

第三回 調査技術ゼミ

2005年10月21日

『 文字コードとフォレンジック』 文字の検索

ネットエージェント株式会社

フォレンジック調査と文字

- フォレンジック調査では、フィルタによる絞込みと、検索により調査時間の短縮が行われる
- 文字コードの影響により、フィルタや検索が失敗する(発見できない)危険性が存在する
- 文字の歴史は長く、様々な文字コードが存在している
- 調査員は文字コードを理解しているか？
ツールの結果を信用してよいのか？

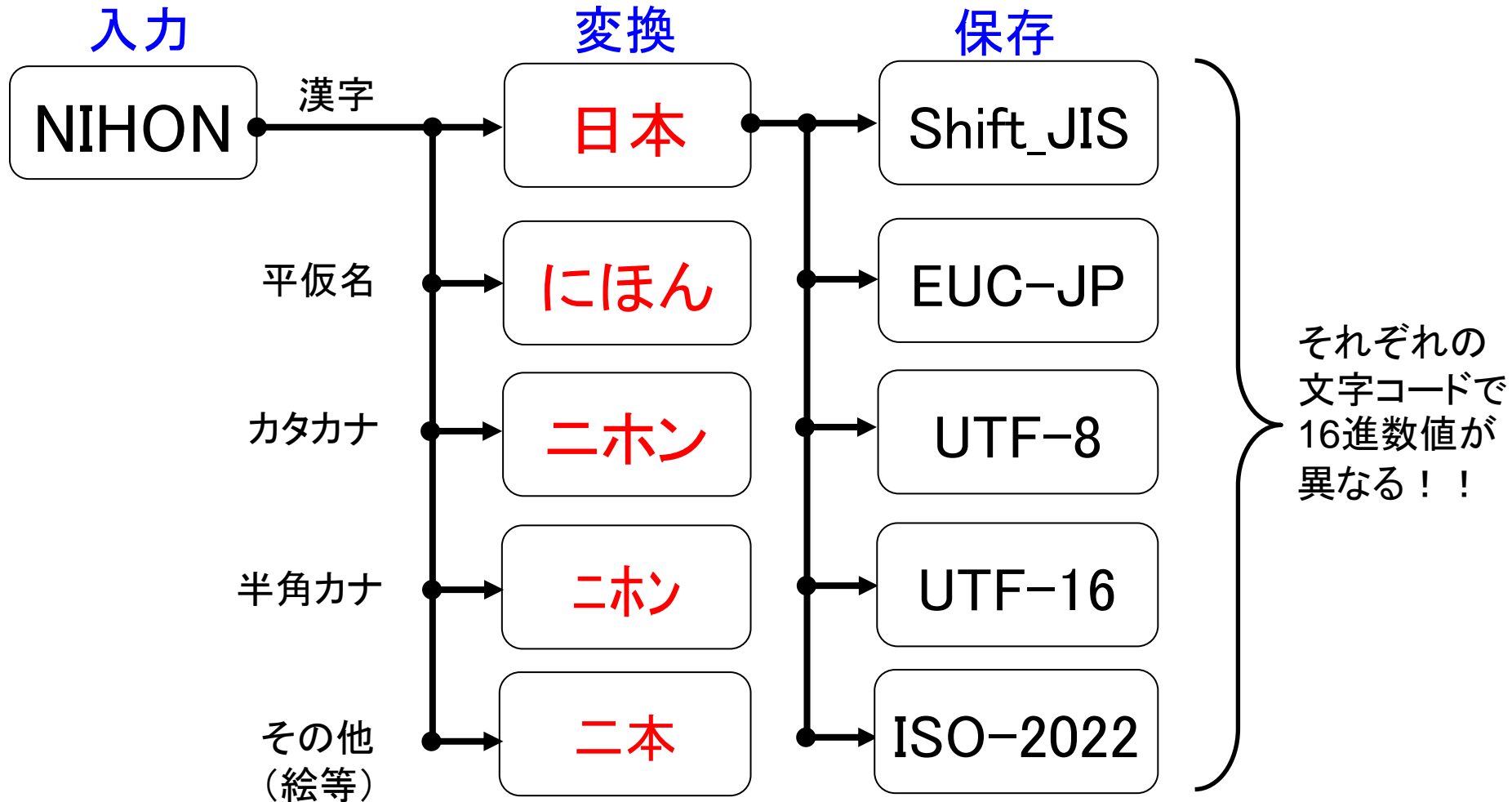
基本的な調査方法

- 検索
（16進検索）
- 文字列抽出
（インデックス化）

目視確認



文字列と文字コードの関係



16進形式での検索

- 文字列を16進数へ変換
- 16進数(パターン)で検索する
例) $A = 0x41 = 0100\ 0001$
- 利点
読めない文字でもバイト列が分かれば検索することができる(外国語の検索など)
- 問題点
パターンが1bitでも異なると検出できない

インデックス検索

- ・ 文字列を抽出しインデックス化することで高速な検索が可能
- ・ 先頭文字の入力によるジャンプ機能
- ・ 対応製品
FTK Asia, Paraben Text Searcher
- ・ 日本語文字列抽出ツール
istrings, jstrings

FTK Asia

- 日本語表示・検索に対応
 - ・ライブ検索
 - ・インデックス検索
- キーワードのエクスポート機能あり
- ファイルのヘッダーによるカテゴリライズ化
- 詳細については
<http://www.ubic.co.jp/>

istrings

- 日本語および Unicode 対応の strings
- istringsで抽出した日本語文字列に対して、別のツールであいまい検索が可能に

例) CP932とUTF-16LEの抽出結果を wiconv でUTF-16LEへ変換し結合

```
istrings -i CP932 -f -p -c Test.E01 | wiconv -f 932 -t 1200 >> log.txt  
istrings -i UTF-16LE -f -p -c Test.E01 >> log.txt
```

完成した log.txt を Word で UTF-16LE として読み込み、あいまい検索を行う

