

UML

(2) シーケンス図

# 本日のお題

第一節: UML を正しく覚えよう

第二節: シーケンス図の記法

# 第一節

UMLを  
正しく覚えよう

オブジェクトを

図面（ダイアグラム）

に描くための手段

= UML

システム開発をする  
中では、関係者とU  
MLを使って連携す  
る場面が、最近では  
普通にあります。

あなたは

UML読めますか？

『正確に』

UML読めますか？

Unified 統一

Modeling 模式化

Language 言語

OMG (Object  
Management Group)

で管理されており、

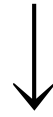
ISOでも

標準化されている。

UML規格仕様に規定  
された通りの意図を讀  
み取る必要がある。

= 正しく読めなければ  
意味がない。

間違った読み方



誤った仕様解釈



バグ発生

ゆえに、

**正確**な読み方を

学びましょう。

第一節  
おわり

## 第二節

### シーケンス図の記法

シーケンス図とは、

時間的な流れに沿って、オブジェクトやクラス間のやり取り(メソッド呼び出しやオブジェクト生成など)を記述する、振る舞いを示した図。

# シーケンス図で扱う対象

- 処理フロー
- インスタンス
- × クラスの静的構造
- 時間的な順序

設計レベルの  
詳細なシーケンス図  
は、そのまま  
ソースコードに  
変換することが  
できます。

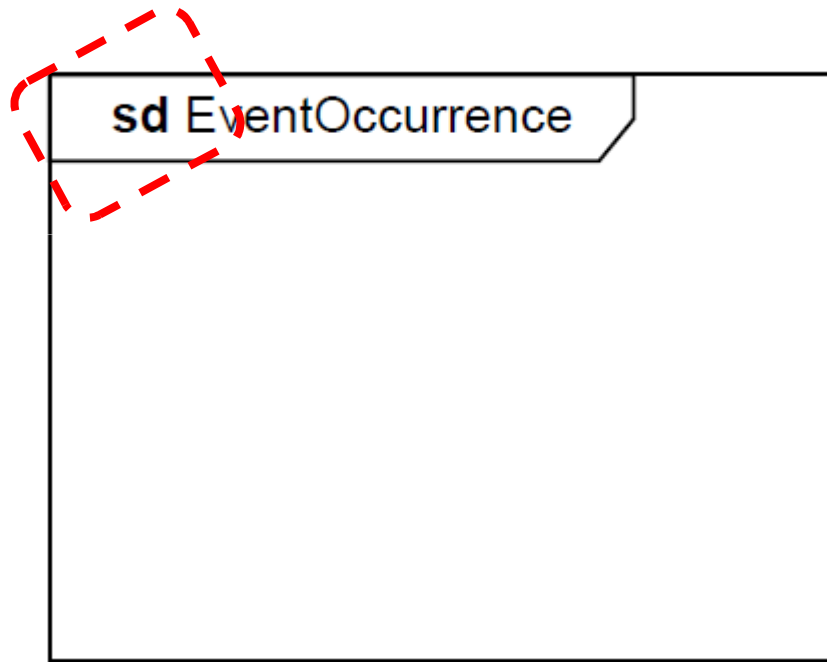
すべての記法を覚える  
必要はありません。

今日のスライドとサン  
プルに出てくる記法だ  
け覚えよう。

推獎

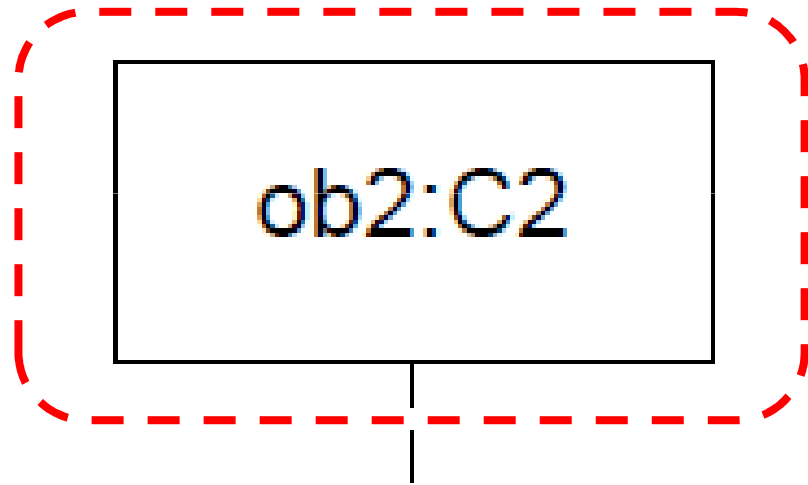
UML 2.0

# フレーム (Frame)



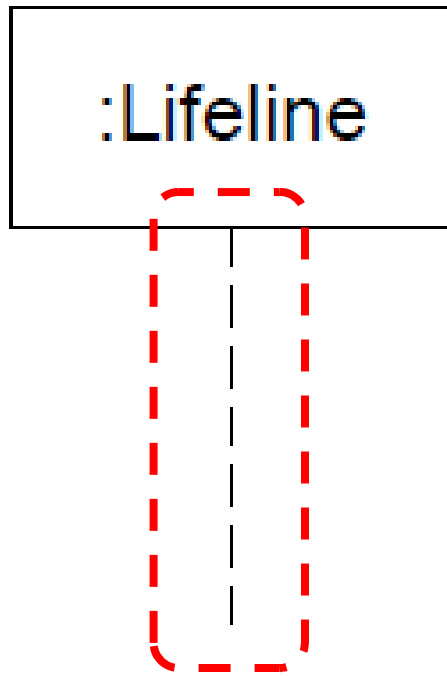
sd。シーケンス図であることを示す。相互作用の名前を記述します。

# ヘッド (Head)



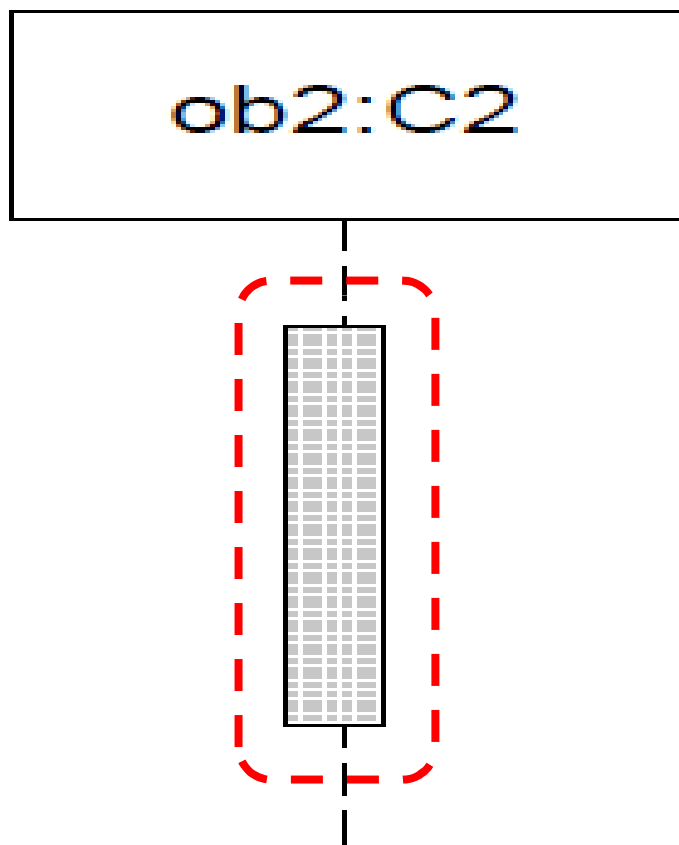
オブジェクト  
名：クラス名  
と書く。  
オブジェクト  
名は省略して  
もよい。

# 生存線 (Lifeline)



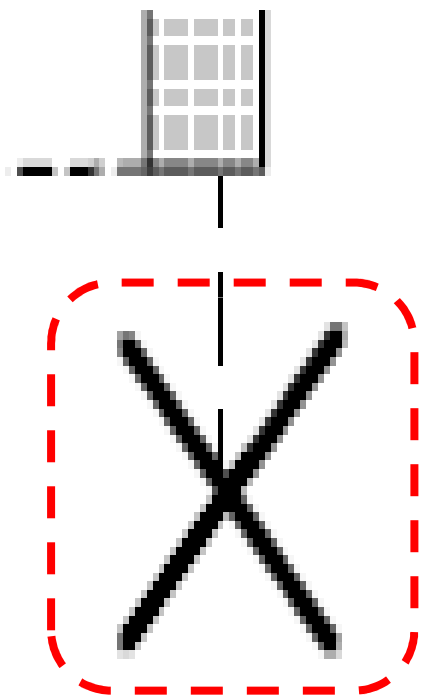
その相互作用において、ヘッドのオブジェクトが生存（メモリ上に存在）している期間です。

