

Internal Use Only (非公開)

TR-SLT-0034

Interchange Format タグ付与ガイドラインおよび
タギングツールマニュアル
Guideline for Interchange Format Tagging and Tagging Tool Manual

浅見 克志
Katsushi ASAMI

2003年3月17日

Interchange Format は発話の意味を表現する中間言語である。この中間言語の表現形式は音声翻訳研究国際コンソーシアム C-STAR (Consortium for Speech Translation Advanced Research) が定めたものである。(現在は欧米の研究機関・大学で構成される NESPOLE! (Negotiating through Spoken Language in e-commerce) プロジェクトに継承)

本テクニカルレポートでは、この中間言語による意味表現を旅行会話基本表現集(BTEC)に付与した際の、付与指針について述べる。また、付与作業に使用するタギングツールのマニュアルも含む。

(株) 国際電気通信基礎技術研究所
音声言語コミュニケーション研究所
〒619-0288 「けいはんな学研都市」 光台二丁目2番地2 TEL : 0774-95-1301

Advanced Telecommunication Research Institute International
Spoken Language Translation Research Laboratories
2-2-2 Hikaridai "Keihanna Science City" 619-0288, Japan
Telephone: +81-774-95-1301
Fax : +81-774-95-1308

©2003 (株) 国際電気通信基礎技術研究所
©2003 Advanced Telecommunication Research Institute International

もくじ

1	付与タグ	2
1.1	Speech Act (発話行為)	2
1.2	Concept (概念)	3
1.3	Argument, Value (属性名、属性値)	3
2	タギング単位と発話分割	4
3	“no-tag” とする発話	4
4	入力フォーマット	5
5	タギングツール及びIFタグ付き旅行会話基本表現集データの管理	5
6	文献	5
付録1	Speech Act 付与例	6
付録2	IF タギングツール 操作マニュアル	16

1 付与タグ

- Interchange Format(IF)[1]に基づく Dialogue Act(:DA)[2]を SDU 毎に付与する。
- DA≡Speaker: Speech Act [+ Concept [+ (Arguments)]]
(大規模旅行対話表現集へのタギングの際には、Speaker は省略する)
- 該当する DA が存在しない場合、可能な限り Speech Act もしくは Speech Act + Concept を付与する。該当する Speech Act が存在しない場合、“no-tag”とする。

1.1 Speech Act (発話行為)

- DA リスト (“concept-arg.hier.db.exp.txt”) を遵守。このリストは、IF タギングツールに内蔵されているので、このツールを用いて選択入力する限り、逸脱はない。
- 文脈を考慮して決定する。ただし、大規模旅行会話表現集の場合、基本的に各表現は独立しているので、文脈を考慮する必要はない。

例) 「c:それは空席はありますか？」 「a:はい。」

→ 「はい」は affirm

例) 「c:はい、観光局でございます。」

→ 「はい」は greeting

表 1 主な Speech Act

Speech Act	コメント
accept	申し出や提案に対する受理。受け入れ。
acknowledge	話者が「分かっている」「理解した」ということを相手に伝達。
affirm	既知の事象の確認。確認の問いに対する肯定。
greeting	会話の始まりを示すフレーズ
closing	会話の終了を示すフレーズ
introduce-self	話者の個人情報(名前etc)や所属などを相手に示す
introduce-topic	話題の導入。前振り。
thank	感謝の意を表す
offer	話者が主体となる行為などの提案。「お手伝いしましょうか？」など
please-wait	待機の依頼。
apologize	謝罪。申し訳ない気持ちの表示。
give-information	情報の提示。
request-information	新しい情報提示の要求
request-suggestion	提案／提示の要求
request-affirmation	確認の要求。再確認。いいかどうかの念押し。
request-action	聞き手に行動／動作を要求／依頼
suggest	聞き手が主体となる行為などの提案。おすすめ。
negate	否定。
reject	提案や意見への拒絶。切り上げ。
verify	前項目までの内容の確認。「(×事象×)ですね」など
request-verification	前項目までの内容の確認の要求。「(×事象×)ですよ」など

1.2 Concept (概念)

- DA リスト (“concept-arg.hier.db.exp.txt”) を遵守。このリストは、IF タギングツールに内蔵されているので、このツールを用いて選択入力する限り、逸脱はない。
- 文脈を考慮せず、該当発話の情報のみから決定する。(文脈から明らかでも、該当発話から識別できない場合は、概念を展開しない。) 大規模旅行会話表現集の場合、基本的に各表現は独立しているので、文脈による概念の展開は考えにくい。ただし、常識的・一般的な状況を想定しての概念の展開を行なわないように注意する。

例) 「それは空席はありますか？」

→ (可能) request-information + availability

→ (不可) request-information + availability + flight

→ (不可) request-information + availability + transportation

1.3 Argument, Value (属性名、属性値)

- 原則として、属性名=属性値の定義リスト「argument-values.txt」に従うが、常識的な範囲で未定義のものも用いる。(迷うようなら用いない)

定義リスト「argument-values.txt」は IF タギングツールに内蔵されているので、このツールを用いて選択入力する限り、定義に従うことになる。未定義の属性名または属性値を使用した場合は、“IF タギングツール 操作マニュアル” に従い、追加登録しておく。

- 属性値に未定義のものを用いる場合は、その属性名に対して既に定義されている属性値から容易に類推できる範囲で追加する。

例) give-information + available + transportation(transportation-type=(subway & bus))

属性名 “transportation-type” の定義リストに属性値 “subway”、“bus” は無いが、他の定義属性値 “shuttle”、“flight”、“train” より類推される。

- 属性名に未定義のものを用いる場合は(主に “features”) は、他の Concept、または Speech-Act に定義されている属性名の範囲内で。

例) 「ニューヨークからロスまでです。」 : give-information + features(origin=new_york, destination=los_angels)

Concept “features” の定義リストに属性名 “origin”、“destination” は未定義だが、Concept “flight” 等には定義されている。

- 固有名詞 (人名、地名、施設名、等) は、リストに無い属性を用いてもよい。このとき、値は英発話の該当部分に基づいて作る、スペースは “ ” はアンダースコア “_” に書き換える。小文字は全て大文字に書き換える。

例) personname = naoko_suzuki

ただし、同一対象を指す別の値が既に登録されている場合は、新規登録を行なわない。

例) ホテル名に `kyoto_century` が既にあれば、`kyoto_century_hotel` は登録せず、`kyoto_century` を用いる。

- “;” (disjuncts)、“&” (conjuncts) は同種の属性値に対して用いる。異種の属性値を結合するときには、“,” (list) を用いる。

例) `transportation-type=(bus ; subway)`

例) `price=(currency=dollar, quantity=100)`

- 結合子 “;” (disjuncts)、“&” (conjuncts)、“,” (list) 間は任意の順序で並べてよい。

例) `price=(currency=dollar, quantity=100) ←→ price=(quantity=100, currency=dollar)`

2 タギング単位と発話分割

- タギング単位 = Semantic Dialogue Unit(:SDU)
- SDU の定義 (C-Star Web page より引用)

“An SDU(Semantic Dialogue Unit) is a portion of an utterance that has a dialogue act(There may be several dialogue acts in a multi-sentence utterance).”

- 発話が複数の文から構成される場合は、基本的に一文 (句点「。」単位) を 1 SDU としてタグを付与する。(入力フォーマットでは既に分割してあるので、作業者が特に注意する必要はありません)
- 重文・複文の場合は、以下の基準に従い、作業者が適切な SDU に分割の上、個々にタグを付与する。

- 焦点が異なるため、別 Concept を付与するのが妥当な場合は分割する。

例) 「便はニューヨークからロスまでで、ニューヨークは 1 3 時発です。」

→ 「SDU1:便はニューヨークからロスまでで、」、

「SDU2:ニューヨークは 1 3 時発です。」

- Speech Act が異なる重文は分割する。

例) 「すみません、満席となっております。」

→ 「SDU1:すみません、」「SDU2:満席となっております。」

- 慣用句として一続きで使われる表現は分割しない。

例) 「はい、そうです。」

例) 「エアパシフィックをご利用くださいませ、どうもありがとうございます。」

3 “no-tag” とする発話

以下に該当する発話に対しては、ドメイン外発話として “no-tag” を付与する。

- 適切な Speech Act が定義リストに存在しない場合。

4 入力フォーマット

- タグ入力は Excel 2000 で動作する IF タギングツールを利用して行なう。
- タギング対象の発話データは、Excel のブックファイルの Source シートにあるので、このシートの A 列に “IF:” と記述されている行にタグを入力する。
- A 列に “Comment” と記述されている行は、作業者が自由に使ってよい。
- “Comment” 行は必要に応じて追加してもよい。ただし、SDU 毎に連続した行にすること。
- 未定義の属性名や属性値を用いた場合、できるだけその旨を “Comment” 行に記述すること。
- 作業時に疑問点などが発生した場合は、“Comment” 行にその内容等を記述する。
- IF タギングツールの詳細は、“IF タギングツール 操作マニュアル” を参照のこと。

5 タギングツール及び IF タグ付き旅行会話基本表現集データの管理

本テクニカルポートに記述する IF タギングツールは、音声言語コミュニケーション研究所 第2研究室で管理する。

また、本テクニカルポートに記述するタグ付与指針に基づいて、旅行会話基本表現集(BTEC)の一部(約10000発話)にIFタグを付与した。このデータも音声言語コミュニケーション研究所 第2研究室で管理する。

なお、タグ付与作業は2001年6月～2001年11月に株式会社アイアール・アルトに依頼して実施した。この作業では、作業者の違いによる発話の解釈の揺れと、付与タグに対するその影響を最小限にとどめるため、作業者は1人とした。(担当作業者：同社 河野 香織氏)

6 文献

[1] <http://www.is.cs.cmu.edu/nespole/db/index.html>

[2] L. Levin, D. Gates, A. Lavie, and A. Waibel, “An interlingua based on domain actions for machine translation of task-oriented dialogues,” Proc.ICSLP-98, pp.1155-1158, 1998.

付録1 Speech Act 付与例

- (i) 以下に各 Speech Act の発話例を示す。いずれの場合も、あくまで例であり、実際のタグ付与にあたっては、想定される場面や相手話者の直前発話などを十分考慮する必要がある。例文中のアスタリスク(*)付の文は、特に注意が必要である。

