Universitatea
din București

# Metode de dezvoltare software

## Arhitecturi software
## - câteva exemple -

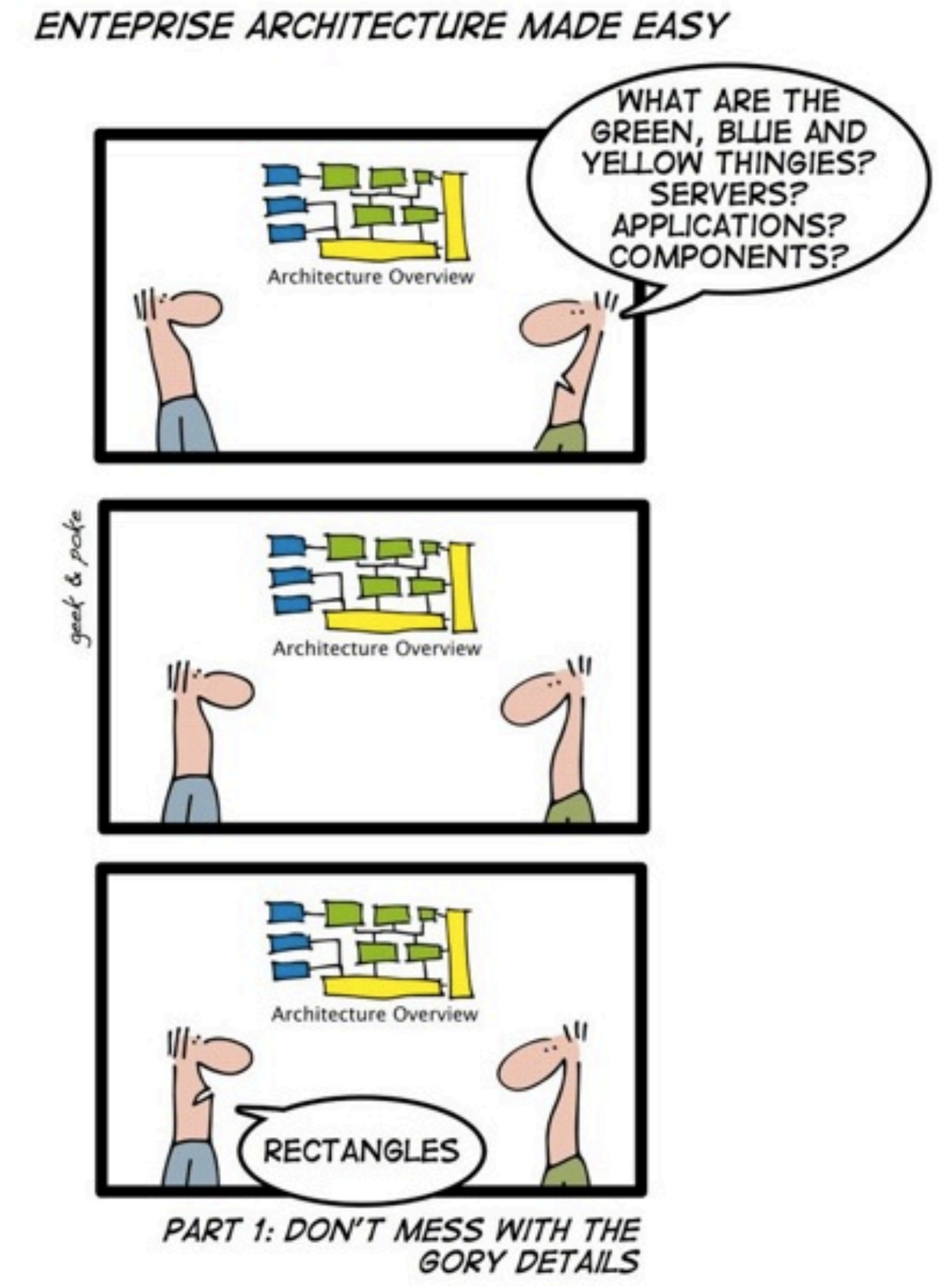
13.03.2017

Alin Ștefănescu

# Arhitecturi software

de la wikipedia.org

# Arhitecturi software… în practică



# Arhitectura unui sistem

- Arhitectura software este un subdomeniu important al ingineriei software
- Aceasta reprezintă **împărțirea optimă a unui sistem complex în diverse componente, evidențiind relațiile dintre acestea**.
- este esențială pentru a avea un sistem funcțional și scalabil
- de știut: jobul de ”arhitect software” este bine văzut și bine plătit: https://www.glassdoor.com/List/Highest-Paying-Jobs-LST_KQ0,19.htm

# Diverse atribute de calitate

Arhitectura unui sistem trebuie adaptată la atributele de calitate cerute: De exemplu:

- **performanță**: trebuie paralelizat cît mai mult, descompunând sistemul în procese cooperante; trebuie ținut sub control comunicare și accesul la date
- **precizie:** trebuie optimizată structura datelor și modul în care valorile sunt prelucrate
- **securitate**: trebuie gestionate bine restricțiile de comunicare și acces; trebuie analizate componentele cele mai vulnerabile
- **portabilitate și reutilizare**: trebuie minimizate depedențele puternice între componente.

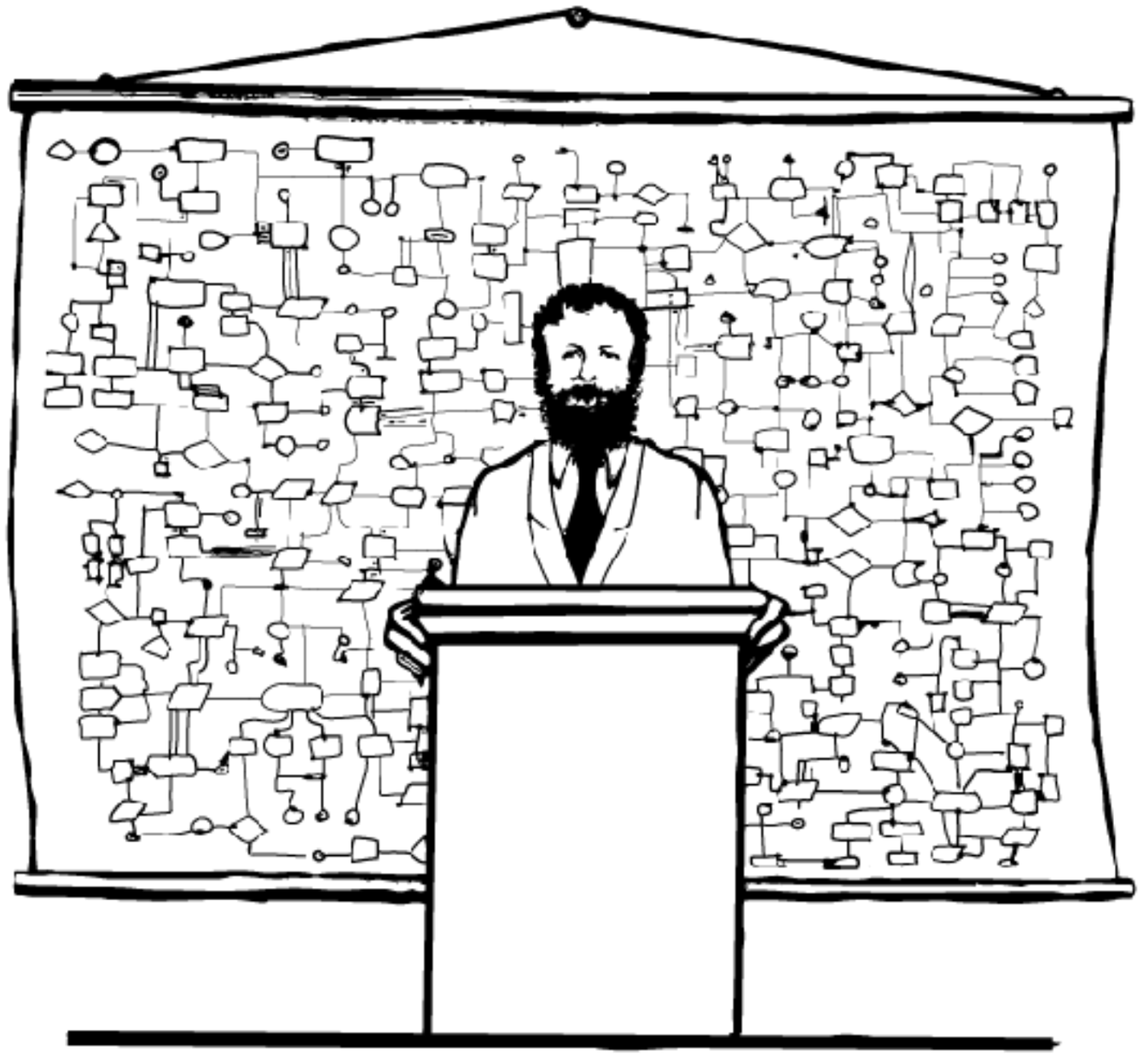
# Stiluri și perspective arhitecturale

- Arhitectura software de obicei prezintă anumite **perspective** ale sistemului, de obicei părțile cele mai importante

- Exemple de perspective:
  - proces
  - data flow
  - comportament
  - deployment
  - relații între module
  - etc.