# 活性区間 (Execution Specification)



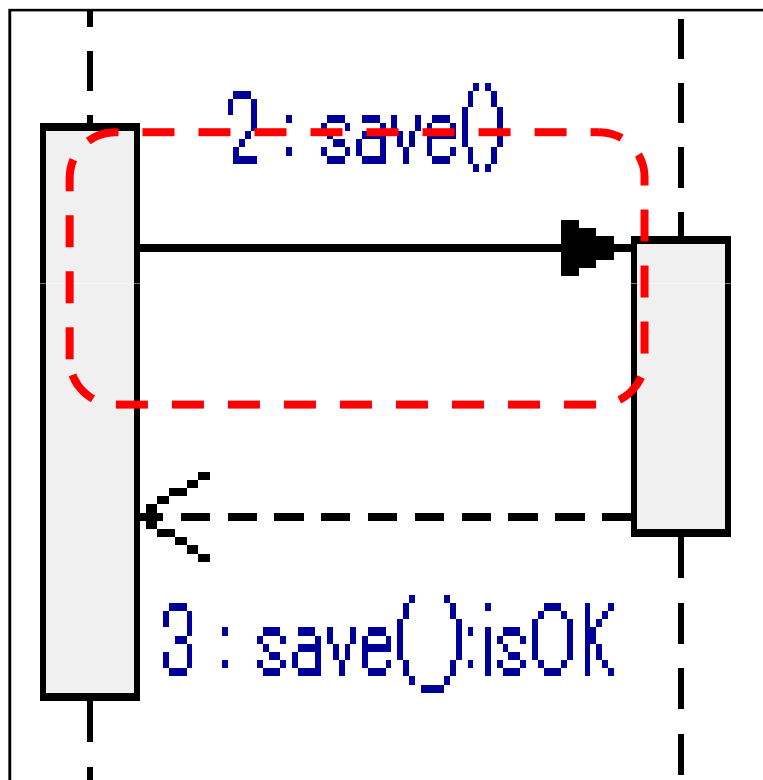
実行指定。生存線の中でのふるまいや行動の1単位の実行の(スタックに積まれている)明示期間です。

