

〔公 開〕

TR-C-0151

S T R による仕様記述法と

仕様確認支援手法の評価

鶴亀 崇昭  
Takaaki TSURUKAME

1 9 9 6 . 3 . 1 5

A T R 通信システム研究所

## 目次

1	はじめに	2
2	STR 記述法の評価と拡張	2
2.1	STR の記述対象	2
2.2	STR の記述法とルールの適用規約	3
2.3	STR の記述性の評価	4
2.3.1	STR 記述仕様の制約	4
2.3.2	制約に対する仕様記述の可否検討	4
2.3.3	記述仕様の改善案	6
2.4	未検討サービスに関する拡張検討	7
2.5	考察	12
3	仕様確認支援手法の評価と拡張	13
3.1	手法の概要	13
3.2	手法の確認可能範囲	14
3.3	手法の拡張検討	15
3.4	考察	16
4	おわりに	17
4.1	STR 記述法について	17
4.2	仕様確認支援手法について	18

## 1 はじめに

今日通信サービスの分野では、通信サービスの基盤整備が急速に進んできている。このような状況下、ユーザのニーズを反映した多種多様なサービスの開発が急務である。サービスの開発では完全な仕様を生成するため仕様記述者が専門家であることが望ましいが、サービスの早期確定のためには非専門家の参画が必要と考えられる。現在まで非専門家による仕様記述から通信ソフトウェアの自動生成を行う手法が提案されている。 [1] [2] [3]

本レポートではそのうち、仕様記述法の STR(State Transition Rule) 手法と仕様の確認支援手法である IPS(InterPreter System) について、評価及び拡張の検討を述べる。

以降、第2章では、STR手法の記述性の評価と拡張検討について、また、第3章では、IPSの確認機能の評価と拡張検討について、第4章では、まとめを述べる。

## 2 STR 記述法の評価と拡張

STRの記述仕様は、文献[4]にまとめられている。本稿のSTRの記述性の評価はSTRの記述法、及びルールの適用について行う。以降、2.1節で記述対象を、2.2節でSTRの記述法とルールの適用規約を述べる。

### 2.1 STRの記述対象

[STRの記述対象]

STRは、通信システム(ネットワーク)を介して実現されるサービスのうち、システムをブラックボックスとして、対象物間の接続に関するサービスの一連の振舞いを記述対象としている。

ここで「対象物」とは、通信サービスをユーザが利用する時の端点となるもので、一般的には端末を指す。サービスの「振舞い」とは、対象物からのイベント入力と対象物の状態を要因として、一意に状態が遷移する動作を意味する。これらは、利用者の観点から認識可能な事象である。

従って、次のサービスは記述対象外となる。

- 対象物間の接続以外の振舞いを対象とするサービス

対象物の振舞いが、システム(ネットワーク)が管理する接続以外のものである場合、記述対象としない。

## 2.2 STR の記述法とルールの適用規約

### [STR の記述法]

STR は、サービス仕様を以下の構文を持つプロダクションルールの集合として記述する。

現状態 イベント: 次状態

ルールは一つのイベントのみ含み、イベントによって影響を受ける対象物の状態を示す。現状態と次状態は対象物の状態を示す状態記述の集合から成る。ルールはイベント記述と状態記述の記述要素から形式的に表される。

イベントの記述は、以下のイベント名と識別子によって行う。

イベント記述 = イベント名 (識別子 [, 識別子])

イベント名は、対象物の状態を遷移させる動作名を示し、引数である識別子は、実状態中の対象物に一対一対応するものである。対象物の単独の動作において識別子は一つ、他の対象物との関係を表す動作では相手の識別子と共に二つの識別子を引数としてとる。

状態記述は、以下の状態識別子と対象物を表す識別子によって行う。

状態記述 = 状態識別子 (識別子 [, 識別子])

状態識別子は、遷移する対象物の状態名である。状態識別子はイベント名と同様に、対象物単独の状態において自身の識別子を一つ、他の対象物との関係を表す場合に二つの識別子を引数としてとる。

イベント名や状態識別子は、サービスの手続きや機能を内包する任意の意味を持つことが可能である。但し、それらは、対象物間の接続に関するもので、且つユーザが認識可能な事象に限られる。

### [ルールの適用規約]

ルールの適用は、実状態を  $G_c$ 、ルールの現状態を  $S_c$ 、実状態中に生起するイベントを  $G_e$ 、ルールのイベントを  $E$  とすると、以下の条件を満たした場合に可能と判定される。

$$(G_c \supseteq S_c) \wedge (G_e = E)$$

適用ルールが複数ある場合は、ルールの組合せを考え、現状態が包含関係にあるかどうかを判断して、包含される方を適用候補から除外する。ルールの適用によって、実状態のうちルールの現状態に一致する状態を、ルールの次状態に置き換える。この時、ルール中の識別子は実状態中の対象物に一対一に対応する。

