



▶ SHAREPOINT POR KO: KNOCKOUTJS	3
▶ COLECCIONES DE SITIOS "HOST-NAMED" EN SHAREPOINT 2013	9
▶ LA COMUNICACIÓN Y EL ROL DE LA INTRANET CORPORATIVA (IC).....	12
▶ 10 PUNTOS PARA ENTENDER LA GESTIÓN ESTRATÉGICA DEL PORTFOLIO DE PROYECTOS EN PROJECT SERVER 2010.....	14
▶ MADPOINT. GRUPO DE USUARIOS DE SHAREPOINT DE MADRID.....	17
▶ ENTREVISTA CON ALBERTO DIAZ MARTIN.....	21
▶ CAMPO PARA GEOLOCALIZACIÓN EN SHAREPOINT 2013.....	25
▶ CONEXIÓN DE WEBPARTS DE FILTROS Y CUADRO DE MANDOS DE PERFORMANCEPOINT MEDIANTE PROGRAMACIÓN.....	27
▶ NOVEDADES EN SOCIAL ENTERPRISE DE SHAREPOINT 2013.....	31
▶ EXPLOTANDO LOS SERVICIOS DE EXCEL DESDE APLICACIONES EXTERNAS MEDIANTE SUS SERVICIOS WEB.....	33
▶ NOVEDADES DE PROJECT SERVER 2013.....	38
▶ CASO PLAN CEIBAL.....	42
▶ COMPARTIR APLICACIONES DE SERVICIO ENTRE GRANJAS SHAREPOINT 2010.....	45

Vol. 13
Septiembre 2012

Español

Comparti MOSS

REVISTA ESPECIALIZADA EN
TECNOLOGIAS SHAREPOINT

Bienvenidos a CompartiMOSS, la primera revista electrónica sobre SharePoint dirigida al mundo de habla hispana.

Editorial

16 de julio del 2012: Microsoft presenta su nueva versión de Office...

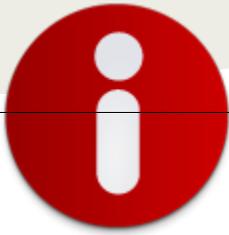
(Continúa en la primera página)



En esta Edición:

- ✓ Colecciones de Sitios "Host-Named" en SharePoint 2013
- ✓ Campos para Geolocalización en SharePoint 2013
- ✓ Novedades en Social Enterprise de SharePoint 2013
- ✓ Novedades de Project Server 2013
- ✓ SharePoint por KO: KnockoutJS
- ✓ Más, mucho más.





Staff

CompartiMOSS es una publicación independiente de distribución libre en forma electrónica. Las opiniones aquí expresadas son de estricto orden personal, cada autor es completamente responsable por su propio contenido.

Dirección General

- ✓ Gustavo Vélez
- ✓ Juan Carlos González
- ✓ Fabián Imaz

Contáctenos

gustavo@gavd.net
jgonzalez@gruposodercan.es
fabiani@siderys.com.uy

Blogs

<http://www.gavd.net>
<http://geeks.ms/blogs/ciin>
<http://blog.siderys.com/>

Facebook

<http://www.facebook.com/group.php?gid=128911147140492>

Escríbanos:

compartimoss@hotmail.com

Visítenos:

http://www.gavd.net/servers/compartimoss/compartimoss_main.aspx

Editorial

16 de julio de 2012: Microsoft presenta su nueva versión de Office, incluyendo a SharePoint 2013, cumpliendo con el ciclo de renovación de productos que usualmente se da cada tres años. Aunque con sustanciales modificaciones, SharePoint 2013 es de nuevo una evolución con respecto a SharePoint 2010, pero está lejos de ser una revolución. Principalmente tres campos han sido modificados o añadidos a la plataforma: extensas modificaciones en la Característica de Publicación, reemplazo de los diferentes tipos de motores de búsqueda por una versión unificada y el marco de trabajo de Aplicaciones.

Otra multitud de aspectos han recibido también actualizaciones: algunas modificaciones en los componentes "sociales", cambios en aspecto y usabilidad, algunos componentes nuevos (para "Educación", por ejemplo), pero, como ya lo mencionamos anteriormente, estos son los cambios que se pueden esperar para adaptar el servidor a las exigencias de las empresas modernas, no una revolución.

En CompartiMOSS estamos siempre en la línea de frente en cuanto a innovaciones, por lo que empezamos directamente en este número con artículos sobre SharePoint 2013. Como la nueva versión no se espera comercialmente sino hasta dentro de unos meses, por varios números seguiremos publicando artículos sobre SharePoint 2010, reflejando la situación que nos tocara vivir como profesionales de SharePoint: por algún tiempo, inclusive algunos años, tendremos que convivir con ambas versiones.

Deseamos que puedan disfrutar –e implementar- las ideas presentadas en los artículos de la presente edición. Recuerden que en lo referente a los artículos sobre SharePoint 2013, se trata de una versión Beta que puede cambiar en la versión final. Pero que esto no ponga cortapisas a su curiosidad: descargue e instale a SharePoint 2013. Sin duda le gustará...

Septiembre, 2012

Gustavo Vélez – gustavo@gavd.net

Juan Carlos González jgonzalez@gruposodercan.es

Fabián Imaz - fabiani@siderys.com.uy





SharePoint por KO: KnockoutJS

... Knockout, una de las nuevas librerías de JavaScript nos permite dotar a SharePoint de un modelo de programación MVVM (Model-View-ViewModel)...

Los tiempos están cambiando para los programadores de SharePoint. Con la llegada de la versión Preview de SharePoint 2013, se ha puesto de manifiesto que tenemos que pulir considerablemente nuestras habilidades de programación en JavaScript. El nuevo modelo de desarrollo de apps de SharePoint 2013 casi impone hacer los desarrollos en JS con la ayuda del modelo de objetos de cliente (JS CSOM) o bien de la interfaz REST de SharePoint.

Para ayudar a mejorar en las habilidades de JS con SharePoint, en este artículo os enseñaré como Knockout, una de las nuevas librerías de JavaScript nos permite dotar a SharePoint de un modelo de programación MVVM (Model-View-ViewModel), más dinámico y que nos permite hacer interfaces muy usables.

¿Qué es Knockout?

KnockoutJS (o Knockout a secas) es una librería para crear interfaces dinámicas con JavaScript siguiendo el patrón MVVM, semejante a como lo hace Silverlight o WPF. La característica principal del patrón MVVM es que en vez de usar el modelo de datos directamente, se usa un modelo ajustado a cada vista (de allí el nombre ViewModel) donde mediante bindings se ligan las propiedades del modelo con los controles de la vista. Estos bindings suelen ser bidireccionales para mejorar la usabilidad de la vista. El "pegamento" que hace posible este enlace es Knockout.

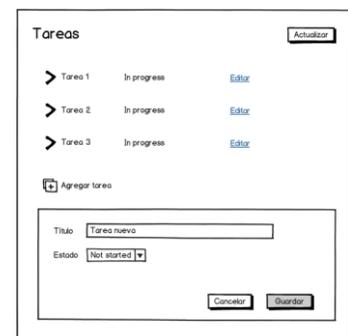
Vamos a ver un ejemplo sencillo de Knockout para construir un pequeño gestor de tareas de SharePoint. Primero usaremos datos "hardcoded" y luego usaremos la interfaz REST de SharePoint para obtener los datos y posteriormente modificarlos.

Para usar Knockout, tenemos que bajarnos el fichero JS desde la página de descargas y referenciarlo desde nuestro HTML.

```
<script src="Scripts/knockout-2.0.0.js" type="text/javascript"></script>
```

El tablón de tareas

La idea de este artículo es tener un pequeño tablón de tareas de una lista de tareas en SharePoint, para poder verlas, modificarlas y añadir nuevas tareas sin ejecutar nada en el lado servidor. La maqueta de lo que queremos conseguir es esta:



Como se ve, habrá una lista de tareas, mostrando su título y su estado, junto con el botón de edición de cada una de las tareas. Habrá también una zona para crear o modificar la tarea, con los controles de edición y botón de guardar.

El ejemplo completo se puede bajar desde mi blog, y en este artículo iré enseñando las piezas clave de la solución.

La vista

Empezamos con un HTML estático sencillo para conseguir la vista. El fragmento de HTML para mostrar las tareas es una lista normal (elemento UL) con varios SPAN.

```
<div id="taskListPanel">
  <ul>
    <li>
      <div class="task-item">
        <span class="taskTitle">Titulo</span>
        <span>Estado</span>
        <a href="#">Editar</a>
      </div>
    </li>
  </ul>
</div>
```





El ViewModel

Como modelo de vista usaremos un array de objetos de JavaScript llamado `datosTareas`, con las propiedades `Id`, `Title` y `Estado`.

```
var datosTareas = [  
  { Title: "Tarea 1", Id: 1, Estado: "Not  
started" },  
  { Title: "Tarea 2", Id: 2, Estado: "In  
progress" },  
  { Title: "Tarea 3", Id: 3, Estado:  
"Completed" }  
];
```

Para convertir este array estándar de JS en un modelo de Knockout, tenemos que usar la función `observableArray()` de Knockout para inicializar una colección observable (la que permitirá el binding con la vista). Esta función toma un objeto y lo convierte en un "observable" de Knockout.