# 削除イベント (DestructionEvent)



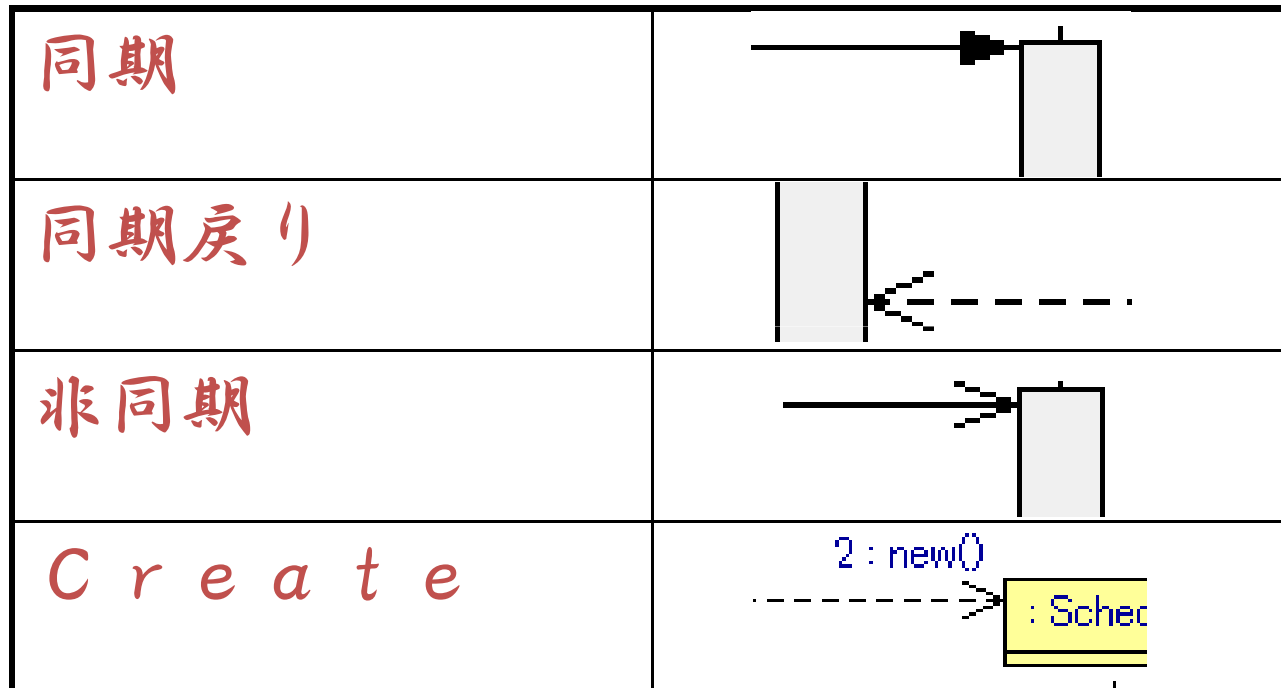
メモリ上から  
オブジェクト  
が削除される  
事を意味しま  
す。

# メッセージ (Messages)

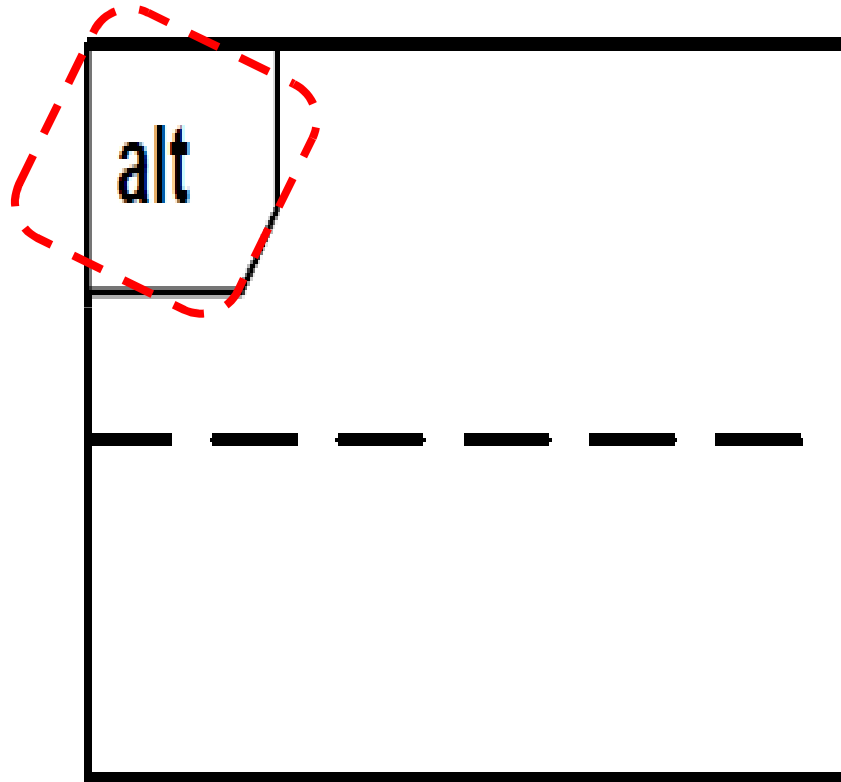


メソッドの  
呼び出しです。

# よく見かけるメッセージ



# 複合断片 (Combined Fragment)

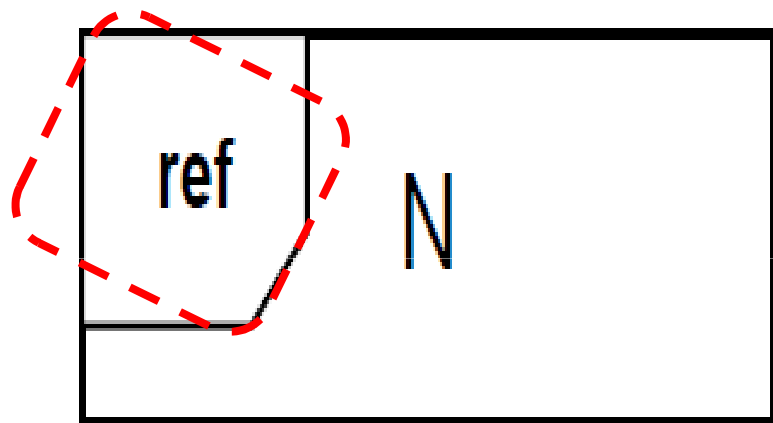


条件分岐や  
ループ等を  
シーケンス図  
上で表現する  
ことが  
できます。

# よく見かける複合断片

<i>a l t</i>	IF~ELSE~分岐
<i>o p t</i>	IF分岐
<i>l o o p</i>	FORループ
<i>s t r i c t</i>	順序を厳守
<i>c r i t i c a l</i>	クリティカル セクション