## 2.3 STR の記述性の評価

文献 [5] では、PBX の電話サービスを対象として記述実験を行っており、STR 記述法の網羅性と有効性が確認されている。また、STR のルールにおける状態、イベントの意味は前節で述べたように、任意のものをとることが可能であり、記述対象内においてルールのとる意味的限界は示されない。そこで本稿では、STR 記述仕様の制約から仕様不可となる領域を洗い出し、改善可能であるかどうかを明確にすることで記述性の評価に代える。

まず、前節で述べた STR の記述仕様 (記述法とルールの適用規約) から制約を抽出し、影響が考えられるサービスについて仕様記述の可否を検討する。

### 2.3.1 STR 記述仕様の制約

以下に STR 記述仕様の制約を挙げる。

#### 1. 記述要素のとる識別子数の制約

STR 記述法では、イベント記述、状態記述のとる識別子の数を一つもしくは二つに限っている。

#### 2. ルール中のイベント数の制約

STR 記述法では、ルール中に記述するイベントの数は一つとしている。

#### 3. ルール中の識別子と実対象物の対応

STR ルールの適用規約は、ルール中の識別子が実状態中の対象物に一対一対応することとしている。これは、記述要素のとる識別子の意味からも導かれる制約である。

### 2.3.2 制約に対する仕様記述の可否検討

検討は以下の点について考慮している。

◇ 通信システムで実際に有り得るサービスかどうか。

実際の通信システムで存在しないサービスは、記述対象外であり、STRの記述可否をはかる必要はない。

◇ 記述者（ユーザ）の認識するサービスの手順を記述できるかどうか。

記述者が認識している動作の手順や振舞いを、正しく反映出来るかどうかによって記述の可否を判断する。

次に制約と、制約から影響が考えられるサービス記述の可否を述べる。

### 1. 記述要素のとり識別子数の制約

- 三つ以上の対象物間の関係を同時にもつサービス

状態記述の識別子の個数の制約により、一つの状態記述で三つ以上の対象物の関係を同時に示すことは出来ない。しかし、状態記述は任意個の記述がルール中で許されており、その組合せで三つ以上の関係を同時に表すことが出来るので、仕様記述の問題にはならない。

- 三つ以上の対象物を持つイベントが生起するサービス

例えば、呼び出したい二つの端末の番号を並べてダイヤルすることで、二端末を一齐に呼び出すサービスは、識別子の個数の制約から、現状のSTRでは記述できない。この場合、イベントを分けてルール記述する対応としても、記述者の認識する手順を正しく記せたとはいえない。このため記述法の制約改善が必要と考えられる。

### 2. ルール中のイベント数の制約

- 複数のイベントの並列処理を行うサービス

STRの記述法では、ルール中のイベントは一つとして限っている。但し、記述対象とする通信システムにおいて、イベントの実行は排他的に行うものであり、複数イベントを一ルール中に記述する必要はない。

### 3. ルール中の識別子と実対象物の対応

- 不特定多数の対象物に対するサービス

記述要素のとり識別子は、ルール適用時に実状態に存在する対象物に一対一に対応する。不特定多数の対象物に対応する適用は考えられていない。

例えば、ある時点において空きの状態を持つ対象物全てに、呼び出しをかけるといったサービスが考えられる。実状態中の対象物数が定まっている場合には、対象物数のパターンだけルールで記述をすることで対応できるが、動的に対象物の数が増加する場合には記述できない。従って、ルール適用による制約の改善が必要と考えられる。

以上、現在の STR 記述仕様の制約から、下記のサービスに関する仕様記述が不可となることが判明した。

- 三つ以上の対象物を持つイベントが生起するサービス
- 不特定多数の対象物に対するサービス

### 2.3.3 記述仕様の改善案

仕様記述の不可を招く制約に関して、改善検討を述べる。

#### \* イベント識別子数の制約改善

識別子の個数制約を設けないイベントの記述を導入する。主体となる対象物を第一識別子で表し、関係する対象物を二番目以降の識別子とすると、前の二つの端末を同時に呼び出す例は以下の様に記述される。

(例) 二端末同時ダイヤル可能 (A), 空き (B), 空き (C)

二端末同時ダイヤル (A,B,C):

呼び戻し (A,B), 呼び出し (B,A), 呼び戻し (A,C), 呼び出し (C,A).

イベントの識別子数を増加する案は、ルールの適用上問題がないため、記述法の改善が可能である。

#### \* 識別子と実対象物の対応関係の制約改善

任意個の数の対象物に対応する変数を状態記述の引数として導入する。併せて、この変数でルールの現状態に記述される対象物が実状態に存在する場合、全てルールの適用が可能となる様に、ルールの適用規約を拡張する。以下に例として、先の一斉呼び出しサービスの変数によるルール記述とルール適用前後の実状態を示す。

(例) 不特定多数の対象物に対応する変数を S とした場合

ルール) ダイヤル可能 (A), 一斉呼び出し可能 (A),

空き (S), 一斉呼び出し加入登録 (S)

一斉呼び出し (A) :

呼び戻し (A,S), 呼び出し (S,A), 一斉呼び出し加入登録 (S).

