

# お断り

## ■ 当資料について

- ▶ 全権利は作者(Hideyuki Yahagi)が所有します。
- ▶ 記載されている情報はすべて一般に公開された情報、あるいは独自に確認された情報に基づいており、各社の機密事項は含まれていません。
- ▶ 内容は特定の環境において正確もしくは妥当であることが確認されています。全ての環境において適用されることを保証するものではありません。
- ▶ 当資料の適用あるいは参照から派生した損失・損害について作者は一切の責を負いません。また、作者は内容に対する技術的な支援、問題判別、導入作業などの作業を行なうものではありません。
- ▶ 記載中、作者以外が作成したソフトウェアに関してはそれぞれのライセンスに準じます。
- ▶ 文書中の商標はその所有者に帰属します。



© Copyright Hideyuki Yahagi 1999. All rights reserved.

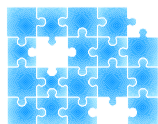
# 目次

- オープン環境におけるサーバーとしてのAS/400
  - ▶ サーバーに求められる機能
  - ▶ AS/400のインターネット機能
- ILE-RPGプログラミング手法
  - ▶ C言語用APIの利用
  - ▶ 例:HTTPログ形式の変換
  - ▶ 例:Cランタイム関数を利用したCGI
  - ▶ ILE-RPGでのsocketの利用
  - ▶ 例:SENDMコマンド
- オープンソース・ソフトウェアのAS/400へのポーティング
  - ▶ オープンソース・ソフトウェア
  - ▶ AS/400へのポーティング
  - ▶ 例:zlib(ファイルの圧縮/解凍)
  - ▶ 例:count(Webアクセスカウンター)
  - ▶ 例:Analog(HTTPログ解析)
  - ▶ 例:AFP2PDFコマンド(AS/400印刷出力をPDFに変換)
- まとめ

© Copyright Hideyuki Yahagi 1999. All rights reserved.

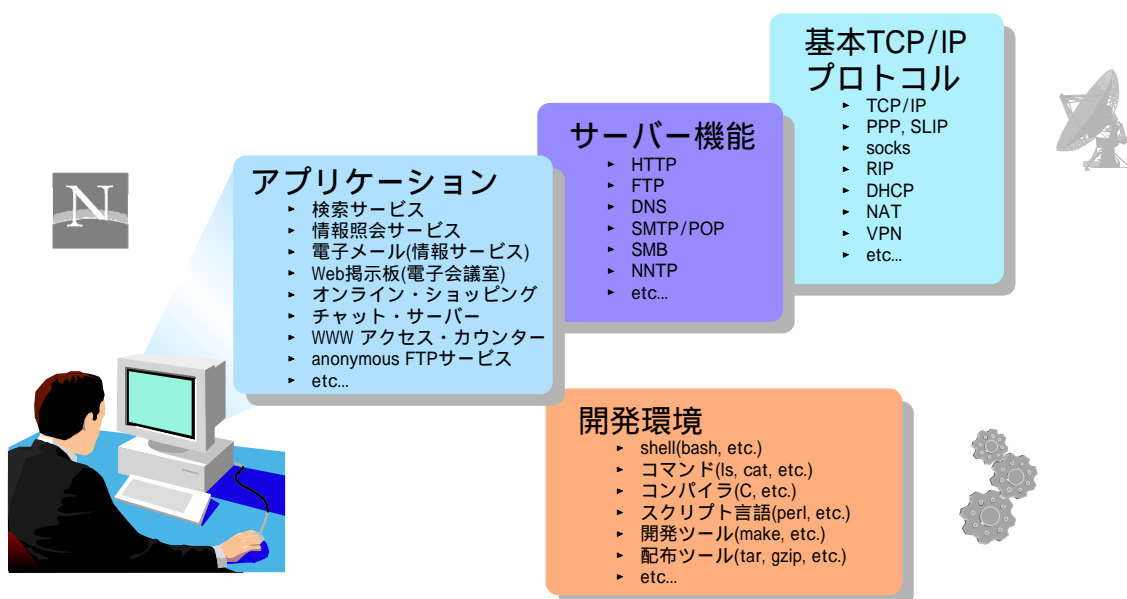
# オープン環境におけるサーバーとしてのAS/400

AS/400はインターネット、あるいはオープン環境のサーバーとして多岐にわたる機能を提供しています。Webアプリケーション・サーバーとしてAS/400を活用する上で他に何が必要でしょうか。



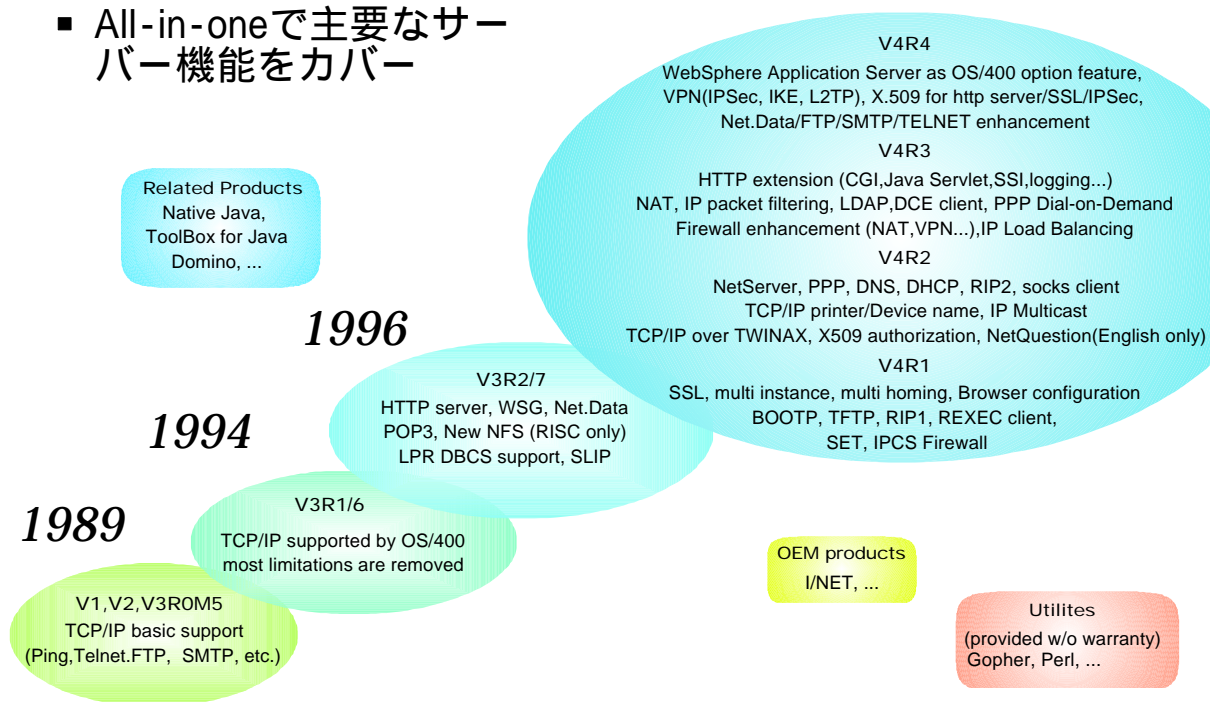
## サーバーに求められる機能

- UNIXサーバーは基本機能を各種ツールが補完



## AS/400のインターネット機能

- All-in-oneで主要なサーバー機能をカバー



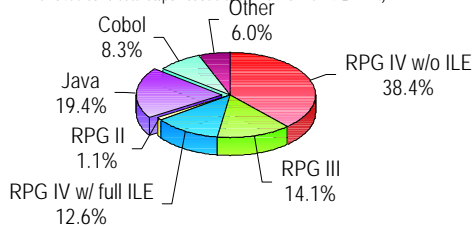
© Copyright Hideyuki Yahagi 1999. All rights reserved.

## AS/400の有効活用

- ILE-RPGの活用

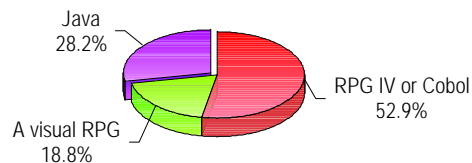
October 1998 'What language are you using now for new AS/400 development?'

