УДК 004.057.3

Н. С. Василенко

ПРЕДСТАВЛЕННЯ ЕМОДЗІ У ФОРМАТІ UTF-16

Вінницький національний технічний університет

Анотація

Розглянуто історію та структуру формату UTF-16. Показано, що собою представляють так звані «сурогатні пари» в цьому форматі. При дослідженні представлення емодзі в UTF-16 було виявлено і описано чіткі закономірності.

Ключові слова: UTF-16, сурогатні пари, Юнікод, емодзі

Abstract

A look into the history and structure of the UTF-16 format. Shown what the so-called "surrogate pairs" are in this format. The research of the representation of emojis in UTF-16 revealed concrete patterns, described in this work.

Keywords: UTF-16, surrogate pairs, Unicode, emojis

Вступ

Емодзі стали невід'ємною частиною сучасного спілкування в інтернеті, оскільки вони дозволяють коротко передавати емоції, реакції, характерно виділяти секції тексту. Для коректного збереження й обміну такими символами в програмному забезпеченні використовується стандарт Unicode. Найпоширенішим способом є формат UTF-8, але формат UTF-16 є теж в ужитку, в першу чергу в ОС Windows. Так як коди більшості емодзі не вміщуються в межі одного 16-бітного числа, то їх представлення у UTF-16 потребує використання спеціального механізму, такого як сурогатні пари. Також різні емодзі можуть мати свої особливості при представленні. Актуальність теми зумовлена широким використанням емодзі у месенджерах, соціальних мережах і прикладному програмному забезпеченні.

Результати дослідження

В 1991 році була представлена перша версія Юнікоду – Unicode 1.0. Його виникнення пов'язано з необхідністю вміщення символів різноманітних мов світу в одному кодуванні, адже попередньо домінуючий формат ANSI [1] виділяв на кожний символ 8 біт, тобто сумарна кількість можливих символів становила 256, останні 128 з яких виділялись під окремі типи мов, створюючи таким чином нові формати (наприклад, Windows-1251 для кирилиці), що не були сумісні між собою. Це породжувало неможливість суміщення тексту на, як приклад, українській та португальській мовах, в умовах, забігаючи трохи наперед, епохи інтернету, це було недопустимо. А для східноазійських мов використовувалося кодування MBCS, яке теж, як і ANSI, поділялось на несумісні формати.

Юнікод цю проблему вирішував [2]. Перший формат UCS-2, замість одного байту як у ANSI, використовував два байти на символ. Таким чином це перетворювало коди символів в 16-бітні числа, а сумарна кількість символів вже могла становити 65536. Цього достатньо для одночасного представлення різних мов, що використовують як латиницю, так і кирилицю, до того ж і східноазійських мов, ієрогліфи яких займають тисячі місць. У 1993 році виходить перша версія Windows з ядром NT (New Technology) – Windows NT 3.1, що є найстаршим предком сучасних версій Windows, і використовує вона вже формат Юнікоду UCS-2 для представлення рядків та тексту.

Скоро після виникнення формату UCS-2 стало зрозуміло, що і 65536 символів це замало. Тоді це було спричинено через масу певних китайських ієрогліфів, математичних, історичних символів тощо, що не вміщалися. Тому в 1996 році був представлений стандарт Unicode 2.0 і формат UTF-16 [3]. На перший погляд він ідентичний до UCS-2: на символи знову виділяється по два байти. Таким чином забезпечується сумісність з UCS-2. Різниця тут у появі так званих «сурогатних пар». Сурогатна пара є комбінацією двох двобайтових символів. Перший може набувати значень від U+D800 до U+DBFF, а другий – від U+DC00 до U+DFFF (коди в Юнікоді складається з префіксу U+ та самого коду у шістнадцятковій системі числення). Така комбінація утворює новий символ. Це дозволяє формату UTF-16 підтримувати до 1114112 символів.

Принцип роботи сурогатних пар продемонструємо на прикладі смайлика з кодом U+1F600 (в десятковій системі 128512, що більше 65535, тому сурогатна пара тут є необхідною для UTF-16). Віднімаємо від 0x1F600 0x10000, результат – 0xF600. Таким чином код вміщається в 20 бітів. Тепер розділимо цей код навпіл: перші 10 бітів представлять число 0x3D, а останні – 0x200. Пора створювати сурогатну пару. Першу частину додаємо до 0xD800 (0xD800 + 0x3D = 0xD83D), а другу – до 0xDC00 (0xDC00 + 0x0200 = 0xDE00). Результат – 0xD83D 0xDE00. Цей розрахунок для легшого розуміння додатково зображено на рис. 1.

