

〔非公開〕

TR-C-0040

通信技術文書体系化システム

島 健一
KENICHI SHIMA

1990. 2. 6

ATR通信システム研究所

通信技術文書体系化システム

島 健一

1990年2月6日

もくじ

1	はじめに	2
2	通信技術文書体系化システム COSMOS の起動法	3
2.1	機能概要	3
2.2	保存場所	3
2.3	起動方法	4
3	通信技術文書体系化システム デモ (SUCCESS) の起動法	5
3.1	機能概要	5
3.2	保存場所	5
3.3	起動方法	6
4	COSMOS 初期操作環境の作成	7
4.1	本の作成 ～ COSMOS 原始文書の作成 ～	7
4.2	重要語の切り出し	7
4.3	文間リンク (参照) の獲得	7
4.4	COSMOS 参照機能実現オブジェクトの作成	7
5	COSMOS 基本オブジェクトの獲得	8
5.1	「キーワード」と「視点」を設定する	8
5.1.1	「キーワード」と「視点」とは	8
5.1.2	「キーワード」と「視点」を作成する方法	8
5.1.3	「キーワード」と「視点」を作成する方法 ～ 自動作成 ～	10
5.1.4	「キーワード」と「視点」の変更と取消方法	11
5.2	「キーワード」を体系化する ～ キーワードリンク ～	13
5.2.1	「キーワードリンク」とは	13
5.2.2	「キーワードリンク」を作成する方法	13
5.2.3	「キーワードリンク」の取消方法	14
5.3	文書に関係付ける ～ 文間リンク ～	14
5.3.1	文間リンクとは	14
5.3.2	文間リンクを作成する方法	14
5.3.3	「文間リンク」の取消方法	15
6	ウィンドウ操作	16
6.1	共通操作	16
6.1.1	マウス操作	16

6.1.2	キーボード操作	18
6.2	文書ウィンドウ (Section Window)	18
6.2.1	タイトルエリア	18
6.2.2	ディスプレイエリア	18
6.2.3	コマンドメニューエリア	19
6.2.4	コマンド入力エリア	20
6.3	キーワードウィンドウ	21
6.3.1	ディスプレイエリア	21
6.3.2	コマンドメニュー	21
6.3.3	コマンド入力エリア	22
6.3.4	マウスドキュメンテーションライン	22
6.4	要約検索ウィンドウ	23
6.4.1	コマンド実行結果表示エリア	23
6.4.2	コマンドメニューエリア	23
6.4.3	コマンド入力エリア	24
6.5	ルールウィンドウ	25
6.6	疑問点ウィンドウ	25
6.7	キーワードエディタ	26
6.7.1	ウィンドウの各エリアの説明	26
6.7.2	操作方法～マウスによる操作～	26
6.7.3	操作方法～コマンドメニューからの操作～	27
7	要約検索	29
7.1	検索ルールの記述	29
7.1.1	システム述語	29
7.1.2	ユーザ定義基本述語	31
8	本の読み履歴ケース	34
8.1	「本の読み履歴ケース」とは	34
8.2	操作方法	34
8.2.1	「見だし」ケースをオープンする	34
8.2.2	複数の「本の読み履歴ケース」を切替える	35
8.3	「本の読み履歴ケース」を作成する方法	35
8.3.1	「本の読み履歴ケース」の作成方法～その一～	35
8.3.2	「本の読み履歴ケース」の作成方法～その二～	36
9	COSMOS オブジェクト一覧	37
10	電子化ドキュメント化の仕様	53
10.1	はじめに	53
10.2	文書資料のワープロ化仕様	53
10.2.1	文書資料の変換	53
10.2.2	SDL 図のイベントオブジェクト仕様への変換	56
10.2.3	SDL 図式エディタを用いた変換	58

10.3	文分割プログラムについて	58
10.3.1	オリジナルの文章	60
10.3.2	変換後の文章	62
11	ドキュメント解析プログラムの起動法	63
11.1	機能概要	63
11.2	保存場所	63
11.3	起動方法	63
12	ドキュメント解析プログラムの構成	64
13	ドキュメント解析用ファイルの構造	65
14	プログラムの仕様説明	66
14.1	text-to-sentence テキスト・ファイル変換プログラム	67
14.1.1	形式	67
14.1.2	機能	67
14.1.3	戻り値	67
14.1.4	使用例	67
14.1.5	制限	67
14.2	make-word-address 単語のインデックス作成プログラム	68
14.2.1	形式	68
14.2.2	機能	68
14.2.3	戻り値	68
14.2.4	使用例	68
14.2.5	制限	68
14.3	make-figure-address 図表のインデックス作成プログラム	69
14.3.1	形式	69
14.3.2	機能	69
14.3.3	戻り値	69
14.3.4	使用例	69
14.3.5	制限	69

第 1 章

はじめに

通信分野における概念を表す用語を技術文書から体系的に集め、通信に関する知識を抽出することを支援するシステム COSMOS (COmmunication document SuMmarizing and Organizing System) の操作法について述べる。

このシステムは、ハイパーテキストを用いた通信技術文書 (CCITT 勧告書) の構造化、検索システムであり、通信分野の概念的な用語 (信号、状態など) の定義、用語間の階層関係を明らかにし、それをさまざまな視点から検索できる。

また、COSMOS は、章立てなどの文章構造の自動的な構造解析機能、参照関係の自動的な抽出機能、重要語彙・重要語句の抜き出し

(キーワード自動抽出) 機能、概念の階層化の際の相互関係の表示機能を持ち概念体系を作成する上での支援ができる。

ここでは、上記の研究で重要となる試作ツールの概要について述べる。特に、システムソフトウェアのインストール手順、操作法に付いて詳しく述べる。

また、合わせて、文書を電子化する方法、および、その文書の解析法に付いても述べる。これらは、基盤となるカードシステムの機能を用いて作成されている。テクニカルレポート『通信ソフトウェア設計支援システム』も合わせて参照されたい。

最後に、本試作ツール開発の機会を与えて下さった山下 紘一社長、竹中 豊文室長、門田 充弘元室長 (現 NTT 交換研究所) に感謝します。

また、本試作システムに関して討論していただいた通信ソフトウェア研究室の各位に感謝します。

第 2 章

通信技術文書体系化システム COSMOS の起動法

2.1 機能概要

通信技術のエキスパートの方が、電子化された文書（マニュアル）を理解し活用することを支援します。エキスパートの方はこれまでの自分の経験、知識を活用し、自分の立場、目的、視点でマニュアルを読み、理解することを自然に進めることができます。

一般に文書（マニュアル）を読み進める中で、大切な語句（キーワード）に下線を引きます。同じ様に、電子化されたマニュアルを本支援プログラムを使って下線を引きながら、キーワードを取り出し、柔軟に体系化します。

本支援プログラムの機能特徴は次の通りです。

1. 文章解析
2. 重要語自動きりだし
3. 参照関係構造解析
4. キーワード階層構造表示
5. キーワード検索

検索を柔軟にできるように検索用言語として述語型ルールを採用しました。利用者がルール式を書くことで、強力で柔軟な検索能力を利用して、適切な検索結果を得ることができます。

本支援プログラムで得られる「要約」は、エキスパートの方の「視点」に基づく「キーワード」体系です。全体の体系を把握することができる「キーワード」階層構造を見ながら、検索した「キーワード」に関連する文書を容易に確認することができます。

2.2 保存場所

cosmos 関連のファイルは論理パス名として card-system:cosmos; の下で管理されています。ATR 通信システム研究所では、その絶対パス名は

- clm01:>local>binary>cosmos>***.*.*
- clm13:>local>binary>cosmos>***.*.*

のどちらかが対応します。利用されるマシンのサーバマシンにより異なります。

以下で、個々のファイルのインストール先を述べます。

1. プログラム

card-system:cosmos;*.lisp

card-system:cosmos;*.bin

2. i-interface data

card-system:cosmos;card-system-data;i-interface.lisp

3. cosmos 環境生成のためのコントロールファイル

スクリプトファイルや辞書ファイルなどがある。

card-system:cosmos;control;*.*

4. 文間リンク獲得時の作業ディレクトリ

文間リンク自動獲得時に作業用として使われるディレクトリである。

card-system:cosmos;transaction;*.*

2.3 起動方法

1. load system cosmos

cosmos システム環境の設定がなされます。

2. load file sys:site;card-system.translations

3. load system chart-editor

chart-editor システム環境の設定がなされます。

4. create card system manager i-interface :system type cosmos

『i-interface 文書』用のマネージャを作成

5. select + A

『i-interface 文書』用のマネージャ画面の選択

6. データロード

データは次のものをロードして下さい。

card-system:cosmos;card-system-data;i-interface.lisp

第 3 章

通信技術文書体系化システム デモ (SUCCESS) の起動法

3.1 機能概要

平成元年6月の開所式のデモンストレーション効果をあげるために COSMOS システムの機能特徴を原始人をモデルにアイコン表現しています。

アイコンでは COSMOS の機能を次のステップで表現しています。

1. 文書の構造化 (input 原文)
2. キーワード自動抽出 (input 構造化文書 1)
3. 視点、キーワード付け (input 構造化文書 2)
4. キーワードの階層化 (input キーワード)
5. キーワードの体系化 (input 階層化キーワード)
6. 日本語による検索 (input 体系化キーワード)

注意事項

本プログラムは開所式以降のバージョンアップを行なっていません。バージョンアップの全ては COSMOS システムにて行なわれています。

3.2 保存場所

本プログラムの保存場所は次の通りです。

1. プログラム

```
clm13:>success>*.lisp  
clm13:>success>*.bin
```

2. i-interface data

```
clm13:>success>card-system-date>i-interface.lisp
```

3.3 起動方法

1. load system fed

FED (フロント管理) システム環境を設定します。

2. load system sdl-editor-card-system

card-system システム環境を設定します。

3. load file clm13:>success>system-compile.lisp

SUCCESS システム環境を設定します。

4. create card system manager i-interface :system type sdl-editor-card-system

『i-interface 文書』用のマネージャを作成

5. set package success

6. select + A

『i-interface 文書』用のマネージャ画面の選択

7. データロード

データは次のものをロードして下さい。

```
clm13:>success>card-system-data>i-interface.lisp
```

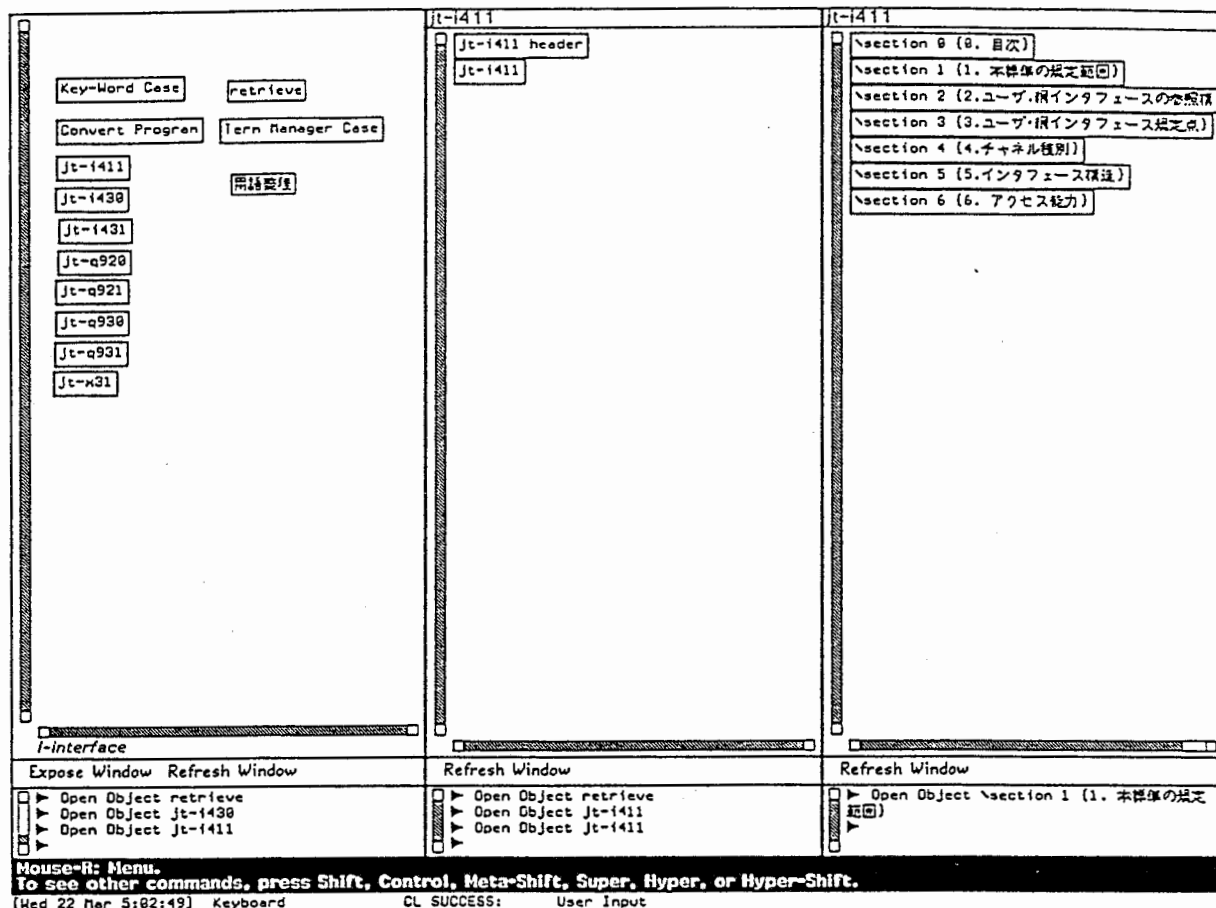
アイコンウィンドウを表示するまでの操作は次の通りです。

1. manager-window 上の「cosmos-demo-case」ケースオブジェクトをオープンする。

2. 「cosmos-demo-case」ケースウィンドウ上の「COSMOS Flow」オブジェクトをオープンする。

4. 2 カードシステムマネージャ

本支援プログラムのメインウィンドウです。全ての操作がここから始まります。
 Select + A キーでいつでも呼び出して表示させることができます。
 以下でウィンドウに表示されているものについて解説します。



