# White Paper
## Unicode im BS2000

Im BS2000 unterstützen wichtige System- und systemnahen Programme jetzt auch UNICODE-Zeichensätze.
Ziel der Entwicklungsmaßnahme war, die derzeitigen für BS2000 verfügbaren EBCDIC-Varianten um die im westeuropäischen Sprachraum relevanten Unicode-Zeichen zu ergänzen. Bei Bedarf kann auch kyrillisch unterstützt werden.
Für die BS2000 systeminterne Kommunikation werden die unterstützten Zeichensätze um UTF-16 (im Umfang von UCS-2) ergänzt. Für die Sortierung und Normalisierung stehen standardmäßig Tabellen zur Verfügung, die die Zeichen innerhalb der Codepoints U+0000 bis U+2FFF unterstützen.
Damit ist sichergestellt, daß alle europäischen Zeichen einschließlich der in den Meldezeichen der öffentlichen Verwaltungen verwendeten Sonderzeichen unterstützt werden, da sie vollständig im unterstützten Wertebereich enthalten sind.

**Inhalt**

## Motivation

Einige EU-Richtlinien und deren Umsetzung in nationale Gesetzgebungen, zwingen unsere Kunden, ihre EDV-Anwendungen an folgende Richtlinien anzupassen:

- Jeder EU-Bürger hat Anspruch auf korrekte Schreibung seines Namens in lateinischer Schrift, diakritische Zeichen eingeschlossen,
- Internationale Postrichtlinie für die korrekte Schreibweise einer Auslandsanschrift,
- BundOnline 2005-Richtlinie,
- Richtlinie des Bundesministerium des Inneren für den Datenaustausch zwischen Behörden.

Ziel von Unicode ist, für alle weltweit vorkommenden Zeichen eine weltweit eindeutige Codierung festzulegen. Die BS2000-Implementierung leistet dies für den westeuropäischen Raum.

Einzelheiten zu Unicode (inklusive Glossar) siehe Unicode Homepage http://www.unicode.org/

## Zielsetzung

Mit der Unicode-Unterstützung im BS2000 werden die in BS2000-Systemen verfügbaren EBCDIC-Zeichensätze um zusätzliche Zeichen erweitert, die im europäischen Sprachraum künftig benötigt werden. Dies erfolgt durch den Einsatz ausgewählter Unicode-Codepoints additiv zu den bisherigen EBCDIC-Varianten.
Durch die Unicode-Unterstützung wird die weitere Nutzung von BS2000-Systemen auch unter den erweiterten Anforderungen an die zu unterstützenden Codezeichen gewährleistet. Im Sinne von evolutionärer Erweiterung geschieht dies unter weitgehender Sicherung des investierten Vermögenswertes der kundenseitigen Anwendungsbeständen.

## Art der Entwicklung

Erweiterung der in BS2000-Systemen verfügbaren EBCDIC-Zeichensätze um zusätzliche Zeichen / Glyphen. Dies wird durch den Einsatz ausgewählter Unicode Codepoints additiv zu den bisherigen EBCDIC-Varianten erfolgen.
Wenn im Folgenden von Unicode Unterstützung im BS2000 die Rede ist, so ist hier ausschließlich diese Erweiterung des Codesets gemeint.

Im Vordergrund steht die Überlegung, was benötigt der Anwender, damit er seine bestehenden Anwendungen um Unicode-Datenfelder erweitern kann. Hierbei wird davon ausgegangen, dass die Anzahl der Felder, die auf Unicode umgestellt oder die zusätzlich eingefügt werden müssen, gering ist. Es wird sich im allgemeinen um Namens- und Adressfelder handeln.

Wesentliches Ziel ist, dass der Kunde nur solche Anwendungen / Anwendungsteile modifizieren muss, die auch die erweiterte Funktionalität nutzen. Ein Mischbetrieb mit den bisherigen („alten") Anwendungen muss sichergestellt werden.

