Vol.2014-DPS-160 No.11
2014/7/25

スマートシステムのための共生コンピューティングモデル

藤田 茂[1,a)]

概要：情報機器や家電，屋内設備等に対して「賢い」技術が求められている．我々は，機器側のシステムが，利用者のモデルを持って利用者の行動や意図を推定し，利用者にサービスを提供し，利用者の側がそのことに過度な驚きを示さずに，提供されたサービスを受け入れることを，「賢い」と捉えている．本稿では，我々がこれまでに研究してきた共生コンピューティングフレームワークの中で，賢い情報システムを構築するための設計モデルを定義し，この設計モデルに基づいて実装を行うことで，スマートシステムを構築するための概要を示す．

1. はじめに

スマートシステムという単語には，センサーネットワークから得られる大量のビックデータをクラウド上のサービスを使って処理し，現実世界へ応答を返すという意味 [2] や，いわゆる賢いサービスとして，利用者の行動を予測して，キッチンでの作業を支援するサービスを提供する行うスマートハウス [3] などの意味があると思われる．また，エネルギー生成，消費，回生を行うシステムや，太陽光や風力などを取り込んだエコシステムとしての意味を持たせた場合もあるが，本稿では対象としていない．

本稿では，スマートシステムを，“人と情報システムが互いに相手の行動を認知しているという前提のもとに，情報システムが動作を行い，人がサービスを受けるシステム”と定義する．

このスマートシステムを設計開発するために，1) 事前に情報システムと利用者が利用するサービスを十分に検討し最適なタイミングで適切なサービスを提供する手法，2) 利用者のサービス利用状況を利用者のフィードバックを得て機械学習し徐々に利用者に適したタイミングでサービスを提供する手法，3) 利用者モデルを構築して，提供したサービスが利用者に受け入れらたかの評価によってモデルを修正する方法と，その融合形態が示されている [3].

スマートホームという例では，家庭内の活動が，時刻によって発生確率が異なることが発見され，この確率に基づいてサービスを提供することで，利用者の満足度を高めることが可能であると示されている．

このように，ある特定の利用者，特定の環境，一連の動作の系列を時刻を追って機械学習とモデル修正，知識修正を行うことで，多くのルーチンワークに対しては，スマートなサービスを提供することができると考えられる．

しかし，我々は利用者の考える「賢いサービス」とは，ルーチンワークに対するサービスを越えたサービスの提供にあるのではないかと考えた．この「賢いサービス」の実現のためには，情報システムの側が利用者のことを認知し，そのことを利用者が知っていること，利用者もまた情報システムのことを認知し，情報システムの側がそのことを知っていることという，共認知のモデルを元にサービスを構成することで，ルーチンワークを越えた「賢いサービス」を提供できると考えた．

2. 共認知モデル

本稿で情報システムの共認知とは，情報システムが外界を知覚し，その外界の中の特定の利用者の行動モデルを，外界のモデルの中でシミュレーションできることを指す．定義上は，人間の側からも同様に，人間が外界を理解し，その外界の中の情報システムの行動をシミュレーションできること，という定義になる．

形式的にモデルを定義すると，情報システム I と利用者 U の間に共認知が成立している場合，情報システムにとっての外界 W_i と　利用者にとっての外界 W_u，情報システムからみた利用者 U_i と利用者から見た情報システム I_u の間の関係を，

$$\begin{cases} I(t+1) = U_i(t) \in W_i(t) \\ U(t+1) = W_u(t) \in I_u(t) \end{cases}$$

と表すとする．ただし，$t, t+1$ は単位時刻の推移を示すも

1 千葉工業大学情報科学部
Faculty of Information and Computer Science, Chiba Institute of Technology, 2-17-1, Narashino-shi, Chiba-ken, 275–0016, Japan
a) fujita@cs.it-chiba.ac.jp

