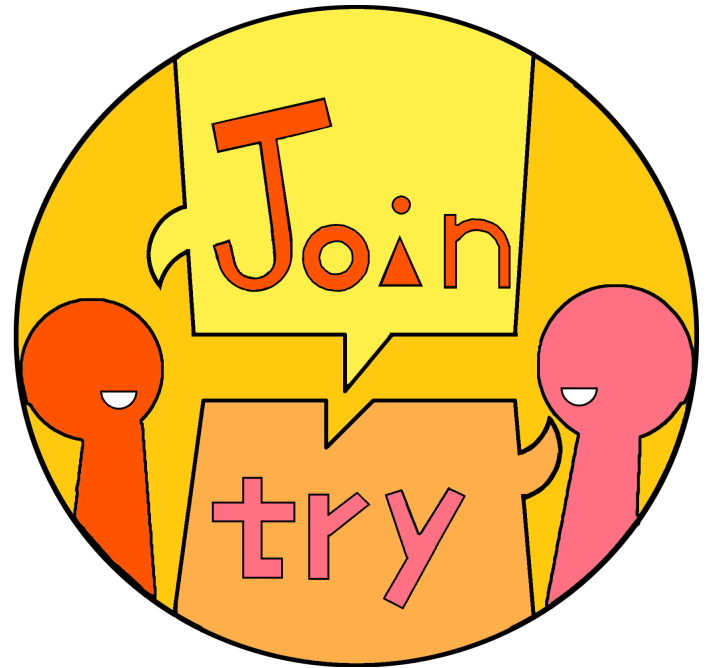

前期PBL成果発表会

(2013-08-03)

子ども向け
プログラミング学習ツール
「Jointry」



秋口PBL(角、金本)

