A Standard for Web Services Transactions

Dr Mark Little
Engineering Director, Red Hat

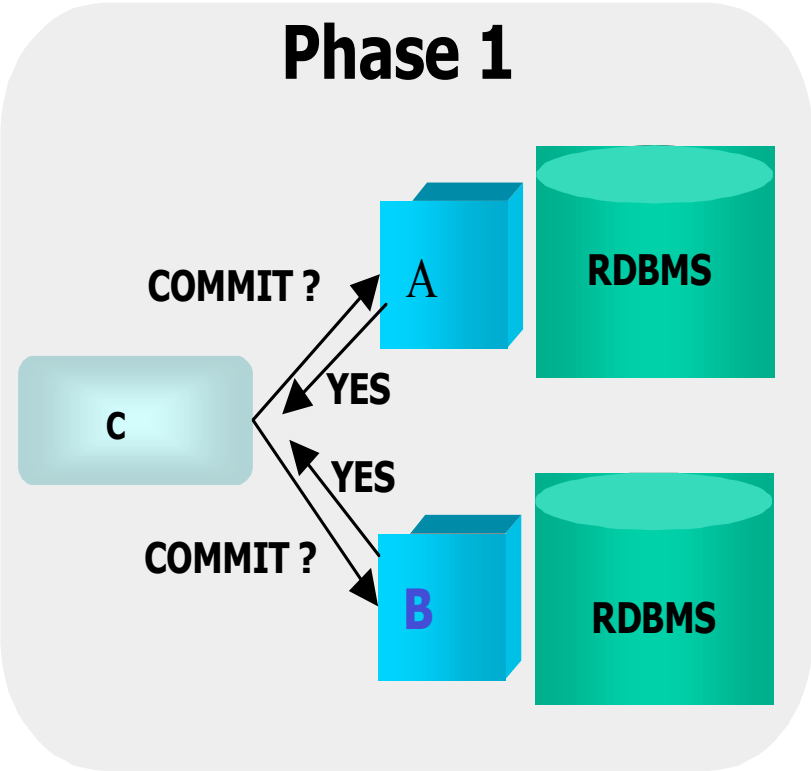
Overview

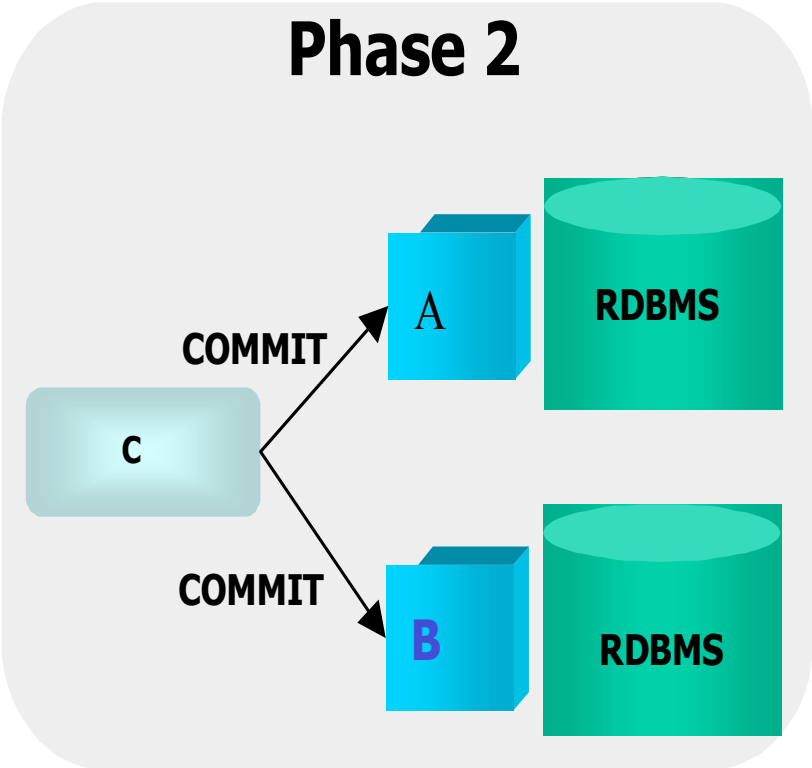
- **Transactions and why they are important**
- **Web services and the problems they present**
- **WS-Coordination, WS-AT and WS-BA**
- **Example scenario**
  - WS-AT and WS-BA

