Instruct

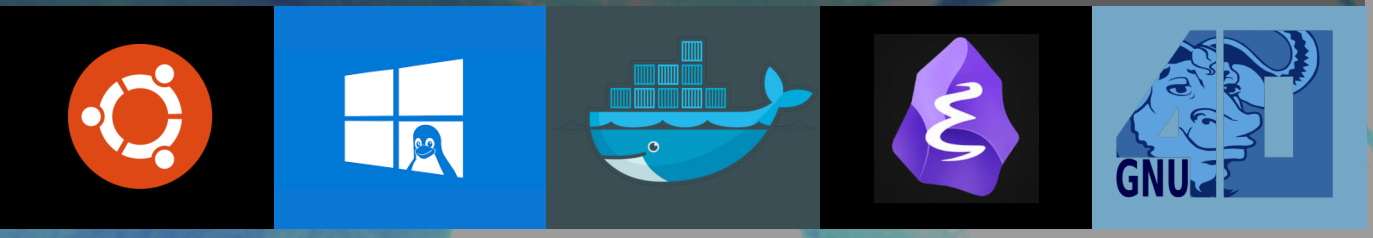
Author: Yuri Tricys
Date: December 19th, 2025

PIDP:3270

CAPSTONE PROJECT

OPEN SOURCE SOFTWARE DEVELOPMENT:

BUILDING EMACS FROM SOURCE: DIGITAL MEDIA



vcc Build Hello

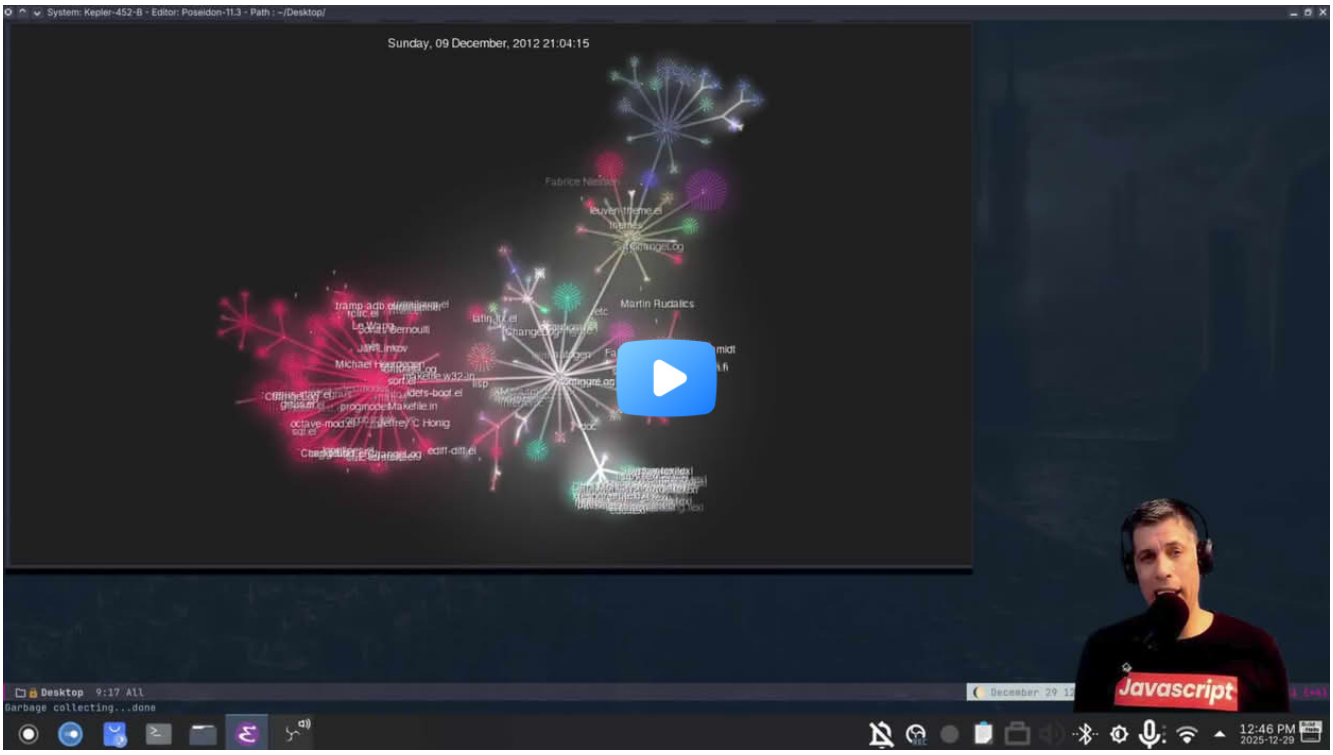
Introduction

This document introduces the digital media lesson series on **Building Emacs From Source**. The goals of the lesson series are to inspire students to learn more about the Emacs text editor, help them decide if they want to install from a package-manager or build Emacs from source, then demonstrate how to build Emacs from source along with additional related concepts, like downloading and verifying source files, the GNU Auto-tools, Performance tuning for Emacs builds, and how to build Emacs in a Docker container.

1. Introduction Video [3:14]

The first video introduces the series, setting the stage for more technical content to come. It covers the overall path of the series, including downloading and verifying Emacs source files, understanding the GNU build tools, working through dependencies and configure options, and then compiling Emacs on Linux Ubuntu 25.04.

It also mentions how the same ideas transfer to WSL/WSLg on Windows and to macOS.



