



Zarządzanie treścią (CMS)

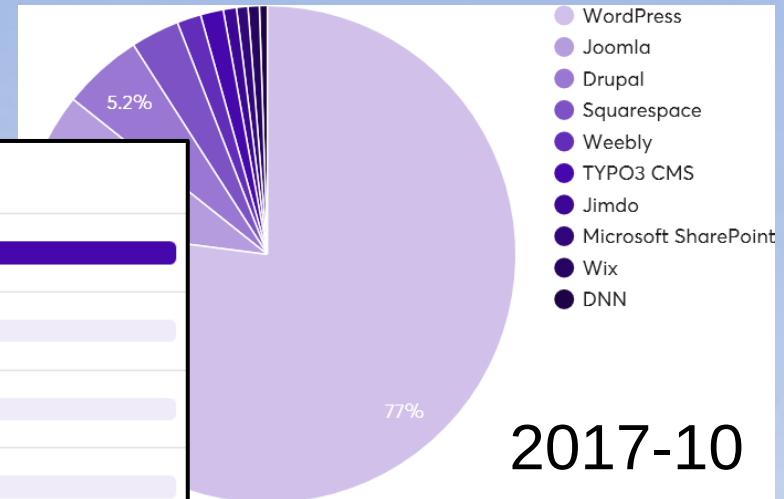
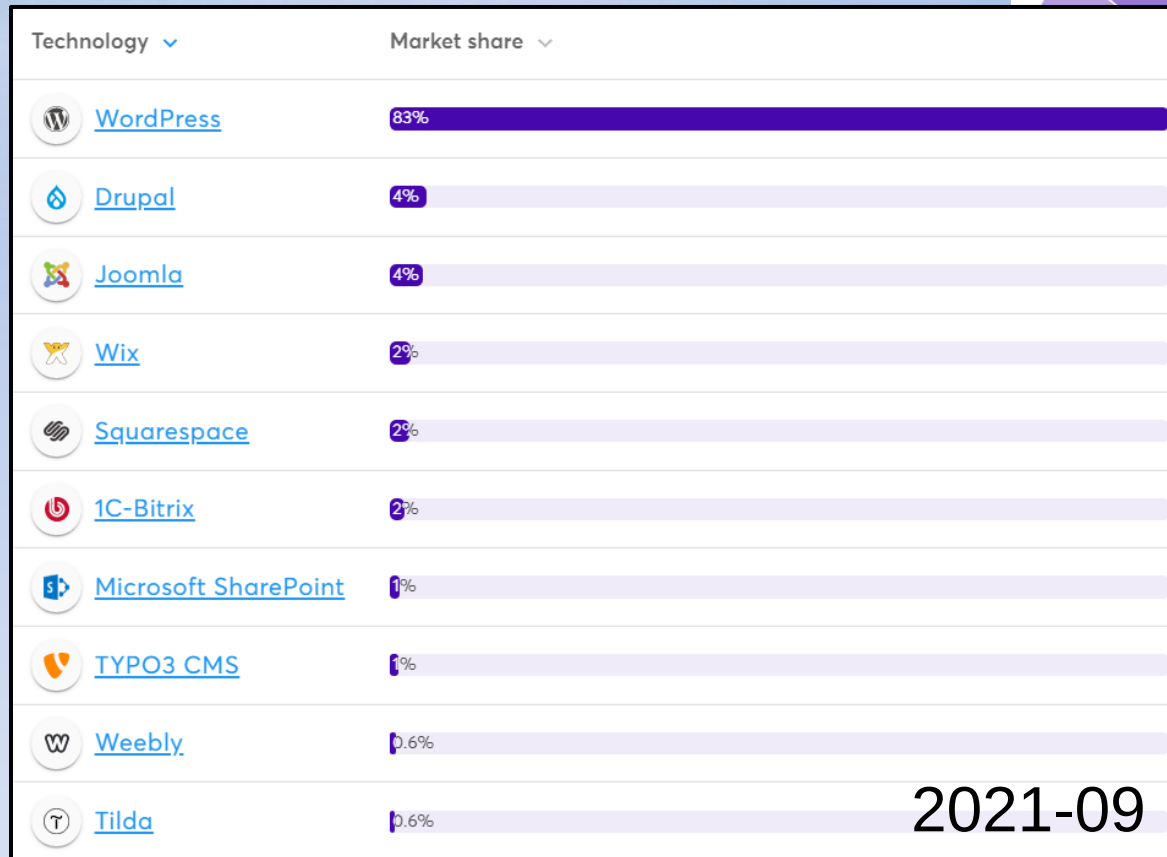
Wykład 05: Systemy zarządzania treścią - omówienie popularnych rozwiązań (2)

dr inż. Mariusz Trzaska,
mtrzaska@mtrzaska.com, <http://www.mtrzaska.com>

Zagadnienia

Ciąg dalszy poprzedniego wykładu.

Popularność CMS wg Wappalyzer (wtyczka do przeglądarek)



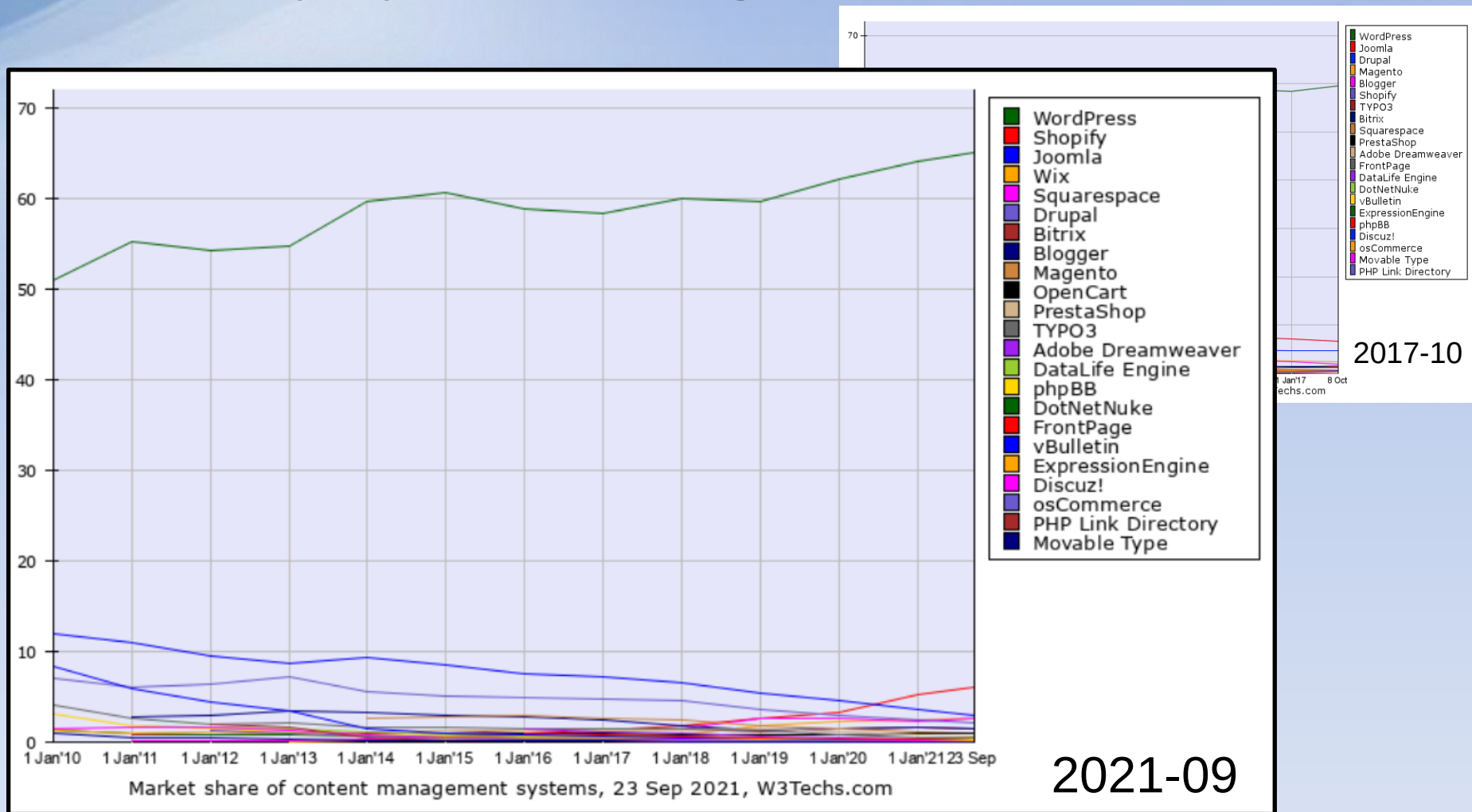
<https://www.wappalyzer.com/categories/cms>

Systemy na których działają najpopularniejsze witryny wg. w3techs

- W3Techs analizuje 10 000 000 najpopularniejszych witryn w Internecie w/g serwisu Alex'a.
- Około:
 - 34.7% w 2021-09,
 - 39.4% w 2020-10,
 - 52% w 2017-10,
 - 71% w 2013-01witryn korzysta z nierozpoznanego/własnego rozwiązania.