ディスプレイエリア

文書名

作業対象となる文書名

Mouse-L クリックでボリュームマネージャを開きます。

Key-Word Case

Mouse-L クリックでキーワードウィンドウを開きます。

ルールガード

Mouse-L クリックで要約検索ルールを記述してあるルールウィンドウを開きます。

用語整理

Mouse-L クリックでキーワードの類似性を記述するウィンドウが開きます。

4. 5 文書ウィンドウ (Section Window)

「目次ウィンドウ」で目次の section 名を Mouse-L クリックすると開きます。作業をするための文書を表示します。

また、「文書ウィンドウ」では

1. 「キーワード」と「視点」の設定
2. 「キーワード」の体系化
3. 「パラグラフ」の関係付け

ができます。

以下でウィンドウに表示されているものを解説します。

The screenshot shows a window titled "Section Window" with a scrollable list of sections. The top section is "jt-1411 : 2.ユーザ網インタフェースの参照構成". It contains a paragraph about "参照構成" (Reference Structure) and a button labeled "参照構成". Below it is a subsection "subsection 2-1 (2.1 機能群)" with a paragraph about "機能群" (Function Groups) and a button labeled "機能群". This subsection contains two sub-subsections: "subsubsection 2-1-1 (2.1.1 網仕様(装置)1 (以下NT1と略記))" and "subsubsection 2-1-2 (2.1.2 網仕様(装置)2 (以下NT2と略記))". The first sub-subsection lists seven items: (1) 伝送路仕様, (2) レイヤ1 (伝送路保守) 機能及び(レイヤ1)に関する動作状態(監視), (3) タイミング, (4) 結電, (5) レイヤ1多変換, (6) レイヤ1融合判別, (7) (ポイント・ポイント), (ポイント・マルチポイント)(ユーザ・網インタフェース)(終端). The second sub-subsection contains text about OSI model layers and LAN/ATM protocols. At the bottom of the window, there is a menu bar with options like "Keyword View SW", "Next Section", "Prev Section", "Refresh Window", "Search Paragraph", "Swich Section", and "Window Full". Below the menu bar, there are command lines and a status bar with the text "Mouse-L: Show Section with this term; Mouse-R: Menu. To see other commands, press Shift, Control, Meta-Shift, Super, Hyper, or Hyper-Shift." and "CL SUCCESS: User Input".

タイトルエリア

文書名

作業している文書名を表示します。

section

表示している文書の section 名を表示します。

(作業過程で subsection や subsubsection などを開いて表示させているときはその名称)

TTC 標準用語

表示されている本文中にある TTC 標準用語の存在を () で表示します。

Hyper + Mouse-L クリックでその用語が文書中の他のどこに存在するかを表示するウィンドウが開きます。

ウィンドウ中の section、subsection などの名称を Mouse-L クリックすると指示された section の文書ウィンドウが開きます。

The screenshot shows a software interface with a main document window and a search window. The document window title is "IT-1411: 2. ユーザ・網インタフェースの参照構成". It contains a table of contents with sections like "2.1 機能群" and "2.1.1 網終端(装置)1 (以下NT1と略記)". The search window title is "検索群" and lists sections from "section 2" to "ELSE BOOKS MENU". At the bottom, there is a menu bar with options like "Keyword View SW", "Next Section", "Prev Section", "Refresh Window", "Search Paragraph", "Switch Section", and "Window Full". Below the menu bar, there are commands for "Section Window command: Next Section", "Section Window command:", and "Section Window command: Show Section With This Term (機能群 . \npar 1)".

コマンドメニューエリア

以下のコマンド機能は Mouse-L クリックすることで実行指示します。

Keyword View SW

本文中に設定されている「キーワード」と「視点」を表示します。

ただし、表示状態にある場合は表示を中止します。

実行指示後に Refresh Window を Mouse-L クリックします。

Next Section

現在表示している段落の1つ次の段落を表示します。

ただし、現在表示している段落が section 中の最後の subsection であるような場合は section を表示します。

Prev Section

現在表示している段落の1つ前の段落を表示します。

ただし、現在表示している段落が section 中の最初の subsection であるような場合は section を表示します。

4. 6 キーワードウィンドウ (Keyword Case Window)

カードシステムマネージャで Keyword Case を Mouse-L クリックすると開きます。設定したキーワードが視点によって分類されて表示されます。また、「キーワードウィンドウ」では「キーワード」の体系化ができます。体系化の詳しい手順は「4. 5 文書ウィンドウ」に記述されています。以下でウィンドウに表示されているものについて解説します。

ディスプレイエリア

視点名

キーワード設定時に指定した視点が表示されます。コマンド "View-Point Hierarchy" を実行後に「視点」を Mouse-L クリックするとその視点の中で構成されている知識構造をツリー階層図で表示します。

キーワード

文書ウィンドウで設定された「キーワード」を「視点」ごとに表示します。キーワードリンクによって構造化されている「キーワード」はキーワードの横に (*) を表示します。

段落名時表示

「キーワード」をマウスポインタでつかみ、Super + Mouse-L クリックします。ウィンドウが開いてその「キーワード」の所属する段落名を表示します。マウスポインタをウィンドウから外すことによってウィンドウを閉じます。

Show Paragraph

「キーワード」を選び出したパラグラフを表示します。
実行指示後、「パラグラフ」を参照したい「キーワード」を Mouse-L クリックします。
続けて「キーワード」を Mouse-L クリックすることで、同時に複数の「パラグラフ」を参照することができます。
Return キーを押し、パラグラフを表示します。

選ばれたキーワードは次のパラグラフに付けられています。

Dynamic Window Pane 894

[Jt-(411)]--\section 4 (4.チャンネル個別)\per 1

構成 チャンネル;構成

定義 チャンネル;定義

チャンネルとはインタフェースの情報伝送容量の指定された部分を表し、以下に示すチャンネル種別がある。

[Jt-(411)]--\subsection 4-1-1 (4.1.1)\per 1

定義 Bチャンネル;定義

Bチャンネルはタイミングをもつ64kbit/sのチャンネルであり、種々のユーザ情報番号を送送するために使用される。ただし、BチャンネルによりISDNの回線交換用シグナリング情報を伝送することはない。このシグナリング情報は、4.2節に示すDチャンネルによって伝送される。

[Jt-(411)]--\subsection 4-2-1 (4.2.1)\per 1

定義 Dチャンネル;定義

Dチャンネルは5章に規定されているように、16kbit/s又は64kbit/sのビットレートととも、第一義的にはISDNによる回線交換用のシグナリング情報を運ぶために用いられる。
Dチャンネルは、TTC標準JT-0920、JT-0921、JT-0930及びJT-0931に従う階層化されたプロトコルに従うものとする。

[Jt-(411)]--\subsection 4-3-1 (4.3.1)\per 1

定義 Hチャンネル;定義

Hチャンネルはタイミングを持ち、次に示す速度のビットレートと有するチャンネルである。
(1)H0チャンネル:384kbit/s
(2)H1チャンネル:1536kbit/s

Refresh Window Show Section

Parag Window command: Refresh
 Parag Window command: Show Hierarchy With This Keyword 参照点
 Parag Window command: Refresh
 Parag Window command:

Mouse-B: Menu.
To see other commands, press Shift, Control, Meta-Shift, or Super.

[Wed 22 Mar 5:10:04] hananaka CL SUCCESS: User Input

View-Point Hierarchy

キーワードの階層構造を表示します。
キーワードリンク構造を確認したい「視点」を Hyper + Mouse-L クリックします。
コマンドメニューの「Refresh Window」を Mouse-L クリックします。
指定された「視点」による「キーワード」の持つ「キーワードリンク」構造を表示します。

コマンド入力エリア

キーボードやマウスから実行されたコマンドの履歴を表示します。

マウスドキュメンテーションライン

マウスの操作によって何が実行できるかを表示します。

4. 7 要約検索ウィンドウ

体系化した知識構造から要約を検索するウィンドウです。

どのウィンドウからでも Select + B キーを押すことで開きます。

体系化された知識構造を様々な場面で生じる疑問に答えるために「要約」として検索します。ここで得られる「要約」は利用者自身の理解に基づくものです。

検索を柔軟にできるように検索用言語として Prolog を採用しました。ルール式を書くことで、プロログの強力で柔軟な検索能力を利用して、利用者は適切な検索結果を得ることができます。

検索の方法と検索ルールの記述方法は「5 検索と要約」でくわしく説明します。

The screenshot shows a terminal window titled "Retrieve Window" with two main panes. The top pane, "Case Window Pane With Timeout 131", displays a hierarchical tree structure of search results for the keyword "チャンネル". The bottom pane, "Case Window Pane With Timeout 132", shows a table of related terms and a list of Prolog commands.

Case Window Pane With Timeout 131

```

<<---- 部分 ---->>
チャンネル;定義
├── Bチャンネル;定義
├── Dチャンネル;定義
└── Hチャンネル;定義

[ No Key-word-link object ]
Hチャンネル;利用方法
Dチャンネル;利用方法
Bチャンネル;利用方法
チャンネル;構成
    
```

Case Window Pane With Timeout 132

異常状態	基礎説明	構成	図示	能力	利用場面
利率	経緯	実現条件	説明	物理的構成	利用方法
経費	規定	実現方法	対象	方法	例
活用制限	規定範囲	実行条件	注意事項	目的	
活用目的	形態	実行方法	定義	用語	

Prolog Show Keyword Link Show Paragraph Show View Point

```

Retrieve Window command: Show View Point
Retrieve Window command: Retrieve Window Show Paragraph
key-word [default: 空欄点]: 空欄点
Retrieve Window command: Show Section Name 経緯等
Retrieve Window command: Retrieve Window Show Paragraph
key-word [default: 空欄点]: 空欄点
[17:06:46 Unable to hardcopy window: Output packet about to be lost. Continue to return it and proceed.]
Retrieve Window command: Prolog

: (summary "チャンネル" *x)
Retrieve Window command:
    
```

Mouse-L, -R: Retrieve Window Show Paragraph.
 To see other commands, press Shift, Control, Meta-Shift, or Super.
 [Wed 22 Mar 5:09:09] Keyboard CL SUCCESS: User Input

結果表示エリア

検索の結果を表示します。

コマンド実行結果表示エリア

メニューにある Show Keyword Link と Show View Point の実行によって得られた結果を表示します。

第 4 章

COSMOS 初期操作環境の作成

4.1 本の作成 ～ COSMOS 原始文書の作成 ～

TEX 文書を COSMOS ドキュメント用の原始文書に変換するステップとして『本の作成』があります。関数 `text-to-sentence` の実行によって文書交換されますが、交換が失敗した場合には、エラーメッセージとエラーファイル名が表示されます。

=== やってみよう ===

『WELCOME COSMOS』ケース内の『本の作成』ケースを OPEN し、その実行説明文に従ってください。

4.2 重要語の切り出し

COSMOS 環境を整えるステップとして『重要語の切り出し』があります。任意の文字列が使われている section および keyword の検索を可能にするために、その文字列による重要語の切り出しを自動的に行なわせます。

=== やってみよう ===

『WELCOME COSMOS』ケース内の『重要語の切り出し』ケースを OPEN し、その実行説明文に従ってください。

4.3 文間リンク (参照) の獲得

COSMOS 環境を整えるステップとして『文間リンク (参照) の獲得』があります。図表などの参照箇所の指示や、説明等を他の section で行なうなどの指示があるパラグラフと、その関連のあるパラグラフとの間に、自動的にリンクを設定します。

=== やってみよう ===

『WELCOME COSMOS』ケース内の『文間リンク (参照) の獲得』ケースを OPEN し、その実行説明文に従ってください。

4.4 COSMOS 参照機能実現オブジェクトの作成

COSMOS 環境を整えるステップとして『COSMOS 参照機能実現オブジェクトの作成』があります。これらのオブジェクトを通して、検索・参照・変更等の機能が提供されます。

=== やってみよう ===

『WELCOME COSMOS』ケース内の『COSMOS 関連機能オブジェクト作成』ケースを OPEN し、その実行説明文に従ってください。

第 5 章

COSMOS 基本オブジェクトの獲得

5.1 「キーワード」と「視点」を設定する

5.1.1 「キーワード」と「視点」とは

「キーワード」(keyword)には各々の「パラグラフ」(paragraph)に関係を持つ重要な短い語句を設定します。その「パラグラフ」を代表するような語句や、様々な立場からそのパラグラフを検索し呼び出す時の索引になる語句です。また、どの様な語句から疑問が生じたときにその「パラグラフ」を読み直す必要があるかを考えて「キーワード」にしても良いでしょう。

「視点」(view-point)は「キーワード」と対になるもので、どの様な立場からそのキーワードを選び出したかを表します。これは後で「要約」を取り出すときの「視点」になります。「キーワード」と同じように、どの様な立場から疑問が生じたときその「パラグラフ」を読み直す必要があるかを考えて決めても良いでしょう。つまり、疑問が生じたときには「キーワード + 視点」で必要な「要約」を検索します。

ここで「キーワード」と「視点」との関係を説明します。作業を進めるうちにある語句を「キーワード」に含めるべきか、「視点」とするべきかを迷うときがあります。例えば、「ループバック機構」をキーワードにするとき、

視点	キーワード
ケース1 機構	ループバック
ケース2 保守	ループバック機構

と2つ以上のケースが考えられます。その選択の基準は1つの「視点」につく「キーワード」の数です。ある1つの立場からの「キーワード」の数が多ければ、それを「視点」として登録します(ケース1)。しかし、数が少なければ、それは「視点」とするのではなくキーワードの付属語と考えられます(ケース2)。