ⓒ 2014 Information Processing Society of Japan

のとする．

3. 共生コンピューティングの概念

3.1 受動システムから能動システムへ

利用者に対して情報を提示するシステムは，図 5 に示すように，利用者からの明示的な要求に応答を返すもの，利用者に対して情報をプッシュするもの，一定の時刻に情報をプッシュするもの，利用者からの要求に対して別途情報を取得し加工した後に提示するもの，などが用いられている．

しかし，いずれもシステムが情報をプッシュしようとしたときに，利用者が不在である，異なる利用者が情報を受け取ってしまう，利用者が多忙であって情報を受け取る余裕が無い，利用がシステムの想定する情報を既に得てしまっている，などの事象が考えられる (図 6)．また実際に，情報検索研究において，利用者のスキルや知識を考慮に入れた研究が行われている．

情報システムが，利用者のことを想定していないという点に着目し，共生コンピューティング基盤では，パートナーエージェントというプログラムが，利用者に 1 対 1 で対応する仕組みを取っている (図 7)．

現時点では設計モデルのみであるが，情報システムの側が，利用者のことを常に認知することを前提とすることで，受動システムの持つ課題を解決しようとしている．この過程をモデル化するに当たって，Shoham のエージェント指向プログラミング [4] を用いる．

3.2 構成要素

共生コンピューティングは，人と既存情報システムとの間に存在する双方の認識のギャップを埋めて，人の活動を支援する情報システムの概念モデルである．

この共生コンピューティングを実現するために，利用者と一対になってサービスを提供するパーソナルエージェント (PA, Personal Agent) と，既存情報システムを利用して，パーソナルエージェントにサービスを提供する社会エージェント (SA, Social Agent) からなる共生空間 (Symbiotic Space) を構成する．

利用者は，パーソナルエージェントを介して，既存情報システムや，社会エージェントによって提供されるサービスを享受する．利用者の情報は，ユビキタスインフラストラクチャーや，既存情報システムを介して得られる，信号（分類，意味付けされていない値）として，共生コンピューティング基盤に与えられる．この信号から利用者の求めるサービスや利用者の状態を推定することは，そのままの形では困難である．

そこで，これらの信号から，いわゆる知的エージェントとしてのパーソナルエージェント，社会エージェントを作成可能にするために，記号への変換を行う機能が，パーセプチュアルファンクション (PF, Perceptual Function) である．

一方，既存情報システムの代表例として，ウェブサービスとそのウェブサービスを利用する社会エージェントと，パートナーエージェントとのやり取りを，信号として獲得し，その信号から新たな意味を付与した記号への変換を行う機能が，ソーシャルファンクションである．これら，共生コンピューティングの構成要素である，利用者，パーソナルエージェント，社会エージェント，共生コンピューティング基盤，既存情報システムの関係を図 1(共生コンピューティングフレームワーク) に示す．

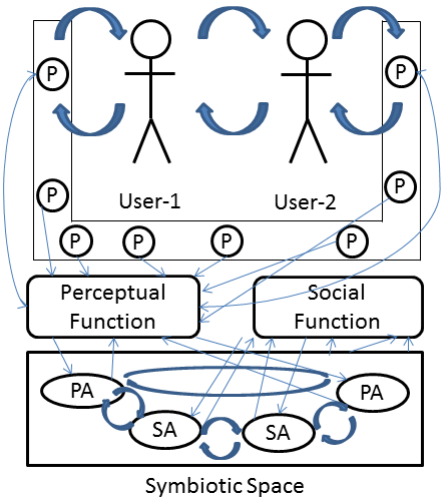

図 1 共生コンピューティングフレームワーク

Real Space からの情報は，信号 (signal) として共生コンピューティング基盤へ送られる．共生コンピューティング基盤は，信号 (signal) を記号 (icon) へ変換する．同様に，ソーシャル機能は，共生空間 (Symbiotic Space) 内のエージェントの相互作用を信号 (signal) として得て，同様に記号 (icon) へ変換する．

Real Space から得られる信号 (signal) は，以下の情報を持つ．

$signal =< time, position, type, value >$.

