

# Software Design

## 3-1. Modeling of Control and Data Flow

Natsuko Noda  
nnoda@shibaura-it.ac.jp

Copyright© Natsuko NODA, 2014-2023

1

# ソフトウェア設計論

## 3-1. 制御やデータの流れのモデリング

野田 夏子  
nnoda@shibaura-it.ac.jp

Copyright© Natsuko NODA, 2014-2023

2

## This section's topic

- Modeling of control and data flow
  - Activity diagram

Today's lesson is constructed by two parts.  
This section is the Part I.

Copyright© Natsuko NODA, 2014-2023

3

## 本セッションのお題

- 制御やコントロールの流れのモデリング
  - アクティビティ図

本日の授業は2つの部分から構成されます。  
こちらがパートIです。

Copyright© Natsuko NODA, 2014-2023

4

# Activity diagram

Copyright© Natsuko NODA, 2014-2023

5

## Activity and action

- To process information or data, we need fixed steps or the order to execute that processing.
- Activity diagrams show such flow of processing.
- Activity: Behavior of information systems that is described by a sequence of actions
- Action: Lower level behavior

The relation between activities and actions is relative.  
An action is relatively lower and more primitive than an activity.

Copyright© Natsuko NODA, 2014-2023

7

# アクティビティ図

Copyright© Natsuko NODA, 2014-2023

6

## アクティビティとアクション

- 情報やデータを処理するためには、それを実行するための決まった手順や順序がある
- アクティビティ図は、このような処理の流れ(順序)を示すものである
- アクティビティ：情報システムの振舞いをアクションの系列として記述したもの
- アクション：より下位の振舞い

アクティビティとアクションの関係は、あくまでも相対的なものである。  
アクションは、アクティビティよりも相対的に言ってより下位、あるいはより基本的なものを示す

Copyright© Natsuko NODA, 2014-2023

8

cf. SE-J, 2.2.3

## Activity diagram

- The purpose of an activity diagrams is to describe one activity.
  - An activity diagram shows how what kinds of actions collaborate to realize the activity.
- An activity may be
  - a work (task) in the real world.
  - a process executed on a computer.
- Similar to flowchart, but the principal difference is that an activity diagram supports parallel behavior.

Copyright© Natsuko NODA, 2014-2023

9

## Example

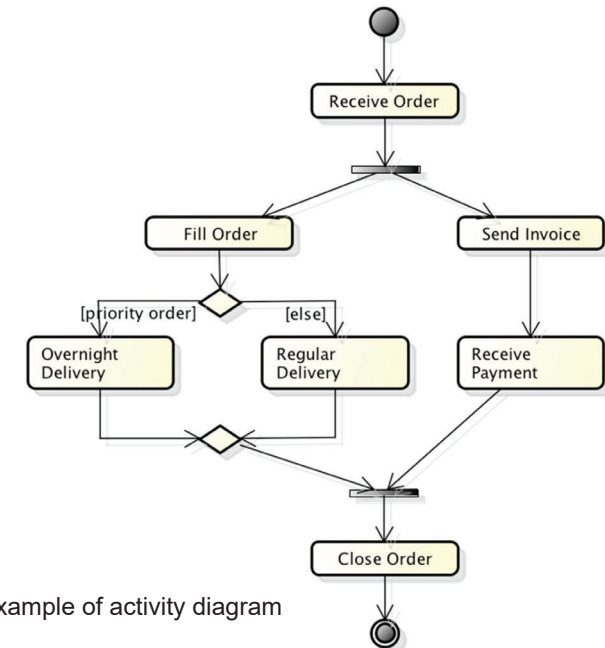


Figure 1. Example of activity diagram

Copyright© Natsuko NODA, 2014-2023

11

## アクティビティ図

- アクティビティ図の目的は、アクティビティを記述すること
  - つまり、どのようなアクションが協調してそのアクティビティを実現するのかを示す
- アクティビティの内容は様々
  - 現実世界の仕事、業務
  - コンピュータ上の処理
- フローチャートに類似している。しかし重要な違いは、アクティビティ図は並行な振る舞いを表現できること