5.1.2 「キーワード」と「視点」を作成する方法

ここでは前節で説明された「キーワード」と「視点」を実際に設定してみましょう。
たとえばこんなパラグラフがあるとします。

『チャンネルにはBチャンネルやCチャンネルがある。』

このパラグラフに”種類”という「視点」で”チャンネル”という「キーワード」を設定してみましょう。

実行手順は次の通りです。

マウス・キーボード等に関する共通操作についての詳細は、「4章 ウィンドウ操作」を参照して下さい。

1. キーワードの選択

文書中にキーワードとして適切な語句があった場合

文書を読みながらキーワードになる語句をリージョン指定します。

=== やってみよう ===

この場合は"チャンネル"をリージョン指定します。

注：リージョン指定

文書の語句に下線を引くこと

語句の先頭の文字をマウスポインタで指示します。

Control + Mouse-L ドラッグしながら (マウスボタンを押したままの状態で)

マウスを右に移動します。

語句の末尾まで下線を引いて、マウスボタンを離します。

2. パラグラフの選択とキーワード及び視点の決定

マウスポインタでパラグラフオブジェクトをつかみます。

=== やってみよう ===

この場合は par2 をつかみます。

Hyper + Mouse-L クリックします。

"Attributes window" が開きます。

1で文書中の語句をリージョン指定してある場合はその語句が title 表示されます。

語句が適切であることを確認します。

語句が適切でない場合、ここで書き換えることができます。語句を Mouse-M クリックした後にキーボードから語句を記入します。

リージョン指定しなかった場合は空欄の title が表示されますので、空覧を Mouse-L クリックした後にキーボードから語句を記入します。

次に、一覧の中から適切な「視点」を Mouse-L クリックすることで選択します。

適切な「視点」がない場合は、"create new" をクリックして下さい。

=== やってみよう ===

この場合は"の種類"をクリックして下さい。

"done" を Mouse-L クリックするか End キーを押すことで「キーワード」及び「視点」を決定します。

視点の選択で "create new" を選択した場合には「視点」設定ウィンドウが開きます。

title に "視点を記入して下さい" と表示されますので、表示を Mouse-L クリックした後にキーボードから語句を記入します。

"done" を Mouse-L クリックするか End キーを押すことで「視点」を決定します。

取り消したい場合は、"Abort" を Mouse-L クリックします。

このようにして選択された「視点」「キーワード」に対して、システムがチェックを行います。

そして同一視点での類似語があった場合は、それらのキーワードをそのパラグラフと共に表示します。

ここでそれらのキーワードから選ぶ場合は、そのキーワードを Mouse-L クリックし、今選択した「視点」「キーワード」で新しくキーワードを作る場合は、「新しくキーワードを作成する」を Mouse-L クリックします。

”done” を Mouse-L クリックするか End キーを押すことで決定します。

取り消したいときは ”abort” を Mouse-L クリックして下さい。

また、表示された類似語を見て、もう一度「キーワード」のタイトルを付け直したいときには、”Collect title” を Mouse-L クリックして下さい。

最初の ”Attributes Window” に戻ります。

もう一度今までの作業をくり返して下さい。

さて上記の操作で「キーワード」と「視点」がパラグラフにつきました。

その確認表示をさせてみましょう。

=== やってみよう ===

このウィンドウ (文書ウィンドウ) のコマンドメニューの ”Keyword View SW” と ”Refresh” を順に Mouse-L クリックします。

5.1.3 「キーワード」と「視点」を作成する方法 ～ 自動作成 ～

そのパラグラフにある文章の構文パターンから、「キーワード」と「視点」を決定し自動獲得する方法があります。

ここでは、その操作方法を記述します。

★「辞書ファイルカード」の設定

自動獲得を実現するには、文法カード、辞書カード、文章パターンカードの3つの辞書ファイルカードが必要です。次にこれらのカードの作成方法を示します。

a. 「文法カード」

card-system::basic-card で作成します。

そのインスタンス変数 keyword-string には ”grammar” をセットします。

カードオブジェクトをオープンし、そのウィンドウのなかで構文解析のための文法を定義します。

ただし、パッケージの変更を念頭において文法定義のマクロ defgram は、prs::defgram のかたちで使用して下さい。

b. 「辞書カード」

card-system::basic-card で作成します。

そのインスタンス変数 keyword-string には ”dictionary” をセットします。

カードオブジェクトをオープンし、そのウィンドウのなかで構文解析のための単語辞書を定義します。

ただし、パッケージの変更を念頭において、辞書定義のマクロ defdict は prs::defdict のかたちで使用して下さい。

c. 「文章パターンカード」

card-system::basic-card で作成します。

定義する視点別に分けて、各カードを作成して下さい。

そのインスタンス変数 keyword-string には ”view-point” をセットします。

カードオブジェクトをオープンし、そのウィンドウのなかでその視点に対応する文章パターン解析式を Lisp 式で記

述します。

Lisp 式は構文解析結果の構造リストを引数とし、戻り値が ("キーワードタイトル 視点タイトル" ("キーワードタイトル" "視点タイトル")) のかたちでキーワード / 視点の情報リストを集めたリストとなるような Lambda 式で記述します。

「視点 EXPERT-CASE (card-system::basic-case で1つだけ作成、その keyword-string には "expert-case" をセット)」に格納します。

★構文解析環境の設定

文章パターン解析式に引き渡す文章の構造リストを生成するために、parser による構文解析を用います。その環境設定の方法を次に示します。

a) 「文法カード」のロード

「文法カード」オブジェクトをオープンさせたウィンドウのなかで、Hyper + R クリックしてメニューを開きます。

そのなかの "EntryGrammar" をクリックします。

この実行によって、「文法カード」の内容と共に「辞書カード」の内容もロードします。

b) 「辞書カード」のロード

「辞書カード」オブジェクトをオープンさせたウィンドウのなかで、Hyper+R クリックしてメニューを開きます。

そのなかの "EntryDictionary" をクリックします。

この実行によって「辞書カード」の内容と共に「文法カード」の内容もロードします。

★「パラグラフ」の選択と「キーワード」及び「視点」の決定

マウスポインタで par1,par2 などの表示のあるパラグラフ名をつかみます。

Mouse-R クリックします。

パラグラフメニューが開きます。

"Get viewpointfrom sentence pattern" を Mouse-L クリックします。

そのパラグラフのなかで、構文解析と文章パターン解析の結果、抽出されたキーワード / 視点がメニューとなって表示されます。

そのメニューのなかで掲げられているキーワード / 視点のうち、実際に作成するものは "yes" の箇所を Mouse-L クリック、作成しないものは "no" の箇所を Mouse-L クリックします。そして最後に "DONE" を Mouse-L クリックすれば、そのパラグラフに選択したキーワード / 視点が付けられ、"Abort" を Mouse-L クリックすれば取り消されます。

5.1.4 「キーワード」と「視点」の変更と取消方法

「キーワード」「視点」の変更・取り消しは、このウィンドウ (文書ウィンドウ) で行う場合と、キーワードエディタで行う場合の2つが考えられます。

また、その「キーワード」に「キーワードリンク」が付けられていないときには、文書ウィンドウで変更・取り消しができますが、既に「キーワード」が付いている場合には、キーワードエディタでしか変更・取り消しすることができません。

もしも「キーワードリンク」が付いている状態で文書ウィンドウでの変更を行った場合には、その「キーワードリンク」が無条件に取り消されてしまいますので注意して下さい。

なおキーワードエディタ上での操作方法は4章で述べられます。

a. 文書ウィンドウでの変更

変更したい「キーワード」が付けられている「パラグラフ」を Mouse-R クリックして Menu を表示し、”Change attribute keyword” を Mouse-L クリックします。その「パラグラフ」に付けられている「キーワード」がポップアップウィンドウに表示されますので、変更したい「キーワード」を Mouse-L クリックして下さい。また「キーワード」をクリックせずにこのポップアップウィンドウからマウスを外すと変更処理に入らずに処理を終了します。これ以降の手順は「キーワード」の設定と同じですので、そちらを参照して下さい。

=== やってみよう ===

このパラグラフについている”変更・取消キーワード”という「キーワード」のタイトルを変更してみましょう。

b. 文書ウィンドウでの取り消し

取り消したい「キーワード」が付けられている「パラグラフ」を Mouse-R クリックして Menu を表示し、”Remove Keyword” を Mouse-L クリックします。

その「パラグラフ」に付けられている「キーワード」がポップアップウィンドウに表示されますので、取り消したい「キーワード」を Mouse-L クリックして下さい。

「キーワード」をクリックせずにこのポップアップウィンドウからマウスを外すと、取り消し処理に入らずに処理を終了します。

=== やってみよう ===

さきほど変更した「キーワード」を取り消してみましょう。

5.2 「キーワード」を体系化する ～ キーワードリンク ～

5.2.1 「キーワードリンク」とは

「キーワードリンク」(keyword link)は文書から選び出した「キーワード」をお互いに関係付けることで、知識を体系化するための機能です。

同じ視点で選び出されたキーワードは互いに何らかの関係を持つと考えられます。それらのキーワードを選び出してきたパラグラフとは関係なく、独立して自分の理解に基づいて「キーワード」を用いて構成します。

同じ「視点」の「キーワード」で階層化された1つの知識体系を作ることが基本です。

作業を進める中で、異なった「視点」を付けていても同じ知識体系の中に入れたいことがあります。システムの機能としては制限を設けてありませんので構造化はできます。しかし、一般にその場合はキーワードの「視点」が変化しています。「視点」を変えることが望ましいでしょう。

5.2.2 「キーワードリンク」を作成する方法

ここでは前節で説明された「キーワードリンク」を実際に設定してみましょう。
つぎのパラグラフに以下のようなキーワードが設定されています。

『ヒトの顔は目や鼻や口からなるが、目は水晶体や網膜からなり、鼻は鼻腔などからなり、口は歯や舌や顎からなる。』

上位概念になる「キーワード」をマウスポインタでつかみます。

=== やってみよう ===

この場合”ヒトの顔”という「キーワード」をつかみます。

Hyper + Mouse-L クリックします。

下位概念になる「キーワード」を Mouse-L クリックします。

=== やってみよう ===

この場合”目”という「キーワード」を Mouse-L クリックします。

ウィンドウが開いて「キーワードリンク」の一覧が表示されます。

一覧の中から適切な「キーワードリンク」を Mouse-L クリックすることで選択します。

適切な「キーワードリンク」がない場合、マウスポインタをウィンドウから外します。

「キーワードリンク」設定ウィンドウが開きます。

”リンク名を記入して下さい”と title 表示されますので、title を Mouse-M クリックして、キーボードから記入します。

=== やってみよう ===

この場合”構成要素”と記入してみましょう。

”done”を Mouse-L クリックするか End を押すことで「キーワードリンク」を決定します。

取り消したい場合は”Abort”を Mouse-L クリックします。

間違ってもマウスポインタがウィンドウから外れてしまったためにこのウィンドウが表示された場合は、何も記入せずに”Abort”を Mouse-L クリックします。

さて上記の操作で「キーワードリンク」ができました。

その確認表示をさせてみましょう。

==== やってみよう ====

「キーワード」をマウスポインタでつかみ、Hyper + Mouse-M をクリックします。

ウィンドウが開いてその「キーワード」に関係するすべての「キーワードリンク」階層構造が「キーワードリンク」名ごとに表示されます。

ウィンドウの外を Mouse-L クリックすることで元のウィンドウにもどります。

なおこの確認表示は「キーワード」が表示されているすべてのウィンドウで行なえます。

5.2.3 「キーワードリンク」の取消方法

「キーワード」階層構造が表示されている全てのウィンドウで変更できます。

階層構造上の「子キーワード」をマウスポインタでつかんで Mouse-R クリックします。

Option window で "Remove higher keyword-link from this object" を選択します。

ウィンドウが開いて Remove することを確認しますので、Mouse-L クリックします。

Remove 操作を中断する場合はマウスポインタをウィンドウから外します。

5.3 文書に関係付ける ～ 文間リンク ～

5.3.1 文間リンクとは

「文間リンク」は「パラグラフ」間の関係付けを、文書の構成に基づいて客観的に行うための機能です。パラグラフ同士が密接に関連を持っている場合があります。例えば「つまり」などの接続詞でつながれて、前のパラグラフの詳細が次のパラグラフに記載されていることがあります。また、図や表を参照することが指示されていることもあります。検索によって「要約」を表示するときこれらの関係が大切な働きをします。実際にどんなものなのでしょうか？ ちょっと見てみましょう。

==== やってみよう ====

このパラグラフについている "→" を Mouse-R クリックして表示されるメニュー中の、"Show between paragraph link" を選んで下さい。

するとこのパラグラフが関連づけられているパラグラフの一覧表がポップアップされるので、それを Mouse-L クリックして下さい。

さあ、いかがですか？

文間リンクの機能の一つとしては、このように前のパラグラフの説明等が次のパラグラフでも続けて行なわれている場合に、それらのパラグラフを結び付けることがあげられます。

5.3.2 文間リンクを作成する方法

では実際に文間リンクを作成してみましょう。手順は次の通りです。

1、対象となるパラグラフを選択します。

マウスポインタで par1、par2 などと表示のあるパラグラフ名をつかみ、Hyper + Mouse-M でクリックします。

==== やってみよう ====

ここでは、このパラグラフを Hyper + Mouse-M クリックしてください。

2、関係付けたいパラグラフを選択します。

マウスポインタで par1、par2 などと表示のあるパラグラフ名をつかみ Hyper + Mouse-M でクリックします。

=== やってみよう ===

ここでは、このパラグラフを Mouse-L クリックしてください。

3、ウィンドウが開いて文間リンクの一覧が表示されます。

一覧の中から適切な「文間リンク」を Mouse-L クリックして選択します。

適切な「文間リンク」がない場合、マウスポインタをウィンドウから外すと「文間リンク」設定ウィンドウが開きません。

”リンク名を記入して下さい”と title 表示されますので、Mouse-L クリックした後、キーボードから記入します。

”done” を Mouse-L クリックするか End キーを押すことで「文間リンク」を決定します。

取り消したい場合は ”Abort” を Mouse-L クリックします。

間違っマウスポインタが文間リンク一覧のウィンドウから外れてしまったためにこのウィンドウが表示された場合は、何も記入せずに ”Abort” を Mouse-L クリックします。

=== やってみよう ===

ここでは、「順序関係」というリンク名をセットしてください。

さあこれで文間リンクができました。確認表示してみましょう。

=== やってみよう ===

ウィンドウを Refresh してみてください。

文書ウィンドウのパラグラフ名の横に、文間リンクが付いていることが矢印で表示されます。上下関係がある場合には親パラグラフには ”→”、子パラグラフには ”←” がそれぞれつきます。

「パラグラフ」を Mouse-R クリックして Menu を表示し、”show between paragraph link” を Mouse-L クリックします。その「パラグラフ」に付けられている「文間リンク」とそのリンクでつながっている「パラグラフ」をポップアップウィンドウに表示します。

もしもそれらの「パラグラフ」を見たい場合は、その「パラグラフ」を Mouse-L クリックして下さい。

また、クリックせずにマウスをポップアップウィンドウから外すともとのウィンドウに戻ります。

5.3.3 「文間リンク」の取消方法

「パラグラフ」が表示されている全てのウィンドウで解消できます。

「パラグラフ」をマウスポインタでつかんで Mouse-R クリックします。

Option window で ”Remove between-paragraph-link from this object” を選択します。

ウィンドウが開いて Remove することを確認しますので、Mouse-L クリックします。

Remove 操作を中断する場合はマウスポインタをウィンドウから外します。

第 6 章

ウィンドウ操作

6.1 共通操作

6.1.1 マウス操作

本支援プログラムの基本的な操作はマウスでできます。本説明書では次の用語と表記法を用います。不明な点はシンボリクスのマニュアルなどを参照して下さい。

- マウスポインタ

マウスの指示位置を表すために画面上に表示される矢印

- マウスポインタでつかむ

マウスポインタがオブジェクトを指し示すと四角い枠が現れたり、枠が強調されたりする変化が起こります。この状態をマウスポインタがオブジェクトを「つかむ」といいます。オブジェクトは単なる文字列ではなく、意味を持ったデータの集合体です。

- クリックする

マウスポインタがオブジェクトをつかんだ状態でマウスの左ボタンを「クリック」(押してすぐ放すこと)します。反応はつかんだオブジェクトによって異なります。シンボリクスマウスの3つのボタンのクリックをそれぞれ

左ボタン	Mouse-L クリック
中ボタン	Mouse-M クリック
右ボタン	Mouse-R クリック

と表記します。

キーボード操作と組み合わせて

マウス操作はキーボードのいずれかのキーを押しながら行う場合があります。それらを

Hyper + Mouse-L
Super + Mouse-M
Control + Mouse-L
Super + Meta + Mouse-L

などと表現します。

- リージョン指定

文書の語句に下線を引くことです。

語句の先頭の文字をマウスポインタで指示します。

Control + Mouse-L ドラッグ(マウスボタンを押したまま)しながらマウスを右に移動し、語句の末尾まで下線を引いて、マウスボタンを離します。

- 操作メニューの表示

Mouse-R クリックするとその場で実行可能な操作のメニューを表示します。そのメニューから操作を選んでマウスポインタでつかみ Mouse-L クリックすると実行します。

マウスポインタがオブジェクトをつかんでいる場合はそのオブジェクトに対して実行可能な操作メニューを表示します。同じ方法で実行します。

Mouse-R をダブルクリック(2回続けて素早く押しすぐ放すこと)をするとシステムメニューをいつでも表示します。同じ方法で実行します。

6.1.2 キーボード操作

キーボードの基本操作方法はシンボリクスの基本操作にしたがいます。

★操作の中断

本支援プログラムでの操作の中断は、システムとして特に定義されていない場合は Abort、Sift + Abort、Control + Abort のいずれかで可能です。

本支援プログラムでは特別な変更や定義はしていませんので、操作はシンボリクスのマニュアルをご覧ください。

6.2 文書ウィンドウ (Section Window)

このウィンドウが文書ウィンドウです。

作業をするための文書を表示します。

また、「文書ウィンドウ」ではつぎのようなことが可能ですが、これらの操作方法は3章を参照してください。

- 1、キーワードと視点の設定
- 2、キーワードの体系化
- 3、パラグラフの関係付け

以下でウィンドウに表示されているものを解説します。

6.2.1 タイトルエリア

● 文書名

作業している文書名を表示します。"ウィンドウ操作"と表示されているところです。

● section

表示している文書の section 名を表示します。

(作業過程で subsection や subsection などを開いて表示させているときはその名称)

6.2.2 ディスプレイエリア

★ subsection、subsection

Mouse-L クリックすることでその subsection、subsection だけを表示させることができます。

つまり、文書量が大きい場合(1画面以上)は作業効率を上げるため、subsection、subsection 名を Mouse-L クリックして小さな単位に分割して表示させます。

=== やってみよう ===

subsection"文書ウィンドウ"を Mouse-L クリックしてみてください。

タイトルエリアも"ウィンドウ操作"から"文書ウィンドウ"に書きかわります。

★パラグラフ (pra 1、par 2 など)

=== やってみよう ===

Mouse-L クリックすることでそのパラグラフだけを表示させることができます。

文間リンクが設定されている場合には矢印(←、→)で存在が表示されます。

★本文

これを読み進めながらキーワード付けなどの作業をします。

★視点とキーワード

=== やってみよう ===

既に設定されている「視点」と「キーワード」を表示します。

表示の有無は Keyword View SW を Mouse-L クリックすることで切り替わります。

キーワードリンクで構造化されている「キーワード」はキーワードの横に (*) を表示します。

★階層構造一時表示

「キーワード」をマウスポインタでつかみ、Hyper + Mouse-M クリックします。

ウィンドウが開いてその「キーワード」に関係するすべての「キーワードリンク」階層構造を「キーワードリンク」ごとに表示します。

ウィンドウの外を Mouse-L でクリックすることで元のウィンドウにもどります。

=== やってみよう ===

「キーワード」はキーワードの横に (*) が表示されているものを選び、Hyper + Mouse-M クリックしてください。

★図、表 (fig 1-1、tab 2-4 など)

図表の存在を表します。

★TTC 標準用語

表示されている本文中にある TTC 標準用語の存在を () で表示します。

Hyper + Mouse-L クリックでその用語が文書中の他のどこに存在するかを表示するウィンドウが開きます。

ウィンドウ中の section、subsection などの名称を Mouse-L クリックすると指示された section の文書ウィンドウが開きます。

6.2.3 コマンドメニューエリア

以下のコマンド機能は Mouse-L クリックすることで実行指示します。

★ Keyword View SW

=== やってみよう ===

本文中に設定されている「キーワード」と「視点」を表示します。

ただし、表示状態にある場合は表示を中止します。

実行指示後に Refresh Window を Mouse-L クリックします。

★ Next

=== やってみよう ===

現在表示している段落の1つ次の段落を表示します。

ただし、現在表示している段落が section 中の最後の subsection であるような場合は section を表示します。

★ Prev

=== やってみよう ===

現在表示している段落の1つ前の段落を表示します。

ただし、現在表示している段落が section 中の最初の subsection であるような場合は section を表示します。

★ Refresh

==== やってみよう ====

ウィンドウを書き換えます。ウィンドウ内の表示が乱れたときに Mouse-L クリックして実行します。

また、Keyword View SW など画面の表示を変更するコマンドを実行指示した後に Mouse-L クリックして表示変更を実行します。

★ Search Par.

==== やってみよう ====

指定した文字列を含む「パラグラフ」をすべて表示します。

実行指示後に検索文字列を記入し、Return キーを押します。

★ Swich

==== やってみよう ====

表示させたい段落を section、subsection などの番号を指示することで表示させます。実行指示後に "2-3-2" + Return のようにキーボードから入力して実行します。

★ Full、Half

==== やってみよう ====

Half で文書ウィンドウの表示幅を縦半分にします。

複数の段落で同時に作業を進めたいときに利用します。

Full で元へ戻します。

実行後に Function + Refresh を実行することをお勧めします。

★ Index View SW

==== やってみよう ====

本文中にある TTC 標準用語に () をつけて表示します。

ただし、表示状態にある場合は表示を中止します。

実行指示後に Refresh Window を Mouse-L クリックします。

★ Show SDL

現在参照中の本に含まれるすべての SDL 図をポップアップウィンドウに表示します。

表示させたい SDL 図を Mouse-L クリックで選択します。

選択せずにマウスをこのポップアップウィンドウから外すと、もとのウィンドウに戻ります。

6.2.4 コマンド入力エリア

キーボードやマウスから実行されたコマンドの履歴を表示します。

★マウスドキュメンテーションライン

マウスの操作によって何が実行できるかを表示します。

6.3 キーワードウィンドウ

カードシステムマネージャで Keyword Case を Mouse-L クリックすると開きます。

設定したキーワードが視点によって分類されて表示されます。

また、「キーワードウィンドウ」では「キーワード」の体系化ができます。

体系化の詳しい手順は3章に記述されています。

以下でウィンドウに表示されているものについて解説します。

6.3.1 ディスプレイエリア

★視点名

キーワード設定時に指定した視点が表示されます。

=== やってみよう ===

「視点」上での Mouse-R クリックメニューの "Show hierarchy with this viewpoint" によってその視点の中で構成されている知識構造をツリー階層図で表示します。

★キーワード

文書ウィンドウで設定された「キーワード」を「視点」ごとに表示します。

キーワードリンクによって構造化されている「キーワード」はキーワードの横に (*) を表示します。

★階層構造一時表示

=== やってみよう ===

「キーワード」をマウスポインタでつかみ、Hyper + Mouse-M クリックします。

ウィンドウが開いてその「キーワード」に関するすべての「キーワードリンク」階層構造を「キーワードリンク」ごとに表示します。

ウィンドウの外を Mouse-L でクリックすることによって元のウィンドウにもどります。

6.3.2 コマンドメニュー

以下のコマンド機能は Mouse-L クリックすることで実行します。

★Serch Key Word

指定した文字列を含むキーワードだけを表示します。

実行指示後、検索する文字列を入力すると、指定した文字列を含むキーワードだけを表示します。

=== やってみよう ===

上記の説明に従って、指定する文字列だけを含むキーワードを表示してみましょう。

その後、Refresh Window の説明を読んでから、それを実行してください。

★ Refresh Window

ウィンドウを書き換えます。ウィンドウ内の表示が乱れたときに実行します。

また、Search Key Word など画面の表示を変更するコマンドを実行指示した後に表示変更します。

★ Reset Serch Mode

=== やってみよう ===

Serch Key Word による表示を標準の一覧表示に戻します。

実行後、Refresh Window を行なって下さい。

★ Show Paragraph

=== やってみよう ===

「キーワード」を選び出したパラグラフを表示します。

実行指示後、「パラグラフ」を参照したい「キーワード」を Mouse-L クリックします。

続けて「キーワード」を Mouse-L クリックすることで、同時に複数の「パラグラフ」を参照することができます。

Return キーを押し、パラグラフを表示します。

6.3.3 コマンド入力エリア

キーボードやマウスから実行されたコマンドの履歴を表示します。

6.3.4 マウสดキュメンテーションライン

マウスの操作によって何が実行できるかを表示します。

6.4 要約検索ウィンドウ

カードシステムマネージャで 検索 Case を Mouse-L クリックして開きます。

そのウィンドウ内で Hyper + Mouse-R クリックによってポップアップするメニューのなかの "Create Retrieve Case" を実行します。

この実行によって作成された Case (汎用検索ケース) を開くと要約検索ウィンドウがオープンします。

このウィンドウは体系化した知識構造から要約を検索するウィンドウです。

体系化された知識構造を様々な場面で生じる疑問に答えるために「要約」として検索します。ここで得られる「要約」は利用者自身の理解に基づくものです。

検索を柔軟にできるように検索用言語として Prolog を採用しました。ルール式を書くことで、プロログの強力で柔軟な検索能力を利用して、利用者は適切な検索結果を得ることができます。