accept

「それで結構です」

「結構です」*

「はい」*

「ええ」*

suggest、offer に対する答えなど

「〇〇はいかがでしょう？」=suggest

「〇〇いたしましょうか？」=offer

→「それをお願いします」；「はい」

「そうですね」「ええ、コーラを下さい」「それは妥当なところです」「そう思います」

など

acknowledge

「分かりました」

「かしこまりました」

「はい」*

request-action に対する答えなど

「禁煙席をお願いします」=request-action

→「はい、かしこまりました」「分かりました」「承知しました」など

affirm

「はい、そうです」

「そのとおりです」

「はい」*

「ええ」*

verify-affirm、request-information に対する答えなど

「〇〇でいいんですね？」=verify-affirm

「〇〇ですか？」=request-information

→「はい、その通りです」「はい、〇〇です」「大丈夫です」「それで十分です」「それ

で結構です」「あります」など

apologize

「お待たせいたしました」

「すいません」

「失礼ですが」

「あいにくですが」

謝罪の意。特に相手の発話に対する返答ではない。

強いて返すなら、相手にクレームを付けられたとき = give-information

「パソコンが壊れてるんですけど」 = give-information

→ 「大変申し訳ございません」

closing

「お越しをお待ちしております。」

「失礼いたします」

「よろしく願います」 *

「おやすみ」「また」

「気を付けてお帰りください」「今日は楽しかったです」

会話の終了。特に相手の発話に対する返答ではない。

別れの挨拶。

delay-action

「できるだけ、一週間前には、はっきりとお返事いたします。」

「十月三日に子供が増えるかどうか電話しますので、」

「じゃそれどちらかにするかはもう一度、ご連絡するようにいたします。」

「また電話します」「3日前までにはお返事いたしますので」

「彼にお引き合わせしましょう」

特に相手の発話に対する返答ではない。

今ではなく、あとで発話者が行動することを伝える。

give-information

「カードはビザカードですけど。」

「たぶん午後六時ぐらいになると思います。」

「お部屋の方はシングルツイントリプルとございますが。」

特に相手の発話に対する返答ではないとき、

「当店で、魚料理を得意と致しております」

request-information に対する答えのとき

「これは何ですか？」 = request-information

→「これは古代の食器です」

greeting

「もしもし」

「はい」＊

「こんにちは」

挨拶

introduce-self

「担当しましたのは、わたくしジョンでございますので。」

「ニューヨークシティホテルでございます。」

「二〇七号室の野澤です。」

特に相手の発話に対する返答ではない。

自分の名前・所属を相手に提示。「〇〇が好きです」などの情報は含まない。名乗る、身分を述べる発話。

introduce-topic

「十七日に関しては、」

「ダブルのお部屋の場合、」

「その予約のことでちょっとご相談したいんですが。」

「予約の事でご相談したいのですが」←“予約をする”わけではない

特に相手の発話に対する返答ではない。

キーワード＝“関して” “場合” など。本題の導入。

negate

「いいえ、」＊

「そうじゃなくって、」

「特に無いんですけども、」

「いいえ」「私は悪くありません」「確かではありません」など

相手発話に対する否定。

「〇〇ですか？」＝request-information

→「違います」

offer

「お客様のために予約をお取りしましょうか」

「そのほかになにかございますか」

「東京成田空港二十一時ご出発の便をご予約させていただきます」

「ご予約いたしましょうか？」 「〇〇をご予約させていただきます」

「おつりはとっておいてください」

特に相手の発話に対する返答ではない。

発話者の行為の意思表示。

please-wait

「少々お待ちくださいませ」

「ただいま確認をいたしておりますので、しばらくお待ちください」

「お調べいたしますので、少々そのままお待ちくださいませ」

特に相手の発話に対する返答ではない。

発話者が相手に待つように依頼する。

reject

「いいえ、以上です」

「結構です」 *

「ありません」

「賛成できません」

「いいえ、払い戻してください」

offer や **request-information** 等の提案や意見に対する拒絶。

request-action

「部屋のタイプを、変えてほしいんですけども。」

「お願いします。」 *

「宿泊の、予約をお願いしたいんですけど。」

「パーマをお願いします」

「電話があったと伝えてください」

特に相手の発話に対して返すわけではない。

聞き手に **action** を求める発話。もしくは、話者の目的、希望が相手の行為によって実現可能な場合。

request-affirmation

「お間違いないでしょうか。」

「それでもよろしいでしょうか。」

「その辺は大丈夫でしょうか。」

「これでお間違いないでしょうか？」

「わかった？」

「目を開けてもいいですか」「借りていいですか」

特に相手の発話に対する返答ではない。

それまでの発話内容に関して聞き手に確認を要求する。再確認であり念押し。

相手に自分が行動しても良いか尋ねる。

キーワード：「大丈夫」「いい」「間違いない」など+ですか？」

request-delay-action

「それでは、わたくしどもの都合もごございますので、一週間前に必ず、連絡お願いいたします。」

「お分かりになりましたら連絡頂けますでしょうか。」

「それでは、一週間ほどしましたら連絡を入れて頂きたいと思います。」

発話の任意時間後に相手に動作をするように要求。

delay-action + 「してもらえますか?」「してください」

request-information

「それと当日のですね、チェックインは何時頃になりますでしょうか。」

「人数の変更とかは、できますか。」

「連絡先の電話番号頂けますでしょうか。」

新たな情報を要求。

具体的な情報(時間・場所など)を要求するときと、

Yes か No かを要求するときとある。

request-suggestion

「ホテルの紹介を幾つかしてほしいんです」

「何かほかに方法はないでしょうか」

「どの方法が一番いいと思いますか」

相手に提案・提示(suggest)を要求。

キーワード：「おすすめ」「他に」「紹介」「いいと思いますか」など

request-verification-delay-action

「一週間前までに、じゃこちらの方からご連絡を入れるということでよろしいですか。」

request-verification-give-information

「それではトリプルのお部屋を御用意してよろしいでございますか。」

「四条河原町のニュー京都ホテルで間違いないでしょうか。」

「ホテルから出ているんですか。」*

cf 「ホテルから出ているんですね。」→verify-give-information

「ホテルから出ているんですか」は、相手が発した情報の確認を相手に求める意図がある。

一方、「ホテルから出ているんですね」は、相手が発した情報を話者自身が確認している。

request-verification-request-action

「二万円のお部屋でお取りしてよろしいでしょうか。」

「では、一旦この便をキャンセルいたしますが、よろしいでしょうか。」

「八月十一日の同じ二〇一便に変更をしたいということですか。」

request-verification-request-delay-action

「じゃ必ず、十月三日までに連絡をいただけますでしょうか。」

request-verification-request-suggestion

「午前中でよろしいでしょうか。」

suggest

「そうですね、小さいお子様連れでしたら、バスのほうが便利かと思いますが。」

「まず、京阪電車で七条駅から三条駅まで出ていただきます。」

「ではバッテリー公園までバスでいらっしやって、そこからご覧になってはいかがですか」

「〇〇はいかがでしょう?」「〇〇がいいですよ」等

相手に対して推奨案などを提案

「おすすめのコースは?」=request-suggest

→「本日のおすすめはAコースでございます」

thank

「ありがとう」「ありがとうございました」

「どうも」

相手の行為や情報提供に対する感謝の表示。

verify-give-information

「ワシントン工科大学のフィリップス様でいらっしやいますね。」

「京都駅からですね。」

「十日にこちらにご到着ということでございますね。」

verify-introduce-self

「京都観光ホテルさんですね。」

verify-offer

「シングルにキャンセルが出た場合には変えさせていただきます。」

verify-request-action

「お部屋の変更でございますね。」

「それでは、ご予約入れさせていただきますので」

「ご用意させていただきます。」

verify-request-delay-action

「じゃ、人数、確定しましたら三日前までにお電話いただけるようお願いします。」

「五日前までに連絡を下さい。」

verify-request-information

「わたしの電話番号も知りたいとおっしゃっていましたよね。」

- (ii) 以下に過去のタグ付与の事例に見られる、特に注意すべき事例について示す。ただし、実際にタグ付与を行うにあたって個々に示す指針に従うと不自然・不整合になる場合は、適宜 ATR 音声言語通信研究所の担当者に相談すること。

① 「はい、結構です」などの発話

- ・ accept、affirm で迷うことがある。

質問や確認に対する応答（の場面が想定される）としての発話の場合、affirm とする。

提案などに対する応答（の場面が想定される）としての発話の場合、accept とする。

② 「ほかに何かございますか」と「ほかに何かご変更とかございますか」

- ・ 「ほかに何かございますか」は offer + help-again とするが、「ほかに何かご変更とかございますか」は「変更」に影響されて request-information + change + features とする例がある。

この場合は、いずれの場合も offer + help-again とする。

cf 「どのようなご変更でしょうか」 → request-information + change + features

③ 「～たい」「～たいのですが」などの発話

- ・ give-information + preference、request-action で判断が難しい場合がある。基本的には次の指針でタグを付与する。

(a) 表層から相手への要求と判断できる（～してもらいたい、～していただきたい）

Yes → request-action No → (b)

(Yes の該当例)

- ・ 「お金を返してもらいたい。」
- ・ 「貴重品を預かっていただきたいのですが。」

(b) (特に SDU 分割した場合など) 前後の文から適切なタグを判断できる。

Yes → 適切なタグ No → (c)

(Yes の該当例)

- ・ 「この小包を日本に送りたいのですが／一番速い方法はなんですか？」

<前半> give-information + preference

<後半>request-information

前半部だけを見ると、request-action か give-information + preference か判断に迷うが後半部を考慮すると、前半部は直接的に動作要求をしていないと判断できるので、give-information + preference となる。

- (c) 発話内容の実現に相手の動作はほとんど必要ない場合、あるいは選択肢・属性や条件に関する希望や好みを発話する場合。文のニュアンスとして、「話者の要求」ではなく「話者の希望」と捉えるほうが自然と考えられる発話。

Yes → give-information + preference No → (d)

(Yes の該当例)

- ・「上半身を鍛えたい」、「泳ぎたい」 → 相手の動作はほとんど必要ない。
- ・「君の手を握りたい」 → 「話者の希望」と捉えるのが自然。
- ・「あと2,3日ここに泊まりたいのですが」 → 条件に関する希望を述べていると考えられ、かつ延長期間が曖昧であることから、「希望」的なニュアンスが強い。
- ・「濃いコーヒーが飲みたい」、「ワッフルが食べたい」 → 好みに関する内容と考えられ、「希望」的なニュアンスが強い。

- (d) 動作要求の発話であることを仮定した上で、その動作に直接対応する人との対話を想定でき、さらに発話内容から相手に要求している動作を特定できる。その動作は、相手が当然行うべき動作（サービス）である。

また、希望的なニュアンスが弱い。

相手に要求している動作を特定できる場合でも、無理にお願いしているような場合などは希望的ニュアンスが強いと判断する)

Yes → request-action No → give-information + preference

(Yes の該当例)

- ・「席を予約したいのですが」 → 相手話者：旅行代理店の窓口など
要求動作：予約手続き
- ・「貴重品を受け取りたいのですが」 → 相手話者：ホテルフロントなど
要求動作：預けた貴重品を出す
- ・「車を借りたいのですが」 → 相手話者：レンタカー業者
要求動作：車のレンタル手続き

(No の該当例)

- ・「友人と一緒に座りたいのですが」 → 相手話者：他の乗客
要求動作：席の交換

ここで、席の交換は他の乗客が当然行うべき動作ではなく、無理にお願いしていると考えられるので、希望的ニュアンスが強いと判断する。

- ・ 「ピーフシチューを分けて食べたいのですが」 → 相手話者：ウェイター
要求動作：特定できない

④ 「よろしくお願ひします」などの発話

状況に応じて **Speech Act** が変化する。

- ・ 対話の終わりの挨拶と考えられる場合は、**closing** とする。
- ・ 対話の終わりではなく「こちらこそ～」となるような場合は **greeting** とする。
(「こちらこそ」を補って置換しても対話の流れが損なわれない場合)
- ・ 対話の終わりではなく「それで～」となるような場合は **request-action** とする。
(「それで」を補って置換しても対話の流れが損なわれない場合)
- ・ 判断のつかないときは **request-action** とする。

⑤ 将来の可能性に関する発話

「もしかしたら子供が、来るかも知れないので。」

「一名直前に増える可能性があるんですけども。」

Speech Act は **give-information** とする。**Concept** には「将来の可能性」に関する定義がないので、暫定的に **Concept** 以下は記述しない。

⑥ 伝聞情報に関する発話

「そちらには、カジノのほかに、家族が楽しめる場所があると聞いたんですが」

Speech Act は **give-information** とする。**Concept** には「伝聞情報」に関する定義がないので、暫定的に **Concept** 以下は記述しない。

⑦ 「…よろしいでしょうか」などの発話で、他の指針では判断が難しい発話

- (a) 「～してもよろしいでしょうか」をサービスを受ける立場から発話、あるいはサービス関係が不明な場合

「許可を求めている」と考えると、現状のタグ定義ではタグ付与が難しいが、
「可能性に関する問合わせ」と考え、**request-information + availability** とする。

(例)

「この電話をお借りしてもよろしいでしょうか」

- (b) 「～してもよろしいでしょうか」をサービスを受ける立場から発話する場合
(ただし、相手の要求などに対する応答としての発話は除く)

相手に対するサービスの申し出と考え、offer とする。

(例)

「お皿をお下げしてもよろしいですか」

ただし、「お皿を下げてください」に対する応答である場合は、request-verification-request-action とする。

⑧その他判断に迷う事例

- ・ 「そうして頂けると、ありがたいんですが。」
相手に何らかの動作を要求しているので request-action
- ・ 「どの部屋でも、結構です。」
希望を示していると考えられるので、give-information + preference
- ・ 「じゃ、こちら、ツインとトリプル両方、おさえておきましょうか。」
相手に提案しているので、suggest + reservation + features + room (room-type=(twin & triple))
- ・ 「じゃ、そのお部屋とって頂けますか。」
give-information、suggest などの後にこのような発話がある場合、accept と迷うが、要求が明確に現れているので、request-action とする。

付録2 IF タギングツール 操作マニュアル

1	本ツールの概要.....	17
2	操作方法 – 基本.....	17
2.1	起動方法.....	17
2.2	タギング対象文の選択.....	17
2.3	Dialogue Act Editor フォーム (図 3)	17
2.4	Speech Act の選択.....	19
2.5	Concept の選択.....	20
2.6	Argument の選択.....	21
2.7	Argument-Value の入力.....	22
2.8	Speaker の選択.....	23
2.9	作成した DA 系列の確定.....	24
3	操作方法 – Argument の Value 入力時の特殊な操作.....	25
3.1	Value Class からの Value の選択.....	25
3.1.1	1 個の Class で構成される Value.....	25
3.1.2	複数の Class で構成される Value.....	25
3.2	Argument を含む Value の選択.....	26
3.2.1	1 個の Argument を含む Value.....	26
3.2.2	複数の Argument を含む Value.....	26
3.3	Value Class への新規 Value の追加.....	27
3.4	Argument への新規 Value の追加.....	28
3.5	Value の結合 (図 16)	29
3.6	Argument – Value の再編集 (図 17)	31
3.7	数字,アルファベットのリスト入力 (図 18)	32
3.8	マニュアル編集.....	32
4	発話分割.....	33

1 本ツールの概要

- 本ツールは Microsoft® Excel 2000 上で動作する Interchange Format(IF)タギングツールである。
- 本ツールは VBA で作成されており、IF タギング対象データ、IF 定義データと一体化された、一つのファイルで提供される。
- 本ツールにより、IF 定義データに示される仕様を満たす IF タグの付与作業を簡便に行なうことができる。

