SharePoint to the cloud—learn how Microsoft ran its own migration

If you’re considering moving your SharePoint sites to the cloud, there are a number of things to think about first. Do you trust the cloud enough to make the move? If you decide to go, how will you deal with all of the sites that your employees have built over the years? If you’re like most companies, you have customized your SharePoint sites extensively, and you wonder how to move those customized sites to the cloud.

Figure 1. The Microsoft IT Office 365 migration journey. By fall 2015, Microsoft migrated 97 percent of its SharePoint sites and portals to the cloud. Today, employees have 191,000 OneDrive for Business profiles and Microsoft has 76,000 SharePoint sites, portals, and Office 365 groups.

Understanding our SharePoint situation

Microsoft employees rely on SharePoint, so we in Microsoft IT asked ourselves the same questions when it came time to move to the cloud. When we started our migration in 2011, employees were maintaining more than 70,000 SharePoint on-premises team and publishing sites, and 114,000 personal MySites. Divisions and groups within our company had built and were operating 240 custom portals handcrafted to do things like share news with employees, find information from groups like IT and Human Resources, and search for campus maps. So when it was time for us to migrate to the cloud, it was clear it wouldn’t be simple.

Rather than tackle migration all at once, we took a measured approach, encouraging our SharePoint site owners to consider if they should migrate at all. If they wanted to migrate their sites, we asked them if they wanted to start fresh so they could build their site just the way they wanted. For larger sites and portals, we used migration techniques ranging from moving as-is (lift and shift) to starting from scratch in the cloud.

This gradual approach worked. By fall 2015, we had moved 97 percent of our sites to the cloud. Even better, we were able to eliminate 50 percent of all the on-premises SharePoint sites on our company system, cutting costs as employees created new project and team spaces directly in the cloud. Along the way, employees could take advantage of new features and capabilities that came with the move to the cloud. And because SharePoint Online is more secure than typically managed, on-premises versions of SharePoint, we were able to make the way our employees and contingent staff store and save data safer—without sacrificing productivity.
These are the big migration challenges we faced:

- **Having too many sites**: There were far too many sites for us to consider them individually. First, we identified and shut-down sites that were not being used. Then we used a combination of automation and site-owner engagement to migrate sites that employees were actively using and wanted to keep.

- **Trusting the cloud**: Like any customer would be, at first we were cautious about moving to the cloud—so we kept our highly secured data on-premises. But once we saw that our migration efforts were working and that we had the expected security controls in place, we began moving all sites to the cloud regardless of their security level. We held a few sites back for regional compliance controls and complexity reasons. Today, we put our most secure sites in the cloud—even sensitive product and financial information.

- **Moving highly customized sites**: SharePoint enables companies to build highly customized internal communication portals and business solutions and, like many customers, Microsoft divisions and groups took full advantage of this by building dozens of multi-layered sites to communicate with employees. Moving these sites with their widely varying customizations intact and functional was a major challenge.

**Governance 101: Get persnickety about the right plan**

Once you decide to move your SharePoint sites to the cloud, the first thing to do is ask yourself what your goals are—how do you want your migration to go and what does success look like to you? Being precise about what you want to accomplish and what your guidelines will be before you start will help you focus on important details and will make it easier to make critical decisions during migration.

We knew we had a site proliferation challenge—our employees had built thousands of sites that, for a variety of reasons, were not being used. To address this challenge, we built an internal governance solution, AutoSites. AutoSites makes sure each site has two active full-time employee owners, and it requires those owners to annually confirm that their site is viable and needed. The tool also notifies us when a site owner leaves the company, which allows us to flag the team to find a replacement. If we don’t get a response, the site is automatically locked down and eventually deleted.

