



ESB for SharePoint

White Paper

April 2010

motion10
10

Microsoft®
GOLD CERTIFIED

Partner



Index

Introduction	3
SharePoint as universal user interface	3
The role of Business Connectivity Services	4
Pitfalls.....	Fout! Bladwijzer niet gedefinieerd.
ESB for SharePoint.....	4
BizTalk Server ESB.....	5
A practical example.....	Fout! Bladwijzer niet gedefinieerd.
The road to success.....	Fout! Bladwijzer niet gedefinieerd.
Conclusions.....	12



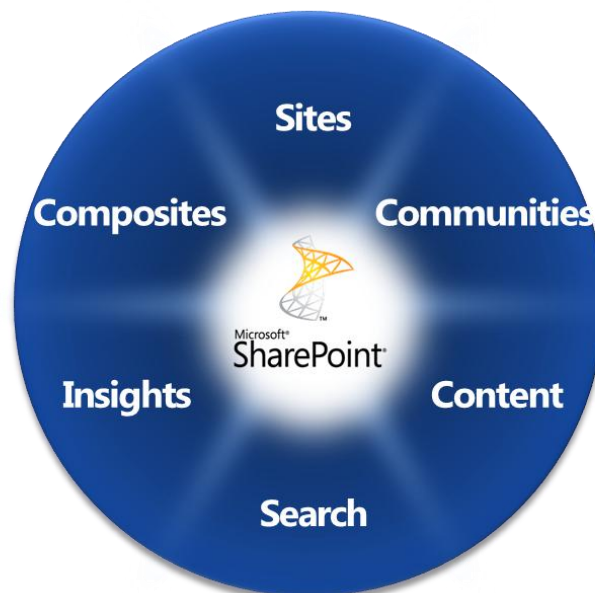
Introduction

SharePoint is one of the most successful Microsoft server products. Most people know this product primarily as an environment for collaboration and document sharing. Many people will also know it to be a platform to build websites, while others see it as a portal. The fact is that SharePoint provides a central location for people to work from. It brings together all required user functionality.

This white paper examines the combination of SharePoint and the Microsoft Enterprise Service Bus (ESB) BizTalk Server, in order to achieve a manageable overall solution for universal access to all relevant information and business processes.

SharePoint as universal user interface

As mentioned, SharePoint is the central place where everything comes together. Below you see the new 2010 chart with the components that constitute SharePoint:



This white paper focuses in particular on the 'Composites' piece of the chart. 'Composites' is the part where SharePoint provides the capabilities to join and present all sorts of information from backend systems with related documents and information from the web. Thus 'Composites' constitutes a central place where people do their work. Composites (or Composite Applications) actually is SharePoint's term for (enterprise) mash-up.





The role of Business Connectivity Services

BCS (Business Connectivity Services) is the technology that is used to access data in SharePoint 2010. It deals with data that comes directly from the backend systems' database or through web services. BCS is the successor to BDC (Business Data Catalog). Whereas BDC was limited in terms of real two-way traffic between users and backend system, BCS has become a mature technology. You can exchange data with an SQL Server, a Web service (using WCF) or with .Net types. It is possible to develop access to this external data source with just SharePoint Designer. This way, a portal page can be created which display data from different sources.

Pitfalls

SharePoint's success is also often the cause of a proliferation of many one-on-one connections with backend systems, databases and web services. The result is often something we call 'spaghetti integration'. The availability of these sources cannot be guaranteed and the clarity, consistency and accuracy of the compiled information can become increasingly uncertain.

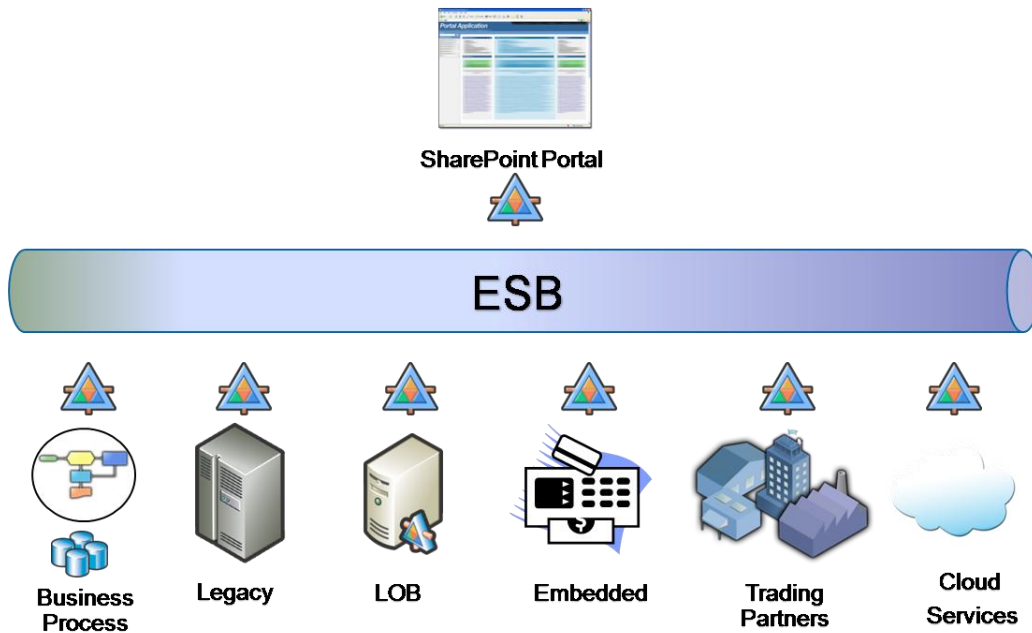
Upgrading a backend system to a new version, or worse, replacing such a system by another, can have serious consequences for all these direct interfaces. It turns into a management nightmare. Also, there is almost no insight into the transactions flowing between all the backend systems and the SharePoint frontend.

Furthermore, it makes little sense to leave the composition of application services within complex scenarios to the frontend. Often there is quite a bit of business logic involved, which should not be resolved in frontend code.

ESB for SharePoint

The Enterprise Service Bus (ESB) offers a solution to this problem. Microsoft's ESB - BizTalk Server - is perfect for creating an abstraction layer between the applications (frontend and backend). Through 'loose' couplings, flexible routing, format and protocol conversion and proactive exception handling, any SharePoint environment can be turned into a manageable entity once again.





Business processes can be modeled within the ESB, which can combine the various backend systems' functionalities and is able to publish them as composite services. The business process that runs in the ESB is responsible for orchestrating the interaction with all interfaces and presenting this process as a single transaction that can be rolled back if necessary.

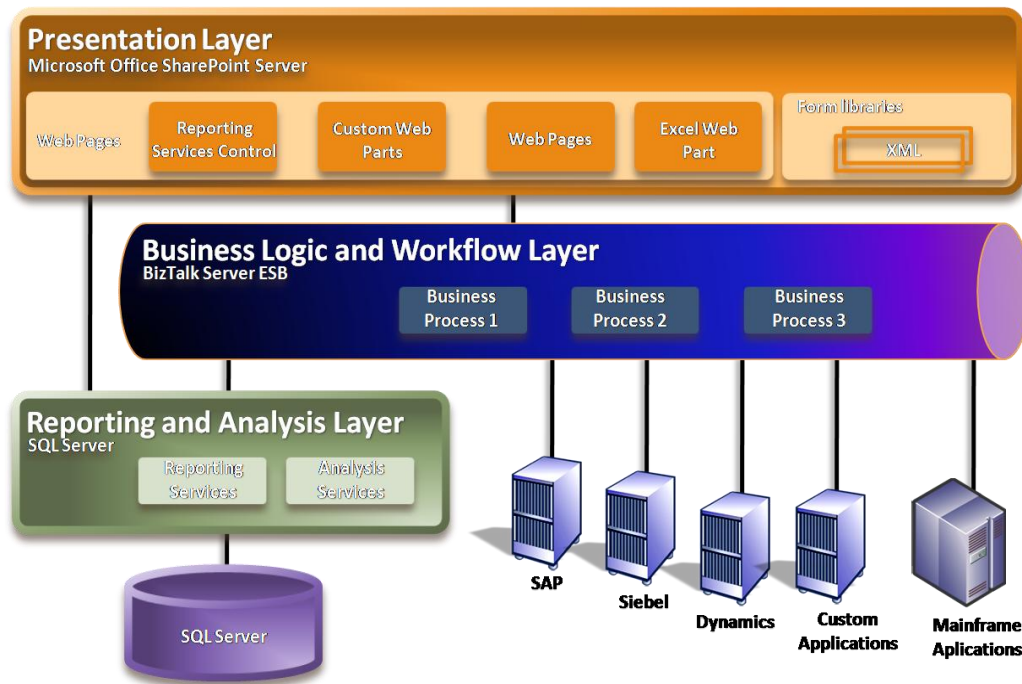
Also, SharePoint's workflow capabilities can be invoked from this orchestration within the ESB. This allows the user to be involved within the automated processes at points where human interaction is required, such as when data deviates from standard business rules, or when the manual input of information is required.

