

# BREW 既知の問題点及びプログラミング Tips

2005.03.08 クアルコムジャパン

## 1. AEEMediaUtil

### 1.1. AEEMediaUtil\_CreateInstance による SMAF のロードについて

ジャンル：バグ

対象：A5304T、SDK2.0.1.5

現在、MMD\_BUFFER を指定して、SMAF データを AEEMediaUtil\_CreateMedia() で読み込もうとすると、NULL が返ります。これは、ISHELL\_DetectType() が SMAF データを識別できない為です。

もし、バッファ上の SMAF データを再生したい場合は、AEECLSID\_MEDIAPHR/AEECLSID\_MEDIAMMF (データフォーマットに依存) を指定して IMedia オブジェクトを ISHELL\_CreateInstance していただき、このオブジェクトに IMEDIA\_SetMediaData して下さい。

(2003.02.27)

## 2. I3D インタフェース

### 2.1. I3D インタフェースの廃止について

ジャンル：Tips

対象：2.1 エミュレータ 及び 実機

I3D インタフェースは仕様変更に伴い、3.0 で廃止されることになりました。新しい 3D 機能については別途リリースされる予定です。

(2003.12.11)

補記

2004 年 3 月、OpenGL ES 互換の新しい 3D 機能 API がリリースされました。

<https://brewx.qualcomm.com/brew/sdk/download.jsp?page=dx/devmisc>

上記ページよりダウンロード可能です (2004.04.08 現在)。

(2004.04.08)

## 3. IStream インタフェース

### 3.1. IASTREAM\_Cancel

ジャンル：ドキュメントエラー

対象：API リファレンス 2.0.1.5

IASTREAM\_Cancel()は API リファレンスで下記のように定義されていますが、

```
void IASTREAM_Cancel(IAStream * pIAStream, PFNNOTIFY pfn, void * pUser)
```

正しい定義は下記の通りです。

```
void IASTREAM_Cancel(IAStream * pIAStream)
```

(2003.02.27)

補記

API リファレンス 2.1 (英語版)にて修正されました。日本語版は 3.0 にて修正されました。

(2003.05.20)

## 4. IBitmap インタフェース

### 4.1. IBITMAP\_BltOut

ジャンル：Tips

対象：ALL

アプリケーションから直接 IBITMAP\_BltOut をコールすると EUNSUPPORTED が返ります。IBITMAP\_BltIn をご使用ください。

(2003.02.27)

### 4.2. IBITMAP\_GetPixel

### 4.3. IBITMAP\_SetPixel

ジャンル：Tips

対象：ALL

これらの API は、DDB に対してしか使用できません。

(2003.07.11)

## 5. IBTAG インタフェース

### 5.1. 仕様変更について

ジャンル：Tips

対象：ALL

Bluetooth 関連の API は 3.0 で全面的に改定されます。

(2003.11.13)

## 6. IBTSDP インタフェース

### 6.1. 仕様変更について

ジャンル：Tips

対象：バージョン 2.1 までの SDK

Bluetooth 関連の API は 3.0 で全面的に改定されます。

(2003.11.13)

## 7. IBTSIOPORT インタフェース

### 7.1. 仕様変更について

ジャンル：Tips

対象：ALL

Bluetooth 関連の API は 3.0 で全面的に改定されます。

(2003.11.13)

## 8. ICallHistory インタフェース

### 8.1. ICALLHISTORY\_EnumNext

ジャンル：ドキュメントエラー

対象：BREW 2.1 新インタフェースリファレンス

API リファレンス中には下記のように定義がありますが、

```
const AEECallHistoryEntry ICALLHISTORY_EnumNext
(
    ICallHistory *pche,
    int *pnErr
);
```

正しくは下記の通りです。

```
const AEECallHistoryEntry * ICALLHISTORY_EnumNext
(
    ICallHistory *pche,
    int *pnErr
);
```

(2003.05.20)

## 9. ICamera インタフェース

### 9.1. AEECameraNotify

ジャンル：ドキュメントエラー

対象：BREW2.1 新インタフェースリファレンス(SDK2.1.0 Ja)

本構造体の「定義」の項では、「AEECameraStatus」という名前でこの構造体が定義されていますが、正しくは「AEECameraNotify」です。

(2004.07.12)

## 10. ICipher インタフェース

### 10.1. ICIPHER\_Flush

ジャンル：Tips

対象：A5304T、2.1 以前の全ての SDK と実機

現在、ICIPHER インタフェースではブロック暗号アルゴリズムをサポートしておりません。このため、この関数は現在使う必要がありません。

(2003.02.27)

## 11. IClipboard インタフェース

### 11.1. ICLIPBOARD\_GetSequence

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

説明には、クリップボードからデータを削除した際にはカウントが減ると記述がありますが、実際にはカウントは減りません。一方的に増えます。

(2003.02.27)

補記

API リファレンス 2.1 (英語版) にて修正されました。日本語版は未修正です。

(2003.05.20)

### 11.2. ICLIPBOARD\_MaxFreeBlock

ジャンル：バグ

対象：エミュレータ 2.0.1.5

現在、この API は正常に動作しません。第 3 引数の pdwUsed は常に 0 が返ります。

(2003.02.27)

## 12. IDatabase インタフェース

### 12.1. IDATABASE\_CreateRecord

ジャンル：バグ

対象：2.1 以前の SDK 及び実機

IDATABASE\_CreateRecord は、レコードの作成に失敗した場合、NULL を返すようデザインされていますが、現在、ファイルシステムが full の時に、作成に失敗しても NULL を返さないというバグが報告されています。

(2003.02.27)

なお、実際にレコードが作成されたかどうかは、IDBRECORD\_GetID でレコードの ID を調べることで確認することが出来ます。作成されたレコードの ID が 0xffff の場合はレコードの作成に失敗していますので、作成された IDBRecord オブジェクトをリリースしてください。

```
pme->m_pDBRecNew = IDATABASE_CreateRecord(pme->m_pDB, pDBFields,
                                           iNumFields);

if ( pme->m_pDBRecNew == NULL )
    return NULL;

if ( IDBRECORD_GetID(pme->m_pDBRecNew) == 0xffff )
{
    IDBRECORD_Release(pme->m_pDBRecNew);
    pme->m_pDBRecNew = NULL;
    return NULL;
}

else
    return pme->m_pDBRecNew;
```

(2004.02.12)

## 13. IDateCtl インタフェース

### 13.1. DateCtl Property の新規追加

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

DP\_CENTERED が 2.0.0.14 より追加されました。

IDATECTL\_SetProperties、IDATECTL\_GetProperties で使用できます。(上記 2 つの API リファレンスには記述されていません。)

(2003.02.27)

### 13.2. IDATECTL\_EnableCommand

ジャンル：Tips

対象：全ての SDK および実機

IDATECTL\_EnableCommand() を AEECLSID\_DATEPICKCTL に対して使用した

場合、IDATECTL\_Redraw()をコールした際に、EVT\_COMMAND ( wParam は 0xffff)がアプリに通知されることがあります。このイベントは、IDATECTL\_SetDateなどで、異なる月の日付をセットした後に再描画を行った場合に発生します。IDATECTL\_EnableCommand をコールしていない場合には上記イベントは発生しません。

(2004.02.12)

### 13.3. IDATECTL\_GetDate

### 13.4. IDATECTL\_SetDate

**ジャンル：ドキュメントエラー**

対象：2.0.1.5 API リファレンス

これらの API の Year、Month、Day の値は unsigned int と記述されていますが、ヘッダ上は signed int となっています。

(2003.06.23)

### 13.5. IDATECTL\_GetDateString(1/2)

**ジャンル：バグ**

対象：2.1 以前の SDK、2.0 の全ての端末

IDATECTL\_GetDateString において、以下の日付フォーマットを使用すると、月名のところで文字列が切り捨てられてしまうバグがあります。

DFMT\_DD\_MON\_YYYY

DFMT\_DD\_MON\_YY

DFMT\_MON\_DD\_YYYY

DFMT\_MON\_DD\_YY

(2003.07.11)

### 13.6. IDATECTL\_GetDateString(2/2)

### 13.7. IDATECTL\_GetDayString

### 13.8. IDATECTL\_GetMonthString

**ドキュメントエラー**

対象：2.0.1.5 API リファレンス

これらの API の nMaxChars、pnChars は unsigned int と記述されていますが、ヘッダ上は signed int となっています。

(2003.06.23)

## 14. IDBMgr インタフェース

### 14.1. IDBMGR\_OpenDatabase

**ジャンル：バグ**

対象：2.0 以前の全ての実機

IDBMGR\_OpenDatabase()コール時、レコードの状態によってはデータベースファイルの更新作業が行われますが、この際、ディスクスペースに十分な空き容量がないとオリジナルのデータベース自体が削除されてしまう不具合があります。このため、データベースをオープンする前に、ファイルシステムの空き容量を確認してください。

(2004.04.02)

## 15.IDBRecord インタフェース

### 15.1.IDBRECORD\_GetID

**ジャンル：ドキュメントエラー**

**対象：2.0.1.5 API リファレンス 及び 3.0 API リファレンス**

本 API は、エラーコードとして AEE\_DB\_ENULLREC (0x302) を返すとありますが、これは誤りです。本 API の戻り値のうち、0x0000 ~ 0xFFFFE までの値は全て有効なレコード ID です。逆に、DB レコードが無効である場合には、0xFFFF が返ります。

(2004.04.02)

### 15.2.IDBRECORD\_Remove(1/3)

**ジャンル：Tips**

**対象：ALL**

IDBRECORD\_Remove()をコールしてレコードを削除しても、物理的なデータベースファイルそのものは直ちにアップデートされません。削除した数が閾値に達した時にはじめて物理的なファイルがアップデートされます。

なお、閾値の値はメーカー様定義であり、実装によって異なります。標準では、無効なレコードの数が 5 以上、かつ、データベースの無効なレコードの数がデータベースの有効なレコードの数の 50%以上に達した場合にデータベースの圧縮が行われます。なお、ここで言う「無効なレコード」の数は、あるレコードのデータを異なるサイズのデータで更新した場合にも増加します。

例えば、

- 30 個のレコードを作成し、10 個のレコードを削除した場合、有効なレコードは 20、無効なレコードが 10 となり、無効なレコード(10)が有効なレコード(20)の 50%に達するためデータベースは圧縮されます。
- 30 個のレコードを作成し、10 個のレコードを更新したあとに、5 個のレコードを削除した場合、有効なレコードは 25 に対し無効なレコードは  $10+5=15$  となり、データベースは圧縮されます。

なお、2.0 機においては[IDBMGR\\_OpenDatabase\(\)に関するバグ情報](#)も参照してください。

(2003.09.10)

### 15.3. IDBRECORD\_Remove(2/3)

**ジャンル：Tips**

対象：2.1 以前の SDK および実機

IDBRECORD\_Remove()のコール後、物理的なデータベースファイルのアップデートを行う場合には、ワークスペースとして作成したデータベースファイルと同サイズの空き容量が必要となります。なお、2.0 機においては[IDBMGR\\_OpenDatabase\(\)に関するバグ情報](#)も参照してください。

(2004.04.02)

3.0 以降では、この挙動は変更され、データベース圧縮の作業用スペースは不要となりました。なお、3.0 においては、[IDBRECORD\\_Remove\(\)に関するバグ情報\(3/3\)](#)も参照してください。

(2005.09.02)