Copyright© Natsuko NODA, 2014-2023

10

## 例

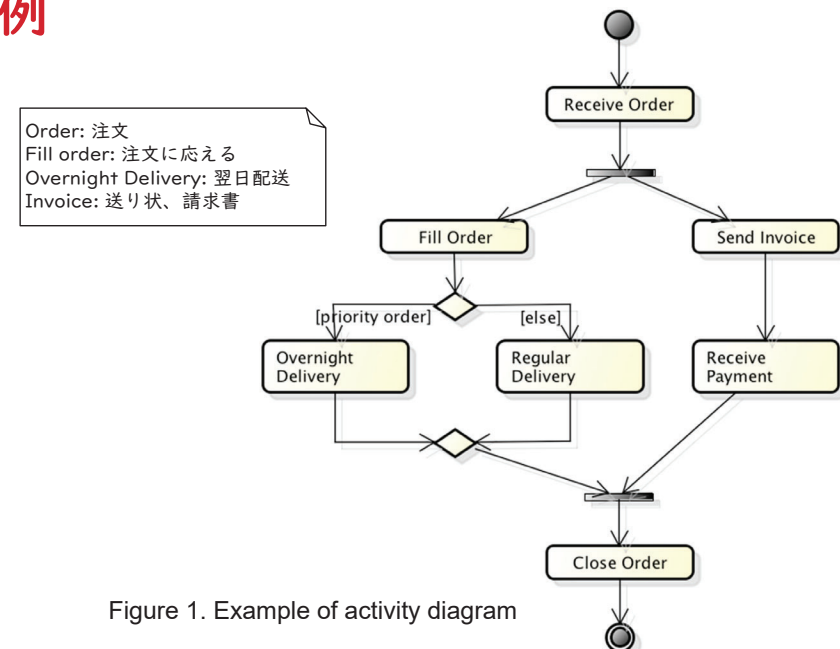


Figure 1. Example of activity diagram

Copyright© Natsuko NODA, 2014-2023

12

## Basic notations of activity diagram

Here, basic and often used notations are explained. More advanced notations are introduced in the supplementary material.

Copyright© Natsuko NODA, 2014-2023

13

## アクティビティ図の基本的な記法

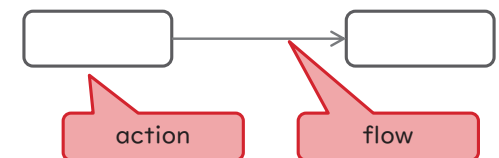
ここでは基本的、かつよく使われる記法のみを説明する。さらに発展的な記法に関しては補足資料で紹介。

Copyright© Natsuko NODA, 2014-2023

14

## Basic elements

- Action
  - one step in an activity; node of the diagram
  - Notation: rounded rectangle
- Flow
  - flow between actions; connects actions
  - Notation: arrow

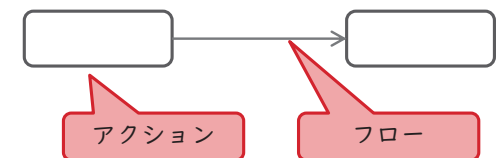


Copyright© Natsuko NODA, 2014-2023

15

## 基本要素

- アクション
  - アクティビティの1ステップ。ダイアグラムのノード
  - 記法：角丸長方形
- フロー
  - アクション間の処理の流れ。アクションを接続
  - 記法：矢印