CP932、CP949

- ・ パスの区切り文字が変換される
- ・ 外部ツールの呼び出し時、パストラバーサルが発生する危険性がある
- ・ 利用するツールでトラバーサルが発生しないかを検証しておく必要がある

\

 information from
FileFormat.Info

U+005C

¥

 information from
FileFormat.Info

U+00A5

≠

 information from
FileFormat.Info

U+20A9

<http://d.hatena.ne.jp/hasegawayosuke/20051108#1131410804>

<http://d.hatena.ne.jp/hasegawayosuke/20050823#1124761789>

<http://www.microsoft.com/globaldev/drintl/columns/019/default.aspx#EED>

ISO 2022系

- Windows 環境のコードページ
 - 50220 ISO 2022 Japanese with no halfwidth Katakana
 - 50221 ISO 2022 Japanese with halfwidth Katakana
 - 50222 ISO 2022 Japanese JIS X 0201-1989
- ISO 2022 を16進数で検索する場合、指定した文字列によってはエスケープシーケンスの影響を受ける
- EFE 5.04a ではヒットしない
例) 伊B原
- エスケープシーケンスを検索する

エスケープシーケンス(1)

■RFC1468 ISO-2022-JP

ASCII	ESC (B	¥x1b¥x28¥x42
JIS X 0201左面(JISローマ字)	ESC (J	¥x1b¥x28¥x4A
JIS C 6226-1978(78JIS)	ESC \$ @	¥x1b¥x24¥x40
JIS X 0208-1983(83JIS)	ESC \$ B	¥x1b¥x24¥x42

■RFC2237 ISO-2022-JP-1

JIS X 0212-1990(補助漢字)	ESC \$ (D	¥x1b¥x24¥x28¥x44
-----------------------	------------	------------------