ルール適用前の実状態

= {ダイヤル可能 (T0), 一斉呼び出し可能 (T0),

空き (T1), 一斉呼び出し加入登録 (T1),

空き (T2), 一斉呼び出し加入登録 (T2),

空き (T3)}

ルール適用後の実状態

= {呼び戻し (T0,T1), 呼び戻し (T0,T2),

呼び出し (T1,T0), 一斉呼び出し加入登録 (T1),

呼び出し (T2,T0), 一斉呼び出し加入登録 (T2),

空き (T3)}

イベント識別子数とルール適用規約の制約について上記の改善案で対応が可能であることを確認した。但し、これらの改善案については、記述法とルール適用についてだけでなく、サービス競合の仕様検証支援をはじめとする様々な支援の適用も考えてさらに検討を進める必要がある。

## 2.4 未検討サービスに関する拡張検討

前節まで、既存の STR の制約について改善案を述べた。本節では、これまでに記述が未検討のサービスに対して、STR 記述仕様の拡張検討を行う。拡張対象として以下のサービスについて述べる。

### (1) 全ての対象物に共通する要素を含むサービス

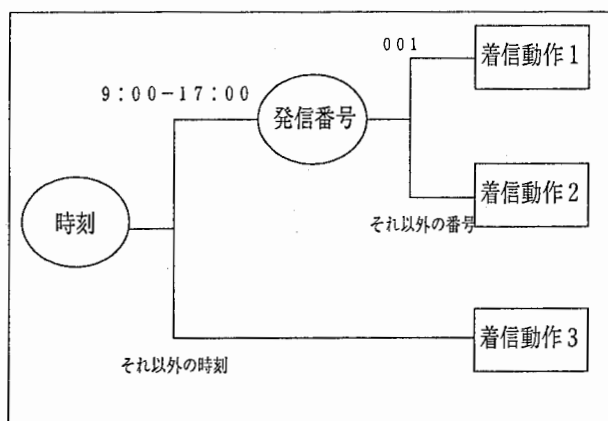
ここで要素とは、対象物の遷移を起こす要因である。要素が全ての対象物に対して共通となる例として、絶対時間 (時刻) や時間帯がある。既存の STR 記述仕



様では対象物の状態に依存した相対時間を示す記述が用意されているが、共通の時刻である絶対時間の記述について特に考慮されていなかった。

## (2) ユーザに依存する条件を含むサービス

ユーザに依存する条件を含むサービスの例として、ユーザの設定する条件を、実条件値と比較判定して、サービスの動作を選択実行するスクリーニングサービスがある。実際のサービス例を図1に示す。



○ 条件の種類

□ 選択される動作

図1: スクリーニングサービス例

このサービスは着信時に時刻と発信番号による条件判定を行い、その結果を反映して着信動作1～3のうちの一つを選択・実行することを示している。この様にユーザに依存する条件とは、ユーザ毎に設定される条件であり、ユーザに共通するサービスの振舞いの記述とどの様に併せて記述するかは未検討であった。

以下に、(1)のサービスに対して、既存の記述法による記述案と新たな記述の拡張案の二案について比較評価を行う。

### (1) 全ての対象物に共通する要素を含むサービスの記述案

#### 1. 対象物に依存した共通要素の記述を行う(案1).

共通要素を従来の対象物を引数にとる記述要素で表す。例えば17:30の時刻を、対象物Aが時刻17:30を持つと言う意味のイベント、「時刻17:30を持

つ (A)」で記述する。時間帯の様に要素の範囲がある場合には、イベントによって、それぞれ始点と終点を示す状態記述を付与することで対応可能である。以下に時刻 17:30 以降にオフフックすると、登録した相手に自動的にダイヤルするサービスについてルールの記述例を示す。

(案 1 によるルール記述例)

空き (A) 時刻 17:30 を持つ (A): 空き (A), 自動ダイヤル可能 (A).

空き (A), 自動ダイヤル可能 (A), 自動ダイヤル登録 (A,B), 空き (B)

オフフック (A):

呼び戻し (A,B), 呼び出し (B,A), 自動ダイヤル可能 (A),

自動ダイヤル登録 (A,B).

## 2. 対象物に依存しない状態として記述を行う (案 2).

共通要素を対象物に依存しない状態として、ルールの現状態に記述する。既存の状態記述の様に識別子は取らない。例えば、前例の時刻 17:30 はそのまま、「時刻 17:30」として記述する。時間帯の場合には、例えば「時間帯 17:30-18:30」となる。ルールの記述例を以下に示す。

(案 2 によるルール記述例)

空き (A), 自動ダイヤル登録 (A,B), 空き (B), 時間帯 17:30-18:30

オフフック (A):

呼び戻し (A,B), 呼び出し (B,A), 自動ダイヤル登録 (A,B).

