

# 非情報系学生を対象とした ソフトウェア開発演習の 設計と継続的改善

2017/3/11

(株)日立製作所 ICT事業統括本部

居駒幹夫 高橋英男

東京大学分子細胞生物学研究所

恒川直樹

東京大学生産技術研究所

西村勝彦 平野敏行 佐藤文俊



本資料は[クリエイティブ・コモンズ 表示 4.0 国際 ライセンス](https://creativecommons.org/licenses/by/4.0/)の下に提供されています。

# 本資料について

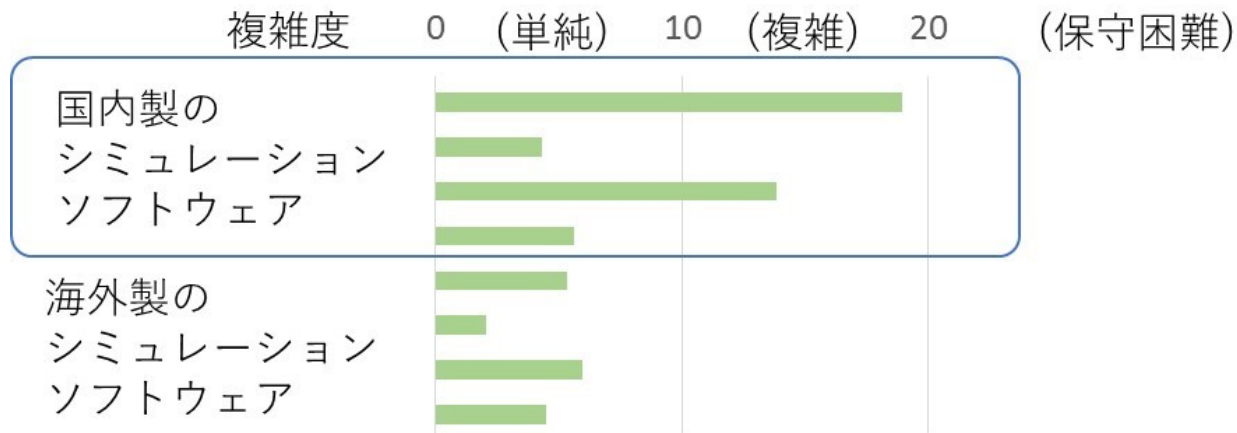
- 本資料は、一般社団法人 情報処理学会 情報処理教育委員会 情報システム教育委員会主催による第9回情報システム教育コンテスト（ISECON2016）の本審査用資料を元に再編集されたものです。
- 本資料（居駒幹夫, 高橋, 恒川直樹, 西村勝彦, 平野敏行, 佐藤文俊, 「非情報系学生を対象としたソフトウェア開発演習の設計と継続的改善」, ISECON2016, 2017.3.11）は、  
[クリエイティブ・コモンズ 表示 4.0 国際 ライセンス](#)  
の下に提供されています。

# Introduction

- 理工学系大学院生を対象とした科学技術系シミュレーションのソフトウェア開発演習
- 情報系の基礎的な教育を受けていない受講者に対し、情報システムの保守性や品質を高めるための考え方・開発プロセス・開発環境を、短期間で教えることが目標
- 東京大学のシミュレーションの専門家と(株)日立製作所のソフトウェア技術者が連携
- 2009年から8年間実施し継続的に改善中

# 背景：科学技術シミュレーションの現状

- 最古から活用されている情報システム
- 日本の現状
  - スーパーコンは，地球シミュレータ，京などで健闘しているが，ソフトウェア分野では，海外製が席捲
  - 日本製シミュレーションソフトウェアは保守性に難点

