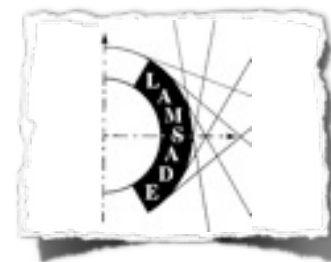


# Projet ANR JCJC PERSO

## Pervasive Service Composition

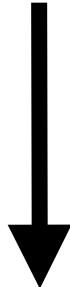
Appel «Jeunes Chercheurs»  
Projet JC07\_186508  
Décision ANR-07-JCJC-0155-01  
2007 - 2010  
<http://anr-perso.ibisc.fr/>



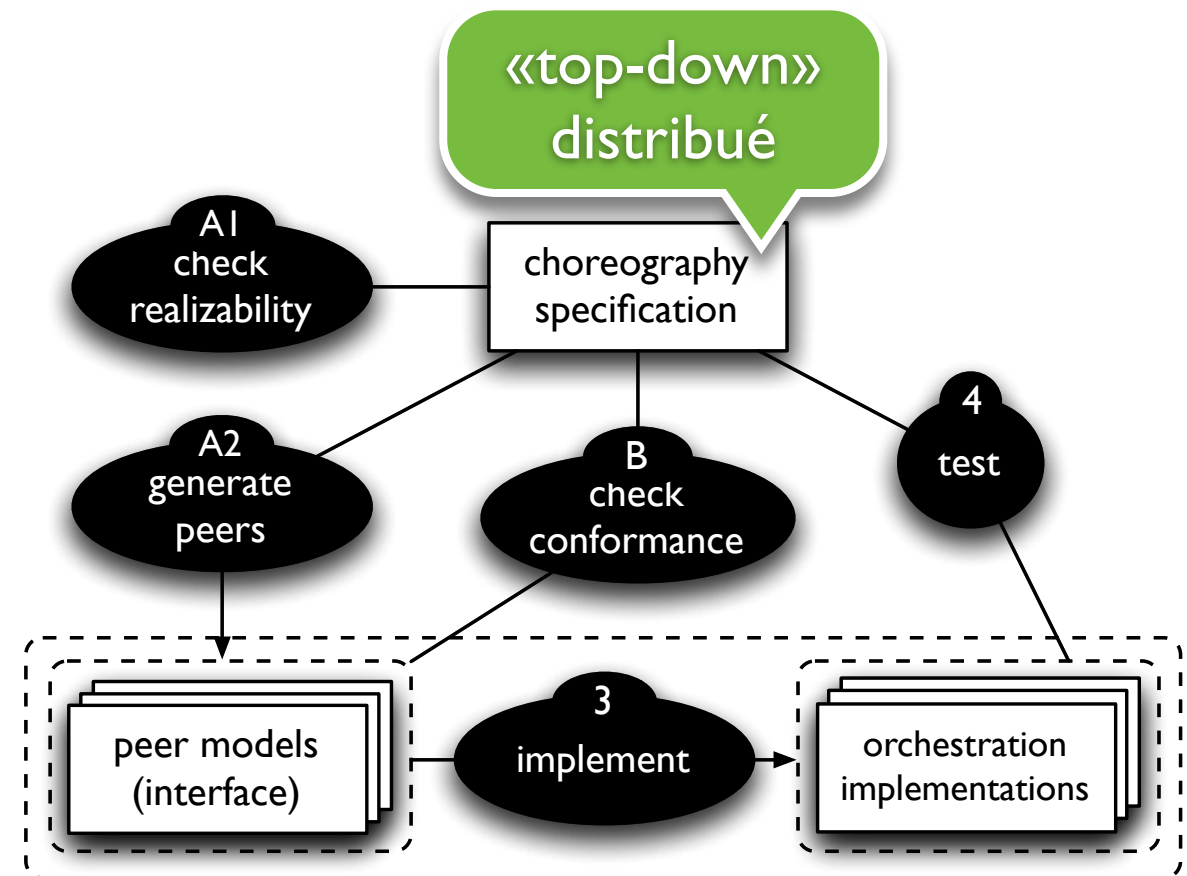
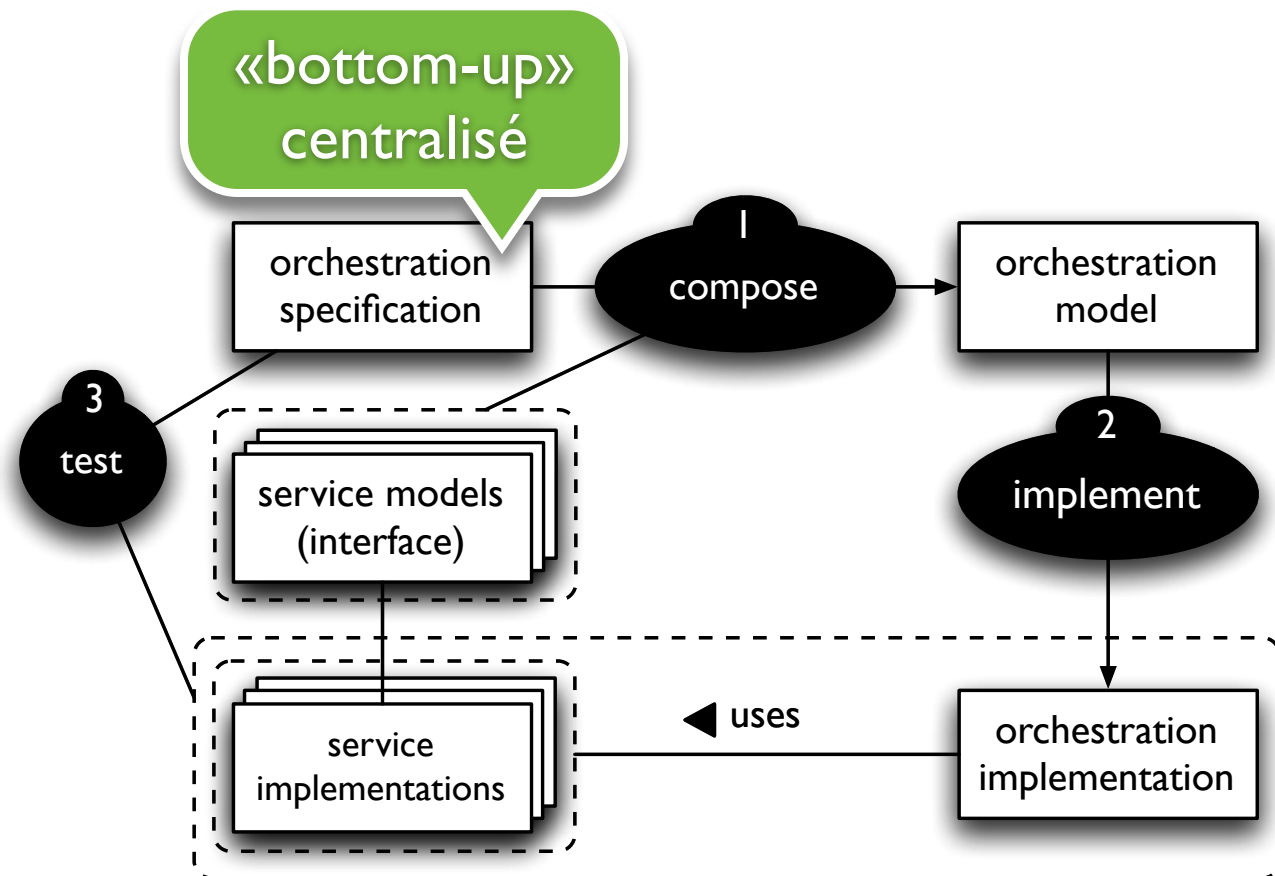
**Pascal Poizat**

Université d'Evry Val d'Essonne;  
LRI - CNRS et Université Paris Sud

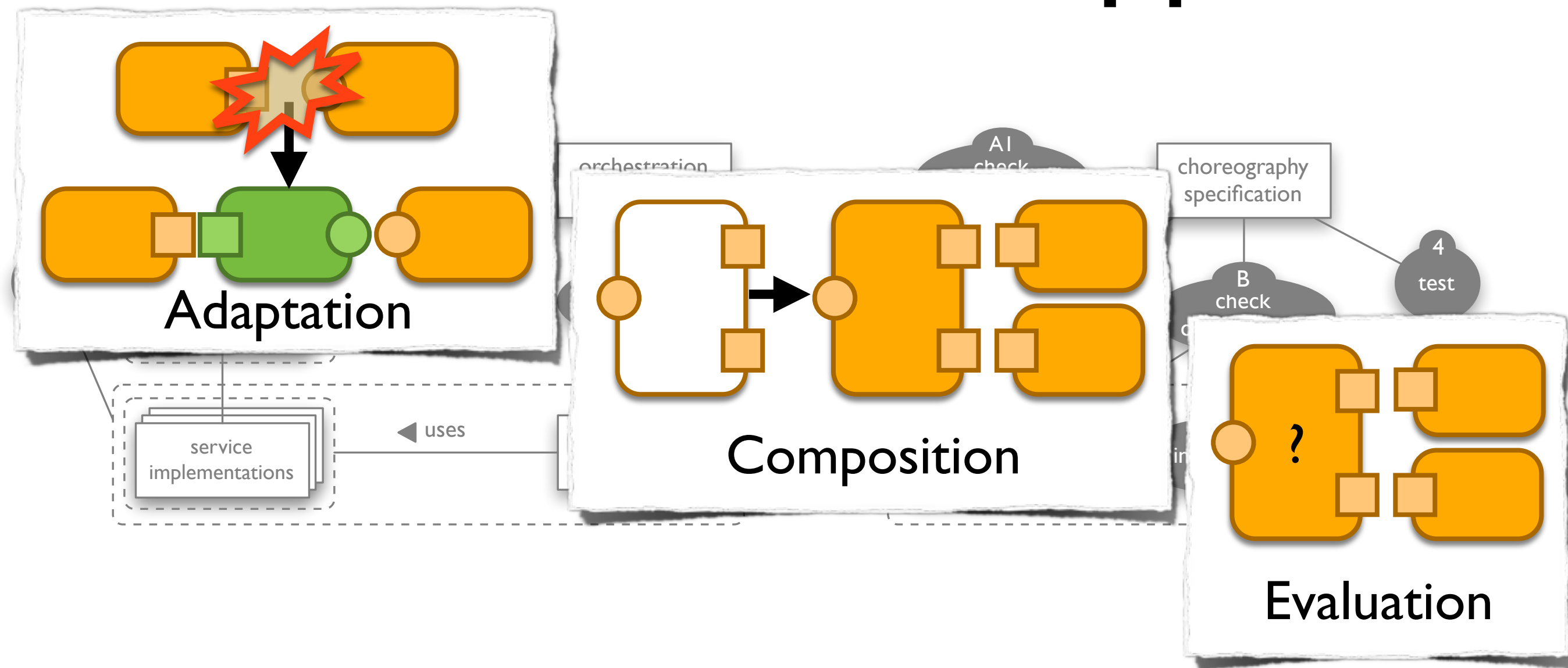
# Contexte

- **Internet & informatique diffuse**
  - ▮▮▮▮➡ accès accru aux données et aux moyens de calcul
- **ingénierie** logicielle à base de **services**
  - évolution modules → classes → services
  - ▮▮▮▮➡ encapsulation (données/calculs), faible couplage
- logiciel **pour (et par) l'utilisateur final**
  - «end-user programming»
  - ▮▮▮▮➡ nécessité d'automatisation et de transparence
- **composition automatique de services**
  - ▮▮▮▮➡ réutilisation + création de services à valeur ajoutée
  - ▮▮▮▮➡ adéquation aux besoins utilisateurs + transparence
  - ▮▮▮▮➡ «software-as-a-service»

# Processus de développement

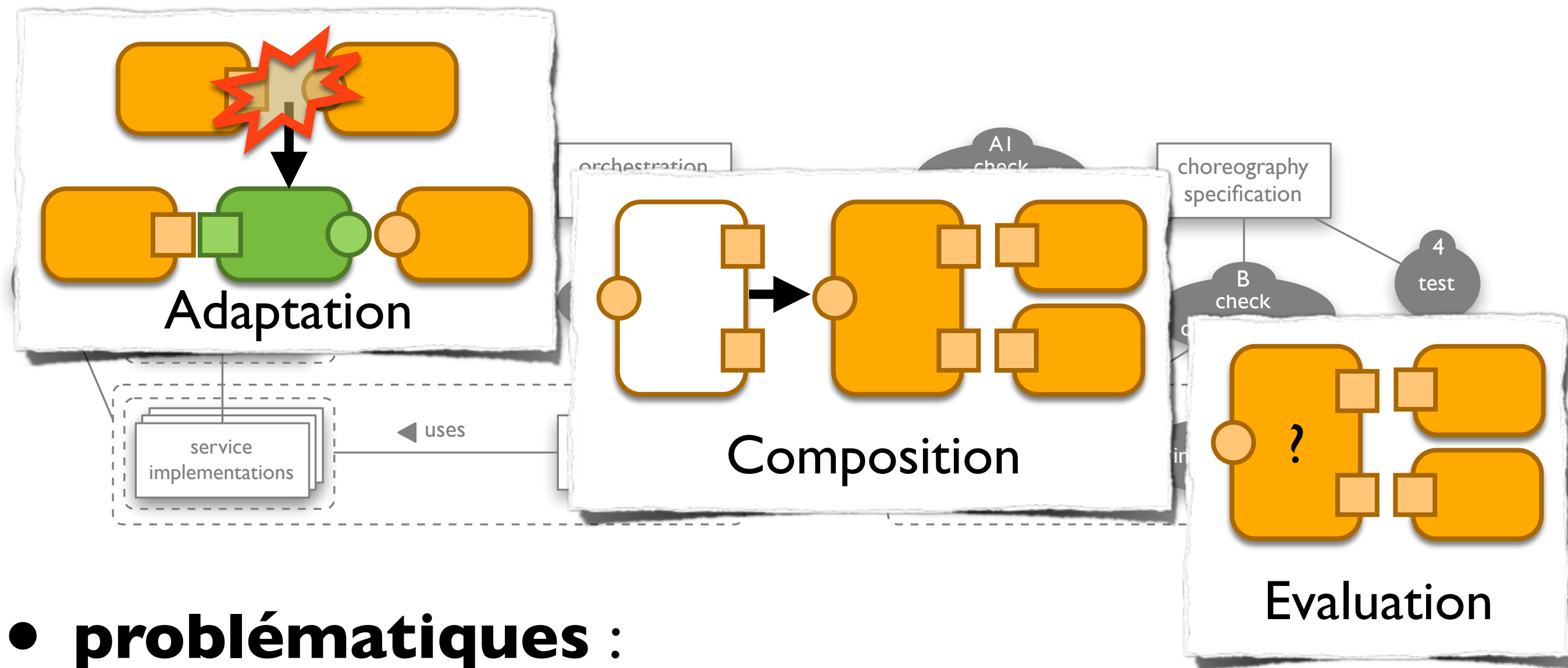


# Processus de développement



- **3 grandes activités :**
  - **adaptation**
  - **composition**
  - **évaluation**

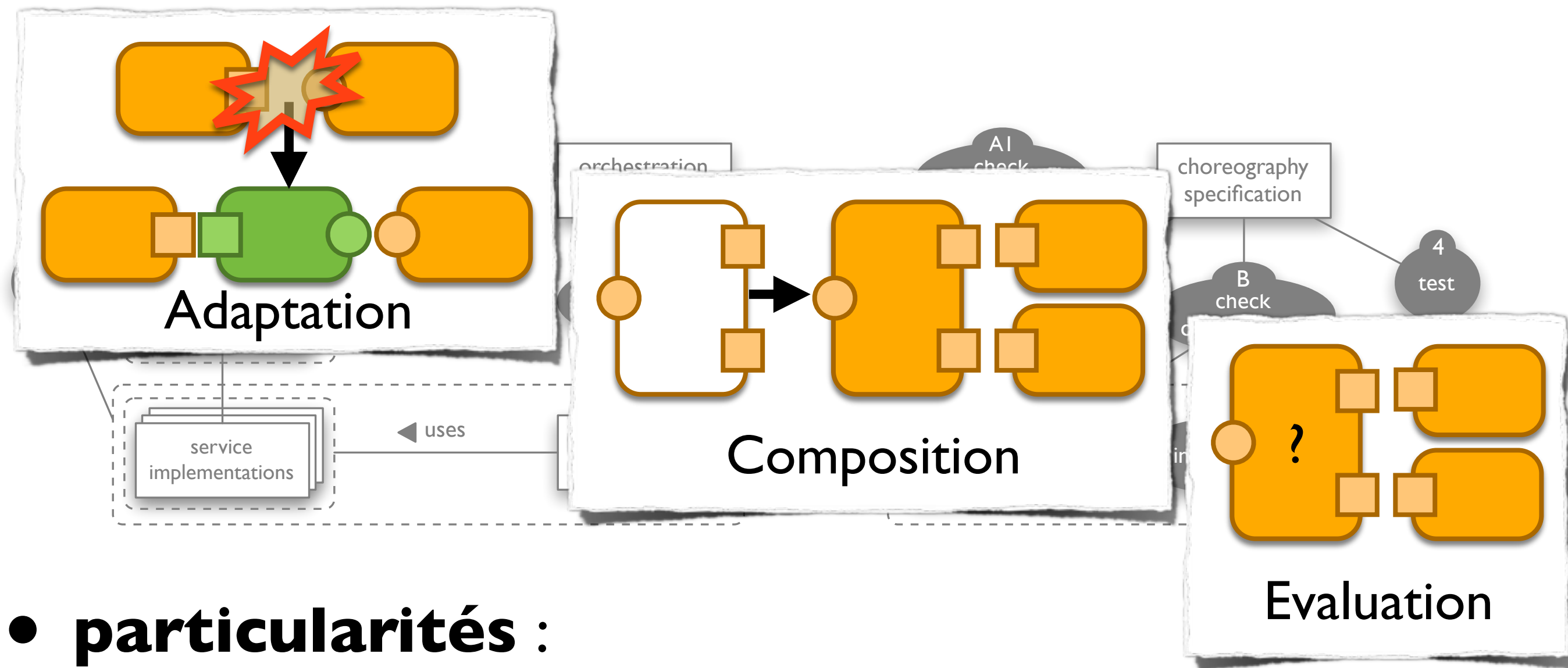
# ANR PERSO



- **problématiques :**

- **support** de **différents types d'interfaces**  
respect de **contraintes** et satisfaction de **besoins comportementaux, non fonctionnels et sémantiques**
- composition **adaptive** (résolution des **incompatibilités**)
- composition **distribuée**

