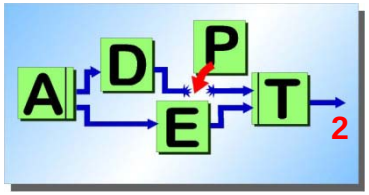
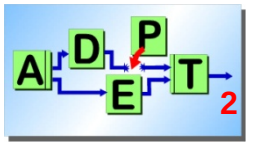


Flexible and Adaptive Process-oriented Information Systems – Challenges and Solutions –

Peter Dadam, Manfred Reichert
Institute of Databases and Information Systems
University of Ulm
D-89081 Ulm, Germany

www.informatik.uni-ulm.de/dbis
{peter.dadam, manfred.reichert}@uni-ulm.de



Current Hype in IT Magazines

EAI

BPEL

SOA

Web Services

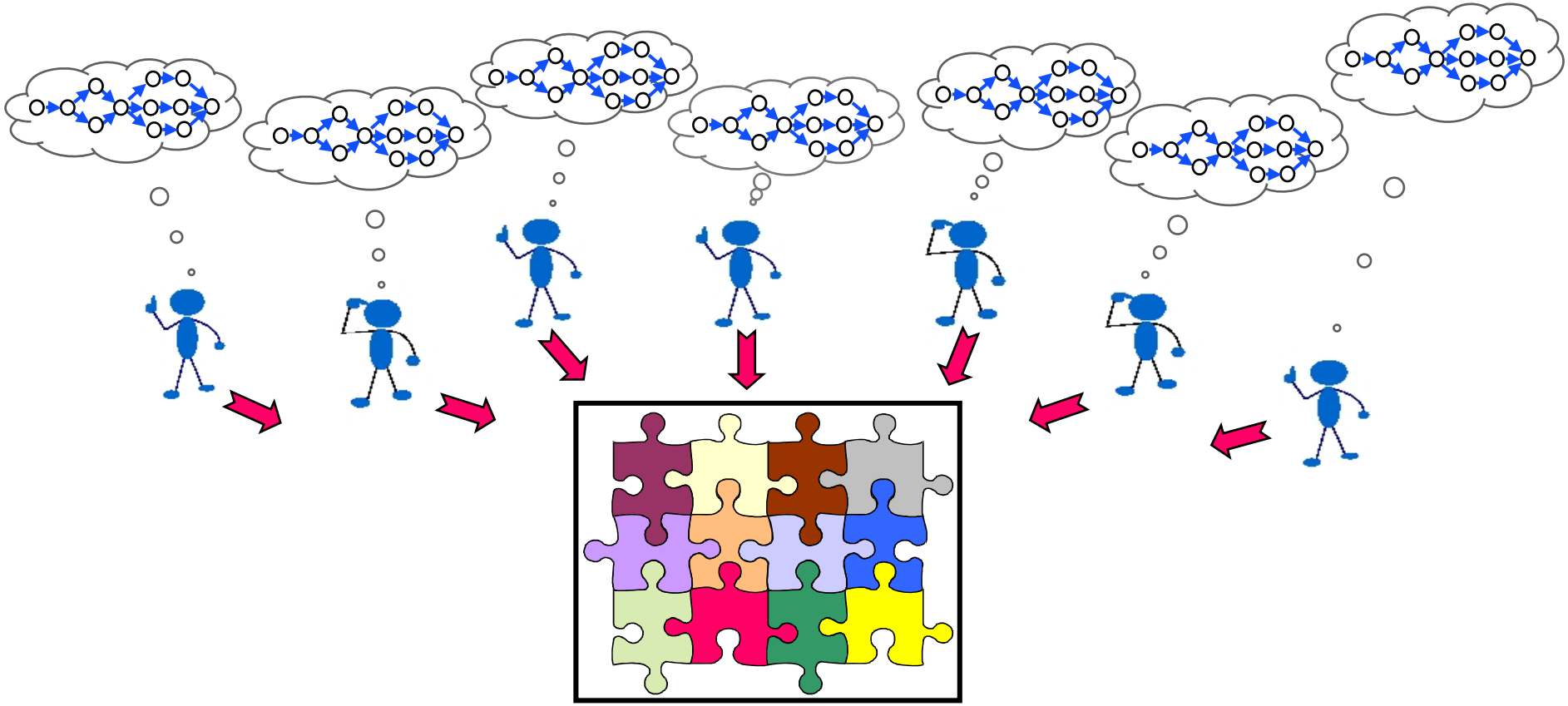
Process Orientation

Process Optimization

Workflow Management

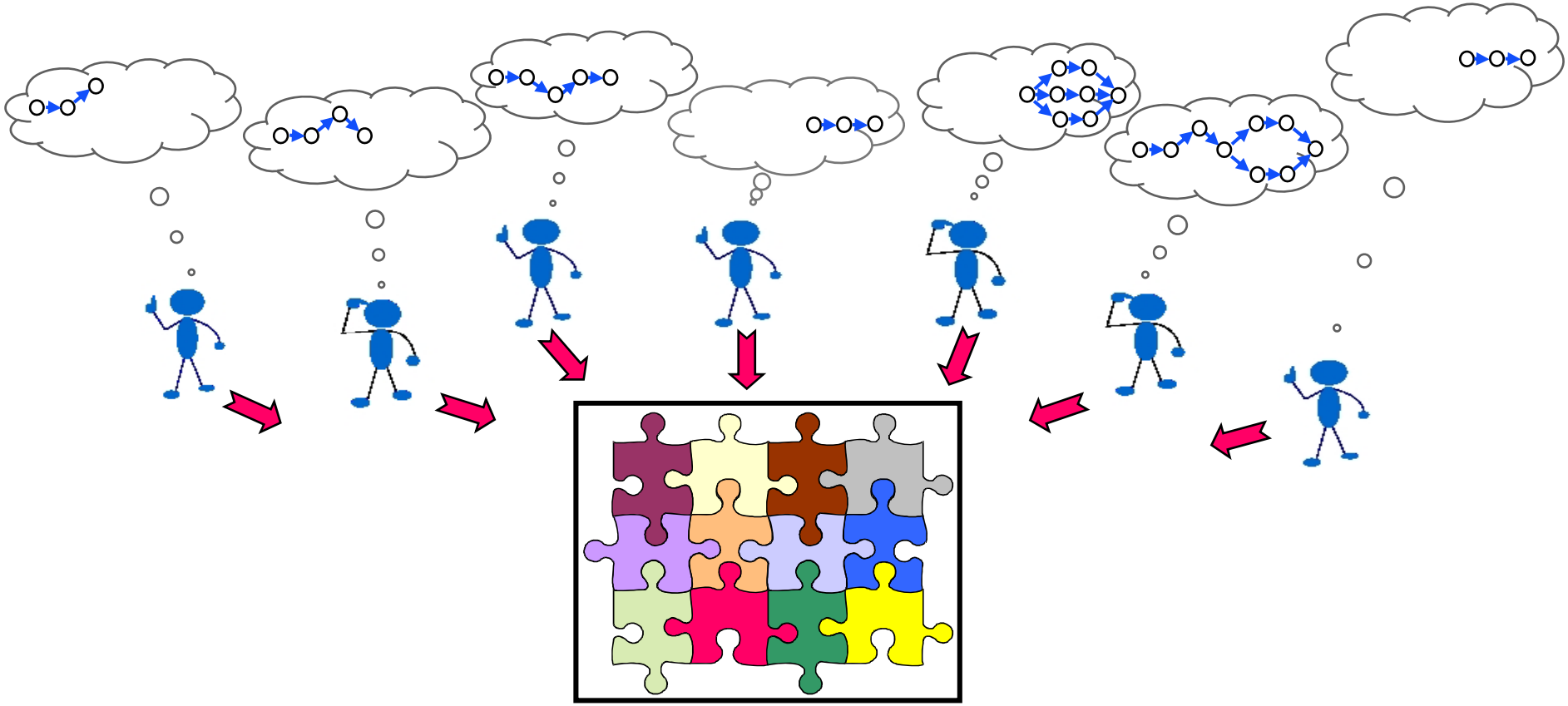
❒ Situation of today

- users interact with monolithic, function-oriented application systems
- processes only in the users' minds – with only partial knowledge of the process