2 操作方法 — 基本

2.1 起動方法

- ・ Excel 2000 を起動し、本ツールを含む作業対象ファイルを読み込む。
- ・ 読み込みを開始すると、マクロ確認メッセージ (図 1) が表示されるので、[マクロを有効にする] をクリックする。

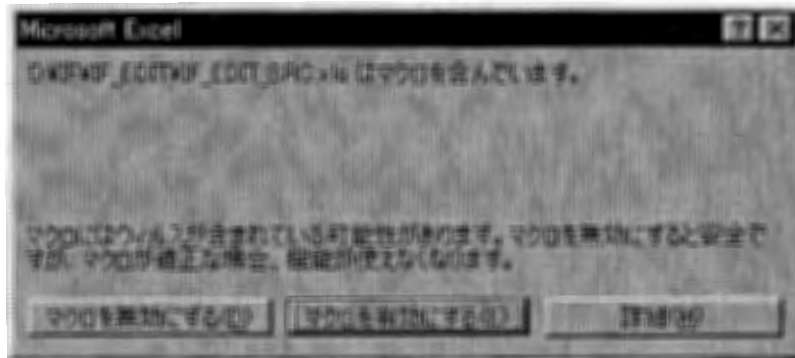


図 1 マクロ確認メッセージ

- ・ 作業ファイルは、(Source)(value)(value-class)(CombinationList)の4つのシートで構成される。(Source)以外のシートには、原則として直接変更を加えないこと。

2.2 タギング対象文の選択

- ・ 作業対象ファイルの(Source)シートで、A列が“IF:”となっている行をダブルクリックする (図 2) と、Dialogue Act Editor フォーム (図 3) が表示される。
- ・ Dialogue Act Editor フォームが表示されている間は、フォーム上の[Previous][Next]ボタンによって、タギング対象文の選択を前後に移動できる。

⚠ (Source)シートで、A列が“IF:” “Comment”以外になっている行およびA列は編集しないこと。A列が“IF:” “Comment”以外になっている行をダブルクリックすると警告メッセージが表示されるが、セルは書込み禁止になっていないので、不用意に変更しないよう注意する。

2.3 Dialogue Act Editor フォーム (図 3)

- ・ Source ボックスには、現在選択されているタギング対象文が選択されている。
- ・ Current DA ボックスには、現在選択されているタギング対象文に付与されている DA(Dialogue Act)が表示される。DA が付与されていなければ、何も表示されない。
- ・ Speech Act リストボックスには、Speech Act のリストが表示される。
- ・ Concept リストボックス(1~5)には、Speech Act、Concept の選択状況に応じて、選択可能な Concept のリストが表示される。

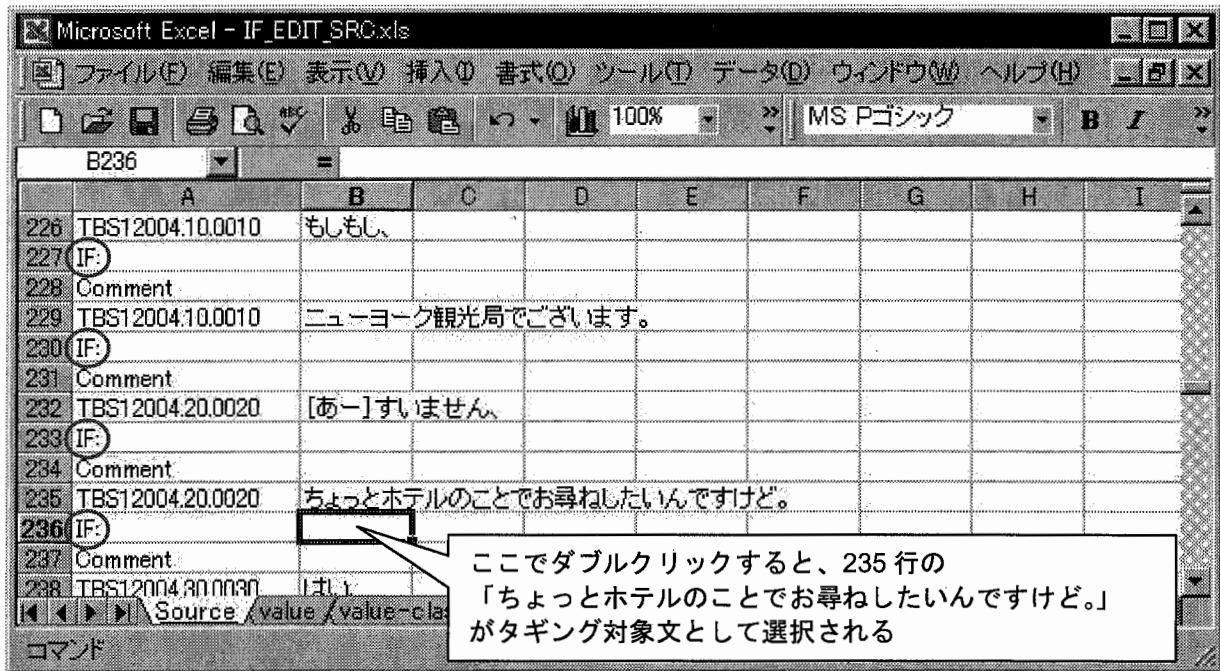


図 2 (Source)シートでのタギング対象文の選択

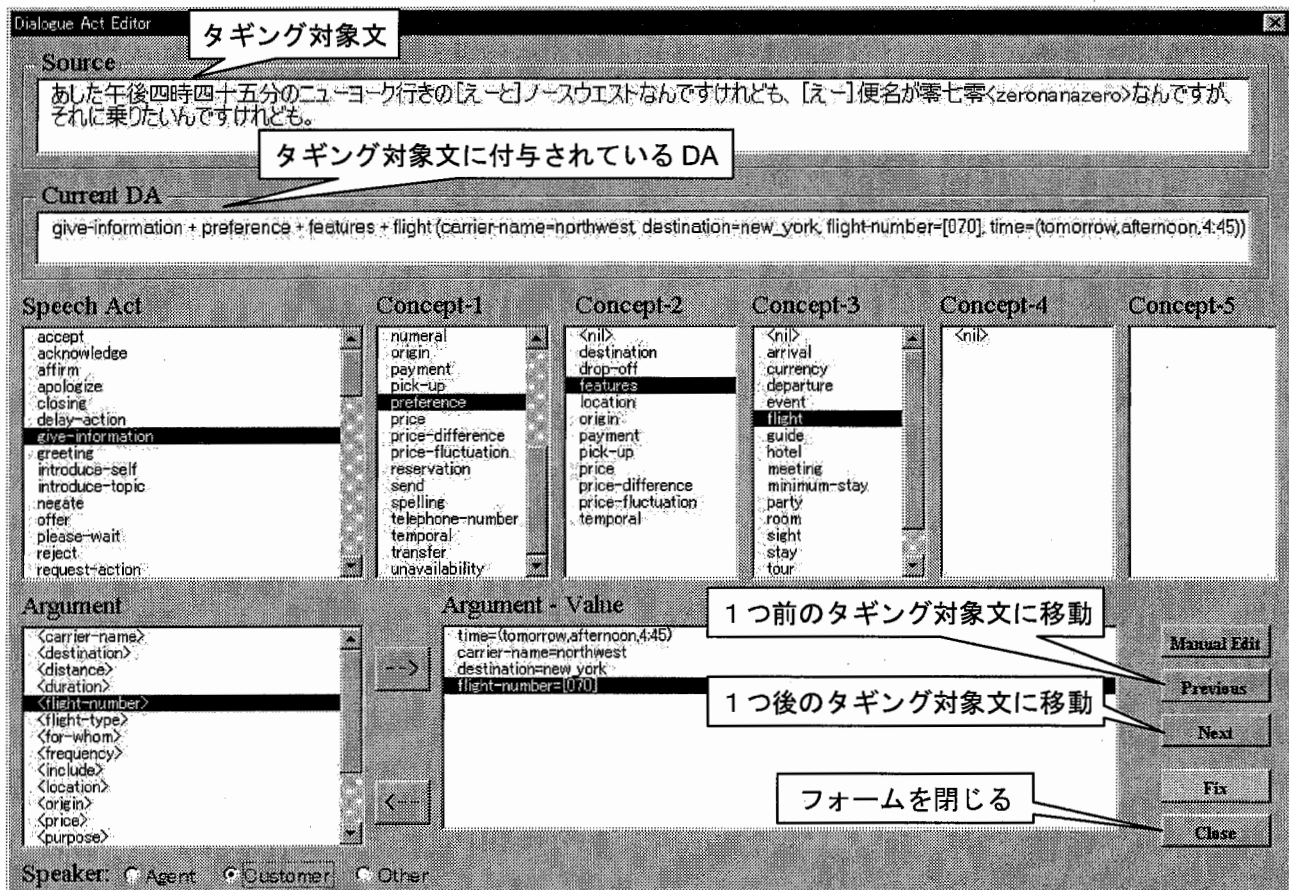


図 3 Dialogue Act Editor フォーム

- Argument リストボックスには、Speech Act、Concept の系列が一つに絞りこまれたとき、選択可能な Argument のリストが表示される。このリストは、Shift キー、Ctrl キーを押しながらクリックすることにより、複数の項目が選択可能である。
- [-->]ボタンは、Argument リストボックスで1個以上の項目を選択すると有効になる。この

ボタンをクリックすると、Argument リストボックスで選択した項目が Argument-Value リストボックスにコピーされる。

- [←]ボタンは、Argument-Value リストボックスで1個以上の項目を選択すると有効になる。このボタンをクリックすると、Argument-Value リストボックスで選択した項目が削除される。
- Argument-Value リストボックスで、リスト項目をダブルクリックすると、Argument Editor ダイアログボックスが表示されるので、選択された Argument (属性名) に対する Value (属性値) の入力を行なう。
- [Manual Edit]ボタンは、Argument-Value ボックスで1個以上の項目を選択すると有効になる。このボタンをクリックすると、“Argument Editor<Manual>”ダイアログボックスが表示され、式<属性名=属性値>をマニュアル編集できる。ただし、仕様を逸脱しないよう、注意が必要である。
- [Previous][Next]ボタンをクリックすると、タギング対象文の選択を前後に移動できる。
- [Fix]ボタンは、現在編集中の DA が、仕様に則って成立する状態 (well-formed な状態) にある場合、有効になる。このボタンをクリックすると、編集された DA が(Source)シートに転記される。同時に Current DA ボックスに表示される。ただし、Speaker を選択していない状態でクリックした場合は、編集した DA は転記されず、Speaker の選択を要求するメッセージが表示される。
[Fix]ボタンをクリックして、DA を(Source)シートに転記した後も、DA の編集・修正は可能である。
編集中の DA が well-formed な状態で、かつ(Source)シートに1回も転記していない間、このボタンのフェースは黄色で表示される。
- [Close]ボタンをクリックすると、Dialogue Act Editor フォームを閉じる。

2.4 Speech Act の選択

- Dialogue Act Editor フォームの Speech Act リストボックスで、タギング対象文にもっともふさわしいと思われる Speech Act をクリックして選択する。(図 4)
- Speech Act 以下の Concept, Argument, Value の選択/入力後でも、随時 Speech Act を再選択できる。ただしその場合、Speech Act 以下の Concept, Argument, Value はリセットされる。
- Speech Act を選択すると、選択された Speech Act に接続可能な Concept が Concept-1 リストボックスに表示される。