(<http://www.news400.com/features/poll2/doublepoll/PrevPoll2.cfm?QID=24>)



September 1999 'For AS/400 business programming which language is the better long-term choice?'

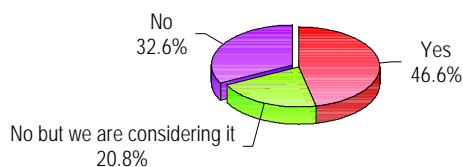
(<http://www.news400.com/features/poll2/doublepoll/PrevPoll2.cfm?QID=35>)



- オープンソース・ソフトウェアの活用

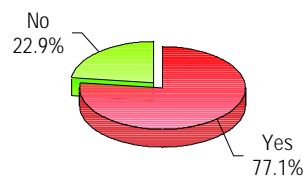
July 1999 'Does your shop incorporate some version of Linux among your supported platforms?'

(<http://www.news400.com/features/poll2/doublepoll/PrevPoll2.cfm?QID=33>)



July 1999 'Should IBM put Linux on the AS/400?'

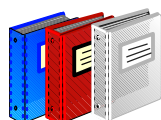
(<http://www.news400.com/features/poll2/doublepoll/PrevPoll2.cfm?QID=9&Q2=yes>)



© Copyright Hideyuki Yahagi 1999. All rights reserved.

# ILE-RPGプログラミング手法

ILE-RPGからUNIX-type APIやCのランタイム・ライブラリーを利用することにより、これまでC言語が使用されていた用途にILE-RPGを適用することが可能となります。



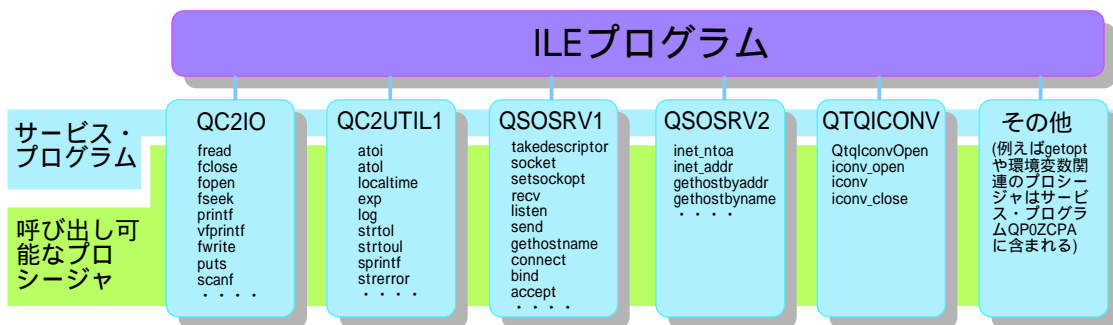
## C言語用APIの利用

- C言語(UNIX)の基本知識が必要
  - ▶ 変数の型(整数、浮動小数点、X'00'で終端された文字列など)、構造体、ポインター、null値
  - ▶ ヘッダーファイル、マクロ、プリプロセッサ
  - ▶ 標準入出力
- APIの習得
  - ▶ Cライブラリー関数
    - 標準入出力、ユーティリティなど
    - 「AS/400 Advanced Series ILE C/C++ for AS/400 Run-Time Library Reference Version 4」(SC09-2715-00)を参照
  - ▶ UNIX-type API
    - 低レベルI/O、socket、ストリーム・ファイルの操作、シグナル、環境変数など
    - 「AS/400e series System API Reference OS/400 UNIX-Type APIs Version 4」(SC41-5875-02)を参照
  - ▶ その他のAPI
    - 文字コード変換(iconv)など



## システム提供の(C言語用)API

- サービス・プログラム(バインド・ディレクトリー)で提供
  - ▶ 公開されている情報は限定的
  - ▶ ILE-C以外の言語からの利用の可否は記載無し
  - ▶ OS/400のバージョンによってバインドの要否が異なる場合あり
  - ▶ DSPBNDDIR/DSPSRVPGMコマンドなどで内容を表示可能
- ILE環境をサポートする言語から使用可能
  - ▶ ヘッダーはC言語用のみ提供



© Copyright Hideyuki Yahagi 1999. All rights reserved.

## ILE-RPGによるC言語用APIの呼び出し

- 関数のプロトタイプ宣言の作成
  - ▶ ILE-RPG用にUNIX-type関数をプロトタイプ宣言

C言語での定義	<code>ssize_t read(int fildes, void *buf, size_t nbyte);</code>
ILE-RPGでの定義	<pre> D read          PR          101 0  EXTPROC('read') D fd            101 0  VALUE D buffer_p      *        VALUE D buffer_length 101 0  VALUE           </pre>

- 一般の変数型はマニュアル「ILE C/C++ for AS/400 Run-Time Library Reference Version 4」(SC09-2715-00)の「1.3.7 Data Type Compatibility」に従って定義
- typedefにより定義されている型はQSYSINC/SYS(TYPES)などの定義を調査

- 関数の呼び出し

- ▶ 必要な場合はQSYSINC/H(FCNTL)などを参照し、「#define」された定数を記述

C言語での呼び出し	<code>FileDes = open(fileName, O_WRONLY O_CREAT O_EXCL, S_IRWXU);</code>
ILE-RPGでの呼び出し	<pre> C          EVAL          FileDesc = open(%ADDR(fileName) : 26 : 448)           </pre>

© Copyright Hideyuki Yahagi 1999. All rights reserved.

## 例 - HTTPログ形式の変換

### ■ HTTPログ形式

- ▶ DDSとCOMMON HTTP構成のLogFormatディレクティブで指定

DDS ... 物理ファイル(例えばQUSRSYS/ACCESSLOG.Q0971001)

COMMON ... ストリーム・ファイル

アクセス時刻: 029/Sep/1997:14:58:47 +0000Z  
クライアント: proxy2.roppongi.japan.ibm.co  
URLアドレス: "GET /cca/main.htm HTTP/1.0"

-----

遅隔 ID: -  
権限 ユーザ: -  
REQ 状況: 200  
バイト数: 1334  
予約済み: -

EBCDIC

ASCII

proxy2.roppongi.japan.ibm.co - -  
[29/Sep/1997:14:58:47 +0000] "GET  
/cca/main.htm HTTP/1.0" 200 1334

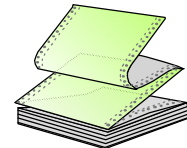
参考: V4R3以降は参照元とユーザー・エージェントが追加

9.188.29.11 - - [11/Oct/1999:15:45:42 -0900]  
"GET /cgi-bin/rpgcgi.pgm?KEY=02030 HTTP/1.1"  
200 388 "http://test400.roppongi.japan.ibm.  
com:8081/c1.html" "Mozilla/4.0 (compatible;  
MSIE 5."

### ■ コンパイルと実行

- ▶ 必要なAPI(UNIX-type APIおよびiconv)は暗黙的にバインド

```
> CRTBNDRPG PGM(ALOGCONV) DFTACTGRP(*NO)
  プログラム ALOGCONV がライブラリー ANA に入っている。最高の重大度は 00
  。 99/10/11 の 14:32:53 に作成されました。
> OVRDBF FILE(ACCESSLOG) TOFILE(QUSRSYS/ACCESSLOG) MBR(*ALL)
> CALL PGM(ALOGCONV) PARM('/tmp/accesslog.txt')
```



© Copyright Hideyuki Yahagi 1999. All rights reserved.