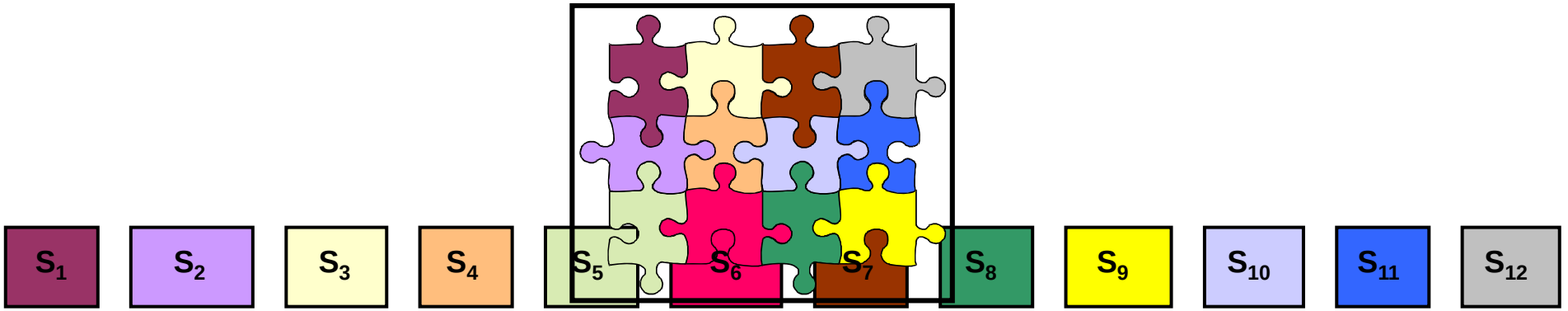
❒ Vision of SOA

- individually callable application functions (“services”)



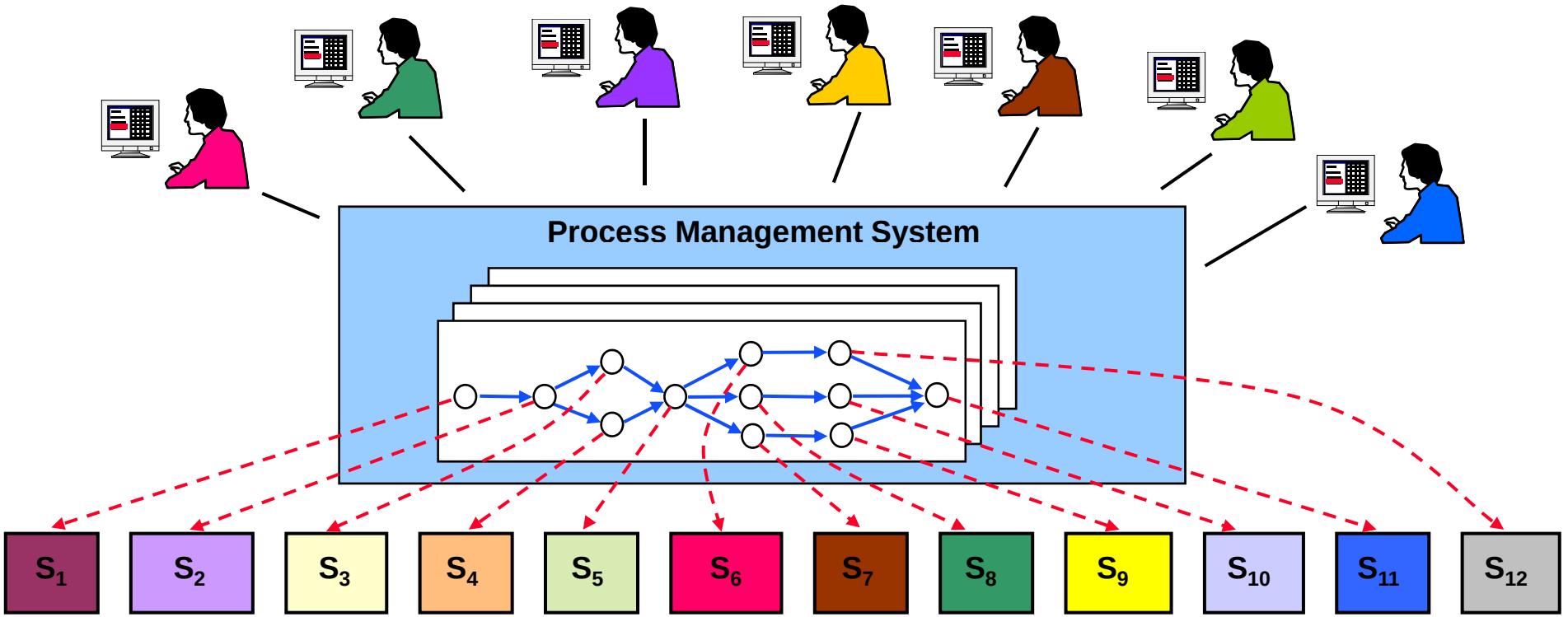
❒ Vision of SOA

- individually callable application functions (“services”)