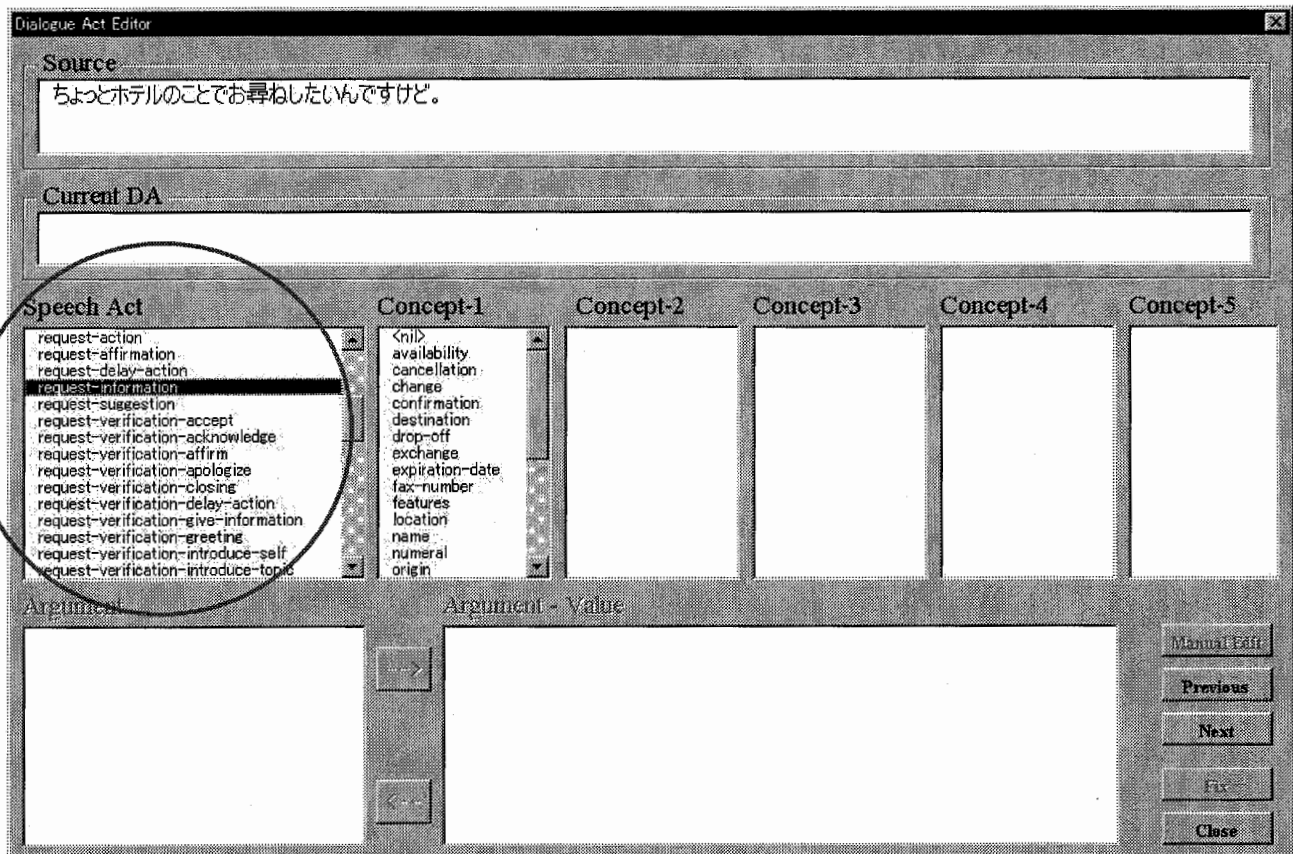


図 4 Speech Act の選択

2.5 Concept の選択

- Dialogue Act Editor フォームの Concept リストボックス(1~5)で、タギング対象文の内容の表現に必要な Concept をクリックして選択する。(図 5)
- Concept は複数個組合せが可能であるが、その組合せパターンは制限されているため、Concept の選択を進めていくと、接続可能な Concept が絞り込まれる。
- 最後の Concept リストボックスの項目が 1 個に絞り込まれたとき、そこまでの Speech Act - Concept の系列が 1 個に絞り込まれたことになる。この状態に達すると、Argument リストボックスに、その系列に対して選択可能な Argument のリストが表示される。

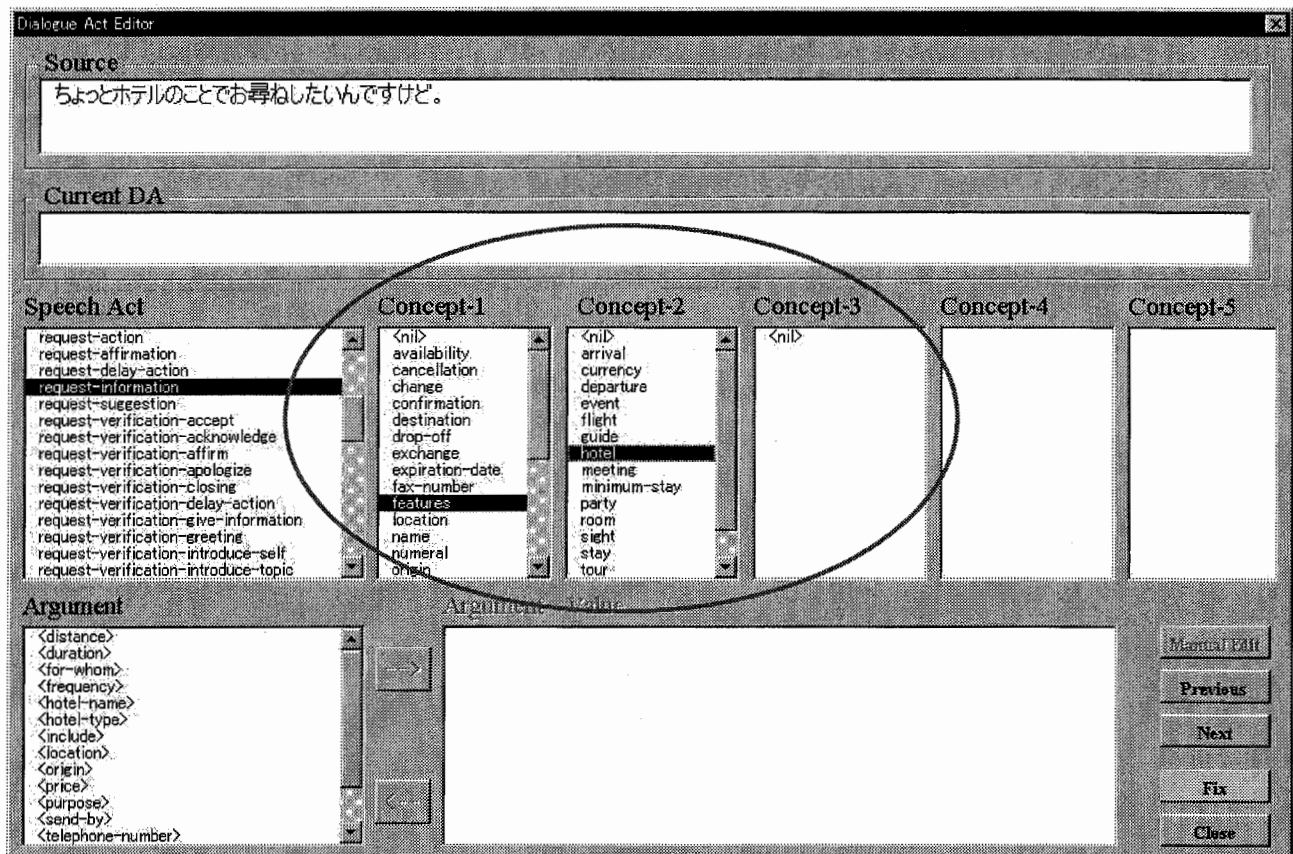


図 5 Concept の選択

2.6 Argument の選択

- Dialogue Act Editor フォームの Argument リストボックスで、タギング対象文の内容の表現に必要な Argument をクリックして選択する。(図 6) Argument を選択する必要が無いときは、そのまま、DA を確定する。(2.9 作成した DA 系列の確定)
- Shift キー、Ctrl キーを押しながらかlickすることにより、複数の項目が選択可能である。
- [->] ボタンをクリックすると、Argument リストボックスで選択した項目が Argument-Value リストボックスにコピーされる。
- Argument リストボックスの項目をダブルクリックした場合も、その項目が Argument-Value リストボックスにコピーされる。
- Argument リスト中の <OTHER-arg> を選択して [->] ボタンをクリックする、あるいは <OTHER-arg> をダブルクリックすると、Argument Index リストボックスが表示され、定義済み Argument の一覧から任意の Argument を選択できる(図 7)。このリストボックスでは、Shift キー、Ctrl キーを押しながらかlickすることにより、複数の項目が選択可能である。項目選択後、[OK] ボタンをクリックすると、選択した項目が Argument-Value リストボックスにコピーされる。
- Argument Index リストボックスの項目をダブルクリックした場合も、その項目が Argument-Value リストボックスにコピーされる。

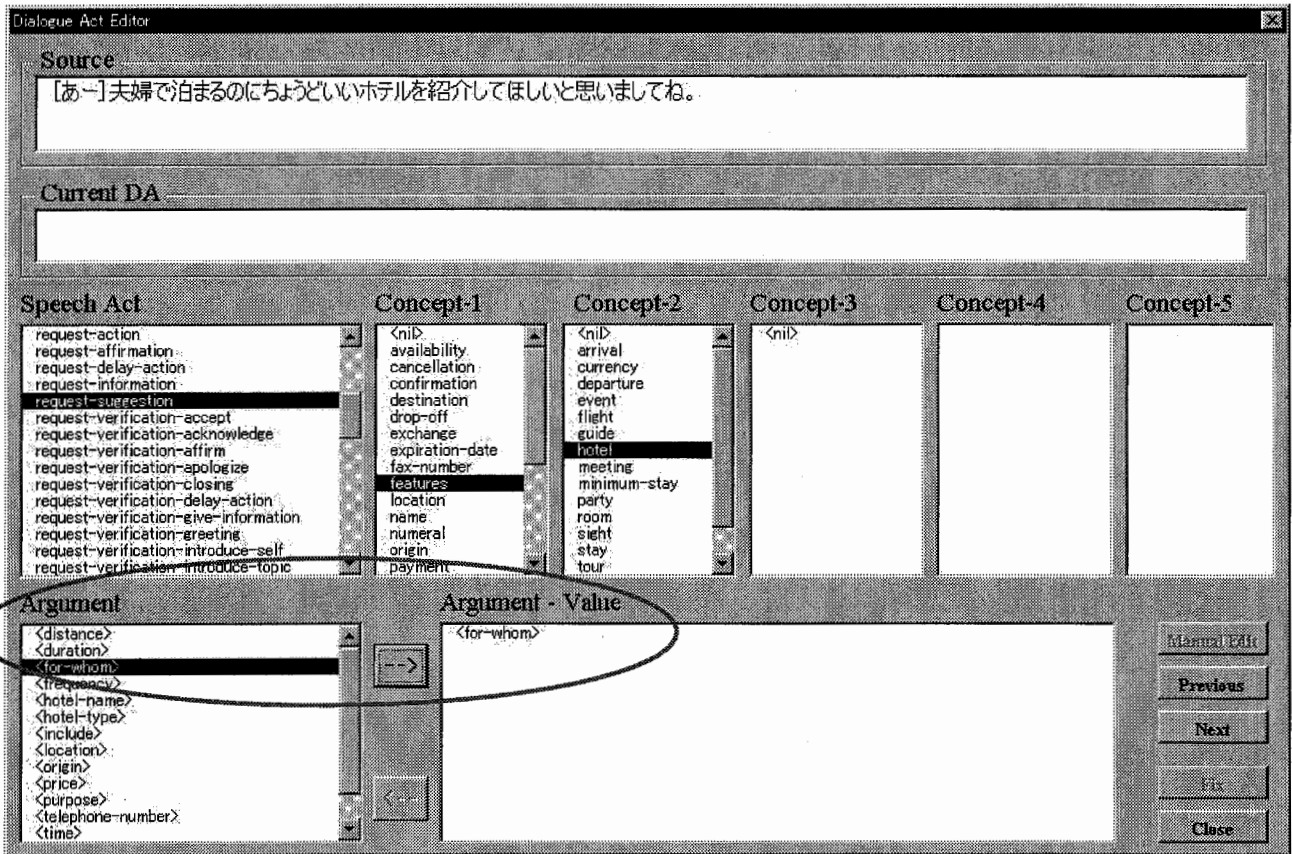


図 6 Argument の選択

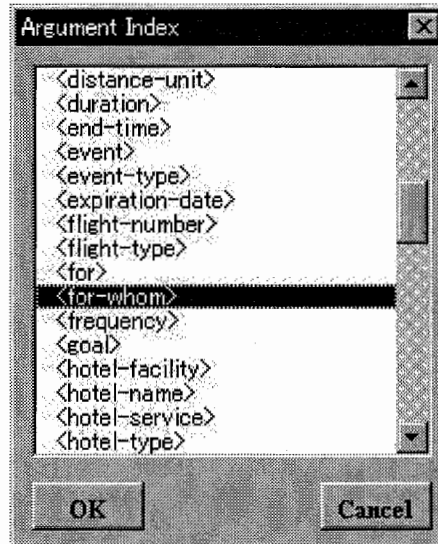


図 7 Argument Index リストボックス

2.7 Argument-Value の入力

- Dialogue Act Editor フォームの Argument - Value リストボックスで、項目をダブルクリックすると、Argument Editor ダイアログボックスが表示される。(図 8)
- Argument Editor ダイアログボックスの最上段のテキストボックスには選択された Argument が表示される。このテキストボックスは編集できない。
- Argument Editor ダイアログボックスの 2 段目のテキストボックスには、Argument Editor ダイアログボックスの最下段の定義済み Value リストで選択された Value が表示される。定義済み value リストに適切な Value がない場合、このテキストボックスに直接入力できる。「3.4 Argument への新規 Value の追加」を参照のこと。

- [OK]ボタンを押すと、Dialogue Act Editor フォームの Argument-Value リストボックスの選択項目が、選択または直接入力された Value によって式<属性名=属性値> (Argument=Value)に置き換えられる。(図 9)
- Value の入力に関しては、いくつかの特殊な入力パターンがあるので、「3 操作方法 - Argument の Value 入力時の特殊な操作」を参照のこと。