## 使用されている関数

### ■ UNIX-type APIおよびiconv()

- ▶ ストリームファイルの操作
  - unlink() ... ストリームファイルの削除
  - open() ... ストリーム・ファイルのオープン。CCSIDを指定可能
  - write() ... データの書き出し
  - close() ... ストリーム・ファイルのクローズ
- ▶ 文字コード変換
  - iconv() ... 文字コード変換の実行
  - iconv\_open(), iconv\_close() ... 文字コード変換ルーチンの準備/開放

### ■ AS/400の文字コード変換機能

	提供方法	特徴	注意点
iconv()	OS/400に組み込み	他のプラットフォームでも使用可能。変換できる文字コードの組み合わせが豊富	C言語用のAPIインターフェースのためILE-RPGから使うには定義を作成する必要あり。iconv自体のオープン/クローズが必要
QDCXLATE	OS/400に組み込み	S/38時代からの実績	変換できる組み合わせが限定されている(日本語DBCSはソフトJISのみ)
ToolBox/400 漢字変換機能	ライセンス・プログラムの一機能として提供	JEFなど特殊な日本語漢字コードの変換が可能	日本語以外は不可。ライセンスの購入が必要
RPGのXLATE命令	RPGコンパイラの機能	変換テーブルをユーザーが定義可能	DBCSの変換不可

© Copyright Hideyuki Yahagi 1999. All rights reserved.

# HTTP ログ形式の変換(1/2)

```

0001.00 FACCESSLOG IP E DISK
0002.00 *****
0003.00 Dunlink PR 101 0 EXTPROC('unlink')
0004.00 D VALUE
0005.00 Dopen PR 101 0 EXTPROC('open')
0006.00 D VALUE
0007.00 D 101 0 VALUE
0008.00 D 100 0 VALUE OPTIONS(*NOPASS)
0009.00 D 100 0 VALUE OPTIONS(*NOPASS)
0010.00 Dwrite PR 98 0 EXTPROC('write')
0011.00 D 98 0 VALUE
0012.00 D VALUE
0013.00 D 98 0 VALUE
0014.00 Dclose PR 98 0 EXTPROC('close')
0015.00 D 98 0 VALUE
0016.00 *
0017.00 Dfd S 98 0
0018.00 DBytesWrt S 98 0
0019.00 *****
0020.00 * OS/400 National Language Support APIs V4 2.1 Data Conversion APIs
0021.00 Diconv_o PR 52 EXTPROC('iconv_open')
0022.00 D VALUE
0023.00 D VALUE
0024.00 Diconv PR 100 0 EXTPROC('iconv')
0025.00 D 52 VALUE
0026.00 D VALUE
0027.00 D VALUE
0028.00 D VALUE
0029.00 D VALUE
0030.00 Diconv_c PR 101 0 EXTPROC('iconv_close')
0031.00 D 52 VALUE
0032.00 *
0033.00 Diconv_t DS
0034.00 D return_value 101 0
0035.00 D_cd 101 0 DIM(12)
0036.00 *
0037.00 Dtoctype DS
0038.00 D IBMCCSID_2 8 INZ('IBMCCSID')
0039.00 D toccsid 5 INZ('00437')
0040.00 D conv_rvd_2 19 INZ('ALLX'00')
0041.00 *
0042.00 Dfromcode DS
0043.00 D IBMCCSID_1 8 INZ('IBMCCSID')
0044.00 D fromccsid 5 INZ('00037')
0045.00 D conv_options 7 INZ('0000010')
0046.00 D conv_rvd_1 12 INZ('ALLX'00')
0047.00 *

```

```

0048.00 Dibuf S 32767
0049.00 Dobuf S 32767
0050.00 Dibuf_p S
0051.00 Dobuf_p S
0052.00 Disav S
0053.00 Dosav S
0054.00 Dibufflen S 100 0
0055.00 Dobufflen S 100 0
0056.00 Diconv_ret S 100 0
0057.00 *****
0058.00 * other variables
0059.00 Drcd_ctr S 98 0
0060.00 Dlf S 2 INZ(X'0A')
0061.00 Dnull C CONST(X'00')
0062.00 *
0063.00 ILOGSRC 10
0064.00 /EJECT
0065.00 *****
0066.00 * Main routine
0067.00 *****
0068.00 C *ENTRY PLIST path 32
0069.00 C *
0070.00 * initialize
0071.00 C N02 EXSR #ICONV_0
0072.00 C N02 EXSR #OPEN
0073.00 C N02 SETON 02
0074.00 * main
0075.00 C 10 EXSR #ICONV
0076.00 C 10 EXSR #WRITE
0077.00 * closing
0078.00 CLR EXSR #CLOSE
0079.00 CLR EXSR #ICONV_C
0080.00 /EJECT
0081.00 *****
0082.00 * Sub routine
0083.00 *****
0084.00 * open output file (STMF)
0085.00 C #OPEN BEGSR
0086.00 *
0087.00 C EVAL path = %TRIM(path) + null
0088.00 C IF -1 = unlink(%ADDR(path))
0089.00 C ENDIF
0090.00 * mode = O_CREAT + O_WRONLY + O_TRUNC + O_CODEPAGE
0091.00 * permissions = S_IRWXU + S_IROTH
0092.00 * AsciiCodePage = 819
0093.00 C EVAL fd = open(%ADDR(path) :
0094.00 C 8 + 2 + 64 + 8388608 : 448 + 4 : 819)

```

© Copyright Hideyuki Yahagi 1999. All rights reserved.

# HTTP ログ形式の変換(2/2)

```

0095.00 *
0096.00 C IF fd = -1
0097.00 C EVAL buffer = 'create() failed.'
0098.00 C EXSR #QUIT
0099.00 C ENDF
0100.00 *
0101.00 C ENDSR
0102.00 *****
0103.00 * initialize ICONV
0104.00 C #ICONV_0 BEGSR
0105.00 *
0106.00 C EVAL iconv_t = iconv_o(%ADDR(toctype)
0107.00 C : %ADDR(fromcode))
0108.00 *
0109.00 C IF return_value = -1
0110.00 C EVAL buffer = 'iconv_o() failed.'
0111.00 C EXSR #QUIT
0112.00 C ENDF
0113.00 *
0114.00 C ENDSR
0115.00 *****
0116.00 * convert string
0117.00 C #ICONV BEGSR
0118.00 *
0119.00 C EVAL ibuf =
0120.00 C %TRIMR(LOGADD) + ' - ' + LOGTIM + ' ' +
0121.00 C %TRIMR(%SUBST(LOGURL : 2 : 254)) + ' ' +
0122.00 C %TRIMR(%EDITC(LOGSTS : '3')) + ' ' +
0123.00 C %TRIMR(%EDITC(LOGBYT : '3')) + null
0124.00 C
0125.00 C EVAL obufflen = 0
0126.00 C EVAL ibuf_p = %ADDR(ibuf)
0127.00 C EVAL obuf_p = %ADDR(obuf)
0128.00 C EVAL isav = ibuf_p
0129.00 C EVAL osav = obuf_p
0130.00 C EVAL iconv_ret = iconv(iconv_t
0131.00 C : %ADDR(ibuf_p)
0132.00 C : %ADDR(ibufflen)
0133.00 C : %ADDR(obuf_p)
0134.00 C : %ADDR(obufflen))
0135.00 *
0136.00 C IF iconv_ret <> 0
0137.00 C EVAL buffer = 'iconv() failed.'
0138.00 C EXSR #QUIT
0139.00 C ENDF
0140.00 *
0141.00 C ENDSR

```

```

0142.00 *****
0143.00 * write string to STMF
0144.00 C #WRITE BEGSR
0145.00 *
0146.00 C EVAL rcd_ctr = rcd_ctr + 1
0147.00 C EVAL %STR(obuf_p - 1 : 2) = If
0148.00 C BytesWrt = write(fd
0149.00 C : osav : obuf_p - osav)
0150.00 *
0151.00 C IF BytesWrt <> (obuf_p - osav)
0152.00 C EVAL buffer = 'write() failed.'
0153.00 C EXSR #QUIT
0154.00 C ENDF
0155.00 *
0156.00 C ENDSR
0157.00 *****
0158.00 * close output file (STMF)
0159.00 C #CLOSE BEGSR
0160.00 *
0161.00 C IF -1 = close(fd)
0162.00 C EVAL buffer = 'close() failed.'
0163.00 C EXSR #QUIT
0164.00 C ENDF
0165.00 *
0166.00 C ENDSR
0167.00 *****
0168.00 * close ICONV
0169.00 C #ICONV_C BEGSR
0170.00 *
0171.00 C IF iconv_c(iconv_t) = -1
0172.00 C EVAL buffer = 'iconv_c() failed.'
0173.00 C EXSR #QUIT
0174.00 C ENDF
0175.00 * display completion message
0176.00 C EVAL buffer = 'Total record processed - ' +
0177.00 C %TRIMR(%EDITC(rcd_ctr : '3'))
0178.00 C DSPLY buffer 52
0179.00 *
0180.00 C ENDSR
0181.00 *****
0182.00 C #QUIT BEGSR
0183.00 *
0184.00 C DSPLY *INLR = '1' buffer 52
0185.00 C EVAL
0186.00 C RETURN
0187.00 *
0188.00 C ENDSR

```

© Copyright Hideyuki Yahagi 1999. All rights reserved.



