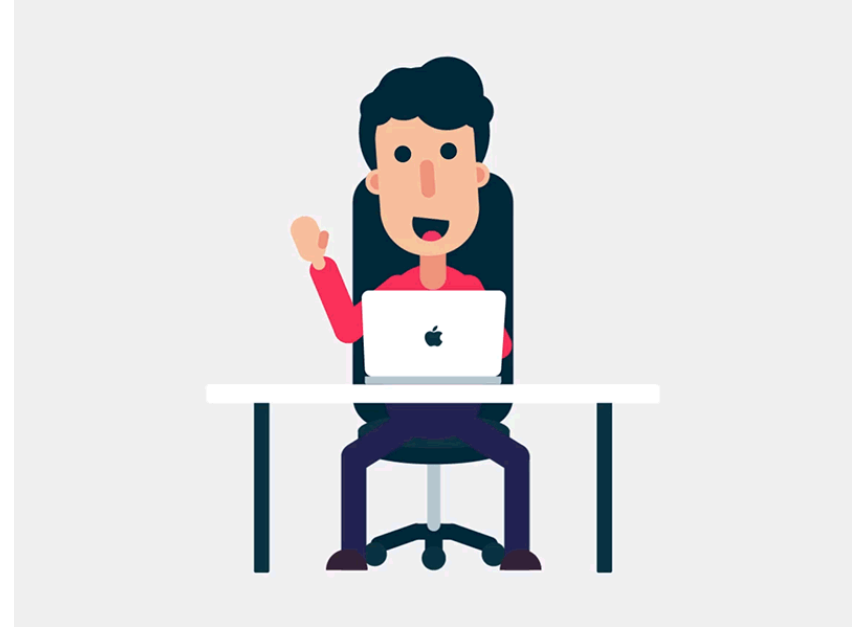# Composer Scripting

Hello!

I am Kunal Kursija

- Mumbai, India
- Drupal Engineer - **@axelerant**
- Twitter: @kunalkursija
- Drupal: https://www.drupal.org/u/kunalkursija
- LinkedIn: https://www.linkedin.com/in/kunalkursija/

What is a script?

In Composer's terms, Script can either be a PHP-Callback(static methods only) or CLI commands.

Where are scripts placed?

Scripts are defined at the root level of your composer.json.

```
"scripts": {
    "key1": "value",
    "key2": [
        "value1",
        "value2"
    ]
},
```

Composer Events

Composer fires different events during its execution, Which allows other packages to hook into composer and perform certain tasks.

Categorising Events

```
kunalkursija@kunals-MacBook-Pro > ~ > composer install
```
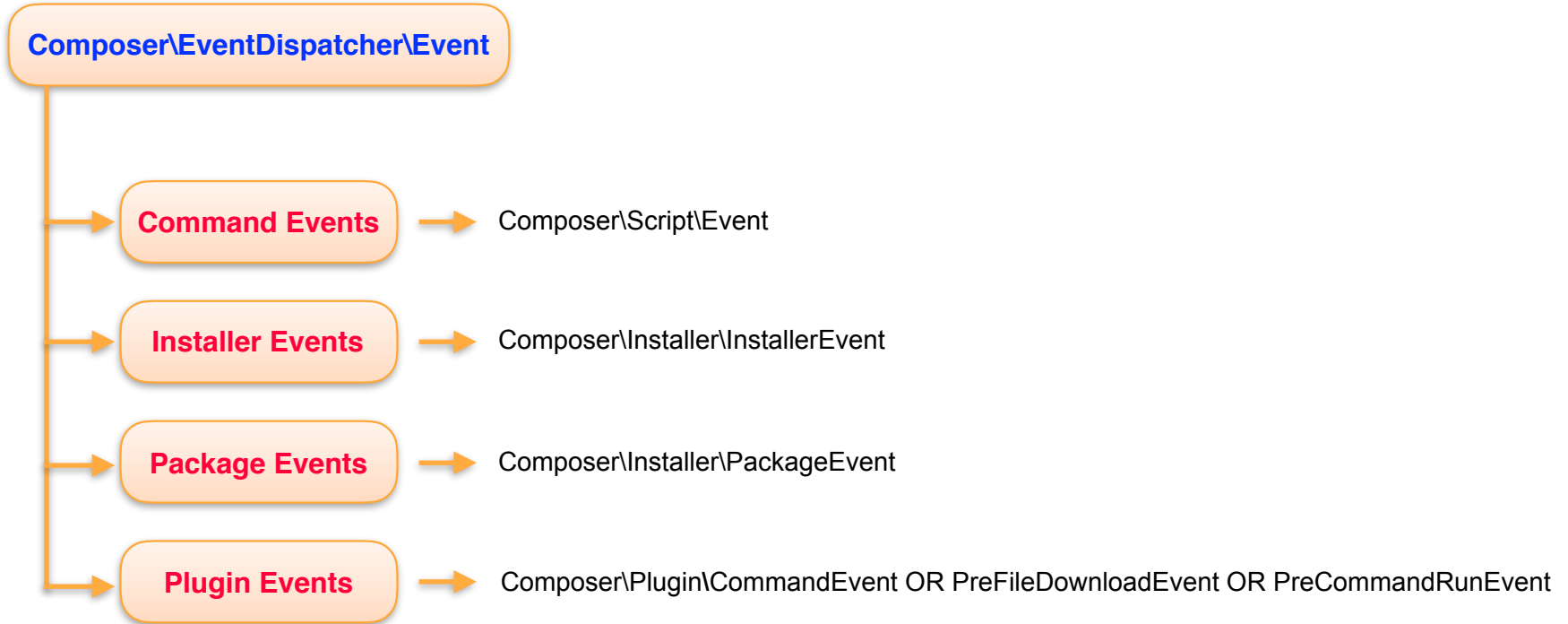
Command Events   Installer Events   Package Events   Plugin Events