źródło: https://w3techs.com/technologies/overview/content_management/all

Udziały rynkowe wg. w3techs.com

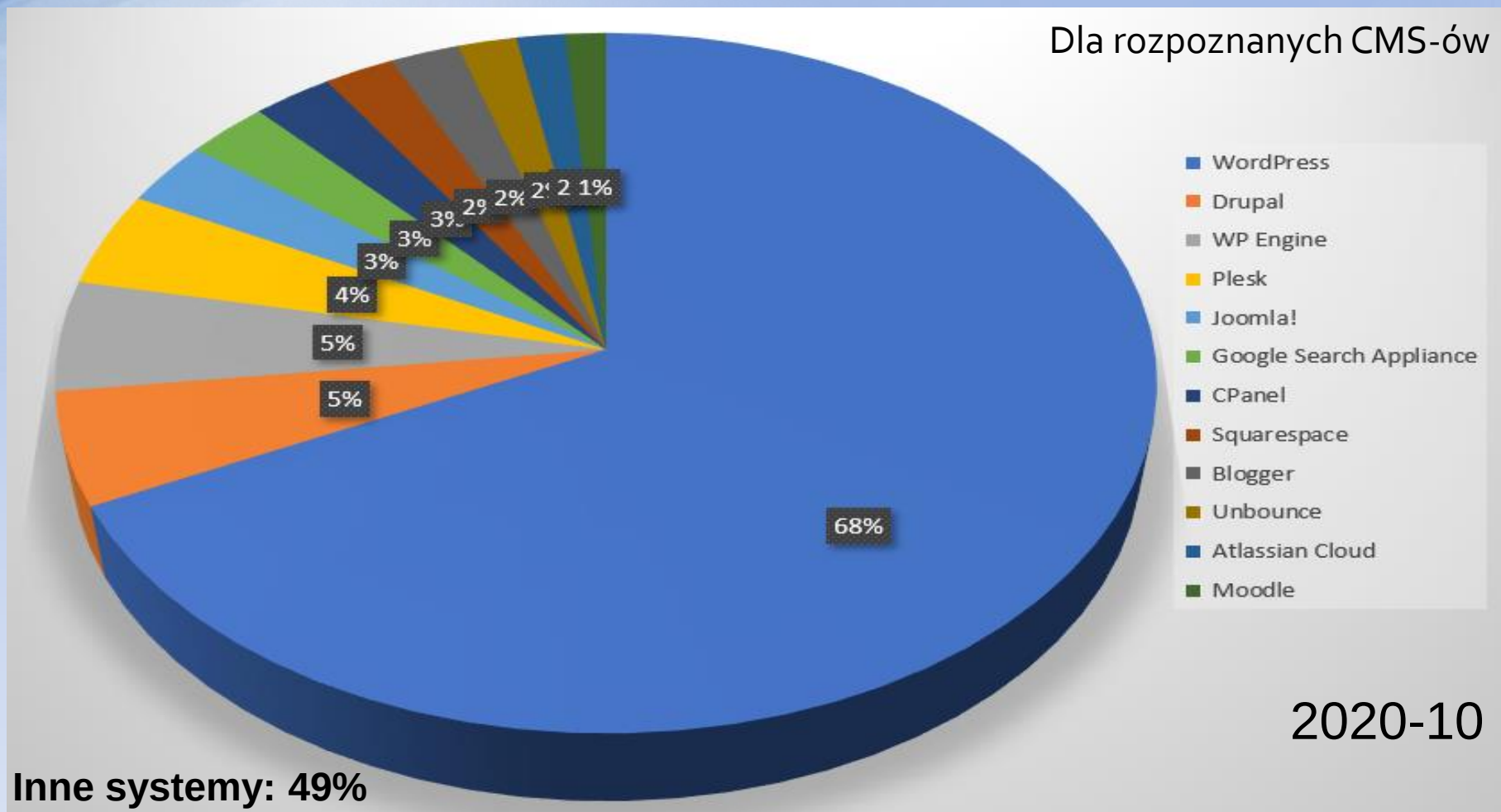


http://w3techs.com/technologies/history_overview/content_management/ms/y

Systemy na których działają najpopularniejsze witryny wg. BuiltWith

- BuiltWith jest kolejnym serwisem badającym popularność technologii webowych.
- Przeanalizowano najpopularniejsze witryny (1 mln., 100 tys. i 10 tys.) w Internecie w/g serwisu Quantcast.
- Własne rozwiązania systemów:
 - 1 000 000 najpopularniejszych: 49%,
 - 100 000 najpopularniejszych: 61%,
 - 10 000 najpopularniejszych: 65%.

