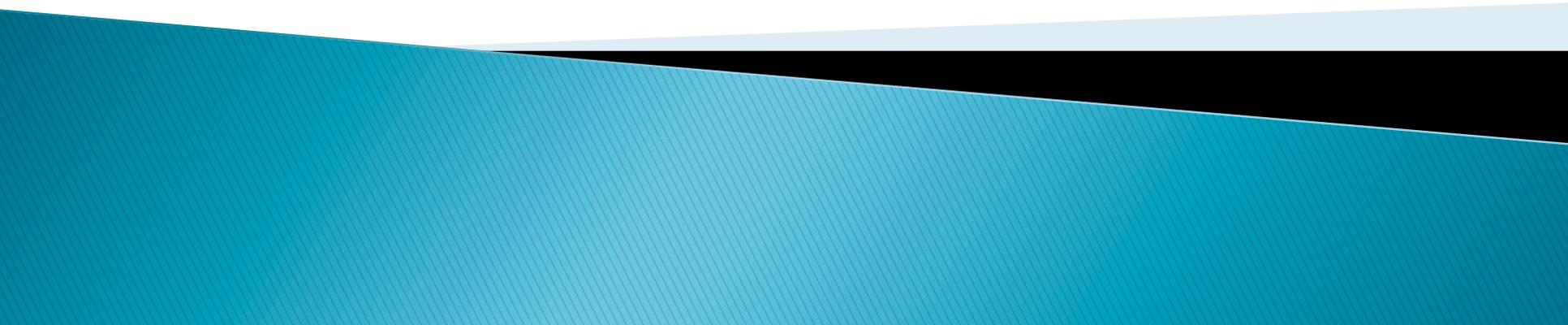


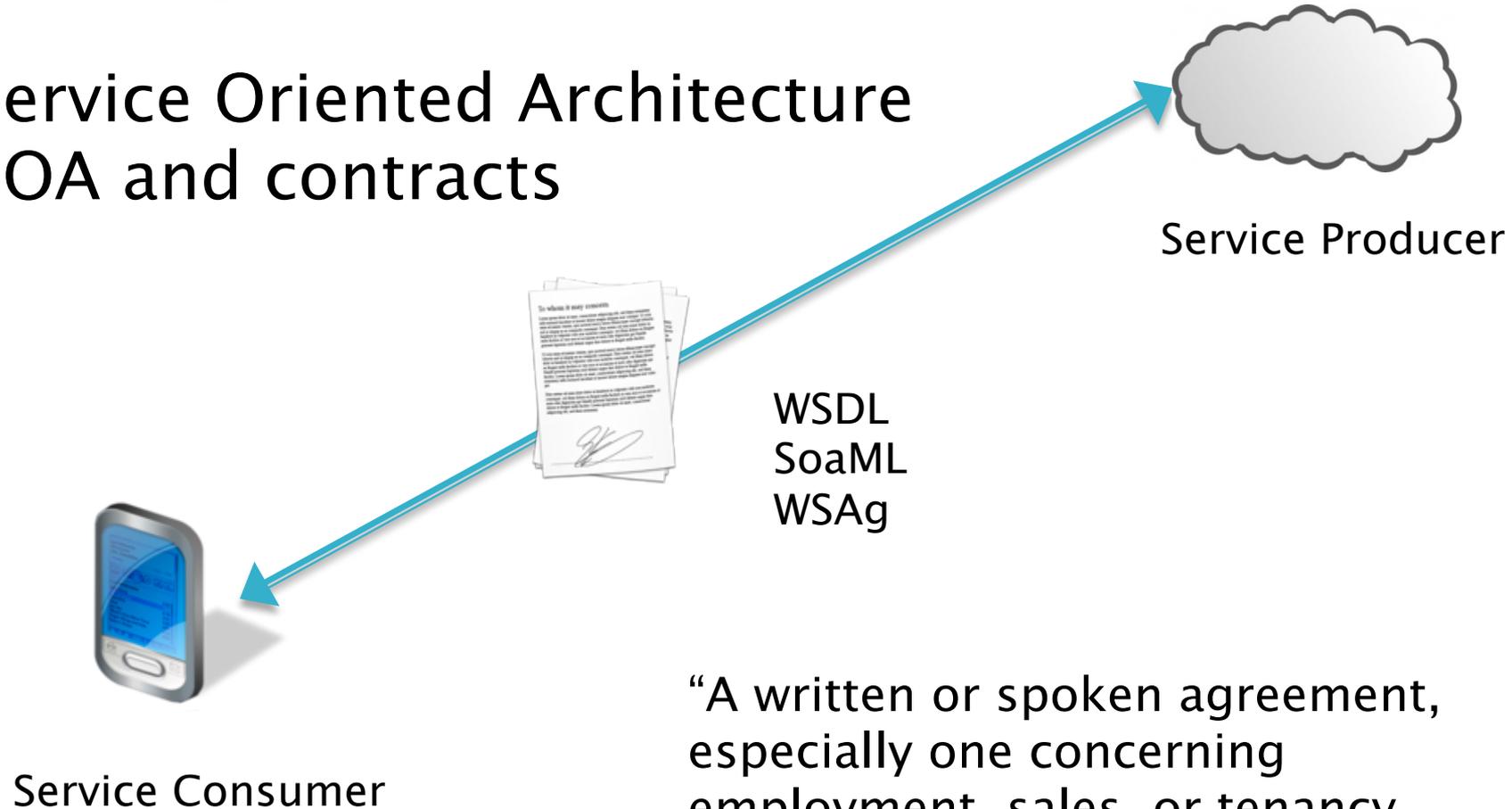
# A Commitment Based Approach for Service Agreement Specification: Modeling Language and Methodology

