

HKSCS-2004

Support for Windows Platform

*Windows XP Font Pack for ISO 10646:2003 + Amendment 1
Traditional Chinese Support (HKSCS-2004) update for Windows
XP and Windows Server 2003*

June 2010

Version 1.0

The information contained in this document represents the current view of HKITF on the issues discussed as of the date of publication. HKITF cannot guarantee the accuracy of any information presented after the date of publication. This paper is for informational purposes only.

HKITF MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

Product company names mentioned herein may be the trademarks of their respective owners.

Contents

Background	1
Introduction	2
Windows XP Font Pack for ISO 10646:2003 + Amendment 1 Traditional Chinese Support.....	3
Target Users	3
Supported Operating Systems	3
Installation Instructions	4
Support Matrix.....	7
HKSCS Support	8
ChangJie IME Input	10
Unicode Code Point Input.....	10
Save as non-Unicode.....	11
Display non-Unicode	11
Save as Unicode 3.0	11
Save as Unicode 4.1	11
Display Unicode 3.0.....	12
Display Unicode 4.1.....	12
Usage Guidance	13
Entering HKSCS-2004 Characters	13
Saving HKSCS-2004 Characters	13
Displaying HKSCS-2004 Characters	13
Data Migration	13
Additional Considerations.....	14
Mixed Unicode Code Points.....	14

Font Fallback Takes Precedence over Font Linking	15
Raster and Vector Fonts.....	16
Glyph Changes	16
Appendix A: About HKSCS.....	17
HKSCS Code Point Assignments	18
HKSCS-2008.....	19
Appendix B: HKSCS Input	20
HKSCS Input in Windows Vista or Above	20
HKSCS Input in Windows XP and Windows Server 2003	23
Appendix C: HKSCS Display	25
Unicode to Glyph Index Mapping	25
Fonts for HKSCS-2004	25
Appendix D: Character Code Conversion.....	26
Microsoft Character Code Conversion Routines for HKSCS-2004	26
Appendix E: Code Pages and Unicode	28
Code Page and Character Set.....	28
Unicode	35
Appendix F: ISO/IEC 10646 and Unicode	41
References	42

Background

The Hong Kong Information Technology Federation (HKITF) was founded in 1980 as a non-profit, non-political trade association to provide a forum in which the IT-related business in Hong Kong can work together for the benefit of the industry and to maintain a high level of business practice amongst the members.

HKITF works closely with the government to promote the development of the local IT industry. As a member of CLIAC, HKITF proposed to set up an information sharing forum on ISO10646 latest development, code point standard, and how existing application will be affected. In March 2010, a HKITF Focus Group was formed and invitations to join the Focus Group were sent to OGCIO, Banks, Utilities Companies, Telco, Media, and Partners, with Professor Lu Qin of HK Polytechnic University offered to act as the honorable advisor.

The objectives of the Focus Group were to:

- draw stakeholders' awareness on HKSCS latest development, code point standard and promote adoption of ISO/IEC 10646 for all new release of HKSCS
- collect feedback from key industry representatives
- have a common understanding on HKSCS compatibility issues with different versions of ISO 10646 standard through scenario discussion
- discuss current issues and concerns on Chinese characters processing from a business perspective

As a result, this whitepaper on HKSCS 2004 for Windows platform was produced in June 2010 with industry best practice and guidelines to facilitate the public and private sector on future platform migration.

Introduction

To facilitate electronic communication within the Government of the Hong Kong Special Administrative Region (HKSAR) that involves Chinese characters and special symbols commonly used in the HKSAR, the Government developed the Government Common Character Set (GCCS) in 1995. This character set was later enhanced by the Government, and was renamed Hong Kong Supplementary Character Set (HKSCS).

To support HKSCS in Microsoft Windows platforms, Microsoft first released HKSCS add-on support packages for Traditional Chinese versions of Windows 98, Windows Millennium Edition, Windows NT 4.0, and Windows 2000. The support packages included support for characters defined in HKSCS-1999. An updated support package was released for Windows 2000 and Windows XP to support HKSCS-2001.

By 2004, all HKSCS characters were approved to be included in the ISO 10646 and Unicode standards, specifically ISO/IEC 10646:2003 + Amendment 1 and Unicode 4.1. Windows Vista or above¹ supports Unicode 4.1, therefore inherently supports all characters defined in HKSCS-2004 without additional add-on support packages.

However, Windows XP and Windows Server 2003 platforms do not have any built-in fonts necessary to display some of the Unicode 4.1 characters, including some characters in HKSCS-2004. If you are using one of these platforms, you may not be able to view all of the HKSCS-2004 characters.

To provide the fonts necessary for users who are using Windows XP or Windows Server 2003 to properly display HKSCS-2004 characters, an add-on font pack for Windows XP and Windows Server 2003 is now available: **Windows XP Font Pack for ISO 10646:2003 + Amendment 1 Traditional Chinese Support (HKSCS-2004) update for Windows XP and for Windows Server 2003**.

¹ **Windows Vista or above** includes Windows Vista, Windows 7, Windows Server 2008, and Windows Server 2008 R2.

Windows XP Font Pack for ISO 10646:2003 + Amendment 1 Traditional Chinese Support

The **Windows XP Font Pack for ISO 10646:2003 + Amendment 1 Traditional Chinese Support (HKSCS-2004) update for Windows XP and for Windows Server 2003** ("the Font Pack") provides the necessary fonts for Windows XP and Windows Server 2003 to display HKSCS-2004 characters.

Note: In the context of this document, the terms "ISO 10646:2003 + Amendment 1" and "Unicode 4.1" are interchangeable. It also applies to the terms "ISO 10646-1:2000" and "Unicode 3.0". For more information about ISO/IEC 10646 and Unicode standards, see **Appendix F: ISO/IEC 10646 and Unicode**.

Target Users

The Font Pack targets Windows XP and Windows Server 2003 users who need fonts to display HKSCS-2004 characters.

You should install the Font Pack if you are using Windows XP or Windows Server 2003, and:

- You want to view documents that may contain HKSCS-2004 characters.
- You want to view web sites that may contain HKSCS-2004 characters.
- You have applications that may display HKSCS-2004 characters.

For example, your operating system is Windows XP and you received a document that contains a list of names. One of the names has the character "錄", which is a newly added character in HKSCS-2004. If you do not install the Font Pack, you will not be able to view the character using any of the built-in fonts.

Important: You do not need to install the Font Pack if you are using Windows Vista or above, including Windows 7. The operating system already provides the fonts necessary to display HKSCS-2004 characters.

Supported Operating Systems

The Font Pack applies to any language edition of:

- All supported x86-based versions of Windows XP and Windows Server 2003
- All supported x64-based versions of Windows Server 2003 and Windows XP Professional x64 Edition

Important: Although the Font Pack can be installed on any language edition of Windows XP and Windows Server 2003, the Font Pack is only supported when **East Asian language** files are installed, and **Language for non-Unicode programs** option is set to **Chinese (Hong Kong S.A.R.)**. See **Installation Instructions** section for more details.

Installation Instructions

Before Installing the Font Pack

Before installing the Font Pack, you must make sure:

1. You have **installed files for East Asian languages**.
2. Your **Language for non-Unicode programs** is set to **Chinese (Hong Kong S.A.R.)**.
3. You have reviewed the article at <http://support.microsoft.com/kb/977801>.

Install files for East Asian languages

To check if you have installed files for East Asian Languages, follow these steps:

1. Click **Start**, click **Control Panel**, click **Date, Time, Language, and Regional Options**, and then click **Regional and Language Options**.
2. Click the **Languages** tab.
3. Under **Supplemental language support**, if the checkbox for **Install files for East Asian languages** is checked, you have already installed files for East Asian languages.

To install files for East Asian languages, follow these steps:

Note: To install files for East Asian Languages for the first time, you may be prompted to provide the media for your operating system or service packs.

Important: You must be logged on as an administrator or a member of the Administrators group in order to complete this procedure. If your computer is connected to a network, network policy settings may also prevent you from completing this procedure.

1. Click **Start**, click **Control Panel**, click **Date, Time, Language, and Regional Options**, and then click **Regional and Language Options**.
2. Click the **Languages** tab.
3. Under **Supplemental language support**, select the **Install files for East Asian languages** checkbox.
4. Click **OK** or **Apply**.

You will be prompted to insert the Windows CD-ROM or point to a network location where the files are located.

5. After the files are installed, you must restart your computer.

Language for non-Unicode programs

To check if you have Language for non-Unicode programs set to Chinese (Hong Kong S.A.R.), follow these steps:

1. Click **Start**, click **Control Panel**, click **Date, Time, Language, and Regional Options**, and then click **Regional and Language Options**.
2. Click the **Advanced** tab.
3. If the box under **Language for non-Unicode programs** shows **Chinese (Hong Kong S.A.R.)**, you already have Language for non-Unicode programs set to Chinese (Hong Kong S.A.R.).

To set Language for non-Unicode programs to Chinese (Hong Kong S.A.R.), follow these steps:

Note: The Chinese (Hong Kong S.A.R.) language option is available after you installed files for East Asian languages.

Important: You must be logged on as an administrator or a member of the Administrators group in order to complete this procedure. If your computer is connected to a network, network policy settings may also prevent you from completing this procedure.

1. Follow the steps in the **Install files for East Asian languages** section above to install files for East Asian languages.
2. Click **Start**, click **Control Panel**, click **Date, Time, Language, and Regional Options**, and then click **Regional and Language Options**.
3. Click the **Advanced** tab.
4. Under **Language for non-Unicode programs**, select **Chinese (Hong Kong S.A.R.)**.
5. Click **OK** or **Apply**.
6. You will be prompted with the following message:

"The required files are already installed on your hard disk. Setup can use these existing files, or Setup can recopy them from your original Windows CD-ROM or from a network share.

Would you like to skip file copying and use the existing files? (If you click No, you will be prompted to insert your Windows CD-ROM or to supply an alternate location where the needed files may be found.)"

7. Click **Yes** to skip file copying.
8. Click **Yes** to restart your computer.

How to Install the Font Pack

To install the Font Pack, follow these steps:

1. Use your browser to navigate to <http://support.microsoft.com/kb/977801>.
2. Select the update to one of the following files depending on your operating system:
 - All supported x86-based versions of Windows XP and Windows Server 2003
 - All supported x64-based versions of Windows XP Professional x64 Edition and Windows Server 2003
3. At the **File Download - Security Warning** dialog box, click **Run**.
4. At the "**Do you want to run this software**" prompt, make sure Publisher is **Microsoft Corporation**.
5. Click **Microsoft Corporation**, and make sure the digital signature is ok. Click **OK**.
6. If the digital signature is ok, click **Run** to continue. Otherwise click **Don't Run** to abort.
7. Read the license agreement, and click **Yes** if you choose to accept the terms of the license agreement, or click **No** if you choose not to accept the terms of the license agreement.
8. If you chose to accept the terms of the license agreement, installation will proceed and you will be prompted to restart your computer. Click **Yes** to restart your computer for the new settings to take effect.

Note: After you installed the Font Pack, read the release note for the Font Pack at %windir%\hkscshlp.txt.

How to Uninstall the Font Pack

When you uninstall the Font Pack, you may be prompted to provide the media for your operating system or service packs.

To uninstall the Font Pack, follow these steps:

1. Click **Start**, point to **Settings**, and then click **Control Panel**.
2. Double-click **Add or Remove Programs**.
3. Click **MS HKSCS-2004 Support**.
4. Click the **Change/Remove** button.
5. Click **Yes** to restart your computer for the changes to take effect.

Important: If you had the HKSCS-2001 Support Package installed prior to the installation of the HKSCS-2004 Font Pack, you should reinstall the HKSCS-2001 Support Package to regain HKSCS-2001 support after uninstalling the HKSCS-2004 Font Pack.

Support Matrix

The following table provides support information for HKSCS-2004 characters on different platforms and HKSCS support combinations.

HKSCS Support	Windows XP or Windows Server 2003				Windows Vista or above
	None	HKSCS-2001	HKSCS-2004	HKSCS-2001 + HKSCS-2004	N/A
ChangJie IME Input	✗	▲	✗	▲	✓
Unicode Code Point Input	●	●	●	●	●
Save as non-Unicode	✗	▲	✗	▲	✗
Display non-Unicode	✗	▲	✗	▲	✗
Save as Unicode 3.0	●	▲ ●	●	▲ ●	●
Save as Unicode 4.1	●	●	●	●	✓
Display Unicode 3.0	✗	▲	✓	✓	✓
Display Unicode 4.1	✗	✗	✓	✓	✓

Note:

- ✗ Does not support HKSCS-2004 characters.
- Supports HKSCS-2004 characters when the characters are entered by Unicode code points using Chinese (Traditional) - Unicode IME or Alt-X function in Microsoft Office XP or above (Outlook 2002 or Word 2002).
- ▲ Supports HKSCS-2001 characters.
- ✓ Supports HKSCS-2004 characters.

