

Unicode

Neues aus dem GSE-Arbeitskreis BUNID und der FG String.Latin



Wer bin ich?



Teamleiterin bei Datev

RZ Application Services

Zuständig für:

- Sprachen am Mainframe (Assembler, Cobol, Java auf z)
 - Entwicklungsumgebungen (ID/z, SEWB, git)
 - Architekturberatungen zur Modernisierung
 - Basissoftware (generelle Module etc.)
 - Vizechairman der BUNID
- Teilnehmer der FG String.Latin

„Es gibt nur eins, was auf Dauer teurer ist als Bildung, keine Bildung.“

John. F. Kennedy, amerikanischer Präsident

Agenda

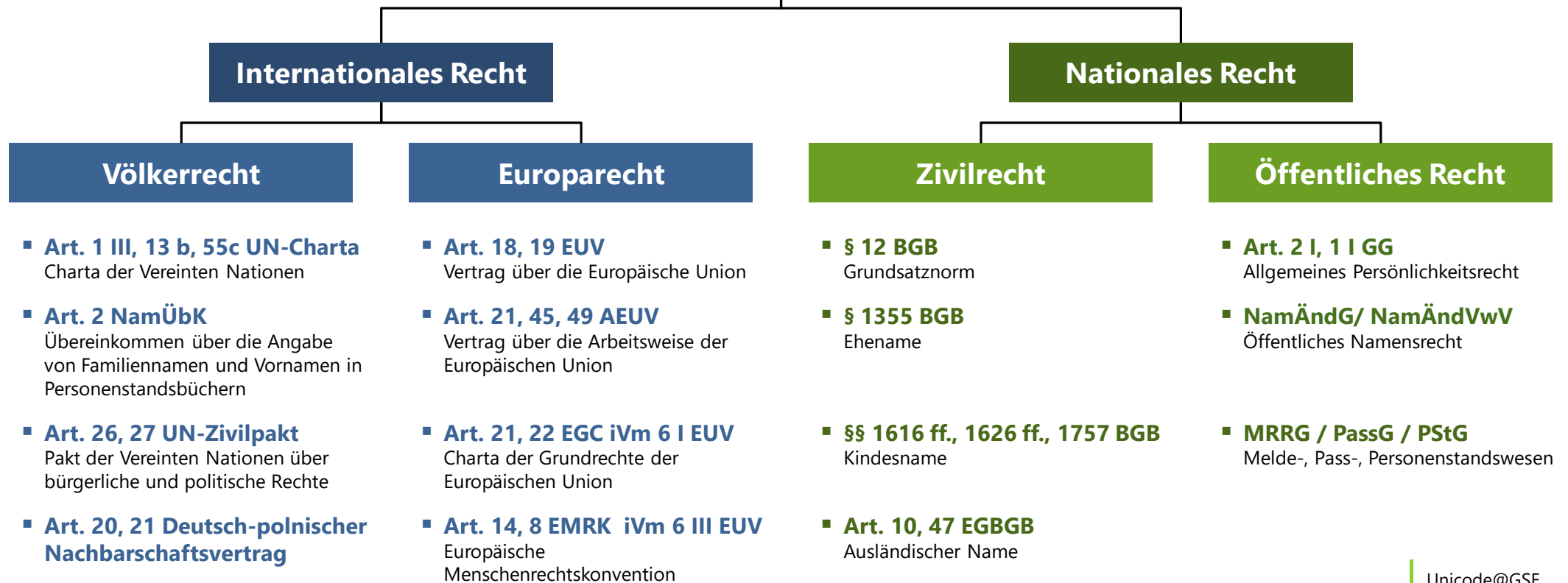
- Warum beschäftigen wir uns mit Unicode?
 - Rechtliche Einordnung
 - Entscheidung IT-Planungsrat
 - KoSIT -> FG String.Latin
 - Historie
 - KoSIT Standard
 - Datev-Set
- GSE BUNID
 - Was macht die BUNID?
- FG String.Latin
 - String Latin 1.2
 - DIN Spec
- Eingabe/Fonts/Druck
- Unicode@Datev

Warum beschäftigen wir uns mit Unicode?

„Globaler Wettbewerb bedeutet auch globaler Bildungswettbewerb.“

Willi Berchthold, Branchenverband BITKOM

Rechtliche Einordnung



➔ Grundsätzlich besteht das Recht auf korrekte Schreibweise des eigenen Namens, jedoch handelt es sich dabei um (noch) *nicht-angewandtes* Recht

Entscheidung 2014/04 des IT-Planungsrats

1. Unter Bezug auf § 1 Abs. 1 Satz 1 Nr. 2 des Vertrages über die Errichtung des IT-Planungsrats und über die Grundlagen der Zusammenarbeit beim Einsatz der Informationstechnologie in den Verwaltungen von Bund und Ländern (IT-Staatsvertrag) beschließt der IT-Planungsrat die **verbindliche Anwendung des Interoperabilitätsstandards „Lateinische Zeichen in UNICODE“** als Mindeststandard.
2. Für IT-Verfahren, die dem **bund-länderübergreifenden Datenaustausch oder dem Datenaustausch mit Bürgern und Wirtschaft dienen**, werden folgende Fristen für die Konformität laut Anlage 1 festgelegt:
 - mit Beschlussfassung - für IT-Verfahren, die neu aufgebaut oder in wesentlichem Umfang überarbeitet werden,
 - **drei Jahre nach Beschlussfassung** - für andere IT-Verfahren.
3. Die Mitglieder des IT-Planungsrats tragen in ihrer jeweiligen Gebietskörperschaft dafür Sorge, dass, sobald möglich, sämtliche IT-Verfahren konform zu diesem Standard sind, wenn nicht zwingende fachliche oder wirtschaftliche Gründe dagegen sprechen.
4. Der Standard „Lateinische Zeichen in UNICODE“ wird im Auftrag des IT-Planungsrats von der Koordinierungsstelle für IT-Standards (KoSIT) herausgegeben. Der Standard ist im Bundesarchiv, Potsdamer Straße 1, 56075 Koblenz, für jedermann zugänglich und archivmäßig gesichert niedergelegt.
5. Der Standard und darauffolgende Änderungen werden im Bundesanzeiger bekannt gemacht.

