

平成 18 年度 公立はこだて未来大学卒業論文

デザインプロセスにおけるグループワーク編成支援ツールの提案  
— サブタイトル —

田中 康介

情報アーキテクチャ学科 1015013

指導教員 姜 南圭

提出日 2019 年 1 月 29 日

Suggestion of group work support tool in the design  
process  
— English Subtitle —

by

Kosuke Tanaka

BA Thesis at Future University Hakodate, 2018

Advisor: Prof. Kang

Department of Media Architecture  
Future University Hakodate  
January 29, 2019

**Abstract**— Currently, problems that designers have to deal with are considering the relationship with the users and the society. Therefore, it becomes difficult to find the best answer with the sensitivity and creativity of individual designers. As a result, it is required to organize teams by members with various skills and to conduct creative activities systematically. In the previous studies, results are gotten by distributing human resources in a well-balanced manner. But, it has not been reported what kind of influence the member composition has on creative activities. Consequently, we conducted a survey about group work in creative activities. As a result, most people chose group work rather than personal work for different reasons such as expanding their own vision. In this research, we suggest a web application, named ‘ Skill Pentagon ’ for organizing more various groups by visualizing the user’s skill.

**Keywords:** Design Process, Group work, Web application, Keyword4, Keyword5

**概要:** 現在デザイナーが対処すべき問題は、ユーザや社会との関係性をより丹念に考慮することが求められるようになり、デザイナー個人の感性や創造性だけでは最適な答えを出すことは難しくなっている. このことから、多種多様なスキルを持つメンバーによってチームを組み、組織的に創造活動を行うことが求められている. 先行研究では、人的リソースをバランスよく配分することで全体的な成果を挙げたことが報告されている. しかし、個人の創造活動がメンバー構成によってどのように変化したかは報告されていない. そのため、公立はこだて未来大学にて創造活動におけるグループワークの現状についてのアンケート調査を行った. その結果、視野が広がるなどの理由から、自分と異なるスキルを持つ人とグループワークを支持する人が大多数であるということが分かった. 本稿ではユーザのスキルを可視化することにより、より多種多様なグループ編成を行うための Web アプリケーションの提案を行う.

**キーワード:** デザインプロセス, グループワーク, ウェブアプリケーション, キーワード 4, キーワード 5

# 目次

|              |                          |          |
|--------------|--------------------------|----------|
| <b>第 1 章</b> | <b>序論</b>                | <b>1</b> |
| 1.1          | 背景 . . . . .             | 1        |
| 1.2          | 対象とする領域 . . . . .        | 1        |
| 1.3          | 研究目標 . . . . .           | 1        |
| <b>第 2 章</b> | <b>関連研究</b>              | <b>2</b> |
| 2.1          | ちんちんプログラム . . . . .      | 2        |
| 2.1.1        | Smalltalk-80 . . . . .   | 2        |
| 2.1.2        | Java 3D . . . . .        | 2        |
| 2.2          | グラフィックスシステム . . . . .    | 2        |
| 2.2.1        | DirectX . . . . .        | 2        |
| <b>第 3 章</b> | <b>プログラミング言語 FUN</b>     | <b>3</b> |
| 3.1          | 提案する言語 FUN の特徴 . . . . . | 3        |
| 3.2          | 言語仕様 . . . . .           | 3        |
| 3.3          | 実装方法 . . . . .           | 3        |
| 3.3.1        | 開発環境 . . . . .           | 3        |
| 3.3.2        | OS に対する依存性 . . . . .     | 3        |
| <b>第 4 章</b> | <b>実験と評価</b>             | <b>4</b> |
| 4.1          | 保守性に関する評価 . . . . .      | 4        |
| 4.1.1        | Fortran との比較 . . . . .   | 4        |
| 4.1.2        | Java との比較 . . . . .      | 4        |
| 4.2          | 実行速度 . . . . .           | 4        |
| 4.2.1        | Fortran との比較 . . . . .   | 4        |
| 4.2.2        | Java との比較 . . . . .      | 5        |
| 4.3          | 利用者によるアンケート . . . . .    | 5        |
| 4.3.1        | 初心者 . . . . .            | 5        |
| 4.3.2        | 上級者 . . . . .            | 5        |
| <b>第 5 章</b> | <b>考察</b>                | <b>6</b> |
| 5.1          | 評価結果 . . . . .           | 6        |
| 5.2          | 評価結果 . . . . .           | 6        |

|       |                 |   |
|-------|-----------------|---|
| 第 6 章 | 結論と今後の展開        | 7 |
| 6.1   | まとめ . . . . .   | 7 |
| 6.2   | 今後の方針 . . . . . | 7 |

