



ATOK10 / ATOK11 for Windows

アプリケーションプログラムインターフェース仕様書

- * 「ATOK11」は、株式会社ジャストシステムの著作物であり、
「ATOK11」にかかる著作権その他の権利はすべて
株式会社ジャストシステムに帰属します。
- * 「ATOK」は株式会社ジャストシステムの登録商標です。
- * その他、記載された製品名は各社の商標または登録商標です。

第2版：1998年12月22日
(C) 1998 株式会社ジャストシステム

ご使用条件

株式会社ジャストシステム（以下「弊社」）は、以下に定めるドキュメント、サンプルプログラム〔ソース/実行形〕、開発用ファイル（以下総称して「本技術情報」）を受領するお客様（以下「受領者」）が下記の条件にご同意されることを条件として、本技術情報の使用を許諾いたします。受領者が本技術情報のご使用を開始された場合は、下記条件に同意いただいているものとみなします。（本使用条件は、ドキュメントの本文、および開発用ファイル、サンプルプログラムの「License.txt」にも記載されています。）

ドキュメント : ATOK10 / ATOK11 for Windows
アプリケーションプログラムインターフェース仕様書
サンプルプログラム : (該当するものはありません)
開発用ファイル : ATOKLIB.H、 ATOKLIB.LIB、 ATOK10WC.LIB

記

1 定義

「アプリケーション」とは、本使用条件に従い本技術情報を使用して受領者が開発するプログラムまたはソフトウェア製品などをいいます。

2 受領者は、受領者個人または受領者の属する法人などの内部的な範囲において、個人的または商業的とを問わず、アプリケーション開発を目的とする場合に限り、以下に定める範囲で本技術情報を使用することができます。

- (1) 本技術情報を、閲覧、ダウンロード、複製して使用すること。
- (2) サンプルプログラムのソースコードについては、改変し、アプリケーションに組み込むこと。
- (3) アプリケーションを第三者に頒布すること。

3 受領者は以下の行為を行わないものとします。

- (1) 本使用条件に定める以外の目的での本技術情報の使用。
- (2) 本技術情報の一部または全部の、第三者への頒布、販売、再使用許諾、送信、Web サイトなどネットワークサーバへの掲示。
- (3) ドキュメント、開発用ファイルの一部または全部の、改変、リバースエンジニアリング。
- (4) 本技術情報を利用することによる、弊社 ATOK シリーズ競合製品の開発。
- (5) アプリケーションの開発、評価および使用の際に、真正品以外の ATOK または弊社製品を使用すること。

4 受領者はアプリケーションを頒布する際、有償であるか否かを問わず、以下を遵守するものとします。

- (1) 弊社の書面による許諾なく、弊社の社名や製品名等のロゴを使用しないこと。
- (2) アプリケーションが弊社製品に対応する旨を表示したり、マニュアル等で商品内容や操作を説明したりする場合を除き、弊社の社名および商品名を表示しないこと。
- (3) 前項に従って、製品名を記載する場合は、「ATOK」が弊社の登録商標である旨をあわせて表示すること。
- (4) アプリケーションについて弊社による推奨、監修、品質保証等がある、もしくは、アプリケーションが弊社から販売されているかの誤認を与えるような表示を行わないこと。
- (5) アプリケーションの頒布により生じた第三者との紛争に関しては、受領者の責任と負担により解決するものとし、弊社に損害を与えないこと。

5 本技術情報にかかる著作権その他の権利は、弊社に帰属するものであり、受領者は、本技術情報に付されている著作権、商標権その他の知的財産権の表示を削除、変更することはできません。

6 弊社は、

- (1) 本技術情報は現状有姿で提供され、その品質および機能が完全であることおよび受領者の使用目的に適合することを保証するものではなく、本技術情報についての瑕疵担保責任および保証責任を一切負いません。本技術情報の選択導入は受領者の責任で行っていただき、本技術情報の使用およびその結果についても同様とします。
- (2) 本技術情報およびアプリケーションの使用または使用不能から生ずる直接的または間接的損害については一切責任を負いません。

7 弊社は受領者に対し、本技術情報に関するサポートは一切行いません。

8 弊社は、受領者が本使用条件のいずれかの条項に違反した場合または弊社の著作権を侵害した場合には、本技術情報の受領者による使用を終了させることができます。その場合、受領者は、受領者が有する本技術情報及びその複製物を必ず破棄して下さい。

9 本技術情報は随時変更されることがあります。

以 上

目 次

1 . 概 要	1
2 . 関数仕様書	2
2 - 1 . 関数一覧	2
2 - 2 . 各関数の仕様	3
3 . サンプル	16
3 - 1 . 未確定文字列表示をアプリケーション側で行う	16
3 - 2 . 未確定文字列表示はIMEに行わせる	17
4 . 付 録	18
4 - 1 . ATOKComposition 取得関数	18
4 - 2 . ATOK Composition 構造体定義	18
4 - 3 . 未確定文字表示色構造体定義	20
4 - 4 . ATOK 辞書設定用構造体定義	21
4 - 5 . 辞書エラー種類一覧	21
4 - 6 . ATOK マウスイベント処理関係定義	22
4 - 7 . ATOK 機能キー番号定義	22
4 - 8 . 単語登録品詞取得用構造体定義	23
4 - 9 . 品詞名一覧	23

1 . 概 要

本書は ATOK10 for Windows および ATOK11 for Windows のアプリケーションインターフェースの仕様を記述している。

ATOK アプリケーションインターフェースは、IME 共通のコントロールをベースに ATOK 固有のものを追加している。これにより ATOK アプリケーションインターフェースを使用するアプリケーションは、使用中の IME を意識することなく制御が可能になる。

また、ATOK アプリケーションインターフェースを使用するための特別な手続きの必要はない。

なお、ATOK アプリケーションインターフェースは DLL(ATOKLIB.DLL)で提供される。

ATOK アプリケーションインターフェースを使用するにあたり、Windows95 の IME 制御についてを一読されることをお勧めする。

Multilingual IME Specification for Application

Microsoft Windows95 プログラマーズガイド(第 30 章)

(関数、メッセージの詳細については Microsoft Win32 Software Development Kit (SDK)または IMM.H を参照。)

また、ここでいう ATOK 固有の情報の追加とは、次のとおり。

ATOK 固有のモードの設定

ATOK 固有の出力情報の獲得

- ・ ATOK のユーザーインターフェースを実現するための情報(ATOK の表示をアプリケーション側で行う場合)
- ・ ネイティブな Unicode(書体情報含む)

開発用ファイルについて

必要となる開発ファイルは次のとおり。

- ・ ATOK11 でご利用の場合... ATOKLIB.H、 ATOKLIB.LIB
- ・ ATOK10 でご利用の場合... ATOKLIB.H、 ATOK10WC.LIB