AutoSites did not eliminate the need for a good governance plan. We still needed to ask site owners to think very carefully about what they wanted to do with their SharePoint sites and to map out who needed access and at what level. Figuring this out before migration started was a must.
Just as we asked our site owners to consider governance before migrating their sites, we had a number of challenges to consider before we got started. Here are some of them:

**Table 1. Tackling governance**

<table>
<thead>
<tr>
<th>Our challenge</th>
<th>What we did</th>
</tr>
</thead>
<tbody>
<tr>
<td>We had 70,000 SharePoint sites to migrate.</td>
<td>We decided to take a mixed approach. We automated as much as possible by building repeatable processes that did most of the work. We also let our employees migrate their sites, which ranged from asking them if they wanted to keep the site, helping them start new sites, working with them on scheduling, and recruiting them to test the new site to make sure it worked properly.</td>
</tr>
<tr>
<td>One of the first things to do when you move a SharePoint site to the cloud is decide who gets access and at what level. This is a critical step before you begin migration.</td>
<td>We built this into the front end of the migration process. If a site owner wanted to migrate a SharePoint team site to the cloud, the site owner had to decide who would have access and at what level. We preserved their permissions as much as possible to minimize the work site owners needed to do.</td>
</tr>
<tr>
<td>We needed to make sure that we aligned our on-premises security configurations with the cloud to preserve secure access to migrated sites in the ways our site owners intended. Getting this right was crucial to getting owners of sensitive sites to agree to make the move.</td>
<td>Most Azure Active Directory (AD) security groups were replicated in the cloud to provide the same user-to-group linkage found on-premises. The exception was domain-calculated groups (groups without fixed membership) that we did not carry forward and had to be remapped. By replicating on-premises security groups and by re-mapping broader calculated groups to cloud equivalents, we preserved the same security access levels as on-premises.</td>
</tr>
</tbody>
</table>
| One of the main reasons we wanted our employees to move their SharePoint sites to the cloud was to make it easier for them to work anywhere on any device. This meant giving them access even when they were not on the corporate network. It also meant giving them access no matter what device they were using or where they were connecting. If our employees can’t use our technology when and how they want to use it, they will use less-secure technologies to get their work done and hide their actions from IT controls. | While it may seem counterintuitive, moving our SharePoint sites to the cloud made them more secure. There are many reasons why, but the larger concept is common sense—the cloud is more secure because Microsoft has invested heavily in the cloud. Updating on-premises sites with modern controls would be costly and hard to manage. Examples of how the cloud is more secure:  
  - Requiring two-factor authentication to access sites in the cloud.  
  - Allowing IT to wipe your device if it is stolen or lost.  
  - Providing encryption at rest with rights protection (data is encrypted when you download something and forget about it).  
  - Providing encryption in transit (data is encrypted as you move from site to site). |
| All of our employees have to classify data they create or work with in three levels: top-secret high business impact (HBI), moderate business impact (MBI), and low business impact (LBI). Each team had to decide what data to allow in the cloud, and so did we. | Initially, we allowed only MBI and LBI data in the cloud. Once Office 365 added encryption at rest and Azure Rights Management Service (RMS) to the cloud, we allowed HBI (top secret) data to be saved in the cloud. Now our highly confidential data sits in the cloud, including financial data that could influence markets if it leaked and sensitive product specifications and plans. |
| Laying a strong migration foundation meant considering everything we could think of ahead of time. | We built a governance plan, established our policies for using SharePoint in the cloud, mapped out when to require employees to move their sites to the cloud and when to allow them to stay on-premises, and then we built out how to handle hosting. |
Choosing what to move

Once we set up our governance plan, we had to decide what would go and what would stay. Here are the main types of SharePoint sites we moved to the cloud:

Table 2. Sorting out what to move and where to put it

<table>
<thead>
<tr>
<th>Workload</th>
<th>Where we put it</th>
<th>Why</th>
</tr>
</thead>
<tbody>
<tr>
<td>Companywide internal portals and internal search</td>
<td>All portals move to Office 365 SharePoint publishing portals.</td>
<td>Desire for a common intranet experience available where employees need it.</td>
</tr>
<tr>
<td>Business personal sharing (storing documents in the cloud</td>
<td>OneDrive for Business, the cloud-based place where employees store their work.</td>
<td>Single destination available on any device when employees are ready to work.</td>
</tr>
<tr>
<td>instead of on PCs)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Group and organization collaboration (how teams store and</td>
<td>Cloud-based SharePoint team sites and groups. These are the sites that made up</td>
<td>To make team collaboration sites available to employees to access whenever they need,</td>
</tr>
<tr>
<td>access shared work)</td>
<td>most of our migration work.</td>
<td>regardless of location or device.</td>
</tr>
<tr>
<td>Regionally regulated group and organization collaboration</td>
<td>Retained on-premises SharePoint or Office 365 dedicated sites to host assets in</td>
<td>On-premises allows the company to choose the region where to host its sites.</td>
</tr>
<tr>
<td>(extranets)</td>
<td>a specific region.</td>
<td></td>
</tr>
<tr>
<td>Partner collaboration (extranets)</td>
<td>Office 365 SharePoint Online team sites with external sharing AND on-premises</td>
<td>On-premises SharePoint extranet for managing our partner’s identity. Sharing with existing</td>
</tr>
<tr>
<td></td>
<td>dedicated extranet.</td>
<td>individual external accounts can be done on SharePoint Online.</td>
</tr>
</tbody>
</table>

By the time we finished our migration, we had moved 97 percent of our internal SharePoint sites to the cloud, including our most secure sites. We decided to leave about 7,000 SharePoint sites on-premises, mainly to support content based on the geographic region that the customer lives in. The last 3,000 or so sites were custom portals—sites that were built by hand with intensive customization. We handled these on a case-by-case basis, which we’ll talk about in just a bit.

Fresh start: No migration is the best migration

We encouraged Microsoft employees to think of migration as a fresh start. We asked, “Is there something you always hated about the way your team site works?” Instead of migrating old, unloved sites to the cloud, we encouraged them to build new sites exactly the way they always envisioned they should work.

Organizations change over time—new teams are created and new projects are assigned. We decided to allow teams to start building in the cloud well before migration started. This gave teams time to think about what they could do in the cloud if they just started fresh. It also had the secondary effect of stopping nearly all new development on-premises. To encourage this line of thinking, we made our default new site creation go to the cloud through a hosting options page where employees could choose what types of sites or experiences they needed through a self-service provisioning experience.

We did this for an entire year before migration started and the results were very strong—we were able to retire half of our SharePoint sites before we even started migration. Most were rebuilt as part of our start fresh initiative and the rest were completely shut down because new projects or teams had superseded them and no one was using them.
Getting security right was a challenge throughout the migration. We did not allow high-security sites to be migrated until we proved that we could get it right with low- and medium-security sites. We quickly proved that SharePoint Online is safe, and now we support all levels of confidentiality. Now, when employees want a new team collaboration space or a publishing site, they are all created on Office 365 unless the employee applies for an exception.

We also had the challenge of personal spaces to contend with. Our employees used MySite to share social information about themselves and to share content. In the lead-up to migration, we let employees know that their MySite would be retired within four months and gave them instructions on how to move their content to their new SkyDrive Pro/OneDrive for Business profile sites.

Over time, MySites were set to read-only to give employees time to retrieve forgotten content. Now, OneDrive for Business adoption is far higher than we ever had with MySites. Altogether, our employees have stored more than 400 terabytes of content on OneDrive for Business versus 8 terabytes previously stored on MySites.

“Lift and shift” approach

After subtracting all of the fresh starts, shutdowns, staying on-premises, and custom portals, we were left with about 25,000 utility (self-service) sites that we needed to migrate.