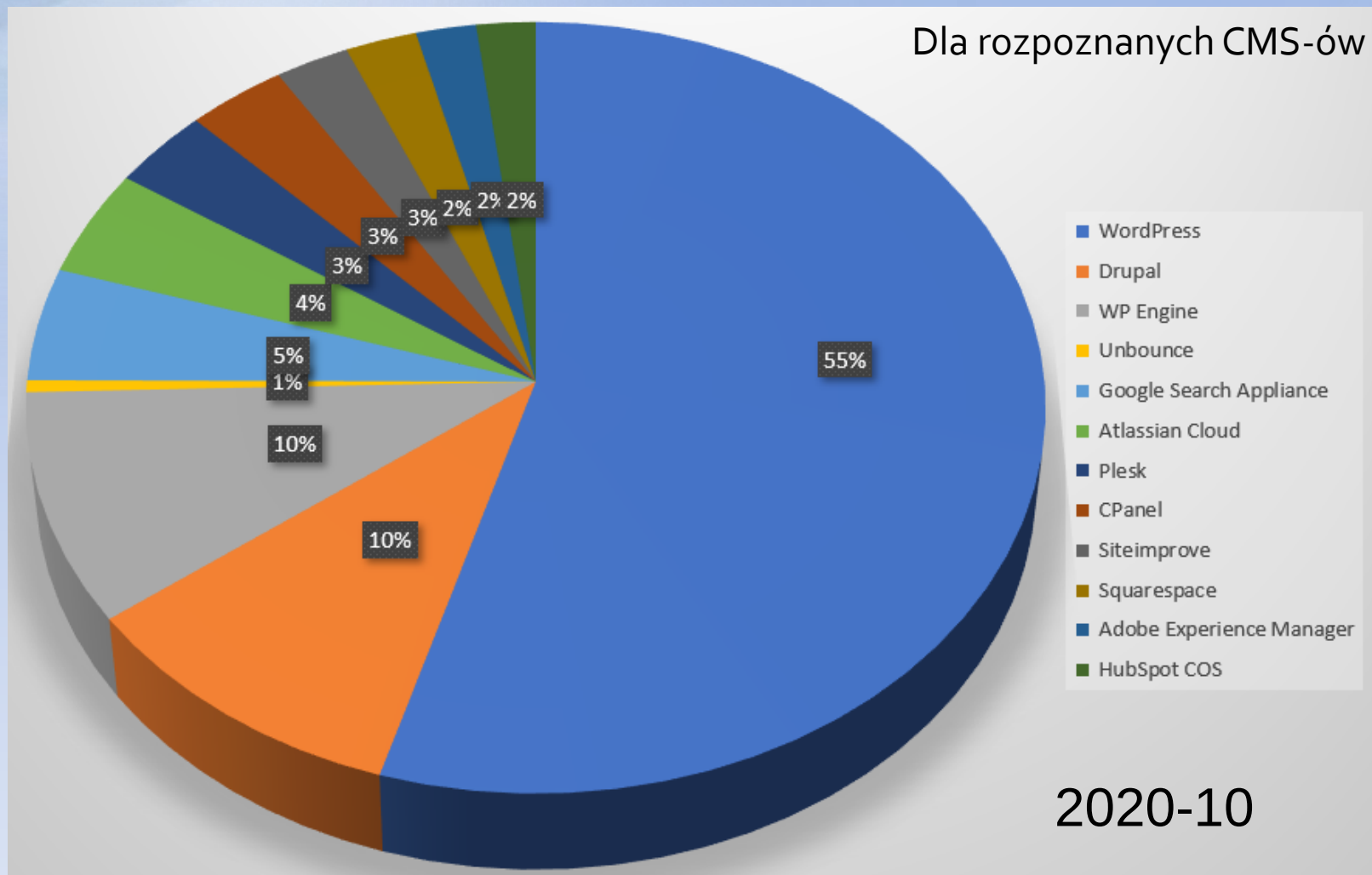
Udziały rynkowe dla 1 000 000 witryn



<http://trends.builtwith.com/cms>

Udziały rynkowe dla 100 000 witryn

Dla rozpoznanych CMS-ów

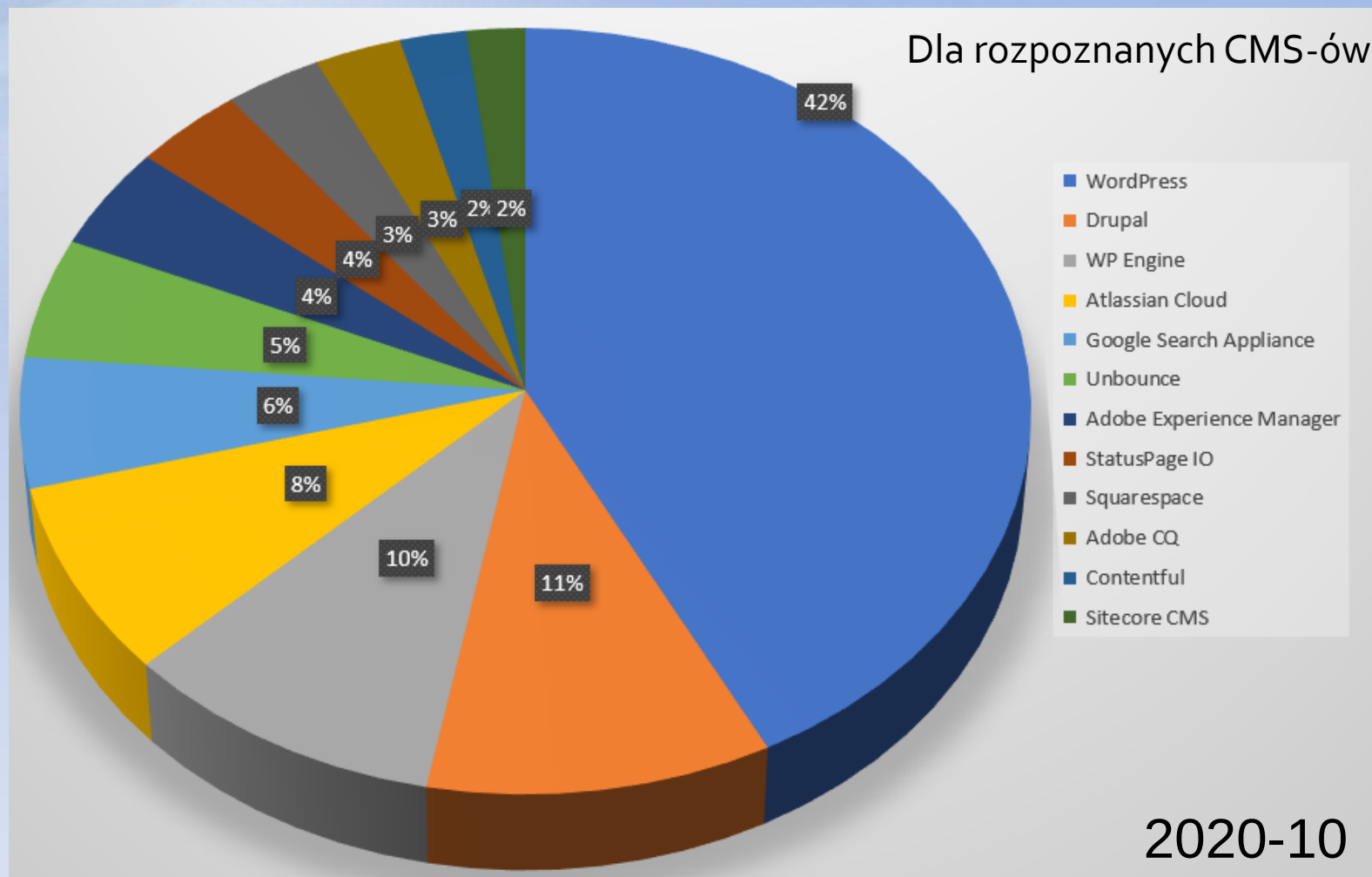


2020-10

Inne systemy: 61%

<http://trends.builtwith.com/cms>

Udziały rynkowe dla 10 000 witryn



Inne systemy: 65%

<http://trends.builtwith.com/cms>

Oferta usług dla poszczególnych CMS

CMS	Guru.com
WordPress	110 296
Joomla!	28 039
Drupal	16 560
Squarespace	1 647
Moodle	1 384
DotNetNuke	1 233
Umbraco	521
MODx	500
Liferay	352
TYPO3	246
Concrete5	205
Alfresco	201
Unbounce Landing Pages	187
Plone	119
Xoops	75
Movable Type	37

Uwagi

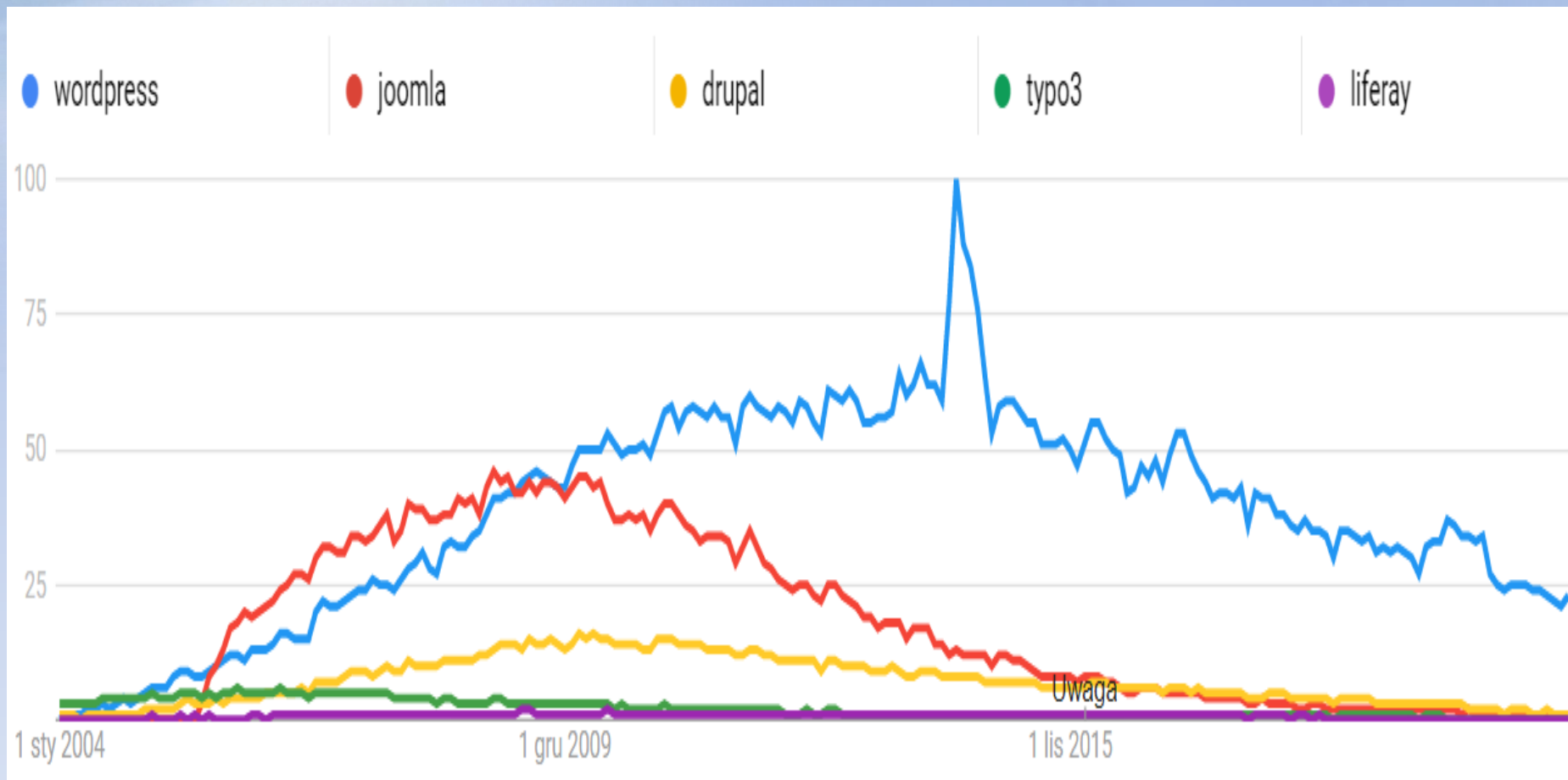
Guru (ponad 100 000 wykonawców) to witryna kojarzące zleceniodawców z wykonawcami (zwykle „wolni strzelcy”).

Zródło: opracowanie własne 2020-10

Popularność dla silników wyszukiwarek (Google, Yahoo!, Bing)

- Wyniki dla poszczególnych słów kluczowych:
 - *content management system*
 - *open source content management system*
 - *open source cms*
 - *cms*
 - *web cms*
 - *web content management system*
- Aktualnie ten sposób nie jest zbyt miarodajny, ponieważ:
 - często zwracane są wyniki artykułów dotyczących tej tematyki,
 - historia wyszukiwania wpływa na wyniki.

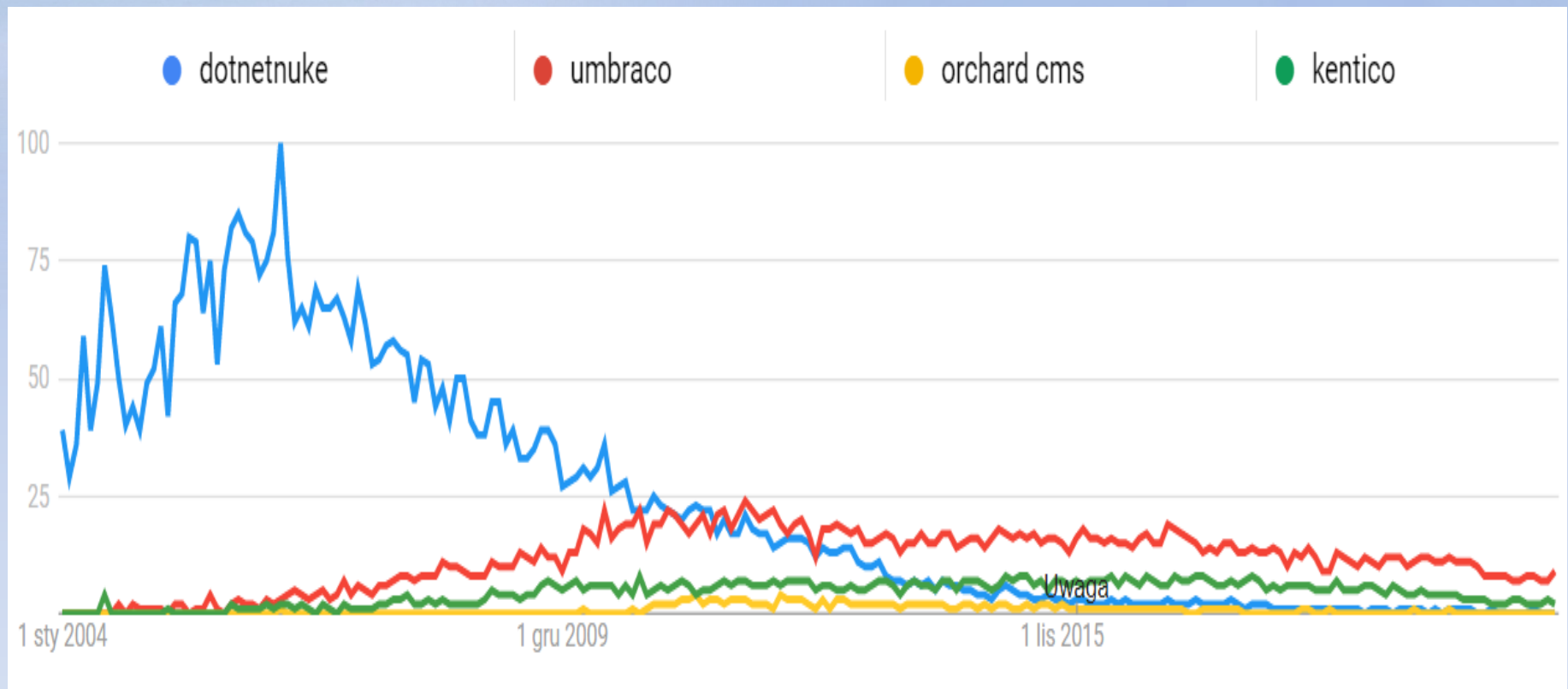
Popularność wg Google Trends



Popularność w [wyszukiwarce Google'a](#): 2004 - 2021

Popularność wg Google Trends (2)