指導教員:秋口教授

副指導教員:加藤教授、土屋助教、長尾助教

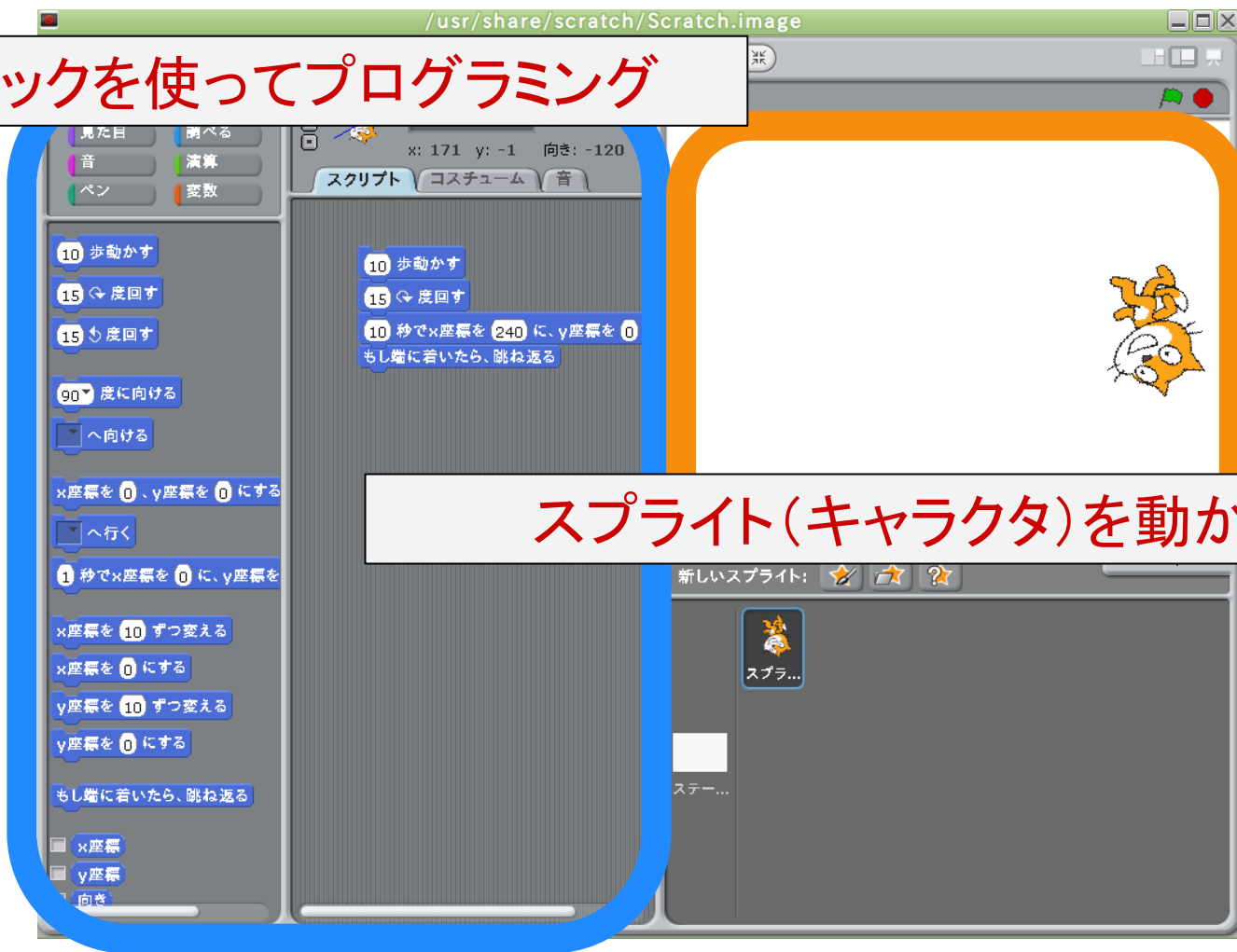
Jointry (ジョイントリ) の概要

- 「Jointry」は、**小学校高学年向けのプログラミング学習ツール**であり、キーボードを使わずにプログラミングの学習を可能とするソフトウェアである。
 - 既存のプログラミング学習ツール (Scratch等) とは違い、**友達や先生と一緒に学習できる共同プログラミング機能**が備わっている。
-

プロダクトイメージ(Scratchの例)

ブロックを使ってプログラミング

スプライト(キャラクタ)を動かす





「Jointry」の由来

- ブロックを接続する(**joint**)
 - プログラミングを一緒にやる(**join**)
 - プログラミングに挑戦してみる(**try**)
-

事前調査

秋口PBLのテーマ(および制約)

- アプリケーションを開発するための**基盤となるソフトウェア**を開発する
 - 一昨年のPBL → UMLエディタ
 - 開発を「支援する」ソフトウェアも含まれる
 - 第一言語として**Java**を使用する
 - これまでのPBLの成果(ライブラリ)を再利用する
 - メンバーがJavaを日常的に使っていない
 - 学習目的を兼ねている
-

市場の動向(2013年4～7月)

「子ども プログラミング」の検索結果

- ベネッセ、小中学生向けの**プログラミング・ワークショップ**開催、新たな学びの場として定期化へ
 - 子ども向け**プログラミング教育のイベント**
 - 【レポート】**ブロック遊びのようなプログラミング言語**でカスタムできる"手書き特化"タブレット端末「enchantMOON」- 端末名の由来も紹介
 - Light-Botは**プログラミングの概念(if-thenなど)を使ってプレイするゲーム**
 - イギリスの課外**プログラミング学習支援**団体Code ClubがCodecademyらの支援
 - プログラマブルロボットmOwayduino (Arduinoを使用)はPython, Java, Scratchでプログラミングできる
 - ミニロボ操りサッカーゲーム **児童ら仕組みなど学ぶ**
 - 絵だけでプログラムが作れる**「ビスケツ塾」幼児コース**のモニター募集
 - **子どものためのプログラミング言語Scratch**がブラウザから使えるようになった
 - プログラミングをミニ四駆のような、**子どもから大人まで楽しめる**ホビーにしたい
 - CodeNow、少数派グループの**高校生向けプログラミング教室**をニューヨークでも展開
 - 30台のRaspberry Piがカメルーンの田舎の**中学校でコンピュータ教育**に使われる
-

子ども向けプログラミング講座

1. CoderDojo Tokyo
2. Tech Kids CAMP
3. 東大サマーキャンプ2013プログラミング講座
4. 夏休みゲームプログラミング教室
5. 小学生からはじめるわくわくプログラミング

「夏休みに参加できる！子供向けプログラミング講座まとめ」より
<http://techacademy.jp/magazine/770>

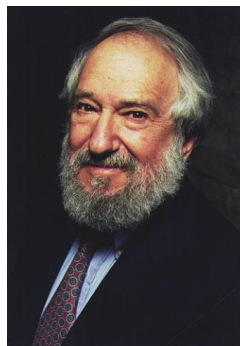
小学生からはじめるわくわくプログラミング (7/25発売)