### 15.4. IDBRECORD\_Remove(3/3)

**ジャンル：バグ**

対象：3.0 および 3.1 SDK

データベースのサイズが 64KB を超える場合、IDBRECORD\_Remove()を行ってデータベースの圧縮が行われるとデータベースが壊れる可能性があります。

この問題は SDK でのみ発生し、3.1 の実機では発生しません。

(2005.09.02)

## 16. IDialog インタフェース

### 16.1. サスペンド時の解放処理

**ジャンル：Tips**

対象：ALL

サスペンド時、BREW 実行環境は全てのダイアログを閉じようとします。このため、アプリケーションの側で ISHELL\_EndDialog を呼ばなくとも、ダイアログは閉じられ、EVT\_DIALOG\_END イベントが通知されます。IDialog オブジェクトは実行環境が自動的に Release しますので、IDIALOG\_Release をコールする必要はありません。

(2003.10.09)

### 16.2. ダイアログの解放

**ジャンル：Tips**

対象：ALL

ISHELL\_EndDialogなどでダイアログを終了させる場合、IDialog オブジェクト自体は実行環境により自動的に Release されます。アプリケーションの側で IDIALOG\_Release を呼ぶ必要はありません。

(2003.10.09)

## 17. IDIB インタフェース

### 17.1. インタフェース全般の説明、使用例： のセクション

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

ビットマップのフォーマットのサンプルが表になっていますが、項目名がずれております。正しくは以下の通りです。

	nPitch	pBmp (ビット配列の先頭)
Top-down BMP file	12	0
Bottom-up BMP file	-12	96
Top-down packed bitmap	9	0
Bottom-up packed bitmap	-9	72

(2003.02.27)

## 18. IDisplay インタフェース

### 18.1. IDISPLAY\_Update

### 18.2. IDISPLAY\_UpdateEx

ジャンル：Tips

対象：All

IDISPLAY\_Update/UpdateEx の動作が 2.0 と 2.1 で変更されました。

2.0 では、IDISPLAY\_SetDestination を行っても、IDISPLAY\_Update/UpdateEx は常にデバイスビットマップからディスプレイを更新していました。

2.1 では、IDISPLAY\_SetDestination でデバイスビットマップと異なるビットマップが宛先ビットマップに指定されている場合、IDISPLAY\_Update/UpdateEx はディスプレイを更新しなくなりました。

(2003.10.09)

## 19. IFileMgr インタフェース

### 19.1. IFILEMGR\_EnumNext

### 19.2. IFILEMGR\_EnumNextEx(1/2)

**ジャンル：バグ**

対象：2.1 以前のエミュレータ

SDK 上、これらの関数で正常なファイル作成日時を取得することが出来ません。実機上では正しい値を取得できます。

(2003.09.10)

### 19.3. IFILEMGR\_EnumNextEx(2/2)

**ジャンル：バグ**

対象：2.1 以前のエミュレータ及び実機

現在、本 API にバグがあり、AEEFileInfoEx.nClasses に  $2 * \text{sizeof}(\text{AEECLSID})$  以上の値を指定した場合、pClasses に正しく ClassID を取得できません。nClasses は必ず  $1 * \text{sizeof}(\text{AEECLSID})$  を指定してください。

なお、このバグは 3.0 で修正されています。また、IFILE\_GetInfoEx() ではこの問題は発生しません。

(2004.05.13)

### 19.4. IFILEMGR\_GetFileUseInfo

**ジャンル：Tips**

対象：3.1 以前の SDK 及び実機

IFILEMGR\_GetFileUseInfo() で取得できる AEEFileUseInfo.wFilesUsed 及び AEEFileUseInfo.dwSpaceUsed は、アプリ起動後、アプリにアクセス権のあるフォルダになんらかの変更が加えられたときにのみ値が変更されます。

「なんらかの変更」とは、

- ・ IFILE\_Write()/IFILE\_Seek()/IFILE\_Truncate() による各ファイルのサイズの変化
- ・ IFILEMGR\_OpenFile()/IFILEMGR\_Remove() によるファイルの作成/削除
- ・ IFILEMGR\_Mkdir()/IFILEMGR\_Rmdir() によるディレクトリの作成/削除

を指します。

(2005.09.06)

### 19.5. IFILEMGR\_GetFreeSpace

**ジャンル：Tips**

対象：エミュレータ 2.0.1.5

SDK 上、この関数は正常な数字を返しません。HDD の空き容量を返してしまうようです。

(2003.02.27)

## 19.6. BREW 3.0 以降におけるファイル名の扱いについて

ジャンル：Tips

対象：3.0 以降の BREW シミュレータ及び実機

BREW2.x 以前では、BREW はファイル名の大文字小文字を区別しませんでした。3.0 以降では大文字小文字を明確に区別するようになりました。

ただし、大文字小文字の区別を行うのは、ファイル名が“fs:”で始まる、3.0 形式でのファイル名に限られます。2.x 以前の形式の、“MyApp.bar” のようなファイル名でファイルにアクセスした場合、ファイル名は実際には全て小文字として扱われます。

例えば、実際のファイル名が “/brew/myapp/myapp.bar” の場合、“myapp.bar”, “MyApp.bar”, “fs:/~/myapp.bar” は OK ですが、“fs:/~/MyApp.bar” は NG です。また、例えば実際のファイル名が “/brew/myapp/MyApp.bar” の場合、“fs:/~/MyApp.bar” は OK ですが、“myapp.bar”, “MyApp.bar”, “fs:/~/myapp.bar” は NG です。

(2005.03.09)

## 20. IFont インタフェース

### 20.1. インスタンスの作成

ジャンル：Tips

対象：ALL

AEECLSID\_FONT を指定しても、IFONT インタフェースのインスタンスは ISHELL\_CreataeInstance では作成できません。

実際にサポートされているフォントに応じ、CLSID を指定して作成する必要があります。ただし、ディベロッパ側でこのクラスを直接コールする必要は通常ありません。

(2003.02.27)

### 20.2. IFONT\_GetFontInfo

### 20.3. IFONT\_GetInfo

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

API リファレンスには IFONT\_GetFontInfo という API について説明がありますが、正しい API 名は IFONT\_GetInfo です。

(2003.05.15)

補記

API リファレンス 2.1 (英語版) にて修正されました。日本語版は未修正です。

(2003.05.20)

## 20.4. IFONT\_MeasureText

**ジャンル: ドキュメントエラー**

対象: 2.0.1.5 API リファレンス (英語版)

この関数の第 3 引数に「-1」を渡すと特別な動作をする旨記述がありますが、これは誤りです。第 3 引数には必ず有効な数字を渡してください。(日本語版ドキュメントにはこの記述はありません。)

(2003.02.27)

## 21. IGetLine インタフェース

### 21.1. IGETLINE\_Advance

**ジャンル: Tips**

対象: 全ての SDK 及び実機

IGetLine はシーケンシャルにデータを読み込むことを前提としたインタフェースであり、IGETLINE\_Advance() が利用可能な範囲は限定的です。特に、 $2^{15}$  以上の値を指定すると、IGETLINE\_Advance() は内部的に失敗します。

(2005.04.12)

## 22. IGraphics インタフェース

### 22.1. IGRAPHICS\_StretchBlt(1/2)

**ジャンル: バグ**

対象: 3.0 以前の全ての SDK 及び実機

IGRAPHICS\_StretchBlt() の描画結果は、IGRAPHICS\_SetDestination() によるターゲットビットマップの指定に関わらず、デフォルトの IDisplay に出力されます。

(2004.11.15)

### 22.2. IGRAPHICS\_StretchBlt(2/2)

**ジャンル: バグ**

対象: 3.0 以前の全ての SDK 及び実機

IGRAPHICS\_StretchBlt() で指定する変形前及び変形後のビットマップの大きさ(縦\*横\*ビット深度/8) が 32767 を超えると、ヒープ残量に関わらず端末がクラッシュします。

(2004.12.13)

## 23. IHTMLViewer インタフェース

### 23.1. 画像の表示について

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

SDK2.0.1.1 より、ドキュメントには記載されておりませんが、HVP\_LOADALLIMAGE というプロパティが定義されました。これにより、<image> タグ内で、width/height の指定を行わなくても画像を正しく表示できるようになります。この指定を行わない場合、必ず width/height を指定する必要があります。

(2003.02.27)

### 23.2. ネットワーク経由の JPEG、BCI の表示について

ジャンル：バグ

対象：3.0.1 シミュレータ

[3.0.1 SDKのIIMAGE SetStream\(\)におけるバグ](#)により、リモートサーバにある JPEG/BCI イメージリソースを表示する場合、正常に表示できないケースがあります。JPEG、BCI の表示に関して上手く表示出来ない場合、ローカルにファイルを置いて正常に表示できることをご確認ください。

```
<img src = "http://localhost/img/test.jpg">
```

```
<img src = "file:///img/test.jpg">
```

(2004.07.05)

### 23.3. IHTMLVIEWER\_GetError (1/2)

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

対象となる HVEERROR が存在しないため、この API は現在意味を持ちません。

(2003.02.27)

### 23.4. IHTMLVIEWER\_GetError (2/2)

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

API リファレンスでは、この関数は以下のように定義されていますが、

```
int IHTMLVIEWER_GetError(IHtmlViewer * pIHTMLViewer);
```

正しくは以下の通りです。

```
int IHTMLVIEWER_GetError(IHtmlViewer * pIHTMLViewer, HVOpt * ppVal);
```

ppVal にエラーが格納され、戻り値としては、成功時には SUCCESS が、失敗時には EUNSUPPORTED が返ります。

ただし、IHTMLViewer\_GetError(1/2)の項で述べましたように現在この関数は動作しません。

(2003.02.28)

### 23.5. <form>エレメント中の<img>エレメント

#### ジャンル：バグ

対象：3.1 以前の全ての SDK 及び実機

現在、<form>エレメント中に<img>エレメントが存在した場合、送信されるデータに不要な&が生成される不具合があります。

(2005.04.12)

### 23.6. 単一画像の複数配置について

#### ジャンル：Tips

対象：3.1.2 SDK 及び実機

同一 HTML ソース内に、単一のソースを持つ画像を複数配置した場合、サイズについては最初の情報が全てに適用されます。

```
<html>
<br>
<br>
</html>
```

このような HTML ソースがあった場合、image1、image2 は共に 20x20 の大きさで描画されます。

(2005.06.15)

### 23.7. 描画完了前のキー入力について

#### ジャンル：Tips

対象：ALL

IHTMLViewer による描画時、画像データなどのロード・描画が完了する前にビューワがテキストボックスにキーイベントを渡して文字入力モードに移行した場合、画像データの描画完了時にテキストボックスからフォーカスが外れそれ以降キーイベントがテキストボックスに渡せなくなります。このため、文字入力モードから復帰できなくなります。

HTML の描画時は、描画が完了するまでキー入力を IHTMLViewer に渡さないようにしてください。

(2005.09.06)

### 23.8. 3.1 における HViewFocus.nState について

ジャンル：Tips

対象：3.1 の SDK 及び実機

PFNHVIEWNOTIFY に通知される HVN\_FOCUS イベントにおいて、フォーカスの当たったアイテムがラジオボタンもしくはチェックボックスの際、HViewFocus.nState にボタン/ボックスの状態が通知されるとありますが、3.1 ではこのフラグは常に FALSE となります。

(2006.03.08)