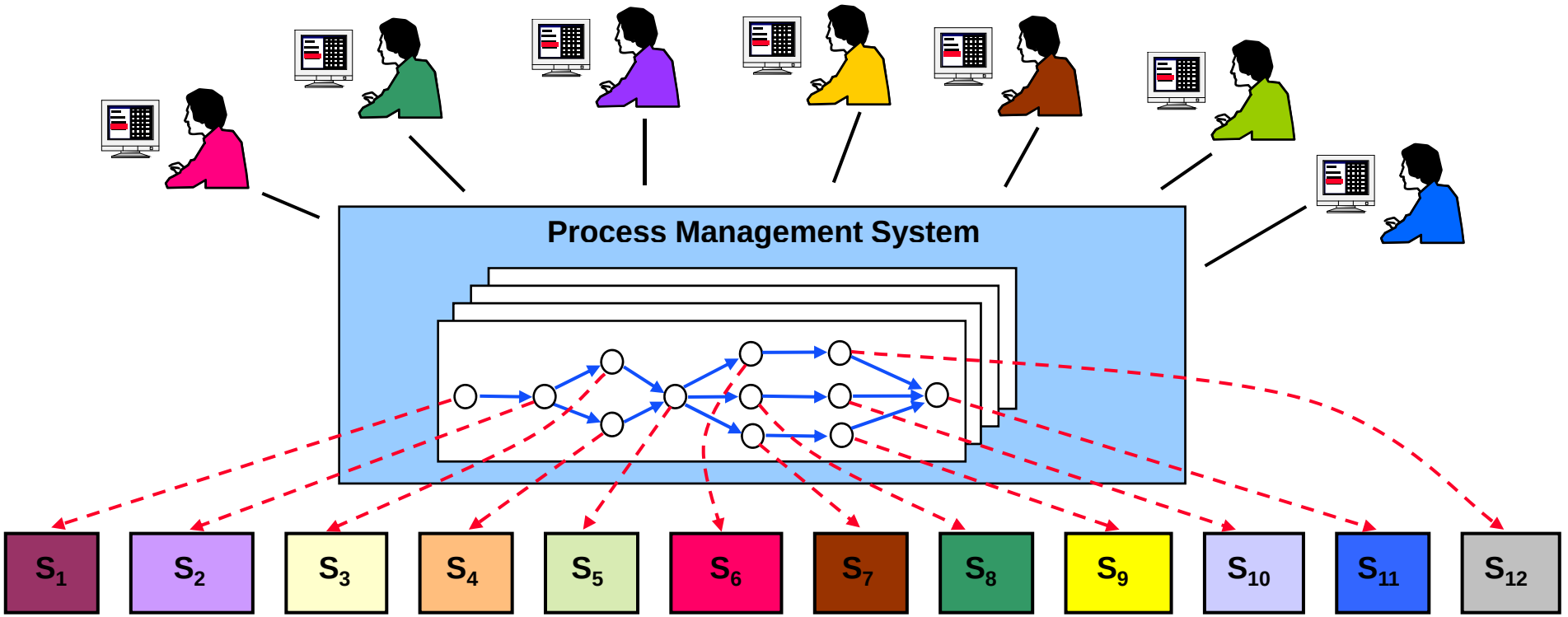
Vision of SOA

- individually callable application functions (“services”)
- combined by explicitly defined processes
- whose execution is supported by a process management system



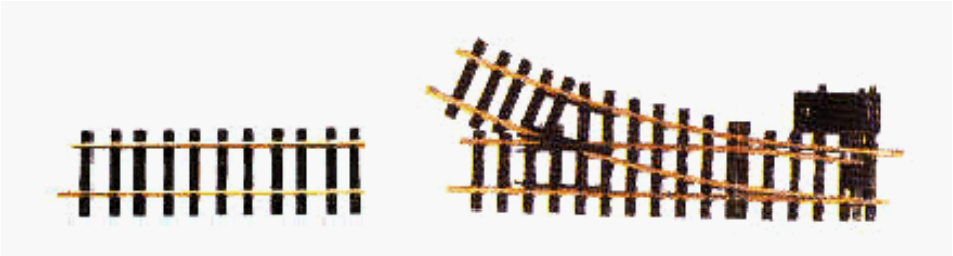
Goals of SOA

- improvement of process quality (reduction of errors, waste time, …)
- increased flexibility
- …



Different aspects of flexibility – very important to discriminate

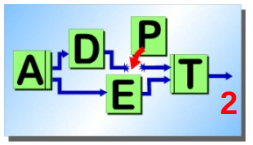
flexibility at build-time



How fast can new processes be implemented?

flexibility at run-time

How to handle exceptional cases during process execution?



