

# Composer Plugins 101



Extending Composer's default functionality



These slides: [bit.ly/composer-plugins-101](https://bit.ly/composer-plugins-101)



# Composer is an application



# Drupal is an application



**Drupal's functionality can be extended with additional modules.**



**Composer's functionality can be extended  
with additional plugins.**



**Composer plugins are added to a project as a dependency.**



**Composer plugins often have configuration specified in the "extras" section of the project's composer.json file.**





**Starting with Composer 2.2, plugins must be given permission to run via the "allow-plugins" setting in the project's composer.json**



# Plugins we will discuss

- [composer/installers](#)
- [oomphinc/composer-installers-extender](#)
- [drupal/core-composer-scaffold](#)
- [cweagans/composer-patches](#)
- [davidrjonas/composer-lock-diff](#)
- [topfloor/composer-cleanup-vcs-dirs](#)
- [joachim-n/composer-manifest](#)



## composer/installers

- Included as part of drupal/recommended-project Composer template.
- Allows for dependencies to be installed in places other than the project's /vendor directory.
- Demo



# oomphinc/composer-installers-extender

- Companion plugin to composer/installers that allows for any arbitrary package type (such as npm packages) to be defined and then handled by the composer/installers plugin.
- Demo



# drupal/core-composer-scaffold

- Included as part of drupal/recommended-project Composer template.
- Allows for Drupal scaffolding files to be installed outside of the core/ directory.
- Allows for customization of Drupal scaffolding files.
- Demo



# cweagans/composer-patches

- Allows for easy (seriously!) patching of project dependencies (like Drupal core and contrib modules)
- Demo



# dauidrjonas/composer-lock-diff

- Displays easy-to-read summary of changes after a "composer update" (via comparison with Git HEAD).
- Demo



# topfloor/composer-cleanup-vcs-dirs

- Automatically removes .git directories from dependencies when all Composer dependencies are committed to the project repository.
- Demo





# joachim-n/composer-manifest

- Creates a simple "composer-manifest.yaml" file in the project root listing all dependencies (and their exact version numbers) currently used in a project.
- Demo



**What about the drupal/core-project-message  
dependency?**



## drupal/core-project-message

- Provides a welcome message the first time a new Drupal 9 codebase is setup using the drupal/recommended-project template.
- It is normally not needed.
- Demo (to remove!)



# Other useful plugins?



# Questions?