Para nuestro caso usaremos una función llamada `ViewModel()` a la que pasaremos un parámetro array de objetos. Se expondrá una propiedad llamada `tareas`, que usaremos para binding.

```
var ViewModel = function (elementos) {  
  var self = this;  
  self.tareas =  
    ko.observableArray(elementos);  
};
```

Si se trata de una propiedad que no sea array, se usaría la función análoga `observable()`.

El enlace de datos con Knockout

El primer paso es decorar los elementos HTML que queremos enlazar con datos mediante el atributo `data-bind`. En el caso de los elementos `SPAN`, usaremos la sintaxis `text: Nombre`, que cargará el valor textual de la propiedad `Nombre` de nuestro view model.

Para que la lista `UL` se rellene con los datos del array observable, usaremos la sintaxis `foreach: Nombre` que rellenará el objeto tantas veces como objetos haya en el array. Para cada elemento se hará un binding individual con los valores actuales del objeto.

```
<div id="taskListPanel">  
  <ul data-bind="foreach: tareas">  
    <li>  
      <div class="task-item">  
        <span class="taskTitle" data-  
bind="text: Title"></span>  
        <span data-bind="text:  
Estado"></span>  
        <a href="#">Editar</a>  
      </div>  
    </li>  
  </ul>  
</div>
```

Al cargar la página, queremos realizar el binding de los datos de las tareas. Para ello tenemos que construir el view model usando la función `ViewModel()` explicada anteriormente y luego indicar a Knockout que haga el binding mediante la función `ko.applyBindings()`.

En este caso usaremos jQuery con su evento de `document.ready`, aunque se podría hacer con la función nativa de SharePoint `executeOrDelayUntilScriptLoaded`.

```
$(function () {  
  var vm = new ViewModel(datosTareas);  
  ko.applyBindings(vm);  
});
```

Como resultado de la ejecución del binding, se muestran los datos de las tres tareas "hardcoded".

Tareas



El mapeo

Hasta ahora hemos usado la declaración automática de las propiedades del view model de la tarea, usando la función `ko.observableArray()`. En la vida real, los objetos de nuestro view model no son una mera copia del modelo de datos real. Seguramente habrá propiedades calculadas o añadidas en el view model y no todas las propiedades del modelo real serán visibles en el view model.





En nuestro caso añadiremos una propiedad más al view model, llamada `IsInEdit`, que nos indicará si una tarea está en edición en la vista o no. Ya que estamos cambiando las maneras de realizar el mapeo, usaremos una extensión de Knockout llamada `ko.mapping` que nos ayudará a escribir menos código repetitivo.

En el view model original, declaramos la propiedad `tareas` como un array vacío y la rellenamos mediante un método reusable llamado `getTareas`. Este método usa la extensión `ko.mapping` para construir los objetos observables a partir de los datos originales y un mapeo. Para usarla, tenemos que referenciar a otro fichero JS:

```
<script src="Scripts/knockout.mapping-latest.js" type="text/javascript"></script>
```

La operación que nos interesa es `ko.mapping.fromJS` que toma tres parámetros: el objeto origen, el mapeo a aplicar y el objeto destino donde se creará el objeto observable.

```
var ViewModel = function (elementos) {
    var self = this;
    self.tareas = ko.observableArray();

    self.getTareas = function () {
        ko.mapping.fromJS(elementos,
            mapping, self.tareas)
    };
};
```

Aquí declaramos el objeto de mapeo. La propiedad interesante en este momento es la `create`, que define como se crearán los nuevos objetos observables a partir del objeto de modelo. En este mapeo usaremos la función auxiliar `CrearTaskViewModel` al que le pasamos la propiedad `data` que contiene el objeto JS a partir del cual se realiza el mapeo.

```
var mapping = {
    'create': function (options) {
        return new
        CrearTaskViewModel(options.data);
    }
};
```

Declaramos una función `CrearTaskViewModel` en la que usaremos la operación `ko.mapping.fromJS` para crear un objeto observable a partir de un objeto JS, pasándole un mapping vacío para que haga el mapeo de todas las propiedades del objeto. Además, definiremos la propiedad nueva `IsInEdit` y añadiremos un método para cambiar el

estado de edición de la tarea, llamado `toggleEditar`.

```
var CrearTaskViewModel = function (data) {
    var self = this;
    ko.mapping.fromJS(data, {}, self);

    self.IsInEdit = ko.observable(false);

    self.toggleEditar = function () {
        self.IsInEdit(!self.IsInEdit());
    };
};
```

Cambios en la vista

Para acomodar los cambios en el view model, primero vincularemos el botón Actualizar para que invoque la operación `getTareas` del view model, usando la opción `click` del parámetro `data-bind`. Esto vinculará el clic en el botón con una función del view model.

Añadiremos un control `input` textual a la visualización de la tarea que se mostrará si la tarea está en modo edición, mientras que el `span` del título de la tarea se mostrará si la tarea no está en modo edición. Para ello en el parámetro `data-bind` especificaremos la opción `visible`.

Aprovechamos para mover el HTML estático a una web part de editor de contenido en modo XHTML. Los ficheros JS y CSS los subiremos a la biblioteca de estilos del sitio en el que hemos insertado la web part.

```
<div id="taskPanel">
    <h1>Tareas</h1>
    <button id="btnActualizar" data-
bind="click: getTareas">Actualizar</button>
<div id="taskListPanel">
    <ul data-bind="foreach: tareas">
        <li>
            <div class="task-item">
                <span class="taskTitle" data-
bind="text: Title, visible:
!IsInEdit()"></span>
                <input type="text" data-
bind="text:Title, visible: IsInEdit()" />
                <span data-bind="text:
Estado"></span>
                <a href="#" data-bind="click:
toggleEditar">Editar</a>
            </div>
        </li>
    </ul>
</div>
```





Interacción con SharePoint

Ahora nos falta sustituir los datos "hardcoded" por los datos de SharePoint reales provenientes de una lista de tareas.

Para obtener los datos, modificaremos el método `getTareas` de nuestro view model para realizar una llamada Ajax hacia la interfaz REST de SharePoint (el fichero `ListData.svc`). Añadiremos una nueva propiedad llamada `lista`, que contendrá el nombre de la lista en SharePoint.

```
self.lista = ko.observable("Tareas");
self.getTareas = function () {
    var ajaxUrl =
        _spPageContextInfo.webServerRelativeUrl +
        "/_vti_bin/listdata.svc/" + self.lista();
    $.ajax({
        type: "GET",
        url: ajaxUrl,
        contentType: "application/json;
charset=utf-8",
        dataType: "json",
        cache: false,
        processData: true,
        success: function (data, status, xhr)
        {
            if (status == "success" && data) {
                ko.mapping.fromJS(data.d.results, mapping,
                    self.tareas)
            },
            error: alert
        });
    });
};
```

Al obtener los datos de SharePoint, tendremos un objeto JSON con las todas propiedades de los elementos de la lista. Modificaremos la función `CrearTaskViewModel` para mapear las propiedades de SharePoint con las del view model, generaremos la colección de observables que necesita nuestro modelo. Al trabajar con la versión de SharePoint en español, las propiedades `Title` y `Estado` del view model se llaman `Título` y `EstadoValue` en el objeto JSON.

```
var CrearTaskViewModel = function (data) {
    var self = this;
    ko.mapping.fromJS(data, {}, self);

    self.Title = ko.observable(data.Título);
    self.Estado =
        ko.observable(data.EstadoValue);

    self.IsInEdit = ko.observable(false);

    self.toggleEditar = function () {
        self.IsInEdit(!self.IsInEdit());
    };
};
```

Ahora, veremos el resultado de los cambios. Los datos ahora vienen directamente desde SharePoint.

Lista de tareas en SharePoint



Lista de tareas en nuestro panel de tareas con Knockout





Edición y creación de nuevas tareas

Hasta ahora todo lo que hemos usado se ha limitado a mostrar las tareas, pero no las hemos editado. Ahora haremos los cambios pertinentes para que se puedan crear y modificar las tareas y eso se lleve a cabo en SharePoint, con la ayuda de Knockout y jQuery.

Cambios en el view model

Añadiremos también la función nuevaTarea que agregará una tarea vacía con identificador 0 a la colección de tareas observables, y la pondremos en edición.

```
self.nuevaTarea = function () {  
    var tarea = new CrearTaskViewModel({  
        Título: "", Identificador: 0 });  
    tarea.IsInEdit(true);  
    self.tareas.push(tarea);  
};
```

Cambios en el mapeo

Para evitar que el plugin de mapping de Knockout cree objetos repetidos, le indicaremos a la función de mapeo la propiedad que identifica las entidades observables, mediante la opción key. La función auxiliar ko.utils.unwrapObservable se usará para evitar devolver un observable (es decir, devolverá siempre el valor subyacente original).

```
var mapping = {  
    'key': function (data) {  
        return  
ko.utils.unwrapObservable(data.Id);  
    },  
    'create': function (options) {  
        return new  
CrearTaskViewModel(options.data);  
    }  
};
```

Cambios en la función CrearTaskViewModel

Añadiremos una nueva propiedad NombreAccion para reflejar el nombre de la acción, que será "Editar" o "Guardar", dependiendo de si estamos en edición o no.

```
self.NombreAccion = ko.computed(function  
( ) {  
    return self.IsInEdit() ? "Guardar" :  
"Editar";  
});  
  
self.toggleEditar = function (elemento,  
evento) {  
    var estaEnEdit = self.IsInEdit();  
    if (estaEnEdit) {  
        self.guardar(elemento, evento);  
    }  
    else {  
        self.IsInEdit(!self.IsInEdit());  
    }  
};
```

Cambios en la vista

Añadiremos un botón nuevo que invocará la función nuevaTarea del view model.

Guardar datos en SharePoint

Usaremos el mismo método Ajax de jQuery pero esta vez en vez del método GET será un POST. Determinaremos mediante la propiedad Id de la tarea si es una tarea nueva (Id = 0) o es una tarea a modificar (Id > 0). Para enviar los datos a la URL de REST de SharePoint, usaremos el parámetro data y lo llenaremos con las propiedades que queremos modificar.

Para las ediciones añadiremos en la cabecera de la función Ajax dos parámetros: If-Match con la propiedad ETag de la tarea recibida desde REST (para identificar el elemento a modificar) y X-HTTP-Method con el valor MERGE para evitar hacer un POST de todas las propiedades de la entidad. En el caso de creación de tareas nuevas, se usará un POST sin MERGE ni ETag.

```
self.guardar = function (elemento, evento) {  
    var element = evento.currentTarget;  
    var context = ko.contextFor(element);  
    var $root = context.$root;  
    var ajaxurl =  
_spPageContextInfo.webServerRelativeUrl +  
"/_vti_bin/listdata.svc/" + $root.lista();
```





```
if (self.Id() > 0) {
    ajaxurl += "(" + self.Id() + ")";
    $.ajax({
        type: "POST",
        url: ajaxurl,
        contentType:
"application/json; charset=utf-8",
        dataType: "json",
        beforeSend: function (xhr) {
            xhr.setRequestHeader("If-
Match", self.__metadata.etag());
            xhr.setRequestHeader("X-
HTTP-Method", 'MERGE');
        },
        data: JSON.stringify({ Título:
self.Title() }),
        cache: false,
        processData: true,
        success: function (data,
status, xhr) {

            if (status == "success") {
                $root.getTareas();
            }
            self.IsEditing(false);
        },
        error: function (xhr, status,
statusText) {
            alert(xhr.responseText);
        }
    });
}
else {
    $.ajax({
        type: "POST",
        url: ajaxurl,
        contentType:
"application/json; charset=utf-8",
        dataType: "json",
        data: JSON.stringify({ Título:
self.Title() }),
        cache: false,
        processData: true,
        success: function (data,
status, xhr) {

            if (status == "success") {
                $root.getTareas();
            }
        },
        error: alert
    });
}
};
```

Vamos a ver ahora los resultados de estas últimas modificaciones:

Edición de una tarea

Creación de una nueva tarea

Conclusión

En este artículo hemos visto que con unas pocas líneas de código en JavaScript podemos crear interfaces más dinámicas y usables aprovechando la inmediatez de la respuesta del modelo de programación en el navegador. El mundo de programación de JavaScript ha madurado mucho en los últimos años y ya está siendo hora de ponernos al día con las librerías más comunes para aplicarlas a nuestros desarrollos en SharePoint.

Edin Kapic

Key Consultant, Pasiona Consulting S.L.

<http://www.pasiona.com>

<http://www.edinkapic.com>

<http://spblogedin.blogspot.com>

@ekapic





Colecciones de Sitios "Host-Named" en SharePoint 2013

... se detalla el mecanismo de configuración de colecciones de sitios "Host-Named" (HNSC, de las siglas en inglés Host Named Site Collections)...

Nota: El contenido del presente artículo se basa en la versión "Public preview" de SharePoint 2013, por lo que es posible que en la versión RTM Microsoft pueda introducir cambios en torno a la configuración y uso de colecciones de sitios "Host-Named" en SharePoint 2013.

En este artículo se detalla el mecanismo de configuración de colecciones de sitios "Host-Named" (HNSC, de las siglas en inglés Host Named Site Collections) en SharePoint 2013 a través de la adición de nuevos comandos PowerShell que facilitan añadir múltiples URLs a una colección de sitios independientemente de que por aplicación web únicamente se dispongan de 5 zonas (Default, Intranet, Extranet, Internet y Custom) y sin necesidad de tener que configurar las rutas de acceso alternativo. Pero antes de empezar con cuestiones de configuración, no está de más tratar de responder a la pregunta: ¿Qué es un HNSC? Lo podemos definir como un mecanismo que permite consolidar aplicaciones web existentes en colecciones de sitios individuales, que puedan tener sus propias URLs "Top". El uso de HNSC proporciona ventajas como:

- Escalabilidad, ya que no aplica el conocido límite de 20 aplicaciones web por granja (y por lo tanto 20 sitios Top), pasamos a miles de colecciones de sitios que pueden tener sus URLs Top.
- Cada colección de sitios tiene asignado un nombre de DNS único.
- Se pueden seguir usando las rutas manejadas disponibles a nivel de aplicación web.
- Se soporta HTTPS en la definición de URLs Top.

En cambio, los HNSC introducen desventajas como:

- Sólo se pueden crear y configurar mediante PowerShell como sucedía en SharePoint 2010, aunque el proceso de configuración es más simple en SharePoint 2013.
- No se pueden usar las rutas de acceso alternativo, ya que los HNSC se definen por defecto en la zona "Default".

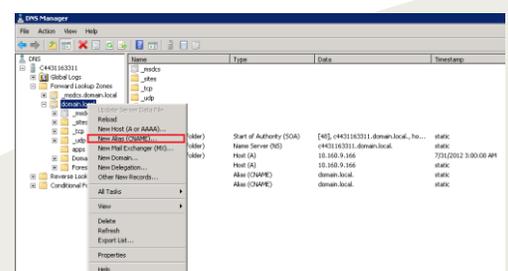
Aunque los HNSC son una opción a tener en cuenta cuando se necesita trabajar con URLs Top a nivel de colección de sitios, es importante tener en cuenta que hay ciertos escenarios en los que su uso se tiene que descartar:

- Se quiere usar la capacidad de "Self-Service Site Creation" para que usuarios no administradores puedan crear colecciones de sitios.
- Se requiere el uso de rutas administradas únicas del tipo "/sites" o "/sitios" a nivel de aplicación web.
- Se prima disponer de un aislamiento máximo a nivel de aplicación web en lo que a uso de pools de aplicaciones se refiere.

Preparación del entorno

Una vez que se han visto conceptos generales de los HNSC, a continuación se detallará como realizar su puesta en marcha comenzando con unos pasos previos de configuración en el entorno de SharePoint 2013 Public Preview:

- En primer lugar, el escenario de partida en el que se van a crear HNSC es una instalación de tipo granjas de SharePoint 2013 Public Preview con el correspondiente controlador de dominio configurado y habilitado, lo que permite agregar nuevos dominios o alias al dominio existente a través del "DNS Manager".
- Para añadir un nuevo alias, abrimos el DNS Manager, seleccionamos el nombre de nuestro dominio y hacemos clic con el botón derecho del ratón para pulsar sobre la opción "New Alias (CNAME)...".





