PARASOFT®
We make software work.™

PARASOFT®

# BPEL Maestro®

DATA SHEET

## Orchestrate Web Services with BPEL Maestro®

To help enterprises implement their Web service-enabled business processes, Parasoft developed BPEL Maestro, an integration platform that provides a production-grade BPEL runtime engine and a BPEL toolkit. Parasoft's BPEL Maestro enables flexible, interoperable integration applications to be implemented and deployed effectively and robustly by providing a platform upon which the vision of a service-oriented architecture can be realized today.

Parasoft BPEL Maestro features full support for WS-BPEL 2.0. In addition, Parasoft BPEL Maestro was designed with minimal proprietary extensions, making it very simple to integrate into your application (OEM) or environment.

## BPEL Maestro® Runtime Engine

BPEL processes need to be executed by a runtime engine that supports a number of requirements. The BPEL Maestro engine provides the abstractions and infrastructure particularly suited to Service-Oriented Architectures such as native support for Web service standards, as well as asynchronous messaging, process persistence, and transactional models consistent with long-term processes.

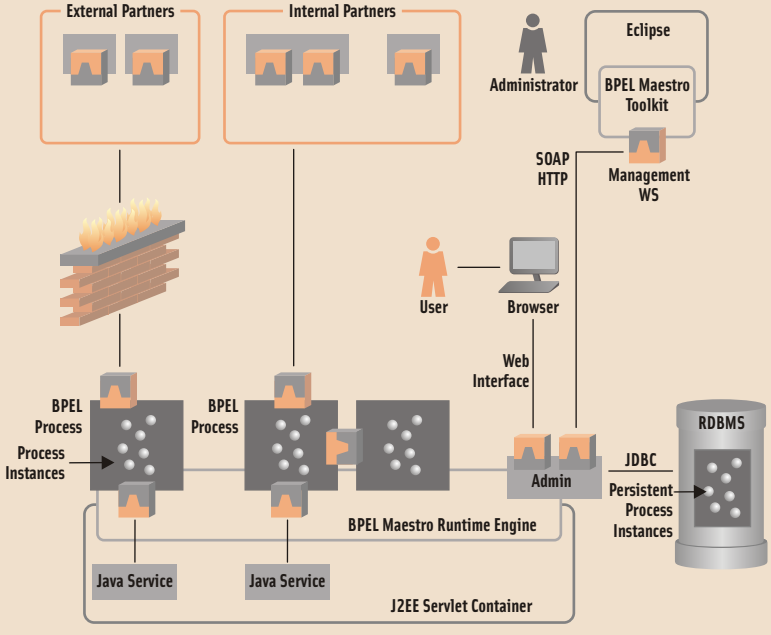
## WSDL-Centric View of Web Services

BPEL Maestro implements a WSDL-centric view of Web services and provides for a flexible, standards-based approach to application integration by supporting Web services standards such as WSDL, SOAP, XML, HTTP, JMS, WS-Addressing, and WS-I Basic Profile. Without conformance to such standards, a combination of monolithic homegrown solutions and vendor lock-in to proprietary solutions is inevitable.

The defining technical characteristic of a service for BPEL Maestro is that it be described in a WSDL. Every message exchange is described in a BPEL process in terms of portTypes and operations defined in the WSDL. Therefore, because the BPEL Maestro runtime engine is WSDL-centric and supports Web service standards, more flexibility is available on the messaging layer.

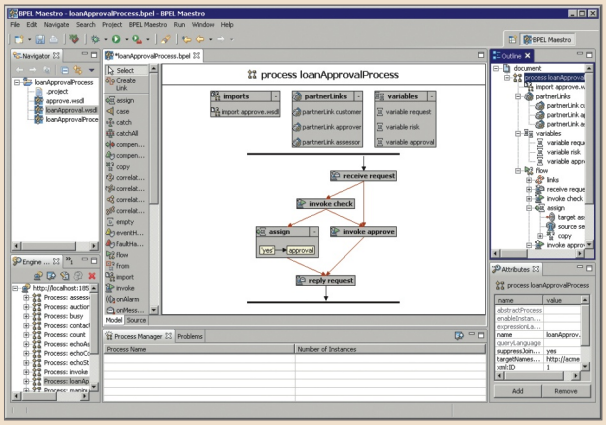
## Stateful Conversations

In order to properly and successfully orchestrate Web services, stateless modules must be composed into functional and stateful interactions. For simple business processes, stateless services are adequate. However, for more complex processes to receive multiple messages, they must be able to remember the salient points from the beginning of the conversation instead of needing to start from scratch when it receives another message.




