

システムの持続可能化技術基盤と言語処理系の開発 — 共生情報システムの実現へ向けて —

藤田茂¹ 滝雄太郎² 白鳥則郎³

概要：情報システムは社会の基盤であり、近年その持続可能性が重要な課題となっている。ここで、システムの持続可能性とは、システムを取り巻く内外の環境の変化、具体的には、社会、法、利用者、技術、障害発生などの変化や災害、コロナ禍などにおいても、これらの変化を効果的に吸収し、機能を維持・発展させサービスを提供し続けることを指す。このような持続可能性を有する情報システムとして既に我々は、共生情報システムを提案しており、本論文では、基本アーキテクチャとして新たにサステナブル層を導入し、この構成として、共生層、サステナブル基盤層、データ安全層を提案している。本論文では、特にこれらのサステナブル基盤層の記述言語としてポスト・エージェント指向に基づく言語「Double-21」とその処理系を提案する。Double-21 で記述されたシステムは、恒常性、発展性を持ち変化の認識を行い、システムの内外への働きかけを行いシステムの持続性を目指す。

キーワード：持続可能性, ポスト・エージェント指向, 共生情報システム

1. はじめに

情報システムは、高速化、大容量化、信頼性向上、廉価化、利用者体験の向上などを目指して研究開発されてきている。また、情報システムの普及に伴って、社会を維持するためのインフラとして重要性が日々増しており、安定性、保守性への要求が拡大している。情報システムの利用者は、概ねスマートフォンを所有する世代として、本邦では中学生から高齢者まで幅広い。この結果、情報システムに対する知識を一律に求めることは難しい。悪意ある攻撃者は、情報システム自体に対する攻撃を行ったり、利用者の知識不足や操作ミスを突いて、甚大な被害を巻き起こしたりしている。これと別に、災害（地震、豪雨、台風、大規模火災、など）を原因として、情報システムのインフラである電力、ネットワーク、計算機・ネットワークを構成するハードウェアに障害が発生することがあり、情報システムの持続性維持がますます困難になっている。情報システムの重要性の増加に伴って、いわゆる GAFA と呼ばれる巨大情報サービスが保存する情報と利用者個人の関係が複雑化し、アメリカ合衆国を起源とするこれら巨大情報サービスと、ヨーロッパ諸国の利用者の間の意識の齟齬から、法規制が新たに定められ、本邦の情報システムも対応を求められている。また、拡大する利用者数に比例するように、情報システムに保存されたデータ（画像、動画、顧客情報、など）が流出する事態が発生している。データ流出のみならず、データが意図せずして超長期に渡って保存されることから、二次被害（誹謗中傷）、三次被害（誹謗中傷の情報劣化によるデマ）が発生している。

現行の情報システムは、これらの変化を吸収させるために、人間が情報システムを設計開発して運用を持続している。このため、開発運用のコストは増加する一方である。

我々の研究グループでは、これら情報システムの持続性を実現するために、情報システムを取り巻く変化を 1) 利用者の変化、2) 社会の変化、3) 自然の変化としてとらえている（図 1）。また、1-3) の変化に対してシステム自身が変化を吸収するシステムを提唱してきた[1]。

変化 1) 利用者

システムが対象とする利用者は、子供・大人・高齢者と様々である。また、これら利用者自身の変化（成長、習熟、老化）に伴って、システムに対する要求やシステムが提供するサービスを変化させる必要がある。例えば、a1) 子供であった利用者が成長した場合、年齢制限を課す必要が無くなる。a2) 利用者がシステム利用に習熟した場合、確認を求める操作やインストラクションを提示する必要が無くなる。a3) 利用者の加齢に伴って視力減退が発生した場合、フォントサイズを大きくする必要がある、などである。

変化 2) 社会

人間社会は、様々な要因から法を変え、その解釈、運用を変える。この変化に従うことがシステムに要求される。例えば、ウェブアクセスのトラッキングに対する利用者の承認を求める動作は、EU での法成立によって、EU からのアクセスがありうるウェブサイトでは対応を余儀なくされた。また人間社会の文化は日々変化するために、利用者の意図しない行為が、おもいがけず社会へ影響を与えることがある。例えば、スマートフォンで撮影した画像のロコミサイトへの投稿（例：Google Maps）は、全世界への公開と、ほぼ永続的な画像保存と意図しない利用というリスクを持つが、これを全ての利用者（子供・高齢者）が理解していると期待することは出来ない。変化する社会情勢と構成員である人間からの要求は、既に 1 で述べた利用者変化、3 で述べる自然変化へ対応するための政策変更を引き起こす。

1 千葉工業大学情報科学部
Faculty of Information and Computer Science, Chiba Institute of Technology
2 千葉工業大学大学院情報科学研究科
Graduate School of Information and Computer Science, Chiba Institute of Technology
3 中央大学研究開発機構
Research and Development Initiative, Chuo University

