

# Reasoning Tasks and Mediation on Choreography and Orchestration in WSMO

Michael Stollberg

WIW 2005, June 6-7, Innsbruck, Austria



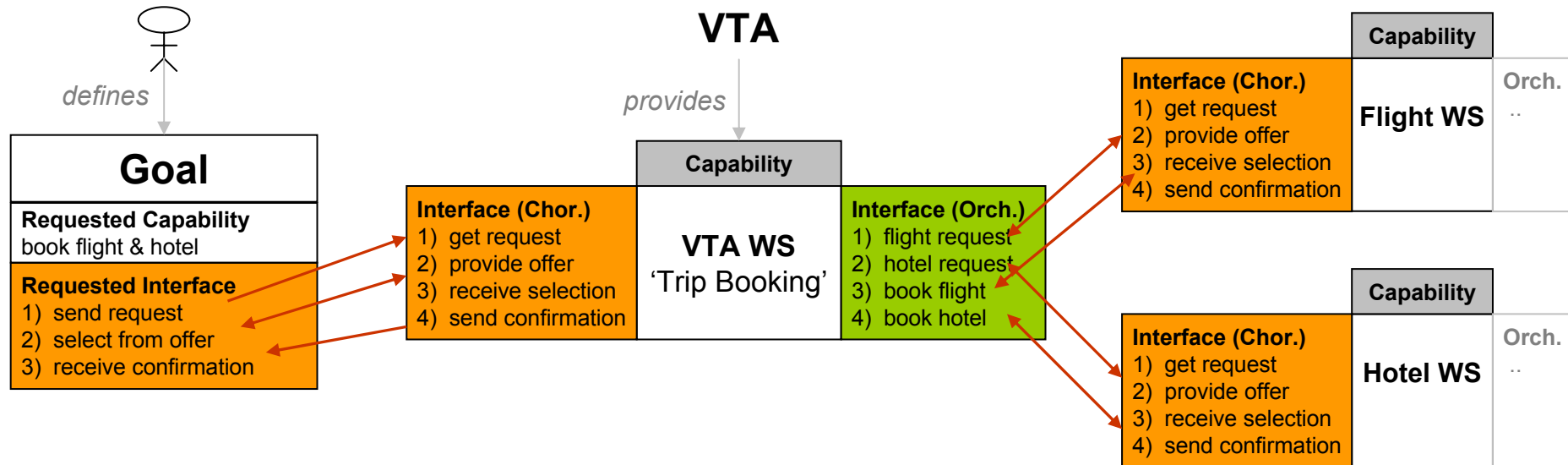
# Content

---

1. Terminology Clarification
2. Choreography: Description, Choreography Discovery as Reasoning Task, and Mediation Requirements
3. Orchestration: Description and Choreography Discovery as Reasoning Task
4. Conclusions



# What we are talking about



**Behavior Interface**  
how entity can interact

**Choreography**  
interaction between entities

**Orchestration**  
service aggregation for realizing functionality

Terminology Definitions from:

Barros A.; Dumas, M.; Oaks, P.: Standards for Web Service Choreography and Orchestration: Status and Perspectives. Accepted for 1st International Workshop on Web Service Choreography and Orchestration for Business Process Management at the BPM 2005, Nancy, France, September 2005)



