# Role-based Multi-Purpose Workflow Engine.

Sebastian Richly
**Sebastian Götz**
Uwe Aßmann
Dirk Habich

Funchal, Madeira, Portugal, 08.06.2010

# 01 Motivation

- big and fragmented WfMS ecosystem
  - many specializations exist

- new requirements, standards or domains
  ➔ adapt engine

- integration currently on a per-case base with high effort

- lack of **runtime flexibility**, that is
  - ability to adapt *unanticipated*

# 02 Concept

The features of our approach:

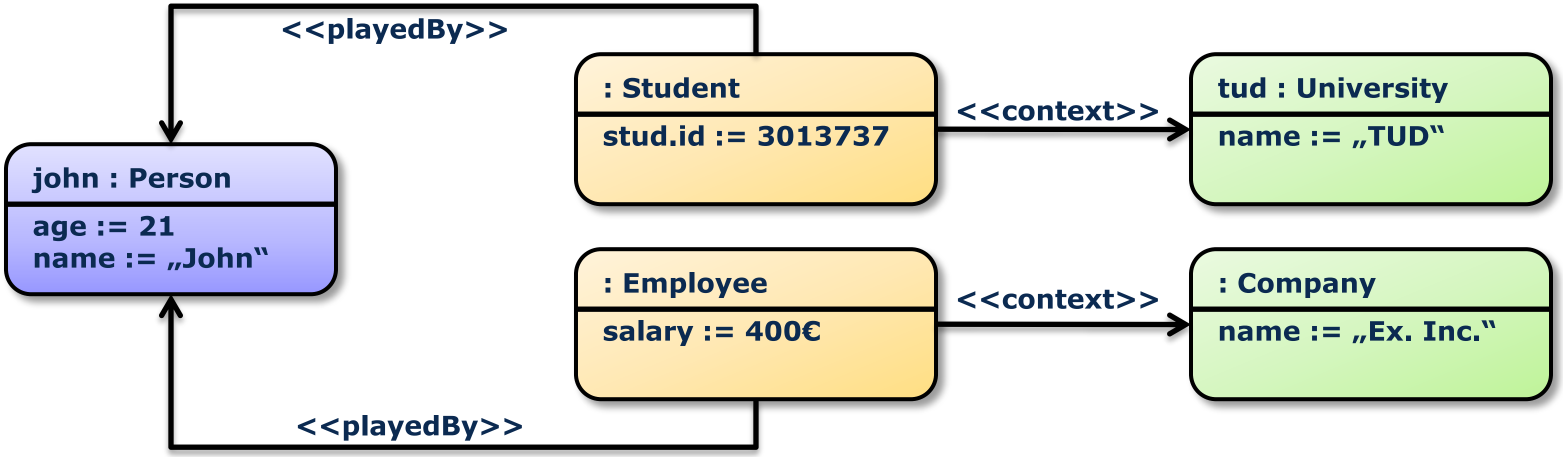
1. Support for all petri-net based process languages
2. dynamic extension of new element types
3. runtime flexibility
4. integration of different process languages
5. mix of domain concepts

## 1) Support for all petri-net based process languages

- Engine based on **Workflow Nets** [1]
  - restricted Petri Nets
    - requires input and output place
    - must not have dangling places
  - able to represent all petri net based workflow languages
  - prerequisite for integration of different workflow languages (*least common multiple*)
  - provides formal techniques of petri-nets