<持続可能なシステムとは？> -- 変化(利用者、社会、災害、コロナ禍)を吸収 --

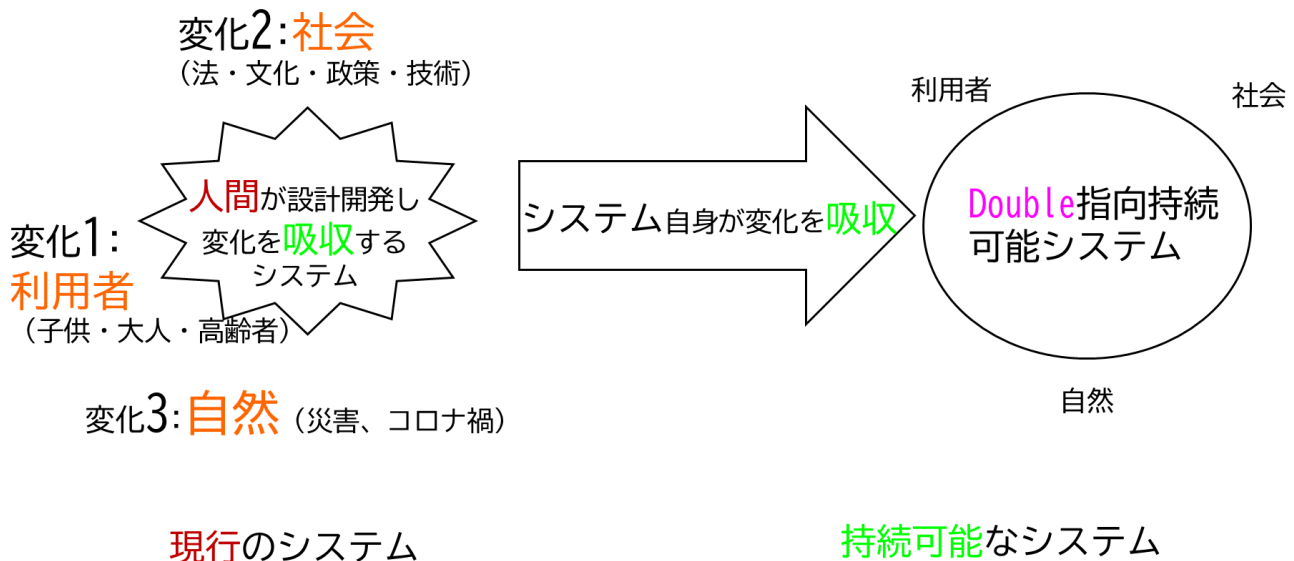


図1 システム自身が変化を吸収する持続可能なシステム

例えば、高齢化社会へ適応するためのユニバーサルデザインの義務化などである。また、情報システムではない技術変化が情報システムへ変化を求めることがある。例えば太陽光発電や風力発電の発達と、電力網ネットワーク維持である。

変化3) 自然

現在、猛威を振るっているコロナ禍は、自然変化の一つである。また、地球温暖化を原因とする異常気象の発生は、ヨーロッパでの水害の発生や、2021年8月時点での九州地方、中国地方を代表とする豪雨被害などとして顕在化している。十年前に発生した東日本大震災のみならず、熊本地震など、地震被害もある。これら災害に際してデータ保存を行うことは、地理的に距離のある複数のデータセンターへの保存によって達成している。これらの対応も人間が行っており、運用に係わるコストは増加する一方である。また単純に複数拠点にデータを保存することは、意図せぬデータ流出を招く恐れがある。

我々の研究グループでは、これらの変化に対してシステム自身が変化を吸収する Flexible システムの概念提案を行い[7]、概念を実証するための枠組みとして、オブジェクト指向、エージェント指向の次の考え方として、ダブル指向 (Double oriented) を提唱した[1]。

メッセージパッシング・カプセル化に代表されるオブジェクト指向のモデル、さらに自律性・エージェント間コミ

ュニケーションプロトコルを導入したエージェント指向のモデルが提唱されてきて一定の効果を上げてきた。しかし、依然として変化 1,2,3 への対応については、人間の作業が必要である。ダブル指向では、オブジェクト指向、エージェント指向に加えて、持続可能のための働きかけを内包することで、人間・環境・システム全体としての調和を達成し、持続可能な人間社会・環境・システムを構成する。