# ANR PERSO



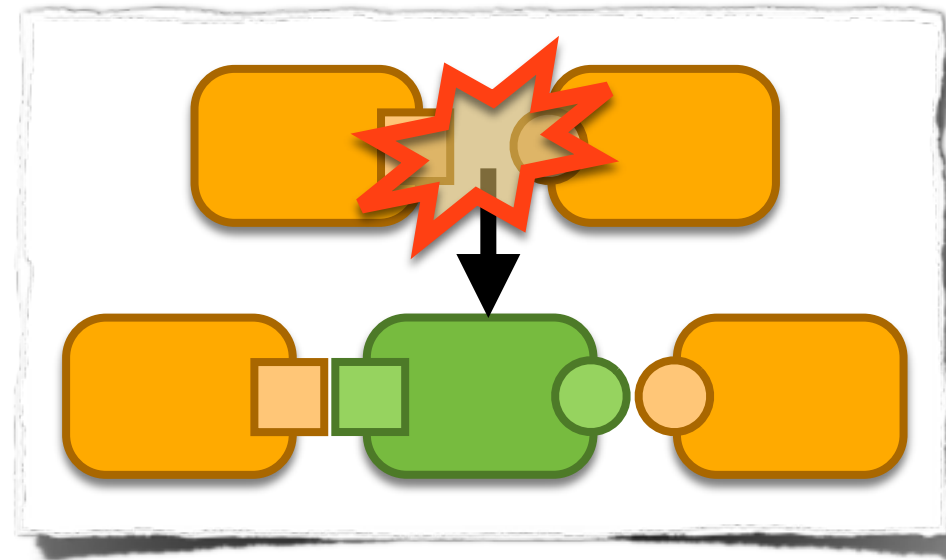
- **particularités :**

- **combinaison MDE + méthodes formelles**

- ➡ généricité

- ➡ rigueur et outillage des solutions proposées

- utilisation de **techniques «exotiques»** en génie logiciel (optimisation combinatoire et analyse de performance)



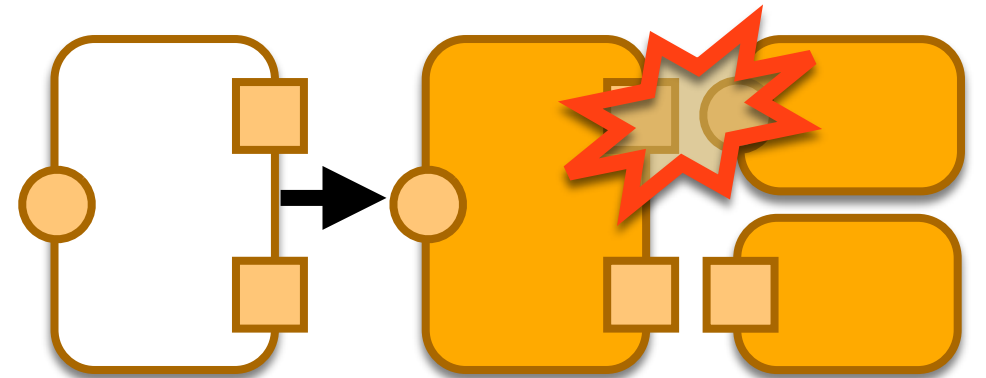
# Adaptation

# Problème

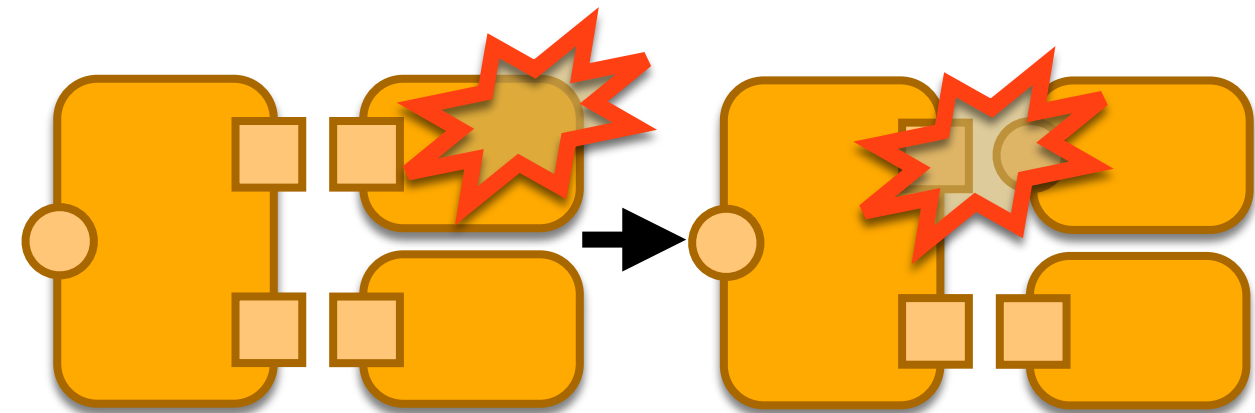
- **fait :**  
les composants sont  
**développés séparément**  
par différents tiers



limitation de la réutilisation



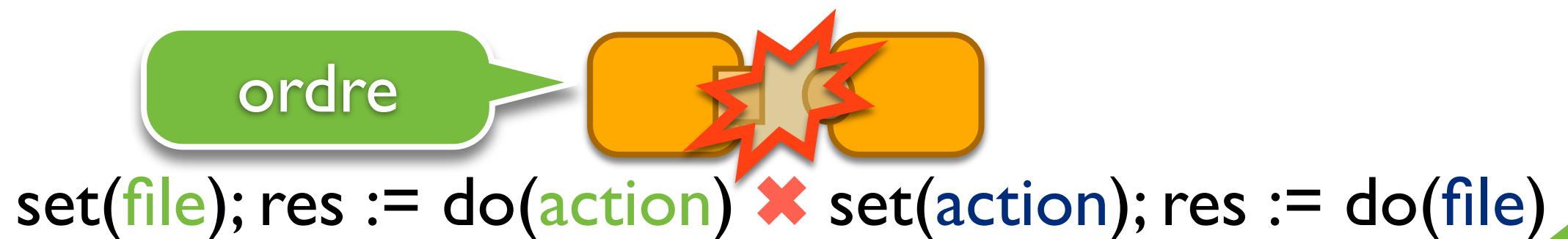
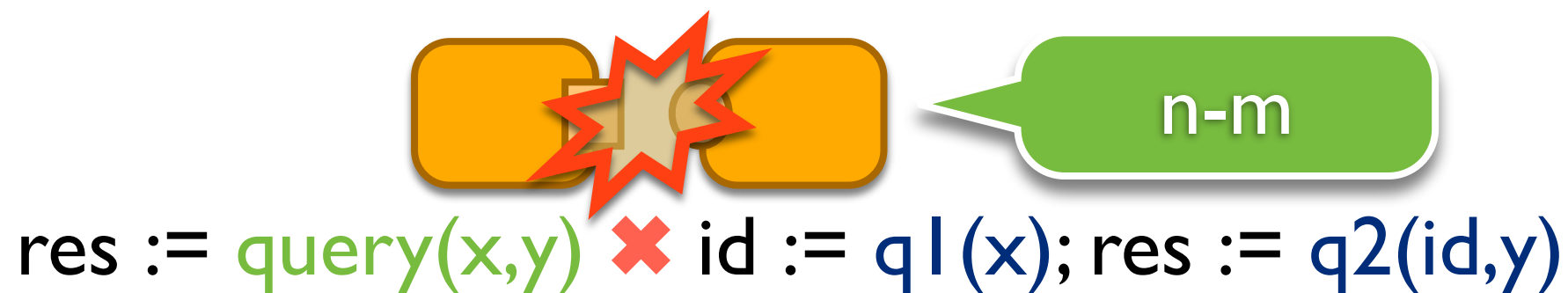
limitation de la composition



limitation de la réparation

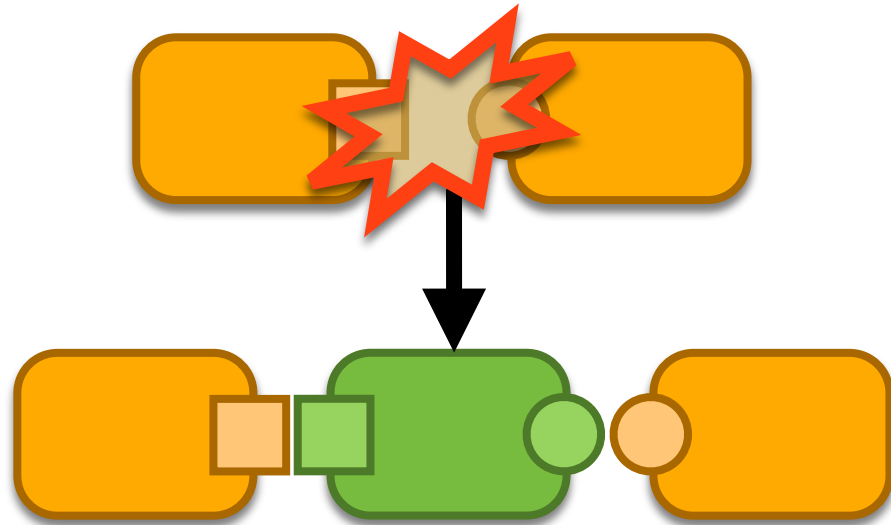


# Incompatibilités



# Adaptation

- création d'**adaptateurs**

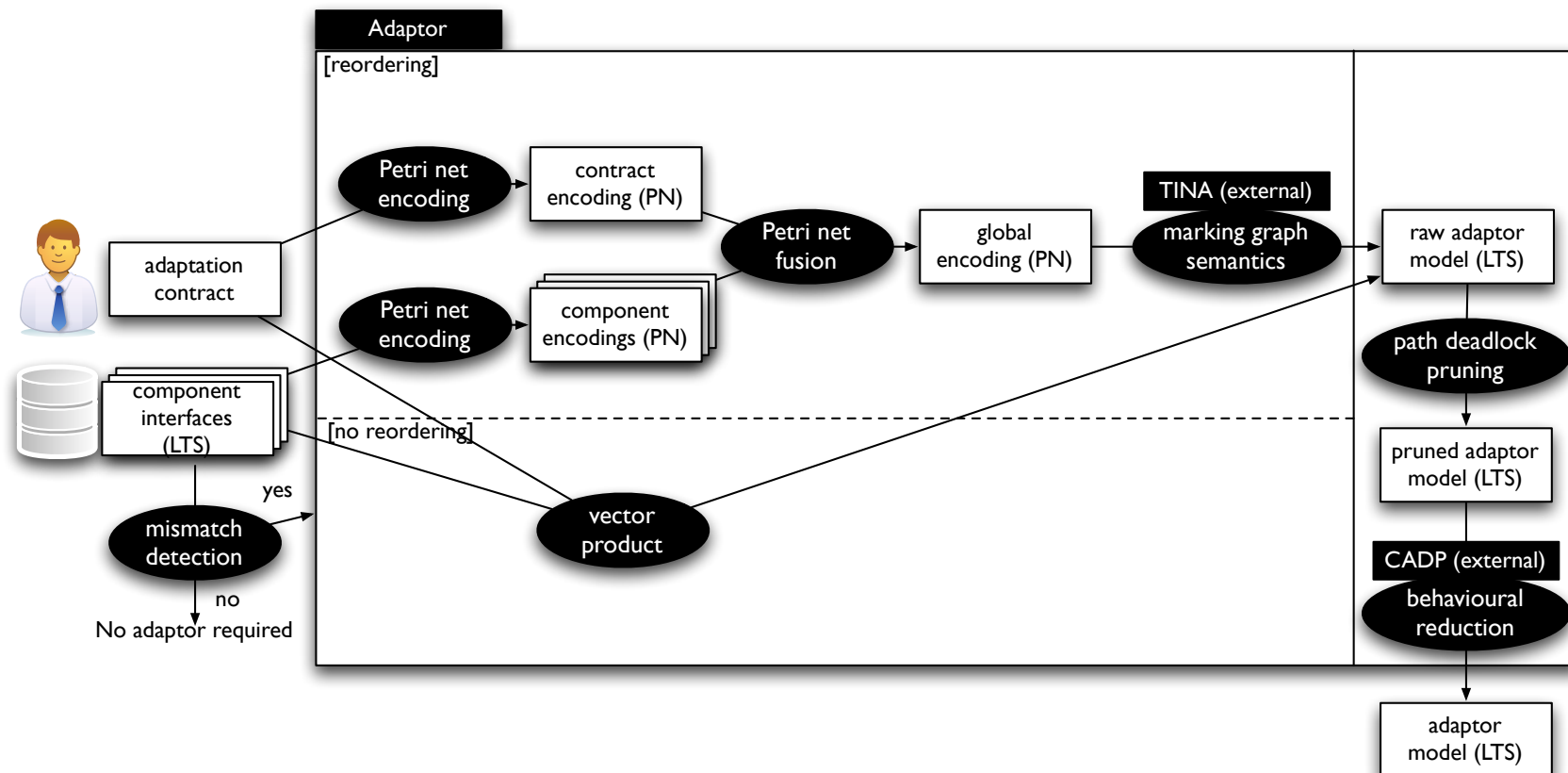


- **non intrusif**  
≠ évolution
- **dynamique**  
≠ paramétrage
- **modèles + code**  
≠ intergiciels adaptatifs
- **données + buffering**  
≠ synthèse de contrôleur

- approches ad-hoc  
non générales / manuelles  
*[Schmidt and Reussner, 2002],*  
*[Benatallah et al, 2005],*  
*[Dumas et al, 2006]*
- approches **restrictives**  
suppression d'interactions  
n-aires, pratiques (outils)  
*[Inverardi and Tivoli, 2003]*
- approches **génératives**  
renommage, réordonnancement  
binaires, théoriques  
*[Yellin and Strom, 1997],*  
*[Bracciali et al, 2005],*  
*[Brogi et al, 2006]*

