Übersichtsblatt:
Unicode

Grundidee:

Bisher gab es für bestimmte geographische Regionen immer einen bestimmten Zeichenvorrat. So hatten die Amerikaner beispielsweise ihr US-ASCII und die Chinesen hatten ihre Schriftzeichen. Im europäischen Raum war das Alphabet in der Regel sehr dicht an US-ASCII, aber in bestimmten Einzelfällen, z.B. die deutschen Umlaute, genügte der ASCII-Vorrat nicht um alle benötigten Zeichen darzustellen.

Unicode ist der Versuch einen international einheitlichen Zeichenvorrat für alle Regionen bereitzustellen.

UTF-16:

Zunächst war die Idee den benötigen Platz für die neuen Zeichen einfach dadurch zu erhalten, dass man für jedes Zeichen ein zusätzliches Byte bereitstellt. Ein Zeichen wird jetzt also nicht mehr wie z.B. bei US-ASCII mit 8 sonder mit 16 Bit Codiert.

Zur automatischen Erkennung jeder UTF-16 Datei ist das erste Byte immer: U+FEFF

Vorteile:

- Es kann ein wesentlich größerer Zeichenvorrat bereitgestellt werden.

Nachteile:

- Jedes Zeichen verbraucht nun doppelt so viel Platz.
- Der erzeugte Bereich genügt immer noch nicht um wirklich annähernd alle Zeichen (auch Chinesisch, Japanisch oder Koreanische) unterzubringen.

UTF-8:

Daher wurde die Nachfolgeversion UTF-8 entwickelt. Hierbei werden jedem Zeichn zwischen 1 und 6 Byte zugeordnet. Diese Variable Codelänge ermöglicht es gleichzeitig einen enorm viel größeren Zeichenvorrat bereitzustellen als bei US-ASCII oder UTF-16 und gleichzeitig den Speicheraufwand dafür nicht größer werden zu lassen als unbedingt nötig.

Anzahl Bytes:	Unicode Zeichenbereich:	UTF-8 Codierung:
1	U+00000000 bis U+0000007F	0xxxxxxx
2	U+00000080 bis U+000007FF	110xxxxx 10xxxxxx
3	U+00000800 bis U+0000FFFF	1110xxxx 10xxxxxx 10xxxxxx
4	U+00010000 bis U+001FFFFF	11110xxx 10xxxxxx 10xxxxxx (noch eins Byte)
5	U+00200000 bis U+03FFFFFF	111110xx 10xxxxxx 10xxxxxx (noch zwei Byte)
6	U+04000000 bis U+7FFFFFFF	1111110x 10xxxxxx 10xxxxxx (noch drei Byte)

Die x Stellen werden dann mit den entsprechenden Unicodezeichen aufgefüllt.