### 23.9. IHTMLVIEWER\_SetViewState()

ジャンル：Tips

対象：全ての SDK 及び実機

IHTMLVIEWER\_SetViewState()を行うと、ラジオボタンなどの選択状況はそのまま復元されますが、チェックボックスの選択状況は失われることがあります。これは、IHtmlViewer が、1 つの HTML ドキュメント中に同名のチェックボックスが複数存在することを想定していないためです。

全てのチェックボックスにユニークな name を設定することで、この問題を回避することが出来ます。

(2006.03.08)

## 24. Image インタフェース

### 24.1. JPEG イメージの取り扱い

ジャンル：Tips

対象：ALL

JPEG のデータの読み込みは非同期で処理されるため、下記のようなコードは失敗する恐れがあります。

```
pi=ISHELL_LoadResImage(pMe->a.m_pIShell, "image.bar", IMG_JPEG1);  
if(pi)  
    IIMAGE_Draw(pi);
```

より安全に処理するには、下記のように、IIMAGE\_Notify によってコールバックを登録し、イメージのロード完了時にコールバック内で IIMAGE\_Draw をコールしてください。

```
ISHELL_CreateInstance(pMe->a.m_pIShell, AEECLSID_JPEG, (void **)&pi);
```

```
if (pi)
{
    IIMAGE_Notify(pi, pfn, pUser); // pfn は PFNIMAGEINFO 関数へのポインタ、
                                // pUser はユーザコンテキストデータ

    IIMAGE_SetStream(pi, stream); // stream は画像ファイルもしくは
                                // 画像データへのポインタ

    // コールバック関数内で IIMAGE_Draw()
}
}
```

(2003.02.27)

## 24.2. PNG イメージの取り扱い

**ジャンル：Tips**

対象：ALL

PNG イメージのデコードは画像データ + 40KB ほどのヒープを瞬間的に使用します。ヒープ残量を確認してからイメージをロードするようにして下さい。

(2003.02.27)

## 24.3. PNG イメージの透過処理について

**ジャンル：Tips**

対象：ALL

2.1 から、PNG イメージに対しては AEE\_RO\_MASK を指定できなくなりました。以前は、tRNS チャンクに透過色の情報を持たないような PNG データに対して AEE\_RO\_MASK を指定した場合、カラーパレットの index0 の色を透過表示していましたが、2.1 以降ではこのような操作は行えません。

なお、tRNS チャンクに透過色の情報を持つ PNG データに対し、AEE\_RO\_COPY を指定した場合、透過色の指定は無視されます。この動作は 2.1、2.0 で共通です。

(2003.10.09)

3.1 では、2.0 以前と同様、透過色情報のない PNG データに対し AEE\_RO\_MASK を指定した場合は、インデックス 0 番の色を透過表示を行うよう変更されました。透過色情報を持つ PNG に対し、AEE\_RO\_COPY を指定した場合は、2.1 と同様透過されずに表示されます。

(2005.06.15)

## 24.4. BCI イメージの取り扱いについて

**ジャンル：Tips**

対象：ALL

BCI ファイルを IImage オブジェクトにセットした場合、IImage オブジェクト自体

を解放するまでソースの BCI ファイルを削除することは出来ません。

(2003.11.13)

## 24.5. 8bit グレースケール JPEG の表示について

**ジャンル：Tips**

対象：2.0.1.5 エミュレータ、2.1.1.5 エミュレータ、弊社ライブラリを使用する端末 BREW2.0 及び 2.1 において、弊社 JPEG ライブラリに不具合があるために、現在 8bit グレースケールの JPEG を表示することが出来ません。

なお、JPEG デコーダを独自に実装している一部の端末はこの影響を受けません。

(2004.05.13)

## 24.6. IIMAGE\_SetStream

**ジャンル：バグ**

対象：3.0.1 シミュレータ

JPEG、BCI の IIMAGE\_SetStream() に問題があり、File 以外のストリームをセットした場合に正常に動作しません。

JPEG の場合、4KB を超えるイメージの描画に必ず失敗します。

BCI の場合、PFNIMAGEINFO への Notify が返らないケースがあります。

このバグは 3.0.1 SDK に特有の問題であり、実機及び他のバージョンの SDK では発生しません。IHTMLViewer の [「ネットワーク経由の JPEG、BCI 表示について」](#) も併せてご参照ください。

(2004.07.05)

# 25. IMedia インタフェース

## 25.1. コールバックの不具合

**ジャンル：バグ**

対象：2.0.1.5 エミュレータ

SDK 上、IMEDIA\_SetMuteCtl や IMEDIA\_SetAudioDevice を行った場合、コールバック時に渡される MM\_CMD\_SETMEDIAPARM の値が MM\_PARM\_VOLUME で渡される不具合があります。

(2003.02.27)

## 25.2. サスペンド時のオブジェクトのリリースについて

**ジャンル：バグ**

対象：A5304T

SMAF 再生中、他のアプリを起動しようとする時 (一発起動キー or ISHELL\_StartApplet) 再生中のアプリケーションに対して EVT\_APP\_SUSPEND

がかかります。このとき、再生をストップして IMEDIA オブジェクトを解放するのが正しい動作ですが、SMAF 再生エンジンの不具合により、この操作（オブジェクトの解放）を行うと端末にリセットがかかることがあります。これに対処するには、以下のようにコーディングしてください。

基本的なシーケンスは以下の通りです。

- 1) EVT\_APP\_SUSPEND を受信。
- 2) イベントハンドラ内で、IMEDIA\_Stop をコール。
- 3) イベントハンドラでは TRUE を返してサスペンドする。
- 4) IMEDIA\_Stop のコールバックが呼ばれたら、このコールバック内で IMEDIA\_Release をコールして IMedia オブジェクトを解放する。

EVT\_APP\_SUSPEND で直ちに IMedia オブジェクトを解放するのではなく、IMEDIA\_Stop のコールバック関数内で解放を行うのがポイントです。

以下に、簡単なサンプルを示します。

```
// サスペンド時のイベントハンドラ
case EVT_APP_SUSPEND:
    if (pMe->pMedia != NULL)
    {
        pMe->bSuspend = TRUE;
        IMEDIA_Stop(pMe->pMedia);
    }

    return TRUE;

// IMEDIA_Stop のコールバック
MediaStopCB(...)
{
    if (pMe->bSuspend)
    {
        // サスペンド処理中なら、IMedia オブジェクトを解放
        IMEDIA_Release(pMe->pMedia);
        pMe->bSuspend = FALSE;
    }
}
```

```
return;  
}
```

(2003.02.27)

### 25.3. IMedia オブジェクトの解放処理について

**ジャンル：Tips**

対象：2.1 以降の実機及び SDK

2.1 において、IMedia の動作に以下の変更が加えられました。

1. IMEDIA\_RegisterNotify で登録したコールバックは、IMEDIA\_Release 時に自動的に解放されるようになりました。
2. 上記コールバックが呼ばれている状態では、IMEDIA\_Release が行えなくなりました。

これらの変更に伴い、「サスペンド時のオブジェクトのリリースについて」で述べたような、「IMEDIA\_Stop に対するコールバックを待った後に IMEDIA\_Release() をコールする」解放処理は不要となるばかりでなく、そのような処理自体行えなくなりました。

2.1 端末においては、IMedia を解放する必要がある場合は、IMEDIA\_Stop コール後、即 IMEDIA\_Release をコールしてください。

(2003.10.09)

### 25.4. MMD\_ISOURCE

**ジャンル：Tips**

対象：2.1 以前の SDK

MMD\_ISOURCE は SDK でサポートされません。実機でのサポートは実装依存になります。

(2003.09.10)

### 25.5. 各 API、パラメータのサポートについて

**ジャンル：Tips**

対象：ALL

IMEDIA インタフェースは抽象インタフェースであり、再生するメディアのフォーマット及び実装により、実際にサポートされている API やパラメータは異なります。現在サポートしている API/パラメータについての詳細なリストはありません。

(2003.02.28)

## 25.6. SetMediaParm

## 25.7. GetMediaParm

**ジャンル：ドキュメントエラー**

対象：2.0.1.5 API リファレンス

IMEDIA\_SetMediaParm/GetMediaParm の各パラメータの説明文中、フォーマットが崩れて意味不明な箇所があります。正しくは以下の通りです。

nParamID	操作	説明	p1	p2
MM_PARM_AUDIO_PATH	取得 / 設定	メディアパス	MM_YPATH_XXX	(p1=MM_YPATH_LOCAL_RINGERの時のみ) SilentTimerMS。繰り返し再生時のトラック間の無音時間を定める(ミリ秒単位)。0 に設定された場合、繰り返しを行わない。
MM_PARM_TICK_TIME	取得 / 設定	MM_TICK_UPDATEの周期	デフォルトは 1000 (ミリ秒単位)。0 にセットすると、MM_TICK_UPDATE イベントは発生しない。	

(2003.02.27)

## 25.8. IMEDIA\_GetClassID

**ジャンル：バグ**

対象：2.1 以前の SDK 及び A5304T、A5306ST、A1304T

現在、MM\_STATE\_IDLE の時にこの API をコールした場合、API の戻り値としては SUCCESS が返るにも関わらず、正しい ClassID が取得できない ( BREW が第 2 引数に渡されたポインタの中身に何の値も設定しない ) というバグがあります。

(2003.10.09)

## 25.9. IMEDIA\_SetTickTime

**ジャンル：バグ**

対象：2.0.1.5 エミュレータ

この API は、SDK 上で正常に動作しません。

(2003.02.28)

## 25.10. MM\_PARM\_MUTE

**ジャンル：ドキュメントエラー**

対象：2.0.1.5 API リファレンス

IMEDIA\_GetMediaParm()/SetMediaParm() では、このパラメータは「0 でミュート、1 でアンミュート」とありますが、これは誤りです。IMEDIA\_SetMuteCtl() の記述が正しく、「1 でミュート、0 でアンミュート」となります。

(2003.11.13)

## 25.11. MM\_STATUS\_DONE

## 25.12. MM\_STATUS\_ABORT

**ジャンル：バグ**

対象：エミュレータ 2.0.1.5 及び 2.0 端末

Phrase データ再生時、IMEDIA\_Stop()をコールして MM\_STATUS\_ABORT の通知を受け取った後、IMEDIA\_Play()をコールして再度再生を行った場合、再生終了時に MM\_STATUS\_DONE の代わりに MM\_STATUS\_ABORT が通知されます。

(2004.09.10)

## 26. IMemAStream インタフェース

### 26.1. IMEMASTREAM\_Cancel

**ジャンル：ドキュメントバグ**

対象：2.0.1.5 API リファレンス

IMEMASTREAM\_Cancel()は API リファレンスで下記のように定義されていますが、

```
void IMEMASTREAM_Cancel(IMemAStream * pIMemAStream, PFNNOTIFY pfn, void * pUser)
```

正しい定義は下記の通りです。

```
void IMEMASTREAM_Cancel(IMemAStream * pIMemAStream)
```

(2003.02.27)

補記

API リファレンス 2.1 (英語版) にて修正されました。日本語版は未修正です。

(2003.05.20)

## 27. IMenuCtl インタフェース

### 27.1. メニュープロパティ

**ジャンル：Tips/ドキュメントエラー**

対象：API リファレンス 2.0.1.5

メニュープロパティについては複数箇所に記述があり、その記述内容に若干の異同があります。

「データ構造(Data Structure)」の AEE メニュープロパティ(AEE Menu Properties) のページに記載されているのが正しい情報です。

(2003.05.20)

