szwajcarski scyzoryk programisty PHP

Krzysztof Dyszczyk

WordCamp Łódź 2019

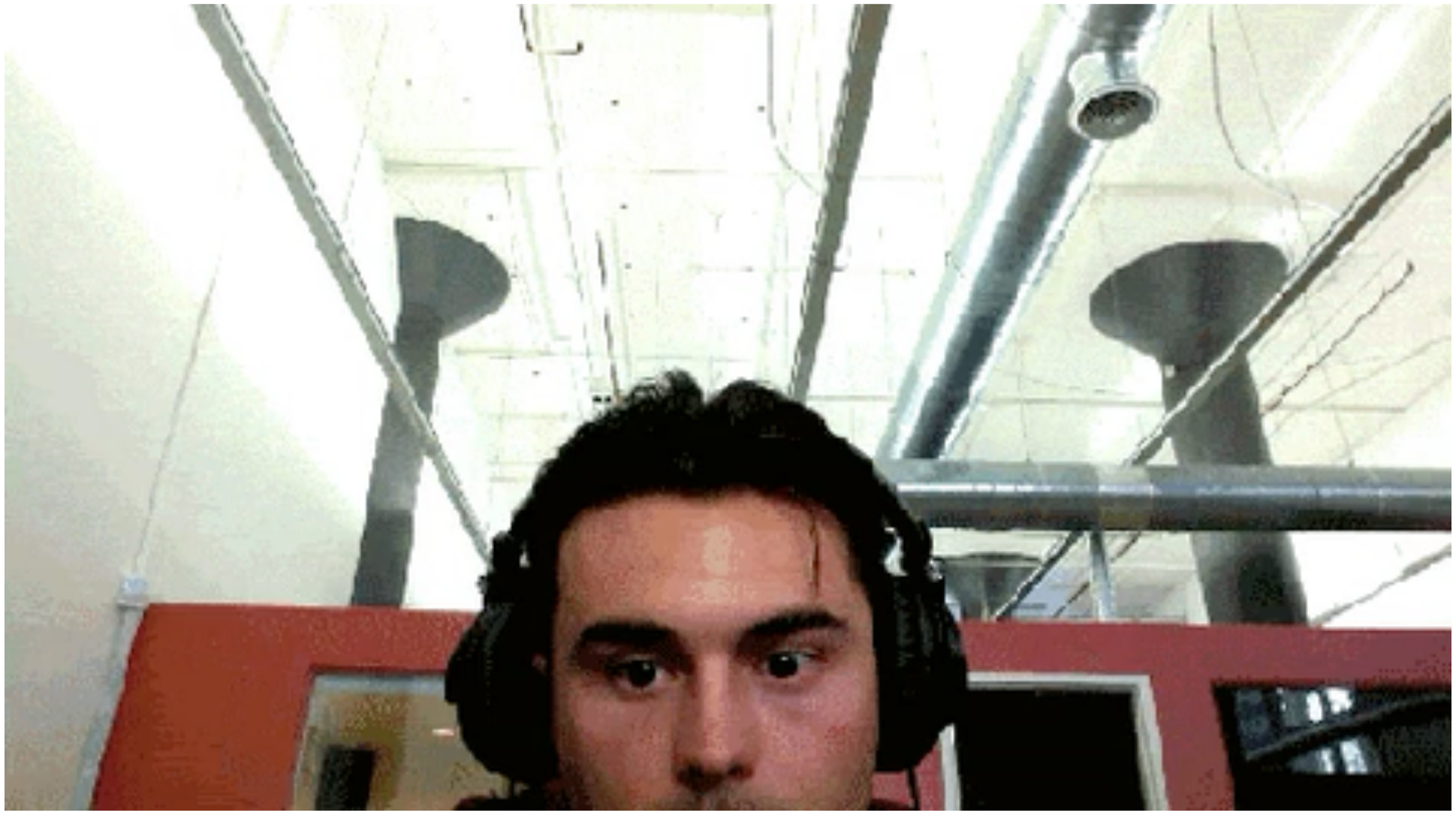
WPDESK

INSPIRELABS

WooCommerce

Upgrade the minimum supported version of PHP to 7.0 and WordPress to 5.0 with the next release