HKSCS Support: HKSCS-2001 refers to the "HKSCS - 2001 support for Windows 2000 and Windows XP" package available from <http://www.microsoft.com/hk/hkscs> ("HKSCS-2001 Support Package")

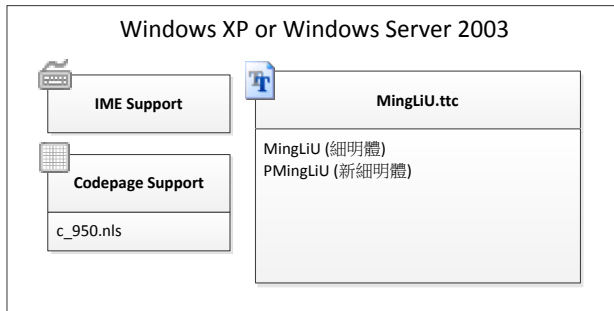
HKSCS Support: HKSCS-2004 refers to the "Windows XP Font Pack for ISO 10646:2003 + Amendment 1 Traditional Chinese Support (HKSCS-2004) update for Windows XP and Windows Server 2003" font pack available from <http://support.microsoft.com/kb/977801> ("HKSCS-2004 Font Pack").

HKSCS 2001 Characters refers to the 4818 characters defined in the HKSCS-2001 specification.

HKSCS 2004 Characters refers to the 4941 characters defined in the HKSCS-2004 specification, which includes the 4818 characters defined in the HKSCS-2001 specification plus 123 newly included characters.

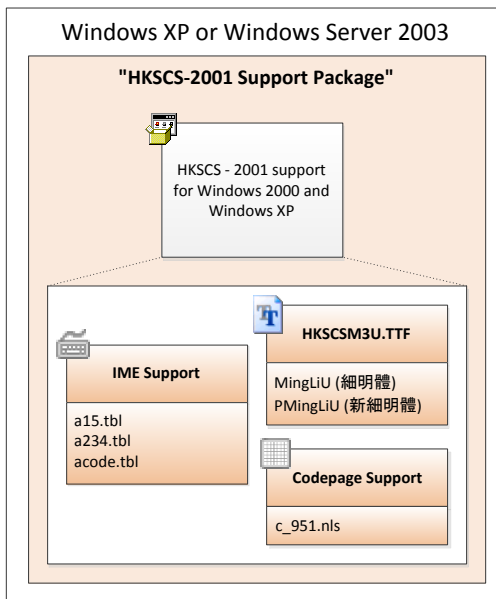
HKSCS Support

Windows XP or Windows Server 2003 without any HKSCS Support



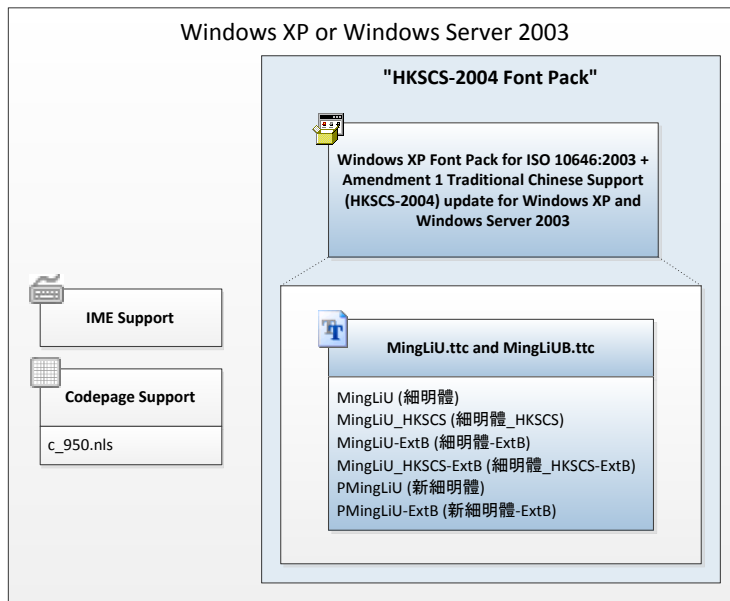
Windows XP and Windows Server 2003 provide IME and font support for Traditional Chinese characters, but do not support any HKSCS characters.

Windows XP or Windows Server 2003 with HKSCS-2001 Support



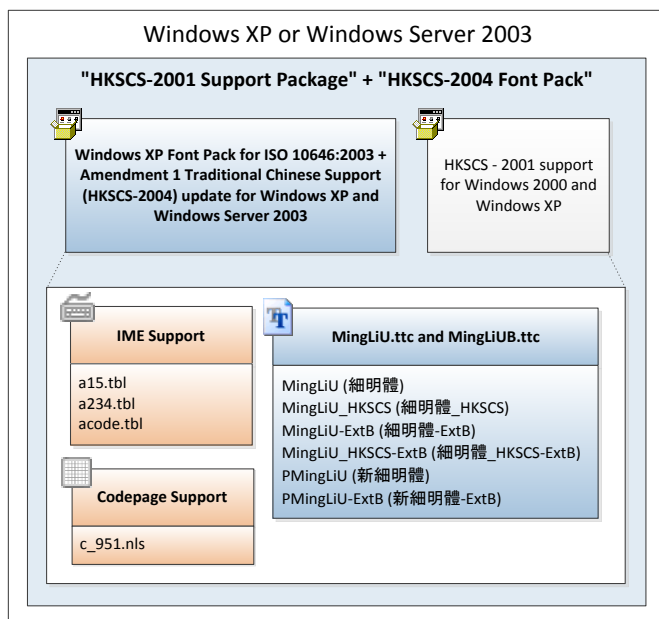
The HKSCS-2001 Support Package includes IME and font support for HKSCS-2001 characters by their Unicode 3.0 code points. It also includes a special code page that replaces the system's code page 950 to support HKSCS-2001 characters in non-Unicode format. The special code page includes Big-5 code points for HKSCS-2001 characters with mappings to their corresponding Unicode 3.0 code points.

Windows XP or Windows Server 2003 with HKSCS-2004 Support Package



The HKSCS-2004 Font Pack is a font pack which includes only font support for displaying HKSCS-2004 characters.

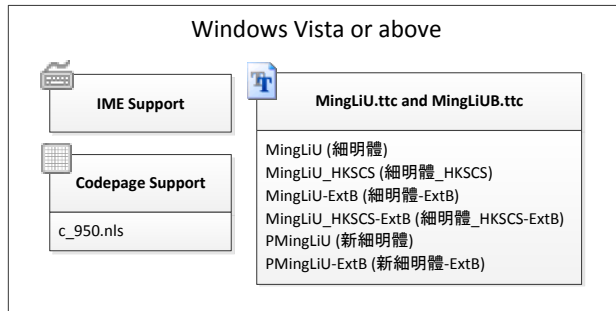
Windows XP or Windows Server 2003 with HKSCS-2001 Support Package and HKSCS-2004 Font Pack



If you install both HKSCS-2001 Support Package and HKSCS-2004 Font Pack, the fonts from the HKSCS-2004 Font Pack replace the corresponding fonts from the HKSCS-2001 Support Package.

The IME and code page support from the HKSCS-2001 Support Package continue to exist in the system, although they only support HKSCS-2001 characters by their Unicode 3.0 code points.

Windows Vista or above



Windows Vista or above provides IME and font support for Traditional Chinese characters with Unicode 4.1 code points, including all HKSCS-2004 characters. Code page 950 continues to provide support for Big-5 characters, but does not include support for HKSCS-2004 characters.

ChangJie IME Input

The built-in ChangJie Input Method Editor (IME) in Windows XP and Windows Server 2003 does not provide support for HKSCS-2004 input sequence. The HKSCS-2001 Support Package provides ChangJie IME support for HKSCS-2001 characters only. The ChangJie IME that comes with Windows Vista or above supports HKSCS-2004 input sequence when you enable HKSCS support.

Note: For more information on how to enable HKSCS support for ChangJie IME in Windows Vista or above, see **Appendix B: HKSCS Input**.

Unicode Code Point Input

You can enter HKSCS-2004 characters by entering their Unicode code points using Chinese (Traditional) - Unicode IME in Windows XP or Windows Server 2003, or using Alt-X function in Microsoft Office XP or above (Outlook 2002 or Word 2002). Windows Vista or above does not include Chinese (Traditional) - Unicode IME.

Note: For more information on how to enter HKSCS-2004 characters by entering Unicode code points, see **Appendix B: HKSCS Input**.

Important: For HKSCS-2004 characters with Unicode 3.0 code points in the Private Use Area (E000 to F8FF), you should consider using their Unicode 4.1 code points instead.

Save as non-Unicode

Save as non-Unicode refers to saving characters in "Big-5", "ANSI", or "Plain Text" format in a typical application. When you save HKSCS characters in non-Unicode format, the characters are encoded using their Big-5 code point assignments.

You cannot save HKSCS-2004 characters in non-Unicode format. You can save HKSCS-2001 characters in non-Unicode format if you have installed the HKSCS-2001 Support Package.

Important: If you have the HKSCS-2001 Support Package installed, you can save HKSCS-2001 characters in non-Unicode format. However the characters cannot be displayed in Windows Vista or above. You should save the characters as Unicode instead.

Display non-Unicode

You cannot display HKSCS-2004 characters in non-Unicode format. You can display HKSCS-2001 characters in non-Unicode format if you have installed the HKSCS-2001 Support Package. For the 123 new characters introduced in HKSCS-2004, you can only display them in Unicode format.

Save as Unicode 3.0

Save as Unicode 3.0 refers to saving characters using their Unicode 3.0 code points, typically identified as "Unicode", "Unicode big endian", or "UTF-8" format.

You can enter HKSCS-2004 characters by entering their Unicode 3.0 code points using Chinese (Traditional) - Unicode IME in Windows XP or Windows Server 2003, or using Alt-X function in Microsoft Office XP or above (Outlook 2002 or Word 2002). The values are preserved when you save these characters.

If you have installed the HKSCS-2001 Support Package, you can also enter HKSCS-2001 characters by using ChangJie IME and save them in Unicode format, in which case the Unicode 3.0 code points are used when you save the characters.

Important: For HKSCS-2004 characters with Unicode 3.0 code points in the Private Use Area (0xE000 to 0xF8FF), you should consider using their Unicode 4.1 code points instead.

Save as Unicode 4.1

Save as Unicode 4.1 refers to saving characters using their Unicode 4.1 code points, typically identified as "Unicode", "Unicode big endian", or "UTF-8" format.

You can enter HKSCS-2004 characters by entering their Unicode 4.1 code points using Chinese (Traditional) - Unicode IME in Windows XP or Windows Server 2003, or using Alt-X function in Microsoft Office XP or above (Outlook 2002 or Word 2002). The values are preserved when you save these characters.

In Windows Vista or above, you can also enter HKSCS-2004 characters by using ChangJie IME and save them in Unicode format, in which case the Unicode 4.1 code points are used when you save the characters.

Display Unicode 3.0

Display Unicode 3.0 refers to displaying characters that are encoded with Unicode 3.0 code points.

You can display HKSCS-2004 characters in Unicode format by their Unicode 3.0 code points in Windows Vista or above, or if you have installed the HKSCS-2004 support package in Windows XP or Windows Server 2003. The HKSCS-2001 Support Package only supports displaying HKSCS-2001 characters in Unicode format by their Unicode 3.0 code points.

Display Unicode 4.1

Display Unicode 4.1 refers to displaying characters that are encoded with Unicode 4.1 code points.

You can display HKSCS-2004 characters in Unicode format by their Unicode 4.1 code points in Windows Vista or above, or if you have installed the HKSCS-2004 support package in Windows XP or Windows Server 2003. The HKSCS-2001 Support Package does not support displaying HKSCS-2001 characters in Unicode format by their Unicode 4.1 code points.

Note: For more information on HKSCS display support, see **Appendix C: HKSCS Display**.

Usage Guidance

Entering HKSCS-2004 Characters

You should enter HKSCS-2004 characters by entering their Unicode 4.1 code points using Chinese (Traditional) - Unicode IME in Windows XP or Windows Server 2003, or using Alt-X function in Microsoft Office XP or above (Outlook 2002 or Word 2002). The values are preserved when you save these characters.

Note: For more information on how to enter HKSCS-2004 characters by entering Unicode code points, see **Appendix B: HKSCS Input**.

Saving HKSCS-2004 Characters

When you save information that contains HKSCS-2004 characters in Windows XP or Windows Server 2003, you should always save as Unicode format. This allows the characters to be displayed properly in Windows Vista or later operating systems.

You should always prefer saving characters in Unicode format by their Unicode 4.1 code points. All characters in HKSCS-2004 now have non-PUA code point assignments in Unicode 4.1. These code points are standardized and thus facilitate interoperability.

Note: For more information about Unicode code point assignments, see **Appendix A: About HKSCS**.

Displaying HKSCS-2004 Characters

The proper display of HKSCS-2004 characters in Windows XP or Windows Server 2003 depends on how the characters are encoded and which font is selected. In most cases the HKSCS-2004 characters can be displayed properly after you installed the Font Pack. In some cases you may need to either change the character format to Unicode or select specific fonts like MingLiU_HKSCS or MingLiU_HKSCS-ExtB to display the characters.

Note: For more information about font selection, see **Appendix C: HKSCS Display**.

Data Migration

If you have documents and data that contain HKSCS characters encoded in either Big-5 encoding or Unicode with Private Use Area (PUA) code points, you should consider converting those characters to Unicode 4.1 code points.