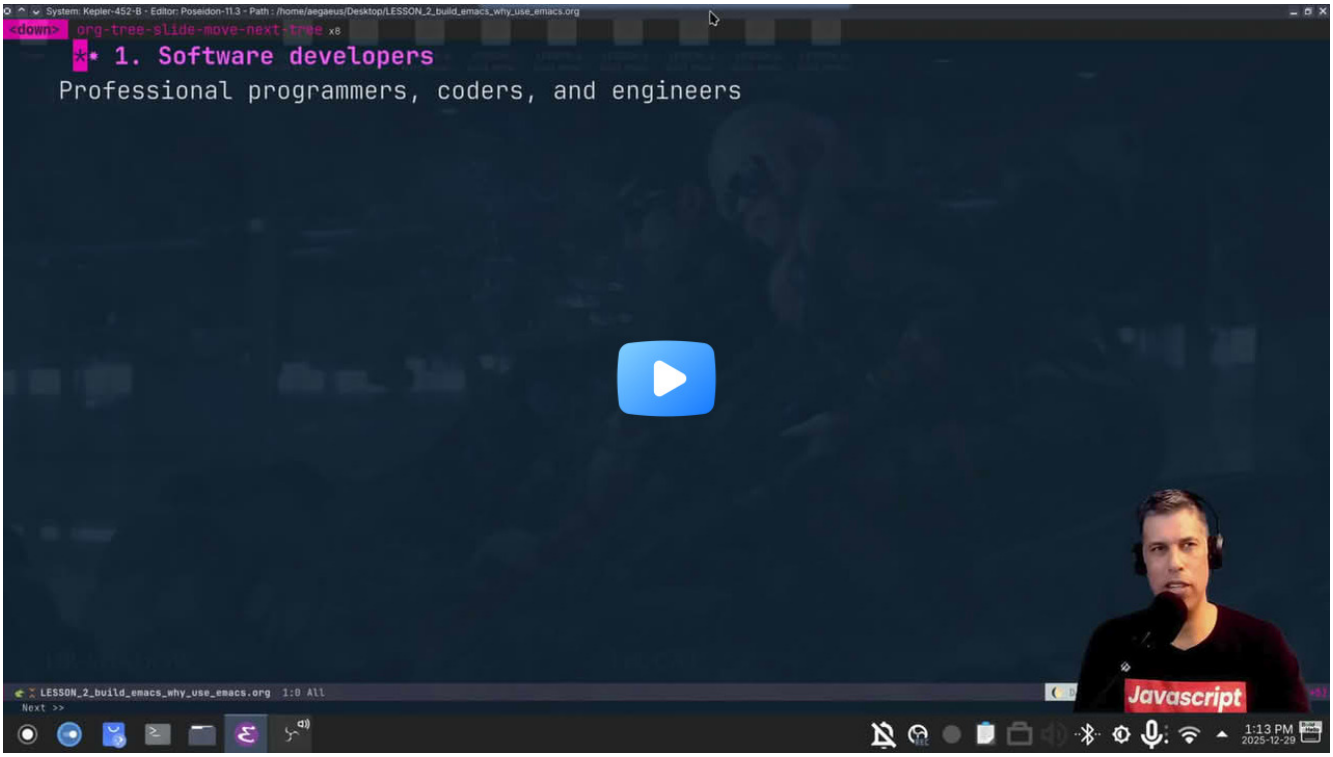
Source
https://youtu.be/En1DzRqkA0k

2. Reasons to Use Emacs [8:49]

This is a big-picture “why Emacs?” lesson. It explains Emacs as an IDE and a unified workspace that helps users stop app-switching and keep their hands on the keyboard. It discusses who uses Emacs (developers, writers, researchers, data folks, even musicians), what people use it for (writing, note-taking, email, projects, reproducible research, AI interaction), and the big “trade-off”: that the learning curve is real, but the payoff is huge, if you stick with it. The goal is to give people a strong foundation for using Emacs and confidence to continue learning.

The lesson is the foundation for the rest of the series. It sets the big picture of what Emacs is and how it can help students be more productive and focused.

It also gives some indication of the nature of the Emacs community, and why that community makes learning Emacs a good decision in terms of the probability of future obsolescence.