Pora porzucić pisanie **legacy code**




COMPOSER

Cele na dziś

- rozumiemy czym jest Composer
- umiemy używać istniejących pakietów
- umiemy pisać własne pakiety

Fajnie instaluje różne rzeczy

PHP-ML

Guzzle

Gaufrette

Omnipay

Intervention Image

ImageWorkshop

PHPWord

PHP Geo

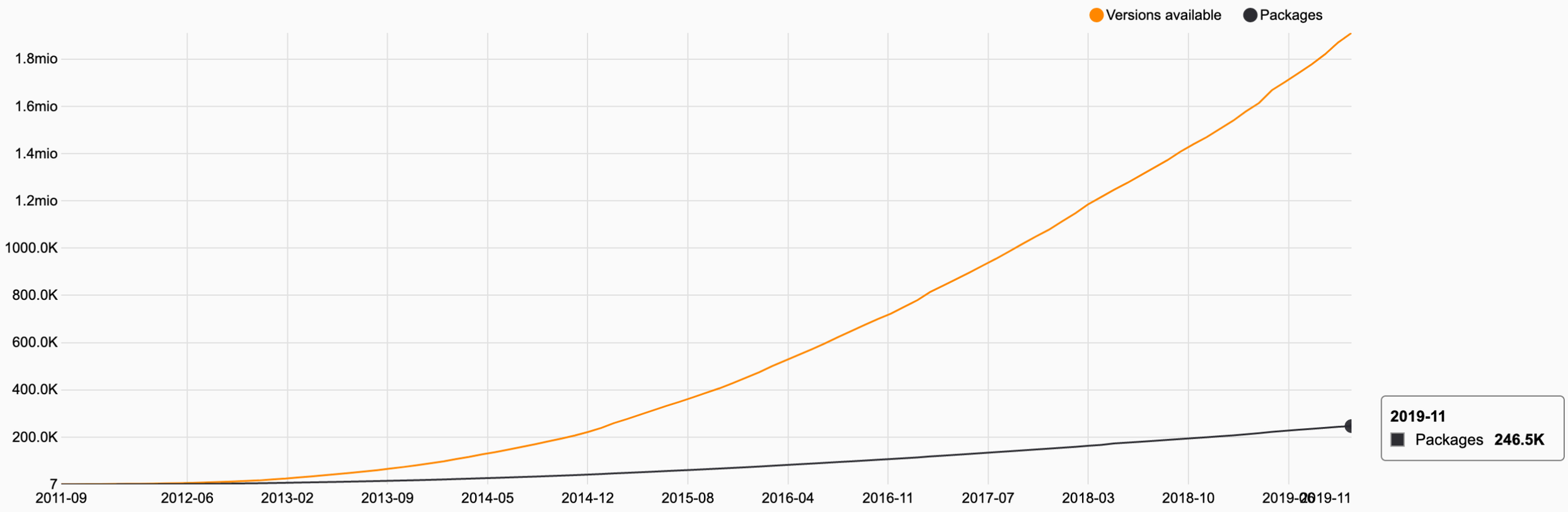
PHP CS

PHPStan

Faker

Whoops

Packages/versions over time



Autoloader

Zabawki dla cool kids

Zdarzenia

Skrypty

Instalatory

Pluginy

...

Zarządzanie zależnościami

Komputer uruchomiony?

Dostęp do sieci?

Przeglądarka WWW?

Konto na GitHub lub Gitlab?

Klient GIT?

git --version

PHP CLI?

php --version

WordPress?

IDE?