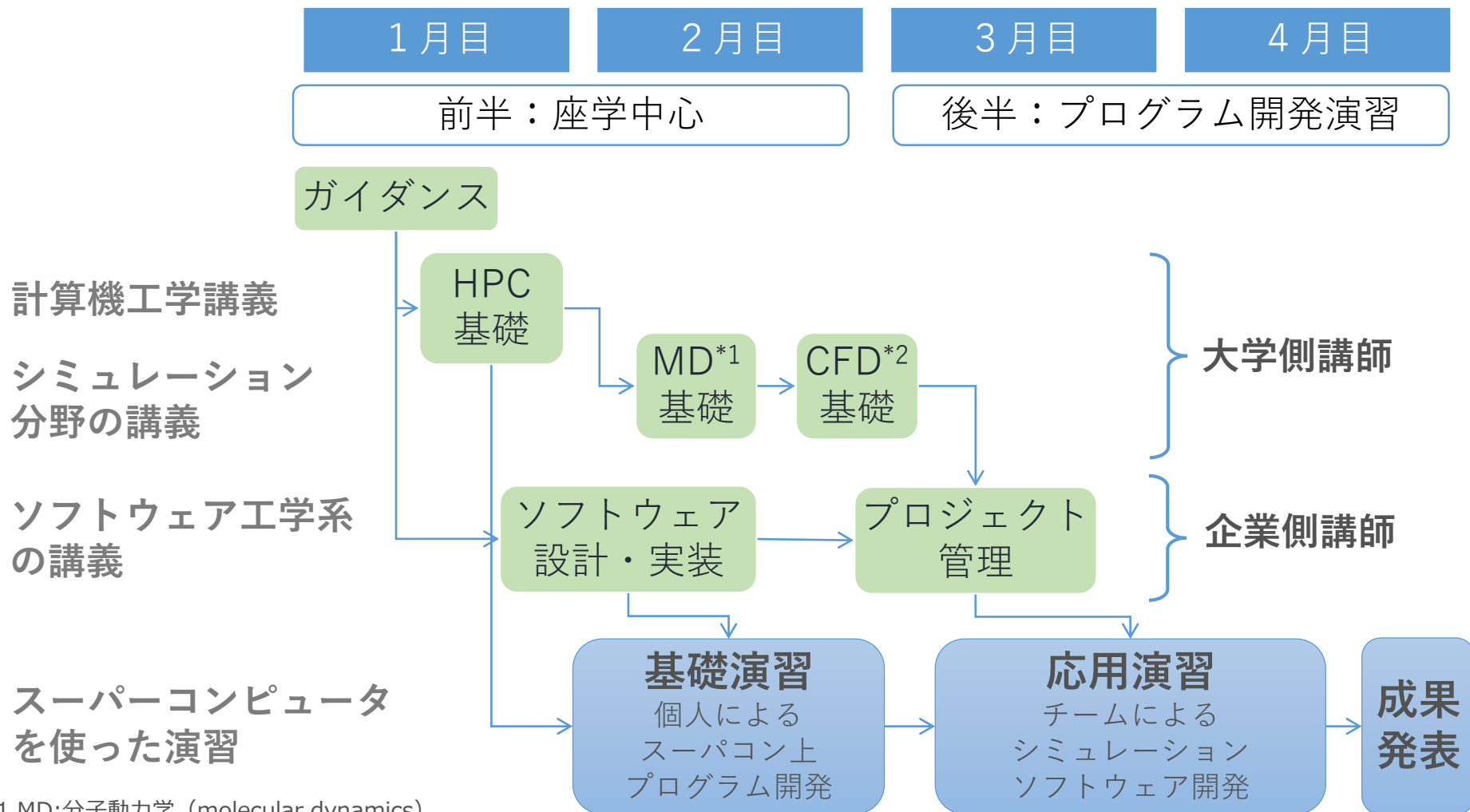


国内外シミュレーションソフトウェアの保守性(複雑度)\*1比較

\*1 McCabe. "A Complexity Measure". IEEE Transactions on Software Engineering: pp.308-320. で保守性を評価

# Syllabus

## 演習の概略スケジュール



\*1 MD:分子動力学 (molecular dynamics)

\*2 CFD:数値流体力学 (computational fluid dynamics)