検索の方法と検索ルールの記述方法は「5 検索と要約」でくわしく説明します。

6.4.1 コマンド実行結果表示エリア

コマンドの実行結果や、検索結果の表示などに使います。

検索の実行結果は本表示エリアに「●」ボタンで標示されます。

この「●」ボタンを Mouse-L クリックすることで検索結果としてのキーワード構造が標示されます。

6.4.2 コマンドメニューエリア

以下のコマンドは Mouse-L クリックすることで実行します。

★検索

検索のために Prolog の実行状態に入ります。

=== やってみよう ===

「検索」を Mouse-L クリックしてください。

Prolog の初期化ファイルが load されて、Prolog 述語の記述モードになります。

★ Show Keyword Link

「キーワードリンク」の種類の一覧を表示します。

=== やってみよう ===

「 Show Keyword Link 」を Mouse-L クリックしてください。

★ Show Paragaraph

=== やってみよう ===

「キーワード」の所属する「パラグラフ」を表示します。

実行指示後、「パラグラフ」を参照したい「キーワード」を Mouse-L クリックします。

続けて「キーワード」を Mouse-L クリックすることで、同時に複数の「パラグラフ」を参照することができます。

Return キーを押し、パラグラフを表示します。

★ Show View Point

「視点」の種類の一覧を表示します。

=== やってみよう ===

「 Show View Point 」を Mouse-L クリックしてください。

6.4.3 コマンド入力エリア

キーボードやマウスから実行されたコマンドの履歴を表示します。
キーボードからの検索式の入力、実行指示もこの領域で行います。

★マウスドキュメンテーションライン

マウスの操作によって何が実行できるかを表示します。

6.5 ルールウィンドウ

検索式を記述して蓄積するウィンドウです。

カードシステムマネージャから「ルールカード」を Mouse-L クリックすることで開きます。

検索の方法と検索ルールの記述方法は「5 要約検索」でくわしく説明します。

ルール記述には、現在2つのカードを提供しています。

検索式 (Prolog 式) を記述するカード「retrieve」と類似語を定義するカード「用語整理」の2つです。

「用語整理」は Prolog 処理系内での検索述語に展開されるだけでなく、cosmos 操作時の類似キーワードの検索時にも利用されます。

6.6 疑問点ウィンドウ

「キーワード」の設定作業過程で生じた疑問や意味の解らなかつた語句を記述した Question Card の一覧を表示するウィンドウです。

=== やってみよう ===

まず疑問カードを作りましょう。

パラグラフ上での Mouse-R クリック で表示されるメニュー内の「question card」を実行して下さい。

疑問や意味の解らなかつた語句を記述するための Card を作成します。

まず Card のタイトルを指定します。

そして Card を マウスで開き、そのなかに内容を記述します。

さあ、これで疑問カードができました。

=== やってみよう ===

さあ、作成した「疑問カード」を、「疑問点ウィンドウ」を開いて見てみましょう。

このウィンドウは、カードシステムマネージャから「Question Case」を Mouse-L クリックすることで開きます。

「疑問カード」の一覧が表示されますので、マウスポインタでつかんで Hyper + Mouse-L クリックします。

「疑問カード」に記入された「パラグラフ」や「キーワード」に関係のあるキーワード階層構造を表示します。

6.7 キーワードエディタ

「キーワード」や「視点」或は「キーワードリンク」について編集するウィンドウです。
カードシステムマネージャから「Keyword Editor」を Mouse-L クリックすることで開きます。
また「キーワード」を Mouse-R クリックして Menu から "Edit Keyword" を選択することによっても、このウィンドウに入ることが出来ます。

6.7.1 ウィンドウの各エリアの説明

- ディスプレイエリア
各コマンドの結果が表示されます。
- Current Candidates エリア
現在編集作業の対象となるキーワードを表示します。
「キーワード」の Menu から "Edit Keyword" を選択してキーワードエディタに入った場合は、それらの「キーワード」を表示します。
- Book Marks エリア
実行したコマンドの履歴を表示します。ここに表示されているコマンドを Mouse-L クリックすると、前に表示したそのコマンドの結果を再び見ることが出来ます。

6.7.2 操作方法 ～ マウスによる操作 ～

★ describe

=== やってみよう ===

Current Candidates エリアで「キーワード」を Mouse-L クリックするとその「キーワード」に対する「視点」と「パラグラフ」名を表示します。

ディスプレイエリアで「キーワード」を Mouse-L クリックするとその「キーワード」とその「キーワード」に付けられている「パラグラフ」の内容をすべて表示します。

★ show-links

=== やってみよう ===

Current Candidates エリアで「キーワード」を Mouse-M クリックするとその「キーワード」に付けられている「キーワードリンク」とそのリンクによってつながっている「キーワード」を表示します。

★ show-partial-hierarchy

=== やってみよう ===

ディスプレイエリアで「キーワード」を Mouse-M クリックするとその「キーワード」に付いている「キーワードリンク」の tree 構造の中から、その「キーワード」を中心としてその親と子に当たる部分のみを表示します。

もしその「キーワード」に「キーワードリンク」がセットされていない場合は、「キーワード」と「視点」を表示します。

★ describe with similar keyword

=== やってみよう ===

ディスプレイエリアで「キーワード」を control-shift-Mouse-L クリックすると、その「キーワード」とその類似キーワードについて「視点」と「パラグラフ」名を表示します。

類似キーワードを指定するルールについては、「5 要約検索」で詳しく説明します。

*これらの操作は「キーワード」を Mouse-R クリックして表示される Menu から起動することが出来ます。

6.7.3 操作方法 ～ コマンドメニューからの操作 ～

★ Show Candidates

=== やってみよう ===

キーボードから編集したい「キーワード」の文字列を入力して下さい。

その文字列を含むタイトルを持つ「キーワード」があれば Current Candidates エリアにそれらの「キーワード」を表示します。

★ Create New Keyword

=== やってみよう ===

新しく「キーワード」を作成します。作成の手順は「キーワード」作成の項を参照してください。

ただしここで作成した場合は、対象となる「パラグラフ」を持ちません。「パラグラフ」を付ける場合はディスプレイエリアに表示されている「キーワード」を Mouse-R クリックして、Menu から "Store paragraph to this keyword" を Mouse-L クリックします。

ここで、対象となる「パラグラフ」を Mouse-L クリックして選択します。

選択が終われば「Return」を押すことによって処理が終了します。

★ Synthesize Concept

=== やってみよう ===

いくつかの「キーワード」を統合して一つの「キーワード」にします。

まず、統合したいいくつかの「キーワード」(ディスプレイエリアに表示されているもの)を Mouse-L クリックします。

「キーワード」を選び終わると「Return」を押して下さい。

次に新しく作る「キーワード」の作成に移ります。

この手順は「キーワード」作成の手順と同じですのでそちらを参照してください。

「キーワード」の作成が終了すると統合処理に入りますが、このとき統合する「キーワード」の中に「キーワードリンク」を持つものがなければそのまま処理が進みます。

また、「キーワードリンク」を持つ「キーワード」が1つだけあれば、やはりその「キーワードリンク」を新しい「キーワード」が継承して処理されます。

しかし、2つ以上の「キーワードリンク」に対して「キーワードリンク」が付けられていた場合は、画面にそれらの「キーワード」ト「キーワードリンク」を表示しますので、必要な「キーワードリンク」の矢印を Mouse-L クリックすることによって選択して下さい。選択が終われば「Return」を押して下さい。

★ Devide Concept

一つの「キーワード」を分割します。

まず、分割したい「キーワード」(ディスプレイエリアに表示されているもの)を Mouse-L クリックします。分割して新しく作る「キーワード」の作成の画面になりますが、作成の手順は「キーワード」作成の手順と同じですのでそちらを参照して下さい。

次に新しく作る「キーワード」に前の「キーワード」のどの「パラグラフ」を入れるかを選択するウィンドウが開きます。ここで、前の「キーワード」に入れるものは "move" を、また前の「キーワード」にも残しておいて、新しい「キーワード」に入れるものは "copy" を選択して下さい。

選択が終われば "done" を、また処理を中断したいときは "abort" を Mouse-L クリックして下さい。

=== やってみよう ===

"Synthesize Concept" によって統合した「キーワード」を対象にして実行してみましょう。

★ Refresh Window

各エリアをクリアします。ただし、Current Candidates エリアに表示されている内容は残ります。

★ Clear Window

各エリアをクリアします。Current Cadates エリアの内容もクリアされます。

第 7 章

要約検索

7.1 検索ルールの記述

「要約検索」は「要約検索ウィンドウ」において行います。

要約検索には「prolog 処理系」を採用しています。利用者はこの「Prolog 処理系」において、いかに示す述語を用いて検索式を記述することで「要約」を得ることができます。

7.1.1 システム述語

★ユーザ述語の定義

(assert (ユーザ述語 引き数)
& rest body)

★ユーザ述語の削除

(retract ユーザ述語)

★副作用を期待した lisp 式の評価

(eval (function arg1 arg2 ...) *variable)
本述語の評価値は常に t(真)

★評価値を期待した lisp 式の評価

(call function arg1 arg2 ...)

★評価値, 副作用を期待した lisp 式の評価

(peval (function arg1 arg2 ...) *variable)

★ prolog を指向した lisp 関数の実行

(pcall function arg1 arg2 ...)

△具体例△

(summary *x):-

(キーワード *x が文間リンクとして○○○と●●●
を持っていればそのリンク先の paragraph)

↓

(summary *x):-

```
(and (pcall keyword *x *x1)
      (pcall link *x1 *x2 ○○○)
      (pcall link *x1 *x2 ●●●)
      (print *x2) )
```

このような形式でのルール記述を可能ならしめるためには？

lisp 関数は原則として次の要件を満足するものでなければならない。

引き数が nil で呼ばれたときは、その引き数に当てはまる object の list を return-value とする。

引き数が複数存在するときは該当ペアの object-list の list を return-value とする。

すべての引き数が nil 以外のときは、引き数群の関連の真偽を return-value とする。

★ユーザ定義述語カードの読み込み
(read-user)

★定義述語の一覧表示
(predicate)

★定義されたユーザ述語の内容表示
(listing ユーザ述語)

★*x と *y をコンスしたものが *z である
(cons *x *y *z)

★*x はリスト *y の要素である
(member-of *x *y)

★*x と *y をアペンドしたものが *z である
(append *x *y *z)

★true
(true)

★not
(not *x)

★真の評価が出るまで処理を続ける
(or pred1 pred2 ...)

★偽の評価が出るまで処理を続ける
(and pred1 pred2 ...)

★*x は変数である。(その評価時点では、変数以外と unify されていない。)
(var *x)

7.1.2 ユーザ定義基本述語

★検索範囲の指定

- "jt-i411" の 2 章に検索範囲を限定する。

(parent-list '("jt-i411" "2"))

- 検索範囲の解除

(delete-parent-list)

- 検索範囲の判断

(contain-parent-list *x)

- 検索範囲の指定を有効 / 無効

(set-parent-list-flag)

(reset-parent-list-flag)

★キーワードの類似性を指定する。

- 同等定義

(key-word-eq "REJ" "リジェクト")

- 類似定義

(key-word-nearly-eq "... "???")

- 部分定義

(contents-of "レイヤ" "レイヤ1 ")

(contents-of "レイヤ" "レイヤ2 ")

★キーワードの類似性を判断する

- 同等定義

(key-word-equal *x *y)

- 類似定義

(keyword-nearly-equal *x *y)

★キーワードの類似性検索を有効 / 無効にする

- 同等定義

(set-key-word-eq-flag)

(reset-key-word-eq-flag)

- 類似定義

(set-key-word-nearly-eq-flag)

(reset-key-word-nearly-eq-flag)

- 部分定義

(set-contents-of-flag)

(reset-contents-of-flag)

★視点の類似性を指定する

(group g1 "定義")

(group g1 "用語定義")

★視点の類似性検索を有効 / 無効にする。

(set-same-group-flag)

(reset-same-group-flag)

★*x と *y は類似性のある視点である

(same-group *x *y)

★キーワード *k は *v を視点に持っている。

(has-key-word *v *k)

★キーワード *k はキーワードリンク *l を持っている

(has-key-word-link *k *l)

★キーワード *k のキーワードリンク *l におけるルートは *r である
(is-root *r *k *l)

★キーワードリンク *l におけるルート *r からのツリー構造を表示する
(show-tree *l *r)

★画面をクリアする
(clear-screen)

★文字列 *string を *format で出力する
(format *format *string)

★キーワード *key-word を画面に表示する (present-self)
(display-key-word *key-word)

★ *x はキーワードリンクを持っていない
(alone *x)

★ *y は文字列 *x に部分一致するタイトルを持つキーワードである
(bubun-itti *x *y)

第 8 章

本の読み履歴ケース

8.1 「本の読み履歴ケース」とは

画面の右側に表示されているケースウィンドウです。

本を読む時、『「ここら辺り」と「ここら辺り」を読もう』というような簡単な戦略を立てて読みますが、この読み方戦略を簡単に表現できるようにしたのが、この「本の読み履歴ケース」です。

ここでは本のセクション・サブセクション・サブサブセクション・キーワード・パラグラフ等の項目を「見出し」として表示することができ、それらの「見出し」を通じてユーザは「読みたい」箇所をダイレクトに選び、読むことができます。

この時、「目次ウィンドウ」と異なるのは次のような点です。

- 常に画面上に表示されている
- 一度見た箇所(一度オープンした「見出し」ケース)をもう一度見ることが容易である
- 次にどこを読めばいいのかを指示してくれる
- 「見出し」ケースを列挙して表示するだけでなく、図式表示も可能である
- 一冊の本に対して複数の「本の読み履歴ケース」を作成することができ、それらの間に順序を設けることができる
- 順序関係上の前後関係の有無がすぐにわかるように表示する

8.2 操作方法

8.2.1 「見だし」ケースをオープンする

基本的には表示されている「見だし」ケースを Mouse-L クリックします。

その「見だし」ケースの内容としてオブジェクトが1個存在するだけならば、そのオブジェクトに対応するウィンドウが開きます。

内容オブジェクトが2個存在するならば、画面を分割して各オブジェクトの持つウィンドウが開きます。

3個以上存在する場合、それらのオブジェクトを「見だし」ケースとするような次世代の「本の読み履歴ケース」を作成し、表示します。

8.2.2 複数の「本の読み履歴ケース」を切替える

「見だし」ケースのオープン操作によって、その次世代の「本の読み履歴ケース」が表示されるようなことが起こります。こうした時、一つ前に表示されていた「読み履歴ケース」に戻って表示させたい、あるいはさらに次々世代の「読み履歴ケース」を表示させたいといった要望が出てくるでしょう。

このように「本の読み履歴ケース」を切替える操作は、以下の通りです。

一つ前に戻る …… Hyper + R メニュー中 “Prev Reading Case”

一つ次のものを見る …… Hyper + R メニュー中 “Next Reading Case”