第 2 版(1999 年 12 月 22 日)での変更点

- ・ 開発用ファイルに ATOK10WC.LIB を追加。
(追加したファイルは ATOK10 で利用するとき必要)
- ・ ATOK11、 ATOK10 それぞれで必要となる開発ファイルを明記。
- ・ 関数「AT_SendVkeyEvent」の説明に「ATOK11 から有効」を追加。

2 . 関数仕様書

2 - 1 . 関数一覧

無 印...ATOK11 / ATOK10で利用可能

* 印...ATOK11で利用可能

No.	名前	機能
1.	AT_SetKoteiInput	固定入力モード設定
2.	AT_GetKoteiInput	固定入力モード取得
3.	AT_SetEijiInput	英字入力モード設定
4.	AT_GetEijiInput	英字入力モード取得
5.	AT_SetKutouten	句読点組み合わせ設定
6.	AT_GetKutouten	句読点組み合わせ取得
7.	AT_SetKougotaiTransMode	口語体変換モード設定
8.	AT_GetKougotaiTransMode	口語体変換モード取得
9.	AT_SetFukugougoTransKind	複合語変換モード詳細設定
10.	AT_GetFukugougoTransKind	複合語変換モード詳細取得
11.	AT_SetDefaultDicNo	基本辞書セット番号設定
12.	AT_GetDefaultDicNo	基本辞書セット番号取得
13.	AT_SetDicFileNameSet	辞書セットごとの辞書ファイル名設定
14.	AT_GetDicFileNameSet	辞書セットごとの辞書ファイル名取得
15.	AT_SetDicLearnMode	辞書セットごとの学習モード設定
16.	AT_GetDicLearnMode	辞書セットごとの学習モード取得
17.	AT_SetOkuriKind	送りがなモード設定
18.	AT_GetOkuriKind	送りがなモード取得
19.	AT_SetDicFileSetNickname	辞書セットごとの名称設定
20.	AT_GetDicFileSetNickname	辞書セットごとの名称取得
* 21.	AT_SendMouseEvent	未確定文字操作に関するマウスイベントを送る
* 22.	AT_SendVkeyEvent	ATOK 機能キーイベントを送る
23.	AT_GetIMECompCollInfo	IME 未確定文字列表示色テーブルの取得
24.	AT_ImmGetAtCompositionPacket	ATOKComposition 一括取得
25.	AT_ImmGetKanjiOn	漢字 ON/OFF 状態取得
26.	AT_ImmSetKanjiOn	漢字 ON 設定
27.	AT_ImmSetKanjiOff	漢字 OFF 設定
28.	AT_ImmSetSentenceMode	変換モード設定
29.	AT_ImmGetSentenceMode	変換モード取得
30.	AT_ImmSetRomanMode	漢字入力モード設定
31.	AT_ImmGetRomanMode	漢字入力モード取得
32.	AT_ImmSetKanjiInputMode	入力文字種設定
33.	AT_ImmGetKanjiInputMode	入力文字種取得
34.	AT_ImmSetInputModeEx	入力モード設定 (固定入力含む)
35.	AT_ImmGetInputModeEx	入力モード取得 (固定入力含む)
36.	AT_ImmRegisterWord	単語登録
37.	AT_ImmUnregisterWord	単語削除
38.	AT_ImmGetRegisterWordStyle	品詞リスト取得

2 - 2 . 各関数の仕様

1.

名前	int WINAPI AT_SetKoteiInput(HIMC hIMC, int mode)
機能概要	固定入力モード設定
入力	HIMC hIMC 使用するInputContextハンドル int mode ... 設定する固定入力モード ATKOTEI_OFF ... 固定解除 ATHIRAGANA_KOTEI ... ひらがな固定 ATZENKAKU_KATAKANA_KOTEI ... カタカナ固定 ATZENKAKU_EISU_KOTEI ... 英数固定 ATHANKAKU_KATAKANA_KOTEI ... カタカナ半角固定 ATHANKAKU_EISU_KOTEI ... 英数半角固定
出力	正常終了 ... 以前の固定入力モード 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	

2.

名前	int WINAPI AT_GetKoteiInput(HIMC hIMC)
機能概要	固定入力モードの読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... 現在の固定入力モード ATKOTEI_OFF ... 固定解除 ATHIRAGANA_KOTEI ... ひらがな固定 ATZENKAKU_KATAKANA_KOTEI ... カタカナ固定 ATZENKAKU_EISU_KOTEI ... 英数固定 ATHANKAKU_KATAKANA_KOTEI ... カタカナ半角固定 ATHANKAKU_EISU_KOTEI ... 英数半角固定 異常終了 ... AT_NOTATOK 選択中のIMEがATOKでない
特記事項	

3.

名前	int WINAPI AT_SetEijiInput(HIMC hIMC, int mode)
機能概要	英字入力モードの設定
入力	int mode ... 設定する英字入力モード AT_OFF ... OFF AT_ON ... ON
出力	正常終了 ... 以前の英字入力モード 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	

4.

名前	int WINAPI AT_GetEijiInput(HIMC hIMC)
機能概要	英字入力モードの読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... 現在の英字入力モード AT_OFF ... OFF AT_ON ... ON 異常終了 ... AT_NOTATOK 選択中のIMEがATOKでない
特記事項	

5.

名前	int WINAPI AT_SetKutouten(HIMC hIMC, int mode)
機能概要	句読点組み合わせの設定
入力	HIMC hIMC 使用するInputContextハンドル int mode ... 次のビットの組み合わせで指定する ATKUTOUTEN_KIGOU ・ ... 1 / ... 0 ATKUTOUTEN_KAKKO 「 」... 1 []... 0 ATKUTOUTEN_KUTEN 。 ... 1 0 ATKUTOUTEN_TOUTEN 、 ... 1 , ... 0
出力	正常終了 ... 以前の句読点組み合わせの設定値 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	指定された値が範囲外の場合は、設定せずにエラーで復帰する。

6.

名前	int WINAPI AT_GetKutouten(HIMC hIMC)
機能概要	句読点組み合わせの読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... 現在の句読点組み合わせ設定値 次のビットの組み合わせで取得できる ATKUTOUTEN_KIGOU ・ ... 1 / ... 0 ATKUTOUTEN_KAKKO 「 」... 1 []... 0 ATKUTOUTEN_KUTEN 。 ... 1 0 ATKUTOUTEN_TOUTEN 、 ... 1 , ... 0 異常終了 ... AT_NOTATOK 選択中のIMEがATOKでない
特記事項	

7.

名前	int WINAPI AT_SetKougotaiTransMode(HIMC hIMC, int mode)
機能概要	口語体変換モードの設定
入力	HIMC hIMC 使用するInputContextハンドル int mode ... 設定する口語体変換モード AT_ON ... OFF AT_OFF ... ON
出力	正常終了 ... 以前の口語体変換の設定 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	未確定文字がある場合、あるいは指定された値が範囲外の場合は、設定せずにエラーで復帰する。

8.