# Exemple….

- în continuare, prezint câteva exemple de arhitecturi cu care am lucrat eu, iar apoi și alte exemple



"Now that you have an overview of the system, we're ready for a little more detail"

# Exemplul 1 - din domeniul Business Software (SAP)

## SAP is the world's leading business software company

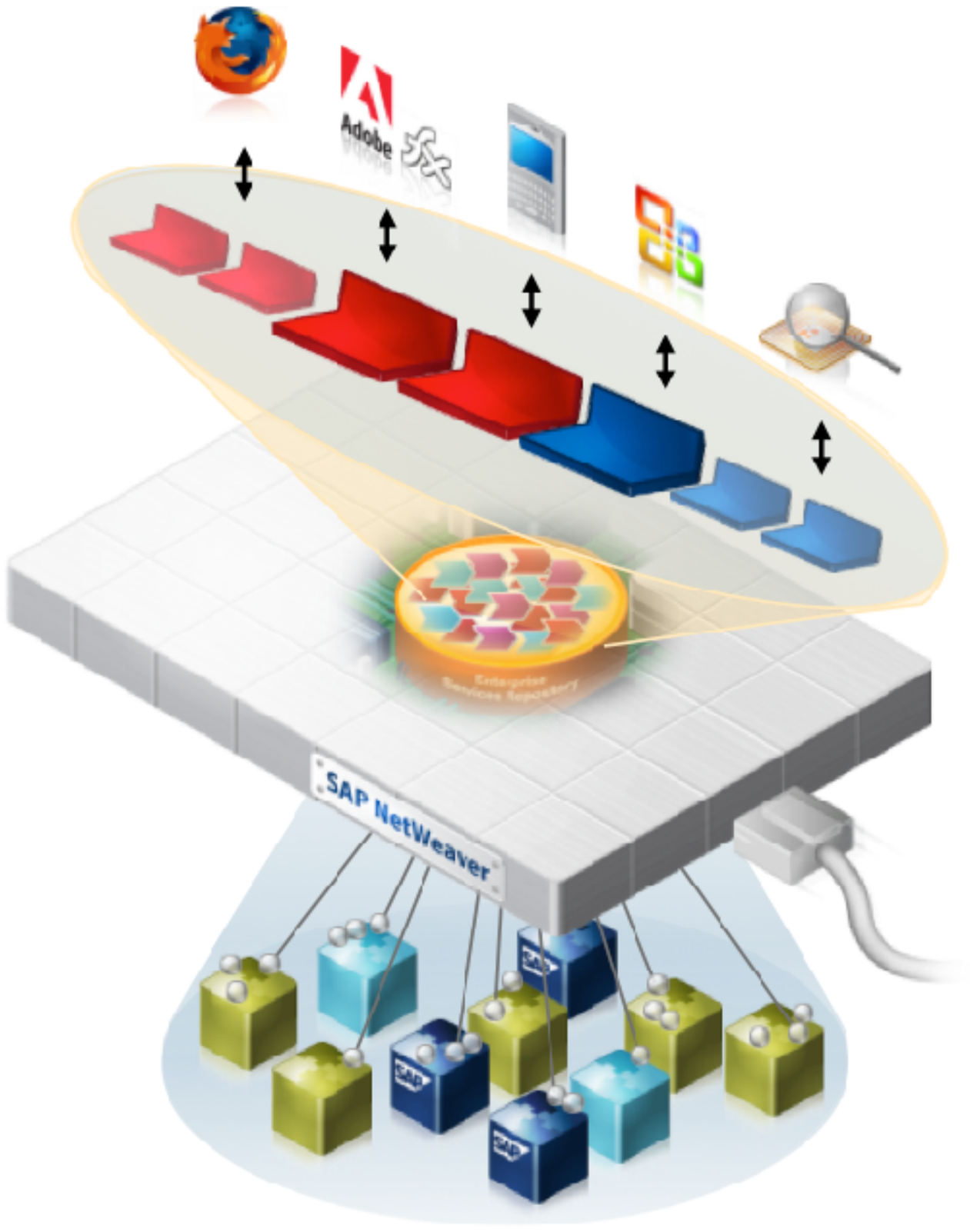
**SAP is**

- **#1** in enterprise applications
- **#1** in SME applications
- **#1** in business analytics
- **#1** in enterprise mobility

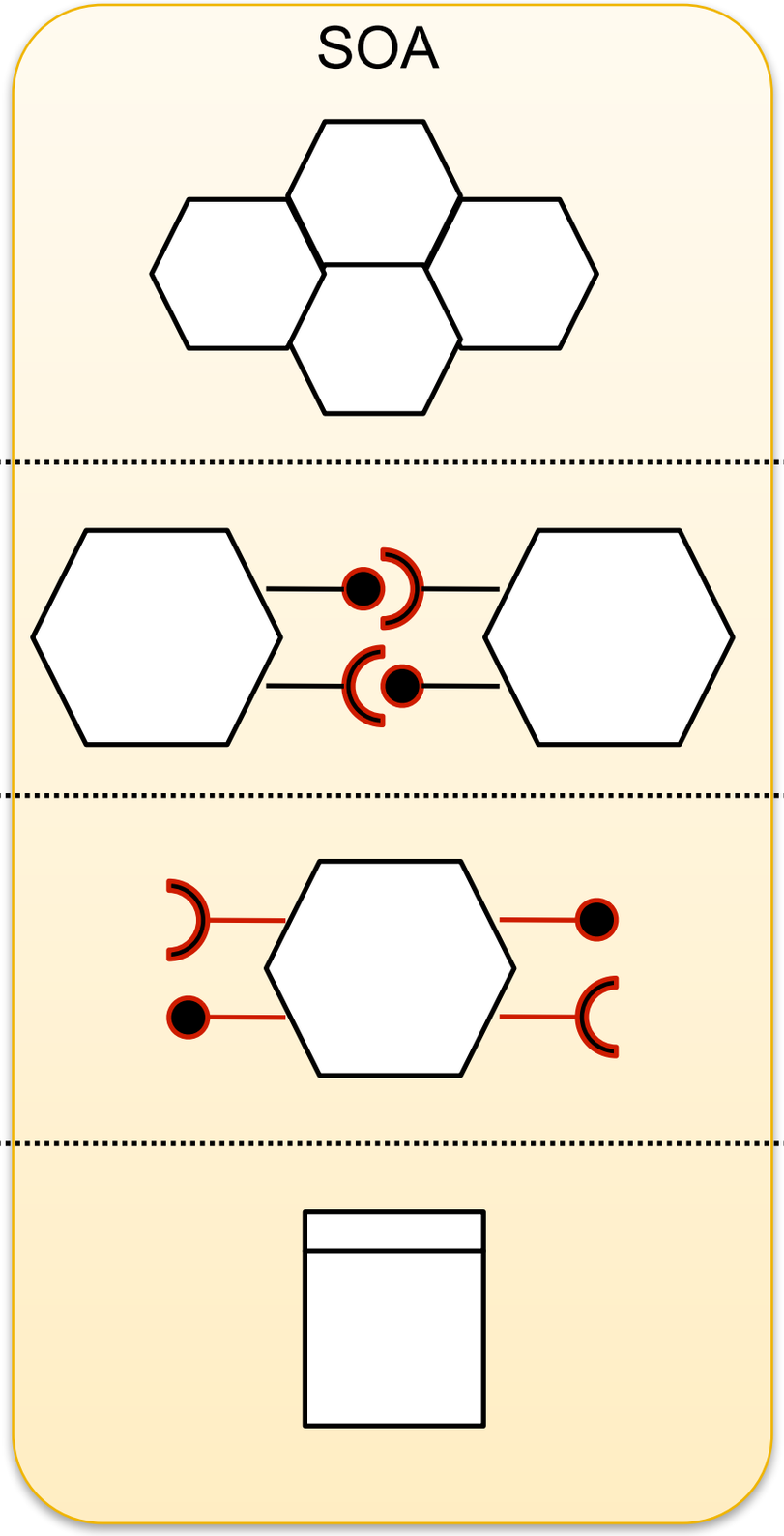
# Sisteme ERP (Enterprise Resource Planning)

| | | | | |
|---|---|---|---|---|
| | 3rd party Services | | | |
| Analytics | Strategic Enterprise Management | Financial Analytics | Operations Analytics | Workforce Analytics |
| Financials | Corporate Governance | Financial Accounting | Management Accounting | Financial Supply Chain Management |
| Human Capital Management | Employee Relationship Management | Employee Lifecycle Management | Employee Transaction Management | Workforce Deployment |
| Operations: Value Generation | Purchasing | Inventory Management | Manufacturing | Distribution | Sales Order Management | Service Order Management |
| Operations: Support | Product Structure Management | Project Management | Quality Management | Asset Management |
| Corporate Services | Travel Management | Environment, Health & Safety | Incentive & Commission Management | Real Estate Management |
| Solution and Integration Platform | People Integration | Information Integration | Process Integration | Application Platform |