- En la ventana que se abre, configuramos el alias especificando el nombre y como FQDN el dominio principal al que llegaremos navegando desde el botón "Browse" de manera que nos quede la siguiente configuración:

- Abrimos una ventana con el símbolo del sistema y ejecutamos "ipconfig/flushdns" para refrescar la configuración de DNS y que los cambios realizados tengan efecto en nuestro entorno. A continuación hacemos un ping a uno de los alias creado (intranet.domain.local en este caso):

```
C:\Users\Administrator> ipconfig /flushdns
Windows IP Configuration
Successfully flushed the DNS Resolver Cache.
C:\Users\Administrator> ping intranet.domain.local
Pinging domain.local [10.160.9.166] with 32 bytes of data:
Reply from 10.160.9.166: bytes=32 time<1ms TTL=128
Ping statistics for 10.160.9.166:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\Users\Administrator>
```

Creando un HNSC y agregando URLs Top

Una vez que a nivel de DNS se han realizado todas las configuraciones necesarias, ya estamos en disposición de crear y configurar nuestro primer HNSC lo que sólo es posible por medio de comandos Windows PowerShell:

- Abrimos la Consola de Administración de SharePoint 2013 y ejecutamos los siguientes comandos PowerShell:

```
New-SPSite "http://domain.local" -
HostHeaderWebApplication "http://c4431163311" -Name
"Portal" -Description "Portal de Clientes" -OwnerAlias
"domain/administrator" -language 3082 -Template "STS#0"
Set-SPSiteURL -Identity http://domain.local -URL
http://intranet.domain.local
Get-SPSiteURL -Identity http://domain.local
```

- El comando "New-SPSite" permite crear una colección de sitios de tipo HNSC añadiendo ya un Host-Name identificado en este caso por http://domain.local que es una URL de tipo Top y que se usará por defecto para acceder a la colección. El parámetro "HostHeaderWebApplication" identifica la aplicación web donde se está creando el HNSC. El resto de parámetros del comando son los habituales para crear una colección de sitios. Como resultado de la ejecución de este comando, se muestra por pantalla un listado con la URL del HNSC y la versión de SharePoint (15 o SharePoint 2013 en este caso).

```
PS C:\Users\Administrator> New-SPSite "http://domain.local" -HostHeaderWebApplication "http://c4431163311" -Name "Portal" -Description "Portal de Clientes" -OwnerAlias "domain/administrator" -language 3082 -Template "STS#0"
Url                                     CompatibilityLevel
-----
http://domain.local                     15
PS C:\Users\Administrator>
```

- Para añadir una nueva URL Top al HNSC basta con ejecutar el nuevo comando "Set-SPSiteURL". Este comando recibe como parámetros la URL del HNSC al que se le va a agregar una nueva URL Top y a continuación la(s) nueva(s) URL(s) a añadir que tienen que estar definidas como entradas (dominios o alias) en el DNS.
- Para obtener las URLs Top que se han agregado a un HNSC, utilizaremos el comando Get-SPSiteURL indicando cuál es el HNSC para el que queremos listar sus URLs Top.

```
PS C:\Users\Administrator> Set-SPSiteURL -Identity http://domain.local -Url http://intranet.domain.local
PS C:\Users\Administrator> Get-SPSiteURL -Identity http://domain.local
Url                                     Zone AppSiteDomainId
-----
http://domain.local/                   Default
http://intranet.domain.local           Default
PS C:\Users\Administrator> Set-SPSiteURL -Identity http://domain.local -Url http://extranet.domain.local
PS C:\Users\Administrator> Set-SPSiteURL -Identity http://domain.local -Url http://sample.apps.domain.local
PS C:\Users\Administrator> Get-SPSiteURL -Identity http://domain.local
Url                                     Zone AppSiteDomainId
-----
http://domain.local/                   Default
http://extranet.domain.local           Default
http://intranet.domain.local           Default
http://sample.apps.domain.local        Default
PS C:\Users\Administrator>
```





- Finalmente, si queremos eliminar una URL Top añadida a un HNSC es suficiente con ejecutar el comando "Remove-SPSiteURL" de acuerdo a la siguiente sintaxis:

Remove-SPSiteURL -URL "http://intranet.domain.local"

- Como se puede apreciar, el único parámetro que se necesita en "Remove-SPSiteURL" es la URL Top a eliminar.

Para concluir, y una vez que hemos realizado las configuraciones de creación del HNSC y adición de una o varias URLs Top, abrimos una sesión de nuestro navegador web y especificamos la URL añadida a la colección comprobando que accedemos a la misma sin problemas como muestra la última figura.

Conclusiones

SharePoint 2013, como en versiones previas, soporta la configuración y uso de HNSC simplificando el proceso de agregación de URLs Top a este tipo de colecciones de sitios a través de la adición de nuevos comandos PowerShell.

Aunque los HNSC proporcionan un mecanismo potente para poder trabajar con URLs "Top" a nivel de colección de sitios, hay ciertos escenarios en los que no es posible su uso.

Referencias

- [1] Concepto de Colecciones de Sitios "Host-Named" para SharePoint 2010 en Microsoft TechNet: <http://technet.microsoft.com/en-us/library/cc424952.aspx>
- [2] Consideraciones a la hora de usar Colecciones de Sitios "Host-Named" en SharePoint 2010: <http://blogs.msdn.com/b/markarend/archive/2012/05/30/host-named-site-collections-hnsc-for-sharepoint-2010-architects.aspx>

Juan Carlos González Martín

Arquitecto de Soluciones

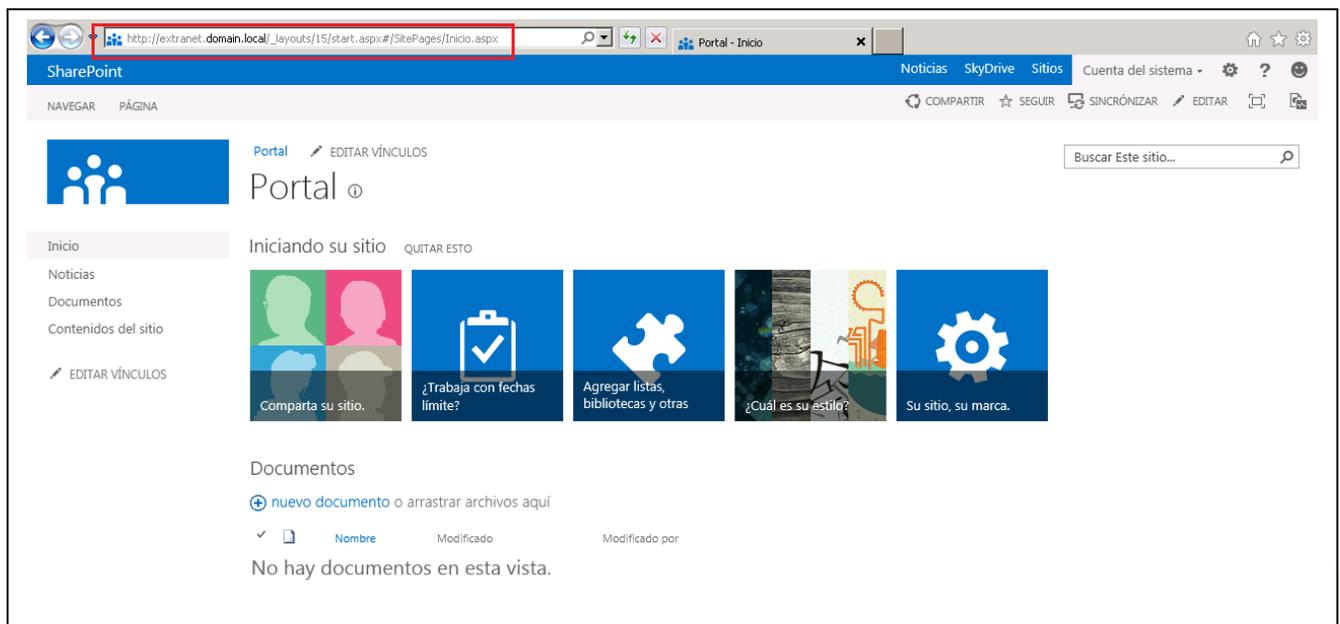
MVP de SharePoint Server

Centro de Innovación en Integración (CIIN)

Microsoft Innovation Center Cantabria

<http://geeks.ms/blogs/ciin>

@jcg1978





La comunicación y el rol de la Intranet Corporativa (IC)

Introducción

Esta publicación tiene como objetivo general describir y destacar la función de liderazgo que los profesionales de la comunicación empresarial tienen en el proceso de desarrollo interdisciplinario de una intranet corporativa (IC).

Poniendo el foco en, las IC, como soluciones comunicacionales basadas en tecnología de Internet, MS SharePoint, constituyéndose en un elemento estratégico para potenciar el intercambio de información, la colaboración, la gestión de proyectos y las nuevas formas de comunicación dentro de las organizaciones.

Según la autora Tittel, Stewart, 1997 El valor real de una Intranet no reside en el hardware o el software que forman se superestructura, si no en los métodos para distinguir, manipular y distribuir los datos.