8.3 「本の読み履歴ケース」を作成する方法

今ここで表示されているのは、「COSMOS 操作オンラインマニュアル」という本のための、あらかじめシステムによって作成された「本の読み履歴ケース」です。

ここでは、このケースの作成手順を説明します。

8.3.1 「本の読み履歴ケース」の作成方法 ～ その一 ～

1. 「カードシステムマネージャ」上の TOOL-BOX 中の「本の読み方作成ケース」を開きます。
2. Hyper + Mouse-R クリックメニュー中の「Create Reading Case」を Mouse-L クリックします。
3. 「本の読み履歴ケース」の項目を作成するためのポップアップが表示されます。
 - ・ 項目の実質内容となるオブジェクト (複数指定が可能)
 - 1 個の場合 …… そのまま表示
 - 2 個の場合 …… 画面分割表示
 - 3 個以上の場合 ……

次のようなオブジェクトをマウスでキャッチし、Mouse-L クリック によって 指定します。

”section”, ”subsectoin”, ”subsubsection”

”keyword”, ”paragraph”, その他リンク等

*また、他の「本の読み履歴ケース」をオブジェクト として指定することも可能です。

- ・ 項目タイトル

タイトルを指定しなければ、項目として指定したオブジェクトの タイトルがセットされます。

4. 項目を指定終了する時は、このポップアップに何も入力せずに「DONE」します。

5. 「本の読み履歴ケース」のタイトルを指定します。

8.3.2 「本の読み履歴ケース」の作成方法 ～ その二 ～

図式表示による「本の読み履歴ケース」を作成する方法を以下の通りです。

キーワードリンクによる階層構造表示（プレゼンテーションタイプ）をキャッチし、Mouse-L クリックすると、この階層構造表示をとりこんだケースができます。

第 9 章

COSMOS オブジェクト一覧

COSMOS におけるオブジェクトの機能、作成状況等の情報を以下に整理します。

★ワープロ化文書の cosmos オブジェクトへの変換ケース

機能概要	COSMOS 初期操作環境を設定するための実行ケースを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	create manager
存在個数 (条件)	only one
認識方法	:keyword-string="welcome"
対応スクリプト	:welcome

★本棚ケース

機能概要	「本」オブジェクトを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	create-book
存在個数 (条件)	only one
認識方法	:keyword-string="book-case"
対応スクリプト	:book-case

★本ケース

機能概要	「巻 (TABLE-OF-CONTENTS)」オブジェクトを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create-book
存在個数 (条件)	複数
認識方法	:keyword-string="book"
対応スクリプト	:book

★目次（巻）ケース

機能概要	「章、節」オブジェクトを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create-book
存在個数（条件）	複数
認識方法	:keyword-string="table of contents"
対応スクリプト	:table-of-contents

★章ケース、節ケース、パラグラフカード

機能概要	「章、節」ケースは「パラグラフ」オブジェクトを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create-book
存在個数（条件）	複数
認識方法	:keyword-string="section","subsection","paragraph"
対応スクリプト	なし

★パラグラフケース

(パラグラフ毎のテンポラリなもの、当パラグラフは作成時に trash-case に格納される)

機能概要	パラグラフの内容、付加情報（キーワード、リンク等）を表示する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	paragraph open
存在個数（条件）	複数
認識方法	:keyword-string="paragraph-case"
対応スクリプト	:paragraph-case

★キーワード

機能概要	パラグラフから抽出した重要語句であり、そのパラグラフを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	キーワード作成時
存在個数（条件）	複数
認識方法	:keyword-string="key-word"
対応スクリプト	なし

★ This-is キーワード

機能概要	同名のタイトルを持つキーワードを格納するケースオブジェクトである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	キーワード作成時にもしなければ作成
存在個数 (条件)	複数
認識方法	:keyword-string= " thisis-keyword "
対応スクリプト	なし

★視点

機能概要	キーワードがそのパラグラフ内で、その何について書かれているかを示すもので、キーワードを格納する
大域変数管理の有無	yes
変数名	*view-point*
Select-A 画面表示の有無	no
作成タイミング	キーワード作成時にもしなければ作成
存在個数 (条件)	複数
認識方法	:keyword-string= " view-point "
対応スクリプト	なし

★パラグラフウィンドウケース

機能概要	
大域変数管理の有無	
変数名	
Select-A 画面表示の有無	
作成タイミング	
存在個数 (条件)	
認識方法	
対応スクリプト	:paragraph-window-case

★文間リンク (参照) の自動獲得ケース

機能概要	文間リンク (参照) の自動獲得を実行するためのケースである
大域変数管理の有無	:no
変数名	:no
Select-A 画面表示の有無	:yes
作成タイミング	:init-extract-key-term
存在個数 (条件)	1
認識方法	:keyword-string="set-bpl "
対応スクリプト	:set-bpl (*set-bpl-script*)

★重要語の切り出しケース

機能概要	重要語の切り出しを実行するためのケースである
大域変数管理の有無	yes
変数名	*term-manager*
Select-A 画面表示の有無	yes
作成タイミング	init-extract-key-term
存在個数 (条件)	1
認識方法	:keyword-string= " term manager "
対応スクリプト	なし

★キーワードエディタケース

機能概要	キーワードを編集したり、キーワードリンクによる階層構造作成のためのケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= " keyword-editor "
対応スクリプト	:keyword-editor (*keyword-editor-script*)

★キーワードメモケース (大域変数管理をしているもの)

機能概要	
大域変数管理の有無	yes
変数名	*key-word-memo-case*
Select-A 画面表示の有無	no
作成タイミング	(create-keyword-memo-case) から作成する。
存在個数 (条件)	1
認識方法	:keyword-string= " keyword memo "
対応スクリプト	:keyword-memo (*key-word-memo-case-script*)

★キーワードメモケース (大域変数管理をしていないもの)

機能概要	キーワードに関連して、整理しておきたいメモ事項を記述する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	キーワードメモケースの作成ケースから作成する。
存在個数 (条件)	複数
認識方法	:keyword-string= " keyword memo "
対応スクリプト	:keyword-memo (*key-word-memo-case-script*)

★検索ケース (汎用検索ケースを作成するケース)

機能概要	汎用検索ケースを作成するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= " tool-create-retrieve "
対応スクリプト	:retrieve-window (*tool-create-retrieve-script*)

★汎用検索ケース

機能概要	PROLOG 式による検索を実行するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	"検索ケース" の object-window-menu で作成する。
存在個数 (条件)	複数
認識方法	:keyword-string= " retrieve-window "
対応スクリプト	:retrieve-window (*retrieve-window-case-script*)

★文間リンク表示ケース (キーワード関連文間リンク表示ケース)

機能概要	キーワード関連の文間リンクを表示するケースである
大域変数管理の有無	yes
変数名	*key-word-display-bpl-case*
Select-A 画面表示の有無	no
作成タイミング	com-show-bpl-from-key-word 時に存在しなければ作成。
存在個数 (条件)	1
認識方法	:keyword-string= " display-bpl-case "
対応スクリプト	:show-bpl (*display-bpl-window-case-script*)

★文間リンク表示ケース (巻、章、節関連文間リンク表示ケース)

機能概要	巻、章、節関連の文間リンクを表示するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	com-show-bpl
存在個数 (条件)	複数
認識方法	:keyword-string= " display-bpl-case "
対応スクリプト	:show-bpl (*display-bpl-window-case-script*)

★検索 BPL ケース

機能概要	検索結果関連の文間リンクを表示するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= "retrieve-bpl-case "
対応スクリプト	なし

★文間リンク表示ケース

機能概要	検索結果関連の文間リンクを表示するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	汎用検索ケース作成時
存在個数 (条件)	汎用検索ケース毎に 1 個
認識方法	:keyword-string= "display-retrieve-bpl-case "
対応スクリプト	:show-bpl (*display-retrieve-bpl-window-case-script*)

★キーワード階層構造表示 (キーワード階層構造表示ケース)

機能概要	キーワード階層構造を表示するケースである
大域変数管理の有無	yes
変数名	*keyword-hierarchy-case*
Select-A 画面表示の有無	no
作成タイミング	(create-popup-window)
存在個数 (条件)	1
認識方法	:keyword-string= "display keyword hierarchy case "
対応スクリプト	:keyword-hierarchy (*keyword-hierarchy-script*)

★検索 Hierarchy Case

機能概要	検索結果関連キーワード階層構造表示ケースを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= "retrieve-hierarchy-case "
対応スクリプト	なし

★キーワード階層構造表示 (検索結果関連キーワード階層構造表示ケース)

機能概要	検索結果関連のキーワード階層構造を表示するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	汎用検索ケース作成時
存在個数 (条件)	汎用検索ケース毎に 1 個
認識方法	:keyword-string= " display-retrieve-keyword "
対応スクリプト	:keyword-hierarchy (*display-retrieve-keyword-script*)

★キーワード階層構造表示 (視点内キーワード階層構造表示ケース)

機能概要	視点内のキーワード階層構造を表示するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	com-show-hierarchy-with-this-view-point
存在個数 (条件)	各視点毎に 1 個
認識方法	:keyword-string= " view-point hierarchy case "
対応スクリプト	:keyword-hierarchy-view (*view-point-hierarchy-case-script*)

★キーワード階層構造表示ケース

機能概要	キーワード階層構造を表示するケースを作成するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	キーワード階層表示ケースの作成の object-window-menu から
存在個数 (条件)	複数
認識方法	なし
対応スクリプト	:keyword-hierarchy(*keyword-hierarchy-case-script*)

★キーワード (視点) ケース (Select-A 画面に表示されているもの)

機能概要	キーワードを視点毎に整理して表示するケースアイコンである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	keyword-string= " keyword case "
対応スクリプト	:keyword-case (*keyword-case-script*)

★キーワード (視点) ケース (Select-A 画面に表示されないもの)

機能概要	キーワードを視点毎に整理して表示するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	キーワードケースの作成ケースの object-window-menu から。
存在個数 (条件)	複数
認識方法	keyword-string= "keyword case "
対応スクリプト	:keyword-case (*keyword-case-script*)

★視点自動獲得用辞書カード

機能概要	視点の自動獲得実行のための構文解析用の辞書を定義する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= "dictionary "
対応スクリプト	:view-acquire-dic (*view-acquire-dic-script*)

★視点自動獲得用文法カード

機能概要	視点の自動獲得実行のための構文解析用の文法を定義する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= "grammar "
対応スクリプト	:view-acquire-gram (*view-acquire-gram-script*)

★視点自動獲得用視点決定ルールカード

機能概要	視点自動獲得時、文章パターンから視点を決定するルールを定義する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	init-create-cosmos-objects
認識方法	: keyword-string= "view-expert-rule "
対応スクリプト	なし

★視点自動獲得用視点決定ルールケース

機能概要	視点自動獲得用視点決定ルールカードを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
認識方法	:keyword-string= "view-expert-rule-case "
対応スクリプト	なし

★ツールケース

機能概要	諸機能を持つケース作成ケースを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= "tool-box "
対応スクリプト	:tool-box (*tool-box-script*)

★キーワード階層構造表示ケース作成ケース

機能概要	キーワード階層構造表示ケースを作成するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= "tool-create-keyword-hierarchy "
対応スクリプト	:tool-create-keyword-hierarchy (*tool-create-keyword-hierarchy-script*)

★キーワードケース作成ケース

機能概要	キーワードケースの作成を行なうケースでツールケースに格納されている
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= "tool-create-keyword-case "
対応スクリプト	:tool-create-keyword-case (*tool-create-keyword-case-script*)

★キーワードメモケース作成ケース

機能概要	キーワードメモケースを作成するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= " tool-create-keyword-memo-case "
対応スクリプト	:tool-create-keyword-memo-case (*tool-create-keyword-memo-case-script*)

★文分割ケース

機能概要	文分割による変換ファイルの作成ケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	文分割ケース作成ケースから作成する
存在個数 (条件)	1
認識方法	:keyword-string= " text-to-sentence "
対応スクリプト	:text-to-sentence (*text-to-sentence-script*)

★文分割ケース作成ケース

機能概要	文分割ケースを作成するケースである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= " text-to-sentence-case "
対応スクリプト	:text-to-sentence-case (*tool-create-text-to-sentence-case-script*)

★検索ルールカード

機能概要	検索ルールを定義する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	任意
存在個数 (条件)	複数
認識方法	:keyword-string= " retrieve "
対応スクリプト	なし

★類似語定義カード

機能概要	用語整理カードとして、キーワードや視点等の類似関係を定義する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	任意
存在個数 (条件)	複数
認識方法	:keyword-string= "similar-word "
対応スクリプト	なし

★座標カード

機能概要	COSMOS オブジェクトの初期画面上の座標データを定義する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create-manager
存在個数 (条件)	1
認識方法	:keyword-string= "coordinates-of-cosmos-object "
対応スクリプト	なし

★疑問ケース

機能概要	疑問カードを格納する
大域変数管理の有無	yes
変数名	*question-case*
Select-A 画面表示の有無	yes
作成タイミング	(set-question-card) 時にもし存在しなければ作成
存在個数 (条件)	1
認識方法	:title= "Question Case "
対応スクリプト	なし

★疑問カード

機能概要	意味不明の語句や理解出来なかった語句等を書き留めておくカードである
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	(set-question-card)
存在個数 (条件)	1
認識方法	:keyword-string= "question-card "
対応スクリプト	なし

★SDL ケース

機能概要	SDL 図オブジェクトを格納する
大域変数管理の有無	yes
変数名	*cosmos-sdl-case-in-this-manager*
Select-A 画面表示の有無	yes
作成タイミング	(open-gr) 時にもし存在しなければ作成
存在個数 (条件)	1
認識方法	:keyword-string= "sdl temporary case "
対応スクリプト	:sdl-case (*sdl-case-script*) 今はまだない！！

★ Book-Title カード

機能概要	「本」オブジェクトのタイトルをとりまとめる
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= "book-title "
対応スクリプト	:book-title (*book-title-card-script*)

★ Fig Tab File カード

機能概要	図表のタイトルおよび番号をとりまとめる
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	init-create-cosmos-objects
存在個数 (条件)	1
認識方法	:keyword-string= "fig-tab-file "
対応スクリプト	:fig-tab-file (*fig-tab-file-card-script*)

★ Text Sequence リンク

機能概要	本の章、節、パラグラフ間の順序関係を管理する
大域変数管理の有無	no
変数名	
Select-A 画面表示の有無	no
作成タイミング	create-book
存在個数 (条件)	1
認識方法	:title= "text sequence "
対応スクリプト	なし

★ 疑問リンク

機能概要	疑問の発生したパラグラフと、疑問カードとを関連づけるリンクである
大域変数管理の有無	yes
変数名	*question-link*
Select-A 画面表示の有無	no
作成タイミング	(set-question-card) 時にもし存在しなければ作成
存在個数 (条件)	1
認識方法	:title= "question-link "
対応スクリプト	なし

★検索ケースリンク

機能概要	汎用検索ケースと、それに対応するキーワード階層構造表示ケースとを関連づけるリンクである
大域変数管理の有無	yes
変数名	*retrieve-case-link*
Select-A 画面表示の有無	no
作成タイミング	汎用検索ケース作成時にもし存在しなければ作成
存在個数 (条件)	1
認識方法	:title= "Retrieve Case Link "
対応スクリプト	なし

★キーワードリンク

機能概要	キーワード間の関係を管理する
大域変数管理の有無	yes
変数名	*key-word-link*
Select-A 画面表示の有無	no
作成タイミング	キーワード作成時にもし存在しなければ作成
存在個数 (条件)	1
認識方法	:keyword-string= "key-word-link "
対応スクリプト	なし

★文間リンク

機能概要	パラグラフ間の関係を管理する
大域変数管理の有無	yes
変数名	*between-paragraph-link*
Select-A 画面表示の有無	no
作成タイミング	キーワード作成時にもし存在しなければ作成
存在個数 (条件)	1
認識方法	:keyword-string= "between-paragraph-link "
対応スクリプト	なし

★検索結果オブジェクト格納ケース

機能概要	検索結果をこのケースに格納してから、このケースを BPL 表示ケースに格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	(show-bpl-from-keyword) (display-result-of-paragraph)
存在個数 (条件)	複数
認識方法	:keyword-string= "object for bpl "
対応スクリプト	なし

★ Initial Process Control ケース

機能概要	初期操作環境の設定に必要なデータカードを格納する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	yes
作成タイミング	create manager
存在個数 (条件)	1
認識方法	:keyword-string= "initial process control "
対応スクリプト	:initial-process-control

★ スクリプトカード

機能概要	COSMOS オブジェクトにセットされるスクリプトを管理する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create manager
存在個数 (条件)	1
認識方法	:keyword-string= "script "
対応スクリプト	:script-value-set (*script-value-set-script*)

★ Term Manager 辞書カード

機能概要	TTC 標準用語として切り出したい語句を定義する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create manager
存在個数 (条件)	1
認識方法	:keyword-string= "term dictionary "
対応スクリプト	:term-dictionary (*term-dictionary-script*)

★ 文間リンク定義カード

機能概要	文間リンクとするリンクのタイトルや属性を定義する
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create manager
存在個数 (条件)	1
認識方法	:keyword-string= "set-bpl-control "
対応スクリプト	:set-bpl-control (*ser-bpl-control-sript*)

★ COSMOS 操作説明書自動作成ケース

機能概要	COSMOS 操作説明書の自動作成を行なう
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create manager
存在個数 (条件)	1
認識方法	:keyword-string= "init-create-operation-manual "
対応スクリプト	:init-create-operation-manual

★本自動作成ケース

機能概要	本の作成を行なう
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create manager
存在個数 (条件)	1
認識方法	:keyword-string= "init-create-book "
対応スクリプト	:init-create-book

★重要語の自動切り出しケース

機能概要	「本」オブジェクトから、term-manager 辞書に従って 重要語の自動切り出しを行なう
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create manager
存在個数 (条件)	1
認識方法	:keyword-string= "init-extract-key-term "
対応スクリプト	:init-extract-key-term

★文間リンク自動作成ケース

機能概要	「本」オブジェクトから、文間リンク定義カードに従って 文間リンク自動作成を行なう
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create manager
存在個数 (条件)	1
認識方法	:keyword-string= "init-auto-bpl-acquisition "
対応スクリプト	:init-auto-bpl-acquisition

★ COSMOS 参照オブジェクト自動作成ケース

機能概要	COSMOS 参照オブジェクト (キーワードエディタ、キーワードケース等) の自動作成を行なう
大域変数管理の有無	no
変数名	no
Select-A 画面表示の有無	no
作成タイミング	create manager
存在個数 (条件)	1
認識方法	:keyword-string= "init-create-cosmos-objects "
対応スクリプト	:init-create-cosmos-objects

第 10 章

電子化ドキュメント化の仕様

10.1 はじめに

通信知識体系化では、文書からの知識整理を考えている。そのため支援ツールとして、

1. 電子化(ワープロ化)文書の作成
2. 文分割プログラム
3. SDL 図式エディタ
4. カードへの変換プログラム

が必要となる。

10.2 文書資料のワープロ化仕様

10.2.1 文書資料の変換

1. 記号はすべて全角で入力する。
数字、記号、漢字、ひらがな、カタカナはすべて全角。(特に、半角カナは使用禁止とする)
2. 読点(、)、句点(。)はすべて(、。)に統一する。
文章の中に、読点として(、)、句点として(。)があればすべて読点(、)、句点(。)に統一する。
ただし、アルファベットの区切り、数字の中、記号の中はすべてそのまま、(、)。(。)を用いる。
3. 入力は \LaTeX 形式に従う。章だて、節、段落付けは以下の形式で行う。
 - 章の前には \section {章の名前} を付ける。
 - 節の前には \subsection {節の名前} を付ける。
 - さらに細かい節の前には、 \subsubsection {節の名前} を付ける。
 - それよりさらに細かい節の前には、 \subsubsubsection {節の名前} を付ける。
 - 段落の区切りの前には、 \par を付ける。
 - 段落の区切りではなく、文の切れ目には、 \newline を付ける。
項目分け等で(。まる)がなく、改行(文間が空いているなど)がある場合、 \newline を用いる。
 - 段落の区切りには、 \par のみを付ける。 \newline は付けない。

- ページの始まりは、`\startpage{ページ番号}` を付ける。
また、ページの終わりでは、`\clearpage{ページ番号}` を付ける。
- ローマ数字はすべて (I、アイ)、(V、ブイ) で表現する。
例: I ⇒ I、IV ⇒ IV
- 記号 Km など はすべて 英文字に置きかえる。
例: Km ⇒ Km、① ⇒ (1)
- 章の名前、節の名前、図、表の名前の中に、・(中点)がある場合、ピリオドに置きかえる。
- ①は(1)に置きかえる。
- 図、表によって、文が切れている時は切らないで、続けて書く。
- マイナス(-)記号は英数のマイナスを用いる。この時、ひらがなの長音(ー)は使わない。
本文中のひらがなの長音(ー)は長音(一)のままで良い。

4. 目次、索引

- 目次は1つの section とする。項目毎に `\newline` を入れる。

```
例: \section{目次}
    1. 概要..... 1
    \newline
    2. 概念と用語..... 2
    \newline
    ...
```

- 索引は大見出しがある場合、1つの subsection とする。項目毎に `\newline` を入れる。

```
例: \subsection{U}
    unit data ユニットデータ ... 1
    \newline
    Unknown 不定 ..... 2
    \newline
    ...
```

5. 図、表について

図は、以下のように変換する。

```
\begin{figure 図の番号}
  図に対する説明、コメントなどがあれば書く。
  ...
  \caption{図のタイトル、標題}
\end{figure}
```

表についても同様に、以下のように変換する。


```
\begin{table 表の番号}
  表のデータを横並びに書く。
  例: a & b & c & \\ \hline
  ...
  \caption{表のタイトル、標題}
  表に対する説明、コメントなどがあれば書く。
\end{table}
```

6. 脚注について

本文中の対応するところで、`\footnote{ 脚注の内容 }`と書く。
ただし、(注)のように、本文中にある場合はそのまま本文として書く。

7. 以上まとめると、記号(コマンド)には以下の種類がある。

章だてについて、

```
\section{}
\subsection{}
\subsubsection{}
\subsubsubsection{}
```

ページの区切りについて、

```
\clearpage{}
\startpage{}
```

段落付けについて、

```
\par
段落の区切りではなく、(。まる)がなく文の切れめの時について、
\newline
```

表について、

```
\begin{table}
...
表の線は &
表の後ろの線は \hline
\caption{表の題名}
\end{table}
```

図について、

```
\begin{figure}
...
\caption{図の題名}
\end{figure}
```

脚注について
`\footnote{}`

その他の数学記号について
上線 `\overline{}`
上カッコ `\overbrace{}`
下カッコ `\underbrace{}`
文字の上につく場合、`\hat`

上付き、下付きに関しては、
たとえば、 x^2 (上付き) $\Rightarrow x \wedge 2$ 、 x_2 (下付き) $\Rightarrow x _ 2$ に変換する。

その他の記号については、ワープロ上にあるキーを使ってもよい。
*、・(中黒)、◎、○、△、▲、□、◇、×、←、→、_(記号で使われる下線)、~、
↑、/、 α 、 β 、などはそのまま全角で入力する。

10.2.2 SDL 図のイベントオブジェクト仕様への変換

SDL 図をイベントオブジェクト仕様(SALの形式)へ変換する。詳しくは、論文¹を参照のこと。

1. すべての状態に番号を付ける。原則としては、上から下、左から右(Prologと同じ)
2. 符号(+、-、|)を付ける。
+ は受信、- は送信を表す。| は or を表す。
3. 条件分岐があれば、それを分けて書く。
4. ループの場合は要注意

以下に例を示す。

```
p1::{\n  s0: +() -> s1; もし、注があればここに書く。 \n  s1: +() -> s2; \n  . . . \n  s16: +() -> exit; }\n\n\\begin{figure} 図 6-1/JT-I430{\nレイヤ1起動::{\n  s0: X\\under2 1=8 X\\under2 2=10->s1; \n  s1: F=0->s2; \n  s2: C=0->s3; \n  s3: 信号待ち; \n  s3:+(Eビット受信)->s4|+(データ要求提出受付)->s19; \n  s4:.if(E=1?=N)->s2|.if(E=1?=Y)->s5;
```

¹「並行プロセスを基本とした交換プログラム仕様の階層的検証法」(伊藤、市川(NTT))

```

s5:.if(C=11?=N)->s6|.if(C=11?=Y)->s3;
s6:C=C+1->s7;
s7:.if(C ≥ X\under2 1=N)->s3|.if(C ≥ X\under
2 1=Y)->s8;
s8:X\under2 1=8->s9;
s9:.if(C ≥ X\under2 2?=N)->s11|.if(C ≥ X\under
er2 2?=Y)->s10;
s10:X\under2 2=10->s11;
s11:.if(F=1?=N)->s3|.if(F=1?=Y)->s12;
s12:.if(C ≥ X\under2 1?=N)->s3|.if(C ≥ X\under
er2 1=Y)->s13;
s13:-(最初のDビット送出)->s14;
s14:Eビット待ち;
s14:+(Eビット)->s15;
s15:.if(D=E?=N)->s2|.if(D=E?=Y)->s16;
s16:.if(Dビットはまだあるか=N)->s18|.if(Dビットはまだある
か=Y)->s17;
s17:-(次のDビットの交信)->s14;
s18:i=1なら X\under2 1=9 i=2なら X\under2 2=11-
>s1;
s19:.if(優先クラス?=1)->s20|.if(優先クラス?=2)->s2
1;
s20:i=1->s22;
s21:i=2->s22;
s22:F=1->s12;}
\newline
D:Dチャンネルビット
\newline
E:Dエコーチャンネルビット
\newline
F:フレームが待状態であることを示すフラグ
\newline
C:Dエコーチャンネルで検出された連続した「1」の数
\newline
X\under2 1: 優先クラス1のしきい値
\newline
X\under2 2: 優先クラス2のしきい値
\newline
i : 優先クラス識別子
\newline
\caption{(CCITT I.430)D-チャンネルアクセスの実施法のSDL表示}
\end{figure}

```

10.2.3 SDL 図式エディタを用いた変換

SDL 図をイベントオブジェクト仕様 (SAL の形式) へ変換すると共に SDL 図式エディタを用いた変換も行なう。SAL への変換は、テキストとしての処理を考慮したものであり (検証など)、図式への変換は図の意味を考えながらの処理を考慮したものである。図 1 参照。

10.3 文分割プログラムについて

ワープロ化された文章をある構造を持った単位に分割する。そのプログラムの仕様を以下に記述する。

- 入力 form: (text-to-sentence 入力 file 名, &optional append-switch, 出力 file 名, default-chapter-number)
- 出力 form: 入力 file 名 + 章番号
入力 file に複数の章番号がある場合は、複数の file を出力する。ただし、append-switch が T の時は複数の file をまとめて一つにして出力する。このときは、出力ファイル名も同時に指定すること。
章番号が指定されていない章については、デフォルト章番号が採用される。複数存在するときは、デフォルト章番号を一つずつ加算して使用する。(デフォルトは 0)
- 管理すべき変数:
 1. 章番号
 2. 段落番号
 3. 文番号
- 定義:
 1. 章:
\`\section`、または \`\subsection` で始まり、次の \`\section`、\`\subsection` の手前で終る範囲をいう。
 2. 章番号:
\`\section`、または \`\subsection` の直後の { から } までの数字の . (ピリオド)、-(マイナス)・(中点) を -(ハイフオン) に変換したもの。また、図番号、表番号についても同様とする。
 3. 段落:
 - (a) 段落の切れ目
 - i. \`\section`、または \`\subsection` の直後の { } の後
 - ii. \`\par` の直後 (但し、図、表 中では、\`\par` は段落の切れ目とはならない)
 - (b) 図、表
 - i. \`\begin{table}`(または、`figure`) で始まり \`\end{...}` で終わる範囲をいう。
 - ii. 図、表は段落の一種とする。
 - iii. 図中の \`\footnote{` で始まり } で終わる範囲は図、表の一種とする。\`\caption{` と同じ扱である。
 4. 文:
 - (a) 通常の文の切れ目は、原則として (。まる) である。これ例外には、? がある。また、\`\newline` を含むこともある。
 - (b) 表、図中の文の切れ目の優先順位は、(。まる)、\`\hline`、"return"、\`\newline` の順である。