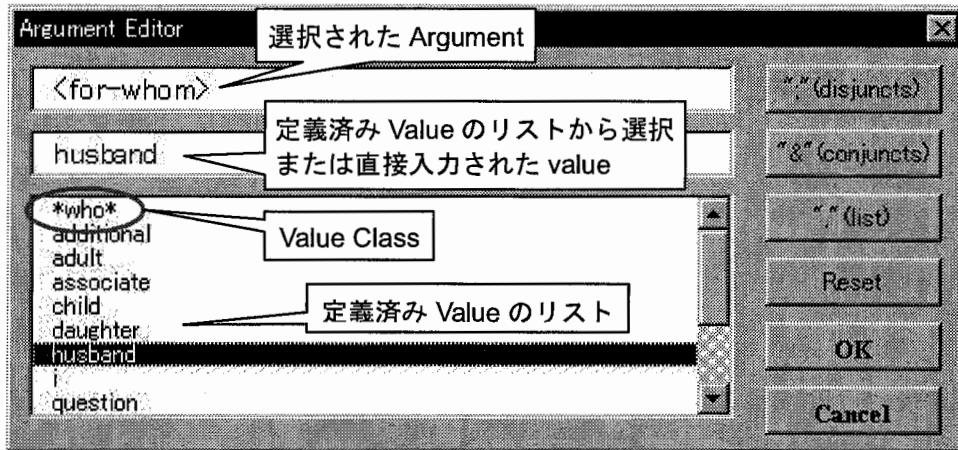


図 8 Argument Editor ダイアログボックス

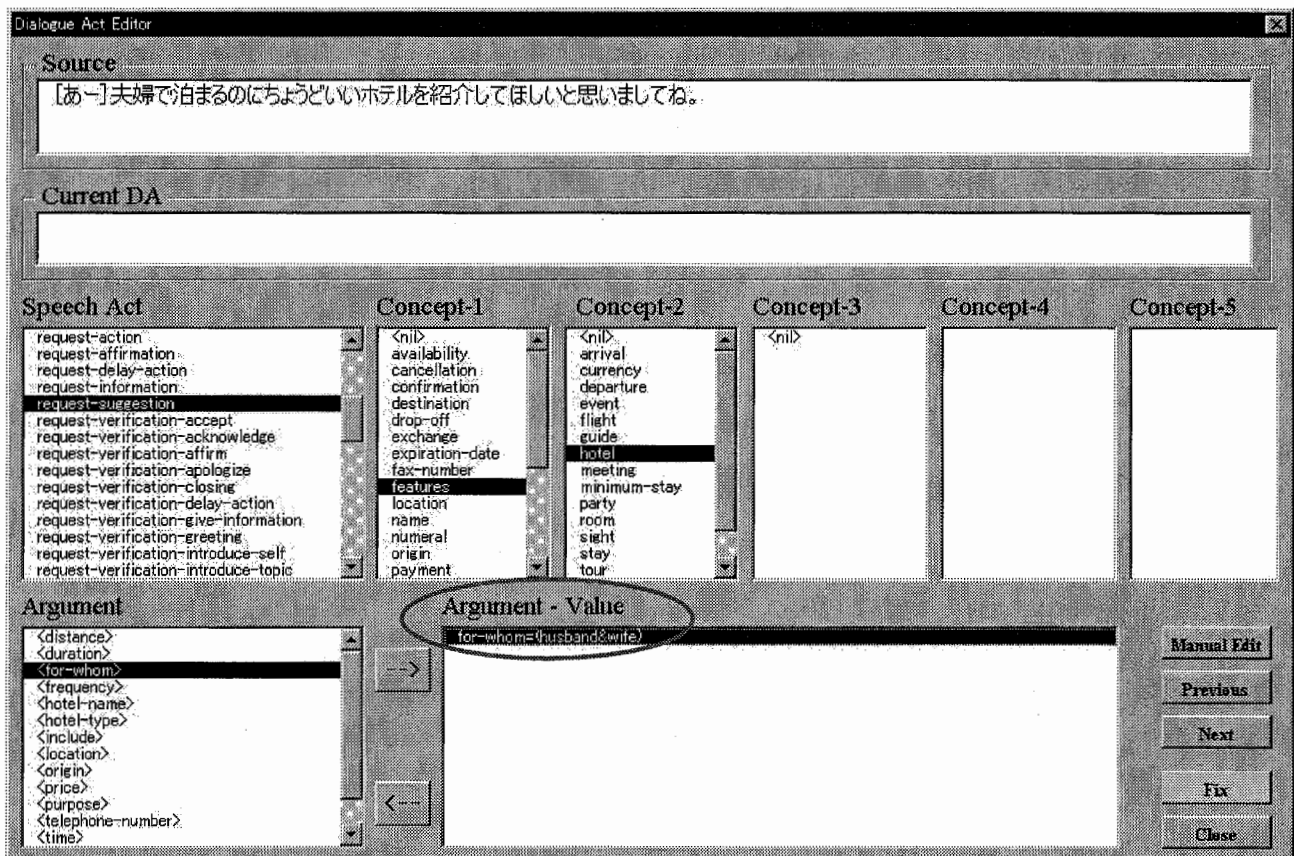


図 9 Argument-Value の入力

2.8 Speaker の選択

- タギング対象文を発話していると考えられる話者の役割 (Agent / Customer / Other) を選択する。(図 10)
- Agent は店員、ホテルフロントなどサービスを提供する、あるいは応対をする立場の話者である。
Customer は旅行客など、サービスを受ける立場の話者である。

Other は Agent / Customer 以外、あるいは話者の役割を判断できない場合に選択する。

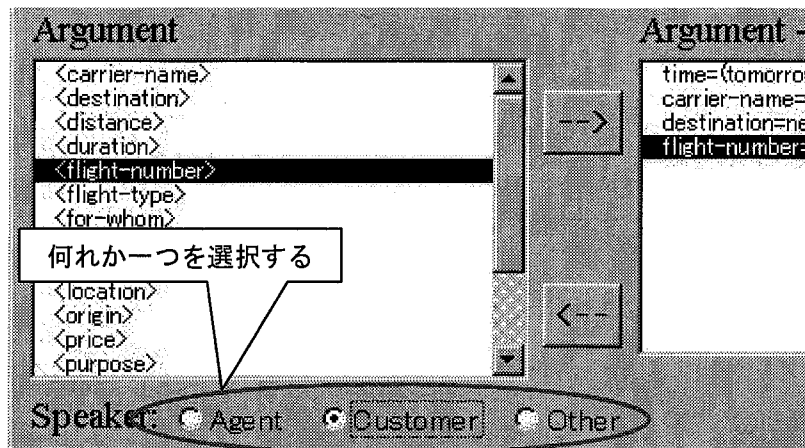


図 10 話者役割の選択

2.9 作成した DA 系列の確定

- Dialogue Act Editor フォームの[Fix]ボタンをクリックすると、編集された DA が(Source)シートに転記される。同時に Current DA ボックスに表示される。
- Speaker を選択していない状態で[Fix]ボタンをクリックすると、Speaker の選択を要求するメッセージが表示されるので、Speaker を選択してから再度[Fix]ボタンをクリックすると、編集された DA が(Source)シートに転記される。
- [Fix]ボタンをクリックせずに[Close]、[Previous]、[Next]ボタンをクリックすると、編集された DA は保存されず破棄される。

3 操作方法 — Argument の Value 入力時の特殊な操作

3.1 Value Class からの Value の選択

3.1.1 1個の Class で構成される Value

- Argument Editor ダイアログボックスの定義済み Value リストに Value Class (*に囲まれた value) がある場合、Class 毎に定義されている Value を選択できる。(例: 図 8 の *who*)
- Argument Editor ダイアログボックスの定義済み Value リストで Value Class をダブルクリックすると、選択された Value Class が Argument Editor ダイアログボックスの2段目のテキストボックスに表示されると同時に、選択された Class 内に定義されている Value のリストボックスが表示される(図 11 Value Class Ref. List)。このリスト内の適当な Value をダブルクリックすると、Argument Editor ダイアログボックスの2段目のテキストボックスに表示されている Value Class と置き換わる。
- 選択された Class 内に定義されている Value のリストボックスで、その Class と関連付けられている他の Class をダブルクリックすると、Argument Editor ダイアログボックスの2段目のテキストボックスに表示されている Value Class と置き換わりと同時に、新たに選択した Class 内に定義されている Value のリストボックスが表示される。
さらに他の Class が関連付けられている場合は、同じ操作を繰り返して処理する。
- 選択された Class 内に定義されている Value のリストボックスで、その Class と関連付けられている他の Argument をダブルクリックすると、Argument Editor ダイアログボックスの2段目のテキストボックスに表示されている Value Class と置き換わる。その後、「3.2 Argument を含む Value の選択」と同様の操作を行なう。

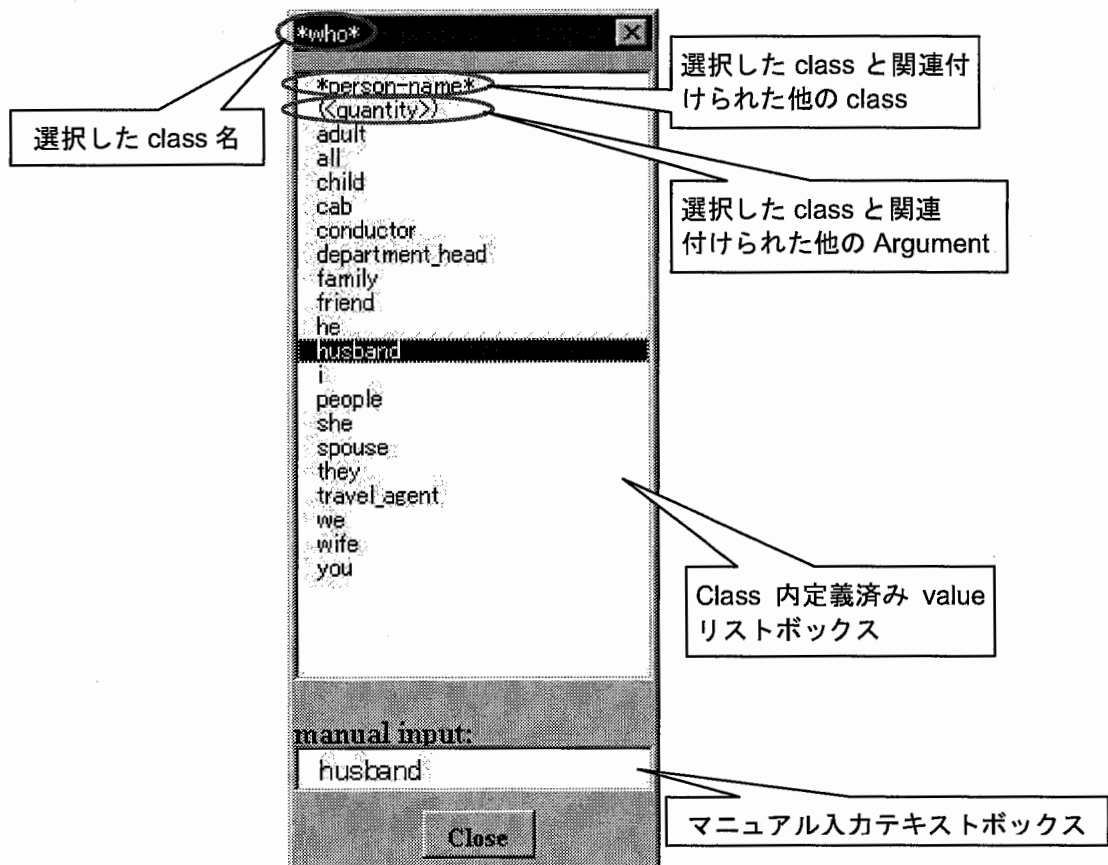


図 11 Value Class Ref. List

3.1.2 複数の Class で構成される Value

- 複数の Class で構成される Value を選択した場合、左端の class から順次「3.1.1 1個の Class で構成される Value」と同様の処理を行なう。
- 例: *first-name* *family-name* を選択した場合
→ *first-name*、*family-name* の順で Value Class Ref. List が表示される。

3.2 Argument を含む Value の選択

3.2.1 1 個の Argument を含む Value

- Argument Editor ダイアログボックスの定義済み Value リストに Argument (<>に囲まれた value) がある場合、Argument の中に他の Argument を選択できる。(例：図 8 の<quantity>)
- Argument Editor ダイアログボックスの定義済み Value リストで Argument をダブルクリックすると、選択された Argument が Argument Editor ダイアログボックスの 2 段目のテキストボックスに表示される。ここで、[OK]ボタンをクリックすると新たに選択した Argument に対する Argument Editor ダイアログボックスが表示されるので、この Argument に対する Value を選択する。(図 12)
- さらに他の Argument を選択した場合、同様の操作を繰り返して処理する。

