

CAPIS モデルにおける設計思考過程の継承に関する研究

—設計思考過程の継承の検証—

A Report on the transfer of design thought process in CAPIS model

—Verification of transfer of design thinking process—

安齊 龍也¹ 大山 勝徳² 武内 惇²

Tatsuya Anzai¹, Katsunori Oyama², Atsushi Takeuchi²

¹ 日本大学大学院 工学研究科 情報工学専攻

¹ Graduate School of Computer Science, College of Engineering, Nihon University

² 日本大学 工学部 情報工学科

² Department of Computer Science, College of Engineering, Nihon University

Abstract: 本論文はソフトウェアの分析・設計において、熟練者が持つ問題解決を行う際の意図や着眼点、常識・専門知識を他の分析者に伝達することを目的とする。しかし、これらは文章表現が難しい暗黙知であることが多く、他の分析者は必ずしも熟練者の設計判断を理解できるとは限らない。そのため、設計思考過程を表現するテンプレートとして CAPIS モデルを提案し、設計思考過程の表現法と利用法の開発を進めている。本稿では、CAPIS モデルにおける継承の有用性と設計思考過程の継承を検証するために、表現項目による思考過程の差異を CAPIS モデルにより可視化し、思考過程の継承と有用性を示す。

1. はじめに

熟練した設計者の多くが定年を向かえることにより、企業にとって重要な知識や技能が失われることが危惧されている。大規模なソフトウェア開発において高い品質を備えたソフトウェアを短期間で開発するには、熟練者の実務経験に裏付けされた設計技術を効果的に利用する方法が重要となる。実務経験に裏付けされた設計技術は、意図や着眼点、常識、専門知識を含み、熟練者の頭の中で無意識のうちに利用されるため、本人以外の技術者が理解することは難しい。

実務経験に裏付けされた設計技術の研究には、失敗知識データベース^[1] (以下、失敗知識 DB) や SECI モデル^[2] を用いた研究が多い。機能的知的外化ツール^[3] を用いた失敗知識と構造化を活用している。しかし、これらの研究では技術者の設計思考過程(以下、思考過程)を継承することが難しい。

本研究では、設計のノウハウを共有するために、CAPIS(CAusality of Problem-Issue-Solution)モ

デル^[4]に基づいて設計作業における設計者の意図や着眼点、常識、専門知識を表現し、かつ、他の設計者が利用できるようにする思考過程の表現法と利用法の研究を進めている^[5]。現在、CAPIS モデルに基づいて記述した思考過程を参照し、設計行為に取り掛かる場合に、知恵が継承されているのか確認することが課題である。

本稿では、CAPIS モデルによる思考過程の継承を検証するために、CAPIS モデル、失敗知識 DB、UML モデルとドキュメントの表現項目の違いによる思考過程の差異を CAPIS モデルにより可視化し、思考過程の継承に有用であることを示す。

2. 継承の定義

設計作業は設計要件を解釈して具体的な設計結果を定義するまでの手順で示すことができる。各手順を実行する思考過程は入力仕様を解釈し、問題を発見し、その問題を解決するための選択肢を想起して設計結果に具体化するまでの試行錯誤を繰り返す知的活動である。(図 1)

