Questions & Answers about ebMS V3

Q: Why the need for ebMS V3, after ebMS V2?

Since ebMS V2 was published as a standard (2002), the technology environment has evolved.

New versions of underlying specifications (e.g. SOAP 1.2) are gaining support, and Web Service-based platforms are now natively supporting features that had to be specified from scratch in ebMS V2 (reliability, security). Also, the experience obtained from V2 deployments leads to a better understanding of some requirements, in particular for small users.

Q: What are significant new features in V3?

- Message Processing Modes: parameters for representing and sharing properties affecting the transfer of messages (level of security / reliability, message exchange patterns, predefined headers etc.)
- Message Pulling. This allows for a light endpoint (no static IP address, or intermittent connectivity) to pull a message over the response of an HTTP connection.
- Message Channels. This increases the control of which messages should be transferred and when.

Q: Doesn't V3 overlap with Web Services technologies?

V3 reuses existing WS standards at protocol level (for reliability, security), but remains independent from the use of WSDL, as the service model may not be suitable for all endpoints.

Web service standards for security (e.g. WS-Security) and reliability (WS-Reliability and WS-ReliableMessaging) are just QoS capabilities that need to be integrated in a coherent business messaging protocol that supports loose coupling of partners. V3 defines this messaging protocol with its additional SOAP extension, that includes standard business metadata and message transfer controls (e.g. message pulling).

V3 is a significant step toward the convergence of ebXML with Web Services protocols, while preserving MOM-style messaging often required in loosely coupled environments.

Q: What are conformance profiles, and why are they not included in the core V3 specification?

The core V3 specification is designed along the lines of, "If you use feature X or underlying specification Y, here is how it must be done." (X could be some level of reliability, Y could be HTTP, SMTP, SOAP 1.1 or 1.2....)

A conformance profile (CP) narrows down which X and Y features or binding must be supported by a class of implementations intended for a messaging style. A few CPs have been defined in an adjunct document separate from the standard, as CP definitions may be updated/extended over time, reflecting user practices.

Q: How about transitioning from V2 to V3? What is the level of compatibility?

Backward compatibility from V3 to V2 could not be preserved mostly because of compliance with a new technology base. However, the V3 specification defines a mapping between V3 and V2 features.

Also, a conformance profile has been defined that shows how to support both V3 and V2. Increasingly, multi-version support is treated at product architecture level rather than specification level (e.g. most SOAP 1.2 stacks today also support SOAP 1.1 in spite of 1.2 not being compatible with 1.1).

Q: Where will features such as routing, status requests, message bundling, be addressed?

These and other features are being considered by the technical committee as part of a future Part 2 to the core specification. The core has been designed to provide the basic functionality required to reliably transfer messages and be a base for future development as required by the user community.