Organization for the Advancement of Structured Information Systems

Business Transaction Protocol Primer

An OASIS Committee Supporting Document

Version: 1.0

3 June 2002

Copyright and related notices

OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS's procedures with respect to rights in OASIS specifications can be found at the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification, can be obtained from the OASIS Executive Director.

OASIS invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to implement this specification. Please address the information to the OASIS Executive Director.

Acknowledgements

Employees of the following companies participated in the finalization of this specification as members of the OASIS Business Transactions Technical Committee:

BEA Systems, Inc.
Bowstreet, Inc.
Choreology Ltd.
Entrust, Inc.
Hewlett-Packard Co.
Interwoven Inc.
IONA Technologies PLC
Oracle Corp.
SeeBeyond Inc.

Sun Microsystems Computer Corp.
Talking Blocks Inc.

And contributions from the following individuals

Alex Ceponkus
Gordon Hamilton
Bill Pope

Development of this primer did not involve a formal and exhaustive review process of the kind used to fashion the BTP 1.0 Committee Specification. As a result it represents a looser consensus from which members of the OASIS BT TC and their companies might demur in particular respects.

The primary authors and editors of the primer were:

Mark Potts
Bill Cox
Bill Pope

Contributions to its writing were made by

Sanjay Dalal
Peter Furniss
Alastair Green
Mark Little
Sazi Temel
Jim Webber

In memory of Ed Felt

Ed Felt of BEA Systems Inc. was an active and highly valued contributor to the work of the OASIS Business Transactions Technical Committee.

His many years of design and implementation experience with the Tuxedo system, WebLogic's Java transactions, and WebLogic Integration's Conversation Management Protocol were brought to bear in his comments on and proposals for this specification.

He was killed in the crash of the hijacked United Airlines flight 93 near Pittsburgh,

on 11 September 2001.

Executive Summary

Increasingly an application is no longer a stand-alone entity. Applications must access other applications and modify data held by other organizations to get their work done.

As business interactions extend over the Internet, a protocol is needed to manage the interactions in that loosely coupled, asynchronous environment, where traditional techniques of resource locking no longer work. We show a detailed set of examples in a travel agency/travel provider scenario to illustrate many concepts of BTP in a real-world business environment.

The Business Transaction Protocol, or "BTP," provides a common understanding and a way to communicate guarantees and limits on guarantees between organizations. The formal rules are necessary for the distribution

of parts of business processes outside the boundaries of an organization.   BTP solves part of the problem for developers of loosely coupled transactions—the coordination and forcing a consistent termination portions. Expertise in the design of compensating actions is still required, but these compensations are local rather than distributed.

Table of Contents

Introduction

This document is a primer on the OASIS Committee Draft of the Business Transaction Protocol, BTP 1.0. [BTP Specification] We do not cover the entire protocol, but we do introduce much of the terminology. We do not discuss the optimisations designed into the protocol (from the contributed base documents, even though they are an important factor in assuring high performance of the protocol.

We suggest that you read this Primer before reading the first section of the BTP Specification. [BTP Model]

We first describe the environment in which BTP will function, define the goals of the Business Transaction Protocol, examine a set of related examples, and conclude with some questions and answers on BTP.

Transactions in Loosely-coupled Systems

Conventional transaction processing in tightly coupled systems supports the so-called *ACID* properties or guarantees—a transaction is

- Atomic: All or nothing. If interrupted by failure, all effects are undone (rolled back).
- Consistent: A consistent result is obtained, allowing clean state changes. The effects of a transaction preserve invariant properties.
- Isolated: Effects aren't visible until all participants agree. A transaction's intermediate states are not visible to other transactions. Transactions appear to execute serially, even if they are performed concurrently.
- Durable: The effects persist after the transaction is complete. The effects of a completed transaction are persistent; they are never lost (except in a catastrophic failure).

Unfortunately, maintaining all of the transactional ACID semantics in a loosely coupled environment is not practical —in part because of the need to use compensating transactions in certain cases, in part because of more complex failures. Typical locking techniques introduce problems where the transactions may last hours, days, or even longer, so that complex lock management algorithms or new interactions need to be introduced.

Isolation is also an issue in a distributed environment, not least in that business issues argue against indefinite locking of resources.

ACID transaction processing, of course, works well in distributed environments today. But the assumptions that make a network of Automatic Teller Machines work well are not present in coordinating a group of autonomous parties. The ATMs are not autonomous in terms of the transaction protocol: they communicate with a centralized database; locking can be done within that database because the database is under the control of a single enterprise; and the loosely coupled ATMs wait synchronously for a transaction to proceed.

Finally, in distributed interactions, communication is less reliable. Connections are intermittent, load plays a significant role in performance, and (in the case of wireless mobile devices) communications may fail altogether.

We have already described the *ACID* properties for [tightly-coupled] transactions. Transaction semantics that work in a tightly coupled single enterprise cannot be successfully used in loosely coupled multi-enterprise networks such as the Internet.

The Business Transaction Protocol relaxes some of the *ACID* properties as summarized in the following table.

Property	*Tightly coupled*	*BTP Atoms*	*BTP Cohesions*
Atomic	All or nothing	All or nothing	Negotiated by participants and coordinator
Consistent	Clean state changes	Clean state changes	Clean state changes. Note that client and coordinator can have interim steps to decide what the finalization set is, but once the set is

			determined the transition to confirmed or cancelled is always clean.
Isolated	Effects aren't visible until all participants agree	Relaxed, visibility of effects controlled by the service	Relaxed, visibility of effects controlled by the service
Durable	Effects persist	Effects persist	Effects persist, some may be volatile

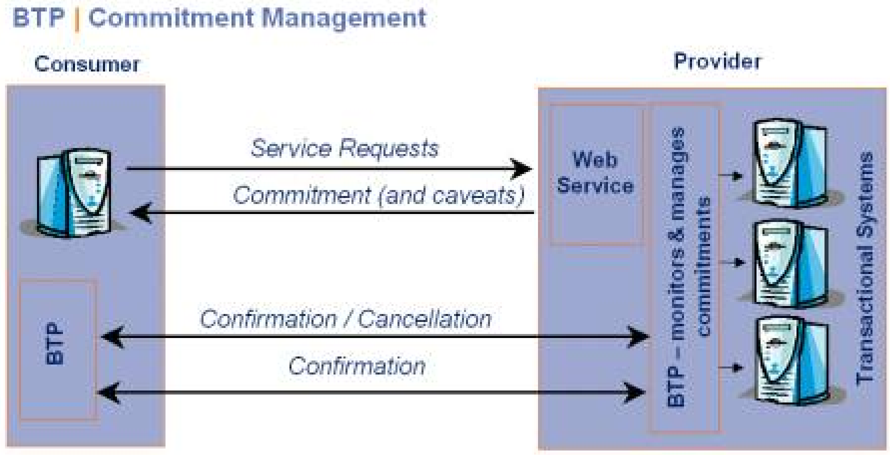
Requirements for Business Transactions

In this section, we will define BTP-specific terms and discuss how traditional transaction semantics can be weakened for the Internet. Terms defined in the BTP Specification are in italics

As more organizations attempt to integrate heterogeneous environments with different transaction coordination semantics, both inside the enterprise and with business partners, a standard coordination protocol is required. BTP is designed to allow the coordination of business transactions that span multiple participants ensuring that a transaction has a consistent without concern for whether the transaction spans disparate applications, developed with disparate technologies, and potentially deployed in different organizations.

In such circumstances, a single party does not control all the resources needed for a transaction. We assume that each participant is autonomous and must manage its own resources while maintaining commitments it has made in a transaction. This mirrors the real world of business, where there are varying degrees of privacy and control over how resources are committed and interactions managed, typically by negotiation.

In their autonomy, the participants in a business transaction may use recorded before- or after-images, or compensation operations to provide a "roll-forward, roll-back" capacity to enable coordination with respect to the overall outcome of the business transaction. Locking can work for short-lived transactions, but compensation is more suitable for longer-lived transactions.



The Provider of a service decides when to commit/unlock, and when to compensate internal transactions/resources, but must provide both *confirm* and *cancel* mechanisms for commitments it makes. The Provider decides how to meet its commitments; the Consumer manages the business transaction within the commitments made by the providers. While participants have autonomous control of their resources, they must also offer some commitment to enable the coordinated and controlled termination of the business transaction.

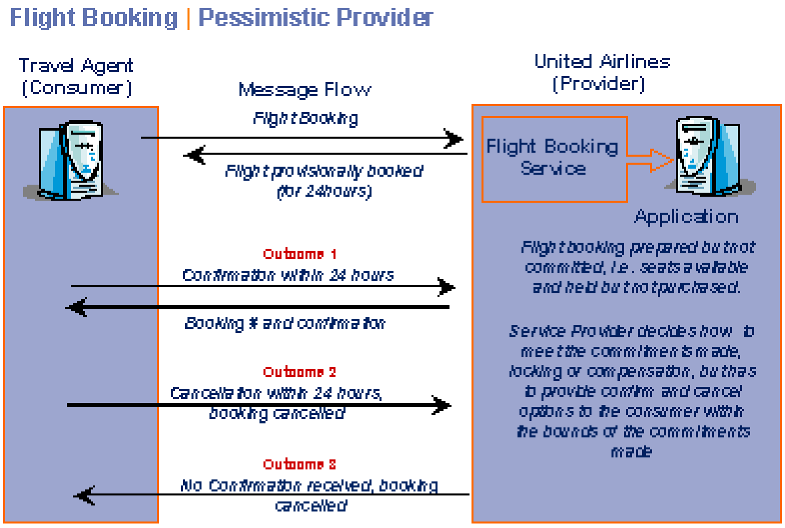
BTP is a transactional protocol that allows independent participants and coordinators to negotiate commitment to a business transaction and allows implementations to manage those commitments to coordinate termination of the business transaction.

