

Turning workflow upside down

Um contributo para a discussão no CEO

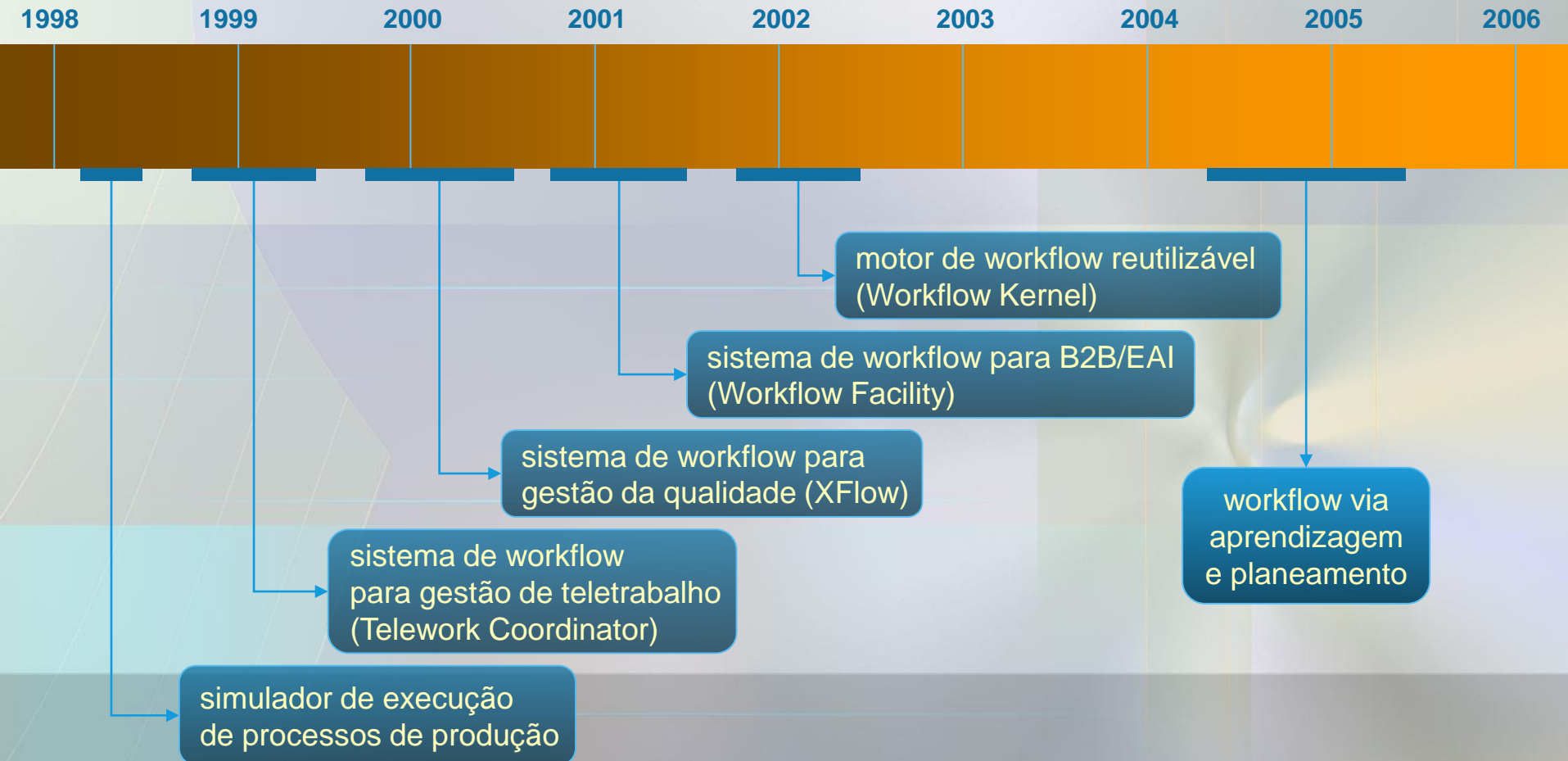
diogo.ferreira@tagus.ist.utl.pt

Agenda

- Part I:
 - rethinking workflow
- Part II:
 - enter CEO
 - workflow upside down
 - plano de acção