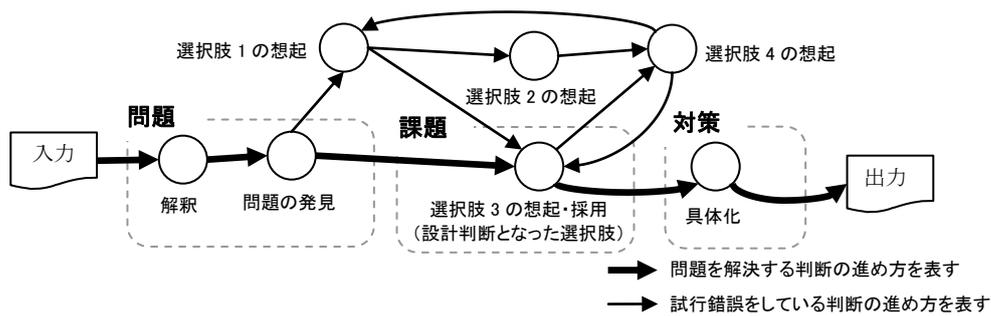


図 1. 設計作業における複雑な思考過程

思考過程は表 1 のようにデータ層、情報層、知識層、知恵層に分類することができる。思考過程の継承は、知識として記憶するだけでは効果が薄い。特に知恵(着眼点)を理解しなければ、利用者にとって知識の内容を実際に応用にすることはできない。着眼点は設計中に注目する失敗の原因や解決(アイデア)の組み合わせと考える。以上のことから、思考過程の継承は以下の 2 点が重要である。

- ① 問題、課題、対策のフローと選択肢を決定する設計判断を理解する。
- ② 熟練者の思考過程と、それを継承する他の設計者の思考過程で表現されている着眼点の類似数が多い。

表 1. ナレッジ階層の記述項目と利用者の動作

ナレッジ階層	記述項目	利用者の動作
知恵	問題を解決する糧の「判断」と、それを正当化する「理由」と「価値」(着眼点)	納得する
知識	常識や専門知識を表す「命題」(「ルール」と「事実」)	蓄積する
情報	データ層で表した記号に関する「意味の説明」	伝える
データ	文章や図で表現された「記号」や「記号例」	認識する

3. 思考過程の表現項目

CAPIS モデル、失敗知識 DB における目的と各モデルが表現している項目を以下に記す。

3.1 CAPIS モデル

CAPIS モデルは、熟練者の意図、着眼点、常識・専門知識を明示し、思考過程を他の設計者が利用可能にすることを目的としている。熟練者の思考過程を表現するために CAPIS モデルは設計行為を問題、

課題、対策に分類している。CAPIS モデルでは課題は問題解決の着眼点であるため、問題や対策に対する知恵を表現している。CAPIS モデルにおける問題、課題、対策の構成を表 2 に示す。

表 2. CAPIS モデルの表現項目

	入力概念モデル	思考概念モデル	出力概念モデル
知恵層	状況	選択肢	方法
	原因	着眼点(問題, 対策)	アイデア
	影響	意義	効果
知識層	事実問題	経験事実	実現方法
	前提条件	実証結果	実現結果
情報層	用語説明	用語説明	用語説明
データ層	設計要件	設計過程	設計結果

3.2 失敗知識 DB

失敗知識 DB は過去の失敗から得た知識や教訓を後世に生かすことを目的としている。失敗の内容を表現するために、シナリオに基づき問題、行動、結果に分類している。失敗知識 DB における問題、行動、結果の構成を表 3 に示す。

表 3. 失敗知識 DB の表現項目(まとめ)

問題	行動	結果
発生情報	対処	後日談
事象	対策	四方山話
経過	知識化	社会影響
原因		シナリオ
背景		

3.3 表現項目の比較

表現項目の違いを明確にするために過去の設計内容を表現し、次の設計に生かすことを目的としているモデルである CAPIS モデルと失敗知識 DB の表現項目の比較を表 4 に記す。UML モデルはソフト

ウェア設計の内容を表現することを目的としているため除外する。また、表現項目は CAPIS モデル基準とする。灰色部は失敗知識 DB が表現していると考えられる項目である。また、追記、シナリオは失敗知識 DB のみが表現している項目である。表 4 から失敗知識 DB や UML モデルだけでは思考の部分表現できないため、知恵継承は行うことができず、知恵継承を行う場合には不十分であることがわかる。

表 4. CAPIS モデルと失敗知識 DB の比較

入力概念モデル	思考概念モデル	出力概念モデル
状況	選択肢	方法
原因	着眼点(問題, 対策)	アイデア
影響	意義	効果
事実問題	経験事実	実現方法
前提条件	実証結果	実現結果
用語説明	用語説明	用語説明
設計要件	設計過程	設計結果
※追記, シナリオ		