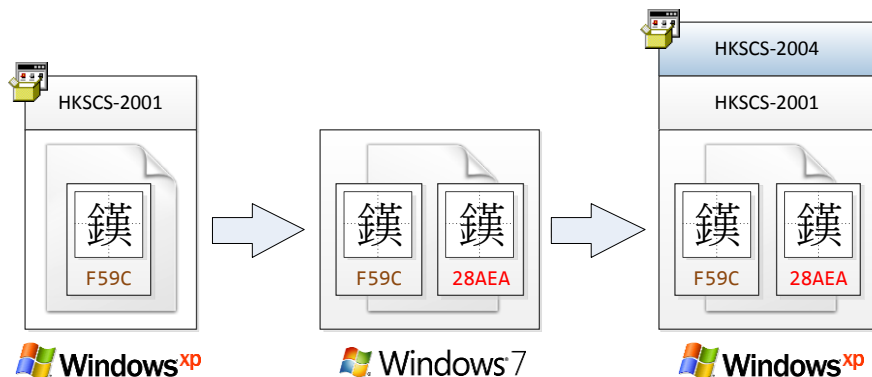
Note: For more information about character code conversion, see **Appendix D: Character Code Conversion**.

Additional Considerations

Mixed Unicode Code Points

"Mixed Unicode code points" refers to the situation when you have a document or data that is saved as Unicode, and both compatibility point and non-PUA code point exist for the same character. When you have mixed Unicode code points, search and sort operations may not produce desired results.

The diagram below illustrates a common scenario where you could end up with mixed Unicode code points:



1. You have an existing document with the character "鎂" that was created in Windows XP with HKSCS-2001 Support Package installed. The character was encoded with Unicode 3.0 PUA code point F59C.
2. You edit the document in Windows 7 using ChangJie IME to enter the same character somewhere else in the document. The character is encoded with Unicode 4.1 code point 28AEA. You can see both characters in Windows 7 because the built-in MingLiU family of fonts supports both Unicode 3.0 PUA (i.e. compatibility points) and Unicode 4.1 non-PUA code points.
3. You edit the document in Windows XP with HKSCS-2001 Support Package and HKSCS-2004 Font Pack installed. You can see both characters in Windows XP because the MingLiU family of fonts that comes with the HKSCS-2004 Font Pack supports both Unicode 3.0 PUA (i.e. compatibility points) and Unicode 4.1 non-PUA code points.
4. When you try to search the character "鎂", only the ones that are encoded with Unicode 3.0 PUA code point are found, unless you also search by its Unicode 4.1 non-PUA code point.

To avoid problems that stem from mixed Unicode code points, you should convert those characters encoded in Unicode 3.0 PUA code points to Unicode 4.1 non-PUA code points.

Note: For more information on character code conversion, see **Appendix D: Character Code Conversion**.

Font Fallback Takes Precedence over Font Linking

Font fallback refers to the mechanism in which a predefined font is selected if the currently selected font does not support a particular character. The mechanism as well as the choice of fallback font are predetermined and cannot be modified.

Font linking refers to the mechanism in which one or more fonts (called "linked fonts") are linked to another font (called the "base font"). Once you linked fonts, you can use the base font to display code points that do not exist in the base font, but that do exist in one of the linked fonts. For example, the Font Pack links MingLiU_HKSCS to Tahoma, which allows you to display HKSCS-2004 characters even if you select Tahoma to display the characters.

Some HKSCS-2004 compatibility points cannot be displayed properly with the MingLiU font in Windows XP and Windows Server 2003 even though the Font Pack links MingLiU_HKSCS to MingLiU. The problem is due to font fallback taking precedence over font linking, and glyphs from the built-in fallback font, Microsoft Sans Serif, are used instead of the glyphs from the linked fonts. In such case you need to specifically select MingLiU_HKSCS to properly display the characters.

In Windows XP and Windows Server 2003, the Microsoft Sans Serif font contains glyph information for the following code points:

- U+E801 to U+E805
- U+F700 to U+F71A
- U+F71D

In Windows Vista or above, the Microsoft Sans Serif font no longer contains glyph information for these code points, thus allowing font linking between MingLiU and MingLiU_HKSCS to work correctly.

The table below illustrates some of the problematic characters:

Code Point	Microsoft San Serif	MingLiU_HKSCS
U+E801	𐄁	鐸
U+E802	𐄂	癉
U+E803	𐄃	𪛗
U+E804	𐄄	孃
U+E805	𐄅	諗

Raster and Vector Fonts

Some raster and vector fonts cannot display certain HKSCS-2004 characters. TrueType and OpenType fonts do not have the same problem.

The Font Pack utilizes surrogate fallback mechanism to support HKSCS-2004 characters that have assigned code points in the Unicode Supplementary Ideographic Plane (SIP). However, the surrogate fallback mechanism does not apply to raster fonts (with OEM character set) and vector fonts, including **Terminal**, **Modern**, **Roman**, and **Script** fonts. If you use these fonts, you can only display characters that have assigned code points in the Unicode Basic Multilingual Plane (BMP).

To identify which HKSCS-2004 character has an assigned code point in the SIP, follow these steps:

1. Download the HKSCS-2004 document at http://www.ogcio.gov.hk/ccli/eng/hkscs/download/e_hkscs_2004.pdf.
2. In the document, locate Table 2.1 Code Table of the HKSCS-2004 in ISO/IEC 10646:2003 and Amendment 1.
3. In the table, locate the HKSCS-2004 character you are interested in, and note the corresponding value on top of the character. For example, "𠂇" has a value of "201A4".
4. If the value is between 0x20000 and 0x2FFFF (i.e. the value begins with a "2" and has 5 characters), the character has an assigned code point in the Unicode SIP. The character cannot be displayed using Terminal, Modern, Roman, or Script fonts.

Glyph Changes

The glyphs for Latin characters have changed in the MingLiU font family. Most of the changes are difficult to see. However, there are noticeable changes for the following characters:

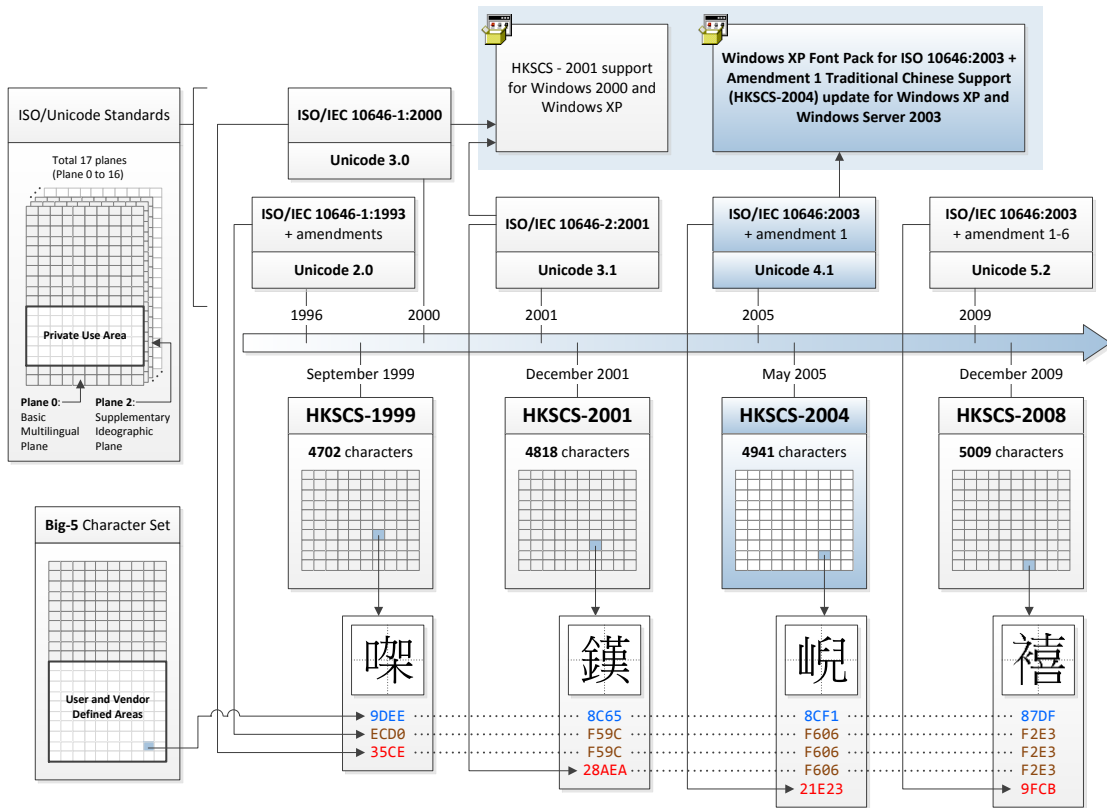
- Caron (also known as Mandarin Chinese third tone, U+02C7)
- Modifier letter acute accent (also known as Mandarin Chinese second tone, U+02CA)
- Modifier letter grave accent (also known as Mandarin Chinese fourth tone, U+02CB)

The following table lists the Unicode code points for the characters, and the glyph changes between Windows XP and Windows 7 for comparison:

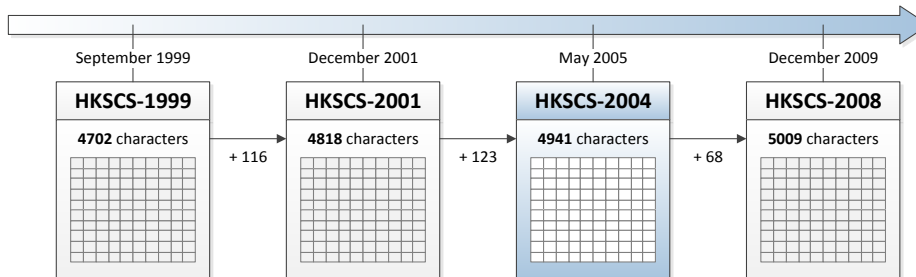
Code Point	Windows XP	Windows 7
U+02C7	✓	ˇ
U+02CA	✓	ˊ
U+02CB	✓	ˋ

Appendix A: About HKSCS

HKSCS defines Chinese characters and special symbols that are commonly used in the HKSAR. The following diagram provides a summary of the different versions of the HKSCS specifications, and their relations to ISO and Unicode standards as well as HKSCS support package and font pack.



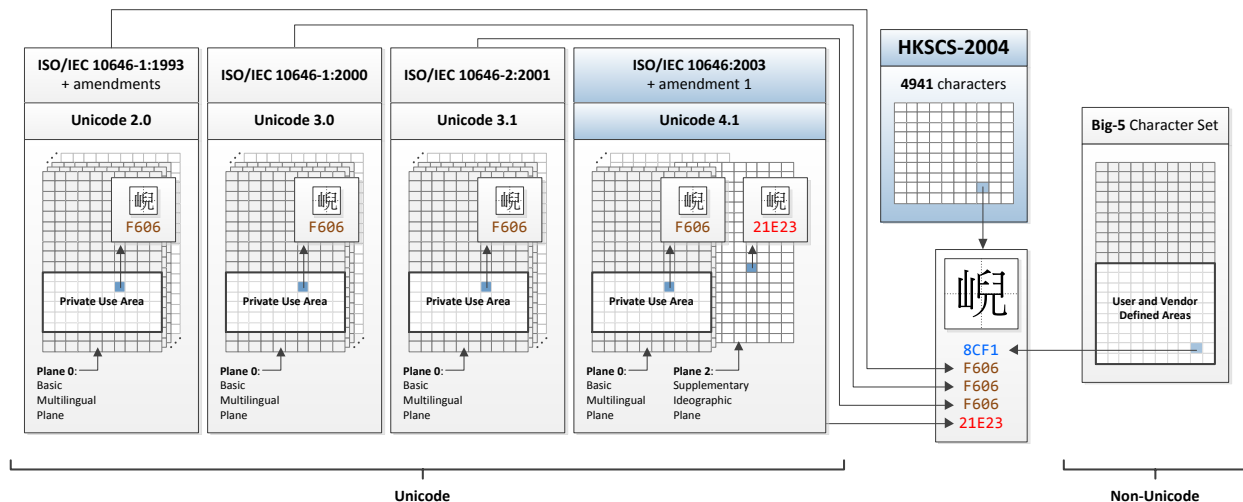
Each HKSCS specification supersedes previous versions and adds additional characters. The HKSCS-2004 specification includes 123 new characters on top of the 4818 characters already included in the HKSC-2001 specification.



HKSCS Code Point Assignments

Each character in the HKSCS specification is assigned a code point in the Big-5 character set's User Defined Area or Vendor Defined Area, and also code points in each of the ISO/Unicode standards. For characters that already have code point assignments in the ISO/Unicode standards, the HKSCS specification refers to the corresponding code points. For characters that do not have existing code point assignments in the ISO/Unicode standards, the HKSCS specification assigns code points from the Private Use Area in the Basic Multilingual Plane.

The diagram below shows code point assignments for a typical character in HKSCS-2004.