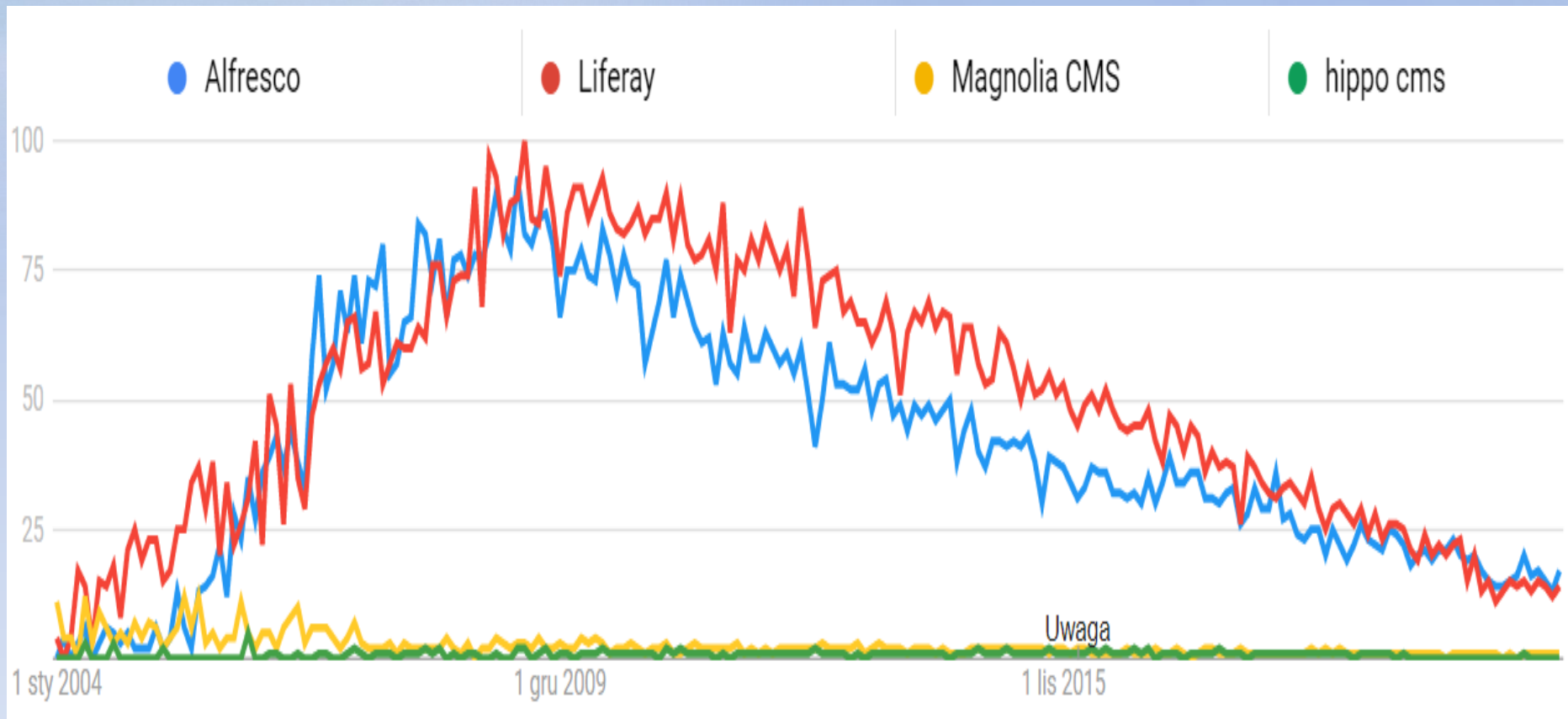
● Systemy oparte na .NET



Popularność w [wyszukiwarce Google'a: 2004 - 2021](#)

Popularność wg Google Trends (3)

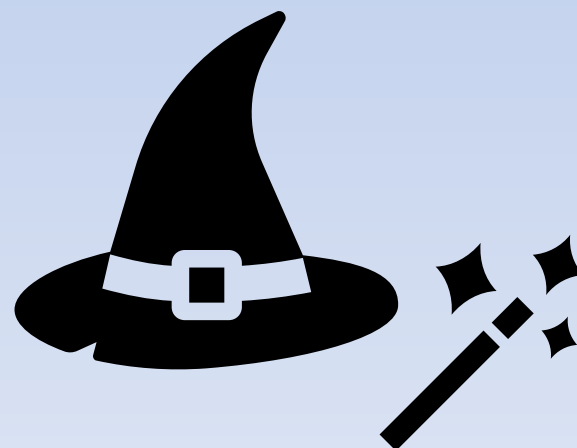
● Systemy oparte na Java



Popularność w [wyszukiwarce Google'a: 2004 - 2021](#)

Kreatory witryn internetowych

- Zwykle otrzymujemy dedykowany hosting;
- Przeważnie dość ograniczona funkcjonalność;
- Bardzo łatwa obsługa;
- Większość rozwiązań jest płatna (abonament), ale niektóre mają darmowy lub testowy plan;
- Popularne możliwości/funkcjonalności:
 - portfolio („wizytówka”),
 - prosty sklep internetowy,
 - formularz kontaktowy,
 - łatwe zmiany szablonów,
 - integracja z własną domeną,
 - wsparcie konsultantów.



Kreatory witryn internetowych (2)

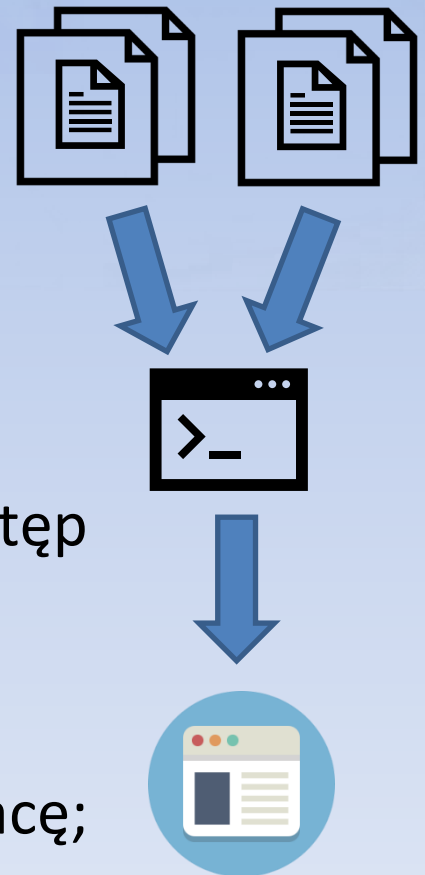
● Popularne rozwiązania

- [Squarespace](#). Płatny, ale ma wariant testowy;
- [WiX](#). Dość rozbudowany.
- [Weebly](#). Ukierunkowany na biznes.
- [Site123](#). Łatwy w obsłudze kreator;
- [Strikingly](#). Ukierunkowany na twórców (portfolio) ze wsparciem sprzedaży;
- [Jimdo](#). Dedykowany dla małych firm;
- [Webnode](#). Dedykowany dla małych firm;
- [Imcreator](#). Pośrednictwo w sprzedaży usług i towarów.

Więcej informacji: <https://www.websitebuilderexpert.com/website-builders/free/>

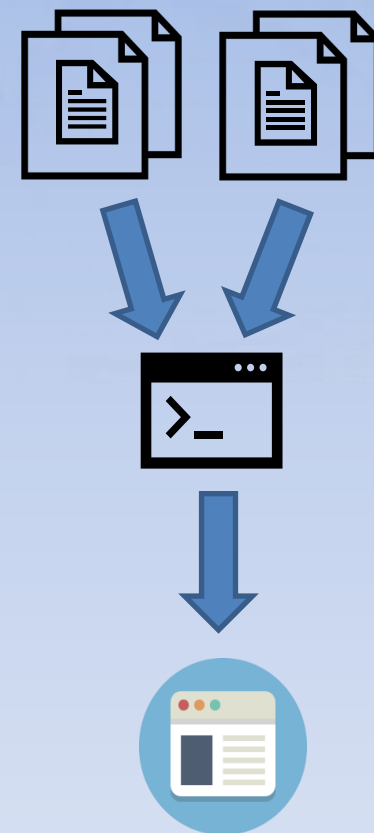
Nowe trendy: CMS-y plikowe (*Flat-file/file-based CMS, static site generators*)

- Nie takie całkiem nowe, bo początki www to właśnie statyczne strony HTML chociaż bardziej „prymitywne”.
- Cała zawartość portalu jest generowana przez administratora do postaci plików HTML/CSS/JavaScript;
- Następnie jest publikowana i dostępna dla odbiorców;
- Możliwość umieszczenia na CDN (*Content delivery network*) co znacząco przyspiesza dostęp do plików;
- Raczej przeznaczone dla zaawansowanych użytkowników (np. programistów);
- Czasami istnieją narzędzia GUI ułatwiające pracę;



Nowe trendy: CMS-y plikowe (2)

- W czasie konsumpcji (przeglądania), nie wymaga uruchamiania kodu, połączenia z bazą danych, itp.;
- Bardzo prosty (tani) hosting. Nie potrzeba uruchamiać kodu programu ani bazy danych.
- Stabilność (brak błędów programistycznych).
- Wersjonowanie treści za pomocą zmian w plikach (np. GitHub).
- Często całość określa się jako **JAM Stack** (**J**avascript – **A**PIs – **M**arkup).