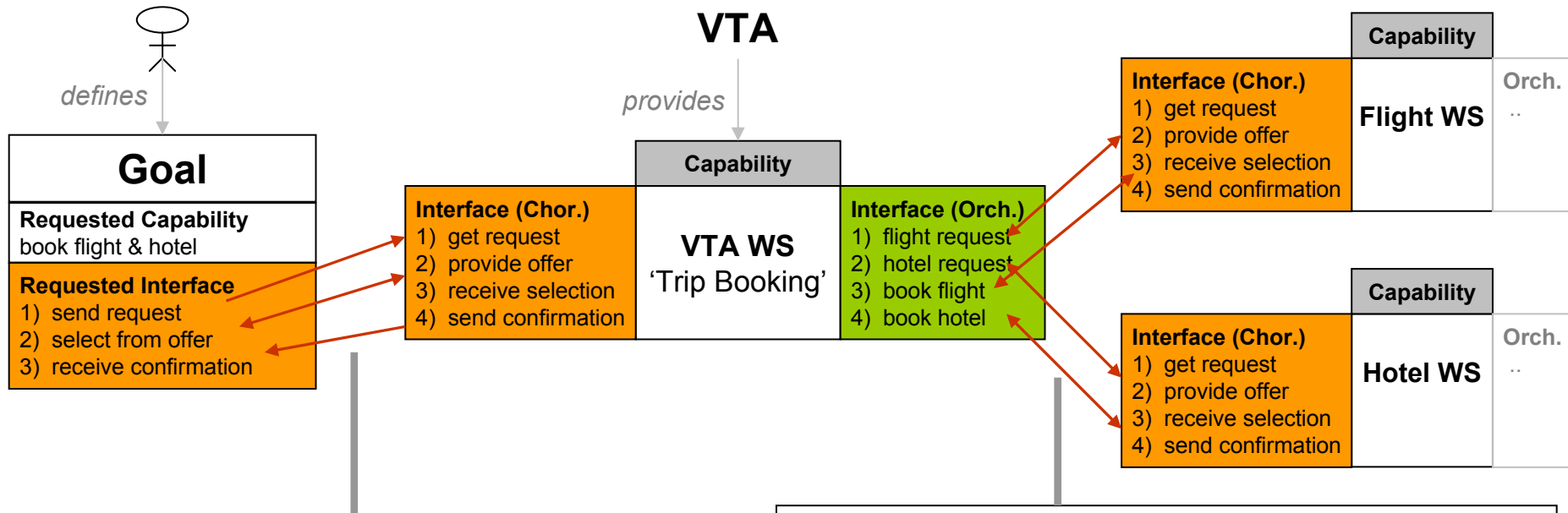
# Main Idea

---

- **“Choreography Discovery”**:
  - given Web Services (Web Services and Goals) with their Behavior Interfaces and Orchestrations as “choreography participants”
  - **does a choreography exists between the choreography participants** (i.e. a ‘valid’ interaction protocol for service usage / realization)?
- What we need for this:
  - formal description model of Behavior Interface & Orchestration
  - “algorithm” for choreography discovery
- Motivation / Context:
  - central reasoning task on Web Service Interfaces
  - allows determining a priori whether the interactions needed to consume / execute Web Services will (or at least can) be successful
  - in conjunction with an ontology-based executable communication technology and “choreography mediation” this can realize the vision of Semantic Web Services / WSMO



# In the example



- both behavior interfaces given ("static")  
 - correct & complete consumption of VTA  
 => existence of a valid choreography?

- VTA Orchestration & Behavior Interfaces of aggregated WS given  
 => existence of a valid choreography between VTA and each aggregated WS?

- **Choreography Discovery** as a central reasoning task in Service Interfaces  
 - 'choreographies' do not have to be described, only existence determination  
 => Questions: choreography discovery algorithm & support from WSMO model



