

Ochutnejte technické SEO

Matěj Velička, Richard Klačko, David Brenner

Taste

OBSAH

O autorech	<u>3</u>
Předmluva	<u>5</u>
1. Průchodnost a indexace webu	<u>7</u>
1.1. Soubor sitemap.xml	<u>7</u>
1.2. Soubor robots.txt	<u>13</u>
1.3. Crawl budget	<u>15</u>
1.4. Interní prolinkování webu	<u>18</u>
1.5. Stránkování obsahu	<u>21</u>
1.6. Přesměrování stránek webu	<u>22</u>
1.7. Duplicitní stránky	<u>25</u>
1.8. Procházení a indexace AJAX webů	<u>26</u>
1.9. Nastavení vícejazyčných stránek a webů cílících na různé země	<u>28</u>
2. Zabezpečení webu (protokol HTTPS)	<u>32</u>
2.1. HTTPS a vyhledávač Seznam	<u>32</u>
2.2. Přejít na HTTPS protokol	<u>33</u>
3. Rychlost načítání	<u>35</u>
3.1. Co všechno má na rychlost načítání vliv?	<u>35</u>
3.2. Jak rychlost načítání měřit?	<u>36</u>
3.3. Jak dobu načítání zlepšit?	<u>37</u>
4. Mobilní použitelnost	<u>44</u>
4.1. Základní pravidla pro „mobile friendly“ web	<u>44</u>
4.2. Mobilní verze webu	<u>45</u>
4.3. Mobile-first indexing	<u>46</u>
4.4. Accelerated Mobile Pages – AMP	<u>47</u>
5. Rozšířené výsledky vyhledávání	<u>53</u>
5.1. Jaké jsou výhody rozšířených výsledků vyhledávání?	<u>54</u>
5.2. Typy rozšířených výsledků vyhledávání v SERPu	<u>54</u>
5.3. Jak dostat stránku do rozšířených výsledků vyhledávání?	<u>64</u>
5.4. Jak ze svých snippetů udělám Rich Snippets/Rich Cards?	<u>64</u>
5.5. Jak strukturovaná data ověřit?	<u>71</u>
Závěr	<u>72</u>

O AUTORECH



Matěj Velička

Svou praxi online marketéra započal roku 2010 v jednom z největších SEO týmů v České republice, a to pod střechou agentury H1.cz. Zprvu se specializoval výhradně na off-page faktory webu, zejména na linkbuilding. Později jej však zaujaly zbožíové vyhledávače a optimalizace feedů e-shopů pro fulltextové výsledky vyhledávání natolik, až se postupně

vypracoval na post SEO konzultanta. O více než 3 roky později přešel do agentury Sun Marketing, kde postavil základy pro SEO oddělení a získal přehled o pokročilejších technikách optimalizace. V současné době má na starost primárně rozvoj oddělení spolu s implementací novinek z oboru do každodenní práce SEO specialistů, dohled nad plynulým chodem zakázek a plnění stanovených strategií.

Richard Klačko

K optimalizaci webů pro vyhledávače se dostal v roce 2015. Tehdy pracoval jako in-house SEO specialista pro mezinárodní společnost Alensa, kde zastřešoval desítky zahraničních e-shopů pro více než 10 evropských zemí. Dále se zde staral o SEO „best practice“ techniky, na nichž spolupracoval s týmem online marketingových specialistů jednotlivých států.

Koncem roku 2016 se Richard stal SEO specialistou v agentuře Sun Marketing. Rychle se osvědčil jako zkušený odborník schopný operativně řešit problémy, efektivně plánovat a nabízet inovativní přístupy. Nezajímá se však pouze o SEO. Online marketing jej baví a zajímá jako celek. Svůj volný čas rád věnuje přátelům nebo rodině a nepohrdne ani točeným pivem, ani dobrým jídlem.





David Brenner

Se SEO začínal, aniž by o tom věděl, při svém studiu na Přírodovědecké Fakultě (2007-2012), kdy vytvořil web Studiumchemie.cz, který dodnes slouží jako podpůrný nástroj pro učitele chemie. Po VŠ následovalo krátké pracovní angažmá na pozici HTML kodéra, kde získal lepší vhléd do programování a kódování webových stránek. Od roku 2013 působil ve společnosti Centrum

Holdings (dnes Economia) na pozici SEO specialisty, během toho také začal vytvářet weby a poskytovat poradenství související se SEO na vlastní živnost. V roce 2016 nastoupil na pozici SEO specialisty do Sun Marketingu. Zde si kromě práce na zajímavých klientech například vyzkoušel, jaké je vést workshop pro studenty Digisemestru nebo přednášet na konferenci SEO Restart. V současnosti pracuje jako SEO specialista ve společnosti Seznam.cz. Kromě SEO, programování a všeho co se točí kolem webů, ho baví hlavně navštěvování koncertů a kina, sportování (badminton, fotbal, běh, běžky...) a kibicování politiky.

O agentuře Taste

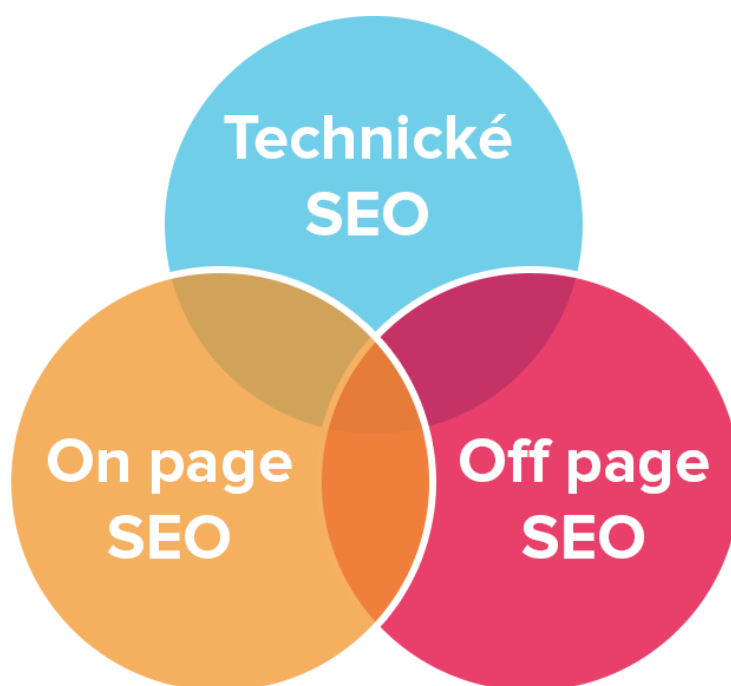
[Digitální agentura Taste](#) vznikla koncem roku 2017 spojením online marketingové agentury Sun Marketing, analytické a konzultační společnosti Medio Interactive a kreativní agentury Digistory. Taste u každého klienta usiluje o smysluplné propojení pečlivě promyšleného cross-channel mixu. Co to znamená? Díky službám sdružených společností, jež se navzájem doplňují, nabízí klientům komplexní řešení ve všech oblastech online marketingu. V rámci Sun Marketingu dokáže pokrýt výkonnostní online kampaně a jejich optimalizaci, Medio si zase poradí s datovou analytikou a User Experience, kdežto Digistory nabídne kreativní zpracování na poli obsahového marketingu, copywritingu i grafického designu.

Publikace Ochutnejte technické SEO vznikla pod křídly Sun Marketingu, který od svého založení v roce 2006 ušel kus cesty. Z agentury zaměřené na PPC kampaně se v průběhu desetiletí stala autorita na poli výkonnostního online marketingu, jehož je problematika SEO nedílnou součástí. Ukrojte si část know-how Sun Marketingu a vychutnejte si požitek z úspěšných webových stránek.



PŘEDMLUVA

Optimalizace pro vyhledávače (SEO) se již od počátku dělila na On-page a Off-page optimalizaci. **On-page SEO** stojí zejména na optimalizaci obsahu a jeho architektuře. V **Off-page SEO** jde především o autoritu webových stránek, tedy o získávání kvalitních a relevantních zpětných odkazů (linkbuilding) a pravděpodobně také o signály ze sociálních sítí. Postupem času nabrala na významu další disciplína, která se se dvěma předchozími často doplňuje a prolíná. Je jí **technické SEO**. I když ve své podstatě spadá technické SEO do množiny on-page faktorů, v této publikaci jej rozebereme jako samostatnou oblast.



Proč se technické SEO dostalo v posledních letech do popředí zájmu? Důvodů existuje více. Například se výrazně **zvýšil podíl mobilních zařízení na internetovém provozu**, což s sebou nese také vyšší nároky na rychlost načítání webových stránek. Na mobilech a tabletech je navíc často třeba jiného uspořádání informací, které se zde zobrazují na výrazně menším prostoru. Také paleta **technologií používaných ke tvorbě webových stránek se hodně změnila**. Velká část webů dnes běží na JavaScriptových frameworkcích, jako jsou Angular nebo React, což je podstatné zejména z pohledu **průchodnosti a indexovatelnosti** takovýchto webů roboty vyhledávačů (crawley). Rozmohl se trend „single page webů“. V optimalizaci hraje rovněž svou roli UX či zabezpečení webu. A konečně, vyhledávače jako Google nebo Seznam.cz provedly v posledních letech výrazné úpravy svých algoritmů. Již nevyhledávají pouze fulltextem, nýbrž se snaží dotazům také lépe porozumět a rovnou na ně odpovídat. Například Google si k tomuto účelu pomáhá algoritmem RankBrain, který využívá strojového učení (Machine Learning).



Publikace je strukturou řazena do pěti ucelených částí. První část tvoří **Průchodnost a indexace webu**. Další kategorií je **Zabezpečení webu**, kde se zaměřujeme na protokol HTTPS. Třetí kapitolou je **Rychlost načítání** stránek webu, dále **Mobilní použitelnost** a poslední kapitola je část **Rozšířené výsledky vyhledávání**. Ta ovšem spadá do publikace **Technické SEO** jen částečně. Naším záměrem je spíše poukázat na možnosti, které Google SERP skrývá.

Poslední kapitolu jsme zařadili do publikace technické SEO proto, že velmi úzce souvisí se strukturovanými daty. Jsme přesvědčeni o tom, že strukturovaná data jsou bohatým zdrojem informací pro vyhledávače, což může pomoci v lepší viditelnosti webu v rozšířených výsledcích vyhledávání. Proto jsme se rozhodli sepsat ty nejdůležitější, které se často vyskytují ve výsledcích vyhledávání v českém prostředí.

Uvědomujeme si, že v této publikaci není obsažena celá problematika, protože do technických faktorů, které ovlivňují vyhledávače, spadá nespočet činností. Do publikace jsme ale zařadili z našeho pohledu to nejzásadnější a nejdůležitější. Pokud si ale myslíte, že v této publikaci chybí nějaká zásadní část, oceníme vaši zpětnou vazbu pro případné doplnění.



1. PRŮCHODNOST A INDEXACE WEBU

Správná indexace webu je jednou ze základních podmínek úspěšné optimalizace webu pro vyhledávače. Jedině stránka uložená v indexu vyhledávače se může zobrazit ve výsledcích vyhledávání. Cílem je dostat do indexu každého vyhledávače co nejvíce obsahově důležitých stránek, respektive stránek, které chceme, aby byly dohledatelné ve vyhledávačích. Zároveň je však potřeba zajistit aby vyhledávače zbytečně neindexovaly stránky, které uživatelům, a ani vyhledávačům, nechceme předkládat. V tomto případě to můžou být například různé duplicity pocházející z filtrace, URL adresy s UTM parametry, stránky s nedůležitým obsahem, stránky ze stránkování, atd.

Počet zaindexovaných stránek zjistíme ve službě Google Search Console, případně u vyhledavače Seznam je to služba Webmaster. Další možností na zjištění orientačního počtu zaindexovaných stránek je využití operátoru site, tedy vložení příkazu site:example.com do vyhledávače. Počet je ale hrubý odhad a vůbec nemusí odpovídat skutečnosti.

U vyhledávače Seznam je nutné pro zobrazení počtu zaindexovaných stránek si zobrazit stránku výsledků vyhledávání ve formátu RSS. Uděláte to například tak, že do URL stránky s výsledky přidáte na konec parametr format s hodnotou rss (tedy „&format=rss“), například:

<https://search.seznam.cz/?q=site%3Aasunmarketing.cz&oq=site%3Aasunmarketing.cz&sourceid=szn-HP&sgld=&thru=&su=e&aq=&format=rss>

Aby byly vyhledávače schopny uložit všechny důležité stránky webu, musíte robotům vyhledávačů zajistit jejich hladký průchod webem. Jednoduše řečeno, je potřeba zařídit, aby byly stránky pro roboty dostupné, indexovatelné a lehce dohledatelné.

1.1. Soubor sitemap.xml

Soubory sitemap.xml umožňují informovat vyhledávače o URL adresách webu, které jsou k dispozici pro procházení a indexaci. Tyto soubory jsou vytvořeny ve formátu XML, ve kterém jsou strukturovaně uvedeny URL adresy podstránek webu s dalšími atributy. Tyto atributy umožňují předávat vyhledávačům další informace jako datum poslední aktualizace stránky (`lastmod`), jak často se mění (`changefreq`) a jakou má relativní důležitost vůči ostatnímu obsahu na webu (`priority`). Všechny tyto atributy jsou volitelné a nijak [neovlivňují pozice ve výsledcích vyhledávání](#). Jejich uvedením pouze umožníte vyhledávačům procházet web inteligentnějším způsobem a soustředit se na obsah, který je skutečně podstatný.



Kód souboru sitemap.xml může vypadat následovně. Zde je ukázka nejjednoduššího souboru sitemap.xml, kterou tvoří jeden odkaz:

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">
  <url>
    <loc>http://www.example.com/</loc>
    <lastmod>2017-09-01</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.8</priority>
  </url>
</urlset>
```

V XML kódu souboru sitemap.xml se mohou vyskytovat následující tagy:

- `<urlset>` - povinný tag, který obaluje celý kód a odkazuje se na referenční standard pro soubor sitemap.xml.
- `<url>` - povinný tag, který obaluje kód pro každou URL webu.
- `<loc>` - povinný tag, ve značkách `<loc>` se nachází URL adresy webu, které požadujete po robotovi projít a indexovat.
- `<lastmod>` - nepovinný tag, ve kterém vyhledávači říkáte, kdy byla stránka naposledy změněna, udává se ve formátu YYYY-MM-DD.
- `<changefreq>` - nepovinný tag, který udává, jak často se stránka v průměru mění, může nabývat několika předdefinovaných hodnot: always, hourly, daily, weekly, monthly, yearly, never.
- `<priority>` - nepovinný tag, který určuje prioritu dané podstránky webu, může nabývat hodnot od nuly do jedné, kdy 0.0 symbolizuje nejnižší prioritu a 1.0 nejvyšší.

Tagy `<changefreq>` a `<priority>` roboti vyhledávačů dost pravděpodobně úplně ignorují. Konkrétně pro Google to potvrdil [John Mueller v rozhovoru v roce 2015](#).

Soubory sitemap mohou být vygenerovány jednorázově různými aplikacemi, jako je například Xenu Linksleuth, [Screaming Frog SEO Spider](#) nebo různé sitemap generátory. Doporučuje se však, aby byly **generovány dynamicky** přímo na serveru. Praktické je to hlavně v tom, že v momentě, kdy vytvoříte novou podstránku webu, je tato podstránka automaticky do souboru sitemap.xml zahrnuta.

Na kontrolu správnosti souboru sitemap.xml existuje řada nástrojů. Nejsnazší je ovšem provést validaci přímo v nástroji Google Search Console.



Nejčastější chyby, které se v mapě stránek nacházejí:

- absence důležitých URL stránek, které chceme indexovat vyhledávači,
- výskyt URL stránek, které jsou přesměrované na jinou stránku,
- výskyt URL stránek, které vracejí stavový kód 404 nebo 410,
- výskyt URL stránek, které jsou kanonizované na jinou stránku,
- výskyt URL stránek s `meta robots` nebo `x-robots noindex`.

1.1.1 Komprese a rozdělení souboru sitemap.xml

Nezkomprimovaná velikost souboru sitemap.xml by podle [doporučení Google](#) i [Seznamu](#) neměla přesáhnout 50 MB nebo 50 tisíc URL. V případě, že je velikost větší, doporučují se obecně následující řešení:

- Soubor sitemap.xml rozdělit na více menších a z URL sitemap.xml na ně odkazovat (viz [sitemap index](#)).
- Zkomprimovat soubor sitemap.xml do formátu GZip.

Obě metody se dají kombinovat. Z jednoho sitemap indexu tak může vést odkaz na více souborů sitemap.xml zkomprimovaných metodou GZip.

1.1.2 Sitemap.xml pro obrázky, videa a zprávy

Vedle standardních souborů sitemap.xml pro podstránky vašeho webu můžete využít také soubory sitemap.xml pro typy dat:

- [sitemap.xml pro zprávy](#),
- [sitemap.xml pro obrázky](#),
- [sitemap.xml pro videa](#).

Tyto soubory obsahují oproti standardním souborům sitemap.xml jiné tagy (u obrázku je to například jeho URL adresa, popisek nebo titulek). Jejich přínos spočívá zejména v tom, že vyhledávači lépe popíšete, jaké typy obsahu se na webu nachází, a zároveň zvýšíte šance na zobrazení tohoto obsahu ve specifických výsledcích vyhledávání, jako je například Google Image Search.



1. Soubor sitemap.xml pro zprávy

Mapa stránek pro zprávy navíc používá tagy, které jsou specifické pro zprávy. Zápis zprávy v souboru sitemap.xml by mohl vypadat následovně:

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
        xmlns=http://www.google.com/schemas/sitemap-news/0.9">
  <url>
    <loc>http://www.example.com/marketing/zprava1</loc>
    <lastmod>2017-09-01</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.8</priority>
  <news:news>
    <news:publication>
      <news:name>Zpráva z marketingu 1</news:name>
      <news:language>cs</news:language>
    </news:publication>
    <news:genres>Novinka, Blog, Zpráva</news:genres>
    <news:publication_date>2017-09-01</news:publication_date>
    <news:title>Titulek zprávy z marketingu 1</news:title>
    <news:keywords>marketing, SEO, PPC, RTB, sociální
site</news:keywords>
      <news:stock_tickers>NASDAQ:A, NASDAQ:B</news:stock_tickers>
    </news:news>
  </url>
</urlset>
```

2. Soubor sitemap.xml pro obrázky

O obrázcích, které jsou na webu, můžete vyhledávač informovat i prostřednictvím souboru sitemap.xml. Nejen, že můžete přidat přímou cestu (URL) k obrázku, ale také můžete vyhledávačům říct o obrázcích doplňkové informace. Zápis obrázku v souboru sitemap.xml může mít následující podobu:

```
<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
        xmlns:image="http://www.google.com/schemas/sitemap-image/1.1">
  <url>
    <loc>http://www.example.com/</loc>
    <lastmod>2017-09-01</lastmod>
    <changefreq>monthly</changefreq>
    <priority>0.8</priority>
```



```

<image:image>
  <image:loc>http://www.example.com/obrazek1.jpg</image:loc>
  <image:caption>Toto je popis obrázku 1</image:caption>
  <image:geo_location>Praha, Česká
    republika</image:geo_location>
  <image:title>Toto je název obrázku 1</image:title>
  <image:license>URL adresa licence na obrázek</image:license>
</image:image>
<image:image>
  <image:loc>http://www.example.com/obrazek2.jpg</image:loc>
  .....
</image:image>
</url>
</urlset>

```

3. Soubor sitemap.xml pro video

Soubor sitemap.xml může vyhledávači sloužit jako skvělý informační kanál o veškerém videoobsahu na vašich stránkách. Navíc tak vyhledávači Google odešlete veškeré informace o videích, čímž Google umožní jejich dohledatelnost ve službě Videá Google. Zápis videa v souboru sitemap.xml může být následovný:

```

<?xml version="1.0" encoding="UTF-8"?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
  xmlns:video="http://www.google.com/schemas/sitemap-
  video/1.1">
<url>
  <loc>http://www.example.com/</loc>
  <lastmod>2017-09-01</lastmod>
  <changefreq>monthly</changefreq>
  <priority>0.8</priority>
  <video:video>
<video:thumbnail_loc>http://www.example.com/obrazek-nahled-
  video.jpg</video:thumbnail_loc>
<video:title>Název videa</video:title>
<video:description>Popisek videa </video:description>
<video:content_loc>http://www.example.com/video1.mp4</video:content_
  loc>
<video:player_loc
  autoplay="ap=1">http://www.example.com/videoplayer.mp4?video=1</video
  :player_loc>
<video:duration>180</video:duration>

```



```
<video:expiration_date>2019-12-
05T19:20:30+08:00</video:expiration_date>
<video:rating>4.5</video:rating>
<video:view_count>1000000</video:view_count>
<video:publication_date>2007-11-
05T19:20:30+08:00</video:publication_date>
<video:family_friendly>yes</video:family_friendly>
<video:restriction relationship="allow">IE GB CA</video:restriction>
<video:gallery_loc
title="Marketing">http://marketing.example.com</video:gallery_loc>
<video:price currency="CZK">10</video:price>
<video:requires_subscription>no</video:requires_subscription>
<video:uploader info="http://www.example.com/uzivatel/honza ">Honza
</video:uploader>
<video:platform relationship="allow">WEB MOBILE</video:platform>
<video:live>no</video:live>
  </video:video>
</url>
</urlset>
```

Vysvětlivky a povinnost/ volitelnost jednotlivých tagů naleznete přímo [v nápovědě o sitemap.xml pro videa od Google](#).

4. Nastavení vícejazyčných stránek a webů s různým geografickým cílením v souboru sitemap.xml

Pokud máte stránky ve více jazycích nebo ve stejném jazyce, ovšem cílící na odlišné země (trhy), je nutné o jednotlivých verzích webu říci vyhledávačům v souboru sitemap.xml. Jak správně nastavit jazykové a geografické cílení v souboru sitemap.xml je popsáno v kapitole [1.9.4](#).

