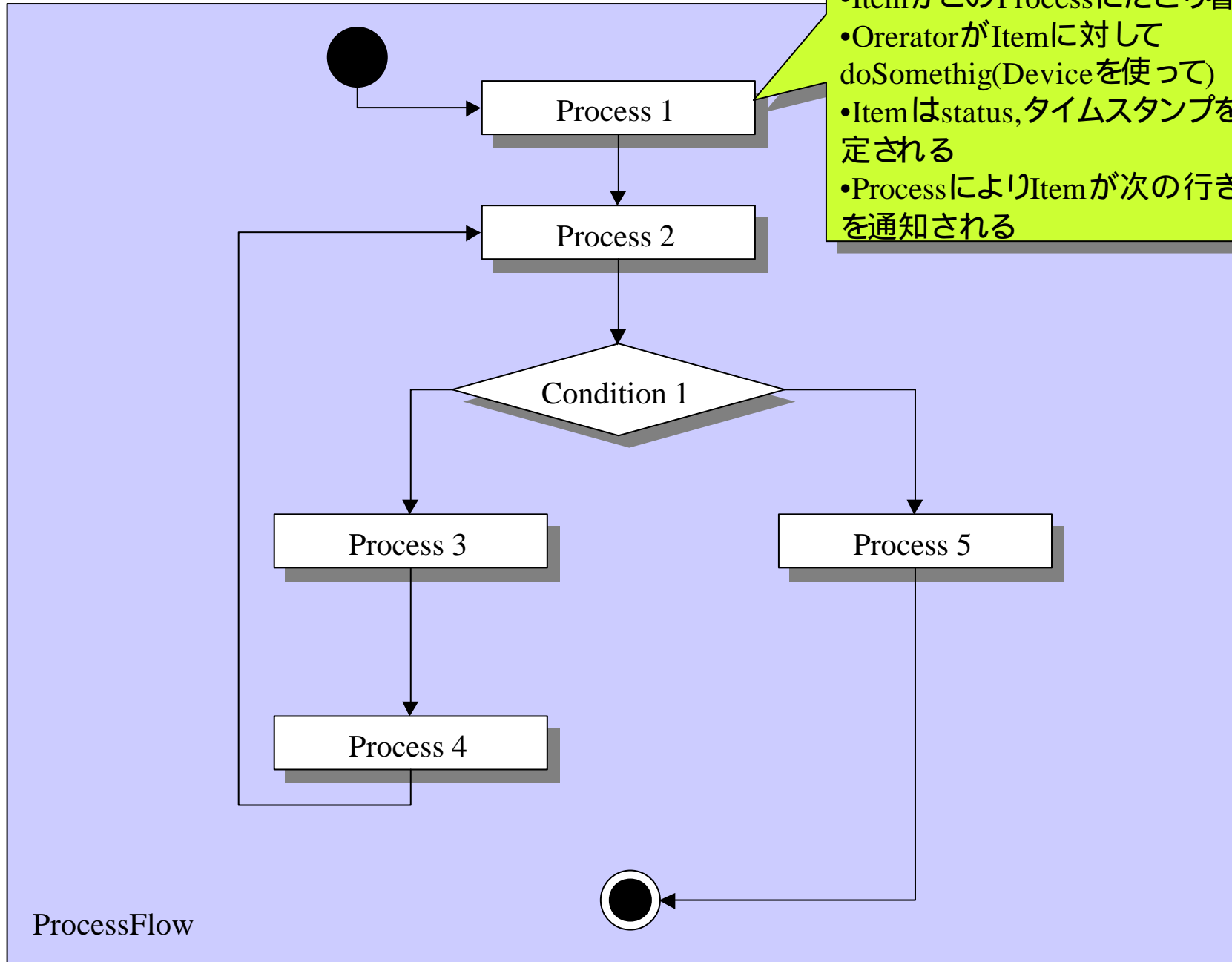


Yet Another Process Flow Manager (draft)

Author: KISHI Yasuhiro

Date: 2005/06/29

概念図



- ItemがこのProcessにたどり着く
- OreratorがItemに対して doSomethig(Deviceを使って)
- Itemはstatus, タイムスタンプを設定される
- ProcessによりItemが次の行き先を通知される