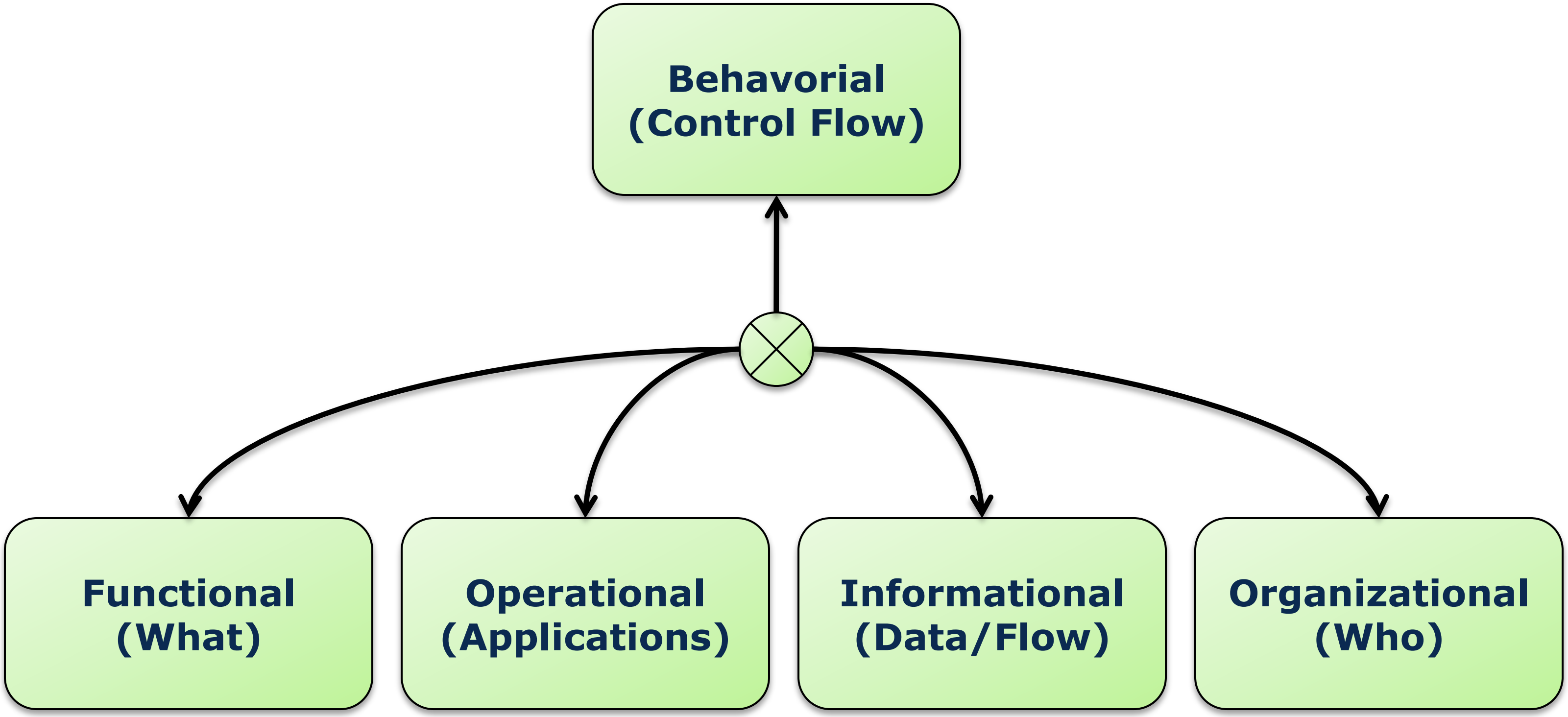
# 02 Concept

- Utilization of **Role-based Programming** [2]
  - Roles as extension to the object-oriented paradigm
  - Concept stems from theater
    - objects start and stop to play roles over time
    - objects are able to play multiple roles concurrently
    - played roles change behaviour and structure of player



