E-Commerce Servers and Back-End Integration: A Case-Based Comparison of Commercial Wholesaling and Retailing Software

By Niels Chr. Juul
&
Claudia Loebbecke

December 1999, no. 13

ISSN 1399-1779 99/13

E-Commerce Servers and Back-End Integration: A Case-Based Comparison of Commercial Wholesaling and Retailing Software

Niels Christian Juul
Copenhagen Business School
Center for Electronic Commerce
Howitzvej 60, DK-2000 Frederiksberg, Denmark
Tel. +45 3815 3789
Fax: +45 3815 2401
E-mail: NCJuul@cbs.dk

and

Claudia Loebbecke
KRAK Chair of Electronic Commerce
Copenhagen Business School
Howitzvej 60, DK-2000 Frederiksberg, Denmark
Tel. +45 3815 2455
Fax: +45 3815 2401
E-mail: Loebbecke@cbs.dk

December 1999

Abstract

With the growing importance of E-Commerce, commercially available E-Commerce servers are increasingly gaining ground. Our research investigates two of the most commonly implemented products, IBM's Net.Commerce (NC) and Microsoft's Site Server Commerce (SSC) Edition. After a short description of the two systems, this paper introduces the case of the shoe wholesaler ROHDE, for which prototypes of both systems have been implemented and investigated. Although our case is business-to-business, the findings are also applicable to the business-to-consumer segment. The ROHDE specific comparison investigates the implementation appropriateness and feasibility of both systems in the light of ROHDE's business requirements. It focuses on the integration of the servers with the installed back-end system. Our findings are summarized and directions for further research are given in the concluding section.

Contents

1. Introduction

Although E-Commerce is much more than Web-commerce, E-Commerce solutions based on Internet technology are growing fast [1, 2, 3]. Diffusion of the technology at the large requires usable building blocks, hence commercially available E-Commerce systems are increasingly gaining ground. Our research covers an E-Commerce application between a wholesaler and a set of retailers or retailer chains enabling retailers to view products and prices, and to place orders. It investigates two of the most commonly implemented products, IBM's Net.Commerce and Microsoft's Site Server Commerce Edition, and provides a comparison of available technologies and functionalities of both products.

A selected prototype implementation illustrates issues regarding the integration of E-Commerce servers and legacy systems. The specific set up in the case company for which prototypes on both systems were implemented requires the legacy systems to continue to be operational after introduction of the E-Commerce systems. The legacy systems are running on an IBM AS/400 server, and both E-Commerce systems run on Microsoft Windows NT Server 4.0 (see Figure 1). Ideally, the E-Commerce systems would run on top of the existing databases integrating smoothly into the existing systems, but in both products a separate database for the E-Commerce system was needed as shown.

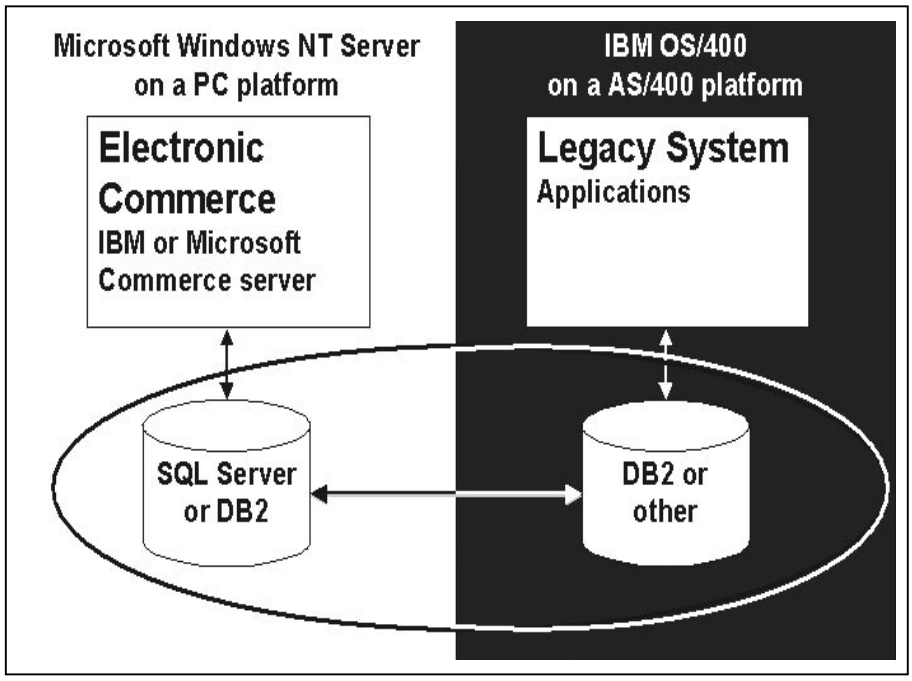

Fig. 1. Architecture of Wholesale IT Solution

2. E-Commerce Systems Overview

This section provides a short introduction of the two systems, which have been implemented during the course of the research, the IBM Net.Commerce Server [4, 5] and the Microsoft Site Server Commerce Edition [6].

2.1 IBM Net.Commerce (NC)

NC is a set of integrated software components that provide a solution for businesses to sell goods and services through an electronic catalog on the Internet. The system is appropriate for both business-to-business and business-to-consumer applications and claims to be capable of integrating with a company's existing business systems. The product comes in a 'START' and a 'PRO' version. The START version is for medium-size or large companies that want to quickly set up an E-Commerce site for a low entry price. The PRO version is for more advanced customers aiming to build a Web site which requires additional functionality, such as advanced back-end integration and advanced cataloging. This paper focuses on the START version.