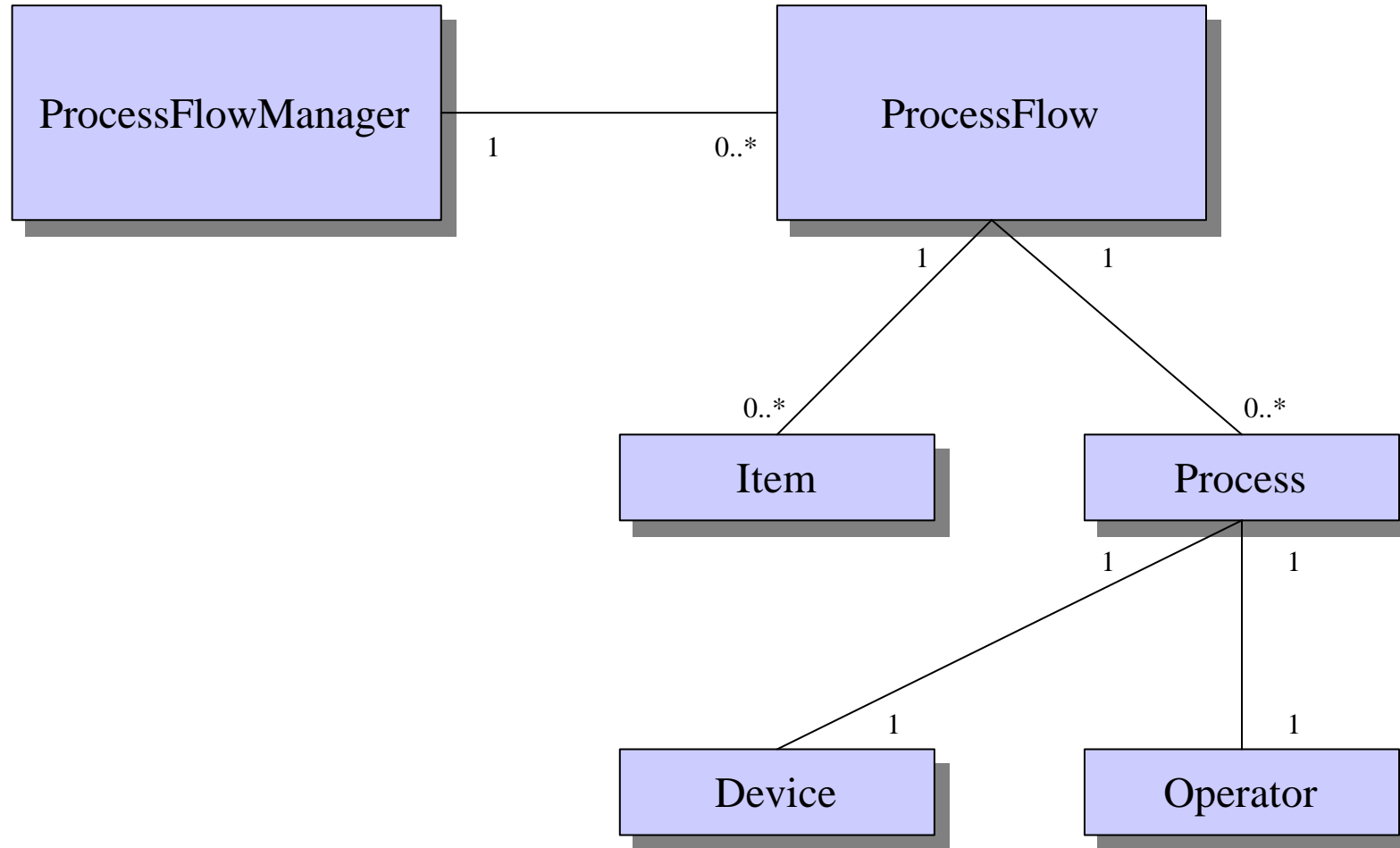
必要となるクラスと処理内容の説明

- Itemに対してはProcessのオブジェクトで構成されたProcessFlowの参照を引き渡す。
- ItemはTraceable インタフェースを実装したものとする。
- Itemはstatus という属性を持ち、Itemが現在位置しているProcessが次のProcessあるいはConditionをItemに伝える。
- ProcessFlowは、複数個のProcessがインスタンス化されたものをグラフ構造にて格納する。
- HistoryManagerは、Itemの状態遷移履歴を格納(非同期)する。 本来はDBを使用する。
- ProcessFlowManagerは全体のオブジェクトの管理をする。