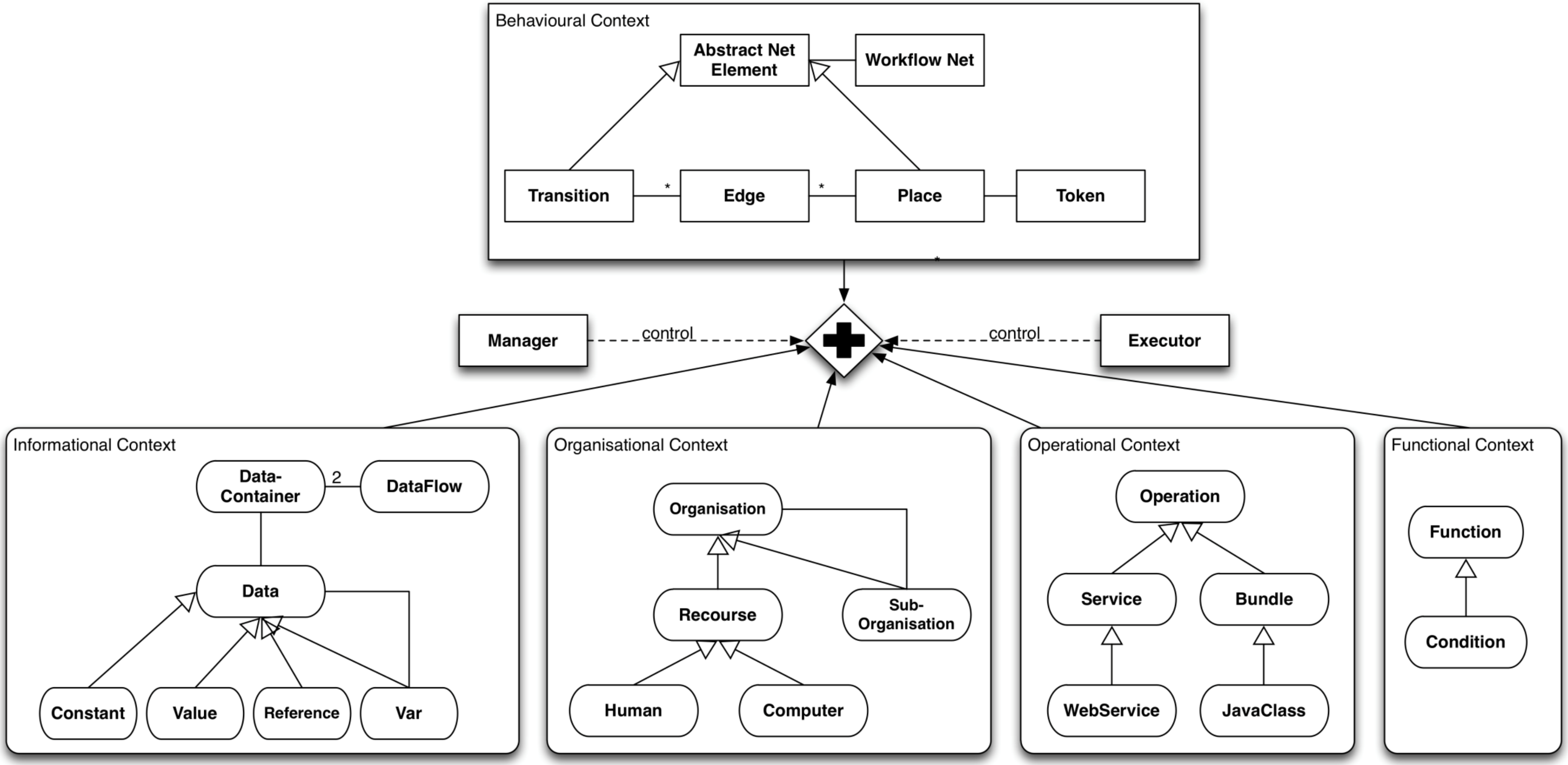
# 02 Concept

- 5 different **workflow aspects** exist[3] :

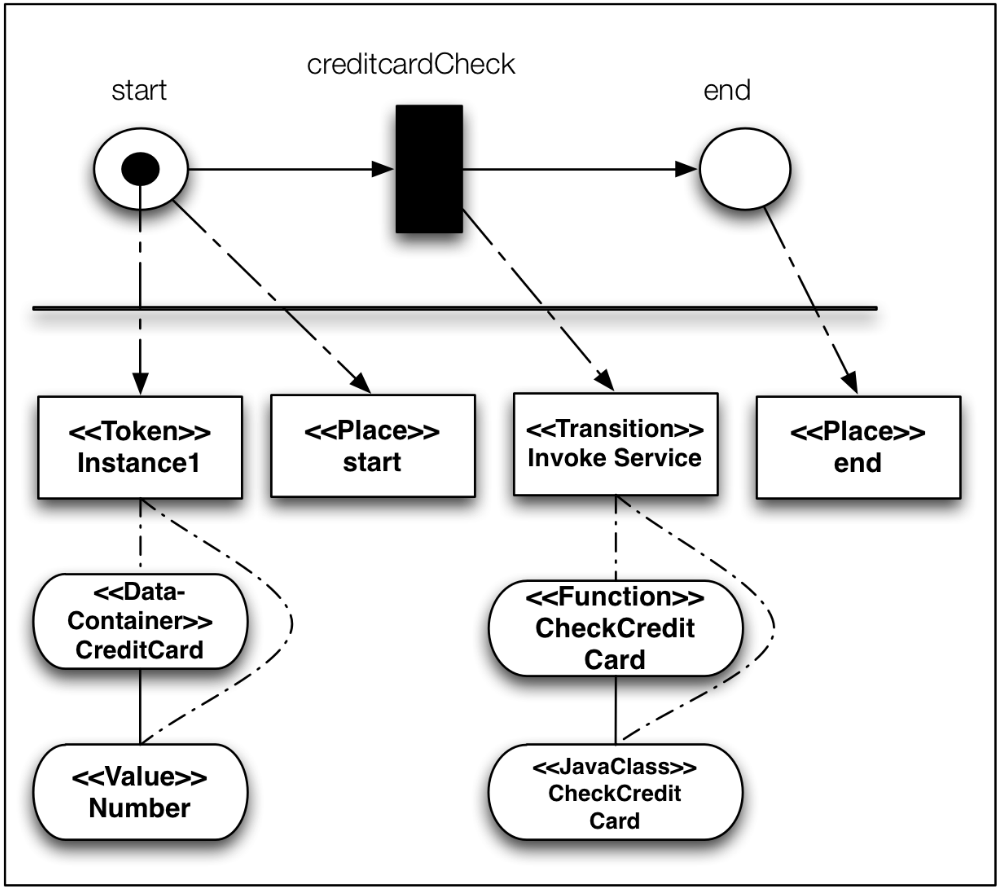


# 02 Concept

