



Tokyo Gakugei University Repository

東京学芸大学リポジトリ

<http://ir.u-gakugei.ac.jp/>

Title	QOCと合理的意思決定法の組み合わせによる設計根拠獲得手法
Author(s)	小林, 祐介; 櫛山, 淳雄
Citation	情報処理学会論文誌, 49(7): 2258-2264
Issue Date	2008
URL	http://hdl.handle.net/2309/95488
Publisher	社団法人情報処理学会
Rights	<p>本著作物の著作権は（社）情報処理学会に帰属します。本著作物は著作権者である情報処理学会の許可のもとに掲載するものです。ご利用に当たっては「著作権法」ならびに「情報処理学会倫理綱領」に従うことをお願いいたします。 Notice for the use of this material The copyright of this material is retained by the Information Processing Society of Japan (IPSJ). This material is published on this web site with the agreement of the author (s) and the IPSJ. Please be complied with Copyright Law of Japan and the Code of Ethics of the IPSJ if any users wish to reproduce, make derivative work, distribute or make available to the public any part or whole thereof. All Rights Reserved, Copyright (C) Information Processing Society of Japan. Comments are welcome. Mail to address editj@ipsj.or.jp, please.</p>

テクニカルノート

QOC と合理的意思決定の組合せによる 設計根拠獲得手法

小林 祐介^{†1} 櫛山 淳雄^{†2}

近年、ソフトウェア設計において、設計根拠獲得の重要性が広く認識されている。そして、設計根拠獲得のために QOC (Questions, Options and Criteria) などの設計根拠モデルが提案されてきた。本稿では、この QOC に焦点を当て、Shum が実証的研究により明らかにした QOC の相対的重みに関する表現能力欠如の問題を合理的意思決定 (Rational Decision Making: RDM) を用いて解決する手法を提案する。RDM を用いることで不明確であった QOC の相対的重みが合理的に算出されるだけでなく、意思決定の結果、QOC に則った設計根拠を獲得することができる。

A Design Rationale Elicitation Method in Combination with QOC and Rational Decision Making

YUSUKE KOBAYASHI^{†1} and ATSUO HAZEYAMA^{†2}

It is widely acknowledged that elicitation of design rationale is important in software design. Thus far, design rationale models have been proposed such as QOC (Questions, Options and Criteria) to elicit design rationale. This study focuses on QOC and proposes a design rationale elicitation method by a rational decision making (RDM) that solves the issues pointed out by Shum, which are lacking in expression power with respect to relative weights for criteria and options in QOC. By using a RDM, not only relative weights for criteria and options in QOC are calculated in a rational way, but also design rationale based on QOC can be elicited as the result of a RDM.

1. はじめに

ソフトウェア設計では意思決定が繰り返し行われる。意思決定は後続の開発作業に影響し、また、類似のソフトウェア設計にとっては再利用可能な知識となるため重要となる¹⁾。しかし、意思決定とその根拠は獲得されることが少なく、時間の経過やステークホルダの入れ替わりにより徐々に失われる。そのため、意思決定過程の再現や知識としての再利用を困難にしている。

このような問題を解決するために、決定事項を裏づける知識や理由を意味する設計根拠を獲得する重要性が広く認識されている²⁾。設計根拠の獲得では、ソフトウェア設計の中で生じる多くの意思決定要素をどのように組織し、それを利用可能な構造に組み立てるかが重要であり³⁾、これまで IBIS (Issues-Based Information Systems)⁴⁾ や QOC (Questions, Options and Criteria)^{5),6)} など多くの設計根拠モデルが提案されてきた。

このような背景をふまえ、本研究ではソフトウェア設計における意思決定とその過程で生じた根拠の獲得に焦点を当てる。とりわけ、知識の再利用を指向している設計根拠モデル QOC に注目し、Shum が実証的研究により指摘した QOC の表現能力の問題点⁷⁾ に対し、合理的意思決定 (Rational Decision Making: 以下、RDM)⁸⁾ の適用を提案する。