この場合、時間帯 17:30-18:30 は、時間帯を管理する機能によって、実状態に自動的に付与/削除する必要がある。

表 1: 共通要素の記述法案の比較

	案 1	案 2
ルール適用規約拡張の必要性	無し	有り
仕様記述量 (相対比較)	多い	少ない
新規種類の要素の記述	随時可能	可能

表 1 に案 1, 2 の比較評価を示す。

- ルール適用規約拡張の必要性

案2は、ルール中の現状態に含まれる共通要素とそれ以外の状態とを分けてユニフィケーションする様に、ルールの適用規約を拡張する必要がある。拡張については特に問題はない。

- 仕様記述量

案1ではイベントによって付与される状態記述の意味的矛盾を考慮して、状態の禁止を組合せ的に記述する必要がある。案2では、実状態中の要素の管理を種類毎の管理機能に任せるため、禁止記述を行う必要がない。

- 新規種類の要素の記述

案1は、従来の任意の意味を持つ記述が可能であり、新規種類の要素の記述も随時可能である。案2では、新しい共通要素の種類毎に、実状態における管理機能を用意することで記述が可能となる。

二つの案は記述者の立場から見ると、記述量の面で案2が適している。また共通要素の意味をより正しく反映していると言える。但し案2は、対象物に依存する状態とは別に、新たに実状態の共通要素を管理する機能を順次追加しなければならない。二つの案の評価は、この管理関数の付与の仕方などシステムの実現も含めてさらに行う必要がある。

## (2) ユーザに依存する条件を含むサービスの記述案

以下に、変数を記述に導入することで、ユーザ依存の条件とユーザ非依存のサービスとの記述分けを行う記述法を提案する。

### 条件の判定を変数によって状態記述する案

変数を引数とする条件判定の記述をルールの現状態に行う。条件は変数によって示され、ルール中に明示的に記述しない。実際の条件はデータ仕様としてルールとは別に記述する。以下に、条件判定の形式的記述と条件の記述について示す。

#### ○ 条件判定記述

実状態中の条件と実条件値との合致判定を、状態の一つとして記述する。条件判定の記述は、条件判定に一意に対応する変数と、その判定毎に設定された各々の条件を一意に識別する分岐番号の二つを引数にとる以下の形式で表す。

図 2: 条件判定記述による拡張ルール

Sp, equal[X,1], equal[Y,1] E: Sn1.  
 Sp, equal[X,1], equal[Y,2] E: Sn2.  
 Sp, equal[X,2] E: Sn3.

E: ルールのイベント

Sp: 対象物の状態集合

Snj: 対象物の次状態集合

表 2: 条件記述例

変数	条件の種類	条件	分岐番号
X	時間	9:00-17:00	1
		それ以外	2
Y	発信番号	001	1
		それ以外	2

条件判定記述 = equal[変数, 分岐番号]

図 1 のスクリーニングサービスにおける条件の判定とそれに応じた着信動作は図 2 の条件判定記述を含む拡張ルールで記述される。条件判定前の対象物の状態 Sp と条件判定の和がルールの現状態となる。

○ 条件記述

条件はルール記述とは別に記述する。記述は、条件判定記述の引数である変数と分岐番号に、条件の種類と条件のデータ、及びそれらの対応関係を記す。条件記述の例を表 2 に示す。

以上述べた条件の記述案以外に、共通要素を含むサービス記述の案 2 の引数を持たない状態の記述案が条件の記述に適用できる。そこで、先に述べた変数を使用して記述する案を I, この引数を持たない状態の記述案を案 II として、二案の比較を表 3 に示す。

● 記述の一貫性

案 I では、記述者が別途条件データの仕様記述を行う必要がある。案 II は、ルールの仕様のみ記述すればよく、記述の一貫性が保てる。

● 仕様記述項目数

表 3: ユーザ依存条件の記述法案の比較

	案 I	案 II(案 2)
記述の一意性	条件のデータ記述が必要	ルール記述のみ
仕様記述項目数	多い	少ない
仕様の再利用性	高い	低い

案 I では、変数と分岐番号、条件、条件の種類について記述する必要がある。

案 II では、条件をそのまま状態として明示的に書くだけでよい。

- 仕様の再利用性

案 I では条件データと切り離したことで、ルール仕様記述の再利用が可能となる。案 II は、ユーザの設定毎に仕様が異なるため、再利用の可能性が低い。

以上、ユーザ依存の条件を変数を使ってルール中に記述し、ユーザ毎の条件データはルール記述と切り離して記述する案 I と条件を明示的に記述する案 II との比較を述べた。案 I は、ユーザ毎の条件記述とユーザに共通なルール記述を分けるので、ルール仕様の増大を防ぐことが出来る。一方、記述を分けることによって、一意性や分かり易さが損なわれるおそれもある。条件をルール中にどこまで記述するかについては、記述のし易さも考慮して、検討を進める必要がある。

## 2.5 考察

本章では既存の STR の記述性を、記述仕様の制約によって仕様記述が不可となるサービスにより明らかにした。また、これらサービスの記述は、下記の改善を行うことで可能になることを確認した。