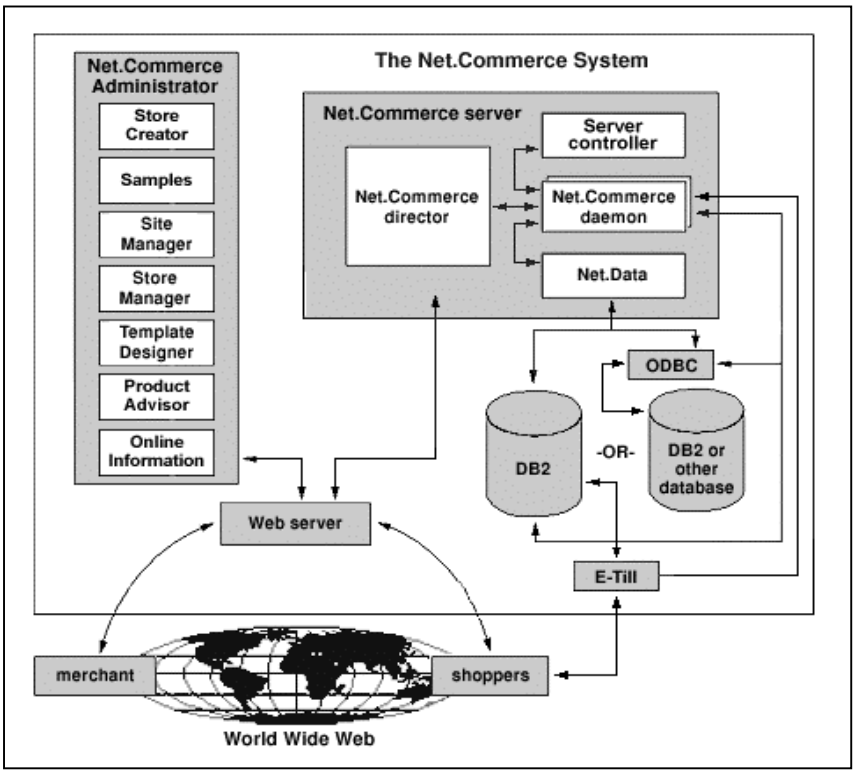

Fig. 2. The IBM Net.Commerce Architecture (Source: IBM)

NC consists of a large number of functional modules (see Figure 2). The *payment server* ensures that payment always takes place over a secure line. There is support for several standards such as SET (Secure Electronic Transaction) [7] and SSL (Secure Socket Layer) [8, 9]. The *Mass Import Utility* can be used to load Net.Commerce product and category tables directly from files generated by the legacy systems. IBM also promises that one can use database tools to synchronize the product database directly from the legacy systems. The *Net.Commerce Administrator* allows managing one's site. It permits creating several stores and provides different roles for the different user categories. It includes modules for 'store creator', 'site manager', 'store manager', 'template designer', 'product advisor' (only in the PRO version), and 'online information'. The *Advanced Catalog Tools* (only in the PRO version) can be used to create intelligent catalogs adapting to the various shopping styles and buying behaviors of customers. These catalogs can provide quick and easy search methods as well as help buyers who may need extra guidance in making product selection. The *Advanced Back-end Integration* (only in the PRO version) includes tools for back-end integration to selected systems like SAP/R3, IBM MQ Series, IBM CICS, and IMS, as well as interface to IBM Global Network UN/EDIFACT converter and infrastructure.

IBM NC is built in layers. The first or bottom layer is a generic infrastructure called Net.Server, which contains the foundation classes for the system. This layer provides services, which are typical for a web-database based application server. It forms an environment for components to be plugged in for handling web requests. Three kinds of components are 'commands', 'overridable functions (OFs)', and 'pages'.

The second layer is the Net.Commerce layer offering a commerce specific object model and a collection of commerce related commands, overridable functions and pages. The Net.Commerce Server is the heart of the IBM NC system. It controls the flow of information that is displayed to and provided by shoppers. The server also controls the flow of information that is accessed by merchant employees when they use the IBM NC Administrator. The server contains two components, the 'daemon' and the 'director'. The IBM *NC server daemon* displays store pages dynamically by executing macros that retrieve data from the IBM NC database. The daemon maintains a continuous connection with the database. It controls shopper registration and ensures that SSL is enabled when required to protect any confidential information. The IBM *NC server director* is a common gateway interface (CGI)

program that enables two-way communication between the Web server and the daemon. The director accesses the store's database through a TCP/IP-connection with the daemon.

2.2 Microsoft Site Server Commerce (SSC)

SSC is an Internet Server, optimized for Microsoft Windows NT operating system with Internet Information Server (IIS), that enables businesses to engage and transact with customers and partners online. It promises secure and scalable order capture, management, and routing while integrating more easily into existing inventory, accounting, and EDI systems. Microsoft SSC further more provides tools to help business managers analyze customer and partner usage data and factor analysis into changes.