https://github.com/dyszczo/composer-workshop

1. Instalacja Composera

A. https://getcomposer.org/

B. Zainstaluj composera.

C. Uruchom composera z konsoli.

composer.phar

php ./composer.phar

php composer.phar

composer

Scaffolding

2. Instalacja WordPress

composer create-project johnpbloch/wordpress

composer.json

konfiguracja

composer.lock

aktualnie zainstalowane pakiety

vendor

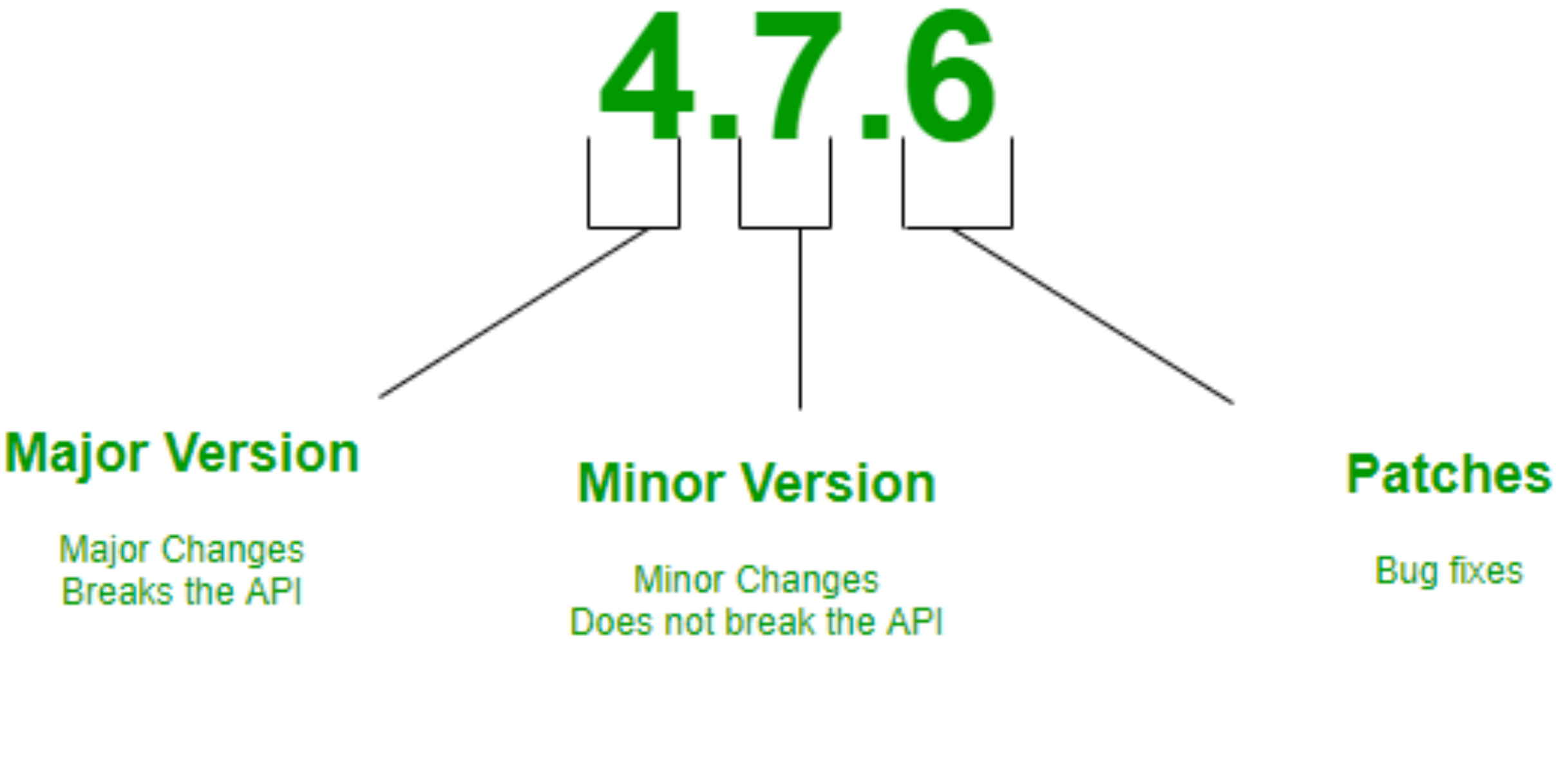
composer info

Pakiet

- wersjonowany kod

```
"require": {
  "php": ">=5.6.20",
  "johnpbloch/wordpress-core-installer": "^1.0",
  "johnpbloch/wordpress-core": "5.3.0"
}
```