## Ausgangssituation im BS2000

Das BS2000 verwendet bei der Bearbeitung von Daten standardmäßig einen 7-Bit EBCDIC-Zeichensatz. Die von Fujitsu definierte internationale 7-Bit EBCDIC-Tabelle, die als System-Standard-Zeichensatz verwendet wird, heißt EDF03IRV. Mit ihr stehen 95 verschiedene darstellbare bzw. abruckbare Zeichen zur Verfügung.
Mittels XHCS unterstützt das BS2000 die Konvertierung einer Reihe von 8-bit ISO-Codes (ISO 8859-x Varianten) in die zugehörigen EBCDIC-Codes, die das System bei der Verarbeitung verwendet.

## Unicode im BS2000-Systemumfeld

In Unicode finden Zeichen der wichtigsten Industriestandard-Zeichensätze wie die ISO-Normen eine 1:1-Entsprechung (das bedeutet, dass bei einer Konvertierung von EBCDIC zu Unicode und zurück das gleiche Ergebnis herauskommt).
Die Speicherung und Übertragung von Unicode erfolgt in unterschiedlichen Formaten. UTF (Unicode Transformation Format) beschreibt Methoden, ein Unicode-Zeichen auf eine Folge von Bytes abzubilden. Jeder Character einer Unicode-Zeichenfolge wird dabei durch eine oder mehrere "code units" dargestellt.

- **UTF-8** ist die verbreitetste Kodierung für Unicode-Zeichen und für alle auf dem Lateinischen Alphabet basierenden Schriften die platzsparendste Methode zur Abbildung von Unicode-Zeichen. Eine Code unit sind 8 bit; jeder Unicode Character braucht 1, 2, 3, oder 4 Bytes.
- **UTF-16** hat neben UTF-8 eine große Bedeutung, so z.B. als Zeichencodierung in Java. Eine Code unit sind 16 bit; jeder Unicode Character braucht entweder 2 oder 4 Byte. UTF-16 ist weitestgehend identisch mit der 2-Byte-Unicode-Darstellung UCS-2.
- **UTF-32** ist eine Kodierung mit konstanter Anzahl Bytes pro Unicode-Zeichen. UTF-32 hat code units mit 32 bit; jeder Unicode-Character braucht genau 4 Bytes. (Wird im BS2000 nicht unterstützt.)
- **UTF-EBCDIC** ist eine Unicode-Erweiterung, die auf dem proprietären EBCDIC-Format von IBM-Großrechnern aufbaut. Code unit sind 8 bit; jeder Unicode-Character braucht 1, 2, 3, 4, oder 5 Byte.
- **UTF-E** wird im BS2000 neu eingeführt; UTF-E im BS2000 ist analog zu UTF-EBCDIC von der IBM und basiert auf einem modifizierten UTF-8.

Im Unterschied zu UTF-8 beginnt die Mehrbytedarstellung bei UTF-EBCDIC und UTF-E erst beim Unicode-Codepoint U+A0. Die ersten 256 Codepoints von Unicode stimmen mit dem Zeichensatz ISO8859-1 überein. D.h. der gesamte zweite Steuerzeichenblock von ISO8859-1 bleibt als Einbyte-Codierung erhalten. Bei der anschließenden Konvertierung von diesem modifierten UTF-8 nach EBCDIC liegt für das BS2000 die Konvertierungstabelle EDF041 zu Grunde.
Die Ein-Byte-Codierung von UTF-E entspricht dem Zeichensatz von EDF03IRV.
Da UTF-E nur eine andere Darstellung von Unicode ist, kann jede andere Unicode Darstellung auf UTF-E abgebildet werden und umgekehrt.