Problemy dynamicznych stron

- Większa podatność na włamanie (wykonywanie kodu);
- Skalowalność wymaga m. in. zmiany serwera webowego i/lub bazodanowego;
- Pamięć podręczna nie zawsze jest idealnym rozwiązaniem (a i tak jest sporo wolniejsza od „płaskiego” pliku HTML);

Wykorzystano: [*Why Static Site Generators Are The Next Big Thing*](#)

Problemy dynamicznych stron (2)

- Wydajność: prosty eksperyment pokazuje, że „statyczne wygenerowanie” portalu (*Smashing Magazine*) i umieszczenie go na CDN, poprawiło wydajność o 600%;
- Wiarygodność/stałość url’i: system generując odpowiedź, może zwracać różną treść;
- Bardziej wymagający hosting pozwalający na uruchamianie kodu (PHP/Java/C#) po stronie serwera.

Wykorzystano: [Why Static Site Generators Are The Next Big Thing](#)

CMS-y plikowe

● Wady CMS-ów plikowych:

- tylko dla zaawansowanych użytkowników lubiących pracę z konsolą poleceń;
- problemy z funkcjonalnością wymagającą uruchamiania kodu; czasami można to obejść korzystając z usług zewnętrznych i/lub uruchamiając kod (JavaScript) na kliencie, np. sortowanie treści według daty lub formularz kontaktowy;

Wykorzystano: [*Why Static Site Generators Are The Next Big Thing*](#)

CMS-y plikowe (2)

- Wady CMS-ów plikowych – c. d.:
 - utrudnione przeszukiwanie portalu przez użytkowników (brak uruchamiania kodu i BD);
 - utrudniona edycja treści; czasami można skorzystać z edytora *on-line* w repozytorium plików z treścią (np. GitHub)
 - proces generowania może być dość długi (problem z małą zmianą „na szybko”), np. gdy trzeba aktualizować linki w całym portalu;

Wykorzystano: [Why Static Site Generators Are The Next Big Thing](#)

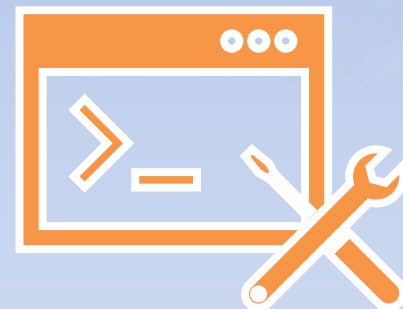
CMS-y plikowe - budowa

- Silnik szablonów umożliwiający opracowanie spójnej struktury portalu; czasami posiadają również wsparcie dla języków programowania;
- Dość rozbudowana hierarchia plików/katalogów;
- Popularne stosowanie składni *Markdown* (np. platforma GitHub);
- Specyficzne definiowanie metadanych, np. na początku dokumentu jako zwykły tekst;

CMS-y plikowe – budowa (2)

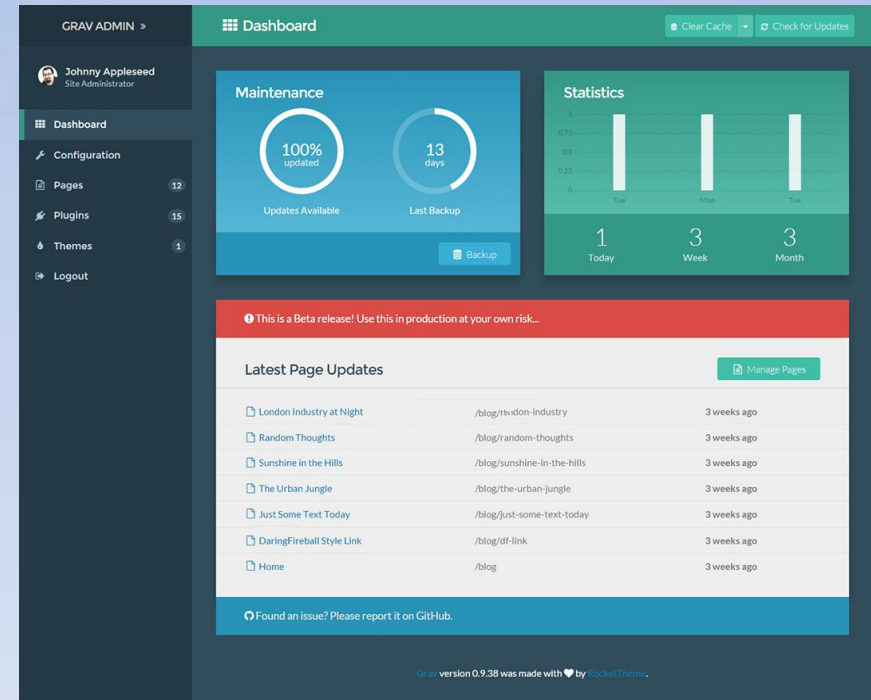
● Rozbudowany zestaw narzędzi (konsola):

- kompilacja zasobów;
- transpilacja (*transpiling*; np. TypeScript → JavaScript);
- minifikacja;
- *bundling*;
- często wykorzystywane są istniejące pakiety służące do przetwarzania/generowania plików, np. Grunt, Gulp, Broccoli.



Popularne CMS-y plikowe (*Flat-file/file-based CMS, static site generators*)

- [Dropplets](#) (bezpłatny),
- [Grav](#) (bezpłatny),
- [Hugo](#) (bezpłatny),
- [Jekyll](#) (bezpłatny),
- [Kirby](#) (komercyjny),
- [Middleman](#) (bezpłatny),
- [Monstra](#) (bezpłatny),
- [Pico](#) (bezpłatny),
- [Publii](#) (bezpłatny),
- [Roots](#) (bezpłatny),
- [Statamic](#) (komercyjny).



Popularne CMS-y plikowe - przegląd

	Language	Framework Used	Focus	Templating
Hugo	Go	None	General	Hugo Specific
11ty	JavaScript	None	General	Multiple
Jekyll	Ruby	None	General	Liquid
Jigsaw	PHP	Laravel (Partial)	General	Laravel Blade
Middleman	Ruby	Rails (Partial)	General	Multiple
Scully	JavaScript	Angular	Application	Angular Based
Gatsby.js	JavaScript	React.js	CMS	JSX (React)
Gridsome	JavaScript	Vue.js	CMS	Vue.js
Next.js	JavaScript	React.js	Application	JSX (React)
Nuxt.js	JavaScript	Vue.js	Application	Vue.js Based
VuePress	JavaScript	Vue.js	Documentation	Markdown/Vue.js Based
Docusarus	JavaScript	React.js	Documentation	JSX (React)
Docsify	JavaScript	Vue.js	Documentation	Markdown/Vue.js Based
Gitbook	SaaS	N/A	Documentation	N/A
Sitesauce	SaaS	N/A	General	N/A
Statamic v3	PHP	Laravel	CMS	Antlers/Laravel Blade

Źródło: <https://www.ample.co/blog/questions-to-ask-before-choosing-a-static-site-generator>

Popularne CMS-y plikowe – przegląd (2)

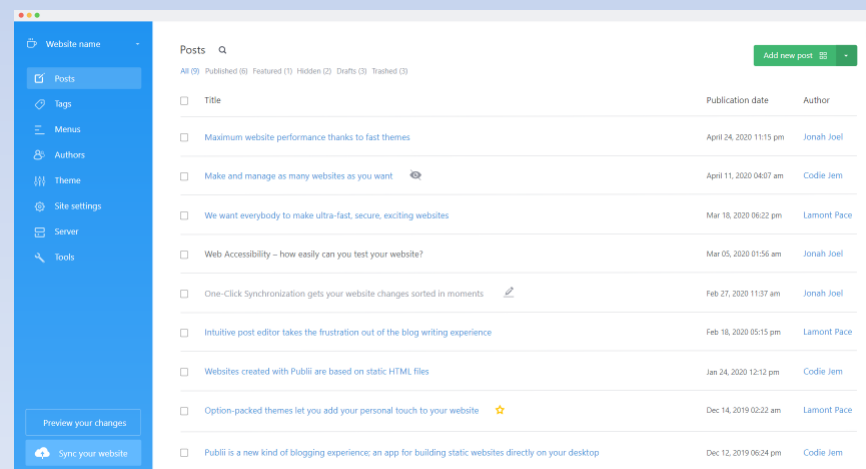
	Hybrid (Sever Side Rendering+SSG)	Data Layer	Default Markdown Flavor	Image Optimization/Transformation
Hugo	✗	Any	Commonmark	✓
11ty	✗	Any	Bring Your Own	✗
Jekyll	✗	Any	Kramdown	✗
Jigsaw	✗	Any	Unspecified	✗
Middleman	✗	Any	Unspecified	✓
Scully	✗	Any	Bring Your Own	✗
Gatsby.js	✓	GraphQL	Unspecified	✓
Gridsome	✗	GraphQL	Unspecified	✓
Next.js	✓	Any	Bring Your Own	✗
Nuxt.js	✗	Any	Bring Your Own	✗
VuePress	✗	Any	Unspecified	✗
Docusarus	✗	Any	GitHub Flavored Markdown	✗
Docsify	✗	Any	Unspecified	✗
Gitbook	✗	Any	Unspecified	✗
Sitesauce	✗	Any	N/A	✗
Statamic v3	✓	Any	Unspecified	✓

Źródło: <https://www.ample.co/blog/questions-to-ask-before-choosing-a-static-site-generator>

Popularny generator stron: Publii

<https://getpublii.com/>

- Dedykowana desktopowa aplikacja (MacOS, Windows, Linux) do wygodnej edycji i zarządzania;
- Wykorzystuje [Handlebars](#) jako język szablonów;
- Lokalne przechowywanie treści w bazie danych SQLite;
- Wygodna synchronizacja/publikacja wygenerowanych stron (HTML5/CSS3/JS);
- Opcjonalne generowanie AMP (*Accelerated Mobile Pages*).
- Dobra dokumentacja;
- Duża elastyczność;
- Projekt w fazie rozwojowej; aktualnie brak np. wtyczek.



Popularny CMS plikowy: Gatsby

<https://www.gatsbyjs.com/>

- Wykorzystuje Node.js oraz React.js;
- Pobieranie danych z różnych źródeł przy użyciu GraphQL;
- Generowane są *Progressive Web Applications* (PWAs).
- Duża liczba wtyczek (2000+) oraz rozbudowana społeczność;
- Dobra dokumentacja;
- Początki nauki mogą być trudne;
- Niezbyt duża szybkość generowania stron.