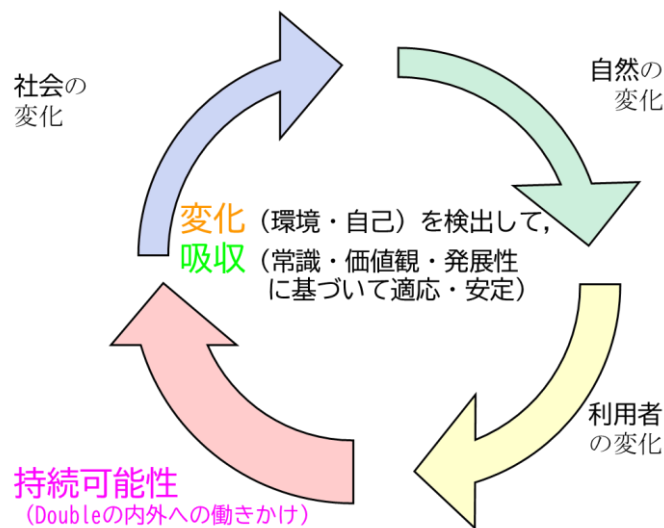
2. システムの持続可能化

Double 指向持続可能システムは、変化 (環境・自己) を検出し、これを吸収するためにシステムの内外へ働きかける。変化については1.で述べた。

Double では、エージェント指向で述べられている自律のみならず、社会、自然、利用者への働きかけと、Double 内部への働きかけによって、システムとそれを取り巻く社会、自然、利用者との調和を達成し、常識・価値観・発展性に基づいて適応・安定を達成する (図2)。

ポスト・エージェント指向モデルである Double は、オブジェクト指向モデルのようにメッセージパッシングによって通信をするモデルであり、エージェント指向モデルのように自律性を持って通信プロトコルに従ってコミュニケーションを行う。ポスト・エージェント指向 (持続指向) モ

<システムの持続可能化>



Double指向持続可能システム

図 2 システムの持続可能化

デルである Double は、自律性のみで自己の恒常性を維持するのではなく、環境への働きかけによって、自己の恒常性を維持しようとする。この過程で社会、自然、利用者といった環境との相互作用を考慮して持続可能性を高める。

自然、利用者、社会とシステムは相互に関係し、独立したパラメータとして制御・観測することが困難である。Double 指向持続可能システムは、これら内外の変化を検出し、システムの持つ常識、価値観、発展性に基づいて適応し、安定する。この時、単にシステム自体が内部を調整制御して安定するのではなく、環境（自然、利用者、社会）への働きかけを行い、その結果としてシステムが安定し全体が適用する点に新しさがある。

3. 持続可能化のための階層モデル

共生情報システム[36]の中で、持続層、共生層として示した層を共生層、サステナブル基盤層、データ安全層と詳細化し、全体をサステナブル層として定義した（図 3）。

物理層、ネットワーク層の上位にサステナブル層を位置づける。このサステナブル層の上位には、アプリケーション層が存在する。

物理層、ネットワーク層の概念は一般的のものであり、トランスポート層やセッション層、データリンク層等を含む。

3.1 データ安全層

システムに存在するデータは安全であることが求められる。従来の情報システムでも、物理層に相当する層では、冗長化などによって安全を保とうとしている、またネットワーク層に相当する層では、通信路の暗号化によって秘密を保持しようとしている。サステナブル層のデータ安全層ではこれに加えて、サステナブル層でデータとして認識されるデータの全てを設定の範囲で永続化、持続可能化、秘匿化を行う。直観的には利用者が想定するデータに対する信頼を保証する層である。