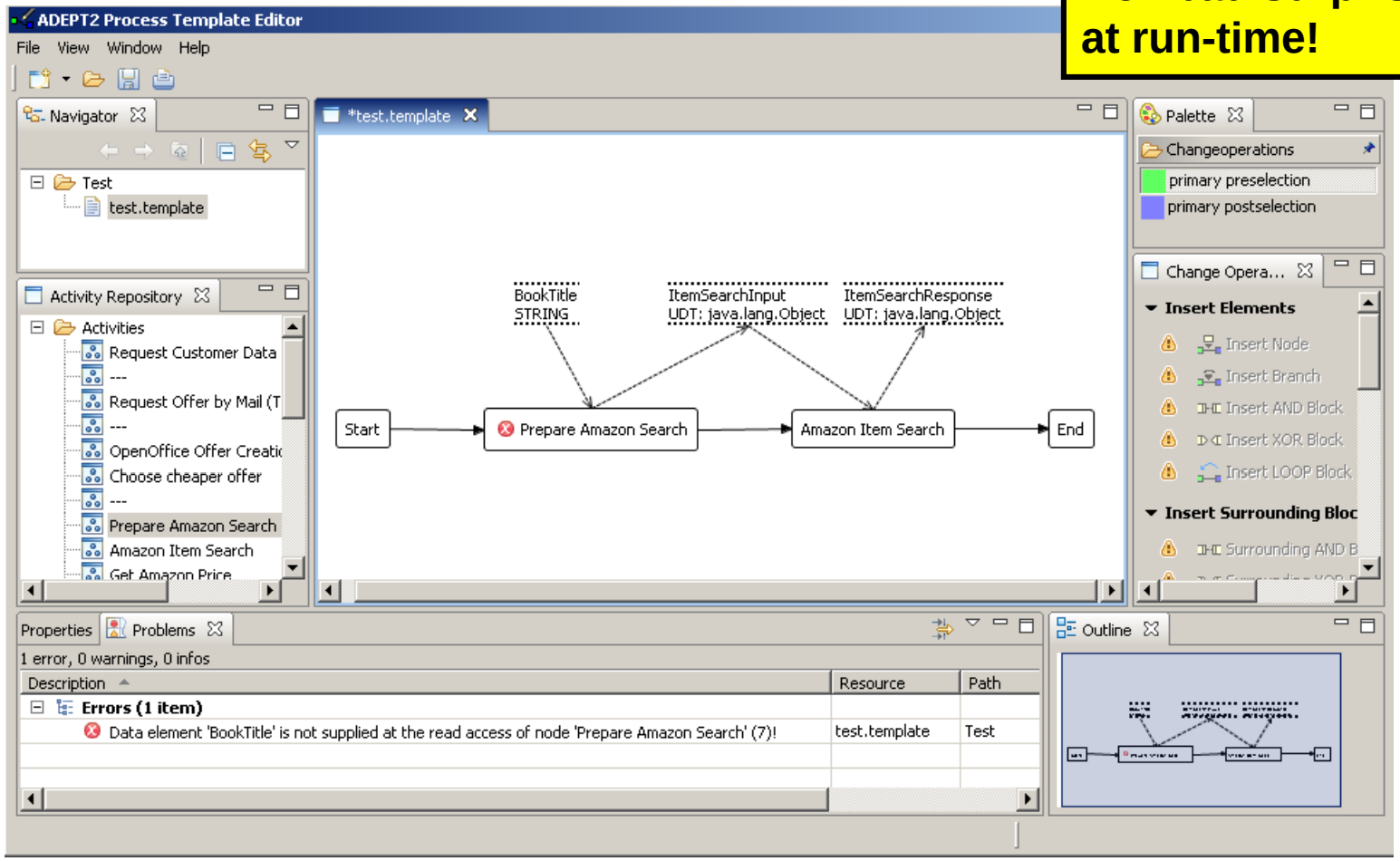
Essential questions when looking at the whole picture

- **How to implement process-aware IS quicker and cheaper?**
- **How to ensure robustness?**
- **How to handle unforeseen cases?**
  - How to support ad-hoc deviations from the pre-modeled execution order without "bad surprises" (= run-time errors)?
  - How to make this task so simple that it can be performed by the end-user?
- **How to support changes of the process template in the context of long-running processes?**