以下では、2章で設計根拠モデル QOC の概要と問題点を、3章では RDM の概要を、4章では QOC の問題点を解決する手法の提案とその利用例を、5章では本手法の有効性、妥当性を、6章ではまとめと今後の課題について述べる。

2. 設計根拠モデル QOC

2.1 QOC の概要

QOC は MacLean らにより提案された Design Space Analysis (以下、DSA) の中で用いられる表記法である^{5),6)}。これは、ある問題に対し、なぜ特定の決定事項が選択されたのかを、解決策や考慮すべき評価基準からなる構造化された空間に整理して配置することで知識化する手法である。IBIS などの他のモデルに比べ、評価基準の要素を持ち、これに基づき整理することが可能であるため、知識の再利用性に富む。図 1 に QOC モデルの表記

^{†1} 東京学芸大学大学院
Graduate School of Education, Tokyo Gakugei University

^{†2} 東京学芸大学
Faculty of Education, Tokyo Gakugei University

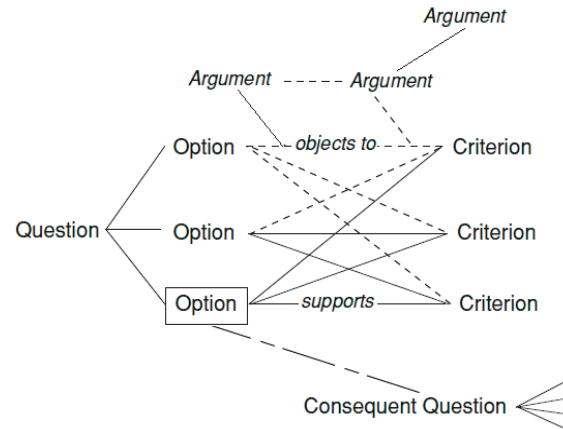


図 1 QOC モデルの表記法
Fig. 1 Notation of QOC.

		CRITERIA	
OPTIONS		Criterion1	Criterion2
Option1		+	-
Option2		-	+

図 2 QOC の設計空間
Fig. 2 The design space by QOC.

法を示す。QOC モデルは要素として解決されるべき問題を表す Question, 解決策を表す Option, 評価基準を表す Criterion, 関係として解決策と評価基準の間に supports (満たす) または objects-to (満たさない) の意味を持たせる Assessment からなる。また、これらの要素, 関係に議論要素を表す Argument を付与することもできる。

図 2 に QOC モデルを用いた設計空間を示す⁶⁾。行は Option の集合を, 列は Criterion の集合を表し, + (プラス) は supports を, - (マイナス) は objects-to の Assessment を表している。

2.2 QOC に対する問題点

Shum は設計過程に QOC を適用した実証的研究を行い, QOC の表現能力について以下の問題点を指摘した⁷⁾。

- Criteria に対する優先度
問題に対して, もしある評価基準を優先する必要がある場合, supports 関係の数を数えるだけで Options を選択するには意思決定としては不十分である。
- Assessment の相対的加重
ある評価基準に対して解決策を評価する場合, supports と objects-to (または assessment なし) だけでは相対的な加重を判断することができない。
これらは, DSA がシンプルに表記でき, 直観的に理解できるという立場をとっており, 実用面を配慮していることに起因する⁷⁾。

これに対し本研究では, 意思決定の観点から, Criteria の優先度や Options の Criteria に対する Assessment は意思決定に影響を与えるため上記問題点を配慮すべきであるという立場をとる。また, 設計知識の再利用の観点からも, 不明確な表現を含む知識の再利用は困難であり, 明確な表現であるべきという立場をとる。

3. 合理的意思決定 (RDM)