1.1.3 Jak se o mých souborech sitemap.xml dozví vyhledávač?

Pokud je soubor sitemap.xml (s tímto přesným názvem) umístěn v kořenovém adresáři webu, pravděpodobně na něj robot vyhledávače narazí sám, protože je to standard. Podpořit přístup robotů k souboru můžete i dalšími způsoby:

- Umístit odkaz na mapu stránek do souboru robots.txt. Adresa stránky by měla být v absolutním tvaru, tedy:

Sitemap: `http://example.com/sitemap.xml`

nikoliv pak:



Sitemap: /sitemap.xml

- Pro Googlebota - nahrát soubor do nástroje [Google Search Console](#).

Alternativy pro informování vyhledávačů o souboru sitemap.xml jsou speciální stránky určené na vložení URL do vyhledávání: <https://www.google.com/webmasters/tools/submit-url>, <https://search.seznam.cz/pridej-stranku>, <https://www.bing.com/toolbox/submit-site-url>.

1.2. Soubor robots.txt

Robots.txt funguje opačným způsobem než sitemap.xml. Můžete v něm zamezit vstupu robotů vyhledávačů na určité části webu. Jedná se o globální standard, který je určen především pro roboty vyhledávačů, aby na webu neprováděly akce, které jsou nežádoucí. Správnou specifikací tak můžete zamezit procházení nedůležitých nebo duplicitních URL vašeho webu (např. URL s parametry apod.). Tímto můžete šetřit tzv. **crawl budget**, který má robot k dispozici pro váš web. O crawl budgetu píšeme více v další kapitole.

Tento soubor vyhledávače vždy načítají jako první, pak procházejí další stránky webu. Ne všichni roboti bohužel jeho obsah dodržují. Roboti, kteří jsou používáni pro získávání emailů, distribuci malware nebo spamboti se těmito doporučeními zpravidla neřídí.

Soubor robots.txt by měl být vždy pojmenován jako „robots“ a měl by být umístěn v kořenovém adresáři domény, tedy např. <http://www.example.com/robots.txt>.

V českém prostředí je důležité do souboru robots.txt vložit URL adresu souboru sitemap.xml (viz kapitola 1.1.3). Pokud adresu mapy stránek Seznambot v souboru robots.txt nenajde, údajně mapu stránek vůbec nenavštíví (tato informace padla ze strany Seznamu na konferenci SEO restart 2016). Kód souboru robots.txt může vypadat následovně:

```
User-agent: *  
Disallow: /wp-admin/  
Sitemap: http://www.example.com/sitemap.xml
```

Například tento kód říká, že roboti mohou web procházet bez omezení s výjimkou složky /wp-admin/ a jejích podadresářů. Obsah souboru, zda v něm nejsou chyby, lze ověřit v nástroji Google Search Console, přehled *Procházení > Nástroj na testování souborů robots.txt*.

Pokud bychom chtěli zamezit robotům vyhledávačů procházet adresář /wp-admin/, kromě stránky /wp-admin/navod-na-nastaveni-meta-dat.html, nastavení by bylo následující:



```
User-agent: *
Disallow: /wp-admin/
Allow: /wp-admin/navod-na-nastaveni-meta-dat.html
Sitemap: http://www.example.com/sitemap.xml
```

1.2.1 Zástupné znaky

Při definování souboru robots.txt lze využívat také zástupné znaky hvězdičku (*) a dolar (\$). Hvězdičku lze použít při nahrazení libovolného počtu znaků. Dolar se používá jako ukončení řetězce znaků.

```
User-agent: *
Disallow: *.pdf$
Sitemap: http://www.example.com/sitemap.xml
```

V uvedeném příkladu zakazujeme všem robotům vyhledávačů procházení URL adres, které končí se znaky „.pdf“, přičemž před tím je libovolný počet znaků.

Seznam v současné době podporuje více zástupných znaků, jejich specifikaci nazvanou Robots.txt 2.0 naleznete v [návodě Seznam](#).

Lidé se často domnívají, že v souboru robots.txt zakážeme robotům také indexaci stránek. V tomto souboru je pouze zakazujeme vyhledávačům procházet. Pokud se stránka v indexu již nachází, následovným zakázáním procházení dané stránky, již uloženou stránku z indexu nesmažeme. Zamezíme pouze opětovné návštěvě stránky robotem vyhledávače z interních odkazů.

Ověřenou metodou, jak můžete funkční stránku z indexu odstranit, je použít nástroj [Google Search Console](#) a odstranit dočasně stránku z indexu Google. Zároveň je však potřeba umístit do hlavičky zdrojového kódu dané stránky meta tag `robots` s obsahem „noindex“, aby se nestalo, že se stránka časem do indexu vrátí. K tomu lze využít v nástroji Google Search Console přehled *Index Google > Odstranit adresy URL*. Více v [návodě Google](#).

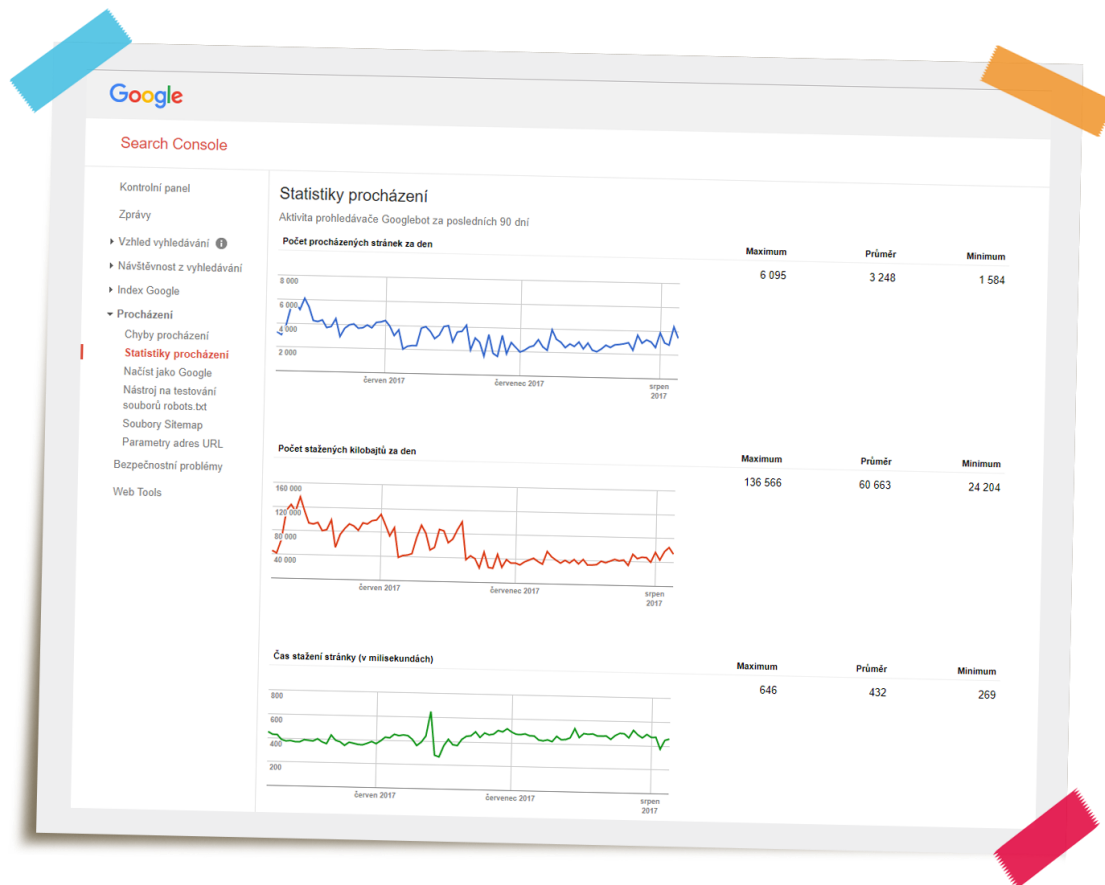


1.3. Crawl budget

Internet v současné době tvoří více než 1 bilión webových stránek. Některé stránky se aktualizují častěji, jiné méně, některé nikdy. Aby byly vyhledávače schopny udržovat svou databázi stránek co nejaktuálnější, musí si stanovit priority při jejich procházení. Z tohoto důvodu vznikl crawl budget. Crawl budget (někdy také crawl space nebo crawl time) je čas, který vyhledávač na vašem webu stráví, a počet stránek na webu, které za návštěvu načte. Crawl budget se v průběhu času mění. Jeho velikost závisí na:

- počtu stránek webu,
- četnosti aktualizace jednotlivých stránek,
- autorita webu (historie domény, kvalita zpětných odkazů apod.),
- velikosti stránek,
- rychlosti načítání stránek, případně dalších faktorů.

Údaje o crawl budgetu Googlebota můžete nalézt v nástroji Google Search Console v sekci *Procházení > Statistiky procházení*. Zde lze vidět, kolik stránek Googlebot prošel denně za posledních 90 dní a také, jaký čas na webu strávil.



Obrázek 1. Google Search Console - statistiky procházení.



1.3.1 Jak crawl budget optimalizovat?

Crawl budget ovlivňuje to, jak rychle naleznou vyhledávače na vašich stránkách nový obsah, ale také to, jak čerstvé varianty stránek budou z tohoto webu ve výsledcích vyhledávání poskytovat. Proto je velmi důležitý. Můžeme jej optimalizovat několika způsoby:

- Optimalizovat velikost webu a jeho rychlost načítání.
- Omezit procházení zbytečných a nehodnotných stránek v robots.txt nebo definovat atribut `rel="nofollow"` u všech odkazů, směřujících na dané stránky.
- Neuvádět zbytečné a nehodnotné stránky v souboru sitemap.xml.
- Snížit počet JS a CSS souborů, na které je z kódu webových stránek odkaz.
- Odstranit řetězce v přesměrování (redirect chain).
- Používat tag `lastmod` v souboru sitemap.xml s aktuální informací o změně stránky.

1.3.2 Řešení filtrace položek

Filtry pomáhají návštěvníkům zužovat výběr položek na stránce na základě vybraných vlastností. Nejčastěji se s filtry můžeme setkat u e-shopů, kde uživatelům usnadňují výběr produktů tím, že si zvolí jen takové vlastnosti, které je zajímají. Pokud mají ale produkty velké množství vlastností, které lze filtrovat, může jejich kombinací vzniknout velké množství stránek, které mají velmi podobný obsah. Toto nám může dělat problém při indexaci a čerpání crawl budgetu. Jako příklad zde uvedeme e-shop s pneumatikami, kde je použito 5 produktových filtrů. Každou pneumatiku lze filtrovat na základě období (3 hodnoty), značky (36 hodnot), šířky (25 hodnot), profilu (15 hodnot) a průměru (13 hodnot). Díky kombinaci jednotlivých parametrů pneumatiky lze získat nespočet různých kombinací, respektive stránek.

V této kapitole se podíváme na řešení filtrů z pohledu jejich dohledatelnosti a na to, jak je nastavit, abychom co nejefektivněji využili crawl budget.

U některých filtrů je vhodné, aby byly vyhledávači indexovatelné a u některých nikoli. Obecně řečeno indexovat chceme takový obsah, který lidé hledají. Zároveň je třeba při návrhu produktových filtrů předcházet vzniku duplicit a zbytečných stránek, které akorát vyčerpávají crawl budget (viz kapitola [1.3](#)).



1.3.3 (Ne)Indexace produktových filtrů

Je vhodné stránky z filtrace indexovat nebo ne? Odpověď není jednoznačná. Některé stránky je vhodné indexovat, některé ne. V tom jak tyto dvě skupiny rozlišit pomůže analýza klíčových slov. Hledají zákazníci mobily podle značky? Ano. Indexovat! Hledají zákazníci mobily podle barvy? Ne. Indexace je zbytečná. Často je to až takto jednoduché. Nejvhodnějším způsobem, jak pro robota filtry na indexovatelné a neindexovatelné rozdělit, je buď zamezit mu k neindexovatelným filtrům přístup prostřednictvím souboru robots.txt, nebo prostřednictvím umístění meta tagu `nofollow` přímo do odkazu v určitém filtru a zároveň zakázání indexaci této stránek meta tagem `robots` s hodnotou `noindex`. Zakázání procházení v souboru robots.txt by mohlo vypadat například takhle:

```
Disallow: /*barva=*
```

Procházení jednotlivých filtrů, pokud je uveden v URL parametr, lze také **zakázat v nástroji Google Search Console**. Tímto způsobem však bude mít zakázaný přístup jedině Googlebot, což v českém prostředí díky vysokému podílu vyhledávače Seznam nestačí. Dalším způsobem, jak zabránit robotům vyhledávačů v procházení filtrů, je nastavit v URL adrese znak `#` a zařídit, aby se každý nepotřebný filtr přepsal v URL až za mřížkou. Tímto způsobem nezajistíme jen to, že stránky vyhledávače nebudou procházet, ale zabezpečíme také to, že se stránky nebudou indexovat. V případě výběru filtru typu „barva“ a „značka“ by tedy zápis URL adresy mohl vypadat následovně:
<http://www.example.com/mobily/?znacka=samsung#barva=cerna>.

1.3.4 Pevné pořadí kritérií filtru v URL adrese

Ať už filtry v URL adrese skládáme prostřednictvím URL parametrů, nebo v rámci hezkých URL, vždy je třeba, aby měla jednotlivá kritéria v rámci filtru vždy stejné pořadí, bez ohledu na to, jakou cestou se uživatel na stránku dostal. **Nemělo by tedy docházet k situaci, že vedle sebe existují například stránky:**

- www.example.com/?filtr=true&typ=bezecke&znacka=adidas
- www.example.com/?filtr=true&znacka=adidas&typ=bezecke

Proto by každý parametr v URL měl mít vždy pevné pořadí.

Kromě vyčerpávání crawl budgetu se jedná také o obsahové duplicity, které si tím zbytečně vytváříme (jedna a ta samá stránka je dostupná z více URL).



1.3.5 Vyhnout se řetězcům v přesměrování (redirect chain)

Řetězec přesměrování, neboli „redirect chain“, je několik po sobě následujících (zbytečných) přesměrování. Spousta webů nemá správně nastavené přesměrování stránek, z původní stránky „A“ do finální stránky „B“. Místo toho často vznikají řetězce přesměrování, tedy z původní stránky „A“ vede přesměrování do druhé stránky „B“ a z této stránky pak vede přesměrování do finální stránky „C“. V takovém případě je vyhledávač nucen navštívit každou „jednotku“, čímž si ukracuje svůj rozpočet na procházení webu (crawl budget). V případě nesmyslně dlouhého řetězce přesměrování může vyhledávač dokonce přestat s procházením, a tedy finální stránku vůbec nenavštíví (čili ani nezaindexuje).

1.4. Interní prolínání webu

Interním odkazem se rozumí odkaz, který je umístěn na webu, po jehož prokliknutí se uživatel dostane na jinou stránku stejného webu. Interní odkaz může mít podobu obrázku (animace) nebo textu. Stejně jako pro uživatele, tak i pro roboty vyhledávačů je velice důležité, jakou podobu tyto interní odkazy mají a jakou informační hodnotu přinášejí. V případě textového odkazu se jedná o tzv. „anchor text“ neboli text odkazu, který uživatelům a robotům sděluje, kam se na webu po prokliknutí přemístí. V případě obrázku zastupuje funkci anchor textu alternativní popis obrázku `alt` nebo jeho titulek `title`. Platí to především pro roboty vyhledávačů, kteří z obrázků nedokážou pochopit jeho význam, a proto tuto informaci čerpají z těchto atributů.

1.4.1 Proč je důležité interní prolínání webu?

Jak už jsme zmiňovali, interní odkazy napomáhají uživatelům a robotům vyhledávačů v lepší orientaci na webu. Díky kvalitně zpracované struktuře interních odkazů ulehčíme vyhledávačům přecházení webu. Dokážeme je tak snadno navigovat napříč webem a dostat je na všechny důležité stránky webu. A nejen to, pomocí interních odkazů dokážeme jiným stránkám předávat jisté množství „link juice“, čímž dokážeme předat hodnotu z jedné stránky na druhou a podpořit hodnocení u vyhledávačů. Předávání „link juice“ funguje podobně jako v případě externích zpětných odkazů, které přinášejí nejen reálnou návštěvnost, ale také zvyšují hodnotu odkazované stránky.

Z pohledu uživatelů je přínosnost kvalitně zpracovaných interních odkazů zřejmá. Docílíme tím toho, že např. uživatel po vstupu na náš web nebude zmatený, neztratí zájem a neodejde z našeho webu dřív, než najde to, co hledal.



Při vytváření kvalitní struktury interních odkazů a propojování webu lze využít několik typů interních odkazů. V závislosti na umístění, kontextu a podobě pak odkazy nabírají také na důležitosti. Pojďme si tedy říct základní typy interních odkazů.

1.4.2 Odkazy v hlavní navigaci (menu)

Odkazy nacházející se v menu jsou ideálním řešením pro rychlou orientaci uživatelů na webu. Důležité je pořádně promyslet strukturu těchto odkazů, aby v každé situaci uživatel dokázal najít nejdůležitější části webu. V navigačním menu by rozhodně neměly chybět nejdůležitější kategorie, resp. podkategorie webu.

1.4.3 Odkazy v drobečkové navigaci

Drobečková navigace je velice užitečná z pohledu uživatele, protože je z ní zřejmé, kde se na webu nachází a jakou cestou se tam dostal (přehled o struktuře stránek). Uživatel se tak snadno orientuje a dokáže jednoduše přejít na nadřazenou stránku webu. Navíc tím dokonale zajistíme provázanost webu a přesun „link juice“ do nadřazených (obvykle také důležitějších) stránek.

V případě drobečkové navigace doporučujeme vždy nasadit také strukturovaná data Breadcrumb, viz kategorie [5.4.3](#).

1.4.4 Odkazy v textu

Téměř každý web disponuje hojným množstvím textového obsahu. Ať už jsou to zajímavosti, FAQ, články z blogu, nebo popisky produktů. Můžeme vše využít v náš prospěch a interními odkazy nasměrovat uživatele k dalšímu užitečnému obsahu, čímž se zvyšuje šance k dosažení požadované akce. Myslete však na to, že odkazy by měly směřovat jen do stránek, u kterých to dává smysl. Tedy na další související obsah.

Výhodou interních odkazů v textu je také to, že můžeme sami vybrat klíčová slova, z kterých odkazujeme na relevantní sekce. Při správném prolinkování a výběru klíčových slov se vám může podařit i mírně zlepšit pozice odkazovaných stránek na zacílená klíčová slova ve vyhledávání.



1.4.5 Odkazy na související obsah/produkty

Určitě jste už někdy brouzdali po webu a prohlíželi si nějaký produkt a zahlídli jste odkazy na „podobné produkty“. Také další chytrý způsob, jak můžete uživatele mírně ovlivnit a případně mu prodat o něco dražší produkt (up-selling), nebo naopak uživatele navedete k sekci „lidé si také koupili“ a místo jednoho produktu ho ovlivníte a koupí rovnou dva nebo více (cross-selling). Typickým příkladem cross-sellingu může být např. produktová stránka nějakého mobilního telefonu, kde budete nabízet jako „doplňkový sortiment“ sluchátka, kryt na telefon, ochranné sklo nebo jakékoliv jiné příslušenství.

1.4.6 Patičkové odkazy

Dalším způsobem, jak prolinkovat web, může být vytvoření odkazů v patičce. Myslete však na to, že odkazy by měly být užitečné především pro uživatele, a proto nedoporučujeme vkládat do patičky odkazy vedoucí na produkty nebo kategorie s jediným cílem „ovlivnit“ vyhledávač a získat tak lepší hodnocení těchto stránek. V případě patičkových odkazů doporučujeme řídit se zvyklostmi a odkazovat na stránky, které jsou v patičkách „zažité“. Jako například kontakt, o nás, obchodní podmínky, atd.

Většina webů má odkazy v patičce fixované, což znamená, že každá stránka webu má v patičce stejné odkazy. Vhodné řešení nadměrného počtu patičkových odkazů může být např. přizpůsobení těchto odkazů v závislosti na sekci webu. Jednoduše řečeno, každá sekce webu bude mít různě nastavené patičkové odkazy vzhledem k relevanci daného obsahu (přizpůsobíte si odkazy z patičky tak, jak chcete).

Interní odkazy jsou zajímavý způsob, jak můžete v jisté míře ovlivnit chování uživatelů na vašem webu, a taky být nápomocní vyhledávačům. Chytře promyšlená struktura interních odkazů s sebou nese množství výhod. Je to hlavně lepší orientace uživatelů, čímž se zvyšují šance, že uživatel po dlouhém bloudění z vašeho webu neodejde, ale dosáhne požadovaného cíle.



1.5. Stránkování obsahu

Stránkování tvoří principiálně stránky s velmi podobným obsahem. Nevhodně řešené stránkování může z pohledu SEO negativně ovlivnit indexaci webu a jeho hodnocení. Nyní si ukážeme, jak stránkování správně ošetřit. Nejprve si však popíšeme stav, kterého chceme dosáhnout.

- Vyhledávače indexují pouze první stránku kategorie či rubriky.
- Vyhledávače neindexují druhou a další stránku.
- Vyhledávače procházejí všechny stránky kvůli indexaci položek (podstránek), které se tam nacházejí.

Správné nastavení je následující. U první stránky nastavit meta robots na „index, follow“, u dalších stránek pak „noindex, follow“. Zároveň na každou stránku umístíme atributy `rel="prev"` a `rel="next"`.