# Stiva SOA (Service-oriented Architecture) la SAP



# Niveluri SOA




Scenariu integrând
mai multe componente

Integrare a
componentelor

Funcționalității oferită
prin **servicii**

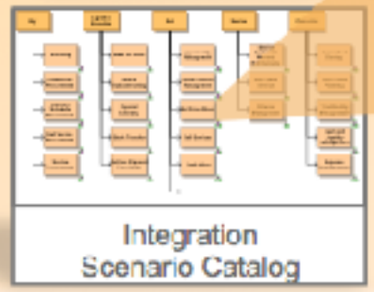
Dezvoltare
funcționalitate

# Modelare SOA la SAP

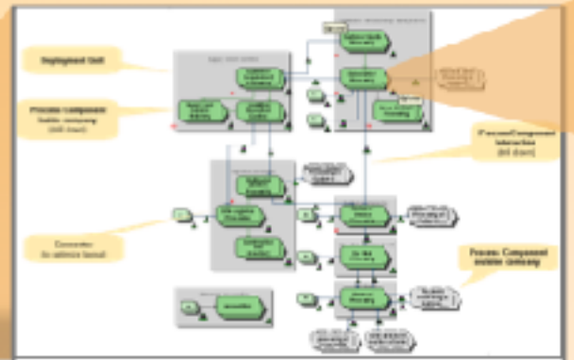
## Process Integration Models (in ARIS/ESR)

Business Scope (in Excel)

