
OASIS WSIA Technical Committee

**Requirements Document
Use Case Report: Orchestrated Producers**

Version <1.0>

Requirements Document	Version: <1.0>
Use Case Report: <Use-Case Name>	Date: <dd/mmm/yy>
<document identifier>	

Revision History

Date	Version	Description	Author
05/Mar/2002	1.0	Orchestrated Producers	Dan Gisolfi, Graeme Riddell, Alan Kropp, Eilon Reshef, Gil Tayar, Rex Brooks, Ravi Konuru, Keven Brinkley, Aditi Karandikar, Monica Martin, Rich Thompson, Charlie Wiecha
14/Mar/2002	1.1	Added Basic Description and Flows, draft	Ravi Konuru

Requirements Document	Version: <1.0>
Use Case Report: <Use-Case Name>	Date: <dd/mmm/yy>
<document identifier>	

Table of Contents

1.	Definition of the Orchestrated Producers use case	3
1.1	Brief Description	3
2.	Actors	2
3.	Flow of Events	2
3.1	Basic Flow	2
3.2	Alternative Flows	3
4.	Diagrams	3
4.1	Relationship between Producers and Consumers in the Orchestrated Producers Use Case	3
4.2	< First special requirement >	4
5.	PreConditions	4
5.1	< Precondition One >	4
6.	PostConditions	4
6.1	< Postcondition One >	4

Use Case Report: Orchestrated Producers

1. Definition of the Orchestrated Producers use case

Definition: Use cases in which the Consumer uses information returned either from Producers it controls, or from interaction with the End-user, to alter the flow of one or more Producers in order to create a unified End-user experience among them. Flow may be altered by skipping selected Producer pages, by inserting Consumer pages within a Producer's flow, or by creating combined pages from multiple Producer's pages.

1.1 Brief Description

The Orchestrated Producer use case captures the scenarios in which the Consumer is able to control the presentation experience of the end-user by selectively showing or altering the page flow of a producer as seen by the End-user under programmatic control. [RK: this means that suppressing the presentation is considered orchestration. Comments?]

The Orchestrated Producer use case extends the Coordinated use case by providing control over coarse-grain (across-page) navigation of a producer flow.

Note that in this use case, the Consumer does not republish itself as a web service -- this step is left to the Republished use case. Thus the consumer executes as a potentially platform-specific application such as a portal server or a J2EE or .Net application.

The following are the examples from the submitted scenarios:

[Traveler's checks] Instead of displaying the Travelers Checks standard "Payment" page, display the bank's Payment page. In this case, the bank is a Consumer that wants to offer payment via bank accounts.

Requirements Document	Version: <1.0>
Use Case Report: <Use-Case Name>	Date: <dd/mmm/yy>
<document identifier>	

[Mortgage center] Add new questions asked, depending on answers given to the original (Mortgage) questions -- in this use case on different pages.

[Financial charting] Skip the Producer's dialog box for specifying symbol to be charted, and provide a Consumer-specified symbol.

[Smart Buyer] Using information input by the End-user the Consumer inserts new "Smart Advisor" pages into the flow that may result in the End-user using a different provider than the one they originally started with.

2. Actors

There are three actors in this use case:

- Producer: one or more WSIA web services
- Consumer: a platform-specific container which instantiates and controls interaction with the Producers on behalf of End-Users
- End-User: a person who interacts directly with the output of the Consumer

3. Flow of Events

3.1 Basic Flow

- The End-User enters a URL pointing to the Consumer into a browser
- The Consumer in Figure 4.1 is instantiated by an application server, which creates a session on it for interaction with the End-User
- The Consumer instantiates WSIA proxies as shown in Figure 4.1 for each of the Producer services it wishes to include in the page to be returned to the End-User, and creates a session on each to represent interaction with the End-User. Note that instantiation of a proxy may not be required if the Consumer chooses to interact with Producers via a framework such as WSIF (Web Services Invocation Framework) which allows direct access to services without use of proxies.
- The Consumer obtains the definition of the context and flow information supported by each Producer. This information may be obtained statically from a directory service such as UDDI for the given Producer or dynamically by querying the Producer service itself.
- The Consumer initializes Producer services, passing the appropriate context information and optionally the kind of flow that it needs from each producer. The consumer may not request a specific flow in the case where there is sufficient information in the flow description for the consumer to perform alterations without interfering with the correct functioning of the producer.
- Producers respond with their output, along with any context and flow information changes that may occur as a side effect of the request made by the Consumer.