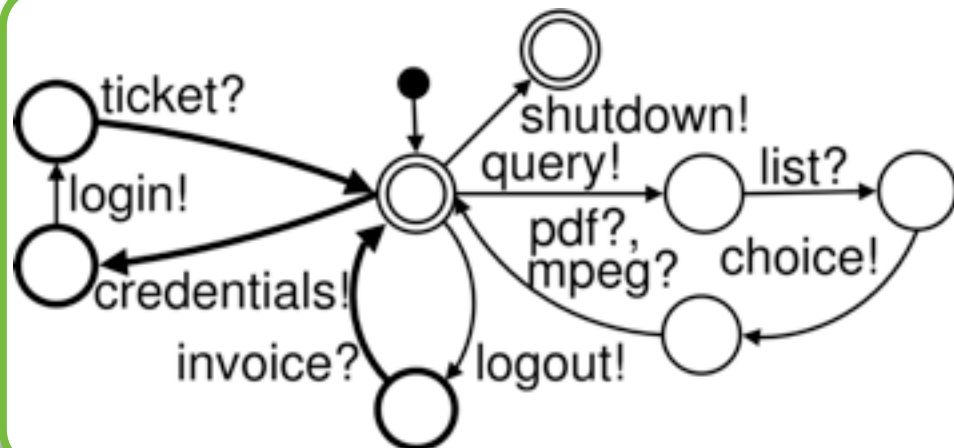
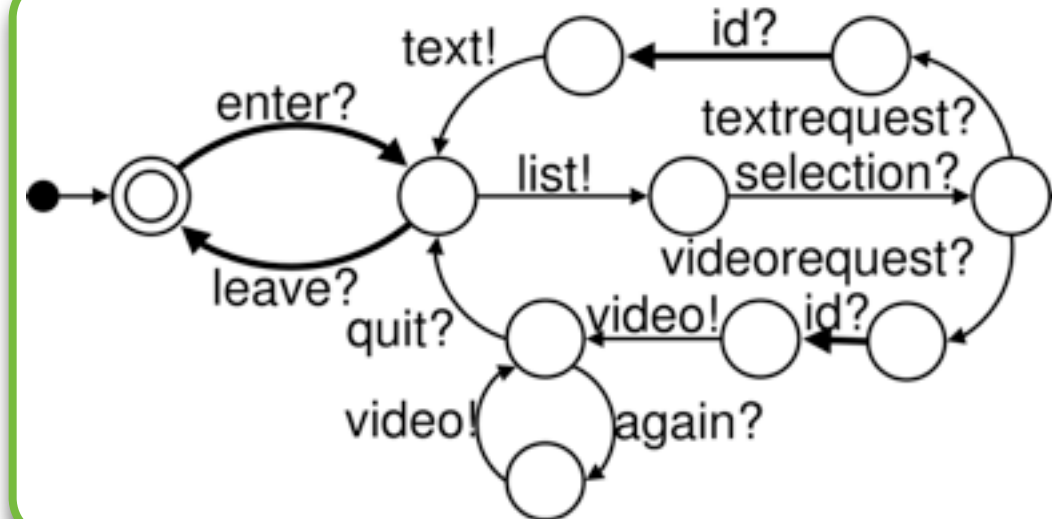
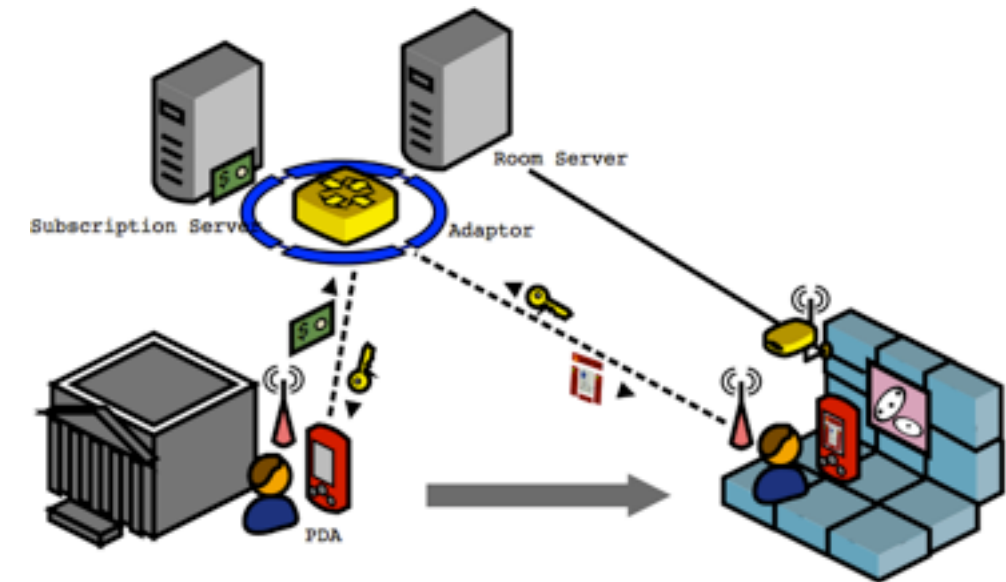
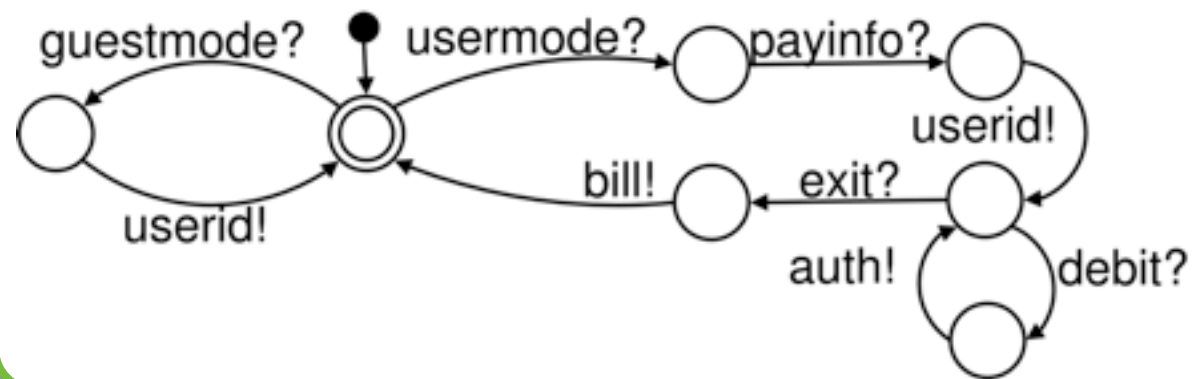
# Adaptation comportementale

restrictif + génératif  
n-aire  
gestion de propriétés  
application à WWF

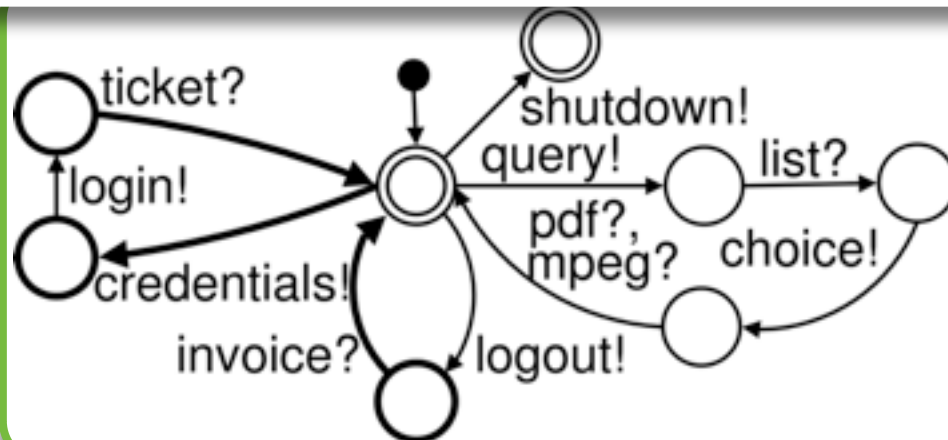
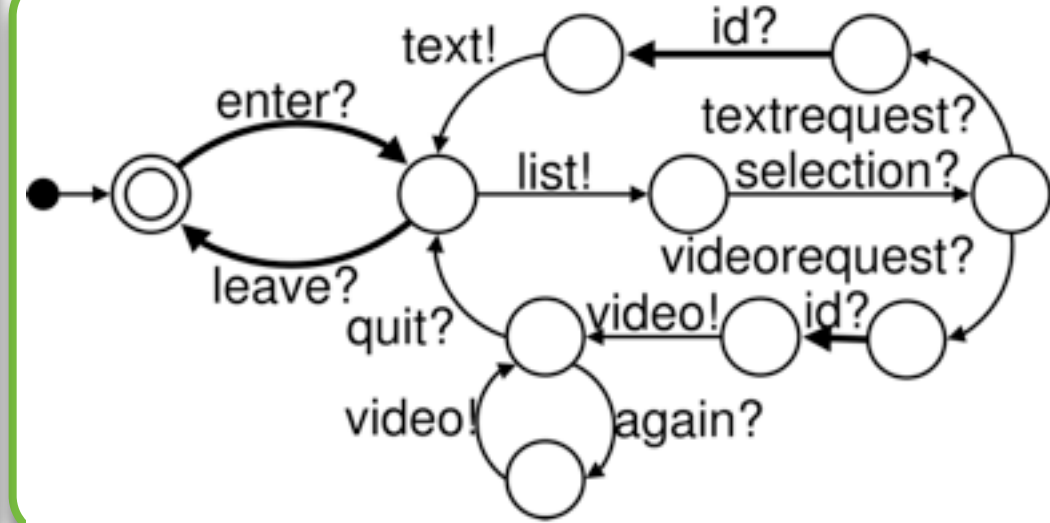
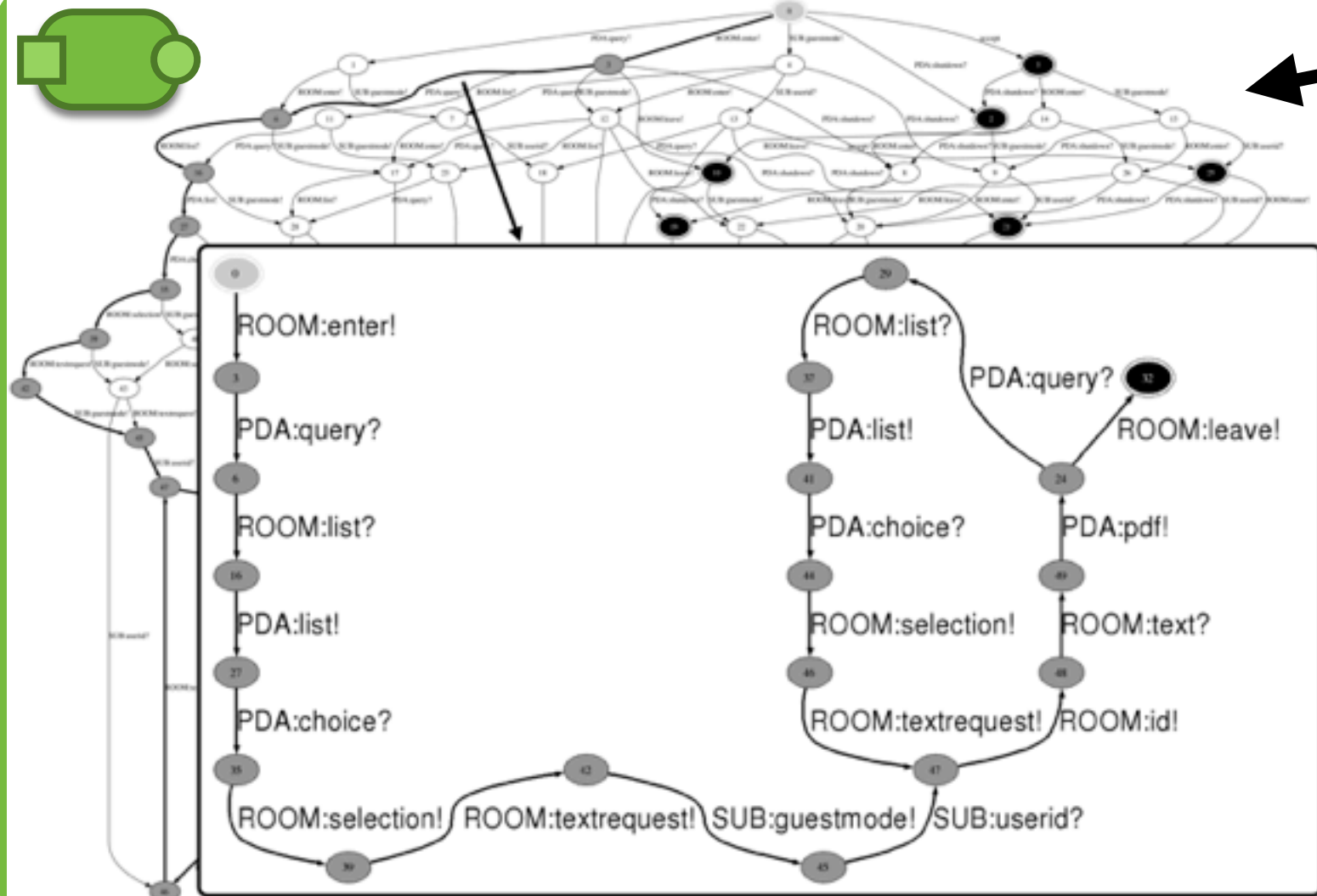
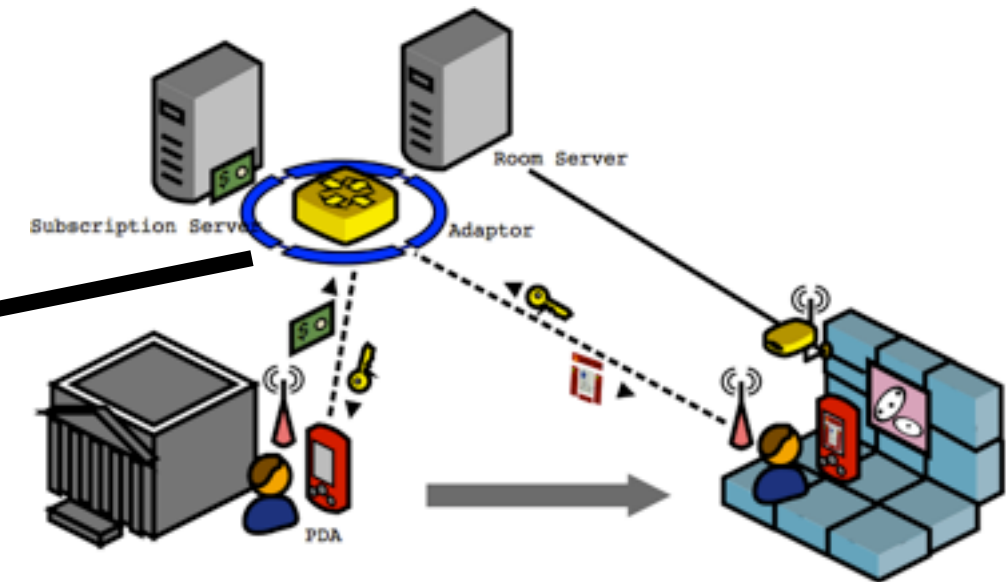
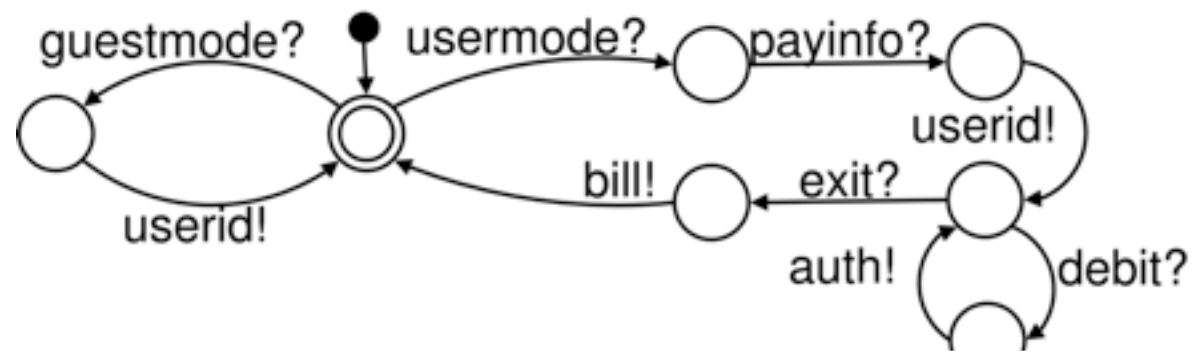


- **interfaces comportementales** des services (LTS)
- **contrat d'adaptation** (LTS étiqueté par vecteurs)  
 $\langle s_i: [m_i | *] \rangle$   
ex:  $\langle s_1: m, s_2: * \rangle, \langle s_1: m_1; s_2: m_2 \rangle$
- étape **générative** : **encodage**  $\Rightarrow$  réseaux de Petri  
+ **fusion** des réseaux  
étape **restrictive** : calcul graphe des marquages  
+ **suppression** chemins bloquants

# eMuseum



# eMuseum





# Adaptation comportementale

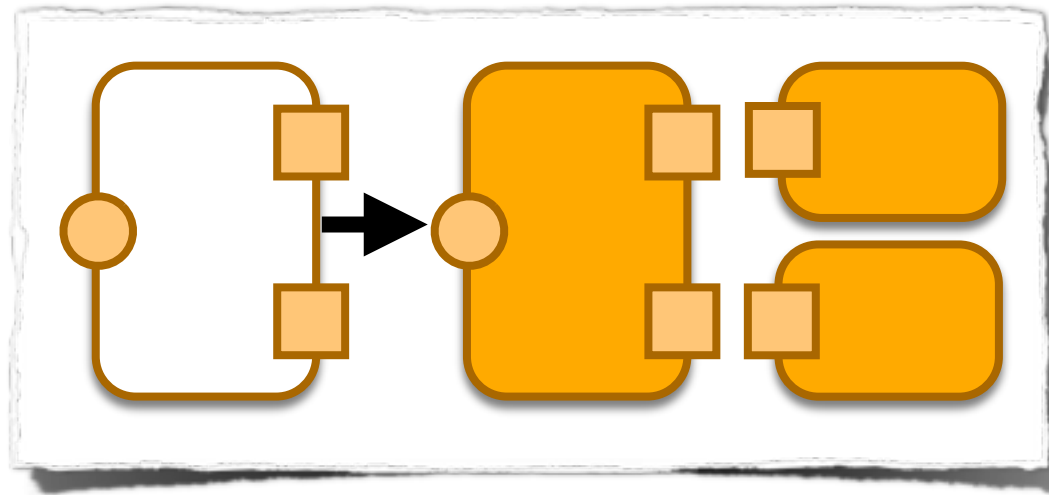
restrictif + génératif  
n-aire  
gestion de propriétés  
application à VWF

approche à la volée (+ efficace)  
support direct des données  
vérification de systèmes adaptés  
application aux VWS «de base»

approche distribuée  
totalement automatique  
(pas de contrat)  
application aux VWS sémantiques