The Business Transaction Protocol

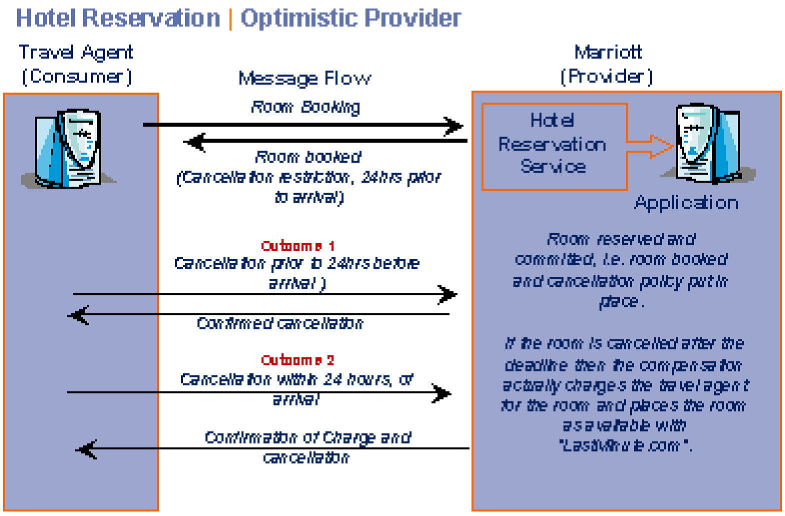
Today service providers often offer their commitments to a business transaction with caveats, or reservations. But there is no standard protocol for those commitments to be made or managed to termination—the coordination and termination of these types of transactions requires out of band *a priori* agreement between developers on both the consumer and provider sides of the transaction.

The idea of negotiating commitment to transactions already exists in many situations today. We will show realistic scenarios where commitments are made for a business transaction in a pessimistic or optimistic way. For example, in the case of booking a flight, many providers take a pessimistic approach to their commitment to the transaction.

N.B. We are using real company names in the following sections, in part to make the examples more realistic, and in part to defer (or even avoid) having to modify many graphics.

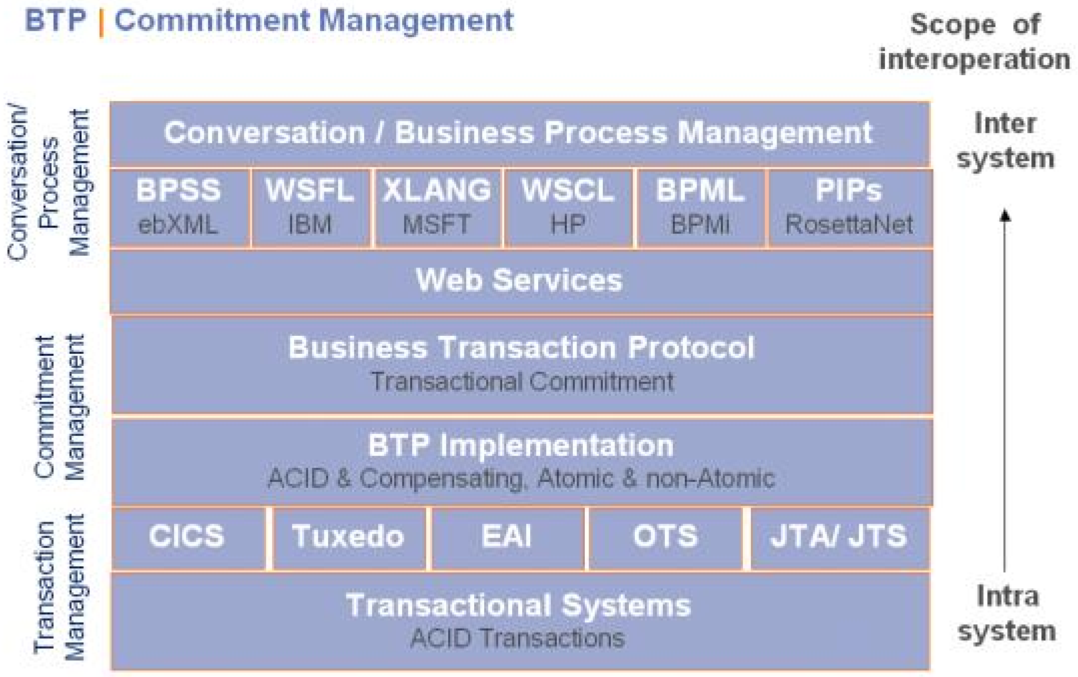

Flight Booking | Pessimistic Provider
Travel Agent (Consumer)
Message Flow
United Airlines (Provider)
Flight Booking
Flight provisionally booked (for 24hours)
Flight Booking Service
Application
Flight booking prepared but not committed, i.e. seats available and held but not purchased.
Service Provider decides how to meet the commitments made, locking or compensation, but has to provide confirm and cancel options to the consumer within the bounds of the commitments made
Outcome 1
Confirmation within 24 hours
Booking # and confirmation
Outcome 2
Cancellation within 24 hours, booking cancelled
Outcome 3
No Confirmation received, booking cancelled

In the case of booking a Hotel Room, many providers take an optimistic approach to their commitment to the transaction.


Hotel Reservation | Optimistic Provider
Travel Agent (Consumer)
Message Flow
Marriott (Provider)
Room Booking
Room booked (Cancellation restriction, 24hrs prior to arrival)
Hotel Reservation Service
Application
Room reserved and committed, i.e. room booked and cancellation policy put in place.
If the room is cancelled after the deadline then the compensation actually charges the travel agent for the room and places the room as available with "LastMinute.com".
Outcome 1
Cancellation prior to 24hrs before arrival)
Confirmed cancellation
Outcome 2
Cancellation within 24 hours, of arrival
Confirmation of Charge and cancellation

BTP's ability to coordinate between services offered by autonomous organizations makes it ideally suited for use in a Web Services environment, and as an underlying protocol for more loosely coupled business transactions semantics defined by conversation and process management standards.

See the diagram on the next page.



Goals and Motivation for BTP

The OASIS Business Transaction Technical Committee was started to support business transactions that constitute units of work across multiple decoupled, distributed parties. The protocol it has defined, the Business Transaction Protocol, is usually abbreviated *BTP*.

BTP solves problems in environments with complex business interactions with potentially unreliable infrastructure over potentially unreliable communication links. The goal of a typical business interaction is to provide a concrete completion or cancellation, under potentially complex business rules that need not (and can not) be understood by all participants.

More specifically, BTP goals include

- Define a model for transactions across the internet, with participants from different organizations

- Compose and coordinate reliable outcomes in the face of potentially unreliable communication channels and infrastructure

- Manage the transaction life cycle

- Support transactions between loosely-coupled systems communicating with each other asynchronously (for enterprise scalability and function)

- Support long-running transactions that might last longer than any business will reasonably reserve its resources for another

- Coordinate multiple related interactions

- Provide a foundation for workflow and business modeling/execution tools

We will use a set of travel reservation examples in this Primer, as many of the problems solved by BTP are present in these common interactions. For a more detailed description of the model of BTP, including a full discussion of error cases and optimizations, see Part One of the specification. [BTP Model]

Atoms and Cohesions

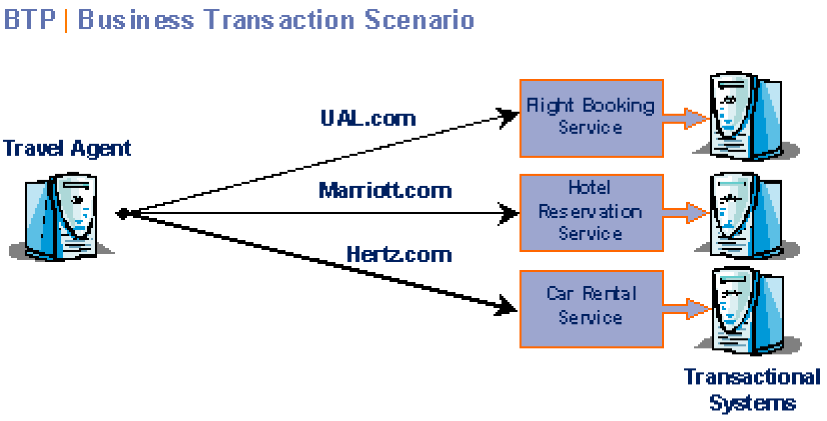
BTP Atomic Transactions, or *atoms*, are similar to transactions in tightly coupled systems, but the isolation property is relaxed, although the transactions are durable (see table below). One *atom coordinator* and zero or more *sub-coordinators* coordinate a transaction; each manages one or more *participants*. Participants act on behalf of services to either accept (confirm) or reject (cancel) the work done by the service within the scope of the atom. In addition, the outcome of an atom is atomic, such that all of the participants will either confirm or cancel.

Cohesive Business Transactions, the term for which is blended into the portmanteau word *cohesions*, similarly relax the isolation property, allowing the effects of a cohesive interaction to be externally visible before the interaction is committed. In addition, a cohesion may deliver different termination outcomes to its participants such that some will confirm whilst the remainder will cancel. Finally, consistency is determined by agreement and interaction between the client (*initiator*) and the coordinator. The *cohesive coordinator* therefore has a more complex role than the *atom coordinator*.

Cohesions may consist of multiple sub-transactions; cohesions and atoms may have multiple participants, thus generating tree diagrams of relationships in a Business Transaction.

The Travel Scenario

To help explain the roles and interactions in the Business Transaction Protocol, we use the following scenario and variations for booking a trip. The trip incorporates a flight, hotel, and car rental reservation and bookings. The scenario covers a number possible outcomes and variations that show how BTP can be used to coordinate many different types of transactional interactions.



We will use this Scenario for five examples:

1. **Single Party Atomic Transaction**

Booking a Flight at UAL.com

2. **Multi Party Atomic Transaction**

Booking a Flight, Hotel and Car in one business transaction

3. **Single Service Type Cohesion**

Price comparison of a flight through UAL.com, BA.com and Qantas.com, subsequently confirming the best option and cancelled the alternatives.

4. **Multi Service Type Cohesion**

Booking a Flight, Hotel, and Car in one business transaction but including a price comparison for the flight portion through UAL.com, BA.com, and Qantas.com, and subsequently confirming one of the Flight options and Hotel and cancelling the Car bookings.