# 相互作用参照 (InteractionUse)



ref。別のsdで定義された相互作用をサブルーチンのように利用するという意味。

第二節  
おわり

## 参考サイト

- **OMG**

正式規格 (1系・2系とも)

<http://www.omg.org/spec/UML/>

- **UMTP**

UMLモデリング推進協議会

<http://www.umtp-japan.org>

おわり

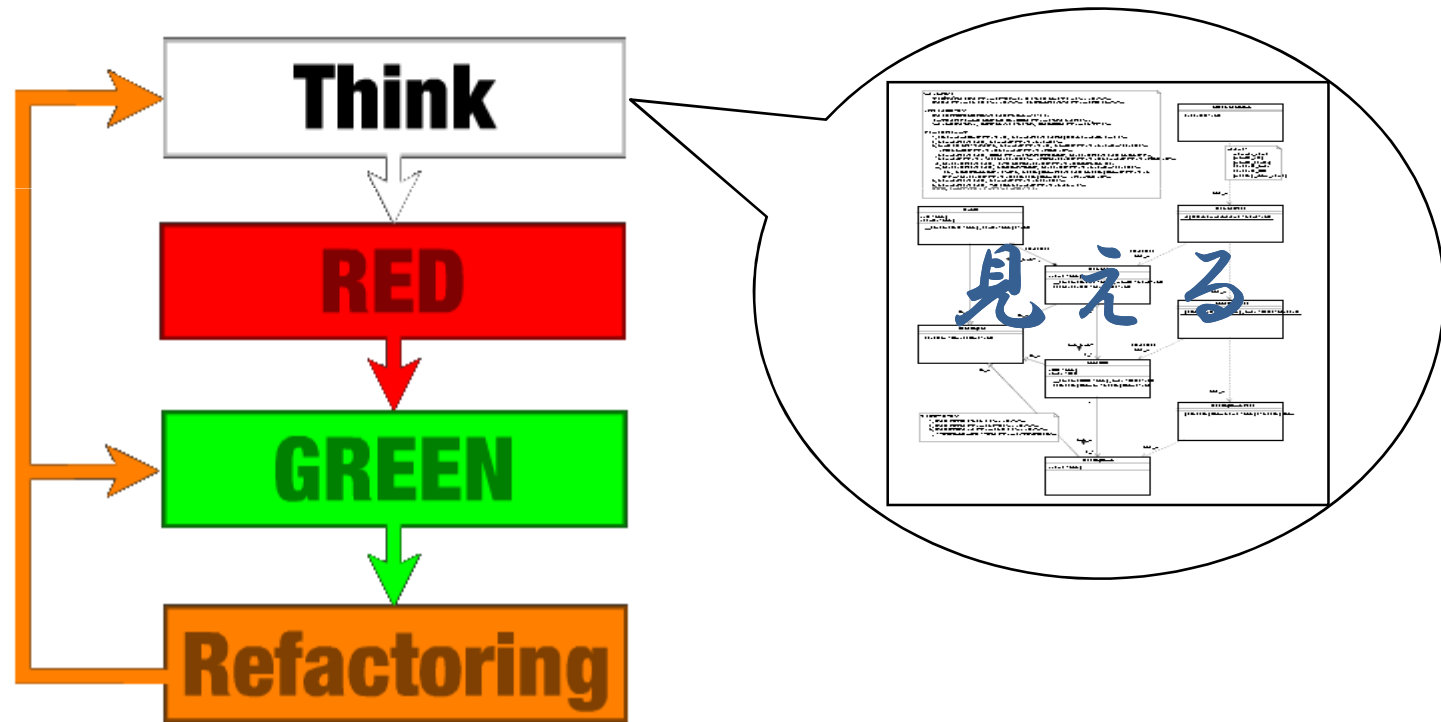
# 発表外おまけ

プログラマは

UMLと

どう付き合うべきか。

UMLは思考を助けてくれる道具です。



UMLを  
脳内で  
ソースコードに  
変換  
できるようにしよう