

The evolution of



by Ashutosh Bijoor

the platform for

on2biz

developed by



Reach1to1 Technologies (P) Ltd.
29/6 Lotus, Garden Lane,
Shere Punjab, Andheri (E)
Mumbai 400093
India
<http://www.reach1to1.com>
<http://www.on2.biz>

Table of Contents

Introduction.....	2
Wolfkrow 0.1 (1996) – the Application Templates Engine.....	2
Wolfkrow 0.2 (1998) – The Virtual Object Storage System (VOSS).....	4
Object Database Advantages.....	4
Relational Database Advantages.....	4
Sitewallah – the web based file sharing application.....	4
Wolfkrow 0.3 (2000) – Multi-tenant Layered Customization Engine (M-LACE).....	5
Order Manager – the document-oriented ERP for small businesses.....	5
Wolfkrow 0.4 (2002) – Forms Wizard (FORMWIZ).....	6
Custom web-based workflow applications.....	6
Wolfkrow 0.5 (2005) – Application Module Registry (DOCMGR).....	7
SAMS- Sales Activity Management System.....	7
Wolfkrow 0.6 (2006) – Object and Registry Caching & AJAX.....	8
On2Biz – online business process management.....	8

Introduction

Wolfkrow is the application framework that Reach1to1 has been developing and using since 1996. The framework represents efforts by the Reach1to1 team to realize the dream of providing customized enterprise software as an online, hosted service, now commonly known as “**Software as a Service**”.

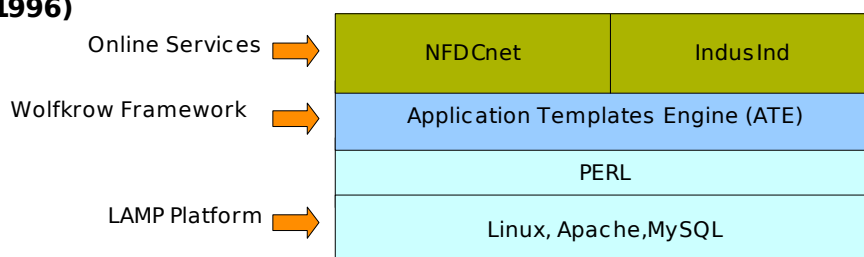
This document describes the evolution of Wolfkrow through the 10 years from 1996 to 2006, describing briefly all the various components that make up Wolfkrow, and their role in creating a stable, scalable platform for deploying and delivering software as a service

Wolfkrow 0.1 (1996) – Application Templates Engine

The development of Wolfkrow started in 1996 when we were contracted to develop the first web based application in India - **NFDCnet** for NFDC (National Film Development Corporation) in 1996. For NFDCnet, we created a web-based **Application Templates Engine** very similar to the numerous web based template engines like PHP, ASP etc. that are available today. We needed to develop our own because at that time, template engines like PHP/ASP/.NET etc. were not yet developed. We based the framework on the Linux operating system, and user Perl as a programming language.

The same framework was later used for implementing IndusInd Bank's online banking application.

Wolfkrow 0.1 (1996)



These initial projects gave us our first experience of developing web based business applications, and the seed of the idea of providing software as an online hosted service using the Internet as the delivery platform.

NFDCnet – the first extranet application in India

NFDCnet was the first extranet / e-commerce application done in India for the [National Film Development Corporation \(NFDC\)](#) which was initiated by Indiaworld - a company started by [Rajesh Jain](#). NFDCnet allowed advertising agencies to book TV advertisement spots on Doordarshan (India's primary TV channel) via the Internet. NFDCnet was considered a huge success, especially because we were able to successfully complete the project in 3 months flat, at one tenth the cost quoted by a reputed software firm for a client-service model application on which they had already worked for over 2 years without completing it!