Atomic transactions

- **Scoping mechanism that provides "all-or-nothing" semantics**
- **Enables shared resources to be protected from concurrent users**
- **ACID properties**
  - Atomic
  - Consistent
  - Isolated
  - Durable

Two-phase commit





Environmental impact

- **ACID transactions implicitly assume**
  - Closely coupled environment
    - All entities involved in a transaction span a LAN, for example.
  - Short-duration activities
    - Must be able to cope with resources being locked for periods
- **Therefore, do not work well in**
  - Loosely coupled environments!
  - Long duration activities!

However …

- Web Services are as much about interoperability as they are about the Web
- In the short term Web Services transactions will be about interoperability between existing TP systems rather than running transactions over the Web

B2B interactions

- **Business-to-business interactions may be complex**
  - involving many parties
  - spanning many different organisations
  - potentially lasting for hours or days
- **Cannot afford to lock resources on behalf of an individual indefinitely**
- **May need to undo only a subset of work**

Relaxing isolation

- **Internal isolation or resources should be a decision for the service provider**
  - E.g., commit early and define compensation activities
  - However, it does impact applications
    - Some users may need to know a priori what isolation policies are being used
- **Undo can be whatever is required**
  - Before and after image
  - Entirely new business processes

Relaxing atomicity

- **Sometimes it may be desirable to cancel some work without affecting the remainder**
  - E.g., prefer to get airline seat now even without travel insurance
- **Similar to nested transactions**
  - Work performed within scope of a nested transaction is provisional
  - Failure does not affect enclosing transaction
- **However, nested transactions may be too restrictive**
  - Relaxing isolation

WS-AT/WS-BA

- **Specifications released by Arjuna, BEA, IBM, IONA and Microsoft**
  - Now OASIS standard
- **Separate coordination from transactions**
  - WS-Coordination
- **Define two transaction models**
  - AtomicTransaction
    - Closely coupled, interoperability
  - Business Activities
    - Compensation based, for long duration activities