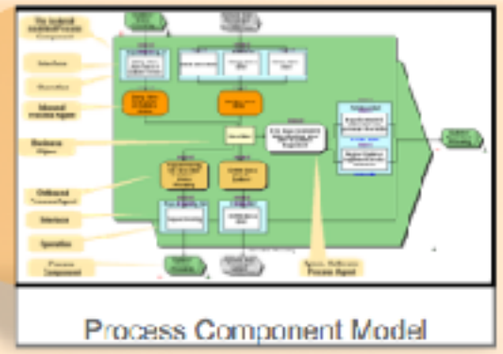
BO Map



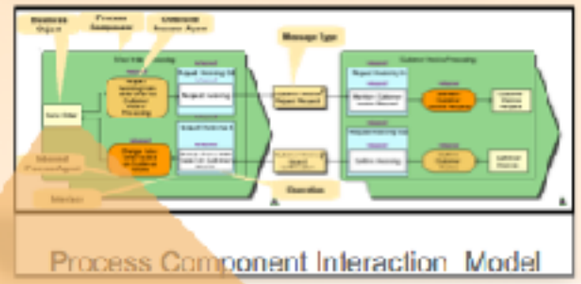
Integration Scenario Catalog



Integration Scenario Model

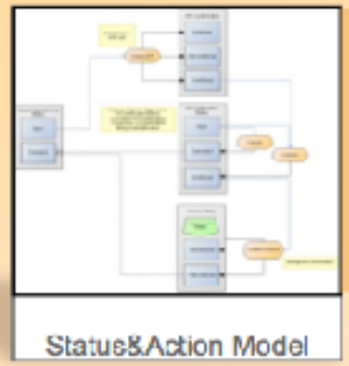


Process Component Model

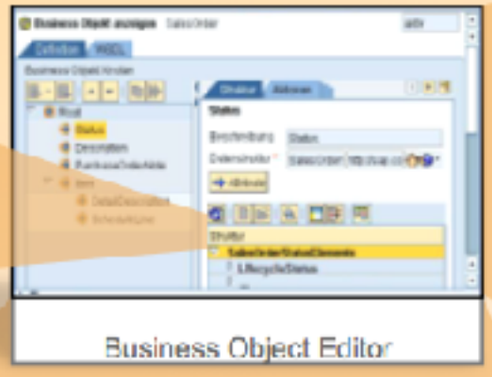


Process Component Interaction Model

## Business Object Models (in ESR)

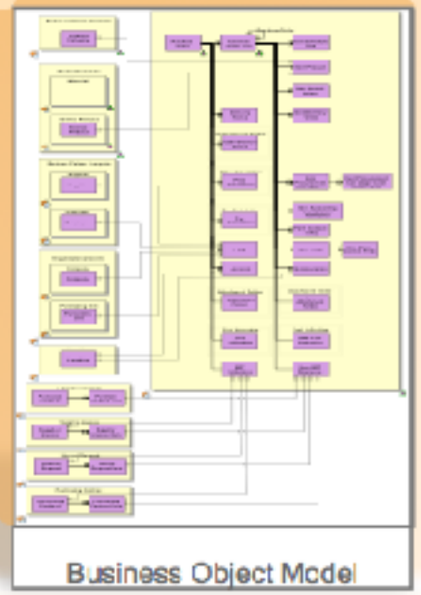


Status&Action Model



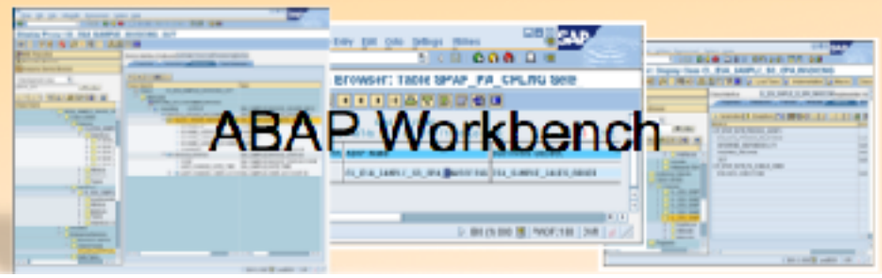