Mattia Salnitri, Fabiano Dalpiaz, Paolo Giorgini



# Concept of contract

- Service Oriented Architecture
- SOA and contracts



“A written or spoken agreement, especially one concerning employment, sales, or tenancy, that is intended to be enforceable by law” [Oxford Dictionary]

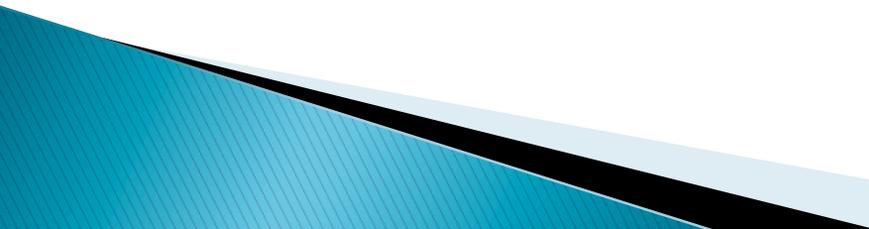
# Modeling SOA

- ▶ Currently modeling languages do not allow for modeling and analyzing contracts
- ▶ Agent Oriented vs Object Oriented Modeling language

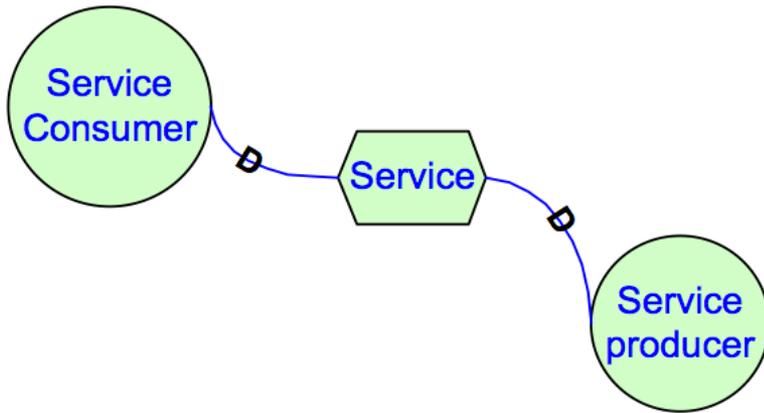


- Agent
- Goal
- Social modeling

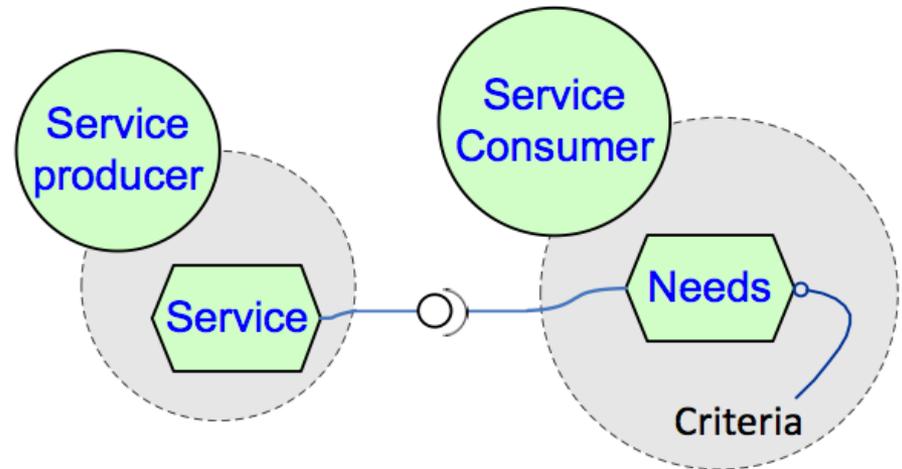
# Objectives of the work

- ▶ Develop an agent-based language able to model contracts
    - Extension of the Tropos modeling language
    - Define a specific analysis process
  - ▶ Develop a method to derive WSAg contracts from goal models
  - ▶ Validate extended modeling language with case studies
- 

# Extension of the modeling language



- It does not model contracts
- Too general
- Incomplete information

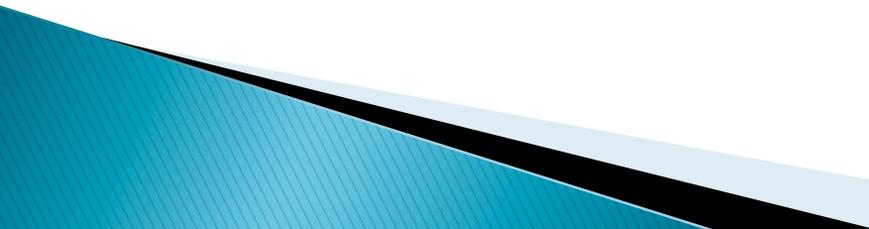


- Focused on contracts
- It models reciprocal dependency
- Specific modeling of contract contents

# The commitment abstraction

*The expression C (debtor, creditor, antecedent, consequent) means that the debtor commits to the creditor in the context that if the antecedent becomes true, the debtor would bring about the consequent.  
[Singh, M.P. et al., Commitment-Based Service-Oriented Architecture]*

C (debtor, creditor, antecedent, consequent)