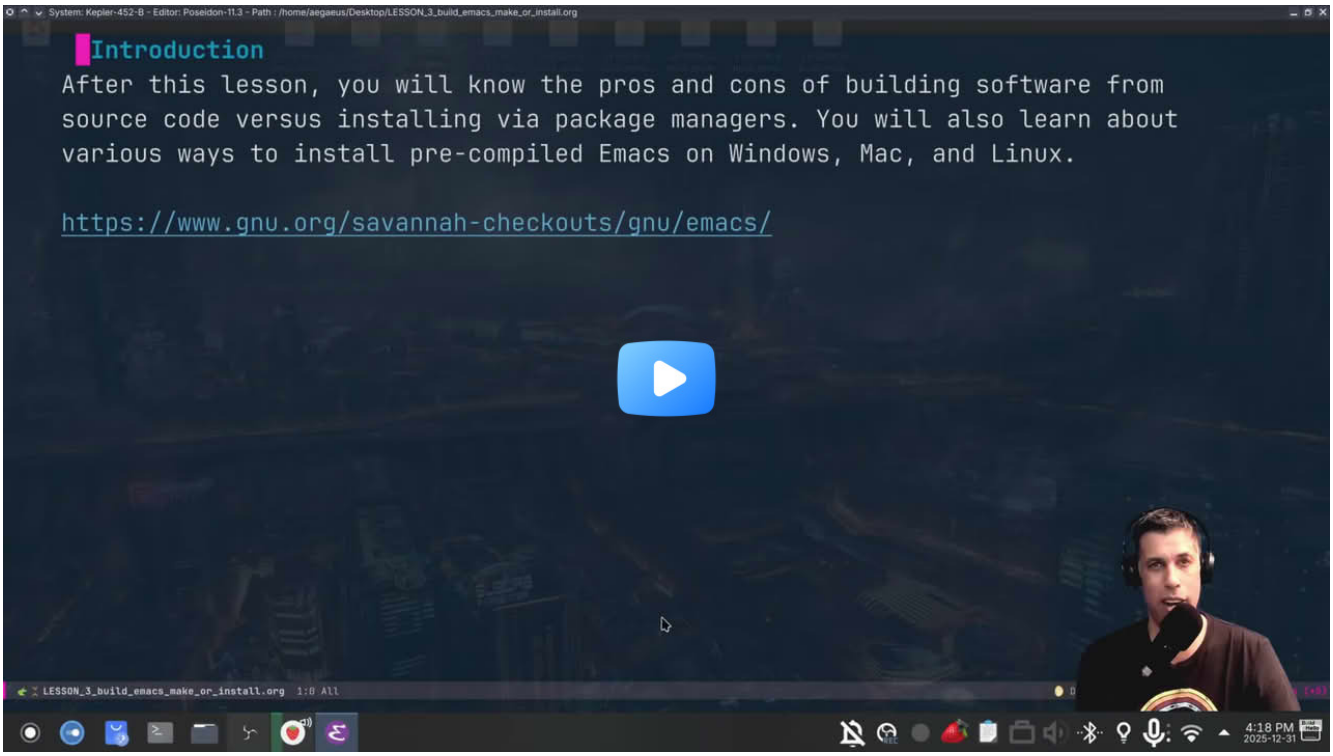


Source
https://youtu.be/hndvNH-uEAg

3. Install or Build From Source [9:01]

This lesson breaks down the practical choice between installing Emacs via a package manager or building from source. It explains why package managers are a good installation method (speed, updates, dependency handling, easy uninstall) and why building from source may be chosen (verifying integrity, custom build options, better optimization, cutting-edge features, reproducible builds, debugging insight, and learning the full process).

It also provides concise install commands for Windows, macOS, and Linux.

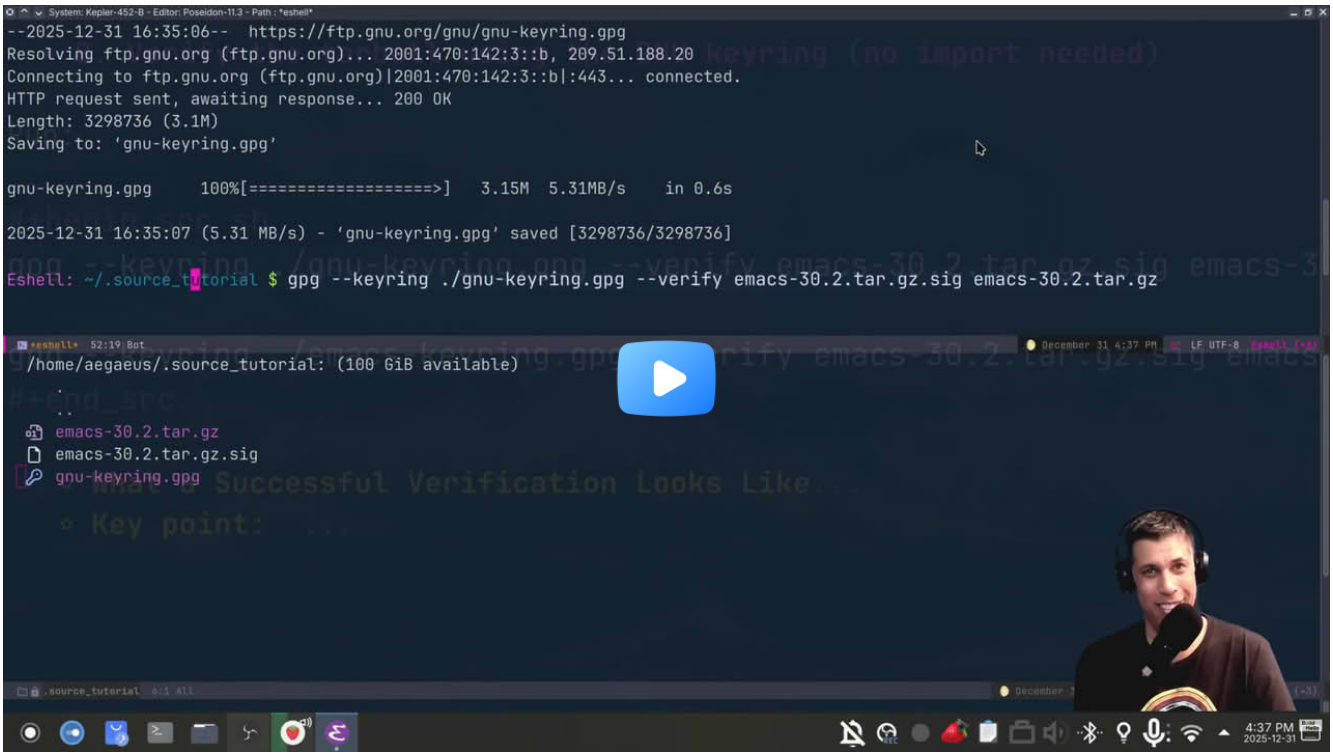


Source:
https://youtu.be/xFfgGps6k-0

4. Download Emacs from Source [14:38]

This lesson covers the security and setup step: where to download Emacs source tarballs from GNU, which files are required (tarball and detached signature), and how to verify downloads using GPG and the GNU keyring.