SSC is targeted towards companies that want to sell their products and services via E-Commerce. It provides components for publishing, such as content management, tag tool and content deployment. SSC offers an engine for searching information on web pages, office documents, public folders, and *open database connectivity* (ODBC) compliant databases. It provides management of memberships and personalized content on the site and push-technology through *active channel server* and *active channel multicast* for delivering push content. Furthermore, SSC contains a knowledge manager and delivers usage analysis, report writing and content analysis. In addition, the ad server allows the management and display of ads on sites in any MIME type file, e.g., images, text, sound, or animation. Finally, SSC contains the Site Builder Wizard, the Site Manager Pages, Commerce Server Objects, Order Processing Pipeline, Commerce Interchange, Microsoft Wallet and buy now.

SSC version 3.0 runs under the Microsoft Windows NT Server operating system on Intel-based and Alpha-based computers. Windows NT Server supports the security and performance requirements of SSC. Windows NT Server includes an integrated Web server, Microsoft Internet Information Server (IIS). IIS provides comprehensive, secure Web site administration for Internet sites running Commerce Server. In addition, IIS version 4.0 supplies the Active Server Pages (ASP) environment for developing Web-based applications and ActiveX™ Data Objects (ADO) version 1.5 for dynamic database access from Commerce Server sites (see Figure 3).

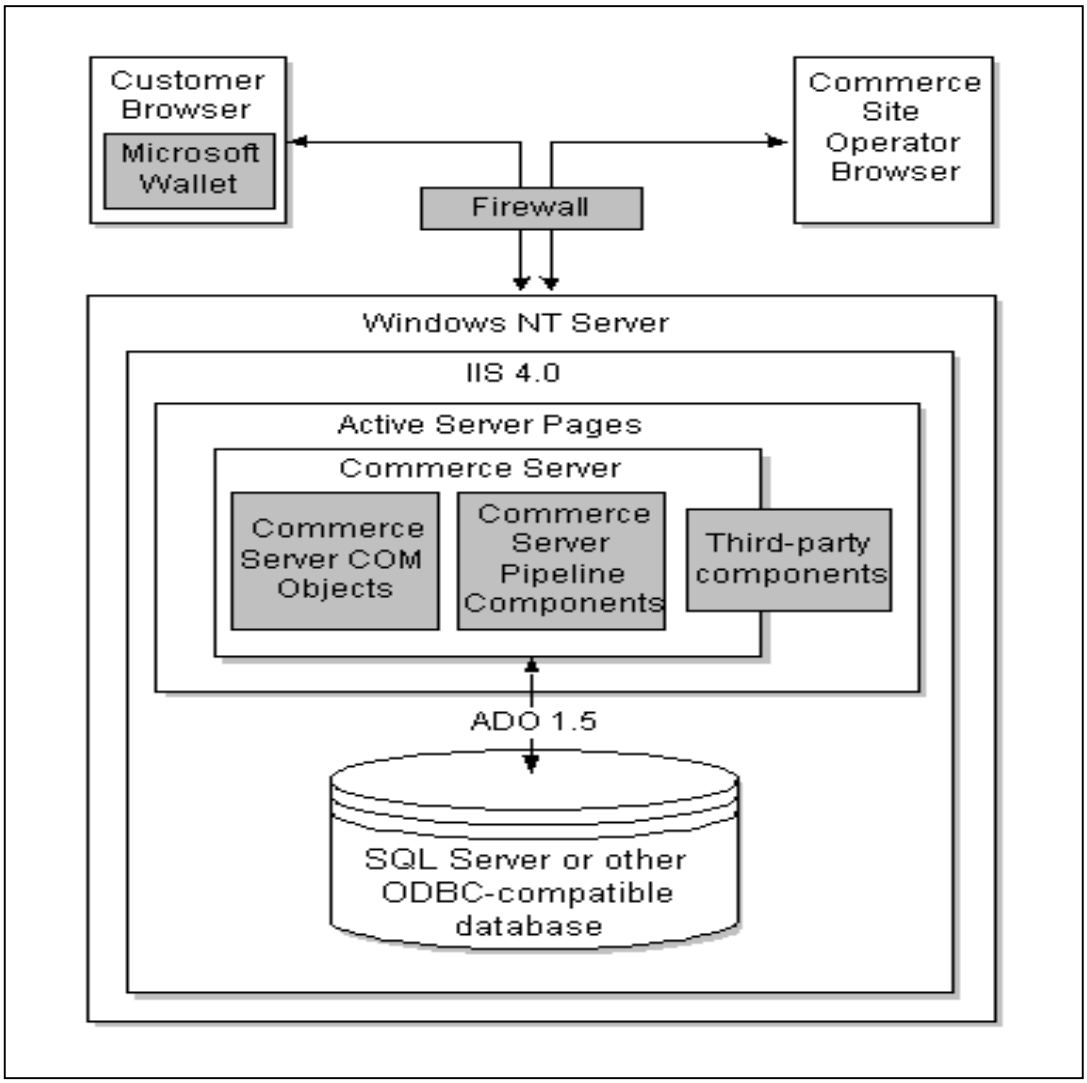

Fig. 3. The Microsoft Site Server Commerce Edition Architecture (Source: Microsoft)

SSC is an object oriented, component-based system. It consists of three types of components, 'Commerce Server COM Objects', 'Commerce Server Pipeline Components', and 'Third-party or in-house developed components'. These components are called from the Active Server Pages (ASP) or from other components. The components of the ASP pages can then access the database on SQL Server or other ODBC compatible DBMS through ActiveX Data Objects (ADO).