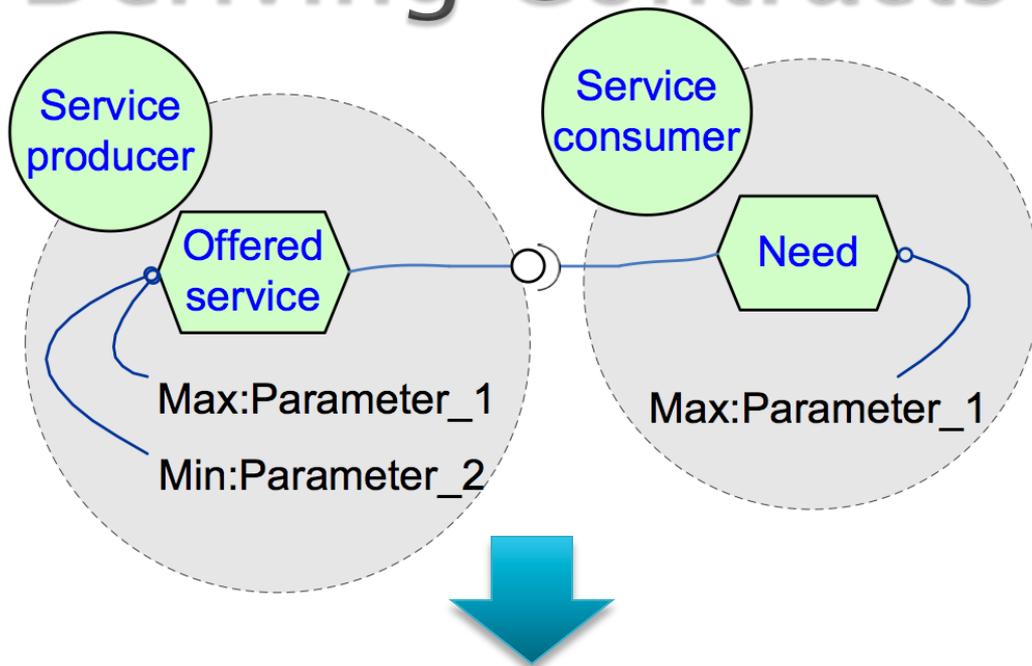


# The commitment abstraction

*The expression  $C$  (debtor, creditor, antecedent, consequent) means that the debtor commits to the creditor in the context that if the antecedent becomes true, the debtor would bring about the consequent.  
[Singh, M.P. et al., Commitment-Based Service-Oriented Architecture]*

- ▶ Every service link is mapped to two commitments
- ▶ Every commitments pair represents an agreement, i.e. a contract

# Deriving Contracts



$C(\text{Cons}, \text{Prod}, \text{Need}[\text{Max:Parameter}_1], \text{Payment})$

$C(\text{Prod}, \text{Cons}, \text{Payment}, \text{OfferedService}[\text{Max:Parameter}_1, \text{Min:Parameter}_2])$

Goal model

Commitments

Contracts

# From commitments to contracts

- ▶ It's not possible to insert all low level details in Tropos modeling language
- ▶ Contract proposal
  - Intermediate step
  - Compliant with WSAg
  - Contains information on service composition and parameter to minimize or maximize



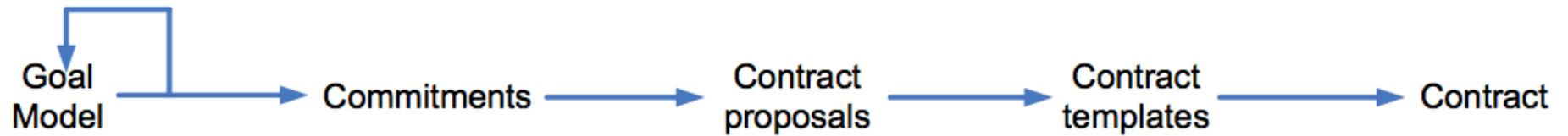
# From commitment to contracts

C(Cons, Prod, Need[Max:Parameter\_1],Payment)

C(Prod, Cons, Payment, OfferedService[Max:Parameter\_1,Min:Parameter\_2])

```
<wsag:Terms>
  <wsag:ServiceDescriptionTerm wsag:Name="Parameters"
wsag:ServiceName="OfferedService">
  <mob:services>
    <mob:service mob:name="OfferedService">
      <mob:Parameter1 mob:type="maximize"/>
      <mob:Parameter2 mob:type="minimize"/>
    </mob:service>
  </mob:services>
</wsag:ServiceDescriptionTerm>
</wsag:Terms>
```

# Methodological guidance



Company base

Company branches

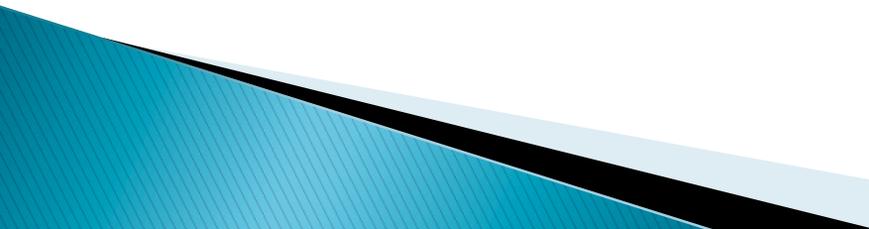
Analysts

Negotiation  
mechanism  
(producer+consumer)