- 三つ以上の対象物を持つイベントが生起するサービス

識別子の個数制限を無くすイベント記述の導入を行う。

- 不特定多数の対象物に対するサービス

実状態中の対象物全てに対応する変数の導入と、それに合わせたルール適用規則の拡張を行う。

続いて、これまで未検討であった二つのサービスについて、以下の記述法の拡張によって記述が可能になることを確認した。

(1) 全ての対象物に共通する要素を含むサービス記述拡張法

引数を持たない条件をルールの状態に記述する。

(2) ユーザに依存する条件を含むサービス記述拡張法

条件を変数によって記述する条件判定記述をルールの状態に記述する。条件のデータ仕様は、ルールの仕様記述とは別に記述して持つ。

(1),(2)の両案に対する支援手法の拡張を検討してそれらの有効性を見極めることは今後の課題である。例えば、仕様の競合検証を考えた場合、共通要素やユーザ依存の条件を現状態に含むルール間で非決定性が冗長に検出されてしまう問題がある。これを防ぐために、今後共通要素及び条件の意味を考えた非決定性の検出支援を検討していく必要がある。

### 3 仕様確認支援手法の評価と拡張

#### 3.1 手法の概要

前章STRにより記述された仕様の確認支援はIPSとしてその手法が提案され、システムが試作されている。手法は、ユーザからの入力によって適用されるルールの判定を行い、ルールによる状態遷移をサービス対象物の状態図の書き換えによって視覚化する。その処理手順の概要を図3に記す。

IPSは、ソフトウェア自動生成システム内において、次のような役割を担っている。

##### □ 仕様の記述支援

システムの上流工程の段階で、仕様要求と記述仕様の合致をみる手段、一部の記述間違いや抜け、仕様の合成時における競合（非決定性）を確認する手段を提供する。これによって、仕様の訂正による手戻りを防ぐことが出来る。

これによって、仕様記述者もしくは要求サービスの内容を理解している者が、仕様に対して意図通りの仕様であるかどうかを見極めることが可能となる。以下にIPSの特徴を記す。

- サービスの実状態表現。

サービス対象物の実状態を図（アイコン）で表示する。これによって、ユーザは対象物の静的な実状態の確認が行える。<sup>1</sup> 図4に、IPSの試作システムの表示画

<sup>1</sup>仕様の各ルール毎の現状態及びイベントによる次状態遷移の確認は、仕様記述支援手法(RAS / GR)[6]で行う。

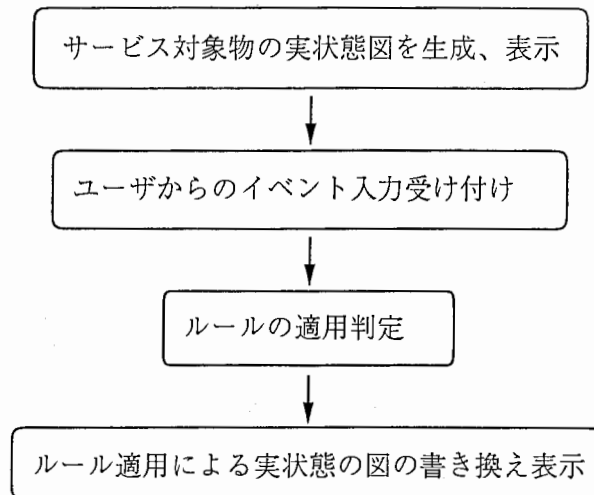


図 3: IPS の処理手順概要

面例を示す。画面では、三つの端末について各々の状態と関係 (path,path,dial-tone) に対応する図が示されている。

- 仕様の動的表現.

実状態において、イベントが発生した時のサービス仕様の振舞いを図の変化によって表現する。ここで、振舞いとは対象物の状態の遷移である。これによってユーザは仕様の動的な確認が可能である。

- ユーザとの対話による仕様確認.

IPS は、ユーザからのイベント入力によって仕様の実行を行う。

- ルールの自動適用判定と非決定性検出

IPS は、ルールを逐次解釈し、実状態における適用性を自動判定する。複数のルールが適用可能な場合、それらをサービス非決定性の候補として表示する。

### 3.2 手法の確認可能範囲

IPS が、STR の仕様記述可能領域についてどこまで確認が可能であるか検討を行った。以下に述べる。

- 状態記述の表現

IPS は、図 4 の IPS 表示画面例で示すように、STR の対象物の状態記述に一意に対応する状態図を用意している。このことで、対象物の種類や情報の種類の変

化に対応できる。他対象物との関係については、関係する対象物を結ぶ線の図形を用意する。これによって通信方向の変化なども表現可能である。

この様に、対象物、及び対象物間の任意の状態記述に対応する図を用意することで、あらゆるものの状態遷移の表現が IPS の手法によって可能である。対象物の数が増える場合でも、表示領域に対象物の状態図を順次追加することで可能である。表示領域の物理的制限は、スクロールや拡大機能などの表示方法によって対処出来る。