The character "峴" is assigned a code point of 0x8CF1 in the Big-5 character set's User Defined Area as it is not in the Big-5 character set. The character already has a code point assignment of U+21E23 in Unicode 4.1, thus the HKSCS-2004 specification refers to the corresponding code point for the character. However the character is not defined in previous versions of ISO/Unicode standards. In this case the HKSCS-2004 specification assigns a previously unused code point of F606 from the Private Use Area in the Basic Multilingual Plane for previous Unicode versions.

Code point assignments in the Private Use Area or User and Vendor Defined Areas are specific to HKSCS. Other systems may assign different characters to the same code points, or none at all, which poses interoperability issues.

All characters in HKSCS-2004 now have non-PUA code point assignments in Unicode 4.1. These code points are standardized and thus facilitate interoperability. Windows Vista or above supports Unicode 4.1 and thus supports non-PUA code points of all HKSCS-2004 characters properly without any additional support packages. The PUA code points that were previously assigned to the characters become compatibility points – they are reserved and will not be reassigned to other characters in future versions of HKSCS specifications.

HKSCS-2008

The HKSCS-2008 specification includes 68 new characters on top of the 4941 characters in the HKSCS-2004 specification. 62 out of the 68 new characters already have non-PUA code point assignments in ISO/IEC 10646:2003 + Amendment 1 and Unicode 4.1 code points, thus are supported in Windows Vista or above. Windows XP or Windows Server 2003 with the HKSCS-2004 Font Pack installed can also display these 62 characters. The remaining 6 characters of HKSCS-2008 (i.e. Unicode code points: U+9FC7, U+9FC8, U+9FC9, U+9FCA, U+9FCB, and U+2ADFF) encoded in ISO/IEC 10646:2003 + Amendments 5 and 6 and Unicode 5.2 code points are not supported by the HKSCS-2004 Font Pack.

The table below lists the 6 characters that are not supported by the HKSCS-2004 Font Pack.

Code Point	Character
U+9FC7	𢆶
U+9FC8	𢆷
U+9FC9	𢆸
U+9FCA	𢆹
U+9FCB	𢆺
U+2ADFF	𪛟

Note: You can find the new contents of the HKSCS-2008 at
<http://www.ogcio.gov.hk/ccli/eng/hkscs/download/68for2008.pdf>


Appendix B: HKSCS Input

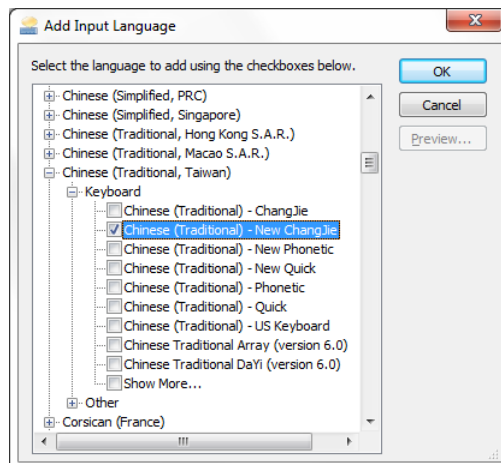
HKSCS Input in Windows Vista or Above

The Traditional Chinese IME including ChangJie/New ChangJie, Phonetic/New Phonetic, and Quick/New Quick in Windows Vista or above provides support for HKSCS-2004 input sequence, but the feature is disabled by default. Once you enable the feature, you can enter all the characters in HKSCS-2004 with a keyboard using the IMEs without additional add-ons or applications.

Enabling HKSCS Support

To enable HKSCS support in Windows Vista or above, follow these steps:

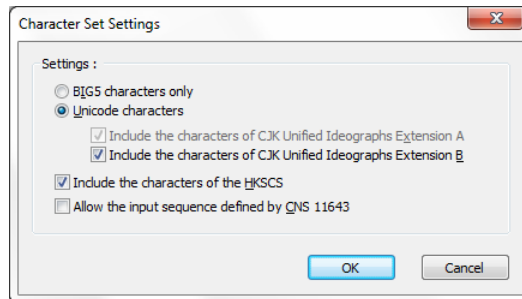
1. Click **Start** , type **intl.cpl** in the **Search programs and files** box, and then press Enter.
2. Click the **Keyboards and Languages** tab, and then click **Change keyboards**.
3. Under **Installed services**, Click **Add**.
4. Locate and expand **Chinese (Traditional, Taiwan)**.
5. Expand **Keyboard**, and select one or more of the following input methods:
 - Chinese (Traditional) – ChangJie
 - Chinese (Traditional) – New ChangJie
 - Chinese (Traditional) – New Phonetic
 - Chinese (Traditional) – New Quick
 - Chinese (Traditional) – Phonetic
 - Chinese (Traditional) – Quick



6. Click **OK**.
7. In **Installed services**, click one of the newly added keyboards, and then click **Properties**.
8. Click **Character Set**.

9. Select **Unicode characters** , and check the following checkboxes:

- Include the characters of CJK Unified Ideographs Extension A
- Include the characters of CJK Unified Ideographs Extension B
- Include the characters of the HKSCS

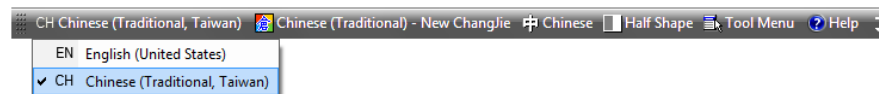


10. Click **OK** three times.

11. In the **Regional and Language** dialog box, click **OK**.

Notice that the **Language** bar appears on the taskbar or floating on desktop. When you position the mouse pointer over it, a ToolTip appears that describes the active keyboard layout.

12. You can now select one of the newly added keyboards from the Language bar to input HKSCS-2004 characters.

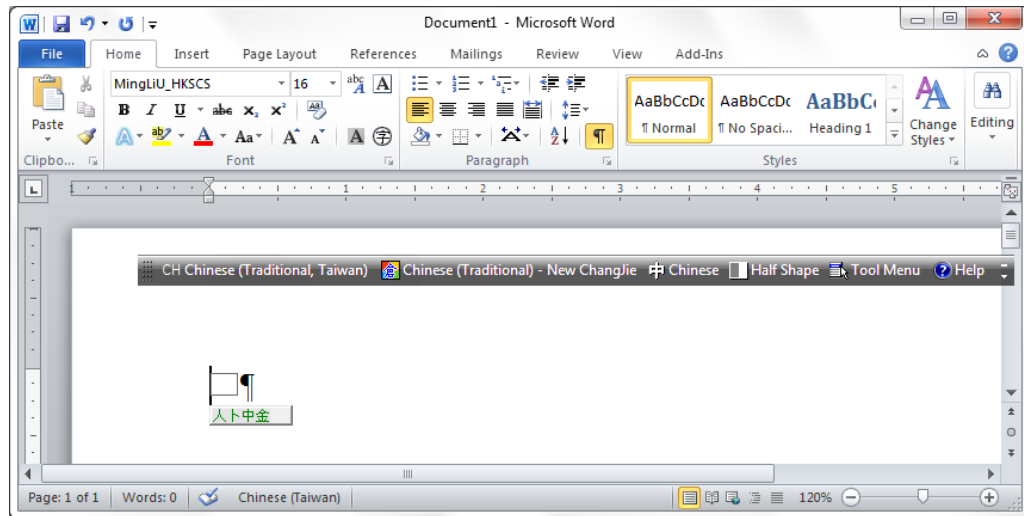


Using ChangJie Input Method

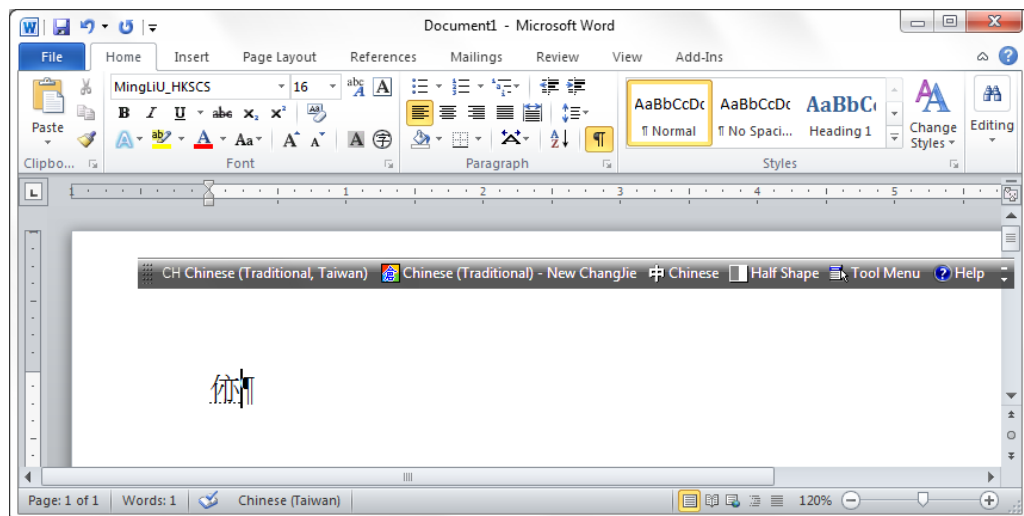
To enter input sequence for HKSCS-2004 characters using ChangJie (also known as Cangjie) input method in Windows Vista or above, follow these steps:

1. Follow the steps in the **Enabling HKSCS Support** section above to enable HKSCS-2004 support.
2. Locate the HKSCS-2004 character you are interested in, and its corresponding Big5 value.
 - Download the HKSCS-2004 document at http://www.ogcio.gov.hk/ccli/eng/hkscs/download/e_hkscs_2004.pdf.
 - In the document, locate Table 2.1 Code Table of the HKSCS-2004 in ISO/IEC 10646:2003 and Amendment 1.
 - In the table, locate the HKSCS-2004 character you are interested in, and note the corresponding Big-5 value (i.e. the bottom value of the character with brackets). For example, "𠂇" has a value of "8CF4".
3. Locate the character's corresponding ChangJie input sequence.
 - Go to **Cangjie Input Code Reference Table in the HKSCS-2004** web page at <http://www.ogcio.gov.hk/ccli/eng/hkscs/terms/terms53.html>.

- Read the Terms of Use, and click **Accept and Download** if you accept the terms, or click **Cancel** to abort.
 - In the document, locate the Big-5 value you noted in step 2 above, and note the Cangjie input code. For example, 8CF4 corresponds to "OYLC" input sequence.
4. Go to the application where you want to insert the character.
 5. From the Language bar, select **Chinese (Traditional) - New ChangJie**.
 6. Enter the input sequence noted in step 3 above.
- For example, enter **OYLC** for the character "你":



Press the Space bar and the character "你" should appear:



Press Enter to confirm the insertion.

HKSCS Input in Windows XP and Windows Server 2003

The Traditional Chinese IMEs in Windows XP and Windows Server 2003 do not provide support for HKSCS input sequence. You can input HKSCS-2001 characters using the IME support from the HKSCS-2001 Support Package, but you cannot use them to input characters that are new to HKSCS-2004. To input HKSCS-2004 characters in Windows XP or Windows Server 2003, you can enter the Unicode code points of HKSCS-2004 characters using one of the following methods.

Insert HKSCS-2004 Characters with ALT-X Hotkey

In Microsoft Office XP (Outlook 2002 and Word 2002) or above, you can enter Unicode code points of HKSCS-2004 characters to insert HKSCS-2004 characters within the application.

To enter Unicode code points of HKSCS-2004 characters, follow these steps:

1. Follow the steps in the **Installation Instructions** section to enable HKSCS-2004 support.
2. Locate the HKSCS-2004 character you are interested in, and its corresponding Unicode code point.
 - Download the HKSCS-2004 document at http://www.ogcio.gov.hk/ccli/eng/hkscs/download/e_hkscs_2004.pdf.
 - In the document, locate Table 2.1 Code Table of the HKSCS-2004 in ISO/IEC 10646:2003 and Amendment 1.
 - In the table, locate the HKSCS-2004 character you are interested in, and note the corresponding code point on top of the character. For example, "𠄎" has a code point of "344A".
3. Go to the application where you want to insert the character.
4. Enter the Unicode code point noted in Step 2 above, and then press **ALT+X** to convert it into the corresponding HKSCS-2004 character. For example, enter **344A** and then press **ALT+X** to produce "𠄎". Press Enter to confirm the insertion.

Note: To enter Unicode code points outside of the Basic Multilingual Plane, you can just enter the Unicode code point. For example, to enter the Unicode code point 201A4 for the character "𐄌", enter 201A4 and then press Alt-X.

Insert HKSCS-2004 Characters with Unicode Input Method

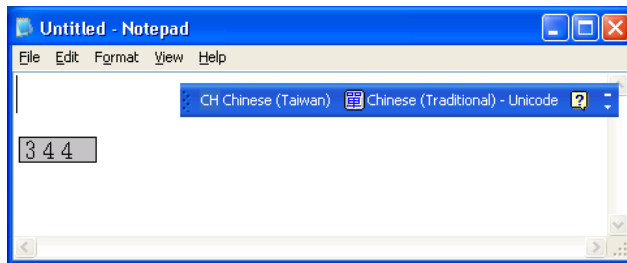
In Windows XP and Windows Server 2003, you can use the **Chinese (Traditional) - Unicode** IME to enter Unicode code points of HKSCS-2004 characters to insert HKSCS-2004 characters in your application.