名前	int WINAPI AT_GetKougotaiTransMode(HIMC hIMC)
機能概要	口語体変換モードの読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... 現在の口語体変換モードの設定 AT_OFF ... OFF AT_ON ... ON
	異常終了 ... AT_NOTATOK 選択中のIMEがATOKでない
特記事項	

9.

名前	int WINAPI AT_SetFukugougoTransKind(HIMC hIMC, int mode)
機能概要	複合語変換モード詳細の設定
入力	HIMC hIMC 使用するInputContextハンドル int mode ... 設定する複合語変換モード詳細 ATFUKUGOUGO_TSUJYO 通常 ATFUKUGOUGO_JINMEIYUSEN 人名優先 ATFUKUGOUGO_CHIMEIYUSEN 地名優先
出力	正常終了 ... 以前の複合語変換モード詳細の設定 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	AT_ImmSetSentenceModeで ATFUKUGOUGO (複合語変換モード)にしたときに有効。 未確定文字がある場合、あるいは指定された値が範囲外の場合は、設定せずにエラーで復帰する。

10.

名前	int WINAPI AT_GetFukugougoTransKind(HIMC hIMC)
機能概要	複合語変換モード詳細の読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... 現在の複合語変換モード詳細の設定 ATFUKUGOUGO_TSUJYO 通常 ATFUKUGOUGO_JINMEIYUSEN 人名優先 ATFUKUGOUGO_CHIMEIYUSEN 地名優先
	異常終了 ... AT_NOTATOK 選択中のIMEがATOKでない
特記事項	

11.

名前	int WINAPI AT_SetDefaultDicNo(HIMC hIMC, int mode)
機能概要	基本辞書セット番号の設定
入力	HIMC hIMC 使用するInputContextハンドル int mode ... 設定する辞書セット番号 辞書1～辞書0 ATDICNO_1～ATDICNO_0
出力	正常終了 ... 以前の辞書セット番号 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	未確定文字がある場合、あるいは指定された値が範囲外の場合は、設定せずにエラーで復帰する。

12.

名前	int WINAPI AT_GetDefaultDicNo(HIMC hIMC)
機能概要	基本辞書セット番号の読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... 現在の辞書セット番号 辞書1～辞書0 ATDICNO_1～ATDICNO_0
	異常終了 ... AT_NOTATOK 選択中のIMEがATOKでない
特記事項	

13.

名前	int WINAPI AT_SetDicFileNameSet(HIMC hIMC, int fno, ATDICFILENAMESET *lpDicNamePack)
機能概要	辞書セットごとの辞書ファイル名設定
入力	HIMC hIMC 使用するInputContextハンドル
	int fno ... 辞書セット番号 辞書1～辞書0 ATDICNO_1～ATDICNO_0
	ATDICFILENAMESET *lpDicNamePack...辞書ファイル名設定構造体アドレス
出力	正常終了 ... AT_OK
	異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	ATDICFILENAMESETのメンバについて・・・付録4-4参照 <ul style="list-style-type: none"> ・設定内容は、ATDICFILENAME_MAXバイト以内のシフトJIS文字列である。 ・NULLポインタを設定した場合、設定されることはない(ただし、システム辞書ファイルパスの項目を除く)。 ・辞書セット内の一部の辞書ファイル名を変更する場合、AT_GetDicFileNameSetにより取得した後、変更する。 未確定文字がある場合、あるいは指定された値が範囲外の場合は、設定せずにエラーで復帰する。

14.

名前	int WINAPI AT_GetDicFileNameSet(HIMC hIMC, int fno, ATDICFILENAMESET *lpDicNamePack)
機能概要	辞書セットごとの辞書ファイル名の読み出し
入力	HIMC hIMC 使用するInputContextハンドル
	int fno ... 辞書セット番号 辞書1～辞書0 ATDICNO_1～ATDICNO_0
	ATDICFILENAMESET *lpDicNamePack...辞書ファイル名設定構造体アドレス
出力	正常終了 ... AT_OK
	異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	ATDICFILENAMESETのメンバについて・・・付録4-4参照 <ul style="list-style-type: none"> ・設定される内容は、ATDICFILENAME_MAXバイト以内のシフトJIS文字列である。 指定された値が範囲外の場合は、設定せずにエラーで復帰する。

15.

名前	int WINAPI AT_SetDicLearnMode(HIMC hIMC, int fno, int mode)
機能概要	辞書セット番号ごとの学習モードの設定
入力	HIMC hIMC 使用するInputContextハンドル int fno ... 辞書セット番号 辞書1～辞書0 ATDICNO_1～ATDICNO_0 int mode ... 設定する学習モード AT_OFF ... OFF AT_ON ... ON
出力	正常終了 ... 以前の学習モード 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	未確定文字がある場合、あるいは指定された値が範囲外の場合は、設定せずにエラーで復帰する。

16.

名前	int WINAPI AT_GetDicLearnMode(HIMC hIMC, int fno)
機能概要	辞書セットごとの学習モードの読み出し
入力	HIMC hIMC 使用するInputContextハンドル int fno ... 辞書セット番号 辞書1～辞書0 ATDICNO_1～ATDICNO_0
出力	正常終了 ... 現在の学習モード AT_OFF ... OFF AT_ON ... ON 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	指定された値が範囲外の場合は、設定せずにエラーで復帰する。

17.

名前	int WINAPI AT_SetOkuriKind(HIMC hIMC, int mode)
機能概要	送りがなモード設定
入力	HIMC hIMC 使用するInputContextハンドル int mode ... 設定する送りがなモード ATOKURIGANA_HONSOKU ... 本則 ATOKURIGANA_HABUKU ... 省く ATOKURIGANA_OKURU ... 送る ATOKURIGANA_SUBETE ... すべて
出力	正常終了 ... 以前の送りがなモード 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	未確定文字がある場合、あるいは指定された値が範囲外の場合は、設定せずにエラーで復帰する。

18.

名前	int WINAPI AT_GetOkuriKind(HIMC hIMC)
機能概要	送りがなモードの読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... 現在の送りがなモード ATOKURIGANA_HONSOKU ... 本則 ATOKURIGANA_HABUKU ... 省く ATOKURIGANA_OKURU ... 送る ATOKURIGANA_SUBETE ... すべて 異常終了 ... AT_NOTATOK 選択中のIMEがATOKでない
特記事項	

19.