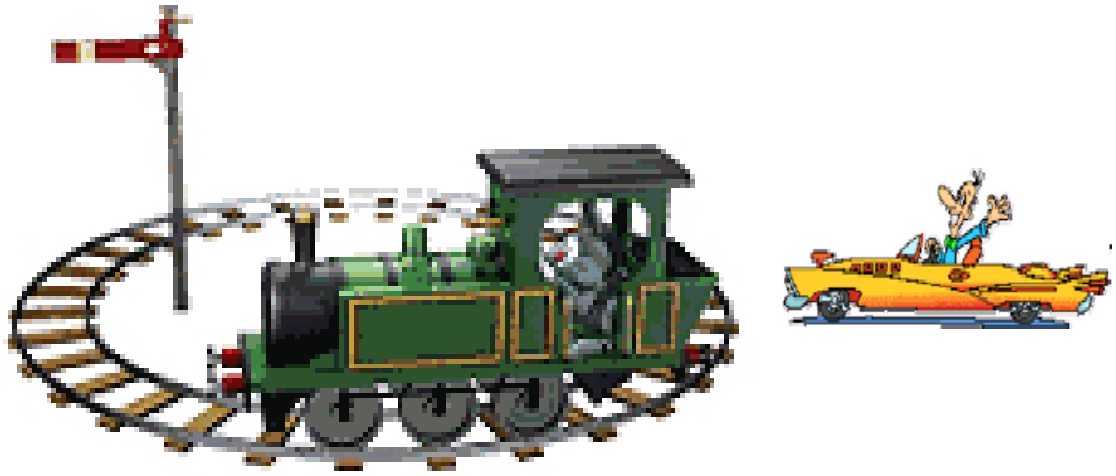
ADEPT$_2$ Editor: “Surprise-free” process models by construction

- **Construction of processes by plug & play fashion**

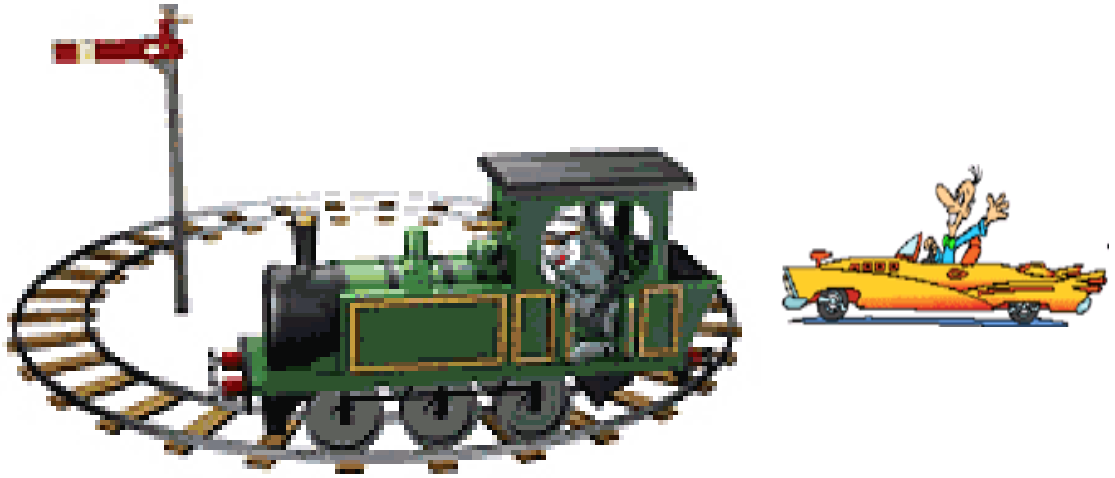
Goal:
No “bad surprises” at run-time!