# ソフトウェア開発演習の概要

	基礎演習	応用演習
開発プログラム	スーパーコン上で並列動作するプログラム	スーパーコン上で並列動作する、流体力学又は分子動力学のシミュレーションソフトウェア
開発体制	チームは組まず個人で開発	2-3名のチームで共同開発
言語, サイズ	C/C++で 200-300行	C++言語で 2000-3000行
プロジェクト管理	無し	チケットによる開発/バグ管理, Gitとの連携
機能以外の評価対象	スケーラビリティ (保守性は対象外)	信頼性, 保守性, スケーラビリティ

# Features

## 本演習の特長

- 特長 1 : 相互触発的な産学連携教育
- 特長 2 : 対象ドメインに正対した教育
- 特長 3 : 良い設計に触れる
- 特長 4 : インタネット活用のコラボレーション

## 特長 1: 相互触発的な産学連携教育

### 課題

#### 教育スコープ

大学側  
分担

企業側  
分担

漏れ

### 従来は弱点補充型の産学連携

- 産学の足りない部分を補って教育設計
- 分担を決めお互いに干渉せず（他方はブラックボックス）
- 結果として漏れの危険性

### 本演習の取組み

#### 教育スコープ

大学側  
分担

企業側  
分担

教育スコープ拡大

### 本演習は相互触発型の産学連携

- 企業・大学の双方が専門分野外にリーチ
- 情報系専門の企業側講師，科学技術シミュレーション専門の大学側講師が，専門外スキルも持った上で，受講者を指導
- 講師側が得た，新たな成果/知見を元に教育スコープを拡大，充実