*The runtime engine contains multiple process instances. Process instances participate in extended conversations with partners through Web services protocols and persist in a RDBMS.*

DATA SHEET




*BPEL Maestro provides an integrated, user-friendly environment for BPEL process authoring.*

## Features

- Provides complete support for WS-BPEL 2.0
- Supports Web service standards, including BPEL, WSDL, SOAP, XML, HTTP, JMS, and WS-Addressing
- Detailed, data-minable audit trails
- Built-in process versioning support
- Historical and predictive analytics capabilities
- Correlates synchronous and asynchronous message exchanges in extended, stateful conversations
- Native support for BPEL standard
- Integrated, full-lifecycle BPEL product that facilitates development, deployment, testing, debugging, and monitoring
- Transactional model supporting both short-term atomic transactions and long-running business activities with compensations
- Conforms to WS-I Basic Profile to facilitate interoperability
- Provides process persistence in RDBMS access through JDBC
- Scalability through automatic dehydration and rehydration

## Benefits

- Increases productivity and ROI by automating business operations that previously required labor and paperwork
- Improves customer service by streamlining transactions and making organizational services available 24/7
- Minimizes deployment costs by leveraging the existing infrastructure and Web services
- Provides abstractions and infrastructure particularly suited to Service-Oriented Architectures
- Enables flexible, interoperable integration applications to be implemented and deployed effectively and robustly

## Platforms

- Windows 2000/XP
- Linux
- Solaris

The BPEL Maestro runtime engine correlates synchronous and asynchronous message exchanges in extended stateful conversations and enables services to remember enough context to continue conversations as extended threads rather than as separate interactions.

## Long-running Processes

An important question arises in implementing any Web service process, namely how long is the process intended to run? This seemingly innocuous question turns out to have major ramifications for architectural requirements, including the type of messaging used, the need for process persistence, and the transactional model. BPEL Maestro provides the features needed to meet these requirements.

**Asynchronous messaging.**
BPEL Maestro supports asynchronous messaging in two ways: JMS-based enterprise messaging systems for reliable asynchronous messaging, and SOAP over HTTP with callbacks via WS-Addressing for lightweight asynchronous solutions.

**Process persistence.**
BPEL Maestro provides persisting process state in a standard RDBMS, accessed through JDBC.

**Scalability through dehydration.**
BPEL Maestro provides the infrastructure to effectively manage how long-running processes are brought into and out of memory by automatically performing dehydration and rehydration for scalability.

## BPEL Maestro® Toolkit

The BPEL Maestro eclipse-based toolkit helps you quickly develop, review, and update BPEL process files and related WSDL and XML Schema documents. Maestro's editor provides three views at differing levels of granularity. The highest-level view is a graphically-based activity diagram that represents the overall process flow; this view provides a coherent global visualization based on UML activity diagrams. In the middle level is a tree structure that reflects the DOM (Document Object Model) of the BPEL process. At the lowest level is a syntax-highlighted text editor with browsing capabilities. The toolkit also includes remote management capabilities, such as deploying and undeploying processes in runtime engines and exchanging messages with deployed BPEL processes. Finally, the toolkit provides a full-featured debugger for debugging either local or remote BPEL processes.

**www.parasoft.com**

**Contact info:**
Parasoft Corporation, 101 E. Huntington Dr., 2nd Flr., Monrovia, CA 91016
Ph: (888)305.0041, Fax: (626)256.6884, Email: info@parasoft.com