

Enterprise Insurance Suite

Arch Insurance Group

Arch Insurance Group, a relative start-up at the time, hired Bantix to architect, design and write a suite of web-based, enterprise-level insurance applications to run on their corporate intranet. Prior to our involvement, the systems in place to support the underwriting functions of their business were ad hoc or off-the-shelf and limited in their ability to accommodate the changing needs that accompanied Arch's growth in the insurance industry.

The applications (or Domains) in the "ArchLink" suite provided separate and distinct functions on a piece of business as it moved throughout the complex life-cycle of a policy. Each domain had to be setup as an individual application that could be worked on by different development groups and that could be deployed independently of the other applications. At the same time, the domains had to be able to communicate with each other in order to allow data to be shared and to facilitate the flow of a potential piece of business throughout the policy process. To keep the application boundaries distinct, yet handle the communication requirements, a service-oriented



architecture was established. Each application had its own data store that it accessed directly, but used service calls to get information or instantiate functionality from the other systems. Additionally, it was required that all applications be consistent in architecture, look and feel, performance and behavior (where possible). To facilitate this, Bantix first designed and developed libraries of reusable code and ASP.NET server controls to be used not only for the initial set of applications, but ultimately for the many applications that would follow as the suite became more complete.

During the initial phase of the project, Bantix's role was to develop the system portal website application called BizShell. It was the entry point into the suite of domain applications, most of which would be written during subsequent project phases. BizShell communicated with all ArchLink applications (via its service layer) to retrieve work in progress (WIP) items that were specific to the currently logged in user. Through these lists of WIP items, BizShell provided instant access to a piece of business, no matter the current stage of the life-cycle, by redirecting to the appropriate domain application. BizShell also provided very sophisticated searching capabilities across all domains.

Bantix was also responsible for developing many of the domain applications for the ArchLink suite. The first of these was the system for managing all customer accounts and business submissions. The application not only managed the data associated with all Arch customers and submissions, but it also incorporated special "clearance" algorithms to help ensure that a customer or a piece of business was not entered into the system more than once. Following the completion of the Account\Submission system, Bantix wrote separate systems for managing Collateral and Documents. Additionally, we were involved with developing applications for managing Policies, Producers and Forms.

To enforce UI consistency across the various applications, Bantix developed well over 60 web server controls and control extenders that were used as the foundation for all of the ASP.NET domain applications. Most of the controls made heavy use of AJAX and JavaScript technologies to improve the overall UI responsiveness of the host applications. By using a set of shared ASP.NET Themes (combining CSS style sheets, skins, images, etc.) for both the controls and the applications, we were able to easily enforce a universal look-and-feel. In addition, because the library of shared code and controls was so comprehensive, it greatly reduced the development effort needed to complete the subsequent ArchLink applications.

Some of the key technologies Bantix used while working on the ArchLink applications included:

SqlServer, SqlServer Full-text Search, ASP.NET, ADO.NET, Ajax, JavaScript, SharePoint, Windows Workflow, C#, XML, Active Directory, SOA