**Đặng Minh Tuấn**

LỰA CHỌN PHƯƠNG PHÁP MÃ HOÁ KÝ TỰ UNICODE

Chào mừng các bạn đón đọc đầu sách từ dự án sách cho thiết bị di động

*Nguồn:* [*http://vnthuquan.net/*](http://vnthuquan.net/)
Tạo ebook: Nguyễn Kim Vỹ.

**MỤC LỤC**

[LỰA CHỌN PHƯƠNG PHÁP MÃ HOÁ KÝ TỰ UNICODE](%22%20%5Cl%20%22bm2)

**Đặng Minh Tuấn**

LỰA CHỌN PHƯƠNG PHÁP MÃ HOÁ KÝ TỰ UNICODE

***Đặng Minh Tuấn***
***Vietkey.Group***
***dangtuan@hn.vnn.vn***

1. **Vai trò của mã hoá dựng sẵn và tổ hợp trong Unicode** :

Dựng sẵn và tổ hợp trong Unicode đều hợp lệ, đều có thể phát triển các ứng dụng đa ngữ, đều thể hiện được các đặc điểm về ngôn ngữ như nhau, đều có thể áp dụng trong tương lai gần cũng như tương lai xa.

1. **Mã hoá ký tự và đặc điểm tổ hợp của ngôn ngữ là 2 vấn đề tách biệt.**

Các đặc điểm ngôn ngữ phục vụ cho người dùng đầu cuối, mã hoá ký tự dành cho các nhà kỹ thuật và phải trong suốt đối với người dùng. Sau khi mã hoá thành dạng nhị phân các đặc điểm về ngôn ngữ không còn được bảo tồn.

1. **Quy định về mã hoá Unicode trong môi trường Web (HTML, XML) của W3C:**

W3c dùng dạng chuẩn hoá NFC là dạng **chuẩn** **hoá thuần dựng sẵn** (Xem phụ lục 9.2).

1. **Các ngôn ngữ thuộc họ Latin đều dùng Unicode dựng sẵn.**

* 1. Các ngôn ngữ châu âu thuộc họ Latin: Pháp, Đức, Hung, Rumani... đều dùng dựng sẵn.
	2. Tiếng Việt cũng thuộc họ Latin, không phải họ Complex Script như Thái, Ả rập...
	3. Tiếng Trung Unicode cũng dùng dạng dựng sẵn (mặc dù đặc điểm hình thái ngôn ngữ là tổ hợp từ 218 bộ) trong môi trường Windows và Linux.

1. **Kỹ thuật cài đặt mã tổ hợp phức tạp**

* 1. Kỹ thuật cài đặt mã tổ hợp rất phức tạp chưa thực hiện được tốt ở nhiều môi trường (đặc biệt vấn đề hiển thị và in ấn - dấu và chữ bị lệch nhau).
	2. Ngay trong môi trường Microsoft Windows cũng không tương đương nhau, Windows 95, 98, Pocket PC2002 (PDA) hỗ trợ Unicode tổ hợp rất kém.

Bảng phân tích về hỗ trợ hiển thị Unicode dựng sẵn và tổ hợp trong các môi trường, "0" là ký hiệu thực hiện hiển thị kém hoặc không thực hiện được.
  **Về sự quyết định chỉ hỗ trợ Unicode tổ hợp của Microsoft**

* 1. Sự khác biệt giữa hỗ trợ và không hỗ trợ là khả năng *chuyển đổi chữ hoa/chữ thường, sắp xếp*...Các tính năng này không có đối với Unicode dựng sẵn trong MS Office 2K, XP không phải là nhược điểm của Unicode dựng sẵn mà là hạn chế của chính những phần mềm này. Trái với quyết định của MS VN không hỗ trợ dựng sẵn, theo tác giả Phạm Kim Long (tác giả Unikey – ( <http://unikey.sourceforge.net/forum/viewtopic.php?t=212>) thì hệ điều hành Windows 2000, XP hỗ trợ tổ hợp và dựng sẵn như nhau – có phần mềm kiểm chứng.
	2. Mã tổ hợp của Microsoft là CP1258 được phát triển từ năm 1995 có sản phẩm là Windows 95 tiếng Việt được đầu tư rất lớn nhưng đã không được thị trường Việt Nam chấp nhận (không phải công ty lớn bao giờ cũng đúng).
	3. Mã Unicode tổ hợp được Microsoft phát triển từ năm 2000 nhưng cũng chưa được sử dụng rộng rãi ở Việt Nam (xem bảng phân tích dưới).
	4. Trước năm 2000, các bộ mã tiếng Việt như TCVN3, VNI... **đều không được Microsoft hỗ trợ** , nhưng CNTT Việt Nam vẫn phát triển, các vấn đề về ngôn ngữ đều được các tổ chức trong nước thực hiện.
	5. Linux là định hướng chiến lược của nhà nước (Bộ KHCN, Đề án 112) hỗ trợ Unicode dựng sẵn với tất các các tính năng mà MS Windows có. Linux có thể thay thế một phần sản phẩm của MS Windows trong tương lai.