[Amazon.co.jp](https://www.amazon.co.jp)「コンピュータ・IT > プログラミング」カテゴリ1位

ビジュアルプログラミング言語の歴史

- AMBIT ('64)
- シーモア・パパート
 - LOGO ('67)
- アラン・ケイ
 - Smalltalk ('70s~)
 - 論文「すべての年齢の子どもたちのためのパーソナルコンピュータ」('72)
- ビル・アトキンソン
 - Apple HyperCard ('87)
- ミッチェル・レズニック
 - Scratch ('03)



(普通の)子どもがプログラミングを しない理由(仮説)

3. 継続の課題

2. 能力の課題

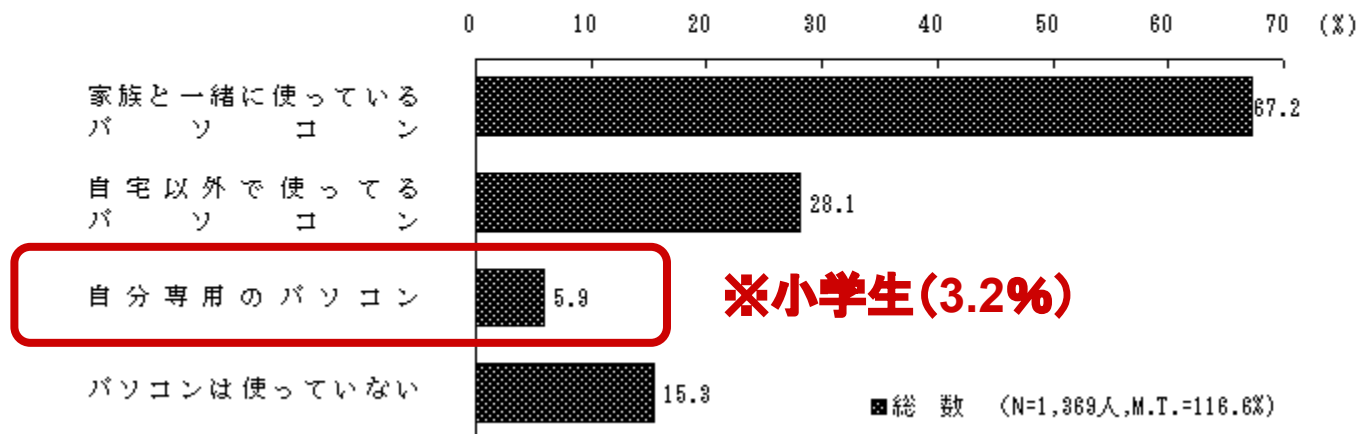
1. 環境の課題

1. 環境の課題

- 自由に使えるパソコンを持っていない
 - ⇒ キーボード操作の習得が難しいのではないかな？

図表Ⅱ-1-2-1 使用状況

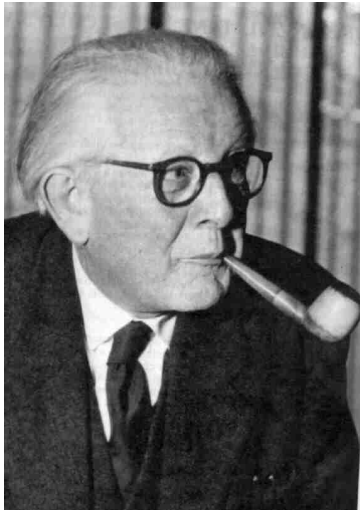
Q10 〔回答票6〕あなたは、パソコンを使っていますか。この中から、あてはまるものをいくつかあげてください。(M.A.)