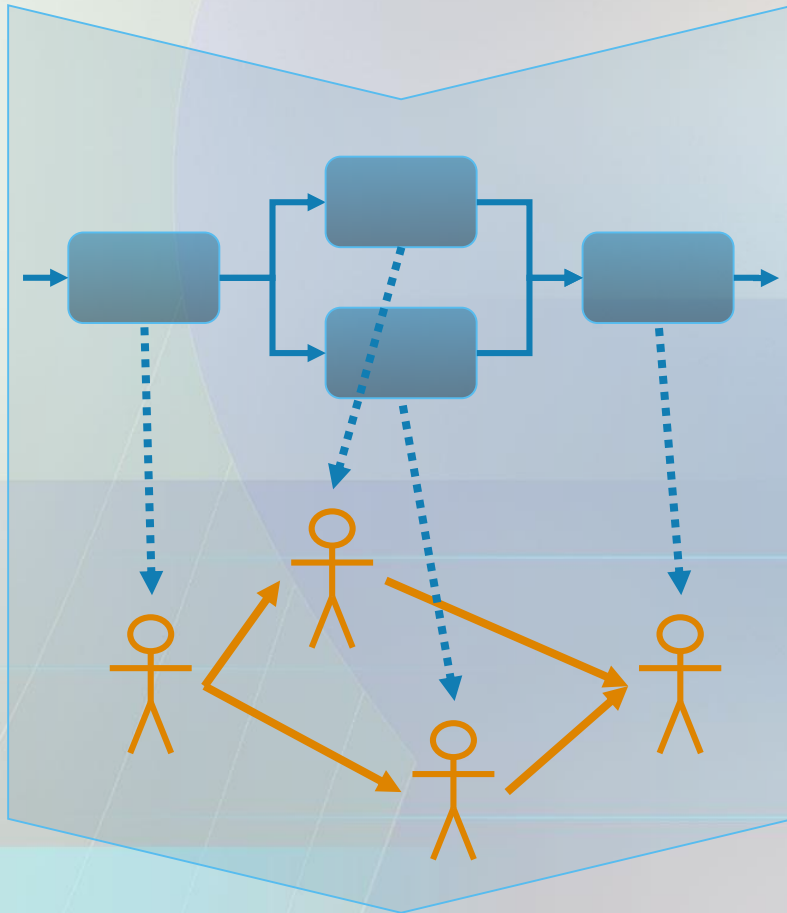
Background info

Timeline

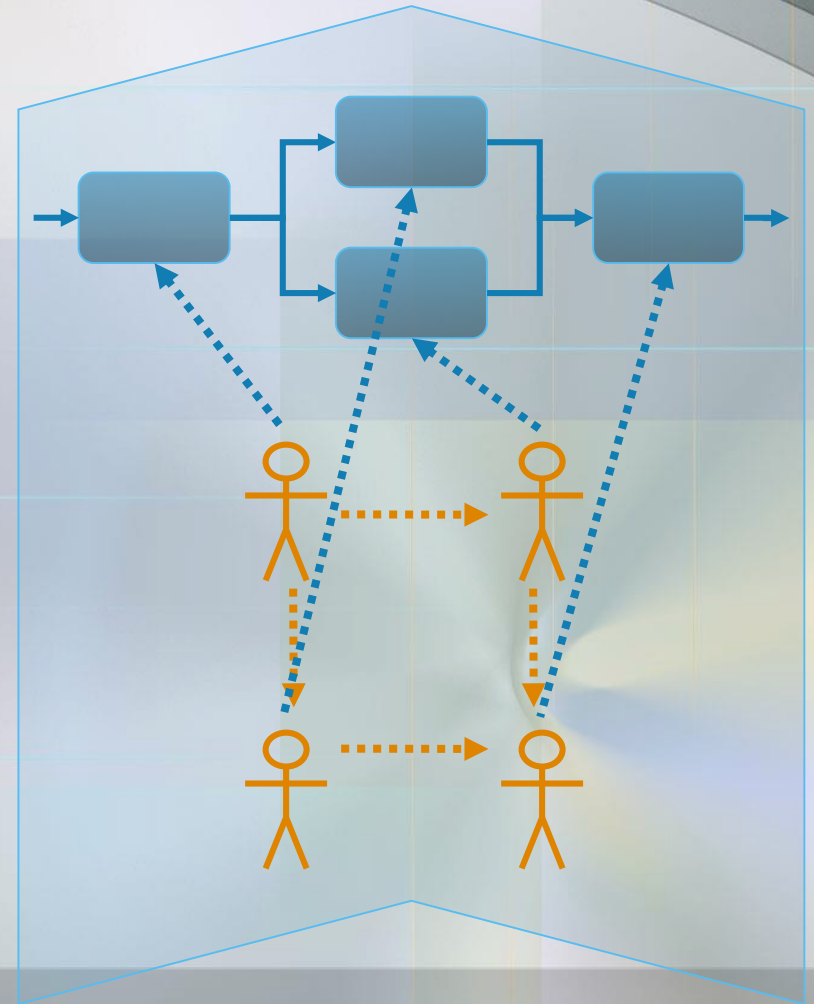


Part I: Rethinking Workflow

Thinking about workflow management

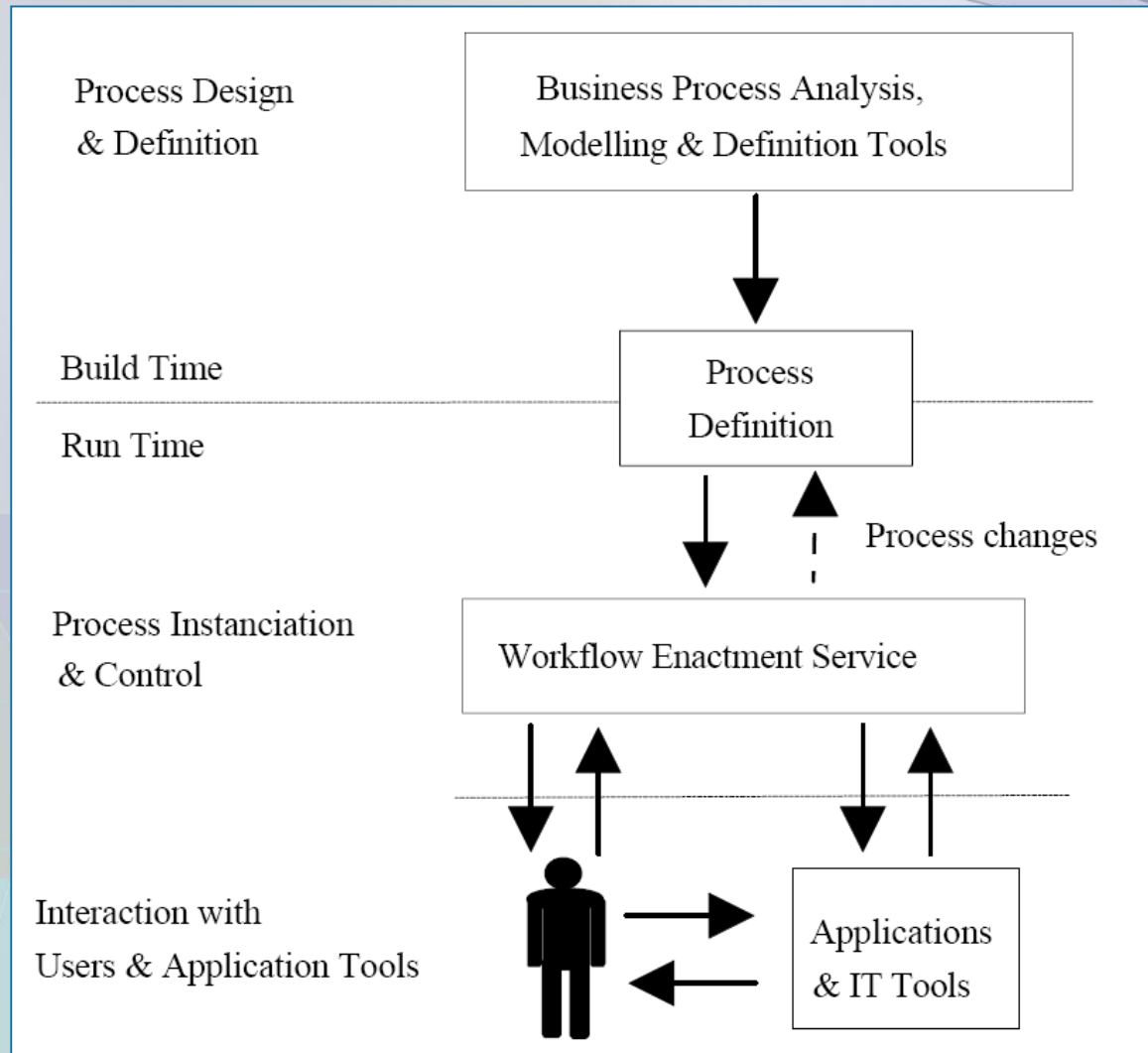


top-down



bottom-up

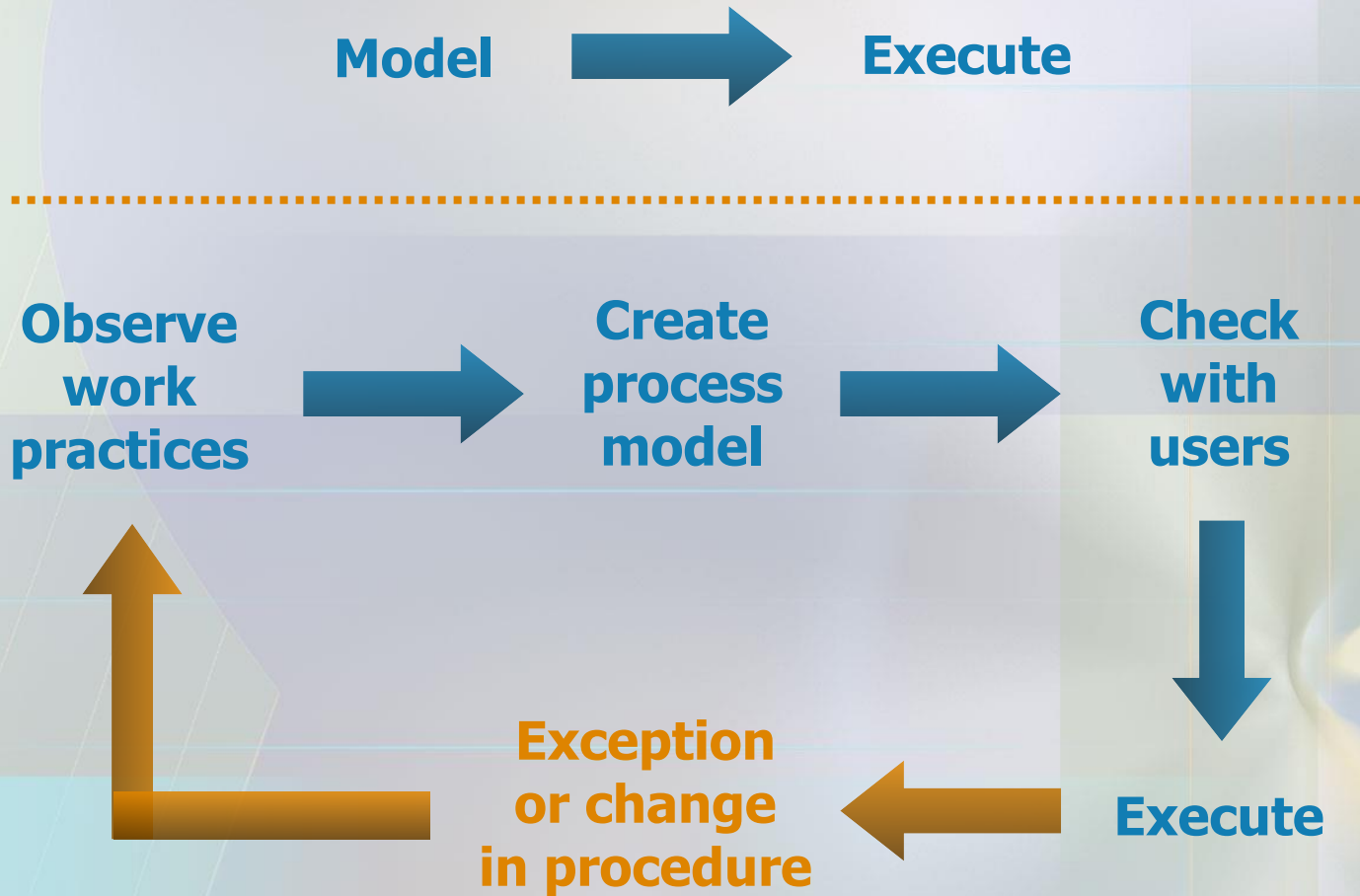
Thinking about workflow management



Tacit knowledge

- (Stenmark, 2001) argues that
 - much of organizational work relies on tacit knowledge
 - formal job descriptions are seldom enough to account for the actions we perform during a working day
 - much of our daily activity is governed by professional interests that are tacit in nature

Workflow with tacit knowledge



Learning workflow activities

`repair_vehicle(C, V)`

Preconditions: `damaged(V) , assessed(C)`

Effects: `+repaired(V) , -damaged(V)`

- in general, users cannot accurately describe the preconditions and effects
- they will be able to say if they can perform the activity or not, given an input state
 - positive vs. negative examples

Inductive Logic Programming

Training examples

(+) daughter(mary,ann)
 (+) daughter(eve,tom)
 (-) daughter(tom,ann)
 (-) daughter(eve,ann)

Background knowledge

parent(ann,mary)	female(ann)
parent(ann,tom)	female(mary)
parent(tom,eve)	female(eve)
parent(tom,ian)	

daughter(X,Y) ← true

daughter(X,Y) ←
X = Y

daughter(X,Y) ←
female(X)

daughter(X,Y) ←
parent(X,Y)

daughter(X,Y) ←
parent(X,Z)

daughter(X,Y) ←
female(X)
female(Y)