Copyright© Natsuko NODA, 2014-2023

16

## Two types of flow

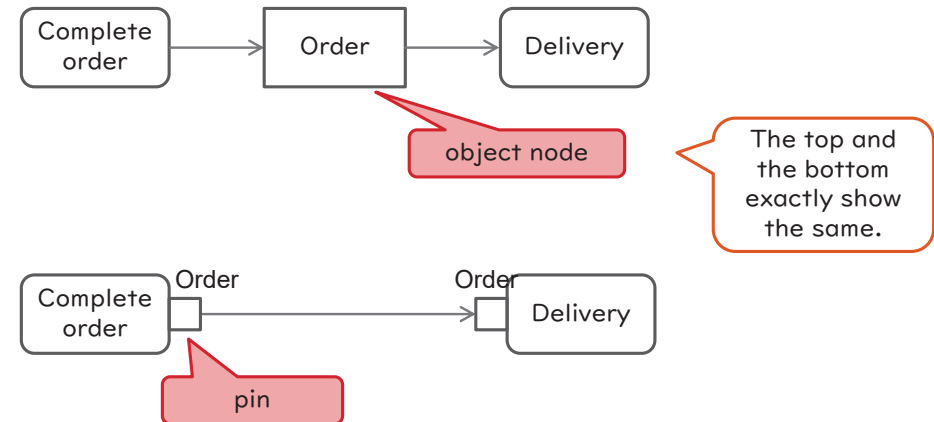
- In activity diagrams, flows may show one of the following two types.
- Control flow
  - Flow of control
  - It shows a sequence in which actions are executed; actions are executed in the order shown by the flows.
- Data flow
  - Flow of data
  - It shows a sequence in which data are transformed; an action transforms data of input flow (data attached to the input flow) to data of output flow.

Copyright© Natsuko NODA, 2014-2023

17

## Detailed notation of data flow

- If data should be clearly shown,
  - there are two types of detailed descriptions



Copyright© Natsuko NODA, 2014-2023

19

## 2種類のフロー

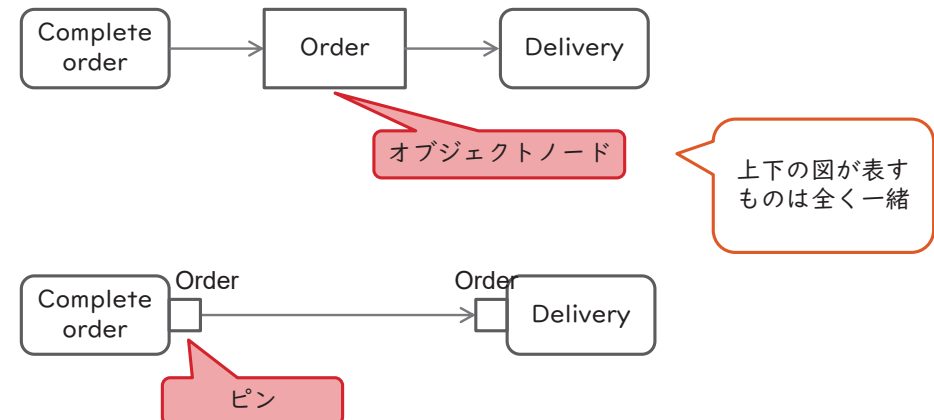
- アクティビティ図のフローが表すのは、次の2種類のどちらか
- 制御フロー
  - 制御の流れ
  - アクションが実行される系列を表す。つまり、アクションはフローが示す順番に実行されていく
- データフロー
  - データの流れ
  - データが変換される系列を示す。アクションは入力フローのデータを入力フローへと変換すると見なされる

Copyright© Natsuko NODA, 2014-2023

18

## データフローの詳細記述

- もしデータフローで流れるデータを明示的に表したければ...
- 2種類の記法がある



Copyright© Natsuko NODA, 2014-2023

20

## Start and end of the flow

- Initial node

- shows the start of the flow



- Activity final node

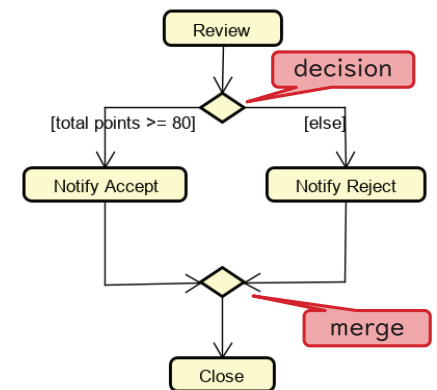
- shows the end of the flow
  - stops all flows in an activity.
  - an activity may have more than one activity final node.



## Conditional behavior

- Conditional behavior is shown by "Decision" and "Merge" nodes.