En ese marco, este trabajo busca posibilitar y brindar a los profesionales de la tecnologías de la información (TI) una visión amplia de los recurso y herramientas profesionales que los especialistas en comunicación organizacional pueden aportar en el proceso de desarrollo y diseño de las IC. De tal forma que posibilite, a los profesionales de TI, conocer la forma de vincularse y participar - con su visión especializada y su bagaje de conocimientos y experiencia- en cada etapa del proceso de desarrollo en conjunto con los especialistas en comunicación organizacional. Es decir, enfocar y hacer comprender, tanto al cliente interno de la empresa como al resto de los especialistas intervinientes, en el negocio del desarrollo y diseño de una IC en SharePoint desde una perspectiva más amplia basada en aspectos de comunicación y diseño y no solamente en tecnologías ingenieriles y/o informáticas.

Se toma la definición sobre la comunicación organizacional de Fernández Collado que la define como: "el conjunto total de mensajes que se intercambian entre los integrantes de una organización y entre ésta y su medio", continuando con su especificación y ampliación como "un conjunto de técnicas y actividades

encaminadas a facilitar y agilizar el flujo de mensajes que se dan entre los miembros de la organización, entre la organización y su medio.

Por lo dicho, el punto de vista se amplía y podemos decir que una intranet es "una nueva forma de comunicarse y de organizar la actividad de la organización, que conlleva una nueva cultura de empresa Este nuevo aspecto del rol de las IC se encuentra directamente relacionado con la economía de las empresas que implementan estas soluciones tecnológicas.

Para comprender el valor estratégico de las IC en cualquier organización, es conveniente repasar las opiniones de Kevin Kelly (1998), quien plantea que la nueva economía tiene tres características principales: es global, apoya lo intangible -las ideas, la información y las relaciones- y está intensamente interconectada.

Para este autor, esos tres atributos generan un nuevo tipo de mercado y de sociedad que tienen sus orígenes en las redes electrónicas que hoy están presentes en todas partes. En todas las economías siempre han existido las redes. La diferencia es que ahora las redes, realizadas y multiplicadas por la tecnología, invaden nuestras vidas tan intensamente que "el sistema de redes" se ha convertido en la metáfora más importante en torno a la cual se organizan todas nuestras ideas, nuestras actividades y nuestra economía.

El autor mencionado afirma que "la comunicación -que en último término es de lo que se trata la tecnología digital y los medios de comunicación- no es únicamente un sector de la economía, la comunicación es la economía" y que "La nueva economía tiene que ver con la comunicación, amplia y profunda. La comunicación es el fundamento de la sociedad, de nuestra cultura, de nuestra humanidad, de nuestra propia identidad individual, y de todos los sistemas económicos. Es por esto que son tan importantes las redes." (Kelly, 1998. p. 20).

... la función de liderazgo que los profesionales de la comunicación empresarial tienen en el proceso de desarrollo interdisciplinario de una intranet corporativa...





En síntesis, este documento aspira a constituirse en un recurso que contribuya a que los profesionales de la IT comprendan la importancia de los profesionales de la comunicación empresarial en el proceso de desarrollo de un IC. Entendiendo la capacidad de estos para introducirse y conocer las necesidades particulares de cada cliente; comprender los objetivos comunicacionales de cada IC; identificar los grupos intervinientes en el proceso de desarrollo; conocer las funcionalidades específicas de una IC.

Aproximación conceptual

Es importante que los profesionales de las TI comprendan que la principal finalidad de la comunicación organizacional es alcanzar los objetivos institucionales y, por lo tanto, la intranet corporativa es el elemento de comunicación organizacional interna que facilita el alcance de dichos objetivos.

La información, la expresión comunicativa y la comprensión serían los tres elementos básicos del fenómeno comunicativo. Por consiguiente, para que se desarrolle un proceso comunicativo "la información ha de ser expresada y esta expresión comunicativa ha de ser comprendida, sólo cuando se procesa el estímulo, se genera la comunicación" (Luhmann, 1995, p. 139).

El concepto que tomaremos para establecer nuestro marco de análisis referido a la comunicación organizacional será el expuesto por Fernández Collado (1997, p. 7) que la define como "el conjunto total de mensajes que se intercambian entre los integrantes de una organización, y entre ésta y su medio", continuando con su especificación y ampliación como: "Un conjunto de técnicas y actividades encaminadas a facilitar y agilizar el flujo de mensajes que se dan entre los miembros de la organización, entre la organización y su medio; o bien, influir en las opiniones, aptitudes y conductas de los públicos internos y externos de la organización, todo ello con el fin de que ésta última cumpla mejor y más rápido los objetivos".

Debido a que el conocimiento puede ser fluido y efímero, las organizaciones sólo pueden comunicarse y colaborar de manera efectiva a través de las Intranet Corporativas y explotar su ventaja competitiva de la distribución del conocimiento y la información, esto siempre y cuando las técnicas y procesos específicos estén adecuadamente puestos en marcha. Ello no siempre sucede ya que, como señala Nielsen (2007) muchas veces las empresas se preocupan más por cuestiones técnicas que por buscar desarrollar una herramienta que se útil para los usuarios.

En primer lugar, la información que se coloca en la Intranet debe ser cuidadosamente seleccionada, organizada y

estructurada de acuerdo a una política coherente de acuerdo con las características de cada empresa (Rowley, 2000). Esta debe estar diseñada contemplando los mecanismos que facilitan la navegación y la búsqueda efectiva del contenido. Lo importante conceptualmente para casi todos los aspectos de la Intranet, es considerar en el análisis previo al diseño que el acceso al conocimiento sea fácil. "Todo debe servir a el objetivo principal de permitir a los usuarios encontrar lo que necesitan" (Futterman, 2001).

Proceso de desarrollo de una IC

Intranet canónica

A continuación se presenta una estructura canónica en la cual los especialistas de comunicación podrán basarse para realizar la primera estructuración de la IC.

Jakob Nielsen propone, en su estudio Usability of Intranet Portals una imagen compuesta por páginas de inicio de 10 grandes intranets superpuestas. Como el gráfico muestra claramente, aunque cada una de las pantallas posee algunas desviaciones, todas comparten las mismas tendencias en su estructura, layout.



Figura 1: Superposición de páginas de inicio de 10 intranets. Fuente: Patty Caya and Jakob Nielsen. (2008) Usability of Intranet Portals- a Report From the Trenches. Experiences From Real-Life Portal Projects - 3rd Edition.

La imagen presentada representa los cuadrantes que conforman la estructura básica de cualquier intranet. Es menester del profesional de la comunicación empresarial trabajara sobre este esquema general para poder definir la estructura de contenidos de la IC teniendo en cuenta las características de cada una de estas áreas.

- Barra horizontal superior: Normalmente se utiliza para el logotipo, la navegación global (a menudo con solapas), y un cuadro para la herramienta de búsqueda. (Fondo de color, 100 píxeles de alto)





- Columna izquierda: Normalmente se utiliza para la barra de navegación detallada y/o con una lista contextual de opciones para el sub-sitio de la IC. (Fondo de color, 200 a 250 píxeles de ancho.)
- Zona central: Por lo general cuenta con una amplia columna (o algunas veces dos columnas) que contiene fotos o ilustraciones, una lista de titulares de noticias, y las cajas con los accesos a los contenidos y aplicaciones más importantes de la IC.
- Aéreas destacadas con banners (Fondo blanco, 400 a 600 píxeles de ancho).
- Columna derecha: Suele utilizarse para un conjunto de módulos, algunos con contenido de color o imágenes (Fondo blanco, 200 a 250 píxeles de ancho).

El conocimiento de la lógica del negocio de la empresa y los usuarios de la IC.

Para definir la interfaz de usuario, sus características comunicacionales y funcionales, los profesionales de la comunicación empresarial deberán conocer los objetivos de la IC, la arquitectura de la información y la estructuración de los contenidos; conocimientos todos basados en la información de la lógica del negocio respecto de la empresa. Es decir, los objetivos a establecerse para una IC dependerán de la organización empresarial en la cual ésta será implementada, su cultura organizacional, la comunicación interna y, finalmente, el área dentro de la empresa que será la responsable de llevar adelante el proyecto. La definición de la tecnología estará definida por el departamento de sistemas, en este caso SharePoint en cualquiera de sus versiones.

Es importante que se realice esta distinción debido a que de esto dependerá la orientación del desarrollo/diseño y la determinación de dónde estará puesto el acento dentro de las características comunicacionales y funcionales de la IC.

Como se analizó en párrafos anteriores, las ICs constituyen una de las herramientas de trabajo colaborativo y de comunicación más importantes dentro de los elementos que maneja un profesional de la comunicación empresarial en su trabajo de programación de la comunicación interna dentro de la empresa. Normalmente las ICs son transversales a varias áreas e incluso países o regiones por lo que debe manejar un delicado equilibrio entre la comunicación, las funcionalidades y los objetivos generales de la compañía siempre teniendo en cuenta las necesidades particulares de cada región, área, e incluso empleado.