Business Object Editor

Proxy generation
in backend system

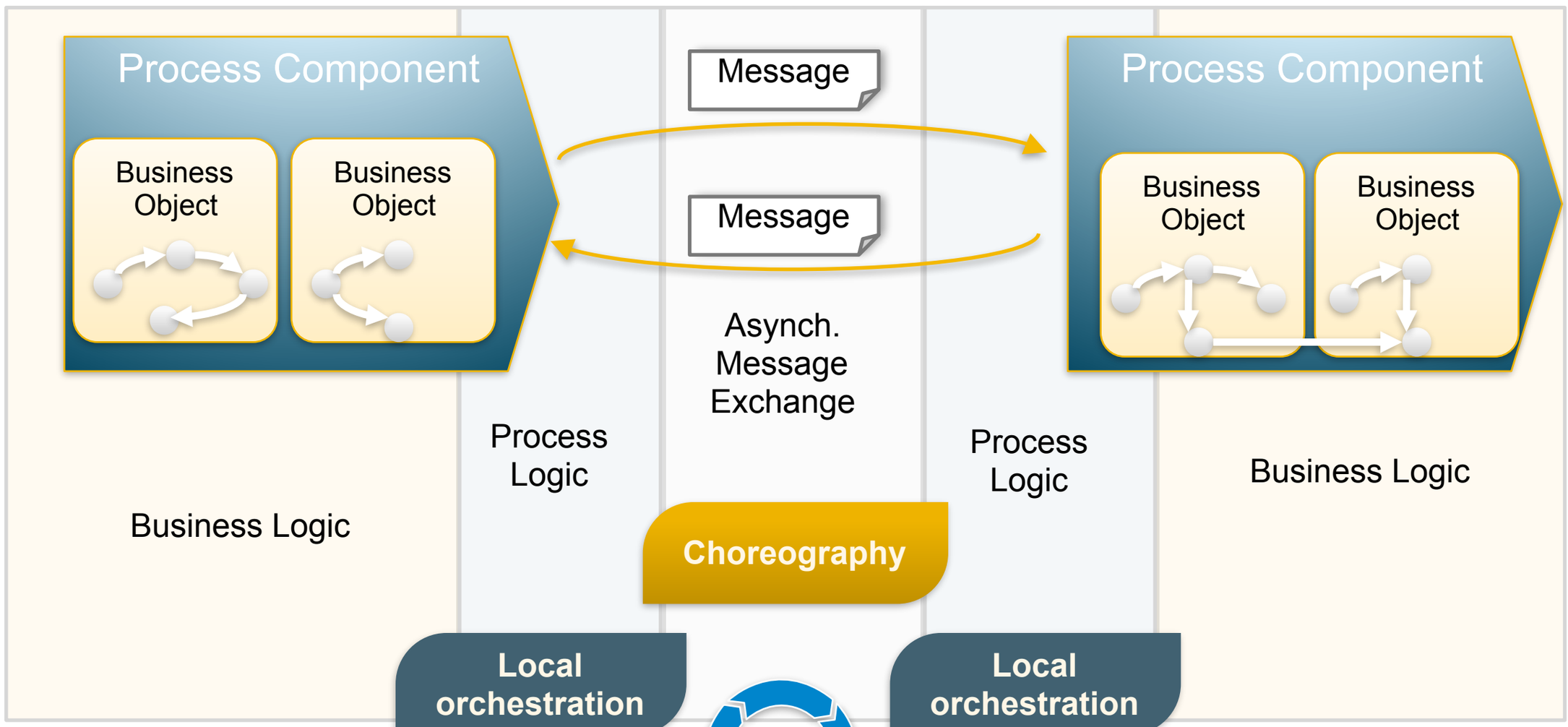


Business Object Model

## Implementation

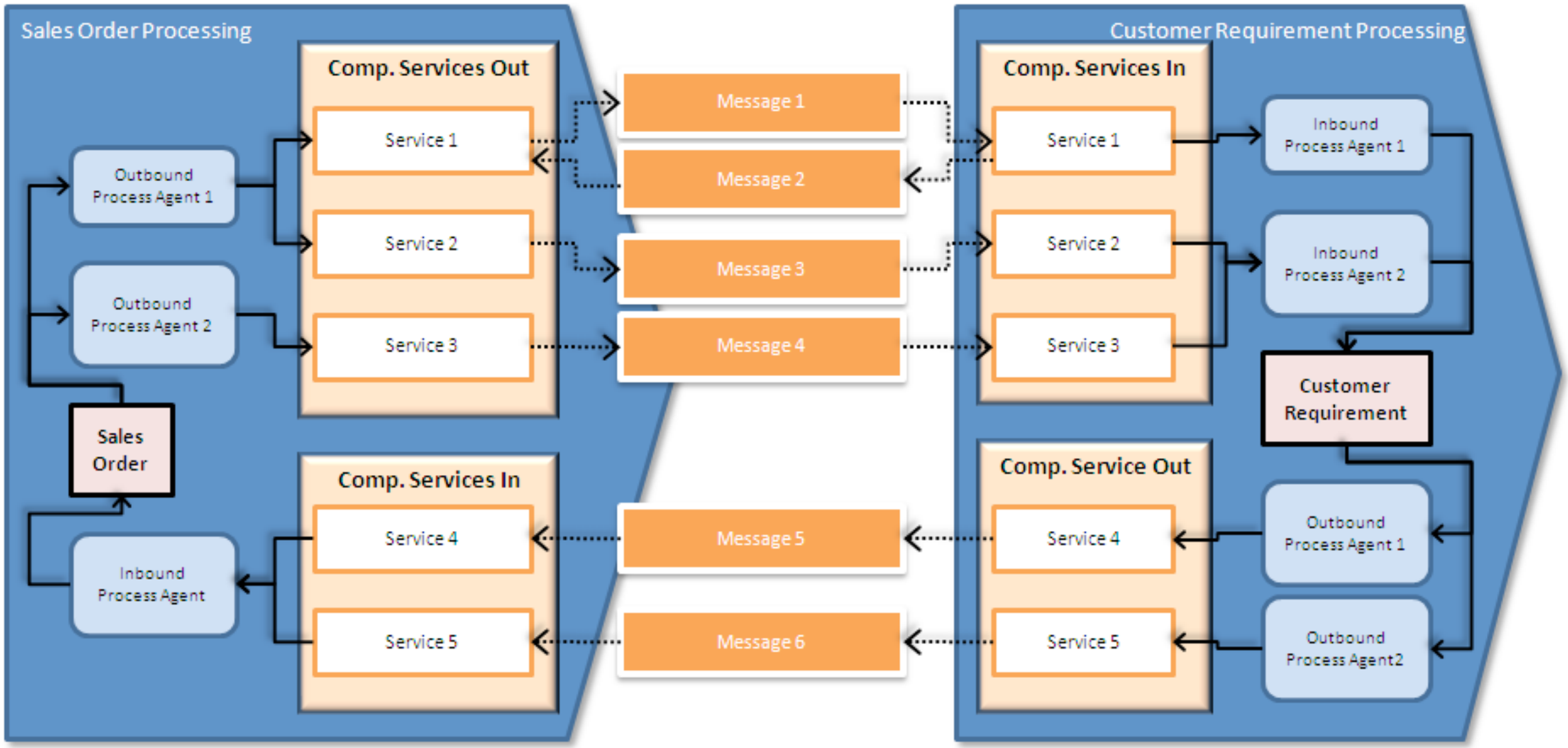


# Comunicare între componente bazată pe mesaje

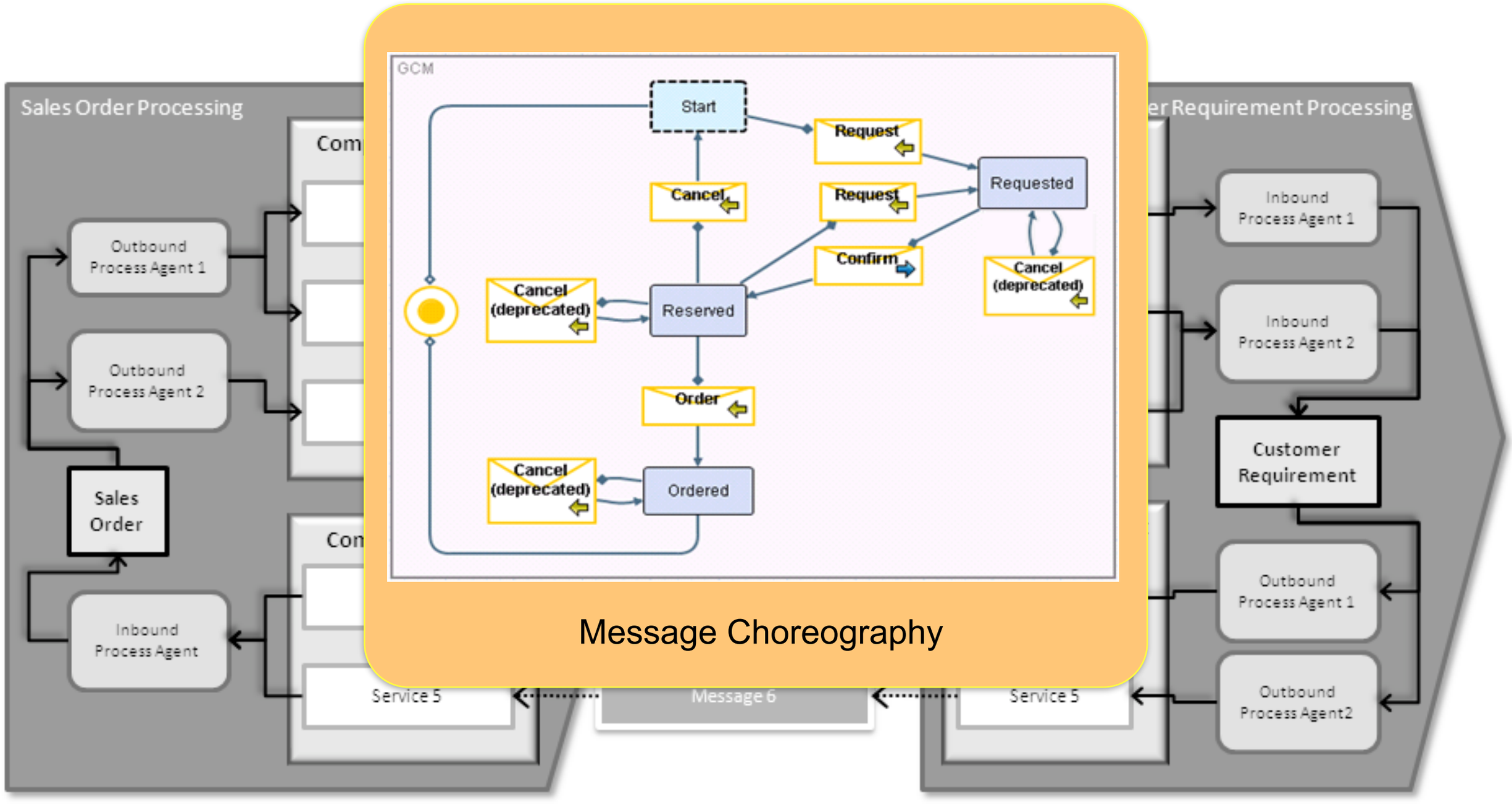



# Modele de interacție între componente

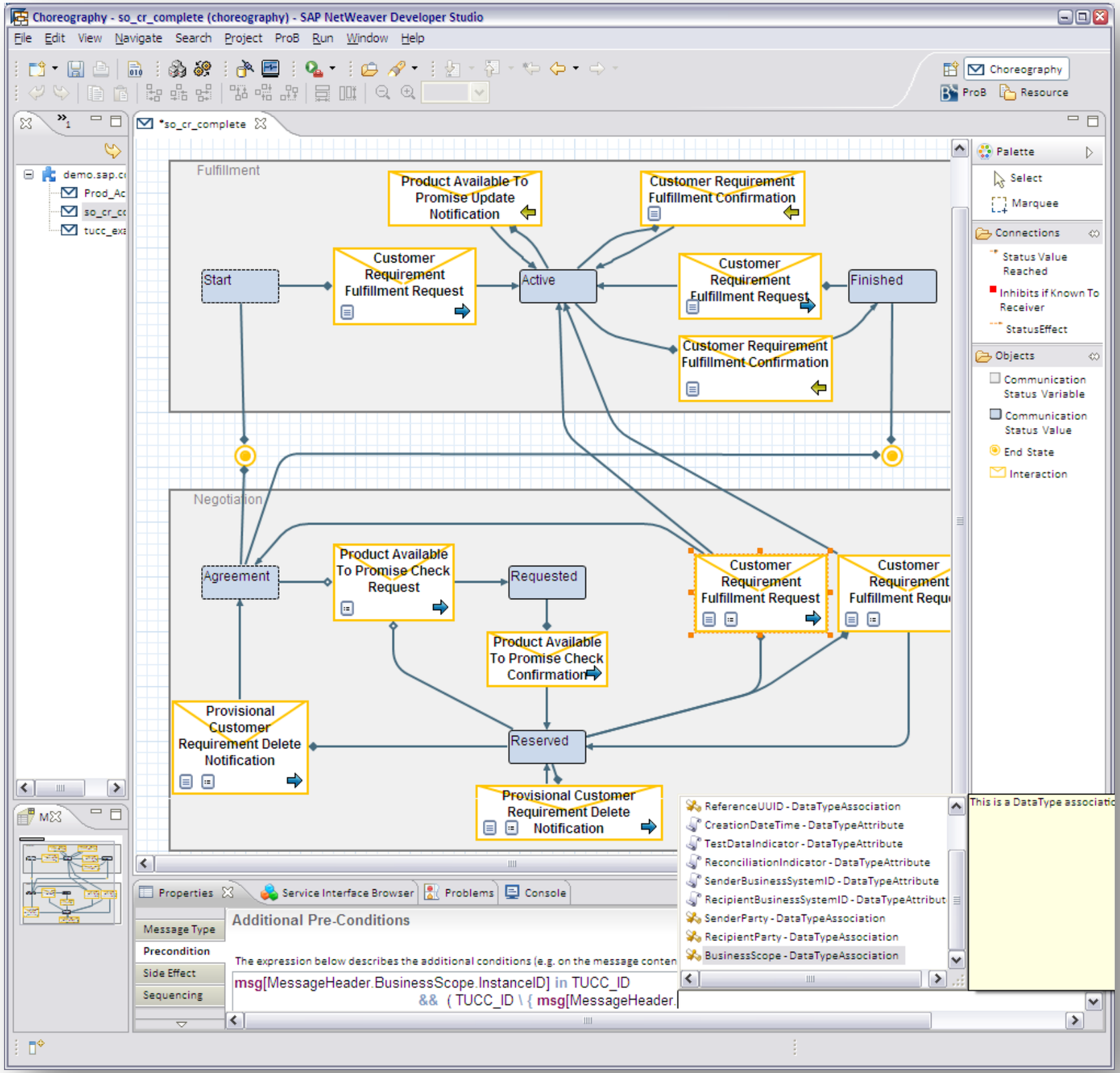
- comunicare bazată pe schimb de mesaje
- descriu doar canalele de comunicație și tipul lor