It also demonstrates downloading with wget/curl/aria2, running signature verification, and extracting the archive to produce a ready-to-build Emacs source directory.



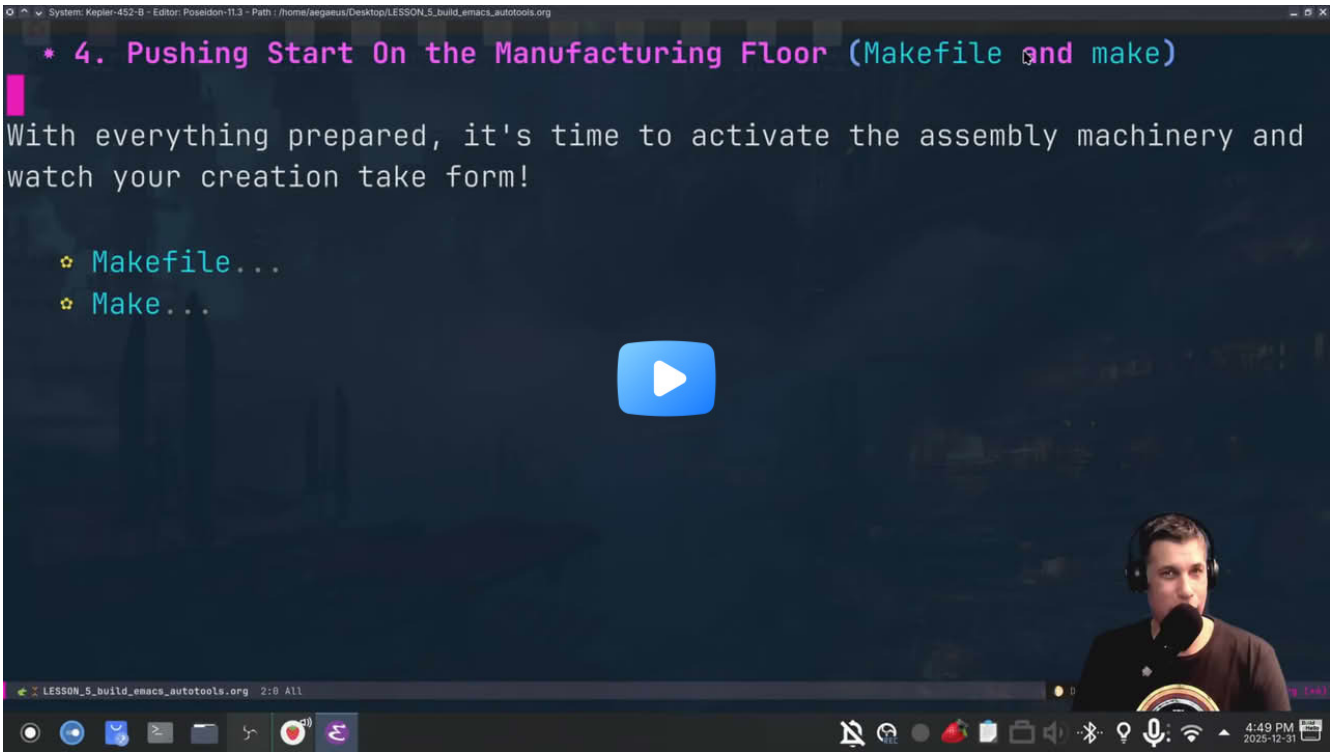
Source:
https://youtu.be/VgSm1NtgXlg

5. Gnu Autotools [8:58]

This lesson explains the GNU build pipeline—autoconf/configure/make/make install – making the process less mysterious.