## 特長 2：対象ドメインに正対した教育

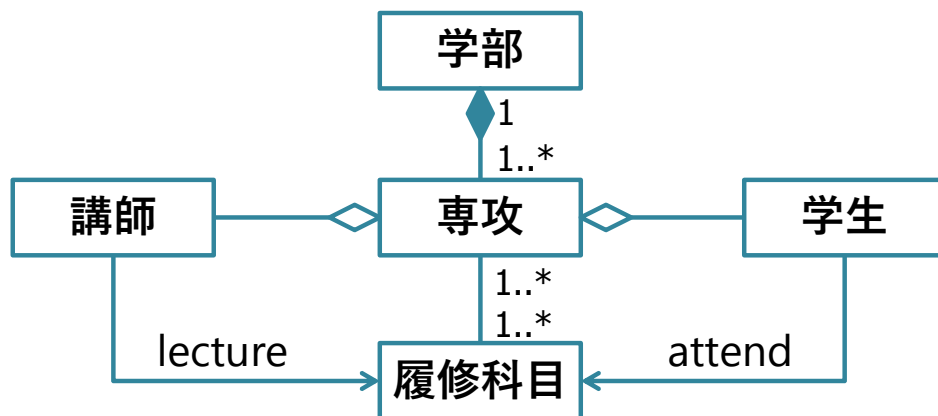
### 課題

- 科学技術シミュレーションにおける「情報システム」「ソフトウェア工学」は業務システムのそれとは異なる

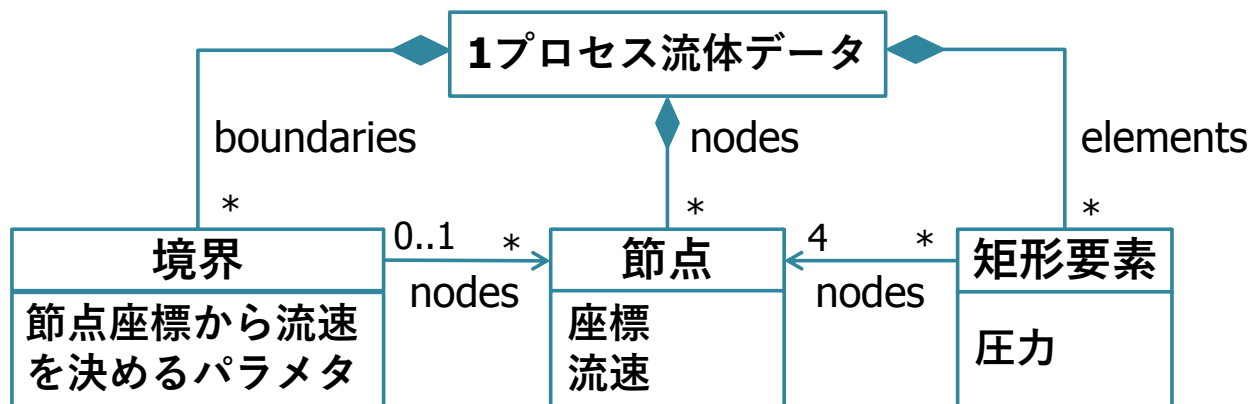
### 本演習の取組み

- 本格的な科学技術シミュレーションソフトウェアを開発
- 情報システムモデリングからテストに至るまで、科学技術シミュレーションの必要技術として教育

# UMLの教育の例



UMLがどのようなものかは理解するが対象分野のモデリングは困難



対象分野でのモデリングを通してUMLも学ぶ

# テスト網羅の例

- 当初は、全て実行文(C0)、全ての分岐(C1)が通るようにテストしましょう  
→ 科学技術系では制御フローではなく  
    インプットドメインのテスト網羅が必要
- 例：以下のメソッドをどのようにテストする？

```
double calcInvariants1(int i, int j){
    return ( 3 * alphaNy_[i] * alphaNy_[j] + betaNy_[i] * betaNy_[j] + gammaNy_[i] * gammaNy_[j]
            + 3 * alphaNx_[i] * alphaNx_[j] + betaNx_[i] * betaNx_[j] + gammaNx_[i] * gammaNx_[j]
            - ( betaxy / alphaxy ) * ( alphaNy_[i] * betaNy_[j] + alphaNy_[j] * betaNy_[i]
                                     + alphaNx_[i] * betaNx_[j] + alphaNx_[j] * betaNx_[i] )
            - ( gammaxy / alphaxy ) * ( alphaNy_[i] * gammaNy_[j] + alphaNy_[j] * gammaNy_[i]
                                     + alphaNx_[i] * gammaNx_[j] + alphaNx_[j] * gammaNx_[i] ) )
            / ( 6.0 * alphaxy * re );
}
```