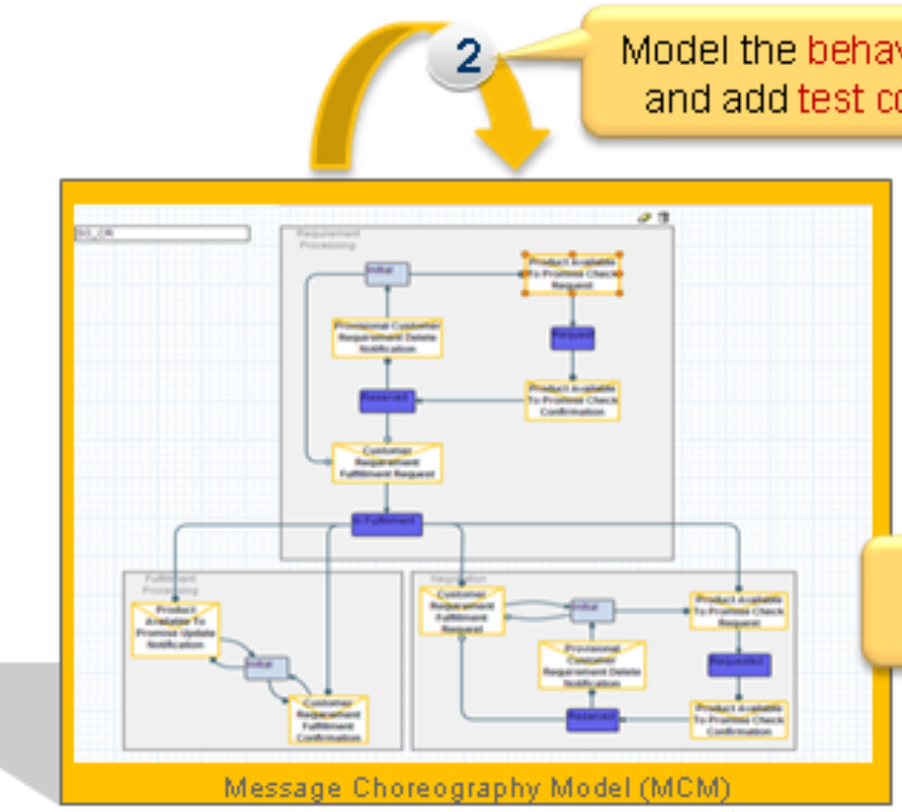
# Model pentru coreografie de mesaje



# Editor pentru modele de coreografii

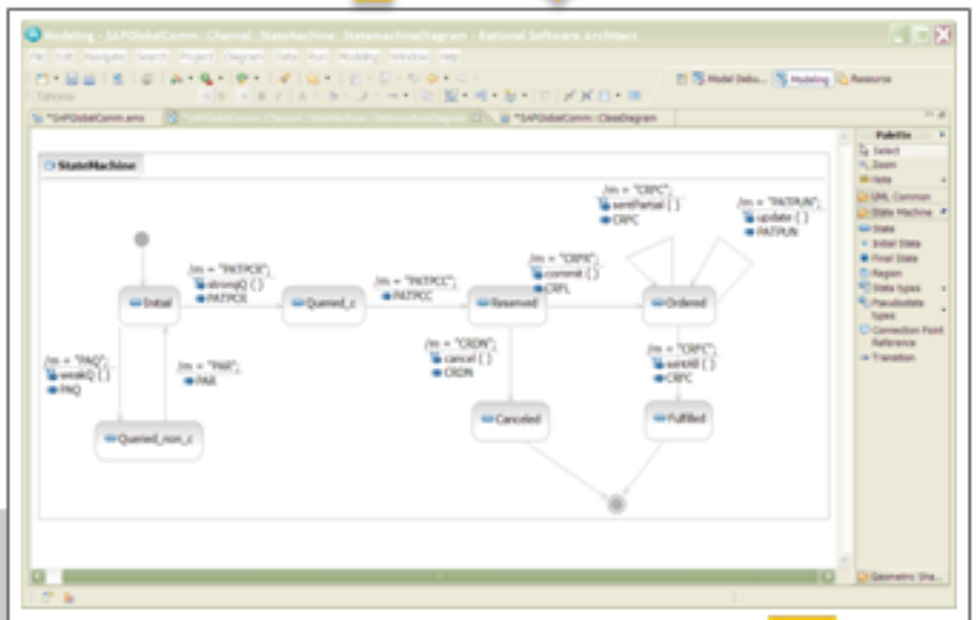


# Proces de generare de teste (implementat în proiectul european MODELPLEX)



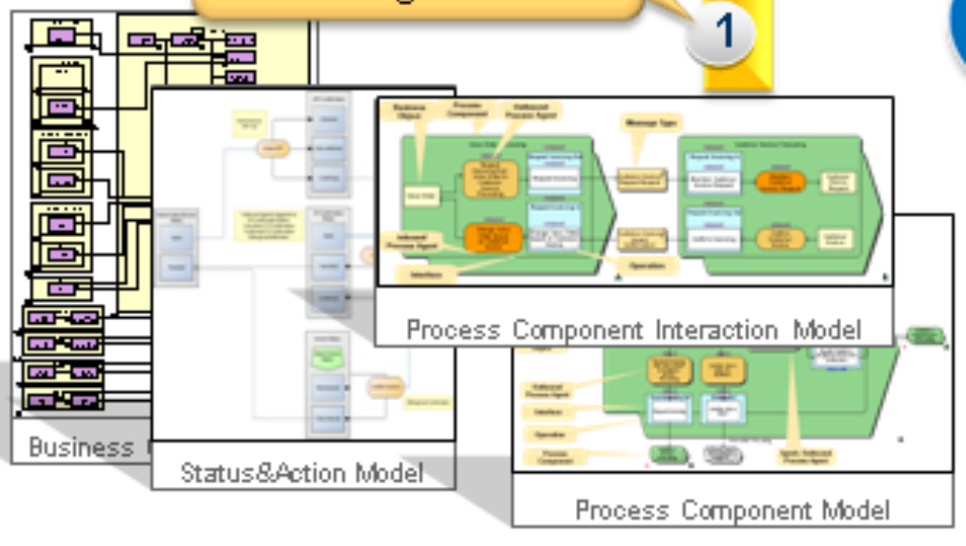

2 Model the behavior (MCM) and add test constraints

3 Transform MCM to UML



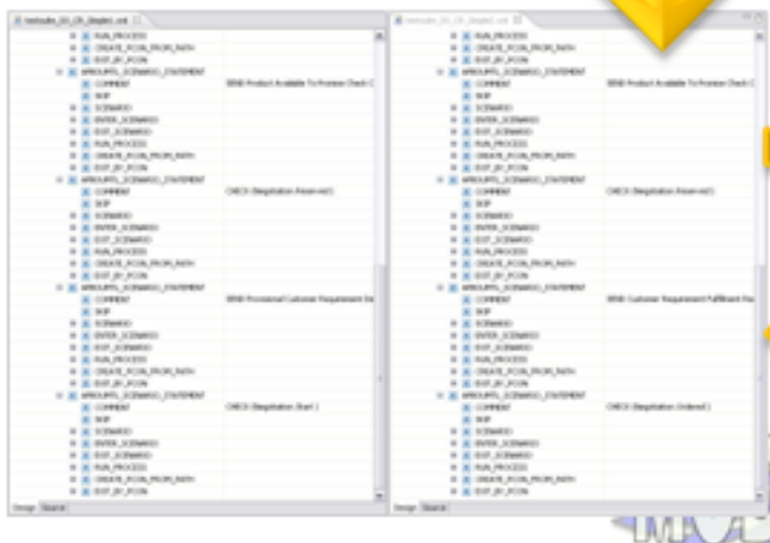

4 Debug the UML model

1 Import structural existing models



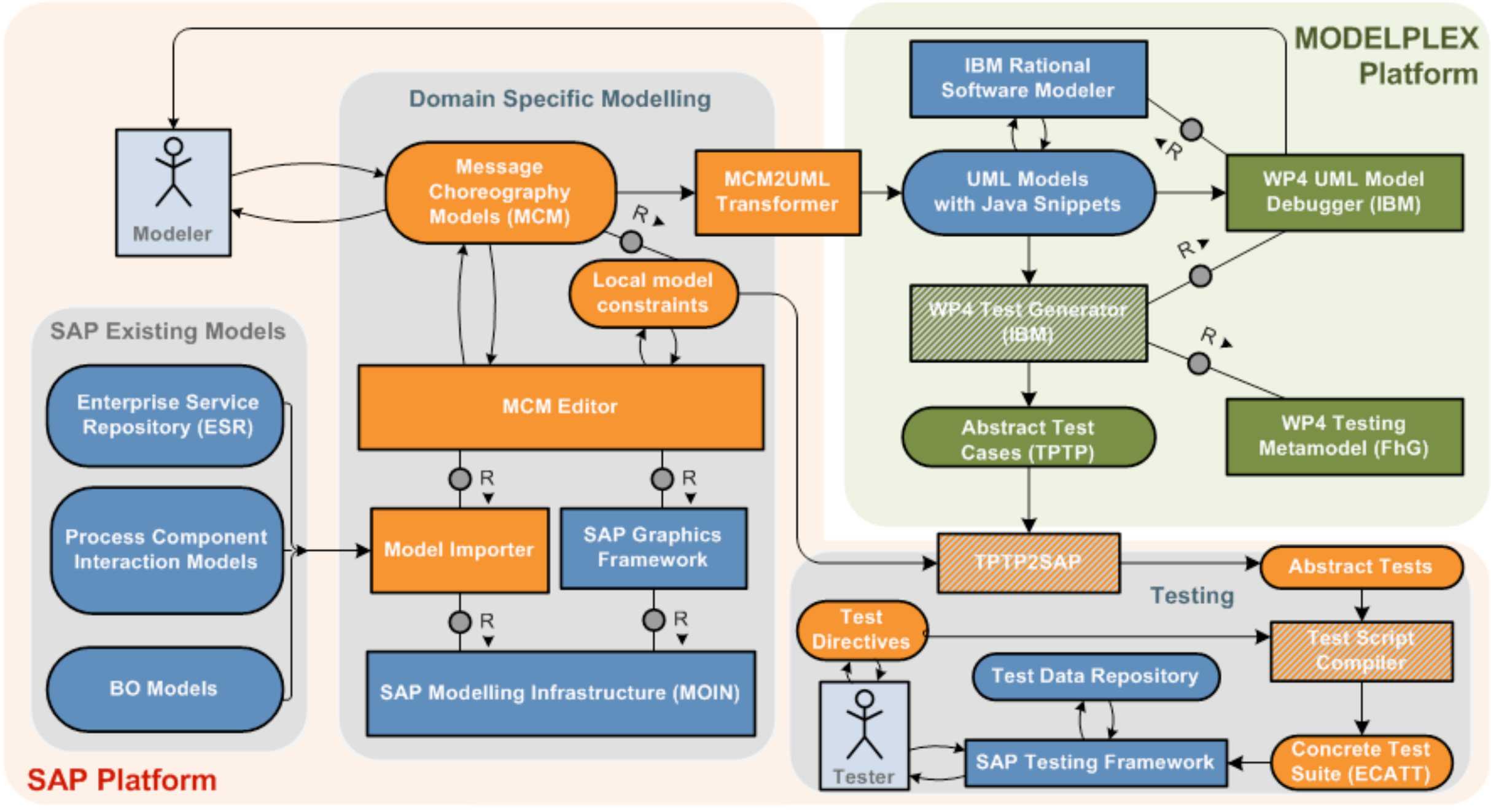


5 Generate test cases



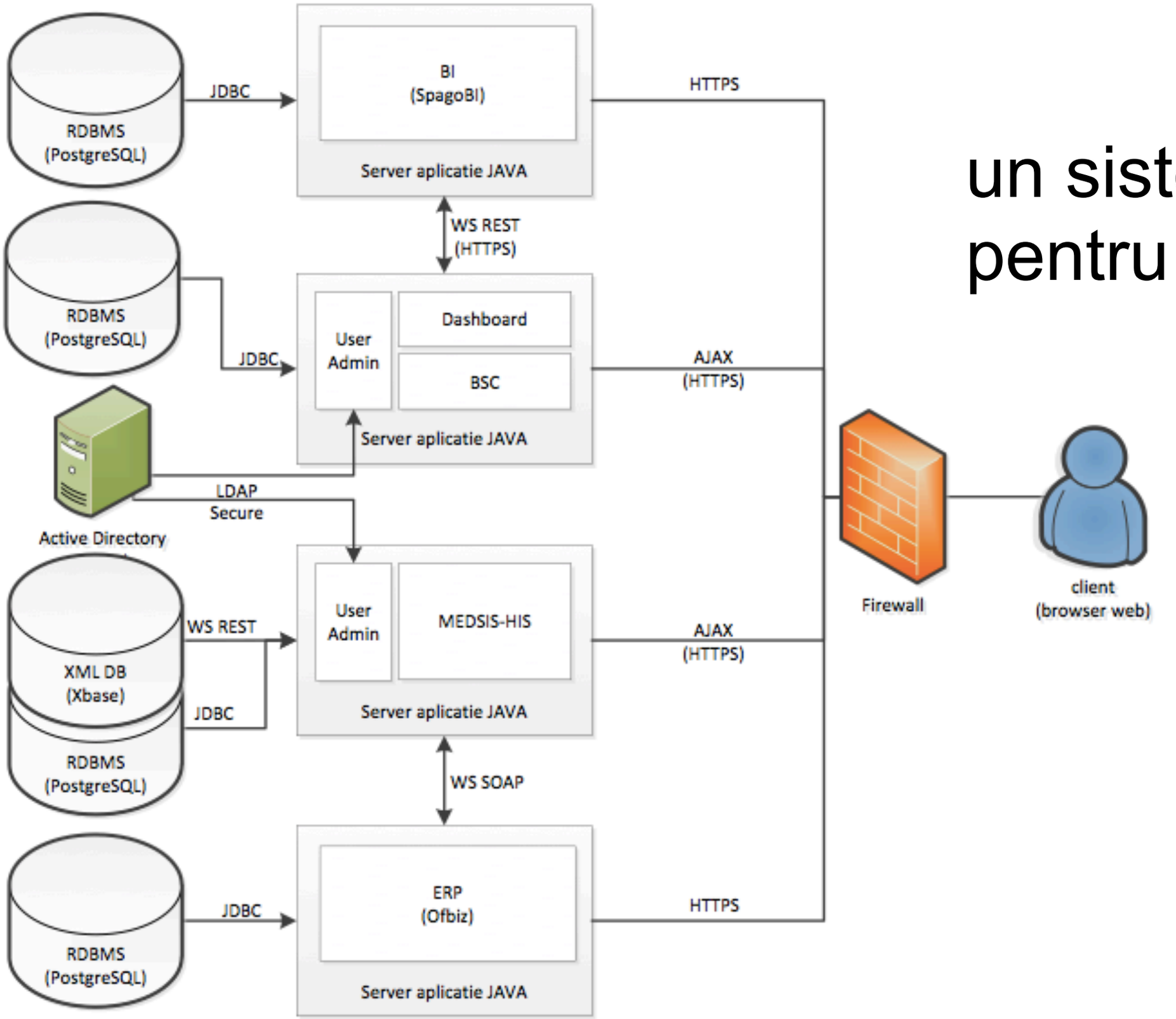
6 Generate test scripts and execute them