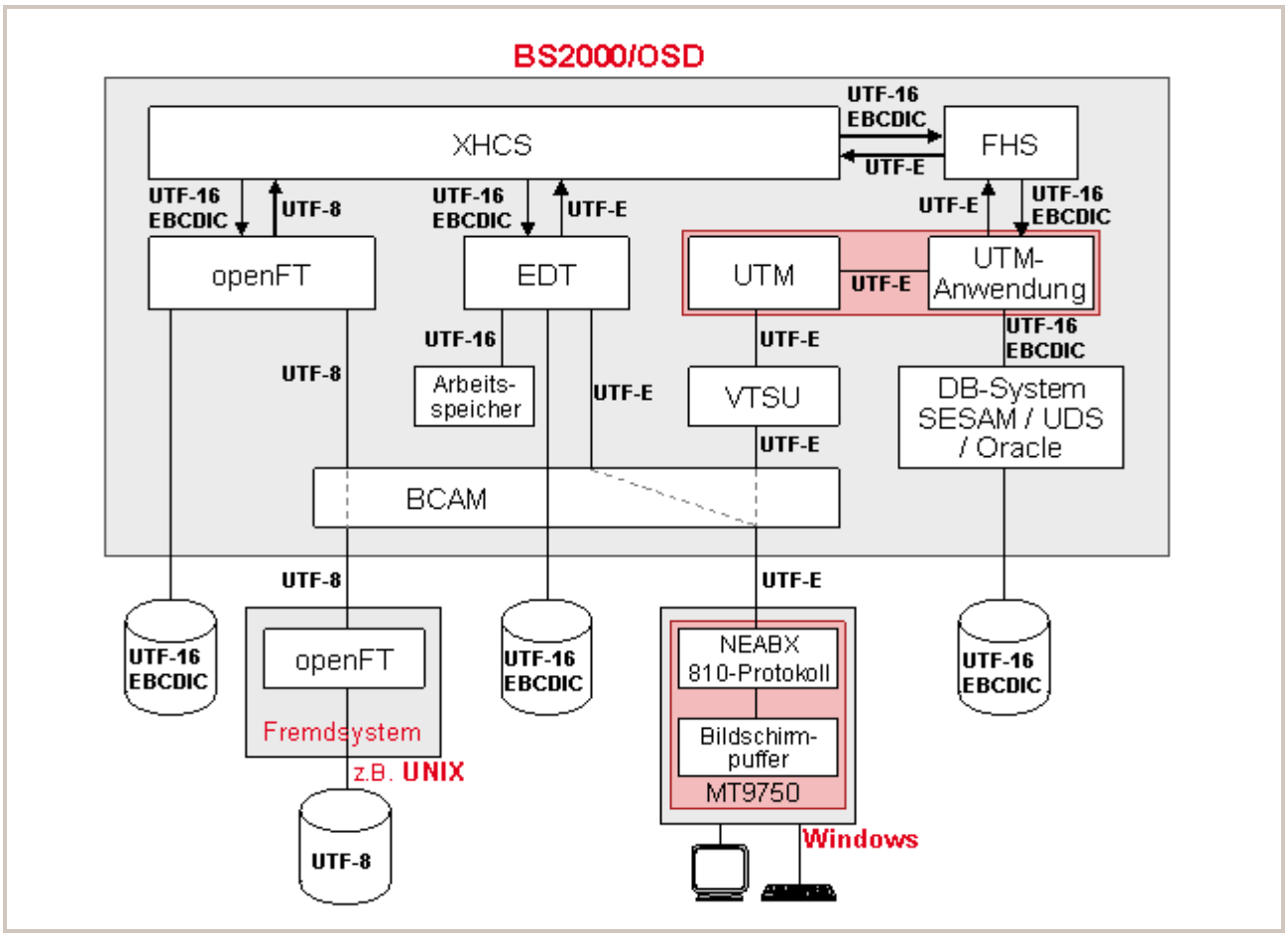
Achtung: Sowohl der Zeichensatz UTF-E bei ORACLE als auch UTF-EBCDIC bei SUN basieren auf den EBCDIC-Tabellen der IBM und unterscheiden sich von dem in BS2000 eingesetzten UTF-E.

Alle Kodierungen mit variabler Anzahl Bytes pro Charakter sind so konstruiert, dass man an beliebiger Stelle im String nicht mehr als 4 Bytes inspizieren muss, um alle Bytes für einen UNICODE-Charakter zu finden. Sie müssen also nicht immer von Anfang an untersucht werden, um die character boundaries an einer Stelle zu finden ("non-overlap").

Basis für die Unicode-Unterstützung im BS2000 sind Erweiterungen im Subsystem XHCS (XHCS ist Bestandteil des Produkts openNet Server).
XHCS

- unterstützt die Codierungsvarianten UTF-8, UTF-16 und UTF-E (variable Länge pro Zeichen).
- umfasst Umsetztabellen der unterstützten EBCDIC-Codes und 8-bit ISO-Codes (ISO8859-1/2/3/4/5/7/9/15) nach Unicode und deren Umkehrabbildung.
- unterstützt weitere Zeichen des Unicode-Zeichensatzes, die im Meldewesen der öffentlichen Verwaltungen als Sonderzeichen verwendet werden - über die in diesen ISO-Codes enthalten Zeichen hinaus.
- Unterstützt für die Normalisierung und Sortierung UTF16-Strings von U+0000 bis U+2FFF.

Unicode-basierte Zeichen werden nur für die zu verarbeitenden / zu verwaltenden Texte bzw. Nutzdaten zugelassen. Für Kommandos und Schlüsselworte werden nur die Zeichen im bisherigen Code-Umfang (EBCDIC) unterstützt.




Unicode im BS2000-Systemumfeld
(Beispielhafte Darstellung für die Dateneingabe, Stand 2007. Für die bessere Übersichtlichkeit wurde auf einige technische Details verzichtet.)

# Unicode im BS2000-Applikationsumfeld

Im Applikationsumfeld sind zwei Arten von Anwendungen zu unterscheiden:

- Systemnahe Anwendungen, die von Fujitsu angeboten und geliefert werden und
- Kundenanwendungen, die spezifisch für den jeweiligen Einsatzfall erstellt werden oder wurden.

Die systemnahen Anwendungen von Fujitsu wurden für die Nutzung von Unicode-Zeichen erweitert.

Die Unicode-Unterstützung der Programmier- und Ablauf-Umgebung erstreckt sich auf folgende Funktionsbereiche:

- Programmierung: COBOL2000, ESQL-COBOL, IFG (Unicode-Datenfelder), AID
- Speicherung von Daten: SESAM/SQL, ORACLE (Unicode-Datenfelder), UDS/SQL (National-Daten)
- Verarbeitung von Daten: EDT, SORT, PERCON, XHCS
- Input/Output: Terminal-Support (VTSU, MT9750, FHS) , Printer-Support (RSO, Spool), File Transfer (openFT)

# Vorbereitungen für eine Unicode-Nutzung

Zusätzlich zur Hochrüstung der BS2000-Systembasis und der systemnahen Anwendungen, sowie der Programmierumgebung auf die erforderlichen Versionen sind vom Kunden auch weitere Vorbereitungen seinerseits zu erbringen. Dabei sind folgende Punkt besonders zu beachten:

- Die Daten und Dateien sind in einen definierten Zustand zu bringen.
- Die im BS2000-System verwendeten CCS-Namen (Coded Character Sets) sind zentraler Bestandteil bei der Behandlung von Daten. Es ist sicher zu stellen, dass der Inhalt und die Bezeichnung der Daten mittels CCS-Namen konsistent ist.
- Die betroffenen Anwendungen sind für die zusätzlich zu verarbeitenden Unicode-Datenfelder zu erweitern.

# Kurzbeschreibung der Unicode-spezifischen Erweiterungen in den einzelnen Produkten

Die Unicode-spezifischen Erweiterungen der einzelnen Produkte können detailiert in den entsprechenden Manualen nachgelesen werden.

## AID

AID bietet eine Low-Level-Unterstützung des Datentyps UTF-16 an, der auch als neuer Datentyp von COBOL unterstützt wird.
Zusätzlich bietet AID bei der Unterstützung durch COBOL den Datentyp UTF-16 auch im LSD (List of symbolic debugging) an. Damit kann dieser Datentyp auch symbolisch getestet werden.

## COBOL2000

Zur Unterstützung der Unicode Funktionalität im BS2000 wird in COBOL2000 der Datentyp NATIONAL (PIC N) in seinen verschiedenen Ausprägungen und die prozedurale Verwendung dieser Daten in Compiler und Laufzeitsystem eingeführt. Die Darstellung der Zeichen erfolgt in UTF-16. Der Zeichenvorrat (Characterset) ist beschränkt auf UCS-2. D.h. es wird vom Compiler nur die zwei Byte Darstellung unterstützt ohne Surrogate.