- extension symbolique des interfaces ( $LTS \rightsquigarrow STS$ ) et des vecteurs
- encodage implicite (réseau de STS)
- absence de blocage encodée en BES  
restriction avec Caesar.Solve
- extraction de modèle ( $BPEL \rightsquigarrow STS$ )  
implantation automatique ( $STS \rightsquigarrow BPEL$ )

- extension sémantique des interfaces ( $LTS \rightsquigarrow SIO LTS$ )
- calcul de clients «parfaits»
- produit sémantique et projection/pairs
- extension des clients parfaits  
par ajout de messages



# Composition

sélection - composition - réparation

# Sélection

## sur critères non-fonctionnels

- **découverte** : **base** de la composition
  - recherche de tous les services présentant une **caractéristique fonctionnelle** souhaitée
- **sélection** : complète la découverte
  - choix du(des) **meilleur(s)** service
  - selon des **critères non-fonctionnels**  
ici, aspects **transactionnels et qualité de service**
- aspects **transactionnels** mais **non QoS**  
[Zeng et al, 2005], [Jaeger et al, 2004], [Zhang et al, 2007]
- aspects **QoS** mais **non transactionnels**  
[Bhiri et al, 2005], [Montagut et al, 2006], [Li et al, 2007]



# Sélection

## sur critères non-fonctionnels

- **services**

- capacité**

- + propriété **transactionnelle** [pac]r?
  - + valeurs de **QoS**

- **besoin**

- workflow** abstrait (capacités)

- + propriété **transactionnelle** [ac]
  - + **poids** sur critères de **QoS**

- **instanciation** du workflow

- **propriété transactionnelle**

- par agrégation structurelle de celles des sous-services

- **max. de la satisfaction** utilisateur (local puis global)
  - par utilisation de techniques d'optimisation combinatoire

# Sélection

## sur critères non-fonctionnels

- **services**

- capacité**

- + propriété **trans**

- + valeurs de **QoS**

- **besoin**

- workflow** abstra

- + propriété **trans**

- + **poids** sur critèr

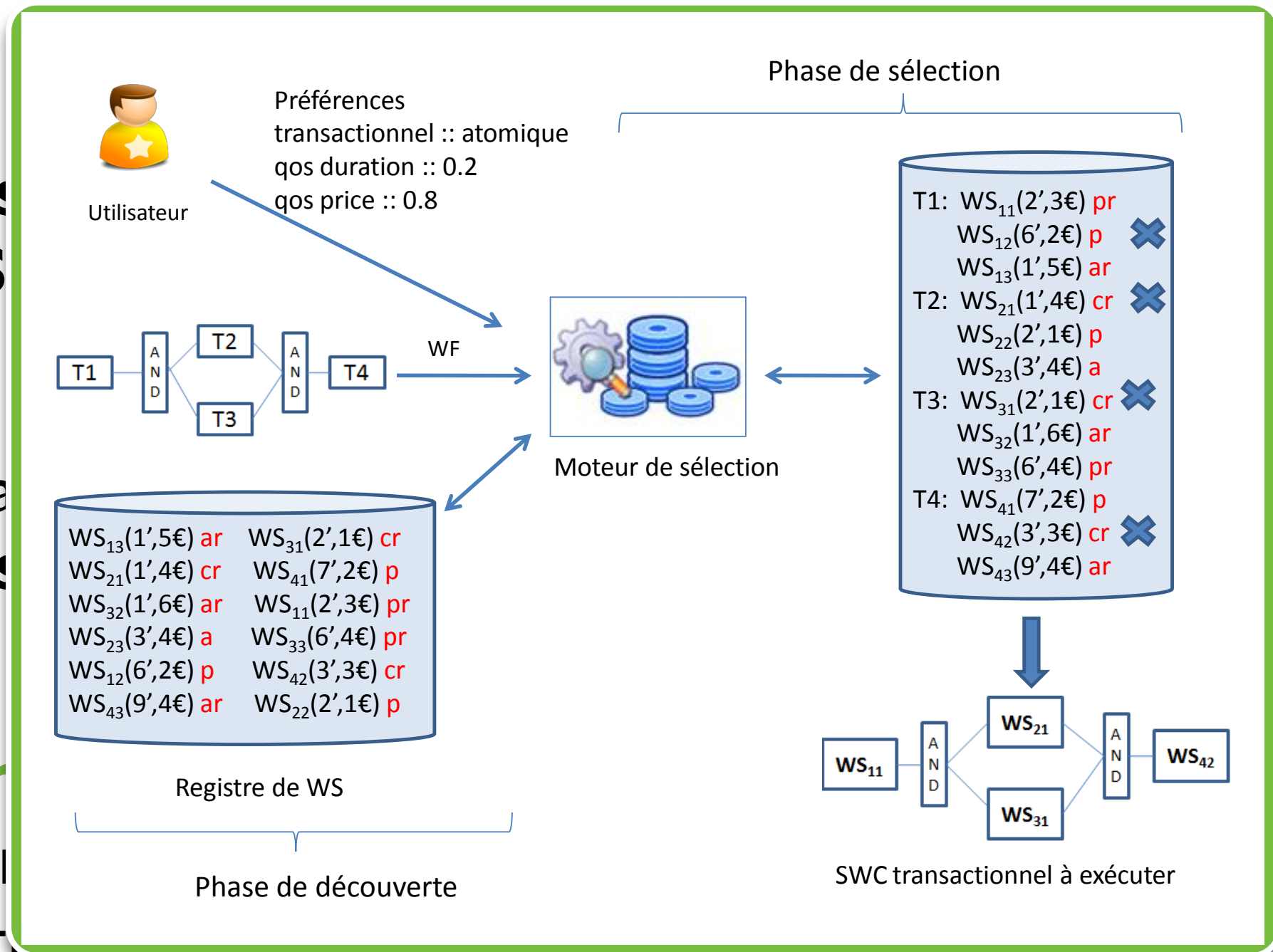
- **instanciation**

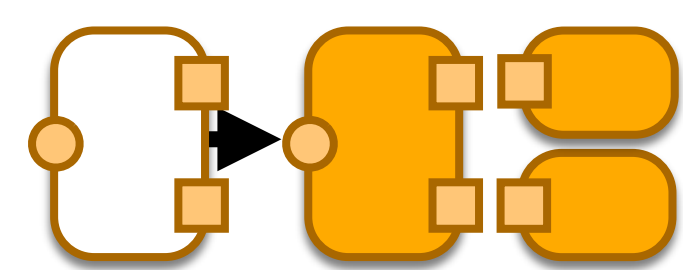
- **propriété tra**

- par agrégation str

- **max. de la satisfaction** utilisateur (local puis global)

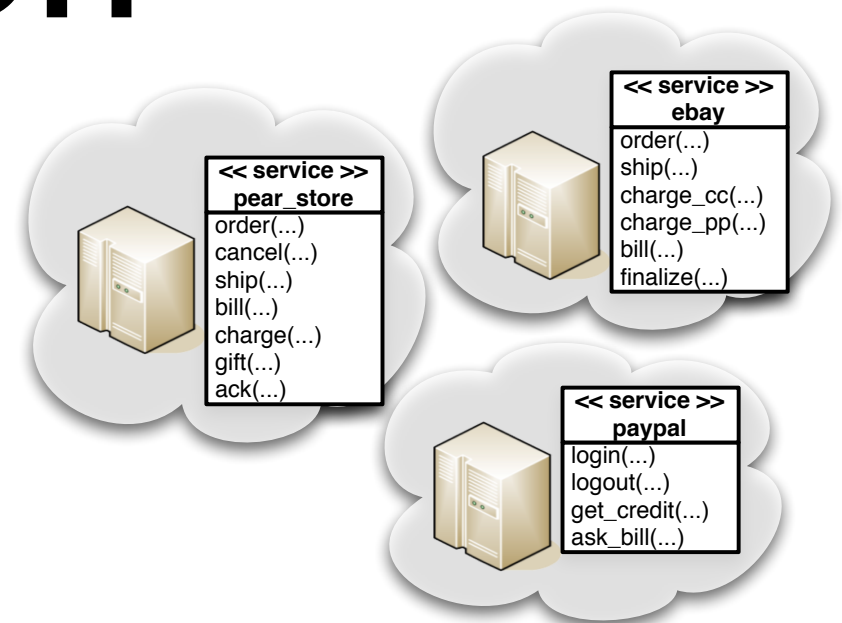
- par utilisation de techniques d'optimisation combinatoire

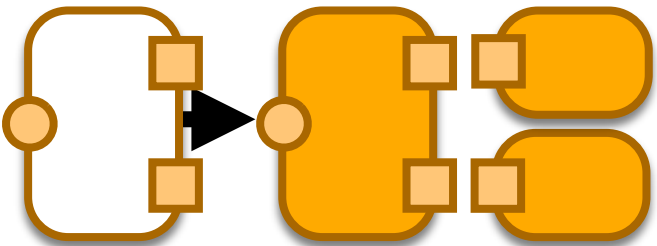




# Composition

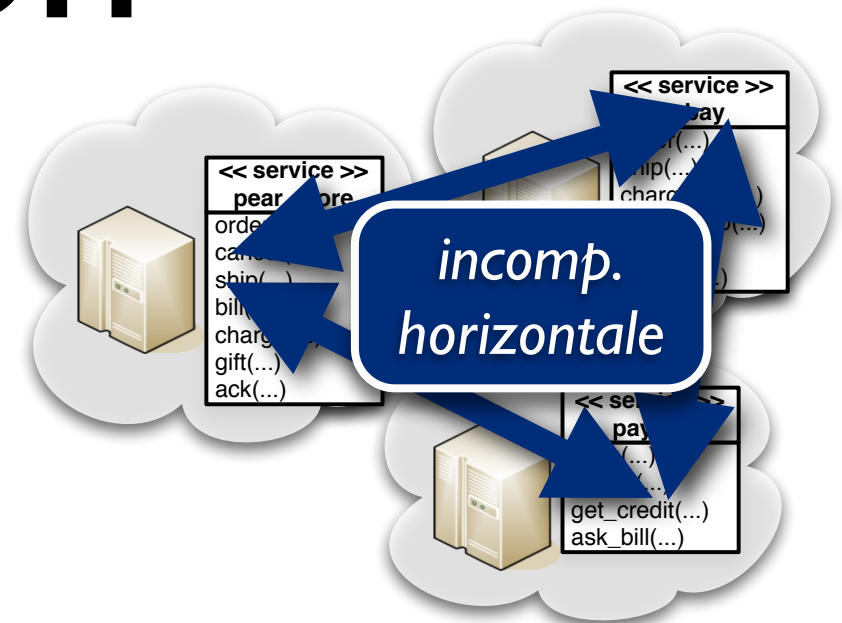
- aller **au-delà** de la résolution de blocages
- composer **automatiquement** des **services** à partir de **besoins** les deux ayant des **conversations**

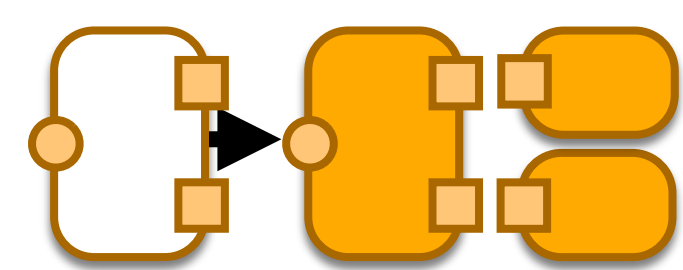




# Composition

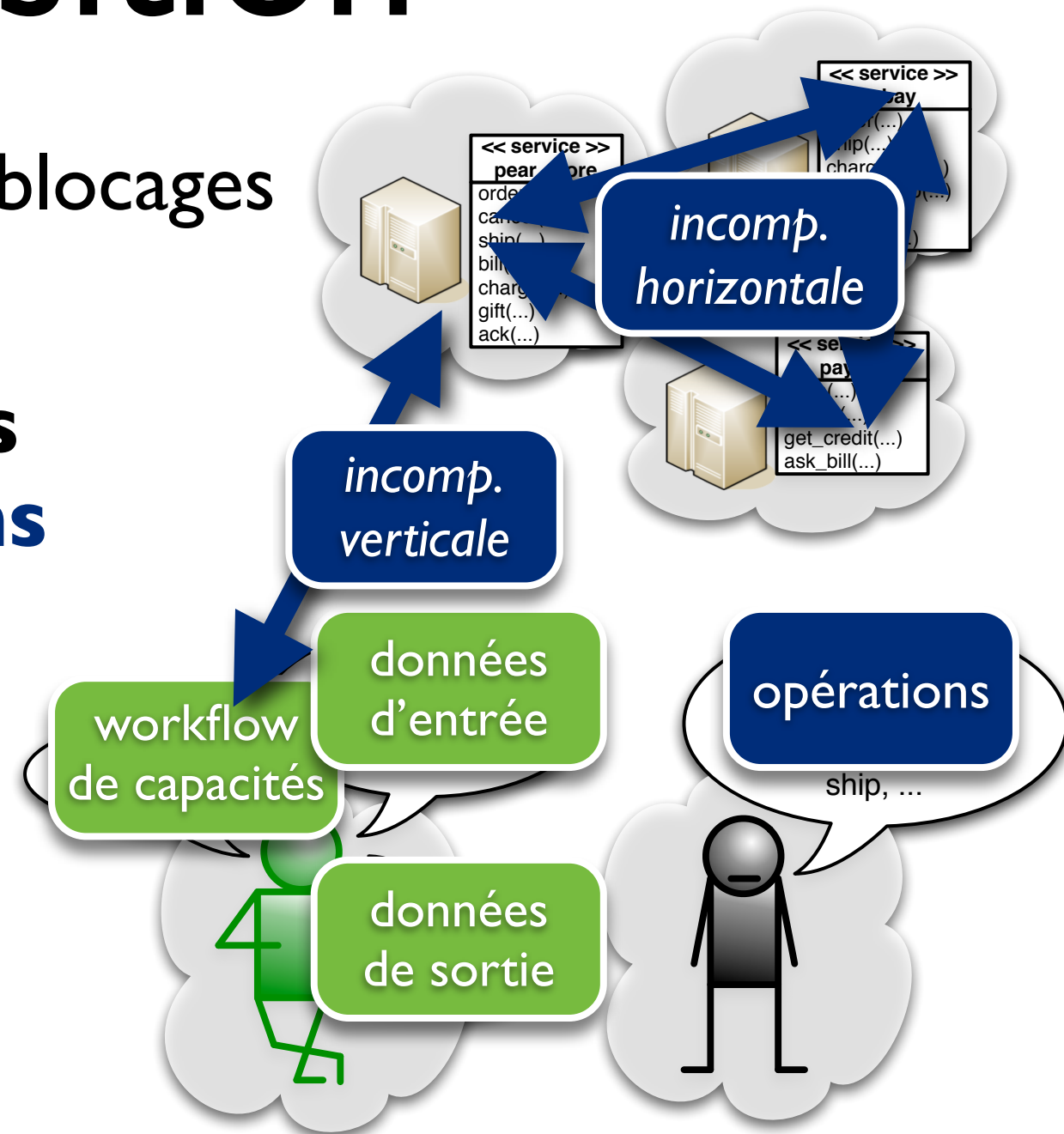
- aller **au-delà** de la résolution de blocages
- composer **automatiquement** des **services** à partir de **besoins** les deux ayant des **conversations**

