



## コンテキスト作成 導入マニュアル

## 1 . はじめに

このマニュアルは、ドリームキャスト用に音声認識プログラムを作成するプログラマーのために、  
グラマーファイルの生成から始まり、実際にアプリケーションプログラムの中で使用できる認識語彙  
ファイル（コンテキスト）にコンパイルするまでの手順をなるべく簡潔に説明したものです。  
音声認識エンジンは、英語、日本語のいずれの言語でも動作しますが、今回は日本語を例にとり、  
以下の説明を行います。

## 2 . BNF グラマーファイルの作成

最初に認識させたい語彙を規定するグラマーファイルを作成します。

NotePad などのスクリーンエディターを使用してください。

離散単語認識のみをご使用の場合、L&H 開発ツールの Lextool でグラマーファイルの作成が可能です。

以下に BNF グラマーファイルの簡単な例：steerplane\_j.bnf を示します。

```
!export <ControlCommands>;
```

```
<ControlCommands>: "オソク" | "ハヤク" | "トマレ" | "ハンテン" | "エンジンテーシ";
```

### 注意：

日本語の認識語彙の表記は、全角カタカナ、（全角）ひらがな、半角数字が使用できます。

BNF ファイルの中で日本語表記をする場合、上記の例のように必ず " "（ダブルクォーテーション）  
でくる必要があります。（半角数字は除く）

グラマーファイルの記述文法の詳細は、別マニュアル：Development Tools User's Guide を参照してください。

## 3 . BNF グラマーファイルのコンパイル

つぎに、この BNF グラマーファイルをコンパイルし、.ctx および.wcl という 2 つのファイルを生成します。

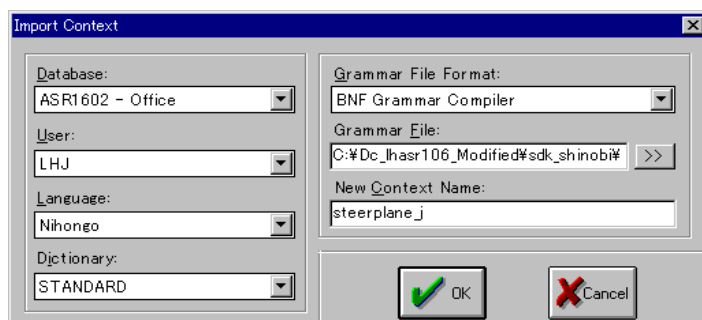
このコンパイルは、Lextool または、AsrBatchTool を使用して実行しますが、今回は、Lextool を使用する  
手順を説明します。AsrBatchTool につきましては、別マニュアル：L&H Automation Tools for ASR Context  
Management Reference Guide を参照してください。

また、Lextool の詳細な説明につきましては、別マニュアル：Development Tools User's Guide を参照して  
ください。

- 1 ) Lextool for Japanese を起動し、Context メニューの中の Import を実行します。

実行時、Language の指定が Nihongo、Grammar File Format が BNF Grammar Compiler に設定されて  
いることを確認してください。

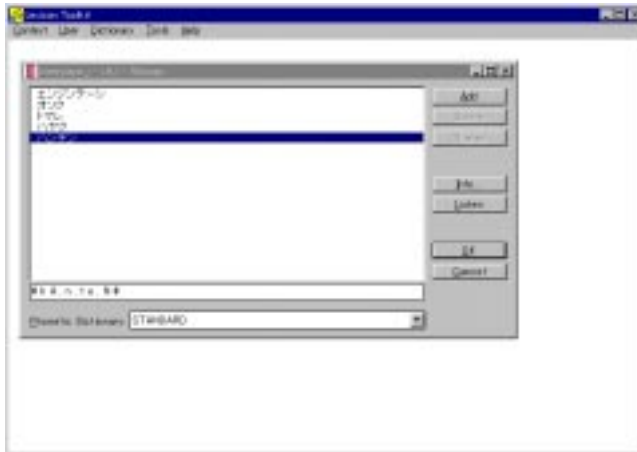
その後、ターゲットの Grammar File を指定し、適当な New Context Name を入力し（今回の場合  
steerplane\_j）、OK ボタンを押します。正常に Import が終了した場合、その旨のメッセージが  
表示されます。



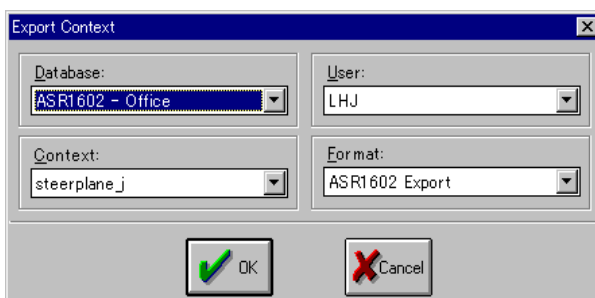
**注意：**Language の指定が Nihongo になっていない場合、新規ユーザー登録をして、そのユーザーの対応言  
語を日本語で登録する必要があります。これには、User メニューの Create を実行し、適当なユーザー名を  
入力し、OK ボタンを押してください。新規ユーザー登録か？聞いてきますので、OK として、つぎの画面  
で、そのユーザーの使用する言語に Nihongo を登録してください。



- 2) つぎに Context メニューで今 Import したファイルを Open してみます。  
下記のように、BNF グラマーファイルで定義した日本語の語彙が表示されるはずです。



- 3) Context ファイルの Export を実施します。  
Context メニューの Export を実行してください。  
実行時、Context 名へ先ほど定義した名前を入力、Format へ AR1602 Export を指定して、OK ボタンを押してください。  
.ctx と .wcl の 2 つのファイルを生成するフォルダ名を順次聞いてきますので、BNF グラマーファイルの存在する場所と同じフォルダを指定して、上記 2 つのファイルを保存してください。(今回は、steerplane\_j.ctx および steerplane\_j.wcl)



- 4) 上記で生成した .wcl ファイル (steerplane\_j.wcl) から 2 つのファイル (.cls と .wrd) を作成します。  
作成には、ConvWcl.exe をコマンドラインから実行します。実行時の書式は、以下の例を参照してください。  
ディレクトリの指定は、.wcl ファイルの存在する所と同じにしてください。2 つの新しいファイルはこの場所に生成されます。

<例> C:\¥...¥context>C:\¥...¥CONVWCL.exe steerplane\_j.wcl steerplane\_j.cls steerplane\_j.wrd

**注意:** このとき .cls と .wrd のファイル名指定の順番は上記のとおりでなくてはなりません。

- 5) 上記までで、BNF グラマーファイルの一連のコンパイルは全て終了しました。

#### 4 . まとめ

以上が認識させる語彙を実際のプログラムで使える形のコンテキストに変換するまでの手順です。  
今回は、孤立する認識単語を例に挙げて説明しましたが、BNF グラマーファイルの記述により、  
連続単語認識、連続数字認識、またこれらを組み合わせた語彙も生成可能です。  
詳しくは、別マニュアル：Development Tools User's Guide の 57 ページ (以降)：THE BNF GRAMMAR  
LANGUAGE を参照してください。

以上