Uses an “Emacs robot” analogy: using ‘configure’ as the blueprint step (checking the system and planning the build), ‘make’ as the factory floor (compiling and linking), and ‘make install’ as deploying the final product to a market location.

It also clarifies when autogen is required, for example when building from a git checkout versus a release tarball.

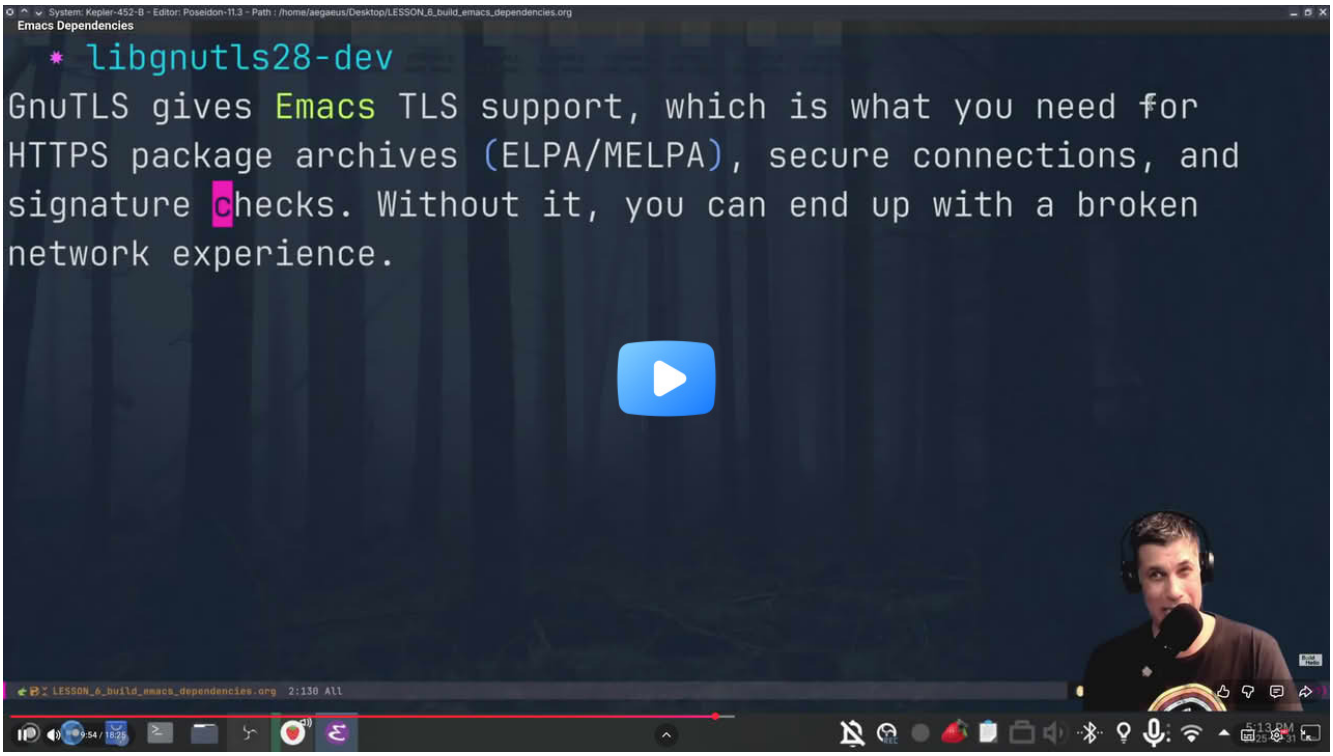


Source:
https://youtu.be/9GBunm7__xY

6. Dependencies [18:26]

This lesson is a dependency walk-through for a modern Emacs build. It describes how Emacs enables features based on available libraries — GUI support, image formats, TLS/HTTPS, SQLite, tree-sitter, native compilation, multilingual text rendering, and more.

It includes a practical Ubuntu-focused package list, explanations of major dependencies' roles, and common issues such as ensuring GCC version matches libgccjit version for native compilation.