To enter Unicode code points of HKSCS-2004 characters, follow these steps:

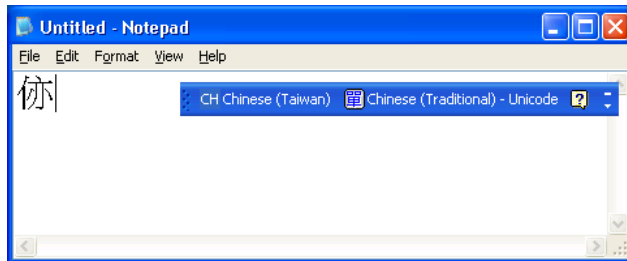
1. Follow the steps in the **Installation Instructions** section to enable HKSCS-2004 support.
2. Click **Start**, and then click **Control Panel**.
3. Select **Regional and Language Options**.

4. Click the **Languages** tab, and then click **Details**.
5. Click **Add**.
6. Select **Chinese (Taiwan)** for Input Language, and then select **Chinese (Traditional) - Unicode** for Keyboard layout/IME.
7. Click **OK** three times.
8. Locate the HKSCS-2004 character you are interested in.
 - Download the HKSCS-2004 document at http://www.ogcio.gov.hk/ccli/eng/hkscs/download/e_hkscs_2004.pdf.
 - In the document, locate Table 2.1 Code Table of the HKSCS-2004 in ISO/IEC 10646:2003 and Amendment 1.
 - In the table, locate the HKSCS-2004 character you are interested in, and note the corresponding value on top of the character. For example, "𡗗" has a value of "344A".
9. Go to the application where you want to insert the character.
10. From the Language bar, select **Chinese (Taiwan)**. If you have more than one Traditional Chinese IME installed, select **Chinese (Traditional) - Unicode** next to it.
11. Enter the code noted in Step 8 above.

For example, enter **344A** for the character "𡗗":



After you entered 344A, the character "𡗗" should appear:



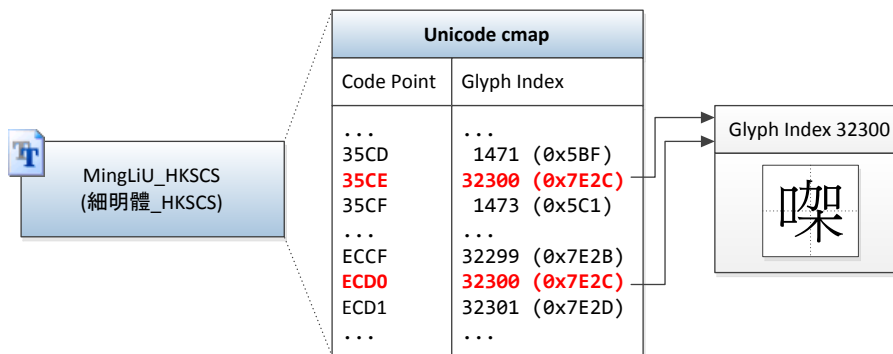
Note: To enter Unicode code points outside of the Basic Multilingual Plane, you need to enter the surrogate pair values. For example, to enter the Unicode code point 201A4 for the character "𐄌", enter D840 and then DDA4.

Appendix C: HKSCS Display

Unicode to Glyph Index Mapping

Each TrueType font, amongst other things, includes a Unicode-to-glyph-index mapping table marked as 'cmap', which maps each supported Unicode code point to a glyph index. Each glyph index corresponds to a particular character's glyph (i.e. graphical representation of the character).

For example, in the MingLiU_HKSCS font that is included in the Font Pack, it supports the character "㗎" with Unicode 3.0 code point U+35CE. The previously assigned code point U+ECD0 in the Private Use Area (PUA) is preserved. The cmap table maps both U+35CE and U+ECD0 code points to the same glyph.



Fonts for HKSCS-2004

The MingLiU family of fonts that is included in the Font Pack includes information for HKSCS-2004 characters. They are variants of the Windows 7 versions of the same files, and support the same set of characters. The table below lists the available fonts that are included in the Font Pack, their corresponding font files, and the characters they support:

Font Files	Font Name	Description
MingLiU.ttc	MingLiU (細明體)	Supports all HKSCS-2004 characters that are defined in the Basic Multilingual Plane (Plane 0), including those in the CJK Unified Ideographs Extension A range (0x3400 to 0x4DBF) and CJK Unified Ideographs range (0x4E00 to 0x9FFF).
	PMingLiU (新細明體)	
	MingLiU_HKSCS (細明體_HKSCS)	Supports the same set of characters as MingLiU and PMingLiU fonts, plus HKSCS compatibility points in the Private Use Area range (0xE000 to 0xF8FF).
MingLiUB.ttc	MingLiU-ExtB (細明體-ExtB)	Supports HKSCS-2004 characters that are defined in the CJK Unified Ideographs Extension B range (0x20000 to 0x2A6DF).
	MingLiU_HKSCS-ExtB (細明體_HKSCS-ExtB)	
	PMingLiU-ExtB (新細明體-ExtB)	

Appendix D: Character Code Conversion

If you have documents and data that contain HKSCS characters encoded in either Big-5 encoding or Unicode with Private Use Area (PUA) code points, you should consider converting those characters to Unicode 4.1 code points.

If you have documents and data that contain HKSCS characters, the characters may be encoded in Big-5 format when:

- You saved documents in "Big-5", "ANSI", or "Plain Text" format.
- You have web pages using Big-5 charset.
- You have a database that stores non-Unicode data type (e.g. char, varchar, text).

If you have documents and data that contain HKSCS characters, the characters may be encoded in Unicode format with Private Use Area (PUA) code points when:

- You saved documents in "Unicode", "Unicode big endian", or "UTF-8" format in Windows XP or Windows Server 2003 with HKSCS-2001 Support Package installed.
- You have web pages using UTF-8 charset which was created in Windows XP or Windows Server 2003 with HKSCS-2001 Support Package installed.
- You have a database that stores Unicode data type (e.g. nchar, nvarchar, ntext) with data from Windows XP or Windows Server 2003 with HKSCS-2001 Support Package installed.

To convert files encoded in either Big-5 encoding or Unicode with PUA code points, you can utilize Microsoft Character Code Conversion Routines for HKSCS-2004.

Microsoft Character Code Conversion Routines for HKSCS-2004

Application developers can use the Microsoft Character Code Conversion Routines for HKSCS-2004 to develop applications that convert HKSCS characters encoded in either Big-5 encoding or Unicode with Private Use Area (PUA) code points to Unicode 4.1 code points.

Note: To download the Microsoft Character Code Conversion Routines for HKSCS-2004, please visit <http://www.microsoft.com/downloads/details.aspx?FamilyID=0e6f5ac8-7baa-4571-b8e8-78b3b776afd7&displaylang=en>.

The installable package consists of a Microsoft Windows Installer file (HKSCS04.msi) which must be installed to your computer. The package contains a dynamic linked library file (hkscs04.dll), a static library file (hkscs04.lib), a header file (hkscs04.h), starter code for two sample applications, and two Visual Studio project files.

To build the sample applications from the Microsoft Character Code Conversion Routines for HKSCS-2004 to perform file conversion, follow these steps:

1. Download and install the Microsoft Character Code Conversion Routines for HKSCS-2004 from <http://www.microsoft.com/downloads/details.aspx?FamilyID=0e6f5ac8-7baa-4571-b8e8-78b3b776afd7&displaylang=en>.
2. You need to have Visual Studio .NET 2003 or later to build the sample applications. If you do not currently have Visual Studio, you can download and install Visual C++ 2010 Express from <http://www.microsoft.com/express/downloads/#2010-Visual-CPP>.
3. Click **Start**, click **All Programs**, click **Conversion Routines for HKSCS-2004**, click **Samples**, and click **Big-5 Sample**.
4. If you have Visual Studio newer than Visual Studio .NET 2003 (e.g. you installed Visual C++ 2010 Express from step 2 above), follow the instructions from the Visual Studio Conversion Wizard to upgrade the project.
5. In Solution Explorer, right-click **Solution 'Big-5 Sample'** and select **Build Solution**. You should see output in the Output window similar to the following:

```
Big5 Sample.vcxproj -> C:\Program Files\Conversion Routines for HKSCS-
2004\sample\Big5 Sample\Debug\Big5 Sample.exe
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped =====
```

6. Repeat Steps 3 to 5 for **PUA Sample**.
7. You can now use Big5-Sample.exe and PUA-Sample.exe to convert text files encoded in either Big-5 encoding or Unicode with PUA code points to Unicode 4.1 code points.

Usage of the sample applications are listed below.

```
C:\>"C:\Program Files\Conversion Routines for HKSCS-2004\sample\Big5
Sample\Debug\Big5-Sample.exe"
usage:
big5-sample <input file> <output file>
Example: big5-sample Big5Input.txt UnicodeOutput.txt

C:\>"C:\Program Files\Conversion Routines for HKSCS-2004\sample\PUA
Sample\Debug\PUA-Sample.exe"
usage:
pua-sample <input file> <output file>
Example: pua-sample PUAInput.txt Unicode4Output.txt
```

Note: The sample applications only support plain text documents. To convert other document types like Microsoft Word documents using the sample applications, save the documents as Plain Text with Unicode encoding before conversion. After the text documents are converted, you can copy and paste the converted text back to the original documents. You may need to reformat the text to regain the original format.

Appendix E: Code Pages and Unicode

Code Page and Character Set

A code page is an ordered set of characters of a given script in which a numeric index (code point) is associated with each character. In the context of code pages defined by Windows, a code page is sometimes called a character set (charset).

Code Pages in Windows

There are three groups of code pages supported by Windows:

1. Windows code pages
2. OEM code pages
3. ISO 8859 code pages

Windows Code Pages

Windows code pages, sometimes referred to as "ANSI" or "Windows ANSI" code pages, consist of two groups of code pages: Single Byte Character Set (SBCS) and Multibyte character sets, in particular the Double Byte Character Set (DBCS). Below lists some commonly used Windows code pages.

SBCS (Single Byte Character Set) Code Pages:

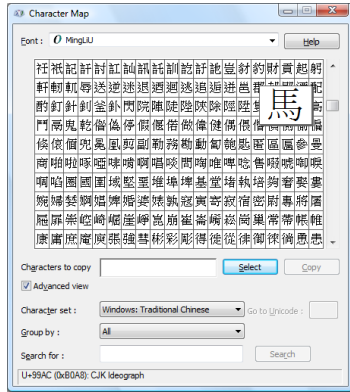
- 1250 (Central Europe)
- 1251 (Cyrillic)
- 1252 (Latin I)
- 1253 (Greek)

DBCS (Double Byte Character Set) Code Pages:

- 932 (Japanese Shift-JIS)
- 936 (Simplified Chinese GBK)
- 949 (Korean)
- 950 (Traditional Chinese Big5)

In DBCS code pages, some code points above 0x80 represent lead bytes. Each lead byte is an index to another set of 256 character block that is associated with that lead byte. The indexed character block is used to interpret the trail byte. The following diagram illustrates the concept with a sample character "馬" that has a value of 0xB0A8 when it is encoded in Windows code page 950, where 0xB0 is the lead byte and 0xA8 is the trail byte.

Windows Codepage 950



0xB0A8

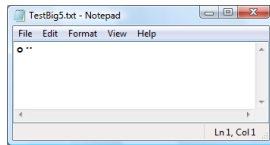
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	B0A0	B0A1	B0A2	B0A3	B0A4	B0A5	B0A6	B0A7	B0A8	B0A9	B0AA	B0AB	B0AC	B0AD	B0AE	B0AF
10	B0B0	B0B1	B0B2	B0B3	B0B4	B0B5	B0B6	B0B7	B0B8	B0B9	B0BA	B0BB	B0BC	B0BD	B0BE	B0BF
20	B0C0	B0C1	B0C2	B0C3	B0C4	B0C5	B0C6	B0C7	B0C8	B0C9	B0CA	B0CB	B0CC	B0CD	B0CE	B0CF
30	B0D0	B0D1	B0D2	B0D3	B0D4	B0D5	B0D6	B0D7	B0D8	B0D9	B0DA	B0DB	B0DC	B0DD	B0DE	B0DF
40	B0E0	B0E1	B0E2	B0E3	B0E4	B0E5	B0E6	B0E7	B0E8	B0E9	B0EA	B0EB	B0EC	B0ED	B0EE	B0EF
50	B0F0	B0F1	B0F2	B0F3	B0F4	B0F5	B0F6	B0F7	B0F8	B0F9	B0FA	B0FB	B0FC	B0FD	B0FE	B0FF
60	B0G0	B0G1	B0G2	B0G3	B0G4	B0G5	B0G6	B0G7	B0G8	B0G9	B0GA	B0GB	B0GC	B0GD	B0GE	B0GF
70	B0H0	B0H1	B0H2	B0H3	B0H4	B0H5	B0H6	B0H7	B0H8	B0H9	B0HA	B0HB	B0HC	B0HD	B0HE	B0HF
80	B0I0	B0I1	B0I2	B0I3	B0I4	B0I5	B0I6	B0I7	B0I8	B0I9	B0IA	B0IB	B0IC	B0ID	B0IE	B0IF
90	B0J0	B0J1	B0J2	B0J3	B0J4	B0J5	B0J6	B0J7	B0J8	B0J9	B0JA	B0JB	B0JC	B0JD	B0JE	B0JF
A0	B0K0	B0K1	B0K2	B0K3	B0K4	B0K5	B0K6	B0K7	B0K8	B0K9	B0KA	B0KB	B0KC	B0KD	B0KE	B0KF
B0	B0L0	B0L1	B0L2	B0L3	B0L4	B0L5	B0L6	B0L7	B0L8	B0L9	B0LA	B0LB	B0LC	B0LD	B0LE	B0LF
C0	B0M0	B0M1	B0M2	B0M3	B0M4	B0M5	B0M6	B0M7	B0M8	B0M9	B0MA	B0MB	B0MC	B0MD	B0ME	B0MF
D0	B0N0	B0N1	B0N2	B0N3	B0N4	B0N5	B0N6	B0N7	B0N8	B0N9	B0NA	B0NB	B0NC	B0ND	B0NE	B0NF
E0	B0O0	B0O1	B0O2	B0O3	B0O4	B0O5	B0O6	B0O7	B0O8	B0O9	B0OA	B0OB	B0OC	B0OD	B0OE	B0OF
F0	B0P0	B0P1	B0P2	B0P3	B0P4	B0P5	B0P6	B0P7	B0P8	B0P9	B0PA	B0PB	B0PC	B0PD	B0PE	B0PF

Windows 950_B0

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
40	馬	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
50	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
60	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
70	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
80	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
90	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
A0	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
B0	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
C0	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
D0	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
E0	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍
F0	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍	鞍

If the same value 0xB0A8 is viewed by the system as encoded in Windows code page 1252, two separate characters (° and °) are interpreted, as shown in the diagram below.