# WSMO Service Interface Description Model

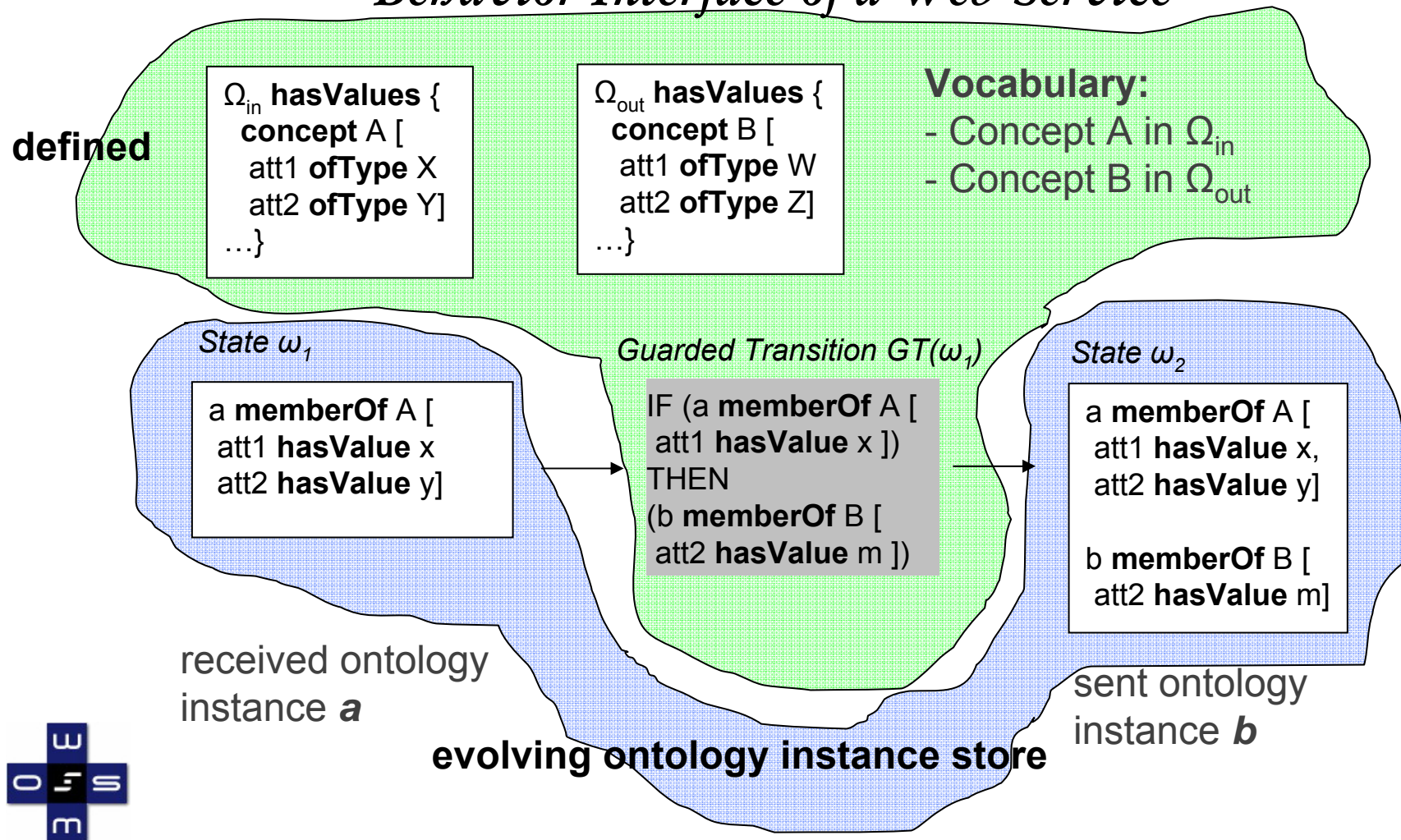
---

- common formal model for Service Interface description
  - ontologies as data model
  - based on ASMs for representing dynamics
  - formal description as basis for reasoning and declarative mediation
  - not restricted to any executable communication technology
- general structure:
  - Vocabulary  $\Omega$ :
    - ontology schema(s) used in service interface description
    - usage for information interchange: in, out, shared, controlled
  - States  $\omega(\Omega)$ :
    - a stable status in the information space
    - defined by attribute values of ontology instances
  - Guarded Transition  $GT(\omega)$ :
    - state transition
    - general structure: *if* (condition) *then* (action)
    - different for Choreography and Orchestration