UML,テスト網羅だけでなく、「コードレビュー」「WBS作成」等でも同じ。  
同様な問題は、業務ソフトウェア以外の全てのドメインにあり

## 特長 3 : 良い設計に触れる

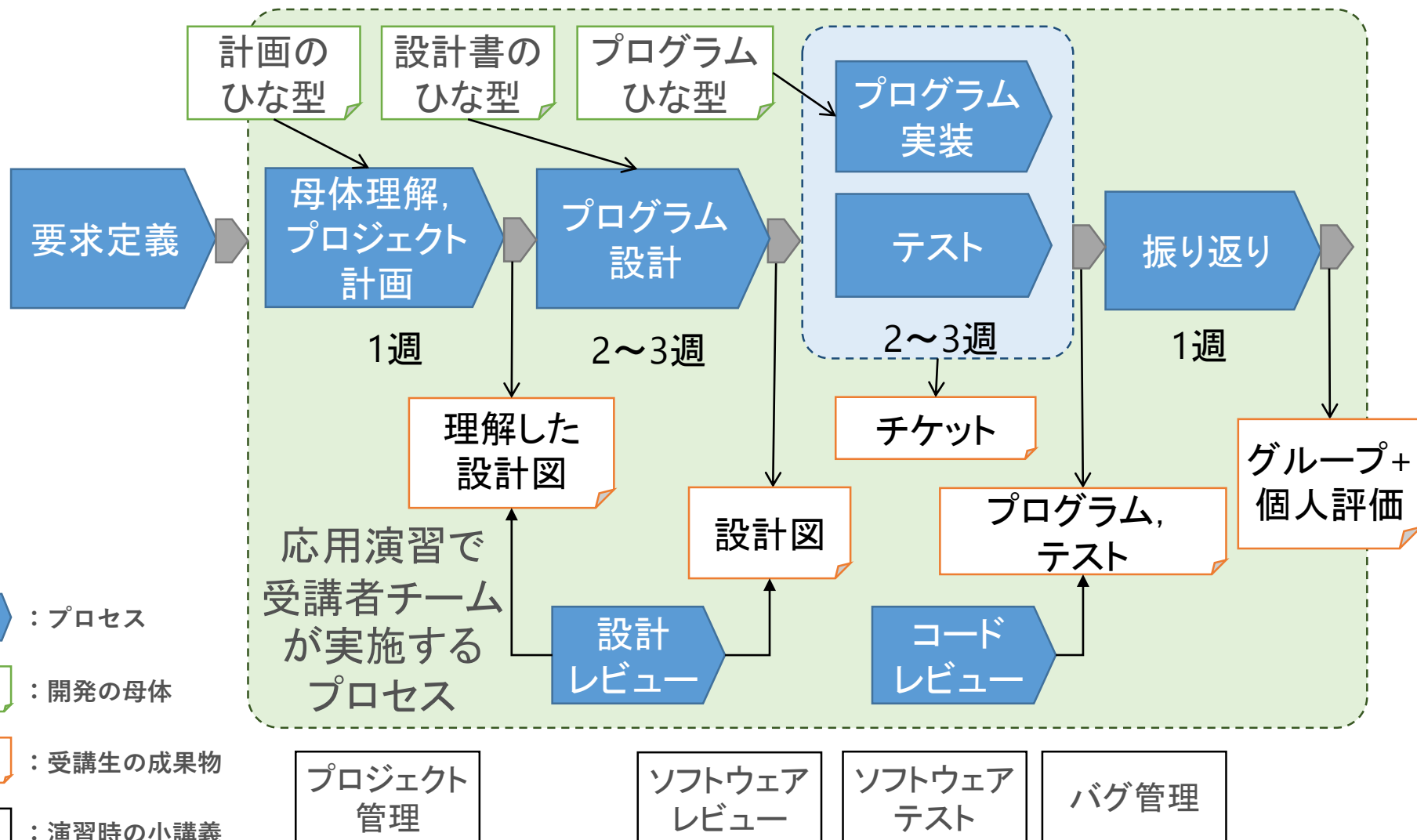
### 課題

- プログラムが成果物の演習では、「動くプログラム」作成に注力しすぎる傾向あり（初期の演習）
- 大学研究室でのプログラム開発の多くは、すでにあるプログラムを母体とした派生開発

