

Standard Unicode a UTF-8

Unicode je univerzální znaková sada, která definuje unikátní číslo (tzv. **Code Point**) pro každý znak bez ohledu na platformu, program nebo jazyk. **UTF-8** je pak nejrozšířenější způsob, jakým jsou tato čísla (znaky) efektivně zapsána do [binární podoby](#).

1. Problém před Unicode (Babylon kódování)

Před nástupem Unicode existovaly stovky různých kódovacích systémů (např. ASCII, ISO-8859-2 pro střední Evropu, Windows-1250).

- Tyto systémy byly nekompatibilní.
- Dokument vytvořený v jednom kódování se v jiném zobrazil jako nesrozumitelná směs znaků (tzv. **Mojibake** – např. „č“ se zobrazilo jako „Ã“).
- Bylo téměř nemožné mít v jednom dokumentu text v češtině, řečtině a japonštině zároveň.

2. Unicode: Jedna tabulka pro všechno

Unicode neřeší, jak se znak uloží v bajtech, ale přiřazuje mu abstraktní číslo.

- Zapisuje se v hexadecimálním formátu s předponou **U+**.
- Například: Velké písmeno **A** je `U+0041`, české **ř** je `U+0159`, a emoji 🐼 je `U+1F60A`.
- Aktuální verze Unicode obsahuje přes 140 000 znaků včetně historických písem (hieroglyfy) a symbolů.

3. UTF-8: Geniální kódování

UTF-8 (Unicode Transformation Format) je kódování s **proměnnou délkou bajtů** (1 až 4 bajty). Dnes tvoří naprostý standard na webu (přes 98 % stránek).

Hlavní výhody UTF-8:

- **Zpětná kompatibilita s ASCII:** Prvních 127 znaků Unicode (anglická abeceda, čísla) se v UTF-8 ukládá jako 1 bajt. Staré ASCII soubory jsou tedy zároveň platnými UTF-8 soubory.
- **Efektivita:** Pro latinu (včetně češtiny) spotřebuje méně místa než formáty jako UTF-32, které používají 4 bajty pro úplně každý znak.
- **Samoopravná schopnost:** Struktura bajtů v UTF-8 umožňuje softwaru poznat, kde znak začíná a kde končí, i když dojde k poškození části dat.

Počet bajtů	Rozsah Code Pointů	Typické znaky
1	U+0000 - U+007F	Základní latina (ASCII)
2	U+0080 - U+07FF	Evropské jazyky, diakritika (č, ř, á)
3	U+0800 - U+FFFF	Čínština, japonština, korejština
4	U+10000 - U+10FFFF	Emoji, historické symboly

4. Další formáty: UTF-16 a UTF-32

Kromě UTF-8 existují i jiná kódování Unicode:

- **UTF-16:** Používá 2 nebo 4 bajty. Je nativním formátem v operačním systému Windows a v programovacích jazycích Java a C#.
- **UTF-32:** Používá fixní 4 bajty pro každý znak. Je velmi neefektivní z hlediska paměti, ale jednoduché pro výpočty (každý znak má stejnou délku).

5. Úskalí: BOM (Byte Order Mark)

Některé editory (zejména na Windows) přidávají na začátek souboru neviditelný znak **BOM**. Ten má identifikovat pořadí bajtů, ale v moderním UTF-8 je často zdrojem problémů v PHP skriptech nebo při zpracování dat v Linuxu. Doporučuje se ukládat soubory jako „**UTF-8 without BOM**“.

Související články:

- [Datový typ Char](#)
- [Přehled datových typů](#)
- [HTML a meta tagy kódování](#)

Tagy: programming unicode utf-8 encoding internationalization text

From:

<http://serviceit.cz/> - IT ENCYKLOPEDIE

Permanent link:

<http://serviceit.cz/doku.php?id=it:sw:unicode>

Last update: **2026/01/02 18:50**