Windows Codepage 1252



	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	B0A0	B0A1	B0A2	B0A3	B0A4	B0A5	B0A6	B0A7	B0A8	B0A9	B0AA	B0AB	B0AC	B0AD	B0AE	B0AF
10	B0B0	B0B1	B0B2	B0B3	B0B4	B0B5	B0B6	B0B7	B0B8	B0B9	B0BA	B0BB	B0BC	B0BD	B0BE	B0BF
20	B0C0	B0C1	B0C2	B0C3	B0C4	B0C5	B0C6	B0C7	B0C8	B0C9	B0CA	B0CB	B0CC	B0CD	B0CE	B0CF
30	B0D0	B0D1	B0D2	B0D3	B0D4	B0D5	B0D6	B0D7	B0D8	B0D9	B0DA	B0DB	B0DC	B0DD	B0DE	B0DF
40	B0E0	B0E1	B0E2	B0E3	B0E4	B0E5	B0E6	B0E7	B0E8	B0E9	B0EA	B0EB	B0EC	B0ED	B0EE	B0EF
50	B0F0	B0F1	B0F2	B0F3	B0F4	B0F5	B0F6	B0F7	B0F8	B0F9	B0FA	B0FB	B0FC	B0FD	B0FE	B0FF
60	B0G0	B0G1	B0G2	B0G3	B0G4	B0G5	B0G6	B0G7	B0G8	B0G9	B0GA	B0GB	B0GC	B0GD	B0GE	B0GF
70	B0H0	B0H1	B0H2	B0H3	B0H4	B0H5	B0H6	B0H7	B0H8	B0H9	B0HA	B0HB	B0HC	B0HD	B0HE	B0HF
80	B0I0	B0I1	B0I2	B0I3	B0I4	B0I5	B0I6	B0I7	B0I8	B0I9	B0IA	B0IB	B0IC	B0ID	B0IE	B0IF
90	B0J0	B0J1	B0J2	B0J3	B0J4	B0J5	B0J6	B0J7	B0J8	B0J9	B0JA	B0JB	B0JC	B0JD	B0JE	B0JF
A0	B0K0	B0K1	B0K2	B0K3	B0K4	B0K5	B0K6	B0K7	B0K8	B0K9	B0KA	B0KB	B0KC	B0KD	B0KE	B0KF
B0	B0L0	B0L1	B0L2	B0L3	B0L4	B0L5	B0L6	B0L7	B0L8	B0L9	B0LA	B0LB	B0LC	B0LD	B0LE	B0LF
C0	B0M0	B0M1	B0M2	B0M3	B0M4	B0M5	B0M6	B0M7	B0M8	B0M9	B0MA	B0MB	B0MC	B0MD	B0ME	B0MF
D0	B0N0	B0N1	B0N2	B0N3	B0N4	B0N5	B0N6	B0N7	B0N8	B0N9	B0NA	B0NB	B0NC	B0ND	B0NE	B0NF
E0	B0O0	B0O1	B0O2	B0O3	B0O4	B0O5	B0O6	B0O7	B0O8	B0O9	B0OA	B0OB	B0OC	B0OD	B0OE	B0OF
F0	B0P0	B0P1	B0P2	B0P3	B0P4	B0P5	B0P6	B0P7	B0P8	B0P9	B0PA	B0PB	B0PC	B0PD	B0PE	B0PF

OEM Code Pages

OEM code pages, sometimes referred to as "Windows OEM" code pages, are used for conversions of MS-DOS-based, text-mode applications. Below lists some commonly used OEM code pages.

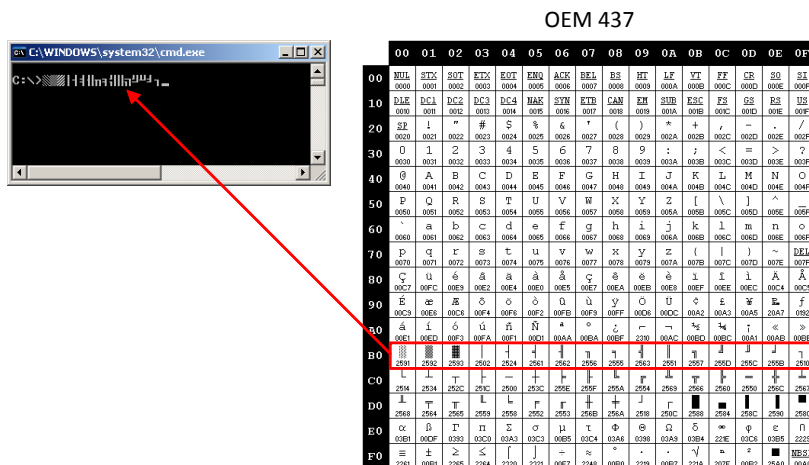
- 437 (US)
- 720 (Arabic)
- 737 (Greek)
- 775 (Baltic)
- 862 (Hebrew)
- 866 (Russian)

The following code pages are used as both Windows ANSI and OEM code pages:

- 874 (Thai)
- 932 (Japanese Shift-JIS)
- 936 (Simplified Chinese GBK)
- 949 (Korean)
- 950 (Traditional Chinese Big5)
- 1258 (Vietnam)

The following diagram shows the OEM 437 code page and a sample sequence of code points from 0xB0 to 0xBF as interpreted by a console window using OEM 437 code page.

OEM 437



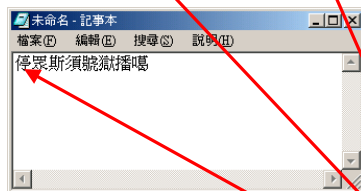
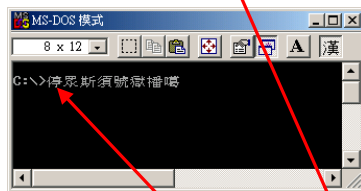
	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	0000	0001	0002	0003	0004	0005	0006	0007	0008	0009	000A	000B	000C	000D	000E	000F
10	0010	0011	0012	0013	0014	0015	0016	0017	0018	0019	001A	001B	001C	001D	001E	001F
20	0020	0021	0022	0023	0024	0025	0026	0027	0028	0029	002A	002B	002C	002D	002E	002F
30	0030	0031	0032	0033	0034	0035	0036	0037	0038	0039	003A	003B	003C	003D	003E	003F
40	0040	0041	0042	0043	0044	0045	0046	0047	0048	0049	004A	004B	004C	004D	004E	004F
50	0050	0051	0052	0053	0054	0055	0056	0057	0058	0059	005A	005B	005C	005D	005E	005F
60	0060	0061	0062	0063	0064	0065	0066	0067	0068	0069	006A	006B	006C	006D	006E	006F
70	0070	0071	0072	0073	0074	0075	0076	0077	0078	0079	007A	007B	007C	007D	007E	007F
80	0080	0081	0082	0083	0084	0085	0086	0087	0088	0089	008A	008B	008C	008D	008E	008F
90	0090	0091	0092	0093	0094	0095	0096	0097	0098	0099	009A	009B	009C	009D	009E	009F
A0	00A0	00A1	00A2	00A3	00A4	00A5	00A6	00A7	00A8	00A9	00AA	00AB	00AC	00AD	00AE	00AF
B0	00B0	00B1	00B2	00B3	00B4	00B5	00B6	00B7	00B8	00B9	00BA	00BB	00BC	00BD	00BE	00BF
C0	00C0	00C1	00C2	00C3	00C4	00C5	00C6	00C7	00C8	00C9	00CA	00CB	00CC	00CD	00CE	00CF
D0	00D0	00D1	00D2	00D3	00D4	00D5	00D6	00D7	00D8	00D9	00DA	00DB	00DC	00DD	00DE	00DF
E0	00E0	00E1	00E2	00E3	00E4	00E5	00E6	00E7	00E8	00E9	00EA	00EB	00EC	00ED	00EE	00EF
F0	00F0	00F1	00F2	00F3	00F4	00F5	00F6	00F7	00F8	00F9	00FA	00FB	00FC	00FD	00FE	00FF

Code pages like 950 are used as both Windows ANSI and OEM code pages. For example, in Traditional Chinese Windows where system language is set to Chinese (Taiwan), both console window and applications like Notepad use code page 950 to interpret non-Unicode text. In this case, the sequence of code points from 0xB0 to 0xBF yields the same display result for both console window and Notepad. There are only eight characters displayed because code points 0xB0 to 0xBF in Windows code page 950 are lead bytes.

噶	播	獄	號	須	斯	眾	停								
0xBF	0xBE	0xBD	0xBC	0xBB	0xBA	0xB9	0xB8	0xB7	0xB6	0xB5	0xB4	0xB3	0xB2	0xB1	0xB0

Windows Codepage 950

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	NUL	STX	SOT	ETX	EOT	ENO	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
10	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
20	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	<	=	>	?	
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL
80		81	82	83	84	85	86	87	88	89	8A	8B	8C	8D	8E	8F
90	90	91	92	93	94	95	96	97	98	99	9A	9B	9C	9D	9E	9F
A0	A0	A1	A2	A3	A4	A5	A6	A7	A8	A9	AA	AB	AC	AD	AE	AF
B0	B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	BA	BB	BC	BD	BE	BF
C0	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CD	CE	CF
D0	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF
E0	E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	EA	EB	EC	ED	EE	EF
F0	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	FA	FB	FC	FD	FE	FF



Windows 950_B0

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
40	度	軟	糾	翊	璽	嶺	蚌	蛭	蛭	衰	東	袁	袂	枉	祇	記
50	許	訖	缸	訕	訕	訕	訕	訕	訕	訕	訕	訕	訕	訕	訕	訕
60	躬	躬	躬	躬	躬	躬	躬	躬	躬	躬	躬	躬	躬	躬	躬	躬
70	郡	郡	郡	郡	郡	郡	郡	郡	郡	郡	郡	郡	郡	郡	郡	郡
80																
90																
A0	停	眾	斯	須	號	獄	播	噶								
B0																
C0																
D0																
E0																
F0																

ISO 8859 Code Pages

The ISO 8859 is a standard for 8-bit encoding and serve as the basis for Windows (ANSI) code pages. Windows code pages are supersets of ISO 8859 code pages, only differ by using printable characters instead of control characters in the 0x80 to 0x9F range. For example, characters like the euro sign (€) and trade mark sign (™) are mapped to this range in Windows code pages.