### 本演習の取組み

- 良い設計の半成果物をモデルのレベルで理解させ、受講者が開発部分も設計図を執筆、レビュー
- ソースコードも半成果物を参考にしてプログラミング

# 良い設計に触れる演習のスケジュール



## 特長4：インターネット活用コラボレーション

### 課題

- 複数キャンパス+企業+自宅間での開発プロジェクトでの連携
- 研究室の学外公開サーバは，保守やセキュリティ確保等の課題で維持困難

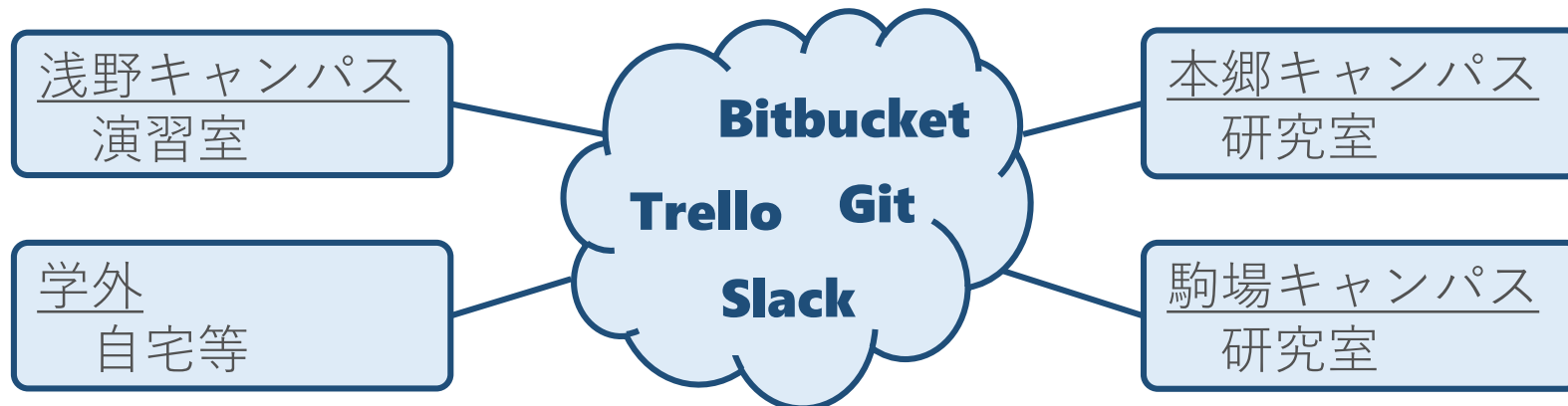
### 本演習の取組み

- フリーのインターネットサービスを活用  
Git, Bitbucket, Slack等

# 演習の開発環境

- 2015年まで
  - 駒場キャンパスの研究室に独自サーバを立て Redmine + Subversion を運用
- 2016年以降
  - 統合プロジェクト管理: Bitbucket
  - バージョン管理: Git
  - メンバ間コミュニケーション: Slack
  - チケット管理\*1: Trello

インターネット上の  
サービスを活用



\*1 チケット管理：2016年はBitbucketの課題管理を使用。2017年以降はTrelloを活用予定

(株)日立製作所 居駒幹夫 高橋英男, 東京大学 恒川直樹 西村勝彦 平野敏行 佐藤文俊,  
「非情報系学生を対象としたソフトウェア開発演習の設計と継続的改善」, ISECON2016, 2017.3.11

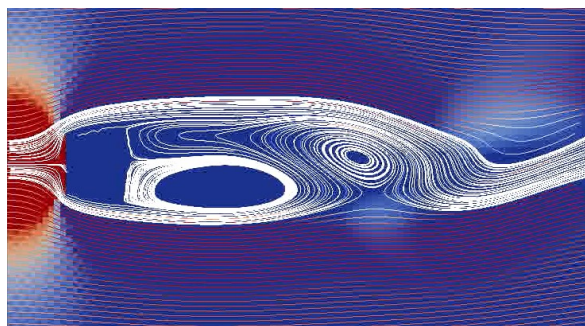
# Results

- 受講生とプロジェクト数の推移\*1

年度	'10	'11	'12	'13	'14	'15	'16
人数	11	9	6	8	13	9	12
応用演習プロジェクト数	4	3	2	3	5	3	4

\*1 2009年はトライアルとして実施したため、2010年-2016年の7回を対象

- 当初想定していた機械系の修士学生の他に、  
材料工学，応用化学，航空宇宙工学等  
シミュレーションを必要とする専攻の受講生が多く参加



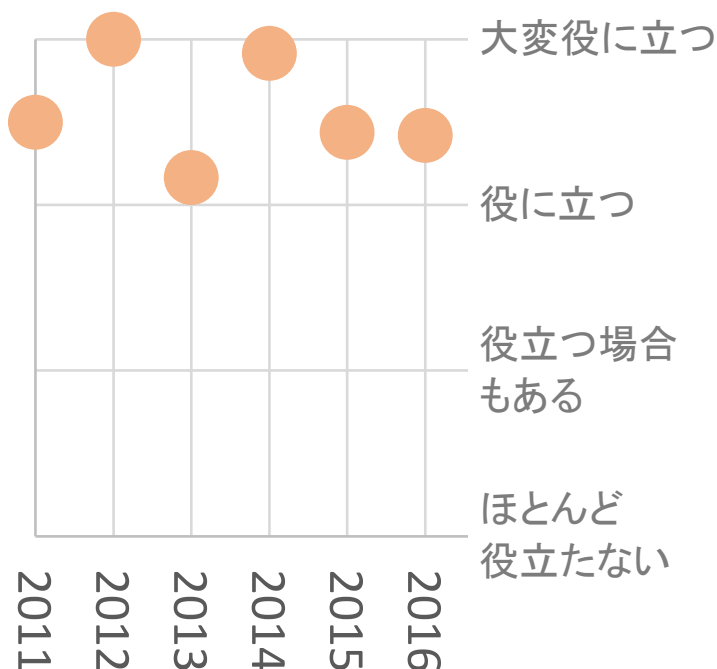
学生のコーディングした  
流体プログラムの出力例  
(角柱の下流に現れるカルマン渦)