# Validation

- ▶ Case study 1: Telecommunication company
  - Actors
    - Service producer: Telecommunication company
    - Service consumer: Tourist role
  - Objectives
    - Validating service composition and contracts derivation
- ▶ Case study 2: Dropbox
  - Actors
    - Service producer: Dropbox company
    - Service consumer: Businessman role
  - Objectives
    - Validating support of new services

# Summary

- ▶ **New ideas**
    - Concept of contract in Agent-Oriented modeling language
    - Use of commitments as contract abstraction
  - ▶ **Benefits**
    - Specific approach for modeling and analysing contracts
    - Automated contract generation
  - ▶ **Disadvantages**
    - Dimension of the diagram
- 

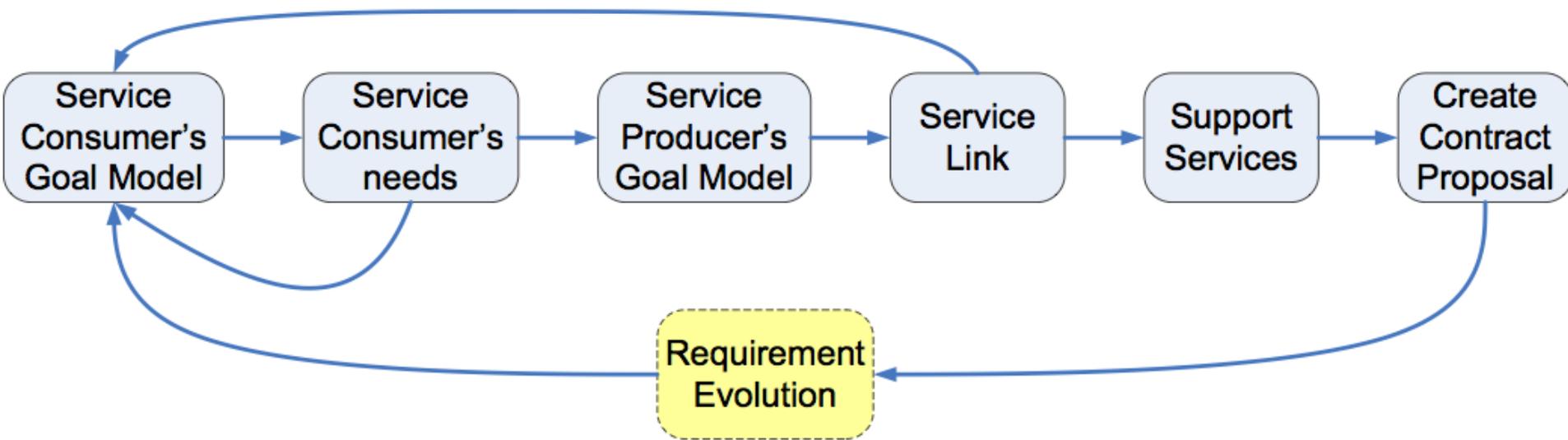
# Future work

- ▶ Implementation of the tool
  - Extension of openOME
- ▶ Specify the how service producer branches will create contracts
  - Cost functions: from contract proposal to contract template
  - Framework WSAg already implemented



Questions?

# Modeling process



# From commitment to contracts

C(Cons, Prod, Need[max:Par1],Payment)

C(Prod, Cons, Payment, OfferedService[max:Par1,min:Par2])

```
<wsag:Terms>
  <wsag:GuaranteeTerm Name="ServProd guarantee"
    Obligated="ServiceComsumer">
    <wsag:BusinessValueList>
      <wsag:Penalty>
        <wsag:AssesmentInterval>
          <wsag:TimeInterval> P5Y2M10DT15H
          </wsag:TimeInterval>
        </wsag:AssesmentInterval>
        <wsag:ValueUnit>EUR</wsag:ValueUnit>
        <wsag:ValueExpr>1</wsag:ValueExpr>
      </wsag:Penalty>
    </wsag:BusinessValueList>
  </wsag:GuaranteeTerm>
</wsag:Terms>
```