Challenge: Ad-hoc deviations at process instance level



Challenge: Ad-hoc deviations at process instance level

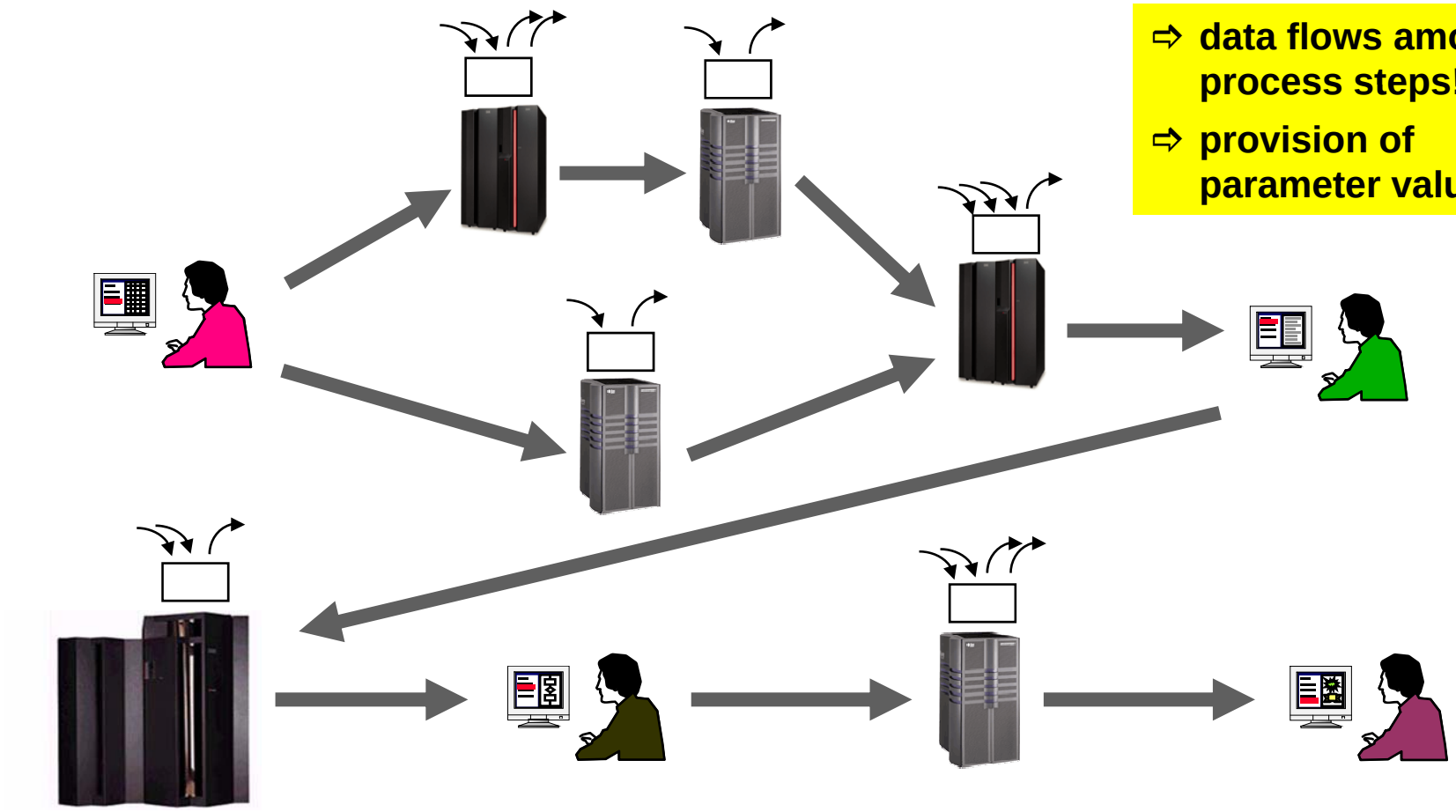


- **Insertion of new steps**
- **Deletion of steps**
- **Shifting of steps to another position**
- **….**

for process instances in execution!

Challenge: Ad-hoc deviations at process instance level

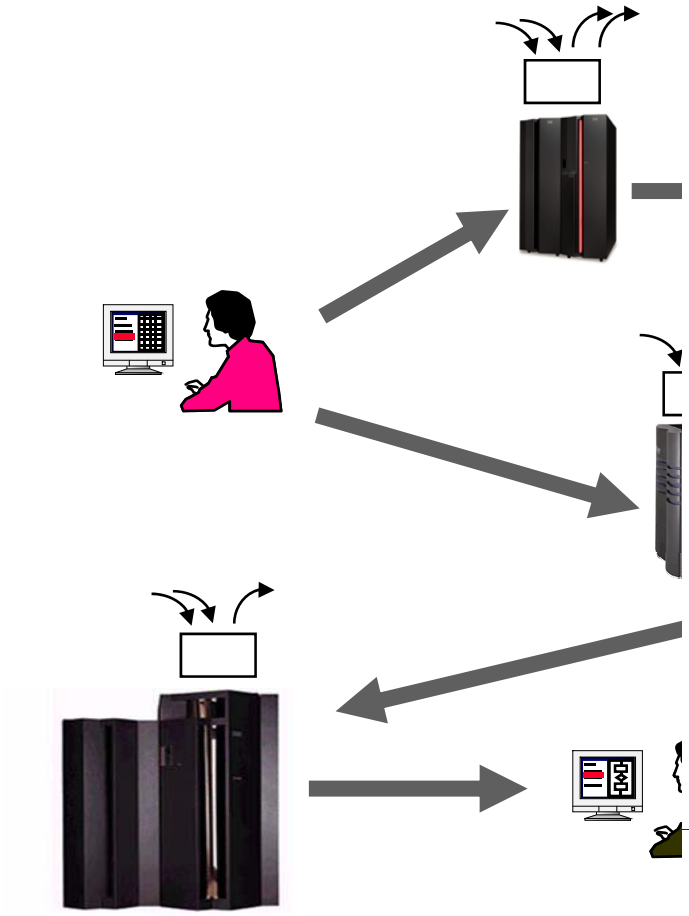
… becomes complicated in more general scenarios



