

## Enterprise Service Bus (ESB) Through BizTalk in Royal Cyber Inc.

The term Enterprise Service Bus is widely used in the context of implementing an infrastructure for enabling Service-Oriented Architecture (SOA). However, real-world experience with the deployment of SOA solutions has demonstrated that an ESB is only one of many components required to build a comprehensive Service-Oriented Infrastructure (SOI).

### Features of ESB

- An Enterprise Service Bus (ESB) is an architectural pattern and a key enabler in implementing the infrastructure for a service-oriented architecture (SOA)
- ESB is only one of many components required to build a comprehensive service-oriented infrastructure (SOI)
- The increasing adoption of SOA and the creation of Web services have revealed an ever growing need to provide a managed layer between services and their consumers
- An ESB provides support for interaction between heterogeneous services and interfaces that might be mismatched, or that might change over time
- An ESB addresses integration problems in a way that maximizes the re-use of services and maintains flexibility

### Introducing the Microsoft BizTalk ESB Toolkit

The BizTalk ESB Toolkit extends the functionality of Microsoft BizTalk Server 2009 to provide a range of new capabilities focused on building robust, connected, service-oriented applications. The BizTalk ESB Toolkit treats BizTalk Server components as individual units of work that can be connected, as desired; to form loosely coupled solutions and dynamic messaging architecture. It functions as middleware that provides tools for rapid mediation between services and their consumers.

The two most valuable features of the toolkit are:

- The Itinerary capabilities which allows you to implement a simple chain of actions on a message path (these actions can be running BizTalk maps, BizTalk Orchestration, or custom .Net code).
- The Management portal sample which allows you to view suspended messages and control other aspects of BizTalk and the ESB

The BizTalk ESB Toolkit provides key building blocks that are required for implementing a comprehensive SOI including:

- **Endpoint run-time discovery and virtualization** - The service consumer does not need to be aware of the service provider location and endpoint details; a new or modified service provider can be added to the ESB, without interruptions to the service consumer.
- **Loosely coupled service composition** - The service provider and service consumer do not need to be aware of service interaction style.
- **Dynamic message transformation and translation** - The mapping definition between distinct message structure and semantics is resolved at run time.
- **Dynamic routing** - Run-time content-based, itinerary-based or context-based message routing.

- **Centralized exception management** - Exception management framework, services, and infrastructure elements that make it possible to create, repair, resubmit, and compensate fault messages that service consumers or BizTalk components submit.
- **Protocol transformation** - Providing the ability for service provider and service consumer to interact via different protocols including WS-\* standards for Web Services. For example, a service provider can send an HTTP Web Service request, which will result in sending a message via Message Queuing.
- **Extensibility** - Provides multiple extensibility points to extend functionality for endpoint discovery, message routing, and additional BizTalk Server adapters for run time and design time.

### **ESB Toolkit Implementation Scope in Client's Project**

In Client project, covered setup standard ESB Toolkit 2.0 exception handling process and Create a build and deploy script. Further covered areas are:

1. ESB Exception Fundamentals
  - BizTalk ESB Toolkit Exception Handling
  - ESB Exception Management
  - Unified Exception Management
2. Capturing Exceptions
  - Routing Messaging Exceptions
  - Routing Orchestration Exceptions
  - Capturing Orchestration Exceptions
  - Extending the Exception Capture Capabilities
3. Extracting and Normalizing Exceptions
  - BizTalk Failed Messaging Normalization
  - Modifying Orchestration Failure Routing
  - The Fault Message Reporting Schema
  - Using the ESB Exception Handling API
  - Capturing Exceptions Using the API
4. Reporting ESB Exceptions
  - The ESB Exception Reporting Process
  - Implementing the ESB Fault Processor Pipeline
5. Using the ESB Management Portal
  - Features of the ESB Management Portal Sample
  - Repairing and Resubmitting Failed Messages
  - Configuring Exception Alerts
  - Automating Exception Processing



© Copyright IBM Corporation 2010  
IBM Global Services  
Route 100  
Somers, NY 10589  
U.S.A.  
Produced in the United States of America  
08-10  
All Rights Reserved

IBM, the IBM logo, [ibm.com](http://ibm.com), Lotus®, Rational®, Tivoli®, DB2® and WebSphere® are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at [ibm.com/legal/copytrade.shtml](http://ibm.com/legal/copytrade.shtml) Other company, product and service names may be trademarks or service marks of others. The information contained in this documentation is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this documentation, it is provided "as is" without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this documentation or any other documentation. Nothing contained in this documentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM (or its suppliers or licensors), or altering the terms and conditions of the applicable license agreement governing the use of IBM software. This document illustrates how one organization uses IBM products. Many factors have contributed to the results and benefits described; IBM does not guarantee comparable results elsewhere.