1. **Chuyển đổi sang Unicode tổ hợp cần kinh phí rất lớn và phức tạp.**

* 1. Theo tính toán của giám đốc công ty VASC, **cần phải 130 triệu USD** (gấp đôi kinh phí cho đề án 112) để chuyển sang dùng Unicode tổ hợp <http://www.itoday.com.vn/itoday/unicode/pbieu_tluan/pbieu_tluan_nat.htm>, theo [http://vnexpress.net](http://vnexpress.net/) , con số chi phí cho bản quyền có thể lên đến hơn 250 triệu USD. Kinh phí nâng cấp phần cứng để chạy được Windows 2000, XP còn lớn hơn nữa.
	2. Cài đặt Unicode tổ hợp phức tạp, quá trình chuyển đổi (upgrade) từ Windows 9x sang Windows 2000, XP rất phức tạp, tốn thời gian, công sức.

1. **Thực tế hầu như tất cả (99.7%) các trang Web Unicode đều dùng dựng sẵn:**

Thí nghiệm: tìm các trang Web tiếng Việt (ở Việt Nam cũng như ở nước ngoài đều) qua máy tìm kiếm **google** cho cả 2 kiểu mã hoá dựng sẵn và tổ hợp. Trường hợp 1 và 2 tìm từ rất phổ thông là "Việt Nam", "Công nghệ" trường hợp thứ 3 tìm từ có tần suất thấp là "Khuyếch đại", kết quả như sau:

**Cụm từ tìm qua google**

**Số trang Dựng sẵn tìm thấy**

**Số trang Tổ hợp tìm thấy**

"Việt Nam "

109.000 (99,68%)

348 (0.32%)

"Công nghệ"

38.600 (99,89%)

44 (0.11%)

“Khuyếch Đại”

134 (100%)

0 (0%)

Kết quả cho thấy trong hơn 3 năm xuất hiện các trang Unicode trên Internet (không chỉ ở VN mà còn ở nước ngoài) thì tỷ lệ dùng mã hoá dựng sẵn chiếm tuyệt đại đa số - hơn 99%.

1. **Kết luận**

* 1. Unicode dựng sẵn và tổ hợp đều bình đẳng trong Unicode.
	2. Lựa chọn Unicode dựng sẵn hoàn toàn phù hợp với bối cảnh hiện tại cũng như tương lai (99.7% các trang Web đã dùng Unicode dựng sẵn), hầu như tất cả các nước thuộc họ Latin đều dùng dựng sẵn.
	3. Chi phí cho Unicode tổ hợp tốn kém và cài đặt phức tạp.
	4. Không có lý do nào thuyết phục để tương lai bắt buộc phải dùng tổ hợp.
	5. Quyết định không hỗ trợ đầy đủ Unicode dựng sẵn cũng không ảnh hưởng đến quyết định của Nhà nước(trước đây đã từng không hỗ trợ trong nhiều năm).

1. Phụ lục

* 1. *Về ý kiến của đại diện Microsoft- Vũ Châu.*

( <http://www.i-today.com.vn/itoday/unicode/pbieu_tluan/pbieu_tluan_vc.htm>)

* 1. Kết luận *"Unicode consortium không khuyến cáo định dạng dựng sẵn, trừ phi không còn cách nào khác đê? biểu diễn một tổ hợp ký tự (Unicode FAQ* ***)"* là không chính xác** , Unicode consortium không hề có khuyến cáo không dùng định dạng dựng sẵn, trong Unicode FAQ không hề có thông tin này. Trái lại W3C lại quy định chỉ dùng NFC là dạng thuần dựng sẵn (xem phần dưới).
	2. "*Cuối cùng, dạng chuẩn NFC (dạng thích hợp dùng cho Web) đã ổn định – không có cách kê´t hợp chữ cái mới nào có thể thêm vào được. Vì thế, việc biểu diễn theo chuẩn NFC của bất kỳ chữ cái dựng sẵn mới nào sẽ vẫn phải dùng các chuỗi phân mã. Các chuối phân mã này có thể được biểu thị bằng cách kết hợp các chuỗi ký tự trong Unicode. Việc bổ sung chữ cái với dấu phụ để tạo ra một ký tự dựng sẵn mới là không thể thực hiện được; và ngược lại còn làm phát sinh một hoặc nhiê`u kiểu chính tả mới, làm phức tạp quá trình thực thi Unicode mà không đem lại lợi ích thực sự nào)"* **luận điểm này cũng không chuẩn xác, không có cơ sở** (xem mục dưới).

* 1. ***Quy định của W3C về dạng chuẩn hoá***

W3C là cơ quan bao gồm hơn 500 tổ chức trên toàn thế giới, chuyên nghiên cứu và đưa ra các quy định và các tiêu chuẩn trong môi trương WEB (HTML, XML).
W3C đã quy định Unicode như là bộ mã ký tự cho HTML (HTML 4.0) và Unicode cũng được dùng cho các đặc tả về XML 1.0 và CSS 2.0. XML là ngôn ngữ mở rộng của HTML và là ngôn ngữ trao đổi dữ liệu rất quan trọng trong các ứng dụng Web-based và Web service.
Các dạng chuẩn hoá được quy định trong phụ chương 15 của tiêu chuẩn Unicode, Phiên bản 3.2.0 ( <http://www.unicode.org/unicode/reports/tr15>) , Tác giả: Mark Davis ( mark.davis@us.ibm.com ), Martin Dürst ( duerst@w3.org ), Ngày: 26/3/2002 bao gồm:

**Dạng chuẩn hoá**

**Mô tả**

**Tham khảo**

Normalization Form D (NFD)

chuẩn hoá thuần tổ hợp

TC Unicode mục 3.6, 3.10, 3.11, phục chương 4

Normalization Form C (NFC)

chuẩn hoá thuần dựng sẵn

Phụ chương Unicode 15, mục 5

Normalization Form KD (NFKD)

chuẩn hoá thuần tổ hợp, phân rã ký tự tương đương

TC Unicode mục 3.6, 3.10, 3.11, phục chương 4

Normalization Form KD (NFKD)

chuẩn hoá thuần dựng sẵn, phân rã ký tự tương đương

Phụ chương Unicode 15, mục 5

Mô hình ký tự trong môi trường Web (The W3C Character Model for the World Wide Web <http://www.w3.org/TR/charmod/>) quy định dùng **NFC (chuẩn hoá thuần dựng sẵn)** cho XML và các chuẩn liên quan.

4.1.3 Lựa chọn dạng chuẩn hoá C (Normalization Form C) - Dịch toàn văn.
Unicode đưa ra 4 dạng chuẩn hoá, các dạng này khác nhau ở 1) chúng đưa các ký tự trong chuỗi text về dạng tổ hợp-decomposed characters (NFD, NFKD) hay dạng dựng sẵn - precomposed characters (NFC, NFKC), 2) chúng có chuẩn hoá các dạng tương thích (NFKD, NFKC) hay không (NFD, NFC).
Trong môi trường Web, một điều quan trọng là không được để mất cái gọi là sự khác biệt tương thích, do đó các dạng chuẩn hoá có chữ ‘K’ sẽ không được quan tâm nữa. Trong 2 dạng còn lại, *NFC có một ưu điểm là hầu như tất cả các dữ liệu cũ (legacy data) cũng như dữ liệu mới được tạo ra từ các hệ thống hiện hành đều đã đang ở dạng này*. NFC có ưu điểm là nhỏ gọn hơn đồng thời cũng phù hợp hơn với quan niệm của người dùng khi nhìn nhận về góc độ hiển thị của ký tự. Do đó NFC đã được chọn như là cơ sở cho các vấn đề chuẩn hoá ký tự đối với môi trường Web.
Tóm lại, NFC được định nghia rằng mọi chuỗi ký tự tổ hợp (bao gồm ký tự cơ sở và một hay nhiều ký tự tổ hợp đi sau đó) đều được thay trong mọi trường hợp có thể bằng một ký tự dựng sẵn chính tắc tương đương. Đoạn văn bản Text trong dạng NFC sẽ không chứa bất kỳ ký tự tổ hợp nào có thể thay thế bởi ký tự dựng sẵn.
**Character Model for the World Wide Web 1.0**
URL: <http://www.w3.org/TR/charmod/>
The **World Wide Web Consortium** (W3C) develops interoperable technologies (specifications, guidelines, software, and tools) to lead the Web to its full potential. W3C has around 500 Member organizations from all over the world and has earned international recognition for its contributions to the growth of the Web.