We call the technique we used for migrating these sites “lift and shift,” named for lifting a site as-is and then shifting it to the cloud with the same capabilities and the same look and feel. Our lift and shift migration started in July 2013 and finished in February 2014. When we started, we were migrating just a few sites per week, but by the time we wrapped up eight months later, we were clearing more than 1,000 sites per week.

Every migration needs a good governance plan and we spent a lot of time marking sure we got ours right. It had three parts:

1. Building a team to run the migration.
2. Creating a database to plan the work.
3. Crafting an execution plan to do the work.

Building the team

We built a team of 11 people to run our migrations of the 25,000 utility site collections. Here’s what it looked like:

- We had a team leader from IT to guide the project and a leader from the SharePoint product group to experiment with new product API patterns that might be helpful. This is something we baked into the migration process as much as we could. It’s also something third-party vendors can help with.
- Four IT engineers ran the migrations and wrote repeatable processes that did much of the work. They had to write new processes or modify others each time a new challenge surfaced. They needed to be able to crawl SharePoint site data and convert it into formats that would allow migration. The engineers were scattered around the globe, which allowed us to have 24-hour coverage. While Microsoft IT used custom processes, much of the work pipeline and APIs that they were experimenting with are now part of third-party migration tools with updated product APIs to support them.
- We pulled in four customer service specialists to support site owners when breakdowns occurred.
- Our communications specialist wrote and sent email communications and FAQs that walked site owners through their migration plan. Communications included a “why” mail from a leader, a mail explaining how the migration would work, a mail that let the site owner it was time to launch, and a wrap-up note when done. Communications escalated when issues popped up.
Creating an inventory database
We loaded sites into a database that we used to categorize and sort them. The more we could assess, divide, and group like sites together, the easier it was to create repeatable processes that would let us understand how to handle all of their unique attributes. To inventory sites, we recommend:

**Step 1:** Get an inventory of the number of sites that you have.

**Step 2:** Filter that inventory for things that would not migrate with feature parity without rework, for example, sites using PowerPivot business intelligence.

**Step 3:** Use that inventory to help schedule when you’re moving what sites, based on whether they are large, classification, template used, etc.

Executing the work
- **Daily scrums:** We held scrums every day for nine months, which allowed us to nimbly respond to issues and surprises.
- **Rescheduling:** Our migration schedule depended on being able to migrate sites as scheduled, as much as possible. We tried to resolve scheduling issues before the week a site was scheduled to migrate. A site owner was only allowed to reschedule twice to limit stalling.
- **Answering questions:** We sent out emails to guide site owners through migration, and also sent them to an internal site where they could search for answers to their questions, walk through Productivity Guides (handy how-to guides that walk customers through our technology), and see the questions that other site managers asked when they were going through their migration.
- **Doing the work:** There are several partner-built migration tools available. We experimented on our own repeatable processes in addition to using third-party tools to learn where to improve the core product migration pipeline as well as to meet our migration goal. Since we completed our migration, those lessons have made it into the product and our third-party products.
- **Rollbacks:** When a migration didn’t work, we rolled the site back to its on-premises location by reopening it from its previous read-only state and then figuring out what went wrong. We had problems with approximately 10 percent of our migrations but were able to migrate most of those by reworking our processes. In the end, we were unable to migrate about 3 percent of our sites (see below for why).

Getting rolling
- **Pacing ourselves:** We started by migrating just five sites. We had to see if our tools were robust enough before we ramped up. We tried to pick a variety of sites to start with, including very large and very small sites. We gradually scaled up until we were migrating 1,200 sites per week. We were successful 97 percent of the time.
- **Challenges popped up:** Some sites had functionality that couldn’t be moved to the cloud. Others were regional and were affected by regulations that required them to remain in that region, versus in a central cloud tenancy. Conflicting URLs were an issue, as multiple on-premises sites were moved into a single tenancy—diplomacy had to be used when teams fought over the same easy-to-remember URL.
- **Site customizations:** With both on-premises and cloud SharePoint sites, we allowed employees to customize as much as our new cloud-based UI allowed. That included master page layouts, list schemas and views, and InfoPath forms and workflows. Support would try to fix issues, but if they couldn’t resolve them, the site owner was asked to undo problematic changes or get an engineering team involved. During migration, some customizations did not migrate cleanly. We sometimes had to create steps to account for new situations that we hadn’t foreseen.