# 例 - Cランタイム関数を利用したCGI

## ■ コンパイル

- ▶ QC2LEをバインドしてコンパイル

```
> CRTBNDRPG PGM(RPGCGI2) DFTACTGRP(*NO) ACTGRP(*CALLER) BNDDIR(QC2LE)
```

## ■ 実行と結果の表示

環境変数の  
セットと  
プログラムの  
呼び出し

```
> ADDENVVAR ENVVAR(QUERY_STRING) VALUE('KEY=02030')
環境変数が追加された。
> QSH CMD('system call rpgcgi2')
```

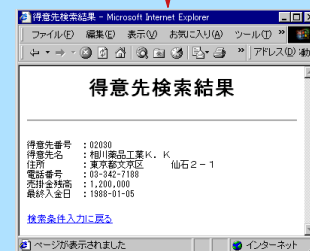
http://(ホスト名:ポート)/(Exec/パス)/rpgcgi2.pgm?KEY=02030

プログラム  
RPGCGI2

プログラム  
RPGCGI2

実行結果の出  
力

```
Content-type: text/html
<HTML><HEAD><TITLE> 得意先検索結果 </TITLE></HEAD>
<BODY><CENTER><H1> 得意先検索結果 </H1><HR></CENTER>
<PRE>
得意先番号 : 02030
得意先名 : 相川薬品工業 K . K
住所 : 東京都文京区 仙石 2 - 1
電話番号 : 03-342-7188
売掛金残高 : 1,200,000
最終入金日 : 1988-01-05
</PRE>
<A HREF="/search.html"> 検索条件入力 </A></BODY></HTML>
端末セッションを終了するためには、実行キーを押してください。
```



© Copyright Hideyuki Yahagi 1999. All rights reserved.

# ILE-RPGによるCGI

```
0001.00 FTKMSP IF E K DISK
0002.00 *
0003.00 Dout PR 256 VALUE
0004.00 D
0005.00 * Prototype for getenv() API
0006.00 Dgetenv PR * EXTPROC('Qp0zGetEnvCCSID')
0007.00 D * VALUE
0008.00 D * VALUE
0009.00 * Variables for getenv() API
0010.00 Denvname S 32
0011.00 Denvccsid S 10I 0 INZ(65535)
0012.00 Denv_p S *
0013.00 * Prototype for fflush() API
0014.00 Dfflush PR * EXTPROC('fflush')
0015.00 D * VALUE
0016.00 *
0017.00 Dnl C X'15'
0018.00 Dnull C X'00'
0019.00 /EJECT
0020.00 * Send HTTP/HTML header
0021.00 C CALLP out('Content-type: text/html' + nl + nl +
0022.00 C '<HTML><HEAD><TITLE> 得意先検索結果 ' +
0023.00 C '</TITLE></HEAD>' + nl +
0024.00 C '<BODY><CENTER><H1> 得意先検索結果 ' +
0025.00 C '</H1><HR></CENTER>' + null)
0026.00 * Read the Environment variable, QUERY_STRING.
0027.00 C EVAL envname = 'QUERY_STRING' + null
0028.00 C EVAL env_p = getenv(%ADDR(envname)) :
0029.00 C %ADDR(envccsid)
0030.00 * Display error if QUERY_STRING is null
0031.00 C IF env_p = *NULL or %STR(env_p) = ''
0032.00 C CALLP out(' <P><B><I><FONT SIZE=2> 環境変数が ' +
0033.00 C '不正です。 </FONT></I></B></P>' + null)
0034.00 * Extract key value
0035.00 C ELSE
0036.00 C EVAL TKBANG = %SUBST(%STR(env_p):5:5)
0037.00 * Search database using key value
0038.00 C TKBANG CHAIN TOKMSR 90
0039.00 * Key not found
```

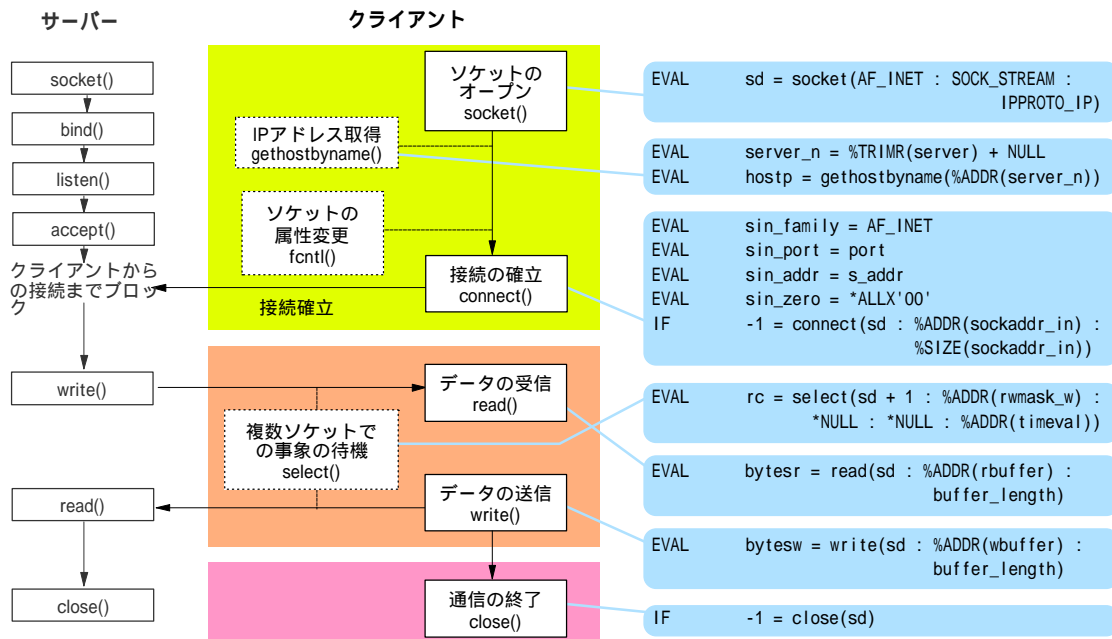
```
0040.00 C 90 CALLP out(' <P><B><I><FONT SIZE=2> 得意先番号が ' +
0041.00 C '見つかりませんでした。(KEY=' +
0042.00 C 'TKBANG + ')</FONT></I></B></P>' + null)
0043.00 * Display search result
0044.00 C TKNYUK datec 10
0045.00 C N90 MOVE
0046.00 C N90 CALLP out(' <PRE> ' +
0047.00 C ' 得意先番号 : ' + TKBANG + nl +
0048.00 C ' 得意先名 : ' + TKNAKJ + nl +
0049.00 C ' 住所 : ' + TKADR1 + TKADR2 + nl +
0050.00 C ' 電話番号 : ' + TKTELE + nl +
0051.00 C ' 売掛金残高 : ' +
0052.00 C %TRIML(%EDITC(TKUJAN:'N')) + nl +
0053.00 C ' 最終入金日 : ' + datec + nl +
0054.00 C '</PRE>' + null)
0055.00 * HTML trailer
0056.00 C CALLP out(' <A HREF="/search.html"> 検索条件入力 ' +
0057.00 C 'に戻る </A></BODY></HTML>' + null)
0058.00 *
0059.00 C CALLP fflush(*NULL)
0060.00 C SETON LR
0061.00 /EJECT
0062.00 C Pout B
0063.00 Dout PI
0064.00 D str 256 VALUE
0065.00 * Prototype for puts() API
0066.00 Dputs PR * EXTPROC('puts')
0067.00 D * VALUE
0068.00 *
0069.00 C CALLP puts(%ADDR(str))
0070.00 C RETURN
0071.00 Pout E
```

© Copyright Hideyuki Yahagi 1999. All rights reserved.