5. **Multi Party Compound Transaction**

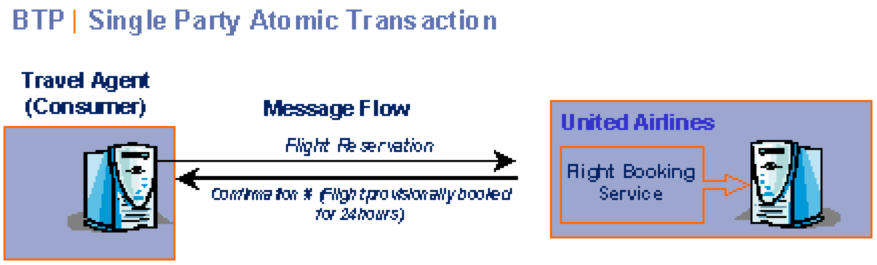
A Consumers view of booking a Flight, Hotel and Car in one business transaction through a Travel Agent Service, where the Travel Agent acts as an intermediary of the consumers behalf.

We indicate alternative steps with a decimal notation—for example, Stage 5.1, Stage 5.2, and Stage 5.3 might all be valid continuations from a Stage 4.

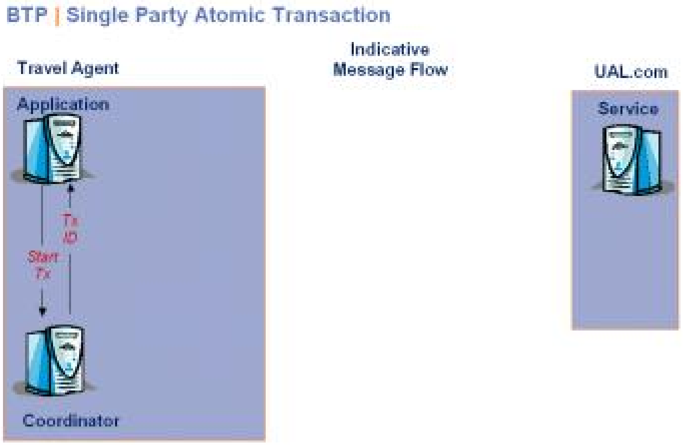
We do not discuss the optimisations designed into BTP (from the contributed base documents), even though they are an important factor in assuring high performance of the protocol. Committee Specification describes the optimisations. [BTP Specification]

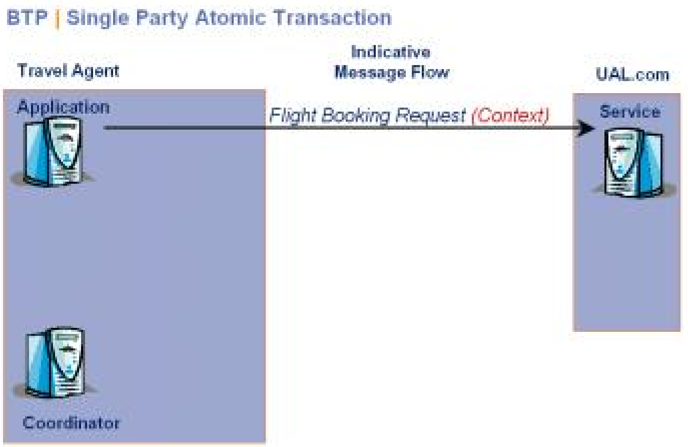
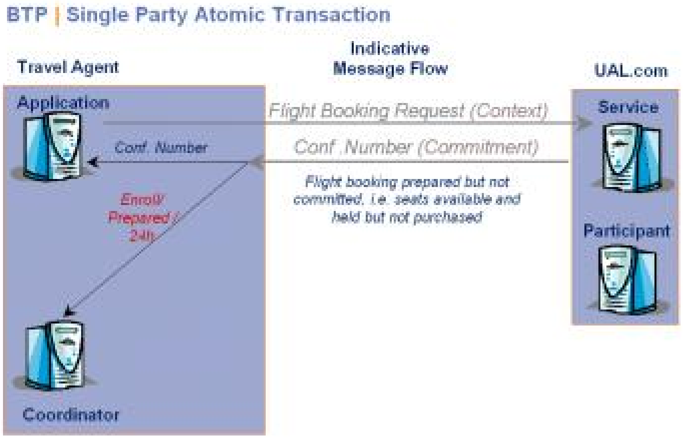
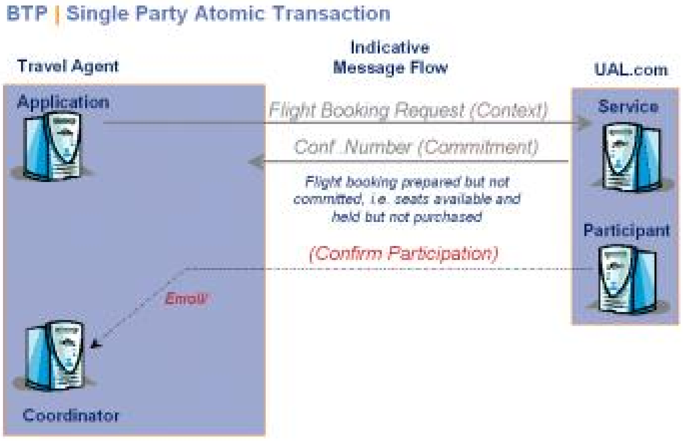
Example 1—Single Party Atomic Transaction

This is the simplest of the four examples in the scenario but one that we will explore in depth to convey the roles, messages and sequence of those messages, the BTP protocol. As we examine the other examples we will be able to apply that understanding to the more complex examples without repeating all the lower level details.

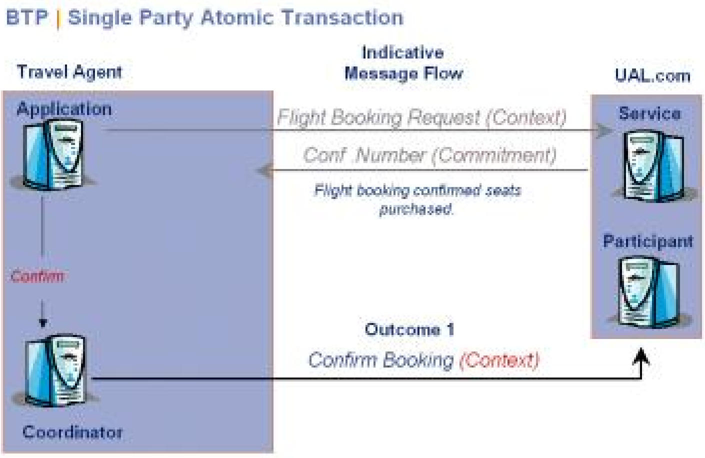

BTP | Single Party Atomic Transaction
Travel Agent (Consumer)
Message Flow
Flight Reservation
Confirmation # (Flight provisionally booked for 24 hours)
United Airlines
Flight Booking Service

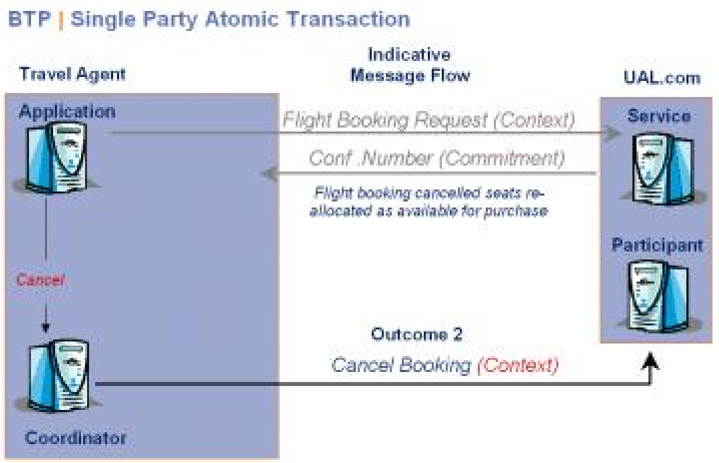
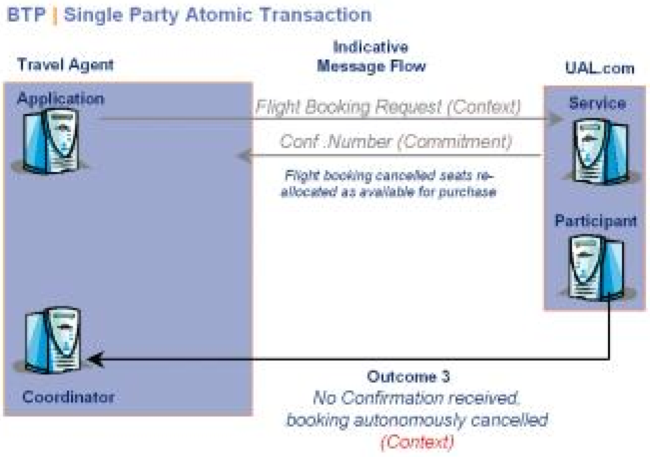
In looking more deeply at this transaction, we will see the phases of the transaction, the message exchanges, and commitments made by the involved parties.

Stage 1: Firstly the Travel Agent (*Initiator*) creates a business transaction (*Context*) for the work it wants to accomplish. It does this through, what we will abstractly refer to as, a Coordinator (*Factory/Coordinator/Decider*).	 BTP | Single Party Atomic Transaction Travel Agent Indicative Message Flow UAL.com Application Service Tx ID Start Tx Coordinator
Stage 2: The Travel Agent (*Initiator*) then makes the Flight Booking Request to UAL.com also propagating the transaction details (*Context*).	