Here are the most common customization challenges that we ran into:

- We found that forms and workflows that supported critical workloads were hard to migrate without business impact, especially when dealing with work that couldn’t be interrupted—like our approvals system. We could support critical workflows in the cloud, but the migration process had to be managed carefully with the site owners.
• InfoPath (a tool that helps people build and complete forms) was a special kind of headache that mostly popped when a team was using an older version of SharePoint. Some sites had very complex, multilayered InfoPath forms. SharePoint on-premises let you design forms how you want to, but the online version limits how many layers you can create in InfoPath forms. A few of these were so complex that we kept them on-premises.

• We had many challenges with hardcoded links to images and other content. The site administrators had to individually fix each hardcoded link after migration. Relative links were fine, and while SharePoint would automatically link content as it is moved, moving to the cloud required that we deal with these renamed links.

• Large and wide lists posed a challenge (i.e., lists with more than 5,000 items or a highly complex schema). SharePoint on-premises sites had exceptions in place to allow expansive queries, and these exceptions were not available in Office 365. While the cloud supported large lists, complex or large views required rework before migration to add in indexes or views.

• Sites with large local term stores were also a challenge to migrate to the cloud as the term set references needed to be remapped manually. That meant these sites also had to remain on-premises until remapping could happen.

Telling our story with custom portals

At Microsoft, SharePoint is the main way we communicate with our employees—all of our major intranet portals and employee experience web sites are built on custom SharePoint solutions. MSW, the company's employee portal was built in SharePoint, as were portals for Human Resources, the company library, finance, legal, IT, sales, internal video sharing, and many others.

We started our migration with about 240 custom portals and SharePoint-based, custom solutions on-premises. For us, any SharePoint site that uses full-trust solutions on SharePoint servers or is important enough to need elevated support is considered a custom portal for migration.

Before migration, custom portals lived on dedicated SharePoint farms, isolating them from our larger, self-service, employee-run sites. Refactoring our custom solutions and portals in Office 365 (using new SharePoint application models to decouple SharePoint from the solution) allowed us to move them to the cloud without affecting other sites.

Our core goal was to make sure these large, complex sites did not lose any functionality when we moved them to the cloud. They needed to be just as powerful as before. We also wanted to position them so that they could take advantage of new capabilities that came with moving to the cloud.

Establishing guiding principles for building a SharePoint portal in the cloud

1. Encouraged site owners to limit the number of customizations they made to keep cost down and retain business agility.

2. Invited site owners to use our new dynamic rendering capabilities to put their content on PCs, tablets, and phones.

3. Built and customized a community-shared responsive design template as a foundation for site owners to build on. This minimized custom page development, gave our intranets some fundamental consistency, and allows us to centrally respond to product and business changes with one package as opposed to changing each portal individually.

4. Established standardized content schemas to make it easier for site owners to publish content in a consistent manner (which improves searching and aggregation across multiple sites). We intentionally separated content design from UI elements so our customers could focus on building great content (not site design).

Building a consistent UI across all portals

Much like we wanted to simplify the way we delivered content on our portals, we simplified the look and feel of our mastheads, our pages, and all of the UI elements of the new SharePoint experience by using the Office 365 standard suite navigation. This concept is extended as you move to the cloud via SharePoint Online, but it's also a big part of
the newer versions of SharePoint itself. Equally important, all sites have the same basic navigation, so using them is similar no matter what site you visit.