# Arhitectura implementării anterioare



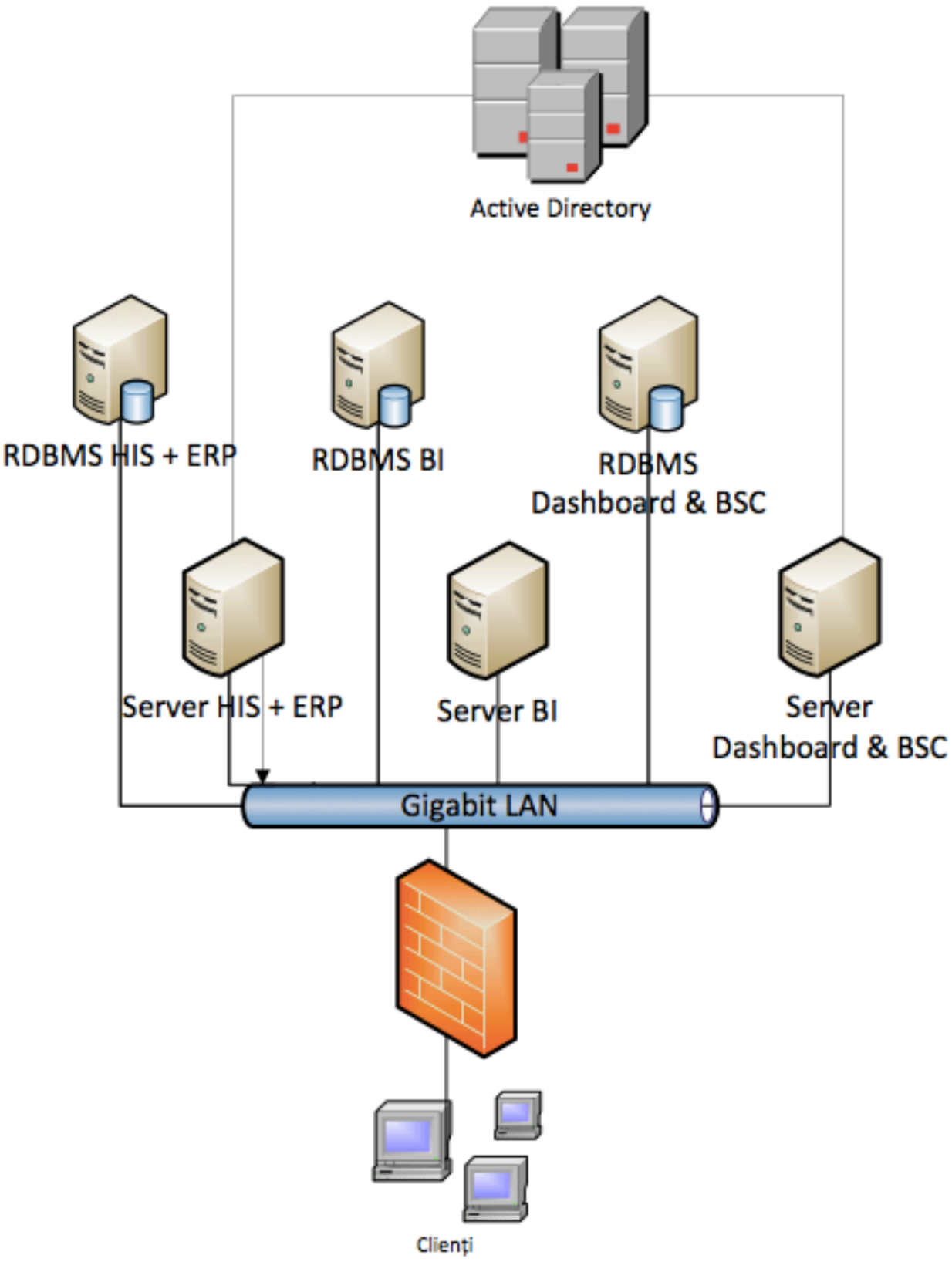

**Legend:** Already existing · Developed by SAP in MODELPLEX WP1 · Developed by WP4 in MODELPLEX · Work in progress

# Exemplul 2 - din domeniul medical




un sistem IT
pentru spitale

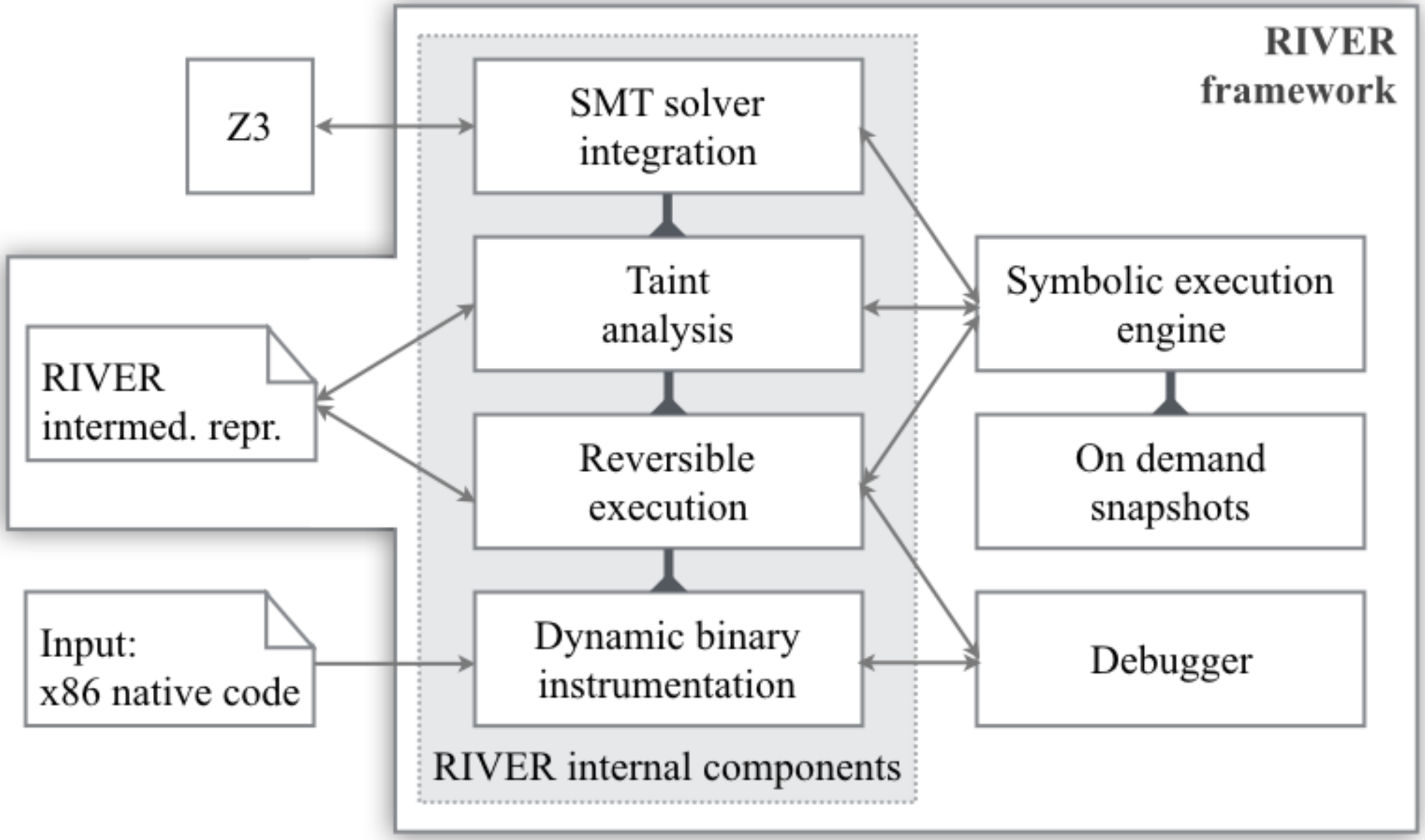
# Exemplul 2 - din domeniul medical



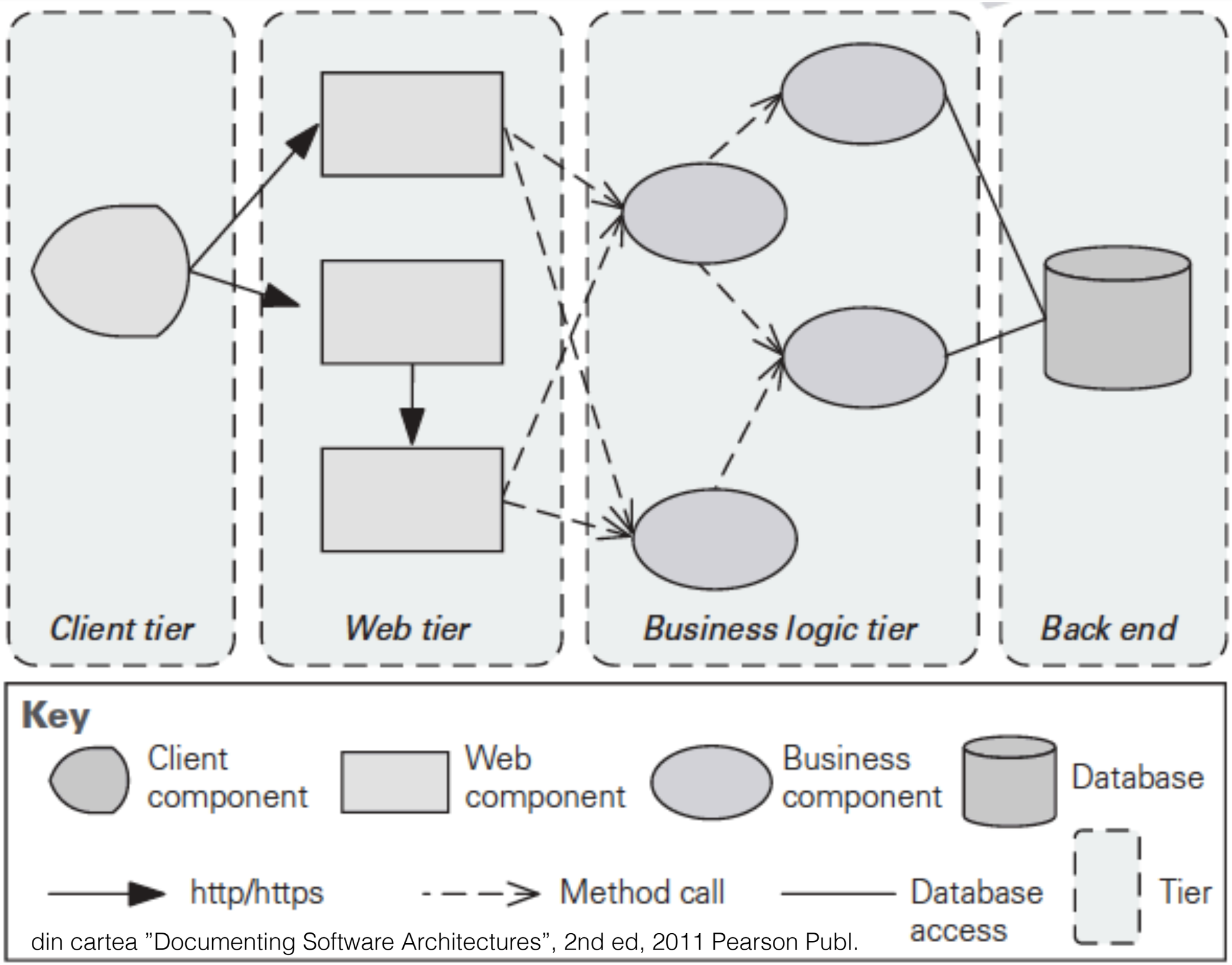

arhitectura hardware

# Exemplul 3 - analiza de executabile

arhitectura unui sistem de analiză dinamică pentru fișiere x86
(dintr-un proiect de cercetare în colaborare Bitdefender - UniBuc)