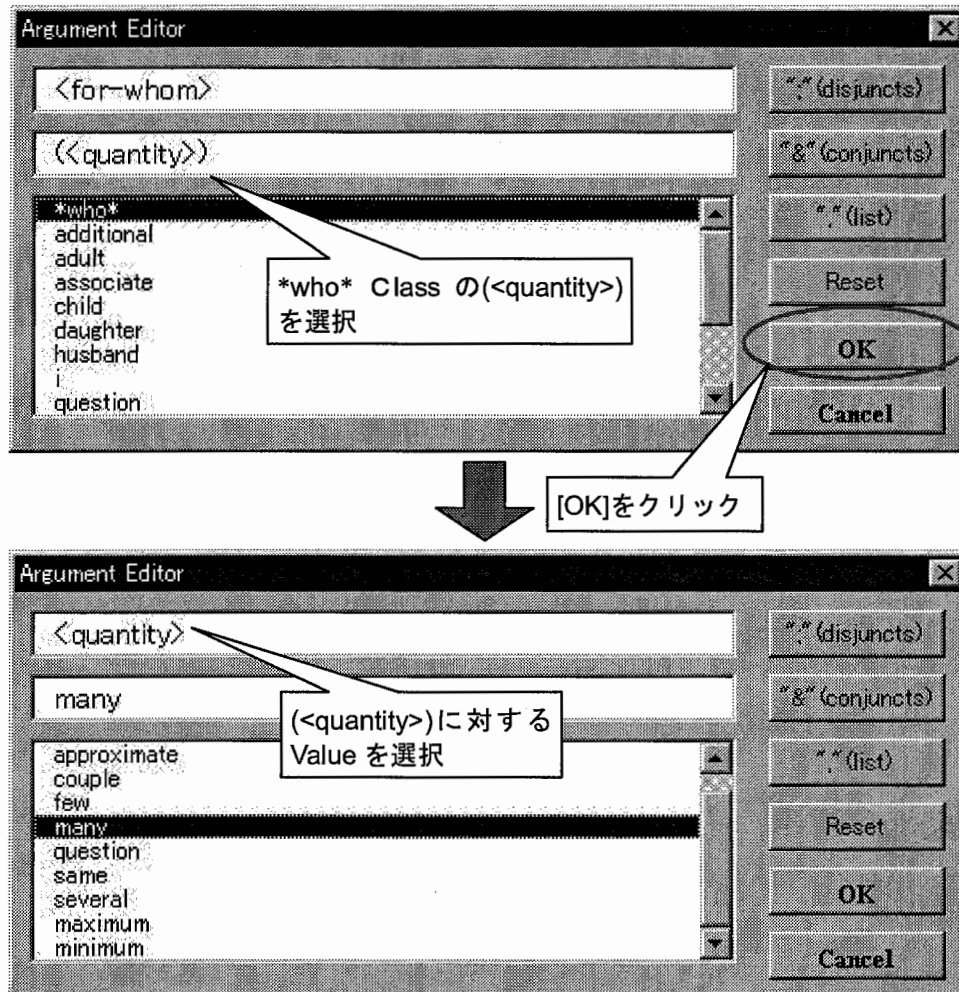


図 12 Argument を含む Value の選択

3.2.2 複数の Argument を含む Value

- 複数の Argument で構成される Value を選択した場合、左端の Value から順次「3.2.1 1 個の Argument を含む Value」と同様の処理を行なう。(図 13)

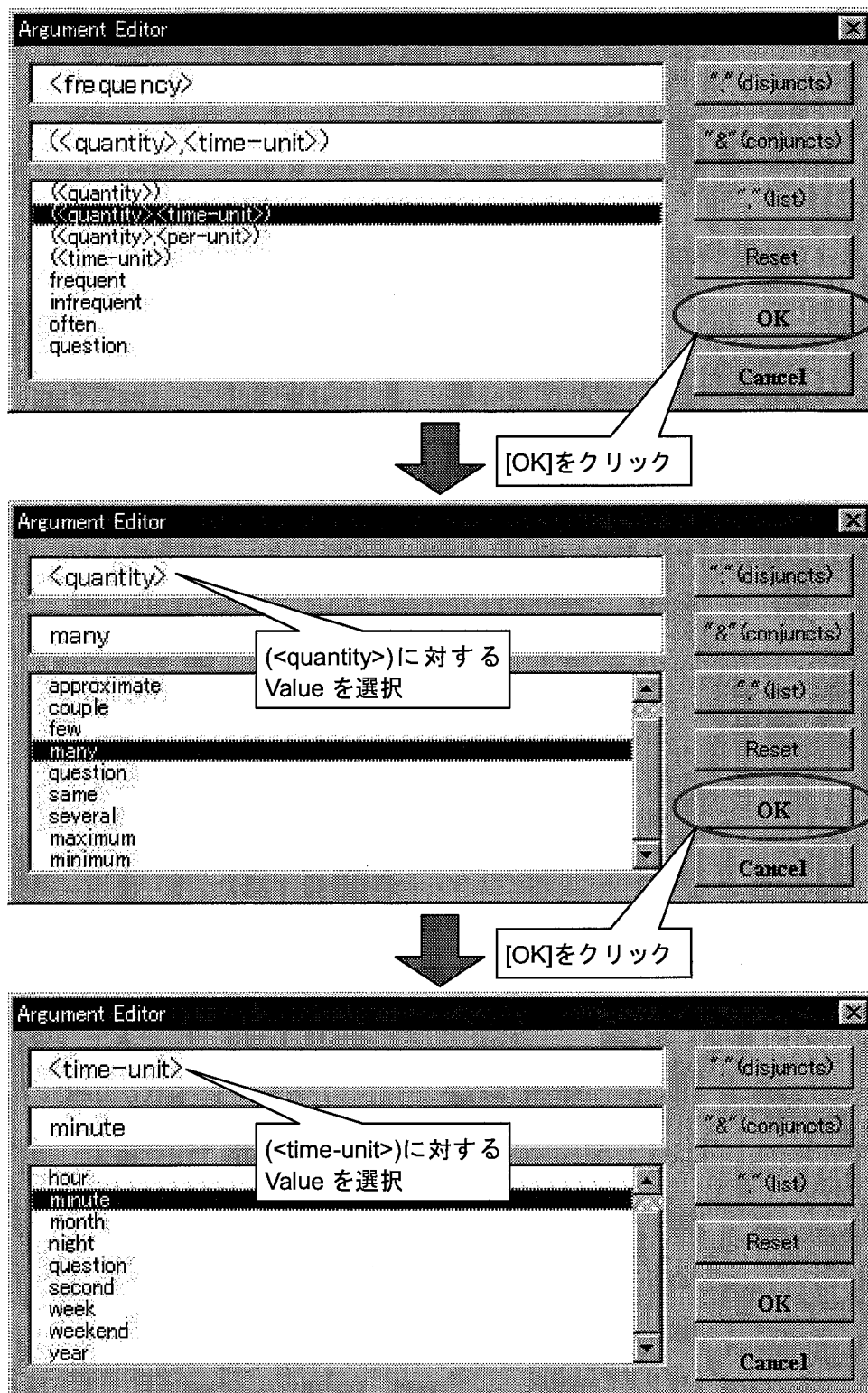


図 13 複数の Argument を含む Value の選択

3.3 Value Class への新規 Value の追加

- 図 11 のマニュアル入力テキストボックスで、定義済み Value リストボックスにない Value を入力すると、その内容は(value-class)シートの定義リストに追加・保存される。この時、追加した日時が同時に記録される。(図 14)

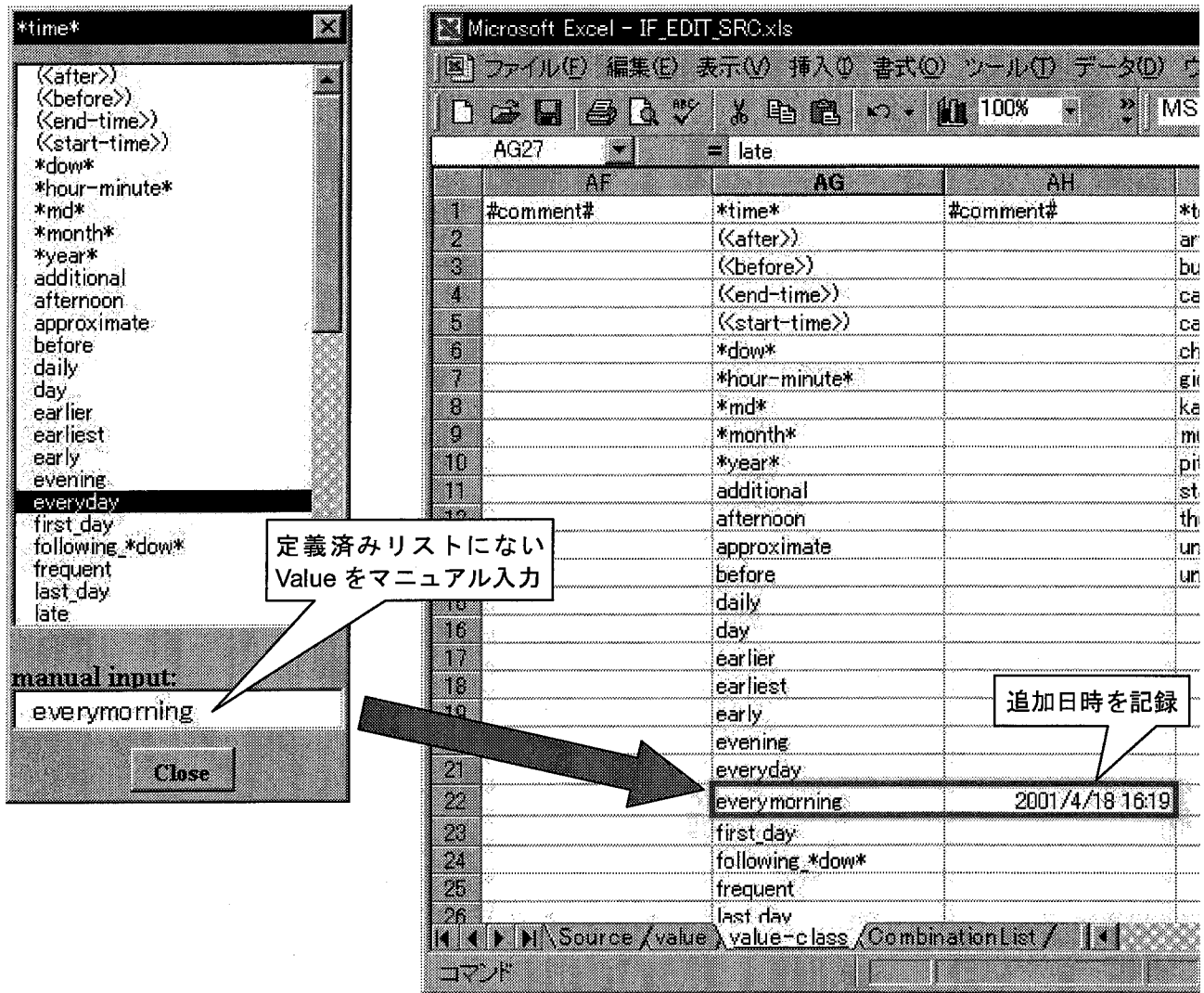
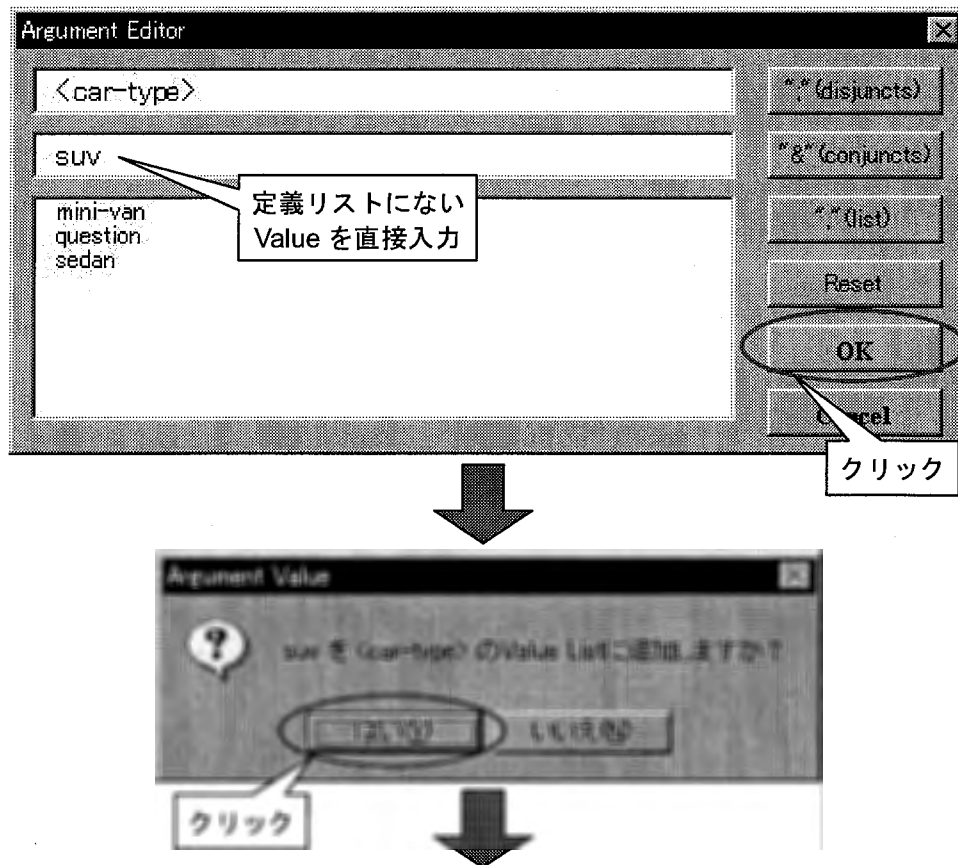