All of Microsoft, whether Office 365, Windows 10, or Microsoft Azure, is taking the same track. Why? So that customers are not surprised when they go from portal to portal, site to site, or product to product. If you build consistent experiences, customers can focus on the work they need to do instead of stumbling from tool to tool, and this helps them be more productive.

**Replicating our taxonomy structure in the cloud**

Microsoft has a standard corporate taxonomy for things like products, countries/regions, languages, etc., that helps us consistently tag and target content. Part of moving to the cloud meant replicating our on-premises SharePoint taxonomy service in the cloud.

To provide the same consistency for our corporate portals and divisional sites, we created enterprise content-types (schemas) for people to tag against. For example, we created an enterprise news content-type with properties allowing site managers to tag content by region or role.

**Company portal shows the way**

The best way to show you what moving to the cloud was like for us is to look at how we moved our employee portal to the cloud. The MSW homepage gets about 1.1 million clicks per month, and the site is visited by nearly 175,000 visitors per month. MSW is our company intranet homepage—employees go there to get company news, learn about major announcements and events, find other internal sites, and search internally.

*Figure 2. We migrated MSW to the cloud without making many changes—those would come later. The main change is that we adopted dynamic rendering, which allowed content to show up well on all devices.*

Our plans to move MSW to the cloud sped up dramatically when CEO Satya Nadella sent us an email asking us why an employee couldn’t open an MSW news article on their phone without logging into the corporate network. Even though MSW was a custom migration, we sought to “lift and shift” the content and information architecture from the
portal so we could do it quickly while keeping all of its core functionality intact. At the same time, we had to update the UI templates as well as refactor the existing solutions. So we could move quickly and simply, we didn’t add much new functionality. Our plan was to add new features after the migration was complete.

Like most of our portals, the on-premises version of MSW was highly customized. Not only is the site the busiest portal at Microsoft, it’s where we keep core employee content. Employees can find everything from campus maps, to expense forms, to meal card balances. MSW is the gateway to get to all other major company sites, it is where employees search for internal content (think of it as our internal Bing), and it is where we publish news for employees.

Moving a giant without sacrificing quality
When it came time to move MSW to the cloud, we weren’t going to remove features that its managers determined were imperative. We had to make sure we didn’t reduce capabilities and quality, and all core features needed to be there.

The first thing we did was inventory exactly what we had with MSW. Here’s how to do it:

**Table 3. Assessing what you have**

<table>
<thead>
<tr>
<th>Type</th>
<th>Questions to ask</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content</td>
<td>Take an inventory of your portal’s content and determine what you are keeping. For MSW, we decided to separate out some content onto another site but retain most content intact for migration.</td>
</tr>
<tr>
<td>Information architecture</td>
<td>Will the structure change? We decided to retain the full MSW site structure and navigation.</td>
</tr>
<tr>
<td>Design</td>
<td>Are you keeping your on-premises design? We chose make MSW more responsive and mobile friendly, but didn’t redesign the look and feel (we saved that for post-migration).</td>
</tr>
<tr>
<td>Custom solutions</td>
<td>Analyze customizations and simplify where possible. Retire, refactor, and replace complexity with out-of-the-box tools or third-party solutions.</td>
</tr>
</tbody>
</table>

MSW migration kicked off in June 2014, and the cloud-based site went live in October 2014. A fast move was a key goal and we sacrificed making more fundamental design or information architecture changes to move fast. A best practice would be to use moving to the cloud as a chance to simplify, modernize, and adopt a mobile-friendly design. We only did the last of those three to move fast.

Mobility was a must—we needed to deliver dynamic rendering out of the gate so that all content would render seamlessly across all devices. We also needed to make the publishing process for articles, images, slideshows, and other form factors simpler and faster.

Not to be overlooked, we went to great lengths to make sure content showed up exactly as it had on-premises. We wanted each sub-page, article type, search results listing, and every other of the hundreds of pages that you could find on MSW to show up exactly like they used to on-premises.