IT-Planungsrat, Entscheidung 2014/04 – Ein einheitlicher Zeichensatz für Datenübermittlung und Registerführung, abrufbar unter:

http://www.it-lanungsErat.de/DE/Entscheidungen/2014/13_Sitzung/13_Sitzung_Entscheidungen.html;jsessionid=3D8F9CB4FE83707578652534A1713A36.2_cid292?nn=5031962#doc5031690bodyText4

Unicode@GSE

25.09.2018

Seite 6

© DATEV eG, alle
Rechte vorbehalten

Fakten zu String.Latin 1.1: Status Quo^{*}

- Verbindlich seit 2011 durch Beschluss des IT-Planungsrats (IT-PLR) für die Öffentliche Verwaltung
- Umstellungsfrist endete 2017
- Viele offene Fragen:
 - Sichere Identifizierung / Transliteration / Phonetik / Altdaten
 - Rechtsgrundlagen / Europäische Vorgaben
 - Technische Umsetzung / Schriftfont / Eingabe
 - Neue Version: Was? / Wann? / Wie? / Übergangsregelungen?
- 2016 Gründung der Fachgruppe String.Latin – mittlerweile mit Mandat des IT-PLR
- FG String.Latin besteht aus Teilnehmern der Ministerien und Behörden, Sozialversicherungsträgern und unabhängigen Firmen

Historie

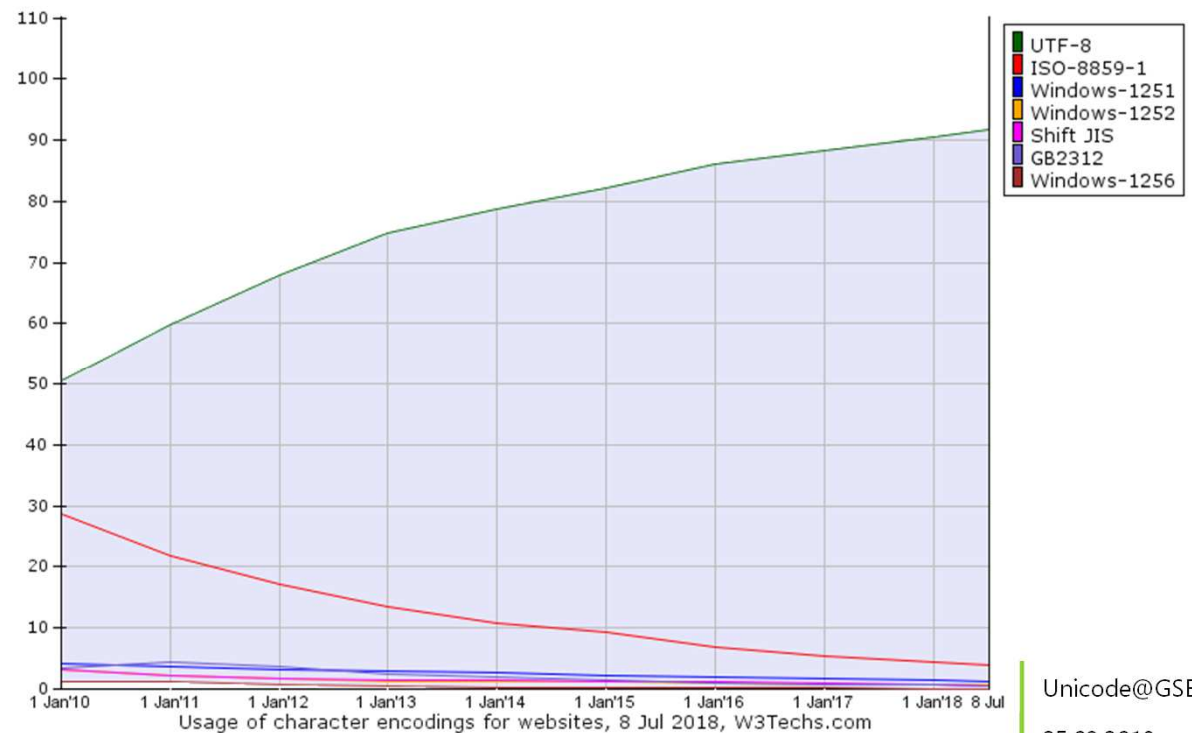


- Erster Entwurf 1988 von Joseph D. Becker

- ~7.000 Zeichen

- 2018: **Unicode 11.0**

- ~137.000 Zeichen in 146 Schriftsystemen (Scripts)
 - Platz für bis zu 1,1 Mio. Zeichen
 - Momentan 91,8% der Websites in UTF-8



Unicode@GSE

25.09.2018

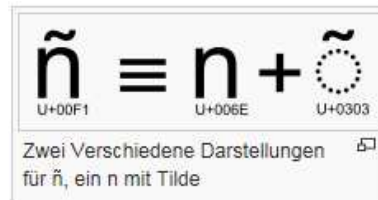
Seite 8

© DATEV eG, alle
Rechte vorbehalten

KoSIT Standard – Lateinische Zeichen in Unicode

■ Lateinische Zeichen in UNICODE V1.1 (V1.2) umfasst 490 (738) Zeichen.

- 411 lateinische Buchstaben
- 79 Sonderzeichen, vor allem aus ASCII, ISO-Latin-1, ISO-Latin-15
- 481 (595) davon werden in UTF-16 immer genau in 2 Bytes kodiert.
- UTF-8 hat dafür eine variable Speicherlänge von 1-3 Bytes.
- 9 (143) davon sind „Composite Characters“. Diese bestehen aus 2 bis 3 kombinierten Zeichen und benötigen somit in UTF-16 2 bzw. 3 x 2 Bytes = 4-6 Bytes.
(analog für UTF-8, UTF-32)



- Die Stringlänge ist in diesem Fall 2/3. (auch wenn nur „ein“ Zeichen gedruckt werden würde).

0044+0302	Đ
004A+030C	Ž
004C+0302	Ł
004D+0302	Ŧ
004E+0302	Š
0064+0302	đ
006C+0302	ł
006D+0302	ț
006E+0302	š

KoSIT Standard

- Aktuell veröffentlicht: KoSIT-Standard 1.1
- In Planung: KoSIT-Standard 1.2 (in Planung: Ende 2018 als DIN)

KoSIT 1.1	KoSIT 1.2
490 Zeichen	738 Zeichen
2 Steuerzeichen	3 Steuerzeichen
9 zweifach komponierte Zeichen	143 zweifach komponierte Zeichen
	11 dreifach komponierte Zeichen

- Auch in KoSIT-Standard 1.2 alle Codepoints in BMP* -> mit UCS-2 darstellbar

* BMP: Basic Multilingual Plane (alle mit UTF16 in 2 Byte darstellbaren Zeichen , etwa 65 000 Zeichen)

Festlegung Zeichensatz bei DATEV



Ausgangslage

- Eingabe von Zeichen außerhalb des KoSIT-Standards soll möglich sein (z. B. Bestückung von Daten über externe Schnittstellen, Partnering)
- Bereits heute gibt es DATEV-Anwendungen, die Unicode im größeren Umfang unterstützen (Stammdatendienst)
- Durchgängige DATEV-weite BMP-Verarbeitung in absehbarer Zeit nicht realistisch → Transliterationen auf andere Single-Byte Character Sets wie DATEV EBCDIC notwendig
- Korrekte und vollständige Transliteration von BMP/"UCS-2" (ca. 65.000 Zeichen) auf Single-Byte Character Sets nur schwer erstellbar

