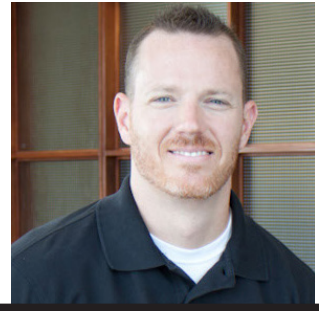


About the Author

Ted Theodoropoulos is the President of Acrowire, an IT consulting firm specializing in technology solutions for legal services. After 10+ years in several corporate positions at Microsoft and Bank of America, Ted pursued his passion and business full-time, driving tangible business results through technology. He helps firms increase productivity and eliminate inefficiencies, ultimately accelerating growth through SharePoint development, software development, business process improvement, virtualization and cloud services. Contact Ted at ted@acrowire.com.



The App Model – Build Alongside, Not On, SharePoint

Microsoft's latest release of SharePoint represents a major milestone for the product in terms of new capabilities and adoption of industry standards. While Microsoft has been touting their commitment to industry standards for some time now, SharePoint 2013 now delivers on that promise with the new "app model."

What's the difference between the old "solution model" and the new "app model"? How can law firms, legal departments and their enterprise content management systems and processes benefit from the app model? Firms are finding out building alongside SharePoint is a better approach than building on SharePoint.

SHAREPOINT AS A DEVELOPMENT PLATFORM

Many organizations that need to build custom Web applications will default to a .NET or open source architecture. While this approach makes sense in many cases, using SharePoint as a framework brings building blocks that can be leveraged to piece together a Web application very quickly. These out-of-the-box features address needs many law firms have, including:

- Content management and intranets
- Extranets and client portals
- Basic document management
- Workflow and process control
- Enterprise search
- Integrated authentication
- Calendaring and collaboration

In the past, the downsides of customizing SharePoint were significant and often locked organizations into the version of the product they customized. It is common to see law firms running SharePoint environments two and three versions back because they know an upgrade will likely break the customizations the firm now depends on. With SharePoint 2013, however, developers can leverage all out-of-the-box capabilities and deploy custom features without the risk. As with any new technology, there is a learning curve, but the benefits will outweigh the drawbacks often.

THE OLD WAY: THE SOLUTIONS MODEL

To understand why the new app model is so important, we must understand the solution model and how custom functionality has been built in previous versions. The most popular way to deploy custom code in SharePoint 2007 and 2010 is by leveraging full-trust solutions (called "full trust" because, once deployed, they have broad access to the internal workings of SharePoint). This makes them extremely powerful and easy to build. Even in SharePoint 2013, fully trusted solutions usually are the quickest way to deploy custom functionality to on-premise environments. (A full-trust solution cannot be used with SharePoint Online.) The full-trust approach has been in place for a long time, which means

it is well-documented and understood. The approach also gives developers access to the most complete application programming interface (API).

As great as this solution model approach sounds, it has a major flaw. The custom code runs directly on the SharePoint server. This presents challenges to administrators who need to keep the environment up and running efficiently. Fully trusted solutions extend the capabilities of the product on the SharePoint server, so when custom code misbehaves it can bring down the entire environment. The deployment of custom code can crash SharePoint, affecting hundreds or thousands of users. It is a classic case of custom code having too much access with administrators having too little control. Even with quality assurance (QA) processes in place, it is difficult to mitigate the risk fully because of how trusted solutions can respond to different environmental factors.

Although full-trust solutions are sometimes quicker to deploy and appear cheaper to develop in the short run, they often lead to a higher overall total cost of ownership (TCO). The savings derived from customizing SharePoint in this way are less like true savings and more like technical debt. This debt drives up TCO through long-term interest costs in the form of:

- Environments that are less stable
- Capacity planning made more difficult
- QA processes that need to be more rigorous
- A product forced to remain in the legacy version
- Deployment models that are clumsy
- Additional environmental complexities
- Developers with specialized knowledge that cost more

Microsoft attempted to address some of the shortcomings of the full-trust solution model with what they called “sandboxed Web parts.” However, that was a failed experiment; the approach has since been deprecated.

THE NEW WAY: THE APP MODEL

There is a widely accepted design principle in software engineering called “separation of concerns” (SoC). SoC promotes the use of a loosely coupled architecture that encapsulates code into modular units that address individual tasks. Separating concerns reduces complexity and makes swapping out pieces of the system less challenging by reducing dependencies.

The SharePoint 2013 app model fully embraces this mantra by mandating that no custom code ever run on the SharePoint server directly. Custom application code must run either in the Web browser (*i.e.*, HTML/CSS/JavaScript) or on a separate server altogether. Instead of building directly on SharePoint, you are building alongside it, which differs from the fully trusted solution approach.