# ILE-RPGでのsocketの利用

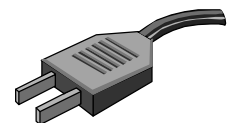
## ■ C言語とまったく同じ手順で使用



© Copyright Hideyuki Yahagi 1999. All rights reserved.

# socketプログラミング上の考慮点

- read/writeのデータ長
  - ▶ read/write実行時に指定したバイト数だけ送信・受信されるとは限らない
  - ▶ 戻り値が0の場合は通常相手がsocketを閉じた事を示す
- タイムアウト処理( V3R7でのテスト結果)
  - ▶ connect()
    - TCP/IPが起動されていないサーバーに接続を試みた場合はただちにエラーになる
    - TCP/IPは稼動しているがsocketの接続ポートが開かれていないサーバーに接続を試みた場合は約3分でエラー。必要ならfcntl()関数で非ブロック化(O\_NONBLOCK)を指定
  - ▶ read()/write()
    - socketでの送受信中に切断した場合、約30分でエラー
    - select()関数と組み合わせてタイムアウトまでの時間をユーザーが制御。select()で使用する「FD\_」で始まるマクロはILE-RPG用にコーディングが必要
- 異常終了時のsocketのクリーンアップ
  - ▶ プログラム(異常)終了時には必ずsocketをclose
  - ▶ オープンされたままのsocketはジョブが終了するまで残る(最大224)
- 文字コード変換
  - ▶ 一般的なサーバーではsocketの先はASCIIなのでコード変換が必要
  - ▶ 許容される文字コードを必ず確認(US-ASCII、ISO-2022-JP、その他)



© Copyright Hideyuki Yahagi 1999. All rights reserved.

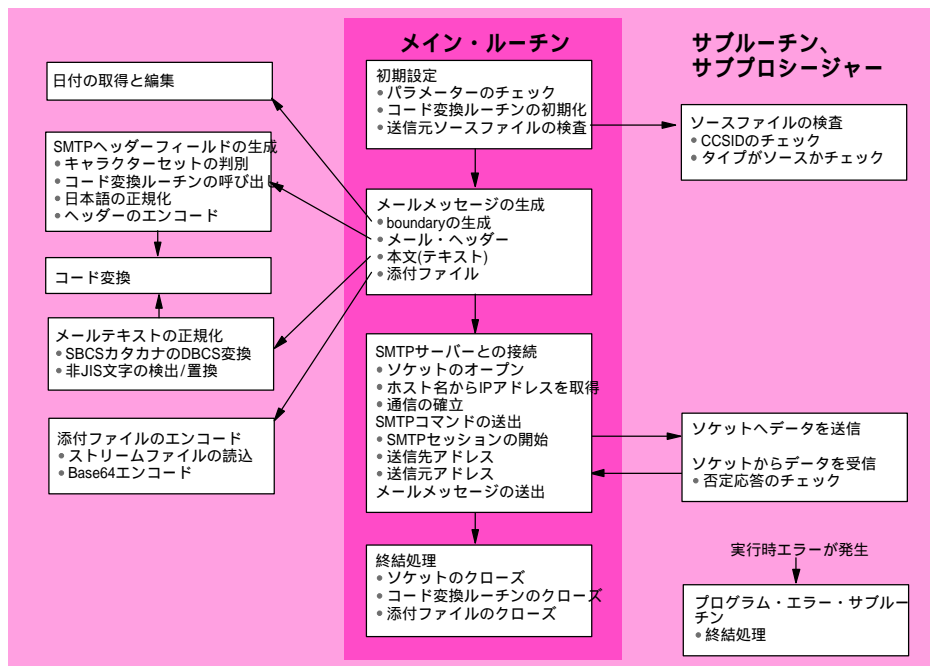
## 例 - SENDMコマンド

- AS/400のコマンドラインから日本語インターネットメールを送信



© Copyright Hideyuki Yahagi 1999. All rights reserved.

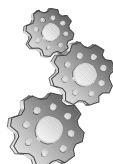
## SENDMコマンド - 処理の概要



© Copyright Hideyuki Yahagi 1999. All rights reserved.

# オープンソース・ソフトウェアの AS/400へのポーティング

ソフトウェアをスクラッチ(何も無いところ)から作成するには膨大な知識とワークロードが必要とされます。インターネット上で配布されているオープンソース・ソフトウェアを有効に利用すれば最小限の労力で優れたソフトウェアをAS/400上でも使用できます。



## オープンソースソフトウェア

### ■ 多様な「オープンソース」ライセンスが存在

- ▶ オープンソース フリーソフトウェア パブリックドメイン
- ▶ オープンソースの定義(<http://www.hams.com/OSD.html>)

	非フリーのソフトウェアとの結合	変更されたソースの占有	元の著作権者の変更に対する特権の保有
■ The FreeBSD Copyright - <a href="http://www.freebsd.org/copyright/freebsd-license.html">http://www.freebsd.org/copyright/freebsd-license.html</a>	許可	許可	なし
■ X Window System License - <a href="http://www.x.org/xlicense.html">http://www.x.org/xlicense.html</a>			
■ Apache License - <a href="http://www.apache.org/LICENSE.txt">http://www.apache.org/LICENSE.txt</a>			
■ GNU GPL(General Public License) - <a href="http://www.gnu.org/copyleft/gpl.html">http://www.gnu.org/copyleft/gpl.html</a>	非許可	非許可	なし
■ The Netscape Public License - <a href="http://www.mozilla.org/NPL/">http://www.mozilla.org/NPL/</a>	許可	許可	あり

### ■ オープンソースの利点

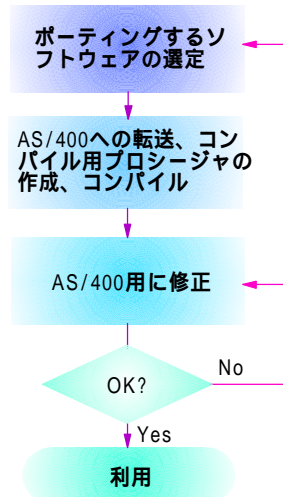
- ▶ (通常は)無料、自由な改変、サポートの終了/陳腐化への保険、高い安全性... など
  - より詳しい情報は「フリーソフトウェアとオープンソースのページ」(<http://openlab.etl.go.jp/freesoft/>)からのリンクなどを参照

## AS/400へのポーティング

### ■ AS/400ポーティングのホームページ

- ▶ AS/400 Porting Center (<http://www.as400.ibm.com/developer/porting/index.html>)

### ■ ポーティングの流れ



- ✓ AS/400上で必要とする機能を定義
- ✓ 商用ソフトを含めて幅広くリサーチ
  - ▶ ライセンス条件を検討
  - ▶ 多種類のOSに移植されているソフトウェアを選択
  - ▶ AS/400でサポートされていない機能を使用していないか確認
  - ▶ 自社で開発環境を持っているか
- ✓ 必要なソースファイルの識別
- ✓ FTPやクライアント・アクセスでAS/400へ転送
- ✓ makeファイルなどを参考にCLPやREXXでコンパイル用のプロシージャを作成
- ✓ コンパイルエラーが発生する場合はこれを修正
- ✓ コンパイルが通ったら動作を確認
- ✓ 必要であれば補助プログラムを開発
- ✓ 修正が困難な場合は別のパッケージも考慮
- ✓ さまざまな実行環境(OS/400のバージョン、日本語/英語環境など)を想定してテストを実施
- ✓ ソース公開の是非を検討
- ✓ オリジナル版のバージョンアップへの対応
- ✓ ドキュメントの作成、メンテナンス、機能追加

© Copyright Hideyuki Yahagi 1999. All rights reserved.