- Both are shown by "diamond" notation.
- Difference is the number of flows. (Ref. next page)



## フローの開始と終了

- 開始ノード

- フローの始まりを示す



- アクティビティ終了ノード

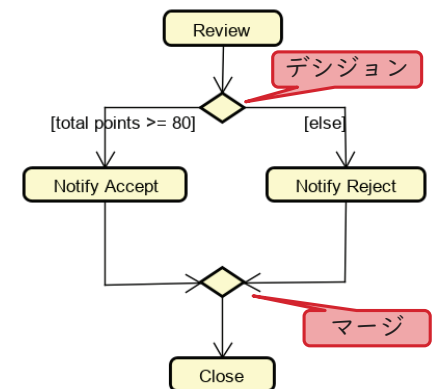
- フローの終わりを示す
  - このノードによりアクティビティ中の全てのフローが終了する
  - アクティビティはひとつ以上のアクティビティ終了ノードを持つことができる



## 条件付きの振舞い (条件分岐)

- 条件により変わる振舞いは、デシジョンノード、マージノードにより表される

- 両方とも、ひし形の記号で表現
- 接続するフローの数が違う (次ページ参照)



## Conditional behavior (Cont.)

### • Decision

- has a single incoming flow and several guarded outbound flows.
  - each outbound flow has a guard (Boolean condition).
- when a decision is reached, only one outbound flow can be taken, according to the guard condition.

### • Merge

- has multiple input flows and a single output flow.
- marks the end of conditional behavior started by the decision.

Copyright© Natsuko NODA, 2014-2023

25

## 条件付きの振舞い (Cont.)

### • デシジョン

- 入力フローはひとつ、出力フローは複数
  - 出力フローのそれぞれには条件(ブール値を取る条件、つまり真偽が決まる条件)がつく
- デシジョンに達すると、条件に従って出ていくフローのうちのただひとつが選ばれる(実行される)

### • マージ

- 入力フローは複数、出力フローはただひとつ
- デシジョンノードで開始された条件付きの振舞いがここで終わることを示す

Copyright© Natsuko NODA, 2014-2023

26

## Parallel behavior

### • Parallel behavior is shown by "Fork" and "Join".

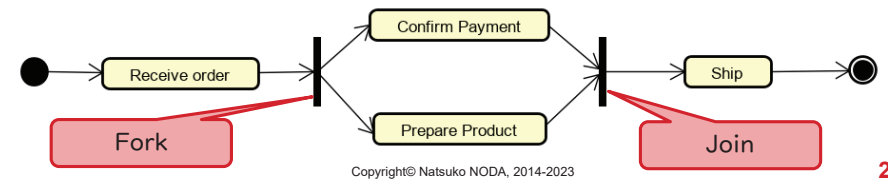
- Both are shown by bold black line.

### • Fork

- splits a flow into multiple concurrent flows.

### • Join

- synchronizes multiple flows.
  - the outgoing flow is taken only when all the incoming flows reach the Join.



27

## 並行な振舞い

### • 並行な振舞いは、フォークとジョインで表現

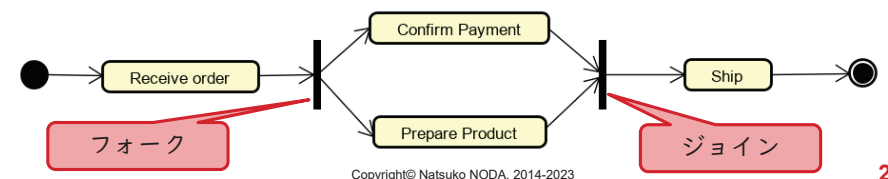
- 両方とも太い黒線で表現

### • フォーク

- フローを複数の並行な複数のフローに分割

### • ジョイン

- 複数のフローを同期。分割したものを合流させる
  - すべての入ってくるフローがジョインに到達した時に、ジョインから出ていくフローが実行される

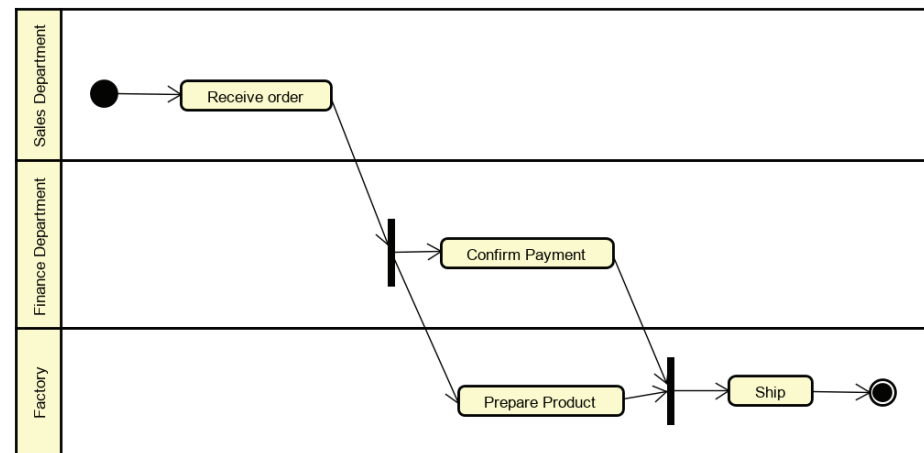


28

## Partition

- An activity diagram shows the sequence of actions.
  - Who does the action is not shown.
- If you want to describe "who" does what, use partition.
  - Divide a diagram into partitions, which shows which actions are carried out by whom.
- To show partitions, use "swim lane" (ref. next page).

## Partition with swim lane

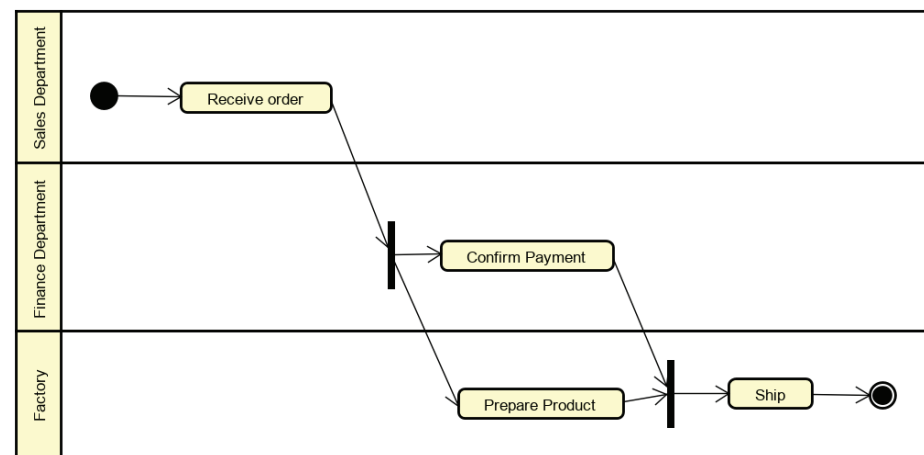


Notice: Swim lanes can be placed horizontally or vertically.

## パーティション (分割)

- アクティビティ図はアクションの系列を示す
  - 誰がそのアクションをするかは示されない
- 「誰が」何をするかまで示したい場合には、パーティション(分割)を使うことができる
  - アクションを行う実体ごとに図をパーティションに分割
- パーティションを示すために、「スイムレーン」を利用することができる(次ページ参照)

## スイムレーンを利用したパーティション



注: スイムレーンは水平方向に配置しても良いし、垂直方向に配置しても良い



## For your review

1. What does an activity diagram depict?
2. Which statement is true?
  - a. Activity diagrams are used only for expression of program logic.
  - b. Activity diagrams are used only for business modeling.
  - c. Activity diagrams are used only for workflow design.
  - d. Activity diagrams can be used for describing and designing various flows in various phases in software development.

## 復習のために

- (この前の2ページは理解の確認をするための復習問題です。簡単な英語で表現されていますので、日本語訳は省略します。英語の理解にもトライしてみましょう。)

## For your review

3. What is the similarity between flowcharts and activity diagrams? What is the principal difference between them?
4. What is an activity? What is an action?