## 27.2. IMENUCTL\_AddItem(1/2)

## 27.3. IMENUCTL\_AddItemEx(1/2)

**ジャンル** : Tips

**対象** : ALL

リソースファイルから項目テキストを読み込ませる場合、セットできる文字は NULL 終端を含め 128 文字までです。これ以上の文字をセットしようとすると、EFAILED が返り、項目は追加されません。

(2003.07.11)

## 27.4. IMENUCTL\_AddItem(1/2)

## 27.5. IMENUCTL\_AddItemEx(1/2)

**ジャンル** : バグ

**対象** : SDK 2.1

メニュー項目の文字列として先頭に「入」(0x93fc)などの 0x93 を含む文字を設定すると、文字列が右寄せされる不具合があります。なお、実機ではこの問題は発生しません。

(2005.12.09)

## 27.6. IMENUCTL\_GetItemRect

**ジャンル** : バグ

**対象** : 3.1 以前の全ての SDK 及び実機

リストコントロール(AEECLSID\_LISTCTL)に対し IMENUCTL\_GetItemRect() を呼ぶ場合、正常な値を返さない不具合があります。

(2005.02.08)

## 27.7. IMENUCTL\_SetStyle

**ジャンル** : Tips

**対象** : ALL

リストコントロール(AEECLSID\_LISTCTL)に対しては、Style を変更することが出来ません。

(2003.05.20)

## 27.8. IMENUCTL\_SetColors(1/2)

**ジャンル** : Tips

**対象** : 日本の端末

ソフトキーコントロール(AEECLSID\_SOFTKEYCTL)に対しては、色の変更を行うことは出来ません。

(2003.05.20)

## 27.9. IMENUCTL\_SetColors(2/2)

ジャンル：Tips

対象：ALL

リストビューコントロール以外では、タイトルの文字色と背景色を変更することは出来ません。

(2003.06.24)

## 28. IModule インタフェース

### 28.1. IMODULE\_FreeResources

ジャンル：Tips

対象：ALL

この API は、通常ディベロッパの方が直接コールする必要はありません。  
AEEModGen.c および AEEAppGen.c を用いることで、この API を使わずに安全にリソースを解放できます。

(2003.03.03)

## 29. IModuleSupport インタフェース

### 29.1. インタフェースのサポート

ジャンル：Tips

対象：3.0/3.1 の SDK 及び実機

3.0、3.1 の SDK には IModuleSupport というクラスが定義されていますが、現在のクラスは実装されておられません。

(2005.02.08)

## 30. INetMgr インタフェース

### 30.1. INETMGR\_GetHostByName

ジャンル：Tips

対象：All

INETMGR\_GetHostByName()のプロセスが進行中、INETMGR\_Release()を行っても、GetHostByName のプロセスはキャンセルされません。別途、CALLBACK\_Cancel()をコールして明示的にキャンセルする必要があります。

(2004.11.11)

### 30.2. INETMGR\_OpenSocket

ジャンル：ドキュメントエラー

対象：API リファレンス 2.0.1.5

API リファレンスには、ソケット数が上限に達している時、INETMGR\_OpenSocket が NULL を返すとありますが、これは誤りです。BREW は、ソケットを Connect あるいは Bind あるいは SendTo する際にはじめて「実際の」ソケットを作成します。従って、特権レベルの設定に誤りがなければ、INETMGR\_OpenSocket は常に ISocket オブジェクトを返します。

ソケット数が上限に達した場合、ISCOCKET\_Bind、\_Connect、\_SendTo に対して AEE\_NET\_EMFILE が返ります。

(2003.03.24)

補記

API リファレンス 2.1 (英語版) で修正されました。日本語版は未修正です。

(2003.05.20)

## 31. IPeek インタフェース

### 31.1. IPEEK\_Peek

ジャンル：ドキュメントエラー

対象：API リファレンス 2.0.1.5

リファレンスには IPEEF\_Peek が IPEEK\_END を返すとありますが、現在この API はストリームの終端に達しても IPEEK\_UNK を返します。ストリームの終端に達した状態で IPEEK\_Advance をコールした場合には IPEEK\_END が返ります。

(2003.09.10)

### 31.2. IPEEK\_Readable

ジャンル：ドキュメントエラー

対象：API リファレンス 2.0.1.5

IPEEK\_Readable の API リファレンスにおけるプロトタイプに誤りがあります。正しくは、下記の通りです。

```
void IPEEK_Readable(IPeek * pIPeek, AEECallback *pcb)
```

(2003.02.27)

補記

API リファレンス 2.1 (英語版) にて修正されました。日本語版は未修正です。

(2003.05.20)

## 32. IPort インタフェース

### 32.1. シリアルポートの利用

ジャンル：Tips

対象：SDK3.0 及び 3.0 以降の実機

シリアルポートへの接続は AEECLSID\_SERIAL を用いて IPort クラスを作成する必要があります。

```
IPort * pIPort;  
ret = ISHELL_CreateInstance(pIShell, AEECLSID_SERIAL, (void **)&pIPort);
```

また、AEECLSID\_SERIAL は「保護されたクラス」となっていますので、MIF の「依存ファイル」に AEECLSID\_SERIAL ( 0x01001042 ) を指定する必要があります。

(2004.09.10)

## 32.2. IPORT\_Open()

ジャンル：Tips

対象：SDK3.0 及び 3.0 以降の実機

一般的に、IPort の利用法は以下のようになります。

1. IPORT\_Open(pPort, PORT\_NAME) でポート名 PORT\_NAME を開く
2. 上記コールはほぼ確実に AEEPOR\_WAIT エラーとなるので、  
IPORT\_Writeable() でコールバックを登録する
3. Writeable()のコールバック内で以下のようにコールする  
IPORT\_Open(pPort, NULL);  
これで、1.で指定したポート名 PORT\_NAME が Open される  
(ここで再度\_WAIT エラーとなるケースもある)
4. 以降、pPort への読み書きが可能となる

(2004.09.10)

## 33. IPosDet インタフェース

### 33.1. IPOSDET\_GetGPSInfo

ジャンル：バグ

対象：3.1.2 SDK

アプリから IPOSDET\_GetGPSInfo()を利用した場合、シミュレータを終了する際にメモリアクセス違反のエラーが発生します。実機ではこの問題は発生しません。

(2005.12.09)

## 34. IRSA インタフェース

### 34.1. 大きなデータの暗号化/復号化について

ジャンル：Tips

対象：ALL

IRSA インタフェースは CPU を大量に消費し、時間のかかるオペレーションです。そのため、modulus を超えるような大きいデータの暗号化/復号化への使用はおすすめできません。

(2003.02.27)

### 34.2. IRSA\_ModExp (1/2)

ジャンル：バグ

対象：SDK2.0.1.5

IRSA\_ModExp のヘッダファイルに誤りがあります。  
正しくは、下記の通りです。

```
#define IRSA_ModExp(p, pbIn, cbIn, ppbOut, pcbOut, pr, pcb)
GET_PVTBL(p, IRSA) -> ModExp(p, pbIn, cbIn, ppbOut, pcbOut, pr, pcb)
```

お手数ですがヘッダを修正してご使用ください。

補記

このバグは 2.1.0 にて修正されています。

(2003.02.27)

### 34.3. IRSA\_ModExp (2/2)

ジャンル：Tips

対象：2.0 以前の全ての SDK 及び実機

2.0 以前の SDK 及び実機では、この関数はブロッキングモードで動作します。つまり、IRSA\_RSA と同じ動作をします。

(2005.09.06)

## 35. IShell インタフェース

### 35.1. ISHELL\_CloseApplet

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

API リファレンス中、本 API は、「現在アクティブなアプレットをクローズする」とありますが、正しくはこの API をコールしたアプレット（自アプレット）をクローズします。

(2003.06.23)

### 35.2. ISHELL\_DetectType

ジャンル：バグ

対象：A5304T、エミュレータ 2.0.1.5

現在、SMAF データを正常に識別できません。アプリケーション側で独自に識別する必要があります。

(2003.02.27)

### 35.3. ISHELL\_GetAppPrefs

### 35.4. ISHELL\_GetPrefs

**ジャンル：バグ**

対象：2.1 以前の SDK 及び実機

現在、これらの関数の引数の wVer に、間違ったバージョン番号を指定すると、設定されているデータが失われてしまうというバグがあります。

(2003.02.27)

### 35.5. ISHELL\_GetAppAuthor

### 35.6. ISHELL\_GetAppCopyright

### 35.7. ISHELL\_GetAppVersion

**ジャンル：ドキュメントエラー**

対象：2.0.1.5 API リファレンス

これらの関数は、取得した文字列の長さをバイト数で返すとリファレンスにありますが、実際には文字列の長さを文字数 (AECHAR 文字列) で返します。

(2003.02.27)

### 35.8. ISHELL\_GetDeviceInfo

**ジャンル：バグ**

対象：3.1.2 SDK

3.1.2 SDK では、ISHELL\_GetDeviceInfo() で dwPlatformID が取得できないバグがあります。この問題は実機では発生しません。

(2005.09.06)

### 35.9. ISHELL\_GetTimerExpiration

**ジャンル：Tips**

対象：All

この関数では、ISHELL\_SetTimerEx() を使って登録したタイマーの残り時間を取得することはできません。

(2005.09.06)

### 35.10. ISHELL\_GetResSize

### 35.11. ISHELL\_LoadResData

### 35.12. ISHELL\_LoadResDataEx

**ジャンル：Tips**

対象：2.1 以前の SDK 及び実機

上記 3 つの API において、現状 `RESTYPE_BINARY` は使用できません。なお、`RESTYPE_IMAGE` には任意のデータを格納できますので、2.0 においてはこちらをご利用ください。`RESTYPE_IMAGE` の仕様は将来に亘ってサポートされます。詳しくは、[「リソースファイルにサウンドデータを格納するには」](#) をご参照ください。

(2003.02.27)

### 35.13. ISHELL\_LoadResObject

**ジャンル :** Tips

**対象 :** 2.1 以前の SDK 及び実機

`HTYPE_SOUND` は現バージョンではサポートされません。( `.bar` ファイルが `SOUND` リソースの格納に対応していない為です。) 詳しくは、[「リソースファイルにサウンドデータを格納するには」](#) をご覧下さい。

(2003.02.27)

### 35.14. ISHELL\_OnInit

### 35.15. ISHELL\_OnExit

**ジャンル :** Tips

**対象 :** ALL

上記二つの API は、ダウンローダブルなアプリケーションでは意味を持ちません。

(2003.02.27)

### 35.16. ISHELL\_OnAppClose

### 35.17. ISHELL\_OnModUnload

**ジャンル :** Tips

**対象 :** ALL

上記二つの API は、`AEEApplet_New` 時に登録する `PFNFREEAPPDATA` 関数での処理で代用できますので、使う必要はありません。( `ModUnload` とは、モジュールがヒープから解放されるタイミングを指します。ROM から削除されるタイミングのことではありません。)

(2003.02.27)

### 35.18. ISHELL\_RegisterSystemCallback

**ジャンル :** Tips

**対象 :** ALL

`ISHELL_On*` 系の API で内部的にコールしている関数です。`ISHELL_On*` API をコールする方がタイプ量も少なく済みますので、そちらをご利用ください。

(2003.02.27)

## 35.19. ISHELL\_LoadImage

## 35.20. ISHELL\_LoadResImage

### ジャンル：バグ

対象：2.1 以前の SDK 及び実機

上記二つの API にて、データのロード失敗時には NULL が返ることになっていますが、あるメモリ状況下でデータのロードに失敗しても NULL が返らない場合があります。