名前	int WINAPI AT_SetDicFileSetNickname(HIMC hIMC, int fno, LPWSTR lpStr)
機能概要	辞書セットごとの名称設定
入力	HIMC hIMC 使用するInputContextハンドル int fno ... 辞書セット番号 辞書1～辞書0 ATDICNO_1～ATDICNO_0 LPWSTR lpStr...辞書セットの名称アドレス
出力	正常終了 ... AT_OK 異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	設定する内容は、文字数 ATDICFILESETNICKNAME_MAX 以内のUnicode文字列である。 ...付録4 - 4 . 参照 未確定文字がある場合、あるいは指定された値が範囲外の場合は、設定せずにエラーで復帰する。

20.

名前	int WINAPI AT_GetDicFileSetNickname(HIMC hIMC, int fno, LPWSTR lpStr)
機能概要	辞書セットごとの名称の読み出し
入力	HIMC hIMC 使用するInputContextハンドル int fno ... 辞書セット番号 辞書1～辞書0 ATDICNO_1～ATDICNO_0 LPWSTR lpStr...辞書セットの名称アドレス
出力	正常終了 ... AT_OK 異常終了 ... AT_NOTATOK 選択中のIMEがATOKでない
特記事項	設定される内容は、文字数 ATDICFILESETNICKNAME_MAX 以内のUnicode文字列である。 ...付録4 - 4 . 参照 指定された値が範囲外の場合は、設定せずにエラーで復帰する。

21.

名前	int WINAPI AT_SendMouseEvent(HIMC hIMC, int fnc, int posX, int posXinf)
機能概要	未確定文字操作に関するマウスイベントを送る
入力	HIMC hIMC 使用するInputContextハンドル
	int fnc マウスイベント(定義ATMSE_*参照)
	int posX 未確定文字先頭からの文字位置(1~100)
	int posXinf 文字位置詳細情報 ATMSP_FCHAR 文字位置前半 ATMSP_BCHAR 文字位置後半
出力	正常終了 ... AT_OK
	異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	ATOK11から有効。 マウスイベント処理関係の定義は、付録4 - 6 . を参照。

22.

名前	int WINAPI AT_SendVkeyEvent(HIMC hIMC, int Vkeyno)
機能概要	ATOK機能キーイベントを送る
入力	HIMC hIMC 使用するInputContextハンドル
	int fnc ATOK機能キー番号.
出力	正常終了 ... AT_OK
	異常終了 ... AT_FAIL 設定エラー AT_NOTATOK 選択中のIMEがATOKでない
特記事項	ATOK11から有効。 ATOK機能キー番号は、付録4 - 7 . を参照。

23.

名前	int WINAPI AT_GetIMECompCollInfo(HIMC, ATIMECOMPCOL_TBL *)
機能概要	IME未確定文字列表示色情報の取得
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... AT_OK
特記事項	sizeof(ATIMECOMPCOL_TBL)の表示色情報がコピーされる。 未確定文字表示色構造体定義は付録4 - 3 . を参照。

24.

名前	BOOL WINAPI AT_ImmGetAtCompositionPacket(HIMC hIMC, DWORD dwIndex, LPATOKCOMPOSITIONPACK lpATOKCompositionPack)
機能概要	ATOK Composition情報の一括取得
入力	HIMC hIMC 使用するInputContextハンドル
	DWORD dwIndex 取得するComposition情報 (要求内容はIMM.H定義のGCS_xxに準拠する)
	LPATOKCOMPOSITIONPACK lpATOKCompositionPack ... 取得するComposition情報格納エリアのポインタ
出力	正常終了 ... TRUE
	異常終了 ... FALSE
特記事項	ATOKCOMPOSITIONPACKの詳細は、付録4 - 2 . を参照。 取得データはUnicode文字列で格納される。

25.

名前	BOOL WINAPI AT_ImmGetKanjiOn(HIMC hIMC)
機能概要	漢字ON状態の取得
入力	HIMC hIMC ...使用するInputContextハンドル
出力	現在の漢字ON/OFF状態 FALSE ... OFF TRUE ... ON
特記事項	

26.

名前	int WINAPI AT_ImmSetKanjiOn(HIMC hIMC)
機能概要	漢字ON
入力	HIMC hIMC ...使用するInputContextハンドル
出力	正常終了 ...以前の漢字ON/OFF状態 AT_OFF ... OFF AT_ON ... ON 異常終了 ...AT_FAIL 設定エラー
特記事項	

27.

名前	int WINAPI AT_ImmSetKanjiOff(HIMC hIMC)
機能概要	漢字OFF
入力	HIMC hIMC ...使用するInputContextハンドル
出力	正常終了 ...以前の漢字On/Off状態 AT_OFF ... OFF AT_ON ... ON 異常終了 ...AT_FAIL 設定エラー
特記事項	

28.

名前	int WINAPI AT_ImmSetSentenceMode(HIMC hIMC, int mode)
機能概要	変換モード設定
入力	HIMC hIMC ... 使用するInputContextハンドル int mode ... 設定する変換モード ATRENBUNSETSU ... 連文節 ATFUKUGOUGO ... 複合語 ATJIDOHENKAN ... 自動
出力	正常終了 ... 以前の変換モード 異常終了 ... AT_FAIL 設定エラー
特記事項	未確定文字がある場合、あるいは指定された値が範囲外の場合は、設定せずにエラーで復帰する。

29.

名前	int WINAPI AT_ImmGetSentenceMode(HIMC hIMC)
機能概要	変換モードの読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... 現在の変換モード ATRENBUNSETSU ... 連文節 ATFUKUGOUGO ... 複合語 ATJIDOHENKAN ... 自動 異常終了 ... AT_FAIL 設定エラー
特記事項	

30.

名前	int WINAPI AT_ImmSetRomanMode(HIMC hIMC, int mode)
機能概要	設定する漢字入力モード
入力	HIMC hIMC 使用するInputContextハンドル int mode ... 設定する漢字入力モード ATKANJI_ROMAJI ... ローマ字漢字 ATKANJI_KANA ... カナ漢字
出力	正常終了 ... 以前の漢字入力モード 異常終了 ... AT_FAIL 設定エラー
特記事項	未確定文字がある場合、あるいは指定された値が範囲外の場合は、設定せずにエラーで復帰する。

31.

名前	int WINAPI AT_ImmGetRomanMode(HIMC hIMC)
機能概要	漢字入力モードの読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	正常終了 ... 現在の漢字入力モード ATKANJI_ROMAJI ... ローマ字漢字 ATKANJI_KANA ... カナ漢字 異常終了 ... AT_FAIL 設定エラー
特記事項	

32.

名前	int WINAPI AT_ImmSetKanjiInputMode(HIMC hIMC, int mode)
機能概要	入力文字種の設定
入力	HIMC hIMC 使用するInputContextハンドル int mode ... 設定する入力文字種 ATHIRAGANA ... 全角ひらがな ATZENKAKU_KATAKANA ... 全角カタカナ ATZENKAKU_EISU ... 全角英数 ATHANKAKU_KATAKANA ... 半角カタカナ ATHANKAKU_EISU ... 半角英数
出力	正常終了 ... 以前の入力文字種 異常終了 ... AT_FAIL 設定エラー
特記事項	

33.