# Service Interface Example

## *Behavior Interface of a Web Service*



# What do we learn from this?

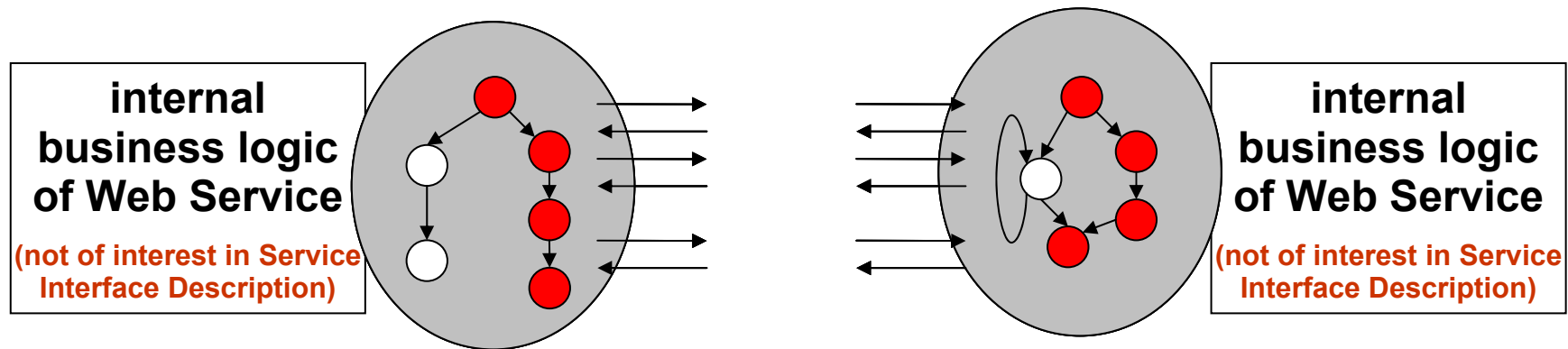
---

- a formal model as basis for reasoning and mediation on Service Interface definitions
- Descriptions from different perspectives:
  - **Behavior Interface:** subjective perspective
  - **Choreography:** global perspective of interaction
  - **Orchestration:** subjective perspective of ‘Orchestrator’
- WSMO model is very basic:
  - implications on communication technology
  - implicit assumptions on Choreography and Orchestration
  - very rudimentary support for ‘process definitions’
  - strong ontology support





# Choreography Discovery



- a valid choreography exists if:
  - 1) **Information Compatibility**
    - compatible vocabulary
    - homogeneous ontologies
  - 2) **Communication Compatibility**
    - start state for interaction
    - a termination state can be reached without any additional input



# Information Compatibility

---

**If choreography participants have compatible vocabulary definitions:**

- $\Omega_{in}(S1)$  and  $\Omega_{shared}(S1) = \Omega_{out}(S2)$  and  $\Omega_{shared}(S2)$
- determinable by Intersection Match from Discovery
- $SI_{S1}, SI_{S2}, O, M \models \exists x. (\Omega_{S1(in \cup shared)}(x) \wedge \Omega_{S2(out \cup shared)}(x))$
- more complex for multi-party choreographies

**Prerequisite: choreography participants use homogeneous ontologies:**

- semanticInteroperability(S1, S2, ..., Sn)
- same ontologies in Service Interfaces, or usage of respective OO Mediators (technique not yet addressed in WSMO)



# Communication Compatibility

---

- Definitions (for “binary choreography” (only 2 services), more complex for multi-party choreographies)

## Valid Choreography State:

$\omega_x(C(S1, S2))$  if informationCompatibility ( $\omega_x(S1), \omega_x(S2)$ )

- means: action in GT of S1 for reaching state  $\omega_x(S1)$  satisfies condition in GT of S2 for reaching state  $\omega_x(S2)$ , or vice versa

## Start State:

$\omega_\emptyset(C(S1, S2))$  if  $\Omega S1(\omega_\emptyset) = \emptyset$  and  $\Omega S2(\omega_\emptyset) = \emptyset$  and  $\exists \omega_1(C(S1, S2))$

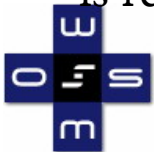
- means: if initial states for choreography participants given (empty ontology, i.e. no information interchange has happened), and there is a valid choreography state for commencing the interaction

## Termination State:

$\omega_T(C(S1, S2))$  if  $\omega_T(S1) = \text{noGT}$  and  $\Omega S2(\omega_T) = \text{noGT}$  and  $\exists \omega_T(C(S1, S2))$

- means: there exist termination states for choreography participants (no action for transition to next state), and this is reachable by a sequence of valid choreography states

- Communication Compatibility given if there exists a start state and a termination state is reachable without additional input by a sequence of valid choreography states