3.1 合理的意思決定 (RDM) の概要

RDM とは, 選好に影響する意思決定である⁸⁾。これは意思決定の際, 数学的計算を利用することで特徴付けられ, 代替案の中から最適な選択を行う際に用いられる。

RDM は 3 つの特徴を持つ⁹⁾。

- **Decision alternatives.** 代替案はアクションの集合, または各アクションに対して考えられる結果の集合として表現される。
- **Utility function.** 各アクションはその結果に基づいて重み付けがされる。
- **Probabilities.** 決定事項は, 特定の案が選択された場合に生じる結果に応じた情報を保持する。

RDM は 3 つの前提を持つ¹⁰⁾。

- ① アクションや結果の集合が既知である。
- ② 代替案の組合せの大きさや, 数学的計算にかかる時間は問題にならない。
- ③ 意思決定者の目的は最適な選択である。

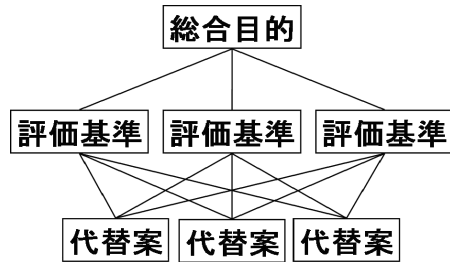


図 3 AHP の階層構造
Fig. 3 Structure of AHP.

3.2 Analytic Hierarchy Process

Saaty により提案された Analytic Hierarchy Process (以下, AHP) は RDM の 1 手法である¹¹⁾.

AHP の手順としては, まず問題の要素を総合目的, 評価基準, 代替案の関係でとらえ, 図 3 のような階層構造を構築する. その後, 各評価基準の重要度を一対比較によって求め, さらに各評価基準の観点から代替案の重要度を求める. 最後にそれら両者から総合目的の観点での代替案の重要度を算出する.

4. 提案手法

本研究では, 2.2 節で述べた Shum の指摘, すなわち設計根拠モデル QOC の表現能力欠如に対して, RDM, とりわけその 1 手法である AHP を適用することでこれを解決する. RDM を適用することで, その特徴の 1 つである Utility function から, QOC の Assessment に対し重み付けを行い, 上記問題の解決が期待できる.

4.1 概念モデル

図 4 に本研究が提案する概念モデルを示す. モデル要素は従来の QOC モデル要素と同じではあるが, 任意の要素間に Assessment の関係を持たせている点が異なる. そして, この Assessment は数値を持つ. 以下でこの数値について説明する.

- **Question-Criteria 間 Assessment の値 (w_i)**
Question に対する Criterion $_i$ ($i = 1, 2, \dots, m$) の重要度
- **Criteria-Options 間 Assessment の値 (v_{ij})**
Criterion $_i$ ($i = 1, 2, \dots, m$) に対する Option $_j$ ($j = 1, 2, \dots, n$) の重要度

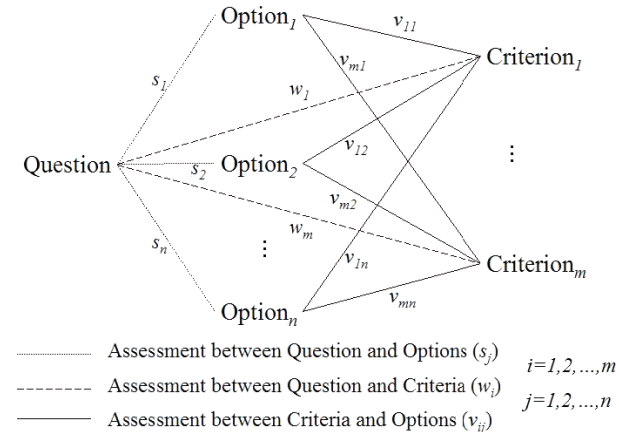


図 4 概念モデル
Fig. 4 Our conceptual model.

