

expense@work has been designed from the ground up as an Enterprise Application capable of supporting thousands of users and the many hundreds of thousands of resulting transactions. Built on .NET technology standards, expense@work meets the most stringent security, reliability and scalability performance criteria.

Introduction

This paper deals with expense@work's scalability credentials, focusing on how its architectural framework has been designed with scalability in mind.

*"Scalability is
all about
Systems Architecture"*

Scalability

Scalability is the capability to increase resources in order to yield a linear increase in service capacity. Put another way, scalability is the capacity to handle additional users or transactions without fundamentally altering the application's architecture or design.

In designing expense@work to be truly scalable, steady performance is achieved by adding additional resources such as servers or memory as the load increases. While other factors do come into play, when we talk about expense@work being a truly scalable application we have sought to achieve a direct or linear relationship between the amount of resources added and the resulting increase in performance.

The foundations of this scalability were laid in the design of expense@work's Systems Architecture. By using

Microsoft's .NET technologies, expense@work has been built using solid object-oriented techniques and developed as a scalable n-tier application.

Because expense@work has been developed using VB.NET and ASP.NET it benefits from the framework in a number of key areas:

- Performance
- Scalability
- Reliability
- Ease of Deployment

expense@work has been designed with scalability in mind and by using ASP.NET it benefits from features specifically tailored to improve performance in clustered and multiprocessor environments.

In addition, processes are closely monitored and managed by the ASP.NET runtime, so that if one misbehaves (such as leaks or deadlocks), a new process can be created in its place, which helps keep expense@work constantly available to handle requests.

Architectural Framework

At the heart of expense@work's architectural framework is an

integrated, standards driven philosophy that enables users to work, collaborate and run their business in real time.

Its highly configurable and flexible design allows deployment of expense@work as a true, multi-company, multi-language and multi-currency application.

By designing expense@work with a zero footprint client, its setup, deployment, and administration can all be handled centrally with no requirement for modifications on the client side. The result is a true web enabled application which enables your employees to access and manage their forms from anywhere.

Some of the key components of expense@work's architectural framework are:

- SQL Server Database
- XSL & XML where appropriate
- Zero footprint deployment
- Flexible reporting and BI engine
- Powerful workflow engine
- Robust security engine

Continued...

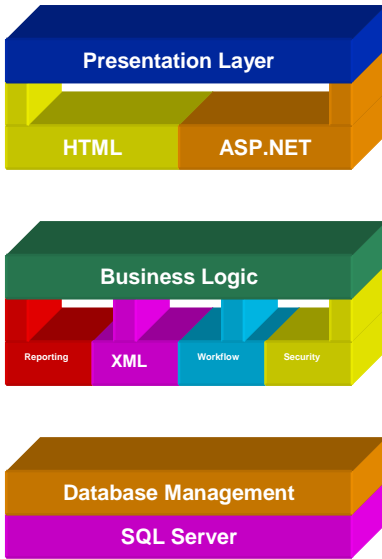


Fig 1: Architectural Framework

Database Management

The database engine used to power expense@work is Microsoft SQL Server. SQL Server has been designed as an industrial strength RDBMS and delivers scalability for e-commerce, data warehousing, and line-of-business solutions.

Independent benchmark testing has shown that SQL Server can handle over 700,000 transactions per minute (709,220 to be precise!) and has established its position as one of the fastest databases in the world. On the Transaction Processing Performance Council (TPC) TPC-C performance list, SQL Server and Microsoft Windows® 2000 hold three of the top five Clustered TPC-C results.

Presentation Layer

By utilising browser technology as its delivery mechanism, expense@work’s user interface is simple to use, easy to customise, rapidly deployable and highly scalable.

Its secure, role-based flexibility and intuitive interface enables organisations to achieve rapid and

painless user acceptance and to roll out expense@work with the minimum of training. Its zero footprint design enables fast and painless deployment for tens, hundreds or thousands of users.

Optimisation

One of the most powerful aspects of expense@work’s scalability is in the design and optimisation of its data structure and reporting engine.

- expense@work’s sophisticated calculation algorithms are optimised to avoid excessive data access by using advanced caching techniques to store control and configuration data.
- expense@work allows the construction of alternative index’s which greatly improves reporting speed for key management reports.
- Reports will monitor their own performance efficiency thus enabling administrators to see how many records have been accessed and how long it took to access them. The alternative indexing features of expense@work can then be utilised to enhance reporting speed.

Integration

expense@work’s integration framework and XML capabilities enables organisations to exchange real time data with other enterprise applications such as CRM and Financial Systems. Using XML, expense@work’s data structures can be passed among processes in distributed environments providing a more flexible and neutral

communication medium than binary solutions such as RPC and CORBA.

Security

As an enterprise enabled application capable of supporting many thousands of users, expense@work was designed with a highly configurable and robust security engine at the heart of its business logic. In addition, expense@work avails of the considerable security capabilities inherent in SQL Server.

To be able to access expense@work data a user must pass through two stages of authentication, one at the SQL Server level and the other at the database level.

These two security levels are implemented using logins names and user accounts respectively. A valid login id and password enables expense@work to connect to the SQL Server and a valid login id is required to access an expense@work database.

During a new connection request, SQL Server verifies the login name supplied using either Windows authentication mode and Mixed mode.

Within the expense@work application itself, further security is available to the Systems Administrator.

Summary

expense@work has been designed with scalability in mind both at the application and architectural level. This design philosophy allows organisations to address their business needs in a timely fashion, and addresses the cost concerns of running a powerful application such as expense@work within the their existing and future IT infrastructures.