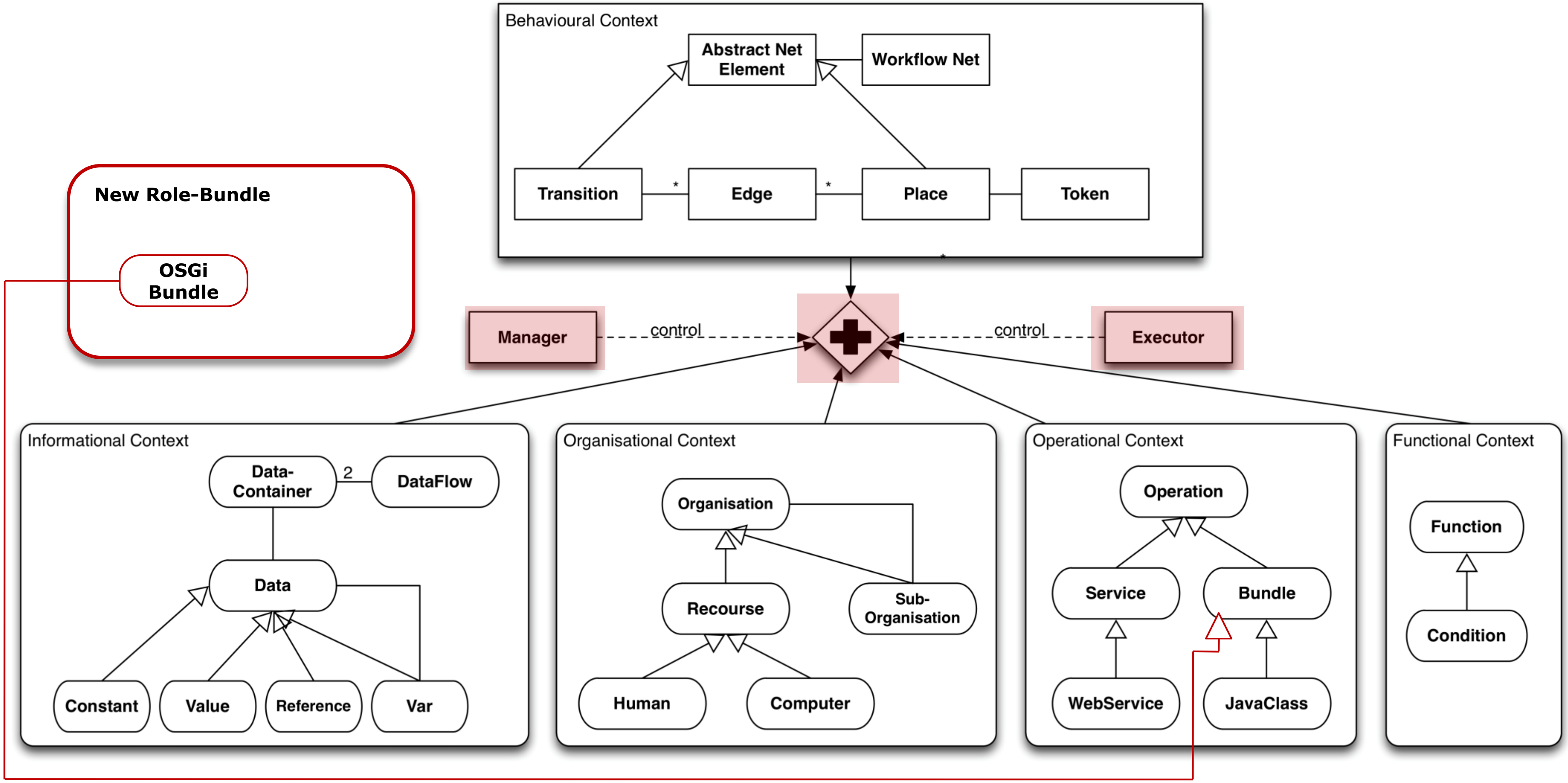
- Workflow Aspects in **Role Space**:



## Credit Card Example:

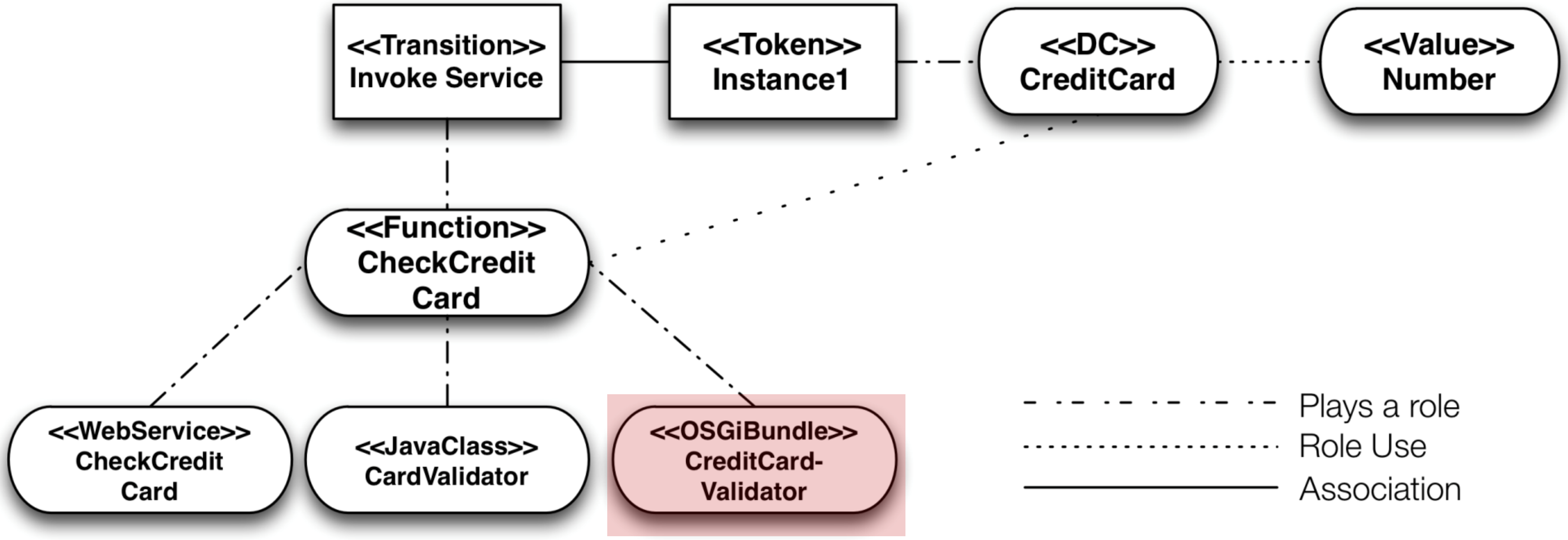

creditcardCheck
start
end
<<Token>> Instance1
<<Place>> start
<<Transition>> Invoke Service
<<Place>> end
<<Data-Container>> CreditCard
<<Function>> CheckCredit Card
<<Value>> Number