Wolfkrow 0.2 (1998) – The Virtual Object Storage System (VOSS)

In 1998, we created the first version of **Virtual Object Storage System (VOSS)** in Wolfkrow. VOSS is a hybrid of an [object database](#) and a relational database developed in PERL and MySQL. The VOSS hybrid object and relational database technologies give the benefits of both systems:

Object Database Advantages

- persistent objects are stored and retrieved very fast as they do not require joins between multiple tables as required in a relational database
- allows every object in an object store to have an independent structure, allowing the application to be highly customizable. For example, customization of forms used to edit every object in VOSS can be carried out without having to redesign the RDBMS tables
- relationships between objects can be followed very easily without having to query the database, leading to faster response times in a web application

Relational Database Advantages

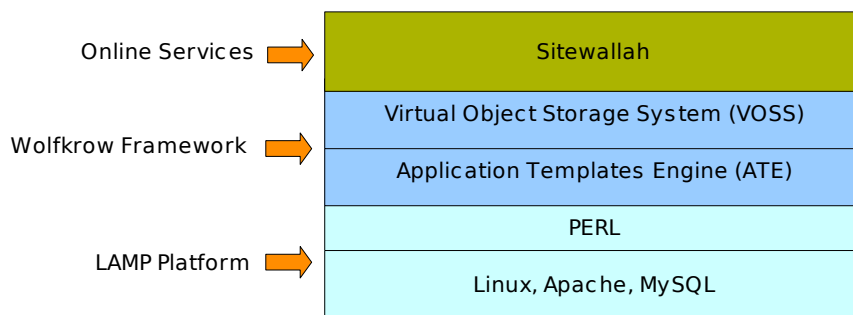
- faster queries over multiple objects that share a common sub-set of data. Primarily useful for generating reports and analytics
- Relational databases are used by VOSS as an indexing device for the objects in the object store

Sitewallah – the web based file sharing application

VOSS was primarily used to empower Reach1to1's first online software service called **Sitewallah**. It allowed users to maintain folders and files in an online shared space. The user interface of Sitewallah was similar to a file manager, with permissions for read, write, delete access controlled by the VOSS.

As an online service, Sitewallah was way ahead of time. It never really took off because the connectivity in India at that time was too poor - most companies had dial-up connections. Connectivity costs were also too high, making Sitewallah quite unusable as a web-only application.

Wolfkrow 0.2 (1998)



Wolfkrow 0.3 (2000) – Multi-tenant Layered Customization Engine (M-LACE)

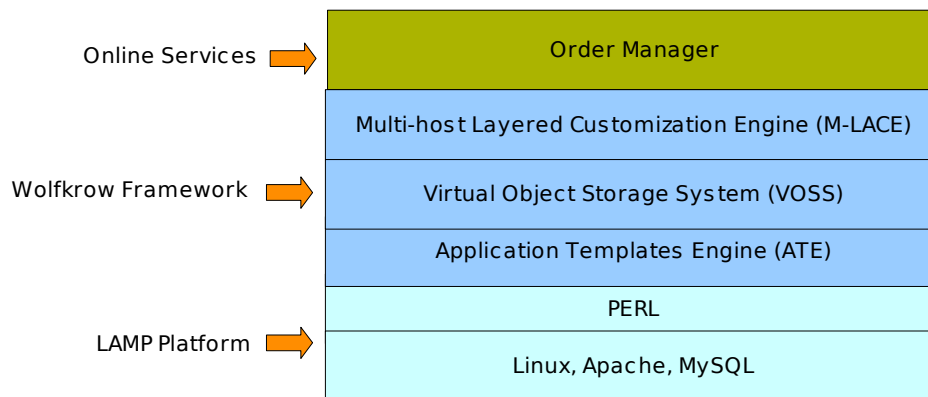
We realised quite early that the key to successful software delivery on the web was the ease of customization. Even in Sitewallah, every customer we got required customized documents. And all customized versions had to reside on the same server (multi-tenant).

In 2000, we developed the first version of **Multi-tenant Layered Customization Engine (M-LACE)**, which allows for a single server to host customized versions of the same application for multiple client installations. M-LACE allows customization of application modules in layers:

- **Customization of the User Interface:** The user interface can be customized for any one client by web / UI designers
- **Customization of the Data Structure and Business Logic:** can be customized for any client by application engineers without affecting the User Interface or Core Application on the same server
- **Shared Core Application:** The core application can be shared between all client installations and can be maintained by the core product development team

