– Online store for Niki shop

By

Carbon soft Solutions

**The Application and its overall architecture**

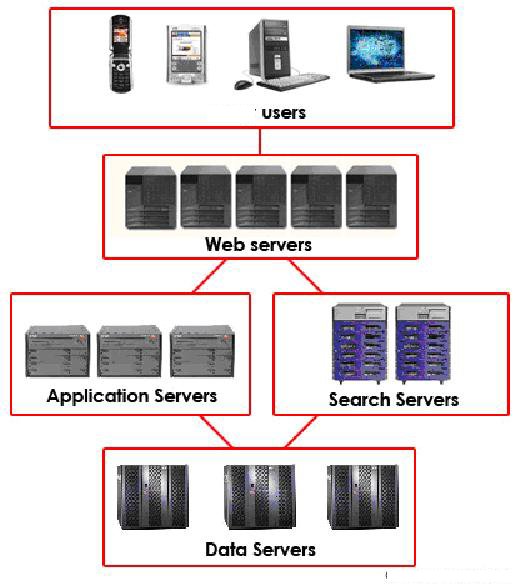
Niki shop is a eCommerce Platform for the search of computer peripherals and laptops and hard disks, Printers etc.

The architecture for a system of this size goes through many iterations. An architectural solution of this level is not only based on the software architecture but also on the system architecture, since the components that populate the system are not just web servers and web clients but it also has databases, security servers, application servers, proxy servers, and transaction servers. The system under discussion has a 3-tier architecture with a web-enabled device (a browser), application and transaction servers, and databases at the data services layer. The following diagram illustrates the relations between these components.



**Component model and its interactions**

Web servers, Web clients, databases, security servers, application servers, proxy servers, transaction servers.



**Web browsers for Modifiability**

The end user typically interacts with the website through a web browser. Web browsers

support user interface modifiability in a wide variety of ways, as the user interface that the

browser supports is not hardwired but it is specified via HTML.

**HTTPS for Security**

Once the user has submitted a request, it must be transmitted to the target web site. This

Transmission may often be sensitive information such as credit card, and HTTPS (HTTP

Secure) is used for this purpose. HTTPS uses Secure Sockets Layer as a sub protocol underneath HTTP. This level of encryption is adequate for the exchange of commercial

**Databases for Performance, Scalability, and Availability**

Finally, the request for service arrives at the database, where it is converted into an instruction to add, modify, or retrieve information. Current database systems frequently use

internal replication for performance, scalability, and high availability. They also use caching

for faster performance.

**Technological Aspects of the NIKI Shop Architecture**

The databases are segmented into functional areas, for example user hosts, item hosts,

account hosts, feedback hosts, transaction hosts, etc. This supports functional decoupling

and isolation. There are no client side transactions. Single database transactions through

anonymous PL/SQL blocks. There are no distributed transactions. It is all based on careful

ordering of db operations, with asynchronous recovery events.

Although the J2EE framework is the bases of the Application Tier, the system uses only the

basic functionality of J2EE and keeps the application tier cocmpletely stateless. There is no

session state and all transient states are maintained in cookie or scratch database. The

application tier is partitioned into the presentation, business and integration tiers.

Niki Shop has built a software-based Integration Tier. This contains both a data access layer

(DAL) and a services framework. The DAL is Niki Shop's internally-developed pure Java

Object-Relational (OR) mapping solution.

It leverages both component-oriented and service-oriented architecture technologies. Niki Shop

has built its own service architecture and uses it to enable integration across disparate

technology stacks. An example of this is the open interoperation between C++ and Java

technologies via services.

Within the Application Tier there is vertical code partitioning where applications (buying,etc.) depend on domains but there is no inter dependancy among shared domains.

Many EJB servers can coordinate transactions that involve multiple objects residing in

various processes in a distributed system. If the distributed transaction implementation

provided by the EJB server incurs additional remote calls in coordinating transactions, using

distributed transactions can slow down an EJB system considerably, inhibiting overall

system scalability. Therefore Niki Shop had to achieve The application stack is a very core J2EE

set of design patterns. Application objects are the equivalent of session beans, and the business object which are the equivalent of entity beans. scalability by developing their own

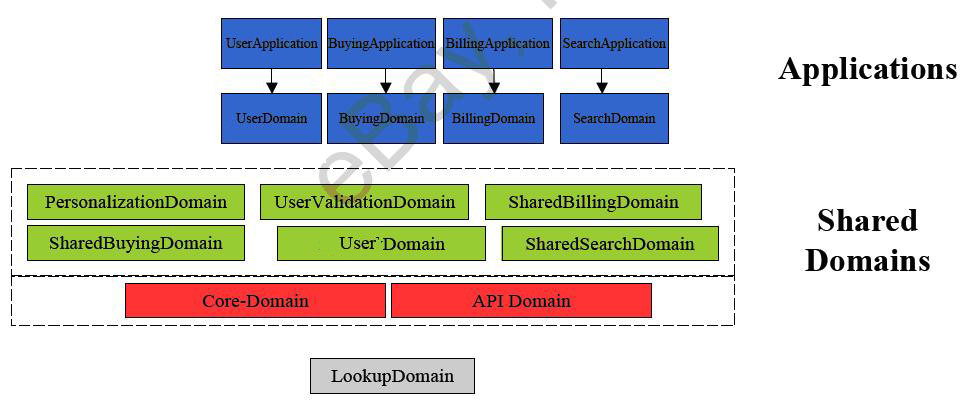
solution for transactions.

This combined application and data architecture "creates the secret sauce for extreme

scalability" for Niki Shop.4

The main technologies used in other areas are

* MSXML framework for presentation layer (even in Java)
* Oracle databases, WebSphere Java (still 1.3.1)
* J2EE: use servlets, JDBC, connection pools.
* JavaScript, JSON, AJAX, Flash/ActionScript, PHP, Perl, Python, .NET e.g. C#, ASP, VB, or Java, and JSP.



**Duration of the** **Project**

The Total project has taken a period of 3 months to develop and 1 month to test each and every part of the application databases API payment gateway and all and has been taken 1 month to quality check the entire application

Total a period of 5 months has been taken to go live