● 処理:

1. 文の分割

(a) \section、\subsection の処理

直前の章の file をクローズし、新たな file を作成する。段落番号、文番号をリセットする。

(b) "space" の処理

"space space" のような一個以上の "space" は単一の "space" に変換する

(c) \par の処理

i. 通常は段落番号を更新する。

ii. 直前の段落が空の場合は段落番号の更新は行なわない。

(d) \newline の処理

i. 通常は読み飛ばす。

ii. 表、図中の \newline は文番号の更新とする。

(e) .(ピリオド) の処理

.(ピリオド) は、そのまましておく。

(f) ,(コンマ) の処理

,(コンマ) は変換せず、そのまま出力する。

(g) "return" の処理

i. 通常は読み飛ばす。

ii. 表、図中の "return" はその表、図中に (。まる)、\hline、が無い場合に限り、その行を文と見なし、文番号の更新とする。

(h) 表、図中の処理

表、図が段落中に現われた場合は、現段落の終了後に表、図を置く。

(i) 段落の処理

段落が更新された時は \par + 段落番号を挿入する。

(j) 文の処理

文が更新された時は、文頭に \sen + 文番号 を挿入する。

(k) 行番号の処理

各行の先頭に、各章ごとに通しの 行番号を付ける。出力ファイルが一つのときは、連番とする。(append-switch が on の時)

(l) \startpage、\clearpage の処理

\clearpage の後の L は削除する 文の途中に \clearpage がある場合は前文の最後に \clearpage を書く。

(m) \hline の処理

文番号を更新する。ただし、図表類のみに限る。

(n) \caption の処理

\caption の後の }迄を一文とする

(o) その他

次の部分は変換テーブルを用いて変換する。この場合、後のスペースは削除する。テーブルは後でエディットできるようにすること。この部分は現在のところそのまま出力すること。

ただし、昔の電子化文書の場合は以下のように変換する。関数 (japanese:jis-to-char) を用いて、対応する JIS 漢字コードを出す。

(a)\bullet	• (JIS 漢字コード表 x(16進)2126)
(b)\rightarrow	→ (jis x217A)
(c)\under2	_(under bar)
(d)\sim	~(childer)
(e)\alpha	α (jis x2641)
(f)\ll	\ll(変換なし)
(g)\gg	\gg(変換なし)
(h)\ast	*(asterisk)
(i)\box	□ (jis x2222)
(j)\bigtriangleup	\bigtriangleup(変換なし)
(k)\times	× (jis x215F)
(l)\bigcirc	○ (jis x217B)
(m)\mu	μ (jis x264C)
(n)\omega	ω (jis x2658)
(o)\odot	◎ (jis x217D)
(p)\uparrow	↑ (jis x222C)
(q)\div	/(devide)
(r)\gamma	Γ (jis x2623)
(s)\doteq	\doteq(変換なし)
(t)\diamond	◇ (jis x217E)
(u)\wedge	(jis x224A)
(v)\hat2	^(hat)

ただし、次の部分はそのままとしておく。

- (a)\hat
- (b)\overline
- (c)\overbrace
- (d)\underbrace

変換テーブルに無い "\… " はそのままにし、別ファイルに書き出す。

2. インデックスの作成

(a) index から本文中の文をだす。

(b) 図、表の対応 index を文章中からだす。

==> 図 1 = clm02:>ikeno>text>teltst-2-2.text 3 \subsection 2-2 \fig 1 \page 7

(c) (a) と (b) を使って、単語、または図、表を、mouse sensiteve な 領域として、クリックしたらその対応表をつくり、本文を表示する。

以下にオリジナルの文章、および変換後の文章を示す。

10.3.1 オリジナルの文章

\section{4. 配線構成の形式}

\newline

```
\par
ユーザ・網インタフェースの電気的特性は、ユーザ構内の様々な配線構成についてのある特定の仮定に基づいて決定される。これらの仮定は、本標準の付録 1 に記述する追加事項とともに、4.1 節と 4.2 節の 2 つのおもな配線構成で確認される。図 4-1/JT-I430 にユーザ構内の配線の一般的参照構成を示す。
```

```
\newline
\clearpage{4}
\startpage{5}
\begin{figure 図 4-1/JT-I430}
TEO IAO TR TE1 IA1 TEn IAn B TR IB NT
\newline
TR: 終端抵抗
\newline
I: 電氣的インタフェース
\newline
B: NT が終端抵抗 (TR) を内蔵した場合の IB の位置
\caption{(CCITT I.430) ユーザ構内における配線の形態}
\end{figure}
```

```
\newline
\subsection{4.1 ポイント・ポイント構成}
```

```
\newline
\par
ポイント・ポイント配線構成とは相互接続回路上でただ 1 つの送信部とただ 1 つの受信部とが相互に接続されることを意味する。
```

```
\par
ポイント・ポイント構成を図 4-2/JT-I430 に示す。
\begin{figure 図 4-2/JT-I430}
TE TR TR NT
\newline
TR: 終端抵抗
\newline
\caption{(CCITT I.430) ポイント・ポイント構成}
\end{figure}
```

```
\clearpage{5}
\startpage{6}
\newline
\subsection{4.2 ポイント・マルチポイント構成}
```

10.3.2 変換後の文章

200 \section 4 {4. 配線構成の形式}

201 \par 1

202 \sen 1 ユーザ・網インタフェースの電気的特性は、ユーザ構内の様々な配線構成についてのある特定の仮定に基づいて決定される。

203 \sen 2 これらの仮定は、本標準の付録1に記述する追加事項とともに、4.1節と4.2節の2つのおもな配線構成で確認される。

204 \sen 3 図4-1/JT-I430にユーザ構内の配線の一般的参照構成を示す。

205 \endpage 4

206 \startpage 5

207 \fig 4-1

208 \sen 1 \begin {figure 図4-1/JT-I430}

209 \sen 2 TEO IA0 TR TE1 IA1 TEn IAn B TR IB NT

210 \sen 3 TR: 終端抵抗

211 \sen 4 I: 電氣的インタフェース

212 \sen 5 B:NTが終端抵抗(TR)を内蔵した場合のIBの位置

213 \sen 6 \caption {(CCITT I.430) ユーザ構内における配線の形態}

214 \sen 7 \end {figure}

215 \subsection 4-1 {4.1 ポイント・ポイント構成}

216 \par 1

217 \sen 1 ポイント・ポイント配線構成とは相互接続回路上でただ1つの送信部とただ1つの受信部とが相互に接続されることを意味する。

218 \par 2

219 \sen 1 ポイント・ポイント構成を図4-2/JT-I430に示す。

220 \fig 4-2

221 \sen 1 \begin {figure 図4-2/JT-I430}

222 \sen 2 TE TR TR NT

223 \sen 3 TR: 終端抵抗

224 \sen 4 \caption {(CCITT I.430) ポイント・ポイント構成}

225 \sen 5 \end {figure}

226 \endpage 5

227 \startpage 6

228 \subsection 4-2 {4.2 ポイント・マルチポイント構成}

第 11 章

ドキュメント解析プログラムの起動法

11.1 機能概要

ドキュメント解析プログラムは以下のモジュールで構成され、ドキュメント解析用のファイルとそのインデックス・ファイルを作成することができます。

1. テキスト・ファイル変換プログラム

電子化されたテキスト・ファイルをドキュメント解析用のファイルに変換します。

2. 単語のインデックス作成プログラム

ドキュメント解析用ファイルと単語を指定して、ファイル内での参照インデックスを作成します。

3. 図表のインデックス作成プログラム

ドキュメント解析用ファイルに含まれる図表の参照インデックスを作成します。

11.2 保存場所

ドキュメント解析プログラムは、以下のディレクトリに保存されています。

```
CLM01:>local>binary>text>
```

11.3 起動方法

実行方法は、各々以下のとおりです。

1. テキスト・ファイル変換

Command: Load File CLM01:>local>binary>text>text-to-sentence

Command: (text-to-sentence *input-file* &*optional append-sw* *output-file* *default-chapter-number*)

2. 単語のインデックス作成

Command: Load File CLM01:>local>binary>text>make-word-address

Command: (make-word-address *input-file* *word* *output-file* *output-sw*)

3. 図表のインデックス作成

Command: Load File CLM01:>local>binary>text>make-figure-address

Command: (make-figure-address *input-file* *output-file* *output-sw*)

第 12 章

ドキュメント解析プログラムの構成

ドキュメント解析プログラムは以下のモジュールより構成されています。

1. テキスト・ファイル変換プログラム

電子化されたテキスト・ファイルをドキュメント解析用のファイルに変換します。

2. 単語のインデックス作成プログラム

ドキュメント解析用ファイルと単語を指定して、ファイル内での参照インデックスを作成します。

3. 図表のインデックス作成プログラム

ドキュメント解析用ファイルに含まれる図表の参照インデックスを作成します。

第 13 章

ドキュメント解析用ファイルの構造

ドキュメント解析用ファイルは大きく章、段落、文という構造になっています。尚、段落には図、表を含みます。ファイルの各行は以下の様な形をしています。

ファイル内連番 ドキュメント内容識別子 ドキュメント内容

ドキュメント内容識別子には章、段落の始まりを示すものや、ひとつの文であることを示すも、ページの区切りを表すものがあります。

1. 章の始まり

`\section`、`\subsection` でドキュメント内容は章番号、章名が入ります。

2. 段落の始まり

図の時 `\fig`、表の時 `\tab`、以外は `\par` で、ドキュメント内容には図表の時は、図表番号、図表名が入り、それ以外は段落の番号が入ります。

3. 文を表す

`\sen` でドキュメント内容は文の内容が入ります。

4. ページの区切り

`\page` でドキュメント内容はページ番号が入ります。

第 14 章

プログラムの仕様説明

次ページ以降に以下のプログラムについての仕様を説明します。

`text-to-sentence` テキスト・ファイル変換プログラム

`make-word-address` 単語のインデックス作成プログラム

`make-figure-address` 図表のインデックス作成プログラム

14.1 text-to-sentence テキスト・ファイル変換プログラム

14.1.1 形式

(text-to-sentence input-filename &optional append-sw output-file default-chapter-number)

input-filename 入力ファイル名を指定する。

append-sw 出力ファイルをひとつにするかを指定する。

devide の時分割、append の時ひとつに出力する。(デフォルトは devide)

output-file append-sw が append の時出力ファイル名を指定する。

default-chapter-number 章番号が指定されていない章が現れた時にセットする章番号を指定する。

(複数存在する時は指定した値に 1 ずつ加えたものを使用する。デフォルトは 0)

14.1.2 機能

指定された入力ファイルに対して、以下の処理を行いドキュメント解析用ファイルを作成します。

1. 章によるファイル分割

章の始まりを示す `\section`、`\subsection` から次の章の始まりまでをひとつの章として、ファイルに出力します。

2. 段落の作成

段落は `\newline`、`\begin` (図表を表す) で始まり、次の段落の始まりまでをひとつの段落にします。(尚、図表は `\end` で終わります。)

3. 文の作成

段落は複数の文で構成されます。文は . ? 。 等で区切られます。

4. 文字列の変換

文中の \ で始まる文字列は変換テーブルを使用して変換します。

14.1.3 戻り値

正常に処理が終了した時は t、異常終了したときは nil とエラーメッセージを表すストリングがマルチ・バリューで戻ります。

14.1.4 使用例

(text-to-setnetce "clm01:>text>data>tel9.text" 'append "clm01:>text>data>tel9.htest")

14.1.5 制限

特に無し。

14.2 make-word-address 単語のインデックス作成プログラム

14.2.1 形式

(make-word-address input-filename word output-filename output-sw)

input-filename 入力ファイル名を指定する。

word 検索する単語を指定する。

output-filename 作成したインデックスを出力するファイル名を指定する。

output-sw 出力を規存のファイルにアペンドするかどうかを指定する。

14.2.2 機能

指定された入力ファイルと単語から以下の形式のインデックスを作成して指定されたファイルに出力します。

単語 = 入力ファイル名 ファイル・アドレス 章名 段落名 段落内番号 ページ

14.2.3 戻り値

正常に処理が終了した時は t、異常終了したときは nil とエラーメッセージを表すストリングがマルチ・バリューで戻ります。

14.2.4 使用例

(make-word-address "clm01:>text>data>te19.htext" "単語")

14.2.5 制限

特に無し。

14.3 make-figure-address 図表のインデックス作成プログラム

14.3.1 形式

(make-figure-address input-filename output-filename output-sw)

input-filename 入力ファイル名を指定する。

output-filename 作成したインデックスを出力するファイル名を指定する。

output-sw 出力を規存のファイルにアペンドするかどうかを指定する。

14.3.2 機能

指定された入力ファイルから以下の形式のインデックスを作成して指定されたファイルに出力します。

単語 = 入力ファイル名 ファイル・アドレス 図表名 ページ

14.3.3 戻り値

正常に処理が終了した時は t、異常終了したときは nil とエラーメッセージを表すストリングがマルチ・バリューで戻ります。

14.3.4 使用例

```
(make-figure-address "clm01:>text>data>te19.htext" "clm01:>text>index>te19.dat")
```

14.3.5 制限

特に無し。