⇨ **data flows among process steps!**

⇨ **provision of parameter values!**

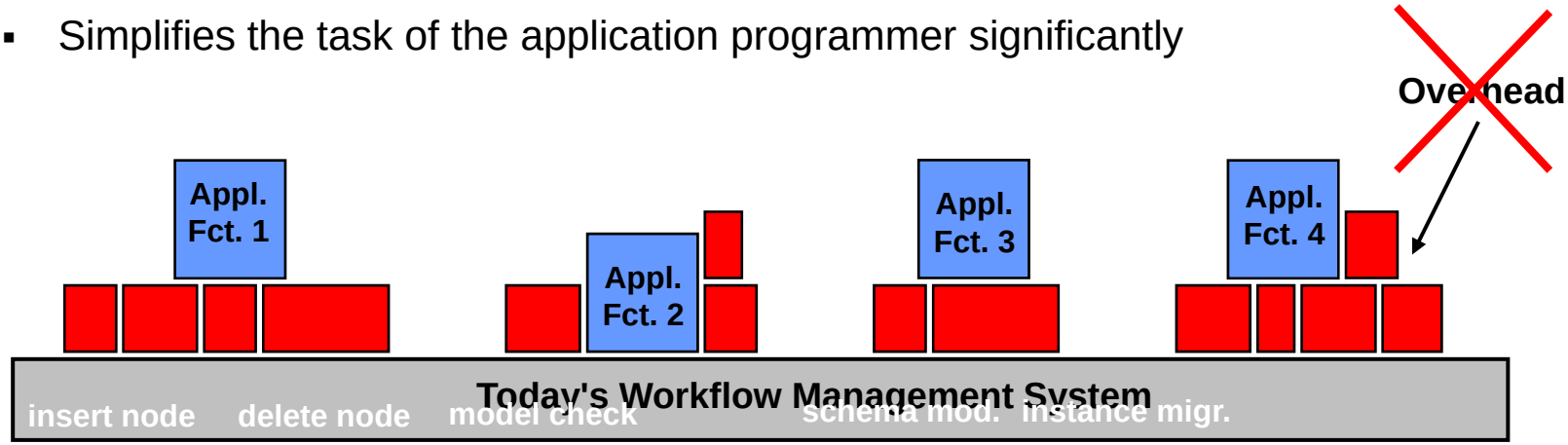
Challenge: Ad-hoc deviations at process instance level



- **Current Situation:**
  - Pre-modeling of all possible cases and exceptional cases required
  - and/or treatment of exceptional cases within the application functions
  - or immature solutions (“at the user’s risk”)
- **Consequences:**
  - High effort for realization
  - Limited applicability
  - Unexpected exceptions cannot be treated

Challenge: Ad-hoc deviations at process instance level

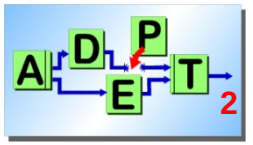
- **ADEPT**
  - Provision of powerful operations for ad-hoc deviations at the system's API
  - Immediate correctness checks – no "bad surprises" at run-time
  - Simplifies the task of the application programmer significantly



ADEPT Process Management System

model to follow: Relational Database Management Systems

Other Research Activities



- Visualization of Large Processes
- Process Configuration and Process Variants
- Data-driven Modeling, Coordination and Adaptation of Processes
- Process Constraint Management & Process Compliance
- Temporal Constraints (e.g., Scheduling, Periodicity)
- Process Variants Mining
- Security in Process-aware IS
- Composition of Process-aware IS (Plug & Play)
- Next Generation Service-oriented Computing
- Evolution of other Aspects (e.g., Org. Models, Access Rules)