- イベントの表現と入力

イベントは、対象物に適用可能なイベント名を示すイベントメニューによってユーザに示している。イベントの入力は、メニューからの実行イベントの選択と、表示領域の対象物図のマウスクリックにより受け付けている。イベントが増加する場合、随時メニューの選択イベントを増やせばよいし、対象物の指定についても、表示画面に必要な対象物を全て生成することで指定が可能である。

以上の様に、IPS は STR 仕様記述に対して確認範囲の論理的な限界は見られず、内容確認手段の提供が全てに対して可能であると言える。

### 3.3 手法の拡張検討

本節では、2章で述べた STR 記述法の改善及び拡張に対応する IPS の機能拡張を検討する。

- 三つ以上の対象物を持つイベント記述に対する拡張

イベント実行時に対象物の指定入力を識別子の個数分、受け付ける様に拡張することで対応可能である。この拡張については既存の機能に影響を与えることなく機能追加と言う形で実現できる。

- 不特定多数の対象物の記述に対する拡張

任意の対象物の状態に対応する変数を含むルール適用の判定機能をインタプリタに追加することで対応可能である。同様に機能追加で実現可能である。

- 全ての対象物に共通する要素を含むサービス記述

共通要素を条件とすると、下記に述べるユーザに依存する要素を含むサービス記述と同様に考えることが出来る。



- ユーザに依存する条件を含むサービス記述

以下の二つの機能が必要である。

1. 条件と条件値の確認手段の提供機能

ユーザが意図する条件下でルールが適用が行われることを確認する手段と、実条件値の内容の確認を行う手段を提供する機能が必要である。

各々の値の表示とルールの適用を同時に表示することで、拡張が可能である。

2. 実条件値の獲得と条件の判定機能

システム内部で実条件値の獲得を行う機能と、その値が条件に当てはまるかどうかの合致判定を行う機能が必要である。

実条件値の獲得は、対象物の情報ファイルを別に入力するか、ユーザに直接問い合わせることで対応が可能である。また、条件の判定機能は、予め条件の種類毎に用意しておくことで対応が可能である。

今回、ユーザ依存の条件を含むサービスへの拡張一環として、上記で述べた機能を追加したスクリーニングサービス対応のIPSを試作した。図5に試作システムの表示画面例を示す。画面は、スクリーニングの際に必要な時間の実条件値について、時間設定要求のウィンドウがポップアップした場面である。この試作システムによってスクリーニングサービスの仕様確認を行い、手法の有効性と拡張性を確認した。

### 3.4 考察

以上、IPSの確認可能範囲と拡張について検討した。IPSの確認範囲は、STRの記述可能領域を全てカバーしており、2章で述べたSTRの記述仕様の拡張についても機能拡張によって対応可能である。

今後、IPSの課題として以下の様に、ユーザインタフェースの向上が挙げられる。

- 場所の表現

現在のIPSの手法は、対象物の場所の概念について特に考慮していない。既存の手法でも、対象物の状態図に場所の名前を示す部品を付加することで、対応が可能であるが、ユーザの側から見ると分かり難い問題がある。そこで、対象物の状態図と分けた場所の表現を別途検討する必要がある。

- イベントのイメージ表現

現在の手法によってユーザに提示されるイベントは、テキストメニューである。そのため、ユーザは、テキストの意味でしかイベントを知る手段がない。イベントの実際の操作を表現するためには、イベントメニューにイベント操作イメージをリンクさせるといった、イベントのイメージ化の検討が必要である。

- ユーザによるイベント入力以外の動作確認手段

既存の手法では、ユーザのイベント入力によってのみ実状態の遷移が実行される。ユーザが完全に仕様の内容を確認できたかどうかは、ユーザに一任されており、ユーザの未確認部分が生じる可能性がある。そのため、イベント以外の入力をユーザから受け付ける手段が必要である。例えば、遷移したい状態を入力することで、その状態の遷移までを自動実行してくれる様な機能が今後必要と思われる。

- 図形用語の統一

状態を示す図形は、意味が曖昧になることが考えられる。そのため、既存の手法では、状態を示す用語を図形用語として対象物の図形と共に表示する。その場合、用語の表記を統一しておかなければ、記述者でない参照者が誤った認識をするおそれがある。そこで、既に提案されている [8] 通信サービスの概念モデルによる意味の分類を行い、表記を統一する検討が必要である。

## 4 おわりに

### 4.1 STR 記述法について

2章では、STRの記述仕様から制約を抽出し、記述不可となるサービスを明らかにして、その改善案を示した。また、未検討であった対象物の共通要素とユーザ依存の条件を含むサービスについて拡張記述案を提案した。改善案、及び拡張案については、各支援手法の適用を考えた総合的な検討を今後行う必要がある。

今回は通信システムの分野に限った内容であったが、今後状態遷移に着目したルール記述を他分野へ応用することが考えられる。例えば、各種機器の制御を行う制御分野などが考えられる。他分野への応用は、時間的記述、状態やイベントの内包的な記述をどの様に定義するかなどが記述に当たっての課題である。

## 4.2 仕様確認支援手法について

STR の記述範囲内で、仕様を確認する手段が手法によって全て提供可能であることを確認した。また、2章で述べた STR 記述仕様の改善・拡張に対応する拡張も可能である。

今後は、ユーザインタフェースを考えて、システムの付加価値を高めていくことが課題である。例えば、場所の表現や、イベントのイメージによる表現、図形の曖昧さの排除や状態指定によるイベントの自動実行機能等の実現について検討を行なう必要がある。

## 参考文献

- [1] 原田, 平川, 竹中, 門田「サービス仕様の自動生成に関する考察」情報処理 39 回全国大会, 5S-5, '89
- [2] Y. Hirakawa Y. Harada T. Takenaka, "Behavior Description for A System which Consist of An Infinite Number of Processes", 1990 BILKENT International Conference on New Trends In Communication, Control, and Signal Processing, pp. 59-68, Jul., 1990.
- [3] Y. Hirakawa and T. Takenaka, "Telecommunication service description using state transition rules", Proc. Sixth Int. Work. Software Specification and Design, pp. 140-147, Oct., 1991.
- [4] STR(State Transition Rule) 記述仕様書, 平成 6 年 4 月, STR2.0
- [5] 原田, STR 手法と通信サービス競合 (非決定性) 検出に関する考察, ATR テクニカルレポート, 1995
- [6] 和田, 河原崎, 太田, 「非専門家を対象とした仕様記述支援法」, 信学技法, SSE95-47, 1995
- [7] 榎木, 小林, 太田「通信サービス開発へのオントロジーの導入」, 信学技法, SSE95-134, 1995
- [8] 榎木, 高見, 太田「通信サービス概念モデルの一考察」, 信学技法, SSE93-77, 1993

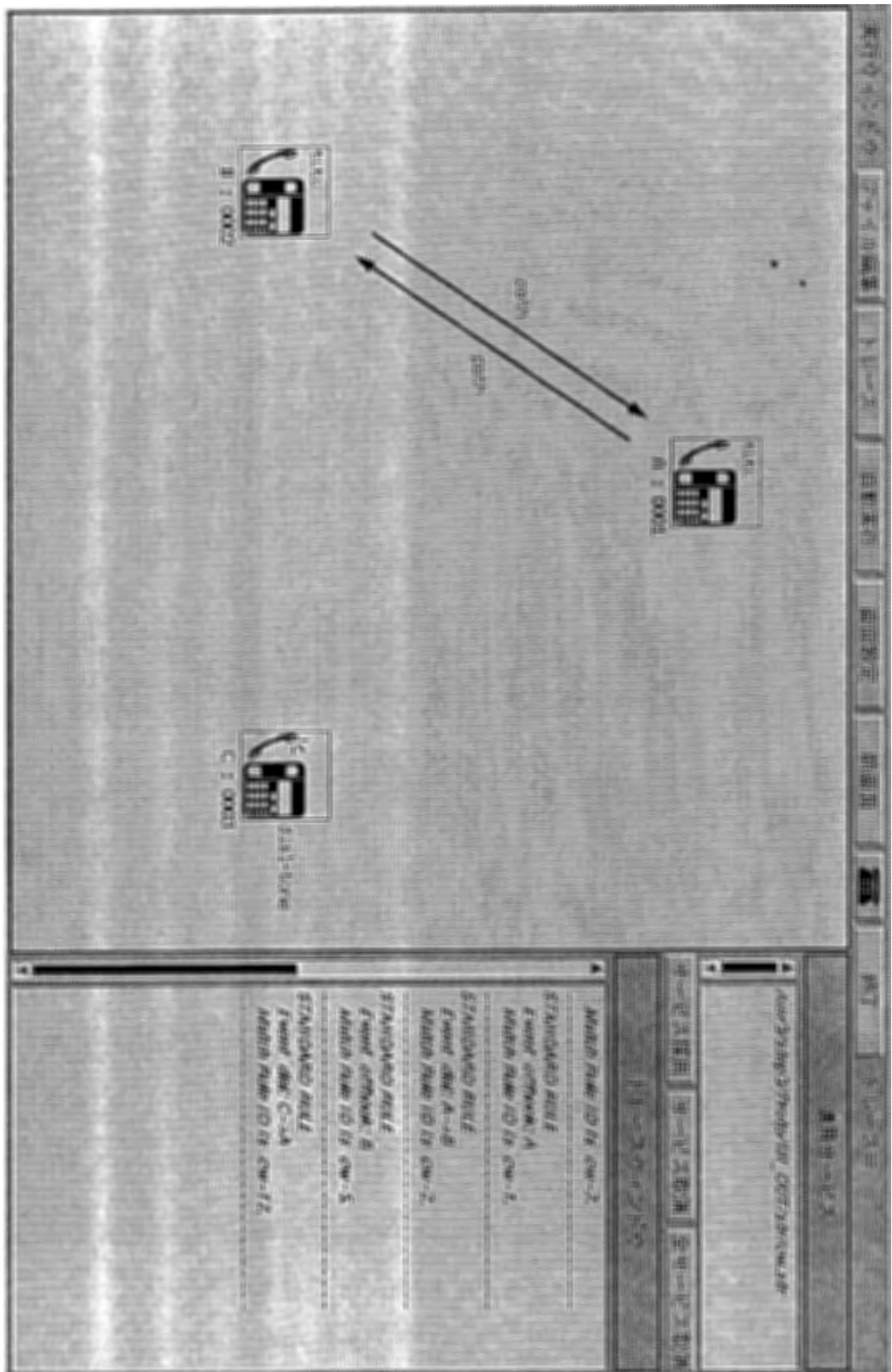


图 4: IPS 表示画面例

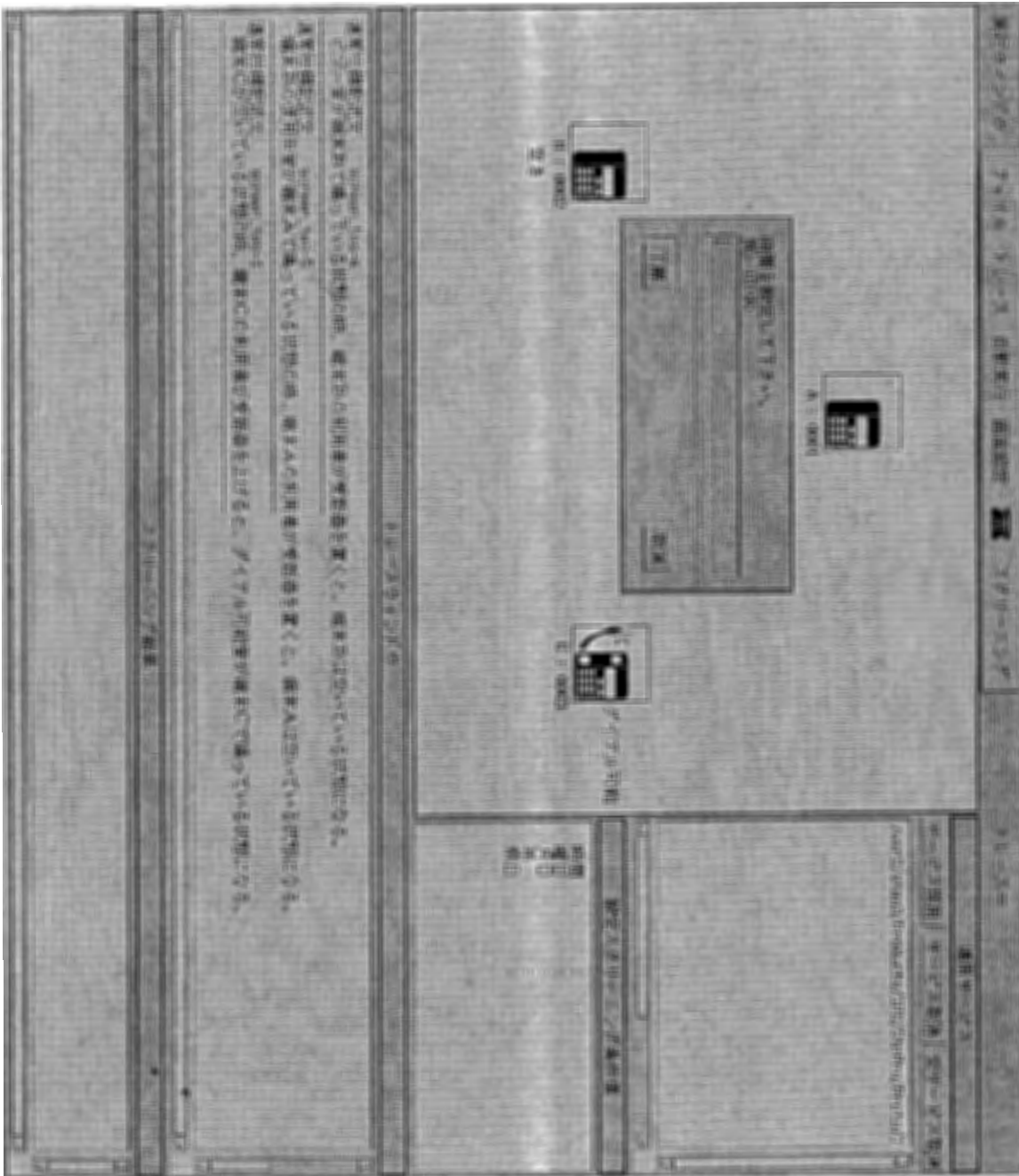


図 5: スクリーニングサービス対応の IPS 表示画面例