How did we do it? Starting with our custom design package with a master pages and templates, we started by building out the framework for the site. Then, we used a third-party tool (provided by Metavis) to map content from on-premises layouts to the cloud version and then used the tool to help us perform the migration itself.

We managed to move all functionality to the cloud, even the very challenging maps program, which shows each building on the Redmond campus and those around the world. The only major feature we dropped was a localized weather report that we decided was too difficult to move to the cloud because of how it varied from place to place. And, besides, it is easy for our audience to get a localized weather report other places.
We inventoried every solution with full-trust, server-side code installed on SharePoint and all client-side solutions. We also had conversations with our stakeholders about what we had to keep, how we could carry forward any solutions that had to be refactored, and how we could simplify wherever possible.

Table 4. Covering our bases

<table>
<thead>
<tr>
<th>Category</th>
<th>What we did</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simplify</td>
<td>Move to out-of-the-box solutions without losing functionality</td>
<td>Custom roll-up web parts: Moved to content by search</td>
</tr>
<tr>
<td>Refactor</td>
<td>Moved from full-trust to SharePoint provider-hosted application</td>
<td>Glossary: A web part that read terms and definitions from SharePoint metadata service</td>
</tr>
<tr>
<td>Refactor</td>
<td>Moved from full-trust to Azure-hosted provider app</td>
<td>Campus maps: Mashup of Bing Maps layered with line of business geo-referenced landmark data such as buildings, cafes, and parking</td>
</tr>
<tr>
<td>Retire</td>
<td>Eliminate functionality completely</td>
<td>Weather app: Deemed not worth the cost to rebuild</td>
</tr>
</tbody>
</table>

Post-migration, pre-launch challenges

Once we migrated MSW, latency was a major challenge. The on-premises version of MSW was supported by three dedicated servers, which meant highly dynamic content (such as new homepage stories) surfaced super-fast because of page caching from those servers. Caching isn’t as viable in the cloud, where you have potentially hundreds of content servers and a far greater variety of sites. When we first moved MSW to the cloud, it could take pages 20 seconds to load; our on-premises average was 2 seconds. By tuning the portal to minimize server-reliant rendering, we reduced page load times. To get there, we changed two major things—dynamic content rendering and navigation.

Content by query replacement

We moved away from our content by query web part, which was a favorite of ours since MSW launched in 2006. This feature dynamically pulled content from libraries across the site. There were seven content query web parts on the MSW homepage when initial migration testing began. Instead of trying to replicate that, we used the SharePoint Online new content by search solution to simply load content from the search index—or we customized a client-side list view of web parts in cases where search latency wasn’t acceptable. For example, for real-time news on the homepage, we decided to use list view to make content show up in minutes instead of the 15 minutes that traditional search indexing could take.

Updating our structural navigation replacement

The MSW SharePoint on-premises portal had around 1,000 subsites because every topic area was its own site. We had used structural navigation to render all of those sites, because in an on-premises portal we could use our three dedicated servers to do real-time iteration on all of the subsites. That server-side rendering wasn’t something we wanted to depend on when we moved to the cloud. Instead, we shifted to managed navigation, where the navigation was pre-defined in the metadata store. For most internal portals, that is perfectly acceptable because the portals don’t have a lot of security trimming in the navigation. For those that do, we used a more search-based navigation approach.

See aka.ms/tune for more information about tuning SharePoint and sites for Office 365.
Benefits of moving to the cloud

- **IT as a Service:** IT was able to move from being a service runner (patching machines, managing service uptime, managing storage, etc.) to being a business enabler. Our focus is now on empowering our businesses and employees to get business value and to enable rapid solution development to achieve business results.

- **Rethink opportunity:** Site and portal owners at Microsoft were encouraged to use migration as an opportunity to start fresh so they could build their SharePoint site(s) just the way they would like, without rushing or building something they wouldn't use or need.