This document is published as part of the W3C Internationalization Activity by the Internationalization Working Group, with the help of the Internationalization Interest Group.
4.1.3 The choice of Normalization Form C
The Unicode Consortium provides four standard normalization forms (see *Unicode Normalization Forms* [[UTR #15]](http://www.w3.org/TR/charmod/#UTR15) ). These forms differ in 1) whether they normalize towards decomposed characters (NFD, NFKD) or precomposed characters (NFC, NFKC) and 2) whether they normalize away compatibility distinctions (NFKD, NFKC) or not (NFD, NFC).

For use on the Web, it is important not to lose the so-called compatibility distinctions, which may be important (see [[UXML]](http://www.w3.org/TR/charmod/#UXML) for a discussion). The K normalization forms are therefore excluded. Among the remaining two forms, NFC has the advantage that almost all legacy data (if transcoded trivially, one-to-one) as well as data created by current software is already in this form; NFC also has a slight compactness advantage and a better match to user expectations with respect to the character vs [grapheme](http://www.w3.org/TR/charmod/o#def-grapheme-string) issue. This document therefore chooses NFC as the base for Web-related text normalization.
NOTE: Roughly speaking, **NFC** is defined such that each combining character sequence (a base character followed by one or more combining characters) is replaced, as far as possible, by a canonically equivalent precomposed character. Text in a [Unicode encoding form](http://www.w3.org/TR/charmod/o#Unicode_Encoding_Form) is said to be in NFC if it doesn t contain any combining sequence that could be replaced and if any remaining combining sequence is in canonical order.
For a list of programming resources related to normalization, see [**D Resources for Normalization**](http://www.w3.org/TR/charmod/#sec-n11n-resources) .

* 1. ***Một số ưu điểm của mã hoá dựng sẵn theo tác giả Richard Gillam***

(Sách *Unicode Demystified – Trang 60-Richard Gillam-6, 2001* ).
(Lược dịch)
Ký tự tổ hợp có ưu điểm là khả năng làm giảm không gian mã hoá và cho phép tổ hợp ký tự bất kỳ có dấu thanh mà chúng ta có thể tưởng tượng được, nhưng mã hoá tổ hợp có một loạt các nhược điểm lớn là nó tốn nhiều không gian (sau khi mã hoá) hơn, khó xử lý, đồng thời phải cần đến công nghệ hiển thị rất phức tạp... Chính vì lý do đó Unicode cần phải có một số lượng lớn các ký tự dựng sẵn.
Rất nhiều chuẩn mã hoá ký tự kể cả Latin1 (trong đó tiếng Việt cũng thuộc họ Latin) được dùng trong hầu hết các ngôn ngữ châu Âu không dùng mã hoá tổ hợp mà dùng mã hoá dựng sẵn. Mã hoá dựng sẵn có quan hệ 1-1 giữa điểm mã và biểu diễn ký tự nên đơn giản trong xử lý. Đối với các hệ Latin, việc chuyển đổi giữa Latin1 với Unicode đơn giản hơn rất nhiều bằng cách thêm phần bù vào mã 8-bit để thành Unicode 16-Bit. Điều này sẽ không thể có dược nếu Unicode không có mã hoá dựng sẵn.

*Unicode Demystified – page 60-Richard Gillam-Thursday, September 6, 2001*

Canonical decompositions

Combining character sequences are great for cutting down on encoding space and allowing for representation of combinations of marks you never thought of, **but they have a couple of big disadvantages** . They take up more space, and they’re harder to process, requiring more sophisticated display technology, among other things.

For these reasons, Unicode also contains a large number of so-called "precomposed characters," code point values representing the combination of a base character and one or more non-spacing marks. Precomposed character all fall under the heading of "compatibility characters," that is, characters that were included in Unicode for compatibility with some other character encoding standard.
Many character encoding standards, including the Latin1 encoding used in most of Europe , use precomposed characters instead of combining character sequences. Users of these encodings are used to needing only a single code point to represent characters like é and ä, and implementations based on these encodings can adhere to the simple one-to-one relationship between code points and glyphs. Going to Unicode represents a significant step in either complexity or encoding size.
With Latin1, there’s the additional consideration that Latin1 forms the basis of Unicode’s
representation of the Latin alphabet. You can convert between Latin1 and Unicode simply by zero-padding to 16 bits or truncating to 8 bits. This wouldn’t be possible if Unicode didn’t have precomposed characters.
The rule in Unicode is that all precomposed characters are compatibility characters, that is, everything you can represent using precomposed characters you must also be able to represent without them. Thus, every precomposed character in Unicode has an equivalent combining character sequence. This is known as its canonical decomposition.