## Cプログラムのポーティング

### ■ 移植が容易なソフトウェアを選択

- ▶ 各社UNIXを含む多数のプラットフォームに移植されている
- ▶ 小規模なパッケージ
- ▶ AS/400でサポートされていない機能/関数を使用していない
  - グラフィック、カーソル制御を使用していないこと
  - fork(), mktemp(), setuid()などはAS/400では提供されていない
- ▶ 実行速度などの要件に合えばJava版も検討



### ■ 基本はUNIX

- ▶ UNIX、VMS、Win32などプラットフォームに選択肢がある場合はUNIX(ANSI-C)を選択
- ▶ 通常はconfigureスクリプトで環境の判別を行ない、ヘッダーファイル中の#define/#ifdefで移植性を確保 AS/400では調査の上、独力で設定

### ■ 「make」の代わり

- ▶ CLPやREXXでコンパイル用のプロシージャを作成
  - IFSにアクセスする場合、コンパイル時にSYSIFCOPT(\*IFSIO)を指定
  - コンパイル・リストを出力するにはOUTPUT(\*PRINT)を指定
  - make中で「-D」により定義されている値をDEFINEパラメーターで指定
  - 必要であればその他の特殊なオプション(ネイティブ・スレッドやテラススペースの使用)を指定
- ▶ 使用頻度と使用形態により、モジュールのまま全てバインドするか/サービスプログラムにしておくかを判断

© Copyright Hideyuki Yahagi 1999. All rights reserved.

## ポーティングの勘所(1/2)

### ■ EBCDIC or ASCII or ?

- ▶ EBCDIC/ASCII/バイナリを意識
  - コンソールI/O、ファイル名などはEBCDIC
  - ファイル中のデータ、通信データは通常ASCIIまたはバイナリ

#### ▶ EBCDICの種類

- CCSID 037 1027
- 実行環境によっては['や']などが化ける

```
> CALL PGM(ZLIB/MINIUNZ)
~~~~~
```

Usage : miniunz [-exv] file.zip [file\_to\_extract]

#### ▶ EBCDICとASCIIで1:1対応しない文字

- EBCDICでは'\'と'¥'は異なる文字

#### ▶ ASCIIプラットフォーム特有のコーディング

```
if ((rep>='a') && (rep<='z'))
#ifdef AS400
    rep -= 0x20;
#else
    rep += 0x40;
#endif
```

### ■ 文字コード変換

- ▶ リテラルは#pragma convertで変換

```
#ifdef AS400
#pragma convert(850)
#endif
if (c == ';') /* terminator */
    break;
if (c == '!') /* EXTENSION ! */
#ifdef AS400
#pragma convert(0)
#endif
```

- ▶ 必要の場合はEBCDIC<->ASCII変換ルーチンを用意し、これを使用

- 漢字をサポートする場合、文字列長が変化
- printf()などの出力は変数を經由して変換
 

```
j = sprintf(wbuf, "%d 0 obj \n", obj_nr);
fprintf(fp, "%s", toascii2(wbuf));
```

- ▶ ケースによってはLIBASCII(wrapper)を検討

- AS/400版(ヘッダーとオブジェクト配布)
 <http://www.as400.ibm.com/domino/libascii.htm>
- OS/390版(ソース配布)
 <http://www2.s390.ibm.com/products/oe/libascii.html>

## ポーティングの勘所(2/2)

### ■ IFS or QSYS.LIB

- ▶ ファイルシステムの差により書式を変更

```
#ifdef AS400
char* versionFile = "tiff/lib/qcsrc(version)";
char* alphaFile = "tiff/lib/qcsrc(alpha)";
#else
char* versionFile = "../VERSION";
char* alphaFile = "../dist/tiff.alpha";
#endif
```

- ▶ DBCS環境でテキスト・モードのオープンを行う場合、CCSIDを指定

```
#ifdef AS400
fp = fopen(buf, "r");
#else
fp = fopen(buf, "r, ccsid=37");
#endif
~~~~~
#ifdef AS400
fd=open(filepath,O_RDWR|
O_CREAT|O_CODEPAGE,0644,37);
#else
fd=open(filepath,O_RDWR|O_CREAT,0644);
#endif
```

### ■ 入出力

- ▶ コンソールから応答値を返す場合は必ず直前に改行

```
#ifndef AS400
printf("... [y]es, [n]o, [A]ll: ",write_filename);
#else
printf("... [y]es, [n]o, [A]ll: \n",write_filename);
#endif
scanf("%1s",answer);
```

- ▶ 標準入出力のリダイレクトはOVRDBFコマンドあるいはQshellを利用

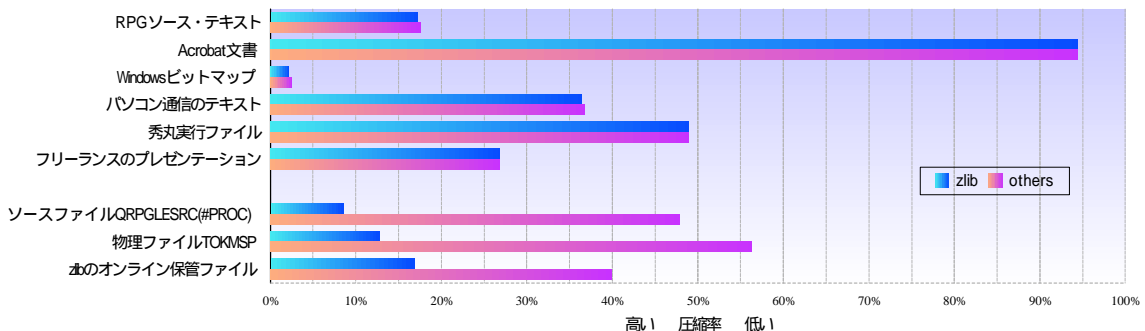
CRTPF	FILE(QTEMP/PIPE) RCDLEN(256)
OVRDBF	FILE(STDOUT) TOFILE(QTEMP/PIPE)
CALL	PGM(ANLGFORM)
DLTOVR	FILE(STDOUT)
OVRDBF	FILE(STDIN) TOFILE(QTEMP/PIPE)
CALL	PGM(ANALOG) PARM('+g-')
DLTOVR	FILE(STDIN)

### ■ その他

- ▶ 必ずしも必要でない関数、機能はコメントアウト
- ▶ 行き詰まったら別のパッケージを再検討

## 例 - zlib, minizip/minizunz

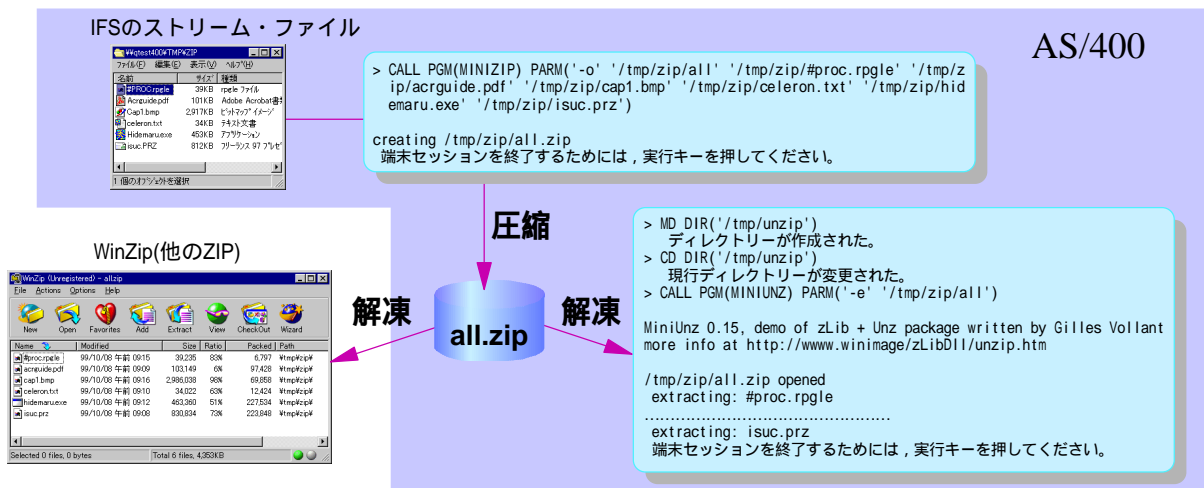
- AS/400上のファイルを圧縮/解凍
  - ▶ ホームページは<http://www.cdrom.com/pub/infozip/zlib/>
  - ▶ IFSのストリーム・ファイル、QSYS.LIBファイルシステムの物理ファイル进行操作可能
- ファイルを圧縮して転送/保管
  - ▶ 他のプラットフォーム上のZIP(WinZipなど)との互換性あり
  - ▶ 通信回線のバンド幅の節約、保管スペースの削減
- 圧縮率はWinZipと同等、SAVxxxより高い



© Copyright Hideyuki Yahagi 1999. All rights reserved.