time は日時情報，position は位置情報，type は種別，value は値である．SA を介して得られない，利用者がウェブブラウザを直接操作して Digital Space にアクセスしている場合なども，上述の Real Space からの信号 (signal) として共生コンピューティング基盤に入力される．

共生コンピューティング基盤が出力する記号 (icon) は，共生空間の中で動作する PA, SA が利用可能な情報である．icon 自体は機械的に生成された識別子であり，それ自体には意味との対応は存在しない，記号 (icon) と意味との対応付けは，PA, SA 内部で行われる．

共生空間 (Symbiotic Space) は，複数のエージェントの動作を抽象的に表現した空間であり，実態は各計算機上で

動作するエージェント実行環境の集合である．このエージェント実行環境の動作は，共生コンピューティング基盤によって観測され，PA–PA 間，PA–SA 間，SA–SA 間の相互作用は，信号 (signal) として Social Function へ送られる．このエージェント間の相互作用の観測結果も，Real Space の信号と同様に分類され，機械的に生成された識別子である記号 (icon) として，再帰的に共生空間へ送られる．

3.3 共生コンピューティング基盤とエージェントの内部構造

共生コンピューティング基盤の主な機能は，信号 (signal) を記号 (icon) に変換するとである．Real Space, Digital Sapce からの信号 (signal) は共生コンピューティング基盤内部の履歴保存機能を介して，属性値化を経て，属性リストと変換され，分類器を経て，機械的に生成された記号 (icon) として，共生空間内のエージェントへ伝えられる．*signal* の定義は，前節に示した．

$icon =< identificationnumber, agentid, itime, iposition, itype, ivalue >$.

identificationnumber は, icon 自身の識別子である．現在の設計では UUID を用いている．agentid は, icon を利用するエージェントの識別子である．itime,iposition,itype,ivalue はそれぞれ，日時，位置，種別，値である．それぞの意味は signal と異なり，抽象化された状態での表現を許容する．例えば,「18632675-66b1-4e36-873b-4c9103b32fa9 は，初夏に，東京で，郭公が，鳴く」ことを表現する，という形で icon を示す．

エージェントは，記号 (icon) を分類器によって，エージェントの内部の推論に用いるシンボルのリストへ変換する．さらにシンボルのリストによって決定器がアクションを決定し，Real / Digital Space に対して動作 (action) を実行する．この動作結果は，さらに再帰的に共生コンピューティング基盤によって観測される．この動作は，再び記号としてエージェントに入力され，評価器によって，得られた記号が妥当であったのかの評価 (feedback) が行われる．

この評価結果は，共生コンピューティング基盤に伝達される．また，評価は，エージェント内部の分類器へも送られ，生成したシンボルが妥当であったのかを繰り返し評価し，分類器のパラメータを調整する．

$feedback =< identificationnumber, agentid, itype, ivalue >$.

それぞれの項目の定義は *icon* に同じである．

3.4 パーソナルエージェント, PA

パーソナルエージェントは，特定の一利用者に対して専従するエージェントである．共生コンピューティングの概念では，利用者に 1 対 1 に対応するパーソナルエージェントが介在する．直接的に身近なパーソナルコンピュータ上でパーソナルエージェントが動作しない場合であっても遠隔地の共生コンピューティング基盤を介して，パートナーエージェントは，利用者の挙動を把握可能である．

一方，利用者の側も，常にパーソナルエージェントが自らのためにサービスを提供するために待機していることを意識している．パーソナルエージェントは，PC や携帯電話，その他の装置を利用して，対応する利用者にパーソナルエージェントが利用者を知覚可能な範囲にとらえていることを定期的に通知する．

このパーソナルエージェント構築のためには，利用者を特定するための機能が不可欠であり，一般的なサービスとして共生空間へ記号 (icon) として利用者情報を通知する機能が必要である．