「青少年のインターネット利用環境実態調査」
(内閣府、平成22年3月)

2. 能力の課題

- プログラミングそのものが難しい
 - ⇒ **抽象的な思考**ができないのではないか？



ジャン・ピアジェ
(発達心理学者)

形式的操作期（11～12歳以降）
になると、抽象的な概念を「シンボル」に結びつけて論理的に扱えるようになる。仮説や演繹的思考ができる。

3. 継続の課題

- 1人でやっていると飽きてしまう
 - ⇒ **フィードバック**が必要ではないか？



既存のプロダクトによる解決策

1. 環境の課題

- a. マウスだけで操作可能にする

2. 能力の課題

- a. ビジュアル(ブロック)プログラミング
- b. ゲームを使ったプログラミング

3. 継続の課題

- a. 成果物をインターネットで公開
 - b. 勉強会やワークショップの開催
-

(参考)調査した既存のプロダクト

1. Scratch
2. Blockly
3. VISCUIT
4. アルゴロジック
5. プログラミン
6. 前田ブロック
7. PenCode
8. App Inventor
9. Snap!

詳細は「秋口PBL2013開発計画書」および
<http://ja.wikipedia.org/wiki/ビジュアルプログラミング言語>

開発計画

本PBLプロジェクトの目的

1. プロダクト(WHAT)の目的

子どもたちが**習得しやすいプログラミング学習環境**を開発する。

- **指標**: 利用者の声を集める(定性的指標)

2. プロセス(HOW)の目的

ユーザーからの**フィードバックを継続的に反映**することで、ユーザー視点に立ったソフトウェアの企画および開発を経験する。

- **指標**: リリース回数とそこに含まれる機能数を記録する(定量的指標)
-

3つの課題に対する解決策案

1. 環境の課題

タッチ操作でプログラミングできる。

2. 能力の課題

目の前のオブジェクトを動かせる。お絵かき機能やカメラ機能を使って編集できる。

3. 継続の課題

ペアプログラミング機能で**誰かと一緒にプログラミングを学習**できる。

想定するユーザー

- 小学高学年(4・5・6年生)を対象にする
 - "研究の結果(中略)8～10歳ぐらいになると、予測し、計画をたてる能力が徐々に発達"
 - アラン・ケイ『アラン・ケイ』鶴岡雄二[翻訳]、浜野保樹[監修]アスキー出版社

3年生		4年生	5年生	6年生
8歳	9歳	10歳	11歳	12歳

- 上記を対象にプログラミングを教える講師
-

※プロジェクトで扱わないこと

- **子どもがプログラミングをやることの是非**
 - 「プログラミングはいいものだ」と仮定する
 - **高度なグラフィックデザイン**
 - デザインの重要性は理解しているが、メンバーの人数を考えて機能の実装を優先する
 - **特定のOSに特化した実装**
 - Javaが動くプラットフォームを対象にする
-

これまでのスケジュール(1Q~2Q)

- 1Q
 - テーマの選定と開発計画の作成
 - 技術調査(主にJavaFXの機能)
- 2Q
 - 最小限の機能を含むデモの開発
 - 簡単なブロック操作とプログラム実行

プロジェクトの進め方

- **年間と四半期**の大まかな計画を立案
 - 必要と思われるタスクをチケットに列挙
 - 手のあいた人がタスクにサインアップ
 - 週2回のコアミーティング(水曜日と土曜日)
 - 土曜日に進捗報告
 - 1. 今週やったこと
 - 2. 来週やること
 - 3. 今後の見通しと問題点
 - 議事録 → BacklogのWikiに登録
-

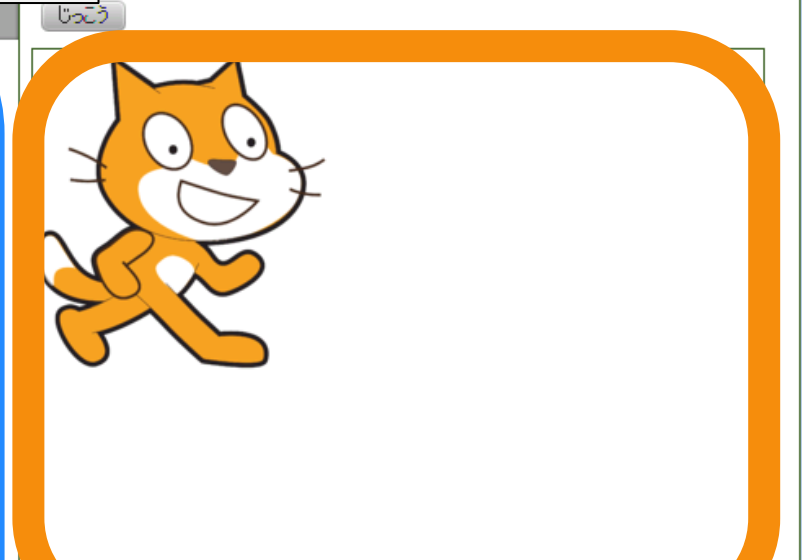
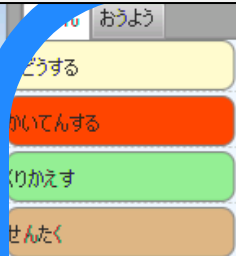
プロジェクトの成果物とメトリクス