状況としては、「IImage オブジェクト（数十バイト）自体」は生成できても、その中の「画像自体」がヒープにロードできないような状況において、これらの API は画像へのポインタをもたない IImage オブジェクトを返します。

プログラム側は、これらの API が返すポインタが本当に有効なものかどうか再度のチェックを行ってください。チェックするには、IIMAGE\_GetInfo を行い、メンバ変数の cx が 0 でないことを確認してください。

```
pMe->pii=ISHELL_LoadResImage(pMe->a.m_pIShell, "wupdate.png", 0);

if (pMe->pii)
{
    IIMAGE_GetInfo(pMe->pii, &paii);
    if (paii.cx==0)
    {
        DBGPRINTF("Failed to load image...!");
        IIMAGE_Release(pMe->pii);
        pMe->pii=NULL;
        return FALSE;
    }
    else
    {
        IIMAGE_Draw(pMe->pii, 0,0);
        return TRUE;
    }
}

return FALSE;
```

補記

PNG、JPEG、BCI の画像をロードする際、上記のチェックを行ってもエラーが検出できないことがあります。

例えば、PNG のイメージのロードには、PNG イメージ自体のファイルサイズ (画像に依存) +PNG デコーダのワークエリア(40KB)+デコードされたイメージの DIB (画像サイズに依存) が必要となりますが、「PNG イメージのファイルサイズ+PNG デコーダのワークエリア」のみが確保できてもデコードされたイメージの DIB がメモリに展開できなかった場合、AEEImageInfo.cx には有効な値が設定されるにも関わらず、画像のロード自体には失敗しています。

現状、PNG、JPEG、BCI の画像を展開する際には、ヒープが十分残っていることを確認してから ISHELL\_LoadImage するようにして下さい。

(2003.02.27)

### 35.21. ISHELL\_LoadSound

### 35.22. ISHELL\_LoadResSound

**ジャンル：バグ**

対象：2.1 以前の SDK 及び実機

上記二つの API にて、データのロード失敗時には NULL が返ることになっていますが、あるメモリ状況下でデータのロードに失敗しても NULL が返らない場合があります。

状況としては、「ISoundPlayer オブジェクト (数十バイト) 自体」は生成できても、その中の「音データ自体」がヒープにロードできないような状況において、これらの API は音データへのポインタをもたない ISoundPlayer オブジェクトを返します。

プログラム側は、これらの API が返すポインタが本当に有効なものかどうか再度のチェックを行ってください。チェックするには、ISOUNDPLAYER\_GetInfo を行い、メンバ変数の dwSize が 0 でないことを確認してください。

```
pMe->psi=ISHELL_LoadResSound(pMe->a.m_pIShell,
                              "wupdate.qcp", 0);

if (pMe->pis)
{
    ISOUNDPLAYER_GetInfo(pMe->pis, &pasi);
    if (pasi.dwSize==0)
    {
        DBGPRINTF("Failed to load sound...!");
    }
}
```

```

        ISOUNDPLAYER_Release(pMe->pis);
        pMe->pis=NULL;
        return FALSE;
    }
    else
    {
        ISOUNDPLAYER_Play(pMe->pis);
        return TRUE;
    }
}
return FALSE;

```

(2003.02.27)

### 35.23. ISHELL\_SetTimer

**ジャンル：バグ**

対象：2.1.1 以前の全ての SDK 及び実機

ISHELL\_SetTimer()で指定可能なタイマー値は int32 の値(単位は msec)を指定できますが、上限値に極めて近い値をセットした場合、直ちにコールバックが呼ばれてしまうケースがあります。

タイマー値としては、 $2^{30}$ msec 以下の値を指定してください。

(2004.11.11)

## 36. ISMS インタフェース

### 36.1. ISMS\_SendMsg

**ジャンル：ドキュメントエラー**

対象：3.1.2 API リファレンス

ISMS\_SendMsg()は、第4引数 \*pnRv に SUCCESS、EFAILED などのエラーコードを非同期で返すとありますが、これは誤りです。\*pnRv に返される値は、上位16bit が AEESMS\_ERRORTYPE\_\* を、下位16bit が AEESMS\_ERROR\_\* を示します。詳しくは、AEESMS.h を参照してください。

(2005.04.12)

## 37. ISocket インタフェース

### 37.1. ISOCKET\_Connect

**ジャンル：仕様変更**

対象：2.0 - 2.1 の SDK および実機

BREW 2.0 以前では、UDP ソケットに対し ISOCKET\_Connect()をコールしてもエラーとはなりませんでしたが、2.1 以降ではエラーとなります。

(2004.02.12)

## 37.2. ISOCKET\_Writeable

**ジャンル：Tips**

対象：2.0, 2.1 の SDK および実機

UDP のソケットに対し、下位レイヤーの PPP 接続がドーマント状態になっている場合に ISOCKET\_Writeable()をコールすると、PPP 接続が再確立されるまでの間、Writeable のコールバックが連続して呼ばれてしまい動作が不安定になります。

(2004.04.02)

## 38. ISound インタフェース

### 38.1. ISOUND\_Get

**ジャンル：ドキュメントエラー**

対象：API リファレンス 2.0.1.5

API リファレンス中には下記のように定義されていますが、

```
void ISOUND_Get(ISound * pISound, const AEESoundInfo * pSoundInfo)
```

正しくは以下の通りです。

```
void ISOUND_Get(ISound * pISound, AEESoundInfo * pSoundInfo)
```

(2003.06.23)

### 38.2. ISOUND\_PlayToneList

**ジャンル：Tips**

対象：ALL

本 API に渡す PlayToneData は、再生終了までアプリケーションの側で保持しておく必要があります。

(2004.04.02)

## 39. ISoundPlayer インタフェース

### 39.1. ISOUNDPLAYER\_SetTune

**ジャンル：ドキュメントエラー**

対象：API リファレンス 2.0.1.5

ドキュメントのプロトタイプは下記のようにありますが、

```
void ISOUNDPLAYER_SetTune(ISoundPlayer * pISoundPlayer, uint8 nStep)
```

正しくは下記の通りです。

```
void ISOUNDPLAYER_SetTune(ISoundPlayer * pISoundPlayer, int8 nStep)
```

(2003.03.24)

補記

API リファレンス 2.1 英語版 にて修正されました。日本語版は未修正です。

(2003.05.20)

## 39.2. ISOUNDPLAYER\_SetVolume

ジャンル：ドキュメントエラー

対象：API リファレンス 2.0.1.5

BREW API リファレンスの ISOUNDPLAYER\_SetVolume() の「副作用」の説明では、コールバックに AEE\_SOUNDPLAYER\_SOUND\_CB が渡されるとありますが、正しくは AEE\_SOUNDPLAYER\_STATUS\_CB が渡されます。

(2003.10.09)

## 40. ISprite インタフェース

### 40.1. DIB のサポート

ジャンル：Tips

対象：ALL

ISprite に渡す IBitmap オブジェクトは、DDB である必要があります。IDIB を渡した場合、Transform 処理は行われません。

(2003.07.11)

### 40.2. TILE\_SIZE\_8X8

ジャンル：バグ

対象：2.1.2 以前の全ての SDK 及び実機

現在、ISPRITE\_SetTileBuffer() で 8x8 のタイルをセットした後、ISPRITE\_Release() をコールすると、タイルバッファがリリースされずメモリリークを起こします。これを防ぐには、アプリケーションの側で、8x8 のタイルバッファに NULL を設定してタイルバッファを明示的に解放してください。

```
ISPRITE_SetTileBuffer(iSprite, TILE_SIZE_8X8, NULL);  
ISPRITE_Release(iSprite);
```

(2004/07/05)

## 41. ISSL インタフェース

### 41.1. ISSL\_Negotiate

### 41.2. ISSL\_Renegotiate

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

ドキュメントには以下の様にプロトタイプが記載されていますが、

```
void ISSL_Negotiate
(
    ISSL *pISSL,
    SSLInfo *pinfo,
    IX509Chain **pCerts,
    AEECallback *pcb, ...
)
```

正しくは下記の通りです。( IWEB\_GetResponse と同じ形式です。)

```
void ISSL_Negotiate
(
    ISSL *pISSL,
    (
        ISSL * pISSL,
        SSLInfo *pinfo,
        IX509Chain **pCerts,
        AEECallback *pcb, ...
    )
)
```

(2003.09.11)

### 41.3. AEE\_NET\_WOULDBLOCK 後の再送信処理について

**ジャンル : Tips**

対象 : 3.0 以前の全ての SDK 及び実機

現在、ISSL のライブラリの仕様として、SSL 通信中に送信元バッファのアドレスが移動すると SSL 処理のエラーとなる制限があります。ISSL\_SetSocket() を用いてソケットベースの SSL 通信を行う場合、ISOCKET\_Write() が AEE\_NET\_WOULDBLOCK を返した場合も、送信が成功するまで、送信元のバッファは移動しないようにしてください。

(2004.12.13)

## 42. IStatic インタフェース

### 42.1. ISTATIC\_SetActive

**ジャンル : バグ**

対象 : 2.1 以前の全ての SDK 及び実機

現在、ISTATIC\_SetActive() によってコントロールをインアクティブにしても、自動

スクロールが止まらないという不具合があります。

(2003.11.13)

## 42.2. ISTATIC\_SetFont

**ジャンル：バグ**

対象：2.1 以前の全ての SDK 及び実機

現在、ISTATIC\_SetFont()によりタイトル部のフォントを変更しても、タイトル部の描画領域の大きさが再計算されず、見た目が崩れる不具合があります。

ISTATIC\_SetText()、ISTATIC\_SetTitle()の中でフォントを指定した場合にはこの問題は起こりません。

(2003.11.13)

## 42.3. ISTATIC\_SetProperties

**ジャンル：ドキュメントエラー**

対象：2.0.1.5API リファレンス

ST\_NOSCROLL をセットした場合、「現行バージョンでは自動スクロールを OFF にするだけ」と記述がありますが、これは誤りです。BREW1.1 以上のバージョンでは、ST\_NOSCROLL を設定した場合、自動スクロールは無効化されますが、AVK\_UP 及び AVK\_DOWN のキーイベントを渡すことで上下へスクロールします。

(2003.09.10)

## 43. ITopVisibleCtl インタフェース

### 43.1. ITOPVISIBLECTL\_GetArgs()/SetArgs()

**ジャンル：ドキュメントエラー**

対象：3.1.2 API リファレンス

これらの関数で、第一引数が IResourceCtl \* となっていますが、正しくは ITopVisibleCtl \* です。

(2005.03.09)

## 44. ITransform インタフェース

### 44.1. インスタンスの作成

**ジャンル：Tips**

対象：ALL

ITransform オブジェクトは、IBitmap のオブジェクトから QueryInterface して下さい。

(2003.02.27)

## 44.2. DIB のサポート

ジャンル：Tips

対象：ALL

ITransform のサポートは実装依存になります。現在、SDK2.0.1.5 上では、DIB においては transform オペレーションをサポートしていません。DDB の IBitmap オブジェクトのみ、transform に対応しております。実機上で何がサポートされるかは実装に依存します。

(2003.02.27)

## 45. IUnzipAStream インタフェース

### 45.1. IUNZIPASTREAM\_SetStream

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

API リファレンス中、この関数のプロトタイプは下記のように定義されていますが、

```
void IUNZIPASTREAM_SetStream
(
    IUnzipAStream * pIUnzipAStream,
    IUnzipAStream * pInIUnzipAStream
)
```

