NAME

ci − WAM knowledge DB library

SYNOPSIS

```
#include <geta/ci.h>
```

DESCRIPTION

CI は各 WAM に固有の知識データベースである. libci は CI へのアクセス手段を与える.

libci を使うには, まず ci_init(3) を最初に呼ぶ. この時点でデータベースファイル $GETAROOT/etc/ci.conf が読み込まれ, 以後, libci の他の関数によりこのデータベースへのアクセスが行われるようになる. wam_init(3) が ci_init(3) を呼びだすため, 通常の使用では wam_init(3) を呼ぶことで libci を初期化することが推奨される. なお, ci_init(3) の呼び出し以降のデータベースファイルへの変更は, 実行中のプログラムには反映されない.

WAM の属性(システム標準)と, その指定が無い時の初期値は以下の通りである. なお, flags の値は整数, データベース上での表現は 16 進数であり, その値は ci_flags(3) により取得することができる. それ以外の属性の値は文字列であり, その値は ci_value(3) で名前を指定することで取得することができる. 属性名は case insensitive である.

```
属性名    省略時の値
handle  (MANDATORY)
short-name      NULL
dataroot        $GETAROOT/data/$handle/
flags   0
```

データベースでの記述は

```
属性名,  :  , スペース, 値
```

または

```
属性名,  :p:  , スペース, 値
```

を1行に書く. (したがって, 値に改行を含めることはできない.) 各 WAM についての記述は handle 属性から次の handle 属性の直前までであり, 最初の handle 属性の値が WAM のハンドルとなる(以後, $handle とも表記する). CW や XR はそれぞれのデータベースを (標準的には) $GETAROOT/data/$handle/ というディレクトリ以下に格納する. 属性名の後に :p: を書いた場合, 値の先頭が / でなければ属性 dataroot の値がその値の先頭に付加される.

上記以外の属性として任意の名前を持つ属性をユーザが用いることができる. 値は全て文字列であり, 値の取得はシステム標準の属性と同じく ci_value(3) により可能である.

DIAGNOSIS

エラーが発生すると, 整数が戻り値の関数は −1 を返し, ポインタが戻り値の関数は NULL を返す.

BUGS

WAM は 128 個までしか定義できない.

FILES

$GETAROOT/etc/ci.conf

LIST OF FUNCTIONS

Function	Description
ci_init	libci を初期化する
ci_handles	ハンドルの一覧を得る
ci_value	ハンドルで示される WAM の指定した属性の値を得る(文字列)

ci_flags ハンドルで示される WAM の flags 属性の値を得る(整数)
ci_print_dataset ハンドルで示される WAM の属性をストリームに書き出す