- 開発計画書: 17 ページ
 - チケット: 46個
 - 例:
 - デモ用のブランチを作成する (S: 10分)
 - スプライトの切り替え処理 (M: 8時間)
 - ブロックの結合 (L: 34時間)
 - ソースコード
 - 行数: 4,125 [`wc -l src/**/*\.(java|fxml)`]
 - クラス数: 55 [`find src -name "**.java" | wc -l`]
-

デモと技術的な説明

デモ版のスクリーンショット

ブロックを使ってプログラミング



スプライト(キャラクタ)を動かす

あたらしいスプライト

使用技術

1. Java 7

- a. PBLの資産を活用するため
- b. Java 8 は時期尚早 (NetBeansとの相性が悪い)

2. JavaFX 2.2

- a. Java 7標準のGUIツールキット
- b. タブレット端末での利用を想定

OracleがiOS と Android用のオープンソースJavaFXを発表
<http://www.infoq.com/jp/news/2013/02/javafx-ios-android>

3. NetBeans 7.3

- a. JavaFXを最初からサポートしていた
 - b. Checkstyle + PMD + FindBugs でコーディング規約
-

JavaFXの概要

- AWT/Swingの後継となるGUIツールキット
 - 2011年10月に2.0リリース
 - バージョン1.0系とは「別物」
 - 日本語の情報がほとんどない
 - FXMLによる自動バインディング
-

FXMLによる自動バインディング

```
<!-- view.fxml -->  
<Pane fx:controller="MyController" ...  
    <Label fx:id="message" />  
    :
```

```
// MyController.java  
public Class MyController implements Initializer {  
    @FXML  
    private Label message;  
    :
```