正しくは以下の通りです。

```
void IUNZIPASTREAM_SetStream
(
    IUnzipAStream * pIUnzipAStream,
    IStream * pInIStream
)
```

なお、このエラーは 3.0 API リファレンスで修正されました。

(2004.04.02)

### 45.2. IUnzipAStream のメモリ使用量について

ジャンル：Tips

対象：All

IUnzipAStream は、実際の ZIP ストリーム読み出し時、32KB 強のメモリを必要とします。これは、CreateInstance 時には確保されず、IUNZIPASTREAM\_SetStream() 後にはじめて確保されます。

ここで内部的にメモリの確保に失敗した場合、その失敗自体はアプリに通知されず、以降全ての読み出しは失敗し 0 を返します。

(2004.11.11)

## 46. IWeb インタフェース

### 46.1. WebOpt について

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

WebOpt について、現在 API リファレンスにまともな記述がありません。

[Appendix A. WebOpt一覧](#) をご参照ください。

(2003.02.27)

### 46.2. Connection: ヘッダについて

ジャンル：Tips

対象：ALL

WEBOPT\_HEADER にて、Connection: Keep-Alive のヘッダを独自に記載しないで下さい。BREW の実行環境が、自動的にヘッダを付加します。

これを禁じるには、WEBOPT\_FLAGS に WEBREQUEST\_NOKEEPALIVE を設定してください。この場合は Connection: ヘッダは付加されませんので、必要な場合は能動的に Connection: Close ヘッダを WEBOPT\_HEADER に追加してください。

(2004.04.02)

### 46.3. 接続先 URL について

ジャンル：Tips

対象：ALL

IWEB\_GetResponse()などで指定する URL は、IWeb のオブジェクト内部ではコピーされません。コールバックにレスポンスが返ってくるまでの間、URL のデータはアプリケーションの側で保持しておく必要があります。

(2004.04.02)

### 46.4. WEBOPT\_CONNECTTIMEOUT

ジャンル：ドキュメントエラー

対象：API リファレンス 2.0.1.5

WEBOPT\_CONNECTTIMEOUT の値に-1 をセットした場合、タイムアウト値が無限になるという説明がありますが、これは誤りです。実際には、ここに-1 をセットした場合はタイムアウト値 0 が設定され、IWEB\_GetResponse()/GetResponseV() は確実に失敗します。

また、WEBOPT\_CONNECTTIMEOUT は値として uint32 を取りますが、内部で

は int32 として扱われます。

(2004.11.11)

## 46.5. 大きなデータの POST について

**ジャンル：バグ**

対象：3.1 以前の全ての SDK 及び実機

IWeb を介して POST でデータを送信する場合、body のサイズが 65536 バイト以上になると端末及び SDK がフリーズするケースがあります。

(2004.12.13)

## 46.6. WEBOPT\_PROXYSPEC 利用時の HOST:ヘッダについて

**ジャンル：バグ**

対象：3.1 以前の全ての SDK 及び 2.1 以前の実機

WEBOPT\_PROXYSPEC を利用して HTTP サーバに接続する場合、リクエストの Host:ヘッダにはターゲットのサーバのホスト名が設定されるべきですが、IWeb はプロキシサーバのホスト名を設定してしまう不具合があります。

(2005.06.15)

## 46.7. HTTPS 接続におけるプロキシの利用について

**ジャンル：仕様**

対象：3.1 以前の全ての SDK 及び実機

現在、IWeb は CONNECT メソッドに対応しておりません。このため、SSL 接続におけるトンネリングを処理することができず、プロキシ経由で HTTPS 接続を行うことはできません。

(2005.06.15)

# 47. IX509Chain インタフェース

## 47.1. IX509CHAIN\_Verify

**ジャンル：ドキュメントエラー**

対象：2.0.1.5 API リファレンス

リファレンスには下記の様にプロトタイプが記載されていますが、

```
void IX509Chain_Verify
(
    IX509Chain *pMe,
    509TrustResult *pTrustResult,
    AEECallback *pVerifyCB,
    ...
)
```

```
);
```

正しくは以下の通りです。( IWEB\_GetResponse と同じ形です。)

```
void IX509Chain_Verify
(
    IX509Chain *pix,
    (
        IX509Chain * pix,
        509TrustResult *pTrustResult,
        AEECallback *pVerifyCB,
        ...
    )
);
```

(2003.09.11)

## 48. ヘルパー関数

### 48.1. GETTIMEMS

### 48.2. GETUPTIMEMS

ジャンル：Tips

対象：全ての実機

これら 2 つの API は非常によく似た機能を提供しますが、以下の点で異なります。

- GETUPTIMEMS は「端末起動時からの経過時間」を、GETTIMEMS は、「当日 0:00 から現在時刻までの経過時間」を返します。
- GETUPTIMEMS は端末内部のクロックを元に計算しますが、GETTIMEMS は CDMA 基地局から通知される時刻を元に計算します。このため、GETTIMEMS の返す値は、通信する基地局が変わるタイミングで大きく変化する可能性があります。アプリケーション内部で経過時間を測るには、GETUPTIMEMS の使用を推奨します。

(2004.09.10)

### 48.3. STREXPAND

### 48.4. STRTOWSTR

### 48.5. WSTRCOMPRESS

### 48.6. WSTRTOSTR

ジャンル：Tips

対象：ALL

[「日本語の扱いについて」](#)をご参照ください。

(2003.05.01)

## 48.7. SNPRINTF

## 48.8. VSNPRINTF

ジャンル：ドキュメントエラー

対象：API リファレンス 2.0

これらの関数は、

1. フォーマット指定子に%fなど浮動小数点に関するものが含まれた場合にはエラーとして「-1」を、
2. バッファとしてNULLが指定された場合には、指定された文字列を出力する為に必要な文字数を、
3. バッファが有効であり、かつ、フォーマット指定子に無効なものが含まれなかった場合、実際に出力した文字数（終端のNULL文字を含む）を返します。

(2003.12.11)

## 48.9. UTF8TOWSTR

## 48.10. WSTRTOUTF8

ジャンル：Tips

対象：ALL

これらのAPIは、UTF8とUTF16の変換を行うAPIです。日本語端末上であっても、UTF8とS-JISの変換を行うAPIではありません。

(2004.04.02)

# 49. データ構造

## 49.1. EVT\_APP\_START

ジャンル：ドキュメントエラー

対象：2.0.1.5 API リファレンス

このイベントのdwParamには、文字列へのポインタではなく、AEEAppStart構造体へのポインタが渡されます。

(2003.12.11)

# 50. エミュレータ

## 50.1. ファイル名の長さについて

ジャンル：Tips

対象：SDK

BREWでは、絶対パス名を含むファイル名の長さに制限があります。これは64文

字と規定されています。

実機上では、BREW アプリケーションは「brew/」ディレクトリに置かれます(\*1)。このため、アプリケーションが使用することのできるファイル名は 59 文字になります。

SDK 上では、これを上手くエミュレートできず、39 文字が限界となります。

\*1 端末により、どこに置かれるかは異なることがあります。その場合、ファイル名の制限も変わります。

(2003.02.28)

## 50.2. エミュレータで表示できない文字について

**ジャンル：バグ**

**対象：エミュレータ**

BREW エミュレータでは、下記 S-JIS 文字を表示することは出来ません。

0xFA40 ~ 0xFC4B

(2003.02.28)

## 50.3. エミュレータの「アプレットディレクトリ」について

**ジャンル：Tips**

**対象：ALL**

BREW エミュレータ自体は日本語のパスを認識しません。このため、エミュレータの「アプレットディレクトリ」の絶対パス名に、日本語が含まれているとアプレットを見つけられなくなり、エミュレータとして機能しなくなります。

(2003.11.13)

# 51. MIF エディタ

## 51.1. ローカル BID の生成について

**ジャンル：Tips**

**対象：ALL**

テスト用に、ローカルの BID を生成する場合は、BREW 内部のクラスと競合しないよう、ClassID に 0x02000000 以上の値を使用してください。

(2004.05.13)

## 51.2. SDK 3.1.2 における MIF エディタについて

**ジャンル：Tips**

**対象：3.1.2 以降の SDK**

MIF エディタ、リソースエディタは SDK 3.1.2 のパッケージから削除され、BREW SDK Tools として別途配布される形式に改められました。これらのツールを必要とする場合は、BREW SDK Tools を別途インストールしてご利用ください。

なお、3.1 において MIF のフォーマットの変更はありませんので、3.0 以前のリソー

スエディタで作成した MIF をそのまま使っても問題はありません。

(2005.03.09)

### 51.3. PNG 画像の扱いについて

**ジャンル：バグ**

対象：MIF エディタ 3.0.3.14 ( BREW SDK Tools 1.0.0 )

現在、MIF のアイコンやサムネイルに PNG フォーマットの画像を指定すると、正しく MIF に画像が格納されず、また、MIF の生成時にオリジナルの PNG ファイルを書き換えてしまう不具合があります。PNG を利用する場合は、SDK3.0 に付属の MIF エディタをご利用ください。

(2005.03.09)

なお、本問題は 2006 年 1 月に公開された BREW SDK Tools 1.0.1 Ja で修正されました。

(2006.03.08)

## 52. リソースエディタ

### 52.1. 文字列に使用できない文字について

**ジャンル：バグ**

対象：2.0.1.5 リソースエディタ

現在、内部変換テーブルの不備により、下記文字は文字列リソースに直接書くことが出来ません。下記文字をリソースファイルに記述したい場合は、バイナリエディタで直接リソースファイルに記述して下さい。

対象文字：S-JIS コードで、

0x8161、0x817C、0x8191、0x8192、0x81CA、0x8740~0x879C

補記

このバグは 2.1 にて修正されました。

(2003.02.28)

### 52.2. リソースファイルにサウンドデータを格納するには

**ジャンル：Tips**

対象：ALL

現在、リソースファイル (.bar ファイル) には、文字列、画像、ダイアログの三種類しか格納することが出来ません。しかし、「画像」データとして、実際には任意のフォーマットを格納することが出来ます。

ISHELL\_LoadResData を用いると、RESTYPE\_IMAGE として格納した任意のデータを読み込むことが出来ます。

```

{
    IMedia * pim;
    AEEMediaData amd;
    byte * buf;
    uint32 dwSize, offset;

    if ( ISHELL_CreateInstance(pMe->a.m_pIShell, AEECLSID_MEDIAPHR,
        (void **)pim) != SUCCESS)
    {
        return FALSE;
    }

    // loaders.bar というリソースファイルから、PHRASEDATA_001 という
//リソース ID の音データを読み出す
    // まず、リソースのサイズを dwSize に取得する
    ISHELL_LoadResDataEx(pMe->a.m_pIShell, "loadres.bar",
        PHRASEDATA_001, RESTYPE_IMAGE, (void *)-1, &dwSize);

    // buf にデータを読み込む
    buf = ISHELL_LoadResData(pMe->a.m_pIShell, "loadres.bar",
        PHRASEDATA_001, RESTYPE_IMAGE);

    // buf の先頭バイトに、実際の音データへのオフセット値が
// 格納されているので、取得。
// リソースのサイズからはオフセット分を減じておく。
// ( オフセットされる部分には、MIME 文字列が収納されています。)
    offset = (uint32)buf[0];
    dwSize = dwSize - offset;

    amd.clsData = MMD_BUFFER;
    amd.pData = buf + offset;    // buf の先頭から offset バイトを
//音データのポインタとして渡す
    amd.dwSize = dwSize;
}