# 学生， 成果物の評価

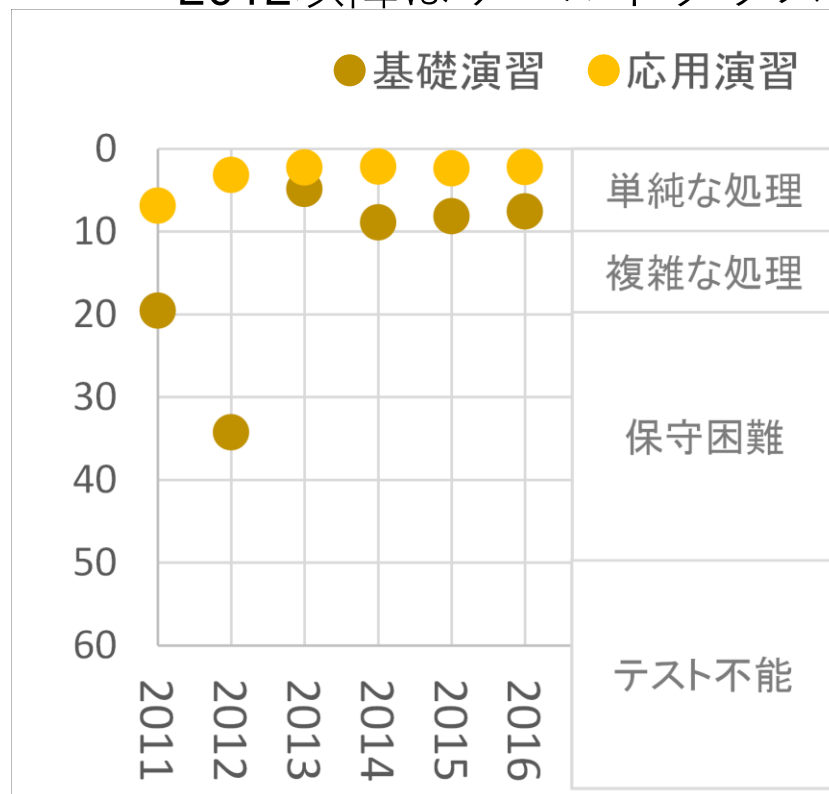
- 学生の評価

- おおむね好評価



- 成果物の保守性評価\*1

- 2012以降はワールドクラス



\*1 評価対象は受講生コーディング部分のみ

(株)日立製作所 居駒幹夫 高橋英男, 東京大学 恒川直樹 西村勝彦 平野敏行 佐藤文俊,  
「非情報系学生を対象としたソフトウェア開発演習の設計と継続的改善」, ISECON2016, 2017.3.11

# Publications

- 教科書を二冊上梓



## 「ソフトウェア開発入門」

シミュレーションソフト設計理論から  
プロジェクト管理まで

佐藤文俊, 加藤千幸編 東京大学出版会 (2014)