- **Anywhere on any device:** Employee adoption of SharePoint services and personal storage increased after moving to Office 365 when our employees could work when and how they needed to on any device. When they were no longer tied to specific machines, employees responded by broadly using OneDrive for Business (and its precursor). They moved significant content off their personal drives onto core team sites, groups, and OneDrive for Business sites. They took advantage of new features and capabilities that came with moving to the cloud, including moving to dynamic rendering that allowed them to publish content once to PCs, tablets, and phones.

- **Service simplification:** We simplified our environments greatly; instead of having multiple disparate SharePoint farms for utility services and for each major customized portal farm, we moved to a single Office 365 tenancy and consolidated the remaining sites on-premises as much as possible. For example, we eliminated separate MSW, divisional, and regional MySite farms, saving engineering effort and hosting costs.

- **Cloud governed:** To help us manage our cloud sites, we built a custom IT site management solution we called AutoSites. This allowed us to better manage site proliferation by requiring a site manager to actively decide that their site is still useful and should be preserved. It also required each site to have a site owner who is with the company, which prevents leaderless sites from floating out there, unmanaged.

- **Technical agility:** We stopped managing our employees’ storage needs, instead leaving that to our cloud service. When we moved to the cloud, our employees went from collectively storing eight terabytes of MySite content on-premises to more than 350 terabytes of data on OneDrive for Business in the cloud. In the past, we would have had to provision disks of new storage capacity to cover all that new use—not anymore.

- **Simplified solution isolation and deployment:** Since business solutions can’t step on each other in the cloud, nor negatively impact the SharePoint farms themselves, our internal customers are able to work with us to get new applications into a central internal application catalog all backed by one Office 365 tenancy—as opposed to separate, on-premises farms for each application.

- **Improve security:** Because SharePoint Online is more secure than on-premises versions of SharePoint, we were able to make the way our employees and contingent staff store and save data much more secure. Significant security enhancements have been rolled into SharePoint Online. Bringing that to on-premises SharePoint farms would be difficult and expensive.

- **Rapid capability evolution:** We were able to move our custom portals over to the cloud with the vast majority of on-premises capabilities intact. Once migrated, we worked with portal owners to use new capabilities afforded by the cloud. For example, sites were able to take advantage of Delve (the Microsoft personalized, internal searching tool) and users could easily discover content shared with them—features that didn’t exist on-premises.

- **Cost reduction:** Our goal was to reduce operation costs while adding new capabilities, including enabling access anywhere on any device, improving look and feel, and using simpler, consistent publishing. We largely were able to do this while also retiring dedicated SharePoint farms, which saved hosting and service engineering costs.
Migration best practices

- Start with simple workloads. Enabling OneDrive for Business or simple team sites starts the migration process without investing in full migration.
- Make sure you have a life-cycle management process in place for SharePoint on-premises environments to help determine what is actually needed and used. Move only what’s needed.
- Anywhere, any-device access empowers employees to work when and where they need to. Getting employees to move their content out of the shadows and into Office 365 cloud services allowed us to better protect the company assets that our employees stored on their SharePoint sites.
- Keep experiences simple and future-proof. Moving to Office 365 offers the opportunity to evolve as our platform evolves, giving you flexibility to take full advantage of the platform without boggling it down.
- Compelling intranets can be built in Office 365, including customized applications, by leveraging the Office 365 and Azure application platforms.

For more information

Microsoft IT Showcase
microsoft.com/ITShowcase

Microsoft IT shares cloud migration insights

Optimizing network performance for Microsoft Office 365
Reinventing the modern business ecosystem
How Microsoft IT manages and governs the internal SharePoint environment
How Microsoft IT uses Microsoft SharePoint and Office 365

Check for roundtable SharePoint discussions
SharePoint Online pricing
SharePoint product site: Learn the new way to work together

© 2016 Microsoft Corporation. All rights reserved. Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. The names of actual companies and products mentioned herein may be the trademarks of their respective owners. This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY.