

ビジネスプロセスの検証ツールの開発

株式会社日立製作所

村田大二郎

開発における問題点

- ・要件定義段階での問題は、後になるほど修正が大変
- ・ビジネスプロセスのデッドロックや詳細化関係
- ・ソースコードを自動生成する場合、検証された元情報が必要

手法・ツールの適用による解決

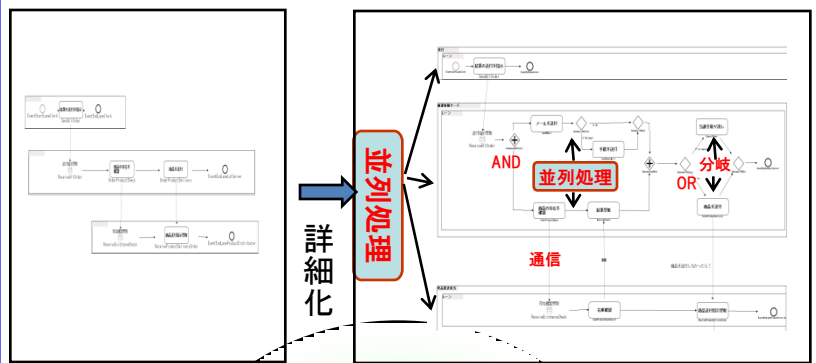
- ・詳細化されるBPMNの検証のため、BPMNからCSPモデルへの変換規則を作成
- ・CSPを用いた検証ツールFDR2のコードをBPMNから生成するツールを開発
- ・CSPモデル実装のためのライブラリJCSPを利用し、BPMNからJavaコードへの生成ツールを開発

ビジネスプロセスのFDR2による検証

ビジネスプロセスのデバッグ

設計者はFDR2のデバッグ結果に表示される名前と同じ名前の、ビジネスプロセス上の要素を見つけ、修正する

ビジネスプロセスエディタ

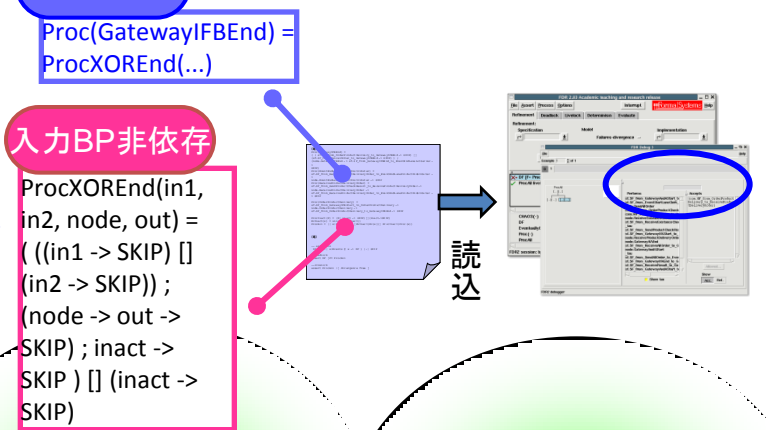


設計者は並列処理を有するBPMNダイアグラム(ビジネスプロセス)を編集

生成ツール(今回開発)

FDR2により検証されたBPMNからJavaソースコードを生成することで、デッドロック・ライブロックの検証されたプログラムを利用可能

FDR2



入力BP依存
Proc(GatewayIFBEnd) = ProcXOREnd(...)

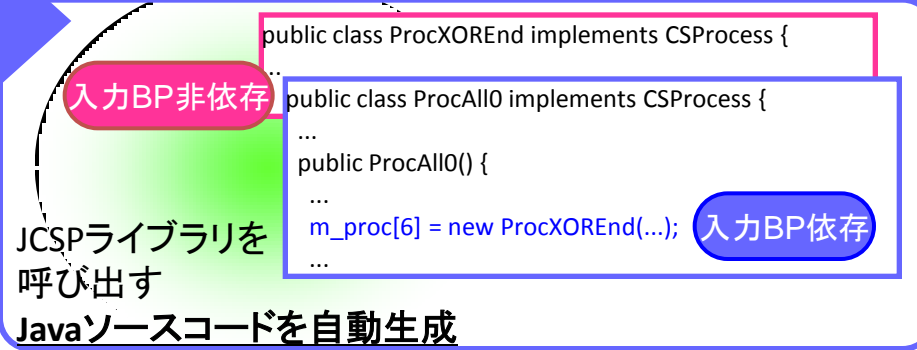
入力BP非依存
ProcXOREnd(in1, in2, node, out) = ((in1 -> SKIP) [] (in2 -> SKIP)); (node -> out -> SKIP); inact -> SKIP [] (inact -> SKIP)

生成ツールを用い、ビジネスプロセスからFDR2コードを自動生成

読込

FDR2コードを読み込み、デッドロックを検証

JCSPによるプロト開発



入力BP非依存

```
public class ProcXOREnd implements CSPProcess {
    ...
    public ProcAllIO() {
        ...
        m_proc[6] = new ProcXOREnd(...);
        ...
    }
}
```

入力BP依存

JCSPライブラリを呼び出す

Javaソースコードを自動生成

BPMN: Business Process Modeling Notation
CSP: Communicating Sequential Processes
JCSP: Communicating Sequential Processes for Java™



トップエスイー修了制作

タイトル

所属

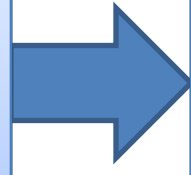
名前

メールアドレス

開発における問題点

手法・ツールの適用による解決

あああ



あああ

ヘッダ

ヘッダ



トップエスイー修了制作

タイトル

所属

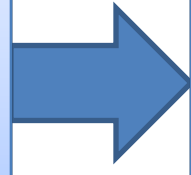
名前

メールアドレス

開発における問題点

手法・ツールの提案による解決

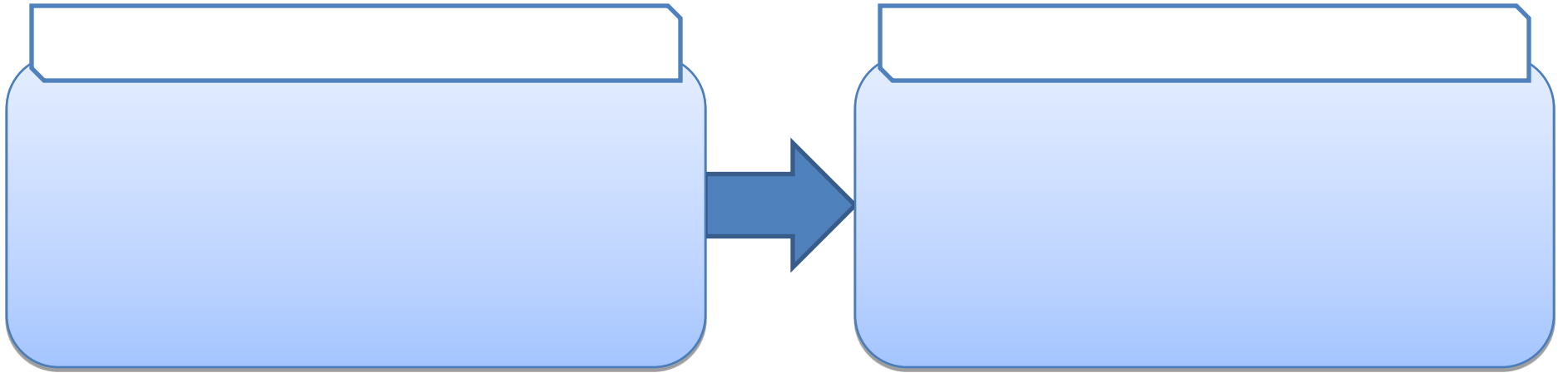
あああ



あああ

ヘッダ

ヘッダ



```

(略)
Proc(GatewayIFBEnd) =
  ( ( sf.SF_from_OrderProductDelivery_to_GatewayIFBEnd -> SKIP) []
  (sf.SF_from_GatewayIFBEnd_to_GatewayIFBEnd -> SKIP) ) ;
(node.GatewayIFBEnd -> sf.SF_from_GatewayIFBEnd_to_EventEndLaneServer -
>
SKIP)
Proc(EventEndLaneProductDistributor) =
sf.SF_from_ReceiveProductDeliveryOrder_to_EventEndLaneProductDistributor -
>
node.EventEndLaneProductDistributor -> SKIP
Proc(ReceiveProductDeliveryOrder) =
sf.SF_from_SendProductCheckResult_to_ReceiveProductDeliveryOrder ->
node.ReceiveProductDeliveryOrder ->
sf.SF_from_ReceiveProductDeliveryOrder_to_EventEndLaneProductDistributor -
> SKIP
Proc(OrderProductDelivery) =
sf.SF_from_GatewayIFBStart_to_OrderProductDelivery ->
node.OrderProductDelivery ->
sf.SF_from_OrderProductDelivery_to_GatewayIFBEnd -> SKIP
ProcInact(P) = (P; inact -> SKIP) [] (inact->SKIP)
EvInact(s) = union(s, {inact})
ProcAll = [] x:Procname @ {EvInact(Ev(x))} ProcInact(Proc(x))

(略)

-- assertion
DF = (!| x:Events @ x -> DF ) |-> SKIP
--deadlock
assert DF [P= ProcAll
--livelock
assert ProcAll :{ divergence free ]
    
```