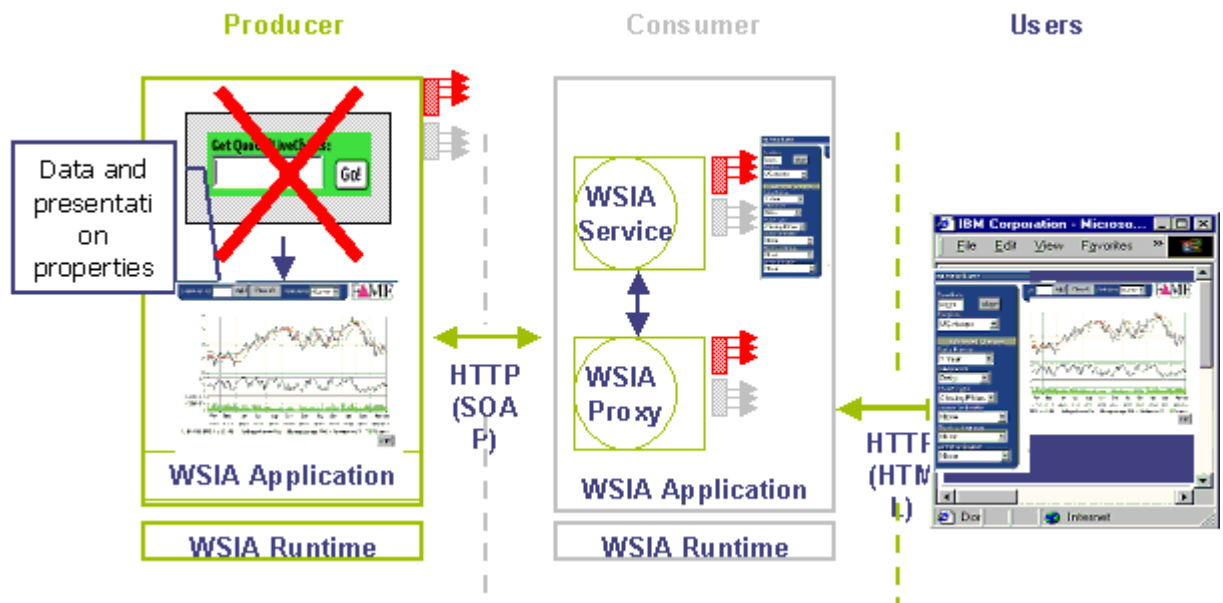
Requirements Document	Version: <1.0>
Use Case Report: <Use-Case Name>	Date: <dd/mmm/yy>
<document identifier>	

- The consumer may statically or dynamic determine the coordinated and orchestrated presentation flow that spans multiple providers. This is the consumer's presentation flow as seen by the End-user.
- For each step in the consumer's presentation flow, the consumer determines the producers that are involved in producing the consumer page and the coordination, if any that needs to be performed.
- The consumer requests the output from the providers providing any coordination information that providers need to customize their output. In addition, the consumer may also customize the actual output from the providers to achieve coordination. Further, it is also possible that the output of the provider is completely suppressed from reaching the End-user.
- End-User interactions with the Consumer's page are directed back to the Consumer. At the Consumer, rewritten URLs are decoded to find the type and the data associated with the request. The consumer uses this information to determine the next consumer page and consequently the set of providers, the provider pages and the adaptations that will be performed on them. The Consumer invokes the relevant Producer components passing any arguments which may or not be directly related to the request data obtained from the End-user. The consumer may look into, modify, or add to the arguments being passed to producer to dynamically customize the request to one or more provider services.

3.2 Alternative Flows

4. Diagrams

4.1 Relationship between Producers and Consumers in the Orchestrated Producers Use Case



Requirements Document	Version: <1.0>
Use Case Report: <Use-Case Name>	Date: <dd/mmm/yy>
<document identifier>	

4.2 < First special requirement >

5. PreConditions

[A precondition (of a use case) is a textual description of any constraints or dependencies that must be satisfied prior to entry of the use case.]

5.1 < Precondition One >

6. PostConditions

[A postcondition (of a use case) is a textual description of any constraints or dependencies that must be satisfied after termination of the use case.]

6.1 < Postcondition One >