CRITERIA		
OPTIONS	Criterion1 [w_1]	Criterion2 [w_2]
Option1 [s_1]	v_{11}	v_{21}
Option2 [s_2]	v_{12}	v_{22}

図 5 提案手法を用いた設計空間
Fig. 5 The design space by our method.

- **Question-Options 間 Assessment の値 (s_j)**

Question に対する Option $_j$ ($j = 1, 2, \dots, n$) の総合評価値

図 5 に, 本手法を適用した場合の設計空間を示す. 従来の設計空間に比べ, 提案手法を用いた設計空間では Criteria に重要度 w_i が, Assessment に重要度 v_{ij} が, Options に総合評価値 s_j が付与されている.

4.2 提案手法の手順

本手法は以下の 4 つの手順からなる. 4.1 節で述べた Assessment の値は AHP に基づき算出される.

[手順0] 意思決定要素の分類, ラベル付け, 構造化

意思決定要素として, 意思決定すべき Question, 考慮すべき Criteria, 考えられる Options を特定, 分類し, 各々に対しラベル(要素に対する名前)をつける. 得られた要素の集合に対し, 概念モデルのように構造化を行う.

[手順1] Question-Criteria 間 Assessment の値の算出

Question に対し, 考慮すべき Criteria に関する一対比較を行い, 一対比較行列を作成する. この一対比較行列は m 次正方形行列で, この成分 a_{ij} は次のような意味を持つ.

$$a_{ij} = (\text{Criterion}_i \text{ の重要度}) / (\text{Criterion}_j \text{ の重要度})$$

この値には 9 点法を用いる¹¹⁾. 対角成分については $a_{ii} = 1$, 対称成分については $a_{ji} = 1/a_{ij}$ をそれぞれ満たす.

この一対比較行列の最大固有値 λ_{\max} と主固有ベクトル W を求める. このとき, 得られた固有ベクトルの成分 w_i ($i = 1, 2, \dots, m$) は Question に対する Criterion_i の重要度を表す.

一対比較の整合性, すなわち一対比較結果の矛盾の有無は, 通常, 以下に表される整合性尺度 (C.I.: Consistency Index) によって表現される. この値が経験的に 0.1 (または 0.15) 以下であれば整合性があると見なし, これよりも大きければ一対比較をやり直す.

$$\text{C.I.} = (\lambda_{\max} - m) / (m - 1)$$

[手順2] Criteria-Options 間 Assessment の値の算出

手順1と同じように, 各 Criterion_i ($i = 1, 2, \dots, m$) に対し Options に関する一対比較を行い, 作成された一対比較行列の主固有ベクトル V_i , および整合度 C.I. を算出する. V_i^t を成分に持つ行列を改めて V とすると, V の成分 v_{ij} ($i = 1, 2, \dots, m, j = 1, 2, \dots, n$) は, Criterion_i に対する Option_j の重要度を表す.

[手順3] Question-Options 間 Assessment の値の算出

手順1で算出した W と, 手順2で算出した V を用いて次式より総合評価ベクトル S を算出する.

$$S = V^t \cdot W$$

この総合評価ベクトルの成分 s_j ($j = 1, 2, \dots, n$) は, Question に対する Option_j の総合評価値を表す.

4.3 提案手法の利用例

4.2 節で述べた本手法の手順を, 利用例を用いて詳述する. 利用例としては, 通信方式設計を考える.

表1 通信方式設計における意思決定要素の例

Table 1 Elements of decision making in this example scenario.

Element of decision making	label
Question	What is optimal communication scheme?
Option ₁	HTTP
Option ₂	HTTP+SSL
Option ₃	HTTP+XML
Criterion ₁	Time Efficiency
Criterion ₂	Security

まず手順0より, 意思決定要素の分類, ラベル付け, 構造化を行う. この結果, 表1に示す要素が特定されたものとする.

次に, 手順1より, Question-Criteria 間 Assessment の値の算出を行う. この例では, Criterion_1 の Time Efficiency より Criterion_2 の Security を重視するものとし, ここで, 式(1)のような一対比較行列 A を与えた場合, この主固有ベクトル W は式(2)のように得られる. W の成分 w_i は Question に対する Criterion_i の重要度を表す. たとえば, $w_2 = 0.666667$ は Question に対する Security の重要度を表している. また, 整合度は $\text{C.I.} = 0$ となり, 整合性は保たれている.

$$A = \begin{pmatrix} 1 & 1/2 \\ 2 & 1 \end{pmatrix} \quad (1)$$

$$W = \begin{pmatrix} 0.333333 \\ 0.666667 \end{pmatrix} \quad (2)$$

次に, 手順2より, Criteria-Options 間 Assessment の値の算出を行う. まず, Criterion_1 , すなわち Time Efficiency に対する Options の重みを算出する. ここで, 式(3)のような一対比較行列 A' を与えた場合, この主固有ベクトル V_1 は式(4)となる. この場合の整合度も $\text{C.I.} = 0.004601$ となり, 整合性は保たれている.

$$A' = \begin{pmatrix} 1 & 2 & 3 \\ 1/2 & 1 & 2 \\ 1/3 & 1/2 & 1 \end{pmatrix} \quad (3)$$

$$V_1 = \begin{pmatrix} 0.539615 \\ 0.296961 \\ 0.163424 \end{pmatrix} \quad (4)$$

同様にして Criteria₂, すなわち Security に対する Options の重みを算出し, 式 (5) の一対比較行列 A'' から式 (6) の主固有ベクトル V₂ が得られたものとする. この場合の整合度も C.I. = 0.012297 となり, 整合性は保たれている.

$$A'' = \begin{pmatrix} 1 & 1/5 & 1/4 \\ 5 & 1 & 2 \\ 4 & 1/2 & 1 \end{pmatrix} \quad (5)$$

$$V_2 = \begin{pmatrix} 0.097390 \\ 0.569541 \\ 0.333069 \end{pmatrix} \quad (6)$$

ここで, これらベクトルの転置を要素として持つ行列 V を式 (7) のように改めて定義すると V の成分 v_{ij} は Criterion_i に対する Option_j の重要度を表す. たとえば, v₂₃ = 0.333069 は Criterion₂ の Security に対する Option₃ の HTTP+XML の重要度を表している.

$$V = \begin{pmatrix} 0.539615 & 0.296961 & 0.163424 \\ 0.097390 & 0.569541 & 0.333069 \end{pmatrix} \quad (7)$$

最後に, 手順 3 より, Question-Options 間 Assessment の値の算出を行う. 式 (2) の W と式 (7) の V から式 (8) より総合評価ベクトル S を算出する. S の成分 s_j は Question に対する Option_j の総合評価値を表す. この例では, s₂ の値が最も高く, この結果から Option₂ の HTTP + SSL が決定事項として選ばれたことが分かる.

$$S = V^t \times W = \begin{pmatrix} 0.244798 \\ 0.478681 \\ 0.276520 \end{pmatrix} \quad (8)$$

5. 議 論

本章では, 提案手法の有効性と妥当性について述べる.

5.1 提案手法の有効性

本研究の提案手法がどのように Shum の指摘を解決するかについて, 4.3 節の利用例をもとに述べる.

図 6 に, 4.3 節の例に対する従来の QOC を用いた設計空間を示す. 見てとれるように, Criteria 間の優先度や Assessment 間の相対的重みは不明確である. これに対し, 提案手法を適用した場合の設計空間を図 7 に示す. QOC に対して Shum が指摘した Criteria 間の優先度が不明確である問題点に対し, 提案手法では RDM を用いることでその Utility function から Criteria 間の優先度を合理的に算出している. この算出結果 w_i は, 意思決定の際に Criteria に対し明確な判断基準を与えるだけでなく, この設計知識を再利用する際, 意思決

QUESTION: *What is optimal communication scheme?*

OPTIONS	CRITERIA	
	<i>Time Efficiency</i>	<i>Security</i>
<i>HTTP</i>		—
<i>HTTP+SSL</i>	—	+
<i>HTTP+XML</i>	—	+

図 6 従来の QOC 設計空間の例
Fig. 6 Example of the design space by QOC.