```

```
IMEDIA_SetMediaData(pim, &amd);  
  
IMEDIA_RegisterNotify(pim, (PFNMEDIANOTIFY)LoadRes_MediaCB, pMe);  
  
IMEDIA_Play(pim);  
  
return TRUE;  
}
```

(2003.02.28)

### 52.3.3.1.2 SDK におけるリソースエディタ

**ジャンル：Tips**

対象：3.1.2 以降の SDK

MIF エディタ、リソースエディタは SDK 3.1.2 のパッケージから削除され、BREW SDK Tools として別途配布される形式に改められました。これらのツールを必要とする場合は、BREW SDK Tools を別途インストールしてご利用ください。

なお、3.1 において bar のフォーマットの変更はありませんので、3.0 以前のリソースエディタで作成した bar をそのまま使っても問題はありません。

(2005.03.09)

## 53. AppLoader について

### 53.1. テスト署名の書き込み

**ジャンル：Tips**

対象：BREW ToolsSuite 2.0.1.2

現在、KDDI 様テスト端末にテスト署名を書き込もうとすると、AppLoader がエラーを表示して正常に書き込めない不具合があります。

この問題は、署名を含むフォルダをフォルダごとドラッグ&ドロップしてフォルダごと書き込むことで回避できます。

(2003.05.15)

補記

ToolsSuite2.1.1 ではこの問題は修正されました。

(2003.07.11)

### 53.2. 日本語ファイル名・フォルダ名について

**ジャンル：Tips**

対象：すべての BREW 端末

BREW 実機環境は日本語のファイル名に対応しておりません。日本語のファイルを

書き込んだ場合、ファイルシステムに不具合をきたし、ファイルの削除が出来ない・アプリケーションの起動が出来ないなどの問題が発生し、BREW が機能しなくなる恐れがありますので、日本語のファイル名を持つファイルは実機上に作成しないで下さい。

(2003.11.13)

## 54. 日本語の扱いについて

### 54.1. WSTR の定義と日本語の取り扱いについて

ジャンル：Tips

対象：ALL

BREW 上、WSTR は「AECHAR の配列」として定義されます。AECHAR とは、2 バイト長固定の文字を指します（実質は uint32 です）。ただし、WSTR の内部の文字コードが何であるかは、端末の実装に依存します。米国の端末は ISOLATIN1、韓国の端末は KSC901、日本の端末は S-JIS... というように、端末によって異なります。

BREW では、基本的に、「1 バイト固定の文字列 (char\* / STR)」もしくは「2 バイト固定の文字列 (AECHAR\* / WSTR)」しか扱うことが出来ません。このため、S-JIS のような、1 バイト 2 バイト混在の文字列を扱う為には、2 バイト長に整えてやる必要があります。逆に、外部と通信を行うような場合は、BREW で扱える WSTR 文字列から、標準の S-JIS 文字列に戻す必要も出てきます。

標準の S-JIS を BREW の WSTR に変換するには、STREXPAND 関数を用います。これは、1 バイト文字に対しては 0x00 をパディングし、2 バイト文字に対しては何も行わずに、S-JIS 文字列を 2 バイト固定の WSTR に変換します。

たとえば、“S-JIS 文字列” という文字列は標準の S-JIS で “53 2d 4a 49 53 95 b6 8e 9a 97 f1” ですが、これに STREXPAND をかけると、“53 00 2d 00 4a 00 49 00 53 00 95 b6 8e 9a 97 f1” という形 (2 バイト x8 文字) に引き伸ばされます。WSTRCOMPRESS はその逆を行います。

以下に簡単な使用例を示します。

```
{  
    AECHAR pszBuf[50];
```

```

STREXPAND((const byte *)"S-JIS 文字列", 12, pszBuf, 100);
IDISPLAY_DrawText(pMe->m_pIDisplay, AEE_FONT_BOLD, pszBuf,
                  -1, 0, 30, NULL, 0);
IDISPLAY_Update(pMe->m_pIDisplay);
}

```

STREXPAND の第 2 引数は、第 1 引数に渡した文字列の長さ (バイト数) です。2 バイトにパディングする前の長さを渡してください。また、第 4 引数は、第 3 引数に渡したバッファの長さ (バイト数) です。

(2003.05.01)

## 54.2. STRTOWSTR、WSTRTOSTR について

**ジャンル : Tips**

対象 : ALL

STRTOWSTR /WSTRTOSTR は、STREXPAND /WSTRCOMPRESS と紛らわしいですが、その効能は大きく異なります。STRTOWSTR は、1 バイト固定の文字列をWSTR にパディングするだけの文字列です。

たとえば、標準の S-JIS で “S-JIS 文字列” は “53 2d 4a 49 53 95 b6 8e 9a 97 f1” ですが、これに STRTOWSTR をかけると、“53 00 2d 00 4a 00 53 00 95 00 b6 00 8e 00 9a 00 97 00 f1 00” という形 (2 バイト x 11 文字) に引き伸ばされます。WSTRTOSTR はその逆を行います。

この API は、ISOLATIN1 のみからなる文字列をWSTR に引き伸ばすことを目的にデザインされています。実際には、STREXPAND を用いても同じ効果が得られますので、この API を使用する意義はほとんどありません。

(2003.05.01)

## 55. ARM 向けコンパイルについて

### 55.1. global / static 変数

**ジャンル : Tips**

対象 : ALL

BREW の構造上、global / static な変数は使うことが出来ません。

(2003.7.14)

### 55.2. 標準関数

**ジャンル : Tips**

対象：ALL

BREW の構造上、C の標準関数は使うことが出来ません。sprintf()、malloc()などは、BREW ヘルパ関数の SPRINTF()、MALLOC()などで代用してください。

(2003.07.14)

### 55.3.new、delete の扱いについて

ジャンル：Tips

対象：ALL

C++でプログラムを記述する場合、new、delete も前記で言う「標準関数」にあたりますので、MALLOC と FREE を用いて下記のようにオーバーロードしてください。

```
void *operator new( size_t size)
{
    return MALLOC(size);
}

void operator delete(void * ptr)
{
    FREE(ptr);
}

void *operator new [] (size_t size)
{
    return MALLOC (size);
}

void operator delete [] (void * ptr)
{
    FREE(ptr);
}
```

(2003.07.14)

### 55.4.浮動小数点数の扱いについて

ジャンル：Tips

対象：ALL

浮動小数点型の数に対して、四則演算の演算子や、int 型などからの代入を行うと、リンカエラーを起こします。代入には FASSIGN\_INT や FASSIGN\_STR、四則演算は FADD/FDIV/FMUL/FSUB をご使用ください。

なお、四則演算の演算子を使用しているかどうかを簡単にチェックするには、makefile の "CODE" の行に「-fpu none」というオプションを追加してコンパイルしてみてください。(このオプションをつけると浮動小数点型の変数が使用できなくなりますので、製品版のコンパイル時にはこのオプションは外す必要があります。)

(2003.07.14)

## 56. BREW Tools Suite

### 56.1. BREW Tools Suite 2.1.2 におけるバージョン設定

**ジャンル：Tips**

対象：BREW Tools Suite 2.1.2

2.1.2 より、BREW 3.0 機に対するサポートが加えられ、デフォルトの設定は BREW3.0 機向けの設定となりました。このため、デフォルトのままでは BREW2.x 機に対して Tools Suite2.1.2 を使用することは出来ません。

Tools Suite 2.1.2 を BREW1.x/2.x 機に対して使用するには、QCOMOEM.ini のバージョンの設定を 2 にしてお使いください。

(2004.02.12)

### 56.2. BREW Tools Suite 2.1.2.4 の接続性

**ジャンル：仕様**

対象：A5503SA

現在、BREW Tools Suite 2.1.2.4 と A5503SA の間で正常に接続が出来ないケースがあります。BREW Tools Suite 3.0 をご利用ください。

(2004.07.05)

## 57. BREW AppWizard

### 57.1. BREW AppWizard 日本語版の問題について

**ジャンル：バグ**

対象：BREW SDK Tools 1.0.0 Ja

BREW SDK Tools 1.0.0 Ja をインストールした場合、Visual Studio .NET 向けの BREW AppWizard が正常に動作しなくなります。

AppWizard で利用するテンプレートの文字コード指定が誤っていることが原因です。修正方法及び修正ファイルについては以下の URL を参照してください。

<http://brewforums.qualcomm.com/ja/showthread.php?s=&threadid=512>

(2005.07.11)

なお、本問題は 2006 年 1 月に公開された BREW SDK Tools 1.0.1 Ja にて修正されました。

## 58. その他

### 58.1. 2.0/2.1 におけるデバッグモードについて

#### ジャンル：Tips

対象：2.0、2.1 を搭載するテスト用端末及びエミュレータ

エミュレータ及びテスト用端末では、「#」キー3回+いずれかのキーを押すことで、デバッグ用の情報をディスプレイ上やログに出力することができます。

###1	ヒープのノードチェックを厳密に行います。エミュレータのみで意味を持ちます。
###2	<p>通信のステータスをディスプレイ上に表示するようになります。 ステータスを示すキャラクタは以下の意味を持ちます。</p> <p>'c'：コネクト中 'C'：コネクトされた 'I'：アイドル 'b'：バインド中 'B'：バインド済み 'R'：TCP ソケット読み込み中 'W'：TCP ソケット書き込み中 'r'：UDP ソケット読み込み中 'w'：UDP ソケット書き込み中 'x'：ソケットクローズ中 'X'：ソケットクローズ済み</p> <p>'^'：PPP オープン中 '='：PPP オープン済み 'v'：PPP クローズ中 '#'：PPP クローズ済み '&lt;'：PPP スリープ移行中 '~'：PPP スリープ状態 '&gt;'：PPP スリープから復帰中</p>
###3	ヒープ上の最大の空きメモリブロックのサイズをディスプレイ上に表示するようになります。

###4	特権レベルの違反があった場合にディスプレイ上にエラー表示するようになります。
###5	MALLOC 失敗時にディスプレイ上にエラー表示するようになります。
###6	100 回に 1 回の割合で、MALLOC に失敗するようになります。エミュレータ上でのみ動作します。
###7	デバッグ出力の動作を同期処理に変更します。ただし、同期モードであっても、NULL ポインタアクセスによるリセットなどではリセット処理が優先されるためメッセージが出力されることは期待できません。再度入力するとデバッグ出力を非同期に戻します。
###8	何も行いません。
###9	ヒープのダンプをログに出力します。
###*	リソースのキャッシュ (ISHELL_LoadRes**()) により内部的に生成されるキャッシュ) を強制的に開放します。
####	現在、利用できません。画面の描画が乱れます。
###0	上記コマンド群によりセットされたデバッグ用のフラグを全て off にします。

(2005.09.06)

## 58.2. 3.0/3.1 におけるデバッグモードについて

**ジャンル : Tips**

**対象 : 3.0/3.1 のシミュレータ及びテスト用端末**

シミュレータ及びテスト用端末では、以下のキーを入力することにより、端末のデバッグキー入力を有効にすることができます。

```
###273933284#
```

(数字部分は、キートップのアルファベットで「brewdebug」になります。)

デバッグキー入力有効時、“「#3 回」+数字キー+「#」” と入力することで、デバッグ用の情報をディスプレイやログに出力することが可能になります。

###1#	ヒープのノードチェックを厳密に行います。エミュレータのみで意味を持ちます。
###2#	通信のステータスをディスプレイ上に「8=R」のような形で表示するようになります。  左端の数字はネットワークの ID を示します。この値は実装に依存し、通