名前	int WINAPI AT_ImmGetKanjiInputMode(HIMC hIMC)
機能概要	入力文字種の読み出し
入力	HIMC hIMC 使用するInputContextハンドル
出力	<p>正常終了 ... 現在の入力文字種</p> <p>ATHIRAGANA ... 全角ひらがな</p> <p>ATZENKAKU_KATAKANA ... 全角カタカナ</p> <p>ATZENKAKU_EISU ... 全角英数</p> <p>ATHANKAKU_KATAKANA ... 半角カタカナ</p> <p>ATHANKAKU_EISU ... 半角英数</p> <p>異常終了 ... AT_FAIL 設定エラー</p>
特記事項	

34.

名前	int WINAPI AT_ImmSetInputModeEx(HIMC hIMC, int mode)	
機能概要	入力モードの設定	
入力	HIMC hIMC	使用するInputContextハンドル
	int mode ... 設定する入力モード	
	ATKANJIINPUT	現在設定中の文字種に従う漢字入力モード
	ATKANJIINPUT_HIRAGANA	全角ひらがな文字種(あ)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_KATAKANA	全角カタカナ文字種(ア)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_EISU	全角無変換文字種(A)の漢字入力モード
	ATKANJIINPUT_HANKAKU_KATAKANA	半角カタカナ文字種(ア)の漢字入力モード
	ATKANJIINPUT_HANKAKU_EISU	半角英数文字種(A)の漢字入力モード
	ATKANJIINPUT_HIRAGANA_KOTEI	全角ひらがな固定(あ)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_KATAKANA_KOTEI	全角カタカナ固定(ア)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_EISU_KOTEI	全角英数固定(A)の漢字入力モード
	ATKANJIINPUT_HANKAKU_KATAKANA_KOTEI	半角カタカナ固定(ア)の漢字入力モード
	ATKANJIINPUT_HANKAKU_EISU_KOTEI	半角英数固定(A)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_EIJI	全角英字入力(*)の漢字入力モード
	ATKANJIINPUT_HANKAKU_EIJI	半角英字入力(*)の漢字入力モード
	ATHANKAKUINPUT	現在設定中半角モードに従う半角入力モード
	ATHANKAKUINPUT_TUJYO	通常半角入力モード
	ATHANKAKUINPUT_HATSUON	発音半角入力モード
	ATHANKAKUINPUT_DOITSUGO	独語半角入力モード
	ATHANKAKUINPUT_FRANSUGO	仏語半角入力モード
	ATCODEINPUT	コード入力モード
	ATKIGOUINPUT	記号入力モード
出力	正常終了 ... 以前の入力モード	
	ATKANJIINPUT_HIRAGANA	全角ひらがな文字種(あ)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_KATAKANA	全角カタカナ文字種(ア)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_EISU	全角英数文字種(A)の漢字入力モード
	ATKANJIINPUT_HANKAKU_KATAKANA	半角カタカナ文字種(ア)の漢字入力モード
	ATKANJIINPUT_HANKAKU_EISU	半角英数文字種(A)の漢字入力モード
	ATKANJIINPUT_HIRAGANA_KOTEI	全角ひらがな固定(あ)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_KATAKANA_KOTEI	全角カタカナ固定(ア)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_EISU_KOTEI	全角英数固定(A)の漢字入力モード
	ATKANJIINPUT_HANKAKU_KATAKANA_KOTEI	半角カタカナ固定(ア)の漢字入力モード
	ATKANJIINPUT_HANKAKU_EISU_KOTEI	半角英数固定(A)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_EIJI	全角英字入力(*)の漢字入力モード
	ATKANJIINPUT_HANKAKU_EIJI	半角英字入力(*)の漢字入力モード
	ATHANKAKUINPUT_TUJYO	通常半角入力モード
	ATHANKAKUINPUT_HATSUON	発音半角入力モード
	ATHANKAKUINPUT_DOITSUGO	独語半角入力モード
	ATHANKAKUINPUT_FRANSUGO	仏語半角入力モード
	ATCODEINPUT	コード入力モード
	ATKIGOUINPUT	記号入力モード
	異常終了 ... AT_FAIL	設定エラー
特記事項		

35.

名前	int WINAPI AT_ImmGetInputModeEx(HIMC hIMC)	
機能概要	入力モードの読み出し	
入力	HIMC hIMC 使用するInputContextハンドル	
出力	正常終了 ... 現在の入力モード	
	ATKANJIINPUT_HIRAGANA	全角ひらがな文字種(あ)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_KATAKANA	全角カタカナ文字種(ア)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_EISU	全角英数文字種(A)の漢字入力モード
	ATKANJIINPUT_HANKAKU_KATAKANA	半角カタカナ文字種(ア)の漢字入力モード
	ATKANJIINPUT_HANKAKU_EISU	半角英数文字種(A)の漢字入力モード
	ATKANJIINPUT_HIRAGANA_KOTEI	全角ひらがな固定(あ)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_KATAKANA_KOTEI	全角カタカナ固定(ア)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_EISU_KOTEI	全角英数固定(A)の漢字入力モード
	ATKANJIINPUT_HANKAKU_KATAKANA_KOTEI	半角カタカナ固定(ア)の漢字入力モード
	ATKANJIINPUT_HANKAKU_EISU_KOTEI	半角英数固定(A)の漢字入力モード
	ATKANJIINPUT_ZENKAKU_EIJI	全角英字入力(*)の漢字入力モード
	ATKANJIINPUT_HANKAKU_EIJI	半角英字入力(*)の漢字入力モード
	ATHANKAKUINPUT_TUJYO	通常半角入力モード
	ATHANKAKUINPUT_HATSUON	発音半角入力モード
	ATHANKAKUINPUT_DOITSUGO	独語半角入力モード
	ATHANKAKUINPUT_FRANSUGO	仏語半角入力モード
	ATCODEINPUT	コード入力モード
	ATKIGOUIINPUT	記号入力モード
	異常終了 ... AT_FAIL 設定エラー	
特記事項		

36.

名前	BOOL WINAPI AT_ImmRegisterWord(HIMC hIMC, WCHAR *lpwzRegReading, WCHAR *lpwzRegString, DWORD dwStyle, DWORD *lpErrorInf)	
機能概要	単語登録	
入力	HIMC hIMC 使用するInputContextハンドル	
	WCHAR *lpwzRegReading	読み文字列(Unicode文字列)アドレス
	WCHAR *lpwzRegWord	表記文字列(Unicode文字列)アドレス
	DWORD dwStyle	品詞番号
	DWORD *lpErrorInf	エラー詳細情報格納アドレス
出力	正常終了 ... TRUE	
	異常終了 ... FALSE	
特記事項	<p>ATOKの単語登録ダイアログを開くことなく単語登録を実行する。 品詞番号はAT_ImmGetRegisterWordStyleで取得した品詞対応表から選択する。 エラー詳細情報は、付録4 - 5 を参照。 エラー詳細情報は、ATOK以外を選択中は常に0である。</p>	

37.