Wykorzystano: [Static Site Generators Overview](#)

Popularny CMS plikowy: Hugo

<https://gohugo.io/>

- Napisany w języku Go;
- Szybkie generowanie stron;
- Używa szablonów z języka Go;
- Bardzo łatwa konfiguracja;
- Dużo wbudowanej funkcjonalności;
- Dobra dokumentacja;
- Brak wsparcia dla XML; używa YAML, JSON oraz CSV.

Wykorzystano: [Static Site Generators Overview](#)

Popularny CMS plikowy: Jekyll

<https://jekyllrb.com/>

- Napisany w języku Ruby;
- Łatwy i bezpłatny hosting na GitHub ([GitHub Pages](#));
- Wbudowany serwer www ułatwiający pracę;
- Wykorzystuje *Markdown* do tworzenia treści, szablony Liquid oraz preprocesor Sass (CSS);
- Bardzo dużo wtyczek (na poziomie generatora) i pomocna społeczność;

Wykorzystano: [Static Site Generators Overview](#)

Popularny CMS plikowy: Jekyll (2)

<https://jekyllrb.com/>

- Łatwa konfiguracja;
- Dobra dokumentacja;
- Dobre wsparcie dla SEO;
- Dość wolne generowanie stron;
- Dodatkowe narzędzia GUI ułatwiające zarządzanie;
- Przykładowa witryna do samodzielnego eksperymentowania, np. Justice ([GitHub](#)).

Wykorzystano: [Static Site Generators Overview](#)

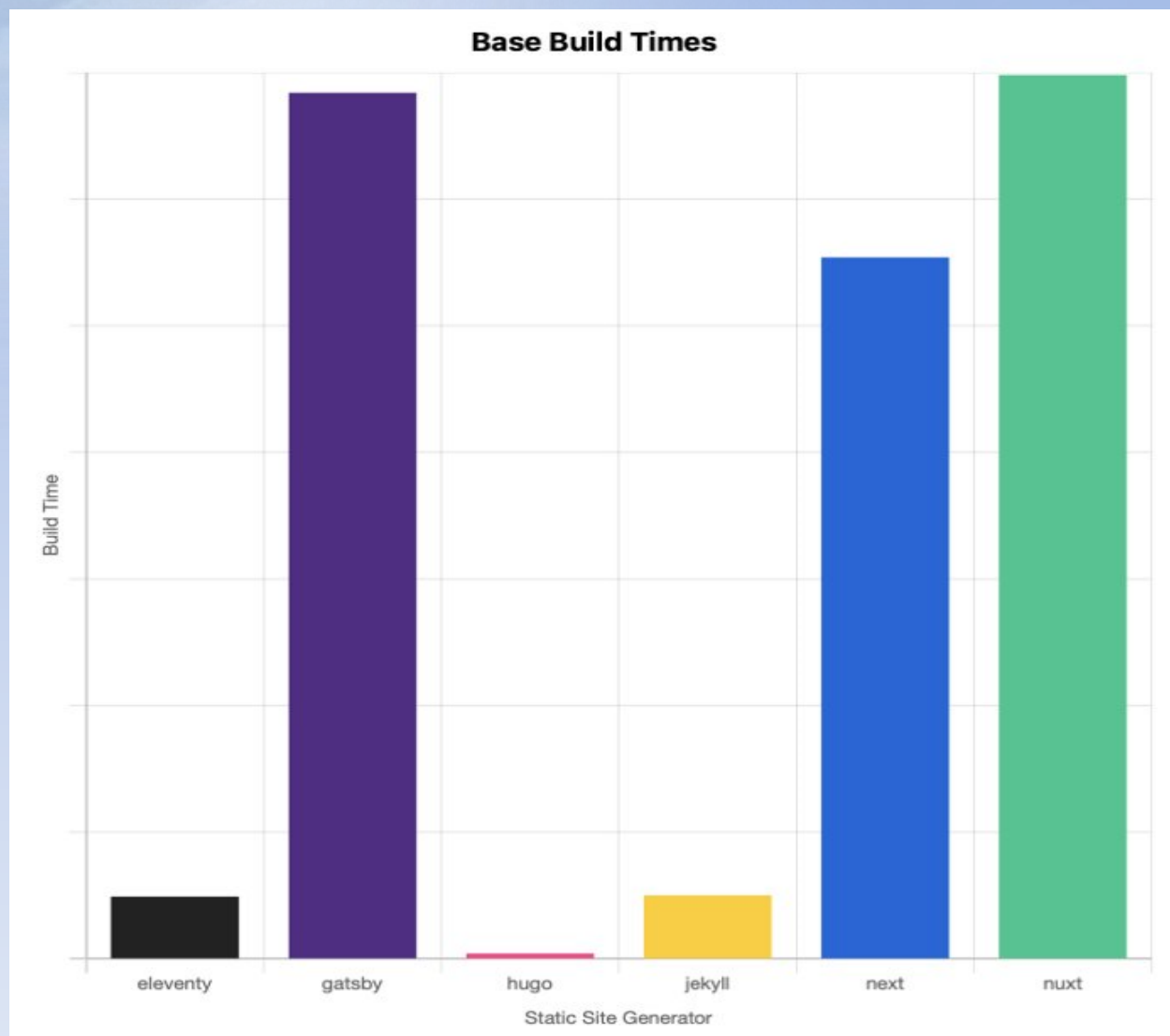
CMS-y plikowe: wydajność

- Różne technologie (m.in. Ruby, Go, JavaScript) wpływają na wydajność.
- Procedura testowa:
 - czas przetwarzania/generowania **pojedynczego pliku**;
 - **mała witryna** zawierająca od 1 – 1024 pliki; przy każdej generacji podwajano liczbę plików: 1, 2, 4, 8, 16, 32,..., 1024;
 - **duża witryna** zawierająca od 1000 – 64000 plików; przy każdej generacji podwajano liczbę plików: 1k, 2k, 4k, 8k, 32k, 64k.

Źródło: <https://css-tricks.com/comparing-static-site-generator-build-times/>

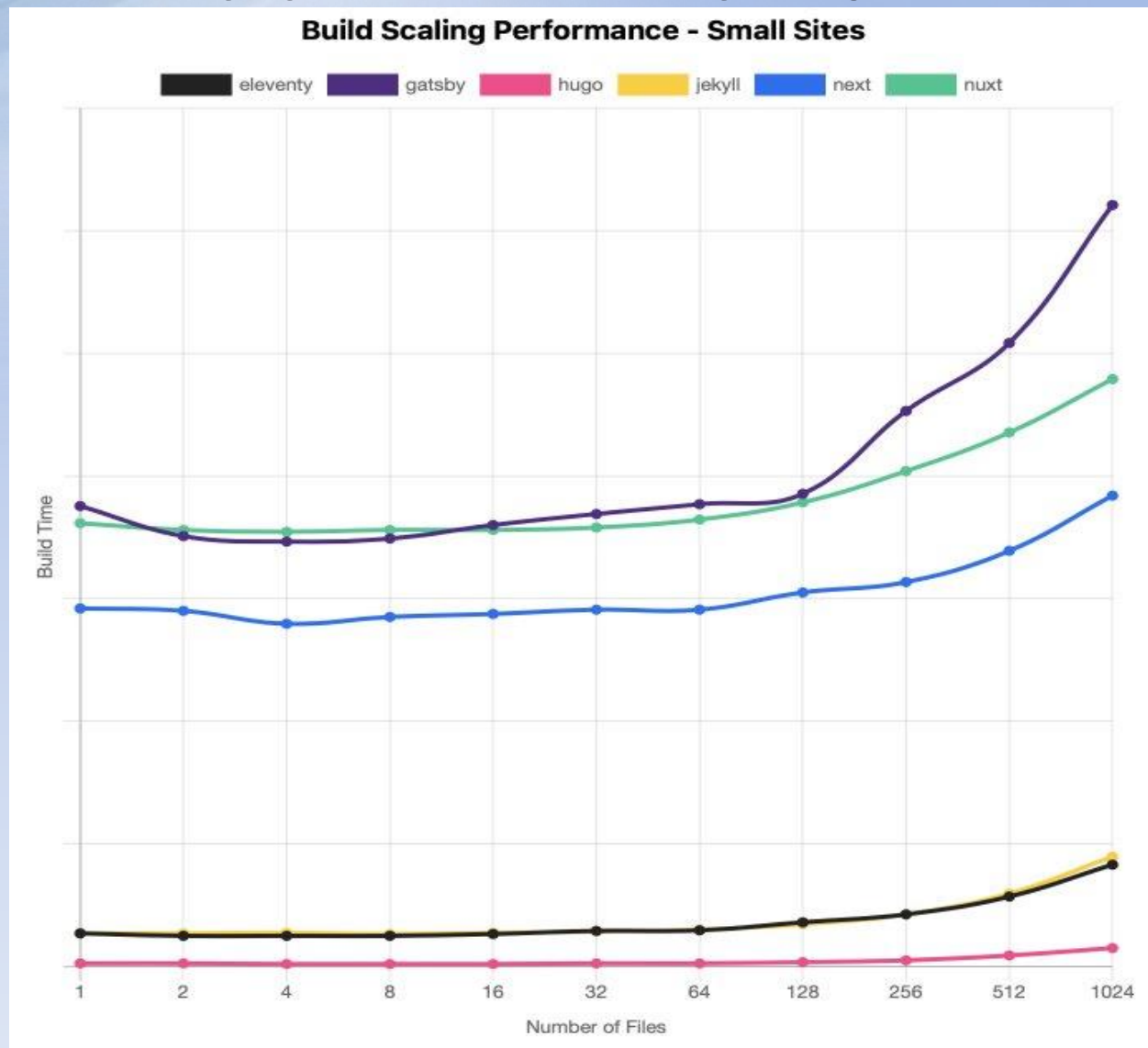
CMS-y plikowe: wydajność (2)