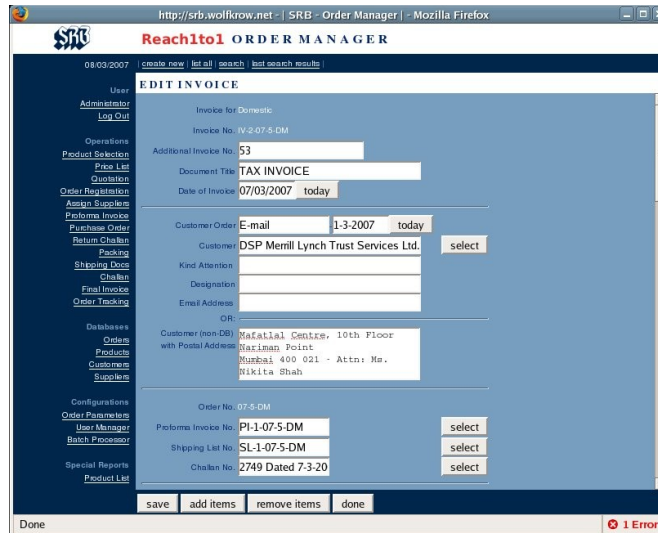
This architecture allowed us to customize our applications at extremely low costs and time (mass customization!). It was perfectly suited for hosting customized applications for multiple customers on the same server.

Wolfkrow 0.3 (2000)



Order Manager – the document-oriented ERP for small businesses

Using **M-LACE**, we developed our first major web-based enterprise application - the **Order Manager**. The Order Manager is a simple document-oriented ERP application that is suitable for a trading/distribution organization. It maintains the entire database of customers, suppliers and products that are sold by the company. It provides for online creation of the entire gamut of documents from price-lists, quotations, order confirmations, proforma invoices, shipping and packing lists, delivery challans etc. All documents are completely customizable.



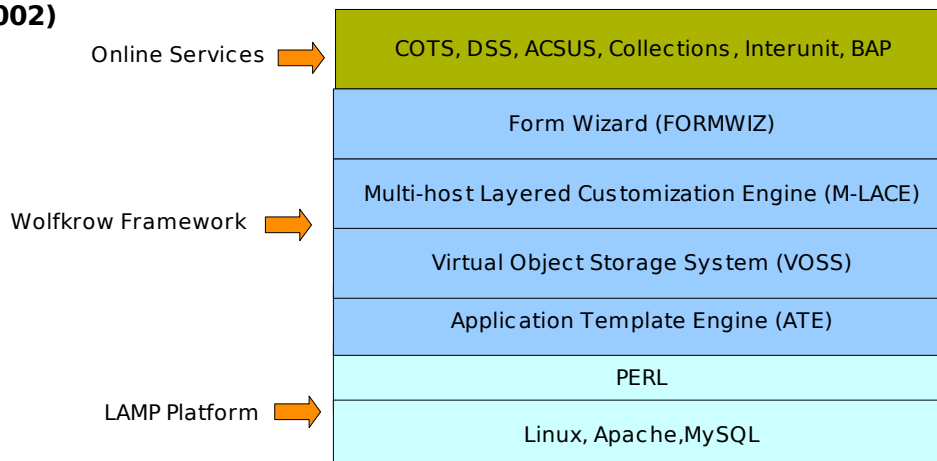
Customers for Order Manager are using the system as a paid online service. Though the Order Manager initially suffered low adoption due to bad connectivity and high cost of hosting, now with broadband connections making fast connectivity affordable, the Order Manager is a fast, efficient and compact ERP application that is perfectly suited for small trading and distribution companies.

Wolfkrow 0.4 (2002) – Form Wizard (FORMWIZ)

We realized that the biggest bottleneck to implementing successful web-based services was creating **usable web interfaces**. Users familiar with conventional enterprise software installed on their local systems find it hard to adapt to the web-based model. The user interface has to be designed so as to function as close to native applications that users are familiar with.

The **Form Wizard** was developed to provide ready-to-use widgets that can be put together in customizable forms and reports that resemble those available in conventional applications.

Wolfkrow 0.4 (2002)



The Form Wizard provides a simple form definition language (not as exhaustive as the [Xforms](#) specification). It allows a low-skilled programmer to easily customize forms using a simple form definition language, with event drivers written in Javascript.

Over the years, we have also developed a large form widget library, which provides ready-to-use components for putting together complex forms in a jiffy.