名前	BOOL WINAPI AT_ImmUnregisterWord(HIMC hIMC, WCHAR *lpwzRegReading, WCHAR *lpwzRegString, DWORD dwStyle, DWORD *lpErrorInf)
機能概要	単語削除
入力	HIMC hIMC 使用するInputContextハンドル WCHAR *lpwzRegReading 読み文字列(Unicode文字列)アドレス WCHAR *lpwzRegWord 表記文字列(Unicode文字列)アドレス DWORD dwStyle 品詞番号 DWORD *lpErrorInf エラー詳細情報格納アドレス
出力	正常終了 ... TRUE 異常終了 ... FALSE
特記事項	ATOKのダイアログを開くことなく単語削除を実行する。 品詞番号はAT_ImmGetRegisterWordStyleで取得した品詞対応表から選択する。 ATOK選択中の場合、品詞番号を ATDELETEALLHINSHI と設定することで、該当する全品詞を対象とすることができる。 エラー詳細情報は、付録4 - 5 . を参照。

38.

名前	UINT WINAPI AT_ImmGetRegisterWordStyle(UINT nItem, LPATREGSTYLEBUF lpAtRegStyleBuf)
機能概要	単語品詞総数を取得する(nItemが0の場合) の単語品詞番号と品詞名の対応表を取得する (品詞名はUnicode文字列で返る)
入力	UINT nItem 対応表の受け入れ数 LPATREGSTYLEBUF lpAtRegStyleBuf 対応表格納バッファアドレス
出力	コピーされた品詞対応表の数
特記事項	対応表取得にはnItem*sizeof(ATREGSTYLEBUF)バイトの領域を必要とする。 付録4 - 8 . を参照のこと ATOK10/ATOK11は同じ品詞体系。品詞名の一覧は、付録4 - 9 . を参照。

3 . サンプル

3 - 1 . 未確定文字列表示をアプリケーション側で行う

```
#include "atoklib.h"

case WM_IME_STARTCOMPOSITION:
    //(未)確定文字表示するための開始処理
    .....
    return( 0 );

case WM_IME_COMPOSITION:
    hIMC = ImmGetContext(hWnd);
    //(未)確定文字列更新情報を取得する
    AT_ImmGetAtCompositionPacket(hIMC, lParam, &ATOKCompositionPack);
    //カーソル位置情報をもとに次候補表示位置を通知する
    ImmSetCandidateWindow(hImc, ..);
    ImmReleaseContext(hWnd, hIMC);
    //取得した(未)確定文字列情報を表示する
    .....
    return( 0 );

case WM_IME_ENDCOMPOSITION:
    //(未)確定文字表示するための終了処理
    .....
    return( 0 );
```

3 - 2 . 未確定文字列表示はIME に行わせる

```
#include "atoklib.h"

case WM_IME_COMPOSITION:
    if ( IParam&GCS_RESULTSTR )
        { //確定文字列の出力があった
          hIMC = ImmGetContext(hWnd);
          //確定文字列更新情報を取得する
          AT_ImmGetAtCompositionPacket( hIMC, IParam,
            &ATOKCompositionPack );
          //取得した確定文字列情報を表示する
          .....
          //未確定文字表示位置を通知する
          ImmSetCompositionWindow( hIMC,...);
          ImmReleaseContext(hWnd, hIMC);
          //確定文字列はここで処理した
          IParam &= ~( GCS_RESULTREADSTR |
            GCS_RESULTREADCLAUSE |
            GCS_RESULTSTR | GCS_RESULTCLAUSE );
        }
    return( DefWindowProc(hWnd, message, wParam, IParam) );
```

4 . 付 録

4 - 1 . ATOKComposition 取得関数

AT_ImmGetAtCompositionPacket ;ATOKCompositionパケットの取得

AT_ImmGetAtCompositionPacket は ImmGetComposition の Unicode 対応版として提供されている。ImmGetComposition は Composition の要素ごとに取得する仕様であるのに対し AT_ImmGetAtCompositionPacketComposition は要素全体を一括取得する。

ATOK の場合、この関数を使用することで、Windows95 上でも Unicode 文字列を取得することができる。

4 - 2 . ATOK Composition 構造体定義

//構造体バッファサイズ定義

```
#define ATRESULTSTRCTR_MAX 100 // 確定文字列表記数Max
#define ATRESULTREADCTR_MAX 100 // 確定文字列読み数Max
#define ATRESULTCLAUSECTR_MAX 100 // 確定文字列文節数Max
#define ATCOMPOSITIONSTRCTR_MAX 100 // 未確定文字列表記数Max
#define ATCOMPOSITIONREADCTR_MAX 100 // 未確定文字列読み数Max
#define ATCOMPOSITIONCLAUSECTR_MAX 100 // 未確定文字列文節数Max
#define ATRSLTSTRB_MAX 128
#define ATRSLTREADB_MAX 128
#define ATRSLTCLSB_MAX 128
#define ATCOMPSTRB_MAX 128
#define ATCOMPREADB_MAX 128
#define ATCOMPCLSB_MAX 128
```