	<p>常変更することはできません。</p> <p>中央の文字は PPP のステータスを示します。各キャラクタは以下の意味を持ちます。</p> <p>'^' : PPP オープン中  '=' : PPP オープン済み  'v' : PPP クローズ中  '#' : PPP クローズ済み  '&lt;' : PPP スリープ移行中  '~' : PPP スリープ状態  '&gt;' : PPP スリープから復帰中</p> <p>右端の文字はソケットのステータスを示します。各キャラクタは以下の意味を持ちます。</p> <p>'c' : コネクト中  'C' : コネクトされた  'I' : アイドル  'b' : バインド中  'B' : バインド済み  'R' : TCP ソケット読み込み中  'W' : TCP ソケット書き込み中  'r' : UDP ソケット読み込み中  'w' : UDP ソケット書き込み中  'x' : ソケットクローズ中  'X' : ソケットクローズ済み</p> <p>ソケットが複数ある場合、ステータスは複数行表示され、各行が個々のソケットのステータスを表します。</p>
###3#	<p>ヒープ上の最大の空きメモリブロックのサイズをディスプレイ上に表示するようになります。</p>
###4#	<p>デバッグ出力の動作を同期処理に変更します。ただし、同期モードであっても、NULL ポインタアクセスによるリセットなどではリセット処理が優先されるためメッセージが出力されることは期待できません。再度</p>

	入力するとデバッグ出力を非同期に戻します。
###5#	モジュールのリストとレジストリ情報をログにダンプします。
###6#	ヒープのダンプをログに出力します。
###7#	リソースのキャッシュをダンプ出力します。
###8#	ファイルのキャッシュをダンプ出力します。
###9#	リソースのキャッシュ ( ISHELL_LoadRes**() )により内部的に生成されるキャッシュ)を強制的に開放します。
###10#	現在、利用できません。画面の描画が乱れます。
###11#	現在、利用できません。
###999#	デバッグのフラグをファイルに保持し、リセット後も状態を保つようになります。
###0#	上記コマンド群によりセットされたデバッグ用のフラグを全て off にします。

再度 “ ###273933284# ” を入力すると、デバッグキー入力は無効になります。

(2005.09.06)

## Appendix A.

### WebOpt一覧

WEBOPT_RESERVED	内部管理用
WEBOPT_RESERVED_LAST	内部管理用
WEBOPT_END	pValを持たない。WEBOPTのターミネータ、以降のWEBOPTを無視する。
WEBOPT_ANY	pValを持たない。GetOpt時、WEBOPTの列挙に用いる。
WEBOPT_REMOVED	pValを持たない。WEBOPTが削除されたことを示す。
WEBOPT_COPYOPTS	(boolean) を取る。TRUEの場合、IWebOptsがWebOptのvalueをコピーするようになる。この操作は一方通行であり、コピーを止めることができるのは空のIWebOptsに対してのみである。
WEBOPT_SZ	内部管理用
WEBOPT_SZ_LAST	内部管理用
WEBOPT_METHOD	(char *) を取る。デフォルトはGET。
WEBOPT_HEADER	(char *) を取る。CRLFでセパレートされる。例: "Name1: val1\r\nName2: val2\r\n"。デフォルトはNULL。
WEBOPT_EXTRAHEADERS	旧形式です。WEBOPT_HEADERを使用してください。
WEBOPT_PROXYSPEC	(char *) を取る。proxyspecフォーマット。
WEBOPT_AUTHBASIC	(char *) を取る。Basic認証情報を送る。未実装。
WEBOPT_USERAGENT	(char *) を取る。もしセットされなかった場合、デフォルトのUser-Agentを送る。NULLがセットされた場合、User-Agent情報を送らない。
WEBOPT_HANDLERDATA	(void *) を取る。ユーザ定義のデータを置く。IWebインタフェースを使用する際、保持しておきたいデータをセットすると、コールバック関数内外で共通のメモリを使用できる。ステータス保持などに使用。
WEBOPT_STATUSHANDLER	(PFNWEBSTATUS) を取る。ステータス処理用のコールバック関数を登録。
WEBOPT_HEADERHANDLER	(PFNWEBHEADER) を取る。ヘッダー処理用のコールバック関数を登録。
WEBOPT_CONNECTTIMEOUT	(int) を取る。タイムアウトの時間をミリ秒で指定。0は端末のデフォルト値使用を意味する。
WEBOPT_CONTENTLENGTH	(long) を取る。Content-Lengthヘッダ。
WEBOPT_IDLECONNTIMEOUT	(int) を取る。無通信時の切断までの時間を秒で指定。デフォルトはPPPのLinger値。-1は無限。
WEBOPT_ACTIVEXACTIONS	(unsigned int) を取る。最大同時トランザクション数を指定。これを超えると、IWebはトランザクションをキューする。
WEBOPT_INTERFACE	内部管理用
WEBOPT_INTERFACE_LAST	内部管理用
WEBOPT_BODY	(ISource*) を取る。POSTするデータ本体を渡す。
WEBOPT_DEFAULT	(IWebOpt*) を取る。IWebOptsをチェーンするために用いる。
WEBOPT_VARBUFFER	pValとして、未確定な、コピー可能な構造体へのポインタを取る。指定された構造体の最初の2バイトがデータサイズを指定し、以下、実際のデータが続く。このサイズ指定はリトルエンディアンである。
WEBOPT_VARBUFFER_LAST	内部管理用
WEBOPT_ASN1	pValとして、ASN.1を内包するコピー可能な構造体へのポインタを取る。最初の1バイトはASN.1のタイプであり、2バイト目から実際のASN.1になる。構造体全体のサイズは、2バイト目から始まるASN.1のサイズに1バイトを加えることで得られる。ただ、ASN.1オブジェクト自体が64KBを超えることは出来ない。また、このオブジェクトはDER/BERでエンコードされていることを想定している。 これらのオプションをセットする際には、(特にWEBOPT_COPYOPTSがセットされている場合)、サイズが有効であるかどうかをチェックすることを推奨する。大きすぎるオプションをセットした場合、巨大なヒープをmallocしようとして、後ろのほうのデータを切り捨ててしまったりすることがある。
WEBOPT_ASN1_LAST	内部管理用
WEBOPT_VARIABLE	pValとして、未確定な、コピー可能な構造体へのポインタを取る。OptIDの下位12bitが指定された構造体のサイズ(バイト数)を示す。OptIDとして使用できる値は0x10000XXX ~ 0x1ffffXXXである (XXXは上記の通りデータのサイズとして使用される)。
WEBOPT_VARIABLE_LAST	内部管理用
WEBOPT_VARIABLE_INC	内部管理用
WEBOPT_SSL_SZ	内部管理用
WEBOPT_SSL_SZ_LAST	内部管理用
WEBOPT_SSL_32BIT	内部管理用
WEBOPT_SSL_32BIT_LAST	内部管理用
WEBOPT_SSL_VARIABLE	内部管理用

WEBOPT_SSL_VARIABLE_LAST	内部管理用
WEBOPT_SSL_INTERFACE	内部管理用
WEBOPT_SSL_INTERFACE_LAST	内部管理用
WEBOPT_SSL_SEC_INFO	(SSLInfo *) を取る。
WEBOPT_SSL_TRUST_MODE	APIリファレンス、ISSL_Negotiate()の項参照。[ISSL_NegotiateV()が重複していますが、最初の方がISSL_Negotiate()のリファレンスです。]
WEBOPT_SSL_ALLOWED_CIPHER_SUITES	APIリファレンス、ISSL_Negotiate()の項参照。[ISSL_NegotiateV()が重複していますが、最初の方がISSL_Negotiate()のリファレンスです。]
WEBOPT_SSL_ALLOWED_VERSIONS	APIリファレンス、ISSL_Negotiate()の項参照。[ISSL_NegotiateV()が重複していますが、最初の方がISSL_Negotiate()のリファレンスです。]
WEBOPT_SSL_WANT_X509_CHAIN	APIリファレンス、ISSL_Negotiate()の項参照。[ISSL_NegotiateV()が重複していますが、最初の方がISSL_Negotiate()のリファレンスです。]
WEBOPT_SSL_X509_CHAIN	APIリファレンス、ISSL_Negotiate()の項参照。[ISSL_NegotiateV()が重複していますが、最初の方がISSL_Negotiate()のリファレンスです。]
WEBOPT_SSL_SESSIONDB	APIリファレンス、ISSL_Negotiate()の項参照。[ISSL_NegotiateV()が重複していますが、最初の方がISSL_Negotiate()のリファレンスです。]
ISSL_SOCKET_IOCTL_RESERVED1	内部管理用
WEBOPT_X509_SZ	内部管理用
WEBOPT_X509_SZ_LAST	内部管理用
WEBOPT_X509_32BIT	内部管理用
WEBOPT_X509_32BIT_LAST	内部管理用
WEBOPT_X509_VARIABLE	内部管理用
WEBOPT_X509_VARIABLE_LAST	内部管理用
WEBOPT_X509_VARBUFFER	内部管理用
WEBOPT_X509_VARBUFFER_LAST	内部管理用
WEBOPT_X509_ASN1	内部管理用
WEBOPT_X509_ASN1_LAST	内部管理用
WEBOPT_X509_ROOT_CERTS	ここにセットされたASN.1がtrustedなroot certであることを示す。
WEBOPT_X509_BRANCH_CERTS	リーフからルートにいたるための中間のcertを示す。
WEBOPT_X509_LEAF_CERT	リーフ(目的サーバ)のcert。
WEBOPT_X509_CHAIN_CERT	チェーンの結果が格納される。IX509CHAIN_Verify()の後に呼ばれなくてはならない。インデックス0がリーフ(目的サーバ)の証明書。
WEBOPT_X509_HOST	証明書と照合すべきホスト名。NULLでターミネートされている必要がある。
WEBOPT_X509_OVERRIDE	(X509TrustOverride*) を取る。OVERRIDEしたオプションのリスト。
WEBOPT_X509_MAX_CHAIN	(unsigned long) チェーンする長さの最大値。ただし、X509CHAIN_MAX_CHAINの値は超えられない。
WEBOPT_X509_EXT_KEY_USE	(32bit ビットベクトル) ASNOIDのセット。拡張情報をチェックするASNOIDを列挙する。
WEBOPT_X509_KEY_USE	(32bit ビットベクトル) ビット配列。証明書の使われ方を提示(giving the intended use)する。ASNOID_KEYUSE_* のいずれか。
HVOPT_N_INDENT	(int) を取る。インデントのピクセル数。
HVOPT_N_VSPACE	(int) を取る。段落の間隔。単位はピクセル。
HVOPT_N_SBWIDTH	(int) を取る。スクロールバーの幅。単位はピクセル。
HVOPT_N_ERROR	(int)。getのみ。HTMLVIEWER固有のエラーが格納される。
HVOPT_RGB_TEXT	(RGBVAL) を取る。文字色指定。
HVOPT_RGB_LINK	(RGBVAL) を取る。リンク文字色指定。
HVOPT_SZ_TYPE	(char *) を取る。MIMEタイプもしくはcharsetを指定。setのみ。
HVOPT_SZ_URL	(char *) を取る。Base URLを指定する。setのみ。
HVOPT_PTR_IWEB	(IWeb *) を取る。HTMLVIEWERにIWebインタフェースをセットする。