	 BTP | Single Party Atomic Transaction Travel Agent Indicative Message Flow UAL.com Application Flight Booking Request (Context) Service Coordinator
Stage 3.1: UAL.Com responds to the request with a confirmation number for the flight it reserved for the Travel Agent. Included in the response, in this particular case, is information and the state of the transaction from UAL.com's (*Participant*) perspective. The response also confirms UAL.com participation in the transaction (*Enroll*) and makes a commitment to the Travel Agent that it will hold this flight on the travel agents behalf for the next 24hours (*Prepared+ Timeout*).	 BTP | Single Party Atomic Transaction Travel Agent Indicative Message Flow UAL.com Application Flight Booking Request (Context) Service Conf. Number Conf. Number (Commitment) Flight booking prepared but not committed, i.e. seats available and held but not purchased Enroll/ Prepared / 24h Participant Coordinator
Stage 3.2: Alternatively UAL.Com confirms with the Travel Agent its participation in the transaction (*Enroll*) separately from responding to the application request. In other longer running scenarios this may be appropriate. UAL.Com would then after a time responds and possibly declare its commitment (*Prepared* + *Timeout*)	 BTP | Single Party Atomic Transaction Travel Agent Indicative Message Flow UAL.com Application Flight Booking Request (Context) Service Conf. Number (Commitment) Flight booking prepared but not committed, i.e. seats available and held but not purchased Participant (Confirm Participation) Enroll Coordinator

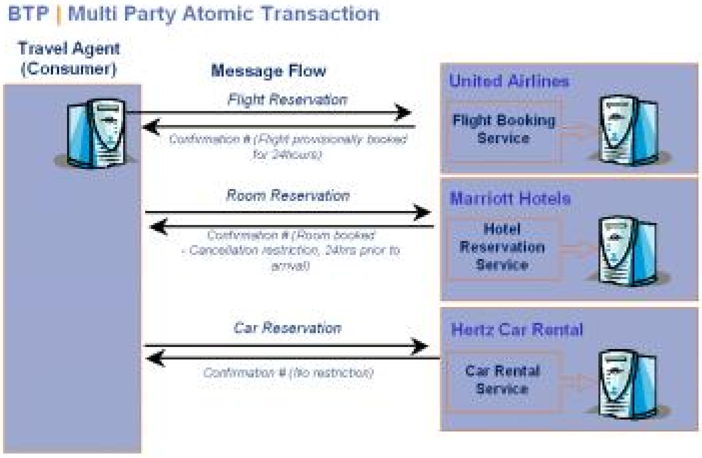
The Travel Agent now has 24 hours to make a decision about whether to accept and confirm the flight booking. After 24 hours UAL is under no obligation to honor its commitment. There are now three possible outcomes for the business transaction.

Stage 4.1: The Travel Agent confirms the booking within 24 hours by informing the Coordinator that it wants to confirm the booking (*Confirm-Transaction*). Because UAL.com has already made a commitment to the transaction, the coordinator simply confirms the booking with UAL (*Confirm*). The request was made within the specified time period, so UAL.com will go ahead and issue the tickets (*Confirmed*) and bill the travel agent. Finally, the Coordinator confirms the successful conclusion of the business	 BTP | Single Party Atomic Transaction Travel Agent Indicative Message Flow UAL.com Application Flight Booking Request (Context) Service Conf. Number (Commitment) Flight booking confirmed seats purchased. Participant Confirm Outcome 1 Confirm Booking (Context) Coordinator

transaction back to the Travel Agent (*Transaction Confirmed*).	
Stage 4.2: The Travel Agent cancels the booking within 24 hours. The Travel Agent does this by informing the coordinator its wish to cancel the booking (*Cancel-Transaction*). The coordinator now simply cancels the booking with UAL (*Cancel*). UAL would respond to the request to cancel, by releasing the seats on the flight (through whatever means it wishes) and confirming this action back to the Travel Agent (*Cancelled*). The Coordinator finally confirms the cancellation of the business transaction back to the Travel Agent (*Transaction Cancelled*).	 BTP | Single Party Atomic Transaction Travel Agent Indicative Message Flow UAL.com Application Flight Booking Request (Context) Service Conf .Number (Commitment) Flight booking cancelled seats re-allocated as available for purchase Participant Cancel Outcome 2 Cancel Booking (Context) Coordinator
Stage 4.3: The Travel Agent neither confirms or cancels the booking within 24 hours. In this case UAL.com autonomously cancels the booking, releasing the seats on the flight (through whatever means it wishes) and informs the travel agent of the action it has taken (*Cancelled*). If the Travel agent then tried to confirm the reservation the Coordinator would report back to the Travel Agent the transaction was cancelled (*Transaction Cancelled*).	 BTP | Single Party Atomic Transaction Travel Agent Indicative Message Flow UAL.com Application Flight Booking Request (Context) Service Conf .Number (Commitment) Flight booking cancelled seats re-allocated as available for purchase Participant Coordinator Outcome 3 No Confirmation received, booking autonomously cancelled (Context)

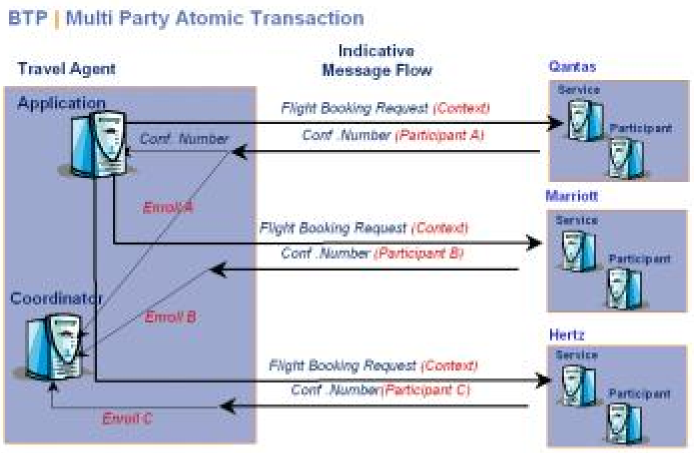
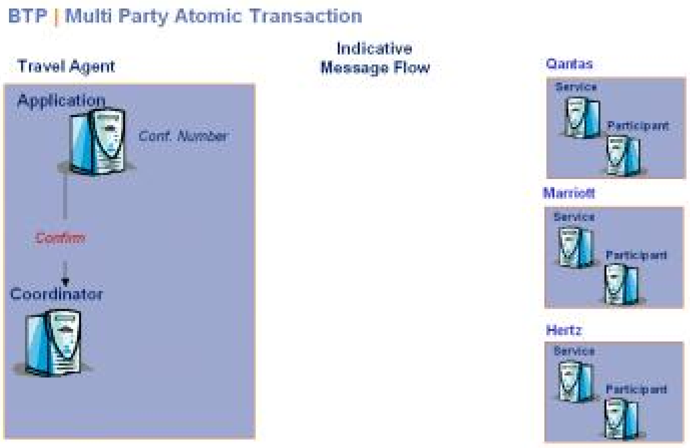
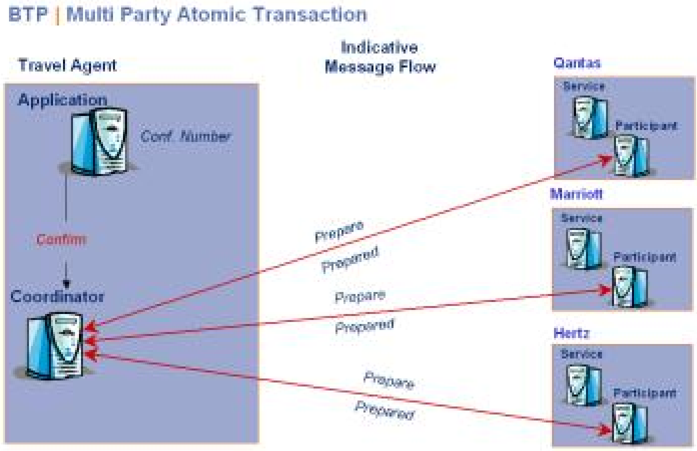
Example 2—Multiple Party Atomic Transaction

This example extends the Single Party Atomic Transaction case by adding more participants to the transaction. In the example below, the same interactions will exist, but this time the trip that we are booking includes multiple components not just a flight. Specifically this trip includes booking a Flight, reserving a hotel room and renting a Car. Without being able, to secure all three components of the trip we do not want to go ahead with the business transaction.

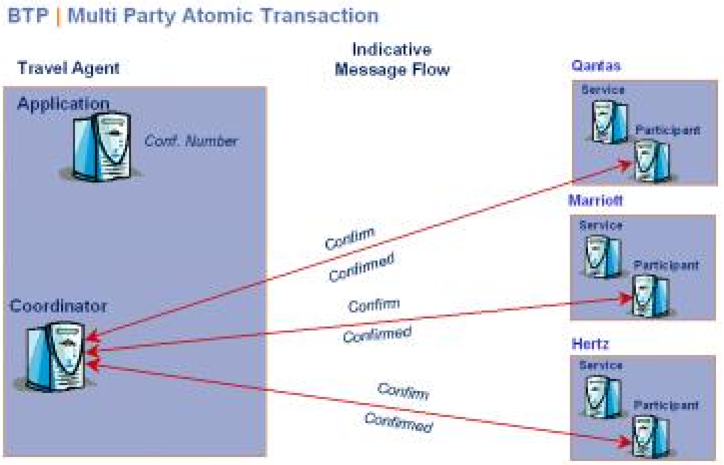

BTP | Multi Party Atomic Transaction
Travel Agent (Consumer)
Message Flow
Flight Reservation
Confirmation # (Flight provisionally booked for 24hours)
United Airlines
Flight Booking Service
Room Reservation
Confirmation # (Room booked - Cancellation restriction, 24hrs prior to arrival)
Marriott Hotels
Hotel Reservation Service
Car Reservation
Confirmation # (No restriction)
Hertz Car Rental
Car Rental Service

In BTP terms the transaction in this case is still atomic insofar as the participants will all see the same outcome: The Travel Agent needs to get agreement commitments to the business transaction and all the bookings need to either, complete successfully or fail, as a single business transaction.

We will not repeat the similar setup stages in the remaining examples.