3. NC vs. SSC: Key Technological Characteristics

This section provides a brief comparison of selected E-Commerce technologies [10, 11, 12] used by both the IBM NC and Microsoft SSC. During our research and prototype building efforts, we have looked at:

- the *protocols* between both the servers and the shopper (Web) or the back-end systems incl. security protocols,
- *content management*, i.e. caching and push/pull issues,
- required and available *browser extensions*,
- *session identifications* throughout a transaction,
- *object-oriented techniques* deployed,
- server-side scripting and database templates, and
- *back-end integration* with existing databases.

3.1 Protocols

Web Protocols: The IBM NC system is based on the HTTP protocol when the shopper's client application (the web-browser) opens a connection to the default Web server (in this case a Lotus Domino Go Web Server). The default Web Server also supports HTTPS, which is a security-enhanced variation of HTTP. HTTPS allows servers and clients to authenticate each other and to define the kind of security used in transmissions. HTTPS is mainly used on the IBM NC manager site. Microsoft SSC also uses HTTP on port 80 for its default web communication, for components requiring secure data transmission it uses HTTPS. For mail, it uses SMTP and for news services it uses NNTP.

Security Protocols: IBM NC can make use of the SET (Secure Electronic Transaction) and the SSL (Secure Socket Layer) protocols. This is interesting when customers wish their payment to take place over a secure line. It is IBM NC's Payment Server Component that primarily makes use of these protocols, but also the login and registration pages of IBM NC as well as the Site Manager uses the SSL protocol by default. An Internet Store created by IBM NC has a digital certificate, which the SSL protocol provides to authenticate the IBM NC server to the shopper. Also SSL encrypts the data and requests between the server and the shopper. Microsoft SSC only supports SSL and this is done through the HTTPS protocol.

Additional protocols: Finally Microsoft SSC uses a set of additional protocols for database access and for its Membership components. For the membership component it uses a proprietary protocol on top of TCP/IP. For accessing databases, Microsoft SSC uses the ODBC protocol.

3.2 Content Management

The IBM NC system does not make use of the Push Content technology. When shoppers click on a link or type the URL for the Electronic Commerce Shop, they will receive the page on their computer. The sites created with Microsoft SSC do not include push content by default, but this can easily be configured using the Site Server Push component (*active channel server* and *active channel multicast*).

Besides using browser side caching, the IBM NC System allows for server side caching. Since HTML pages are not static, but created through Net.Data macros, a lot of time and processing power can be saved if already created pages are kept in a caching directory. By default the naming convention of these HTML pages is based on CategoryID/ProductID and StoreID. If a user requests a Product Category page of an Internet Store, which has already been provided to other shoppers, that page will not be generated through Net.Data macros again but just retrieved from the cache directory.

In general, caching provides faster response to the second and following accesses to the same data, which introduces the risk of providing old outdated data. This problem is quite relevant for the IBM NC caching system since products or templates can be updated. For this to work effectively, the IBM NC system has a Synchronization Daemon, which runs as a background process that synchronizes data used by the Web server with the database.

If a store uses different HTML templates to different Retailer Groups, the IBM NC caching utility supports a naming convention which includes the Shopper Group ID in the filename.

3.3 Browser Extensions

Though IBM argues that the Internet Store should be kept clean regarding Browser extensions (by default an IBM NC-Store only uses HTML and JavaScript on the Browser Side), there is no limitations regarding MIME types, Applets, Plug-ins and Controls. Any of these browser extensions can be included in the Net.Data macros that generate the HTML pages. Using browser extensions introduces dependencies on specific browsers and/or browser configurations as well as security risks to the shopper. On the other hand, HTML developers can take advantage of specific browser features and dynamically select between tailored HTML pages to

Fig. 4. Web page shown in Netscape Navigator (v.4.51) vs. Internet Explorer (v.5.0)

these browsers. I.e. one page could be exposed to Netscape users and another page could be exposed to Internet Explorer users (see differences on Figure 4); frames could be used with some browsers, frame-less pages can be used with others, etc.

Microsoft SSC takes advantage of both ActiveX controls and Java Applets. ActiveX controls are only used with Microsoft Wallet and only in browsers with ActiveX support such as Microsoft Internet Explorer or Netscape Navigator with an ActiveX browser extension. If this extension is not present, it provides standard HTML forms for these transactions instead.

Java Applets is used only in the web interface for administration of SSC and for the store manager site. Even though the ActiveX components used for Microsoft Wallet is platform dependent of a Windows Operating system the elective implementation combined with the Java Applets makes the solution platform independent for both administrators and customers. AxtiveX controls are software components referenced by web pages, which can be automatically downloaded and installed on the client's computer. The advantage of this step is that when the ActiveX is referenced in the future it has already been downloaded and is ready to run again (minimizes waiting time on slow connections).

One disadvantage is that the ActiveX controls have full access to the resources of the client system, which means that opposite the Java's sandbox model, ActiveX controls have full access to the user's computer. The download process of an ActiveX control is not activated all automatically, but the user is given the choice to trust the author of the ActiveX control before downloading. But here there is a potential danger; if you choose to 'always' trust the author (non experienced users tend to do this), then if you encounter a site where another ActiveX control comes from the same

author it activates the downloading and installation process itself without prompting the user.