En el proceso de desarrollo y gestión deben participar tanto los impulsores del proyecto (áreas de recursos

humanos, tecnología de la información, ventas, comunicación, etc.) como quienes serán los usuarios finales. Esto es necesario para que la arquitectura de la información y de contenidos de la IC estén basadas en los procesos de trabajo reales de los usuarios y no en prejuicios o análisis basados en hipótesis no contrastadas con la realidad y el día a día del usuario.

Es por esto que en el momento de diseñar una IC es importante utilizar una metodología de diseño comunicacional centrado en el usuario y en el marco de la lógica de negocio de la empresa y sus objetivos con respecto a la Intranet. Esta modalidad de trabajo permitirá definir y conocer los objetivos generales de la compañía pero también los de las áreas y empleados. El foco está puesto en lograr que los profesionales de la comunicación empresarial puedan interpretar e identificar las necesidades de cada área y brindar experiencias de comunicación y colaboración altamente satisfactorias para los usuarios mediante la creación de diseños amigables, de operación intuitiva y, por ende, fáciles de usar. De esta manera los profesionales de TI de la empresa podrán tener resuelto varios aspectos del proceso de desarrollo de una IC, pudiendo enfocarse en los aspectos referidos a la tecnología.

Determinación general de los objetivos comunicacionales de una IC

Sea cual fuere el cliente, externo o interno, el procedimiento del proceso de desarrollo comienza con la identificación del objetivo de la empresa en referencia a su IC.

Esto, por lo tanto, requiere del trabajo de determinar, en conjunto con el cliente la descripción de cuál es el corazón del negocio de la empresa, Core Business.

Existe, para ese momento, una serie de preguntas que ayudan al profesional en comunicación a comprender qué es lo que espera su cliente de su IC y, de esa forma, definir su estructura básica de contenidos y que el departamento de IT responsable de desarrollar e implementar la solución puedan ir conociendo los alcances del proyecto. . Ellas son:

- ¿Podría definir y/o describir el Core Business de la empresa?
- ¿Cuál es el beneficio principal a enfatizar en esta nueva IC?
- ¿Cuáles son las oportunidades de la empresa que se buscarán desarrollar con la implementación de la IC?
- ¿Cuáles son los riesgos de la empresa que se deberían evitar con la implementación de la IC?





- ¿Desde dónde se origina la necesidad interna de contar con este sistema, desde que área o áreas?
- ¿Cuáles serían los Módulos claves de la IC?
- ¿Quiénes serían los usuarios de la IC?
- ¿Quiénes administrarían la IC?
- ¿Cuál es el área de la empresa que más se verá beneficiada con la implementación de este sistema?
- ¿Cuáles serían los circuitos o procesos internos que se podrán optimizar con este sistema?
- ¿Qué ventajas más sobresalientes reconoce a priori gracias a la implementación de este sistema?
- ¿Cuál es la reacción y los comentarios que le gustaría recibir de los usuarios acerca del desarrollo?
- ¿Quiénes son las personas que piensa que usarán el nuevo sistema con mayor asiduidad?
- ¿Con qué fin desea que los usuarios o clientes ingresen a la IC? Objetivos.
- ¿En qué orden cree Ud. que ellos querrán visitar la información, los nuevos módulos o las nuevas secciones de la IC?

Una vez que se compilen y analicen las respuestas a estas preguntas se podrá obtener una idea inicial de los objetivos que la IC deberá cumplimentar. Pudiendo definirse cuales de esto SharePoint resuelve, out off the Box, y cuales deberán ser desarrollados a medida.

Relación objetivos-funcionalidad en una IC

El éxito del trabajo del profesional de la comunicación empresarial está relacionado con el resultado de este tipo de aplicaciones y este éxito está íntimamente relacionado con el nivel de participación y compromiso de los empleados administradores de los contenidos de la IC. Ésta es una de las claves para generar convocatoria hacia el usos de la IC, ya que con un alto nivel de compromiso en la elaboración de contenidos se garantiza la permanente actualización de los mismos, se genera un dinamismo que transmite la sensación de permanente actualización y, por sobre todas las cosas, se producen contenidos que reflejan la situación actual de la empresa.

Una vez que se hayan establecido los objetivos generales, se puede continuar con la definición de los objetivos relacionados con las funcionalidades que se definan para la nueva IC.

Relación funcionalidad-comunicación en una IC

La interface de usuario presentada en el inicio del capítulo, contiene en sí misma las funcionalidades de la IC. La comunicación es un elemento que se desprende y, a su vez, retroalimenta las funcionalidades de la IC. Se puede poner como ejemplo el denominado "módulo de noticias destacadas" que generalmente se encuentra en la página de inicio (Home) de una IC. Donde las noticias son destacadas a través de un cartel (banner) especialmente diseñado para que tome un sector destacado de la página de inicio.

En este ejemplo, la comunicación se vincula con la funcionalidad del módulo que se desarrolló específicamente para esa sección de noticias destacadas.

El balance entre comunicación y funcionalidad dependerá de que área es la que tiene como responsabilidad el desarrollo de la IC. Si este trabajo recae sobre el área interna de recursos humanos (RRHH), muchas veces acompañados por el área de sistemas -el área que estará a cargo del proyecto-, los requerimientos seguramente estarán enfocados en los aspectos y en las herramientas de gestión y sus funcionalidades. Por ejemplo: formulario de pedido de vacaciones, publicaciones laborales internas (Job Posting), evaluaciones de desempeño etc. Específicamente estarán interesados en las soluciones de gestión que las ICs les brinden a los empleados de la compañía ligados principalmente a las necesidades de esta área. Por el contrario, si es el área de comunicación Institucional e interna la encargada de llevar a cabo el proyecto de desarrollo de la IC, los requerimientos iniciales y el acento en las funcionalidades, estarán puesto en sus aspectos comunicacionales: distintos niveles de lectura, distribución de los contenidos, imagen del portal, publicaciones de noticias y novedades etc.

Es necesario entonces describir los objetivos generales que se esperan de una IC para evaluar el balance que se espera que tenga entre funcionalidad y comunicación. Para esta etapa de definición general, en capítulos posteriores se profundizará sobre estos temas, el profesional de la comunicación empresarial deberá analizar cuáles son los objetivos que representan los intereses y las necesidades de la empresa y de los usuarios. Para ello deberá atender los siguientes aspectos relacionados con la comunicación y las funcionalidades de la IC:

- Participación proactiva de los usuarios en la carga de contenidos: Una manera de lograr este objetivo es desarrollar funcionalidades que permitan la actualización dinámica descentralizada y la participación de diferentes áreas de la empresa en la producción y la generación de contenidos. De esta manera la IC se mantendría siempre actualizada.





- Temas de mayor interés: es importante que los contenidos sean de interés para los empleados de la empresa. La estructura de la página de inicio de la IC y la determinación de sus contenidos se realiza en la etapa de relevamiento de las necesidades de cada área de la empresa a través de entrevistas con los distintos grupos de usuarios, o aéreas internas de la empresa. En estas reuniones presenciales, se determina cuál es la información que se publicará y su nivel de jerarquía dentro de la estructura de contenidos de la IC.
- Canal de comunicación interna: contar con una solución como una IC, que le permite a la empresa establecer un canal de comunicación directo con sus empleados donde centralizar la información, debería ser uno de los objetivos más importantes. Esto unificaría la comunicación institucional.
- Mejora en los niveles de comunicación: canalizando parte de ellos a través de la IC.
- Integración de procesos internos vía la IC: son ejemplo de esto la sección de preguntas frecuentes, mesa de ayuda, formularios de recursos humanos para operaciones simples, etc.

Arquitectura de la información, de una IC

Arquitectura de la información (AI) es definida como: "Disciplina (arte y ciencia) encargada de estructurar, organizar y etiquetar los elementos que conforman los entornos informacionales para facilitar de esta manera la localización (o el acceso) de la información contenida en ellos y mejorar, así, su utilidad y su aprovechamiento por parte de los usuarios" (Pérez-Montoro, 2010).

La especialista en usabilidad Web, Olga Carreras Montoto, menciona los tres pilares de una buena arquitectura de información basándose en Morville y Rosenfeld (2006): el contexto organizacional en el que se desarrolla, el contenido que alberga y los usuarios que la visitan y consultan.

Menciona además los sistemas, estructuras o recursos para estructurar una web y que definen la AI de un sitio. Estos se llaman componentes o anatomía de la arquitectura de información y son los siguientes:

- Sistemas de organización.
- Sistemas de etiquetado.
- Sistemas de navegación.
- Sistemas de búsqueda.
- Vocabularios o lenguajes documentales.