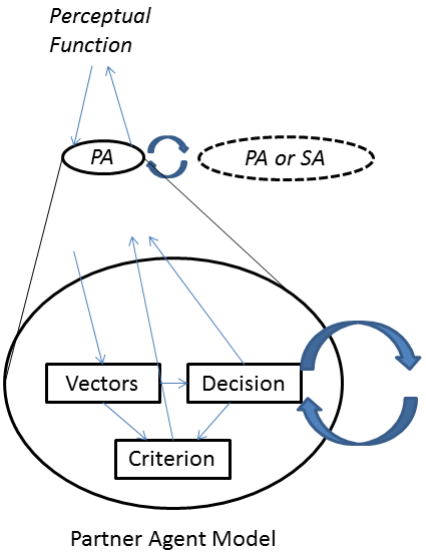

図 2 パートナーエージェントモデル

3.5 実空間知覚機能の設計

抽象的な信号 ← 記号の変換を行う Perceptual Function の設計を示したが，次いでここのセンサーやデバイスに対して記号への変換を行う，実空間知覚機能の概要設計を示す．

実空間知覚機能は，カメラやマイクロホン，圧力センサ，温度センサ，加速度センサ等の実空間から得られる信号を，パートナーエージェント，社会エージェントが処理可能な記号へ変換する機能である．

信号は，デジタルデータに変換されて，実空間知覚機能に取り込まれ，履歴保管機能にそのログが記録される．また，変換を行う属性計測機能によって信号から属性：属性値への変換が行われる．この属性計測機能は，それぞれのセンサから得られる信号を，一次的に記号に変換する機能である．この機能は，実空間知覚機能の設計者によって，設計された仕様に基づいて実現される．設計については，

ⓒ 2014 Information Processing Society of Japan

既存手法である分類アルゴリズムを用いる．分類アルゴリズムは既に多くの手法が提案されており，信号の性質に応じて，設計者が選択することになる．

記号生成機能は，属性：属性値の形式で与えられる情報を，パーソナルエージェント，社会エージェントの実装で用いられる記号へ変換する機能である．記号生成機能は，生成した記号に対する評価を，パーソナルエージェント，社会エージェントから得て，新たな記号生成の助けとする．

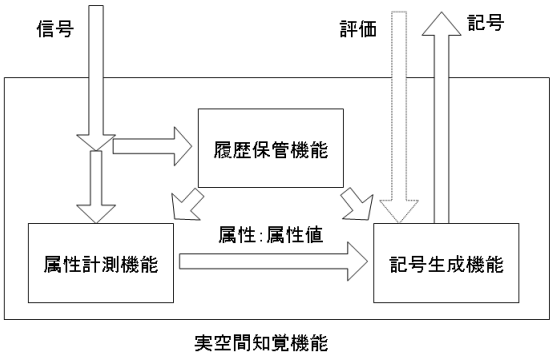

図 3 実空間知覚機能

4. シナリオ

本稿で述べる共生コンピューティング基盤の評価実験のためのシナリオを以下に述べる．

室内に 4 台のウェブカメラ (A, B, C, D) を設置し，常時画像データを取得する．4 では，おおよその撮影可能範囲を図示している．この室内が利用者が移動し，いくつかの椅子に着席する．

この室内を利用して以下の 3 つの機能が, 共生コンピューティング基盤から提供される記号化された情報に基づいて，パーソナルエージェントの機能として実現可能であることを示す．

(1) 利用者追跡機能
室内に設置したカメラからの画像を利用して，利用者特定を行い，パーソナルエージェントの対応する利用者の移動を追跡する．

(2) 利用者への追跡状況の通知機能
利用者の周辺にある装置を利用して，パーソナルエージェントが利用者の位置を把握していることを通知する．

(3) 利用者への情報提供機能
利用者自身のスケジュール，利用者の室内の行動履歴から，利用者の位置と，ウェブから取得される情報を用いて，利用者に提供することが望ましいと思われる情報を提示する機能.