DatevSet

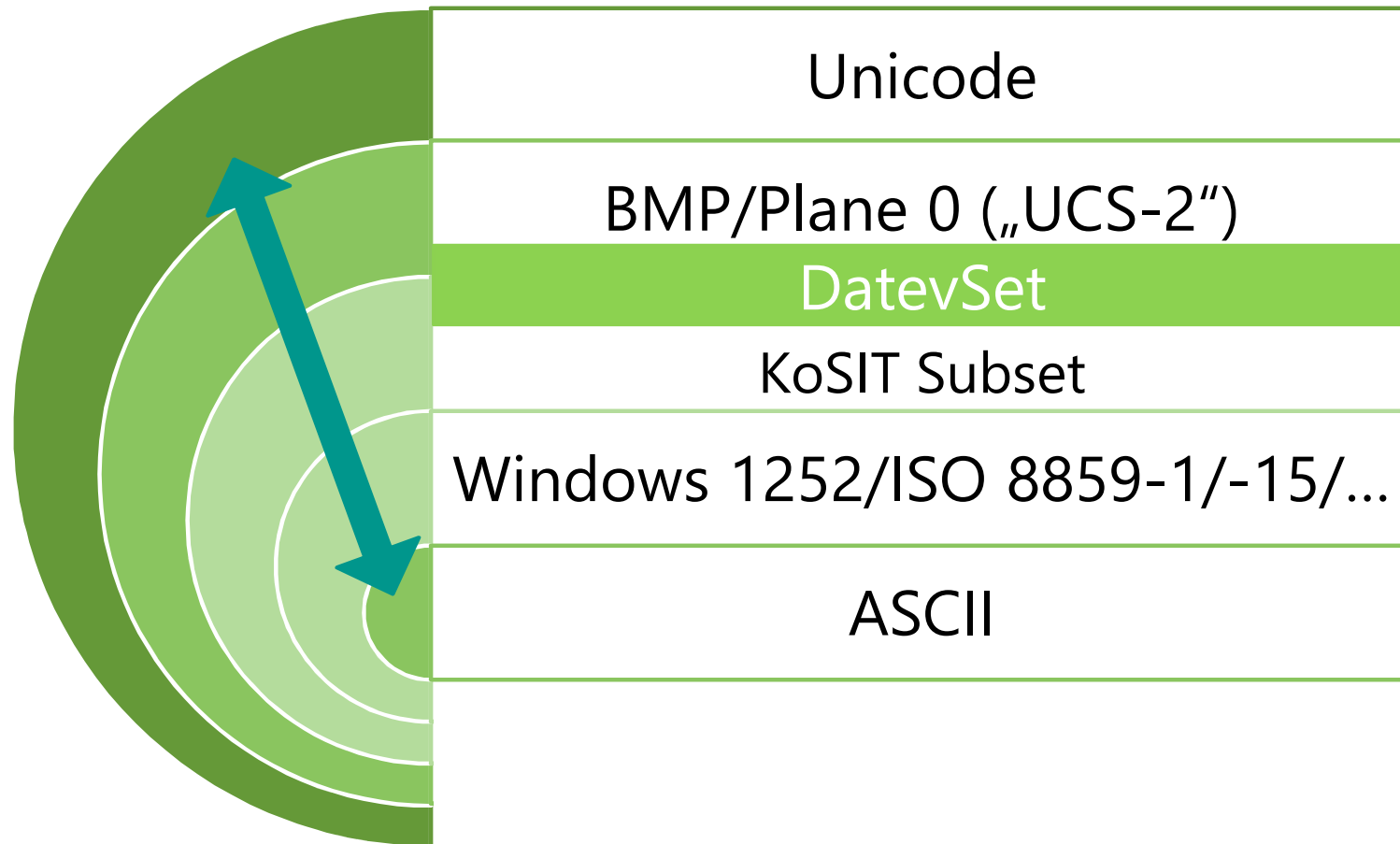
Festlegung

- **Jede Anwendung und jeder Service unterstützt als Zeichensatz grundsätzlich UCS-2, sofern keine gewichtigen Argumente dagegen sprechen, ansonsten aber mindestens DatevSet.**

DatevSet

- interner erweiterbarer Zeichensatz-Standard
- Abstraktionsebene für Konvertierungsalgorithmen
- entspricht aktuell dem KoSIT-Standard String.Latin 1.1 (490 Zeichen) + <SUB>
- Erweiterung bei Verabschiedung eines neuen KoSIT-Standards
- Whitepaper für Datev

Datev-Set

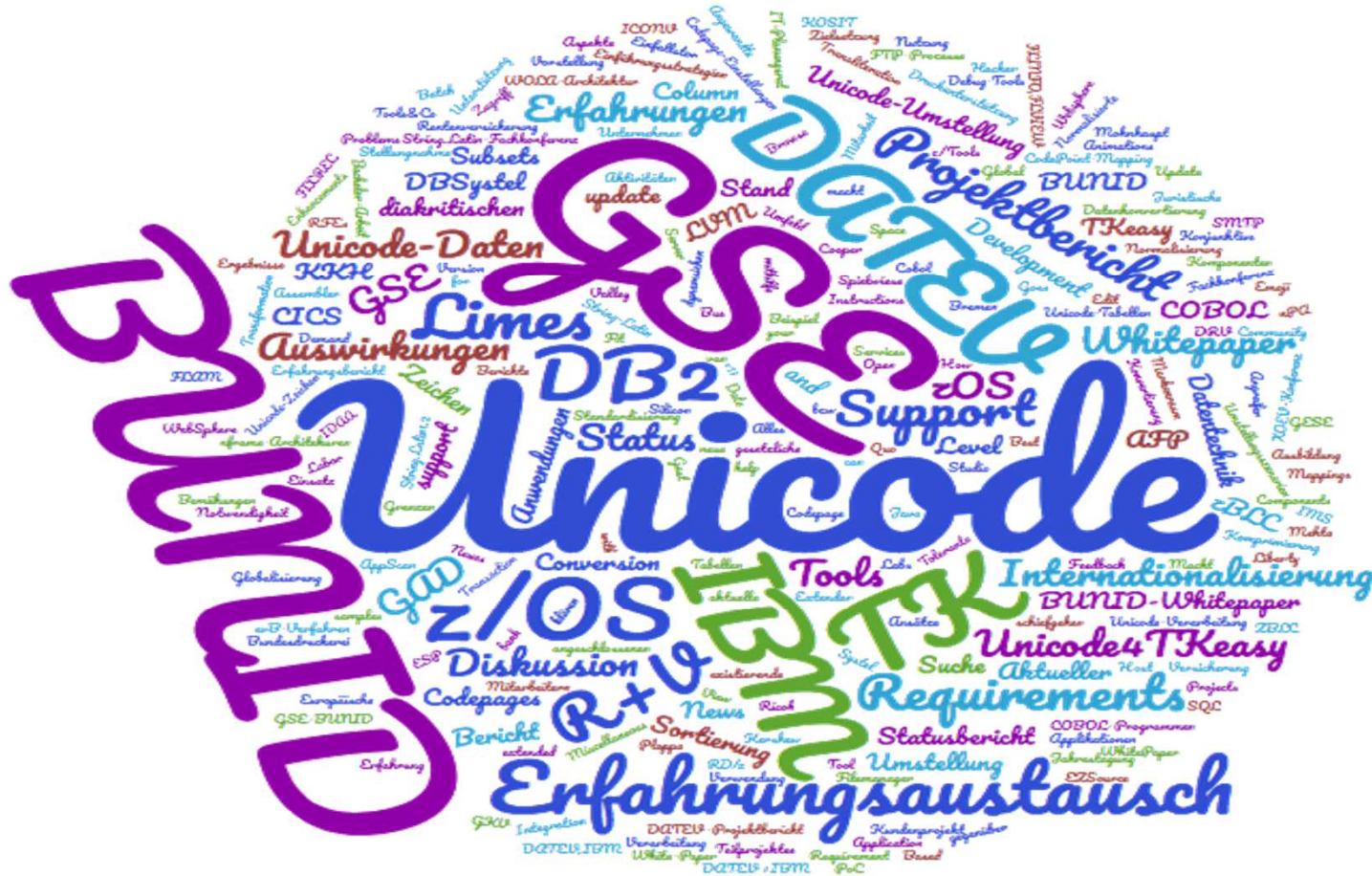


GSE BUNID

„Ist das nicht interessant? Für ein Symbol reicht ein Buchstabe,
aber ein Wort muss mindestens zwei Buchstaben haben“