Another disadvantage of ActiveX-components is that they contain binary code, which makes them platform dependent, which means that not all client software (i.e. browsers) can make use of ActiveX controls. Besides the workaround regarding the Microsoft Wallet, ActiveX controls all other pages is displayed equally in both browsers.

3.4 Session Identification

Opposite static stateless HTML pages, Electronic Commerce almost per definition requires a series of interactions with the customer. IBM NC by default uses SessionID's placed in Cookies to save information about which state information is associated with which user. The IBM NC SessionID is stored in the HTTP cookie given by the server to the browser on initial connection. If the user has accepted a cookie, the server can recognize the user and facilitate the login process. When users logs in, the server side generated SessionID is replaced by the users' login. IBM NC also uses hidden fields in forms, which are returned to the server on consequent requests.

Microsoft SSC uses no cookies by default; but cookies can be enabled if wanted. When cookies are not enabled the shopperID is passed as a Query string parameter in the URL.

3.5 Object-Oriented Techniques

IBM NC is using object technology through the overridable functions. Applications can be distributed among multiple computers. On the other hand, IBM is not using any of the well-known Object Technology standards like CORBA, COM objects or Java Beans/RMI [13].

Microsoft SSC takes great advantages of object technologies through the use of Component Object Model (COM) components. As mentioned before, Microsoft SSC distinguish between three different kinds of COM components; Commerce Server COM Objects, Commerce Server Pipeline Components and Third-party or in-house developed components. The Commerce Server COM objects is used for general functionally of Microsoft SSC, e.g. OrderForm object and DBStorage object. Putting

all database access into a single component can enable using technologies such as connection pooling to improve performance.

The Commerce Server Pipeline components are of more specific purpose and are used in the pipelines. The pipeline is a framework for connecting modular pipeline components together in a sequence, e.g. the purchase pipeline contains components for validating purchase information, payment and acceptance. Finally third-party or in-house developed components can be used as general purpose components or as pipeline components to provide specific functionality such as legacy system integration.

In addition, object-oriented design provides rapid application development through developing reusable components. It provides the ability to distribute components among servers to improve performance. Furthermore, component based solutions can take advantage of TP (Transaction Processing) monitors to ensure atomic consistent transactions as well as taking advantage of ORB (Object Request Brokers) in order to build scalable and robust applications [14].

3.6 Server-Side Scripting and Database Templates

IBM NC does not use plain Server-Side Scripting. Rather the Net.Data macros resemble a combination between Server-Side Scripting (JavaScript) and database-driven templates, where the SQL statements are embedded through special tags in the HTML code.

When shoppers click on a hypertext link or an HTML form button in IBM NC, it calls a Net.Commerce command to execute the macro, which is then loaded. The Web server accesses the macro through Net.Data as DLL. Net.Data expects two parameters: the name of the macro to process, and the HTML section to display. Input parameters are passed into the macro from HTML forms, command parameters, and other global variables, such as the merchant reference number. These can be used in the macro to control processing. Net.Data syntax controls macro flow, and one or more SQL statements are executed to retrieve data from the database. The search results are formatted as an HTML page and returned to the browser (IBM NC Online documentation). The Net.Data macro files are also referred to as templates.

Microsoft SSC uses server side scripting through ASP which supports embedded scripts such as JavaScript and VBScript.

Table I SUMMERY OF TECHNOLOGICAL ISSUES IN E-COMMERCE SERVERS

	Technological Characteristics	IBM NC	Microsoft SSC
A.	Protocols	HTTPS SET, SSL ODBC	HTTPS SSL ODBC, proprietary
B.	Content Management	Pull and Server-side Caching	Push and Pull
C.	Browser Extensions	HTML, JavaScript (+plug-ins)	HTML, (+ActiveX, Java Applets etc.)
D.	Session Indentification	Cookies	Cookies or Query strings
E.	Object-Oriented Techniques	Overridable functions	COM, CORBA
F.	Server-Side Scripting	net.data scripting	ASP (JavaScript, VBScript)
G.	Back-End Integration	Build additional DB, apply of 'E' and 'F'	Build additional DB, apply of 'E' and 'F'

3.7 Back-End Integration

Both, NC and SSC, demand a database within the E-Commerce system and another one within the existing legacy system, thus having problems maintaining consistency. They have to build their own database and mass import data from the legacy system. Hence both servers deploy the technical features of *scripting* and *object-oriented* component communication (see also Section 4).

3.8 Summary of Key Technological Characteristics

Table I summarize our brief comparison of the technological characteristics provided by NC and SSC. To further investigate the appropriateness and feasibility of both servers in a concrete setting, we proceed by presenting the RHODE case. Due to given page constraints, we will focus on 'G: Back-End Integration' as the most cumbersome and crucial issue in building a real-world application.

4. Case: E-Commerce Server Implementation at 'Rohde'

4.1 Case Setting

The wholesaler ROHDE is the Danish division of Erik Rohde Schuh Fabriken, a German shoe factory [15]. ROHDE has 10 full-time employees; it sells and distributes

shoes to retailers in Denmark, Sweden and Norway, with a focus on Denmark. ROHDE sells 10 to 20 thousands pairs of shoes per year. The company handles two types of orders, (1) *main orders*, which are placed before producing the collection of the season and (2) *supplement orders*, which are placed ad hoc during a season. ROHDE receives many supplement orders, often for small quantities, by phone, mail, fax and email. A peak of 200 phone calls per day is not extraordinary. The administration cost of supplement orders is significantly higher than the one of main orders, resulting in higher prices charged for shoes ordered during a season. Two employees are in charge of only quickly handling of supplement orders and thus to service the retailers. The higher prices on supplement orders results in retailers buying larger quantities via main orders, hence requiring them to keep higher stock for a longer time before actual sales take place (low turn over rate). On average pairs of shoes stays in a store for 180 days before they are sold. ROHDE aims at lowering the cost of administration of supplement orders by E-Commerce. It foresees electronic handling of supplement orders as a way to improve its competitiveness by being able to lower prices and to provide a better service to their retailers. Over time, this has led to the idea to extend the project to main orders and perhaps even to take orders from consumers, and hence to streamline their overall flow of orders.