## minizip/minizunzの使用例

- minizip/minizunz
  - ▶ ホームページは<http://www.winimage.com/zLibDll/unzip.html>
  - ▶ zlibのパッケージに含まれる
  - ▶ 機能は限定的(ワイルドカードやマルチボリュームなどは非サポート)








© Copyright Hideyuki Yahagi 1999. All rights reserved.



## 例 - Count (グラフィカル・アクセス・カウンター)

### ■ WWW Homepage Access Counter and Clock!

- ▶ ホームページは<http://www.fccc.edu/users/muquit/Count.html>
- ▶ GIFイメージ(カウンターや時計、その他)をダイナミックに生成し、ブラウザに送信
- ▶ 表示例

デフォルトのカウンター	df=sample.dat	
12時制の時刻	display=clock	
カンマ付き、数字タイプ「B」	df=sample.dat&dd=B&comma=T	
カスタム・イメージ	df=sample.dat&strip=f&ft=5&dd=mah	
エラー表示	(不正な指定)	

### ■ カウンターの利用

- ▶ %%BINARY%%モード(V4R2以降で利用可)でリクエストをASCIIで処理
  - 例: Exec /CGICOUNT/\* /QSYS.LIB/WCOUNT.LIB/\* %%BINARY%%
- ▶ HTMLでは「IMG」タグを使用
  - 例: 

© Copyright Hideyuki Yahagi 1999. All rights reserved.

## Countのファイル構成

### ■ デジットはストリームファイル、その他は物理ファイル

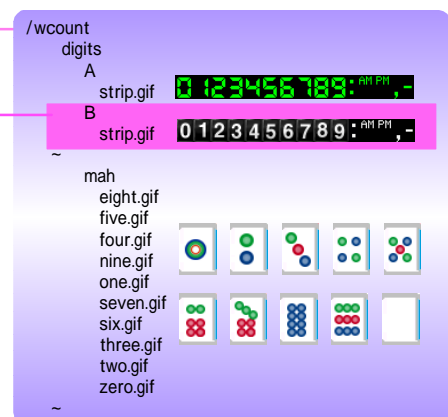
- ▶ WCOUNT/H.CONFIG

```
#define CONFIG_DIR "WCOUNT/CONFIG"
#define CONFIG_FILE "(CONFIG)"
#define DATA_DIR "/QSYS.LIB/WCOUNT.LIB/DATA.FILE"
#define DIGIT_DIR "/wcount/digits"
#define LOG_DIR "WCOUNT/LOGS"
#define LOG_FILE "(LOG)"
```

### ■ カウンター値を記録するデータ・ファイルや表示形式はURLで指定

- ▶ (<http://ホスト名:ポート/Exec/パス...>)  
.../COUNT.PGM?df=sample.dat&dd=B&comma=T&...

/QSYS.LIB/WCOUNT.LIB/DATA.FILE/SAMPLE.DAT.MBR



### ■ カウンターのイメージ

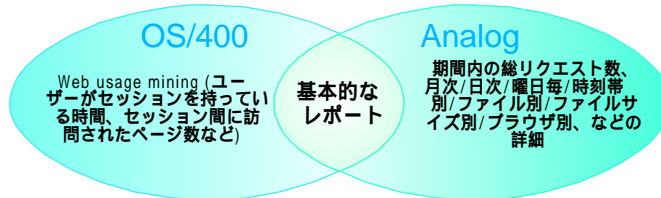
- ▶ 数百種のイメージを「Digit Mania」(<http://www.digitmania.hollowwww.com/>)からダウンロードして使用可能

© Copyright Hideyuki Yahagi 1999. All rights reserved.



## 例 - Analog (HTTPログ解析)

- Analog - The most popular logfile analyser in the world.
  - ▶ ホームページは<http://www.statslab.cam.ac.uk/~sret1/analog/>
  - ▶ HTTPアクセスログを高速に解析し、多彩なレポートを出力
- OS/400のログ解析機能(V4R4～)と相補



### ■ Analogの設定と利用

- ▶ HTTP構成でcommon形式のログを取得するように設定
- ▶ Analog構成ファイルの設定
  - ホスト名や言語情報など一般的な設定情報を記載(詳細はAnalog添付のdocs¥Readme.htmlを参照)
  - コマンドラインの「+g」オプションで指定
- ▶ フォームインターフェースを使用する場合はHTTP構成にディレクティブを追加

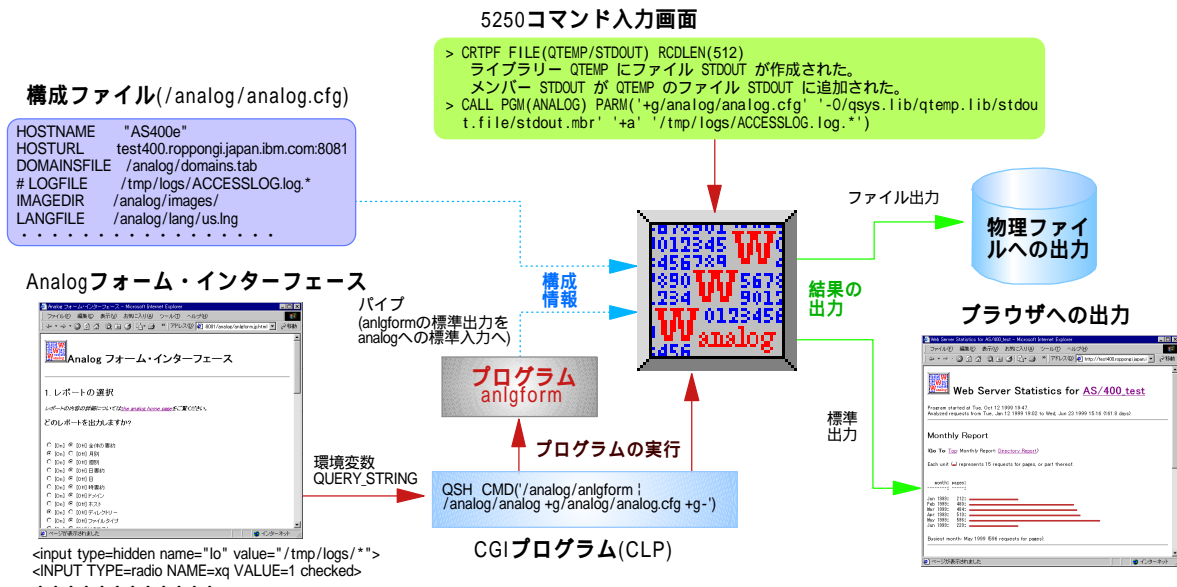
```
Exec /ANACGI/* /QSYS.LIB/ANALOG.LIB/* %%EBCDIC%%
Pass /analog/*
```

例

© Copyright Hideyuki Yahagi 1999. All rights reserved.

## Analogの稼動環境

- 直接プログラムをcall、またはブラウザからフォームインターフェース経由で表示



© Copyright Hideyuki Yahagi 1999. All rights reserved.

