



akansha on 11 Nov 2015

0

Grace hopper doing it, - is again disputed. in 1952 , Grace hopper (from USA) wrote compiler, and in 1952 Alick glennie (from Britain) also wrote compiler . Some people used to put hopper as true claimant, and some people used to put alick glennie as true claimant of first compiler writer.

Now the US exported blind feminism and feminist political correctness took over the world , and without further debating just on historicity and merits; they proclaimed Grace hopper as the first compiler writer, just because they needed a symbol for geek feminism. No discussion happened further, and now with political correctness, no one wants to dig deep that who is the correct owner of the title " the first compiler writer?" - irrespective of their gender and nationality. The debate was shut shut down, without offering any genuine argument.

==> in reality, as true IT persons/computer geeks/truth seekers, we should say that Grace hopper and Alick Glennie both independently implemented the first compiler in 1952. Among them who did it the first, still needs to be investigate further and established.



wiztigers on 29 Oct 2015

1

Another example of open-source COBOL project that started this year : TypeCobol

(<https://github.com/TypeCobolTeam/TypeCobol>  
(<https://github.com/TypeCobolTeam/TypeCobol>)).

It aims to be an incremental compiler able to be used for real-time error detection by any IDE (there are currently basic implementations Eclipse/RDz or VisualStudio).

Currently not quite ready for production, but in active development. :)

[Joshua Allen Holm \(/users/holmja\)](/users/holmja) on 30 Oct 2015

0

wiztigers, thanks for sharing. TypeCobol looks like great project. I hope it matures into a solid, production ready product in the future.

[Ankita \(/users/anikita\)](/users/anikita) on 28 Nov 2015

0





I work at one such company. Lots of old timers still writing COBOL. Seeing the code they write in java is a real treat. Helping them understand OOP is fun when you see them grasp a concept and it kinda blows their mind.



[VbCoen \(/users/vbcoenbtconnectcom/\)](/users/vbcoenbtconnectcom/) on 06 Dec 2015

1

Judging by the amount of work that I have been getting over the last 5 - 10 years and despite being 'retired' Cobol is alive and well.

This applies to both m/f (mainframe) and work on PC / servers much of which uses Gnu Cobol.

Cobol programming people will be needed for a long time yet - why ? well the old saying seems to still apply - If it isnt broke don't fix it.

As can be testified by by the number of sites that have been advised to move away from both Cobol and kit from mainframe to PC/servers to language x only to continue with existing (admittedly with updating) having been very bitten, both by escalating costs and poorly designed and written replacements).



(<http://creativecommons.org/licenses/by-sa/4.0/>)

## Sign up for Opensource.com news

Continue

Find us:

[Privacy Policy](#) | [Terms of Use](#) | [Contact](#) | [Meet the Team](#) | [Visit opensource.org](#)