# Communication Compatibility Example

## A Buyer (e.g. Goal)

$\Omega_{S_1}(\omega\emptyset) = \{\emptyset\}$

**if**  $\emptyset$  **then** request

$\Omega_{S_1}(\omega 1) = \{\text{request(out)}\}$

**if** cnd1(offer) **then** changeReq

$\Omega_{S_1}(\omega 2a) =$   
 $\{\text{offer(in), changeReq(out)}\}$

**if** cnd2(offer) **then** order

$\Omega_{S_1}(\omega 2b) =$   
 $\{\text{offer(in), order(out)}\}$

**if** conf **then**  $\emptyset$

$\Omega_{S_1}(\omega 3) = \{\text{offer(in), conf(in)}\}$

## A Seller (e.g. Service)

$\Omega_{S_2}(\omega\emptyset) = \{\emptyset\}$

**if** request **then** offer

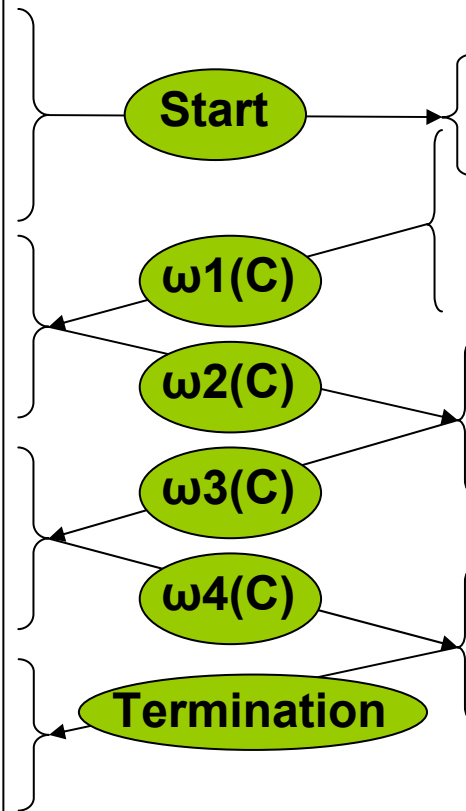
$\Omega_{S_2}(\omega 1) =$   
 $\{\text{request(in), offer(out)}\}$

