

Web Services Transaction

A Comparison of Frameworks

A presentation by Team Two

What is a Transactions?

To the general public, the term “transaction” is used to mean a business exchange where money is traded for goods.

But in the domain of computing, the meaning of “transaction” is more specific...

Transaction

Transactions are a fundamental abstraction in dependable computing systems. Its inherent value is to allow programmers to make something that went wrong appear as if it never happened.

A transaction implies that a group of activities is completed as a unit, so they all succeed or all fail together. In other words, it's a semantic that means "all or nothing".

ACID

■ Atomic

- The transaction will succeed completely or fail completely. This is particularly important when executing business logic that involves the updating of multiple underlying data sources, where the atomicity property turns a set of operations into a single indivisible logical operation.

■ Consistent

- The data store changes over time through a sequence of consistent states. That is, any data that has been updated during the lifetime of the transaction is left in a consistent state at the end of that transaction, irrespective of any failures that have occurred during the transaction.

ACID

- **Isolated**

- The effects of a transaction should be invisible to and isolated from other transaction until it has been committed. This also means that any running transaction believes that it has exclusive access to the resources associated with it.

- **Durable**

- The results are guaranteed to be stored after the transaction completes. That is, once a transaction has been completed, the resulting changes must not be lost even if hardware or software fails.

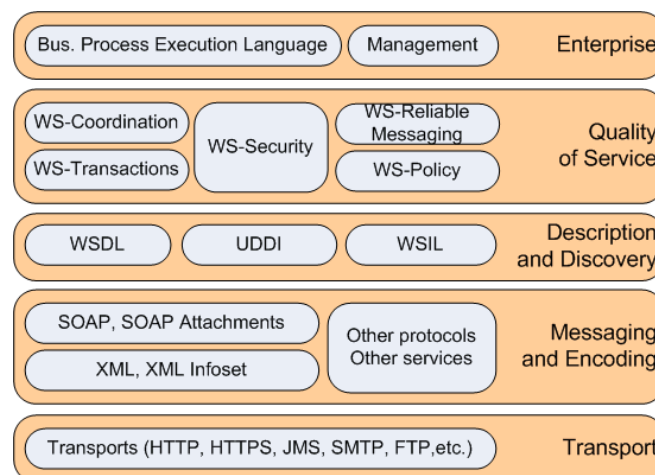
Compensating Transaction

- Compensation is specific to the way business data is managed, so it's always part of business logic.
- This is very different from the automatic rollback provided by databases for ACID transactions.
- Compensation avoids another problem. Locking of your company's data by anyone on the Internet allows denial-of-service attacks.
- Using compensation means that your data isn't locked for a long time, but we can no longer have ACID transactions - at least the Isolation guarantees must be relaxed - because the data is visible between the initial change and the compensation.

Available Frameworks

- There are currently a few standards working towards some form of reliable transaction management using Web Services. The better known ones are:
- The OASIS Business Transaction Protocol, push by Arjuna Technologies Ltd. (formally part of Hewlett-Packard Company), Fujitsu Limited, IONA Technologies Ltd., Oracle Corporation, and Sun Microsystems, Inc.
- WS-Coordination/WS-Transaction push by IBM, Microsoft and BEA

Web Services Standards



OASIS

- **Organization for the Advancement of Structured Information Standards**
- is a not-for-profit, international consortium that drives the development, convergence and adoption of e-business standards.
- produces worldwide standards for security, Web services, conformance, business transactions, supply chain, public sector, and interoperability within and between marketplaces.
- has more than 3,000 participants representing over 600 organizations and individual members in 100 countries around the world.

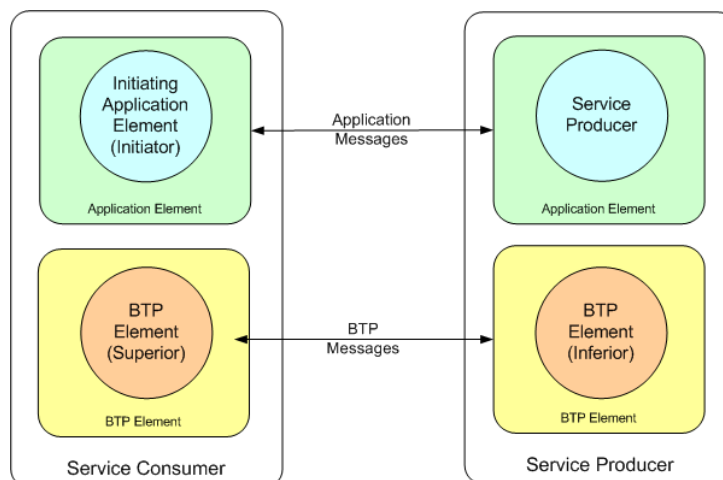
OASIS Business Transaction Protocol

- BTP 1.0 published In May 2002 by the OASIS Business Transaction Technical Committee (BTTC)
- BTP 1.0 is the work of several companies (BEA, IBM, Sun, HP, Choreology, ORACLE, and others) that hopes to become a standardized Internet-based means of managing complex, ongoing business-to-business (B2B) transactions among multiple organizations.
- One of the key design factors that BTP recognizes is that in a business transaction, no single party controls all resources needed - assumed that the respective parties manage their own resources but the activities will be coordinate in a defined manner to accomplish the work scoped by a transaction.

OASIS Business Transaction Protocol

- provision for individual service providers to decide if they want to be part of a transaction or not
 - if they decide that they want to be part of the transaction, they must provide a mechanism to confirm or cancel their commitments to the transaction.
- Although not design specifically for Web Services, the protocol will be especially useful in a Web Services environment.
- The BTP specification was formed to address the needs of inter-organizational transactions and of workflow systems in general. It was also design to overcome the limitations of similar coordination protocols tied to communication protocols.

BTP – How it works



BTP – Types of Transactions

- In traditional transactions, a transaction manager will roll back a transaction if any resource manager participating in the transaction cannot commit or cannot prepare.
- But in BTP, this cannot be assured. With BTP, you have to define the set of participants that must confirm before a transaction can be committed; this group of participants makes up what is known as the confirm-set. The confirm-set may include all or a subpart of all the participants.

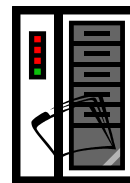
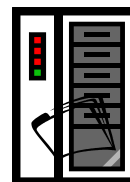
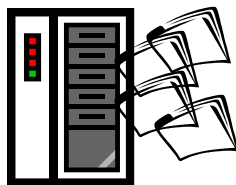
BTP – Types of Transactions

- To cater for this, BTP has defined 2 types of transactions:
 - **BTP Atomic Business Transactions**, or atoms, are like traditional transactions, with a relaxed isolation property.
 - **BTP Cohesive Business Transactions**, or cohesions, are transactions where both isolation and atomicity properties are relaxed.

Atomic Business Transactions (Atoms)

- Atomic business transaction is where all participants have to agree before a transaction can be committed.
- If any participant cannot confirm, the entire transaction is canceled.
- Because BTP transactions do not require strict isolation, it is up to each participating service to determine how to implement transaction isolation.
- In an atomic Business Transaction, the confirm-set is the set of all inferiors and any of the inferior elements has power to veto the transaction.

Atomic Business Transactions (Atoms)



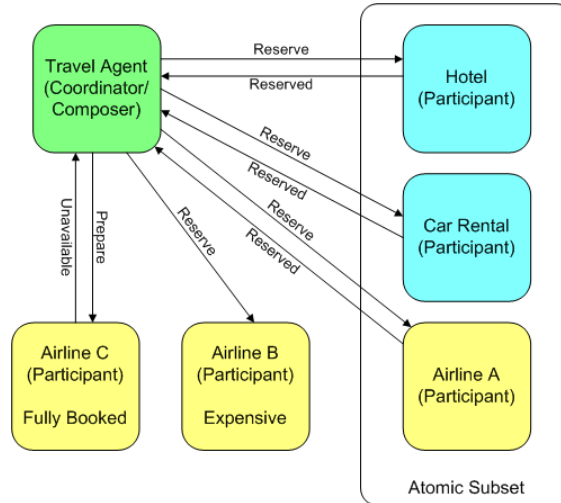
Cohesive Business Transactions (Cohesions)

- Cohesive Business Transactions are transactions where not all involved parties must agree to commit their changes before a transaction is committed. Only some subset of the parties may need to agree.
- The subset of parties that need to agree before a transaction can be completed is usually determined through the business logic in the application.

Cohesive Business Transactions (Cohesions)

- The BTP element (superior) at the service consumer end is called a composer (as oppose to as a coordinator).
- The BTP element associated with the service producer is called a participant.
- The business logic in initiating application element can determine whether the transaction can be completed
- If only one participant must confirm but both eventually confirm, the composer will ask the unwanted participant to cancel or roll back his part of the transaction.

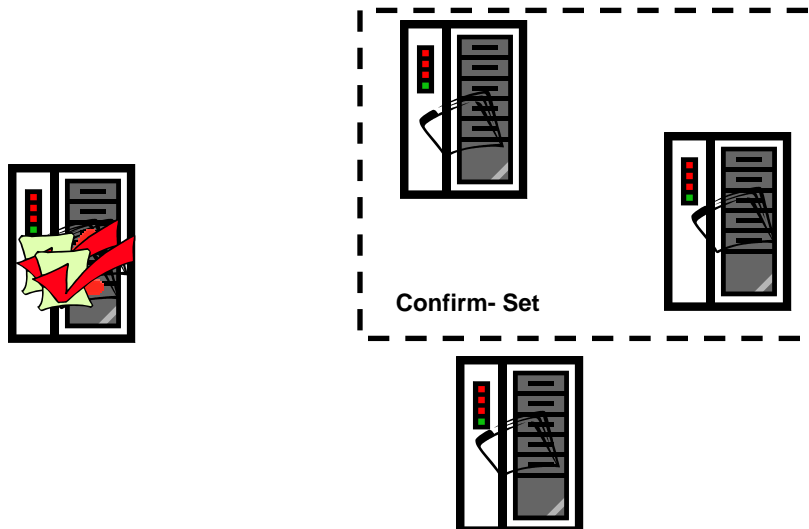
Cohesive Business Transactions (Cohesions)



Transaction in Web Services

19

Cohesive Business Transactions (Cohesions)



Transaction in Web Services

20

Critique of OASIS BTP

- Been worked on longer, considered well-formed and complete;
Although it is pretty straightforward, the specification is very full and hard for customers to digest.
- Uses business logic to control the flow of the transaction; while this seems to give you more control over how the transaction will flow, it reduces what you would expect from a transaction protocol like consistency, isolation, etc.
- when using BTP, you would have to do a lot more work to ensure a transaction is valid.
Because of this reliance on the business logic to flow the transaction, the user or initiator has to be very close to (or even be) the coordinator.

Critique of OASIS BTP

- Critical business information such as the ability for a participant to remain prepared (for example, hold onto a hotel room) for a specific period of time is propagated from the participant to the coordinator; but there is nothing within the protocol to allow this information to filter up to the application/client where it really belongs.
- Because of the lack of flow control in the protocol, in order to use cohesions it is also necessary for Web services to expose back-end implementation choices about participants.

WS-Transaction, WS-Coordination

- In July 2002, BEA, IBM and Microsoft released the specification
- Designed to support business transactions over Web Services
- It provides business process management, transactional integrity and generic coordination.

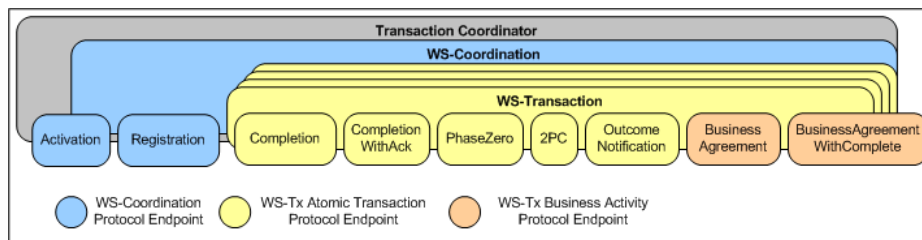
WS-Transaction

- Differentiates it from traditional transaction protocols is that synchronous request/response is not assumed
- It is layered upon the WS-Coordination protocol whose own communication patterns are asynchronous by default

WS-Coordination

- **Activation Service**
 - Enables application to create coordination instance or context
- **Registration Service**
 - Enables application to register for coordination protocol

WS-Coordination Protocol



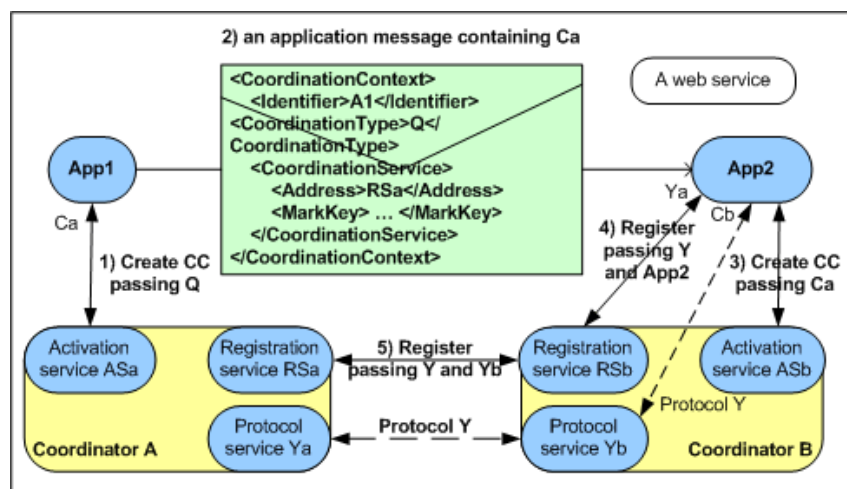
WS-Coordination

- Activation service enables an application to create a coordination instance or context. (Option for Coordination Service)
- Registration service enables an application to register for coordination protocol. (Mandatory for Coordination Service)
- Coordination service type (Stock Trades, Supply Chain). The protocol has to be understood by both ends of the communication

Transaction in Web Services

27

WS-C/WS-Transaction Coordination Service



Transaction in Web Services

28

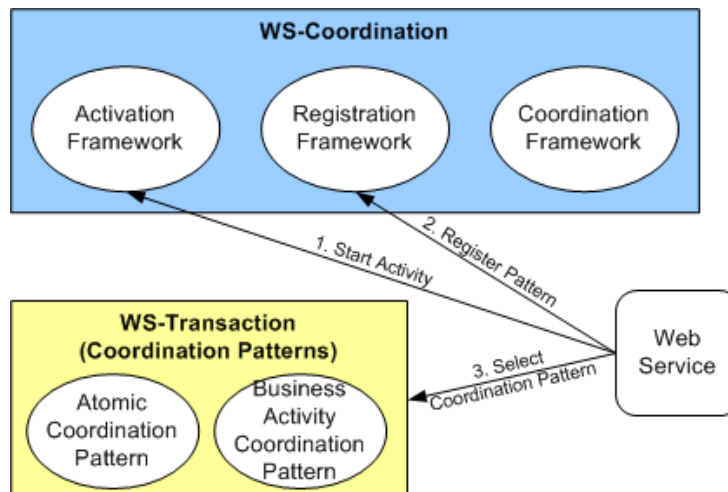
Sample CoordinationContext Envelope between Two Parties

```

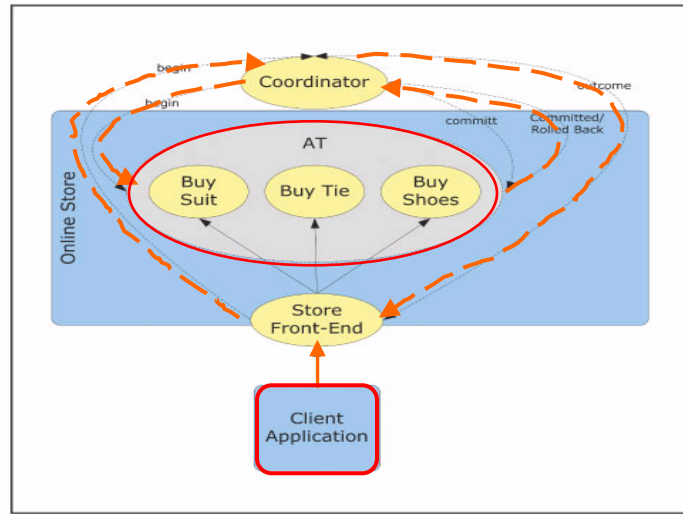
■ <?xml version="1.0" encoding="utf-8"?>
■ <S:Envelope xmlns:S="http://www.w3.org/2001/12/soap-envelope"
■   <S:Header>
■     ....
■     <wscoor:CoordinationType>
■       http://schemas.xmlsoap.org/ws/2002/08/wsba
■     </wscoor:CoordinationType>
■     <wscoor:RegistrationService>
■       <wsu:Address>
■         http://Schedule456.com/mycoordination-service/registration
■       </wsu:Address>
■       <myApp:Myapp:BetaMark> ... </myApp:Myapp:BetaMark>
■       <myApp:EBDCode> ... </myApp:EBDCode>
■     <myService:NestedCreate wsu:MustUnderstand="true"></myService:NestedCreate>
■     </wscoor:RegistrationService>
■   </S:Header>
■   ....
■ </S:Envelope>

```

WS-Transaction Specification



Atomic Transactions – All or Nothing



Transaction in Web Services

31

WS-C/WS-Transaction

- Business Activity Coordination Protocols
 - *BusinessAgreement* Protocol
 - *BusinessAgreementWithComplete* Protocol

Transaction in Web Services

32

BA Coordination Protocol

■ **BusinessAgreement Protocol**

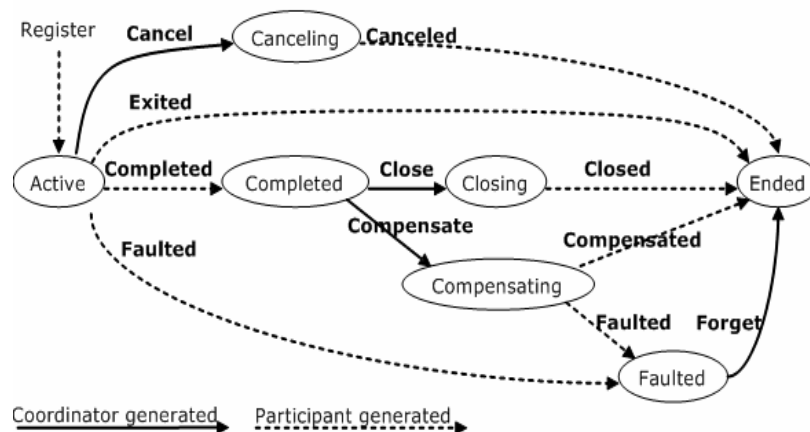
- Coordinators send 4 types of messages in Business Agreement Protocol
 - Cancel, Close, Compensate, Forget
- Participants send 6 types of messages
 - Completed, Faulted, Compensated, Closed, Canceled, Exited

■ **BusinessAgreementWithComplete Protocol**

- Coordinators send 5 types of messages in Business Agreement Protocol
 - Cancel, Close, Compensate, Forget, Complete
- Participants send 6 types of messages
 - Completed, Faulted, Compensated, Closed, Canceled, Exited.

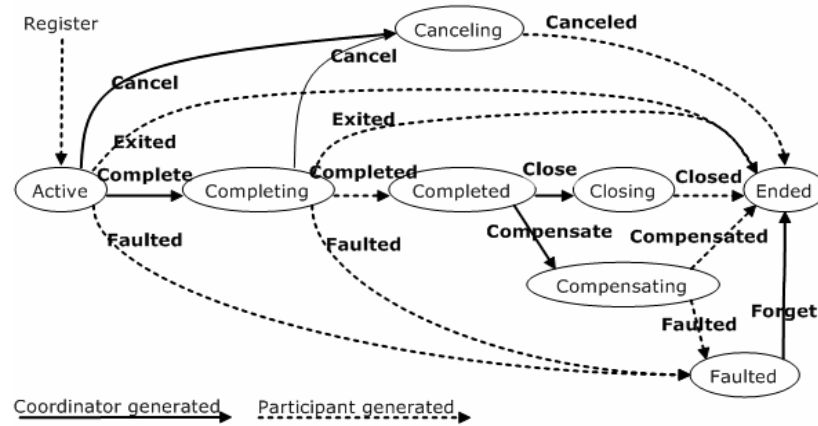
BusinessAgreement Protocol

BusinessAgreement Protocol State Diagram



BusinessAgreementWithComplete Protocol

BusinessAgreementWithComplete Protocol State Diagram



Transaction in Web Services

35

Compare Atomic and BA Protocol

	Atomic Protocol	BA Protocol
Average Execution Time	Short	Long
Scope	Mostly Internal Systems	Interoperates with multiple external systems
Resource Locking	Lock the resource. Prevent changes from other transactions	Don't lock the resources. Flexible isolation policies or compensations
Roll Back	Abort transaction	Use compensation to reverse the effects of the original business task
Request Time out	Abort transaction and retry	Resent request

Transaction in Web Services

36

Critique of WS-C/WS-Transaction

- Applications have to communicate through coordination services, which could exist outside of company firewall.
- Domain-specific coordination protocols have to be created and inserted into coordination services. **Out of box WS-coordination service provides only activity and registration services.**
- Most business logics are defined in the coordination protocol services within the coordinator. **It is not flexible to change the business logics.**

Transaction in Web Services

37

Comparison of Protocols

	BTP	WS-Transaction
Coordination framework	None, tied to 2-phase	WS-Coordination
Transaction framework	General protocol, statically defined	None, but current defined protocols cover typical patterns (AT and BA)
Strict atomic model	Atom, which is atomic only, other properties specified by service (not available via protocol). Uses open-top protocol which makes interoperability with existing transaction systems difficult.	Atomic Transactions, which requires strict ACID properties, specifically for interoperability with traditional transaction systems.

Transaction in Web Services

38

Comparison of Protocols

	BTP	WS-Transaction
Relaxed model	Cohesion allow flexible participant list. Requires participants to be exposed to application/terminator	Business Activity allows flexible participant list.
Scopes	No. Cohesion manages relationship within scope.	Yes, Business Activity manages relationship between scopes. Nested scopes allowed.
Flexible outcomes for consensus groups	Yes, via Cohesions	Yes, via Business Activity.
Flexible participation in consensus groups	Yes, participants can resign from Cohesion.	Yes, participants can exit in Business Activity protocol

Transaction in Web Services

39

Comparison of Protocols

	BTP	WS-Transaction
Service behavior	Services define behavior (not specified by BTP)	Defined by the protocol
Business logic/coordinator separation	Mixed (open-top protocol requires strong coupling between business logic and coordinator)	Distinct
Web services-specific	No, requires a lot of extra effort from the specification/protocol	Yes
Failure recovery	Re-drive protocol	Optimized protocol

Transaction in Web Services

40

Conclusion

- In our review, we have found that both models are relevant and can be used in many of today's transactional context.
- BTP allows finer control and flexibility of implementing business transactions, therefore in cases where the transaction logic and participants are simple, BTP offers a more attractive approach.
- With WS-transactions, the infrastructure to implement it is more extensive as you would also require a coordination service which could incur higher cost.

Recommendation

- Our team's recommendation is to examine the nature of the business transactions and the complexity of the transaction workflow together with security and infrastructure concerns of the organization before determining the right fit.
- We believe that both models have their purposes and will be useful in their own ways depending on environment.

Questions