* 1. ***Ý kiến của Stefan Probst (chuyên gia tư vấn về CNTT của UNDP tại Việt Nam )***

Trích dẫn ý kiến tại diễn đàn chuẩn CNNT của UNDP: <http://www.isoc-vn.org/www/standard>
All,
let me add only a bit:
1) There are more fonts available. The MS set includes around 12 fonts AFAIR.Vietnamese are "famous" for using dozens of fonts mixed together in a single document, without purpose, where in fact they would need much less.
The appearance of many Vietnamese texts (whether printed or hosted) can make a typographer get sick. "More" is not always "Better"!

2) Using a combining file format does not bring any significant advantage regarding the font tables. MS is telling a fairy tale:
"Simple" combining fonts (i.e. one glyph per Unicode character) are not usable at all: There is e.g. only a single Unicode character for each of the combining tone marks, but there are at least two glyphs in every reasonable font: one mark for the lower case characters, and another one (printed slightly higher) for the upper case characters. It is up to the SW to decide which glyph in the font to chose. And if you need already SW to "translate" from the file format to the used glyphs in the font, then this SW can do as well a translation from pre-composed file format to a combining font format. It is just a small routine more.
3) **To have more combining than pre-composed font sets is no advantage** :
For printing, combining characters are of too bad quality, i.e. a pre-composed font set has to be used.
On the Internet, for widest compatibility, only basic fonts should be used.
So what is the purpose of using a combining file format? For quality printing, it had anyway to be converted to a pre-composed format, and for the Internet the increased number of available fonts is no advantage.
And: it still has to be proved, that there are really significant more usable "combining" fonts available for Vietnamese. The "o+", "u+" e.g. are typical Vietnamese characters. If a designer adds those, then he does it for the Vietnamese users. He can then also add the other "special" Vietnamese characters (i.e. the pre-composed ones).
4) **Vietnam cannot afford to do it different than the rest of the world** :
If W3C adopts the present drafts (and it looks like that), then the "must have" standard for the Web is NFC, i.e. fully pre-composed. This has more ramifications than only search engines.
a) If Vietnam uses NFD (i.e. fully de-composed/combining), or "free style" ("do what you want"), then we can immediately kiss international eCommerce in Vietnamese good-bye. Example: an overseas Vietnamese orders from a Vietnamese website something, using his "international standard" SW.
However, since the Vietnamese side is setup to use the "local" standard (combining), which is incompatible with the international one (pre-composed), the order might not be accepted at all - or the wrong items might be delivered.
He might not be able to make an online payment, because the Vietnamese payment gateway does not accept his name (written in Vietnamese). Or Vietnamese cannot order from foreign sellers, because their writing (e.g.
name, address, ...) is not accepted.
The goods might be delivered to the wrong address, because the characters are interpreted differently.
b) There will be soon Domain names in Unicode characters. Of course using pre-composed characters. While the overseas community will be happily writing eMail addresses and Website addresses in "real" Vietnamese characters, the Vietnamese will be left out, because somebody decided to do it different than everybody else in the world.
You may ask why MS is pro combining characters?
Easy. Their whole system is setup like that, and they would have to do some development work to use pre-composed file format in their spell check, dictionary SW, etc. On the other side, Linux is traditionally fully pre-composed. Do you need more reasons?
Furthermore, MS Internet Explorer on Windows 9x (i.e. prior to Win2k) does a fairly bad job on NFC encoded pages. Just have a look at the test page at <http://www.isoc-vn.org/www/standard/browsertest52.html>. The free Mozilla browser (http://www.mozilla.org ; Release 1.2 due to be released within a few days) does a quite better job there.
Well, we are used in Vietnam to short-sighted decisions. The telephone numbering system e.g. changed about 3 times in about 5 years, i.e. you had to inform your contacts, change your stationary etc. to the new numbers, because somebody could not look for some years ahead. I would not wonder, if the standard for the next two years will be NFD/combining, and then there will be big ODA money to convert all documents and databases to NFC/pre-composed two years later, because NFD proved to be not feasible.

My 300 VND for today.
Stefan

* 1. ***Kết quả thí nghiệm tìm kiếm trang Web dựng sẵn và tổ hợp trên Internet (11-2002)***

*Kết quả tìm kiếm từ "Việt Nam " dựng sẵn*
*Kết quả tìm kiếm từ "Việt Nam " tổ hợp*

Lời cuối: Cám ơn bạn đã theo dõi hết cuốn truyện.
Nguồn: http://vnthuquan.net
Phát hành: Nguyễn Kim Vỹ.

Nguồn: Thái Nhi
Được bạn: Thành Viên VNthuquan đưa lên
vào ngày: 27 tháng 12 năm 2003