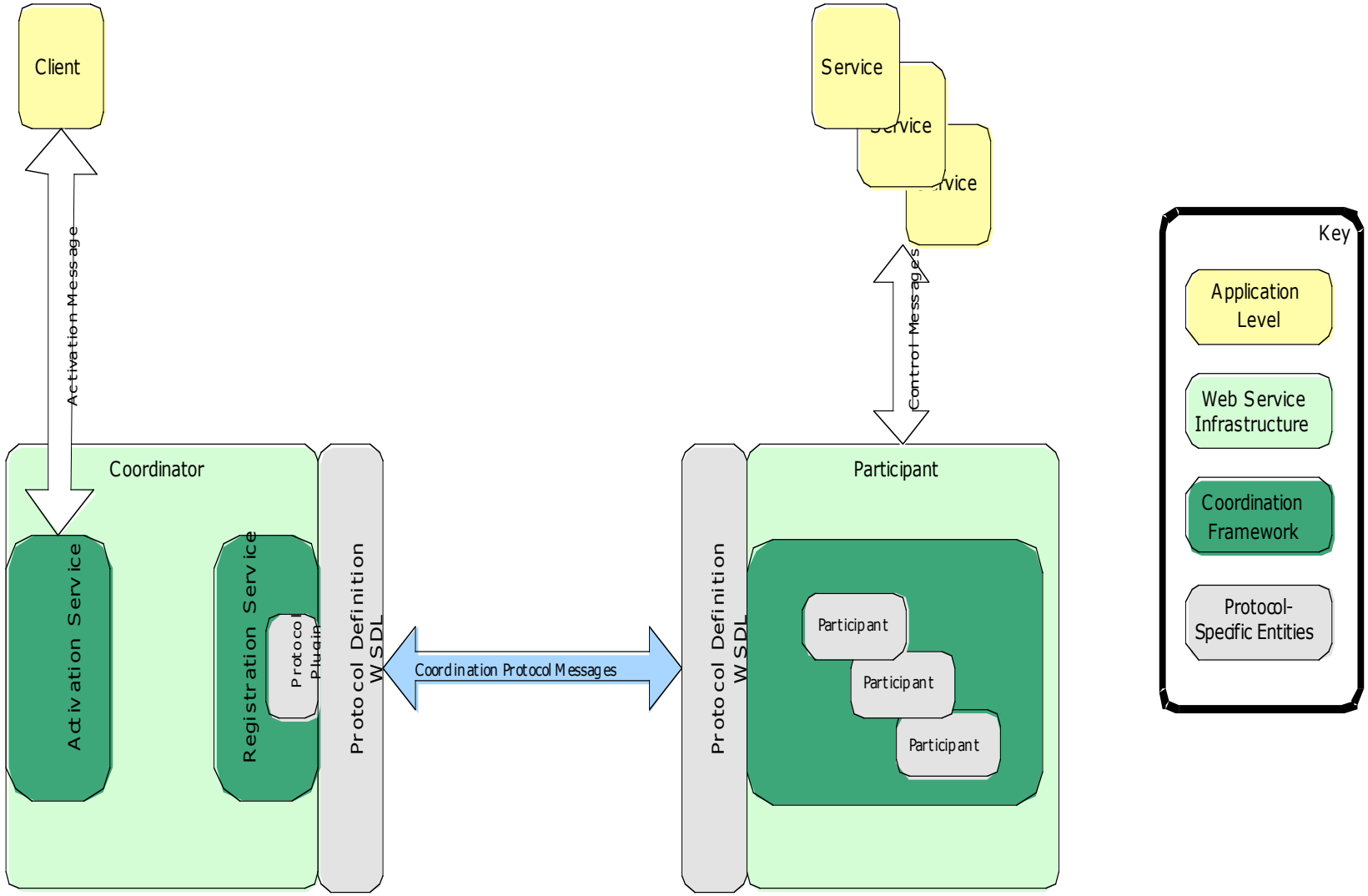
Web Services Coordination

- **Coordination is more fundamental than transactions**
  - Transactions, security, workflow, replication, …
  - But each use may require different protocol
    - Two-phase, three-phase, QoS specific, …
- **Define separate coordination service**
  - Allow customisation for different protocols

WS-Coordination

- **Defines typical roles of coordinator and participant**
  - Coordinator split into two roles
    - Activation service
      - Context
    - Registration service
  - Participant interface is implied by specific protocol

Coordination service



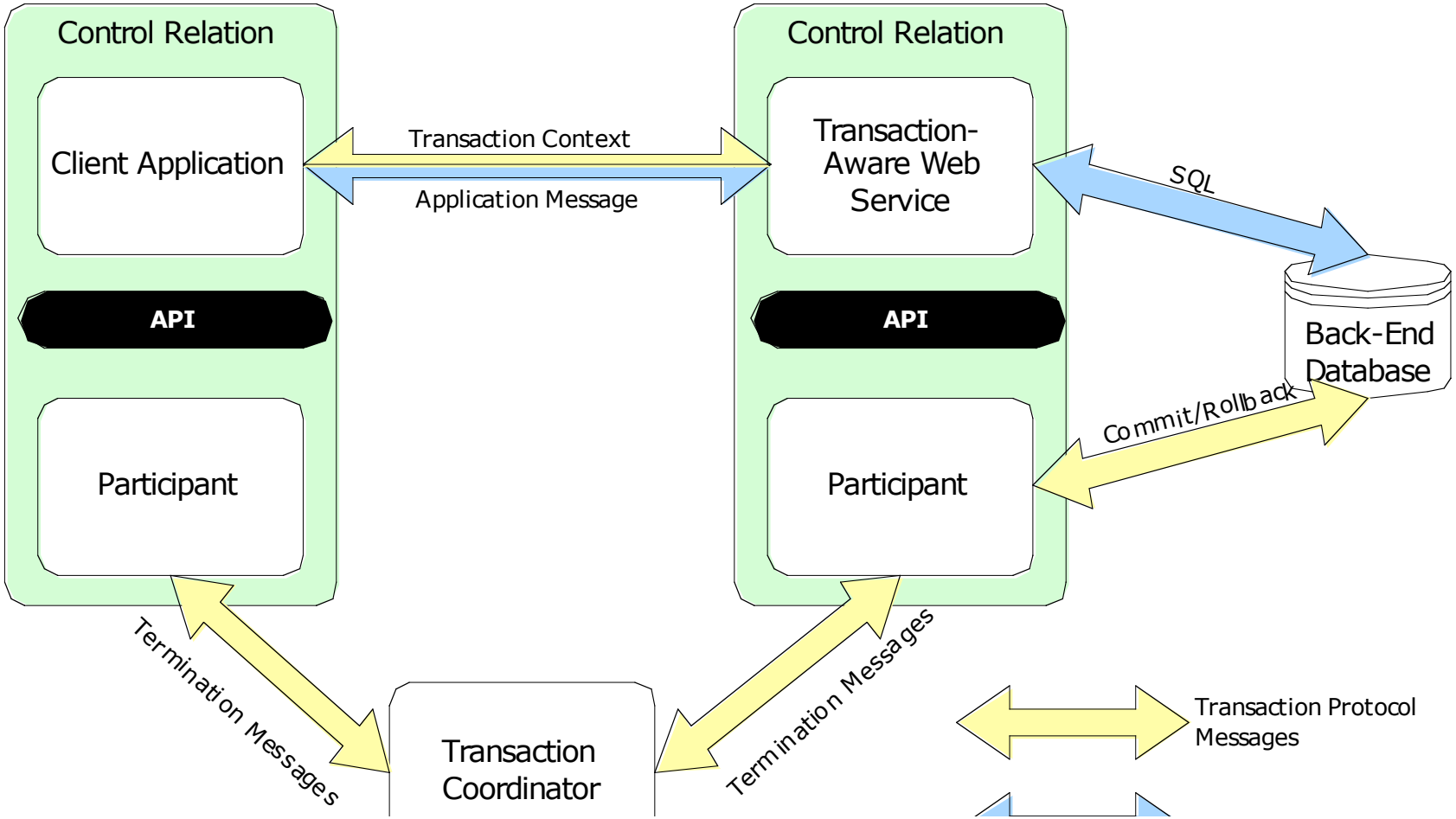
WS-TX protocols

- **Coordinator protocols**
  - Atomic Transaction
    - Completion, DurableTwoPhase, VolatileTwoPhase
  - Business Activity
    - BusinessAgreementWithCoordinatorCompletion, BusinessAgreementWithParticipantCompletion

WS-AtomicTransaction

- **Assume ACID transactions**
  - High degree of trust
  - Isolation for duration of transaction
  - Backward compensation techniques
  - Does not allow heuristic outcomes
- **Integration with existing transaction systems**
  - Important to leverage investments
- **Interoperability between transaction systems**
  - Something of a holy grail to date

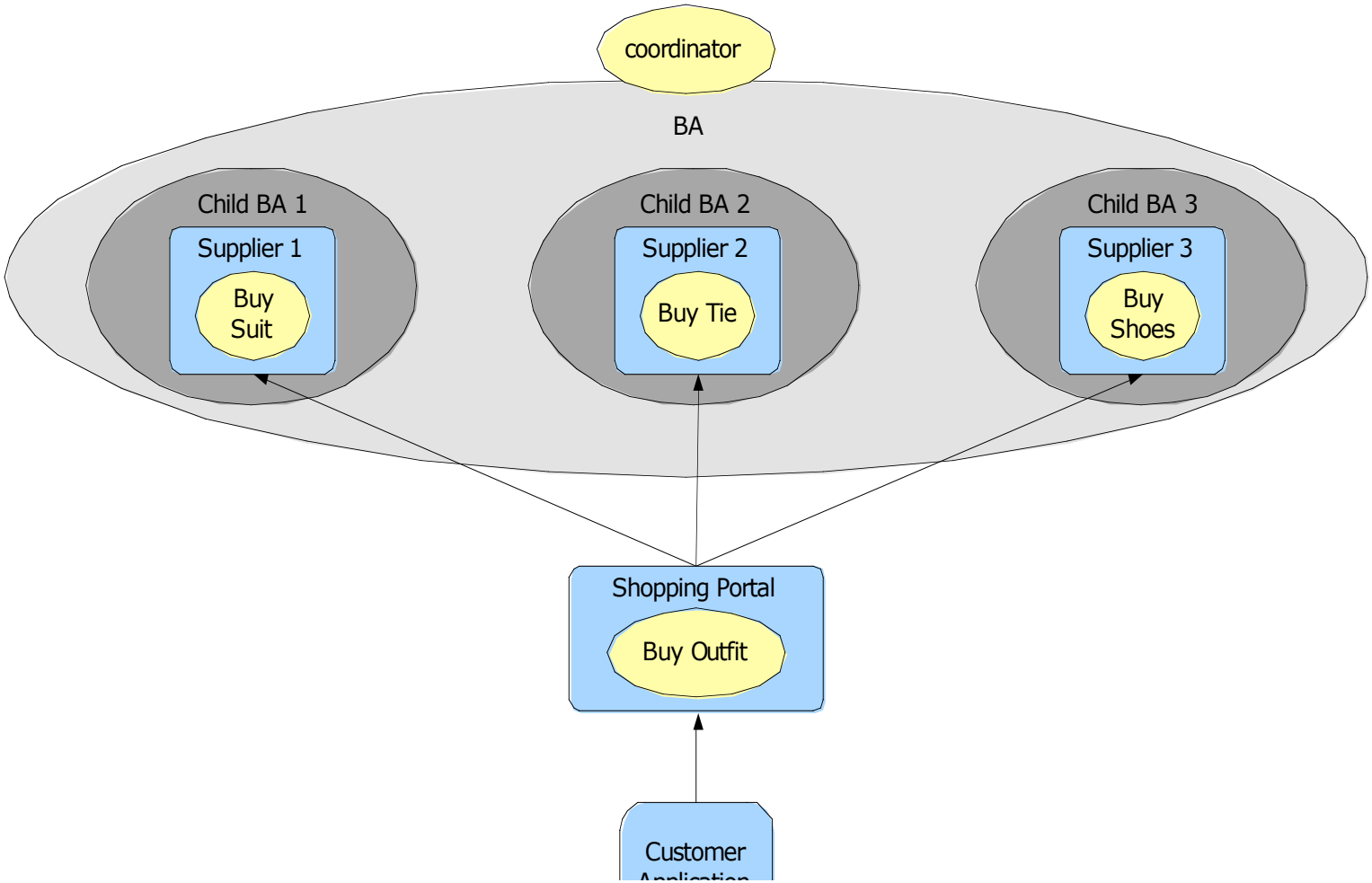
Services, participants and context



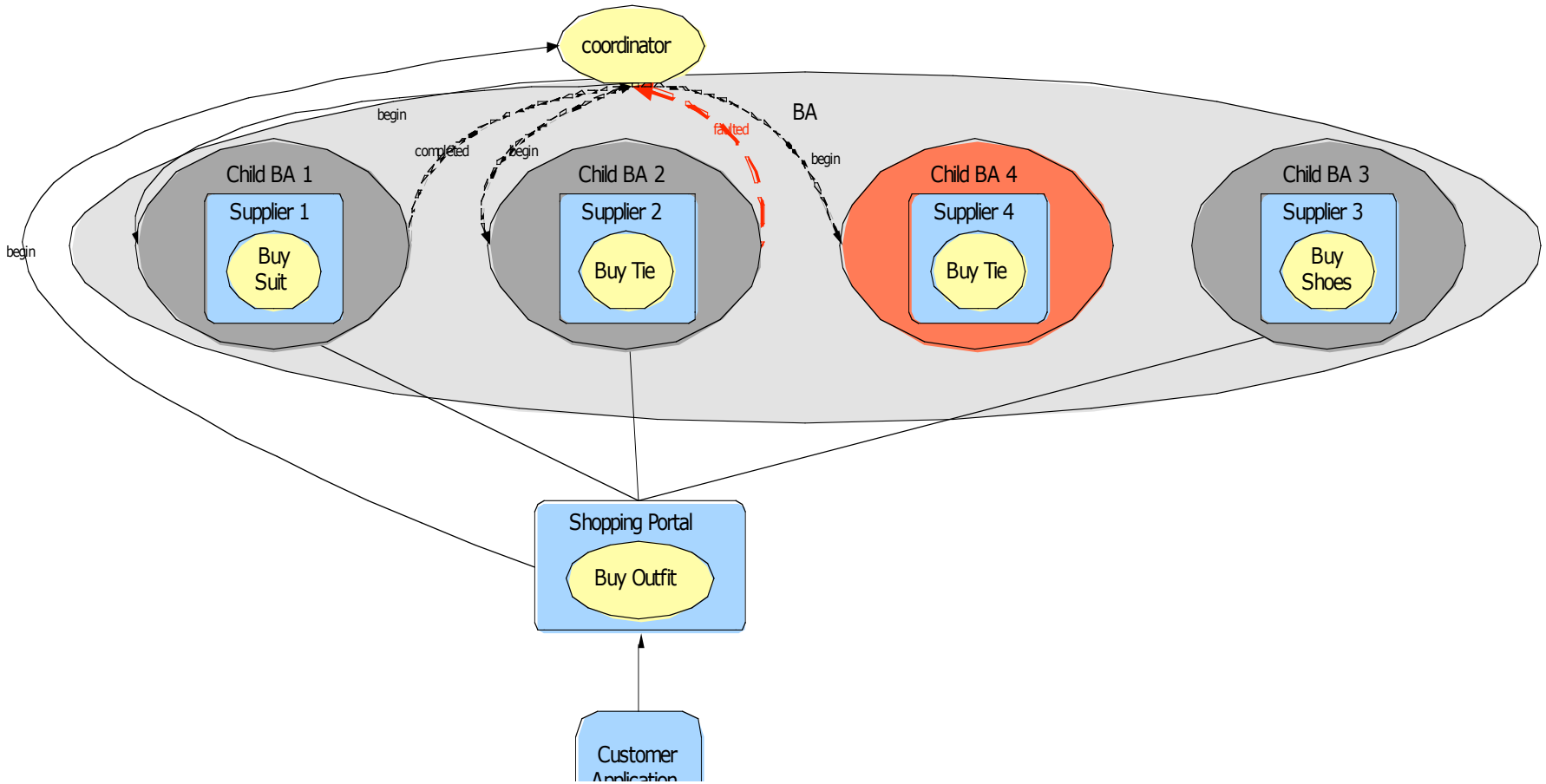
WS-BusinessActivity

- **Workflow-like coordination and management**
- **Business activity can be partitioned into tasks**
  - Parent and child relationships
    - Select subset of children to complete
    - Parent can deal with child failures without compromising forward progress
- **Tasks can dynamically exit a business activity**
- **Tasks can indicate outcome earlier than termination**
  - Up-calls rather than just down-calls

WS-BA example



Compensating BA



Conclusions

- **Transactions are important in mission critical environments**
  - Definition of “transaction” needs to be re-thought for Web Services
  - But ACID transactions still important
- **OASIS WS-TX defines the standard for Web Services Transactions**
  - Multiple protocols for different use cases
  - Demonstrated interoperability between
    - Red Hat/Jboss
    - IBM
    - Microsoft
    - IONA Technologies