Event Classes

Composer\EventDispatcher\Event

Command Events ➡ Composer\Script\Event

Installer Events ➡ Composer\Installer\InstallerEvent

Package Events ➡ Composer\Installer\PackageEvent

Plugin Events ➡ Composer\Plugin\CommandEvent OR PreFileDownloadEvent OR PreCommandRunEvent

Command Events

- pre-install-cmd: occurs before the `install` command is executed with a lock file present.
- post-install-cmd: occurs after the `install` command has been executed with a lock file present.
- pre-update-cmd: occurs before the `update` command is executed, or before the `install` command is executed without a lock file present.
- post-update-cmd: occurs after the `update` command has been executed, or after the `install` command has been executed without a lock file present.
- post-status-cmd: occurs after the `status` command has been executed.
- pre-archive-cmd: occurs before the `archive` command is executed.
- post-archive-cmd: occurs after the `archive` command has been executed.
- pre-autoload-dump: occurs before the autoloader is dumped, either during `install` / `update`, or via the `dump-autoload` command.
- post-autoload-dump: occurs after the autoloader has been dumped, either during `install` / `update`, or via the `dump-autoload` command.
- post-root-package-install: occurs after the root package has been installed, during the `create-project` command.
- post-create-project-cmd: occurs after the `create-project` command has been executed.

Installer Events

- pre-dependencies-solving: occurs before the dependencies are resolved.
- post-dependencies-solving: occurs after the dependencies have been resolved.

Package Events

- pre-package-install: occurs before a package is installed.
- post-package-install: occurs after a package has been installed.
- pre-package-update: occurs before a package is updated.
- post-package-update: occurs after a package has been updated.
- pre-package-uninstall: occurs before a package is uninstalled.
- post-package-uninstall: occurs after a package has been uninstalled.

Plugin Events

- init: occurs after a Composer instance is done being initialized.
- command: occurs before any Composer Command is executed on the CLI. It provides you with access to the input and output objects of the program.
- pre-file-download: occurs before files are downloaded and allows you to manipulate the `RemoteFilesystem` object prior to downloading files based on the URL to be downloaded.
- pre-command-run: occurs before a command is executed and allows you to manipulate the `InputInterface` object's options and arguments to tweak a command's behavior.

Demo

Let's see events in action…

Writing & Executing Custom Composer Commands

- Composer allows us to create & execute custom commands
- Executing PHP scripts
- Executing Shell scripts
- Executing Drush commands
- Executing Composer commands
- Script Referencing

Demo

Let's see custom commands in action…

Questions?

Thank You!

DrupalCamp Atlanta