Below is an example comparison between Windows code page 1252 and ISO 8859-1 code page:

Windows code page 1252:

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	NUL 0000	STX 0001	SOT 0002	ETX 0003	EOT 0004	ENQ 0005	ACK 0006	BEL 0007	BS 0008	HT 0009	LF 000A	VT 000B	FF 000C	CR 000D	SO 000E	SI 000F
10	DLE 0010	DC1 0011	DC2 0012	DC3 0013	DC4 0014	NAK 0015	SYN 0016	ETB 0017	CAN 0018	EM 0019	SUB 001A	ESC 001B	FS 001C	GS 001D	RS 001E	US 001F
20	SP 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL 007F
80	€ 20AC		ƒ 201A	„ 0192	“ 201E	… 2026	† 2020	‡ 2021	~ 02C6	% 2030	Š 0160	< 2039	Œ 0152		Ž 017D	
90		ˆ 2018	˜ 2019	“ 201C	” 201D	• 2022	— 2013	— 2014	™ 02DC	Š 0161	> 203A	œ 0153		Ž 017E	Ÿ 0178	
A0	MSBSP 00A0	¡ 00A1	¢ 00A2	£ 00A3	¤ 00A4	¥ 00A5	¦ 00A6	§ 00A7	¨ 00A8	© 00A9	ª 00AA	« 00AB	¬ 00AC	­ 00AD	® 00AE	¯ 00AF
B0	° 00B0	± 00B1	² 00B2	³ 00B3	´ 00B4	µ 00B5	¶ 00B6	· 00B7	¸ 00B8	¹ 00B9	º 00BA	» 00BB	¼ 00BC	½ 00BD	¾ 00BE	¿ 00BF
C0	À 00C0	Á 00C1	Â 00C2	Ã 00C3	Ä 00C4	Å 00C5	Æ 00C6	Ç 00C7	È 00C8	É 00C9	Ê 00CA	Ë 00CB	Ì 00CC	Í 00CD	Î 00CE	Ï 00CF
D0	Ð 00D0	Ñ 00D1	Ò 00D2	Ó 00D3	Ô 00D4	Õ 00D5	Ö 00D6	× 00D7	Ø 00D8	Ù 00D9	Ú 00DA	Û 00DB	Ü 00DC	Ý 00DD	Þ 00DE	ß 00DF
E0	à 00E0	á 00E1	â 00E2	ã 00E3	ä 00E4	å 00E5	æ 00E6	ç 00E7	è 00E8	é 00E9	ê 00EA	ë 00EB	ì 00EC	í 00ED	î 00EE	ï 00EF
F0	ð 00F0	ñ 00F1	ò 00F2	ó 00F3	ô 00F4	õ 00F5	ö 00F6	÷ 00F7	ø 00F8	ù 00F9	ú 00FA	û 00FB	ü 00FC	ý 00FD	þ 00FE	ÿ 00FF

ISO 8859-1:

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00	NUL 0000	STX 0001	SOT 0002	ETX 0003	EOT 0004	ENQ 0005	ACK 0006	BEL 0007	BS 0008	HT 0009	LF 000A	VT 000B	FF 000C	CR 000D	SO 000E	SI 000F
10	DLE 0010	DC1 0011	DC2 0012	DC3 0013	DC4 0014	NAK 0015	SYN 0016	ETB 0017	CAN 0018	EM 0019	SUB 001A	ESC 001B	FS 001C	GS 001D	RS 001E	US 001F
20	SP 0020	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
30	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
40	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
50	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
60	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
70	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL 007F
80																
90																
A0	MSBSP 00A0	¡ 00A1	¢ 00A2	£ 00A3	¤ 00A4	¥ 00A5	¦ 00A6	§ 00A7	¨ 00A8	© 00A9	ª 00AA	« 00AB	¬ 00AC	­ 00AD	® 00AE	¯ 00AF
B0	° 00B0	± 00B1	² 00B2	³ 00B3	´ 00B4	µ 00B5	¶ 00B6	· 00B7	¸ 00B8	¹ 00B9	º 00BA	» 00BB	¼ 00BC	½ 00BD	¾ 00BE	¿ 00BF
C0	À 00C0	Á 00C1	Â 00C2	Ã 00C3	Ä 00C4	Å 00C5	Æ 00C6	Ç 00C7	È 00C8	É 00C9	Ê 00CA	Ë 00CB	Ì 00CC	Í 00CD	Î 00CE	Ï 00CF
D0	Ð 00D0	Ñ 00D1	Ò 00D2	Ó 00D3	Ô 00D4	Õ 00D5	Ö 00D6	× 00D7	Ø 00D8	Ù 00D9	Ú 00DA	Û 00DB	Ü 00DC	Ý 00DD	Þ 00DE	ß 00DF
E0	à 00E0	á 00E1	â 00E2	ã 00E3	ä 00E4	å 00E5	æ 00E6	ç 00E7	è 00E8	é 00E9	ê 00EA	ë 00EB	ì 00EC	í 00ED	î 00EE	ï 00EF
F0	ð 00F0	ñ 00F1	ò 00F2	ó 00F3	ô 00F4	õ 00F5	ö 00F6	÷ 00F7	ø 00F8	ù 00F9	ú 00FA	û 00FB	ü 00FC	ý 00FD	þ 00FE	ÿ 00FF

Variants of this standard (for example, 8859-2, 8859-5, 8859-13) target different scripts, and each variant corresponds to different Windows code pages.

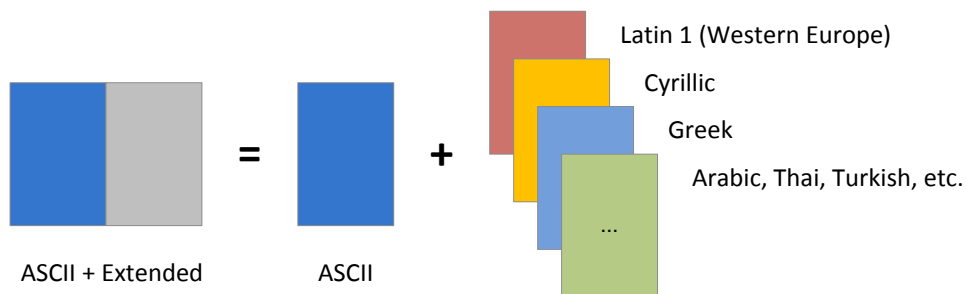
Below lists some example of ISO 8859 code pages and their corresponding Windows code pages.

- ISO-8859-1 (Latin 1) → Windows 28591
- ISO-8859-2 (Latin 2 Central Europe) → Windows 28592
- ISO-8859-15 (Latin 9) → Windows 28605

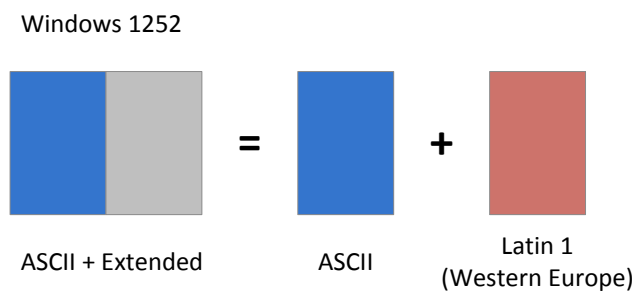
Extended ASCII

Extended ASCII character set corresponds to characters above the ASCII range (32 through 127) in Single Byte Character Set code pages. In Windows code pages, many code points above 0x80 (128) differ between code pages.

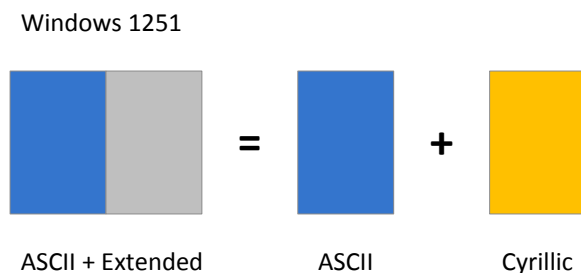
The diagram below illustrates the extended ASCII concept.



For example, in Windows code page 1252 the character set available would be:

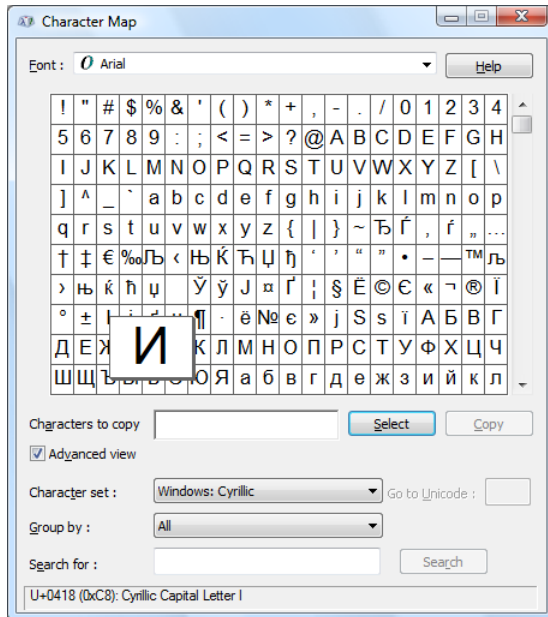


In Windows code page 1251, the character set available would be:

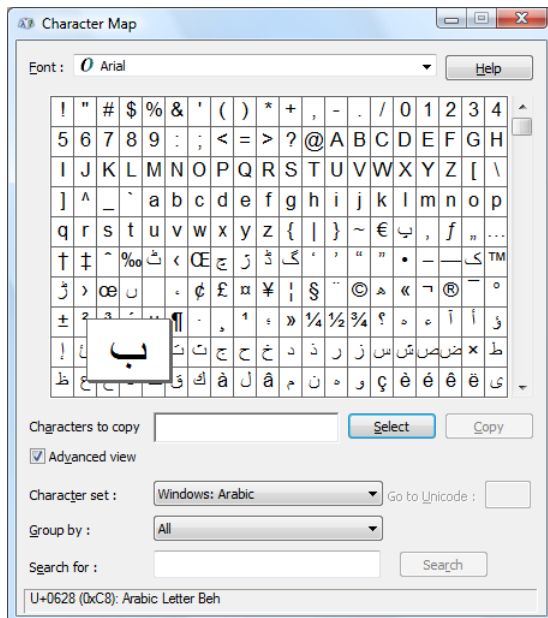


To further illustrate, code point 0xC8 in Windows 1252 code page corresponds to "È". In other code pages code point 0xC8 corresponds to different characters. The diagrams below show the same 0xC8 code point in different Windows code pages rendered with the same font but with different character sets (or "scripts").

Cyrillic:



Arabic:



Unicode

What is Unicode

Unicode is a 16-bit encoding that encompasses over many characters used in general text interchange throughout the world. Each Unicode index refers unambiguously to a given character no matter what the platform, no matter what the program, and no matter what the language.

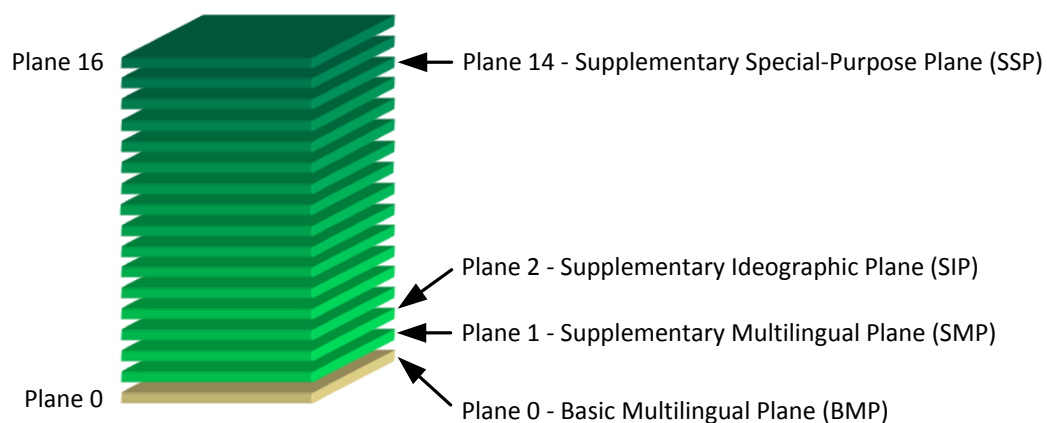
Unicode has a character repertoire, an abstract list of more than one million characters found in a wide variety of scripts including Latin, Cyrillic, Chinese, Korean, Japanese, Hebrew, and Aramaic. Other symbols such as mathematical and musical notations are also included in the character repertoire.

Each character in the character repertoire is assigned to a "code point". Each code point has a specific numerical value, called its scalar value. The scalar value is often expressed in hexadecimal. For example, the character "A" in Unicode is assigned a scalar value of 0x0041.

Code points exist in "code space". The code space consists of a range of scalar values, which are divided across two areas:

- Basic Multilingual Plane (64k in size).
In Unicode, the hexadecimal expression of the values in this lower plane range from 0x0000 to 0xFFFF.
- Supplemental Multilingual Plane (16 additional sections of 64k each).
In Unicode, the hexadecimal expression of the values in this upper plane range from 0x10000 to 0x10FFFF.

The diagram below illustrates the logical stack of code space.



17 Planes (tables) each contain 64K code points
for a total of 1,114,112 possible unique values

Unicode Encoding Forms

Unicode defines a set of unique scalar values for the code points. For example, in Unicode the code point for character "A" has a scalar value of 0x0041, notated as code point "U+0041". The code point for the Ohm sign "Ω" has a scalar value of 0x2126, or code point "U+2126". However, how the scalar value is encoded as data depends on the encoding form.

Unicode defines three character encoding forms (a.k.a. Unicode Transformations):

- UTF-8
- UTF-16
- UTF-32

All three encoding forms can be used to represent the full range of encoded characters in the Unicode Standard.

Note: UTF-8, UTF-16, and UTF-32 are not code pages. They are pseudo code pages in that, for example, when you specify Codepage = 65001 for UTF-8 in a web page, you are requesting the use of the specific Unicode transformation to obtain the resulting encoded character. Character encoding with code pages perform table lookup instead of transformation.

In each of the encoding forms, the Unicode code points are expressed as a sequence of one or more code units. A "code unit" is a single unit within each encoding form. The code unit size is equivalent to the bit measurement for the particular encoding.