図 14 (value-class)シートへの新規 Value の追加

3.4 Argument への新規 Value の追加

- Argument Editor ダイアログボックスの 2 段目のテキストボックスで、定義済み Value リストボックスにない Value を入力し、[OK] ボタンをクリックすると、その内容を定義リストに追加するか問い合わせるメッセージボックスが表示される。ここで、[はい] をクリックすると、その内容は(value)シートの定義リストに追加・保存される。この時、追加した日時が同時に記録される。
- 同時に Dialogue Act Editor フォームの Argument - Value リストボックスの選択項目が、入力された Value によって式<属性名=属性値>(Argument=Value)に置き換えられる。



	V	W	X	Y	Z	A
1	#comment#	car-type	#comment#	carrier-name	#comment#	change-fr
2		mini-van		air_pacific		[any-arg]
3		question		all nippon airway		<<account
4		sedan		american		<<account
5		suv	2001/4/18 16:13	ana		<<address
6				asian airlines		<<admissi

図 15 (value)シートへの新規 Value の追加

3.5 Value の結合 (図 16)

- Argument Editor ダイアログボックスで、結合子ボタン (“,” / “;” / “&”) を利用すると、1 個の Argument に対して、複数の Value を設定することができる。
- 結合子の設定をしていない間は、Argument Editor ダイアログボックスで定義済み Value リストの選択項目を変更すると、2 段目のテキストボックスの内容も書き換えられる。
- 結合した Value を設定する際、[OK] ボタンをクリックすると、その内容を定義リストに追加するか問い合わせるメッセージボックスが表示されることがある。この場合は、[いいえ] をクリックする。

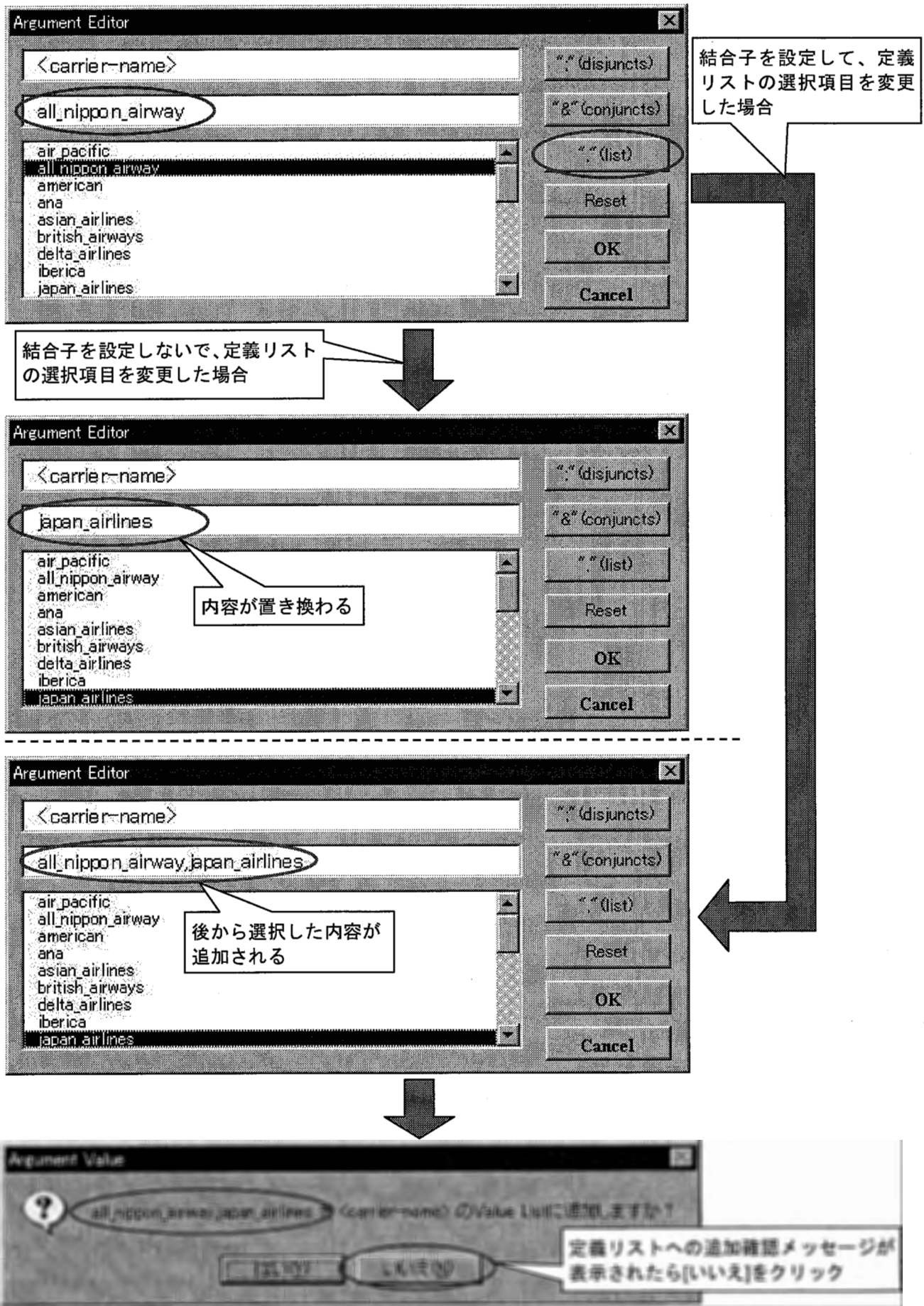
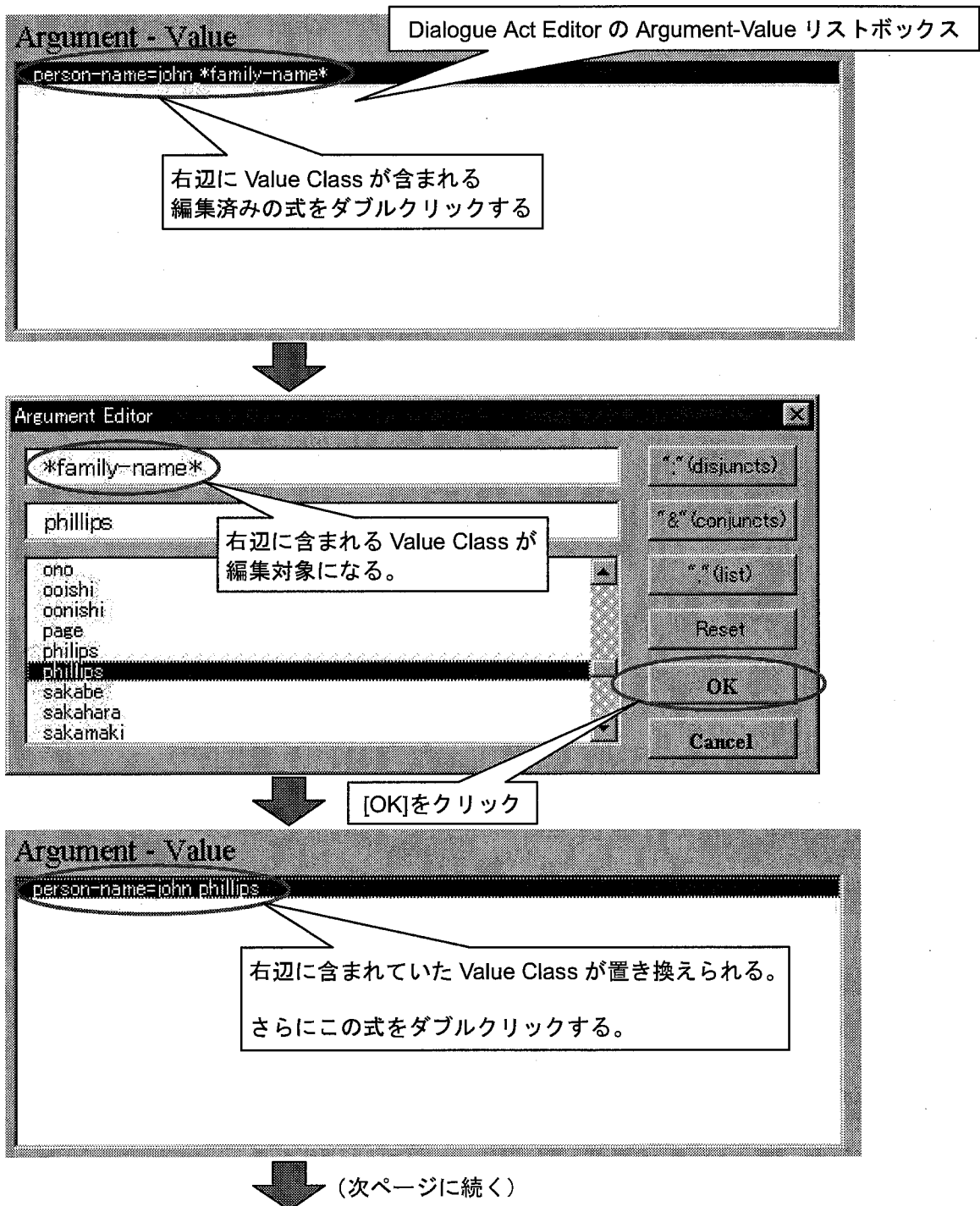


図 16 Value の結合

3.6

Argument - Value の再編集 (図 17)

- Dialogue Act Editor フォームで、編集済みの式<属性名=属性値>(Argument=Value)をダブルクリックすると、再編集することができる。
- ダブルクリックしたときの状態によって、動作が異なる。
 - (i) 右辺に Argument あるいは Value Class が含まれる場合
 - 右辺に含まれる Argument あるいは Value Class の最左にあるものを対象として Argument Editor ダイアログボックスが表示される。
 - (ii) 右辺に Argument あるいは Value Class が含まれない場合
 - 左辺の Argument を対象とした Argument Editor ダイアログボックスが表示される。
 - この時、Argument Editor ダイアログボックスの最上段のテキストボックスには、編集済みの式が表示される。
- 編集済みの式を削除するには、Dialogue Act Editor フォームの[<-]ボタンをクリックする。



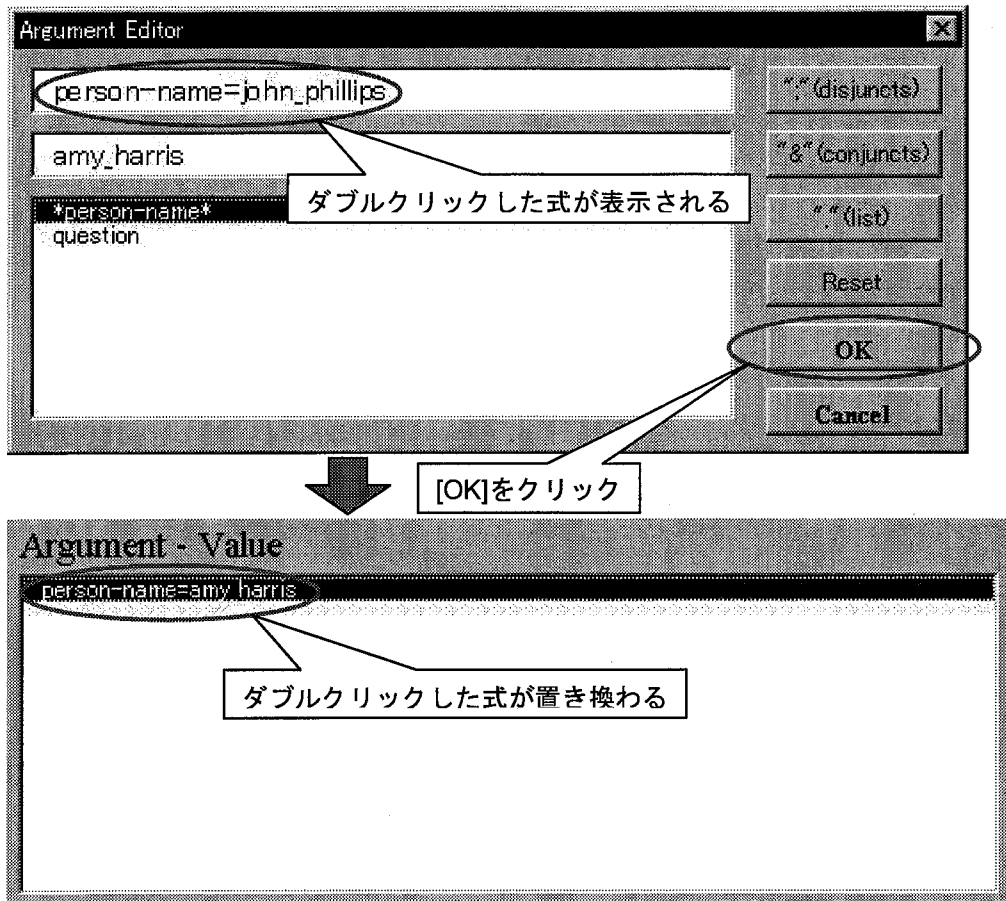


図 17 Argument - Value の再編集

3.7 数字,アルファベットのリスト入力 (図 18)