El profesional de la comunicación empresarial podrá trabajar con un especialista en arquitectura de la información para que en conjunto puedan definir la estructura final de la IC ya

que, tomando la conclusión de Moner (2002), del análisis de la información y del conocimiento asociado a los procesos y procedimientos de la empresa es que se determina la arquitectura de la información.

Estructura de contenidos iniciales, wireframes: El profesional de comunicación deberá conformar un árbol de información simple basado en la recolección de datos que obtuvo mediante las encuestas y las entrevistas mencionadas anteriormente.

Una vez que se ha definido la estructura de los contenidos, parte del trabajo de estructura de la información, una de las variadas formas de agregar valor a las ICs es identificar los procesos de trabajo que la empresa considera más importantes e incluirlos en la estructura que se ha definido para la IC, de manera que los usuarios interactúen directamente con ellos y reconozcan rápidamente que la herramienta les agrega valor a sus tareas diarias. Entendiendo que los procesos de negocios de las empresas son su columna vertebral, este análisis permite una buena aproximación para el planteamiento de una estructura de contenidos realmente útil para los usuarios.

En su artículo, Como manejar procesos nucleares del negocio en su Intranet, Juan Fernando Zuluaga C. propone incluir también dentro del análisis a las labores periféricas, como la contratación del personal (a menos que sea una empresa de contrataciones por outsourcing, por ejemplo) o la liquidación contable. Aunque estos procesos no son de alta importancia, ciertamente la empresa no puede obviarlos. Así mismo propone cinco pasos para optimizar los procesos centrales del negocio de la empresa desde una IC.

1. Incluir los procesos en la arquitectura de la información y contenidos de la IC.

2. Plasmar el flujo de procesos en la página inicial, Home, de la IC.

3. Desarrollar los contenidos en un esquema por fases

(Por ejemplo. Primera Fase: Incluir sólo Manuales y Procedimientos; Segunda Fase: Desarrollar Consultas a los Sistemas de Información Empresariales; Tercera Fase: Interacción completa)

4. Planear junto con el Departamento de Tecnología todas las interfaces a sistemas Corporativos que sean necesarios y que estarán disponibles en un futuro.

5. Permitir a los usuarios la autogestión de las secciones correspondientes a sus áreas de trabajo, con el fin de mantener la retroalimentación constante.

Juan Ibáñez

Lic. en Negocios de comunicación y diseño
jiban.es@paginar.net





10 puntos para entender la Gestión estratégica del portfolio de proyectos en Project Server 2010

... introducir al usuario en la potente herramienta de gestión estratégica de portfolio de proyectos en Project Server...

El propósito de este breve artículo es introducir al usuario en la potente herramienta de gestión estratégica de portfolio de proyectos en Project Server. Esta herramienta no ha sido tan conocida en la versión 2007 por ser una utilidad separada de Project Server. Sin embargo, en la versión 2010, su integración total y natural la convierte en una importante solución para el circuito previo a la ejecución de los proyectos: la elección de los proyectos a ejecutar.

1 ¿En qué consiste la gestión estratégica de proyectos en Project Server?

La mejor explicación a mi gusto es que, así como la gestión de proyectos busca lograr una ejecución exitosa de los mismos, la gestión estratégica del portfolio de proyectos, busca seleccionar los proyectos a ejecutar, detectando cuáles le aportan mayor valor al negocio. Si lo piensan 5 minutos, ¿no es esto demasiado importante como para no considerarlo?

Muchas organizaciones manejan este proceso de esta forma (muy resumido). A principio de un ejercicio fiscal, determinan los objetivos de negocio. Identifican un conjunto de iniciativas y evalúan cómo pueden ayudar a cumplir esos objetivos de negocio. Estiman un costo de alto nivel de esas iniciativas y un beneficio esperado. En algunos casos planifican la necesidad de recursos. Con todos esos datos seleccionan los proyectos que pueden ejecutar de acuerdo a un presupuesto siempre restringido. Una vez que se aprueba esa selección comienzan los proyectos uno a uno, pero eso ya es terreno conocido.

2 ¿Qué son los drivers del negocio?

El primer paso en este proceso es identificar los objetivos de negocio que perseguimos. Para ello, Project Server mantiene una librería de drivers la cual debemos priorizar a través de un método manual o de un interesante algoritmo que nos hace comparar a todos los drivers entre sí y luego

le asigna un peso calculado a cada uno. Más adelante en el proceso, buscaremos aquellos proyectos que mejor estén alineados a nuestros drivers.

3 ¿Cómo se priorizan los proyectos?

El proceso es sencillo. El primer paso es crear un escenario de análisis en el que debemos elegir:

- ¿Qué proyectos vamos a analizar?
- ¿Qué priorización de drivers utilizaremos?
- ¿En qué campo almacenamos el costo estimado del proyecto?
- Y algunos datos más si queremos analizar la disponibilidad de recursos.

Luego, contrastamos la alineación de cada uno de los proyectos contra cada driver y Project Server asigna una prioridad a cada proyecto, teniendo en cuenta la alineación de los mismos, pero también el peso de cada driver.

Lo interesante es que esto lo puede hacer el rol a cargo de la selección de proyectos, pero también puede ser pedido en el workflow de creación de la iniciativa, en donde el dueño de la misma puede crear una primera alineación de su proyecto con los drivers.

El resultado de este paso es una lista priorizada de proyectos.





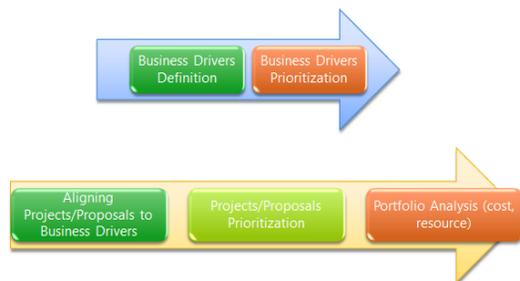
4 ¿Cómo es el proceso de selección de proyectos?

Una vez priorizados los proyectos, Project Server compara los costos con un presupuesto general y selecciona los proyectos que generen mayor valor a nuestro negocio y que puedan ejecutarse con nuestro presupuesto actual. Nos brinda la posibilidad de generar varios escenarios con distintas variantes de presupuesto y salvar cada uno de ellos.

Algo muy interesante es que previo a este análisis pudimos haber establecido dependencias como por ejemplo que un proyecto sólo pueda ser elegido si otro también es elegido. También exclusiones para manejar alternativas entre varios proyectos. Si elijo A, no puedo elegir B o C.

Finalmente no debemos olvidar que siempre podremos sobre-escribir lo que Project selecciona en forma automática, por ejemplo obligando a que un proyecto sea elegido porque es un cambio regulatorio.

Una vez seleccionados los proyectos podemos confirmarlos. Esto, además de registrar ciertos datos en la BD, podría disparar una acción de workflow como la aprobación para efectivamente dar inicio a un proyecto en particular.



5 ¿Es posible hacer un análisis de disponibilidad de recursos?

Si. Se trata de un paso opcional. Si he definido planes de recursos para los proyectos, puedo hacer análisis, también utilizando varios escenarios, para saber si mi disponibilidad de recursos es suficiente. Incluso puedo hacer variar datos cómo la contratación de recursos para determinar si un proyecto es seleccionado o no.

6 ¿Puedo incluir un flujo de trabajo para gestionar la lógica del proceso?

Claro y esta es una de las novedades más interesantes de Project Server 2010. Cada iniciativa puede arrancar con un flujo de trabajo para ir capturando información o realizando aprobaciones parciales. Lo interesante es que una iniciativa que no puede avanzar más en forma individual, puede quedar a la espera del proceso de selección de proyectos. Si la alternativa es seleccionada, la misma puede volver a su flujo individual que puede contener pasos como el inicio formal del proyecto, su ejecución y su cierre, es decir, el circuito completo de gestión de proyectos en la misma herramienta.

Es importante entender que nuestro flujo de trabajo puede no existir, ser simple o tan complejo como lo necesitemos.

7 ¿Cómo construyo los flujos de trabajo?

Los flujos de trabajo en Project Server 2010 deben ser contruidos con Visual Studio. Desde el punto de vista de la arquitectura, constituyen una capa por encima de los flujos de trabajo de SharePoint. SharePoint Designer no tiene lugar aquí. Algunas herramientas de terceros como Nintex soportan Project Server. A favor (si la contra es el desarrollo es obligado) es que hay mucho que se resuelve con parametrización: los campos custom, las fases y etapas del flujo, las PDPs que nos permiten agrupar la información y la obligatoriedad de los campos. Todo, menos la lógica del negocio que le queda a nuestro amigo, el Visual Studio.

8 ¿Qué son los Enterprise Project Types?

Cuando creo una iniciativa, lo hago desde Project Web Application, seleccionado un tipo de proyecto (EPT). Un EPT está caracterizado por un flujo de trabajo, una plantilla de sitio, una plantilla de Gantt y un departamento entre otros datos. Los EPTs le dan una enorme flexibilidad a Project Server, que antes no existía.