The BizTalk Server ESB itself is an integration platform that ensures high availability and scalability, thanks to its built-in cluster architecture and grouping capabilities.

BizTalk Server ESB

BizTalk Server as an integration server is a key part of Microsoft's application platform. It is the heart of .Net technology-based Service Oriented Architectures (SOA) and is often used as the Enterprise Service Bus. The ESB is the technology you need in order to be able to implement a SOA within a company. SOA is the architectural style; ESB is the 'plumbing'.





What exactly is an ESB? The following sections illustrate this by a brief explanation of the ESB's main components.

Loose coupling

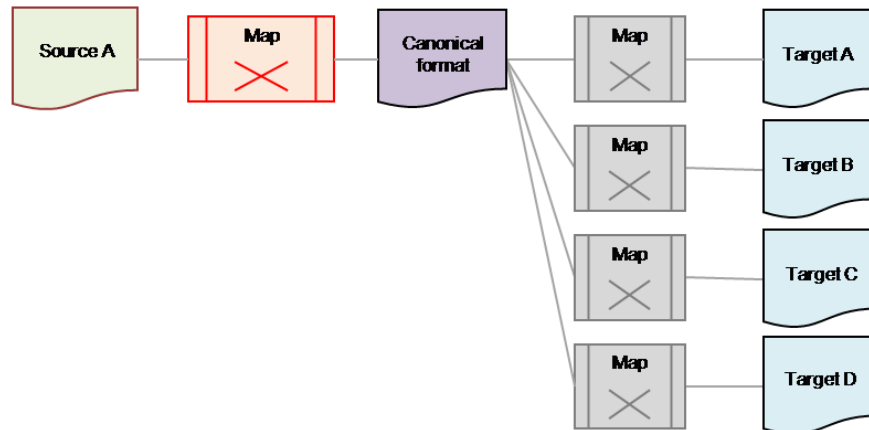
Loose coupling means that there is no direct link to application interfaces through hard coding and / or invocation of APIs, but only through the exchange of messages. Nowadays this usually means XML messaging (mainly web services), but we often see links between applications, based on the exchange of the familiar 'flat' files or other file formats. BizTalk Server can be seen as a multitude of file formats and exchanges, all of which are described by XML schemas with possible extensions to the schemas, in order to be able to handle non-XML formats. The XML schemas are part of BizTalk solutions developed using Microsoft Visual Studio.

Transformation

Transformation is needed because information from one system can almost never be processed directly by another. The message's structure and content is converted from the source to the target. Developers of BizTalk solutions can use the BizTalk Mapper for this purpose. This is a graphical tool integrated into Visual Studio, allowing transformations to be developed by drawing connecting lines and using an extensive library of predefined 'functoids'. The resulting logic map is then compiled to XSLT, which performs the transformations at runtime.

Canonical data model / abstraction layer

By introducing an intermediate format - the so-called canonical data model - an additional abstraction layer can be added which makes it easier to handle changes from backend and frontend applications.