Příklad nastavení první stránky:

```
<meta name="robots" content="follow,index">
<link rel="next" href="http://www.example.com/kategorie/strana=2">
```

Příklad nastavení druhé stránky:

```
<meta name="robots" content="follow,noindex">
<link rel="prev" href="http://www.example.cz/kategorie/">
<link rel="next" href="http://www.example.com/kategorie/strana=3">
```

Dalším možným řešením je vytvořit stránku, která bude obsahovat všechny položky. Následně všechny ostatní stránky s částečným výpisem položek na tuto souhrnnou stránku kanonizovat ([pomocí atributu rel="canonical"](#)). Docílíme tím toho, že budeme mít jednu hlavní „super stránku“, která bude obsahovat všechny položky a na kterou se bude kanonickým tagem odkazovat ze všech stránek s částečným výpisem položek. Kanonické tagy vedlejších „částečných“ stránek by měly zajistit, že se ve vyhledávání zobrazí jenom naše hlavní stránka. Zároveň se ujistíme, že v případě vzniku externích zpětných odkazů vedoucích do „vedlejších“ stránek, se pomocí kanonického tagu účinek ze zpětných odkazů přesune do naší hlavní stránky. Má to však i zápornou stránku a to z pohledu UX nebo rychlosti načtení. Toto řešení tedy nedoporučujeme jako prioritní.



1.6. Přesměrování stránek webu

Pokud dojde ke změně URL či smazání stránky, vyhledávače velmi často drží tuto URL ve svém indexu a také ji zobrazují ve výsledcích vyhledávání mnohdy i několik týdnů. Pokud stránka existovala delší dobu, je pravděpodobné, že za svou existenci získala i nějaké zpětné odkazy, či byla sdílena na sociálních sítích. URL mohou být použity v dokumentech (doc, pdf), či použity v e-mailové komunikaci (např. newsletter). Abychom se vyhnuli zbytečným potížím, měli bychom se snažit o **neměnnost URL adres**. Když vám to ale situace nedovoluje, je potřebné [řešit správné přesměrování](#).

Přesměrování z jedné URL adresy na jinou, je při správě webu velmi častá operace. Může k ní docházet z několika důvodů:

- Změna URL adresy při redesignu webu.
- Přejít na HTTPS a nutnost přesměrovat 1:1 HTTP verze URL adres na HTTPS verze.
- Změny ve struktuře webu zahrnující změny URL.
- Zrušení nebo smazání obsahu a jeho přesměrování na logického následovníka.

Z pohledu SEO při přesměrování může docházet k několika typickým chybám:

- Vznik smyček nebo řetězců v přesměrování.
- Nevhodně zvolený stavový kód.
- Špatně použitý typ přesměrování (JavaScript, Meta Refresh).

U posledního bodu bychom se zastavili. Přesměrování lze totiž realizovat několika způsoby a vždy je třeba zvolit ten správný. Při špatném použití přesměrování může docházet k problémům s použitelností webu a v některých případech i ke ztrátě autority webu a propadu ve výsledcích vyhledávání.

1.6.1 Trvalé přesměrování 301

Přesměrování stavovým kódem 301, které se také nazývá **trvalé přesměrování**, je asi nejčastěji používaným typem. Dle serveru Moz.com při přesměrování [údajně dochází k přesunu 90 % \(a více\) tzv. link juice](#) (kvalita stránky vyhodnocená vyhledávačem na základě zpětných odkazů směřujících na stránku) z původní URL na novou, takže by nemělo docházet k propadům ve výsledcích vyhledávání. Používá se v těchto situacích:

- redesign webu,



- přechod na HTTPS,
- rušení obsahu a nahrazení novým na jiné URL.

1.6.2 Nalezeno 302

Přesměrování stavovým kódem 302 se nazývá **nalezeno**. Byť [John Mueller na svém Google+ profilu](#) potvrdil, že při migraci webu z HTTP na HTTPS nedochází ke ztrátě „link juice“ u vyhledávače Google, stále nebylo ověřeno, že se tak děje ve všech případech. Proto doporučujeme používat protokol 302 jen ve striktně daných situacích:

- Pokud jde skutečně o dočasné přesměrování a přesměrovaná stránka má i v budoucnu figurovat v indexu.
- Při přesměrování uživatele mezi desktop a mobilní verzi webu. Tedy v případě, že uživatel na mobilním zařízení zadá [www.example.com](#) a vy mu chcete nabídnout mobilní verzi stránky (tj. např. [m.example.com](#)). V tomto případě lze použít také stavový kód 301, nicméně [vyhledávač Google doporučuje 302](#).

Dalším důvodem, proč nepoužívat stavový kód 302 k trvalému přesměrování je fakt, že vyhledávače často zobrazují starou URL ve výsledcích vyhledávání.

1.6.3 Dočasné přesměrování 307 (od HTTP 1.1)

Přesměrování stavovým kódem 307 se používá velmi zřídka. Jde o dočasné přesměrování platné jen pro protokol HTTP 1.1 (pro protokol HTTP 1.0 je určeno přesměrování 302). Se stavovým kódem 307 se také můžeme setkat při tzv. interním přesměrování. Jakmile prohlížeč uvidí na webu hlavičku HTTPS Strict-Transport-Security (HSTS), tak ví, že při příštím požadavku již nemá stránku požadovat na protokolu http, ale výhradně na HTTPS (za každou cenu). Prohlížeč si tuto informaci zapamatuje na dobu určitou (na základě nastavení) a automaticky bude při pokusu o otevírání HTTP stránky vykonáno přesměrování pomocí stavového kódu 307 na HTTPS verzi stránky.

1.6.4 Přesměrování přes JavaScript

Pro přesměrování přes JavaScript se používá syntaxe, která se umístí do hlavičky dokumentu. Nedoporučujeme však upřednostňovat přesměrování JavaScriptem před HTTP přesměrováním. Problémem totiž je, že pokud má uživatel vypnutý JavaScript, tak k přesměrování nedojde. Navíc vyhledávač Seznam neumí pracovat [se všemi typy přesměrování JavaScriptu](#).



1.6.5 Co když ale stránka (obsah) definitivně zmizí z webu?

Častý problém, který mnoho správců řeší, je co s URL stránek, které na webu již neexistují. Zde je nutno podotknout, že **ne každá URL musí být nutně přesměrována**. Přesměrování dává smysl pouze tehdy, pokud **došlo k zachování obsahu** stránky (tedy pouze ke změně URL adresy), nebo **obsah je alespoň informačně podobný tomu smazanému**. V tomto případě je přesměrování na místě. Pokud obsah zmizí a informace ze stránky již na webu nejsou, pak je v pořádku, že stará URL vrátí chybovou stránku.

V případě smazání obsahu a zobrazení chybové stránky se nejčastěji používají stavové kódy 404 a 410. První stavový kód říká, že stránka nebyla nalezena, ale v budoucnu může být dostupná. Druhý stavový kód (410) naopak dá vyhledávačům informaci, že stránka na webu již není a ani nebude v budoucnu dostupná.

Doporučujeme ale vždy předem zvážit, jestli není jiné řešení, než stránku smazat, resp. zrušit. Možným řešením je nahradit ji jinou, aktuální stránkou. Co však např. s produktem, který přestaneme prodávat, tj. co se stránkou, na které se onen produkt nachází?

Představte si, že prodáváte Playstation 3 a máte pro něj vytvořenou produktovou stránku na svém e-shopu. Najednou na trh přijde nová herní konzole a vy si řeknete, že Playstation 3 vám už nepřináší tolik zisku (resp. se vám produkt nevyplatí dále prodávat), a tak stránku zrušíte. Existuje však i jiné řešení. Produktová stránka Playstation 3 může mít velmi dobrou přirozenou návštěvnost, tedy byla by škoda o její návštěvníky přijít. Vhodným řešením tak může být stránku nerušit, ale pouze na ni přestat odkazovat z webu, aby se k ní nemohl nikdo dostat pomocí interního odkazu (např. z menu). Stránka ale zůstane v indexu vyhledávačů a bude ji tedy možné dohledat na základě klíčových slov. Jelikož Playstation 3 už nechceme prodávat, na stránku doplníme snadno viditelný text, kterým potenciálního zákazníka informujeme o tom, že produkt již neprodáváme a zároveň jej nasměrujeme k jinému aktuálnímu produktu. Volili bychom např. upoutávku "Zapomeňte na Playstation 3, na trhu je nová herní konzole Playstation 4! Zjistěte více!".



1.7. Duplicitní stránky

Duplicitní či velmi podobný obsah dostupný pod více URL je jeden z nejčastějších technických problémů webu z pohledu SEO. Často se s duplicitami setkáme u větších a funkčně složitějších webů, například e-shopů, kde se pomocí parametrů automaticky generují různé URL adresy, které mají ale stejný nebo velmi podobný obsah. Uvedeme zde několik nejčastějších příkladů, kdy se lze s duplicitami setkat:

- Web je nasazen na verzi bez www a s www současně.
- U e-shopu, pokud je produkt zařazen ve více kategoriích. V případě, že se do URL stránky vypisuje také kategorie, může nastat problém.
- Pokud na web směřujete kampaně s UTM parametry.
- Další častá duplicita nastává při filtraci či řazení produktů v kategorii. Lépe řečeno podobnost, protože ke drobné změně dojde a to k přeskládání produktů.

Jedno z řešení těchto duplicit je `canonical` tag, pomocí kterého se nastaví "hlavní" URL adresa a k ní její varianty.

1.7.1 Kanonizace stránek

Kanonizace funguje z pohledu vyhledávačů podobně jako přesměrování 301, s tím rozdílem, že uživatel není fyzicky přesunut na jinou URL adresu. Používá se typicky v situacích, kdy víme o duplicitách na webu a chceme pro vyhledávače označit jednu kanonickou URL, která je z daných duplicit určená k indexaci. Nutno říci, že vyhledávače nemusí toto označení následovat, jedná se pouze o doporučení, kterým se mohou řídit. Autorita webových stránek se při kanonizaci přenáší podobným způsobem, jako u přesměrování stavovým kódem 301.

Kanonické URL se zapisují v HTML hlavičce pomocí následujícího tagu:

```
<link rel="canonical" href="http://www.example.com/kanonicka-verze.html">
```

A to na stránce, kterou nechceme indexovat, i na hlavní stránce, kterou indexovat chceme (ta bude kanonizována sama na sebe). V praxi to tedy může vypadat takto.

Máme 3 duplicitní stránky:

```
http://www.example.com/hlavni  
http://www.example.com/vedlejsi-1  
http://www.example.com/vedlejsi-2
```



Hlavní stránku chceme označit jako kanonickou.

Do hlavičky obou vedlejších stránek i hlavní stránky tedy zapíšeme:

```
<link rel="canonical" href="http://www.example.com/hlavni.html">
```

Na vedlejších stránkách takto předáme vyhledavačům informaci, že existuje podobná důležitější stránka, kterou chceme indexovat. Na hlavní stránce přidáním kanonizace na sebe sama zabráníme tomu, aby se do indexu dostala jiná, nechtěná verze stránky.

To se může stát například při použití UTM parametrů v URL. Přidáním UTM parametru vznikne nová URL, ale obsah stránky zůstane stejný. Např. při prokliku z emailové kampaně sledované UTM parametrem, by pak URL vypadala takto:

http://www.example.com/hlavni?utm_source=newsletter&utm_medium=email&utm_campaign=valentyn

Pokud by někdo vzal tuto URL s UTM parametrem a odkázal na ni např. na svém webu nebo v diskusním fóru, mohla by se nám tato stránka dostat do indexu vedle hlavní stránky (<http://www.example.com/hlavni>) a vznikla by tak nechtěná duplicita. Proto je nutné každou stránku kanonizovat sama do sebe, tzv. „self canonical“.

1.8. Procházení a indexace AJAX webů

Pokud se dnes chcete podívat na zdrojový kód webu, často se místo kompletního HTML kódu setkáte s kódem, který je vytvořen v jednom z dnes populárních JavaScriptových frameworků, jako jsou Angular nebo React. Například na webových stránkách Zboží.cz vypadá zdrojový kód následovně:



```

1 <!doctype html>
2 <html>
3 <head>
4 <meta charset="utf-8">
5 <meta http-equiv="X-UA-Compatible" content="IE=Edge"/>
6
7 <title>Zboží.cz • tisíce obchodů na jednom místě</title>
8 <meta name="description" content="Na Zboží.cz jsou produkty včetně popisů, recenzí, příslušenství a návodů. Ceny si navíc můžete srovnat od těch nejlevnějších.">
9 <noscript><meta http-equiv="refresh" content="0;url=?_escaped_fragment_"></noscript>
10 <meta name="fragment" content="!" ng-if="showMetaFragment" />
11 <meta name="viewport" content="width=device-width, initial-scale=1" />
12 <meta name="referrer" content="origin">
13
14 <meta name="seznam-vmt" content="GYLHTGLKyxpiKgg2XI6G6w8J3XV7mm3X">
15 <meta name="szn:status" content="">
16
17 <link rel="shortcut icon" href="/img/favicon/favicon.ico?6.68.14" type="image/x-icon" />
18 <link rel="apple-touch-icon" sizes="57x57" href="/img/favicon/apple-touch-icon-57x57.png?6.68.14">
19 <link rel="apple-touch-icon" sizes="60x60" href="/img/favicon/apple-touch-icon-60x60.png?6.68.14">
20 <link rel="apple-touch-icon" sizes="72x72" href="/img/favicon/apple-touch-icon-72x72.png?6.68.14">
21 <link rel="apple-touch-icon" sizes="76x76" href="/img/favicon/apple-touch-icon-76x76.png?6.68.14">
22 <link rel="apple-touch-icon" sizes="114x114" href="/img/favicon/apple-touch-icon-114x114.png?6.68.14">
23 <link rel="apple-touch-icon" sizes="120x120" href="/img/favicon/apple-touch-icon-120x120.png?6.68.14">
24 <link rel="apple-touch-icon" sizes="144x144" href="/img/favicon/apple-touch-icon-144x144.png?6.68.14">
25 <link rel="apple-touch-icon" sizes="152x152" href="/img/favicon/apple-touch-icon-152x152.png?6.68.14">
26 <link rel="apple-touch-icon" sizes="180x180" href="/img/favicon/apple-touch-icon-180x180.png?6.68.14">
27 <meta name="msapplication-TileColor" content="#666666">
28 <meta name="msapplication-config" content="/img/favicon/browserconfig.xml?6.68.14">
29 <meta name="msapplication-TileImage" content="/img/favicon/mstile-144x144.png?6.68.14">
30
31 <link rel="stylesheet" href="/css/all.css?6.68.14">
32
33 <!--[if lte IE 9]>
34 <link rel="stylesheet" href="/css/all.oldie.css?6.68.14">
35 <script src="/js/lib/polyfills/ie8/es5.js?6.68.14"></script>
36 <script src="/js/lib/polyfills/ie8/js16.js?6.68.14"></script>
37 <script src="/js/lib/polyfills/ie8/html5shiv.min.js?6.68.14"></script>
38 <script src="/js/lib/polyfills/ie8/respond.min.js?6.68.14"></script>
39 <![endif]>
40
41 <script src="https://login.szn.cz/js/api/login.js"></script>
42 <script src="/i.in.cz/software/promo/promo-sbrouser.js" crossorigin"></script>
43
44 <script src="/i.szn.cz/config.js?6.68.14"></script>
45 <script src="/js/all.js?6.68.14"></script>
46
47 <body data-dot="zbozi">
48 <ng-view ng-if="!error" data-autoscroll=""></ng-view>
49 <ng-include ng-if="error" src="/app/view/errors/templ/errors.html"></ng-include>
50 <erotic-disclaimer ng-if="showEroticDisclaimer"></erotic-disclaimer>
51 <script src="/i.media.cz/js/im2.js" crossorigin"></script>
52 <script src="/i.media.cz/js/dot-small.js" crossorigin"></script>
53 <script src="/i.media.cz/js/sid.js" crossorigin"></script>
54 <script>Zbozi.init();</script>
55
56 </body>
57 </html>

```

Obrázek 2. Ukázka zdrojového kódu Zboží.cz.

Je to dáno tím, že jsou weby naprogramovány v některém z AJAX (Asynchronní JavaScript a XML) frameworků využívajících JavaScript. V tomto případě je HTML kód překládán z JavaScriptu přímo prohlížečem, což je oproti standardním programovacím jazykům jako PHP nebo ASP rozdílné, protože fungují tak, že se z jejich kódu tvoří HTML přímo na serveru a následně je hotový dokument předán prohlížeči.

1.8.1 Problémy s indexací obsahu v AJAX

S indexací stránek v AJAXu si některé vyhledávače jako např. český Seznam nemusí vědět rady, protože Seznambot JavaScriptový kód kvůli kapacitě nemusí spouštět a nengeneruje tak z něj HTML. Často se tedy stává, že vyhledávač ze stránky zaindexuje pouze navigaci nebo jiný statický obsah webu a hlavní sdělení zůstává nepřístupné v JavaScriptovém kódu.

Zmíněný Seznam poskytuje k indexaci stránek AJAXu alternativní řešení, kterou naleznete v [návodě Seznam](#).



1.8.2 Předrenderování

Kvůli výše zmíněné situaci vznikly nástroje, jako je například [Prerender.io](https://prerender.io), které web převedou do statického HTML kódu. Díky přítomnosti fragmentu:

```
<meta name="fragment" content="!" ng-if="showMetaFragment" />
```

V AJAXové verzi stránky tak vyhledávač ví, kde má statickou HTML verzi stránky hledat. V tomto případě se nachází statická HTML verze stránky <http://www.zbozi.cz/> na URL adrese https://www.zbozi.cz/?_escaped_fragment=. Použití escape fragmentu neboli [hashbangu je pro Google zastaralé](#) už od října 2015, avšak pořád toto řešení akceptuje a používá. Důležité je ale poznamenat, že v druhém kvartálu 2018 Google přestane používat tuto metodu a bude AJAXové stránky kompletně vykreslovat. Nicméně je to stále jediná metoda, jak předložit vyhledávači Seznam obsah stránek v AJAXu.

Nevýhodou této techniky je zbytečné čerpání crawl budgetu, protože robot musí každou stránku navštívit dvakrát (AJAX verze, statická HTML verze).

1.9. Nastavení vícejazyčných stránek a webů cílících na různé země

Značka `hreflang` slouží pro nastavení konkrétního jazyka nebo oblasti pro URL stránky webu. Jednoduše řečeno, pokud máte vícejazyčný web nebo web, který se orientuje na vícero zemí, měli byste tuto informaci sdělit vyhledávačům. Pomocí značky `hreflang` zajistíme, že vyhledávače budou naše weby rozlišovat, pochopí jejich jazyk a oblast, na kterou cílí. Vyhledávače pak dokážou lépe reagovat na vyhledávací dotazy uživatelů.

Pokud máme např. web www.example.com zaměřený na americký trh a zároveň máme web www.example.co.uk zaměřený na britský trh, vyhledávače to na základě značky `hreflang` budou rozlišovat. Nenastane pak situace, že by vyhledávač vnímal tyto dva různé weby jako duplicitní, protože pochopí, že se jedná o „ten samý web“ s „tím samým jazykem“, který je ale orientovaný na odlišné země. Vyhledávač dokáže lépe reagovat na vyhledávací dotazy uživatelů a nebude jim nabízet výsledky určené pro jiný trh, když pro ně existuje lepší alternativa.



1.9.1 Způsoby definování značky hreflang

A. Zápis značky hreflang v HTML kódu stránky.

B. Záhlaví protokolu http.

C. Zápis značky hreflang v souboru Sitemap.xml.

1.9.2 Jak by měl vypadat zápis značky hreflang v HTML kódu?

Představme si, že máme společnost Example s.r.o., přičemž vlastníme webové stránky example.com (zaměřené na americký trh) a example.co.uk (zaměřené na britský trh). Abychom tyto stránky dokázali odlišit, potřebujeme nastavit značku hreflang, a to následovně:

- Do hlavičky HTML kódu stránky `http://www.example.com/` umístěte:

```
<link rel="alternate" hreflang="en-US" href="http://www.example.com/" />
<link rel="alternate" hreflang="en-GB"
href="http://www.example.co.uk/" />
```

- Zároveň do hlavičky HTML kódu stránky `http://www.example.co.uk/` umístěte:

```
<link rel="alternate" hreflang="en-GB"
href="http://www.example.co.uk/" />
<link rel="alternate" hreflang="en-US" href="http://www.example.com/" />
```

Vždy, když stránka A odkazuje na stránku B, musí i stránka B odkazovat na stránku A. Jinak se může stát, že hreflang nebude akceptovaný nebo správně implementovaný. Chyby ve značce hreflang můžeme zjistit přímo v *Google Search Console* > *Návštěvnost z vyhledávání* > *Mezinárodní cílení*.

Nezapomeňte také na to, že na každé stránce webu musí být uvedena skutečná URL adresa ve značce hreflang. Tedy v případě stránky `http://www.example.com/category1/` musí být v hlavičce stránky:

```
<link rel="alternate" hreflang="en-US"
href="http://www.example.com/category1/" />
<link rel="alternate" hreflang="en-GB"
href="http://www.example.co.uk/category1/" />
```



A v případě stránky `http://www.example.co.uk/category1/` musí být v hlavičce stránky:

```
<link rel="alternate" hreflang="en-GB"
href="http://www.example.co.uk/category1/" />
<link rel="alternate" hreflang="en-US"
href="http://www.example.com/category1/" />
```

Pokud však nějakou URL stránku máte jenom na jednom z webů, např. stránka `http://www.example.com/category2/` existuje jenom na doméně `example.com`, **hreflang neuvádějte**, protože daná stránka nemá svou alternativu v jiném jazyce/jiné zemi.

Výše uvedenou specifikace lze použít pro jazykové mutace řešené formou subadresáře (např. `example.com/cz` a `example.com/sk`), doménou prvního řádu (`example.com` a `example.co.uk`), doménou druhého řádku (`example.com` a `another-example.com`) nebo subdoménou (`portugal.example.com` a `czech.example.com`).

1.9.3 Zápis do záhlaví protokolu HTTP

Pokud máte možnost nastavovat svůj server, můžete `hreflang` pro adresy URL dokumentů HTML a jiných souborů (například `.pdf`) uvádět pomocí záhlaví HTTP. Například když máte `.pdf` dokument určený pro americký trh a `.pdf` dokument určený pro britský trh, můžete je definovat následovně:

```
Link: <http://www.example.com/american.pdf/>; rel="alternate";
hreflang="en-US", <http://www.example.co.uk/british.pdf/>;
rel="alternate"; hreflang="en-GB"
```

1.9.4 Hreflang v souboru Sitemap.xml

Pokud nechcete definovat `hreflang` v záhlaví protokolu HTTP ani přímým vložením do HTML kódu stránky, můžete tak učinit přímo v souboru `Sitemap.xml`. Zápis značky `hreflang` by pak měl vypadat následovně:

- V souboru `http://www.example.com/sitemap.xml` přiřadíte k URL adrese alternativu:

```
<url>
  <loc>http://www.example.com/category1/</loc>
  <xhtml:link rel="alternate" hreflang="en-US"
href="http://www.example.com/category1/" />
  <xhtml:link rel="alternate" hreflang="en-GB"
href="http://www.example.co.uk/category1/" />
</url>
```



- V souboru `http://www.example.co.uk/sitemap.xml` pak přiřadíte ke stejné URL adrese alternativu:

```
<url>
<loc>http://www.example.co.uk/category1</loc>
<xhtml:link rel="alternate" hreflang="en-GB"
href="http://www.example.co.uk/category1/" />
<xhtml:link rel="alternate" hreflang="en-US"
href="http://www.example.com/category1/" />
</url>
```

1.9.5 Kde najít správné kódy pro definování jazyku a země v atributu hreflang?

Kompletní seznam kódů pro určení jazykové verze najdete ve [formátu ISO 639-1](#) a seznam kódů pro země najdete ve [formátu ISO 3166-1 Alpha 2](#). Příklad:

- `hreflang="en"` – obsah stránky je v angličtině.
- `hreflang="en-US"` – obsah stránky je v angličtině a je určený uživatelům USA.
- `hreflang="en-DE"` – obsah stránky je v angličtině a je určený uživatelům Německa.
- `hreflang="x-default"` – obsah stránky se nezaměřuje na žádnou specifickou zemi ani jazyk.



2. ZABEZPEČENÍ WEBU (PROTOKOL HTTPS)

HTTP (Hypertext Transfer Protocol) je internetový protokol určený pro výměnu HTML souborů. Funguje na principu dotazu a odpovědi. Uživatel pomocí klienta (prohlížeče) pošle na server dotaz ve formě textu (pro potřeby této publikace postačí uvést URL), server požadavek zpracuje a pošle výsledek (např. HTML dokument).

Nevýhoda HTTP protokolu je ta, že je snadno prolomitelná. Zkrátka mezi serverem a klientem může komunikaci kdokoliv odchytit a třeba i poupravit. Umístit do stránky nebezpečný script, vložit do stránky formulář s cílem vytěžit od uživatele citlivá data apod. Proto je nutné komunikaci zabezpečit. Pro zabezpečení komunikace mezi serverem a uživatelem se používá protokol HTTPS.

V srpnu roku 2014 Google oznámil zanesení protokolu HTTPS mezi hodnotící faktory. Následně weby začaly velmi rychle migrovat na tento zabezpečený protokol. Zejména ve světě, kde Google neměl problém s přechodem.

V říjnu 2017 pak Google přistoupil k dalšímu kroku, a to když u nejpoužívanějšího prohlížeče Chrome (od verze 62) začal varovat návštěvníky webů s nezabezpečeným protokolem.

Na stránce, která není zabezpečená, zobrazí symbol „i“; po rozkliknutí tohoto symbolu prohlížeč informuje uživatele, že spojení s tímto webem není bezpečné. Jakmile se ale na stránce nachází formulář a uživatel jej začne vyplňovat, symbol se rozšíří o slovo „Nezabezpečeno“.

2.1. HTTPS a vyhledávač Seznam

Vyhledávač Seznam se z počátku dlouho nedokázal popasovat s přesměrováním (z HTTP na HTTPS verzi stránky). Zvláště u větších webů. Při změně URL docházelo ke ztrátě určitých signálů, což vedlo ke snížení hodnocení (a reálně tak k poklesu pozic). Toto se od začátku roku 2016 podařilo opravit. Proto by přechod měl být nyní bezpečný. Nicméně i přesto Seznam doporučuje, aby se vždy přechod plánoval na část roku, kdy není sezóna (tedy na období roku, kdy na web přichází nejméně návštěvníků z přirozených výsledků vyhledávání). Dále vyhledávač [Seznam ve své nápovědě uvádí](#), že nedělá mezi HTTP a HTTPS protokolem rozdíl. Protokol HTTPS tak nejspíš nepatří u Seznamu mezi hodnotící faktory.



2.2. Přejít na HTTPS protokol

HTTPS se stalo nyní standardem. Na přechod by se zkrátka nemělo čekat déle, než je nezbytné. Pokud budete řešit přechod na HTTPS, doporučujeme dodržet následující body:

- Ujistěte se, že **každý prvek webu používá HTTPS** (widgety, java script, css, obrázky apod.).
- Použijte **přesměrování 301** ze všech HTTP stránek na nové (identické) HTTPS stránky. Nepoužívejte dočasné přesměrování (302, 307).
- **Přepište všechny interní odkazy v kódu, které směřují na HTTP** adresy vašich stránek. Přepište vše, co je možné. Tedy nejen odkazy v navigaci, ale i odkazy z článků apod. Upravte URL v OG tazích, Twitter cards apod.
- Ujistěte se, že všechny **kanonické tagy poukazují na HTTPS** verzi URL adresy a ne na HTTP než tomu bylo předtím.
- Vyhněte se použití relativních URL ve zdrojovém kódu stránky (např. /strana.html). **Používejte absolutní URL** (<https://www.example.com/strana.html>).
- Před přechodem **zkontrolujte, že se na webu nenachází žádná HTTP stránka** například pomocí nástrojů Screaming Frog SEO Spider Tool, nebo Xenu's Link Sleuth.
- **Upravte URL XML feedů a RSS kanálů.**
- Zaregistrujte **verzi HTTPS v Google Search Console** (stará verze Search Console s HTTP webem neshromažďuje data z HTTPS). Search Consoli na HTTP verzi doporučujeme ještě nějakou dobu po přechodu nechat aktivní (nemazat).
- Nechte stránku **sitemap.xml s URL adresami HTTP po určitou dobu** (než vyhledávač navštíví všechny stránky a zaindexuje přesměrování 301 na stránky HTTPS). **Pak aktualizujte stránku sitemap.xml s novými HTTPS adresami.** Odešlete novou sitemap do nástrojů Google Search Console.
- **Aktualizujte soubor robots.txt.**
- Zkontroluje, zda Trackovací kód z Google Analytics je stejný s kódem, který máte na stránkách nového HTTPS webu. Také **ve službě Google Analytics změňte v nastaveních URL adresu webu z HTTP na HTTPS.**
- Implementujte HTTP Strict Transport Security (HSTS). Toto záhlaví řekne vyhledávačům, aby **přístupovali pouze k HTTPS stránkám**, i když se dostanou na HTTP. To eliminuje přesměrování, urychlí dobu odezvy a poskytne dodatečné zabezpečení.
- **Přenastavte Google Analytics** na HTTPS (*Správce > Nastavení služby > Výchozí adresa URL*).



- **Změňte URL v externích zdrojích** – profily na sociálních sítích, katalogy firem, PPC reklamy, bannerová reklama, PR články apod. Vhodné je si udělat analýzu odkazového profilu a identifikovat tak odkazy, které lze změnit. Pro to mohou posloužit nástroje [Majestic](#) nebo [Ahrefs](#).



3. RYCHLOST NAČÍTÁNÍ

Z pohledu SEO má rychlost načítání jeden základní význam. Jedná se o hodnotící faktor vyhledávačů. U vyhledávače Google měla doposud rychlost načtení stránky zásadní vliv u vyhledávání na desktopu. Od července 2018 s příchodem mobile-first indexing je toto aktuální také ve vyhledávání v mobilních zařízeních (viz článek [Using page speed in mobile search ranking](#)). Mezi další benefity dále patří lepší průchodnost a indexovatelnost webu vyhledávači nebo uživatelská přívětivost (UX).

Žádný limit pro ideální rychlost načtení stránky neexistuje. Obecně se doporučuje, že doba načtení stránky by neměla překročit 4 sekundy. Například [John Mueller ve svém tweetu](#) doporučuje dobu načtení pod 2-3 sekundy. V podstatě ale lze říct, že čím je kratší doba načtení, tím více to ocení uživatelé i vyhledávače.

3.1. Co všechno má na rychlost načítání vliv?

Na to, jak rychle se stránka načte, má kromě rychlosti internetového připojení vliv řada dalších okolností. Načtení webové stránky se skládá z posloupnosti následujících kroků:

- **připojování k IP adrese,**
- **parsování zdrojového kódu,**
- **vytváření Document Object Modelu (DOM) a CSS Object Modelu (CSSOM),**
- **spouštění kódu JavaScriptu,**
- **renderování (vykreslování) stránky.**

Z toho plyne, že dobu načítání a vykreslování webu může **brzdit několik věcí:**

- špatné připojení,
- poloha serveru (vzdálenost od místa požadavku),
- neoptimalizované procesy na serveru (např. databázové dotazy),
- příliš požadavků na server (příliš mnoho souborů ke stažení),
- příliš velké soubory ke stažení,
- soubory kaskádových stylů (CSS),
- JavaScript.

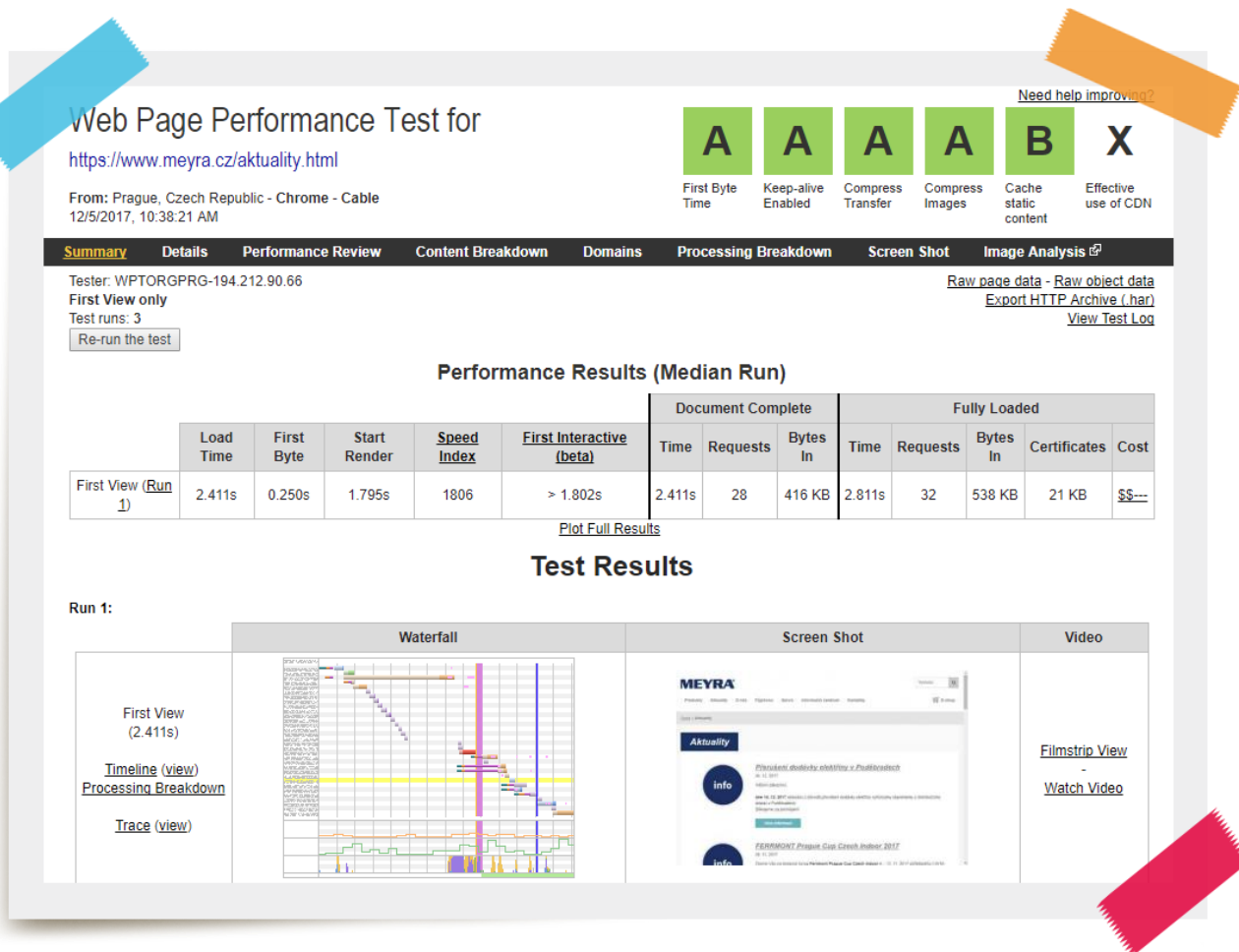


3.2. Jak rychlost načítání měřit?

Ke zjištění rychlosti webu lze využít různé nástroje, které jsou v základní formě zcela zdarma:

- [Webpagatest](#),
- [GTmetrix](#),
- [Pingdom](#),
- [Google PageSpeed Insights](#) a [Google TestMySite](#).

První tři možnosti, tedy Webpagetest, GTmetrix a Pingdom, jsou komplexní a v podstatě rovnocenné. Google PageSpeed insights a Google TestMySite slouží spíše k rychlé kontrole stavu.



Obrázek 3. Výsledek testu rychlosti v nástroji webpagetest.cz.



3.3. Jak dobu načítání zlepšit?

Nástroje zmíněné výše nám často ukáží, jak si na tom naše stránky relativně stojí a následně navrhnou metody, jak současný stav změnit k lepšímu. Nejlepších výsledků často dosáhnete zmenšením a zredukováním počtu statických souborů. Nabízí se ale i další možnosti, jak web výrazně a za vynaložení minimálního úsilí zrychlit.

3.3.1 Optimalizace obrázků

Optimalizace obrázků je zpravidla nejméně pracnou a zároveň nejvíce přínosnou činností v rámci optimalizace rychlosti načítání. Obrázky lze optimalizovat následujícími způsoby:

- Kde to jde, PNG nahradit JPEG formátem.
- Používat progressive JPEG formát.
- Obrázky bezztrátově komprimovat.
- Používat obrázky přímo ve velikostech, v jakých se zobrazují na webu a tuto velikost uvádět v kódu u každého obrázku v HTML atributu.

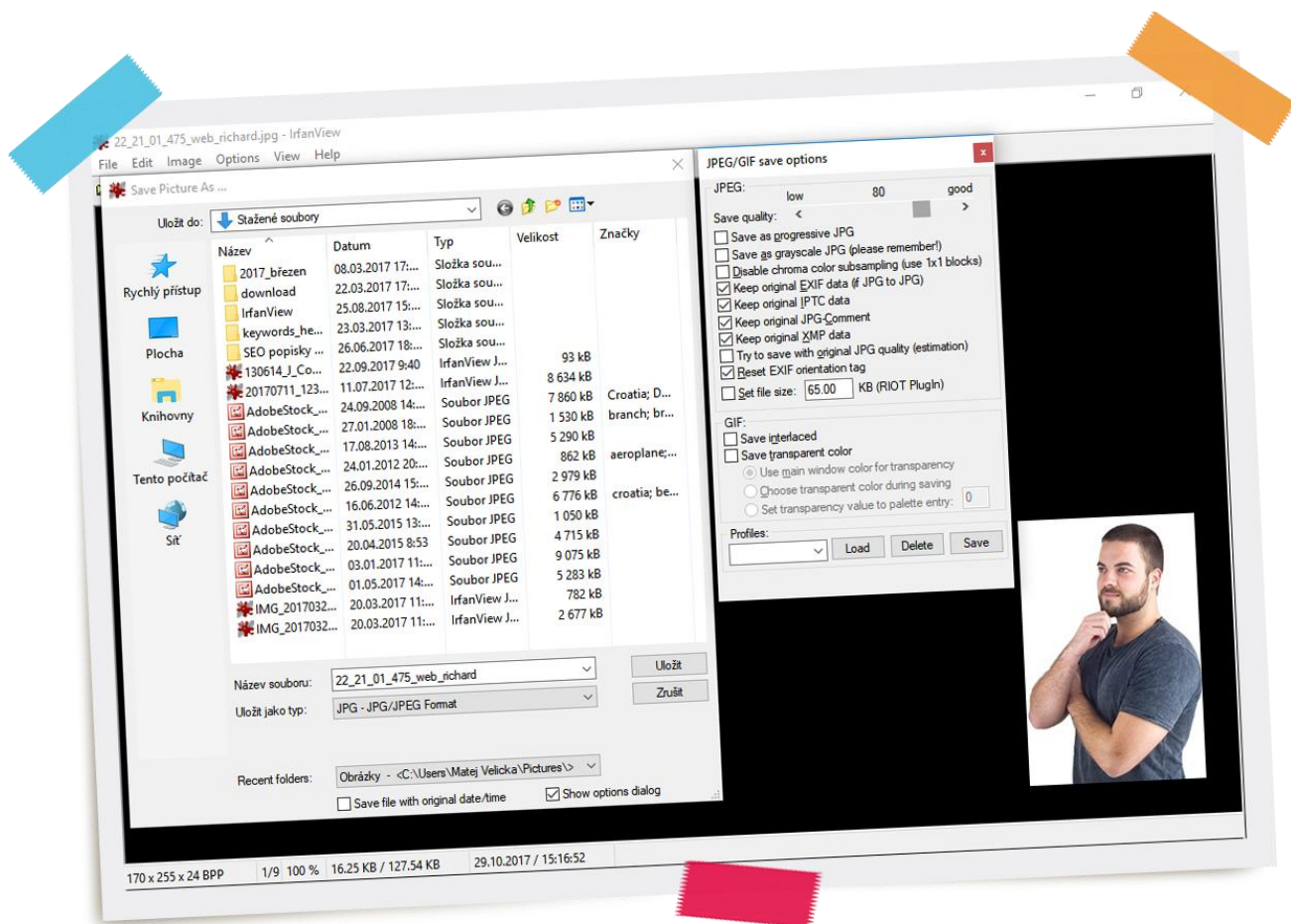
Progresivní JPEG (progressive JPEG) je způsob uložení obrázku, při němž dojde k načtení celého obrázku najednou v nižší kvalitě a k jeho postupnému zaostřování. Prohlížeč při prvním dotazu na server zobrazí jen některé pixely z obrázku a okolní zabarví stejnou barvou. Ty pak postupně vyplňuje správnou barvou, jak přicházejí pakety s kousky obrázku. Tento způsob úpravy je vhodný pro velké obrázky stahované přes relativně pomalé internetové připojení.



Obrázek 4. Ukázka vykreslení obrázku ve formátu progressive JPEG.



Uložit obrázek do formátu progressive JPEG lze například v prohlížeči obrázků IrfanView. Stačí při ukládání souboru zatrhnout „Save as progressive JPG“.



Obrázek 5. Uložení obrázku do formátu progressive JPEG pomocí IrfanView.

Pro rychlejší načtení obrázku je také vhodné jej komprimovat, ideálně bezztrátově. Ke komprimaci obrázků doporučujeme použít aplikaci [Compressor.io](#). Jedná se o velmi jednoduchou webovou aplikaci, kde lze najednou komprimovat soubory, a to i hromadně až do velikosti 10 MB.

3.3.2 Slučování a minifikace souborů

Jak už bylo řečeno na začátku kapitoly, ke zlepšení rychlosti načítání webu často vedou kroky jako redukce počtu souborů a jejich zmenšování. To platí kromě obrázků také pro soubory kaskádových stylů a JavaScriptu.

CSS sprite je technika, která se používá pro slučování většího množství malých obrázků, například ikonek, do jednoho velkého. Z tohoto obrázku (CSS spritu) se následně načte požadovaná ikonka (obrázek) pomocí CSS na konkrétní místo na stránce. Použití CSS spritů vede k výraznému zmenšení počtu dotazů na server a tím i k rychlejšímu načtení stránky.

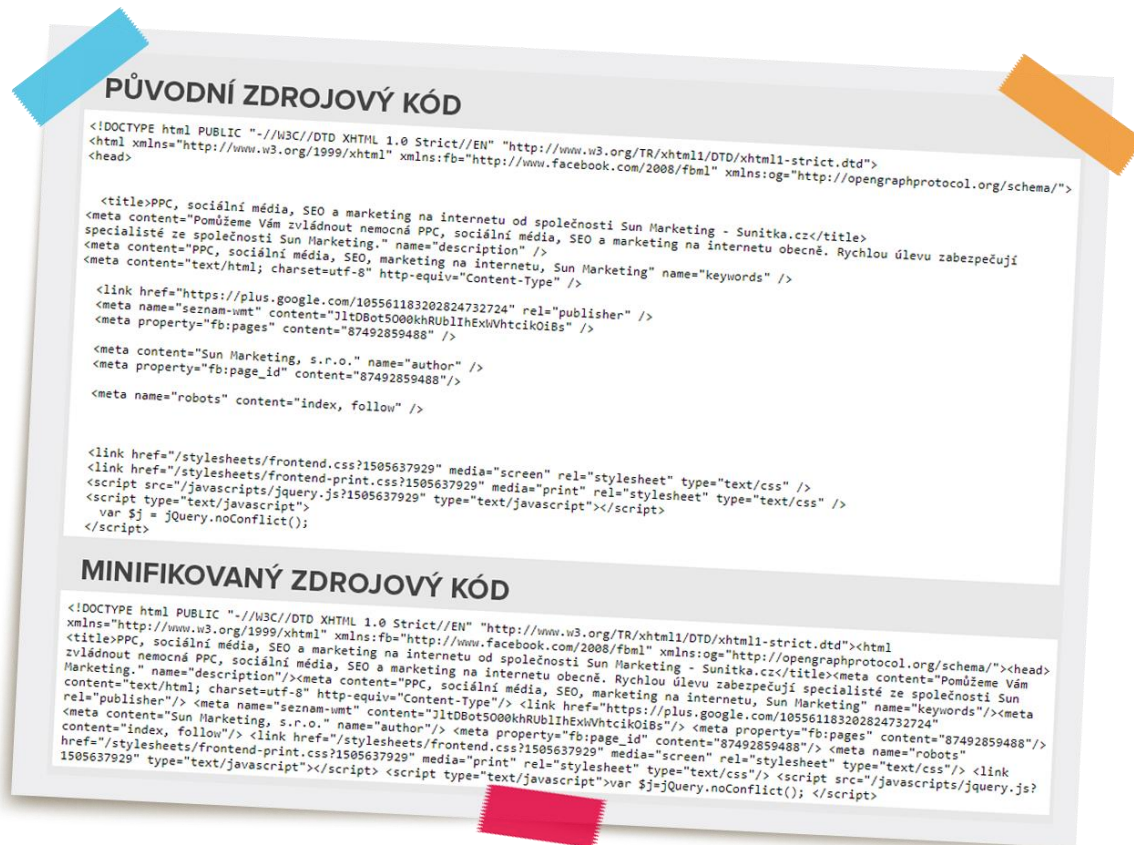


- Všechny obrázky se spojí do jednoho (CSS sprite).
- V kaskádovém stylu (CSS) se vytvoří elementy s rozměry pro jednotlivé obrázky vyříznuté z CSS spritu.
- V HTML kódu se pak použije značka span nebo div, pomocí kterých se na konkrétní místo použije námi určený výřez z CSS spritu.



Obrázek 6. Ukázka CSS sprite.

Minifikace souborů je metoda úspory místa na základě odstranění tzv. prázdných znaků z kódu HTML, JavaScriptu a CSS.



Obrázek 7. Příklad minifikace zdrojového kódu.



Slučování JS a CSS souborů funguje podobně jako CSS sprite u obrázků, jen se slučují jednotlivé soubory JavaScriptu a kaskádových stylů. Opět se tím redukuje počet dotazů na server a tím se zrychluje doba načítání webu.

3.3.3 Optimalizace cache

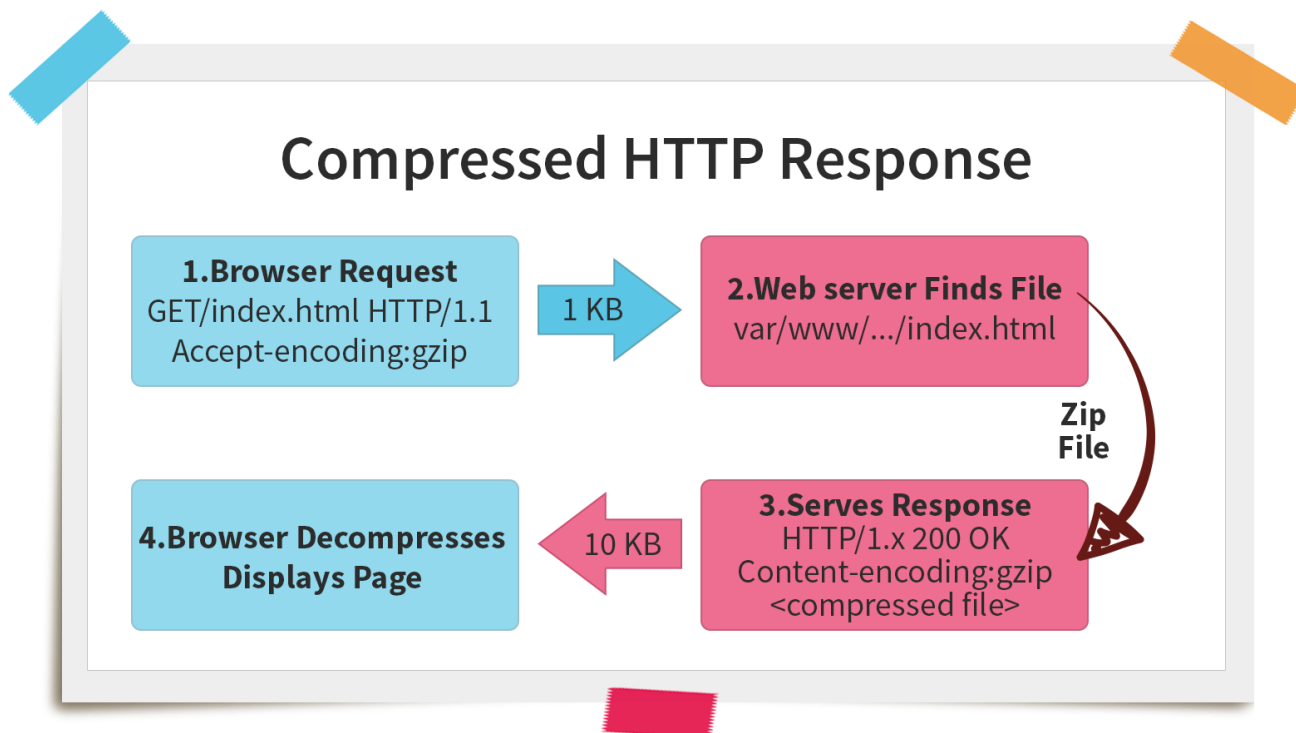
Optimalizací cache říkáte internetovému prohlížeči, na jak dlouho si má určité zdroje uložit do své mezipaměti. S takovými zdroji je při opakovaném načtení schopen pracovat výrazně rychleji. V případě webserveru Apache si cachování můžete nastavit například v konfiguračním souboru `.htaccess`. Kód by mohl vypadat následovně:

```
## EXPIRES CACHING ##
<IfModule mod_expires.c>
ExpiresActive On
ExpiresByType image/jpg "access 1 year"
ExpiresByType image/jpeg "access 1 year"
ExpiresByType image/gif "access 1 year"
ExpiresByType image/png "access 1 year"
ExpiresByType text/css "access 1 month"
ExpiresByType text/html "access 1 month"
ExpiresByType application/pdf "access 1 month"
ExpiresByType text/x-JavaScript "access 1 month"
ExpiresByType image/x-icon "access 1 year"
ExpiresDefault "access 1 month"
</IfModule>
## EXPIRES CACHING ##
```



3.3.4 GZip komprese

GZip komprese vychází z toho, že mnoho souborů, které se při komunikaci server-browser přenáší, je textové povahy. Takový obsah je možné na serveru výrazně komprimovat a ušetřit tak místo při přenosu.



Obrázek 8. GZip komprese.

U valné většiny prohlížečů GZip kompresi na serveru nastavíte velmi snadno. U některých serverů stačí v nastavení povolit GZip kompresi. U serveru Apache je třeba vložit do konfiguračního souboru `.htaccess` následující kód (nebo některou z jeho alternativ):

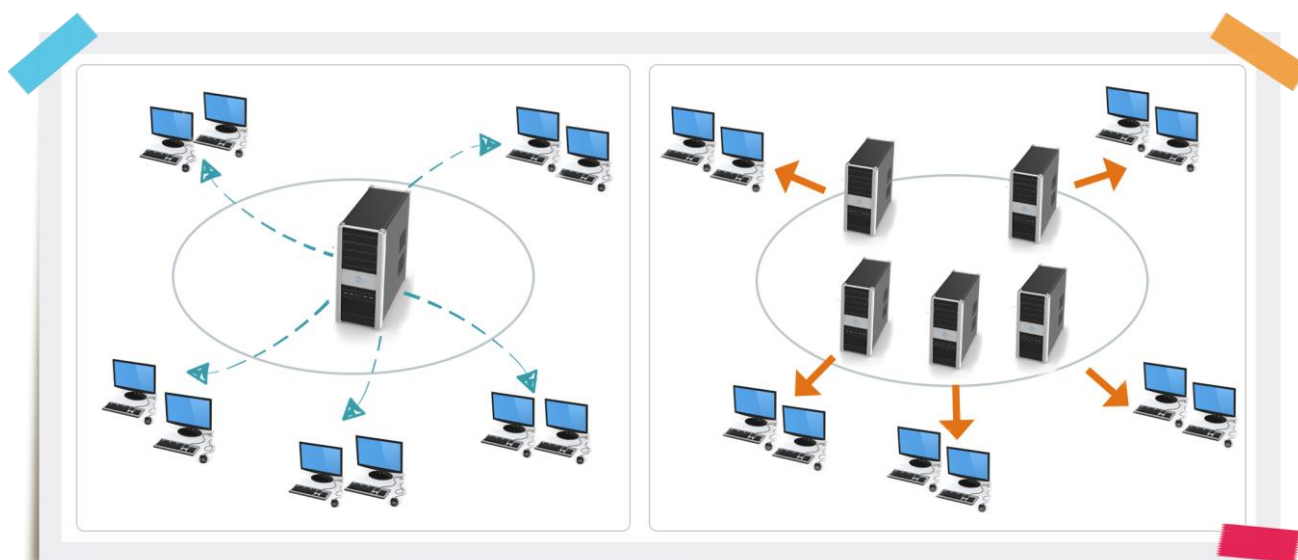
```
<ifModule mod_gzip.c>
mod_gzip_on Yes
mod_gzip_dechunk Yes
mod_gzip_item_include file \.(html?|text|css|js|php|p1)$
mod_gzip_item_include handler ^cgi-script$
mod_gzip_item_include mime ^text/*
mod_gzip_item_include mime ^application/x-javascript.*
mod_gzip_item_include mime ^image/*
mod_gzip_item_include rspheader ^Content-Encoding:.*gzip.*
</ifModule>
```



3.3.5 Využití CDN

CDN (Content Delivery Network) výrazně zrychlí váš web v okamžiku, kdy působíte na globální úrovni a na váš web chodí lidé z různých částí světa. Zjednodušeně se jedná o síť úložišť rozmístěných po celém světě, kde jsou většinou uložena statická data webu (lze ale uložit i dynamicky generovaná data), jako jsou obrázky, videa nebo CSS soubory. Pokud si web otevře například uživatel v Severní Americe, tato data se mu načtou z jemu bližší lokality (např. z New Yorku) a nemusí putovat přes celý svět ze serveru umístěného v ČR. CDN může zrychlit web například i v rámci ČR, protože se pro přenos dat často využívá rychlejší protokol HTTP/2.

Cena za využívání CDN se obvykle určuje specificky, na základě konkrétních požadavků. Mezi nejznámější poskytovatele patří Akamai, Limelight, Level3 nebo Amazon CloudFront.



Obrázek 9. Příklad CDN.

3.3.6 Optimalizace Time To First Byte

Time To First Byte (TTFB) je metrika, která značí dobu odpovědi serveru na první dotaz prohlížeče při načítání webové stránky. Čas se měří od odeslání prvního HTTP požadavku od uživatele k serveru až po přijetí prvního byte nazpět. TTFB ovlivňují zejména tyto faktory:

- [síťová latence](#) (vzdálenost serveru od lokality požadavku, počet síťových uzlů apod.),
- čas, který server potřebuje ke zpracování požadavku.



Průměrná hodnota bývá od 100 do 500 milisekund. [Google však doporučuje](#) držet tuto hodnotu pod 200 milisekund. Pokud je doba odezvy serveru příliš vysoká, způsobí to delší načítání stránky, což může vést ke 2 hlavním problémům:

- Pokud se vaše stránka načítá příliš dlouho, uživatel může odejít hledat odpověď u vašeho konkurenta.
- Doba načítání stránky je hodnotící faktor vyhledávačů, tedy pomalé načítání vašich stránek může negativně ovlivnit pozice ve výsledcích vyhledávání.

Problém s velkou vzdáleností mezi lokalitami požadavku a serveru vyřeší CDN (viz výše). Zbytek optimalizace je primárně záležitostí procesů na serveru, mezi něž patří například databázové operace.

3.3.7 Protokol HTTP/2

I české hostingy pomalu začínají nabízet nový protokol HTTP/2. Díky „multiplexování“ (v jednu chvíli jde po síti více požadavků a odpovědí) dokáže zpracovávat více požadavků a tak přenos dat [výrazně zrychluje](#). Podle [údajů serveru W3techs.com](#) běží v současnosti (srpen 2017) na protokolu HTTP/2 **15,9 % všech světových webových stránek**.

Podmínku pro nasazení protokolu HTTP/2 představuje přechod na zabezpečený protokol HTTPS, což je také jeden z hodnotících faktorů vyhledávače Google.

Hlavní výhodou protokolu HTTP/2 je ovšem zrychlení načítání stránek, díky čemuž se zlepšuje hodnocení webu vyhledávači.



4. MOBILNÍ POUŽITELNOST

Mobilní použitelnost webu je také hodnotícím faktorem pro vyhledávače. Na jaře roku 2015 dokonce Google oznámil aktualizaci algoritmu (v SEO kruzích přezdíváné Mobilegeddon), který měl ve výsledcích vyhledávání na mobilních zařízeních výrazně zvýhodňovat tzv. **mobile friendly** weby. Ve skutečnosti ale k výraznějším turbulencím v mobilních výsledcích vyhledávání Google nedošlo. Je však zřejmé, že Google považuje použitelnost webu za důležitou, a to i v případě jeho mobilní verze.

Existuje několik způsobů, jak web převést do mobilní verze:

- responzivní design webu,
- mobilní a desktopová verze webu (dvě odlišné URL adresy),
- dynamicky načítaný obsah podle zařízení,
- mobilní aplikace, případně AMP.

V dnešní době se těší velké popularitě hlavně první dva přístupy. Dynamicky načítaný obsah podle zařízení je problematický zejména v tom, že se tak může na stejné URL adrese vyskytovat, v závislosti na podmínkách, různý obsah. To může vést k problémům s indexací tohoto obsahu.

4.1. Základní pravidla pro „mobile friendly“ web

Ať už optimalizujete zobrazení na mobilech jakýmkoli z uvedených způsobů, držte se základních pravidel:

- Mějte nakonfigurovaný viewport.
- Používejte dostatečně velké písmo.
- Mějte dotykové prvky umístěné v dostatečné vzdálenosti od sebe.
- Omezte používání pop-up oken.

Viewport je meta značka v hlavičce zdrojového kódu, která prohlížeči říká, zda a jak je web upravený pro mobilní zařízení. Jeho základní nastavení je:

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```



4.2. Mobilní verze webu

Jde o zvláštní verzi webu s vlastní URL adresou, která je přizpůsobena mobilním zařízením. Standardně je dostupná na subdoméně `m.example.com`. Název subdomény může být ovšem jakýkoliv (např. `mobil.example.com`). Stejně tak může být mobilní verze webu umístěna v subadresáři domény druhého řádu, např. `example.com/m/`. Nyní si řekneme několik pravidel, která je potřeba dodržet.

1. Dejte uživateli možnost prokliknout se na desktopovou verzi webu

Důležité je dát uživatelům tuto možnost na každé stránce mobilní verze webu. Nejvhodnějším místem je proto patička webu. Zde je vhodné umístit odkaz (např. „Klasická verze webu“ nebo „Verze pro desktop“), který povede na desktopovou verzi webu.

2. Dejte vyhledávačům vědět o alternativní verzi stránek

Abychom zamezili duplicitám, je nutné vyhledávačům říci o jednotlivých verzích webu. Do hlavičky zdrojového kódu stránky pro desktopy proto vložíme HTML značku `link` s atributem `alternate`, v případě mobilní verze webu pak atribut `canonical`. Tzn., že například na stránku `www.example.com` (verze stránky pro desktopy) umístíme značku:

```
<link rel="alternate" media="only screen and (max-width: 640px)"  
href="http://m.example.com" >
```

Naopak u stránky `m.example.com` (verze stránky pro mobilní zařízení) by měla značka v hlavičce zdrojového kódu vypadat takto:

```
<link rel="canonical" href="http://example.com">
```

Alternativní (mobilní verzi stránky) je vhodné také zanést do souboru `sitemap.xml` desktopové verze webu:

```
<?xml version="1.0" encoding="UTF-8"?>  
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9">  
<url>  
<loc>https://www.example.com/</loc>  
<xhtml:link rel="alternate"  
media="only screen and (max-width: 640px)"  
href="https://m.example.com/" />  
</url>  
</urlset>
```



3. Řekněte vyhledávačům, která stránka je ta hlavní

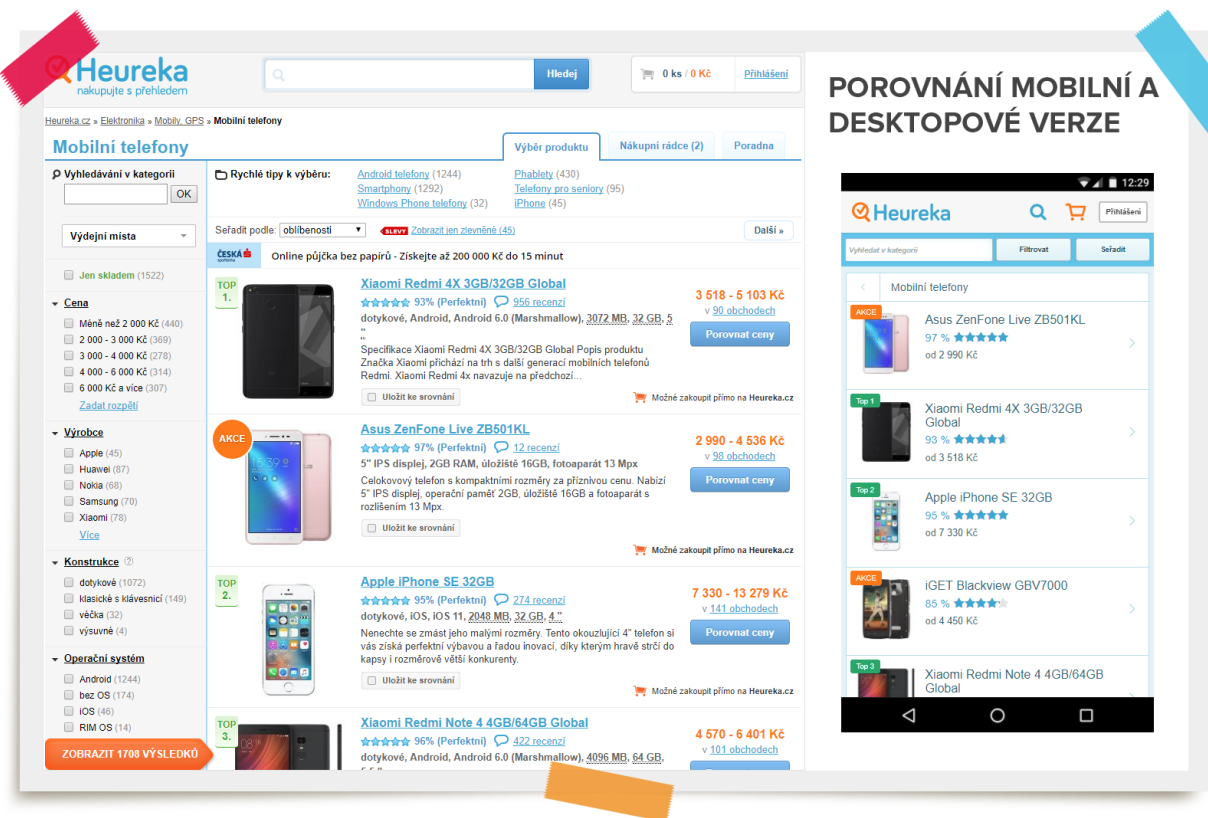
Mobilní verzi stránek webu je důležité kanonizovat. Vždy volíme jako kanonického zástupce desktopovou verzi (tedy jako hlavní verzi). Do obou verzí stránky (jak <http://m.example.com>, tak i <http://www.example.com>) vložíme do hlavičky zdrojového kódu značku `link` s atributem `canonical`:

```
<link rel="canonical" href="http://www.example.com/" >
```

Tím vyhledávačům řekneme, která verze stránky je ta hlavní.

4.3. Mobile-first indexing

V listopadu 2016 přišel Google na svém blogu pro webmastery s [vyjádřením](#), které optimalizaci pro mobily posouvá na další úroveň – Google začne nově posuzovat weby podle toho, jak se zobrazují na mobilních zařízeních. Na podzim 2017 byl Mobile-first Index spuštěn pro některé weby, v průběhu roku 2018 by pak těchto webů mělo rapidně přibývat. Co to znamená? Pokud máte responzivní web, pravděpodobně nemusíte řešit nic. Pokud je ale mobilní verze vašeho webu zkrácenou verzí desktopové verze po obsahové stránce, může se stát, že se váš web propadne v mobilním vyhledávání. Dbejte tedy na to, aby mobilní verze webu obsahovala všechny důležité a vysoce kvalitní obsahové prvky.



Obrázek 10. Příklad mobilní vs. desktopové verze webu heureka.cz.

Další věci jsou strukturovaná data, která by měla být rovněž identická pro všechny verze webu. Při Mobile-first indexingu, je třeba se řídit některými pravidly:

- Obsah by se měl shodovat na všech verzích webu napříč zařízeními (responzivní design).
- Strukturovaná data by se měla nasadit na všechny verze webu napříč zařízeními.
- Roboti vyhledávačů by měli být schopni procházet všechny verze webu (nastavení robots.txt).
- Web by měl být „Mobile Friendly“ (viz kapitola [4.1](#)).