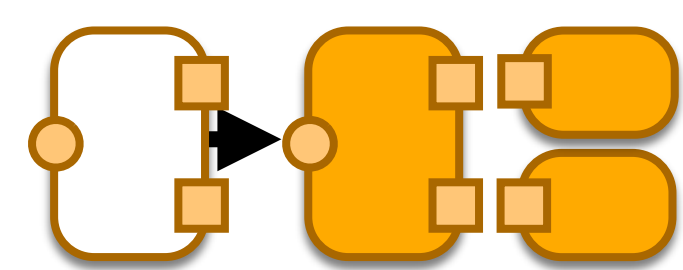




# Composition

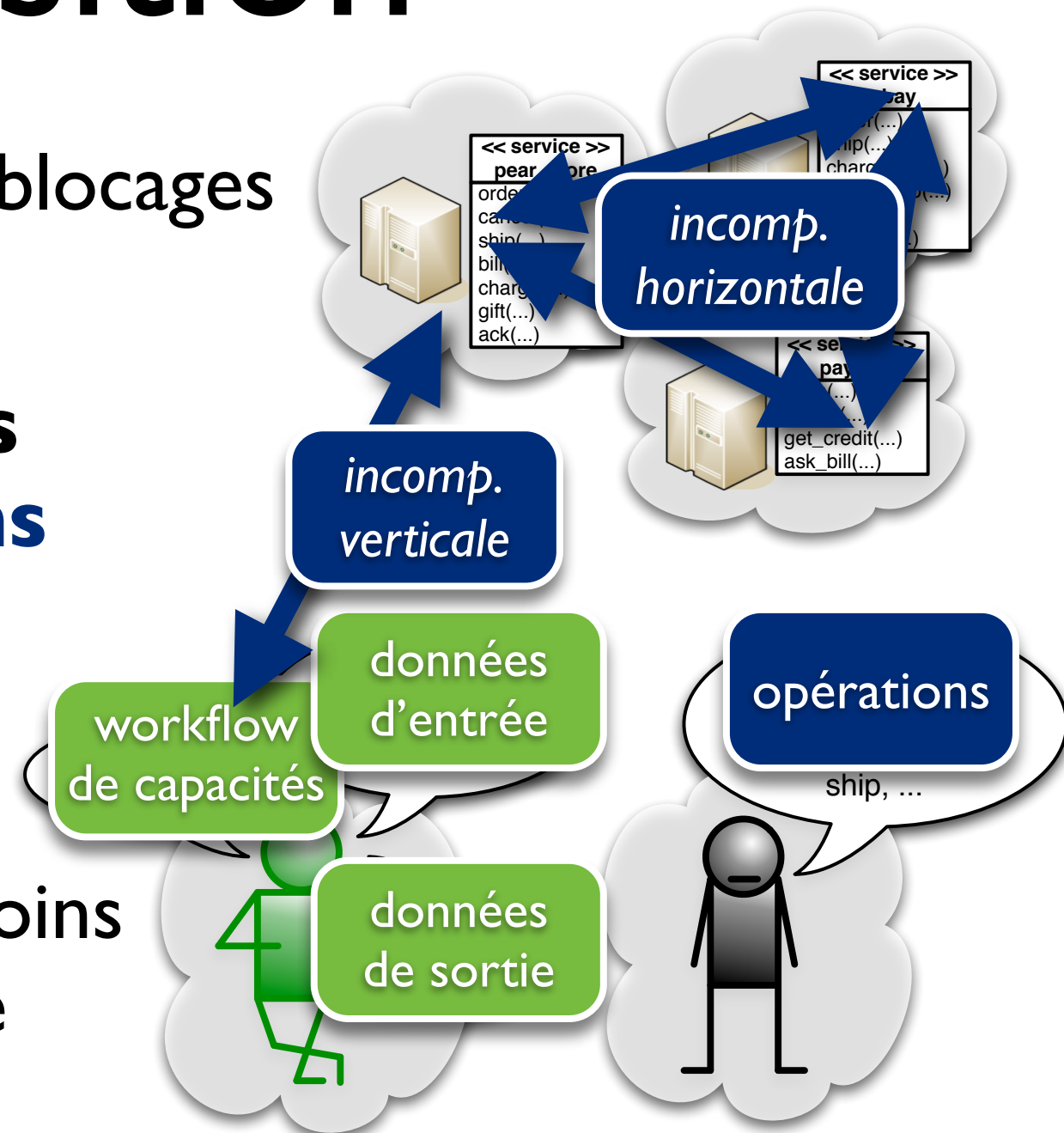
- aller **au-delà** de la résolution de blocages
- composer **automatiquement** des **services** à partir de **besoins** les deux ayant des **conversations**



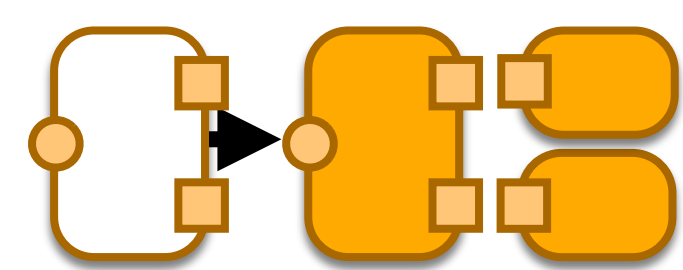


# Composition

- aller **au-delà** de la résolution de blocages
- composer **automatiquement** des **services** à partir de **besoins** les deux ayant des **conversations**
- **nombreuses** approches  
*[Marconi and Pistore, 2009]*
- **conversations** : services **et** besoins  
+ support **données** entrée/sortie  
+ support **capacités**  
*[Ben Mokhtar et al, 2007], [Bertoli et al, 2010]*
- incompatibilités **partiellement supportées**  
et hypothèse d'**intégration** de services **simple**  
*[Ben Mokhtar et al, 2007]*

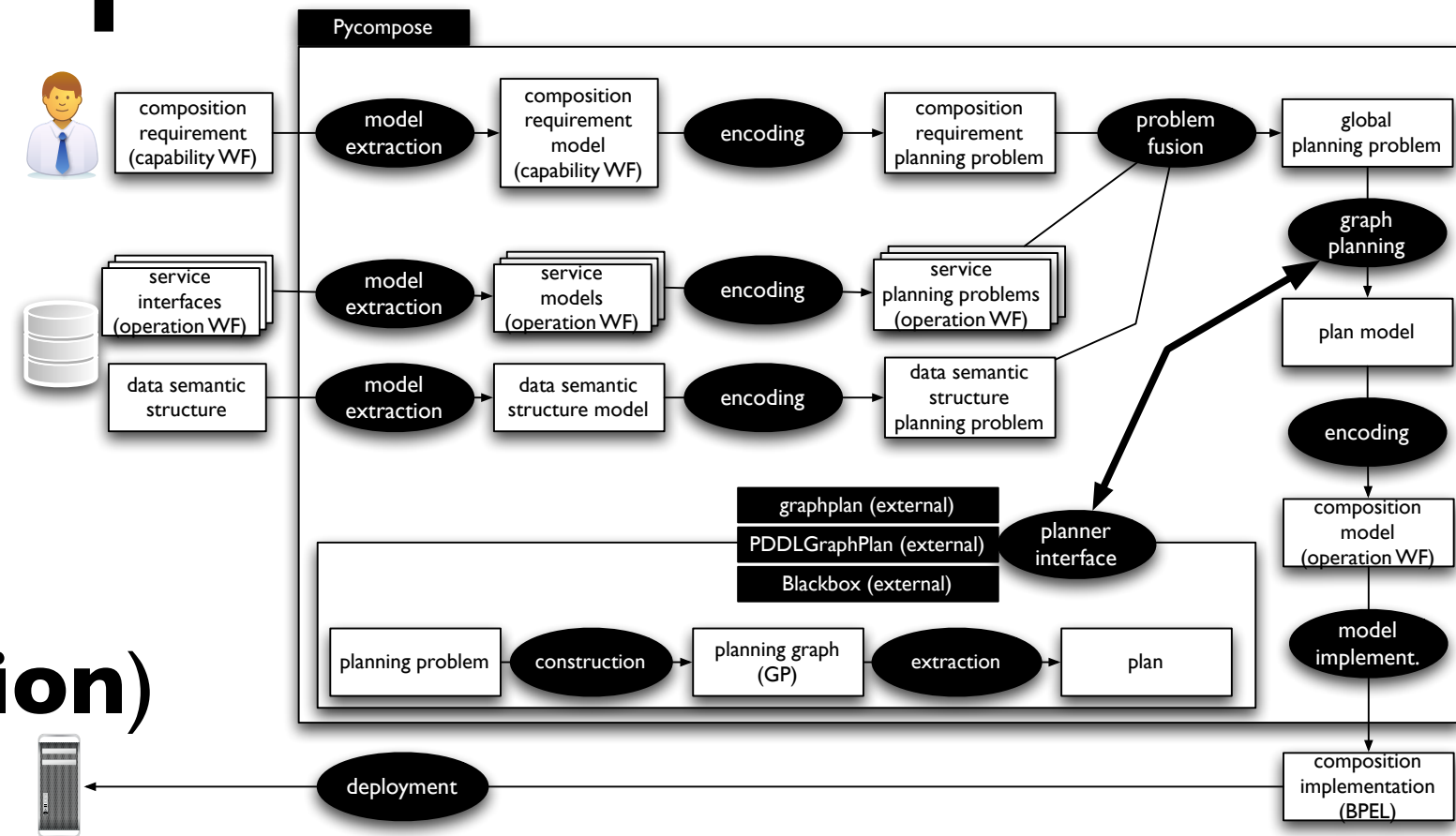






# Composition

- **typage sémantique** des opérations (entrées/sorties/capacité)
- **encodage** en actions propositionnelles (**graphe de planification**)



- de l'**adaptation de données**

casting / composition / décomposition

- des **conversations**

adaptation d'une transformation workflow  $\Rightarrow$  Petri net

[Kiepuszewski, 2003]

- des **opérations**

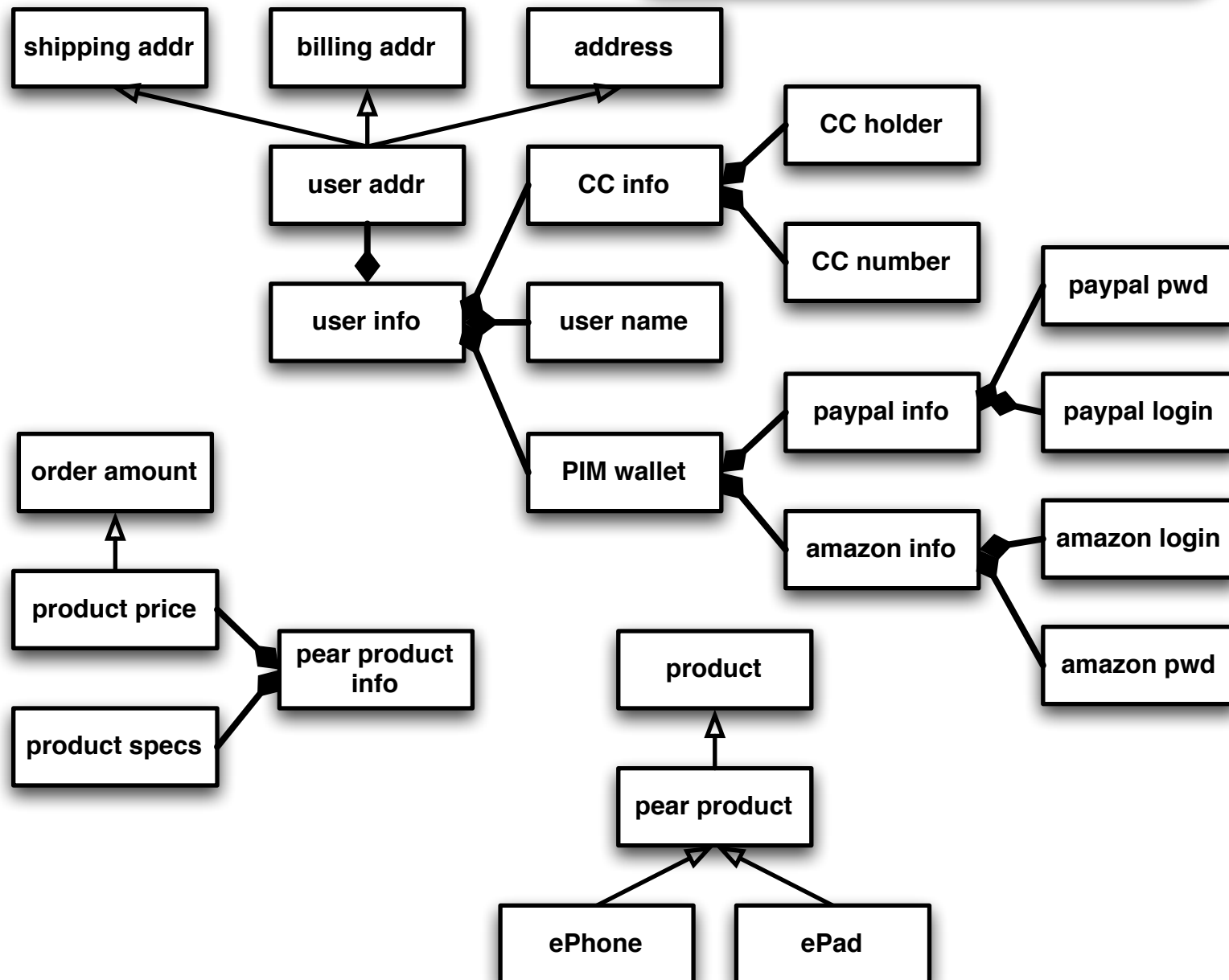
capacité autorisée + entrées  $\rightarrow$  capacité faite + sorties

# eShopping



# eShopping

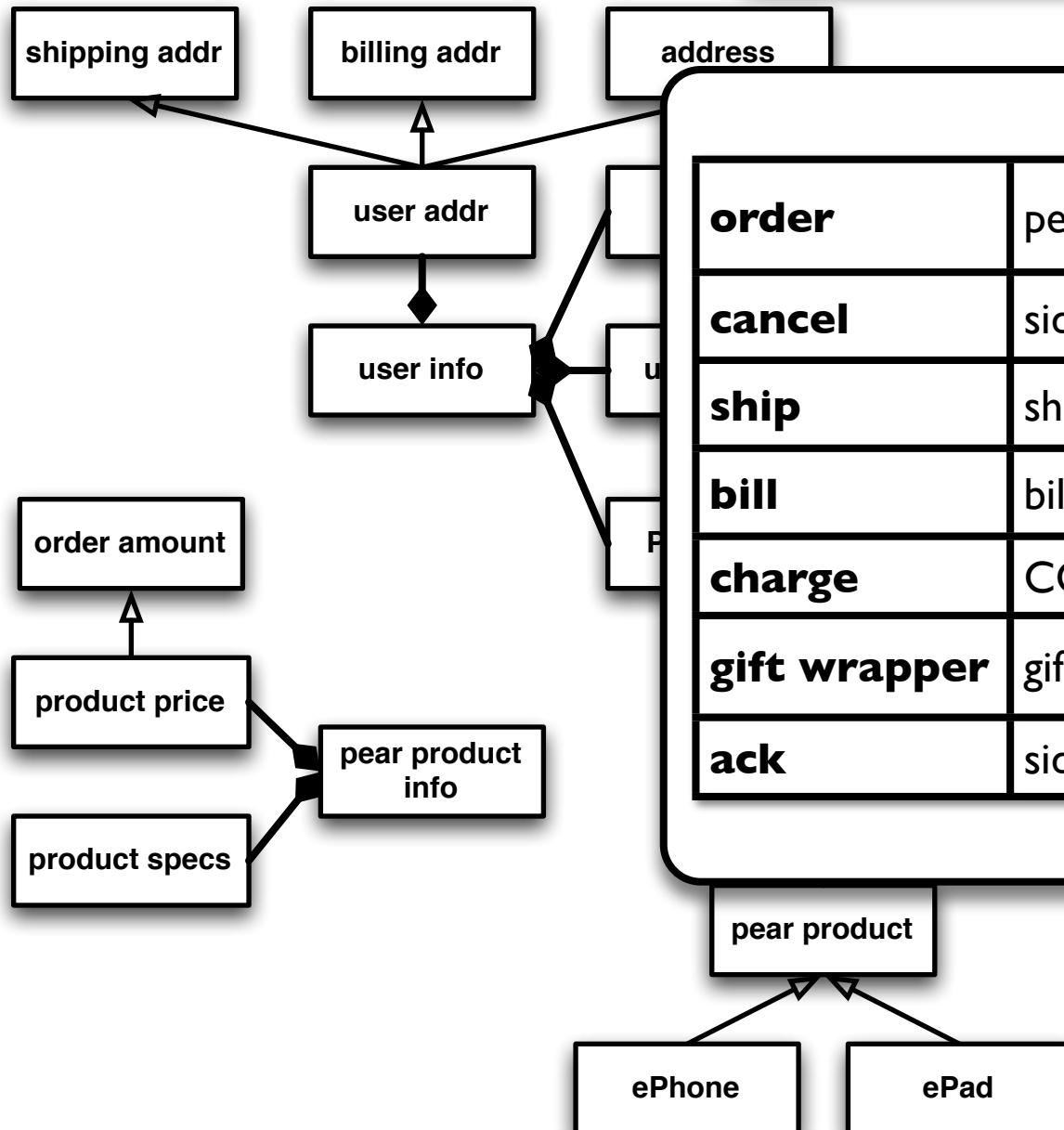
## semantic data description



# eShopping

semantic data description

service operations

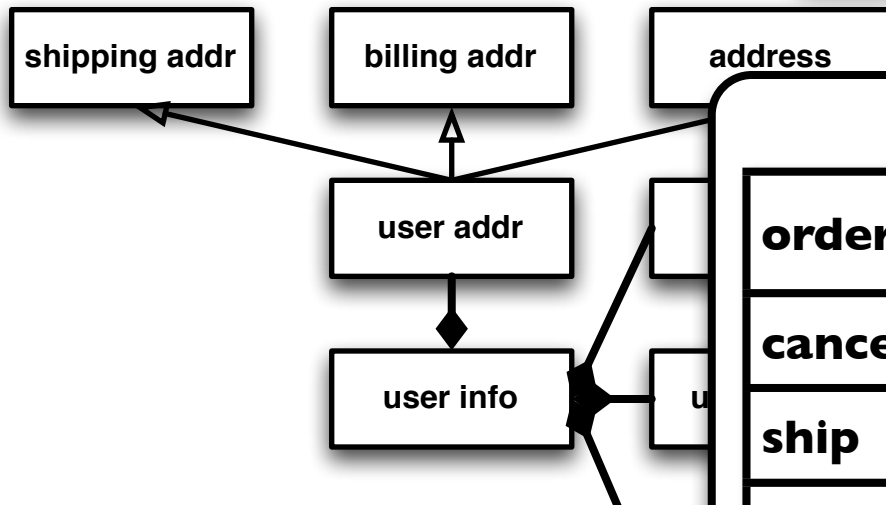


<b>order</b>	pear product	pear product info, sid <sup>PS</sup>	<b>PS</b>	product selection
<b>cancel</b>	sid <sup>PS</sup>			
<b>ship</b>	shipping addr, sid <sup>PS</sup>		<b>SS</b>	shipping setup
<b>bill</b>	billing addr, sid <sup>PS</sup>		<b>BS</b>	billing setup
<b>charge</b>	CC info, sid <sup>PS</sup>		<b>\$</b>	payment
<b>gift wrapper</b>	giftcode, sid <sup>PS</sup>		<b>\$</b>	payment
<b>ack</b>	sid <sup>PS</sup>	tracking number	<b>OF</b>	order finalization