Wolfgang J. Reus, deutscher Journalist

BUNID – Rückblick auf 6,5 Jahre



Unicode@GSE

25.09.2018

Seite 15

© DATEV eG, alle
Rechte vorbehalten

BUNID – Grundlagen

- Herbst 2011
 - -
- Frühjahr 2012
 - Unicode im DB2 Umfeld
 - Requirement: Europäische Sortierung
 - Unicode Assembler Instructions
 - Unicode - Ansätze zur Standardisierung von Subsets
- Herbst 2012
 - Auswirkungen von Unicode und Codepage-Einstellungen auf COBOL und DB2
 - Komprimierung und Konvertierung von Unicode-Daten Limes Datentechnik
 - Angewandte z/OS Unicode Services
 - Unicode - Auswirkungen auf Sortierung
 - AFP - Unicode Druckunterstützung

BUNID – Grundlagen

■ Frühjahr 2013

- Globalisierung bzw. Internationalisierung von Anwendungen
- z/OS Unicode Samples - Einsatz in Batch Anwendungen
- Normalisierte Suche

■ Herbst 2013

- Host on Demand - Status zu "Unicode" vs. "extended Codepages"
- zOS Unicode Miscellaneous
- Unicode with IMS
- Unicode-Verarbeitung mit IBM Integration Bus & WebSphere Transformation Extender
- Tolerante Suche gegenüber diakritischen Zeichen mithilfe von Normalisierung unter DB2 z/OS

BUNID – Grundlagen

- Frühjahr 2014
 - Ergebnisse der Bachelor-Arbeit
- Herbst 2014
 - Juristische Aspekte zu Unicode
 - Unicode Subsets & Best Fit Mappings
 - Vorstellung Whitepaper
 - AppScan - Applikationen als Einfallstor für Hacker
- Sommer 2015
 - Whitepaper
 - WOLA- Architektur
 - Einführungsstrategien

BUNID – Grundlagen

- Herbst 2015
 - Liberty for your Codepage Problems - How Java on zOS can help
 - Unicode als Spielwiese für Angreifer
 - Whitepaper
- Frühjahr 2016
 - Unicode - News von der IBM
- Herbst 2016
 - Unicode News
 - BUNID Whitepaper

BUNID – Grundlagen

- Frühjahr 2017
 - Conversion & Transliteration
 - Conversion in Mainframe-Architekturen
 - Unicode Animations
 - BUNID-Whitepaper
- Herbst 2017
 - CodePoint-Mapping

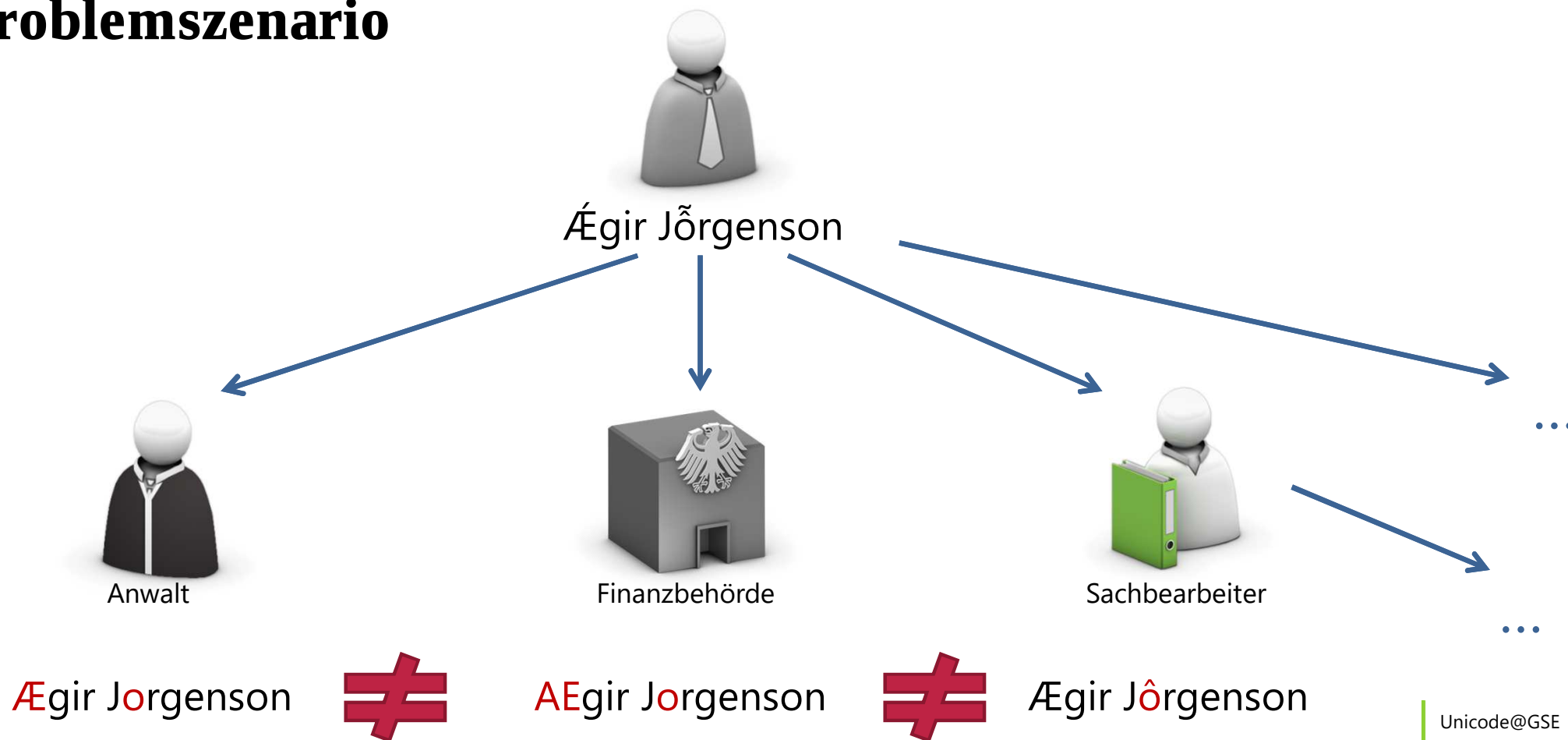
BUNID – Strategische Ausrichtung von IBM

- z/OS Components and Unicode Support
Kershew Mehta, IBM z/OS Development
- Global View on Unicode support, Projects and Requirements
Chris Crone, IBM Silicon Valley Labs
- IBM zOS 2.1 Unicode Support
Bangert, IBM
- GSE Requirements + IBM Stellungnahme
- Unicode Enhancements CICS Transaction Server
IBM CICS Labor, Cooper
- GSE Requirements
Bangert, Kistenberger