- A code unit in UTF-8 consists of 8 bits.
- A code unit in UTF-16 consists of 16 bits.
- A code unit in UTF-32 consists of 32 bits.

The number of code units required to be mapped to a code point varies across encoding forms. For example, when the character "馬" is encoded in UTF-8, the code units that represent the character are 0xE9, 0xA6, and 0xAC. The number of code units in this case is three.

The following sections describe each of the three encoding forms.

UTF-8

UTF-8 encoding form encodes each Unicode code point's scalar value to an unsigned byte sequence of one to four bytes in length, as specified in the following table.

Unicode Scalar Value	Byte 1	Byte 2	Byte 3	Byte 4
00000000 0xxxxxxx	0xxxxxxx			
00000yyy yyxxxxxx	110yyyyy	10xxxxxx		
zzzzyyyy yyxxxxxx	1110zzzz	10yyyyyy	10xxxxxx	
000uuuuu zzzzyyyy yyxxxxxx	11110uuu	10uuzzzz	10yyyyyy	10xxxxxx

Below lists some examples of code points that are encoded in UTF-8 using the specification above.

Code Point	Character	Code Point Bit Pattern	Resulting UTF-8 Encoded Bytes Bit Pattern
U+0041	A	00000000 01000001	01000001
U+00A1	¡	00000000 10100001	11000010 10100001
U+99AC	馬	10011001 10101100	11101001 10100110 10101100
U+200D9	𐤓	00000010 00000000 11011001	11110000 10100000 10000011 10011001

Note that a set of well-formed byte sequences can be deduced from the encoding form specification above. The following table lists all of the byte sequences that are well-formed in UTF-8. A range of byte values such as A0..BF indicates that any byte from 0xA0 to 0xBF (inclusive) is well-formed in that position. Any byte value outside of the ranges listed is ill-formed.

Code Points	Byte 1	Byte 2	Byte 3	Byte 4
U+0000..U+007F	00..7F			
U+0080..U+07FF	C2..DF	80..BF		
U+0800..U+FFFF	E0	A0..BF	80..BF	
U+1000..U+CFFF	E1..EC	80..BF	80..BF	
U+D000..U+D7FF	ED	80..9F	80..BF	
U+E000..U+FFFF	EE..EF	80..BF	80..BF	
U+10000..U+3FFFF	F0	90..BF	80..BF	80..BF
U+40000..U+FFFFF	F1..F3	80..BF	80..BF	80..BF
U+100000..U+10FFFF	F4	80..8F	80..BF	80..BF

From the table above, the following values are disallowed in UTF-8: C0–C1, F5–FF.

UTF-16

UTF-16 encoding form assigns each Unicode scalar value in the ranges 0x0000 to 0xD7FF and 0xE000 to 0xFFFF to a single unsigned 16-bit code unit with the same numeric value as the Unicode scalar value, and assigns each Unicode scalar value in the range 0x10000 to 0x10FFFF to a surrogate pair.

Code units in the range 0xD800 to 0xDFFF are surrogate code points. Values in the range 0xD800 to 0xDBFF are for the first, most significant surrogate ("high surrogate") and 0xDC00 to 0xDFFF for the second, least significant surrogate ("low surrogate"). High surrogate code units are used in UTF-16 as the leading code unit of a surrogate pair. Low surrogate code units are used in UTF-16 as the trailing code unit of a surrogate pair.

In other words:

1. Characters in Plane 0, the Basic Multilingual Plane (BMP), when encoded in UTF-16 encoding form, result in a single 16-bit word that has the same value as the scalar value.

- For characters in the other planes, the encoding will result in a pair of 16-bit words, a surrogate pair, with the first 16-bit word in the range 0xD800 to 0xDBFF and the second 16-bit word in the range 0xDC00 to 0xDFFF.

The table below specifies the bit distribution for the UTF-16 encoding form. Recall that for Unicode scalar values equal to or greater than 0x10000, UTF-16 requires surrogate pairs. Calculation of the surrogate pair values involves subtraction of 0x10000 to account for the starting offset to the scalar value.

Scalar Value Bit Pattern	Resulting UTF-16 Encoded Bytes Bit Pattern
xxxxxxx xxxxxxx	xxxxxxx xxxxxxx
000uuuuu xxxxxxx xxxxxxx	110110ww wwxxxxx 110111xx xxxxxxx

Note: www = uuuu - 1

Below lists some examples of code points that are encoded in UTF-16 using the specification above.

Characters in Plane 0, the Basic Multilingual Plane (BMP), when encoded with UTF-16 encoding form, result in a single 16-bit word that has the same value as the scalar value.

Code Point	Character	Code Point Bit Pattern	Resulting UTF-16 Encoded Bytes Bit Pattern
U+0041	A	00000000 01000001	00000000 01000001
U+99AC	馬	10011001 10101100	10011001 10101100

For characters in the range 0x10000..0x10FFFF, the encoding will result in a pair of 16-bit words, a surrogate pair, with the first 16-bit word in the range 0xD800 to 0xDBFF and the second 16-bit word in the range 0xDC00 to 0xDFFF.

Code Point	Character	Code Point Bit Pattern	Resulting UTF-16 Encoded Surrogate Pair Bit Pattern
U+20058	𪛗	00000010 00000000 01011000	11011000 01000000 11011100 01011000
U+200D9	𪛙	00000010 00000000 11011001	11011000 01000000 11011100 11011001

Note: The following formula can be used to convert a surrogate pair (two 16-bit words) into a 32-bit Unicode scalar value (or UTF-32):

(High Surrogate - 0xD800) * 0x400 + (Low Surrogate - 0xDC00) + 0x10000

The following formula can be used to convert a Unicode scalar value (S) to surrogate pair:

High Surrogate = (S - 0x10000) / 0x400 + 0xD800

Low Surrogate = (S - 0x10000) % 0x400 + 0xDC00

UTF-32

UTF-32 encoding form assigns each Unicode scalar value to a single unsigned 32-bit code unit with the same numeric value as the Unicode scalar value.

Because surrogate code points are not included in the set of Unicode scalar values, UTF-32 code units in the range 0x0000D800 to 0x0000DFFF are ill-formed. Also any UTF-32 code unit greater than 0x0010FFFF is ill-formed.

Unicode Encoding Schemes

Unicode Encoding Scheme is a specified byte serialization for a Unicode encoding form, including the specification of the handling of a byte order mark (BOM), if allowed. The Unicode Standard defines seven encoding schemes, listed below.

- UTF-8
- UTF-16
- UTF-16BE (Big-Endian)
- UTF-16LE (Little-Endian)
- UTF-32
- UTF-32BE (Big-Endian)
- UTF-32LE (Little-Endian)

The terms UTF-8, UTF-16, and UTF-32, when used unqualified, could be ambiguous as they can mean Unicode encoding forms or Unicode encoding schemes.

For UTF-8, this ambiguity is usually innocuous, because the UTF-8 encoding scheme is trivially derived from the byte sequences defined for the UTF-8 encoding form. However, for UTF-16 and UTF-32, the ambiguity is more problematic.

As encoding forms, UTF-16 and UTF-32 refer to code units in memory; there is no associated byte orientation, and a BOM is never used. As encoding schemes, UTF-16 and UTF-32 refer to serialized bytes, as for streaming data or in files; they may have either byte orientation, and a BOM may be present.

When the usage of the short terms "UTF-16" or "UTF-32" might be misinterpreted, and where a distinction between their use as referring to Unicode encoding forms or to Unicode encoding schemes is important, the full terms should be used. For example, use UTF-16 encoding form or UTF-16 encoding scheme. These terms may also be abbreviated to UTF-16 CEF or UTF-16 CES, respectively.

The table below gives an example of the resulting byte streams of characters "繁體中文" encoded in Windows code page 950 ("Big5"), UTF-8, and UTF-16LE.

Character	Unicode Code Point	Windows Code Page 950	UTF-8	UTF-16LE
繁	U+7E41	0xC1 0x63	0xE7 0xB9 0x81	0x41 0x7E
體	U+9AD4	0xC5 0xE9	0xE9 0xAB 0x94	0xD4 0x9A
中	U+4E2D	0xA4 0xA4	0xE4 0xB8 0xAD	0x2D 0x4E
文	U+6587	0xA4 0xE5	0xE6 0x96 0x87	0x87 0x65

The characters all fall into the Unicode range 0x4E00 to 0x9FFF, which is CJK Unified Ideographs, the range of Unicode code points assigned for ideographs used by Chinese characters.

Byte Order Mark (BOM)

A Byte Order Mark can be placed at the beginning of a file or data stream to distinguish between byte orders and to define the encoding scheme. Notepad, for example, uses the BOM prefix to clearly define the encoding of a Unicode encoded text file.

The table below lists the encodings and their associated BOM:

Encoding	Byte Order Mark
UTF-8	0xEF 0xBB 0xBF
UTF-16 (Big-Endian)	0xFE 0xFF
UTF-16 (Little-Endian)	0xFF 0xFE
UTF-32 (Big-Endian)	0x00 0x00 0xFE 0xFF
UTF-32 (Little-Endian)	0xFF 0xFE 0x00 0x00

Note: UTF-16 Little-Endian is the encoding scheme generally used in Windows, commonly abbreviated as just "Unicode".

Technically there is no need for a byte order signature when using UTF-8 because the main purpose of BOM is to define the ordering of bytes. Its usage at the beginning of a UTF-8 data stream or file is neither required nor explicitly recommended by the Unicode Standard, but its presence does not affect conformance to the UTF-8 encoding scheme.

Having a BOM prefix helps to indicate that the file or data stream is using the UTF-8 encoding scheme, which avoids ambiguity. However, when you use such UTF-8 encoded files in some applications, you may get an extra line or unwanted characters at the beginning of the file. This might not be a problem depending on the type of your application. Notepad, for example, takes BOM prefix into consideration.

Note: For more information on choosing where to save UTF-8 encoded files with BOM prefix, you can reference the following resource:

FAQ: Display problems caused by the UTF-8 BOM

<http://www.w3.org/International/questions/qa-utf8-bom>

Appendix F: ISO/IEC 10646 and Unicode

The International Organization for Standardization (ISO) and the Unicode Consortium both agreed to develop a single universal character code standard. The character code standard defines a set of characters (the repertoire), with each character assigned an unambiguous name and an integer number called code point expressed in hexadecimal.

The same characters with the same code points exist in both ISO/IEC 10646 and Unicode standards of comparable versions. For example, the character "A" has a code point of U+0041 named "LATIN CAPITAL LETTER A" in both ISO and Unicode standards. The code point of U+0041 has the same meaning no matter which language you use.

The table below shows the correlation between ISO/IEC 10646 versions and Unicode Standard versions:

ISO/IEC 10646	Unicode Standard
ISO/IEC 10646-1:1993	Unicode 1.1
ISO/IEC 10646-1:1993 + Amendments	Unicode 2.0
ISO/IEC 10646-1:2000	Unicode 3.0
ISO/IEC 10646-2:2001	Unicode 3.1
ISO/IEC 10646:2003	Unicode 4.0
ISO/IEC 10646:2003 + Amendment 1	Unicode 4.1
ISO/IEC 10646:2003 + Amendments 1 and 2	Unicode 5.0
ISO/IEC 10646:2003 + Amendments 1 to 4	Unicode 5.1
ISO/IEC 10646:2003 + Amendments 1 to 6	Unicode 5.2

More information about the correlation can be found in the Unicode Standard publication, Appendix C Relationship to ISO/IEC 10646 at <http://www.unicode.org/versions/Unicode5.2.0/>.

The ISO/IEC 10646:2003 specification and related amendments can be found at <http://www.iso.org/iso/search.htm?qt=10646&sort=rel&type=simple&published=on>

Code charts for recent Unicode versions can be found at <http://www.unicode.org/charts/About.html>.

- Code charts for Unicode 4.1: <http://www.unicode.org/charts/PDF/Unicode-4.1/>
- Code charts for Unicode 3.0: <http://www.unicode.org/Public/3.0-Update/>

References

Windows XP Font Pack for ISO 10646:2003 + Amendment 1 Traditional Chinese Support is available for Windows XP and Windows Server 2003

<http://support.microsoft.com/kb/977801>

Windows XP Font Pack for ISO 10646:2003 + Amendment 1 Traditional Chinese Support

<http://www.microsoft.com/downloads/details.aspx?displaylang=en&FamilyID=1112aa2c-e011-471c-a12c-656e767d3bb8>

Microsoft HKSCS Support Packages for Windows platform

<http://www.microsoft.com/hk/hkscs/>

Microsoft Character Code Conversion Routines for HKSCS-2004

<http://www.microsoft.com/downloads/details.aspx?FamilyID=0e6f5ac8-7baa-4571-b8e8-78b3b776afd7>

HKSCS-2004 Specification

http://www.ogcio.gov.hk/ccli/eng/hkscs/download/e_hkscs_2004.pdf

New contents of the HKSCS-2004

<http://www.ogcio.gov.hk/ccli/eng/hkscs/download/123for2004.pdf>

OGCIO HKSCS Documents

<http://www.ogcio.gov.hk/ccli/eng/hkscs/document.html>