4. 思考過程の差異の比較

本章では表現項目の違いによる設計思考過程の差異を可視化するために、CAPIS モデル記述実験を行ったロボコン参加者を対象として実験を行う。実験はオリジナルコース攻略に関する設計思考過程の記述実験を行い、その結果を考察する。

4. 1 目的

表現項目の違いより思考過程にどのような影響があるのかを可視化することにより、CAPIS モデルが思考過程の継承に必要な項目の多くを表現していることを検証する。

4. 2 実験環境

オリジナルコースは ET ロボコン⁶⁾の 2010 年度と 2011 年度コースを比較した場合に、2011 年度にはなく、UML モデル 2010、CAPIS モデルにより表現した 2010 年度コース攻略法を参考にした場合に、類似した機能のあるコースを作成した。作成したコースを図 2 に示す。また、コースを攻略する際のルールとして

- ① コースの全長は 2m×2m とし、制限時間は設けない
- ② コースの灰色部(①)は進入禁止とする
- ③ コースのスタート位置から反時計回りにスタートし、難所開始、難所終了を通過する
- ④ 高確率攻略を目標とするが、攻略確率に大きな差異がない場合は走行時間の短い案を採用とする

- ⑤ NXT は 2011 年度仕様とする
- の 5 つを設定する。

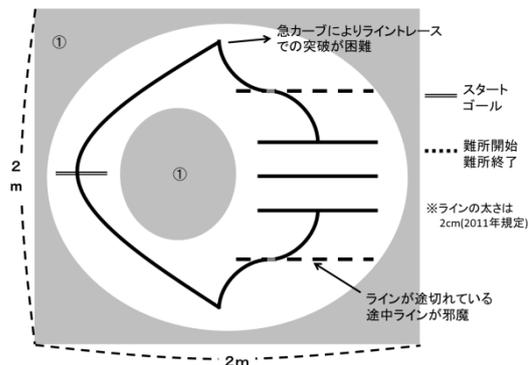


図 2. 実験用オリジナルコース

4. 3 実験手順

- ① 実験対象者を A, B, C の 3 つの班に分割する。
- ② 議題を発表する(オリジナルコースの高確率攻略)。
- ③ 全ての班に過去の UML モデルを配布する。また各班は以下の配布資料を追加する。
 - A) CAPIS モデルに基づいた表現項目
 - B) 追加なし
 - C) 失敗知識 DB に基づいた表現項目
- ④ 配布資料を参照し、個人で議題に取り組む。思考内容はノートなどに記述する。
- ⑤ 制限時間経過後、議題に対する解答を選択し、選択した内容に沿ってユースケース図、クラス図、ステートチャート図を作成する。
- ⑥ 作成した図を参照し、思考過程を記述する。

4. 4 仮説

- ① UML モデルの場合はソフトウェアに対する機能、構造、振る舞いを表すものであり、最終的な対策を表現するため、問題解決に関連する着眼点は少ないと考える。
- ② 失敗知識 DB は問題発生の原因、結果を詳しく表現しているが、その対処、対策を講じた理由の説明といった思考の内容は含まれている量が少ないと考える。そのため、CAPIS モデルでは着眼点(問題寄り)を含んでいる可能性はあるが着眼点(対策寄り)は含んでいないと考える。

4. 5 継承実験結果

実験結果の中で大きな差異がみられた表現項目について以下に示す。

4. 5. 1 選択肢

A, B, C 班別のコース攻略選択肢の数を表 5 に記す。表 5 の 0 内はオリジナルコース作成者であり、2010 年度 ET ロボコン参加者(以下、M とする)の選

択肢数である。表 5 から読み取れるようにオリジナルコース攻略に関する選択肢の数は A 班>C 班>B 班の順になっており、CAPIS モデルを用いた継承は失敗知識 DB を用いた継承に対して選択肢が継承でき、問題解決のための案が増えた。