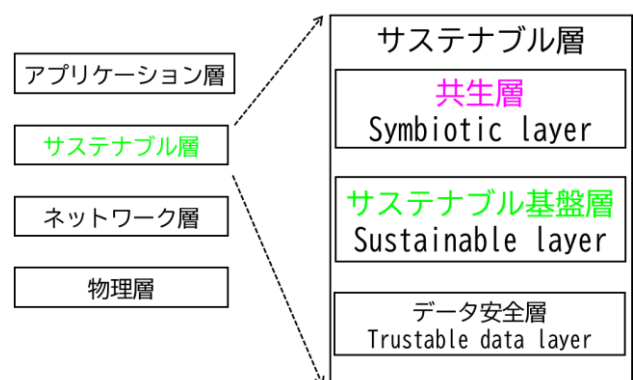


図 3 サステナブル層

このデータ安全層は、利用者から見ると自分のデータが自分の意図と社会の常識に従って保存されているように見える。また、ダブル指向モデルからは、利用しているデー

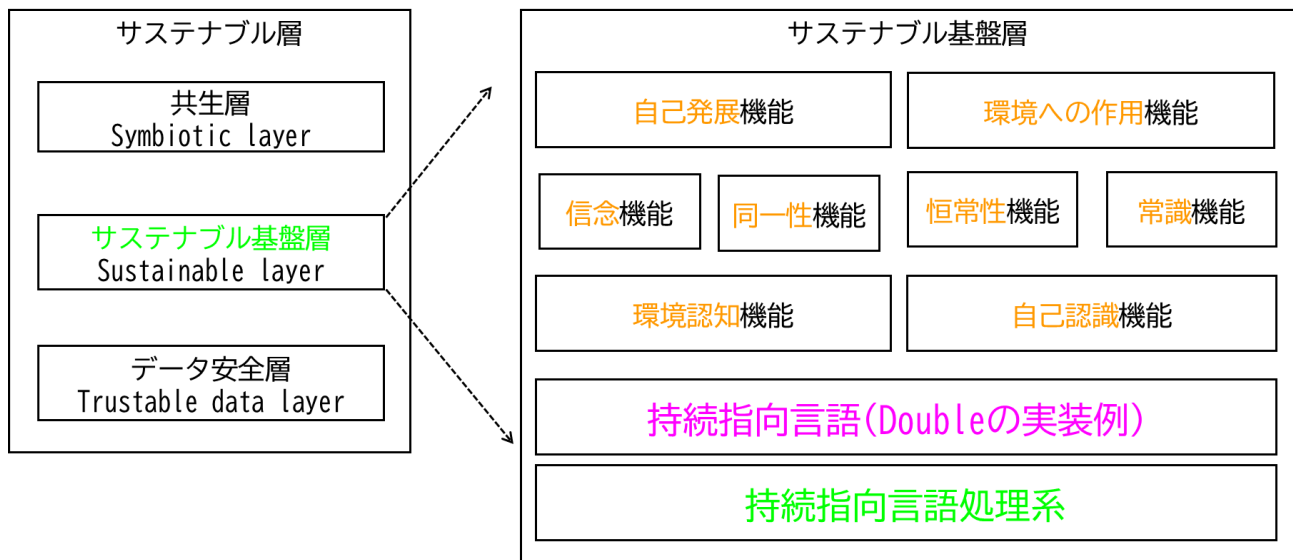


図4 サステナブル基盤層

タが常に存在して利用可能であることを保証している層である。この実装形態の一つとして、秘密分散[29,30]によるデータ保存手法がある。実装形態は秘密分散に限らず、他の暗号化形式でも代替可能である。

3.2 サステナブル基盤層

サステナブル基盤層の詳細は4.で述べる。サステナブル層は、ダブル指向モデルで実現される持続可能性を実現するための副機能からなる層である（図4）。サステナブル層の中核であり、それ自体の機能持続性のために、ポスト・エージェント指向言語 Double-21 のような、持続指向言語によって表現される。

サステナブル基盤層は利用者から見ると、サービスが一貫して提供されるように見える。このサステナブル基盤層はデータ安全層の上に位置し、データの安全性を担保した上でサービスを提供する。ここでデータの安全性とは、単にデータが失われないことのみを意味するのではなく、データへのアクセス制御や、利用状況の把握等の機能を含む。

3.3 共生層

共生層は、社会・環境との調整と調和を実現する。共生層はアプリケーション層の下に位置し、利用者と社会から見たサービスの実態が、そのアプリケーションを取り囲む社会の規範、常識、法、慣習、などに基づいて動作することを示している。

共生層はサステナブル基盤層の上に位置し、サービスの社会的な意味やコンテンツ、コンテキストを解釈した上で、アプリケーション層を構築するためのサービス API を公開する。

4. サステナブル基盤層

4.1 持続指向言語（Double-21）

持続指向モデル(Double)に基づくプログラミング言語が Double-21 である。Double の直観的な表現は文献[1]で与えられており、

Double = <Agent, Common Sense, Belief, ...>.

という形である。

この他、従来のエージェント機能に加えて、1. 信念、2. 常識、3. 発展性、4. 恒常性のための機能を備えている。Double の集合から構成されるシステムは、システムの外部の変化（ユーザ要求変化、誤操作、ワイドネットワーク帯域変化、等）や、内部の変化（CPU 負荷、ローカルネットワーク負荷、他の内部構成要素のレスポンス時間変化、等）に柔軟に対応し、内部構成変更や、外部（計算機環境、ネットワーク環境、利用者、社会、自然）への働きかけによって、安定して動作することが可能になる。

Double-21 のプロトタイプについては5.で述べる。

4.2 持続指向言語処理系

Double-21 の解釈実行を行う、持続指向言語処理系は、単に文法を解釈して計算を実行のみならず、4.3 以降で述べる機能を基に、自らを再帰的に定義することで、持続可能性を言語処理系自身が有するようになる。

例えば、言語処理系自身がデータ安全層の上に構築されているが、データ安全層の維持管理に必要な情報は、サステナブル基盤層の信念機能の一部に保存される。このため、データ安全層の機能の一部は、上位層であるサステナブル基盤層の機能によって実現され、データ安全層の持続性向上に貢献する。