This format also allows for the separation of application domains. Someone who knows frontend application development only needs to take the schema's mapping into account, which describes the composite service to the user interface. For example, the mapping between the SAP backend system to the canonical format can be designed and created by an SAP integration specialist. The introduction of an intermediate format is almost always a good idea (for more information on this topic, see motion10 white paper "Using canonical formats with BizTalk Server").

Dynamic routing

Routing information to the appropriate target systems may dynamically occur within the ESB. For each message that arrives in the ESB, a decision is made about what should be done with it. Think of the transformation to a canonical format, enrichment with address data from CRM systems, performing orchestration X, etc. The actions that take place are described in the so-called Itinerary (also known as the 'routing slip'). So if something needs to change in the sequence of implemented actions, for example, it is just a matter of configuration. Once again, it is not necessary to redeploy the solution.

Composite services

Because the ESB is often used to implement complex business processes which are modeled in, for example, Microsoft Visio, it is ideally suited to present composite services (services which in turn are composed of several other services) to the SharePoint frontend. A request to display an order's status in the portal can be the reason for the ESB to consult different





systems, to consequently show the order's financial and logistical handling status. Since all transactions are handled by the ESB, monitoring and managing of the integration layer can be performed in this central location.

Business Rules Engine

The business rules of an organization should be stored in a central location. This is where decisions are taken, for example to start a workflow for obtaining approval from a department head or manager to process a transaction. Or to determine a discount rate for incoming orders and payment deadlines. Enforcing these rules can be done from this central location, since all transactions are processed by the ESB, during which multiple backend systems can be involved. Defining these rules is a job for information analysts and definitely not for developers.

Exception handling

Since the syntactic and semantic validation of transactions, as well as the rules' applicability is all done within the ESB, this is also the obvious place to set up error handling. The BizTalk ESB makes it possible to view all relevant information about data that has been processed or is waiting to be processed, from one centralized dashboard within the SharePoint environment. Here, it is also possible to correct transactions with reported errors and subsequently resubmit them. This BizTalk ESB management portal is a standard feature and is very user friendly for technical and functional support staff. It offers the capability to proactively and timely be informed about (potential) problems.



Business Activity Monitoring

Monitoring business activity goes a step beyond pure technical and performance monitoring. The ESB can be configured so that during the running and processing of transactions, key business information such as part numbers, quantities, prices, delivery locations, can be tracked and collected. Subsequently, a clear dashboard can be provided, using SQL Server Analysis Services and Reporting Services in the SharePoint Portal. This information enables the business user to make better decisions. This functionality belongs more to business intelligence (BI) and within the SharePoint environment it can be found under the 'Insights' pie of the chart. This combination of technologies enables businesses to discover trends in order flows or to obtain an overview of all orders received for specific items in a specific region during the past week.

A practical example

While reviewing existing processes or designing new functionality, it often becomes clear that there is a need for certain information from different systems to be aggregated, or for information to be modified in several applications simultaneously.

An example of a real-life scenario is a system that allows customers to track shipping containers. The ESB not only controls the interaction with the underlying distributed container terminals, but also retrieves the location and type of each container. This set of application services is published as a composite service by the ESB. The WSDL (Web Services Description Language) which describes this service can be used by SharePoint Designer to configure BCS and to map the data to the user interface controls.

The end user can enter a container number in the SharePoint portal. The ESB then retrieves all relevant information from various legacy systems and returns everything as one result. The container's location is shown in the SharePoint portal, using Bing Maps. Information about the container type is retrieved as a document from within SharePoint and displayed. This way, all information is brought together. The container's status can then be updated or modified, after which this transaction is processed by the ESB to the underlying systems.

All transactions processed by the ESB are monitored and visualized in a SharePoint dashboard, where error handling is redirected to the right support staff. The SharePoint portal is used by both end users and by the support staff who manage the integration environment.