次に各機能が目的の処理を達成していることを確認する手順について述べる．

機能 (1) の確認手順

i 室内のカメラ映像を監視する実空間知覚機能は，室内で発見した人の顔を記号化して，パートナーエージェントへ位置，時間情報と共に送付する

ii パートナーエージェントは，以前に利用者を特定した同種の記号があれば，対応する利用者であるとして追跡を行う

iii 追跡の過程で過去の動作履歴から推定される移動方向との一致度を計り，対応する利用者であることを確認し続ける

上述の情報はパーソナルエージェントへの問い合わせによって，確認することができる．

機能 (2) の確認手順

i 機能 (1) により，パーソナルエージェントは利用者の位置を把握する

ii 利用者に対して，位置を把握していることを通知するために利用者にもっとも近い PC 上のインタフェースを利用して，利用者の注意を引く．

(a) インタフェースはすべてのパーソナルエージェントが利用可能な形式で，室内のすべての PC 上で動作しているものとする．このシナリオでは，PC のスピーカーと PC 画面上に，パーソナルエージェントからのメッセージを表示可能とする．

機能 (3) の確認手順

i 機能 2 により，利用者はパーソナルエージェントが利用者の位置を認識し，位置を認識していることを通知する．

ii パーソナルエージェントは，利用者の行動履歴から利用者がまもなく外出するであろうことを推論する．

iii パーソナルエージェントは，外出時に “天候” が重要な情報であることを知識としてもっており，利用者に外出前に “天候” に関する情報を伝えることが，利用者への情報提供として妥当であることを推論する．

iv ウェブから現在地点に近い地区の天候情報を取得し，利用者に通知する．

以上のシナリオにそって，パーソナルエージェントが，サービスを提供するべき利用者を特定し，その利用者の室内での移動を追跡し，利用者の求めるであろう情報を適切なタイミングで提供できることを示す予定である．