■RFC1554 ISO-2022-JP-2

GB2312-1980	ESC \$ A	¥x1b¥x24¥x41
KSC5601-1987	ESC \$ (C	¥x1b¥x24¥x28¥x43

■ISO-2022-JP-3

JIS X 0213:2000の1面	ESC \$ (O	¥x1b¥x24¥x28¥x4F
JIS X 0213:2004の1面	ESC \$ (Q	¥x1b¥x24¥x28¥x51
JIS X 0213:2000の2面	ESC \$ (P	¥x1b¥x24¥x28¥x50

■いわゆる半角カナ

JIS X 0201 片仮名(半角カナ)	ESC (I	¥x1b¥x28¥x49
JIS X 0201 片仮名(半角カナ)	SO	¥x1B¥x28¥x42¥x0E

■中国語・韓国語

ISO-2022-CN(CP 50227)	ESC \$) A	¥x1b¥x24¥x29¥x41
ISO-2022-KR(CP 50225)	ESC \$) C	¥x1B¥x24¥x29¥x43

※RFC1554では ESC \$ (Ft のパターンが定義されている※

エスケープシーケンス(2)

(例) EnCase Grep 正規表現 * 正規表現は十分テストしてから利用 *

ASCIIとJIS X 0201	¥x1b¥x28(¥x42 ¥x4a)
78JISと83JIS	¥x1b¥x24(¥x40 ¥x42)
JIS X 0201と78,83JIS	¥x1b((¥x28¥x4a)) (¥x24(¥x40 ¥x42))
RFC1468全て	¥x1b(¥x28(¥x42 ¥x4a)) (¥x24(¥x40 ¥x42))
JIS X 0213 1面と2面	¥x1b¥x24¥x28(¥x4F ¥x50 ¥x51)

Unicode制御文字(1)

- U+FEFF; ZERO WIDTH NO-BREAK SPACE (BOM: バイト オーダーマーク) } 見えない
- U+200B; ZERO WIDTH SPACE
- U+200C; ZERO WIDTH NON-JOINER
- U+200D; ZERO WIDTH JOINER
- U+202E; RIGHT-TO-LEFT OVERRIDE } 文字方向
- U+202C; POP DIRECTIONAL FORMATTING

意図的にこれらの制御文字を挟むことが可能！

Unicode制御文字(2)

- `a\xFEFF;bc.txt`
- `Algorithm\xFEFF;ID`
- `test\x202E;ARAH\x202C;.txt`
- `a..\xA5;..\xA5;..\xA5;bc.txt`

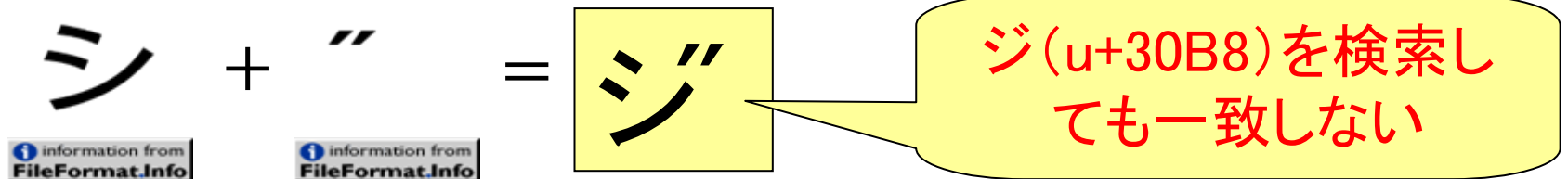
Unicode制御文字の影響

- Windows ではファイル名やレジストリでも Unicode 制御文字を利用できる
- フォレンジックツールのフィルタや検索は制御文字の影響を受ける
- バイナリで確認する
例) MFT レコードを直接確認する
- Unicode 制御文字を検索する
(範囲を狭めて誤検知を防ぐ工夫が必要)

Unicode 文字の合成・結合

- 合成済み文字
- 結合文字列(基底文字+結合文字)
- 合成済み文字+結合文字

- 『シ』
- U+30B7 と U+030B を合成して作成



Unicode正規化(1)

- 正規化形式

NFD (Normalization Form D)