Custom web-based workflow applications

To further validate Wolfkrow as a platform for providing customized web-based workflow applications (and to help us pay our bills), we implemented several custom projects using Wolfkrow:

- **COTS - Customer Order Tracking System**, which allows customers to track the status of every order online via the Web
- **DSS - Distributors Stock Status**, which allowed distributors and channel partners to maintain stock positions of all their products online, so that visitors to the web site could check the availability of any product at any location
- **ARMS - Accounts Receivable Management System** – a system that extracts pending invoices from a back-end ERP system and tracks the payment collection process
- **Inter-Unit Accounts Reconciliation** – a system that allows accounting transactions at multiple locations to be reconciled over the Internet
- **BAP - Branch Automation Program**, which provides basic sales management capabilities to sales branches at multiple locations.

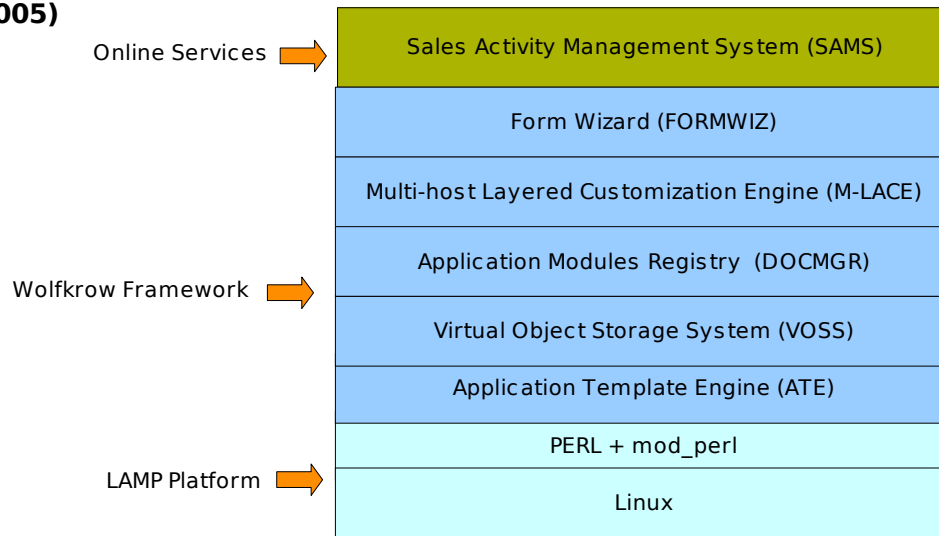
All these web based services that were developed using the Wolfkrow framework. They gave us the opportunity to test the framework thoroughly, and each module gradually evolved, including upgrades of the Virtual Object Storage System, the Application Templates Engine and the **Form Wizard**.

Wolfkrow 0.5 (2005) – Application Module Registry (DOCMGR)

The most significant development in 2005 was the **Application Module Registry** – fondly called as **DOCMGR** - a system that allows different application modules to work in an integrated fashion seamlessly. As additional modules get added on, they declare their linkage points in the registry, thereby allowing other modules to connect seamlessly.

The Registry is similar to the now commonly known **Web Services** architecture that helps application modules to be linked to each other easily, without having to write any intermediate code. The Registry maintains information about hooks for integration of application modules with each other as well as subscription data that permits users to access application modules as part of the Software as a Service (SaaS) model.

Wolfkrow 0.5 (2005)



The **Application Module Registry** is accessed during each and every operation in the application module and acts as an auditor for each module, ensuring that no information is exchanged between modules or users until authorized.

Another major milestone in 2005 was the conversion to mod_perl from the previously used CGI model. This also led to a significant improvement in performance.

SAMS- Sales Activity Management System

The biggest and most important application developed on the Wolfkrow framework in 2005 was **SAMS - Sales Activity Management System**. SAMS is a full-scale enterprise sales management system, which allows an organization to manage their entire sales force at multiple locations. It captures data at every stage of the sales cycle - every lead, opportunity, prospect, sales call, quotation, proposal, etc. is captured in the system, and **SAMS Analytics** generates daily activity reports that measure activity of each sales person, as well as provides an instant view of the sales pipeline, which forms the basis for all sales tracking and very soon, even sales incentives.