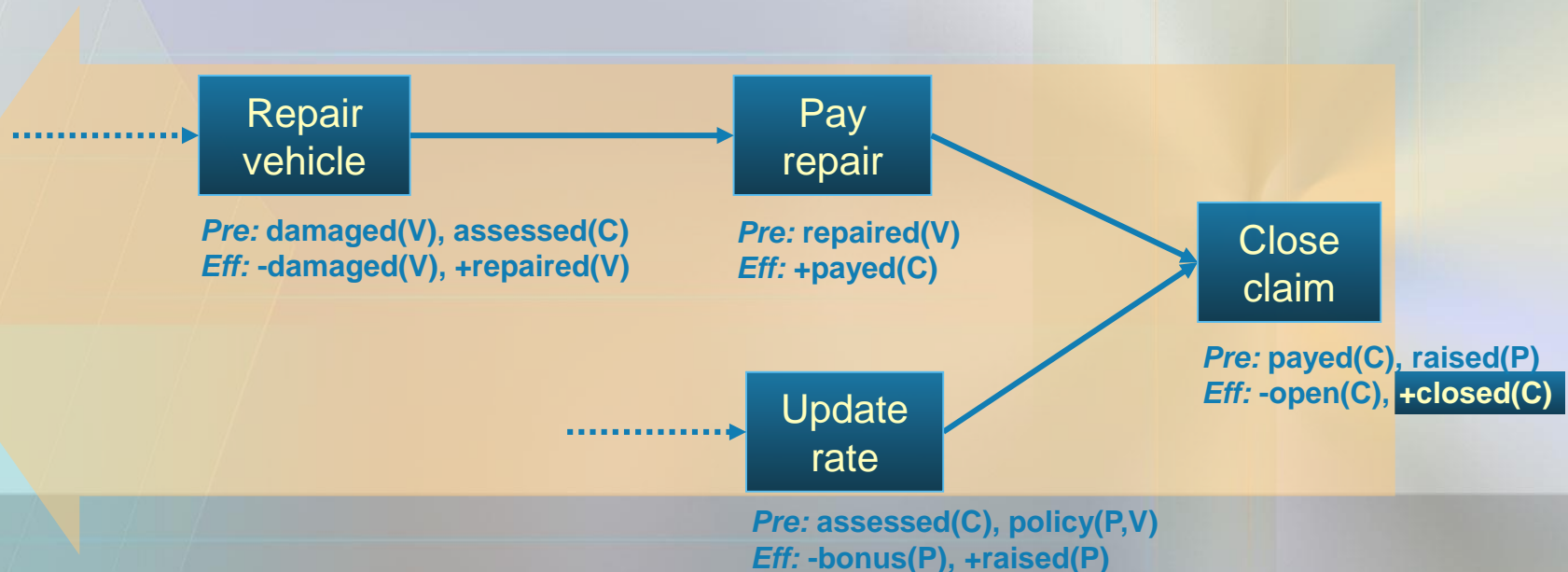
daughter(X,Y) ←
female(X)
parent(Y,X)

Learning workflow activities

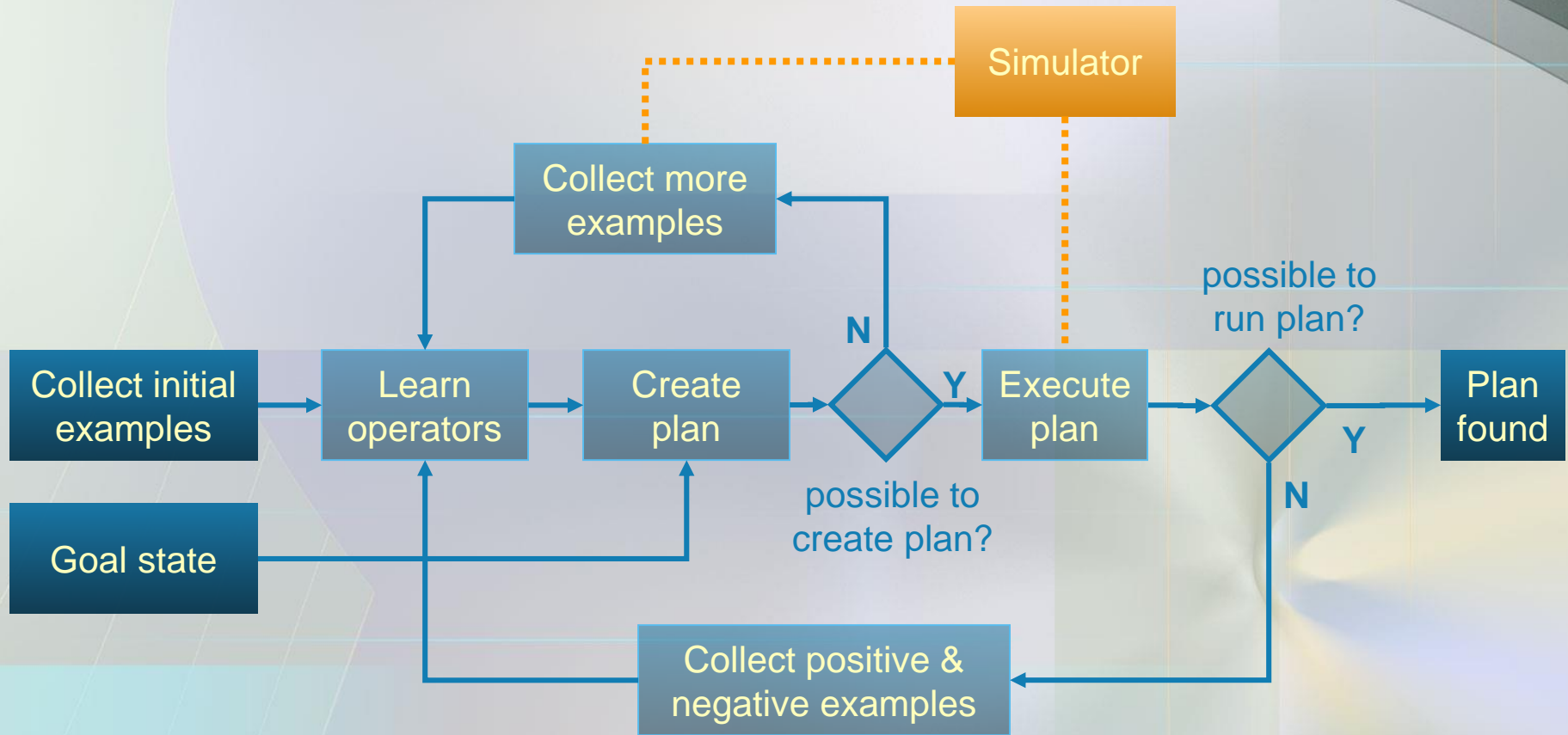
- use ILP to:
 - learn preconditions by specialization
 - positive and negative examples available
 - start with most general rule (top-down search)
 - learn effects by generalization
 - only positive examples available
 - start with most specific rule (bottom-up search)

Planning workflow processes

- Partial-Order Planning (POP)
 - activities as planning operators
 - expert modeler defines the predicates
 - users provide initial examples
 - learn preconditions and effects via ILP
 - generate process model via planning

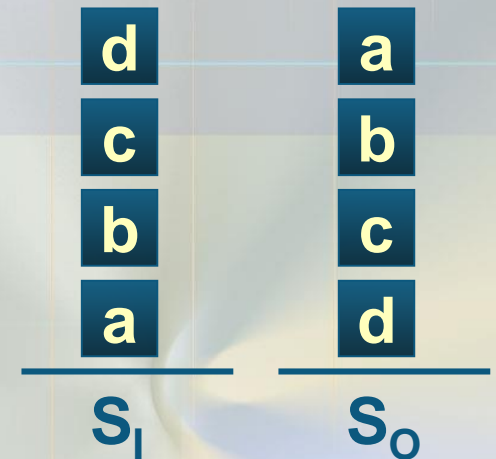


Learning and planning algorithm



Example 1: blocks world scenario

- how to go from S_I to S_O knowing that there are 3 kinds of moves:
 - `movebb(X,Y,Z)`
 - `movebt(X,Y)`
 - `movetb(X,Y)`
- not knowing what they do or how to apply them!
- predicates: *on(X,Y)* , *clear(X)*
- simulator knows true operators