NFC (Normalization Form C)

NFKD (Normalization Form KD)

NFKC (Normalization Form KC)

- D = Decomposition (分解)

C = Composition (合成)

K = Compatibility (互換性)

参照: Unicode正規化とは

<http://homepage1.nifty.com/nomenclator/unicode/normalization.htm>

Unicode正規化(2)

- 例)『ぱ』の正規化

ぱ = U+3071

↓ NFD

U+306F U+309A

ぱ = は ◦

 information from
FileFormat.Info

 information from
FileFormat.Info

- 例) 半角カナの正規化

互換分解により全角に変換される

ハ = U+FF8A U+FF9F

↓ NFKD

U+30CF U+309A

Unicode正規化(3)

- ・ 正規化により、バイト列が変化する
(16進数形式での検索ではヒットしない)
- ・ EFE には、Unicode 正規化形式を指定して検索するオプションが現在のところ無い
- ・ Unicode エディタの利用(正規化)
BabelPad
<http://www.babelstone.co.uk/Software/BabelPad.html>
SC UniPad
<http://www.unipad.org/main/>

Unicode正規化(4)

- ・ Mac OS X のファイル名は正規化されている
- ・ 一部仕様により正規化によって分解されない文字(互換漢字など)が存在する
- ・ Mac OS X の仕様に従った分解を行わなければ、16進形式の検索ではヒットしない危険性がある

- ・ 参照
Unicode に関する微妙な問題
<http://developer.apple.com/ja/technotes/tn1150.html#UnicodeSubtleties>
- ・ 小形克宏の「文字の海、ビットの舟」——文字コードが私たちに問いかけるもの
特別編25
JIS X 0213の改正は、文字コードにどんな未来をもたらすか(8) 番外編:改正
JIS X 0213とUnicodeの等価属性／正規化について(下)
<http://internet.watch.impress.co.jp/www/column/ogata/sp25.htm>

Unicode正規化(5)

- フォレンジック

↓ NFD

フォレンシ[〃]ック

U+3099

- EnCaseで「フォレンシック」と「フォレンジック」を(NFDで正規化後に)検索する
フォレンシ.**{0,1}**ック

Unicode サロゲートペア

- U+10000～U+10FFFFを表現
上位サロゲート U+D800～U+DBFF
下位サロゲート U+DC00～U+DFFF
- 上位サロゲート+下位サロゲートの組合せで
1文字を表す
例) U+10000 = U+D800 U+DC00

𐀀

こちらのパターンで検索する



WORDとUnicode

- Alt+X キーを使うと Unicode 変換が可能
 1. Word を起動する
 2. 変換したい数値を半角英数で入力し確定
例) 3042
 3. Alt+x キーを入力すると変換される
例) あ
 4. 逆方向の変換も可能
例) あalt+x → 3042

EnCase キーワード検索

Edit Keyword

Search expression: 井.{0,4}原

Code Page: GREP

Code Page	Name	Valid	Code
<input checked="" type="checkbox"/> 1	Japanese (Shift-JIS)	•	932
<input checked="" type="checkbox"/> 2	Japanese (EUC)	•	51932
<input checked="" type="checkbox"/> 3	Japanese (JIS)	•	50220
<input type="checkbox"/> 4	Baltic (Windows)		1257
<input type="checkbox"/> 5	Central European (DOS)		852
<input type="checkbox"/> 6	Central European (ISO)		28592
<input type="checkbox"/> 7	Central European (Mac)		10029
<input type="checkbox"/> 8	Central European (Windows)		1250
<input type="checkbox"/> 9	Chinese Simplified (EUC)	•	51936
<input type="checkbox"/> 10	Chinese Simplified (GB18030)	•	54936
<input type="checkbox"/> 11	Chinese Simplified (GB2312)	•	936
<input type="checkbox"/> 12	Chinese Simplified (GB2312-80)		20936

Unicode View: [4E3C].{0,4}[539F]

