

【A6】 Delphiテクニカルセッション

RTLソースを利用するDelphiデバッグ技法

CodeGear

R&D

有澤 雄志

アジェンダ

- RTL の利用準備
- IDE から使ってみる

RTL ソースコード

- ランタイムライブラリのほとんどすべてのソースコードが製品に含まれている。
 - Delphi : RTL と VCL
 - C++ : RTL
- ソースコードが提供されていない物
 - サードパーティ提供のもの (TeeChart など)
 - データベースドライバ (BDE, BDP, DBX)
 - IDE に含まれる物

RTL ソースコードをどう使うか

- 研究
- 再利用
- 機能改善
- デバッグ
- ...

RTL ソースコードをどう使うか / 機能改善

- 特定 CPU / OS に依存した特化版の作成
- 標準 RTL を独自に高速化
 - Delphi2006 で登場した新しいメモリマネージャである FastMM は、元は RTL 高速化グループの成果物

RTL ソースコードをどう使うか / デバッグ

- 深い部分でのソースコードデバッグ
 - 標準関数の内側まで
 - 例外発生時、ソースコードが開く。
- RTL ソースを自分のプロジェクトに登録 (Delphi)
 - 部分的な改造が可能

気をつける点

- 巨大なデバッグ情報
 - リンクが遅くなる...
- コンパイラの設定によっては、生成コードが異なる
 - コンパイルスイッチをよく読むこと
 - 最適化を切った方がデバッグしやすくなる場合...
 - 最適化を切るとバグが消える場合...

RTL ソースコードの場所

- Delphi
 - C:\Program Files\CodeGear\RAD Studio\5.0\source\Win32\rtl
 - C:\Program Files\CodeGear\RAD Studio\5.0\source\Win32\vcl
- C++Builder
 - C:\Program Files\CodeGear\RAD Studio\5.0\source\cpprtl

RTL の利用準備

- デバッグ情報をつけたライブラリを作成する。
 - ライブラリやデバッグ情報が大きくなるとリンクに時間がかかる。
 - 慣れてきたら必要な部分だけをデバッグ版に置き換えてみましょう
 - ライブラリの再構築ができるようになると、RTL を改造し再利用することが容易になる。

RTL の利用準備 - C/C++

- ライブラリを再構築するためのバッチファイルが提供されています。
 - ライブラリ (.lib) の形にする必要がある。
 - ビルド用のヘッダーファイルなどがあるため、個々の RTL ソースをアプリケーションで直接利用するのは難しい。

RTL の利用準備 - Delphi

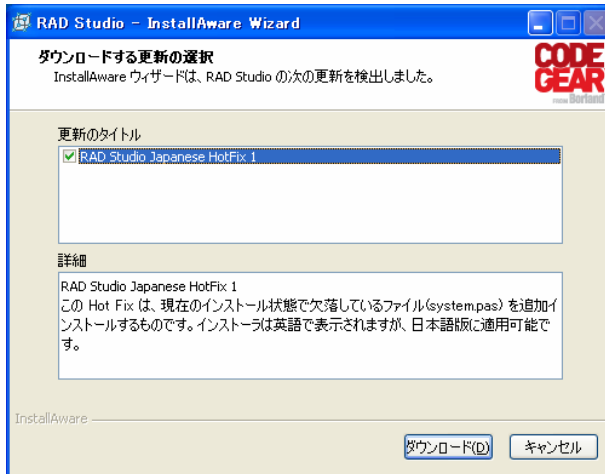
- 基本的にソースコードでの提供
 - 一つの .PAS ソースが一つの .DCU (コンパイル済みユニット) に対応しているので、.DCU と同じ場所に .PAS をコピーするだけで足りる。
- SYSTEM/SYSUTIL ユニットだけは例外
 - これらを含む RTL ソースをコンパイルするために makefile が提供されています。

デバッグ版ライブラリとの違い

	共通	リリース	デバッグ
C/C++	lib	lib¥release	lib¥debug
Delphi	lib	lib	lib¥debug

- 元から提供されているデバッグ版ライブラリとの違いは何か？
 - VCL がデバッグ版かどうか。
 - C/C++ RTL はリリース版しか提供されていない

ビルドの前に / HotFix 1 を当ててください



ビルドしてみよう！ - C/C++ (1)

source¥cpprtl¥readme に手順が書いてあります。

1. cpprtl ディレクトリに移動

```
cd "C:¥Program Files¥CodeGear¥RAD  
Studio¥5.0¥source¥cpprtl"
```

2. Tools ディレクトリを PATH 環境変数に追加

```
>path %path%;%CD%¥Tools
```

3. Build.bat を実行

```
>build
```

ビルドしてみよう！ - C/C++ (2)

- 標準とは異なるディレクトリに RTL ソースコードをインストール/移動した場合は、ソースコードのフルパスを RTLROOT 環境変数を設定してください。
 - 詳しくは付属の Readme を。

ビルドしてみよう！ - C/C++ (3)

- デバッグ情報をつけるには
 - DEBUG 環境変数に 1 を設定してビルド。
`>SET DEBUG=1`

ビルドしてみよう！ - C/C++ (4)

- cpprtl¥lib ディレクトリに新しいライブラリが作成されます。
- RAD Studio¥5.0¥lib の下に上書きすることで変更後のライブラリを使うことができます。
 - cc3280mt.dll などの実行時 DLL は Windows¥System32 に上書きコピーします。
 - オリジナルは消さないように。バックアップは必ずとっておきましょう。

ビルドしてみよう！ - Delphi (1)

1. RTL ディレクトリに移動

```
cd "C:¥Program Files¥CodeGear¥RAD  
Studio¥5.0¥source¥Win32¥RTL"
```
2. LIB ディレクトリを作成。すでにある場合には中を空に。

```
>mkdir Lib  
>del Lib¥*.*
```
3. Make ファイルを実行

```
>make
```

ビルドしてみよう！ - Delphi (2)

- デバッグ情報をつけるには
 - DEBUG 環境変数に 1 を設定してビルド。

```
>SET DEBUG=1
>make -B
```

ビルドしてみよう！ - Delphi (3)

- source¥Win32¥RTL¥lib にコンパイル済みユニット (.dcu) が作成されます。
- RAD Studio¥5.0¥lib の下に上書きすることで変更後のライブラリを使うことができます。
 - オリジナルは消さないように。バックアップは必ずとっておきましょう。

ビルドしてみよう！ - Delphi (4)

- VCL はどうすればコンパイルできる？
 1. .DCU と同じディレクトリにコピー
 2. 自分のプロジェクトと同じディレクトリにコピー
- 実行時パッケージはどうやって再コンパイルする？
 - IDE でパッケージを作成。

ライセンスに関して

- License.txt/.rtf には目を通してください。
- 常識の範囲で！
 - あなたの書いたコードはあなたの物。好きにしてください。
 - あなたが書いていないコードは、誰か別の人の物。その人の意向に従いましょう。

ライセンスに関して

- 現在の「CODEGEAR 使用契約」では...
 - ソースコードそのものは「再配布」できません
 - コンパイルされたバイナリ(.exe / .dll / .bpl) は変更した部分を含め、自由に配布できます。
 - 3.1 再配布可能ファイルを含まないコンパイルされたコードは自由に再配布できます。
- 再配布対象のファイルの再配布は避けてください。
 - どうしても配布したい場合は名前を変える。

デバッグしてみましょう

- デモ

Q & A

- Any question?