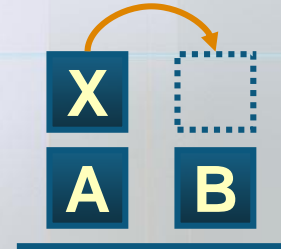
Example 1: blocks world scenario

`movebb (X, A, B)`

P: `clear (X)` , `clear (B)` , `on (X, A)`

-: `on (X, A)` , `clear (B)`

+: `clear (A)` , `on (X, B)`

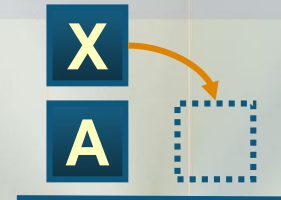


`movebt (X, A)`

P: `clear (X)` , `on (X, A)`

-: `on (X, A)`

+: `clear (A)`

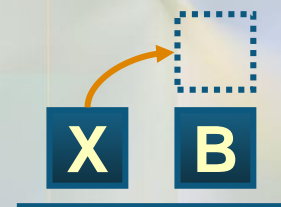


`movetb (X, B)`

P: `clear (X)` , `clear (B)`

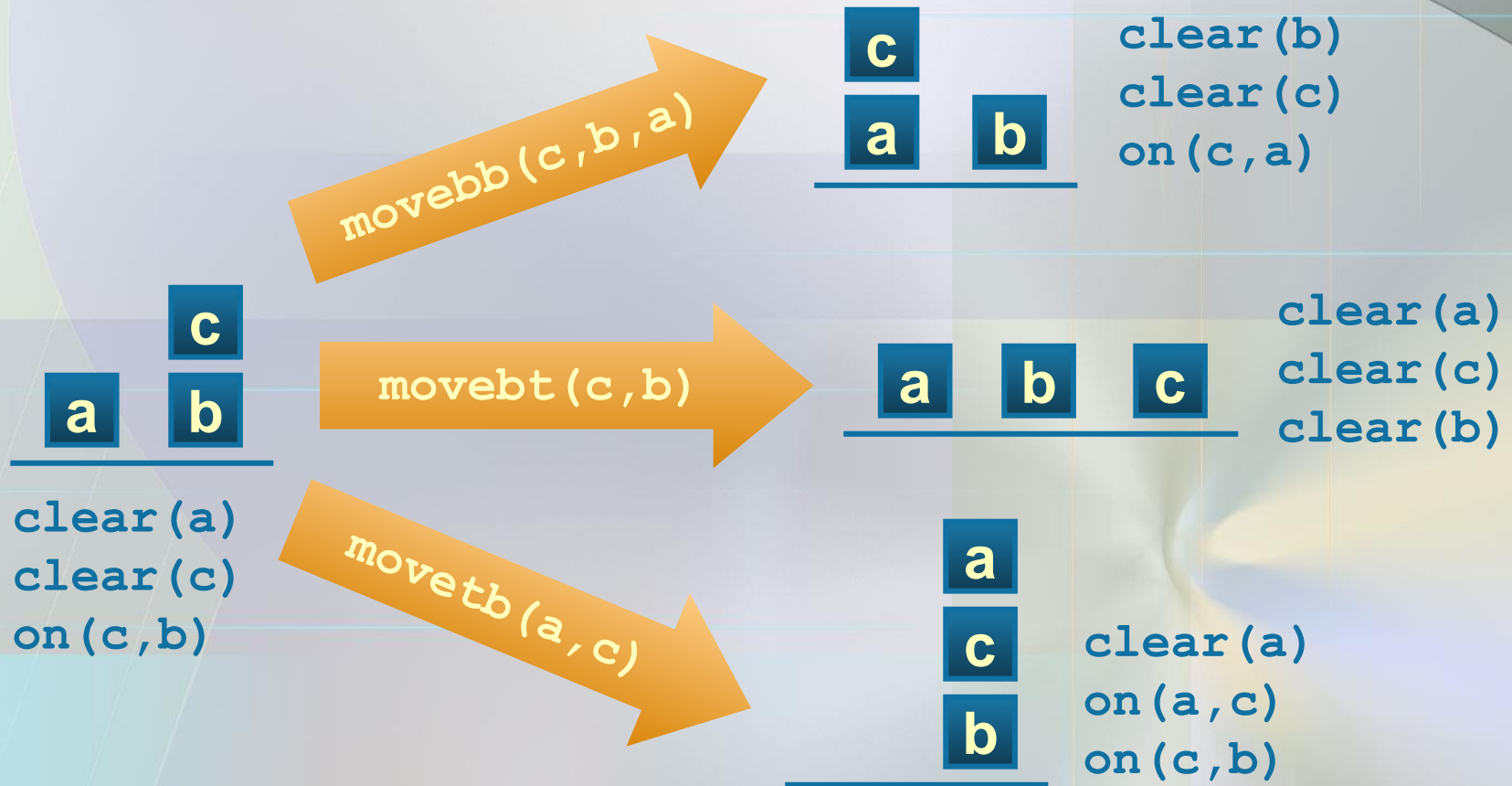
-: `clear (B)`

+: `on (X, B)`



Example 1: blocks world scenario

- Single positive example for each operator

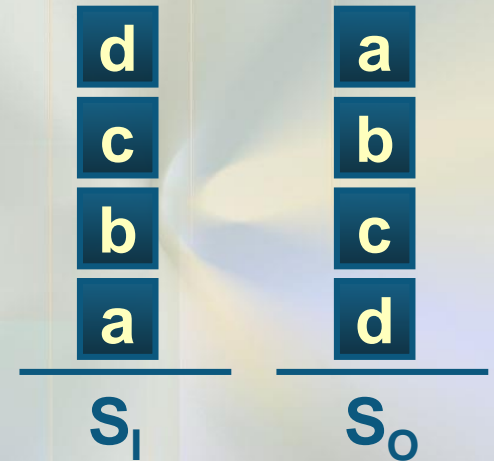


Example 1: blocks world scenario

- Results:

- solved in 5 iterations
- `movebb(X,Y,Z)` correctly learned
- `movetb(X,Y)` missing precondition `clear(Y)`
- `movebt(X,Y)` missing all preconditions
- solution:

1. `movebt(?,c)`
2. `movebb(c,b,d)`
3. `movebb(b,a,c)`
4. `movetb(a,b)`



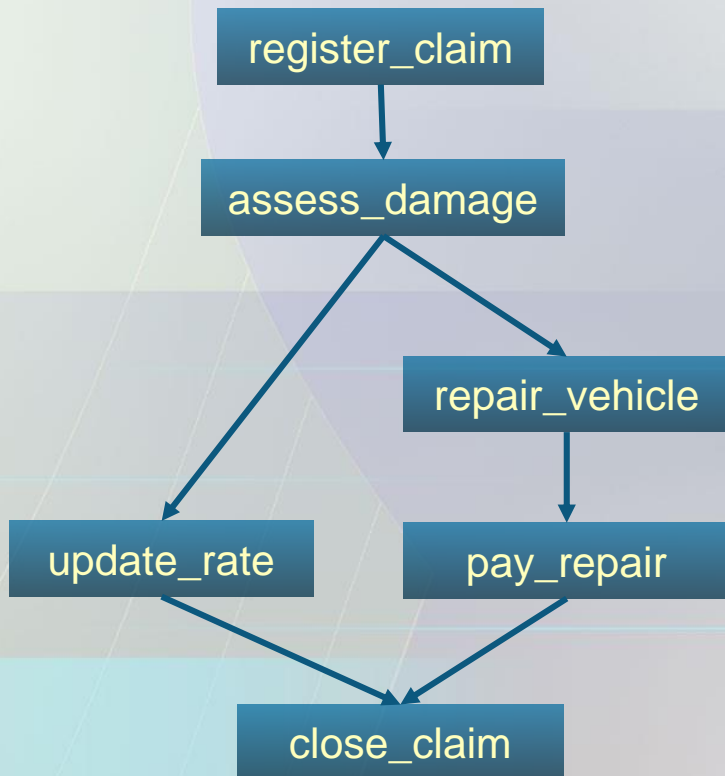
Example 2: insurance claim processing

Operators:

- register_claim(C,V)
- assess_damage(C,V)
- repair_vehicle(C,V)
- pay_repair(C,V)
- update_rate(C,V)
- close_claim(C,V)

Thesaurus:

- claim(C)
- vehicle(V)
- policy(P,V)
- damaged(V)
- assessed(C)
- repaired(V)
- payed(C)
- raised(P)
- bonus(P)
- open(C)
- closed(C)