Źródło: <https://css-tricks.com/comparing-static-site-generator-build-times/>



Mniej = lepiej

CMS-y plikowe: wydajność (3)

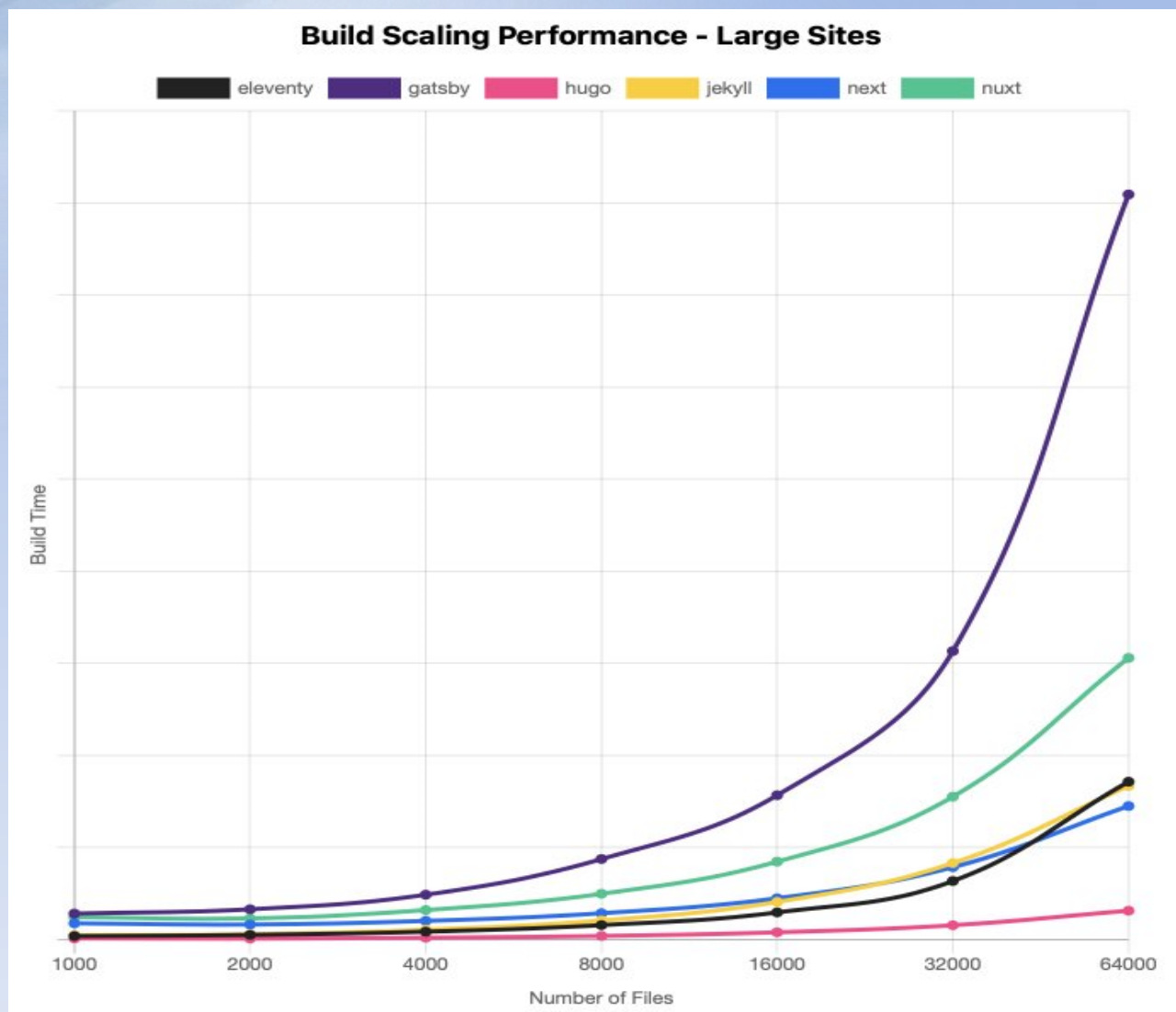


Źródło: <https://css-tricks.com/comparing-static-site-generator-build-times/>

Mniej = lepiej

CMS-y plikowe: wydajność (4)

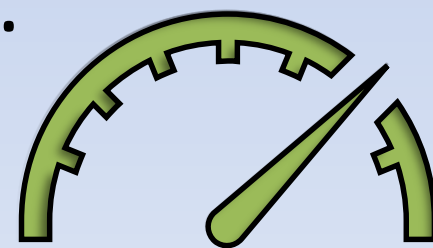
Źródło: <https://css-tricks.com/comparing-static-site-generator-build-times/>



Mniej = lepiej

CMS-y plikowe: wydajność (5)

- Najszybszy był Hugo, bez względu na liczbę plików do przetwarzania;
- Żaden z badanych CMS-ów nie skaluje się liniowo;
- Wolniejsze CMS-y mają większe możliwości, np. więcej wtyczek.



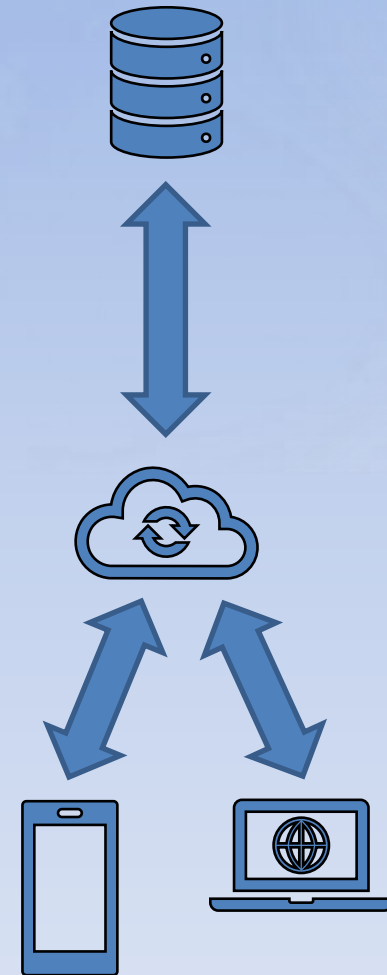
Źródło: <https://css-tricks.com/comparing-static-site-generator-build-times/>

CMS-y plikowe – więcej informacji

- [Mike Neumegen: Jekyll For WordPress Developers](#). Smashing Magazine.
- [Stefan Baumgartner: Using A Static Site Generator At Scale: Lessons Learned](#)
- [Why Static Website Generators Are The Next Big Thing](#)
- [Mathias Biilmann Christensen: Static Website Generators Reviewed: Jekyll, Middleman, Roots, Hugo](#).

Nowe trendy: CMS-y bezgłowe (*Headless/decoupled/API first CMS*)

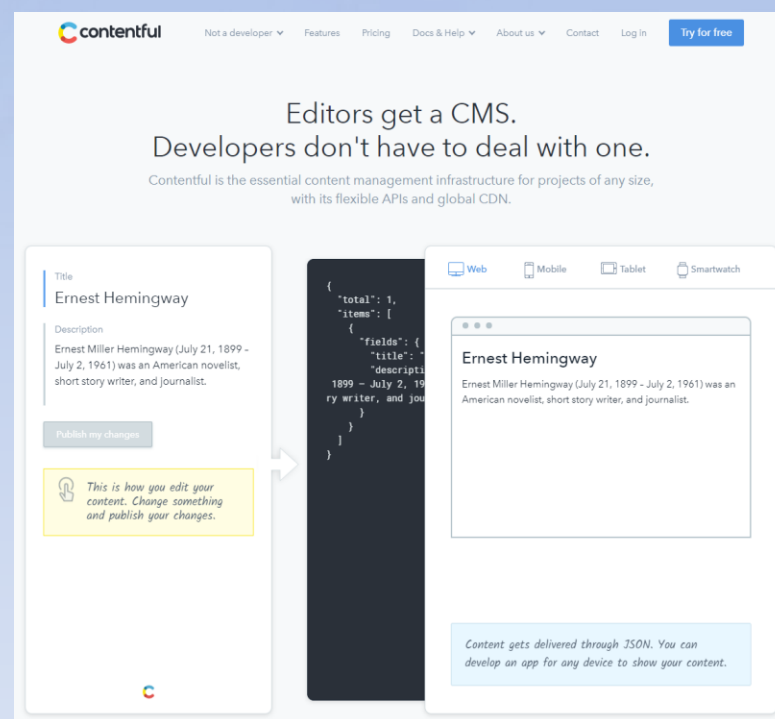
- Podobne do klasycznych CMS-ów, ale nie mamy wbudowanego GUI dla odbiorcy;
- Czasami jest GUI dla administratorów (np. CRUD) do zarządzania i tworzenia treści;
- Możliwość wykonywania operacji za pomocą REST API;
- Do wyświetlania treści dla odbiorców, wykorzystywany jest klient, np. aplikacja JavaScript lub natywna mobilna (np. Android).