持続指向言語処理系自身でセキュリティを向上させるために、持続指向言語処理系を秘密計算[14,36]に基づいて実装することが可能である。このような実装形態をとった場合には、サステナブル基盤層を構成する計算ノードからデータが漏洩したとしても、データ安全性が確保され、またサステナブル基盤層の通信を傍受しても、データは復元できず、改竄の発見が可能である。また、秘密分散と組み合わせることで、ノード故障耐性を向上させることができる。

持続指向言語処理系の構成例は、5.で述べる。この中で、持続指向言語処理系の構成ノードの情報を各ノードが保持することから、ネットワーク速度やノードの計算速度、利用可能なストレージサイズ、平均故障時間等から最適な利用ノードを決定することが可能である。

4.3 環境認知機能

環境には 1)計算機環境、2)ネットワーク環境、3)データ安全層のノード情報、4)自然環境、5)社会環境、6)利用者が含まれる。これらの環境を認知し、変化を検知する機能を持つ。

すべての環境変化を検知する必要は無いが、各サステナブル基盤がこれらの変化を検知し言語処理系へ通知する。また言語処理系は各 Double からの変化通知閾値の設定を受けて、通知間隔、通知の閾値を変更する。

4.4 自己認識機能

Double として実行する要素は、自己の動作を認識し、通常の範囲内での実行であるのか、通常の範囲を逸脱しているのかを、サステナブル基盤層からのデータに基づいて知る。これを自己認識機能と呼ぶ。

実現形態としては、通常の計算機プロセスであるが、Double の要素として、自己の実行パラメータを 4.3 の機能を通じて知り、自己の状態を把握する。

また、4.3 の機能を通じて利用者からの認知を認識し、共認知[8]が成立しているか認識する。

4.5 信念機能

エージェント指向プログラミング[37]で示された Belief は、その時点でのエージェントの知りえる（真であると信ずる）情報である。Double の信念機能は、この Belief を拡張し、Belief の破綻から、Belief の回復を行うための働きかけの系列を記述するように拡張される。4.10 で述べる恒常性機能へ働きかけが伝達される場合には、内部の調整として安定化を図る。一方、4.9 で述べる環境への作用機能へ働きかけが伝達される場合には、外部への調整として安定化を図る。

信念と 4.7 常識機能の一致が見られない場合、上位層である共生層からの制御が行われる場合、信念機能の出力がオーバーライトされる。

4.6 同一性機能

同一性機能は、ネットワーク上に複数のクローンが存在し得る分散システムとしての Double が、見かけ上は同一の存在であることを維持するための機能である。

バックアップや、障害回避の目的のために、ある時刻では、同一 Double に見える計算プロセス/サービスが存在することがある。これを利用者の側から見た時に、異なる Double と見せず、同一の Double とするための機能である。

4.7 常識機能

“常識”は共生層にも存在する。サステナブル基盤層での常識は、サステナブル基盤層間のコミュニケーションによって構成される、環境に対する共有化された知識である。例えば、ネットワーク帯域の時間的な変化、典型的な利用者の平均応答時間、同様のサービスを構成するノードの平均的な故障間隔などである。

Double の記述において、すべての“常識”を人が記載することは現実的では無く、Double の実行の結果、計算機やネットワーク、利用者、自然、社会からのフィードバックが蓄積されて“常識”を形成することになる。

4.8 自己発展機能

Double の内部への働きかけの結果、Double の動作パラメータ、動作ノードの移動、構成要素の変更が発生する。この働きかけに対応して自らの機能を発展変更する機能が自己発展機能である。

自己発展機能の実現形態として、予め準備したプログラム部品を利用する方式、動作パラメータを学習して最適な状態を反映する方式などがあり、我々の研究グループではこれまでに個別の機能について実験を行って報告[21, 38]している。

4.9 環境への作用機能

Double の特徴である環境への作用機能は、環境認知機能からの通知を受けて、常識機能に基づいて信念機能が自己発展機能で変化に対応するか、環境への作用機能を通じて対応するかの決定を受けて動作する。

環境への作用（働きかけ）は、さまざまな手段を必要とする。全てのサステナブル基盤層ですべての働きかけを実装する必要があるわけではない。環境への作用機能は、各システムのプラットフォームに依存する。

4.10 恒常性機能

Double の安定した動作は恒常性機能によって実現される。恒常性は、利用者、社会から見るとサステナブルなシステムとして期待される機能である。自然、社会が変化したとしても、システムが一定の応答を変わずに返すことが、期待される。

恒常性を維持するために、環境認知機能からの変化の通知、自己認識機能による自己の状態検査、信念機構による回復からの働きかけの決定、自己発展機能と環境への作用機能による、システム内外への働きかけによって、システムのサービスの安定性を確保し、利用者、社会から見ると恒常性をもったシステムとなる。