<<JavaClass>> CheckCredit Card

## 2) Dynamic extension of new element types



## 3) Runtime flexibility

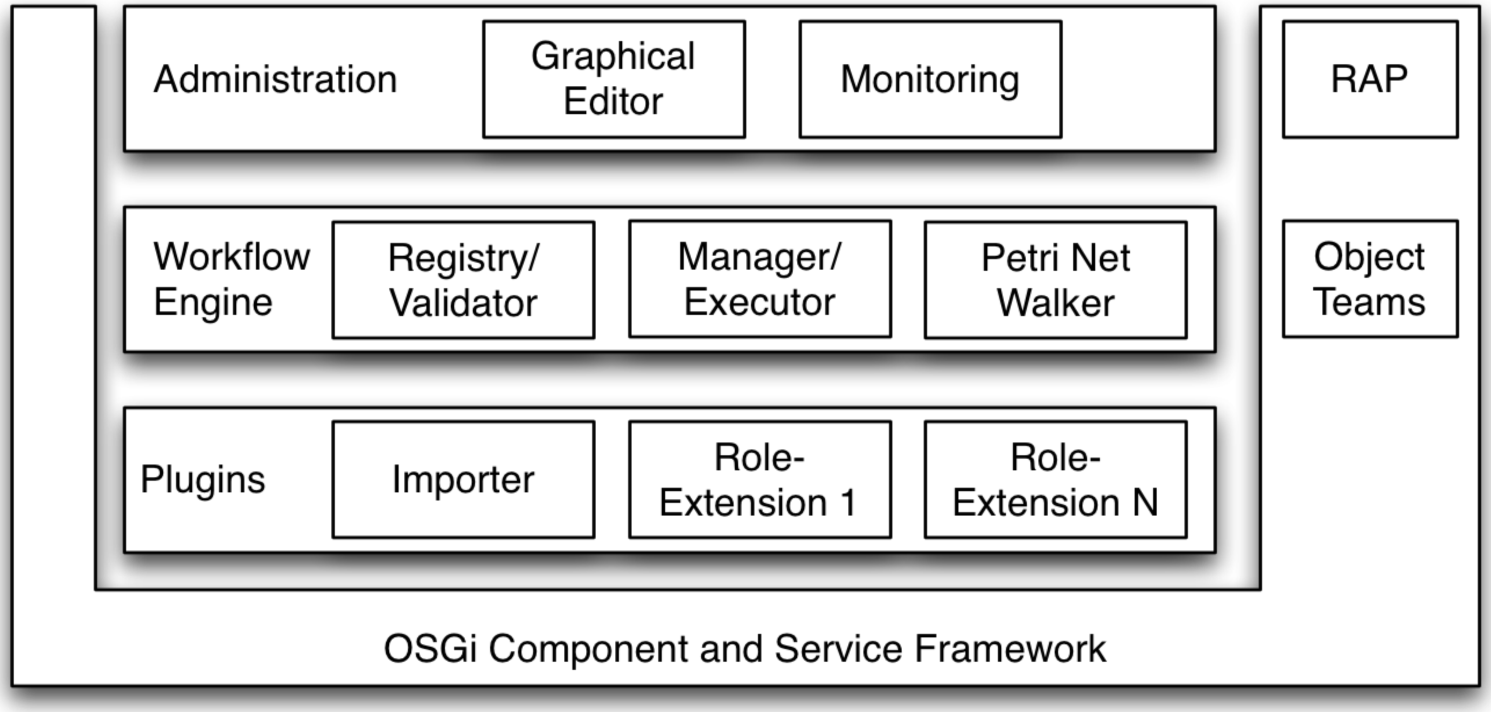
- proactively decide instead of react after failure
- change role set of running processes