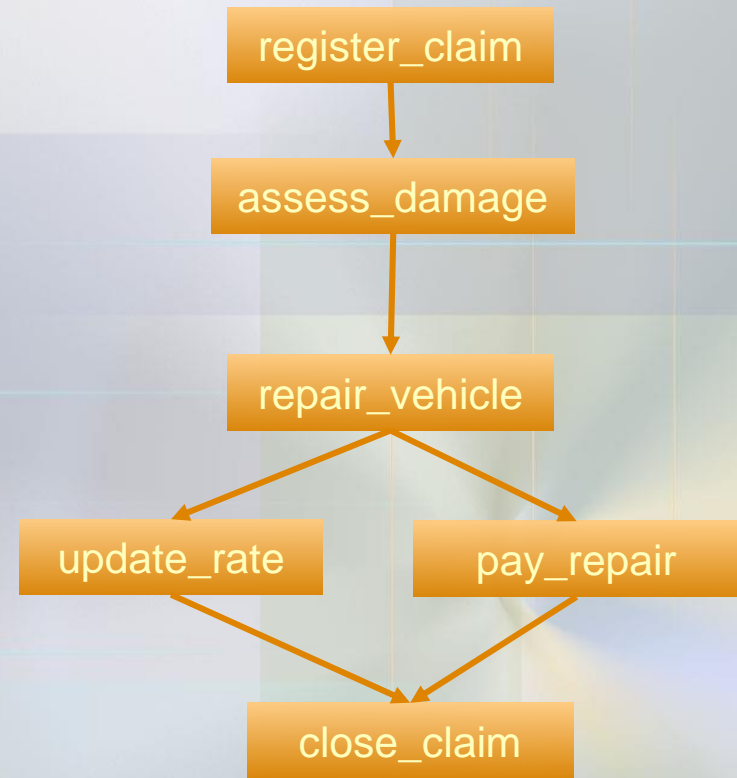
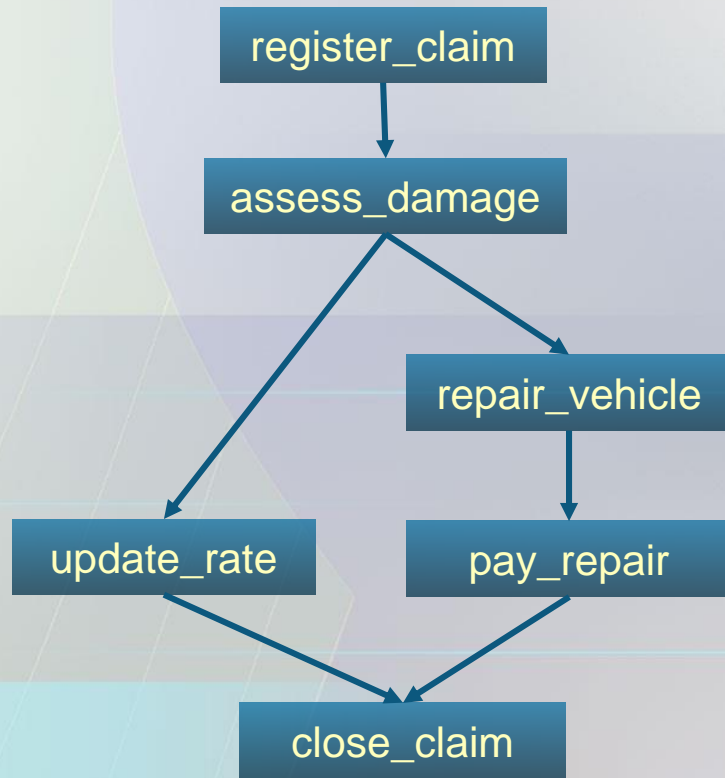


Example 2: insurance claim processing

- inputs:
 - initial state: $vehicle(v)$, $policy(p,v)$, $claim(c)$
 - goal state: $closed(c)$
 - single positive example for each operator

Example 2: insurance claim processing

- Results: solved in 10 iterations, but...



Example 2: insurance claim processing

register_claim(C,V)

P: **claim(C)** , **vehicle(V)** , **policy(P,V)**

R:

A: **damaged(V)** , **open(C)**

assess_damage(C,V)

P: **damaged(V)**

R:

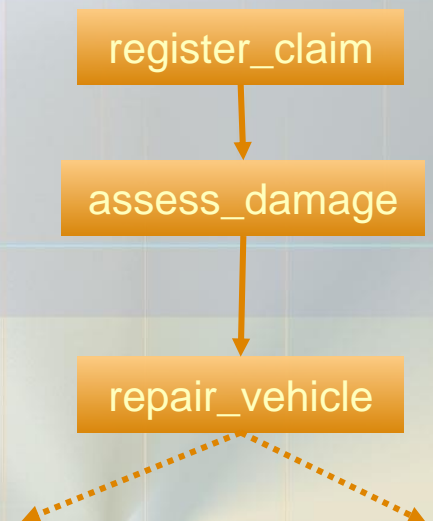
A: **assessed(C)**

repair_vehicle(C,V)

P: **damaged(V)** , **assessed(C)**

R: **damaged(V)**

A: **repaired(V)**



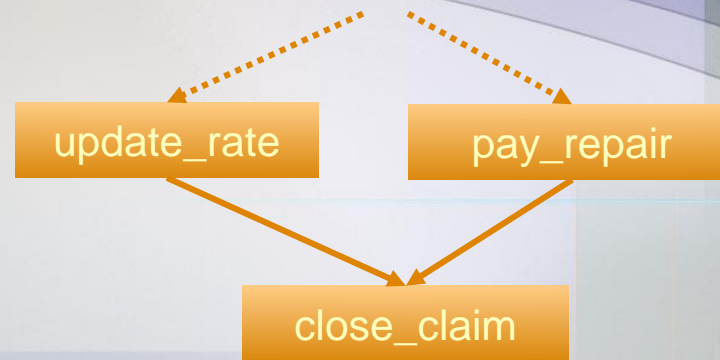
Example 2: insurance claim processing

`pay_repair(C,V)`

P: `repaired(V)`

R:

A: `payed(C)`



`update_rate(C,V)`

P: `assessed(C)` , `policy(P,V)` , `repaired(V)`

R: `bonus(P)`

A: `raised(P)`

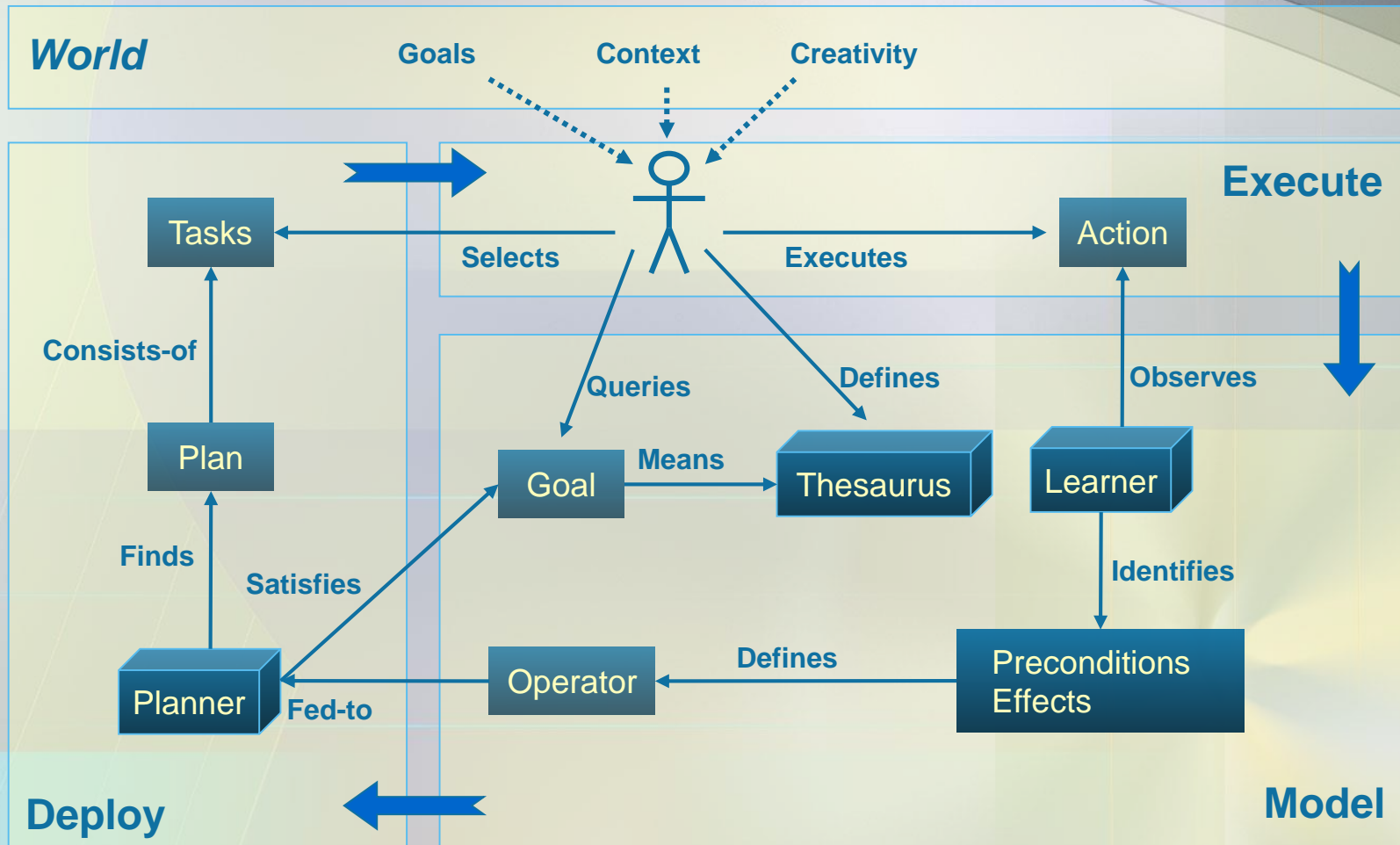
`close_claim(C,V)`

P: `open(C)` , `payed(C)` , `policy(P,V)` , `raised(P)`

A: `closed(C)`

R: `open(C)`

A new workflow life cycle



Remaining challenges

- learning and planning issues
 - efficiency of the planning algorithm
 - planning with sensing actions
 - learning with inconsistent examples
 - multiple agents acting concurrently
 - mixed-initiative planning

Remaining challenges

- other fundamental issues
 - capturing world states (observation)
 - making use of common semantics (ontologies)