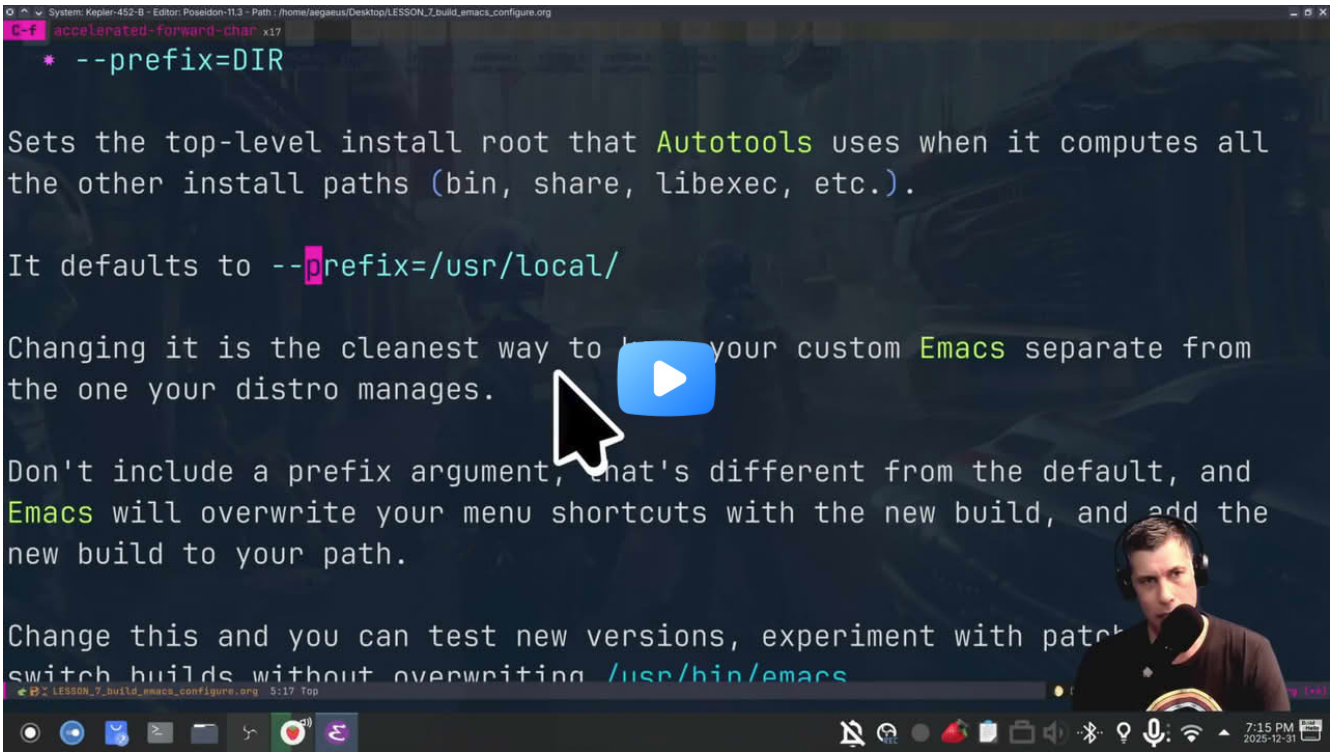


Source:
https://youtu.be/Ri7UZ7g_14Q

7. Emacs Configure Options [17:31]

This lesson focuses on 'configure' options, like how to build for a X11 compositor (windowing system) vs a Wayland/PGTK compositor. It shows how to set install locations with a prefix argument, to maintain multiple versions of Emacs on the same machine, and how to enable native compilation and other performance options.

It also shows how to inspect an existing Emacs installation to determine build options for replication.