<<Transition>> Invoke Service
<<Token>> Instance1
<<DC>> CreditCard
<<Value>> Number
<<Function>> CheckCredit Card
<<WebService>> CheckCredit Card
<<JavaClass>> CardValidator
<<OSGiBundle>> CreditCard-Validator
Plays a role
Role Use
Association

## 4+5) Integration and Intermixture

- possible due to workflow nets
    - language importer written
    - First control flow concepts are mapped
    - Second resources and activities are mapped
- mix of domain concepts possible
    - Tasks can play multiple roles at the same time
    - even if these roles stem from different imported languages!

# 03 Implementation



**http://141.76.65.194/OSPP**

# 04 Related Work

- **XPDL** for interoperability between workflow languages, defined by WfMC
- **XRL**[4] (eXtendible Routing Language) also based on Workflow Nets, changes only at design time (not runtime)
- **AO4BPEL**[5] allows to weave in tasks at runtime, but is tailored to specific languages
- **XSLT transformations** for interoperability of workflow languages, like in [6]

# 05 Conclusion

1. Support for all petri-net based process languages

   ➔ because based on workflow nets

2. dynamic extension of new task types

   ➔ tasks start to play additional roles

3. runtime flexibility

   ➔ proactively decide instead of react on failure

   ➔ extend/change at runtime using role bundles

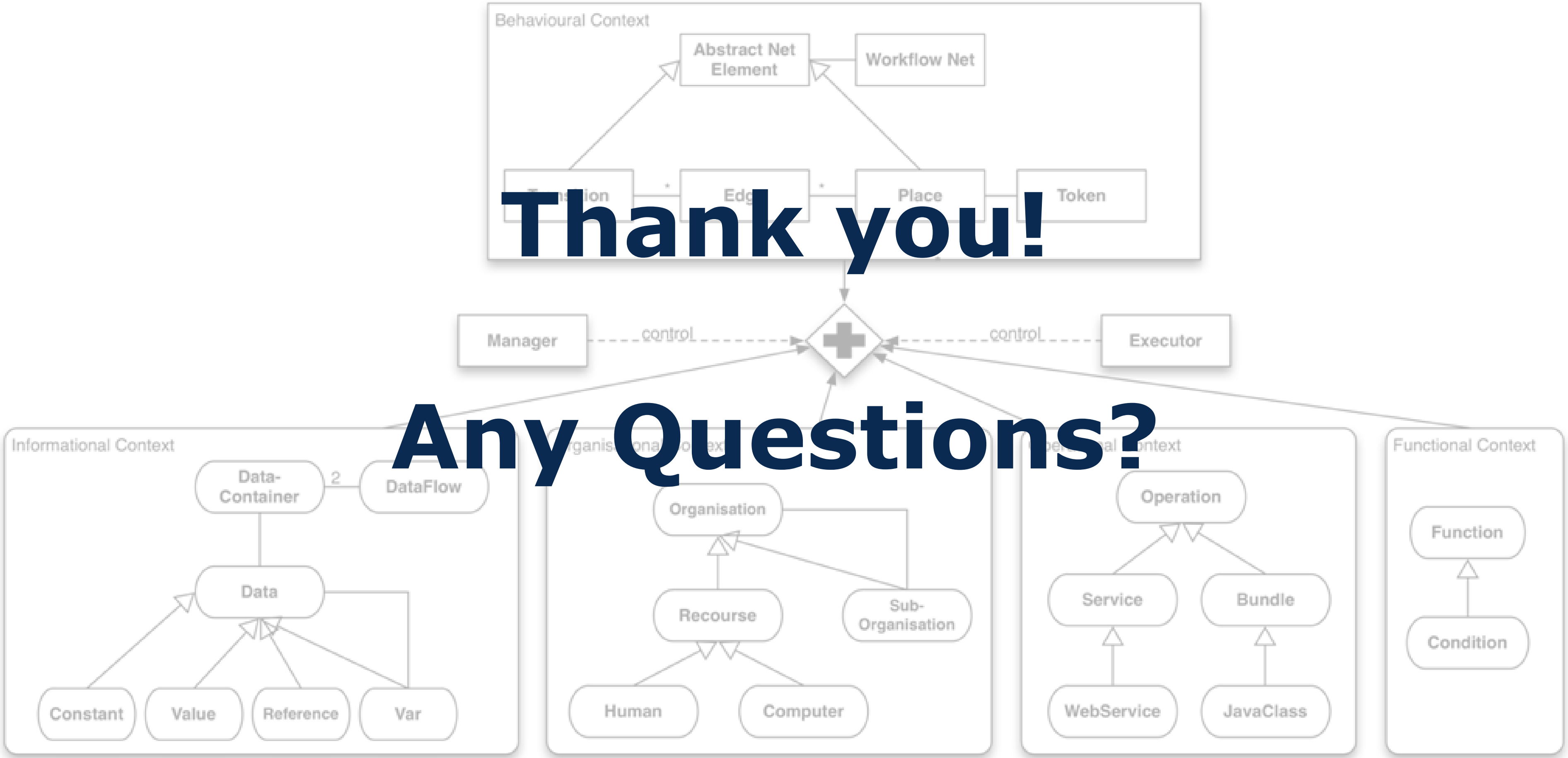
4. integration of workflow languages

   ➔ due to workflow nets

   ➔ domain concept mix possible due to roles

# Thank you!

# Any Questions?



# 06 Literature

[1] Van Der Aalst, W.M.P.: Verification of workflow nets. In: ICATPN ’97: Proceedings of the 18th International Conference on Application and Theory of Petri Nets, London, UK, Springer-Verlag (1997) 407–426

[2] Steimann, F.: On the representation of roles in object-oriented and conceptual modelling. Data & Knowl. Eng. (2000) 83–106

[3] Petkov, S., Oren, E., Haller, A. : Aspects in Workflow Management. In: Technical Report DERI TR 2005-04-10, DERI, 2005

[4] Verbeek, H.M.W., Van Der Aalst, W.M.P., Kumar, A.: Xrl/woflan: Verification and extensibility of an xml/petri-net-based language for inter-organizational workflows. Inf. Technol. and Management (2004) 65–110

[5] Charfi, A., Mezini, M.: Aspect-oriented web service composition with AO4BPEL. In Zhang, L.J., ed.: ECOWS, Springer (2004) 168–182

[6] Huang, L., Akram, A., Allan, R., Walker, D.W., Rana, O.F., Huang, Y.: A workflow portal supporting multi-language interoperation and optimization: Research articles. Concurr. Comput. : Pract. Exper. (2007) 1583–1595