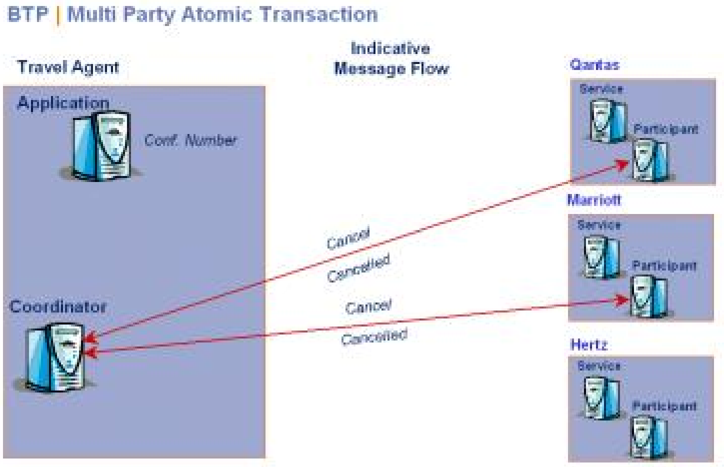
Stage 1: Firstly the Travel Agent (*Initiator*) creates a business transaction (*Context*) for the work it want to accomplish. It does this through a Coordinator (*Factory/Coordinator/Decider*).	
Stage 2: The Travel Agent (*Initiator*) then makes the Service Requests to Qantas.com Marriot.com and Hertz.com also propagating the transaction details (*Context*).	
Stage 3: Qantas.com Marriot.com and Hertz.com (*Participants*) all agree to participate in the transaction (*Enroll*). As in example 1 they could also make commitments with regard to the business transaction in their replies to the travel agent. In this example no commitments are made at this time.	 BTP | Multi Party Atomic Transaction Travel Agent Indicative Message Flow Qantas Service Participant Application Flight Booking Request (Context) Conf .Number (Participant A) Conf. Number Enroll A Marriott Service Participant Flight Booking Request (Context) Conf .Number (Participant B) Coordinator Enroll B Hertz Service Participant Flight Booking Request (Context) Conf .Number(Participant C) Enroll C
Stage 4: Once all parties have agreed to participate (*Enrolled*) and informed the Travel Agent of this (*Context Reply*), the Travel Agent can then confirm the booking. (*Confirm Transaction*).	 BTP | Multi Party Atomic Transaction Travel Agent Indicative Message Flow Qantas Service Participant Application Conf. Number Marriott Service Participant Confirm Coordinator Hertz Service Participant
Stage 5: Because there are multiple parties in this transaction the Coordinator then asks each party (*Participants*) to make a commitment with regard to the overall business transaction (*Prepare*). Qantas.com, Marriott.com, and Hertz.com (*Participants*) all make commitments to the business transaction with caveats in some cases. Positive commitments mean there are seats available on the flight requested through Qantas.com, there are rooms available on the dates requested at the Marriott hotel, and there is a Car available for rent for the period requested through Hertz.	 BTP | Multi Party Atomic Transaction Travel Agent Indicative Message Flow Qantas Service Participant Application Conf. Number Marriott Service Participant Confirm Prepare Prepared Coordinator Prepare Prepared Hertz Service Participant Prepare Prepared
Stage 6.1: The Coordinator receives	

positive commitments from Qantas.com Marriot.com and Hertz.com with regard to the business transaction (*Prepared*). The Coordinator then proceeds with the transaction (*Confirm*), based on the combined positive commitments made by each party. Each party also signals back to the coordinator the verification of success (*Confirmed*). The Coordinator finally confirms the successful conclusion of the business transaction back to the Travel Agent (*Transaction Confirmed*).



Stage 6.2: The Coordinator receives positive commitments from Qantas.com and Marriott.com ***but not*** from Hertz.com with regard to the car rental. Because this business transaction is Atomic in nature, the Coordinator must cancel the flight booking through Qantas.com and the room reservation with Marriott. The Coordinator therefore issues a Cancel instruction to Qantas.com and Marriott.com (*Cancel*). Each party also signals back to the coordinator the verification of cancel request (*Cancelled*).

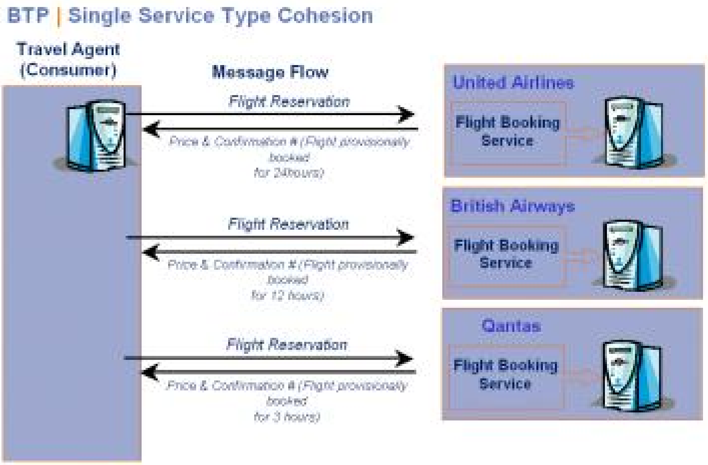
Because Hertz has already cancelled, there is no need to go back to them. Finally, the Coordinator informs the Travel Agent of the failure of business transaction (*Transaction Cancelled*).

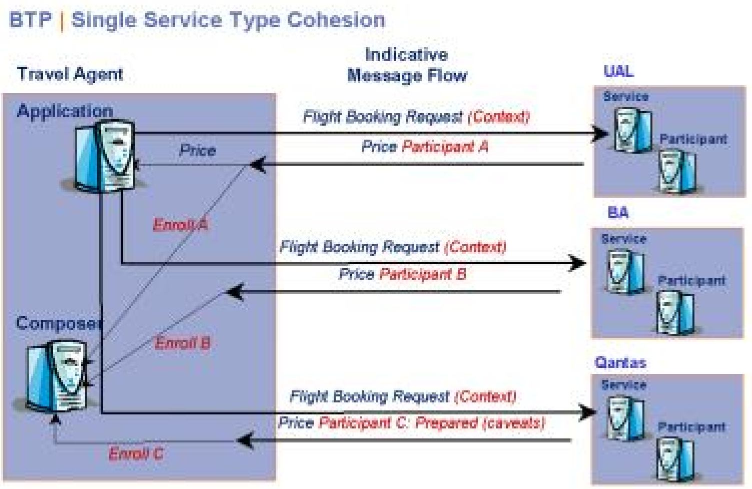
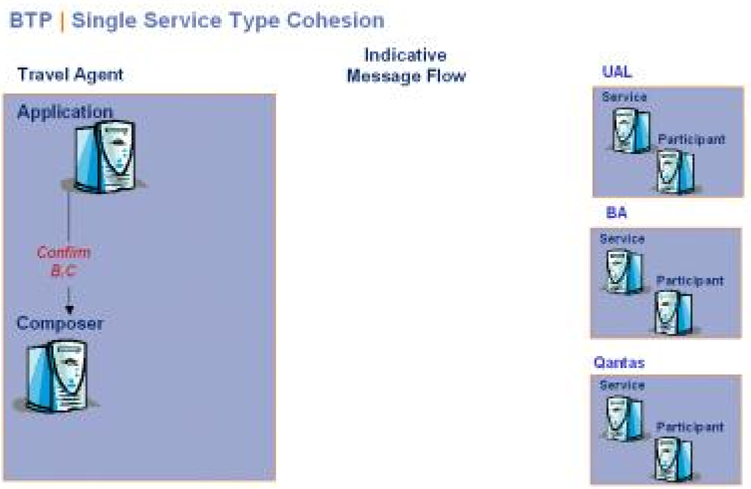


Example 3—Single Service Type Cohesion

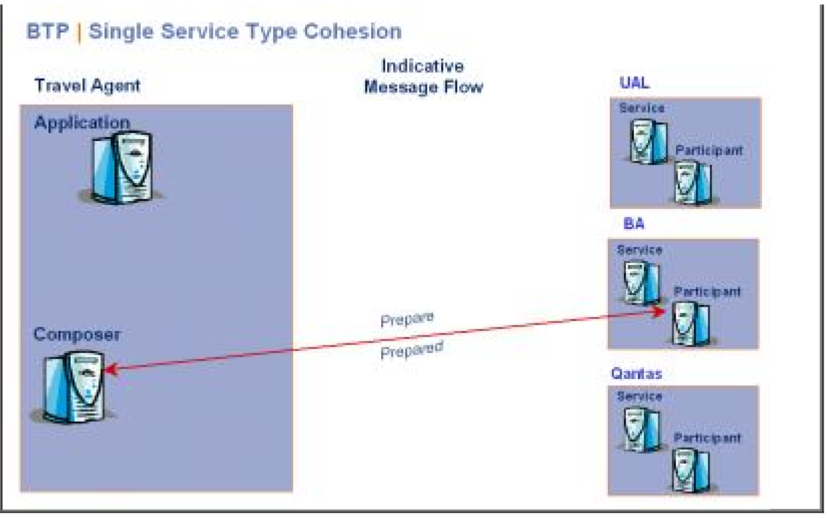
This example discusses a cohesion type transaction that relaxes not only isolation levels but allows for the relaxing of atomic properties prior to confirming or canceling a transaction. In this example, the Travel Agent chooses to start a transaction and book a flight to London. One flight option is direct on UAL and the other has two legs and two different carriers BA and Qantas. Eventually the travel Agent has to decide on one of the flights either the direct UAL flight or the combined Qantas/BA flight. By getting commitments for both the UAL flight or the combined Qantas/BA flight the Travel Agent can decide which to take knowing that they will always get the flight they decide upon as long as they make the booking compliant with any restriction made with the commitments received.

Given that the British Air/Qantas flights need to be taken as a pair, this example could be described with a sub-atom in the cohesion; we have chosen to instead describe business logic in a cohesion that has the same effect of booking either BA/Qantas in combination or United by itself.


BTP | Single Service Type Cohesion
Travel Agent (Consumer)
Message Flow
Flight Reservation
Price & Confirmation # (Flight provisionally booked for 24hours)
Flight Reservation
Price & Confirmation # (Flight provisionally booked for 12 hours)
Flight Reservation
Price & Confirmation # (Flight provisionally booked for 3 hours)
United Airlines
Flight Booking Service
British Airways
Flight Booking Service
Qantas
Flight Booking Service