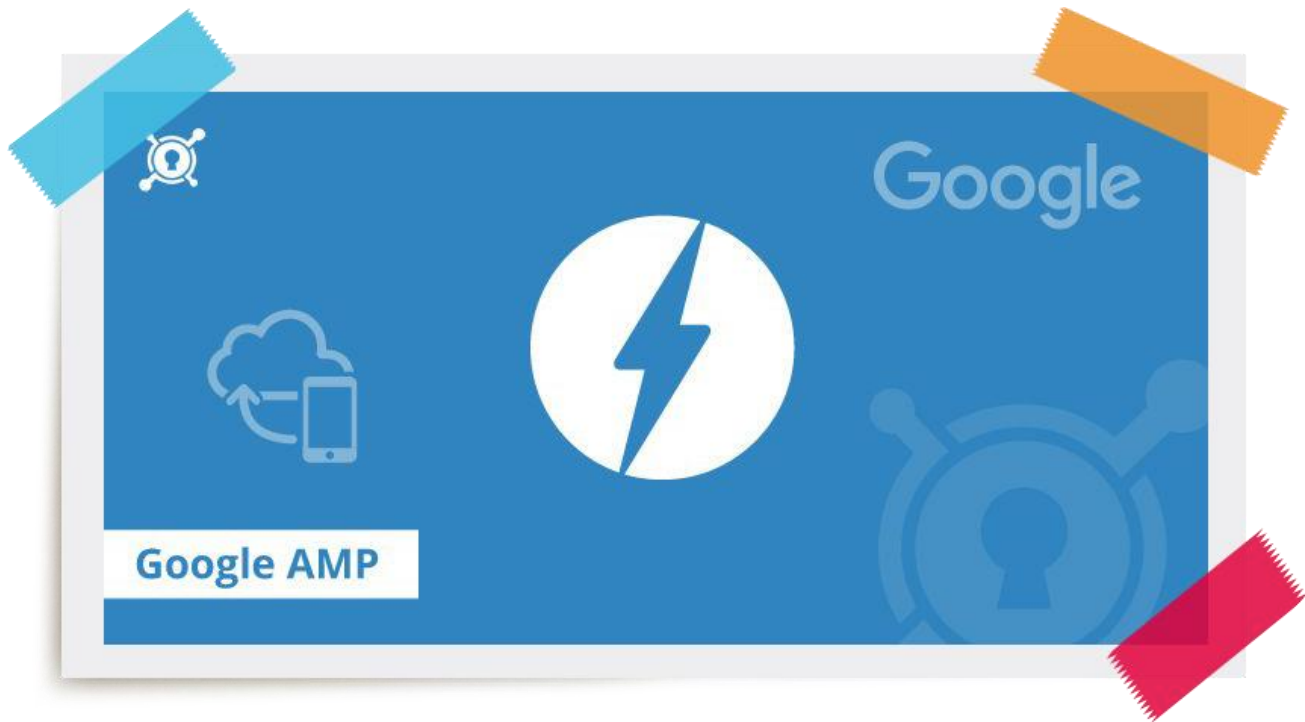
4.4. Accelerated Mobile Pages – AMP

Na ikonku blesku s nápisem AMP už si možná někteří z vás v mobilních výsledcích vyhledávání Google zvykli. Po kliknutí na takto označený výsledek vyhledávání se webová stránka zpravidla načte v řádech desetin sekund a uživatel na mobilu tak nemusí na vykreslení webu čekat dlouhé vteřiny. Za bleskurychlé načtení stránek může [technologie Accelerated Mobile Pages](#). O co se jedná a jak to může webu pomoci?

Webové stránky ve formátu Accelerated Mobile Pages (AMP) jsou pro české uživatele k dispozici zhruba od podzimu 2016. Jedná se o technologii od Google, která umožňuje téměř okamžité načtení webových stránek na mobilních zařízeních, a to i za pomalého připojení. Jak je to možné? Jednoduše. Díky třem věcem:

- **AMP HTML** – standard HTML kódu, který specifikuje, jaké značky (tagy) je a není možné v kódu používat. U některých standardních HTML značek, jako například `img` pro vložení obrázku, je navíc nutné použít pozměněnou AMP značku. V tomto případě `amp-img`.
- **AMP JS** – knihovna JavaScriptu, která umožňuje bleskové vykreslení stránky. Nadměrné používání JavaScriptu často načítání webových stránek brzdí. Proto je použití JavaScriptu ve formátu AMP omezeno na funkcionalitu této knihovny. Knihovna AMP JS je dostupná na [GitHubu](#).
- **AMP Cache** - třetí velmi důležitá součást je mezipaměť Google AMP Cache, do níž se obsah webu načítá. Při načítání webových stránek jsou tak HTML dokument, všechny obrázky a javascriptové soubory načteny téměř okamžitě, a to i díky použití protokolu HTTP/2.





Obrázek 11. Google AMP.

4.4.1 Jaké další věci musí AMP verze stránek splňovat?

- V hlavičce na ně musí odkazovat originální verze stránek ve formátu:

```
<link rel="amphtml" href="https://www.example.com/amp/stranka.html">
```

- V hlavičce AMP verze stránek musí být použita kanonizace na URL originální stránky:

```
<link rel="canonical" href="https://www.example.com/stranka.html">
```



4.4.2 Jak se AMP implementuje?

Jak už zaznělo, AMP kód musí splňovat některé náležitosti vycházející ze standardů AMP HTML. Nejjednodušší šablona kódu stránek v AMP může vypadat následovně:

```
<!doctype html>
<html amp lang="cs-CZ">
<head>
<meta charset="utf-8">
<link rel="canonical" href="http://www.example.com/stranka.html">
<meta name="viewport" content="width=device-width,minimum-
scale=1,initial-scale=1">
<style amp-boilerplate>
body{-webkit-animation:-amp-start 8s steps(1,end) 0s 1 normal both;-
moz-animation:-amp-start 8s steps(1,end) 0s 1 normal both;-ms-
animation:-amp-start 8s steps(1,end) 0s 1 normal both;animation:-amp-
start 8s steps(1,end) 0s 1 normal both}
@-webkit-keyframes -amp-
start{from{visibility:hidden}to{visibility:visible}}
@-moz-keyframes -amp-
start{from{visibility:hidden}to{visibility:visible}}
@-ms-keyframes -amp-
start{from{visibility:hidden}to{visibility:visible}}
@-o-keyframes -amp-
start{from{visibility:hidden}to{visibility:visible}}
@keyframes -amp-
start{from{visibility:hidden}to{visibility:visible}}</style>
<noscript><style amp-boilerplate>body{-webkit-animation:none;-moz-
animation:none;-ms-animation:none;animation:none}</style></noscript>
<script async src="https://cdn.ampproject.org/v0.js"></script>
</head>
<body>Obsah stránky</body>
</html>
```

Při tvorbě stránek je samozřejmě třeba dodržovat standardy AMP HTML a AMP JS.

Ti z vás, kterým běží web na **WordPressu**, mají k dispozici oficiální plugin, který je extrémně snadno použitelný a AMP díky němu na svém webu rozchodíte po pár klicích. Pokud používáte plugin YOAST SEO, nezapomeňte nainstalovat také plugin **Glue for Yoast SEO & AMP**. Ten umožní data z YOAST pluginu (titulky, tagy meta description apod.) zobrazovat také v AMP verzi stránek. Kromě WordPressu je AMP podporováno rovněž na dalších CMS jako **Drupal** nebo **Squarespace**.



4.4.3 Kde všude se moje AMP stránky mohou zobrazit?

Počet platform, které [AMP verze stránek podporují](#) ve svých rozhraních, není úplně velký, zato jsou velmi významné. Kromě Google jsou to třeba Twitter, Pinterest, LinkedIn a další. Nějaký čas s AMP experimentoval i Facebook, v současné době se ale AMP výsledky na jeho timeline neobjevují, což je možná dáno tím, že AMP konkuruje facebookovému formátu Instant Articles.

4.4.4 Testování a validace AMP

Protože AMP je technologie podporovaná Google, můžete chyby na AMP stránkách a počet zaindexovaných AMP stránek sledovat v nástroji Google Search Console.



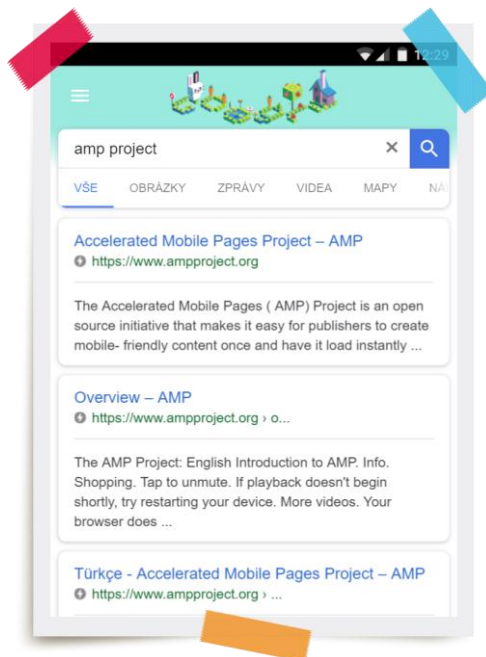
Obrázek 12. Validace a testování AMP.

4.4.5 Je technologie AMP vhodná pro všechny typy stránek?

AMP je technologie značně omezená funkcionalitou. Hodí se tedy spíše pro weby s malou mírou interakce uživatelů. Zpravidla se jedná o obsahové weby a většina významných světových mediálních institucí již AMP používá. Zmínit lze například The Guardian, CNN, The New York Times nebo BBC. Z významných českých médií jsme zatím AMP zaregistrovali pouze u serveru Seznam zprávy.

Jsou ovšem i výjimky. AMP používají také e-shopoví giganti eBay nebo Aliexpress. Vzhledem k náročnosti nasazení AMP na tyto typy stránek bych ale doporučil zvážit, jak velký přínos bude mít zrychlení stránek oproti vynaloženému úsilí.



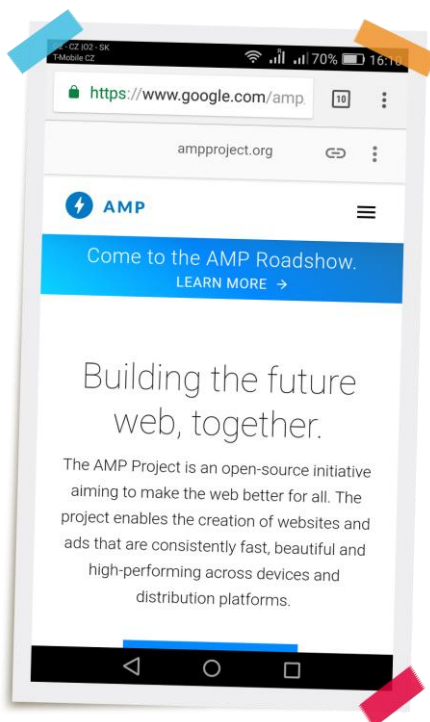


Obrázek 13. Ukázka AMP snippetů v Google SERP.

4.4.6 Zobrazení URL adresy AMP stránky

Zobrazená URL adresa AMP stránky v adresním řádku může být pro uživatele poněkud matoucí. Pokud uživatel na mobilním zařízení navštíví stránku AMP, dostane URL adresu na doméně Google.

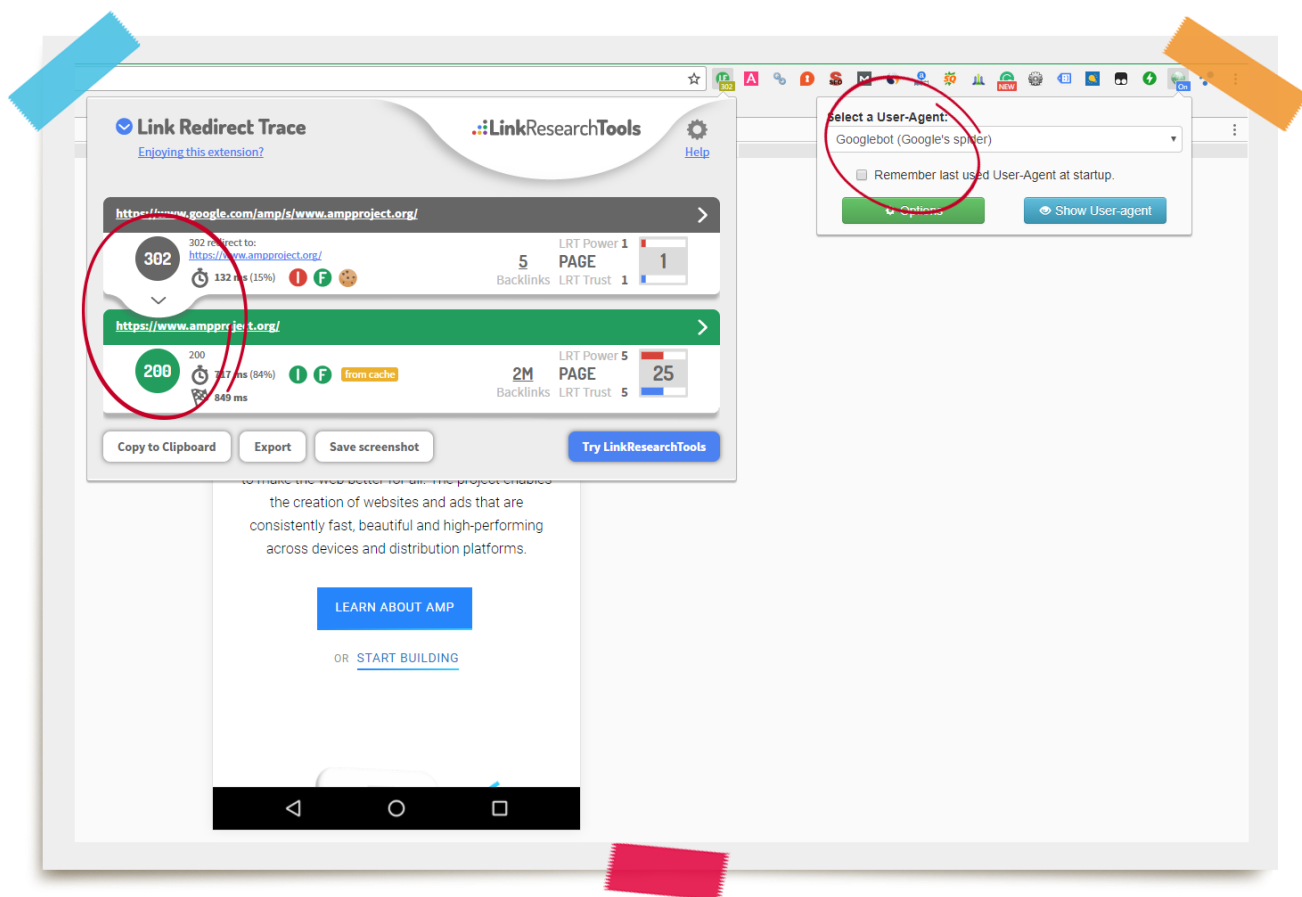
Při načtení stránky <https://www.ampproject.org/> s technologií AMP vidí uživatel v adresním řádku URL adresu: <https://www.google.com/amp/s/www.ampproject.org/>.



Obrázek 14. Zobrazení AMP stránky na mobilním zařízení.



Zkusili jsme ale nasimulovat přítomnost Googlebota a při návštěvě URL <https://www.google.com/amp/s/www.ampproject.org/> se adresa přesměrovala stavovým kódem 302 do původní verze stránky www.ampproject.org/, čímž se této stránce „předá link juice“. Stejně tak dojde k přesměrování při návštěvě stránky z desktopového zařízení, při návštěvě Bingbota nebo Slurp (Yahoo's Spider).



Obrázek 15. Návštěva AMP stránky při simulování Googlebota.



5. ROZŠÍŘENÉ VÝSLEDKY VYHLEDÁVÁNÍ

Přirozené výsledky vyhledávání na Google nejsou jenom o snippetech. Lze se setkat s velkým množstvím dalších formátů, které se souhrnně nazývají „rozšířené výsledky vyhledávání“. Nyní se podíváme, na které typy rozšířených výsledků vyhledávání můžeme na Google narazit, a řekneme si, jaké z nich mohou plynout výhody. Také si ukážeme, co dělat pro to, aby se zde vaše webové stránky mohly objevit.

Úvodem je třeba říct, co je to **snippet**. Jedná se o běžný výsledek vyhledávání, jak jej známe. V nejjednodušší formě **obsahuje titulek, URL adresu a popisek**.



Obrázek 16. Ukázka snippetu v Google SERP.

Odmyslíme-li si reklamní obsah, objevuje se v [SERPu](#) (Search Engine Result Page) [řada dalších formátů dat](#). Patří mezi ně mapy, karusely, grafické prvky nebo různé typy rozšířených informací. Jaké jsou jejich výhody a jak zajistit, aby se v nich zobrazovala také data z vašeho webu?

Rozšířené výsledky vyhledávání zobrazené v SERPu vždy odpovídají typologii dotazu a typu zařízení, které právě používáte (desktop, mobil, tablet). Hledáte kavárnu? Google vám nabídne mapku s podniky v okolí. Chcete si něco uvařit? Zadejte ingredience a Google odpoví receptem s hodnocením, dobou vaření a počtem kalorií. Sháníte informace o firmě? Google vám v pravém panelu přehledně zobrazí její kontaktní informace, logo, recenze a odkazy na web i sociální sítě (tzv. knowledge graph).



5.1. Jaké jsou výhody rozšířených výsledků vyhledávání?

Výhod objevíme hned několik a vzájemně spolu souvisí. Základem je to, že se od standardních výsledků vyhledávání vizuálně odlišují a poutají tak více pozornosti. Zde jsou ale ty hlavní:

- **Uživatelská přívětivost** – rozšířené výsledky vyhledávání jdou výrazně vstříc uživateli. Ať už se jedná o mapku, která se vám nabídne, když bloudíte po Vinohradech a hledáte pivovar, nebo o rozšířené informace o kaloriích a době přípravy jídla, když přemýšlíte o vhodném receptu.
- **Více prostoru ve výsledcích vyhledávání** – díky zobrazení v rozšířených výsledcích vyhledávání můžete docílit toho, že se na jedné stránce SERPu zobrazí vaše stránky vícekrát. Například ve snippetu, v mapce a v obrázkovém carouselu. To vede k vyšší míře prokliku z výsledků vyhledávání na web (CTR).
- **Povědomí o značce a důvěryhodnost** – důležitý samozřejmě není jen proklik na stránku, ale také to, jak je značka vidět. S rozšířenými výsledky vyhledávání toho docílíte a zároveň posílíte důvěryhodnost, protože součástí rozšířených výsledků je často také více doplňkových informací (nebojte se uveřejňovat hodnocení zákazníků, cenu zboží apod., pokud je to zrovna vaší výhodou nad konkurencí).

5.2. Typy rozšířených výsledků vyhledávání v SERPu

V současné době můžeme v SERPu Google narazit na celou řadu rozšířených výsledků vyhledávání. Google si je vědom, že ne na každý dotaz je nejlepší výsledek standardní snippet, proto se snaží velmi často odpověď přizpůsobit dotazu. Google dokonce v jednu dobu také experimentoval s tzv. [Zero-Result SERP](#). Jednalo se spíše o experiment Googlu, kdy na určité dotazy, kdy si byl jist jednou jedinou odpovědí, zobrazil rovnou výsledek v SERPu bez dalších alternativ. Ty byly dostupné až po rozkliknutí tlačítka „Show all results“. Od toho ovšem záhy upustil a nyní se již s tímto výsledkem v Google SERP neseťkáme.

Pojďme se ovšem vrátit k rozšířeným výsledkům vyhledávání a ukázat si ty základní, na které můžeme v českém prostředí Google SERP narazit.

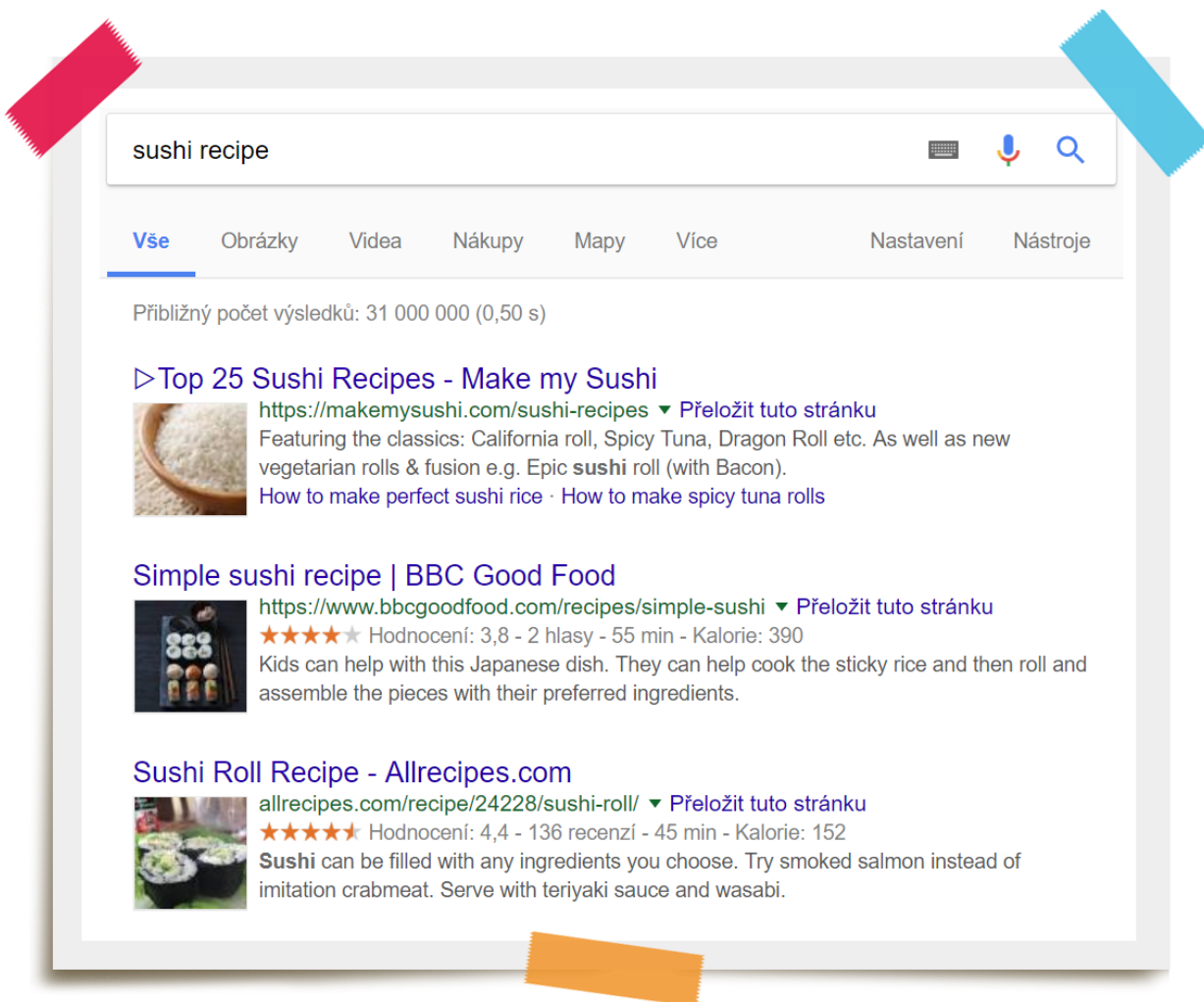


5.2.1 Rich Snippets

První a pravděpodobně nejčastější rozšířený výsledek vyhledávání je Rich Snippets. Jedná se o strukturované úryvky, které se začaly v SERPu Google zobrazovat v roce 2012. Jde o rozšíření založené na formátu strukturovaných dat podle specifikace [Schema.org](https://schema.org). Obecně se doporučuje strukturovaná data nasazovat předně na typy podstránek, pro které již Rich Snippets existují:

- produkty,
- recenze,
- události,
- recepty,
- kurzy.

Předpokladem jsou indexované stránky, které obsahují strukturovaná data. Ani správně nasazená strukturovaná data Google ovšem nemusí ve formátu Rich Snippets zobrazit.



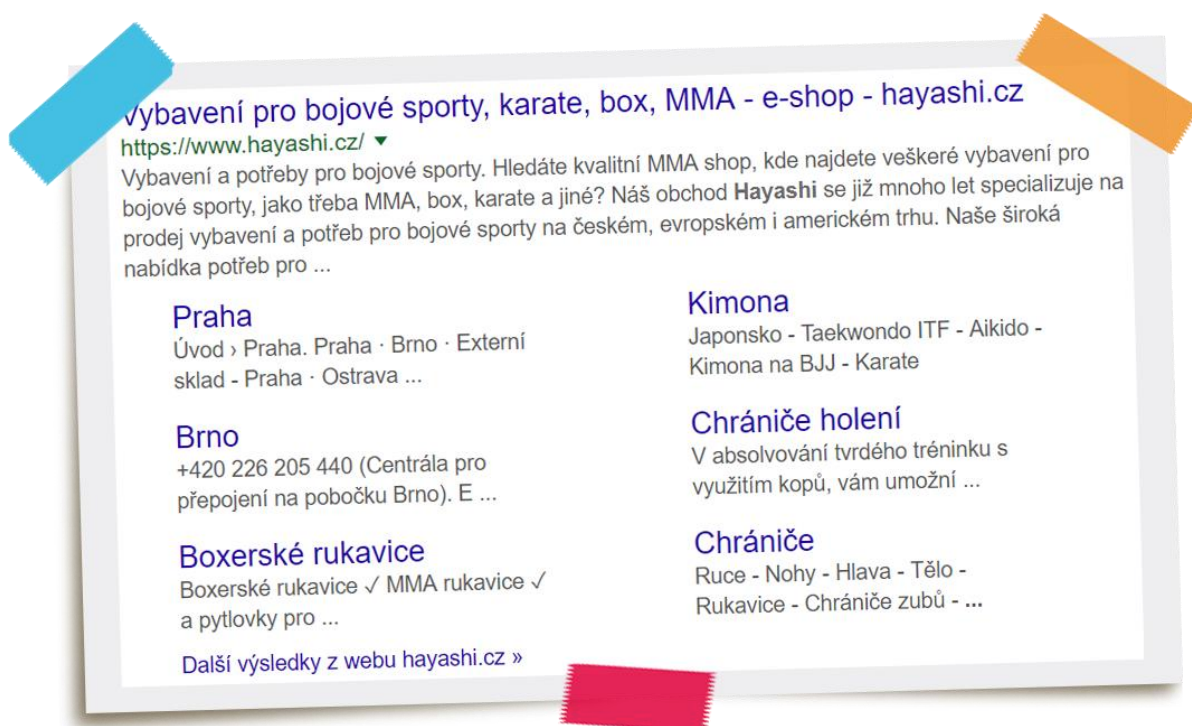
Obrázek 17. Rozšířené výsledky vyhledávání v Google - Rich Snippets.



5.2.2 Sitelink Searchbox & Sitelinks

U větších webů, jako jsou e-shopy nebo velké zpravodajské servery, se na určité typy dotazů zobrazuje přímo ve snippetu také vyhledávací políčko **Sitelink Searchbox**.

Pod snippetem se v některých případech objevují rovněž odkazy na podstránky. Toto rozšíření se nazývá Sitelinks a v současné době **není zcela zřejmé, jak tento výpis ovlivnit**. Dříve to bylo možné v Google Search Console, ale tato funkcionalita byla ukončena.



Obrázek 18. Rozšířené výsledky vyhledávání v Google - Site links.

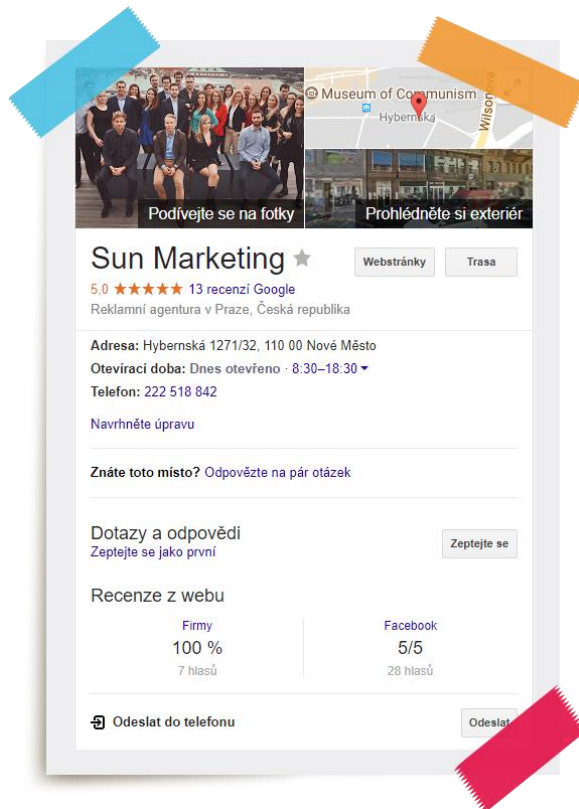
5.2.3 Knowledge Graph panel

Jedná se o **informační box v pravé části SERPu**. K zobrazování dat v tomto rozšíření Google využívá svojí znalostní bázi Knowledge Graph (viz níže). Cílem tohoto výsledku vyhledávání je agregovat základní informace o dotazu na „jedno místo“. Pro zobrazení zde jsou důležité zejména tyto zdroje:

- profil na Google Moje Firma,
- zápis na Wikipedii,
- nasazená strukturovaná data na webu.

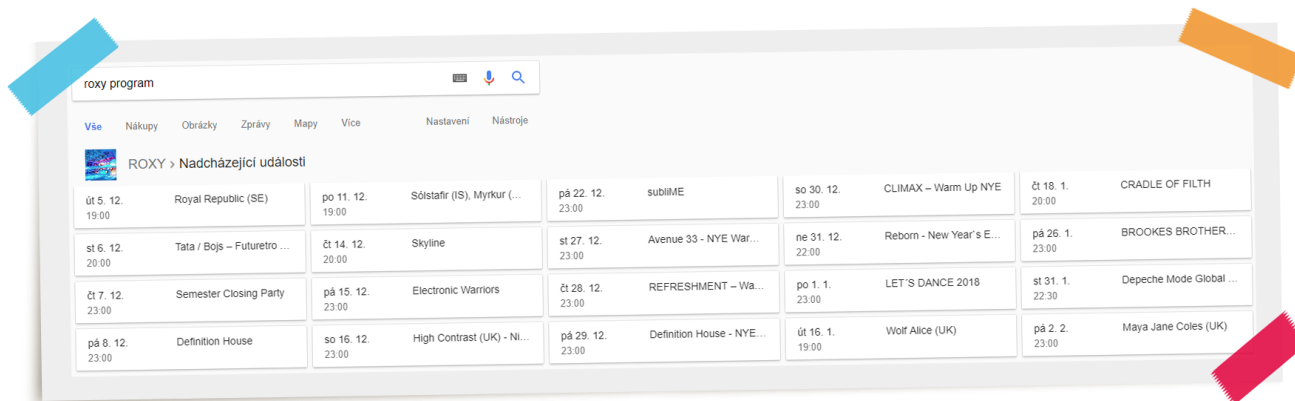
Google na tomto místě velmi často zobrazuje další informace (nejčastěji hodnocení firmy) ze služeb třetí strany, například Facebook, Firmy.cz apod.





Obrázek 19. Rozšířené výsledky vyhledávání v Google - Knowledge Graph.

Odpověď na to, proč dokáže Google vždy nabídnout tak přesné a relevantní výsledky vyhledávání, spočívá v jeho **znalostní bázi nazývané Google Knowledge Graph**. Základem Knowledge Graphu jsou entity, které si Google vytváří ze získaných informací a vazeb mezi nimi. Google spustil Knowledge Graph v roce 2012 s cílem přiblížit se sémantickému vyhledávání a poskytovat přímé odpovědi na dotazy, ne pouze odkazy na stránky jako v minulosti. Ukázkou takové odpovědi může být například SERP, který Google zobrazí na dotaz „Roxy program“. Kdyby byl Google člověk, jeho myšlenkový postup by se dal shrnout asi nějak takto: „Aha, tak ty chceš asi vědět, jaké události nabídne klub Roxy v následující době. Jsou to tyto...“.



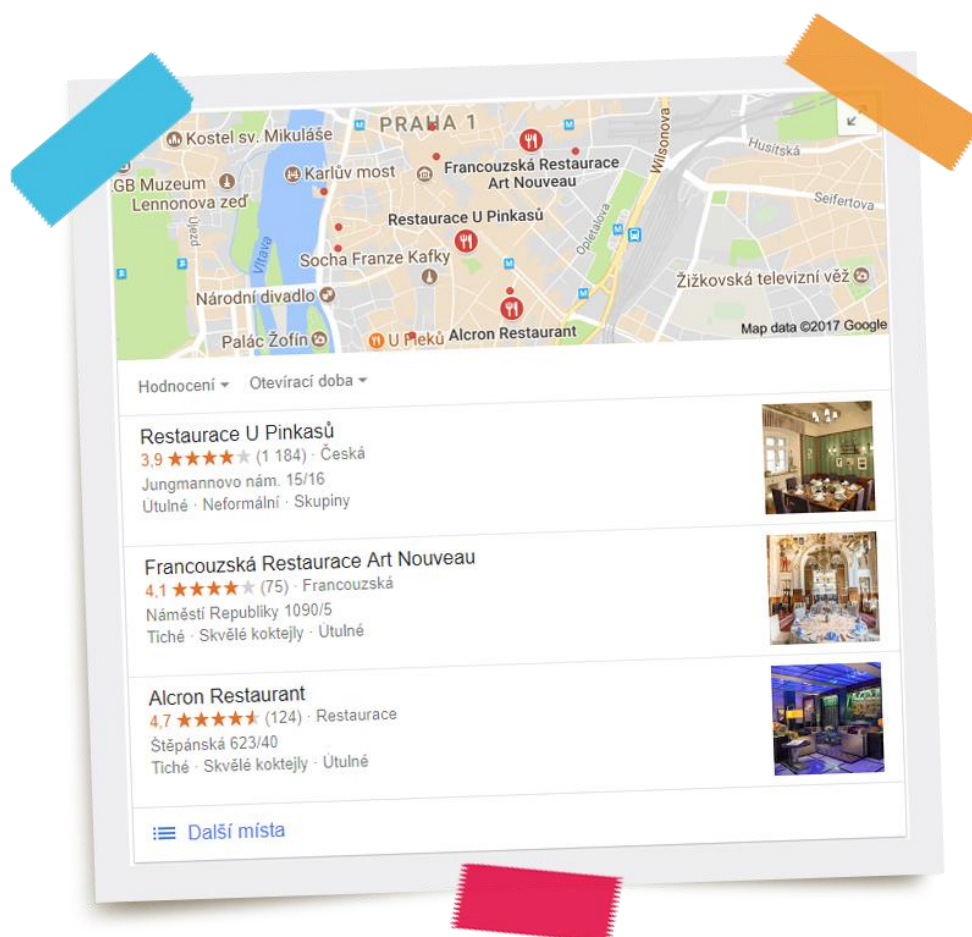
Obrázek 20. Rozšířené výsledky vyhledávání v Google – události.



5.2.4 Local pack

Má podobu **mapky vnořené mezi standardní výsledky vyhledávání**. Na mapce jsou zpravidla ukázány 2-3 provozovny relevantní k dotazu. Často se s takovým výsledkem setkáváme při zadání dotazu, který obsahuje lokaci (město, část obce nebo ulici). Není to ovšem podmínkou. To jaké výsledky se v Local packu zobrazí, hraje velkou roli lokalita, ze které právě uživatel vyhledává. Dalšími faktory pro zobrazení zde jsou:

- informace uvedené na profilu Google Moje Firma,
- počet a kvalita hodnocení od uživatelů na Google Moje Firma,
- historie zápisu,
- síla a relevance webu firmy,
- další data, která Google o firmě posbíral.



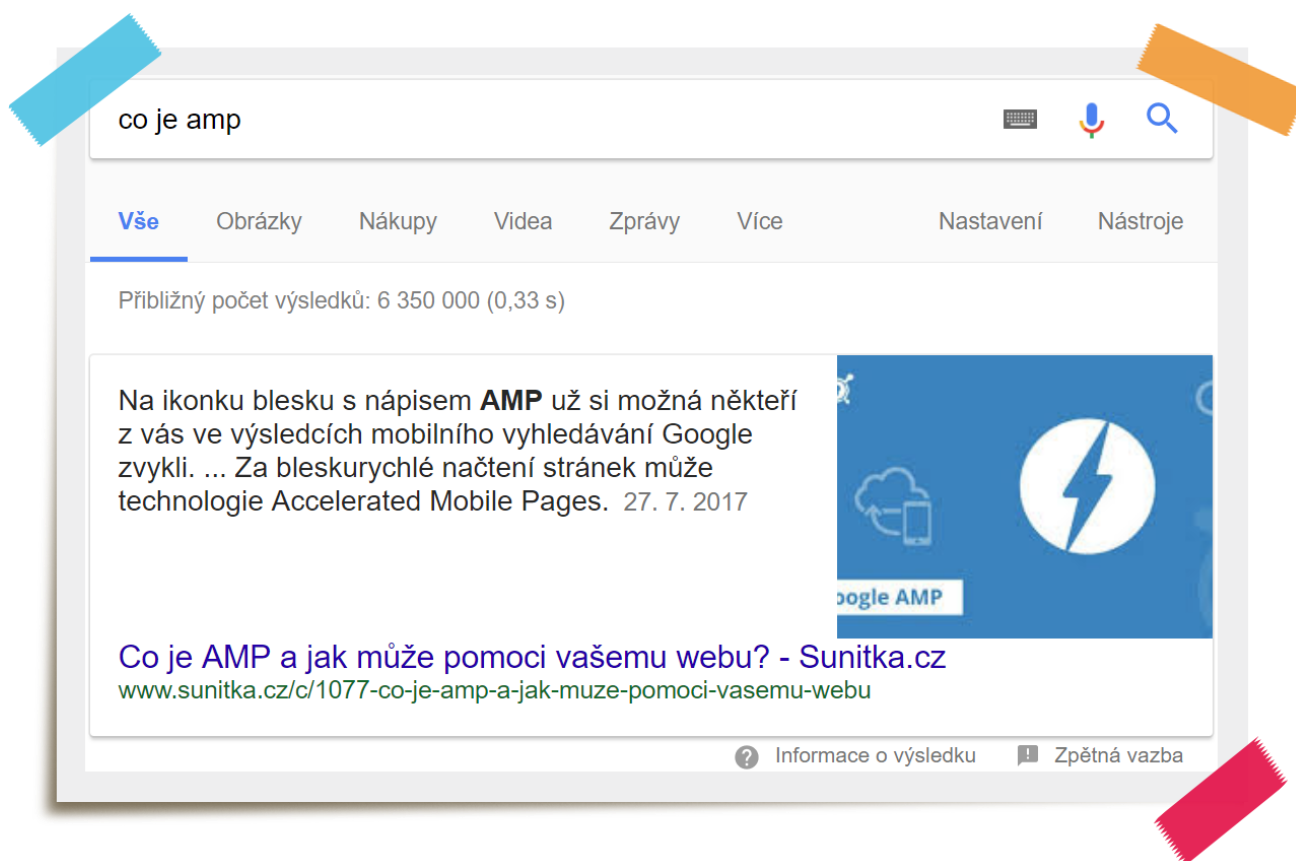
Obrázek 21. Rozšířené výsledky vyhledávání v Google - Local Pack.



5.2.5 Featured Snippet

Featured Snippet je box, který se nachází na tzv. nulté pozici, tedy mezi placenou reklamou (pokud se zobrazí) a prvním organickým výsledkem vyhledávání. Nejčastěji se objevuje jako **slovníková odpověď na otázku**. Hlavní faktory pro zobrazení citace z webu ve Featured Snippetu jsou:

- Důvěryhodnost webu.
- Obsah tvořený s cílem odpovídat na otázky (např. slovník pojmů, FAQ).



Obrázek 22. Rozšířené výsledky vyhledávání v Google - Featured Snippet.



5.2.6 Image box

Jedná se o box s obrázky, které jsou **kontextově blízké k dotazu**.



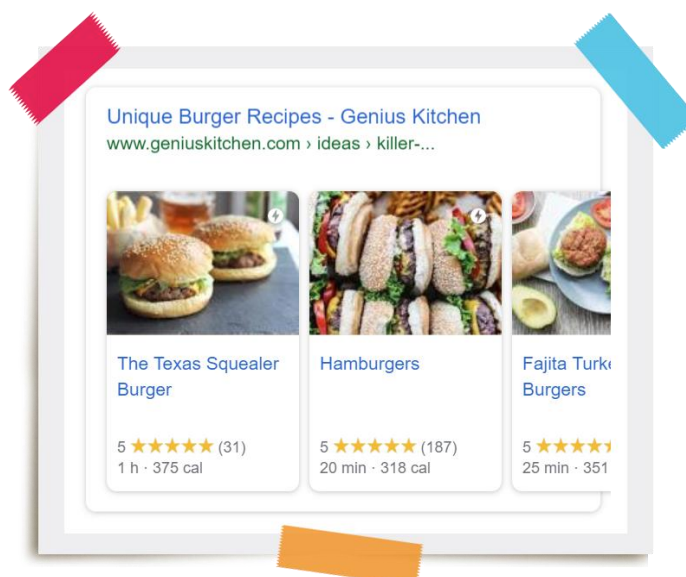
Obrázek 23. Rozšířené výsledky vyhledávání v Google - Obrázky pro dotaz damske sportovni boty.

Pro zobrazení se v Image boxu platí podmínky pro optimalizaci obrázků, tj. správně vyplněný alternativní popis obrázku (parametr `alt`) a název souboru obrázku. Dále textový obsah okolo obrázku a síla spolu s relevancí stránky, kde je obrázek umístěn vůči dotazu.

5.2.7 AMP carousel

AMP carousel představuje formát, který se zobrazuje **výhradně na mobilních zařízeních**. Jedná se o carousel příspěvků odpovídající dotazu ve formátu AMP. Podmínky pro zobrazení v AMP carouselu:

- Nasazené AMP na webu.
- Nasazená strukturovaná data na webu.

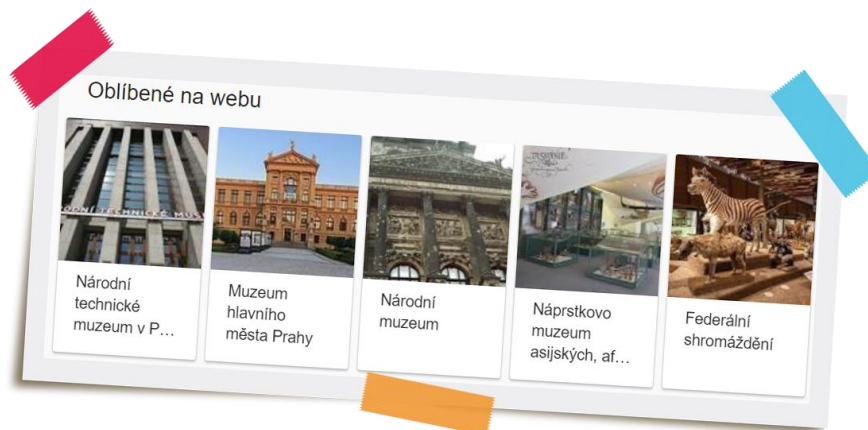


Obrázek 24. Rozšířené výsledky vyhledávání v Google - AMP carousel.



5.2.8 Oblíbené na webu

Oblíbené na webu je carousel, který se ve výsledcích vyhledávání nejčastěji objevuje, pokud se dotaz ptá na nějaký **výčet objektů**. Například plemena psů, muzea v Praze, němečtí vědci apod.

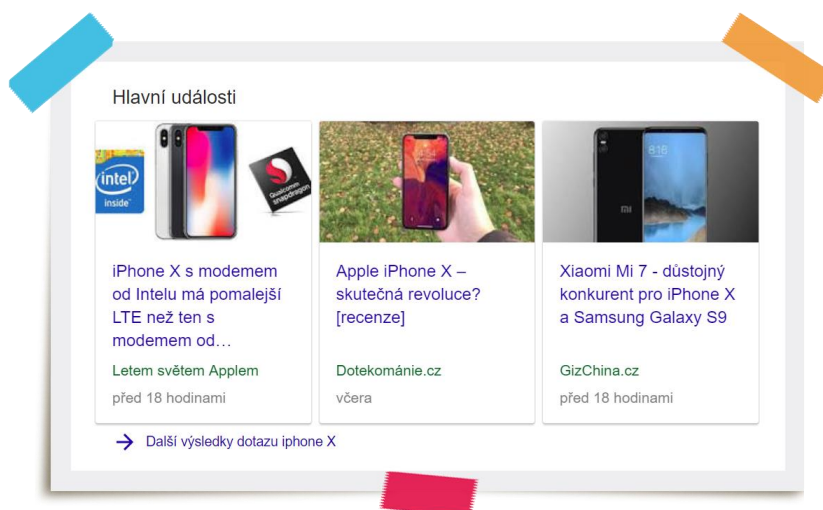


Obrázek 25. Rozšířené výsledky vyhledávání v Google - Oblíbené na webu.