The road to success

Deploying an ESB within an application landscape is an important decision that is right at the heart of the architecture and should therefore be done in a structured way. The motion10 'ESB Roadmap' is based on 10 years of experience in hundreds of BizTalk Server implementations and contains the following main steps:

1. Organize a presentation to demonstrate the possibilities, in order to learn what the solution exactly entails. The people that will be involved in the next steps should attend this, so they will become fully aware of the possibilities and impossibilities.
2. Architecture Design Session (ADS). During a two-day session with the architect(s) and analyst(s) (up to 5-7 fully dedicated people), the following items are determined:
 - a) Scope
 - b) Use case(s)
 - c) High level architecture
 - d) Information flows
 - e) Security requirements
 - f) Management requirements
 - g) Potential risks
 - h) Selection of best practice(s)
 - i) Timeframe
3. Deliver ADS documentation and determine the next step(s).
4. The next step could be a proof of concept (POC) or pilot. Or both.
5. Proof of Concept (POC). A POC must demonstrate that the chosen solution could work well in a production environment. The results of the POC are generally discarded afterwards. During a five-day session, one or more use cases are developed on the basis of the following phases:
 - a) Designing the POC environment
 - b) Developing the use case(s)
 - c) Testing the results
 - d) Presenting and reporting the results
6. Pilot. A Pilot is done to demonstrate that the chosen direction for the solution works and is meant to be taken into actual production. The requirements for a pilot are thus much stricter than those for a POC. Security and exception handling are some of the key aspects of the pilot which should receive full attention. The length depends on the complexity of the project.

During the ADS session, there will be an examination together with the architects and information analysts on which use cases offer the highest ROI (Return on Investment) or which ones have the lowest risks. In other words: which ones offer the highest probability for success in the short term. Sometimes the case with the highest visibility for the business is chosen. It may be that an existing flow is chosen to be replaced by a solution in which the ESB plays a central role, or the decision can be made to create a new flow, with the ESB as a central component in the overall solution. The starting point will usually be the selected use



cases which are at that time (without ESB) difficult to implement (new flows) or which currently suffer from management issues due to the lack of insight or correct tooling (replacing existing flows).

It is very important that the right people are involved during the ADS phase: Both the people who understand the frontend (user interface) side of the solution, as well as specialists in backend integration. A central role is reserved for integration specialists who know all the ins and outs of the ESB and who are able to select and apply the proper integration patterns to the selected use cases.

Ideally, the results of the POC or the pilot will be a useful use case which is able to demonstrate that the combination of SharePoint as a universal user interface and the ESB as an integration layer between SharePoint and the various backend systems, provides an overall solution. This means saving costs on designing new automated business processes or efficiently replacing existing processes. If a pilot was chosen, on completion the system can be put into operation immediately.



Conclusion

SharePoint can be used with great success as a centralized access point for business processes and information. SharePoint also offers a universal user interface that can be deployed companywide.

By using the BizTalk ESB to function as an integration layer beneath SharePoint - which serves as a universal user interface - a highly manageable environment can be created. The spaghetti of one-on-one links with various backend systems can be avoided. Additional advantages in using an ESB are its transactionality, the ability to publish and consume composite services plus the high availability and flexibility of the overall platform.

The deployment of an ESB in a SharePoint environment is a project that must be handled with great care; using the right kind of professionals is crucial. They must determine the best possible architecture. Integration specialists who have a great deal of experience with the ESB should be used. By following a roadmap based on best practices, success can be guaranteed.

By combining SharePoint with the BizTalk Server ESB, it is possible to make the most of the Microsoft Application Platform .

On the author and motion10



Gijsbert 'Gijs' in 't Veld (gijs.intveld@motion10.com) is CTO for business integration specialist motion10. Since 2001, he has been involved in a great number of BizTalk Server implementations worldwide. Gijs is 5 times Microsoft MVP.

motion10 is specialized in integrating digital information from multiple sources. By integrating information, the company enables easy access and helps to extract more value from the data. As a result of this optimal integration, business processes connect better to one another, processes run better, their management becomes easier and costs will be reduced. Thanks to in-depth experience with sophisticated integrations (hundreds of BizTalk Server implementations) motion10 is able to resolve the complexity of IT environments. Based on Microsoft Application Platform technology - BizTalk, SharePoint and SQL Server (BI) – motion10 help organizations to accomplish a robust, future-proof IT environment.

For more information: www.motion10.com.