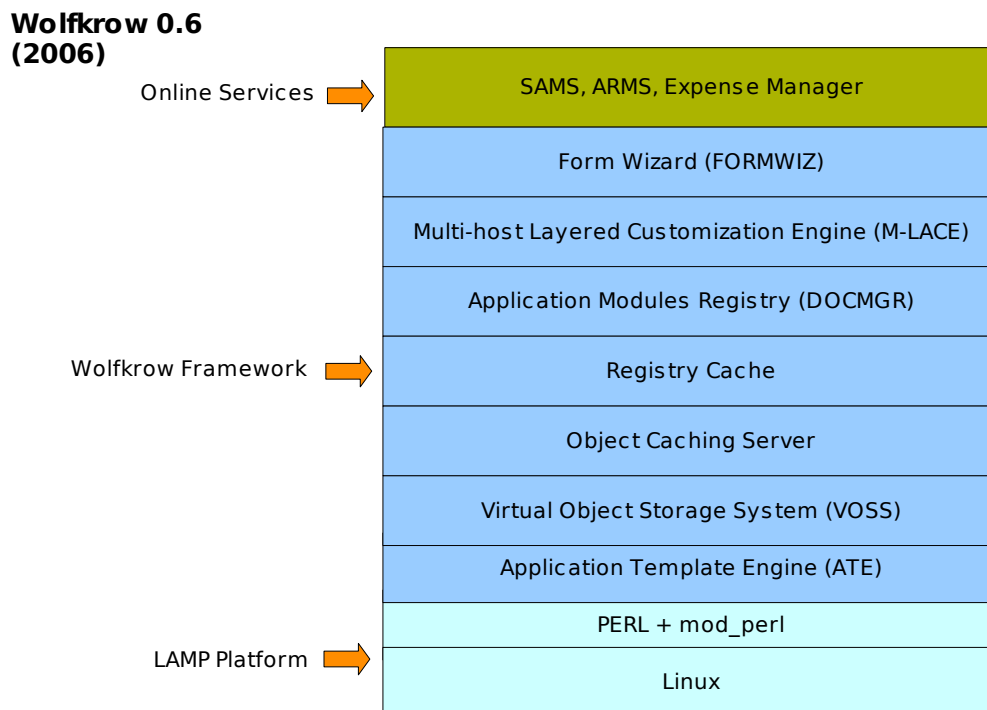
SAMS was completed in December 2005 after over two years of iterative development. The initial users of SAMS have played an extremely important role in helping to verify the usability of the application, to identify bugs that escaped the testers, and most important, allowing us to learn the art of **successful user adoption** - which is far more difficult than the developing software.

Wolfkrow 0.6 (2006) – Object and Registry Caching & AJAX

In August 2006, we installed a major upgrade to the Wolfkrow framework that improved the performance of applications by 25% to 50% based on our benchmark tests.

This upgrade contained the following new systems:

- 1. Object Caching Server:** This add-on to the **VOSS** is an object caching server. It works by creating a cache of recently used objects, so that loading these objects is much faster than retrieving them from the database. The Object Caching Server works as a gateway for each database operation
- 2. Application Modules Registry Caching:** The Application Modules Registry is the auditor for all information exchange in the Wolfkrow framework. Hence, the registry is probably more frequently accessed than the database itself. Till now, the registry data was cached on disk and was therefore re-loaded by every server request in the application. The new Caching server caches the registry in a special shared memory, that can be accessed directly by each request without requiring to re-load it
- 3. AJAX:** With the widespread use of [AJAX](#) in Web 2.0 applications cropping up in 2006, we quickly adopted it in Wolfkrow. We have adopted the popular [jQuery](#) library, an open-source Javascript framework that provides extremely compact code with highly flexibility.



These upgrades have resulted in significantly improved application response times, as well as drastic simplification of user interfaces, leading to higher levels of user adoption.

On2Biz – online business process management

In October 2006, we pre-released On2Biz – a scalable platform for web based business process management. On2Biz combines all the advantages of the Wolfkrow platform, providing high levels of customizability at extremely low costs, high performance and dependability. On2Biz provides simple yet comprehensive facilities to manage multi-stage business processes and performance management facilities. Early users of On2Biz have given a tremendously positive response. So much so that we have now stopped all custom project development, and are completely concentrating on building and deploying the On2Biz service.

Having crossed the 10th anniversary of Wolfkrow, 2007 promises to be the most significant year for us in Reach1to1, as finally all the components of our vision are falling in place. Fast and cheap connectivity, cost-effective hosting, a proven enterprise application platform, a strong core team of developers, a highly supportive set of customers, and an emerging set of business development partners.

Working towards making the dream of providing customized enterprise software as a service come true!