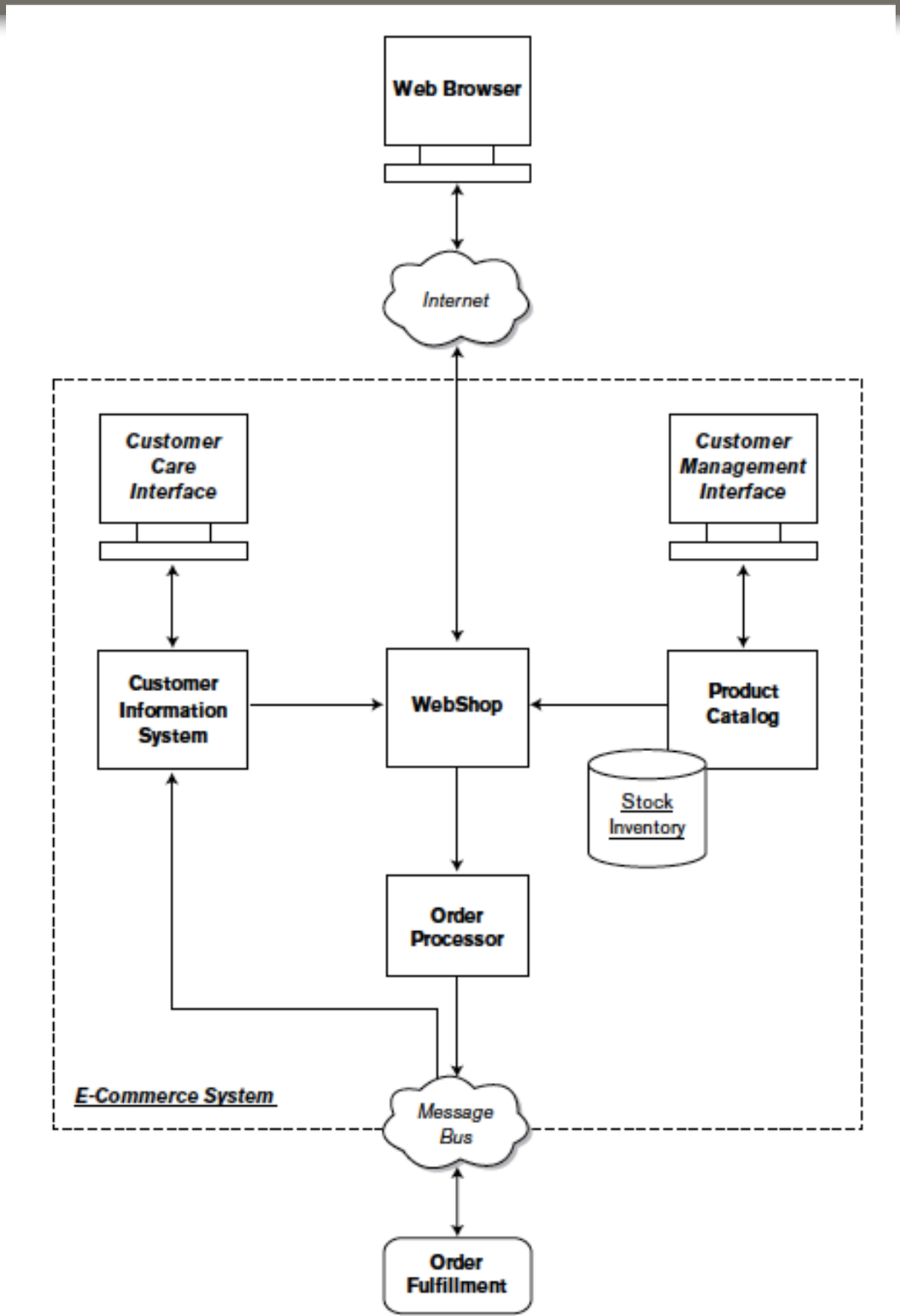


# Exemplul 4 - șablon pt. o aplicație web




din cartea ”Documenting Software Architectures”, 2nd ed, 2011 Pearson Publ.

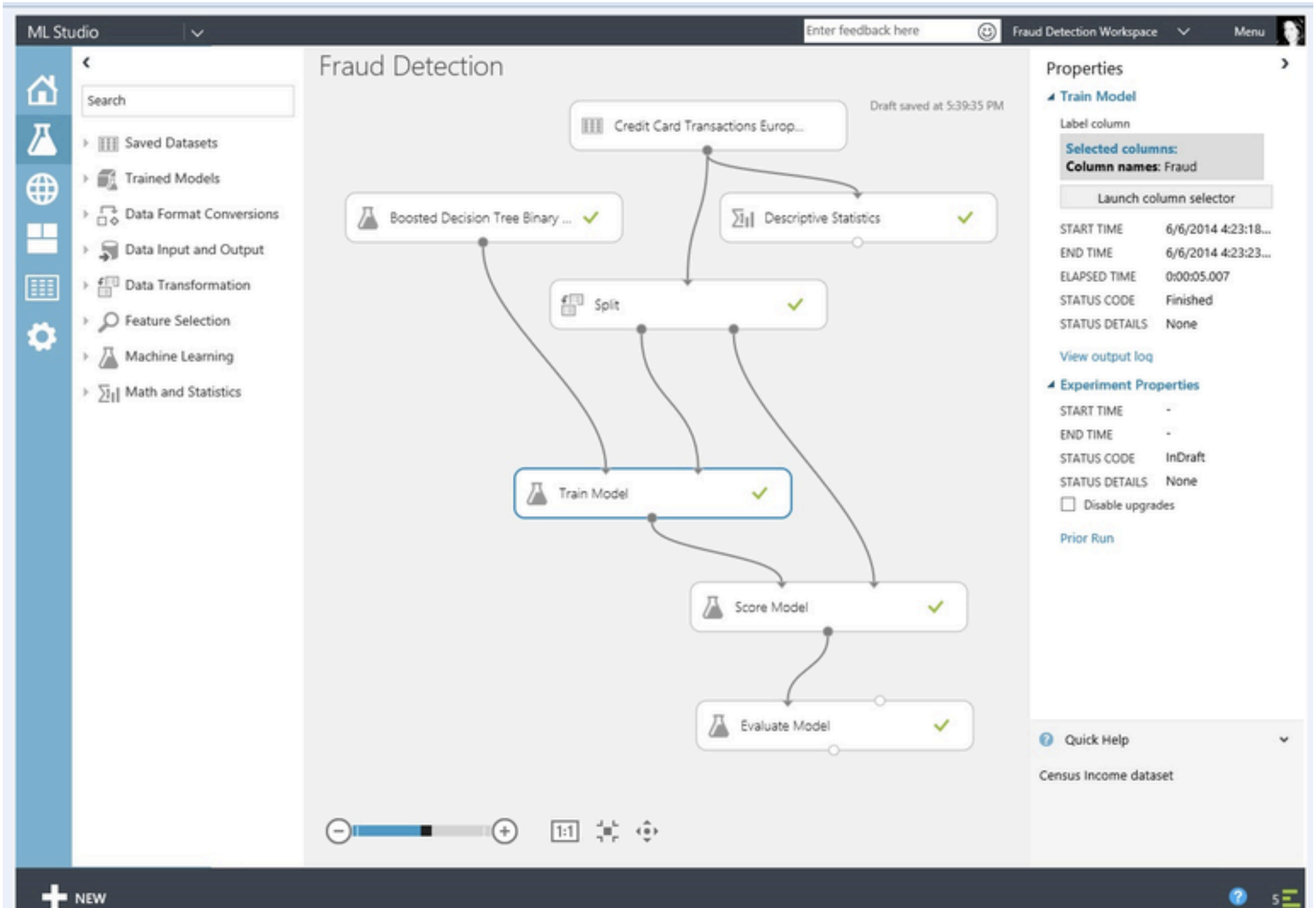
# Exemplul 5 - șablon pt. o aplicație web




din cartea ”Software System Architecture”, 2nd ed, 2008 Pearson Publ.

# Exemplul 6 - machine learning

Azure ML tool în care se poate face direct arhitectura ”executabilă”



# Exemplu  7 - modelare UML a unui joc

`http://rtb-team.sourceforge.net/rtb-team_analysis.htm`

- La linkul de mai sus este un exemplu de proiect software care este documentat prin diverse tipuri de diagrame UML