Popularne CMS-y bezgłowe

- [Built.io \(Contentstack\)](#),
- [ButterCMS](#),
- [Cloud CMS](#),
- [Contentful](#),
- [Directus](#),
- [DNN Evoq 9](#),
- [dotCMS](#),
- [Elemeno](#),
- [GraphCMS](#),
- [NetlifyCMS](#),
- [Prismic](#),
- [Squidex](#),

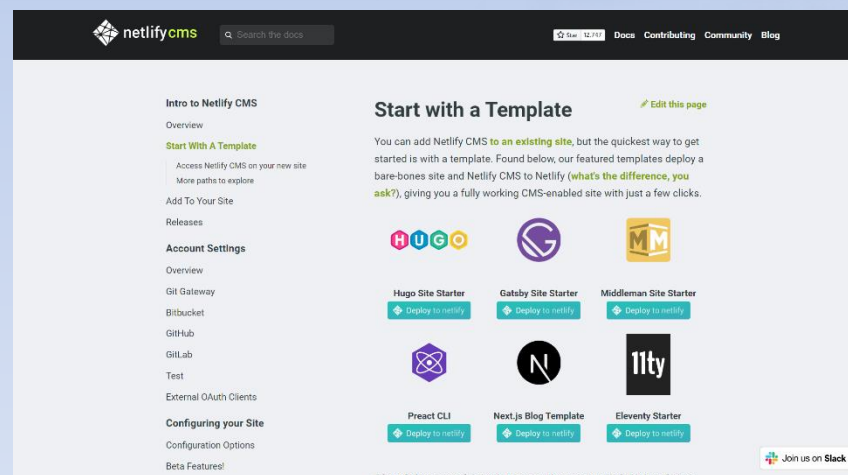
- [Strapi](#),
- [WordPress REST API](#)



Więcej informacji:
[34 Headless CMSs to Put on Your Radar](#)

Popularny CMS bezgłowy: NetlifyCMS

- Otwarte źródła,
- Opcjonalny hosting w Chmurze Netlify.com,
- Technologia JavaScript z React,
- Możliwość współpracy ze statycznymi generatorami, m.in. Hugo, Middleman oraz Gatsby,
- Wsparcie do traktowania GitHub jako repozytorium treści,

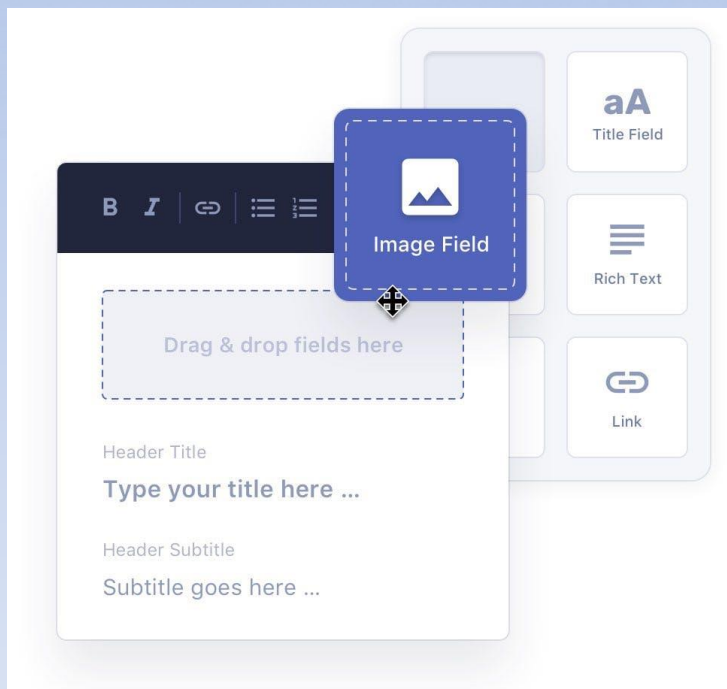


Popularny CMS bezgłowy: NetlifyCMS (2)

- NetlifyCMS może być traktowany jako narzędzie umożliwiające edycję treści (m.in. składnia *Markdown*) i zapis jej do repozytorium;
- Przykładowy sposób pracy (*workflow*):
 - edycja treści (np. artykuł) w prostym GUI zapewnianym przez NetlifyCMS (*backend*),
 - zapisanie ww. treści w repozytorium GitHub,
 - automatyczne przebudowanie i opublikowanie witryny przez np. Gatsby (jako statyczne pliki HTML + CSS + JavaScript).

Popularny CMS bezgłowy: Prismic

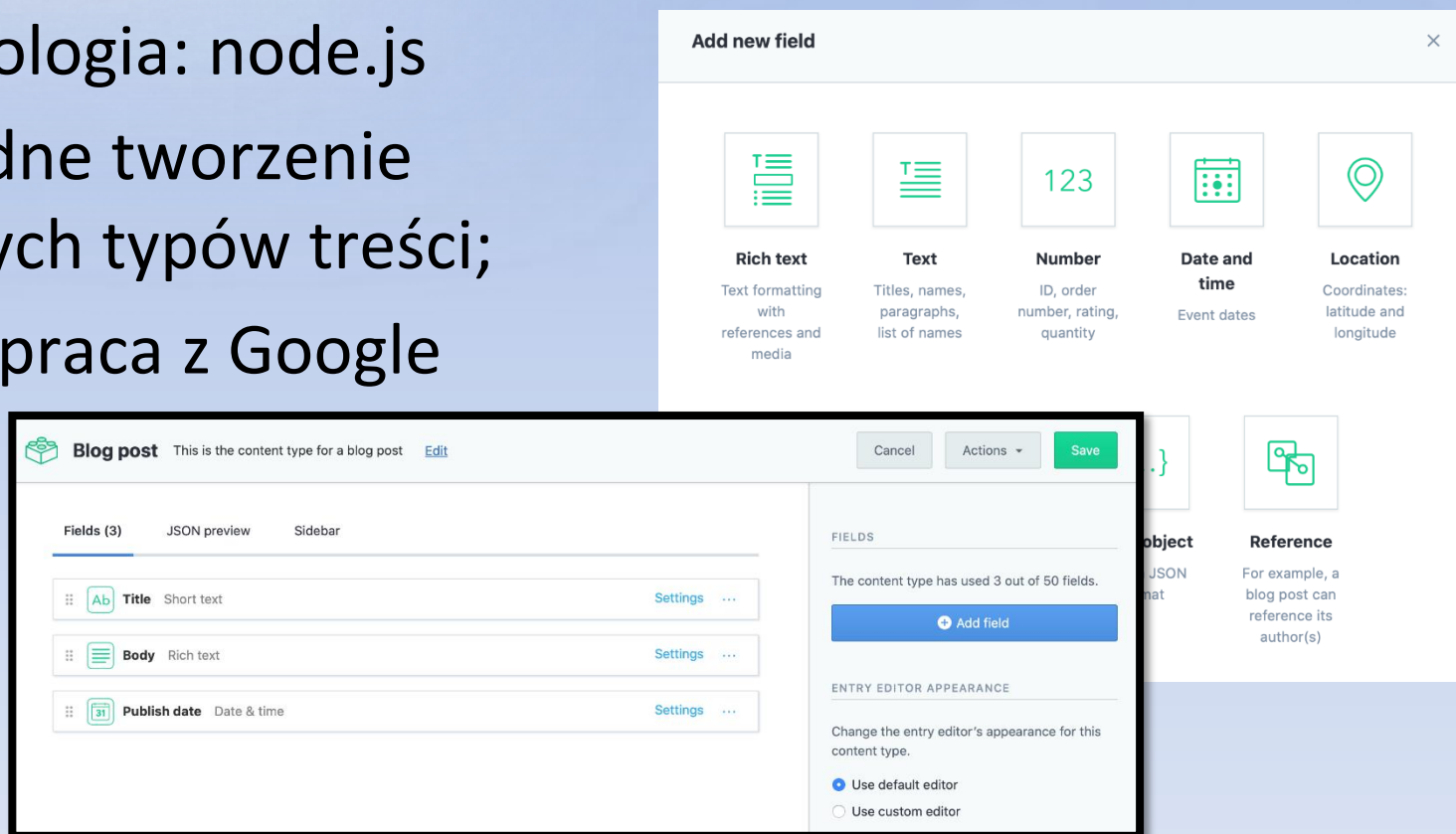
- Modelowanie własnych typów treści z możliwością ich ponownego użycia,



- Wsparcie dla wielu bibliotek, m.in. Gatsby, React, Next, Vue,
- Komunikacja za pomocą standardowego REST API oraz GraphQL,
- Opcjonalny hosting z globalnymi serwerami CDN,
- Dedykowana optymalizacja dla obrazków ([Imgix](https://imgix.com/)).

Popularny CMS bezgłowy: Contentful

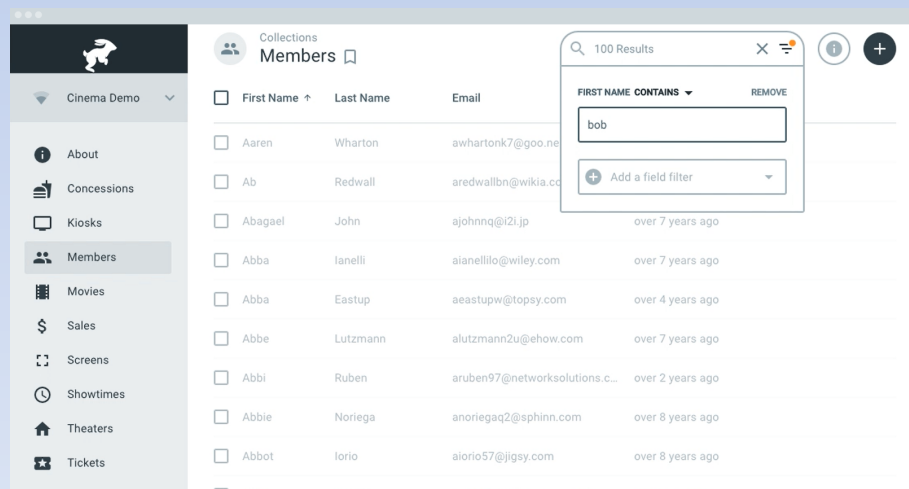
- Wersja bezpłatna;
- Technologia: node.js
- Wygodne tworzenie własnych typów treści;
- Współpraca z Google Docs;
- Wtyczki (HTML + CSS + JS)



Źródło: <https://www.contentful.com/help/contentful-101/>

Popularny CMS bezgłowy: Directus

- Bezpłatny
- Technologia PHP oraz backbone.js;
- Wygodny edytor CRUD;
- Modularny i rozbudowywalny;
- Przechowywanie plików;
- Łatwe komentowanie treści;
- Śledzenie i anulowanie zmian;
- Wsparcie dla wielu języków treści.



Źródło: [10 Most Popular Headless CMS](#), [11 Headless CMS to Consider](#)

Podsumowanie

- Ogromny wybór różnych systemów CMS.
- Przeważnie można znaleźć system spełniający większość oczekiwań.
- Możliwość dokonywania zmian dzięki otwartości kodu (*open source*).
- Warto również rozważyć nowe/alternatywne podejścia.