5.2.9 Hlavní události

V současné době má toto rozšíření podobu **carouselu s kartami pro jednotlivé zprávy**. Podle všeho jsou důležité zejména tyto atributy zpráv:

- aktuálnost,
- důvěryhodnost serveru,
- důvěryhodnost zprávy (Google používá [Fact Check](#)),
- nasazená strukturovaná data na webu.

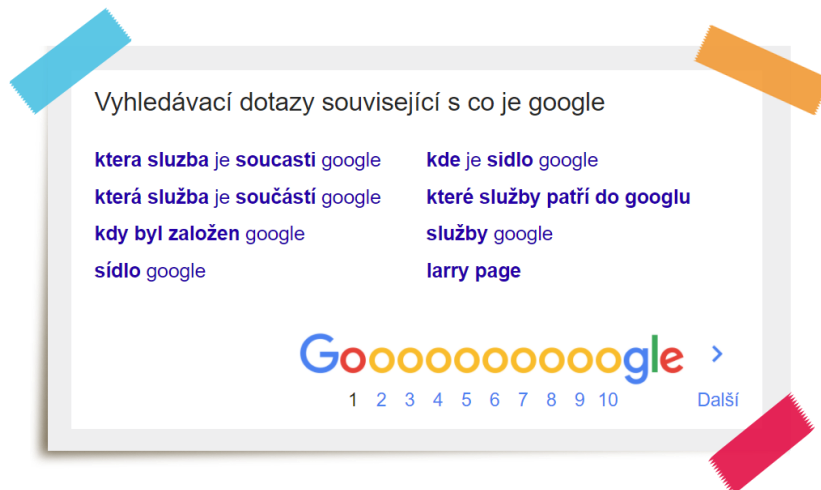


Obrázek 26. Rozšířené výsledky vyhledávání v Google - Hlavní události.



5.2.10 Related Search

Related Search je seznam kontextově blízkých dotazů. Tento výčet se zobrazuje zpravidla mezi posledním přirozeným výsledkem vyhledávání (případně reklamou) na stránce a stránkováním.



Obrázek 27. Rozšířené výsledky vyhledávání v Google - Related search.

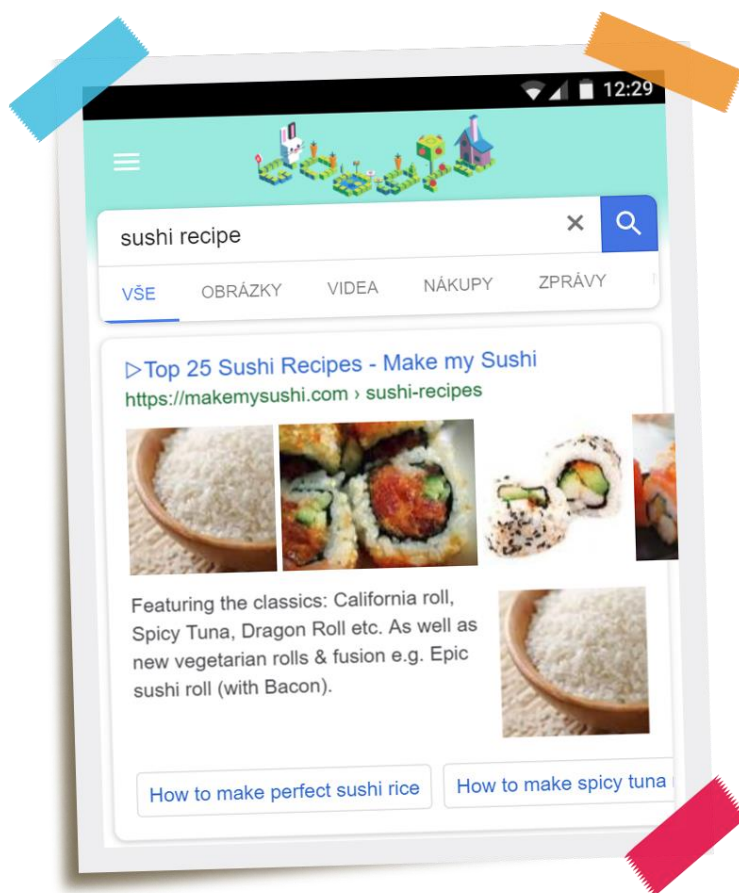
5.2.11 AMP (Accelerated Mobile Pages)

U některých výsledků vyhledávání se na mobilu zobrazuje šedá ikonka blesku. Tato ikonka upozorňuje na to, že po prokliku se dostanete na webovou stránku běžící v režimu AMP, tedy že bude její načtení velmi rychlé. Podrobnější informace o technologii AMP popisujeme v kapitole [4.4](#). Podmínkou pro zobrazení snippetu s tímto symbolem je nasazená AMP verze stránek.



5.2.12 Rich Cards

Rich Cards vznikly z potřeby zobrazovat na mobilu výsledky vyhledávání přehledně a na malé ploše. Jsou to v podstatě Rich Snippets ve formátu karet pro mobilní zobrazení.



Obrázek 28. Rozšířené výsledky vyhledávání v Google - Rich Cards.

Kromě těchto výsledků můžeme narazit přímo ve vyhledávání i na některé miniaplikace, jako jsou Předpověď počasí, Kalkulačka, Lokalizace polohy, Překladač, Konverze měn, Burza, Dopravní spojení, Sportovní informace a jiné.

O zastoupení rozšířených výsledků vyhledávání v českém SERPu Google vyčerpávajícím způsobem hovoří [Filip Podstavec ve svém článku na Lupa.cz](#).



5.3. Jak dostat stránku do rozšířených výsledků vyhledávání?

Čím více informací a kontextu k webovým stránkám Google zná, tím pravděpodobněji se v rozšířených výsledcích vyhledávání zobrazí. Na zobrazení stránky v rozšířených výsledcích mají vliv následující faktory:

- Založení profilu na [Google Moje Firma](#).
- Založení stránky profilu firmy na Wikipedii (pokud to dává smysl).
- Nasazení strukturovaných dat podle specifikace [Schema.org](#) (viz níže).
- Specifikace meta dat, [Open Graph meta dat](#), [Twitter Cards](#).
- Zápisy v relevantních a důvěryhodných zdrojích, které už mají strukturovaná data nasazena: katalogy, oborové katalogy, katalogy událostí apod.

5.4. Jak ze svých snippetů udělám Rich Snippets/Rich Cards?

Rich Snippets a Rich Cards jsou jen vizuální interpretací strukturovaných dat podle specifikace [Schema.org](#), která jsou přítomná na vašem webu. Máte-li správně a smysluplně nasazená strukturovaná data, je dost pravděpodobné, že se ve vašich snippetech budou zobrazovat i rozšířené údaje, jako jsou hvězdičky hodnocení, cena nebo obrázek produktu. Strukturovaná data můžete nasadit čtyřmi různými metodami:

- **Formát RDF-a** – atributy přímo v HTML.
- **Formát Microdata** – atributy přímo v HTML.
- **Formát JSON-LD** – úsek kódu ve formátu JSON, který není závislý na jednotlivých částech kódu HTML.
- **Manuálně označená strukturovaná data** v Google Search Console pomocí [zvýrazňovače dat](#).

Z těchto zmíněných řešení v současné době Google nejvíce podporuje a doporučuje [implementaci ve formátu JSON-LD](#). Výhodou zmiňovaného formátu je, že syntaxe není nijak složitá a zvládne ji vstřebat i člověk, který se v HTML kódu běžně nevyzná. JSON-LD je na HTML struktuře nezávislé. Pouze popisuje, co se na stránce nachází. Příklad zápisu strukturovaných dat pro produkt může vypadat například takto. Do kódu jsou doplněny komentáře vždy za dvěma lomítky.




```

<script type="application/ld+json">
{
  "@context": "http://schema.org/",
  "@type": "Product",
  "name": "3D POVLEČENÍ - ANTIK", // Název produktu většinou nadpis
stránky, HTML značka H1.
  "url": "https://superlevnezbozi.cz/3d-povlezeni-antik/",
  "image": [
    "https://superlevnezbozi.cz/data/cache/thumb_1000-1000-
12/products/761/1500304794/00008992.jpg",
    "https://superlevnezbozi.cz/data/cache/thumb_1000-1000-
12/products/761/1500304808/00009272.jpg"
  ], // Pole obrázků. Jednotlivé obrázky jsou oddělené čárkou. Za
posledním čárka není.
  "description": "3D povlečení ANTIK ve velikosti přikrývky 140x200
a polštáře 70x90 cm. Velmi příjemné a pohodlné povlečení s 3D
efektem.", // Lze vzít první odstavec popisku nebo obsah meta tagu
description.
  "brand": {
    "@type": "Thing",
    "name": "M&K textil" // Značka výrobce.
  },
  "offers": {
    "@type": "Offer",
    "priceCurrency": "CZK",
    "price": "259",
    "priceValidUntil": "2099-01-01",
    "itemCondition": "NewCondition", // Stav zboží, předvolby
jsou dostupné na schema.org/UsedCondition.
    "availability": "InStock", // Dostupnost - předvolby jsou
dostupné na schema.org/InStock.
    "category": "výprodej", // Nadřazená kategorie.
    "seller": {
      "@type": "Organization",
      "name": "M&K Home Textile"
    }
  },
  "aggregateRating": {
    "@type": "AggregateRating",
    "ratingValue": "3.5", // Průměrné hodnocení produktu.
    "reviewCount": "11", // Počet hodnocení.
    "bestRating": "5",
    "worstRating": "1"
  }
}
</script>

```



5.4.1 Strukturovaná data ve formátu JSON-LD

Jak už bylo řečeno, Google doporučuje implementovat strukturovaná data prostřednictvím formátu JSON-LD podle specifikace [Schema.org](https://schema.org). Syntaxe je velmi snadno pochopitelná a některé nejčastější typy strukturovaných dat si ukážeme na příkladech.

5.4.2 Strukturovaná data WebSite

Strukturovaná data typu WebSite popisují webové stránky. Mezi nejčastější vlastnosti, které se v tomto typu strukturovaných dat popisují, patří název webové stránky, její URL adresa a URL adresa interního vyhledávání na webu. Právě fragment vlastnosti URL adresy interního vyhledávání na webové stránce může mít užitečný efekt, že se ve výsledcích může zobrazit rozšířený snippet se Sitelink Searchboxem (viz kapitola 5.2.2). Zde je příklad zápisu strukturovaných dat typu WebSite ve formátu JSON-LD:

```
<script type="application/ld+json">
{
  "@context": "http://schema.org",
  "@type": "WebSite",
  "name": "Název webu",
  "alternateName": "Alternativní název webu",
  "url": "http://example.com/",
  "potentialAction": {
    "@type": "SearchAction",
    "target": "http://example.com/search?q={query}",
    "query-input": "required name=query"
  }
}
</script>
```



5.4.3 Strukturovaná data Breadcrumb

```
<script type="application/ld+json">
{
  "@context": "http://schema.org",
  "@type": "BreadcrumbList",
  "itemListElement": [{
    "@type": "ListItem",
    "position": 1,
    "item": {
      "@id": "https://example.com/notebooky",
      "name": "Notebooky",
      "image": "http://example.com/images/icon-notebook.png"
    }
  }, {
    "@type": "ListItem",
    "position": 2,
    "item": {
      "@id": "https://example.com/ultrabooky",
      "name": "Ultrabooky",
      "image": "http://example.com/images/icon-ultrabooky.png"
    }
  }, {
    "@type": "ListItem",
    "position": 3,
    "item": {
      "@id": "https://example.com/books/authors/annleckie",
      "name": "Ann Leckie",
      "image": "http://example.com/images/author-leckie-ann.png"
    }
  }
  ]
}
```



5.4.4 Strukturovaná data Organization

Strukturovaná data typu Organization popisují organizaci/firmu, jíž stránky patří. Mezi nejčastější vlastnosti, které se v tomto typu strukturovaných dat popisují, se řadí název organizace, URL adresa jejího webu, kontaktní údaje organizace a odkazy na další internetové prezentace společnosti (profily na sociálních sítích, zápisy v katalogích apod.).
Příklad zápisu strukturovaných dat typu Organization ve formátu JSON-LD:

```
<script type="application/ld+json">
{
  "@context": "http://schema.org",
  "@type": "Organization",
  "name": "Example s.r.o.",
  "telephone": "+999 999 999 999 999",
  "email": "info@example.com",
  "url": "http://example.com/",
  "sameAs":
["https://www.firmy.cz/detail/example.html", "https://www.facebook.com
/example/", "https://najisto.centrum.cz/22422426/example/", "http://top
kontakt.idnes.cz/f/example/", "https://www.zlatestranky.cz/profil/exam
ple"]
}
</script>
```



5.4.5 Strukturovaná data LocalBusiness

Strukturovaná data typu LocalBusiness slouží k popsání vlastností kamenné prodejny. Nejčastějšími vlastnostmi zde jsou kontaktní údaje, otevírací doba, případně cenové rozpětí dané provozovny. Příklad zápisu strukturovaných dat typu LocalBusiness ve formátu JSON-LD:

```
<script type="application/ld+json">
{
  "@context": "http://www.schema.org",
  "@type": "Store",
  "name": " Example s.r.o.",
  "url": "http://www.example.com/pobocka-praha",
  "telephone": "+999 999 999 999 999",
  "email": "info@example.com",
  "sameAs": [
    " https://www.firmy.cz/detail/example.html ",
    "https://www.facebook.com/example/"
  ],
  "logo": " http://www.example.com/logo.jpg",
  "image": "http://www.example.com/example.jpg",
  "description": "Příklad je po anglicky example.",
  "address": {
    "@type": "PostalAddress",
    "streetAddress": "Examplová 1, Praha 99 - Example",
    "addressLocality": "Praha",
    "postalCode": "999 99",
    "addressCountry": "Česká Republika"
  },
  "hasMap":
  "https://www.google.cz/maps/place/50%C2%B882920983022'55.4%22N+14%C2%
B029'10.6%22E/@50.0487214,14.4840973,17z/data ",
  "geo": {
    "@type": "GeoCoordinates",
    "latitude": "99.99",
    "longitude": "99.99"
  },
  "openingHours": "Mo, Tu, Th 08:00-17:00 We 09:00-18:00 Fr 08:00-
16:00",
  "priceRange": "100 - 150 000 CZK"
}
</script>
```



5.4.6 Strukturovaná data Product

Jedním z nejčastějších typů strukturovaných dat, se kterými se můžete setkat v SERPu Google, jsou strukturovaná data typu Product. Slouží k popisu charakteristiky produktu. Nejčastější vlastnosti zde představuje název a typ produktu, jeho cena, dostupnost skladem nebo jeho hodnocení od uživatelů. Příklad zápisu pro Product:

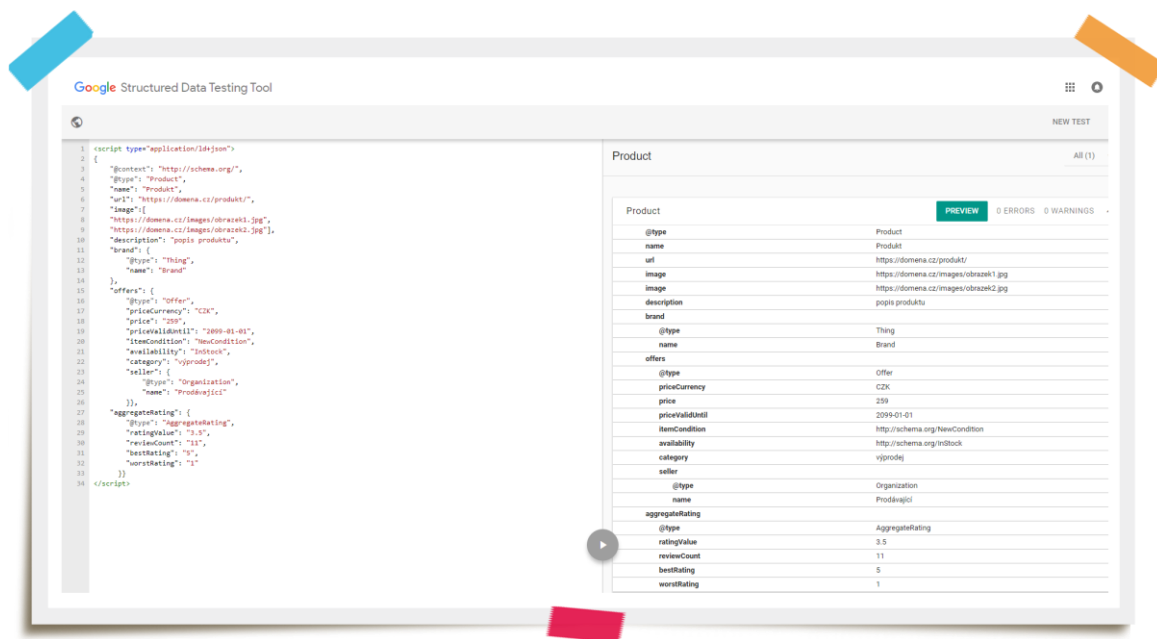
```
<script type='application/ld+json'>
{
  "@context": "http://www.schema.org",
  "@type": "product",
  "brand": "Example",
  "logo": "http://www.example.com/logo.jpg",
  "name": "Example example",
  "category": "Example",
  "image": "http://www.example.com/example.jpg",
  "description": " Příklad je po anglicky example.",
  "aggregateRating": {
    "@type": "aggregateRating",
    "ratingValue": "5",
    "reviewCount": "99"
  }
}
</script>
```

Existuje celá řada strukturovaných dat, zde jsme si ukázali opravdu ty nejdůležitější, jejich základní výčet a typy, které Google zcela jistě podporuje, je na stránce [Search Gallery Google](#). Pro kompletní výčet strukturovaných dat doporučujeme navštívit [Schema.org](#).



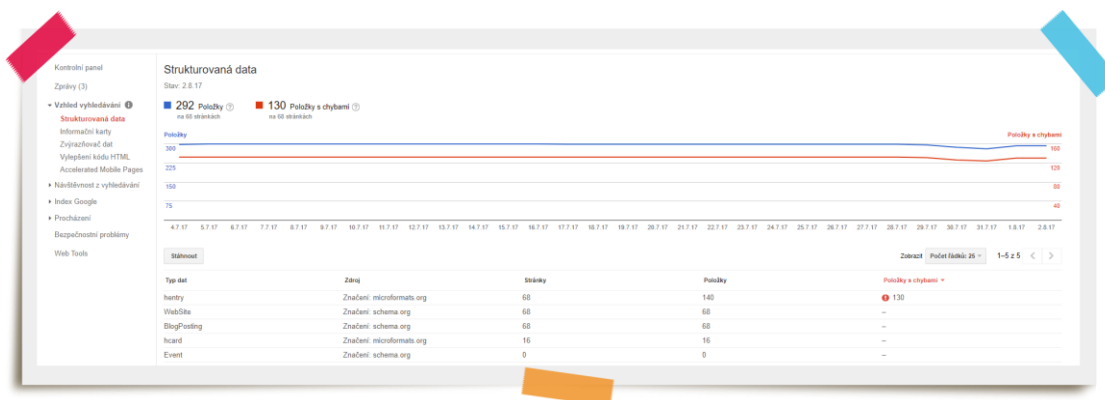
5.5. Jak strukturovaná data ověřit?

K ověřování strukturovaných dat slouží [Nástroj pro testování strukturovaných dat](#) od Google. Nástroj slouží zejména ke dvěma účelům, a to k ladění vznikajícího kódu pro strukturovaná data (může být JSON-LD nebo Microdata) a k zjištění přítomnosti strukturovaných dat na stránce. Na vstupu se tak může objevit buď URL adresa, na které se nástroj snaží strukturovaná data nalézt, nebo úryvek kódu, který nástroj validuje, popřípadě zjišťuje chyby.



Obrázek 29. Nástroj na testování strukturovaných dat.

Ke kontrole strukturovaných dat, které má Google zaindexované, lze využít přehled *Vzhled vyhledávání* > *Strukturovaná data*, v nástroji [Google Search Console](#). V přehledném grafu je zde znázorněn jak počet zaindexovaných stránek se strukturovanými daty, tak počet položek s chybami v syntaxi strukturovaných dat.



Obrázek 30. Strukturovaná data v Google Search Console.



ZÁVĚR

Pokud jste dočetli až sem, získali jste základy pro návrh úspěšného webu, který vyhledávače hladce projdou, budou schopny najít důležitý obsah, porozumět mu a nabídnout jej uživatelům, kteří jej hledají.

I když se na první pohled může zdát, že přestává dávat smysl řešit při optimalizaci technické aspekty webu s čím dál tím pokročilejšími roboty a algoritmy vyhledávačů, publikace ukazuje opak. A to především díky příchodu mnoha nových technologií za poslední roky. Můžeme jmenovat strukturovaná data, jejichž prostřednictvím se vyhledávače snaží lépe pochopit obsah webu. Dramatický nárůst využívání mobilních zařízení, vzrůstající trend použití dynamických prvků webu a použití JavaScriptu.

Technické SEO je v dnešní době disciplína sama o sobě. Do hloubky se touto tematikou zabývají především pokročilejší SEO specialisté často s programátorskou či kodéřskou zkušeností. Do jisté míry tak suplují práci vývojářů. Přestože tato publikace nepokrývá všechny.

Benefity plynoucí ze zvládnutého technického SEO jsou důležité jak pro rozsáhlé a komplikované weby jako e-shopy, tak pro malé firemní prezentace či třeba osobní blog.

Přejeme hodně úspěchů na cestě vstříc vyhledávačům!