Wersjonowanie semantyczne



*

dev-stefan

1.2.*

^1.2.3

~1.2.3

1.1

^5.3 || >= 6.2

~5.3.2 !=5.3.5

3. Wersjonowanie

https://semver.mwl.be/

3.1

Wersja dowolna, ale najnowsza możliwa

3.2

Wersja co najmniej 4.3

3.3

Wersja co najmniej 4.3, ale nie łamiąca kompatybilności

3.4

Wersja co najmniej 4.3.1, nie łamiące kompatybilności, ale z wykluczeniem 4.4.1

3.5

Wersja co najmniej 5.2.2 ale mniejsza niż 5.3

3.6

Najnowsza wersja z gałęzi rozwojowej master

3.7

Wersja 5.0 lub 5.1

Repozytorium

- kolekcja pakietów

Packagist

WordPress Packagist

4. Instalacja wtyczki

- Dodaj WordPress Packagist
- Zainstaluj wtyczkę Yoast Seo

Pakiet nie powinien być instalowany do wnętrza innego pakietu

Bedrock

https://composer.rarst.net/

https://www.smashingmagazine.com/2019/03/composer-wordpress/

composer install

composer update

Autoładowanie

PSR-0 Yoast_Seo_Client

PSR-4 \Yoast\Seo\Client

Classmap

Files

5. Własna wtyczka

- Umieść szkielet wtyczki w WordPress
- Zainicjalizuj Composer composer init
- Dodaj autoloader
- Odśwież autoloader composer dump
- Odwołaj się do klasy Plugin

5.2 Inicjalizacja Composera

- Polecenie composer init
- Package name: zgodne z url w github ;)
- Minimum stability: czy pozwalamy instalować niestabilne zależności?
- Require/require dev: zależności produkcyjne i dla developerów

5.3 Dodawanie autoloadera

- Dopisz sekcję autoload typu classmap
- Dodaj require w pliku wtyczki
- Wykonaj polecenie composer dump
- Uruchom klasę Plugin::run();

Dobre praktyki

```
require __DIR__ . '/vendor/autoload.php'
```

*

dev-master

1.2.*

^1.2.3

~1.2.3

1.1

W środowisku dev normalnie

composer update

composer install

Na produkcji optymalizacja

composer install --no-dev --optimize-autoloader --prefer-dist --no-interaction

Zdefiniuj konkretną platformę produkcyjną

```
config": {
    "platform": {
        "php": "7.1"
  }
}
```

composer.lock dodaj do git (jeśli to nie pakiet)

vendor do ignore

po composer update

testuj

sort-packages: true

6. Używamy pakietów

- Umieść szkielet wtyczki w WordPress
- Przeczytaj opis biblioteki https://gitlab.com/wpdesk/wp-notice
- Dodaj pakiet do wtyczki i odpowiednio użyj aby pokazać notice o wybranej treści

composer require --dev

composer install --no-dev

7. Pakiety dev

- Dodaj pakiet dev https://github.com/WordPress/WordPress-Coding-Standards
- Zwróc uwagę na plik phpcs.xml.dist
- Po instalacji sniffer powinien być dostępny w katalogu vendor/bin
- Użyj sniffera vendor/bin/phpcs .

Prywatne pakiety

Pakiety to doskonałe antidotum na DRY

8.1 Nowy pakiet

- Tworzymy pakiet w nowym katalogu za pomocą composer init
- Umieszczamy katalog src z zawartością w pakiecie
- Uzupełniamy sekcję autoload

8.2 Użycie pakietu

- Możemy dodać katalog jako nowe repozytorium we wtyczce typu **path**
- Używamy pakietu z composera zamiast z wtyczki

8.3 Zdalny pakiet

- Dodaj pakiet do swojego repozytorium git i prześlij na serwer zdalny

- Możesz dodać zdalne repozytorium jako repozytorium typu **vcs**

- Możesz utworzyć konto w https://packagist.org/ i dodać pakiet, jeśli jest publicznie dostępny