ROHDE's *main requirements* to the E-Commerce Internet store can be summarized as follows:

- Administer products (shoes) with two attributes (size & color).
- Individually price products depending on specified intervals of sizes.
- Accept only retailers approved by ROHDE for shopping on the site.
- Accept only several buyers per retailers if approved by their manager.
- Check retailers' credit limits.
- Inform retailers when purchased products are out of stock.
- Import product data from the legacy system.

Currently, all orders are processed through ROHDE 's legacy system, which runs on an AS/400 computer on a mixture of DB2 database and plain files. The part of the legacy system using plain files was developed in the ‘70s. Unfortunately, ROHDE's

existing network bridging software does not allow import/export access to the actual legacy data. The problem originates from the file types used in the old part of the legacy system. In AS/400 there are two different types of files: externally described files and internally described ones. Nowadays, AS/400 applications use externally described files and the PC to AS/400 bridging software can only handle this file type. Therefore, the legacy data could not be bridged to PC files.

4.2 Approach to Further Investigation

With our exploratory prototype implementations, we have studied the installation, the configuration, and the store creation and modification using both server systems. Due to the given page constraint, this section is limited to the issue of 'Back-End Integration' (see II, G). To facilitate the actual E-Commerce server implementation and to investigate the problems of integration, the ROHDE legacy data has been transferred into a similar Microsoft Access Database while preserving the old legacy record structures. The Access Database mimics the back-end system for this comparison. The above emphasized need to create and maintain multiple databases when implementing NC or SSC has been analyzed in more detail along the following five sub-criteria [16, 17]:

- Quality of *data overview* referring to the structuring of data into tables, fields, and relations in the E-Commerce server database;
- Data access for *importing and exporting* data between different DBMS;
- Support for *data transformation* between legacy system and E-Commerce server;
- Ensured *data consistency* between databases, and
- Data *bridging* between different platforms (AS/400 and PC Windows NT).

4.3 Back-End Integration at RHODE with IBM NC

A standard installation of IBM NC uses a DB2 database and creates a database (called MSER by default) in the installation process. The MSER database consists of 237 tables of which 53 are DB2 System Tables. DB2 does not provide any functions

or tools to give an *overview* of the data structure by default. Neither does it provide a graphical user interface to amend the records in the database. IBM NC, on the other hand, provides a very detailed on-line documentation, in which the database structure for the most important tables (30-40 of the 184) is described. This description is given as both an overview and as detailed description. As the documentation is based on HTML pages, the illustrations actually work as 'clickable' maps, where the user can click on the tables in order to get a detailed description of the fields and their data types. The overview itself is split into subsections (i.e. the documentation does not try to show the relations between all 184 tables on one image). The documentation gives a thorough and detailed insight in the database structure, but one has to switch to the Control Center in DB2 every time sample data from the database is needed. By default IBM NC uses one database for all stores created. Hence, it becomes complex to get an overview of ownership and relations between products in the product table etc. Consequently, company product numbers cannot be used as primary keys (i.e. two stores can give different products the same product number). Instead IBM NC automatically generates a unique reference number internally. During store configuration both these numbers are used to identify the same products at different stages, which complicates the configuration for the user.

In the ROHDE case, the legacy system contains 106 products, 997 items, and 3,187 attributes for the Winter season of 1998. Automatically *importing* the data from the legacy system to the DB2 database is therefore necessary, as it would be too time-consuming to manually insert the records through the Store Manager. Since the data structure of the legacy system is different from the data structure of the Net.Commerce system, a *transformation* of the data has to take place. The Import utility in IBM NC imports a text file into the DB2 database and creates reference numbers and relations between records in different tables. The text file *must* have a certain structure. In order to automatically create the text file, we created specific queries to the Legacy System (in Access97) in order to extract data and format them in the structure needed for the input file. These queries have to be exported (each row in the text file represents one record in one or more tables), merged, and the parameter '#ROHDE' has to be added on the first line to allow the import utility to recognize for which store the products should be imported. Having the import file ready for import, the command then has to be typed manually. In the ROHDE case, the import utility ran for 12 minutes and 20 seconds on a Pentium II 400MHz with 128MB RAM in order

to import approximately 5,000 product related records into the DB2 database. The import utility then created reference numbers and references between categories, products, items, attributes, prices etc. Probably, this batch job could be more automated.

IBM NC by default requires either DB2 or Oracle v. 8.04 meaning that unless the company runs these DMBS, data importing/exporting is needed. If NC is build on top of one of these DBMS, all queries in NC has to be customized to access the existing database correctly. Thus, most users will need to export the database from their legacy systems. This raises another problem i.e. to keep the two database copies *consistent* [14]. The NC does not handle simultaneous updates of data by default. Such functionality could be implemented by adding extra OFs.