## EDT

EDT kann ab Version 17.0 Dateien mit mehreren unterschiedlichen Zeichensätzen gleichzeitig bearbeiten. Dabei liegen alle Daten, die der EDT in seinem Bildschirmpuffer einliest bzw. ausgibt, anschließend in UTF-E vor. Intern bearbeitet der EDT die Arbeitsdatei im Hauptspeicher in UTF-16. Verarbeitet werden können Dateien mit den Charactersets EBCDIC (wie heute), UTF-16 und UTF-E.

## ESQL-COBOL

Der Precompiler unterstützt die neuen Datentypen NCHAR und NVARCHAR im Funktionsumfang von SESAM.
ESQL-COBOL erlaubt den Datentyp NATIONAL von COBOL in seinen Declare Sections und lässt seine Verwendung als Hostvariable in SQL Anweisungen zur Weitergabe an die ICSQLE Schnittstelle zu.

## FHS

Sollen in einer Maske neben den heutigen EBCDIC-Feldern UTF-16 Felder eingelesen werden, so kann FHS den Bildschirmpuffer nicht mehr in EBCDIC-Codierung sondern muss ihn in UTF-E anfordern.
Die Konvertierung und Ausgabe eines Formates als Unicode-codierte Nachricht (UTF-E) erfolgt durch FHS, sofern ein Feld des FHS-Formats als ein für Unicode-Ein- oder Ausgaben zulässiges Feld definiert ist. Bei der Eingabe wird der Eingabe-String analysiert und in das für jedes Feld von IFG definierte Coded Characterset (EBCDIC bzw. UTF16) umgewandelt. Für die Codeumsetzung wird XHCS genutzt.

## IFG

IFG ermöglicht für einzelne Felder eines Formats eine Kennzeichnung für Unicode. Dazu wird ein neues Attribut „Unicode" eingeführt. In diesen Feldern werden dann die Inhalte als 2-Byte UTF-16 Characters durch FHS interpretiert. Die Generierung der FHS-Formate und Adressierungshilfen

werden dazu entsprechend angepasst. Formate die Unicodefelder enthalten, können nur von Terminalemulationen ausgegeben werden, die Unicode (UTF-E) unterstützten.

## MT9750 - Terminalemulation

Die MT9750 Terminalemulation unterstützt Unicode für europäische Zeichen im Zeichenumfang von U+0000 bis U+2FFF. Durch dieses Feature ist es möglich, Zeichen aus verschiedenen Character-Sets in einem Formular anzeigen zu können.Im Unicode-Modus können auf einer Tastatur nicht verfügbare Zeichen eines Unicode-Zeichensatzes durch die COMPOSE-Tastenfunktion erzeugt werden. Darüber hinaus können Unicode-Zeichen über die Zwischenablage eingefügt werden.

## OMNIS

OMNIS unterstützt ab V8.5A den Nachrichtenverkehr im Unicode Zeichensatz.

## openFT

openFT für BS2000 unterstützt die Übertragung von DMS- und POSIX-Dateien mit den Unicode-Varianten UTF-16 (Big Endian) und UTF-8. Die Spezifikation der Unicode-Variante erfolgt wahlweise über eine Parameterangabe im TRANSFER-FILE Kommando oder über eine XHCS-Codeset Angabe im Dateikatalog (nur bei DMS-Dateien).

## ORACLE

Oracle-BS2000 bietet die Möglichkeit, UTF-16-Daten in der Datenbank zu speichern, und zwar in Feldern mit bestimmten NCHAR Datentypen. Die Verarbeitung dieser Felder ist im BS2000 mit Ausnahme von Pro*Cobol ebenfalls möglich. Die Beschränkung auf die ausschließliche Verwendung von Unicode-Zeichen im BS2000 auf den Textinhalt ist besonders zu beachten.
In Oracle Database 10g können Unicode-Zeichen im UTF8- bzw. UTF16-Format in Feldern der Datentypen NCHAR, NVARCHAR2 und NCLOB gespeichert und mittels SQL und PL/SQL bearbeitet werden. Unicode-Daten können auch mit den Utilities DataPump, Export, Import, SQL*Loader sowie über Anwendungen mittels der Schnittstellen Pro*Cobol, Pro*C und OCI bearbeitet werden. Zur Konvertierung werden verschiedene in Oracle definierte ASCII-, EBCDIC- und Unicode-Zeichensätze verwendet.