Part II: Enter CEO

A Engenharia Organizacional

- objecto de estudo:
 - a organização e os seus sistemas de suporte
- âmbito:
 - todos os elementos que compõe a arq. empresarial
 - pessoas, funções, actividades, entidades informacionais, ...
- propósito:
 - desenvolvimento metodológico e tecnológico
 - modelar, analisar, conceber, desenvolver, monitorizar, avaliar, (...) *sistemas organizacionais*

A Engenharia Organizacional

- **objectivo último:**
 - **criação do “autocad empresarial”**
 - ferramenta que permite monitorizar e agir, em tempo real, em todos os elementos da arq. empresarial
 - **pressupõe observabilidade e controlabilidade**
 - cf. teoria de controlo
 - cf. aprendizagem & planeamento

O Framework CEO

- formalismo de modelação empresarial
 - incide em objectivos estratégicos, processos e sistemas
 - mostra a interligação entre estes
 - mantém um mapa do alinhamento dos sistemas com a estratégia de negócio
 - base do FCEO
 - UML
 - extensões Eriksson-Penker
 - MIT Process Handbook

Extensões ao FCEO

- extensões em desenvolvimento
 - separation of concerns (Caetano et al, 2004)
 - actors, roles
 - auditoria (Santos, 2004)
 - objectivos de controlo, processos de controlo, recursos de controlo
 - arquitectura de SI (Sérgio, 2002)
 - sistema, serviço, operação
 - observação (Zacarias, 2004)
 - contextos de acção/interacção

Artur Caetano, António Rito da Silva, José Tribolet, "Using Roles to Specify Business Objects Collaborations", IADIS 2004

Carlos Santos, André Vasconcelos, Carla Ferreira, José Tribolet, "Da Framework CEO à Auditoria de Sistemas de Informação", CAPSI 2004

Gonçalo Sérgio, "Sistema de Informação como objecto de Arquitectura de Empresa", TFC, IST, 2002

Marielba Zacarias, Artur Caetano, Sofia Pinto, José Tribolet, "Modelação de Contextos em Engenharia Organizacional", CAPSI 2004

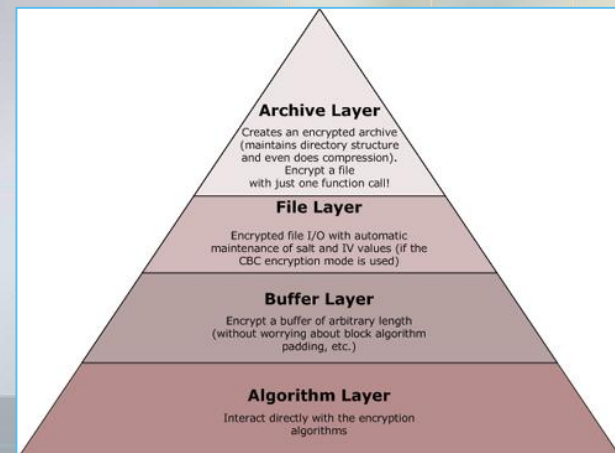
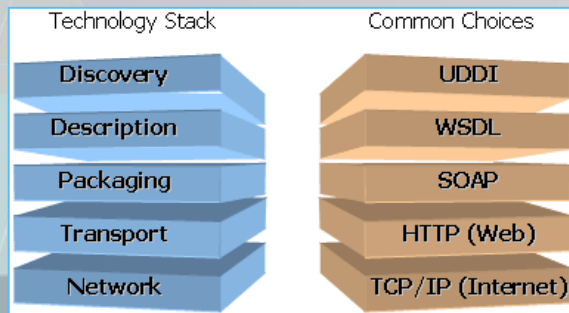
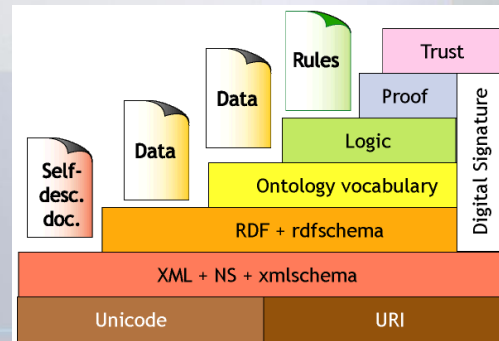
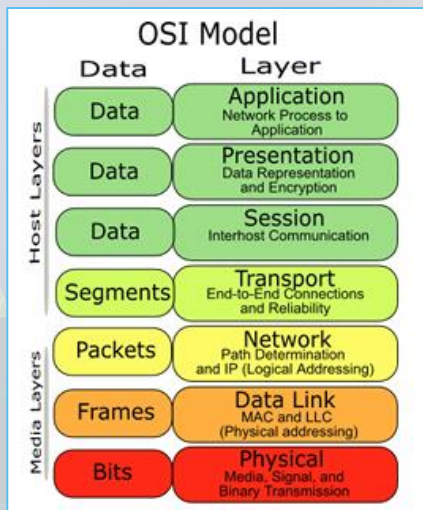
Modelação empresarial

- as primitivas organizacionais
 - activity, role, entity (Sousa et al, 2005)
 - entities, processes, events, rules (Caetano et al, 2000)
 - processes, goals, resources (Sinogas et al, 2001)
 - processes, events, resources, goals, rules (Castela et al, 2001)
 - resources, goals, rules (Mendes et al, 2001)
 - goals, processes, systems (Vasconcelos et al, 2001)
 - actors, action/interaction contexts (Zacarias et al, 2005)
 - business objects, activities, entities, resources, actors, goals (Caetano et al, 2005)

Pedro Sousa, Artur Caetano, André Vasconcelos, Carla Pereira, José Tribolet, "Enterprise Architecture Modeling with the Unified Modeling Language", 2005
Artur Caetano, António Rito Silva, José Tribolet, João Neves, Pedro Sinogas, "The MODIFY Project: Combined Business and System Modeling for Adaptable Enterprise Computing System Design", 2000
Pedro Sinogas, André Vasconcelos, Artur Caetano, João Neves, Ricardo Mendes, José Tribolet, "Business Processes Extensions to UML Profile for Business Modeling", 2001
Nuno Castela, José Tribolet, Alberto Silva, Arminda Guerra, "Business Process Modeling with UML", 2001
Ricardo Mendes, André Vasconcelos, Artur Caetano, João Neves, Pedro Sinogas, José Tribolet, "Representing Business Strategy Through Goal Modeling", 2001
André Vasconcelos, Artur Caetano, João Neves, Pedro Sinogas, Ricardo Mendes, José Tribolet, "A Framework for Modeling Strategy, Business Processes and Information Systems", 2001
Marielba Zacarias, Artur Caetano, Sofia Pinto, José Tribolet, "Modeling Organizations: The Operating Systems metaphor", 2005
Artur Caetano, António Rito da Silva, José Tribolet, "Using Roles and Business Objects to Model and Understand Business Processes", 2005