**if** changeReq **then** offer

$\Omega_{S_2}(\omega 2a) =$   
 $\{\text{changeReq(in), offer(out)}\}$

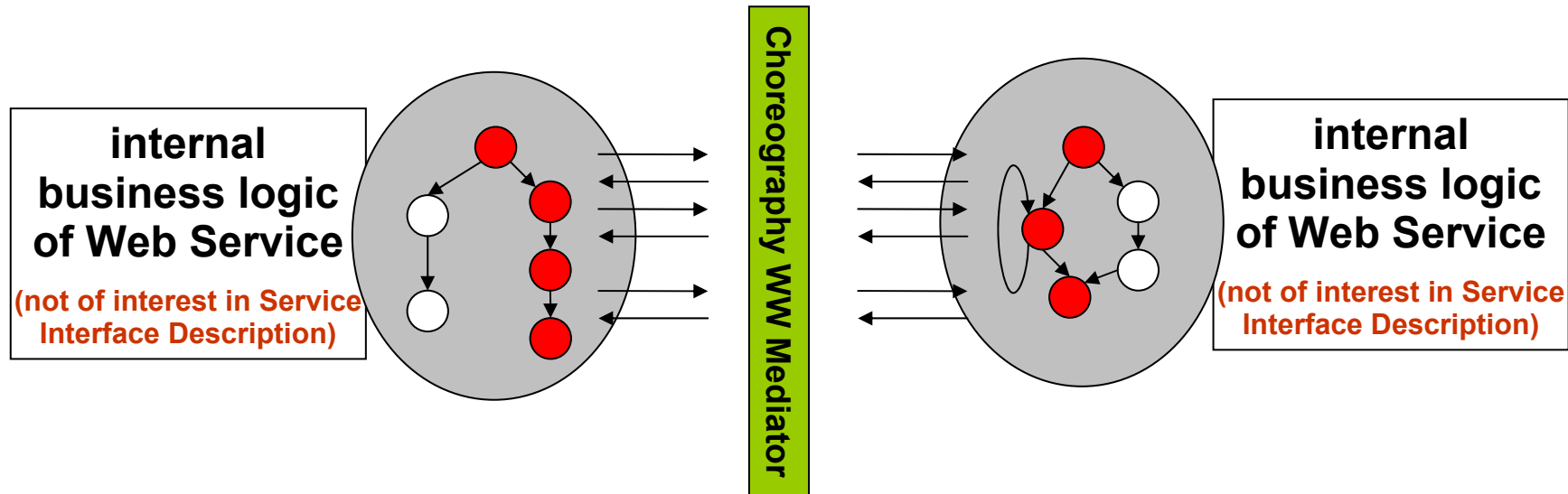
**if** order **then** conf

$\Omega_{S_2}(\omega 2b) =$   
 $\{\text{order(in), conf(out)}\}$



existence of a valid Choreography

# WW Mediators in Choreography

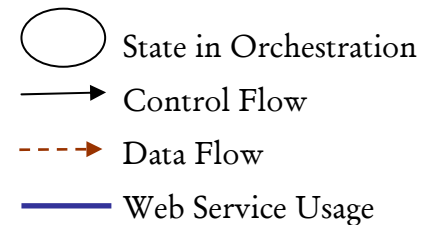
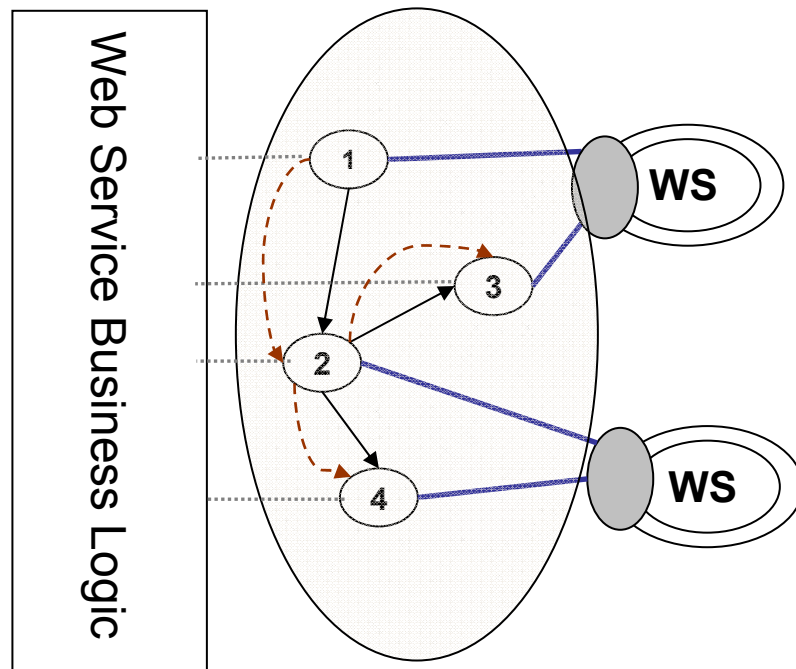


- if a choreography does not exist, then find to define an appropriate WW Mediator that
  - resolves possible mismatches to establish Information Compatibility (OO Mediator usage)
  - resolves process / protocol level mismatches to establish Communication Compatibility (corresponding to the approach for process mediation in WSMX)



# Orchestration

*control structure for aggregation of other Web Services*



- formally described service functionality decomposition
- only those aspects of WS realization wherefore other WS are aggregated
- aggregated WS used via their behavior interface



# Orchestration Description & Validation

---

- Orchestration Description:
  - interaction behavior of “Orchestrator” with “orchestrated Web Services”
  - WSMO Service Interface description model, extension of Guarded Transitions
- Proposal therefore:
  - Guarded Transitions general structure:  
if condition then operation  
operation = (Orchestrator, Web Service, Action)
  - Orchestrator serves as client for aggregated Web Services
- Orchestration Validation:
  - need to ensure that aspired interactions with aggregated Web Service can be executed successfully
  - = > Choreography Discovery for all interaction of Orchestrator with each aggregated Web Service



# Orchestration Validation Example

## VTA Web Service Orchestration

```

if  $\emptyset$  then (FWS, flightRequest)

if flightOffer
then (HWS, hotelRequest)

if selection
then (FWS, flightBookingOrder)

if selection, flightBookingConf
then (HWS, hotelBookingOrder)
    
```

## Flight WS Behavior Interface

```

if request then offer
if order then confirmation
    
```

## Hotel WS Behavior Interface

```

if request then offer
if order then confirmation
    
```

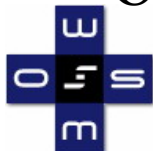
**Start**  
(VTA, FWS)