Because IBM NC is available on the AS/400 platform as well as Windows NT, AIX and Sun Solaris, there is no need to *bridge* data between different platforms. In case one wishes to run IBM NC on a different platform than the legacy system, one has to go through a combination of OFs, ODBC access and Bridging and/or Gateway Software (such as Microsoft SNA Server or IBM Communication Server).

4.4 Back-End Integration at ROHDE with Microsoft SSC

For getting an *overview* of the data structure MICROSOFT SSC provides two tools, the 'manager site' and the 'DBMS'. Through the manager site, some of ROHDE's products and attributes have been created manually in order to track how the specific product information is stored in the database. Microsoft SQL Server Enterprise Manager allows viewing the database tables and structure. SQL Server 7.0 has database diagrams, which automatically document the database structure. SSC creates one set of tables per store, even though several stores use the same database, hence making database management more flexible and reliable (databases can be placed on different servers).

SQL Server Data Transformation Services (DTS) can be used to transform the data from the legacy system to the SSC database. DTS provides a flexible way to run interdependent database actions such as *import/export*, *data transformation* etc. DTS supports virtually any DBMS, which have an ODBC or OLE DB driver, including Oracle and DB2. Transformations consist of a source table or SQL query, a destination table and transformation. The transformation may be a simple copy of the values or a program written in VBScript or JavaScript (See Figure 5 and 6).

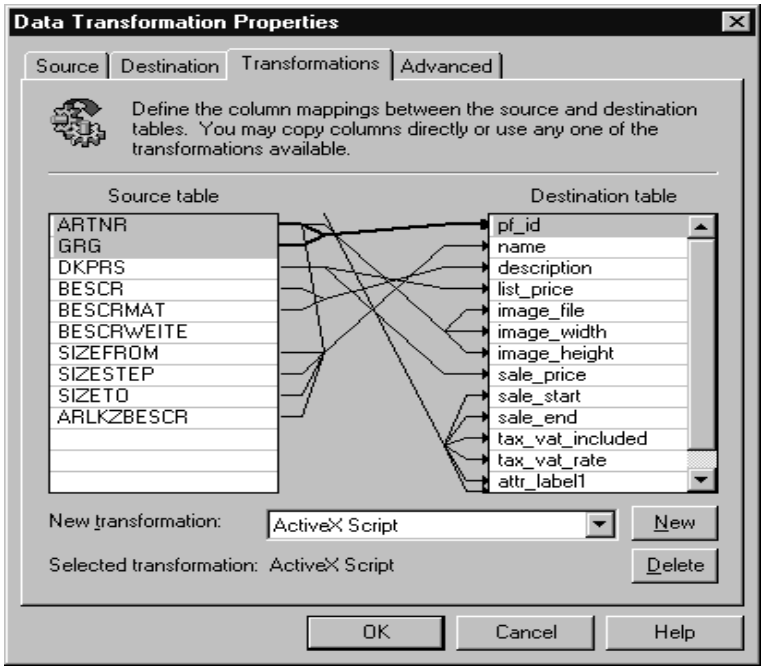

Fig. 5. Overview of data transformation for product import

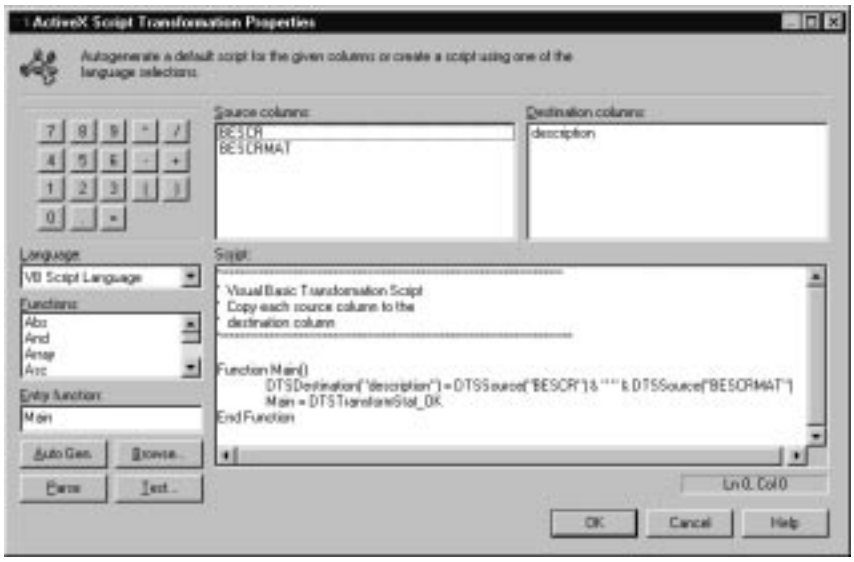

Fig. 6. VB Script transforming product description and material into a single description string.

The DTS handles both transformation and import/export between different DBMS, which has an ODBC or OLE DB driver present. Regarding simultaneous access to the legacy system, e.g. for *consistency* purpose, ROHDE requires that: (1) products are actually in stock, and (2) retailers' credit limit are not exceeded. SSC implements such simultaneous database access to the legacy system through adding components to the pipelines.

As ROHDE legacy systems run on an AS/400 and SSC runs on a PC Server with Windows NT Server operating system, *bridging* is required. Microsoft SNA

Server provides this bridging as well as ODBC drivers for accessing DB2 databases and files on AS/400.