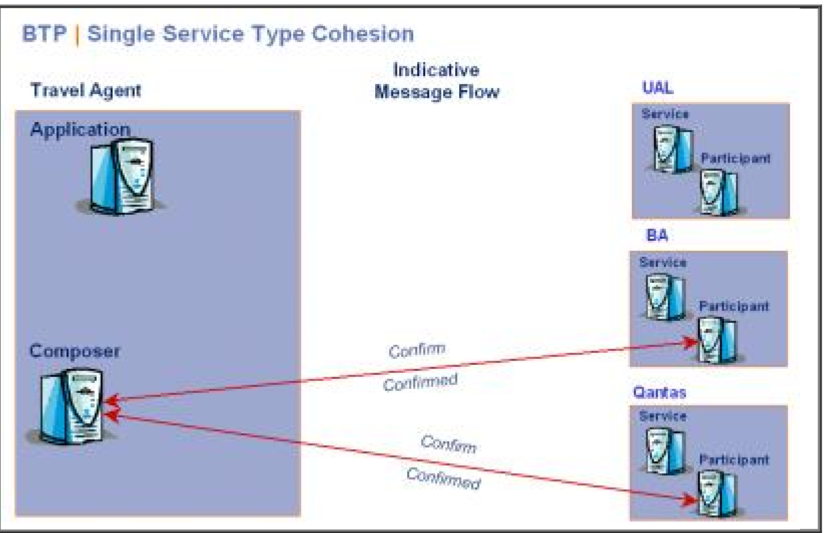
Stage 1: Firstly the Travel Agent (*Initiator*) creates a business transaction (*Context*) for the work it want to accomplish. It does this through what we abstractly refer to as a Coordinator (*Factory/Composer/Decider*).	
Stage 2: The Travel Agent (*Initiator*) then makes the Service Requests to Qantas.com UAL.com and BA.com also propagating the transaction details (*Context*).	
Stage 3: Qantas.com, UAL.com and BA.com (*Participants*) all agree to participate in the transaction (*Enroll*). In this example Qantas also makes a commitment to the transaction (*Prepared*) but UAL and BA do not.	 BTP | Single Service Type Cohesion Travel Agent Indicative Message Flow Application Composer Price Enroll A Enroll B Enroll C UAL Service Participant Flight Booking Request (Context) Price Participant A BA Service Participant Flight Booking Request (Context) Price Participant B Qantas Service Participant Flight Booking Request (Context) Price Participant C: Prepared (caveats)
Stage 4: Based on the Prices returned the Travel Agent decides to go ahead and book the two-legged flight offered by Qantas and BA. (*Confirm Inferiors B, C)*. Because UAL never made a commitment to the business transaction (*Prepared*), i.e. United did not reserve seats, there is no need to cancel the UAL flight.	 BTP | Single Service Type Cohesion Travel Agent Indicative Message Flow Application Confirm B,C Composer UAL Service Participant BA Service Participant Qantas Service Participant
Stage 5: Because the flight chosen involves two parties, Qantas and BA, the	

transaction the coordinator (*Composer*) then asks each party (*Participants*) that has not already done so to make a commitment with regard to the overall business transaction (*Prepare*). Because Qantas has already made a commitment the coordinator (*Composer*) only needs to get a commitment from BA (*Prepare*).

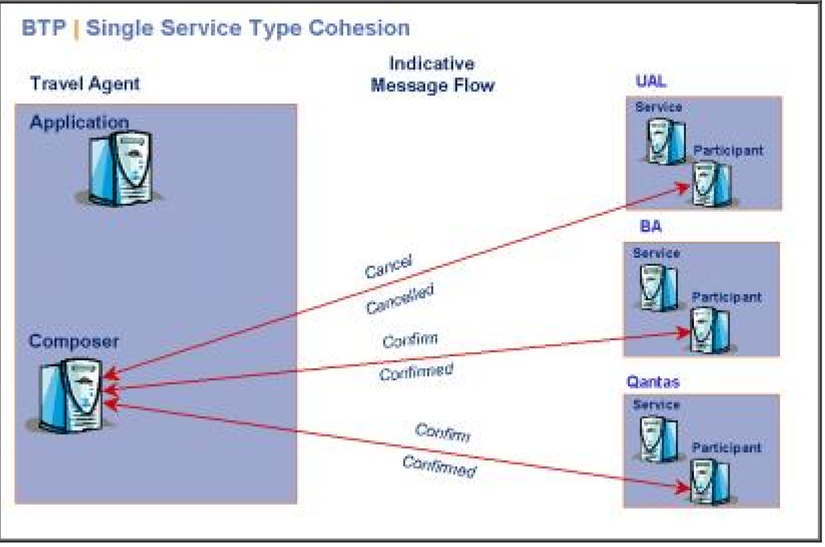


Even though this business transaction is a cohesive transaction (*cohesion*), you will notice that the final set of participants chosen from the cohesion must terminate atomically. In the example BA and Qantas need to make commitment to the transaction and complete as an atomic set, omitting the UAL flight. Again, we could have shown this as a separate atom, but instead showed how to force this outcome from the cohesion.

Stage 6.1: The Coordinator (*Composer*) now has received positive commitments from Qantas.com and BA.com, the requested portions of the business transaction requested by the Travel Agent. The coordinator (*Composer*) therefore goes ahead and confirms the seat reservations offered by BA.com and Qantas.com.



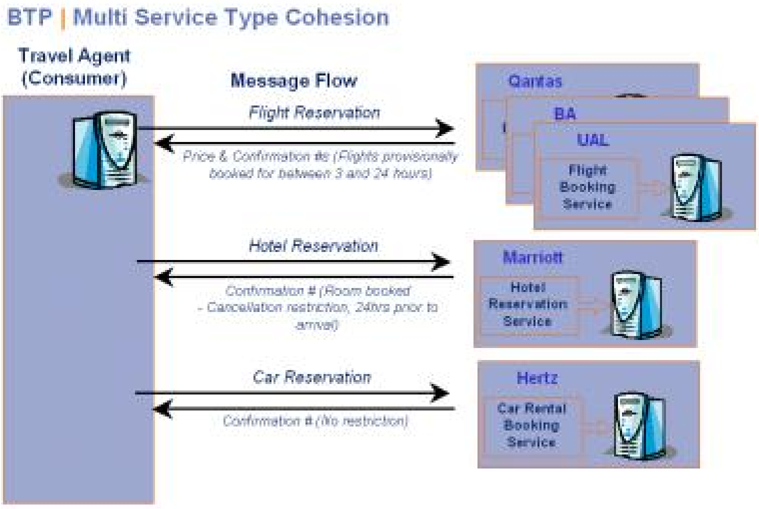
Stage 6.2: If UAL had made a commitment (*Prepared*) then the coordinator (*Composer*) would need to explicitly cancel (*Cancel*) the seats reserved by UAL as part of the business transaction, at the same time as confirming the BA, Qantas flight. Finally, the Coordinator confirms the successful conclusion of the business transaction back to the Travel Agent (*Transaction Confirmed*).

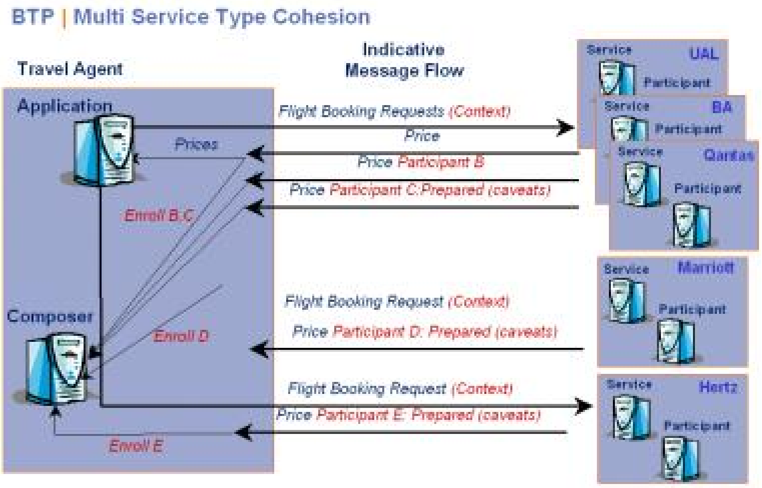


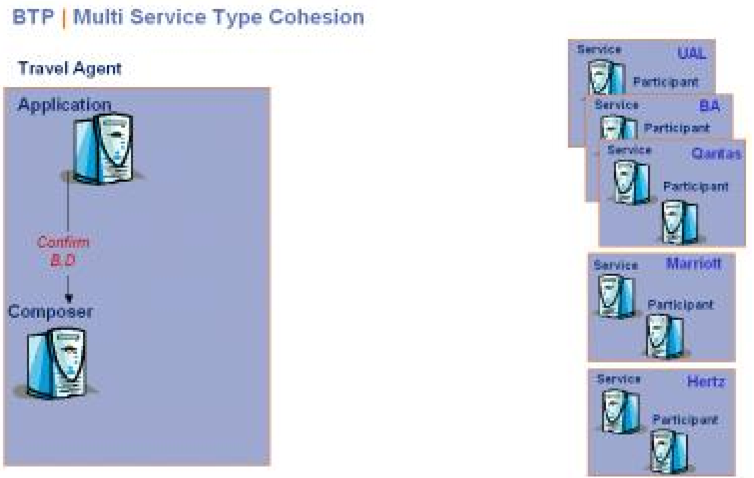
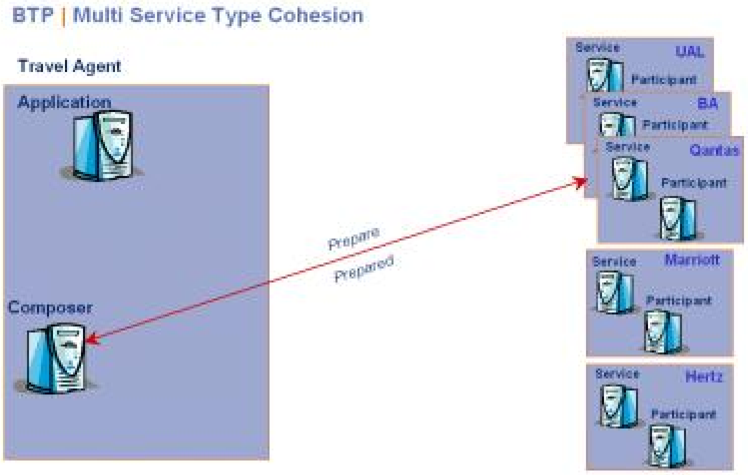
Example 4—Multiple Service Type Cohesion

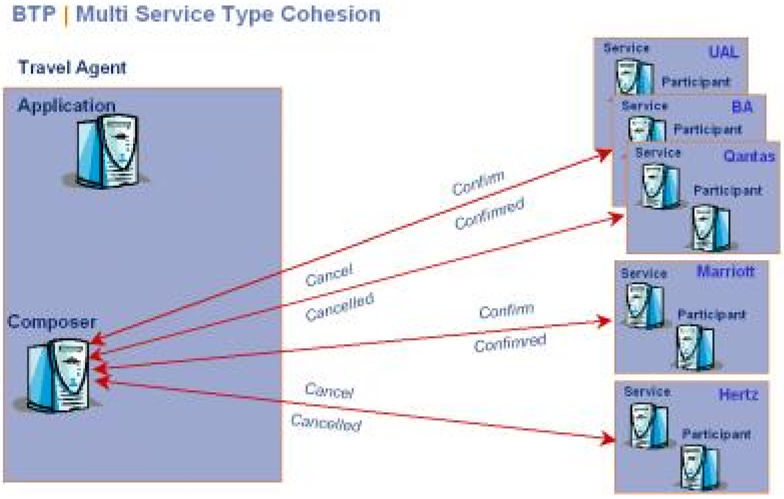
This example extends example 3 and discusses a cohesion type transaction that incorporates multiple service types and multiple services for those types. In this example, the Travel Agent chooses to start a transaction and book a vacation including Flight, Hotel and Car rental. The Hotel is specific to the location of the vacation

(Perth Western Australia), the Car will be rented through Hertz, but the flight is variable—there are options on Qantas, BA, and United.


BTP | Multi Service Type Cohesion
Travel Agent (Consumer)
Message Flow
Flight Reservation
Price & Confirmation #s (Flights provisionally booked for between 3 and 24 hours)
Hotel Reservation
Confirmation # (Room booked - Cancellation restriction, 24hrs prior to arrival)
Car Reservation
Confirmation # (No restriction)
Qantas
BA
UAL
Flight Booking Service
Marriott
Hotel Reservation Service
Hertz
Car Rental Booking Service

Stage 1: Firstly the Travel Agent (*Initiator*) creates a business transaction (*Context*) for the work it want to accomplish. It does this through a Coordinator (*Factory/Composer/Decider*).	
Stage 2: The Travel Agent (*Initiator*) then makes the Service Requests to Qantas.com, UAL.com, BA.com, Marriott, and Hertz, also propagating the transaction details (*Context*).	
Stage 3: Qantas.com, BA.com, Marriott, and Hertz (*Participants*) all agree to participate in the transaction (*Enroll*). UAL is yet to agree to participate. In this example Qantas, Marriott, and Hertz also make a commitment to the transaction (*Prepared*) but BA does not.	 BTP | Multi Service Type Cohesion Travel Agent Indicative Message Flow Application Composer Flight Booking Requests (Context) Price Prices Price Participant B Price Participant C:Prepared (caveats) Enroll B,C Enroll D Flight Booking Request (Context) Price Participant D: Prepared (caveats) Flight Booking Request (Context) Price Participant E: Prepared (caveats) Enroll E Service UAL Participant Service BA Participant Service Qantas Participant Service Marriott Participant Service Hertz Participant
Stage 4: Based on the excellent Price returned by BA, the Travel Agent decides to go ahead and books the trip with the flight from BA, as well as the Marriott Hotel, but decides to do without the car hire after reading about Perth's great public transport system (Confirm Inferiors B, D).	

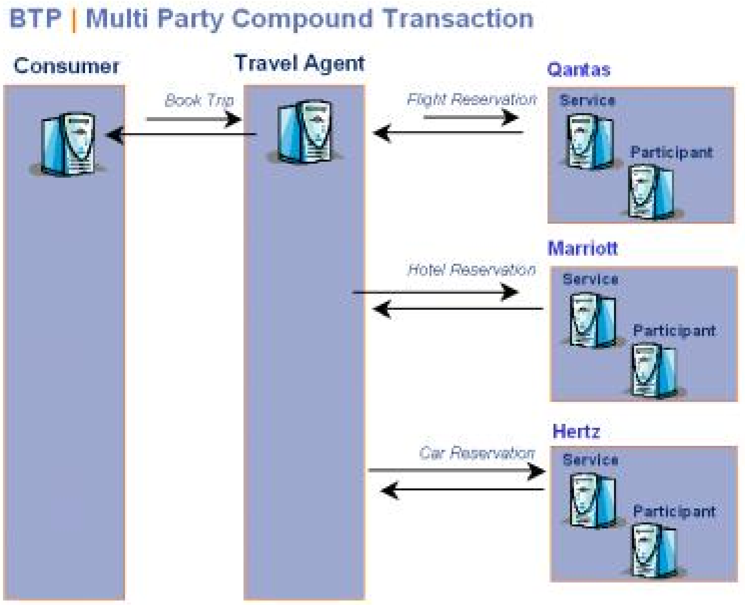
	 BTP | Multi Service Type Cohesion Travel Agent Application Confirm B,D Composer Service UAL Participant Service BA Participant Service Qantas Participant Service Marriott Participant Service Hertz Participant
Stage 5: Because BA has not made a commitment about seats on the flight, the Coordinator (*Composer*) then asks BA (*Participant*) to make a commitment with regard to the overall business transaction (*Prepare*). Because Marriott has already made a commitment the coordinator (*Composer*) only needs to get a commitment from BA (*Prepare*).	 BTP | Multi Service Type Cohesion Travel Agent Application Composer Prepare Prepared Service UAL Participant Service BA Participant Service Qantas Participant Service Marriott Participant Service Hertz Participant

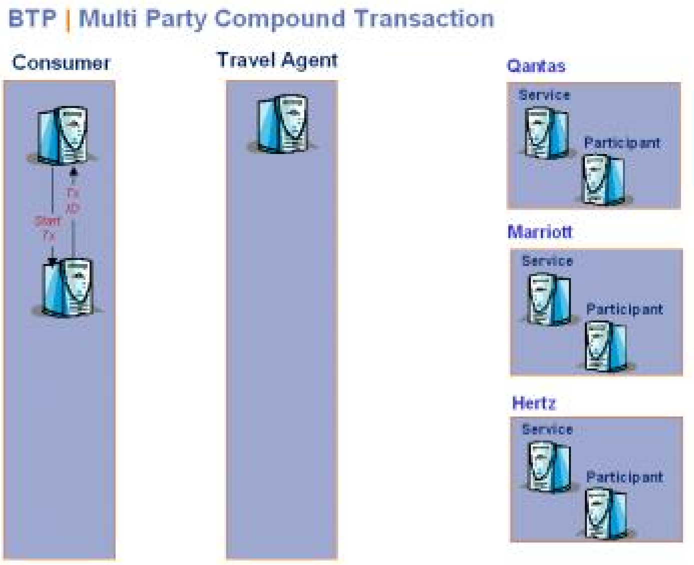
Stage 6: At this stage parties involved in the transaction have made various agreements (*Enrolled*) and commitments (*Prepared*) to the overall business transaction. UAL has neither agreed to participate nor made any commitments; BA has made a commitment based on an explicit request to do so (*Prepare*), Qantas autonomously made a commitment, as did Marriott and Hertz (*Prepared*). Based on the Travel Agents instruction to book the flight and the hotel, the Coordinator needs to confirm with these two parties the purchase of the ticket (with BA) and confirm the room reservation (with Marriott). The Coordinator also has to cancel parties that have made commitments that are no longer required (Hertz and Qantas). The Coordinator confirms the successful conclusion of the business transaction back to the Travel Agent (*Transaction Confirmed*). UAL does not need to be contacted as they did not even agree to participate and	 BTP | Multi Service Type Cohesion Travel Agent Application Composer Confirm Confirmed Cancel Cancelled Confirm Confirmed Cancel Cancelled Service UAL Participant Service BA Participant Service Qantas Participant Service Marriott Participant Service Hertz Participant

are no longer required to complete the business transaction.	

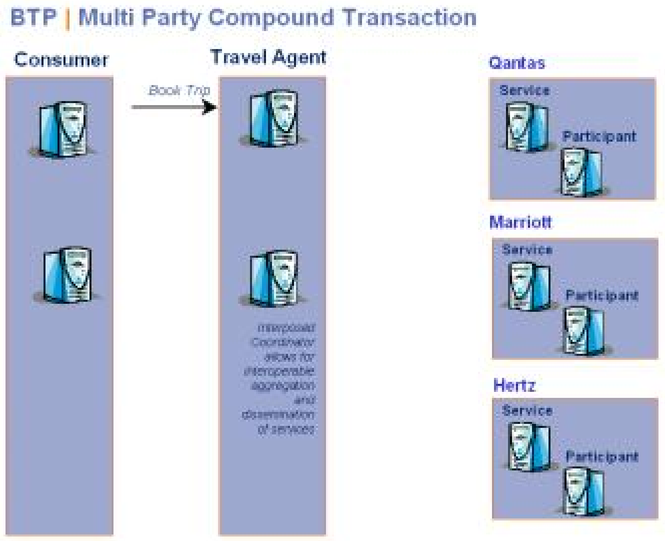
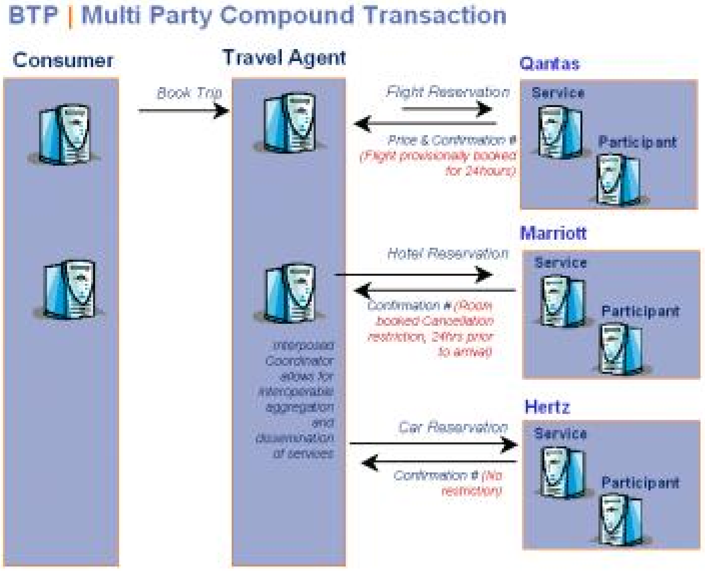
Example 5—Multi Party Compound Transaction

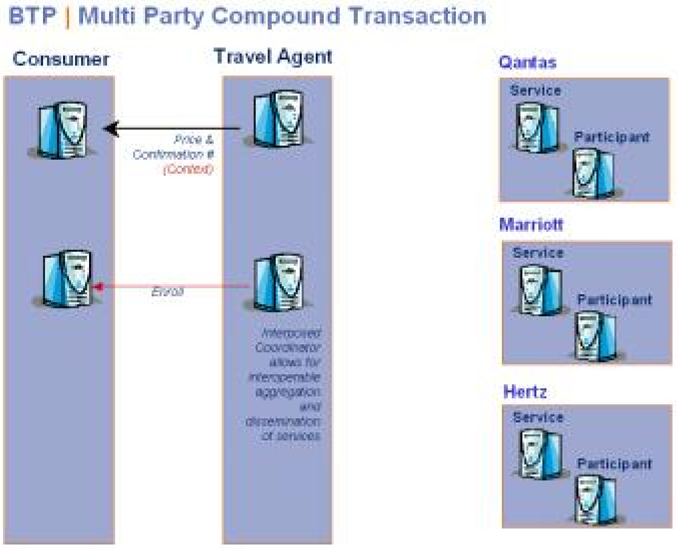
In this example the Travel agent itself offers a service for other consumers to utilise for holiday and trip planning. In effect, the travel agent is an intermediary acting on behalf of the consumer and the consumer never interacts directly with Qantas, Marriott, or Hertz. The same type of interactions we have seen in the previous examples can be compounded or nested whether they are Atomic or Cohesive (in this simple example all portions of the transaction are Atomic) to support more complicated scenarios such as supply chain and intermediaries. This example most closely reflects the real world of the travel agent.

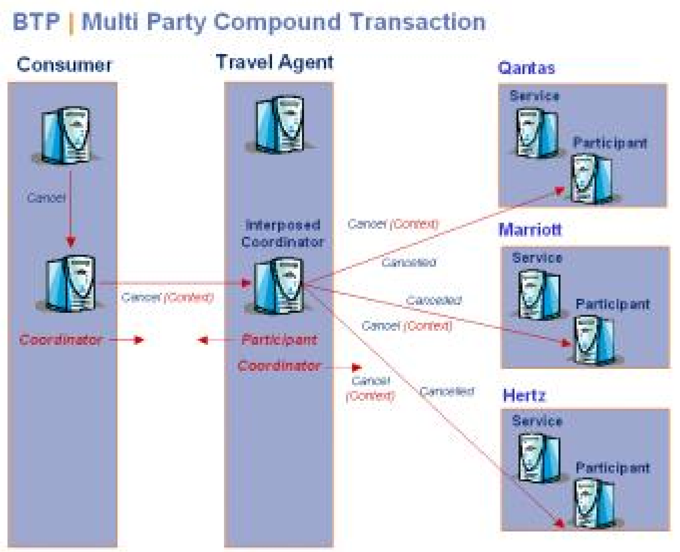
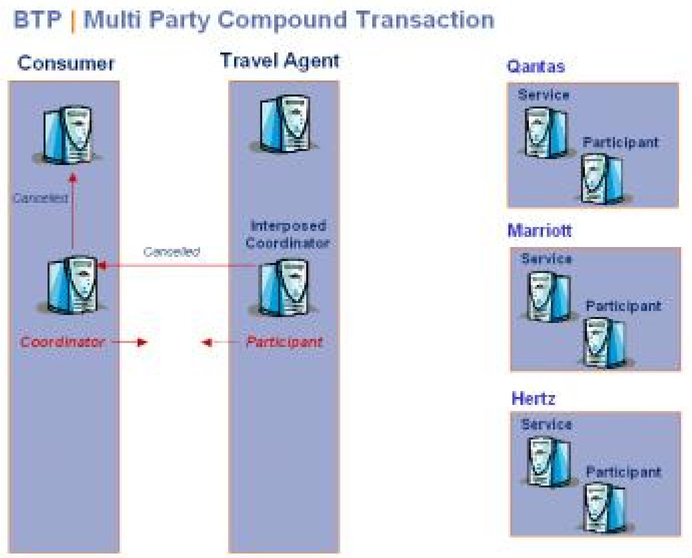


Stage 1: Firstly the Consumer (*Initiator*) creates a business transaction (*Context*) for the work it want to accomplish. It does this through a Coordinator (*Factory/Composer/Decider*).	 BTP | Multi Party Compound Transaction Consumer Travel Agent Start Tx Tx XID Qantas Service Participant Marriott Service Participant Hertz Service Participant

Stage 2: The Consumer (*Initiator*) then makes the Service Requests to the Travel Agent, also propagating the transaction details (*Context*). The Travel Agent now undertakes the task of creating and managing the sub-transactions that make up the overall business transaction.	

	 BTP | Multi Party Compound Transaction Consumer Travel Agent Book Trip Interposed Coordinator allows for interoperable aggregation and dissemination of services Qantas Service Participant Marriott Service Participant Hertz Service Participant
Stage 3: The Travel Agent makes Service Requests to Qantas, Marriott and Hertz based on the Consumer's request. Qantas, Marriott and Hertz (*Participants*) all agree to participate in the transaction (*Enroll*) and confirm that with the Travel Agent. Once all the parties have agreed to participate in the transaction the Travel Agent (*Participant/Coordinator*) itself can agree to be part of the transaction initiated by the consumer (*Initiator*).	 BTP | Multi Party Compound Transaction Consumer Travel Agent Book Trip Flight Reservation Price & Confirmation # (Flight provisionally booked for 24hours) Hotel Reservation Confirmation # (Room booked Cancellation restriction, 24hrs prior to arrival) Car Reservation Confirmation # (No restriction) Interposed Coordinator allows for interoperable aggregation and dissemination of services Qantas Service Participant Marriott Service Participant Hertz Service Participant

Stage 4: In Stage 3 all parties also made a commitment to the Travel Agent with regard to the transaction (*Prepared*). The Travel Agent could also make commitments (*Prepared*) to the Consumer when it agrees to participate in the transaction (*Enroll*). In this case however the Travel Agent simply replies to the Consumer and agrees to participate in the transaction (*Enroll*).	 BTP | Multi Party Compound Transaction Consumer Travel Agent Price & Confirmation # (Context) Enroll Interposed Coordinator allows for interoperable aggregation and dissemination of services Qantas Service Participant Marriott Service Participant Hertz Service Participant
Stage 5: The Consumer (Initiator) now can decide to make the booking, or cancel, depending upon the information returned by the Travel Agent. The consumer (*Initiator*) in this case decides to not purchase the trip offered (*Cancel*). The Coordinator now asks Travel Agent	

<table>
<tr>
<td>(<i>Participant</i>) to cancel the booking. The Travel Agent (<i>Coordinator)</i>, who has already received commitments from the parties (<i>Participants</i>) it is transacting with now must cancel with Qantas, Marriott, and Hertz.</td>
<td>
</td>
</tr>
<tr>
<td><b>Stage 6:</b> Once the Travel Agent (<i>Coordinator</i>) has received confirmation of the requests to cancel from all parties (<i>Participants</i>) it can confirm the cancel operation with the Coordinator, and the Coordinator in turn can confirm the cancellation with the Consumer.</td>
<td>
</td>
</tr>
</table>

Reading the Specification

The BTP specification and Model [BTP Model] more carefully and formally define the terms we are using here. The BTP Specification describes many more potential topologies of actors, along with interaction diagrams, selected state transition diagrams, and detailed state tables.

Questions and Answers

In this section, we answer some common questions about BTP.

- *Is BTP a Web Services protocol? Where does it fit with Web Services?* BTP defines XML messages that can be exchanged over many carriers, including the SOAP/HTTP combination that typifies Web Services. A binding to SOAP 1.1 over HTTP 1.1 is provided in the specification. Providing transaction coordination for Web Services is one of the requirements for BTP.
- *I've read about conversations for web services. Does BTP implement conversations for pairs of parties? For multiple parties?* BTP doesn't implement conversations; binary conversations use a conversation identifier to keep track of the respective parties. BTP can be used to aggregate binary conversations into multiparty conversations. In a similar manner, binary agreements such as RosettaNet PIPs or ebXML Collaboration Profile Agreements could be built into multiparty collaborations.
- *Does BTP attempt to do ACID transactions across the Internet?* No. BTP allows the implementation of coordinated actions. It is not feasible to do full ACID transactions between autonomous parties across

the Internet—the locking constraints would tie up one party's resources without giving them the control necessary to protect their business interests.
- *BTP does 2-phase commit, but it's not ACID. Why?* BTP uses a 2-phase interaction between the service requestor and the service provider to ensure that participants and coordinators agree on the outcome of the transaction. BTP neither requires nor expects the rigid locking of resources that is needed for ACID guarantees.
- *Does BTP do 2-phase locking? I understand that that doesn't work well over the Internet.* BTP does not do 2-phase locking. It is a 2-phase protocol, as there is a preparation and completion phase, but locking is not required for participants—they are autonomous, and locking of another entity's resources is simply not acceptable in the BTP environment and model.

• *How is BTP similar to traditional transaction processing protocols?* BTP is similar to traditional transaction protocols in that it defines a consistent termination for a unit of work, called a transaction. BTP has improved applicability to loosely coupled distributed transactions and environments with the required weakening of "tradition" transactional guarantees (e.g. ACID).

- *Is BTP a workflow language?* BTP does not contain a workflow language. BTP can provide reliable outcomes for workflow systems for inter-related activities. It enables more sophisticated workflows than might otherwise be possible, including reliable aggregation of multiple steps into a single unit of work.
- *Is BTP a business modelling language? Is BTP a Business Process Language?* BTP is neither a business modeling language nor a business process language. As with workflow implementation, BTP can be used to ensure reliable outcomes.
- *How much does an application writer need to know about BTP?* Applications that call BTP enabled services are required to interact with the BTP actors to initiate the communication (via the Initiator) and to bring the transaction to a close via the Terminator. Cohesion use requires further input and interaction from the controlling application to determine how the result set is formed.

Applications that provide a BTP enabled service need to call a BTP Participant to enrol in a transaction. These service side applications need to implement some form of commit and cancel operations. Whether these are compensation based or something else is up to the application writer.

Application developers who have been creating *ad hoc* termination protocols involving compensation will find their work simplified significantly. While the nature of compensation actions requires significant expertise in the domain (such as financial services or workflow management), the developers' job in creating termination and reconciliation protocols is much easier.

References

[BTP Model]

Section 1, OASIS Business Transaction Protocol, Committee Specification 1.0. Download from http://www.oasis-open.org/business-transaction/

[BTP Specification]

OASIS Business Transaction Protocol, Committee Specification 1.0. Download from http://www.oasis-open.org/business-transaction/