**Termination**  
(VTA, FWS)

**Start**  
(VTA, HWS)

**Termination**  
(VTA, HWS)

Orchestration is valid if valid choreography exists for interactions between Orchestrator and each aggregated Web Service, done by choreography discovery





# Conclusions

---

- Choreography Discovery as central reasoning task for WSMO Service Interfaces
  - a priori determination if interactions necessary to consume and execute Web Services
  - on basis of / supported by WSMO formal model for Service Interfaces
  - ontology-based reasoning as basic operation
  - clarification (proposal) for terminology and conceptual aspects
- Very early stage of work
  - No use case, no testing, no implementation
  - More complex scenarios not considered
  - Formal basis very rudimentary
- Open Issues:
  - language for WSMO Service Interface Description (under construction)
  - examination of related work & enhancements
  - “Choreography Discoverer” Implementation



---

< / dreaming the full potential of  
WSMO >

< Q&A >



# Behavior Interface Aspects

---

## *Interface for Consuming a Web Service*

- **External Visible Behavior**
  - those aspects of the workflow of a Web Service where interaction with the client is required for consuming
  - this is static (needs to be consumed *correctly* and *completely*)
- **Communication Structure**
  - messages / communicative acts sent and received
  - their order / process (communicative behavior for service consumption)
  - choreography related errors (e.g. input wrong, message timeout, etc.)
- **Grounding**
  - executable communication technology support
- **Formal Model**
  - reasoning on usability of Web Service / interoperability with clients



# Choreography Aspects

---

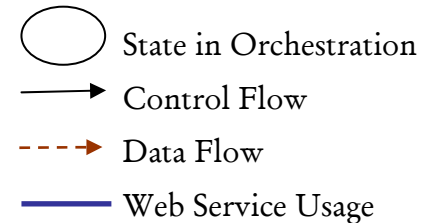
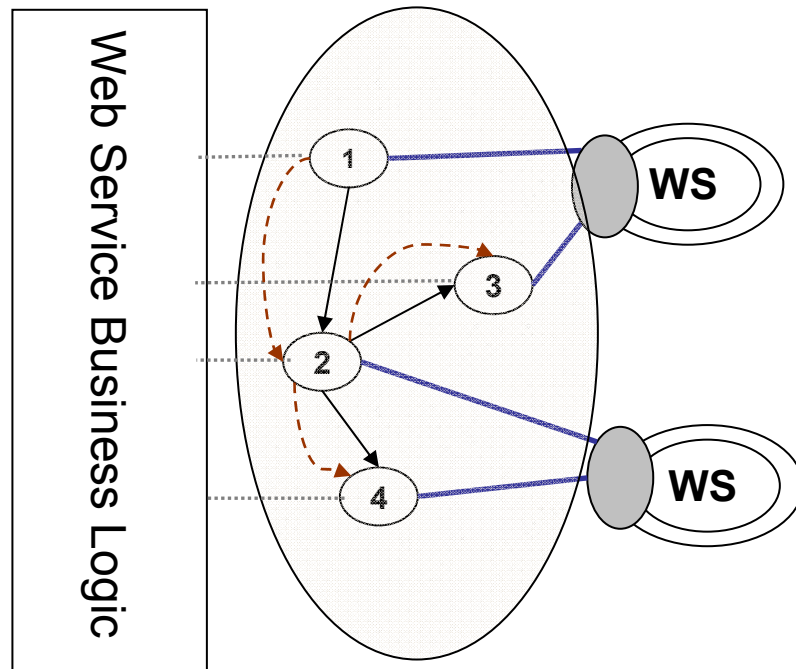
## *Global Interaction between two or more Web Services*

- **Interaction Participants**
  - the actual Web Services / Clients that are ought to interact
  - participation via their Behavior Interface
- **Communication Structure**
  - messages / communicative acts interchanged between partners
  - interaction process (when which interaction between whom)
- **Grounding**
  - not needed, as this is only a control facility
  - communication / interaction execution by participants
- **Formal Model**
  - not needed, as we only have to determine existence of a validity choreography
  - WS-CDL is a language for describing this (but not for Semantic Web Services)



# Orchestration Aspects

## *Control Structure for aggregation of other Web Services*



- formally described service functionality decomposition
- only those aspects of WS realization wherefore other WS are aggregated
- aggregated WS used via their behavior interface