## 「ソフトウェア開発実践」

科学技術シミュレーションソフトの設計

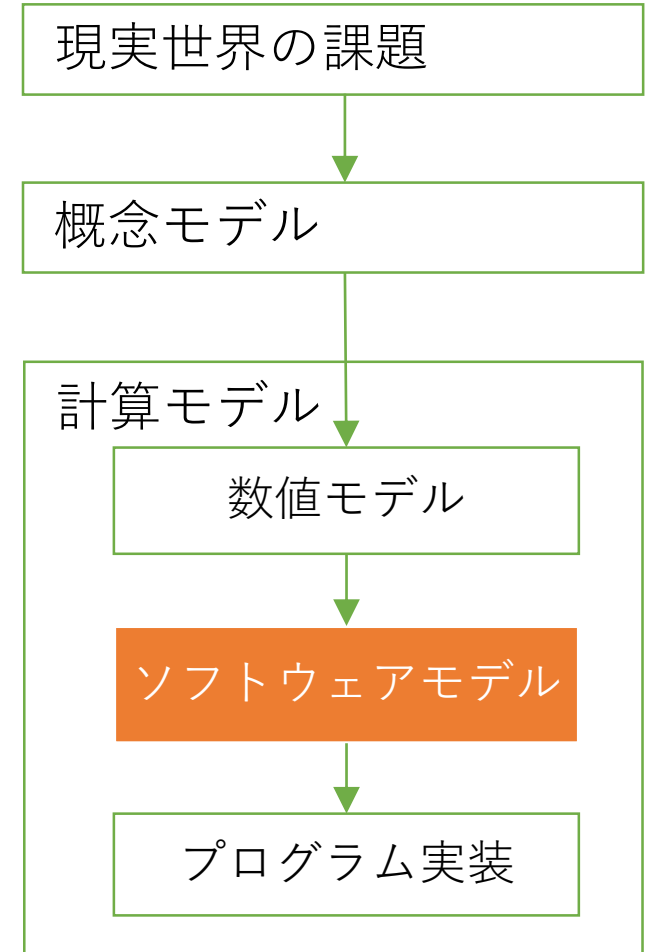
佐藤文俊, 加藤千幸編 東京大学出版会 (2015)

- 教育内容の発表/報告 (教育コンテンツも順次公開予定)

- 居駒, 佐藤他, “非情報系の学生を対象としたソフトウェア開発演習の設計と改善”, 2016, 第137回情報システムと社会環境研究発表会
- 東京大学情報基盤センター スーパーコンピューティングニュース: 2012/11, 2015/9, 2016/9に三回報告

# 本演習教科書の特長

- 従来の教科書
  - 数値モデルから，いきなり（現実には使えない）プログラムのソースコードが示されている。
- 今回の教科書
  - 数値モデルから，ソフトウェアのモデリング技術や，動作するスーパーコンのアーキテクチャ等を考慮したソフトウェアモデルを示し，その上で必要部分のソースコードを示している。



# Summary

- 科学技術シミュレーションソフトウェアの分野で初めてのソフトウェア開発演習
  - シミュレーション開発研究を専門とする大学教員と、情報システム・ソフトウェア工学が専門の企業講師による相互触発型連携
  - シミュレーション分野でのソフトウェア開発に対応した現実的な解を教育
- 受講生の評価も高く，成果物の保守性もワールドクラス
- 8年間実施し，継続的に改善
- 成果を書籍（2冊）として公開し，大学や企業で活用

# 参考文献

- 居駒幹夫, “講義紹介：実践的シミュレーションソフトウェア開発演習”, 東京大学情報基盤センター スーパーコンピューティングニュース, Vol.14, No.6, pp.27-37, 2012年11月.
- 佐藤文俊, 加藤千幸編, “ソフトウェア開発入門: シミュレーションソフト設計理論からプロジェクト管理まで”, 東大出版, 2014年4月, ISBN-4130624547
- 佐藤文俊, “教育利用講義報告「実践的シミュレーションソフトウェア開発演習」”, 東京大学情報基盤センター スーパーコンピューティングニュース, Vol.17, No.5, pp.47-49, 2015年9月.
- 佐藤文俊, 加藤千幸編, “ソフトウェア開発実践: 科学技術シミュレーションソフトの設計”, 東大出版, 2015年11月, ISBN-4130624555
- 居駒幹夫, 高橋英男, 西村勝彦, 平野敏行, 恒川直樹, 佐藤文俊, “非情報系の学生を対象としたソフトウェア開発演習の設計と改善”, 研究報告 情報システムと社会環境, 2016-IS-137, No.2, pp.1-8, 2016年8月.
- 佐藤文俊, 居駒幹夫, “2016年度「実践的シミュレーションソフトウェア開発演習」”, 東京大学情報基盤センター スーパーコンピューティングニュース, Vol.18, No.5, pp.34-35, 2016年9月.