Modelação empresarial

- necessidade de um modelo em camadas

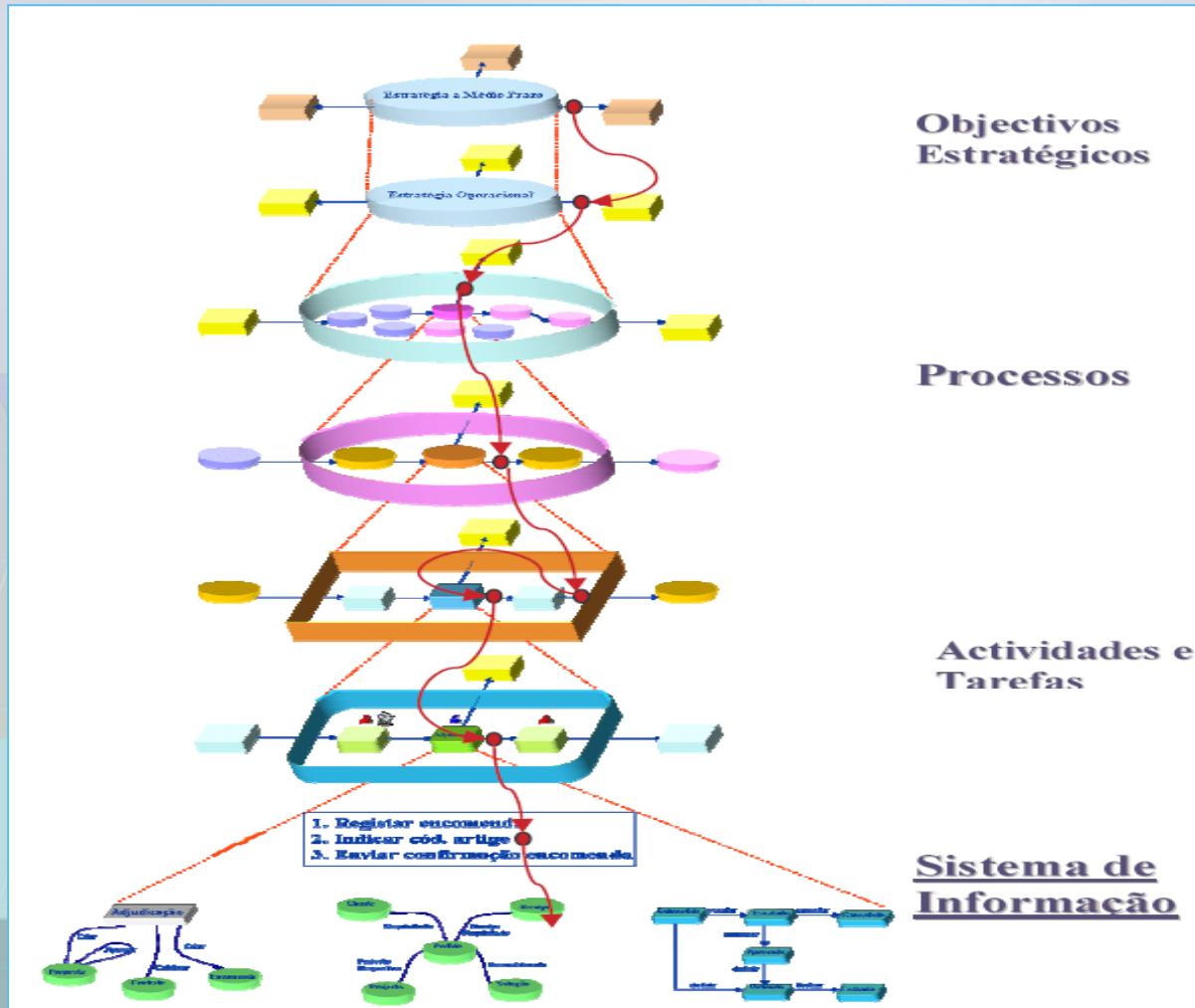


Modelação empresarial

- modelo em camadas – exemplo 1 (*):
 - camada de processos de negócio
 - primitivas: «Actor», «Activity», «Resource»
 - camada de Web Services genéricos
 - primitivas: «ServiceBlock», «ServiceRole», «Service», «Operation», «Document»
 - camada tecnológica
 - primitivas: «Operation», «portType», «Binding», «Port», «Message», «Parameter»
 - mapeamento entre camadas

Modelação empresarial

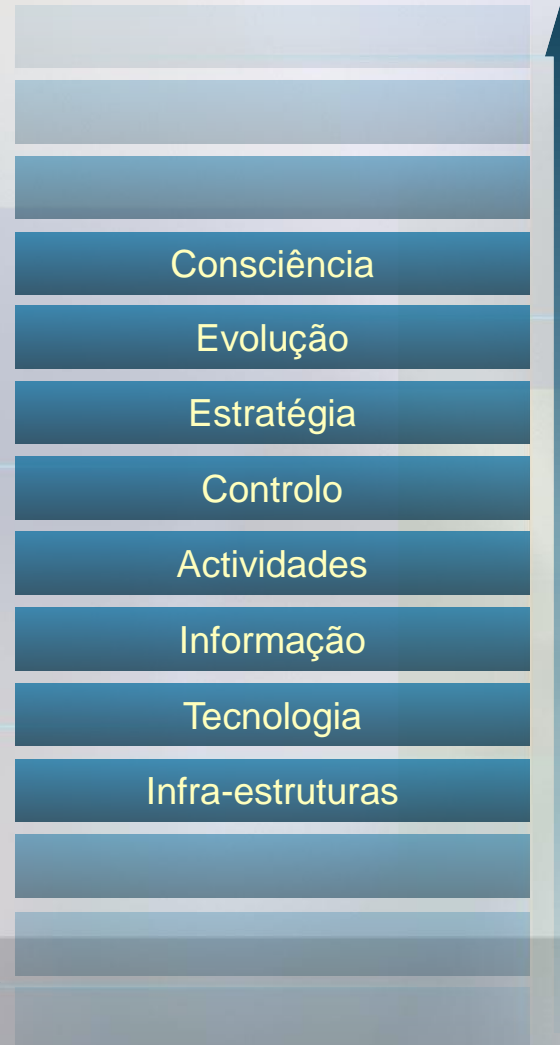
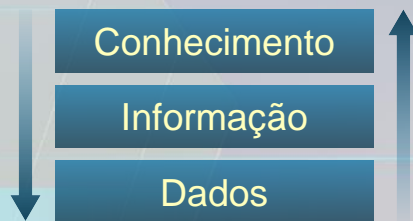
- modelo em camadas – exemplo 2 (*):



(*) Jorge Coelho
"Método LEARN"
SisConsult

Modelação empresarial

- mas como modelar a organização inteira?...
 - que camadas?
 - que primitivas?
 - top-down?
 - bottom-up?
 - middle-out?

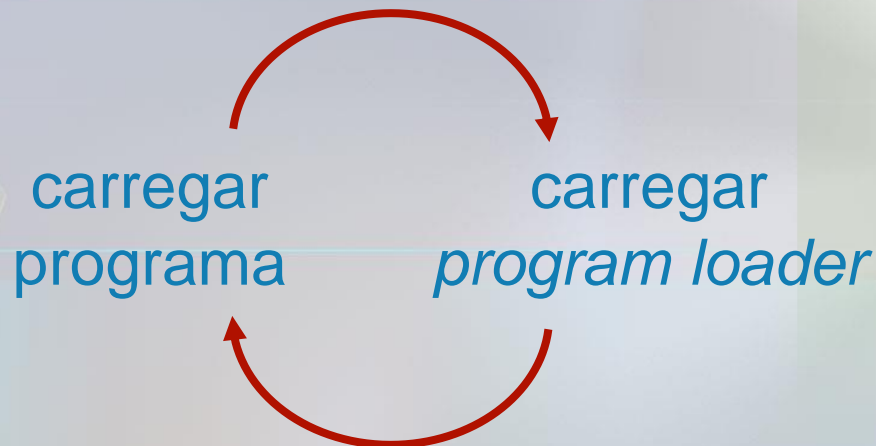




Workflow upside down

The bootstrap paradox

- como é que uma máquina arranca?
 - necessário carregar um programa (OS)
 - necessário ter um *program loader*
 - mas o *program loader* também é um programa!
 - quem carrega o *program loader*?...