Problemszenario

UNICODE

EBCDIC 1141



Abweichungen Transliterationsregeln KoSIT

Codepoint	Zeichen	LVM-Mapping (EBCDIC1141)	DATEV-Mapping (EBCDIC 1141)	KoSIT-Vorschlag
0150	Ö	Ö	O	O
0151	ö	ö	o	o
0170	Ü	Ü	U	U
0171	ü	ü	u	u
01DE	Ä	Ä	A	A
01DF	ä	ä	a	a
01FA	Å	Å	A	A
01FB	å	å	a	a
01FC	Æ	Æ	Æ	AE
01FD	æ	æ	æ	ae
01FE	Ø	Ø	o	o
01FF	ø	ø	o	o
022A	Ő	Ő	O	O
022B	ő	ő	o	o
1EAA	À	À	A	A
1EAB	à	à	a	a
1EAC	Â	Â	A	A
1EC4	Ê	Ê	E	E
1EC5	ê	ê	e	e
1ED6	Ô	Ô	O	O
1ED7	ô	ô	o	o
1EA4	Ä	Ä	A	A
1EA5	ä	ä	a	a
1EA6	Å	Å	A	A
1EA7	å	å	a	a
1EC0	Ê	Ê	E	E
1EC1	ê	ê	e	e
1EBE	Ê	Ê	E	E
1EBF	ê	ê	e	e
1EC6	Ê	Ê	E	E
1EC7	ê	ê	e	e
1ED2	Ő	Ő	O	O
1ED3	ő	ő	o	o
1ED0	Ő	Ő	O	O
1ED1	ő	ő	o	o
1ED8	Ő	Ő	O	O
1ED9	ő	ő	o	o

Codepoint	Zeichen	DATEV-Mapping (DOS CP 437)	KoSIT-Vorschlag
00BC	¼	1/4	n/a
00BD	½	1/2	n/a
00BE	¾	3/4	n/a
00D7	×	*	n/a
20AC	€	EUR	n/a

→ **Gemeinsame Regeln notwendig für konsistenten Datenaustausch!**

KoSIT-Vorschlag aus XML-File http://xoev.de/latinchars/1_1/charset/latinchars.xml

Datev Mappings

- „Unicode“ (BMP) zu **DatevSet** (reine Substitution)
- **DatevSet** zu...
 - EBCDIC 1141
 - DATEV EBCDIC
 - ISO 8859-1 (Latin1)
 - ISO 8859-15 (Latin9)
 - Win1252
 - *CP437, CP500, CP850 (Mapping vorhanden, jedoch aktuell nicht implementiert)*

Unterschiede

- Datev-Abweichung zum KoSIT-Vorschlag bei 15 Zeichen
- LVM-Abweichung zum KoSIT-Vorschlag bei 37 Zeichen
- Fehlende Mapping-Vorschläge im KoSIT-Standard bisher nicht von Hr. Kappenberg spezifiziert (evtl. wird das durch die DINspec verbessert)
- Wie mappen wir? Auf das Grundzeichen oder einem „ähnlichen“ Zeichen?
- Kann man sich auf nationaler Ebene auf Mapping-Tabelle einigen?

Infos zu FG String.Latin

„Unicode enabled us to give the local sense to every one of our users.“

Shai Agassi, Member of SAP Executive Board

Warum String.Latin 1.2?

- Ergänzungen von Standesbeamten
- Alle ISO-Transliterationsregeln abdecken
- Ergänzungen für Namen juristischer Personen (Handelsregistereinträge)
- Tauglichkeit für europäischen Datenaustausch

Warum als DIN SPEC?

- Hemmnisse bei der Umsetzung
 - Finanzierung
 - Verbindliche Festlegung in Ausschreibungen
 - Bekanntheit
- Warten auf europäische Vorgaben (z. B. durch EESSI – Electronic Exchange of Social Security Information)
- Ziele:
 - In Deutsch verfasst, aber zusätzlich englischsprachige Version für EU-Projekte
 - Anschließende Normierung in Richtung ISO- oder CEN-Norm

Warum als DIN SPEC? - Abwägung

- Gegenwind aus dem DIN-Normenausschuss „Transliteration und Transkription“
 - Forderung: Immer alle lateinischen Zeichen aus Unicode unterstützen
 - Risiko, dass Unicode nie vollständig implementiert wird, wenn Länder Unicode einschränken
 - Betroffene Community soll Anwendungsvereinbarungen ohne DIN SPEC / Norm treffen

Warum als DIN SPEC? - Abwägung

- Argumente pro DIN SPEC:
 - Die „betroffene Community“ – die Öffentliche Verwaltung Deutschlands – ist sehr groß
 - Keine Widersprüche zu anderen Normen und konkreter Bedarf vorhanden
 - Auswahl der Zeichen muss fachlich motiviert sein
 - Keine historischen Zeichen unterstützen
 - Risiko zusätzlicher Zeichen: Verschleierung von Namen durch ähnliche Zeichen
 - Für Transliterationen notwendige kombinierte Sequenzen wären nicht explizit aufgeführt
 - Reduzierung erleichtert die Eingabe
 - Abschließende Liste erleichtert Implementation und Test für IT und Druck (hoheitliche Dokumente und Urkunden)
 - Erleichtert Begründung für Investitionen
 - Erleichtert verbindliche Vorgabe in Ausschreibungen
 - Beugt abweichenden europäischen Vorgaben vor

Was ist eine DIN SPEC? – Was ist das DIN?

- Deutsche Institut für Normung e. V. ist eingetragener gemeinnütziger Verein
- Wird privatwirtschaftlich getragen
- Beteiligung aller interessierten Kreise
- Unterstützung des freien Warenverkehrs durch aktive Mitwirkung an der internationalen und europäischen Normung
- Nationale Übernahme internationaler Normen
- Einheitlichkeit und Widerspruchsfreiheit des Normenwerks
- Vermeidung von Doppelarbeit
- DIN trägt aktiv zur Konsensbildung bei

Was ist eine DIN SPEC? – Steuerung durch DIN

- DIN stellt Infrastruktur für effiziente Arbeit
- Runder Tisch führt alle interessierten Kreise zusammen, z. B. Wirtschaft, Forschung, Verbraucher, öffentliche Hand
- Unparteilich
- Organisiert die Arbeit in Normenausschüssen
- Stellt Stand der Technik sicher
- Kartellrechtlich unbedenklich und demokratisch legitimiert
- Übersichtliche und praxisorientierte Regelwerke – dabei präzise und eindeutig

Was ist eine DIN SPEC? – Vorteile

- Schnell: Erstellung und Veröffentlichung innerhalb weniger Monate möglich
- Flexibel: Arbeitsprogramm, Meetingfrequenz u. Ä. bestimmt der Workshop
- Anerkannt: Marke DIN sorgt für zusätzliche Akzeptanz
- Imagefördernd: Autoren (Person und Organisation) werden im Dokument genannt
- Unkompliziert: Planung und Management des Projekts erfolgen durch DIN
- Widerspruchsfrei: Spätere Überführung in Normen möglich

Eingabe/Fonts/Druck

„Lieber ungefähr richtig als genau falsch.“

Franz Josef Strauß, deutscher Politiker

Unicode Fonts im Druck

- Kaum eine der momentan verwendeten Schriften unicodefähig
- Daher: **Neue Fonts im Druck**
- Ausnahme: PDF-Druck 😊 (Font eingebettet)
- UnicodeDoc nur beschränkt einsetzbar
- **Priorität:** möglichst wenig Abweichung auf technischer Ebene:
 - Gleiche Laufweite, Höhe, Abstände,...