Preview Code Page: !"#%&()* *+,-./0123456789; :=>? @ABCDEFGHIJKLMN OPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrs tuvwxyz{|}~ ¡ ¢£¥¦§¨ª«¬®¯°±²³´µ¶·¸¹º»¼½¾¿

複数コードページで文字列を一括して検索！！

EnCase Grep 機能(1)

- 例) 以下のパターンで「日本」を検出

日本

日

本

日 本

- 日[`¥x20¥x0d¥x0a`]*本

EnCase Grep 機能(2)

- 例) 日本の中に0~4文字、何かが挟まる
日. {0, 4} 本
- EnCase grep では、Unicode の1文字は
2byte として解釈される
- ・ 問題点 (誤検知の増加・予想が困難)
u+65E5 u+FFFF u+FFFFu+FFFFu+FFFF u+672C

EnCase Grep 機能 (3)

- OR (a|b) オプションの利用 (EFE 5.04a)
伊原 ¥x88¥xC9¥x8C¥xB4
井原 ¥x88¥xE4¥x8C¥xB4
(伊|井)原
- | 利用は注意が必要 (EFE 5.04a)
伊原 ¥x88¥xC9¥x8C¥xB4
伊腹 ¥x88¥xC9¥x95¥xA0
伊(原|腹) → 不完全な検索となる (EFE 5.04a)
Expression 表示 [88] [C9] [8C] [95 A0 B4]
- 上位バイト・下位バイトの並びに注意する

EnCase Grep 機能 (4)

- Unicode 番号を w オプションで指定した場合、ASCII 以外は現状では動かない
- 詳細については GSI 社のメッセージボード
GSI Japanese を参照
<http://www.encase.com/support/messageboards.asp>

コード変換による影響

- [PRB] SHIFT - JIS と Unicode 間の変換問題
<http://support.microsoft.com/default.aspx?scid=kb;ja;JP170559>
- キーワードとして U+2121 Telephone Sign を登録
- CP932 を対象のコードページとして指定した状態で検索を実行すると、0x8784 がヒット
(U+2121 が 0x8784 へ変換される)
- もし、U+2121 Telephone Sign が 0xfa5a として記録されている場合、EFE の検索ではヒットしない

稀なケースであり、この変換が実務上影響を与えることは無いと思われるが、文字コード変換による影響は常に考慮する必要がある！

フォントの確認

- ・ 専用のフォントセットを利用している場合
- ・ GTフォント
東京大学多言語処理研究会
<http://www.l.u-tokyo.ac.jp/GT/>
- ・ 一太郎 JIS X 0213:2004対応フォント
<http://www.ichitaro.com/2005/taro/toku07.html>
- ・ Shift_JIS-2004 (JIS X 0213:2004)フォント

参考資料

- 文字コード超研究
深沢 千尋 (著), ラトルズ ; ISBN: 4899770510
- Unicode標準入門
トニー グラハム (著), 翔泳社 ; ISBN: 4798100307
- 小形克宏の「文字の海、ビットの舟」
—— 文字コードが私たちに問いかけるもの
<http://www.watch.impress.co.jp/internet/www/column/ogata/>

参考資料: Windows環境におけるコードページ

http://msdn.microsoft.com/library/en-us/intl/unicode_81rn.asp より抜粋

Identifier	Name
932	ANSI/OEM – Japanese, Shift-JIS
1200	Unicode UCS-2 Little-Endian (BMP of ISO 10646)
1201	Unicode UCS-2 Big-Endian
20932	JIS X 0208-1990 & 0121-1990
50220	ISO 2022 Japanese with no halfwidth Katakana
50221	ISO 2022 Japanese with halfwidth Katakana
50222	ISO 2022 Japanese JIS X 0201-1989
51932	EUC – Japanese
65000	Unicode UTF-7
65001	Unicode UTF-8

**解き明かす力、今すぐにでも。
フォレンジック・サービス**

NetAgent

The Forensics Company

<http://forensic.netagent.co.jp/>