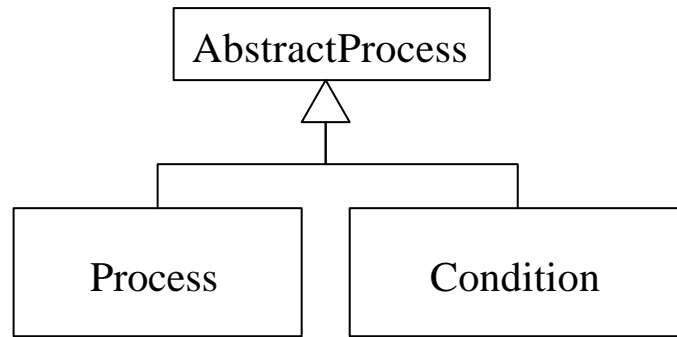
トレーサビリティの本質とは

1. Why, What, Who, When, Where, How をロギングする
2. 現在のステータスを把握できる
3. 現状に至った状態を即座に因果関係として求めることができる

プロセスマネージメント概念(役割分担・関連性について)



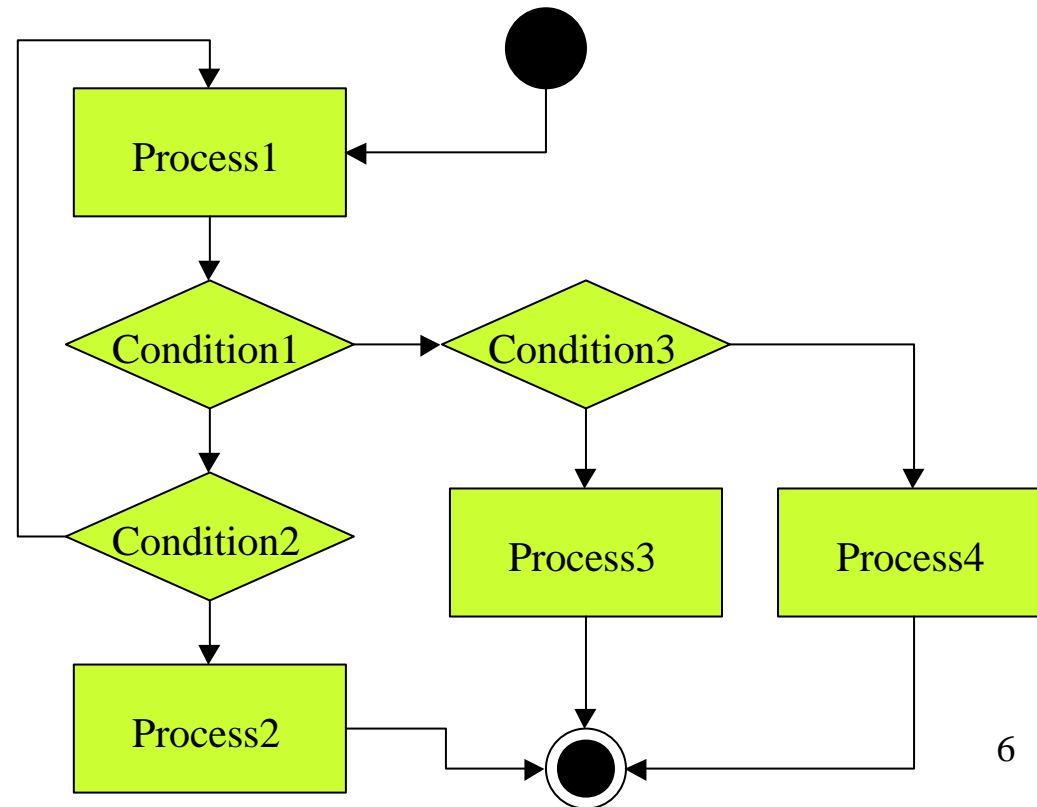
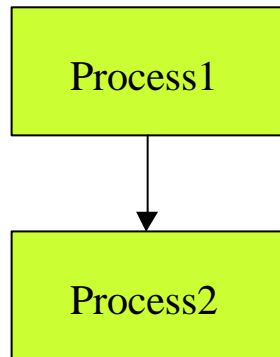
AbstractProcessとProcess,Conditionの関係