Unicode Range	KoSIT-Standard 483 Glyphs	Migration-Fonts (Courier, Times, Helvetica) 1267 Glyphs	Compatil 388 Glyphs	SegoeUI keine HOST Lizenz 2901 Glyphs	Verdana, Calibri, Consolas 652 Glyphs (WGL)	DejaVu Ca. 2700 - 6241 Glyphs
C0 Controls (0000-001F)	0009-000A 000D 3 Glyphs	✓	✓	✓	✓	✓
ASCII (0020-007F)	0020-007E 95 Glyphs	✓	✓	✓	✓	✓
Latin-1 Supplement (00A0-00FF)	00A1-00AC 00AE-00FF 94 Glyphs	✓	✓	✓	✓	✓
Latin Extended-A (0100-017F)	0100-0107 010A-011B 011E-0123 0126-0131 0134-015B 015E-016B 016E-017E 115 Glyphs	✓	5 F ☹	✓	✓	✓
Latin Extended-B (0180-024F)	018F 01A0-01A1	40 F ☹	47 F ☹	✓	45 F ☹	✓

- Alle Zeichen in NFC
- Kombinierte Zeichen: „Umdrehen“ der Codepoints (Z + ◌ → ◌ + Z)
→ Verzicht auf explizites Platzieren der Diakritika

Fonts

- *DATEV Compatil* nicht Unicode-fähig
 - Erweiterung als ursprüngliche Planung verworfen
 - SIL OFL Fonts Liberation und *Google Noto* als Ersatz/Fallback-Font
- Weitere Schriften wie *Gothic, Helvetica, Verdana* sollen durch *Liberation/Deja Vu* bei Bedarf ersetzt werden



Liberation

Google Noto Serif | 32pt

Laufweite

Compatil DATEV | 32pt

Google Noto Serif | 29pt

Laufweite

Compatil DATEV | 32pt

Eingabe von Unicode-Sonderzeichen

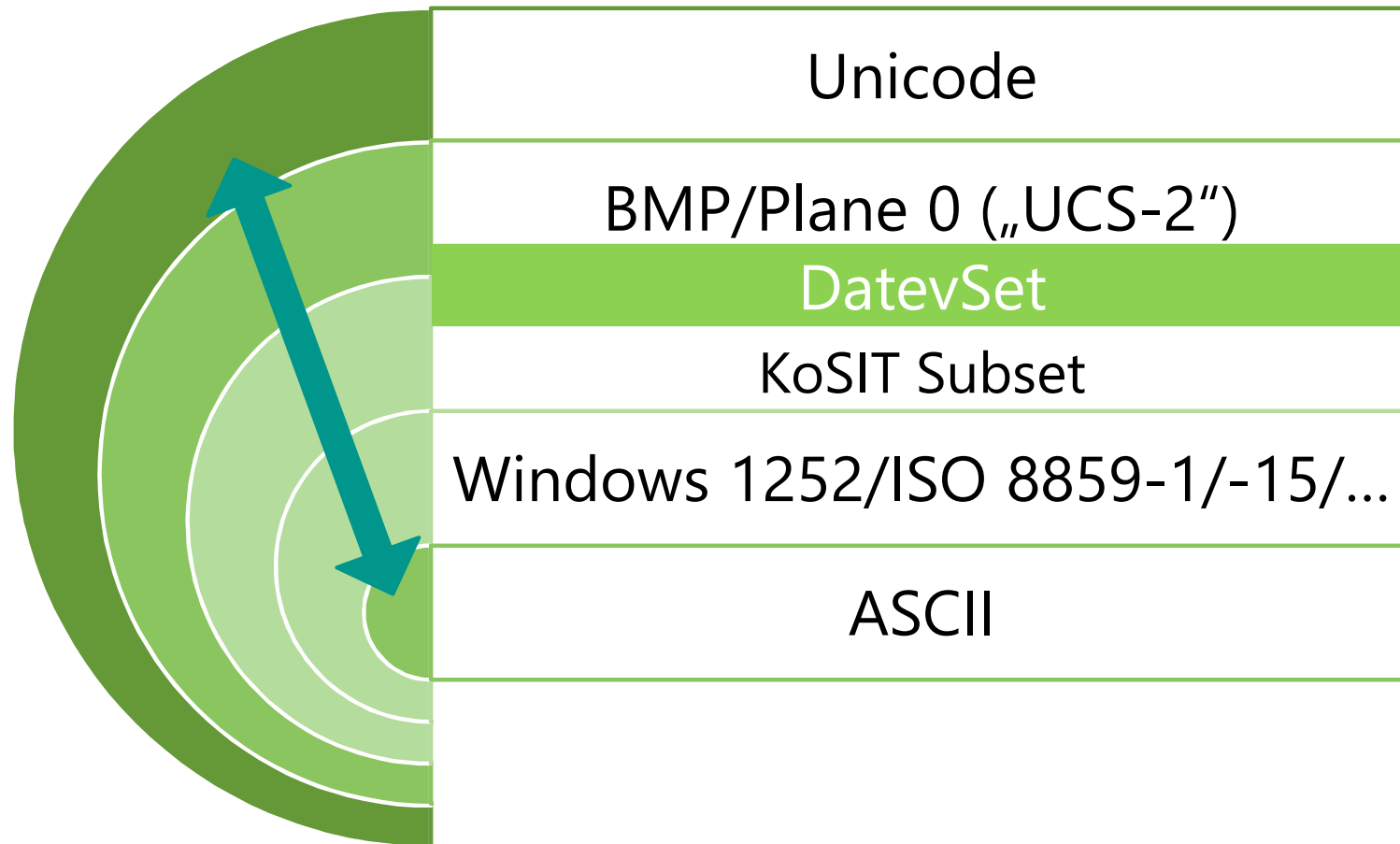
- Online-Umfrage bei Datev mit ca. 650 Befragten
 - Sonderzeichen werden aktuell via Tastenkombinationen (63%) und Zeichentabelle (54%) eingegeben. Rund 30% der Probanden geben Sonderzeichen durch copy/paste z.B. aus dem Internet ein.
 - Als bevorzugte Variante zur Standard-Eingabe von Sonderzeichen in DATEV-Anwendungen nannten die Nutzer die Eingabe via Tastenkombination (50%) oder Zeichentabelle analog zu Microsoft-Programmen (48%). Außerdem wurde die Möglichkeit der Eingabe über ein Kontextmenü (30%) oder eine Virtuelle Tastatur (25%) genannt.