//AtChangeMode(文字列更新情報)構成要素

```
#define ATGCS_COMPREADSTR GCS_COMPREADSTR // 未確定文字列読み更新
#define ATGCS_COMPREADATTR GCS_COMPREADATTR // 未確定文字列属性更新
#define ATGCS_COMPREADCLAUSE GCS_COMPREADCLAUSE // 未確定文字列文節区切更新
#define ATGCS_COMPSTR GCS_COMPSTR // 未確定文字列表記更新
#define ATGCS_COMPATTR GCS_COMPATTR // 未確定文字列属性更新
#define ATGCS_COMPCLAUSE GCS_COMPCLAUSE // 未確定文字列文節区切更新
#define ATGCS_CURSORPOS GCS_CURSORPOS // 未確定文字列カーソル位置更新
#define ATGCS_DELTASTART GCS_DELTASTART // 未確定文字列表示変更位置更新
#define ATGCS_RESULTREADSTR GCS_RESULTREADSTR // 確定文字列読み更新
#define ATGCS_RESULTREADCLAUSE GCS_RESULTREADCLAUSE // 確定文字列文節区切り更新
#define ATGCS_RESULTSTR GCS_RESULTSTR // 確定文字列表記更新
#define ATGCS_RESULTCLAUSE GCS_RESULTCLAUSE // 確定文字列文節区切り更新
```

```
//AtCompositionStrAttr(未確定文字列表記属性)構成要素
//他社IMEの場合、ATCOLINDX_INPUTKOTEI ~ は返りません
//カーソル色は返りません(デフォルト)
#define ATCOLINDX_INPUT          0x00    // 変換可能入力文字色
#define ATCOLINDX_TARGETCONVERT  0x01    // 変換文字色
#define ATCOLINDX_CONVERTED      0x02    // 変換済み注目文節色
#define ATCOLINDX_TARGETNOTCONVERTED 0x03    // 未変換注目文節色
#define ATCOLINDX_INPUT_ERROR    0x04    // 入力文字エラー色
#define ATCOLINDX_INPUTKOTEI     0x06    // 固定入力文字色
#define ATCOLINDX_TARGETNOTCONVERTEDKOTEI 0x08    // 固定入力文節色
#define ATCOLINDX_TARGETCOMMENT  0x09    // 注目文節コメント色
#define ATCOLINDX_COMMENT        0x0a    // 文節コメント色
```

```
//AtResultPrvFaceNameFlg(確定文字フォント書体指定有無)構成要素
//他社IMEの場合、常にATRESULTPRVFONT_FALSE
#define ATRESULTPRVFONT_FALSE    0      // 無し
#define ATRESULTPRVFONT_TRUE     1      // 有り
```

```
//ATOKCOMPOSITIONPACK構造体
typedef struct{
    int    AtChangeMode                ;// 文字列更新情報
    int    AtCompositionStrCtr         ;// 未確定文字列表記数
    int    AtCompositionStrClsCtr      ;// 未確定文字列表記文節数
    int    AtCompositionReadCtr       ;// 未確定文字列読み数
    int    AtCompositionReadClsCtr    ;// 未確定文字列読み文節数
    int    AtCompositionCursorPos     ;// 未確定文字列カーソル位置
    int    AtCompositionDeltaStart    ;// 未確定文字列変更位置
    int    AtResultStrCtr              ;// 確定文字列表記数
    int    AtResultReadCtr            ;// 確定文字列読み数
    int    AtResultStrClsCtr          ;// 確定文字列表記文節数
    int    AtResultReadClsCtr        ;// 確定文字列読み文節数
    int    AtResultPrvFaceNameFlg     ;// 確定文字フォント書体指定有無
    int    AtReserveAreaA[64-12]     ;// リザーブ

    WCHAR  AtCompositionStr[ATCOMPSTRB_MAX] ;// 未確定文字列表記バッファ
    WORD   AtCompositionStrAttr[ATCOMPSTRB_MAX] ;// 未確定文字列表記属性バッファ
    DWORD  AtCompositionStrCls[ATCOMPCLSB_MAX] ;// 未確定文字列表記文節バッファ
    WCHAR  AtCompositionRead[ATCOMPREADB_MAX] ;// 未確定文字列読みバッファ
    WORD   AtCompositionReadAttr[ATCOMPREADB_MAX] ;// 未確定文字列読み属性バッファ
    DWORD  AtCompositionReadCls[ATCOMPCLSB_MAX] ;// 未確定文字列読み文節バッファ

    WCHAR  AtResultStr[ATRSLTSTRB_MAX] ;// 確定文字列表記バッファ
    DWORD  AtResultStrCls[ATRSLTCLSB_MAX] ;// 確定文字列表記文節バッファ
    WCHAR  AtResultRead[ATRSLTREADB_MAX] ;// 確定文字列読みバッファ
    DWORD  AtResultReadCls[ATRSLTCLSB_MAX] ;// 確定文字列読み文節バッファ

    WORD   AtReserveAreaC[ATRSLTCLSB_MAX] ;// リザーブ
    WCHAR  AtResultPrvFaceName[LF_FACESIZE] ;// 確定文字フォント書体名

    BYTE   AtReserveAreaB[1024-64] ;// リザーブ
} ATOKCOMPOSITIONPACK, *PATOKCOMPOSITIONPACK, *LPATOKCOMPOSITIONPACK;
```

4 - 3 . 未確定文字表示色構造体定義

```
#define ATIMECOMPCL_ITEMMAX 16 // 未確定文字表示項目数
// 項目は後述のAtCompositionStrAttrに従い
// ATOK以外のIMEではATCOLINDX_INPUTKOTEI ~は
// 使用しません

typedef struct{ // 表示色アトリビュート構造体
    COLORREF    rgbBack; // 背景色
    COLORREF    rgbText; // 文字色
    BOOL        fUnderLine; // 下線表示有無
} ATIMECOL_STRC;
typedef struct{
    ATIMECOL_STRC  ATIMECompCol[ATIMECOMPCL_ITEMMAX];
} ATIMECOMPCL_TBL, *PATIMECOMPCL_TBL, *LPATIMECOMPCL_TBL;
```

4 - 4 . ATOK 辞書設定用構造体定義

```
//辞書セット番号
#define ATDICNO_1          0          // 辞書セット番号：辞書 1
#define ATDICNO_2          1          // 辞書セット番号：辞書 2
#define ATDICNO_3          2          // 辞書セット番号：辞書 3
#define ATDICNO_4          3          // 辞書セット番号：辞書 4
#define ATDICNO_5          4          // 辞書セット番号：辞書 5
#define ATDICNO_6          5          // 辞書セット番号：辞書 6
#define ATDICNO_7          6          // 辞書セット番号：辞書 7
#define ATDICNO_8          7          // 辞書セット番号：辞書 8
#define ATDICNO_9          8          // 辞書セット番号：辞書 9
#define ATDICNO_0          9          // 辞書セット番号：辞書 0

#define ATASSISTDICMAX     4          // 補助辞書数Max
#define ATDICFILENAME_MAX (256+2) // 辞書ファイル名バイト数Max
#define ATDICFILESETNICKNAME_MAX (80+1) // 辞書セット名文字数Max

// 辞書ファイル設定構造体
typedef struct tagATDICFILENAMESET {
    UCHAR    *pszUserDicName;          // ユーザ辞書ファイルパス
    UCHAR    *pszSystemDicName;       // システム辞書ファイルパス
    UCHAR    *pszAssistDicName[ATASSISTDICMAX]; // 補助辞書ファイルパス
} ATDICFILENAMESET;
```

4 - 5 . 辞書エラー種類一覧

```
enum ATAppError {
    ATREG_ERROR_RECORD_FULL = 1,      // 同じ読みの単語がいっぱい
    ATREG_ERROR_YOMI_FULL,           // この読みではこれ以上登録できない
    ATREG_ERROR_FILE_FULL,           // これ以上登録できない
    ATREG_ERROR_ABORT,               // 辞書が正しく設定されていないので登録を中止
    ATREG_ERROR_HYOKI_ERROR,         // 単語に不適切な文字が含まれている
    ATREG_ERROR_CANT_APPEND,         // 登録できなかった
    ATREG_ERROR_YOMI_OVER,           // 読みが長すぎる
    ATREG_ERROR_YOMI_ERROR,         // 読みにも不適切な文字が含まれている
    ATREG_ERROR_APPEND_NOW,         // すでに登録されている
    ATREG_ERROR_YOMI_TOP,            // 読みの先頭文字が不適切
};
```