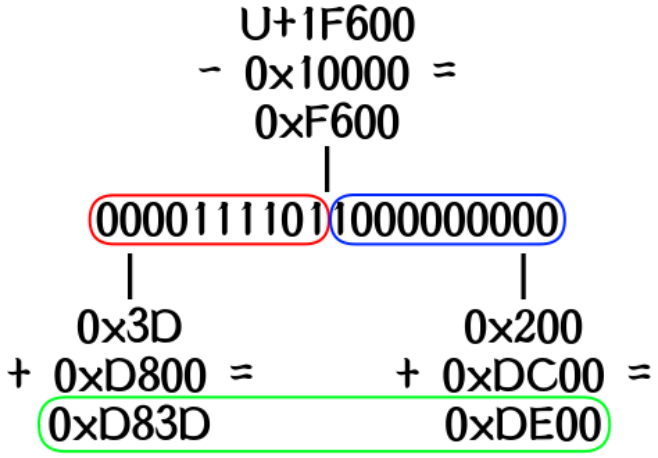

Рис. 1. Конвертація емодзі з кодом U+1F600 в сурогатну пару для UTF-16

Емодзі стрімко набирали популярність впродовж 2000-х років, і в 2010 році стандарт Unicode 6.0 включив емодзі в Юнікод. Їхнє представлення в форматі UTF-16 переважно випало саме на сурогатні пари. Емодзі знаходяться в різних частинах Юнікоду відповідно до їхньої категоризації. Наприклад, смайлики знаходяться на проміжку від U+1F600 до U+1F64F. Сучасний стандарт Unicode 17.0 нараховує близько 5000 емодзі, розподілених на 10 категорій [4].

Тут уже потрібно зазначити, що не всі емодзі мають свій окремий код в Юнікоді, а можуть складатися з декількох кодів. Розглянемо такі можливості:

1. Символ + U+FE0F. Цей код має назву Variation Selector, він показує програмі, що символ, який стоїть перед ним, необхідно представити як емодзі. Річ у тому, що існують певні прості символи, наприклад, символ серця (U+2764). Такі символи, як і літери, є чорно-білими та вмістилися в перші 65535 символів. Variation Selector ж дає знати, що треба саме емодзі серця, а не просто символ. Це випадок, коли сурогатні пари зовсім не застосовуються.

2. Прапори. Кожне емодзі прапорів складається насправді з двох окремих символів, що називаються регіональними індикаторами. Вони складаються з літер англійського алфавіту. Для емодзі прапору потрібно співставити дві таких літери у вигляді регіональних індикаторів, що складають код країни. Як приклад розглянемо емодзі прапору Італії. В Італії код IT. Тому потрібно разом поставити регіональні індикатори I (U+1F1EE) та T (U+1F1F9). Вийде італійський прапор.

3. З'єднання емодзі символом U+200D. Цей символ називається Zero Width Joiner (ZWJ). Він поєднує різні емодзі в одне нове. Наприклад, емодзі пожежника має наступне представлення: U+1F9D1 U+200D U+1F692. Першим кодом є емодзі чоловіка, а третім – пожежної машини. ZWJ утворює з цих двох емодзі емодзі пожежника. ZWJ може поєднувати не лише два емодзі. Емодзі сім'ї є результатом поєднання за допомогою ZWJ чотирьох емодзі: чоловіка, жінки, дівчинки й хлопчика, і має представлення U+1F468 U+200D U+1F469 U+200D U+1F467 U+200D U+1F466. Виходить на емодзі аж 7 символів, а враховуючи використання сурогатних пар в UTF-16, то цілих 11.

4. Емодзі + модифікатор відтінку шкіри. Такі модифікатори знаходяться в проміжку від U+1F3FB до U+1F3FF і відповідають шкалі Фітцпатріка [5]. Наприклад, емодзі пальця вгору з білим відтінком шкіри є комбінацією кодів U+1F44D та U+1F3FB.

Висновки

Отже, UTF-16 є форматом Юнікоду, в якого було ціллю можливість представлення в одному кодування якомога більшої кількості символів. Його особливістю є використання сурогатних пар, що

збільшили максимальну кількість символів з 65536 до 1114112 порівняно з попередником UCS-2. Представлення емодзі у форматі UTF-16 базується в першу чергу як раз на використанні цих сурогатних пар. Крім того, існують ще інші певні особливості при представленні емодзі, які виникають при відображенні емодзі зі звичайних символів, в прапорах, при з’єднанні різних емодзі в одне нове та застосуванні тонів шкіри.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. ANSI Encoding: A Detailed Guide to ANSI Encoding in a Modern Digital World [Електронний ресурс]. – Режим доступу: https://www.autopaintrepairs.co.uk/ansi-encoding/ (дата звернення: 25.05.2025).
2. Jukka K. Korpela. ANSI Encoding: A Detailed Guide to ANSI Encoding in a Modern Digital World. – USA : O'Reilly Media, 2006. – 688 p.
3. Understanding UTF-8 and UTF-16 [Електронний ресурс]. – Режим доступу: https://medium.com/@andygineer/understanding-utf-8-and-utf-16-how-they-help-avoid-encoding-bugs-264cbc55dca3 (дата звернення: 25.05.2025).
4. Emoji Keyboard/Display Test Data for UTS #51 [Електронний ресурс]. – Режим доступу: https://unicode.org/Public/emoji/latest/emoji-test.txt (дата звернення: 25.05.2025).
5. Mindy D. Szeto, Cara Barber, Varun K Ranpariya, Robert Dellavalle. Emojis and Emoticons in Healthcare and Dermatology Communication: A Narrative Review. DOI:10.2196/33851. 2021.

Василенко Назар Сергійович — студент групи 3ПІ-22б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: nazarvasilenko1234@tutanota.com.

Vasylenko Nazar S. — student of the Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: nazarvasilenko1234@tutanota.com.