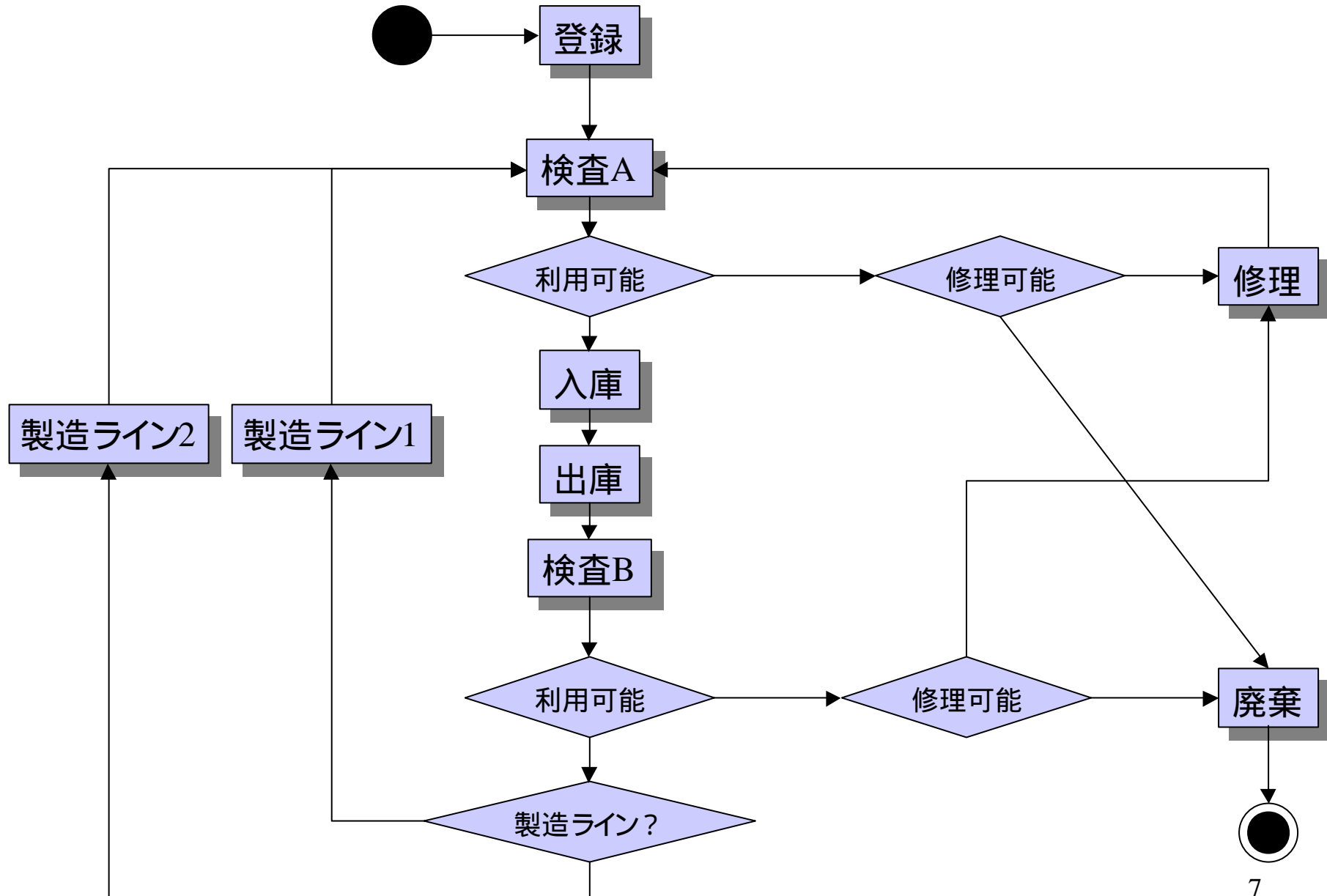
• 終端でない場合、Processは必ず次のステップとしてProcessあるいはConditionを持つ（どちらか一つ）。

• Conditionは内部で判断する真偽の状態により、次の進路として必ず2つのAbstractプロセスを持たなければならない。

課題！
SWITCHステートメントのようなまとめて判断するクラスもあとで必要となる！



プロセスフローのサンプル



課題

1. UIをどう作るか(SWINGベース)
2. Operatorの権限管理(認証、役割)