In addition to the changes in architecture, the app model also changes the underlying technologies used to deliver custom functionality. The fully trusted approach uses server-side solutions typically written in C#, leveraging the server-side object model (SSOM). Apps are not deployed on the SharePoint server, so they can be written in many server-side technologies. While this opens many doors, the app model approach necessitates the need to

interact with SharePoint using the client-side object model (CSOM) via a REST-based API usually accessed through JavaScript. Some SharePoint apps can use client-side code that runs in the browser exclusively, eliminating the need for any server-side infrastructure.

MORE BENEFITS OF THE APP MODEL

1. Hire Fewer Staff with Specialized Skill Sets: According to an annual study by Dice.com, the national unemployment rate for software developers is under two percent as of late 2013. The rate is expected to stay at or below this level in 2014. When the labor market is this tight, it puts upward pressure on payroll costs and makes attracting and retaining talent challenging. This is especially true in the legal industry. While there are innovative and technology-driven law firms out there, the industry is not perceived as progressive or cutting-edge. Law firms usually are not the first place talented technical professionals have on the radar. Regardless of how you feel about that perception, it can make competing for and retaining top technical talent even more challenging for law firms.

Using apps to deliver custom functionality now requires less SharePoint-specific knowledge because developers use industry-standard technologies to build apps, such as HTML, JavaScript, CSS, REST, JSON, OData and OAuth. Developers must still have a working knowledge of SharePoint objects and how to interact with them using the CSOM, but several development and deployment nuances of full-trust solutions have been reduced or eliminated with apps. This lowers entry barriers for developers by reducing the amount of specialized SharePoint domain knowledge required. More developers know how to build apps using open standards over the old SharePoint solution model. The app model reduces the required amount of SharePoint-specific expertise, expanding the pool of candidates in an already tight talent pool.

2. Upgrade and Secure with Fewer Issues: Full-trust solutions are entwined with SharePoint’s underlying infrastructure, so upgrading a customized environment can be extremely high-risk. Even applying patches and service packs could break customizations. Keeping SharePoint up to date is about more than just enabling new capabilities; it is also about maintaining a secure environment that is supported by Microsoft. Security vulnerabilities are identified all the time in the IT world, and keeping products updated is the number one countermeasure against attacks. Having a fragile SharePoint environment that cannot be updated due to tightly coupled customizations creates a difficult dilemma.


It is incumbent upon technology leaders to maintain a reliable and secure IT environment that provides the capabilities the business needs to stay competitive. Deploying tightly coupled SharePoint solutions makes it difficult to achieve that mission by making firms reluctant to deploy the next version or service pack. Customizing SharePoint using the app model drastically reduces the need to shy away from updates by keeping customizations loosely coupled. Apps use a service-oriented architecture that reduces dependencies and does not lock firms into the version of SharePoint currently being used. This will ensure vulnerabilities are mitigated, the latest capabilities are enabled and the environment is supported should a call to Microsoft become necessary.

3. Launch SharePoint Online: Many law firms have not embraced cloud computing for various reasons. Some firms have clients that do not allow it or that have regulatory requirements that make multi-tenant platforms like SharePoint Online (a part of Microsoft's Office 365) a non-starter. Other firms are not comfortable with their data residing outside their network, or they stayed away from SharePoint Online because deploying custom functionality there had significant limitations. The app model does not always address regulatory issues or a lack of comfort with the cloud, but it does remove many limitations developers faced customizing the platform.

For firms not prohibited from using the cloud, SharePoint Online is now a viable alternative. SharePoint is a great product, but it can get very expensive when you factor in the licensing, hardware and manpower required to keep a firm running at peak performance. SharePoint Online uses a cost-effective subscription model and eliminates the need for firms to manage the environment and keep it secure and up to date. There are still limitations to SharePoint Online, but the app model has at least made it worthy of consideration.

4. Reduce Risk and Improve Performance: Code never runs on the SharePoint server using the app model, and it is difficult for the platform to be severely affected by the resource utilization of apps. Apps are built alongside SharePoint, so you can load test them independently. The signal-to-noise ratio goes down dramatically when you can test custom code independently from the SharePoint server. All of this translates into a better user experience and higher service levels, both of which are two of the most important metrics for any IT environment.

MODELING A MOVE TOWARD REAL CHANGE

Leveraging the app model can be a real game changer for your firm and its enterprise content management strategies. As Microsoft continues to make SharePoint more appealing to the next generation of application developers, resisting adoption will not be a sustainable strategy. With the advent of the SharePoint app model, there are ample opportunities for growth for current SharePoint developers and the potential for real change in how your law firm leverages the platform. 



This article was published in ILTA's April 2014 white paper titled "The App Model — Build Alongside, not on, SharePoint" and is reprinted here with permission. For more information about ILTA, visit their website at www.iltanet.org.