4.1 メッセージ形式，データ構造

Shoham による AOP の提案では，Object Oriented Programming との構成要素の差は,

i 基本要素の状態を定義する語
信念 (beliefs)，コミットメント (commitments)，選択 (choices) など

ii メッセージの型
通知 (inform)，要求 (request)，提案 (offer)，拒絶 (de-

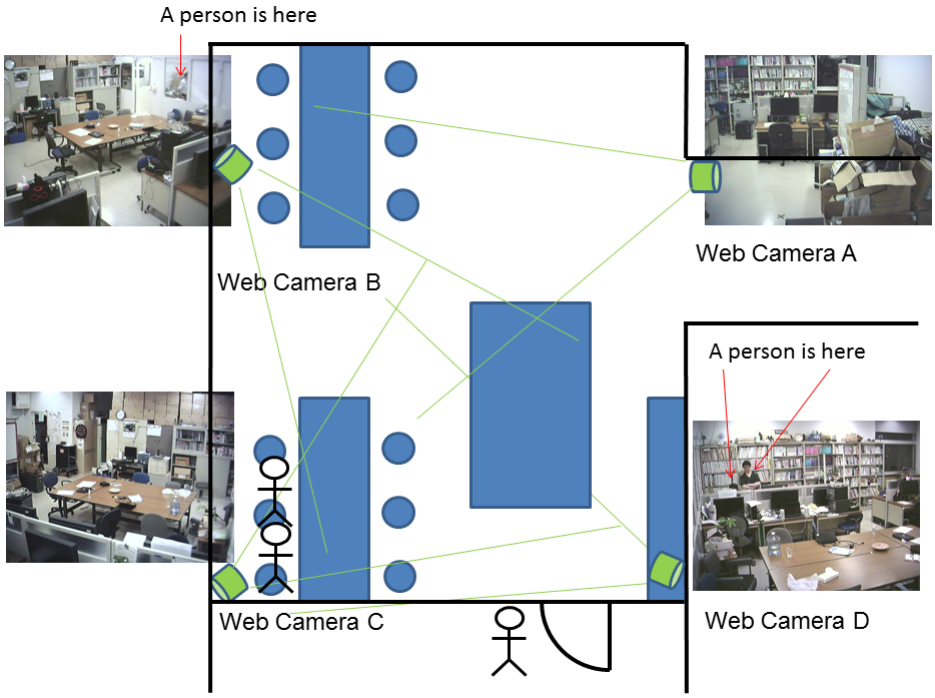

図 4 実験室イメージ

cline) など

iii メソッドの制約

誠実性 (honesty)，一貫性 (consistency) など

にある [4]．本稿でも，開発法論として AOP を踏襲する．

’はじめに’で述べたように，メッセージの型については，すでに FIPA/JADE 等で提案が行われ，エージェント研究者によって複数の試作システムが作られるなど，一定の実績があると思われる．そこで本稿では，エージェント間の通信プロトコルの再定義や，新規のメッセージ形式を提案することは行わない．

また，エージェント間通信で用いられるデータ構造については，既に LOD でも用いられている RDF が事実上の標準となっていると考えられることから，新規にデータ構造を定義することは行わない．

一方で，エージェントが新たな情報を獲得した後の動作について，AOP では単に “心的状態の更新 (Update mental state)” とかかれているのみであって，個別のエージェントに対する設計指針やエージェントのアーキテクチャについて記述がなかった．

本稿では，エージェント指向開発方法論を定めるためには，エージェントのアーキテクチャを新規に定めることが必要であるとして，以下にその要求仕様を示す．

4.2 Shoham のエージェントモデル

まず始めに，Shoham が AOP[4] の中で示したエージェントインタプリタの動作フローを図 8 に示す．

図 8 に示した要素は，以下の 7 項目である．

i Initialize mental state and capabilities
心的状態と能力の初期化を行う

ii Define rules for making new committments
新しいコミットメントを行うルールを定義する

iii Clock
エージェントの動作を駆動する

iv representation of mental state and capablity
現時点のエージェントの心的状態と能力の情報を保持する

v Update mental state
rule に基づいて，incoming message, (4) を使って，(4) を更新する

vi Execute commitments for current time
現時点に実行するべきコミットメントを実行する

vii incomming message, outgoing message
エージェント間通信を実現するメッセージ

エージェントの基本動作は次の 2 つのステップからなる．

i エージェントインタプリタは，メッセージを読み込み，心的状態を更新する．

ii 現時点のエージェントが実行するべきコミットメントを実行する．この結果，心的状態が更新される，またメッセージを他のエージェントに送信する．

Shoham のエージェントインタプリタのモデルでは，mental state の更新がエージェントの振る舞いを決定しているにも関わらず，その仕様の細部はエージェント作成者の設計に委ねられており，エージェント指向プログラミングを行うためには，より詳細なモデルを示す必要があると

考える．

4.3 情報システム構築のためのエージェントモデル

Shoham のエージェントモデルの中の，心的状態更新を次の 5 ステップで実行するように定める．

i representation of mental state and capability の保存を行う．このとき, 時刻 t-1 であるとすると, MSC(t-1) と表記する．

ii 時刻 t における incomming message の取り込み. このとき, incomming message の集合を IM(t) と表記する．

iii IM(t) と MSC(t-1) を Rule によって処理し，新しい representation of mental state and capability, MSC(t) が更新される．

iv MSC(t) に基づいて，コミットメントを実行する．これを C(t) と表記する．

v C(t) によって，新しい MSC(t+1) が更新される．

さらに，エージェントの心的状態 MSC の更新に際して，矛盾が生じた場合，これをなんらかの形で解消する必要がある．しかし，Shoham の AOP では，この矛盾に関しては言及が無く，AOP の作成者に委ねられている．

情報システムの動作中に発生する矛盾は，ソフトウェア工学的な観点からは要求分析が不十分であるとされ，システムエンジニアの対応が必要である．しかし，実システムが既に稼働している状況下においては，矛盾に気づくこと無く処理が継続され，情報システムの外部からの指摘によって，設計時の要求分析あるいは要求獲得が不十分であったことに気づく．

ここで，システムエンジニアが再設計を行ってシステムを改修するか，プログラマが稼働中のコードを改修してシステムを改修する．しかし，この工程は，利用者にとって稼働中のサービスを一旦停止し，再度の実行を伴うことから，大きな負担となる．

一方で矛盾の発生を情報システムの側が自律的に行う，あるいは，さらに推し進めて矛盾解消の手立てを情報システム側が行うことをエージェントモデルの中に取り込む．

4.4 矛盾発見時の処理フロー

データベースの処理において，トランザクションに失敗すれば，その処理自体をなかったものとしてロールバックする．しかし，エージェント指向モデルに基づいて，処理を行った場合，心的状態 MSC(t) が更新されてしまえば，C(t) に基づいて outgoing message は他のエージェントに配送され，そのメッセージを受け取ったエージェントの MSC(t) も更新されてしまう．エージェント指向モデルで情報システムを構築した際には，全体のロールバックという処理を組み込むことは出来ない．

エージェント単独の処理の中で矛盾が発見されるが，この矛盾対応処理について，以下のように定める．

i 矛盾発見

(a) コミットメント C(t) によって変化するであろう MSC(t+1) を予めエージェントが作成する．これを MSC'(t+1) と表記する．

(b) C(t) が実行された後に，MSC'(t+1) と実際に変更された MSC(t+1) を比較する．このとき差が大きければ，矛盾の発見である．

ii 矛盾発生の推定

(a) MSC'(t+1) を生成したルールを原因とする．

(b) MSC'(t+1) の生成要因となった incomming message を原因とする．

(c) MSC(t+1) を生成した C(t) の元となったルールを原因とする．

(d) (2a から 2c) の組み合わせを原因とする．

(i) (2a) と (2b) の組み合わせ

(ii) (2a) と (2c) の組み合わせ

(iii)(2b) と (2c) の組み合わせ

(e) 矛盾発生の推定を放棄する

iii 矛盾解消の試行

(2) で推定された原因を解消するための動作をコミットメントするルールを実行する

上述のフローに類似の概念として，仮説に基づく動作を実行できるインタプリタとして，Agent-1 の提案が [4] に存在するが，実現はされていない．また，エージェントが incomming message を全て真として取り込んでしまうことは，[4] でも意味論とアルゴリズム上の課題であることが指摘されている．incomming message を取り込んだ後に発生する矛盾については，上述のフローで対応可能である．仮説に基づく動作についても，コミットメントとして，部分的な MSC(t) のコピーを作成し，その MSC(t) にエージェント自体ののルールを適応することで対応可能である．

5. おわりに

スマートシステムを作るために，共認知モデルに基づいて動作するエージェントを用いて，システムを設計する概念を述べた．

参考文献

[1] 藤田茂, “共生コンピューティング基盤の設計 (2)”,DPS152,1993

[2] スマートシステム研究グループ, http://www/itri.aist-go.jp/group/ccsg.html 2014 年 7 月 2 日 アクセス

[3] Liming Chen, Chris Nugent, and George Okeyo, “An Ontology-Based Hybrid Approach to Activity Modeling for Smart Homes,” IEEE Transactions on Human-Machine Systems, Vol.44, No.1, February 2014, pp. 92–105.

[4] Y. Shoham, “Agent oriented programming”, Artificial Intelligence, Vol.60, No.1, p.51 - 92, 1993 システムへの適用からの考察”, JAWS2013, ポスター, 2013/9/19 ゲームにおける言語のモデル化”, JAWS2013, ポスター,

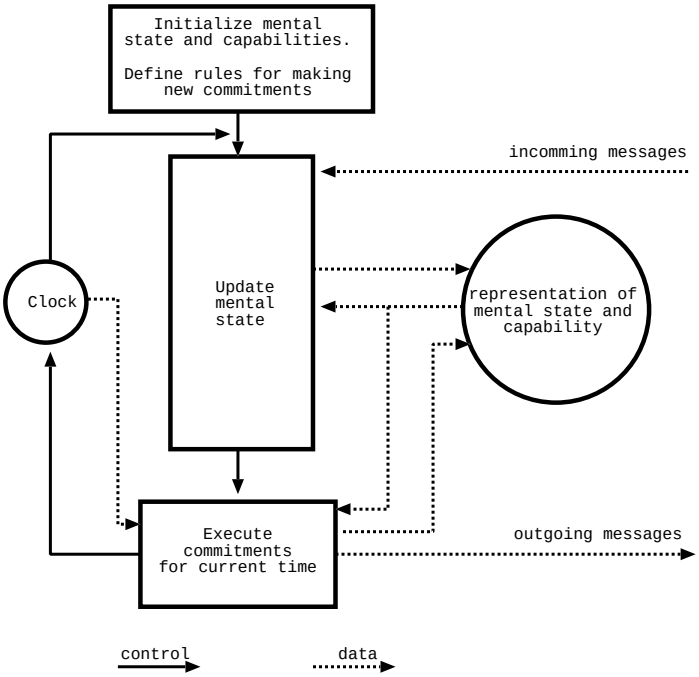

図 8 エージェントインタプリタ [4]

2013/9/20

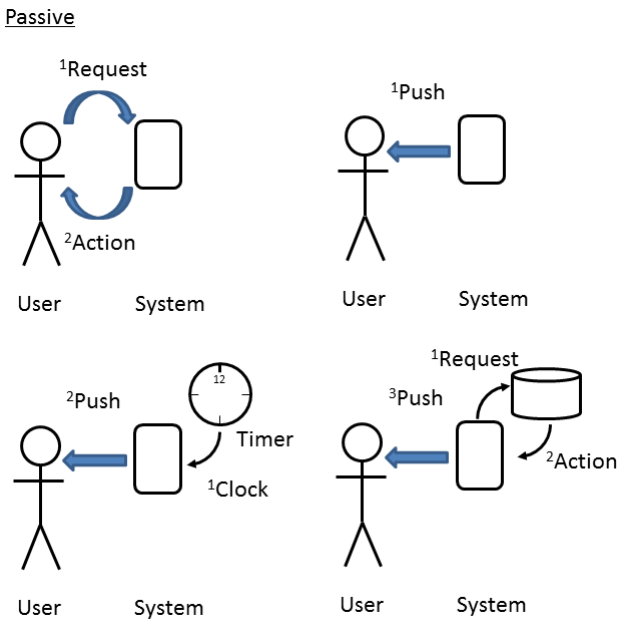

図 5 受動システム-1

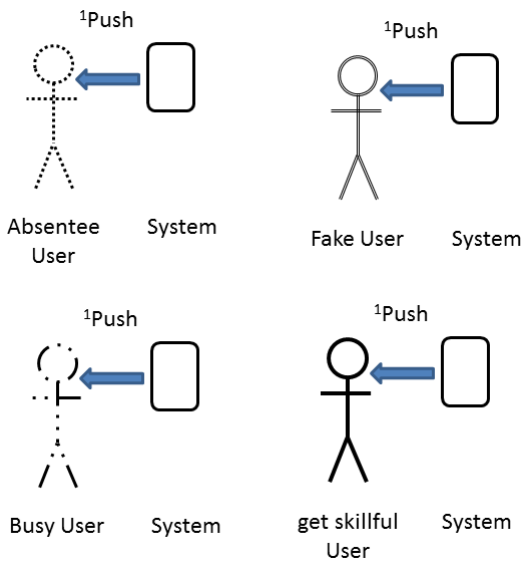

図 6 受動システム-2

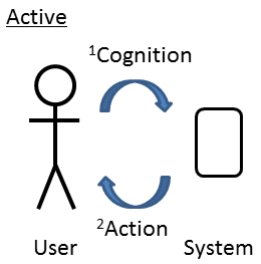

図 7 能動システム

ⓒ 2014 Information Processing Society of Japan