ebXML Business Process Specification Schema

Monica J. Martin
Sun Microsystems

WS-Choreography F2F
Cambridge, United Kingdom
17 December 2003
ebXML BPSS Summary Outline

• Business collaboration
  • Role in eBusiness and Business Process Management
  • Role in ebXML architecture
  • Metamodel

• Business semantics
  • Binary and multi-party collaboration
  • Business transaction protocol
  • Business message and signal exchange
  • Logical business documents

• Message controls and choreography
  • Rules and conditions
  • State synchronization

• Relevance to WS-Choreography and its output
Business Integration Today

- Collaboration: Business semantics
  - Two or more peers
  - Business-oriented goals
- Choreography: Interface and active monitor
  - Message sequencing “interface”
    - Service operations and externally observable behavior
  - Message exchange patterns with timeouts
  - Can be client/server
    - Constrains client using the Service
- Orchestration: Domain actors
  - Run business processes under central control or from one partner's view
    - Stateful
    - Invokes and offers Services
    - Conforms to Choreography for Services used
BPSS Basics

- Business collaboration
  - Comprised of business transaction patterns and business semantics:
    - Binary and multi-party collaboration
    - Business messages and signals
    - Business transaction protocol
      - Business transactions
      - Business transaction activities
      - Business signals
      - Message controls and choreography
      - Business state synchronization
      - Conditions
      - Transitions

![Diagram of business signals and choreography]
BPSS Role in Business Process Management

[1] Functionality does not currently exist in emerging WS-BPEL.
BPSS in the ebXML Architecture
Metamodel for BPSS

- Borne from UML profile for business process (UMM)
  - Process model defined by UML activity diagrams
  - Computable schema
- Geared towards eBusiness
  - Business process description as part of loosely coupled, highly aligned ebXML architecture
- Used with CPP/A
Business Semantics

- Binary and multi-party collaboration: A special type of business process or activity conducted between two or more parties to achieve a commonly defined goal or outcome. Can be decomposed into one or more business transactions.
    - Follows specific business message exchange patterns based on business transaction purpose (notification, commitment, etc).
    - Represents choreography definition state (i.e. the choreography is in the state of executing this business transaction). The business activity is an abstract kind of business state.
Business Semantics [2]

- Business transaction activities: Roles of the binary collaboration are assigned to the execution of the business transaction

  - The abstract roles (of the binary collaboration and business transaction) become explicit when the transaction is used within a business transaction activity within a binary collaboration
Business Semantics [3]

- Business signals: Optional messages that guarantee “state alignment” between parties. Layered atop reliable messaging
  - Receipt Ack: Message received and passed structural validity checks by receiving party
  - Acceptance Ack: System passed business rules of receiving party and is now in a system of record (rules and the system are not exposed in BPSS)
- Exceptions
  - Technical
  - Business
  - Timeouts
- Business Service Interface
  - Runtime software that can isolate the internal communications of a given legacy or other application from the collaboration model, and once built represent the party in a collaboration model
Message Controls and Choreography

- Message controls / choreography: Business collaboration choreography describes ordering and transitions between business transactions or sub collaborations within a binary collaboration

  - Choreography equates to and transitions between business states. Transitions happen between business activities. Requests can occur within the timeToPerform of the binary collaboration

    - A timeToPerform is the period of time, starting upon initiation of the first activity, within which this entire collaboration must conclude from the requester’s perspective

  - Control flow based on state/transition model

- Rules

  - Production rules: Maps UML model to schema. Currently represented syntactically as strings

  - Condition guards and expressions on state transitions: Account for success, failure or timeout

    - Transitions: Can be used to create nested activities

    - Guard: Status of business transaction activity

    - Condition expression: Conditions true in logical document
BPSS into the Future

- Service-oriented architecture
  - BSI Interface binding for business messages and signals
- Updates/enhancements to ebXML BPSS 1.01, later versions
  - Tightening business semantics for controls
  - Enhancements to production rules, state management, timing
- Reuse or reference to other technologies: Understanding terminology and scope are important
  - Other choreography definitions or transaction protocols
    - WS-Choreography output
    - WS-CAF output (coordination)
    - Other higher level business constructs
Potential Mapping BPSS and WS-Chor Output

Trading partners
Business intent
Commercial transaction patterns
Business signals and messages
Business semantics

Order
Sales
Status
Company A
Company B

Order
Sales
Status
Company A
Company B
...n

Enterprise Choreography A
Enterprise Choreography B

Underlying Msg Exchange [1]

Control or data flow
Sequences of interactions
Channels associate roles, participants

PO Request
Receipt Ack
Accept Ack
PO Response
Receipt Ack
Accept Ack

Message exchange between service endpoints
Single local process instance

[1] Between logical business message and application transport
Summary

- BPSS is focused on business semantics for business collaboration
- WS-Choreography is focused on the externally observable behavior of composed services
- Maximize benefit to both
  - Complementary aspects of emerging standards in BPM
  - Leverage existing capabilities
  - Investigate in more detail expected future capabilities, for example:
    - Business rules
    - Choreography
    - Multi-party collaboration