4.5 Towards an Evaluation of NC versus SSC Back End Integration

Any evaluation of the two servers with regard to back-end integration is necessarily subjective and contingency dependent, even more so in the current state of research in progress. In order to better communicate our results with RHODE managers, we refined our five sub-criteria even further. Table II depicts a summary of our comparison and the calculated total score. SSC appears to be superior because of its ease of use and its flexibility of the solution. However, this comparison is heavily based on a comparison of DBMS (DB2 vs. SQL Server).

5. Conclusion and Outlook

It has become apparent that both NC and SSC offer a rich variety of tools and applications, which will increasingly reach the status of 'need to have' software. Based on our experiences with the two E-Commerce servers, we foresee a long and stony path until E-Commerce servers will reach the state of 'off-the shelf' products. Especially the necessary integration with back end systems is still rather tedious. There are still a lot of gaps with 'features that will be available in next version of the

TABLE II NC AND SSC EVALUATION OF BACK-END INTEGRATION CAPABILITIES

Criteria	**IBM NC**	**Microsoft SSC**
Database structure – ease of use	Low (1)	High (3)
Database structure – flexibility	High (3)	Medium (2)
Database structure – documentation	High (3)	Low (1)
Transformation – ease of use	Low (1)	Medium* (2)
Transformation – flexibility	Low (1)	High* (3)
Import/export – ease of use	Medium (2)	Medium* (2)
Import/export – flexibility	Low (1)	High* (3)
Bridging – ease of use	Low (1)	Medium* (2)
Bridging – flexibility	Low (1)	Medium* (2)
Ensuring consistency - ease of use	Medium (2)	High (3)
Ensuring consistency – flexibility	High (3)	High (3)
Documentation on legacy system integration issues	Medium (2)	Low (1)
Total	**21**	**27**

* Using Microsoft SQL Server which is only included as a trial version.

products'.

This research focussed on the back-end integration with a wholesaler's IT-systems. The back-end integration at the reseller's side will be another major challenge to next generation E-Commerce servers. Current solutions (including NC and SSC) simply take advantage of the 'nice and universal' interface between E-Commerce servers and human users provided by web browsers. Even in the light of XML and XML-EDI there is still a long way before the full power of EDI is reached when integrating E-Commerce systems with client IT-systems.

Ongoing research is expected to shed light on more efficient application developments. Among many trajectories, one to be followed to is whether adjustments / adaptations should be made in prototype development and data transformation. To what degree will and should the introduction of E-Commerce solutions in every day business life trigger a major restructuring of the overall legacy landscape?

6. Acknowledgement

The authors highly appreciate that ROHDE has provided time and restricted access to legacy resources. The underlying implementation on the two servers were done by M. Aydas, T. Bjerkan, P. Mertz, and A. Schou during our graduate course in E-Commerce Server Architectures, Spring 1999. IBM graciously provided our lab with the full IBM Net.Commerce server software and likewise our students had access to the Microsoft Site Server.

7. References

[1] G Winfield Treese & Lawrence C. Stewart. *Designing Systems for Internet Commerce.* Addison-Wesley, 1998.

[2] David Angell & Brent Heslop. *The Internet Business Companion.* Addison-Wesley, 1994.

[3] Ravi Kalakota and Andrew B. Whinston. *Electronic Commerce: A Manager's Guide.* Addison Wesley Longman, 1997.

[4] IBM Website about IBM Net.Commerce server product, http://www.software.ibm.com/commerce/net.commerce/

[5] Samantha Shurety. *e-business with Net.Commerce*. First Edition, Prentice Hall, Prof. Technical Reference, 1998.

[6] Microsoft Web site about MS Site Server Commerce Edition, www.microsoft.com/siteserver/commerce/

[7] SET Secure Electronic Transaction LLC Web site at URL: http://www.setco.org including the SET Secure Electronic Transaction Specification.

[8] Alan O. Freier, Philip Karlton, and Paul C. Kocher. *The SSL Protocol*, Version 3.0, Netscape Communications Corp, November 1996.<draft-freier-ssl-version3-02.txt>

[9] Netscape. *Secure Socket Layer*, SSL version 3. Web site available at: http://www.netscape.com/eng/ssl3/

[10] Andrew S. Tanenbaum. *Computer Network*s.Third Edition, Prentice-Hall, 1996.

[11] Simson Garfinkel with Gene Spafford. *Web Security & Commerce*. O'Reilly, 1997.

[12] Robert Orfali, Dan Harkey & Jeri Edwards. *The Essential Client/Server Survival Guide*. Second Edition, John Wiley & Sons, 1996.

[13] Robert Orfali & Dan Harkey. *Client/Server Programming with JAVA and CORBA*. Second Edition, John Wiley & Sons, 1998.

[14] J.D.Trygar. Atomicity in electronic commerce. In *netWorker: The Craft of Network Computing, vol. 2(2), p. 32-43*, April/May 1998.

[15] The information for the ROHDE case has been collected through interviews with the General Manager and the Sales manager at ROHDE, during the spring of 1999.

[16] C. J. Date. *An Introduction to Database Systems.* Addison-Wesley, 1990.

[17] Ramez Elmasri & Shamkant B. Navathe. *Fundamentals of Database Systems*. Second Edition, Benjamin/Cummings, 1994.