QUESTION: *What is optimal communication scheme?*

OPTIONS	CRITERIA	
	<i>Time Efficiency</i> w ₁ =0.333333	<i>Security</i> w ₂ =0.666667
<i>HTTP</i> s ₁ =0.244798	v ₁₁ =0.539615	v ₂₁ =0.097390
<i>HTTP+SSL</i> s ₂ =0.478681	v ₁₂ =0.296961	v ₂₂ =0.569541
<i>HTTP+XML</i> s ₃ =0.276520	v ₁₃ =0.163424	v ₂₃ =0.333069

図 7 提案手法を用いた設計空間の例
Fig. 7 Example of the design space by our method.

定時にどの Criterion を優先したかの明確な根拠を与える．設計時と設計知識の再利用時では優先順位が異なることも考えられるので，この値は再度意思決定を行ううえでの基準となりうる．

また，QOC に対して Shum が指摘した Assessment の相対的重みが不明確である問題点に対し，提案手法では上記同様 Utility function から，Criteria に対する Option の相対的重みを算出している．従来の QOC では supports と objects-to (または assessment なし) だけであった Assessment が，合理的に算出された v_{ij} から Criterion_{*i*} に対する Option_{*j*} の重みを見てとることができる．この v_{ij} は，意思決定の際，Options から最終的な選択を行う際に明確な判断基準を与えるだけでなく，この設計知識を再利用する際，ある Option が Criteria をどの程度満たしているかを見てとることができる．

5.2 提案手法の妥当性

QOC モデルに AHP のような RDM を適用するには，3.1 節で述べた RDM の 3 つの前提を満たす必要がある．以下では，3.1 節の前提をふまえ，提案手法の妥当性について考察する．

① について，1 章で述べたように設計根拠の獲得は意思決定要素を組織し，それを利用可能な構造に組み立てることが必要であり³⁾，記録される要素は十分に知られていることが前提である．このことは ① の前提を満たす．

② について，Zannier らの実証的研究によると，設計根拠は記録される意思決定要素の数が実用的な限度を超えて大きくならないと述べている¹²⁾．そのため，代替案の組合せの大きさや数学的計算にかかる時間はそもそも大きくならない．このことは ② の前提に合う．

③ については，設計根拠は意思決定を正当化するものであり，必ずしも最適な選択でなくてもかまわない．そのため，③ の前提を満たしていない．これに対し，Zannier らの実証的研究によると，ソフトウェア設計における意思決定では RDM と Naturalistic Decision Making¹³⁾ が並行して利用されると述べているため¹⁴⁾，最適な選択が望まれる意思決定の局面は十分にあると考える．また，最適ではない意思決定が望まれる場合には，QOC の Argument 要素で意思決定に対する補足的な説明を加えることで設計根拠を補足する．

5.3 設計根拠獲得のためのコストの問題

設計根拠の獲得には獲得にかかる負荷の問題が繰り返し述べられている^{2),7)}．これに対し，本手法では RDM を取り入れることで，意思決定の結果が設計根拠の表記として得られる．そのため，提案手法では改めて設計根拠を書き起こすコストは生じない．

6. まとめと今後の課題

本稿では，設計根拠モデル QOC に対する Shum の指摘に対し，RDM の Utility function を利用した解決法を提案した．

今後は提案した概念モデルを具現化するシステムの構築，および提案手法の評価を行う．また，ANP (Analytic Network Process) など他の RDM に対しても適用可能かどうか検討を行う．