5. プロトタイプ : Double-21

現在、本稿で述べた概念に基づいて、持続指向モデルに基づいた Double のプロトタイプとして、Double-21 と呼ぶ処理系を実装中である。

Double-21 は、データ安全層として、一般化した N パーティ軽量秘匿関数計算[14]を用いている。このため、データも計算も秘匿性を保ったまま、ノードからのデータ流出があったとしても、データが復元できないという特徴を持つ。また複数のノードにデータが分散しているために、一つのノードが失われても、データ復元が可能である。詳細な議論は、文献[14]で述べた。

本稿で述べた Double の概念をすべて実装するのではなく、データ安全層の上に、自己参照的な言語処理系を構築できることの検証を目的として、Double-21 を設計実装している。このため、プログラミング言語のモデルとしては Lisp を採用している。

6. 関連研究と例題

6.1 関連研究

我々はこれまでに広く共生情報システムに関する研究を行ってきた。災害に強いネバーダイ・ネットワーク[2,3]、人間-コンピュータの共生システムに関する研究[6-12, 16-19]などである。情報システムは思いもよらぬ長寿命であることを期待されることがあり[15]、我々の提唱する共生情報システムのような持続可能性への期待は大きい。

情報システムの信頼性を向上させる研究で示されている検証はコストが高く、コミュニケーションの場にすべて適用することが困難であることが述べられている[13,34]。共生情報システムでも、人間とコンピュータの共生を目指して、安心・安全かつ持続可能なサービスの維持を信用に基づいて実行できる枠組みを目指している。またこれらのシステムを構成する方法として自律的に動作するエージェントからなるマルチエージェントシステムによる手法、人工知能によるコード修正等の技術が提案されている[22-28]。

データ安全層の実現のために様々な暗号化や信頼性向上の技術を検討しているが、我々は現在、秘密分散・秘密計算の技術を利用することを検討している[32]。

6.2 ストレージ不足への対応

ごく単純な持続指向モデルに基づく共生情報システムの動作をシナリオ例として示す。

- 1) ストレージの空き容量不足が環境認知機能より通知される。
- 2) 信念機能が空き容量を確保することを通知する
- 3) 常識機能の持つ情報使って恒常性機能がストレージの増加要求を通知する
- 4) 環境への作用機能が利用者に対してストレージの増加を要求する

ここまでの動作であれば、通常の OS(Windows, MacOS 等)が出す、「ハードディスクの空き容量が不足しました」というメッセージと大差が無い。

持続指向モデルに基づいているので、この後利用者から応答が無い場合でも、共生情報システムは、信念機構の動作により、ストレージ不足への対処を実行する。すなわち

- 5) 信念機構が常識機能の持つ利用者応答時間の範囲を超えて、応答が無いことを検出する
- 6) 環境認知機能からは利用者からの他の応答も無いことを確認する
- 7) 常識機能のもつ情報から、このままストレージの空きが不足する場合に、システムの動作が不可能になることが判明する
- 8) 信念機構はシステム内への働きかけとして、自己発展機能が常識機能の持つ利用頻度の少ないファイルかつ他のノードの持つ情報から復元可能なファイルを検索して、これを削除
- 9) 利用者へファイルの削除を通知

という動作を行う。これまでの情報システムが利用者応答を永遠に待ち続けてストレージ不足を原因とする持続性を失っていたことや、Linux にみられる OOM Killer による無作為にプロセスを消去することが無いという点で、持続可能性を示す例題としている。

6.3 ネバーダイプロフェッサ

研究者の研究活動は論文等にまとめられるが、その研究活動を支える人的ネットワークや研究資料の保存は、一部のノーベル賞級の研究を除いては行われていない。また、近年活発な電子ジャーナルは、出版社の倒産あるいは学会の破綻によって失われる可能性があることが指摘されている[33]。

一方で、研究分野の拡大に伴って、我々の社会を支える科学技術のエビデンスの保存が、今後の科学技術、社会活動の発展に重要である。我々は共生情報システムの応用例として、研究者の意図、データ、研究活動の持続を狙って、ネバーダイプロフェッサの概念を提唱し、共生コンピューティングによる設計を示している[36]。

6.4 デジタル寺院

利用者の死後に、利用者の意図に沿ったデータの活用、あるいはデータの廃棄等の意図を反映した動作を信託するにたる機構として、デジタル寺院の提唱[35]に基づく共生情報システムによる設計を行っている[4,5,31].