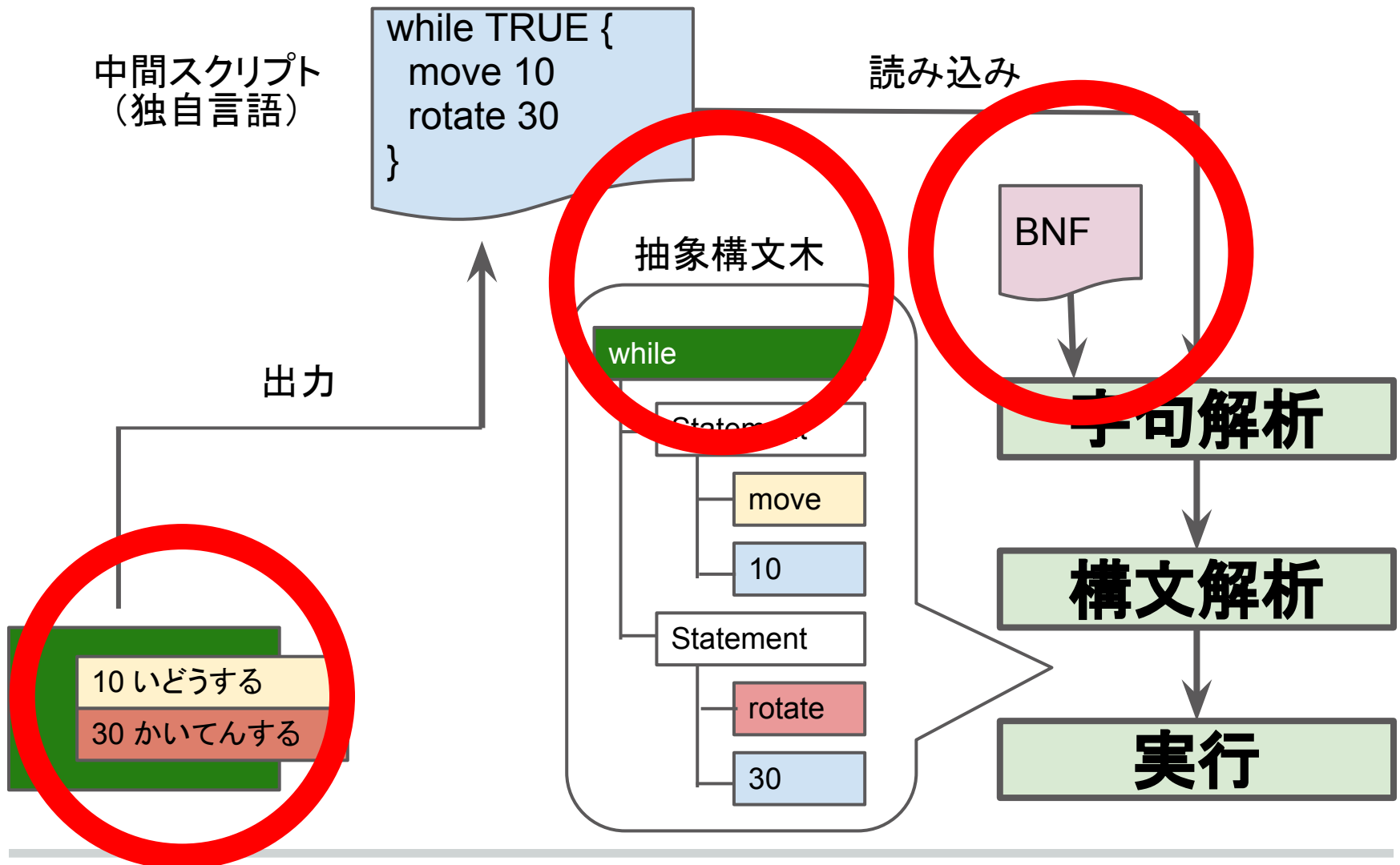


(狭義の)ViewとControllerの分離が可能

主な機能の裏側

1. ブロックと言語処理
2. お絵かき機能

1. ブロックと言語処理



新しいブロックを追加する場合

1. ブロックを作成する

- a. 形状、色、テキストフィールドの有無など

2. BNFの要素を追加する

- a. ブロックと言語要素を対応づける

3. 抽象構文木のノードを追加する

- a. 言語要素の処理内容を eval メソッドに追加する
-

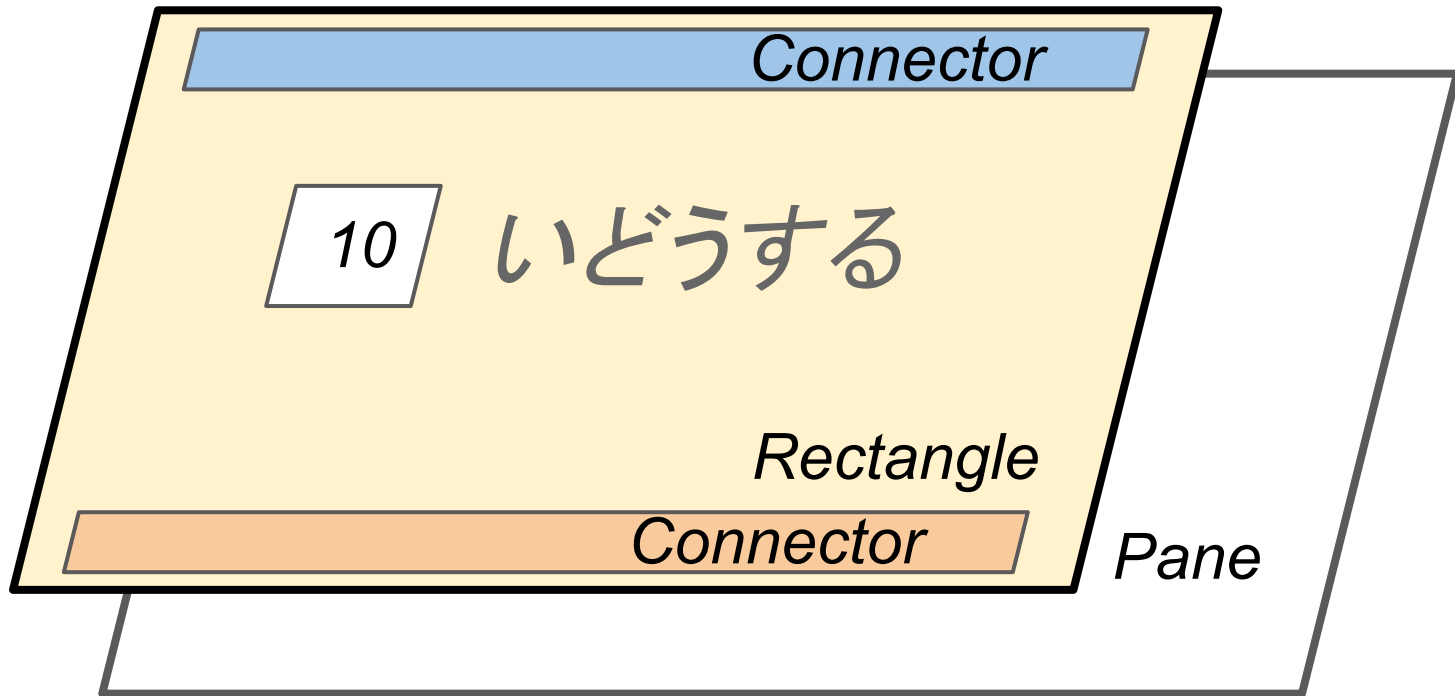
デモ版のBNF定義

```
primary      : ( "[" [ elements ] "]" | "(" expr ")"  
              | NUMBER | IDENTIFIER | STRING ) { postfix }  
factor       : "-" primary | primary  
expr         : factor { OP factor }  
block        : "{" [ statement ] { ( ";" | EOF ) [ statement ] } "  
simple        : expr [ args ]  
statement    : "if" expr block [ "else" block ]  
              | "while" expr block  
              | "rotate" expr  
              | "move" expr  
              | "costume" expr  
              | simple  
program      : [ defclass | def | statemnet ] ( ";" | EOL )
```

独自のステートメントを定義していくことで、
ビジュアルプログラミングらしい構文が作成できる

ブロック同士の接続

Connector 同士の当たり判定で接続処理



2. お絵かき機能

- PBLの既存アプリを書き直したもの
 - Swing → JavaFX
 - 置き換えるときのポイント
 - 描画レイアウト
 - 描画コンテキストの取得
 - 図形描画
 - ドキュメントが揃っていればJavaFXの方が作りやすい
-

- 描画レイアウト

- swing
 - swing.JComponent
- JavaFX
 - Canvas

- 描画コンテキストの取得

- swing
 - JComponent#paintComponentのオーバーライド
 - BufferedImage#createGraphics()
- JavaFX
 - Canvas#getGraphicsContext2D()

- 図形描画

- swing
 - Graphics#drawRect(), Graphics2D#drawRect()
 - JavaFX
 - GraphicsContext#fillRect(), #drawImage()
-

今後の予定

3つの課題に対する解決策案

1. 環境の課題

タッチ操作でプログラミングできる。

2. 能力の課題

目の前のオブジェクトを動かせる。お絵かき機能やカメラ機能を使って編集できる。

3. 継続の課題

ペアプログラミング機能で**誰かと一緒にプログラミングを学習**できる。

これからのスケジュール(3Q～4Q)

- 夏休み
 - デモのブラッシュアップ
 - タブレット端末の入手と検証
 - 3Q
 - 「タッチ操作」と「ペアプロ機能」の実装
 - ユーザーインタビューの実施
 - 4Q
 - フィードバックを受けて引き続き開発
-

考えられるリスク(と暫定的な対応)

- インタビューできるユーザー(小学校高学年の子どもたち)が見つからない
 - ご連絡お待ちしております！
 - pbl2013akiguchi@aiit.ac.jp
 - PBLの既存ライブラリが使えない
 - 既存のライブラリはSwingで作られている
 - が、頑張って対応する.....。
 - JavaFXがiOS/Androidに対応しない
 - **すでに確定済(!!)**
 - Windows 8のタブレット端末を使用予定
-

可能であればやりたいこと

- デザインの洗練
 - ワークショップの開催
 - Javaが得意な人からパッチをもらう
-

Welcome to your PR!!

(Using MIT LICENSE)

<https://github.com/jointry/jointry>