Satis

Satis Press

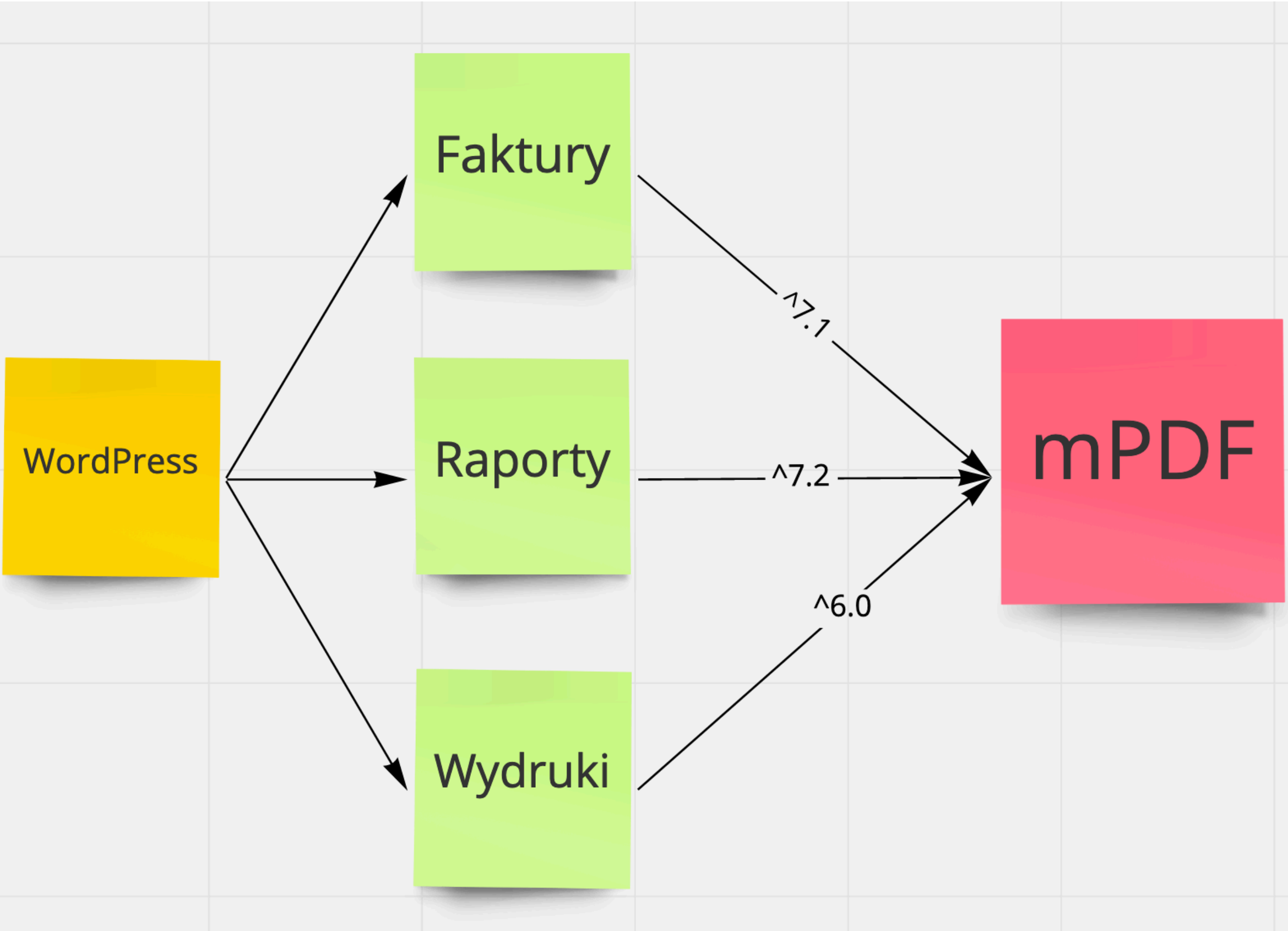
https://github.com/cedaro/satispress

Brak wsparcia WordPress

Dependency Hell

Bedrock

Sage


Faktury
WordPress
Raporty
Wydruki
mPDF
^7.1
^7.2
^6.0

Sterowanie kolejnością

~~Sterowanie~~

~~kolejnością~~

Prefixowanie

9. Prefixowanie

- Dodaj pakiet dev https://github.com/humbug/php-scoper w wersji ^0.12.3
- Zwróć uwagę na plik scoper.inc.php
- Uruchom vendor/bin/php-scoper add-prefix
- Użyj prefixowanych pakietów

Mozart

https://github.com/coenjacobs/mozart

Dziękuję

Krzysiek Dyszczyk

Zatrudniamy!
cv@inspirelabs.pl

WPDESK

INSPIRELABS