参 考 文 献

- 1) 古宮誠一：ソフトウェア設計上の意思決定とその根拠の情報を再利用する方法，電子情報通信学会論文誌，Vol.J84-D-I, No.1, pp.78-89 (2001).
- 2) Tang, A., Barbar, M.A., Gorton, I. and Han, J.: A Survey of the Use and Documentation of Architecture Design Rational, *Proc. 5th Working IEEE/IFIP Conference on Software Architecture (WICSA '05)* (2005).
- 3) Hu, X., Pang, J., Pang, Y., Atwood, M., Sunand, W. and Regli, W.C.: Survey on design rationale: Capture, representation and retrieval, *Proc. 2000 ASME Design Engineering Technical Conferences and the Computers and Information in Engineering Conference* (Sep. 2000).
- 4) Kunz, W. and Rittel, W.: Issues as elements of information systems, Working paper 131, Center for Planning and Development Research, University of California, Berkeley (Aug. 1970).
- 5) MacLean, A., Young, R.M. and Moran, T.P.: Questions, Options, and Criteria: Elements of Design Space Analysis, *Human-Computer Interaction*, Vol.6, No.3&4, pp.201-250 (1991).
- 6) MacLean, A., Young, R.M., Bellotti, V. and Moran, T.P.: Design Space Analysis: Bridging from Theory to Practice via Design Rationale, *Proc. Esprit '91*, pp.720-730, Brussels (Nov. 1991).
- 7) Shum, S.B.: Analyzing the Usability of a Design Rationale Notation, *Design Rationale: Concepts, Techniques, and Use*, Moran, T. and Carroll, J. (Eds.), Mahwah, Lawrence Erlbaum (1996).
- 8) Stirling, W.C.: *Satisfying Games and Decision Making, with Applications to Engineering and Computer Science*, Cambridge University Press, Cambridge, UK (2003).
- 9) Simon, H.: A behavioral model of rational choice, *Quarterly Journal of Economics*, Vo.69, No.1, pp.99-118 (1955).
- 10) Orasanu J. and Connolly, T.: The reinvention of decision making, *Decision Mak-*

ing in Action, Klein, G.A., Orasanu, J., Calderwood, R. and Zsombok, C.E. (Eds.), Ablex, New Jersey, USA (1993).

- 11) Saaty, T.L.: *The Analytic Hierarchy Process*, McGraw-Hill, New York (1980).
- 12) Zannier, C. and Maurer, F.: A Qualitative Empirical Evaluation of Design Decisions, *Proc. 2005 Workshop on Human and Social Factors of Software Engineering*, pp.1-7 (2005).
- 13) Zsombok, C.E. and Klein, G.A.: Naturalistic Decision Making, *Proc. Naturalistic Decision Making Conference 1994*, Mahwah, NJ, Lawrence Erlbaum (Jan. 1997).
- 14) Zannier, C. Chiasson, M. and Maurer, F.: A model of design decision making based on empirical results of interviews with software designers, *Information and Software Technology*, Vol.49, pp.637-653 (2007).

(平成 19 年 9 月 13 日受付)

(平成 20 年 1 月 8 日採録)



小林 祐介

2006 年東京学芸大学教育学部情報教育専攻卒業。2008 年同大学大学院修士課程修了。ソフトウェア設計支援，ソフトウェア開発環境の研究に従事。



榎山 淳雄 (正会員)

1985 年早稲田大学理工学部卒業。1987 年電気通信大学大学院修士課程修了。博士(工学)。日本電気株式会社勤務を経て，1999 年東京学芸大学助教授。2007 年同准教授。1999～2003 年電子情報通信学会和文論文誌 D 編集委員，2003～2007 年本学会論文誌編集委員。協調ソフトウェア開発支援，CSCW，協調学習支援等に関する研究に従事。ACM，IEEE Computer Society，電子情報通信学会，人工知能学会，教育システム情報学会各会員。