## PERCON

In PERCON wurden Erweiterungen zur Konvertierung zwischen Unicode-Daten und anderen Daten (mit kompatiblen CCS) umgesetzt. Die Normalisierung von Unicode-Daten (composed) wird zusätzlich unterstützt.

## RSO

Unterstützung von Unicode-Druckern in RSO, die als Netzdrucker UTF8 codierte Daten akzeptieren. Folgende Codierungen werden unterstützt:
UTF-8 (einschließlich VTSU und RSO Font-Control Zeichen)
UTF-E (einschließlich VTSU und RSO Font-Control Zeichen)
UTF-16 (Textdateien mit und ohne Vorschubzeichen)

Die Fälle 1 und 2 betreffen teilweise die Schnittstelle zu UTM; die Erweiterung ermöglicht das Drucken von Unicode-Text .
Der 3. Fall erlaubt das Drucken von Unicode-Dokumenten auf lokalen (IPDS) und RSO-Druckern.

## SESAM/SQL

Im DB-System SESAM/SQL wird durch Einführung der Datentypen NCHAR und NVARCHAR die Möglichkeit geschaffen, auch Unicode-Zeichen abzuspeichern und mit SQL-Mitteln zu bearbeiten. Hierbei wird neben der Unterstützung der neuen Datentypen durch die DML-Sprachen auch die Nutzung in den verschiedenen Dienstfunktionen (z.B. LOAD, UNLOAD, IMPORT, EXPORT) ermöglicht. Für die Umsetzung von z.B. CHAR zu NCHAR bzw. vice versus bedient sich SESAM/SQL der von XHCS bereitgestellten Aufruffunktionen. Der SESAM-interne Vergleich von Daten erfolgt auch bei nationalen Daten immer binär.

## SORT

SORT unterstützt derzeit eine dreistufige Sortierung von UTF-16-Feldern entsprechend der Unicode-Norm (s. unter http://www.unicode.org/reports/tr10/tr10-9.html). Jedem Unicode-Zeichen wird ein Sortierelement zugeteilt. Die Sortierelemente werden mittels einer von XHCS gelieferten Tabelle (Unicode Default Collation Table) festgelegt.

## UDS/SQL

UDS/SQL ab V2.5 erlaubt die Speicherung und Wiedergewinnung von Unicode-Zeichen in den Datenbanken von UDS/SQL.
In UDS/SQL-Datenbanken können Felder vom Datentyp NATIONAL CHARACTER und NATIONAL CHARACTER VARYING definiert und darin Unicode-Daten in UTF-16-Format gespeichert werden. Metadaten wie beispielsweise Namen von Satzarten, Sets und Realms werden weiterhin in EBCDIC angegeben und gespeichert.
Im Rahmen einer Umstrukturierung neu angelegte Felder vom Datentyp NATIONAL werden mit dem nationalen Leerzeichen initialisiert.

## VTSU

Erweiterung der Funktionalität zur Erkennung einer Unicode-Unterstützung von Partnern, Unicode-Texterkennung, Verarbeitung und Umwandlung mit Hilfe von XHCS.

## XHCS

XHCS ist als Subsystem realisiert. Es wurde um die Unicode-Transformation-Formate UTF-8, UTF-16, UTF-E erweitert.
Es informiert über die Kompatibilität der Codes und die Möglichkeit der Umsetzung. Die Funktionen zur Umwandlung der Zeichensätze mit den erforderlichen Sortiertabellen, zur Wandlung von Klein-/Großschreibung, und zur Unterstützung von Encoding forms in Unicode stehen auch Anwenderprogrammen zur Verfügung.

# Querverweise

- White Paper zu BS2000/OSD-BC V7.0
- Übersichtshandbuch zu 'Unicode im BS2000/OSD' - siehe Manual-Server
- Unicode Homepage http://www.unicode.org/
- The Unicode Standard: A Technical Introduction http://www.unicode.org/standard/principles.html

Kontakt:
Fujitsu
Barbara Stadler
Mies-van-der-Rohe-Straße 8, 80807 München
Deutschland
Telefon: +49 (0)89-62060-1978
E-mail: barbara.stadler@ts.fujitsu.com
Website: de.fujitsu.com
13. August 2015 EM DE