データの保存という観点のみでは、記録メディアの研究開発が行われており、数十万年保存可能なメディアが存在する[20]. しかし、メディアにデータのみを保存しても、その保存意図や利用目的が利用者の意図に合わねば、利用者がデータを後世に残さない場合があると考えると、我々は、デジタル識別子の永続的な保存から始まったデジタル寺院の概念を拡張して、データの永続的な保存をも目指して、デジタル寺院の再定義を行っている。

7. おわりに

情報システム持続可能性が重要な課題となっている。本稿では、システムを取り巻く内外の環境の変化(社会、法、利用者、技術、障害発生などの変化や災害、コロナ禍)を効果的に吸収し、機能を維持・発展させサービスを提供し続ける共生情報システムを実現するための、ポスト・エージェンツ指向言語 Double を実行するサステナブル基盤層について述べた。

Double で記述されたシステムは、恒常性、発展性を持ち変化の認識を行い、システムの内外への働きかけを行いシステムの持続性を目指すことを述べた。

また現在、ポスト・エージェンツ指向モデルに基づいてプロトタイプ: Double-21 を実装中である。

参考文献

- [1] 白鳥則郎, “ポスト・モダン分散システム”, 情報処理, Vol. 36, No.9, pp.88-91, 1995/9
- [2] Norio Shiratori, Noriki Uchida, Yoshitaka Shibata, Satoru Izumi, “Never Die Network towards Disaster-resistant Information Communication Systems,” ASEAN Engineering Journal Part D, Vol.1, No.2, pp.1-22, March 2013 [Invited Paper].
- [3] 白鳥則郎, 稲葉勉, 中村直毅, 菅沼拓夫, “災害に強いグリーン指向ネバーダイ・ネットワーク,” 情報処理学会論文誌, Vol.53, No.7, 1821-1831, July 2012 [招待論文]
- [4] 藤田茂, 樋地正浩, 滝雄太郎, 宮西洋太郎, 角田篤泰, 菅原研次, 白鳥則郎, “デジタル寺院”: 設計と開発へ向けて, 情報処理学会研究報告, Vol.2019-DPS-180, Vol.2019-EIP-85, No.10, pp.1-8, 2019/9/19.
- [5] 藤田茂, 樋地正浩, 滝雄太郎, 宮西洋太郎, 角田篤泰, 菅原研次, 白鳥則郎, “デジタル寺院”: モデルと基盤技術, 情報処理学会研究報告, Vol.2019-MBL-92, Vol.2019-CDS-26, No.10, pp.1-8, 2019/8/30.
- [6] Fujita, S., Sugawara, K., Kinoshita, T., and Shiratori, N., “An Approach to Developing Human-Agent Symbiotic Space”, Proc. of 2nd Joint Conference on Knowledge-based Software, pp.11-18, Bulgaria, 1996.
- [7] Takahide Maemura, Shigeru Fujita, Tetsuo Kinoshita, “Flexible Distributed System for Symbiotic Computing,” 8th IEEE International Conference on Cognitive Informatics (ICCI 2009), pp. 141-144, 2009.
- [8] Kenji Sugawara, Shigeru Fujita, “Non-verbal Interface of a Personal Agent based on Symbiotic Computing Model,” ICCI*CC2011, 2011.
- [9] Shigeru Fujita, Kenji Sugawara, “A Design Method for User Centric System Development by Symbiotic Computing,” Centric2012, 2012.
- [10] Kenji Sugawara, Shigeru Fujita, “Mobile Symbiotic Interaction between a User and a Personal Assistant Agent,” ICCI*CC2012, 2012.
- [11] Norio Shiratori, et.al., “Symbiotic Computing Based Approach Towards Reducing Users Burden Due to Information Explosion”, Journal of Information Processing, 2012.
- [12] Kenji Sugawara, Shigeru Fujita, “Interaction Zone between an office worker,” CSCWD 2011, 2011.
- [13] 西田豊明, @toyoakinishida, on Twitter, 2020/08/17, <https://twitter.com/toyoakinishida/status/1295171859747463168>.
- [14] 滝雄太郎, 藤田茂, 宮西洋太郎, 白鳥則郎: 軽量 N パーティ秘匿関数計算の一般化, 情報処理学会論文誌, Vol. 59, No. 10, pp. 1895-1902, 2018.
- [15] Richard Chirgwin, “Nuke plants to rely on PDP-11 code UNTIL 2050! Programmers and their walking sticks converge in Canada”, Wed 19 Jun 2013 // 05:59 UTC, https://www.theregister.com/2013/06/19/nuke_plants_to_keep_pdp_11_until_2050/ (last accessed 2020/08/19).
- [16] 藤田茂, “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”, 情報処理学会研究報告, Vol. 2014-DPS-160(11), pp.1-8, 2014/07/17.
- [17] 藤田茂, “エージェンツ指向システムによる情報システム構築のためのエージェンツに対する要求仕様”, 情報処理学会研究報告, 2013-DPS-157(11), pp. 1-6, 2013/10/10.
- [18] 藤田茂, “共生コンピューティング基盤の設計(2)”, 情報処理学会研究報告, 2012-DPS-152(2), pp.1-6, 2012/09/06
- [19] 藤田茂, “共生コンピューティング基盤の設計(1)”, 情報処理学会研究報告, 2012-DPS-151(12), pp.1-5, 2012/05/14
- [20] Eternal 5D data storage could record the history of humankind, published: 18 February 2016, <https://www.southampton.ac.uk/news/2016/02/5d-data-storage-update.page> (last accessed: 2020/08/19)
- [21] Takahiro Uchiya, Tetsuo Kinoshita, “Surveillance Architecture of Evolutional Agent System on Repository-based Agent Framework”, Proc. of the 8th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA2013), pp.614-617, 2013.
- [22] HORLING, BRYAN; LESSER, VICTOR., “A survey of multi-agent organizational paradigms”, The Knowledge Engineering Review; Cambridge, Vol.19, No. 4, pp. 281-316, 2004/12.
- [23] Yokoo, M., Sakurai, Y., & Matsubara, S., “Robust multi-unit auction protocol against false-name bids”. IJCAI International Joint Conference on Artificial Intelligence, 1089-1094, 2001
- [24] Esteban Real, Chen Liang, David R. So, Quoc V. Le, “AutoML-Zero: Evolving Machine Learning Algorithms From Scratch”, <https://arxiv.org/abs/2003.03384>, last revised 30 Jun 2020, (last accessed: 2020/08/19)
- [25] Esteban Real, “AutoML-Zero: Evolving Code that Learns”, July 9, 2020, <https://aigoogleblog.com/2020/07/automl-zero-evolving-code-that-learns.html>, (last accessed: 2020/08/19).
- [26] 高橋裕太, 佐藤亮介, 亀井靖高, 鶴林尚靖, “Stack Overflow 投稿を用いた深層学習による自動バグ修正にむけて”, 情報処理学会研究報告, Vol.2018-SE-200, No.3, pp.1-7, 2018
- [27] Zack Whittaker, “Amazon’s facial recognition moratorium has major loopholes”, <https://techcrunch.com/2020/06/10/amazon-rekognition-moratorium/>, June 11, 2020, (last accessed:

2020/08/19).

- [28] Devin Coldewey, "IBM ends all facial recognition business as CEO calls out bias and inequality", <https://techcrunch.com/2020/06/08/ibm-ends-all-facial-recognition-work-as-ceo-calls-out-bias-and-inequality/> June 9, 2020, (last accessed: 2020/08/19).
- [29] Adi Shamir, "How to share a secret", Communications of the ACM, Vol. 22, No.11, pp.612-613, 1997.
- [30] Blakley, G.R., "Safeguarding Cryptographic Keys", International Workshop on Managing Requirements Knowledge, (AFIPS) Vol.48, pp. 313-317, 1997
- [31] 樋地正浩, 橋祐一, 菊池一彦, 藤田茂, 宮西洋太郎, 白鳥則郎, "秘密分散法が切り開くデジタルコンテンツの相続ーデジタル寺院の実現に向けてー", 情報処理学会東北支部研究報告, Vol.2019, 2020/02/08, pp.1-5
- [32] 宮西洋太郎, 韓嘯公, 北上眞二, 金岡晃, 佐藤文明, 浦野義頼, 白鳥則郎, "クラウドサービス利用者の安心感を高める簡易的秘密計算法の提案", 電子情報通信学会情報・システムソサエティソフトウェアインタプライズモデリング研究会, 2014年度, 第1回 SWIM 研究会
- [33] 橋本誠志, "ペーパーレス社会における学会の破産と知的成果のサステナビリティに関する一考察", 情報処理学会研究報告, Vol. 2019-EIP-85, No. 12, pp.1-8, 2019/09/20
- [34] Jin-Hee Cho, Kevin S. Chan, "Building Trust-Based Sustainable Network", IEEE Technology and Society Magazine, Summer, pp.32-38, 2013
- [35] 角田篤泰, 山澤昌夫, 五太子政史, 白鳥則郎, "デジタル・アイデンティティの危殆化に抗う「デジタル寺院」構想", 日本セキュリティマネジメント学会, 第32回全国大会研究報告書, pp.1-6, 2018/6.
- [36] 藤田茂, 滝雄太郎, 白鳥則郎, "持続可能なセキュア共生情報システムの提案とデジタル寺院・ネバーダイプロフェッサへの応用", 情報処理学会研究報告マルチメディア通信と分散処理 (DPS) ,2021-DPS-186(70),1-8 (2021-03-08) , 2188-8906
- [37] Y.Shoham, "Agent oriented programming", Artificial Intelligence, Volume 60, Issue 1, p. 51-92, 1993
- [38] 藤田茂, 木下哲夫, 菅原研次, 白鳥則郎, "分散処理システムのエージェント指向アーキテクチャ", 情報処理学会論文誌, Vol. 37, No.5, pp.840-852(1996)