Unicode@Datev

„Entscheidend ist, was hinten rauskommt.“

Helmut Kohl, deutscher Politiker

Datev-Set



Konvertierungstools

- Für den klassischen Host (COBOL/ASM):
- Für Java:
- Für onPremise (C#):

GMUNICO



seCPconv

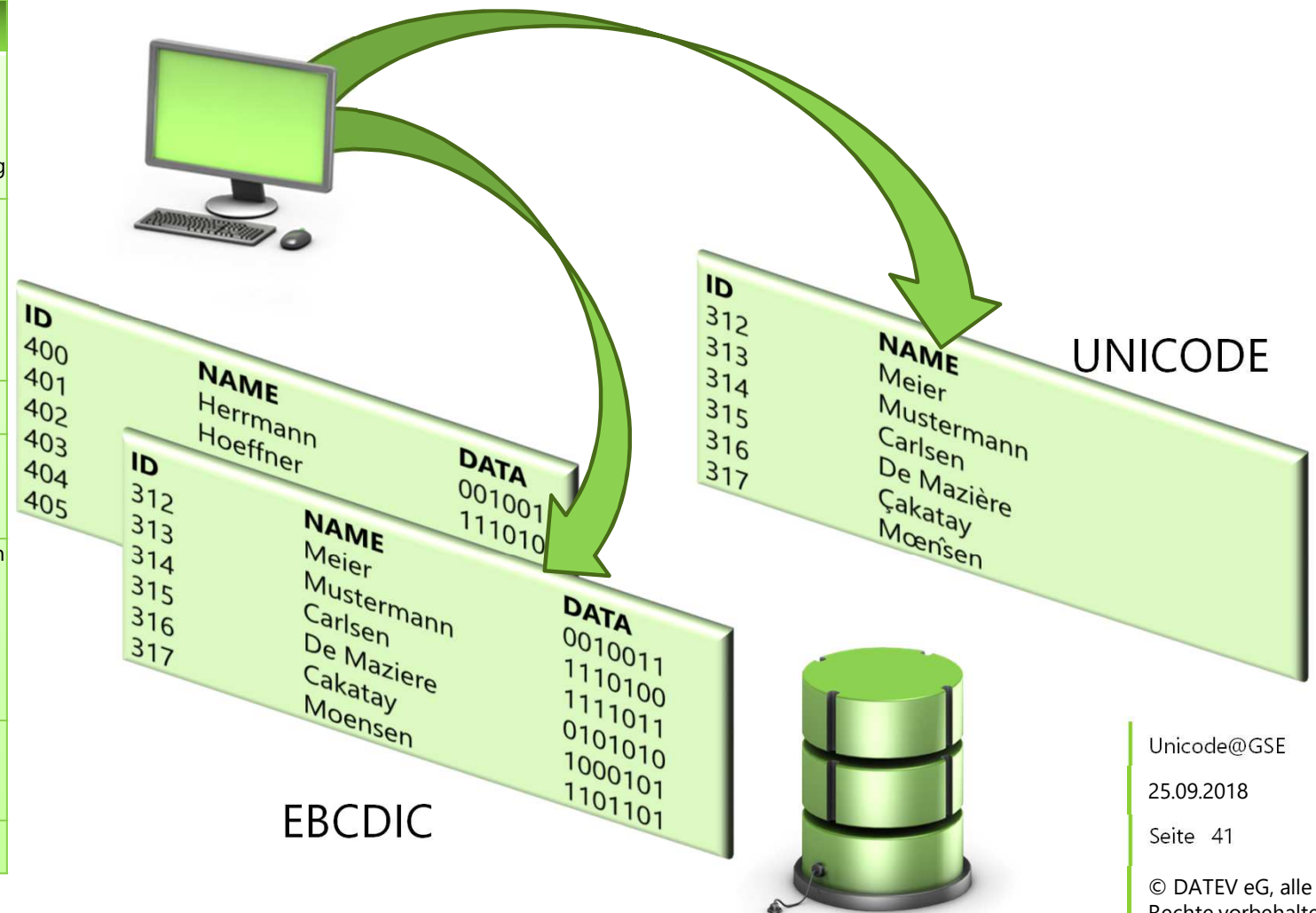


UNICTRAN



EBCDIC + UNICODE teilweise gespiegelt

EBCDIC + Unicode teilweise gespiegelt	
mögliche Ausgangsszenarien (Stand VOR der Umstellung)	- viele Schnittstellen mit EBCDIC-verwendenden Programmen - wenig Spalten mit Diakritika - bereits viel Speicherplatz notwendig
Aufbau	- Unicode-Tabellen entstehen bei Bedarf - Unicode-Spalten entstehen bei Bedarf
Funktionsweise	Anwendung holt Daten aus EBCDIC und Unicode-Tabelle
Spalten	- Primärspalten in beiden Spalten identisch - Problematisch bei Char-Primärspalten
RI	- Keine RI-Verknüpfung zwischen den Tabellen - Konsistenz und Konvertierung durch Anwendung (2 Insert, 2 Update) - ggf. DB2-Trigger
Umstellungsprozess	- Speichererweiterung nach Bedarf - Sukzessive Anpassung der Programmlogik
EBCDIC Auswirkung	EBCDIC-Welt bleibt erhalten