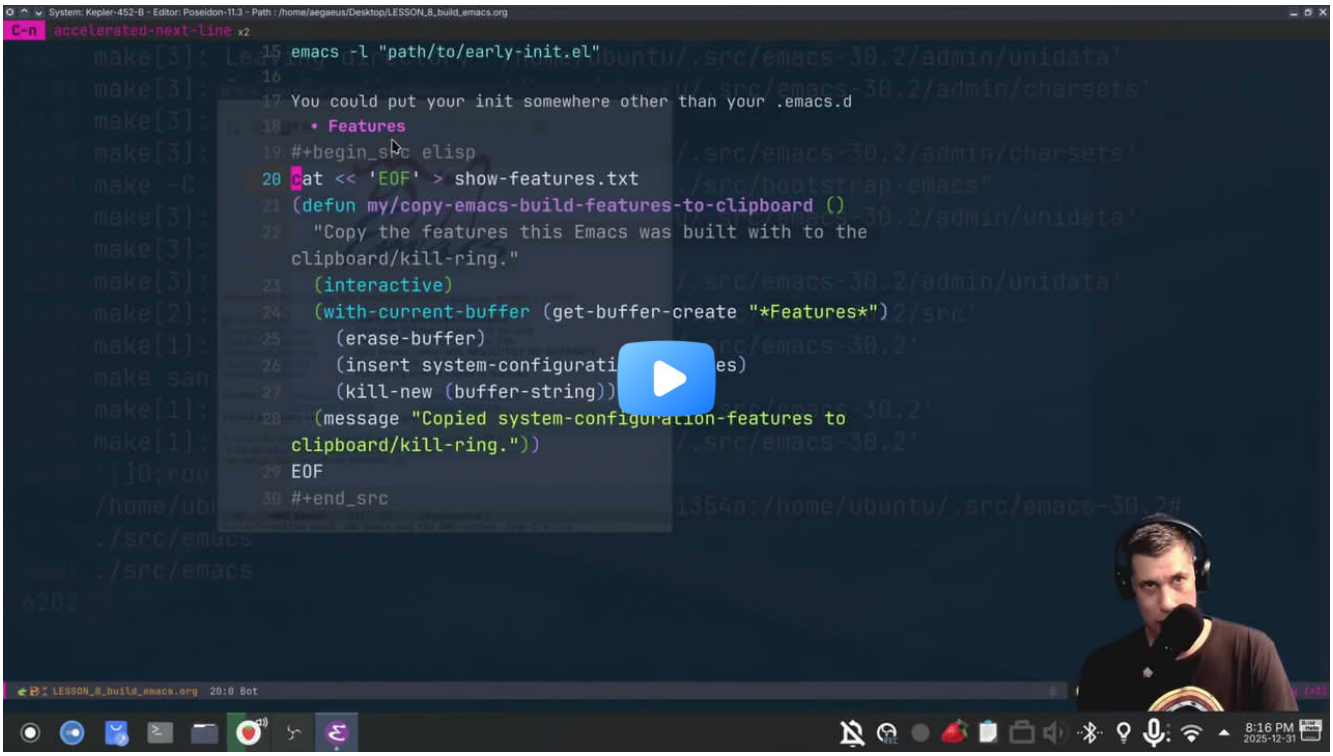


Source:
https://youtu.be/I_uzGnvwZTQ

8. Build Emacs in Docker [41:57][Smart Watch]

Lesson 8 is a hands-on lesson in building Emacs source code from a clean Ubuntu 25.04 Docker container to a working Emacs binary. It runs the full sequence: installing prerequisites, downloading and verifying source files, installing dependencies, configuring with chosen options, compiling with 'make' using multiple cores, and launching Emacs.

It also demonstrates how the feature set changes between a minimal and a more complete Emacs build based on the dependencies installed on the system, which shows how an Emacs build adapts to the available libraries.



Source:
https://youtu.be/fpTV_uHA2ik

Conclusion

This document presented access to the 8 part lesson “Building Emacs from Source”, along with descriptions of each lesson.

The length and classification of each video was also included, to give readers an idea of which lessons may be applicable to those interest in software development, and which lessons might be more interesting to audiences that simply want a better way of managing their computer workflows.

The reader is encouraged to explore the lesson series, and future lessons.

References

[Photograph of a rocket] [Photograph]. (n.d.). Internet Archive. Retrieved December 10, 2025, from https://archive.org/

meaicon. (n.d.). Stream icons [Image]. Retrieved January 5, 2026, from https://www.flaticon.com/free-icons/stream