表 5. 班別詳細

		難所攻略 選択肢	その他問 題選択肢	合計
A 班	対象者 1	5(3/3)	5(5/5)	10(8/8)
	対象者 2	4(3/3)	5(5/5)	9(8/8)
	対象者 3	5(3/3)	5(5/5)	10(8/8)
B 班	対象者 4	2(2/3)	2(2/5)	4(4/8)
	対象者 5	3(2/3)	2(2/5)	5(4/8)
C 班	対象者 6	3(2/3)	5(5/5)	8(7/8)
	対象者 7	3(2/3)	4(4/5)	7(6/8)

4. 5. 2 着眼点

「思考過程の継承」に重要な着眼点を示すため、対象者 1~7 の着眼点を原因、アイデアに分けて示す。

4. 5. 2. 1 原因

対象者 1~7 が原因の箇所として捉えたものの一例を表 6 に示す。原因一箇所の内容は記述した全ての対象者が「コース中の鋭角なカーブ」を示している。また原因一性質の内容でも記述した全ての対象者が「ライントレースでは曲がりきれない」を示している。

表 6. 原因の箇所、性質

項目	内容
原因一 箇所	コースの上端と下端のラインの角度が鋭く 鋭角な部分 ラインが鋭角すぎる 鋭角カーブ 急な角度の部分 コースのカーブが鋭い カーブの角度が鋭い
原因一 性質	ライントレースを行うことができない ライントレースができない 速度を維持したライントレースができない ライントレースでは曲がりきれない ライントレースでは旋回しきれない カーブが曲がりきれない ライントレースでは旋回しきれない
修正後	コース中の鋭角なカーブがライントレースでは曲がりきれない

4. 5. 2. 2 アイディア

対象者 1~7 がアイデアの箇所として捉えたものの一例を表 7 に示す。アイデア一箇所の内容は記述した全ての対象者が「コース中の鋭角なカーブ」を示している。またアイデア一性質の内容でも記述した全ての対象者が「特殊(ラインを無視する)走行を用いてラインを無視する」を示している。

表 7. アイディアの箇所、性質

項目	内容
アイ ディ ア - 箇所	ライントレースでは突破できない部分 鋭角なライン ラインの鋭角すぎる部分 鋭角カーブでライントレースできない部分 急な角度ライン 鋭いカーブ 鋭角な部分
アイ ディ ア - 性質	特殊走行と距離、角度検知を用いる 特殊走行で鋭角なラインを無視する 特殊走行を用いる ストレートランを用いて鋭角カーブを無視 ストレートランを用いてラインを無視する 曲進走行でラインを無視する 特殊走行でラインを無視する
修正 後	コース中の鋭角なカーブは特殊(ラインを無視する)走行を用いてラインを無視する

4. 5. 2. 3 着眼点のまとめ

原因、アイデアをまとめ、導き出した班別の着眼点を表 8 に示し、M の着眼点と比較した。その結果、CAPIS モデルを用いた継承は問題解決のための選択肢に重大な影響のある、着眼点を確実に継承することができた。

4. 5. 3 事実問題

A, B, C 班の事実問題の差異を表 9 に示す。表 9 から C 班の記述内容は A 班の記述内容よりも問題が発生する理由と発生した場合の状況を詳しく述べているため、失敗知識 DB の表現項目を用いた継承は CAPIS モデルの表現項目を用いた継承に対して問題解決のための原因究明が詳しくなる。

4. 6 実験結果の考察

CAPIS モデルは問題、課題、対策という問題解決のための設計思考過程を確実に継承できるため、熟練者の着眼点、知恵を迅速に継承し、活用する場合に優れている。また、失敗知識 DB は問題、行動、結果という問題解決のための原因究明に利用できるため、問題を掘り下げ、新たな手法を立てていく場合に優れていると考えられる。

5. おわりに

本稿では、CAPIS モデルにおける継承の有用性と思考過程の継承を検証するために、表現項目による設計思考過程の差異を CAPIS モデルにより可視化し、知恵の継承と有用性を示した。