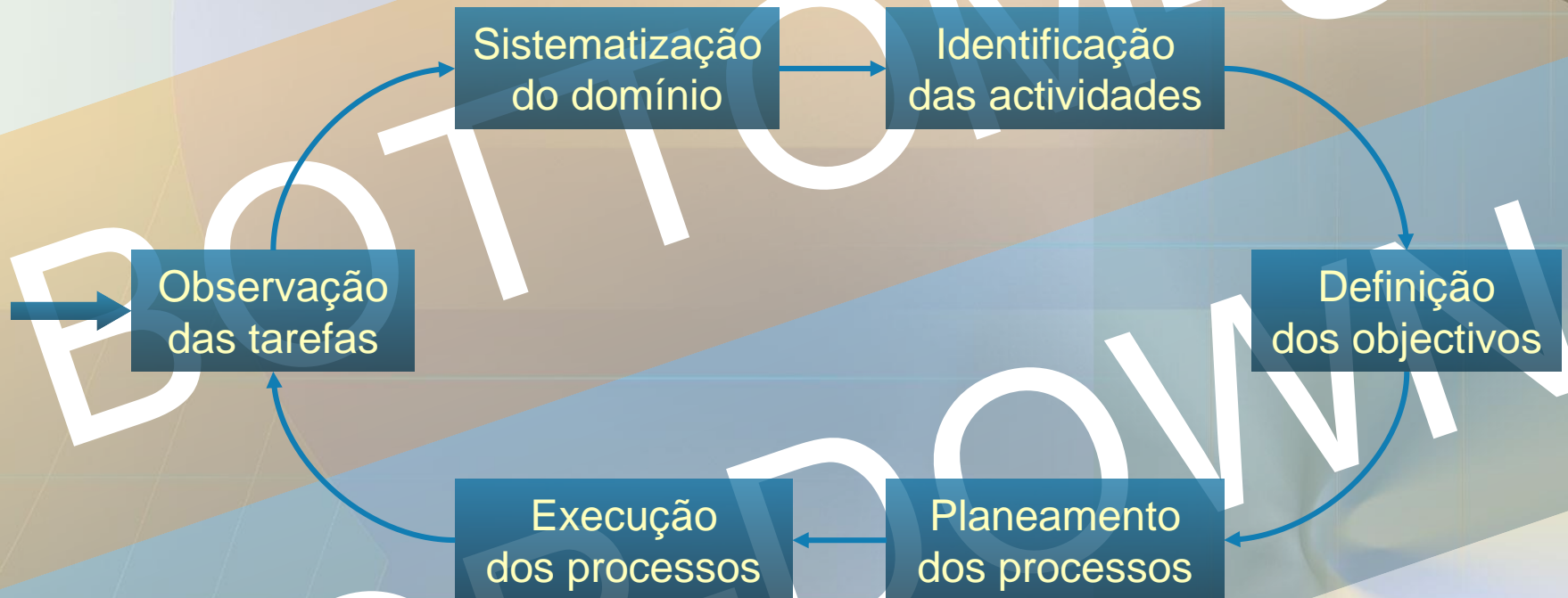


The process paradox

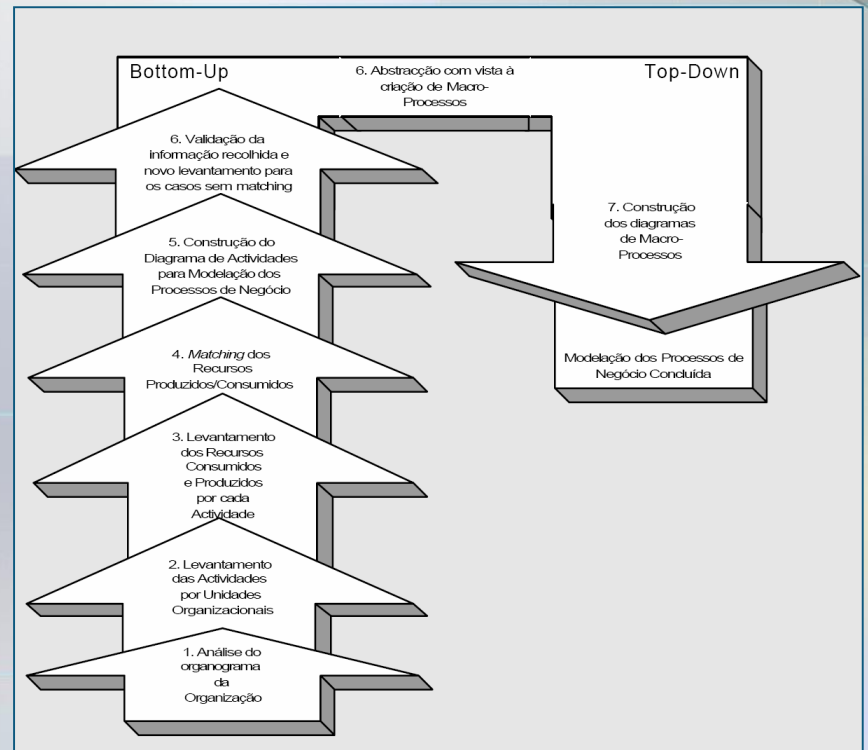
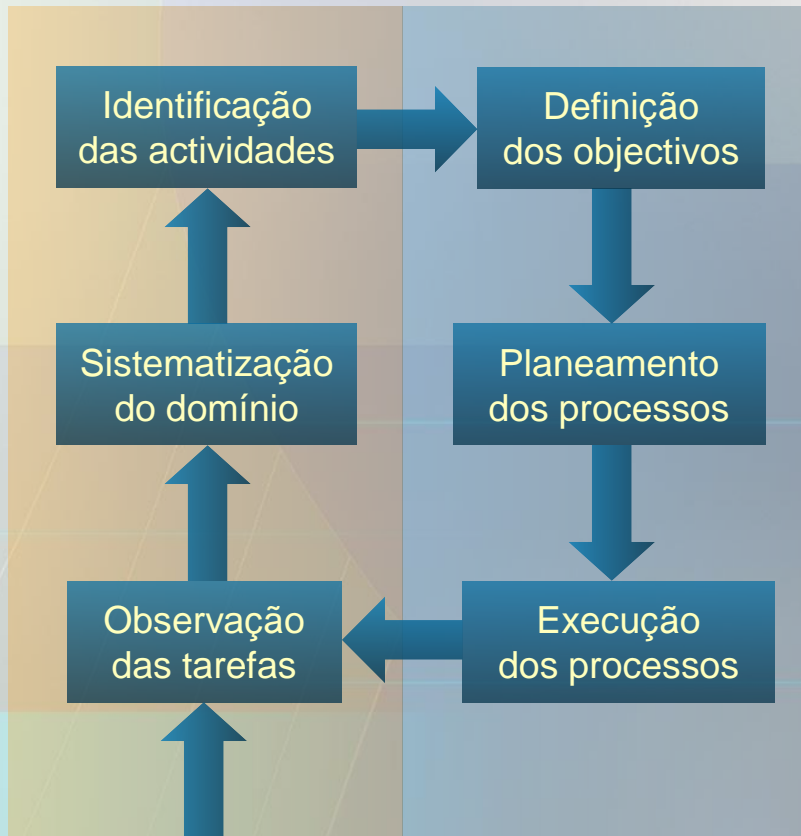
- o que é um processo de negócio?
 - é o que as pessoas fazem
 - actividades estruturadas de algum modo
 - esse modo é um processo!
 - quem define o processo?...



Where to begin



Where to begin



Nuno Castela, José Tribolet, Arminda Guerra, Eurico Lopes, "Survey, Analysis and Validation of Information for Business Process Modeling", ICEIS 2002

Plano de acção

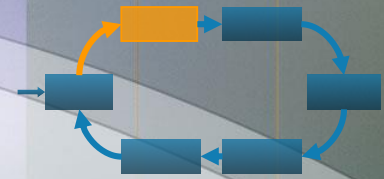
Observação das tarefas

- questões a tratar:
 - detecção automática de contextos de acção
 - infra-estrutura de observação
 - integração com ferramentas existentes
 - resolução das questões éticas associadas
 - repositório de documentos
 - informação processada durante a acção
 - documentos só disponíveis em papel (digitalização+OCR)



Sistematização do domínio

- questões a tratar:
 - extracção dos termos relevantes
 - técnicas de IR
 - resultados contextualizados
 - criação da ontologia do domínio
 - criação manual vs. assistida
 - possibilidade de actualizações
 - formalismo de modelação associado



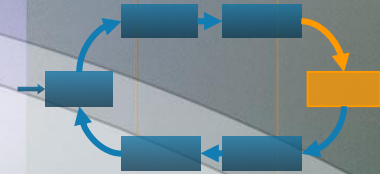
Identificação das actividades

- questões a tratar:
 - aprendizagem de regras de negócio
 - determinar pré-condições e efeitos
 - algoritmos de indução (ILP, Defeasible Logic, etc.)
 - formalismo para expressão de regras (FOL, OCL, StAC,...)
 - representação da incerteza
 - repositório de regras de negócio
 - evolução das práticas de negócio
 - remanescente de regras ECA (Event-Condition-Action)



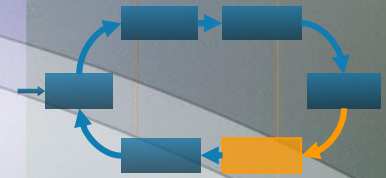
Definição dos objectivos

- questões a tratar:
 - identificação dos objectivos
 - apenas objectivos operacionais
 - identificar o estado final pretendido
 - especificação dos objectivos
 - em termos da ontologia do domínio
 - formalismo coerente com o anterior
 - manter repositório de objectivos possíveis/frequentes



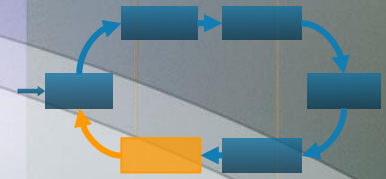
Planeamento dos processos

- questões a tratar:
 - algoritmo de planeamento
 - linguagens de planeamento (formalismo)
 - necessidade de algoritmo próprio? (formalismo e performance)
 - planeamento assistido (*orderings* conhecidos à partida)
 - *sensing actions*
 - re-planeamento contínuo



Execução dos processos

- questões a tratar:
 - mecanismos de interacção
 - interacção através de *tasklist* forneceria contexto
 - detecção da mudança de estado (*framing problem*)
 - erros de execução
 - são exemplos negativos úteis
 - mecanismos de compensação (reposição do estado inicial)
 - acções reversíveis vs. irreversíveis



Conclusão

Trends

- workflow is effectively turning upside down
 - in the research community
 - e.g. process mining
 - in the commercial arena
 - e.g. Ultimus Adaptive Discovery™

Desafios

- grandes desafios para a eng. informática
 - infra-estruturas omni-presentes
 - extracção de informação
 - representação de conhecimento
 - processamento semântico
 - complexidade temporal dos algoritmos
 - formalismos de modelação
 - ...

Implementação

- tecnologias necessárias
 - grande parte já disponíveis ou em desenvolvimento
- competências necessárias
 - todas as existentes na área de SI do DEI
- âmbito da implementação
 - tanto mais bem sucedida quanto mais restrito o âmbito
- futuro
 - alargamento aos restantes elementos da arq. empresarial
 - detecção automática/semi-automática de pessoas, funções, entidades informacionais, estrutura organizacional, etc.
 - reconfigurabilidade operacional/organizacional total

Oportunidade (1)

- exigências do mercado
 - tecnologia de workflow estagnada
 - âmbito de aplicabilidade
 - mecanismos de suporte aos processos
 - necessidade de ferramentas mais “inteligentes”
 - capazes de fornecer um *jump-start*
 - capazes de observar sem exigir conhecimento à partida

Oportunidade (2)

- **causa unificadora**
 - **abrangência tecnológica**
 - todas as competências de SI e mais algumas
 - intersecção entre SI e IA
 - **oportunidade de aplicação**
 - novo cenário para técnicas clássicas
 - pano de fundo para o desenvolvimento de novas técnicas