## 例 - AFP2PDFコマンド (AS/400印刷出力をPDFに変換)

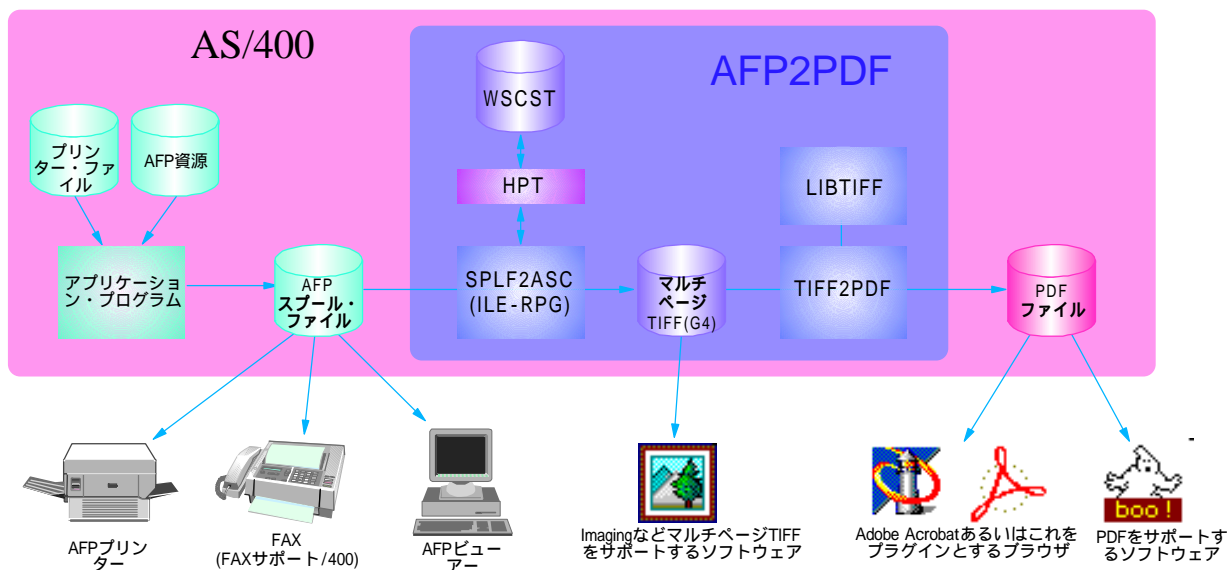
- OS/400上のAFPスプールファイルをPDF形式のストリーム・ファイルに変換
  - ▶ 「TIFF Software」のホームページは<http://www.libtiff.org/>
  - ▶ 「TIFF2PDF」の開発者はRudi Sluijtmann (R.W.Sluijtmann@kub.nl)
- 印刷装置に依存しない表現力の高い文書の配布
- 使用されている技術
  - ▶ AFP(Advanced Function Printing/Presentation - 高機能印刷)
    - 多種のフォント、イメージ、罫線、バーコードなどを含む表現力の高い(印刷)出力を行うためのアーキテクチャー、もしくは機能。詳細はAS/400のAFP関連マニュアルを参照。
  - ▶ HPT(Host Print Transform)
    - OS/400のSCSあるいはAFPDS印刷データストリームをASCIIプリンターで出力可能な形式に変換する機能。詳細はRedbook「AS/400 Printing V」(SG24-2160)などを参照。
  - ▶ TIFF(Tag Image File Format)
    - 主にラスター(イメージ)データの交換に使用される画像形式。カラー/モノクロ、マルチページの画像などを記録。詳細は<http://home.earthlink.net/~ritter/tiff/>などを参照。
  - ▶ PDF(Portable Document Format)
    - 電子文書配信用のフォーマット。文書を作成したアプリケーションに依存せず、各種プラットフォーム上で閲覧を可能とする。詳細は<http://www.adobe.co.jp/products/acrobat/adobepdf.html>を参照。



© Copyright Hideyuki Yahagi 1999. All rights reserved.

## 変換の過程

- AFPスプールファイル TIFFG4イメージ PDF
  - ▶ AFP TIFF変換は新規開発のILE-RPGプログラム、TIFF PDFは2種のソフトウェアをポーティング
  - ▶ 出力用紙がA4固定など機能的な制約あり

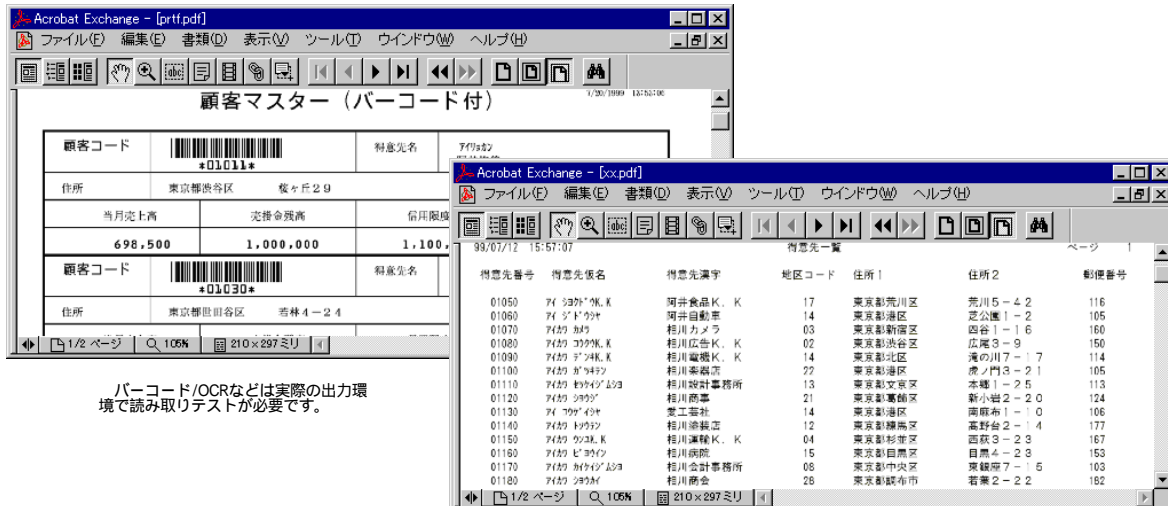


© Copyright Hideyuki Yahagi 1999. All rights reserved.

## PDF出力の考慮点

### ■ イメージとしてPDFを生成

- ▶ 意図に近い印刷出力が得られる(印刷装置による差が出にくい)、機能的な制約が少ない、改ざんが難しい、日本語が使用可能
- ▶ 文字としての処理(Cut&Pasteや検索など)が不可能、出力装置の解像度を生かせない



© Copyright Hideyuki Yahagi 1999. All rights reserved.

## AFPスプールファイルの生成

### ■ 内部(プログラム)記述プリンターファイル

- ▶ OVRPRTFコマンドでDEVTYPE(\*AFPDS)およびフォントを指定

```
> OVRPRTF FILE(QSYSPRT) DEVTYPE(*AFPDS) CDEFNT(QFNT61/XOM13U) IGCCDEFNT(QFNT61/XOM26F)
```

- ▶ 桁ずれを防ぐためDBCS:SBCSが2:1になるフォントを選択(日本語フォントは要購入)
- ▶ フォントにより文字間隔、文字サイズが決定 出力イメージが従来と相違

### ■ 外部記述プリンターファイル

- ▶ 基本的に印刷位置を座標で指定

```
A      R BAR
A      CODE30F9      5      BARCODE(CODE30F9 *AST *HRI)
A      BX              5S 3P  POSITION(&BY &BX)
A      BY              5S 3P
```

印刷装置ファイル

```
C      EVAL BY = IPOS(ROW:LPI)
C      EVAL BX = IPOS(COL:CPI)
C      EVAL CODE30F9 = STR
C      NU1 WRITE BAR
```

RPGプログラム

### ■ APWでは出力をAFPデータストリームに指定可能

- ▶ オプション・フィーチャー「5769-AP1 IPDSプリンター・サポート」が必要
- ▶ MRGAPWコマンドでDEVTYPE(\*AFPDS)を指定

© Copyright Hideyuki Yahagi 1999. All rights reserved.

## まとめ

ILE-RPGの潜在的な機能を有効に活用し、優秀なオープンソースソフトウェアを取り込むことにより、AS/400の卓越したインターネット・サーバーとしての評価はさらに確固たる物となります。



## まとめ

- ほとんどの場合、CでできることはILE-RPGでも可能
  - ▶ AS/400担当者の既存のスキルを活用
  - ▶ 今後ますます重要になる技術(ストリームファイルやsocketの操作)が利用可能
- 優れたポータビリティ
  - ▶ 他プラットフォーム用のソフトウェアを容易に移植

ソフトウェア	総行数の概算 (ヘッダーとプログラム、コメント含む)	AS/400用に変更した箇所
SENDM	3,117	N/A
zlib	12,295	10
count	15,276	48
Analog	12,681	15
libtiff	19,609	17

- インターネット時代のサーバーとしての資質を再認識