Unicode@GSE

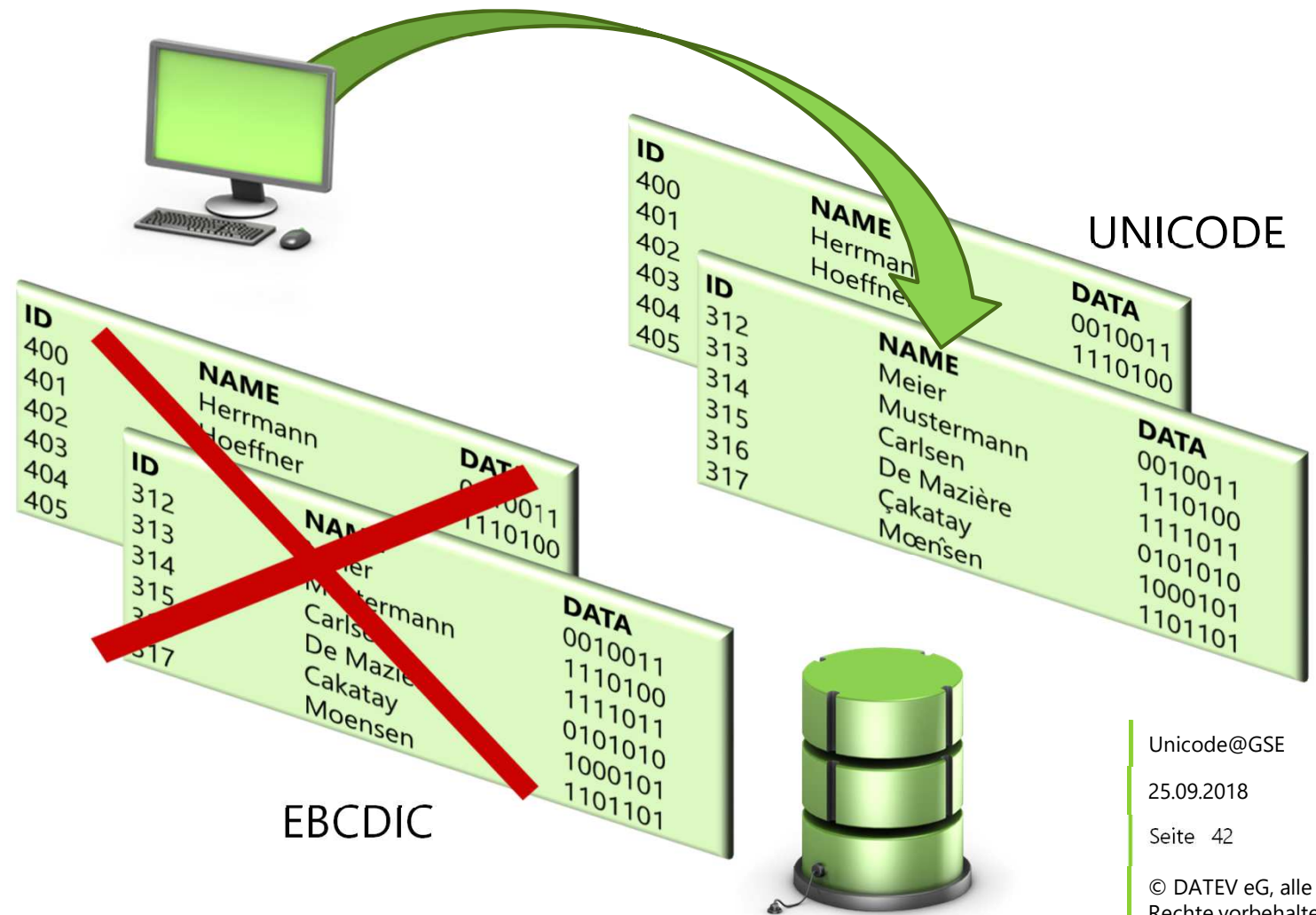
25.09.2018

Seite 41

© DATEV eG, alle Rechte vorbehalten

Stichtagsumstellung UNICODE

Anwendungsbezogene Stichtagsumstellung	
mögliche Ausgangsszenarien (Stand VOR der Umstellung)	- kleine, überschaubare Programme - exklusiver Datenbankzugang - wenig Schnittstellen
Aufbau	- Nur eine Unicode-Tabelle - Nur Unicode-Spalten
Funktionsweise	Anwendung holt Daten ab Stichtag aus Unicode-Tabelle
Spalten	Nur eine Primärspalte in Unicode
RI	- RI-Verknüpfung durchgängig - Konsistenz gewährleistet
Umstellungsprozess	- Speichererweiterung nach Bedarf - Stichtagsumstellung
EBCDIC Auswirkung	EBCDIC-Welt für diese Anwendung abgeschaltet



Unicode@GSE

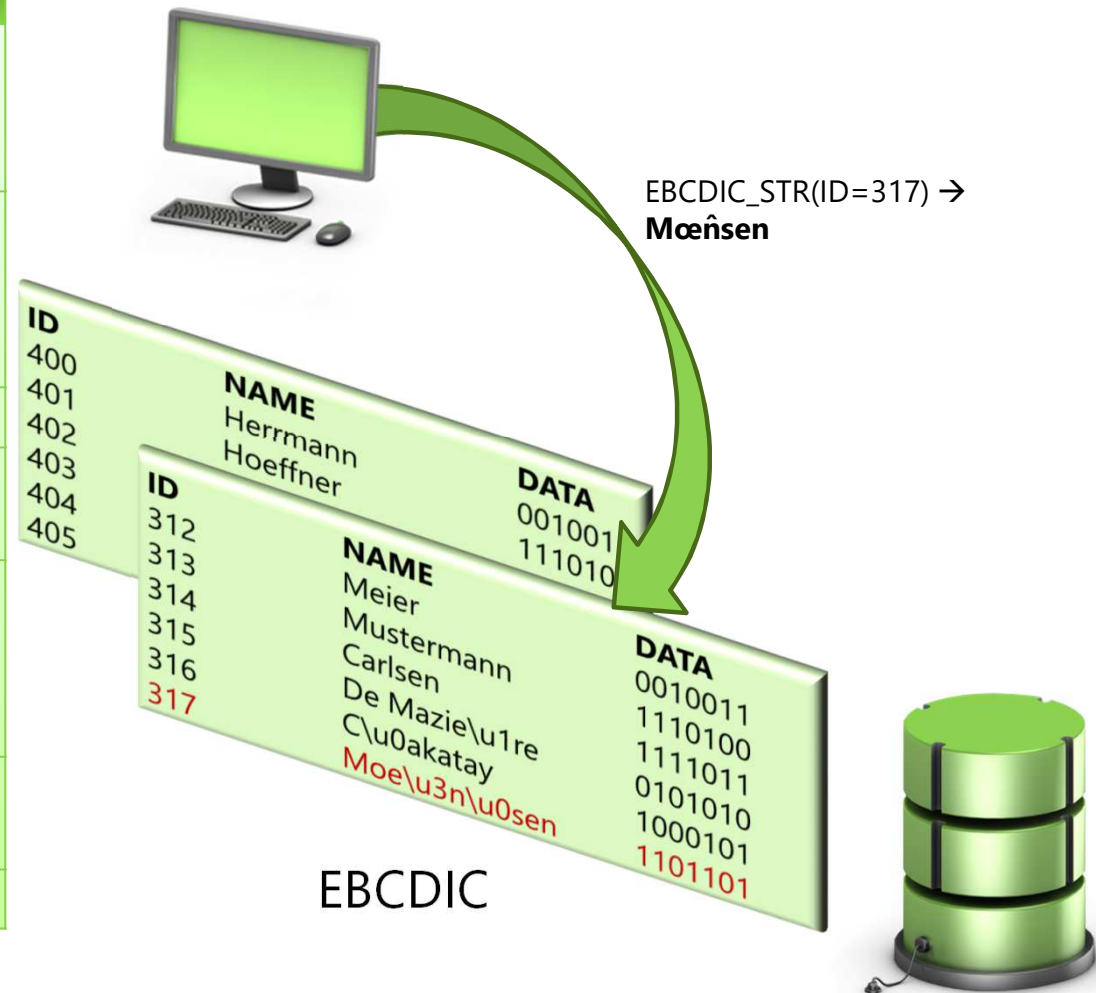
25.09.2018

Seite 42

© DATEV eG, alle Rechte vorbehalten

Pseudo-Unicode in bestehender EBCDIC-Welt

Pseudo-Unicode in bestehender EBCDIC-Tabelle	
mögliche Ausgangsszenarien (Stand VOR der Umstellung)	- nur interne Schnittstellen - keine kanonische Äquivalenz von Nöten
Aufbau	- Nur EBCDIC-Tabellen (273) - Unicode-Daten in EBCDIC-Spalte mit speziellen DB2-Funktionen: - zum Lesen EBCDIC_STR, zum Schreiben UNICODE_STR
Funktionsweise	Nicht-Verwendung der o.g. Funktionen zerstört Daten!
Spalten	Nur eine Primärspalte in EBCDIC
RI	- RI Verknüpfung durchgängig - Konsistenz gewährleistet
Umstellungsprozess	- Speichererweiterung nach Bedarf - sukzessive Anpassung der Programmlogik
EBCDIC Auswirkung	EBCDIC-Welt bleibt erhalten



Umwandlung an externer Schnittstelle (Best Fit Mapping)