4 - 6 . ATOKマウスイベント処理関係定義

```
// イベントは(ボタン状態 + キーボードシフト状態)で定義される
// 左ボタン状態
#define ATMSE_LDOWN          0x0001          // 押す
#define ATMSE_LUP           0x0002          // 離す
#define ATMSE_LDBLCLK       0x0003          // ダブルクリック
#define ATMSE_LDRAG         0x0004          // ドラッグ
// 右ボタン状態
#define ATMSE_RDOWN          0x0011          // 押す
#define ATMSE_RUP           0x0012          // 離す
#define ATMSE_RDBLCLK       0x0013          // ダブルクリック
#define ATMSE_RDRAG         0x0014          // ドラッグ
// 中央ボタン状態
#define ATMSE_MDOWN          0x0021          // 押す
#define ATMSE_MUP           0x0022          // 離す
#define ATMSE_MDBLCLK       0x0023          // ダブルクリック
#define ATMSE_MDRAG         0x0024          // ドラッグ
// キーボードシフト状態
#define ATMSE_KEYSHIFT      0x0100          // SHIFTキー同時押下
#define ATMSE_KEYCTRL       0x0200          // CTRLキー同時押下
// 文字位置詳細
#define ATMSP_FCHAR          0x0000          // 文字位置前半
#define ATMSP_BCHAR          0x0001          // 文字位置後半
```

4 - 7 . ATOK機能キー番号定義

```
#define ATKEY_AMET01        0x91            // AMET1起動
#define ATKEY_AMET02        0x92            // AMET2起動
#define ATKEY_AMET03        0x93            // AMET3起動
#define ATKEY_AMET04        0x94            // AMET4起動
#define ATKEY_AMET05        0x95            // AMET5起動
#define ATKEY_AMET06        0x96            // AMET6起動
#define ATKEY_AMET07        0x97            // AMET7起動
#define ATKEY_AMET08        0x98            // AMET8起動
#define ATKEY_AMET09        0x99            // AMET9起動
#define ATKEY_AMET10        0x9a            // AMET0起動
#define ATKEY_HELP          0xb0            // ヘルプ起動
#define ATKEY_PROPERTY       0xb2            // プロパティ起動
#define ATKEY_DICUT          0xb3            // 辞書ユーティリティ起動
#define ATKEY_CHARPALLET    0xb4            // 文字パレット起動
#define ATKEY_CLICKPALLET   0xb6            // クリックパレット起動
#define ATKEY_RESULTPALLET  0xb7            // 文字列ボックス起動
```

4 - 8 . 単語登録品詞取得用構造体定義

```
typedef struct tagATREGSTYLEBUF {
    DWORD          dwStyle; // 品詞番号
    WCHAR          szDescription[STYLE_DESCRIPTION_SIZE]; // 品詞名称
} ATREGSTYLEBUF, *PATREGSTYLEBUF, *LPATREGSTYLEBUF;
```

4 - 9 . 品詞名一覧

品詞名	品詞の説明	例
名詞	物や物の状態に付ける一般的な名詞	花、鳥
固有人姓	人の姓に付ける固有の名詞	後藤、村井
固有人名	人の名に付ける固有の名詞	和之、友子
固有人他	外国人名など、固有人姓・固有人名以外に付ける固有の名詞	ガリバー、アリス
固有地名	場所に付ける固有の名詞	徳島市、シドニー
固有組織	組織に付ける固有の名詞	ジャストシステム
固有商品	商品に付ける固有の名詞	一太郎、ATOK
固有一般名詞	人名・場所・組織・商品以外に付ける固有の名詞	天の川、ペガサス
名詞サ変	後にサ変動詞「する」を付けることができる名詞	演出-する、援助-する
名詞ザ変	後にザ変動詞「ずる」を付けることができる名詞	念-ずる、命-ずる
名詞形動	物事の姿・状態・性質などを表し、後ろに「だ」・「な」を付けることができる名詞	画期的-だ(な)、大切-だ(な)
名ザ形動	名詞・サ変・形容動詞の要素を持っており、後ろに「する」・「な」を付けることができる名詞	満足-する(な)、いたずら-する(な)
数詞	数量・順序などを数えて示すことば	1、壹
副詞	主に用言(動詞・形容詞・形容動詞)を修飾することば	たいへん、ゆっくり
連体詞	活用がなく、体言だけを修飾することば	あの、この
接続詞	語句を続けるために使うことば	そして、しかし
感動詞	感動・応答・呼びかけなどを表すことば	ありがとう、やあ
独立語	助詞・助動詞が付かず他のことばとつながりを持たないことば	敬具、かしこ
接頭語	他のことばの前に付けて使うことば	御、ご、無
冠数詞	数字の前に付けて使うことば	第、午前
接尾語	他のことばの後に付けて使うことば	後、様
助数詞	数字に付けて種類や単位を表すことば	週、グラム
力行五段	力行の五段活用する動詞	書く、動く
カ行五段	カ行の五段活用する動詞	急ぐ、泳ぐ
サ行五段	サ行の五段活用する動詞	示す、消す
タ行五段	タ行の五段活用する動詞	立つ、勝つ
ナ行五段	ナ行の五段活用する動詞	死ぬ
ハ行五段	ハ行の五段活用する動詞	飛ぶ、遊ぶ
マ行五段	マ行の五段活用する動詞	生む、読む
ラ行五段	ラ行の五段活用する動詞	売る、走る
ワ行五段	ワ行の五段活用する動詞	会う、笑う
ハ行四段	ハ行の四段活用する動詞	言ふ、笑ふ
一段動詞	上一段、下一段活用する動詞	着る、始める
カ変動詞	カ行変格活用する動詞	来る
サ変動詞	サ行変格活用する動詞	熱する、察する
ザ変動詞	ザ行変格活用する動詞	軽んずる、甘んずる
形容詞	物事の性質・状態を表し、言い切るときの形が「い」で終わることば	美しい、広い
形容詞ウ	形容詞の中で語尾がウ音便化したときに語幹の形が変わることば	高う、長う
形容動詞	物事の性質・状態を表し、言い切るときの形が「だ」・「な」で終わることば	豊かだ(な)、うららかだ(な)
形動タリ	物事の性質・状態を表し、語尾が「と」「たる」に活用することば	燦然と(たる)、平然と(たる)
単漢字	助詞・助動詞がつかず、ほかの品詞としての要素を持たない1文字の漢字	垂、以、宇

以上