- Argument Editor ダイアログボックスで、定義済み Value リストから[n]や[a-z]などを選択すると、2段目のテキストボックスに選択した項目が表示され、[]の入力を求められるので、[]に適切な数字、文字などを入力する。“[”“]”はそのまま残す。



図 18 数字,アルファベットのリスト入力

3.8 マニュアル編集

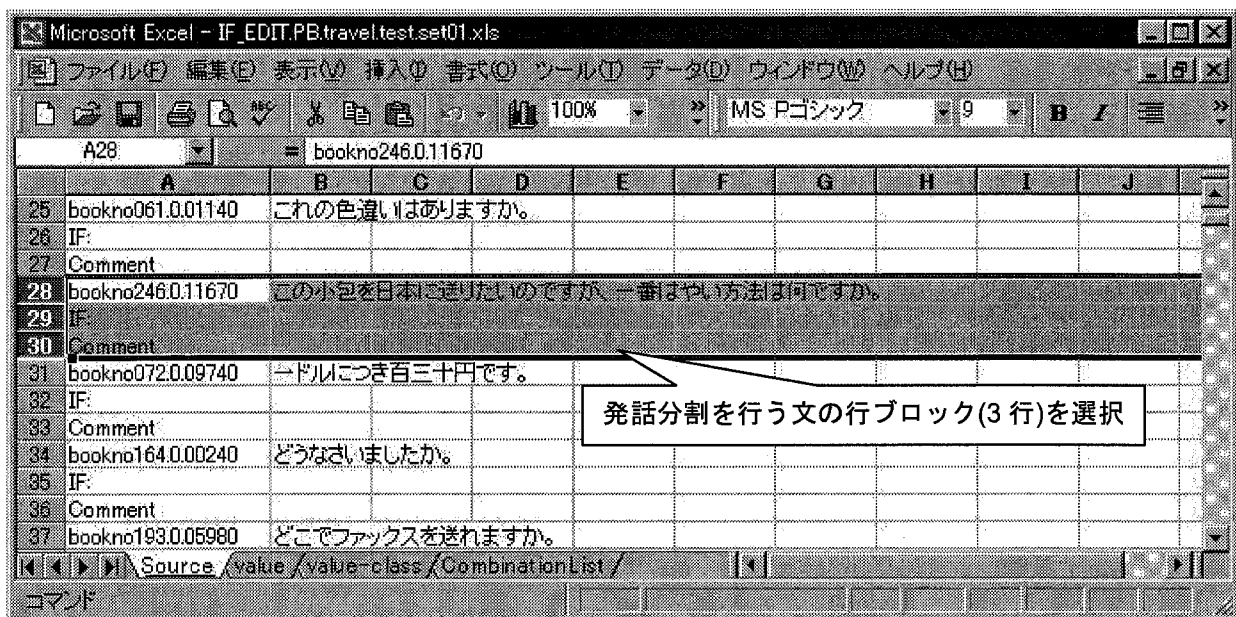
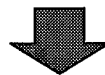
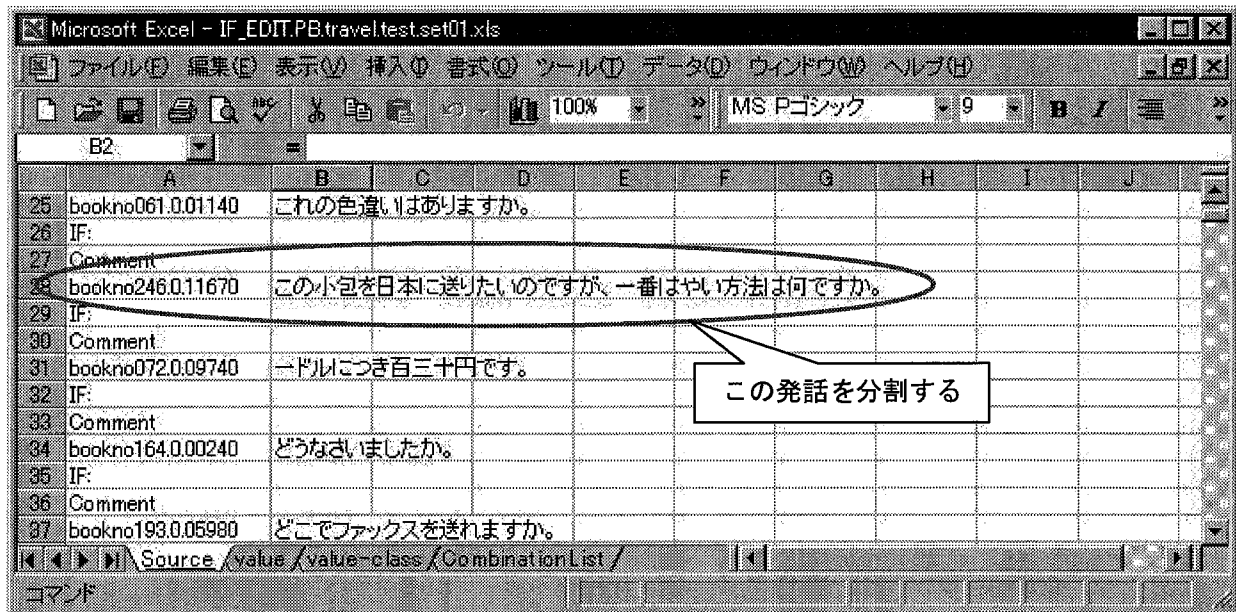
- Dialogue Act Editor フォームの [Manual Edit] ボタンをクリックすると、“Argument Editor<Manual>” ダイアログボックスが表示され、式<属性名=属性値>をキーボードから直接編集できる。ただし、仕様を逸脱しないよう、注意が必要である。



図 19 マニュアル編集

4 発話分割

- 発話分割を行う場合、作業ファイルの(Source)シート上で行う。
- 発話分割手順の例として、「bookno246.0.11670 この小包を日本に送りたいのですが、一番はやい方法は何ですか。」を「bookno246.0.11670.1 この小包を日本に送りたいのですが、」と「bookno246.0.11670.2 一番はやい方法は何ですか。」に分割する手順を図 20 に示す。



(次ページに続く)



(次ページに続く)

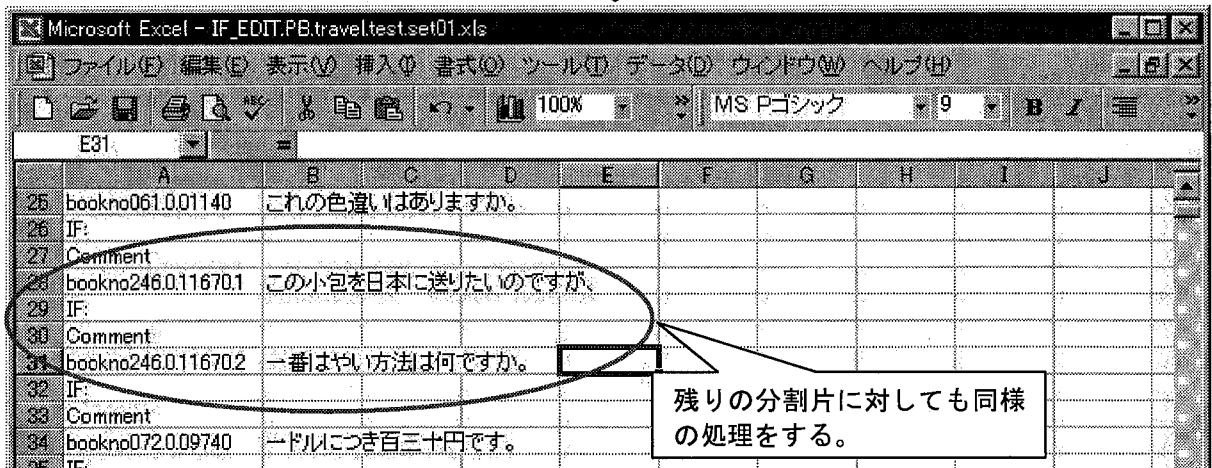
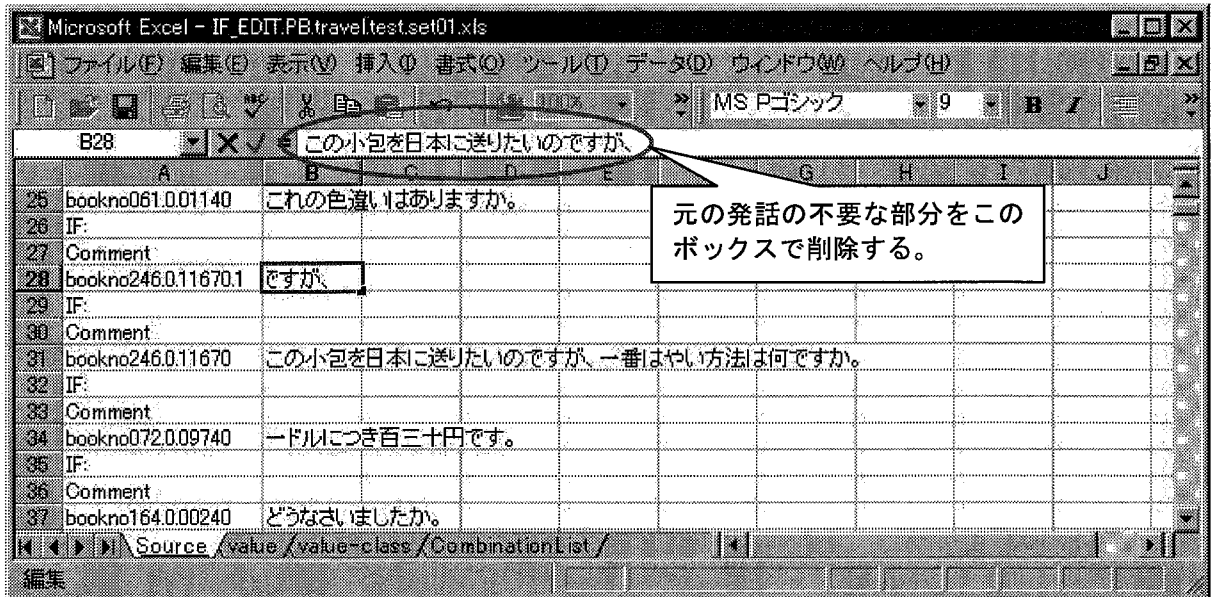
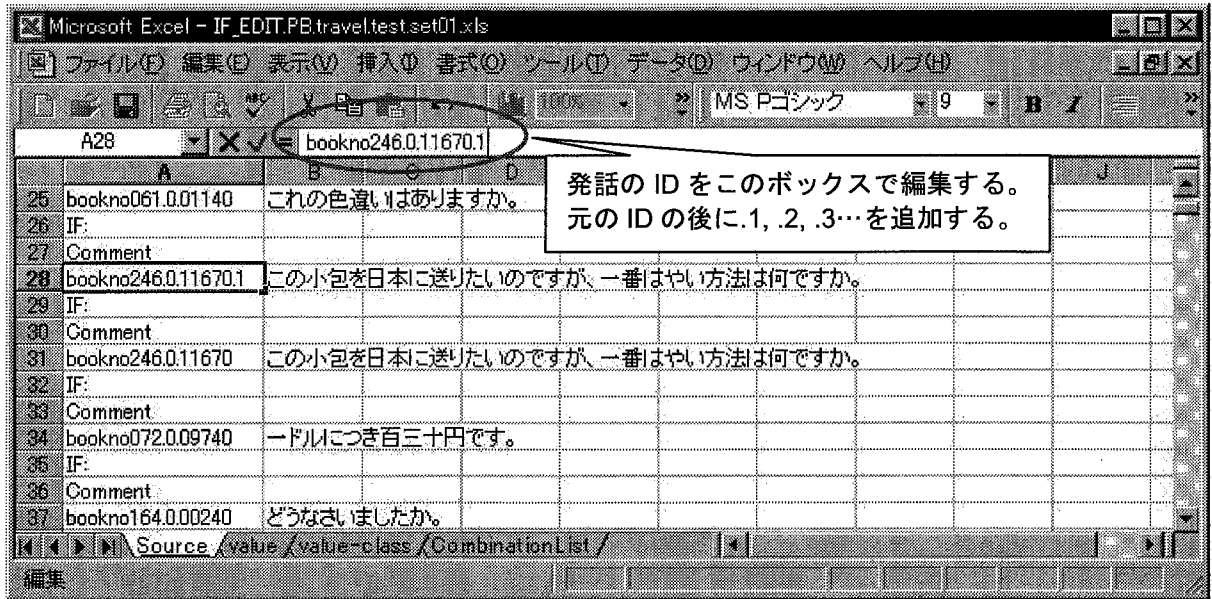


図 20 発話分割例