# eShopping

semantic data description

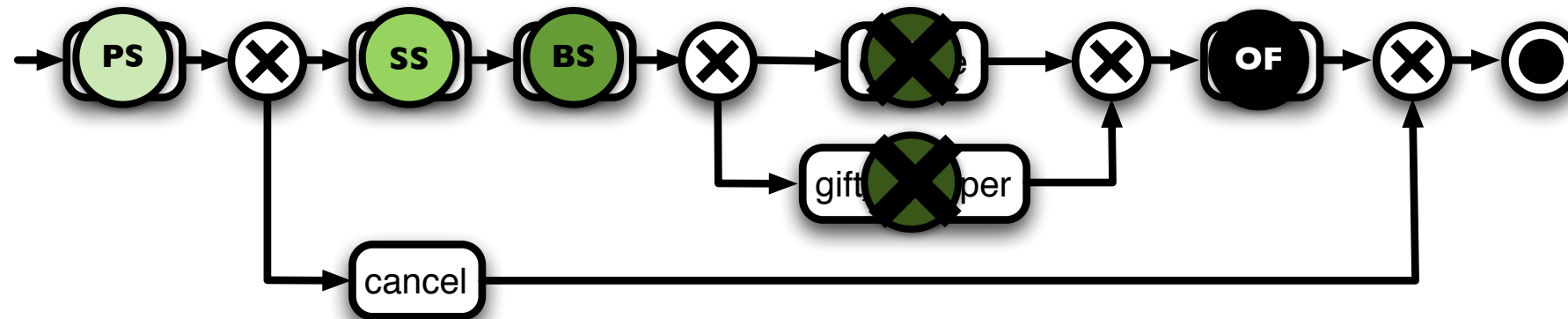
service operations



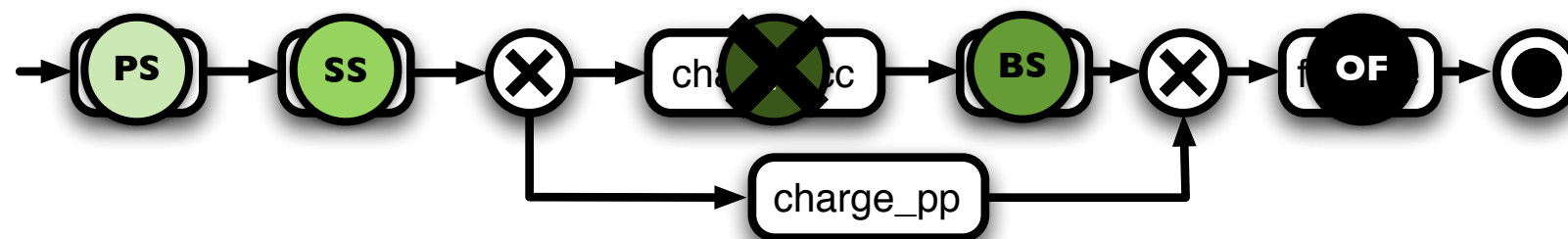
<b>order</b>	pear product	pear product info, sid <sup>PS</sup>	<b>PS</b>	product selection
<b>cancel</b>	sid <sup>PS</sup>			
<b>ship</b>	shipping addr sid <sup>PS</sup>		<b>SS</b>	shipping setup
			<b>BS</b>	billing setup
			<b>\$</b>	payment
			<b>\$</b>	payment
			<b>OF</b>	order finalization

service conversations

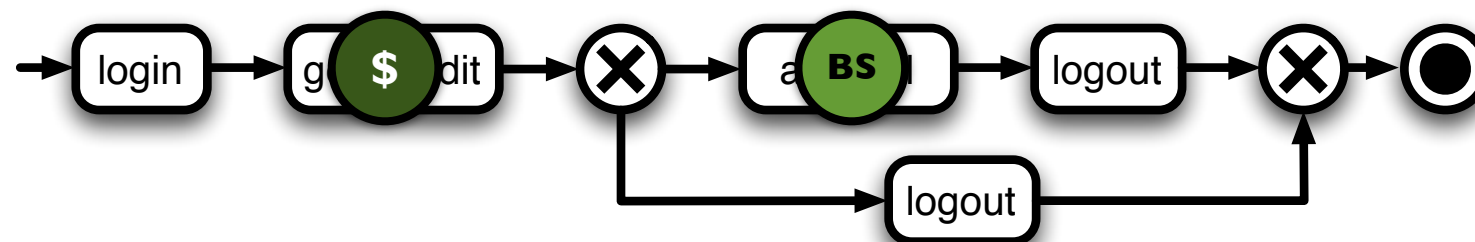
pear\_store



ebay



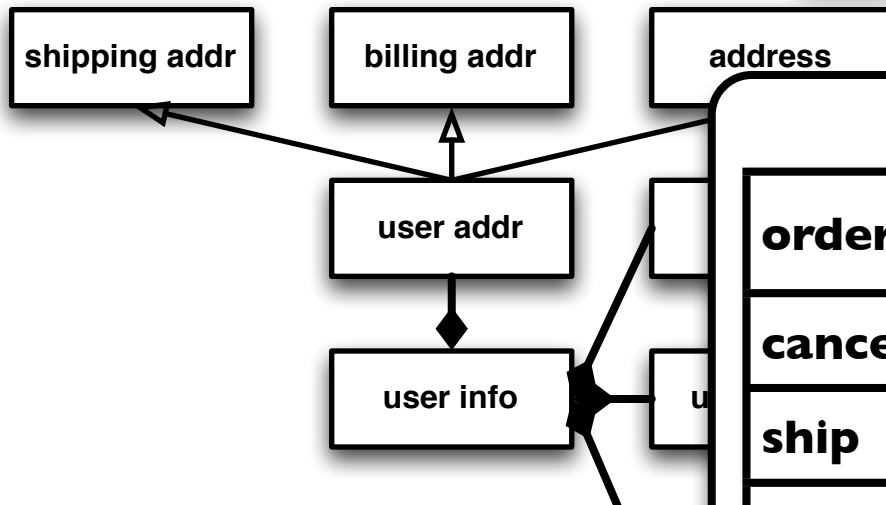
paypal



# eShopping

semantic data description

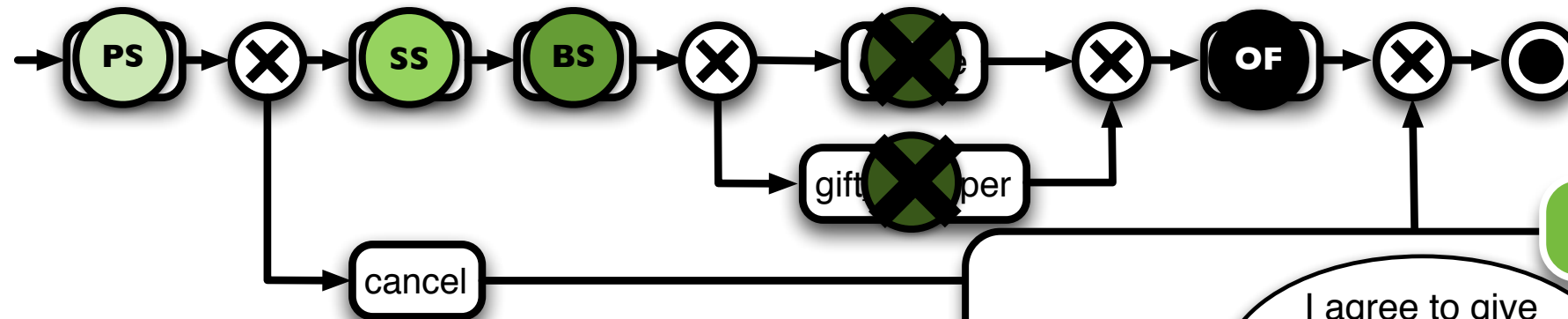
service operations



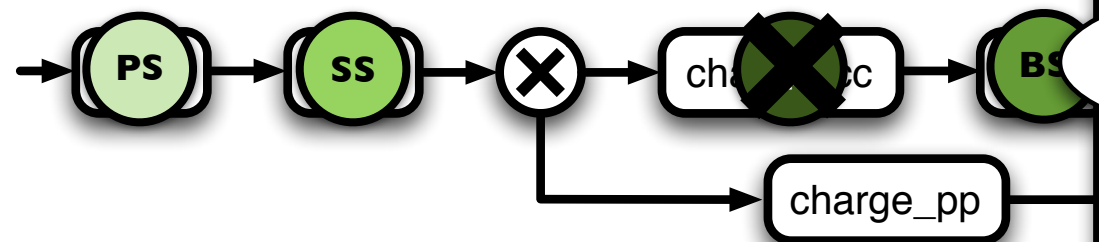
<b>order</b>	pear product	pear product info, sid <sup>PS</sup>	<b>PS</b>	product selection
<b>cancel</b>	sid <sup>PS</sup>			
<b>ship</b>	shipping addr sid <sup>PS</sup>		<b>SS</b>	shipping setup
			<b>BS</b>	billing setup
			<b>\$</b>	payment
			<b>\$</b>	payment
			<b>OF</b>	order finalization