今後の課題として、継承の判定や効果的な利用法を検討する。

表 8. A, B, C 各班別の着眼点の差異

項目	内容
Aー 共通	<ul style="list-style-type: none"> カーブが鋭角でライントレースでは曲がりきれないなら、特殊走行を用いてラインを無視する 難所でラインが途切れているなら、特殊走行を用いて突破する 特殊走行後のラインへの復帰が安定しないなら、曲進走行でラインに沿って接触する 難所の攻略判断に誤りがあり不安定なら、距離、角度検知以外の検知機能を用いる
Aー 個別	<ul style="list-style-type: none"> 倒立が安定しないなら、テール角度を調節する(A) ライントレースできない部分の多いコースなら、ライントレースせず攻略する(A, C)
Bー 共通	<ul style="list-style-type: none"> カーブが鋭角でライントレースでは曲がりきれないなら、特殊走行を用いてラインを無視する 難所でラインが途切れているなら、特殊走行を用いて突破する
Bー 個別	<ul style="list-style-type: none"> なし
Cー 共通	<ul style="list-style-type: none"> カーブが鋭角でライントレースでは曲がりきれないなら、特殊走行を用いてラインを無視する 難所でラインが途切れているなら、特殊走行を用いて突破する 特殊走行後のラインへの復帰が安定しないなら、曲進走行でラインに沿って接触する
Cー 個別	<ul style="list-style-type: none"> ライントレースできない部分の多いコースなら、ライントレースせず攻略する(F)
M	<ul style="list-style-type: none"> カーブが鋭角でライントレースでは曲がりきれないなら、特殊走行を用いてラインを無視する 難所でラインが途切れているなら、特殊走行を用いて突破する 特殊走行後のラインへの復帰が安定しないなら、曲進走行でラインに沿って接触する 難所の攻略判断に誤りがあり不安定なら、距離、角度検知以外の検知機能を用いる

表 9. 事実問題の差異

項目	内容
A 班	<ul style="list-style-type: none"> コースの鋭角な部分がライントレースでは旋回しきれず、コースアウトしてしまう 難所部分ではラインが途切れていてライントレースでは難所を突破できない ラインを無視するための特殊走行後の復帰が不安定である 難所突破判断のミスが発生してしまう
B 班	<ul style="list-style-type: none"> ライントレース中の走行体は急な角度のラインを走ることができずにコースアウトしてしまう ライントレース中の走行体はラインのない部分を走ることができずにリタイアしてしまう
C 班	<ul style="list-style-type: none"> コースの鋭角な部分はライントレースでは旋回しきれず、コースアウトしてしまう コースの鋭いカーブで減速した場合でも光センサの取得値などによりライントレースを維持できない 難所ではライントレースができないため、ライントレースのみでは突破できない 特殊走行後の復帰が環境によるズレで接触角度が鋭角や非接触によりライン復帰が安定しない

参考文献

- [1] 畑村洋太郎:“技術と創造と設計”, 岩波書店, 2006-11.
- [2] 野中郁次郎:“知識創造の経営—日本企業のエピステモロジー”, 日本経済新聞社, 1990.
- [3] 高藤淳, 來村徳信, 溝口理一郎, 機能的知識外化ツール OntoGear を用いた失敗知識の構造化と活用, セマンティックウェブとオントロジー研究会, SIG-SWO-A802-11(2008).
- [4] 大山勝徳, 武内惇, 藤本洋, CAPIS モデル方式による設計思考過程の表現法, 情報処理学会論文誌 Vol47, No8, pp, 136-139(2005).
- [5] 大山勝徳, 八木沼修, 武内惇, 藤本洋, 経験者の思考過程を用いたクラス抽出実験に関する報告, 組込みシステムシンポジウム ESS2006 論文集, pp. 104-108(2006).
- [6] ET ロボコン<<http://www.etrobo.jp>>