# 第1章 序論

本章では、本研究における背景と、研究目的について述べる。

## 1.1 背景

現在デザイナーが対処すべき問題は、ユーザや社会との関係性をより丹念に考慮することが求められるようになり、デザイナー個人の感性や創造性だけでは最適な答えを出すことは難しくなっている。そこで個人の創造性を超えて、多角的な方面からより創造的な解を生み出すために多様な専門性を持つチームによる組織的なデザイン行為のあり方について議論されることが増えてきた。A1 石井らは創造性という観点から一般的な創造活動の認知モデルであるジェネプロアモデルの枠組みを適応し、問題解決においてアイデアを検討する段階に限定したうえで、「独立して考える場合と比較して二人で話し合うという協同には効果がある」と示した [2]。また、Kang は異なる学問や経験、文化を持っている人が集まりグループワークを行うことは、大きな創造性を生み出す潜在能力を持っている [3] としている。これらのことから、自分と異なるスキルや視点を持つ人と共に創造的な作業を行うことにより、自分だけの潜在能力では考えることができなかった成果をあげることがあると考えられる。しかし、グループワークをする際、各グループメンバーのスキルや視点を具体的に目にする機会が少ない問題が指摘されている。

## 1.2 対象とする領域

実用レベルのサイズのプログラムを作成するためのプログラミング言語について研究する。ここで、行うのは3次元グラフィックス向けの言語の設計とそのインタプリタの実装である。

## 1.3 研究目標

完全な処理系の実装を目指すものではなく、プログラミング言語にオブジェクト指向という考え方を取り入れたプログラミング言語を設計し、プロトタイプシステムを作成することにより、オブジェクト指向の概念が、プログラミングの能率向上とメンテナンス性の向上に寄与することを示す。

## 第2章 関連研究

### 2.1 ちんちんプログラム

#### 2.1.1 Smalltalk-80

Smalltalk-80 は 1982 年ごろ，当時ゼロックスにいた...

必要があれば

#### 2.1.2 Java 3D

Java はオブジェクト指向言語で，そこで 3 D グラフィックスを扱うための..

### 2.2 グラフィックスシステム

#### 2.2.1 DirectX

DirectX はマイクロソフトの Windows 上の.....

## 第3章 プログラミング言語FUN

この章では，提案する理論，仮説，モデル，アルゴリズム，方法論，実装のなどの説明を行う．

### 3.1 提案する言語FUNの特徴

この言語の特徴は，..<sup>1</sup>であり，...<sup>2</sup>という従来にない長所をもつ．

### 3.2 言語仕様

言語仕様は以下の通り．

### 3.3 実装方法

この言語は，C言語を用いて記述されている．ソースコードは20に分かれ，コードの大きさは約3000行となった．

#### 3.3.1 開発環境

この言語は，C言語を用いて記述されている．ソースコードは20に分かれ，コードの大きさは約3000行となった．

#### 3.3.2 OSに対する依存性

この言語は，C言語を用いて記述されている．ソースコードは20に分かれ，コードの大きさは約3000行となった．



## 第4章 実験と評価

### 4.1 保守性に関する評価

ここでは、FUN を用いて記述した場合とそれ以外の言語で書いた場合の比較を行なう。

#### 4.1.1 Fortran との比較

同一のゲームを Fortran と FUN で記述してみた。

##### スーパーマリオブラザーズ

一見、このプログラムは Fortran 向きと考えられるが、FUN の TAKOIKA ライブラリを用いて記述すると、非常にコンパクトになる。

##### パックマン

このプログラムはどちらの言語にとっても、有利な要素はない、このことを反映して。

#### 4.1.2 Java との比較

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

### 4.2 実行速度

#### 4.2.1 Fortran との比較

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

#### **4.2.2 Java との比較**

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

### **4.3 利用者によるアンケート**

#### **4.3.1 初心者**

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

#### **4.3.2 上級者**

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

## 第5章 考察

### 5.1 評価結果

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

### 5.2 評価結果

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

## 第6章 結論と今後の展開

### 6.1 まとめ

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

### 6.2 今後の方針

Java 言語との比較では、惨敗であり、FUN は 2 倍の記述量を必要とした。しかし、これは、Java のもつパッケージ IKURA が非常に強力であるためで、同一機能をもつライブラリを用意することにより、FUN にも同様の能力を持たせることができることが判明した。

## 謝辞

本研究において、長期にわたる評価実験に協力いただきました、株式会社〇〇の△△△△様に感謝いたします。

## 参考文献

- [1] アイン・シュタイン, 「相対性理論について」, 2000.

## 付録その1

付録その1(プログラムのソースリストなど)を必要があれば載せる

## 付録その2

付録その2(関連資料など)を必要があれば載せる



*Short Title in English*

## 図 目 次

## 表 目 次