service conversations

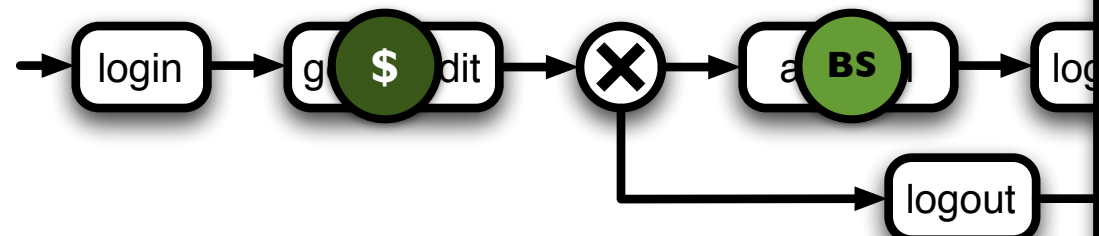
pear\_store



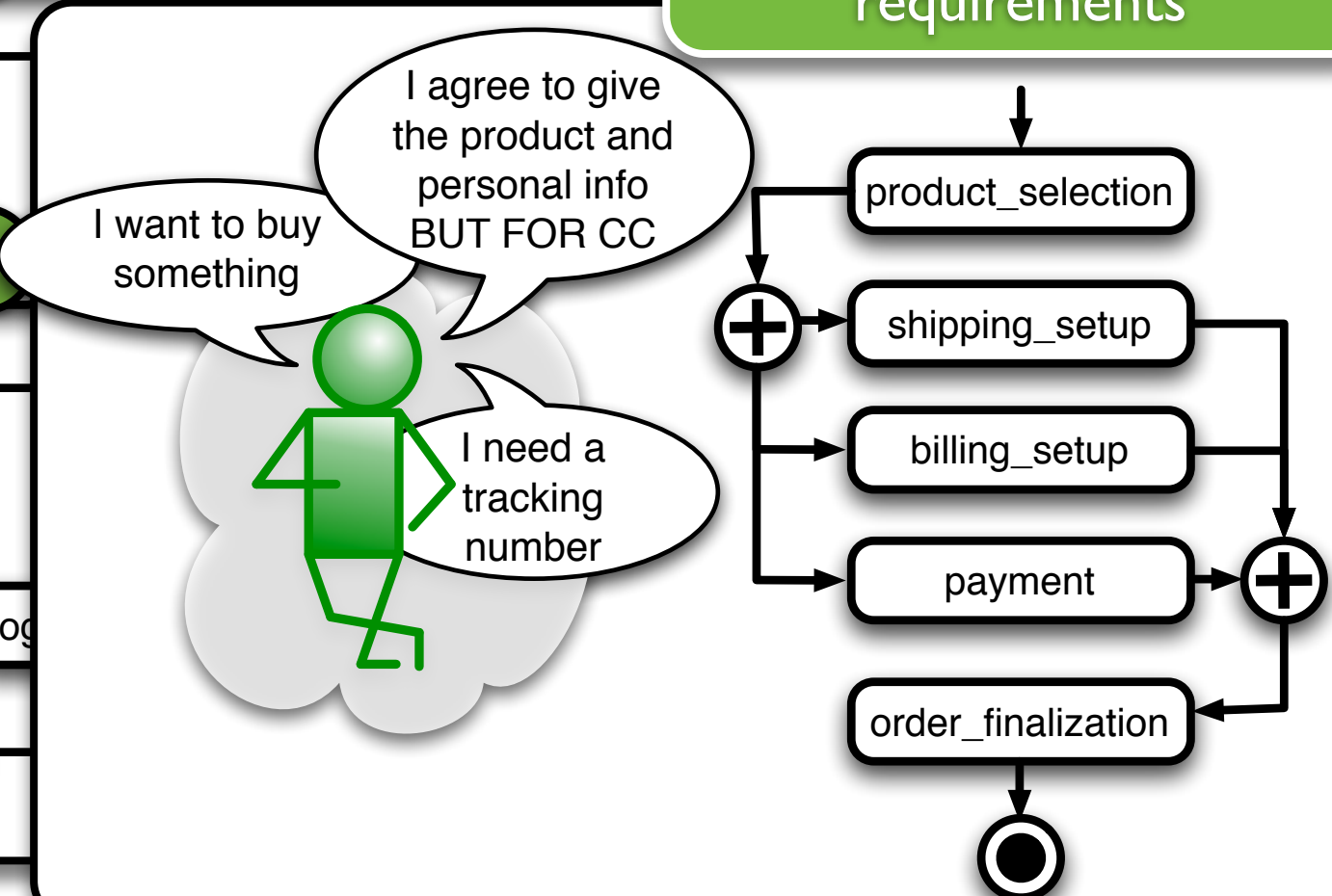
ebay



paypal

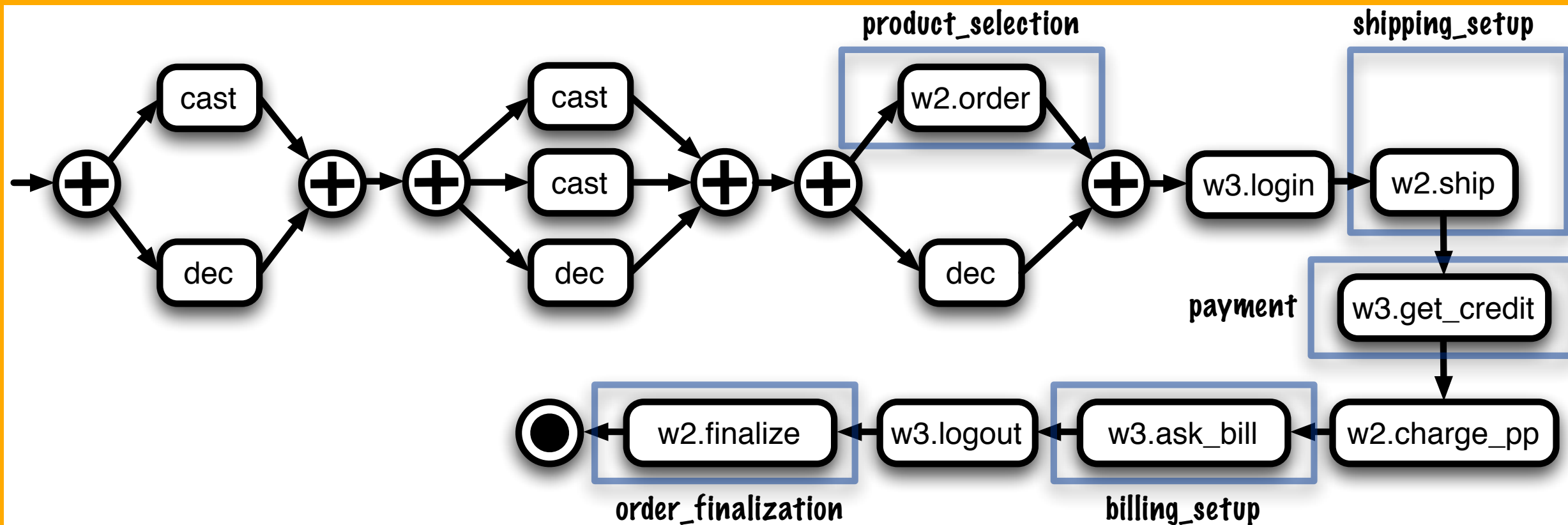


requirements

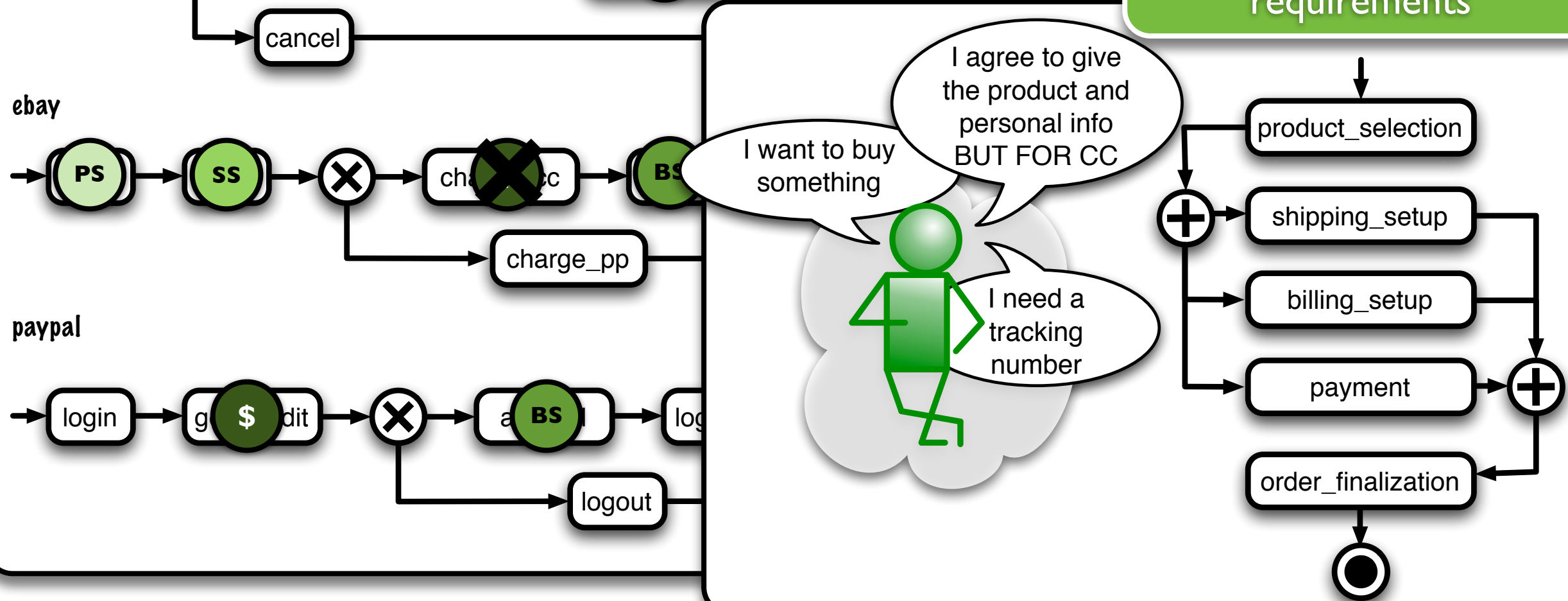


# eShopping

## semantic data description



## requirements



```

receive(user,op,{ePad,user_info}) ;
flow {
  [pear_product := cast(ePad)],
  [user_name := user_info.user_name; user_address := user_info.user_address;
   pim_wallet := user_info.pim_wallet]
};
flow {
  [product := cast(pear_product)],
  [shipping_addr := cast(user_address)],
  [paypal_info := pim_wallet.paypal_info; amazon_info := PIM_wallet.amazon_info]
};
flow {
  [{e_sessionid} := invoke(w2,order,{product})],
  [paypal_login := paypal_info.paypal_login; paypal_pwd := paypal_info.paypal_pwd]
};
{p_sessionid} := invoke(w3,login,{paypal_login,paypal_pwd}) ;
{order_amount} := invoke(w2,ship,{shipping_addr,e_sessionid}) ;
{paypal_trans_id} := invoke(w3,get_credit,{order_amount,p_sessionid}) ;
invoke(w2,charge_pp,{paypal_trans_id,e_sessionid}) ;
invoke(w3,ask_bill,{user_address,p_sessionid}) ;
invoke(w3,logout,{p_sessionid}) ;
{tracking_num} := invoke(w2,finalize,{e_sessionid}) ;
reply(user,op,{tracking_num});
  
```



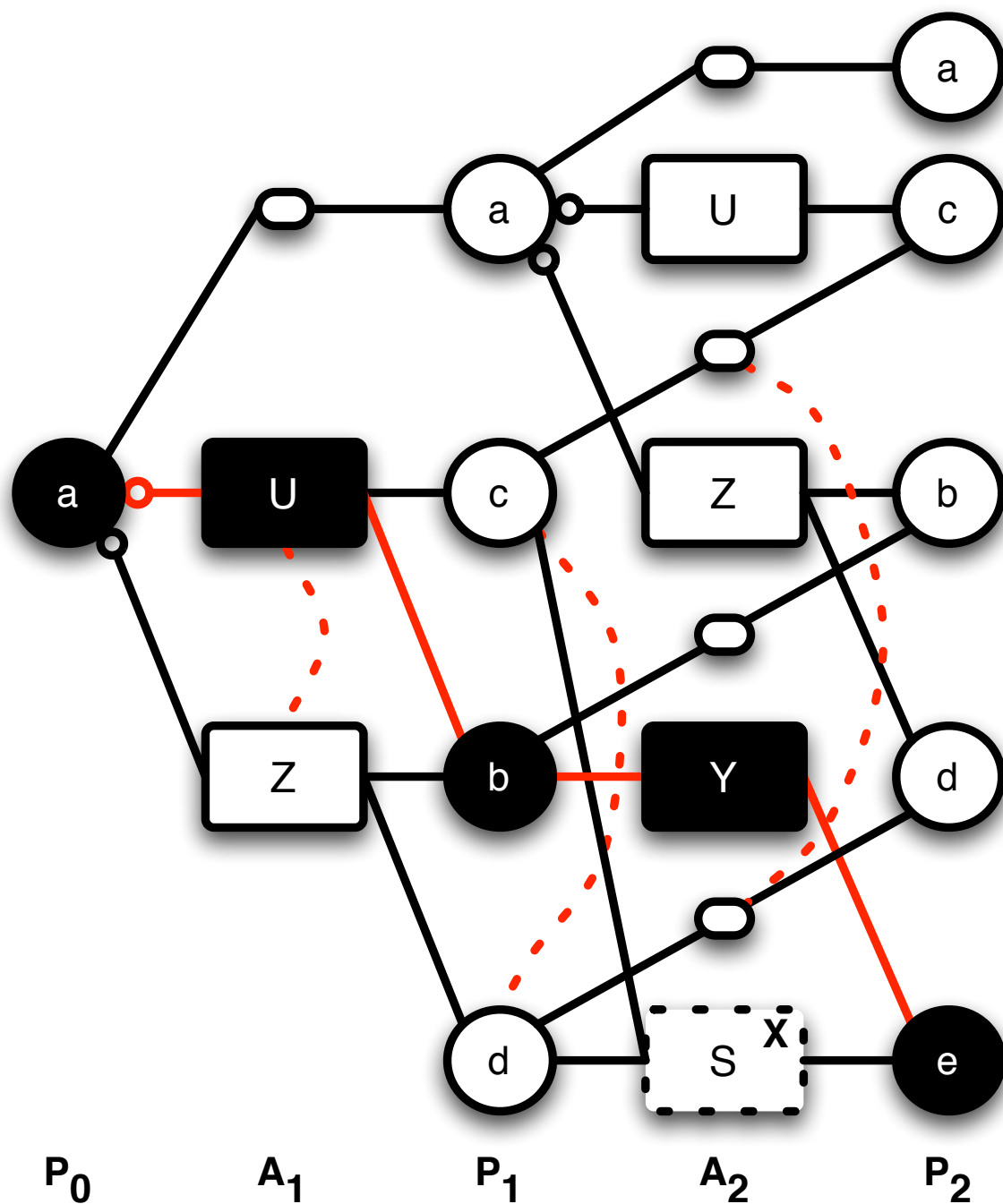


# Réparation

- les compositions de services sont soumises à un **monde changeant** :
  - services indisponibles (pannes, fautes, mobilité)
  - nouveaux services (mobilité)
  - besoins de l'utilisateur modifiés
- **remplacement**  
limité (I-I uniquement), problème des nouveaux besoins
- **re-composition / adaptation**  
potentiellement coûteux, perte de la solution initiale
- alternative : **réparation** ?



# Réparation



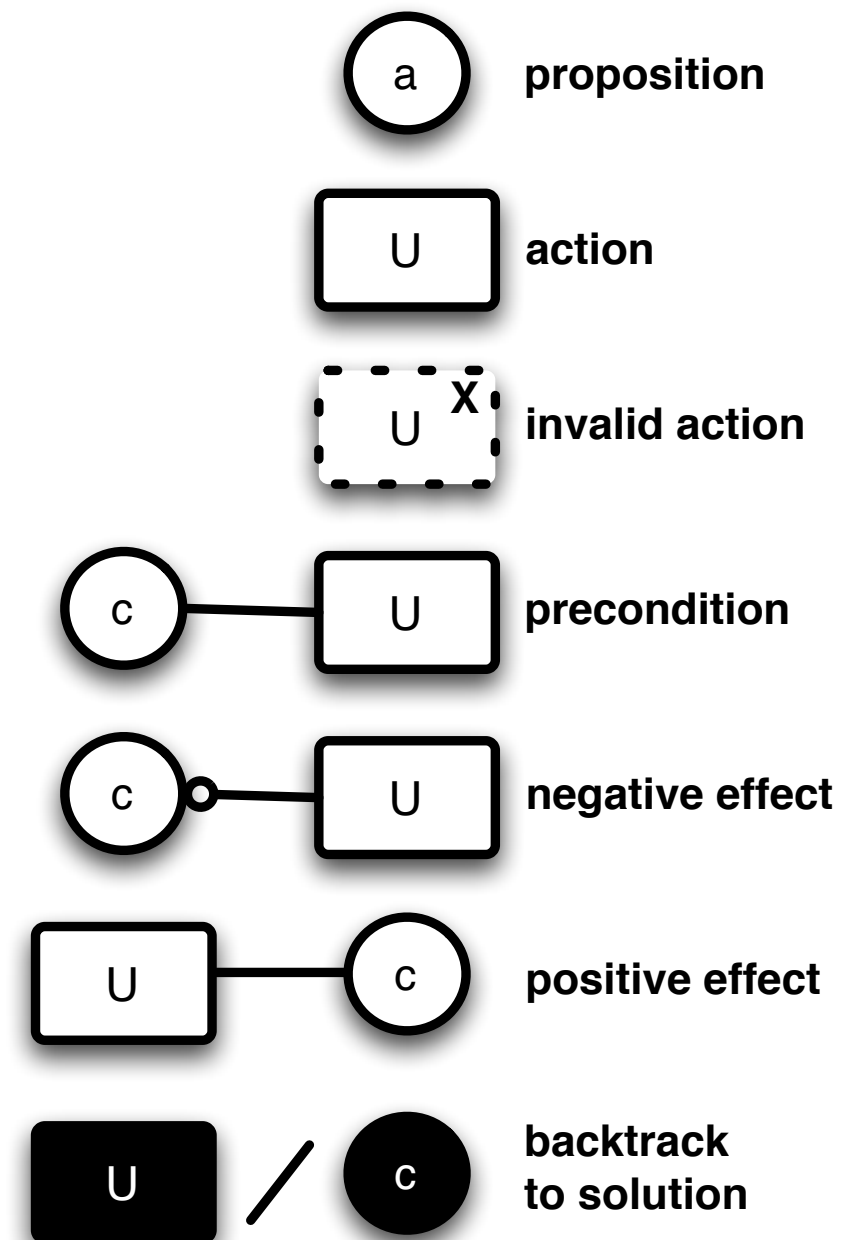
**Z:**  $\text{pre}=\{a\}$ ,  
 $\text{effect}^-=\{a\}$ ,  
 $\text{effect}^+=\{b,d\}$

**U:**  $\text{pre}=\{a\}$ ,  
 $\text{effect}^-=\{a\}$ ,  
 $\text{effect}^+=\{b,c\}$

**Y:**  $\text{pre}=\{b\}$ ,  
 $\text{effect}^-=\{\}$ ,  
 $\text{effect}^+=\{e\}$

**S:**  $\text{pre}=\{c,d\}$ ,  
 $\text{effect}^-=\{\}$ ,  
 $\text{effect}^+=\{e\}$

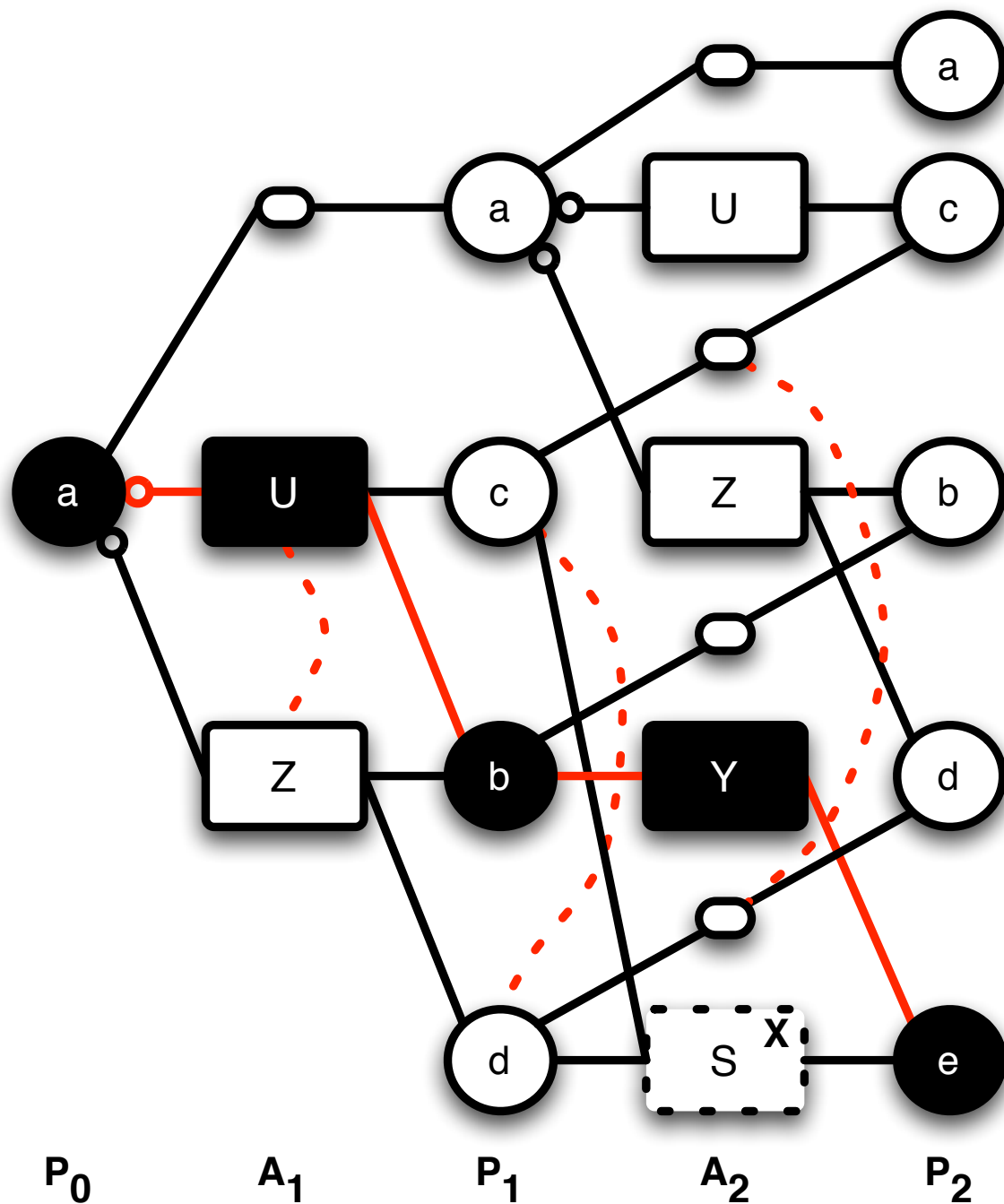
← layers







# Réparation



Z:  $pre=\{a\}$ ,  
 $effect^-=\{a\}$

solution:

$effect^-=\{a\}$ ,  
 $effect^+=\{b,c\}$

Y:  $pre=\{b\}$ ,  
 $effect^-=\{\}$ ,  
 $effect^+=\{e\}$

S:  $pre=\{c,d\}$ ,  
 $effect^-=\{\}$ ,  
 $effect^+=\{e\}$

$a$  proposition

$U;Y$

$U$  invalid action

$c$   $U$  precondition

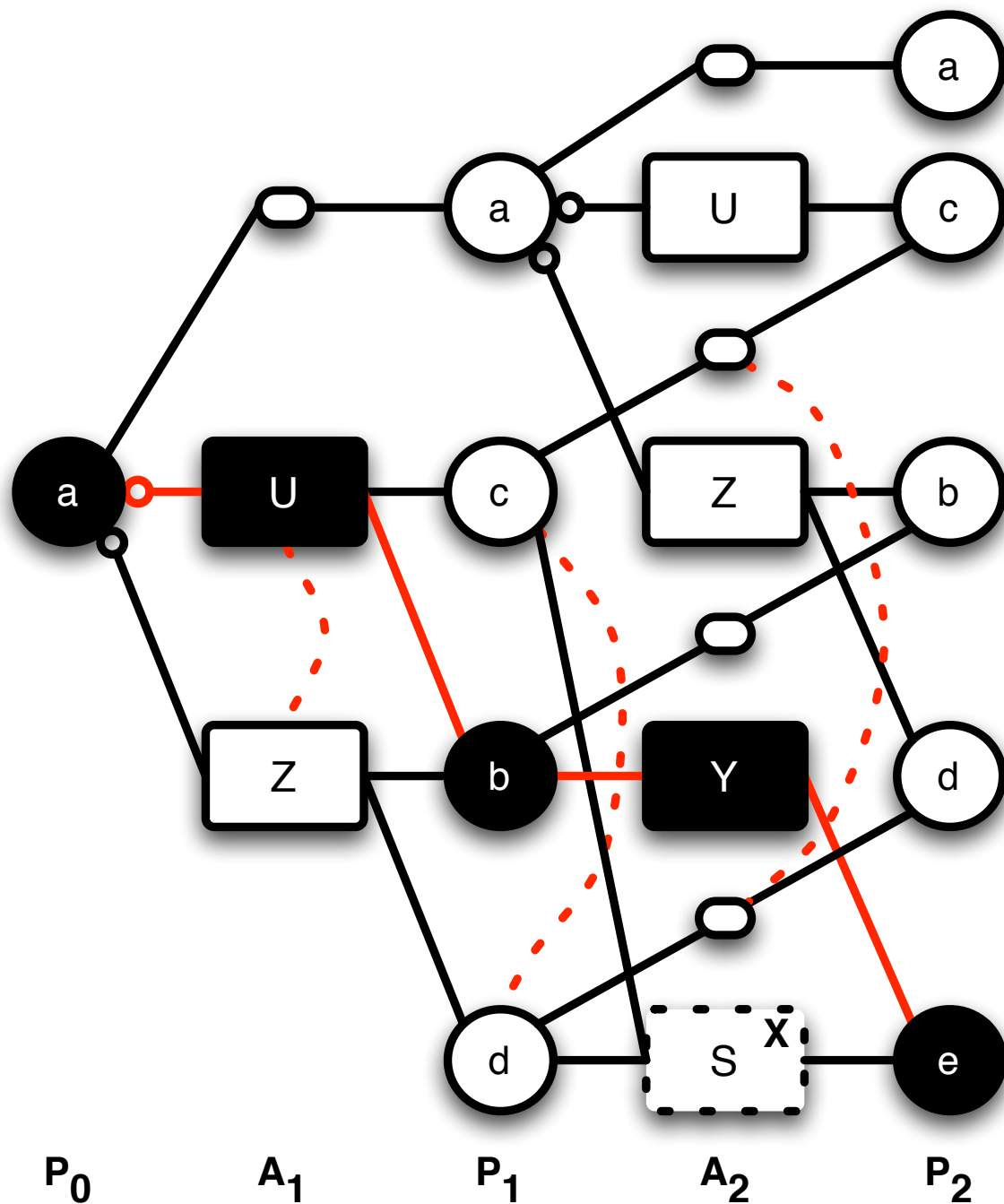
$c$   $U$  negative effect

$U$   $c$  positive effect

$U$  /  $c$  backtrack to solution

← layers

# Réparation



$Z: \text{pre}=\{a\},$   
 $\text{effect}^-=\{a\}$

$a$  proposition

solution:

$U;Y$

*et si U est indisponible ?*  
 autre solution:  $Z;Y$

$\text{effect}^-=\{\},$   
 $\text{effect}^+=\{e\}$

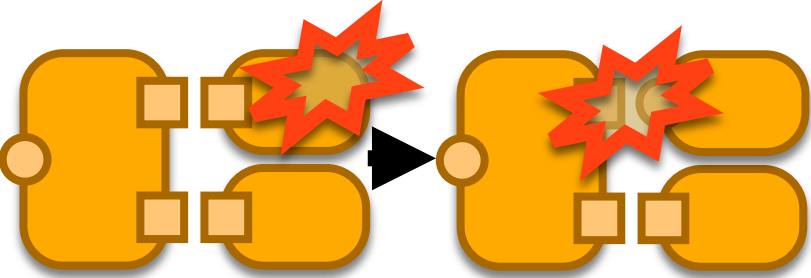
$S: \text{pre}=\{c,d\},$   
 $\text{effect}^-=\{\},$   
 $\text{effect}^+=\{e\}$

$c$  —  $U$  negative effect

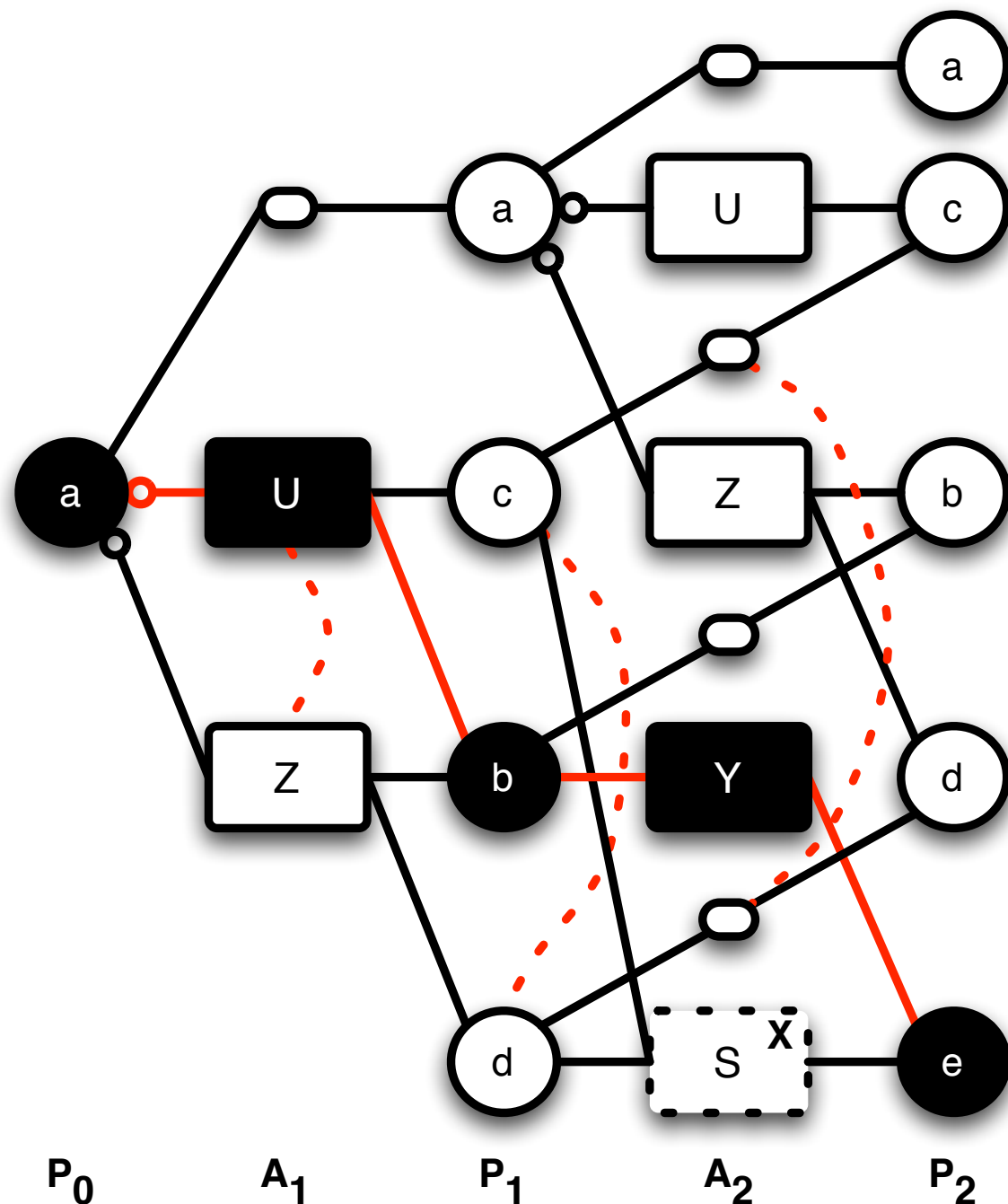
$U$  —  $c$  positive effect

$U$  /  $c$  backtrack to solution

← layers



# Réparation



Z:  $pre=\{a\}$ ,  
 $effect^+=\{a\}$

$\bigcirc a$  proposition

solution:

U;Y

*et si U est indisponible ?*  
 autre solution:

Z;Y

*et si Y est indisponible ?*  
 plus de solution mais une partie  
 du graphe est toujours valide

$effect^+=\{e\}$

U

/

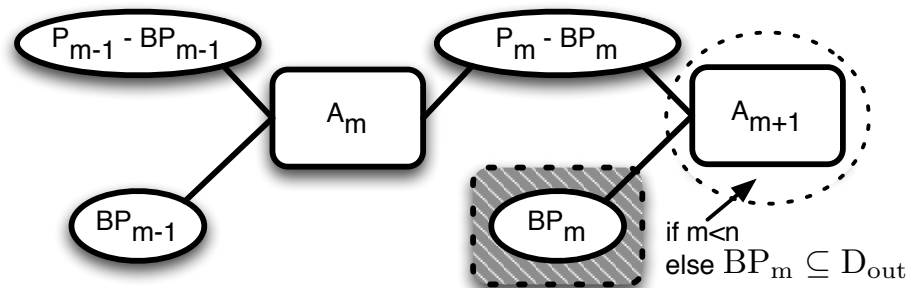
c

backtrack  
to solution

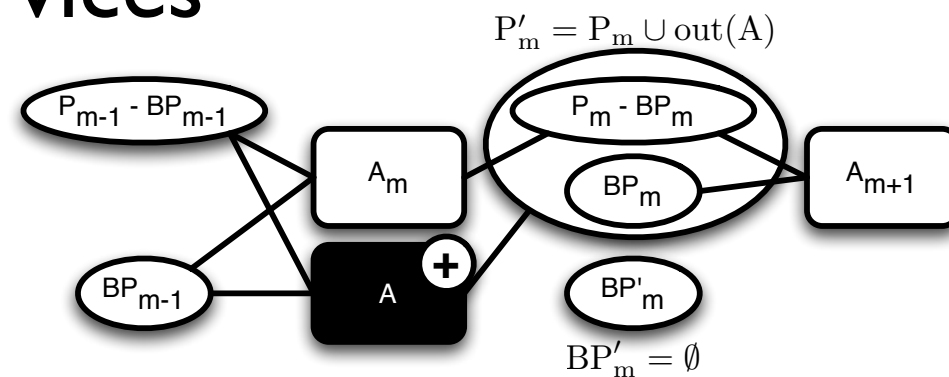


# Réparation

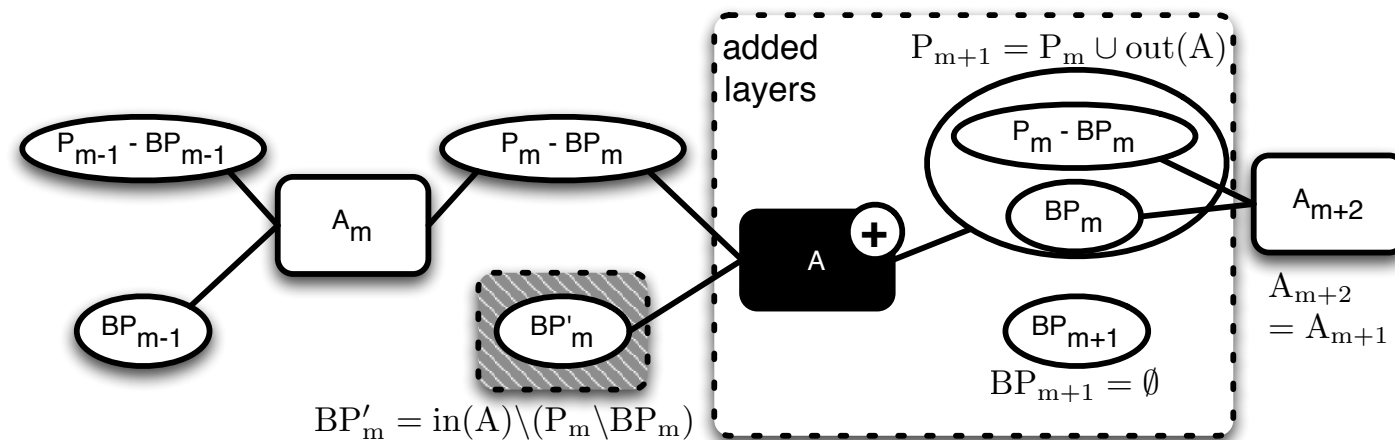
- calcul des **pré-conditions insatisfaites** (Broken Preconditions)
- **insertion** (heuristique) de nouveaux services dans le graphe de planification
  - éventuel ajout de niveaux
  - éventuelle suppression de services



(a)



(b)

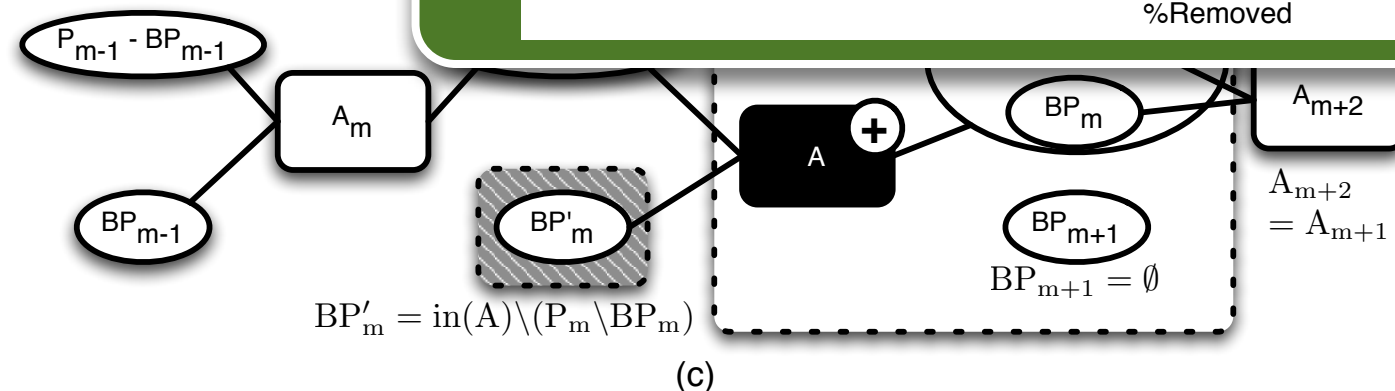
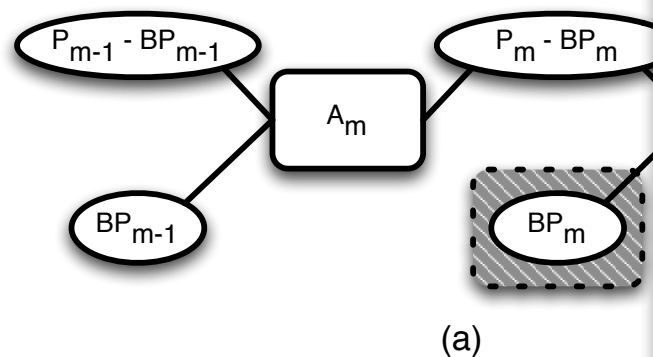


(c)

# Réparation

- calcul des **pré-conditions insatisfaites** (Broken Preconditions)

- insertion** (heuristique dans le graphe de plan)
  - éventuel ajout de niveau
  - éventuelle suppression



# Résultats

- techniques d'**adaptation comportementale**
  - semi-automatique (contrats)
  - automatique et distribuée
- techniques de **composition**
  - sélection sur critères non fonct.
  - composition comportementale et adaptative
  - réparation automatique
- **implémentation**
  - transformations de modèle
  - langage d'orchestration léger
- **évaluation** de performance
- **prototypes**
- **publications**
  - 2 HdR
  - 1 chapitre de livre
  - 6 articles de revues internat.  
IEEE Trans. Software Engineering (2),  
IEEE Trans. Service Computing,  
Int. J. of Web Service Research,  
Int. J. of Web and Grid Services,  
J. of Concur. and Comp.: Pract. and Exp.
  - 18 articles de conf. internat.  
dont 3 @ ICSOC et 3 @ ICWS («A»)

ANR PERSO - appel JCJC 2007-2010  
<http://anr-perso.ibisc.fr/>

```
graph TD; A[ANR PERSO - appel JCJC 2007-2010  
http://anr-perso.ibisc.fr/] --> B(Colloque STIC 2010); B --> C[ANR PIMI - appel VERSO 2010-2013];
```

Colloque  
STIC 2010

ANR PIMI - appel VERSO 2010-2013