9 ¿Tengo que crear un Gant?





La creación del Gant no es obligatoria. Lo normal es que ni aparezca la opción en las primeras etapas de nuestro flujo de trabajo. Sin embargo, no es necesario esperar hasta el inicio del proyecto. Podríamos estar capturando algunos hitos principales incluso antes de la aprobación de una iniciativa. Está naturalmente integrada al flujo de trabajo.

10 ¿Por qué está totalmente integrado con la gestión de proyectos?

Porque en la misma herramienta soporto desde el nacimiento de una idea hasta el cierre del proyecto que la implementa, pasando por un proceso transversal como la selección del conjunto de proyectos que mejor valor le aporten a mi negocio.

Conclusión

Comenzar a trabajar con gestión estratégica del portfolio de proyectos no es una tarea tan complicada en Project

Server ya que ha sido uno de sus 4 pilares de mejora en su versión 2010.

Por supuesto, detrás de una implementación de este tipo, las organizaciones necesitan definir un proceso, si aún no lo tienen.

De todas maneras, existen diferentes niveles de complejidad para comenzar, no es necesario iniciar con toda la funcionalidad completa.

Los animo a intentarlo. ¡Hasta la próxima!

Juan Pablo Pussacq Laborde

SharePoint MVP

<http://surpoint.blogspot.com/>

<http://facebook.com/surpointblog/>

<http://twitter.com/jpussacq/>

The screenshot shows the SharePoint 15 home page. At the top, there is a navigation bar with 'SharePoint', 'Noticias', 'SkyDrive', and 'Sitios'. On the right, there are links for 'Cuenta del sistema', settings, help, and a smiley face. Below the navigation bar, there are options for 'NAVEGAR' and 'PÁGINA', and a row of icons for 'COMPARTIR', 'SEGUIR', 'SINCRONIZAR', 'EDITAR', and a refresh icon. The main content area features a 'SharePoint 15' header with an 'EDITAR VÍNCULOS' link and a search box labeled 'Buscar Este sitio...'. Below the header, there is a 'Inicio' section with a 'Iniciando su sitio' card and a 'QUITAR ESTO' link. The 'Inicio' section includes a sidebar with 'Noticias', 'Documentos', 'Recientes', 'EnumerarListas', and 'Contenidos del sitio', along with an 'EDITAR VÍNCULOS' link. The main content area displays five cards: 'Comparta su sitio.', '¿Trabaja con fechas límite?', 'Agregar listas, bibliotecas y otras', '¿Cuál es su estilo?', and 'Su sitio, su marca.'. Below the cards, there is a 'Documentos' section with a '+ nuevo documento o arrastrar archivos aquí' button and a table with columns for 'Nombre', 'Modificado', and 'Modificado por'. The table is currently empty, with the message 'No hay documentos en esta vista.'



MadPoint

SharePoint Users Group Madrid

MadPoint. Grupo de Usuarios de SharePoint de Madrid

Todos los profesionales, usuarios y amantes de la plataforma SharePoint que vivan en Madrid están de enhorabuena, ya que ha nacido un nuevo grupo de usuarios de SharePoint en su región.

El grupo, bautizado como MadPoint, va a ser un grupo de ayuda, en el que todos los miembros podrán compartir su experiencia con SharePoint mediante herramientas on-line, eventos presenciales, webcast, libros, noticias, artículos, material, etc.

El nombre elegido evoca a Madrid y a SharePoint, y también se puede entender como un punto de encuentro o "el punto de la locura".

MadPoint anima a todos los madrileños a que participen en las actividades del grupo. De momento es posible seguir al grupo en las redes sociales LinkedIn y Twitter. Y próximamente, el 8 de octubre, tendrá lugar el primer evento presencial del grupo, ¡permaneced atentos!

Si deseas seguir las actividades del grupo, puedes hacerlo en:

- LinkedIn: <http://www.linkedin.com/groups/MadPoint-SharePoint-Users-Group-Madrid-4554702>
- Twitter: @MadPoint
- Nuestra web (próximamente): www.madpoint.org.es

Si quieres colaborar activamente con nosotros, puedes contactar con:

- **Mario Cortés:** mariocortesflores@hotmail.com
- **Miguel Tabera:** miguel.tabera@outlook.com





Entrevista con Alberto Diaz Martin

Biografía – Introducción



No todo es tecnología pero yo he decidido vivir con ella desde que mis padres, cuando yo tenía cinco años, me pusieron en las manos un Spectrum ZK. Nací en Santa Cruz de Tenerife, en las Islas Canarias, archipiélago español de clima subtropical con hermosas playas y montañas. Siempre trabajando en Tenerife, soy responsable de proyectos en General de Software, empresa dedicada al mundo de la tecnología, en la que desarrollo software para el mundo empresarial, centrándome en SharePoint y Business Intelligence. Unos 28 años de tecnología por los que he pasado de administrar los sistemas en una gran empresa canaria a ser el responsable de proyectos de desarrollo de soluciones Microsoft y, gracias a todo esto, a mi familia y a mis amigos, soy MVP de SharePoint.

¿Por qué y cómo empezaste en tecnología?

Todo se lo debo a mis padres, que fueron los que invirtieron en ella para mí. Me compraron un Spectrum ZX con muchos juegos en 'cassette' y, desde entonces, me he puesto a desarrollar aplicaciones. Me viene a la mente mi primer juego, copiado del libro del Spectrum, que movía una bolita de colores por la pantalla o mi primer proyecto de software, la aplicación de un call center conectada a una centralita de VoIP, que realicé con mi empresa actual.

Cuando llegué a la época universitaria, pasé de querer estudiar matemáticas, a ver mi futuro en el mundo del desarrollo de software. Me planteaba que tendría más futuro laboral en Informática que en matemáticas y por eso la estudié. No me equivoqué, como informático nunca me ha faltado trabajo y he conseguido logros que nunca me planteé.

He estado con SharePoint desde la versión 2.0, y fue con la versión 2007 cuando me decidí a escribir un blog sobre tecnología. Siempre había leído a los expertos que escriben en Geeks.ms y me propuse colaborar con ellos pidiendo un espacio para escribir sobre .NET y SharePoint. Aquí me nació el gusanillo y las ganas de mostrar lo que se podía hacer con SharePoint, aunque mi primera charla fue en el pre-lanzamiento de Visual Studio 2010. Desde entonces supe que quería compartir conocimiento y conseguir que todos conocieran las bondades de mi mundo, la tecnología.

¿Cuáles son tus principales actividades tecnológicas hoy en día?

Como Arquitecto de Soluciones en General de Software, todos los proyectos con tecnología Microsoft, especialmente los de SharePoint e Inteligencia de Negocio, así como los proyectos de movilidad (Windows Phone, iOS, Android) pasan por mis manos. Esto me ha llevado a especializarme en .NET, Visual Studio y sus tecnologías adyacentes como son SharePoint, SQL Server, Windows Phone, Windows 8 y, porque el mercado de las Apps sigue creciendo, también conozco la tecnología iOS y Android.

En los últimos meses, Office 365, Windows Phone y Windows 8 han sido mis objetivos, plataformas en auge y estratégicas para mi empresa y Microsoft.



¿Cuáles son tus principales actividades NO tecnológicas hoy en día?

Pocas y debería de ser más. Me gusta pasar el tiempo con mi familia y amigos, el cine y viajar. Intento tener el fin de semana para ellas, aunque siempre acabo leyendo el email o el twitter de mi Windows Phone y no desconecto todo lo que debería de desconectar.

¿Cuáles son tus hobbies?

Qué os voy a contar que no hayáis leído ya. Soy un geek compulsivo. Me encanta irme a un hotel, sentarme en la terraza de la habitación, alejado del ruido, sólo con mi portátil y ponerme a desarrollar algún pequeño proyecto que tenga pendiente, algún trocito de código que tengo en mente. Luego está la Xbox, soy un adicto a los shooter y siempre he evitado tener suscripción Gold para no poder jugar online.

¿Cuál es tú visión de futuro en la tecnología de acá a los próximos años?

El mundo de las Apps ha conseguido un reto, y es que hayan nacido pequeñas empresas dedicadas al mundo del software, deslocalizadas geográficamente, viviendo de la venta de pequeñas aplicaciones en el mundo de los smartphones y tablets. Creo que las tiendas de aplicaciones son parte del futuro. Si conseguimos que estas tiendas funcionen en empresas, como lo ha hecho en los usuarios de consumo, a los desarrolladores se les abre un mundo de posibilidades impresionante, lejos de las grandes consultorías y de los mega-proyectos que las compañías reclaman. Parte de mi tiempo lo dedico a buscar pequeñas soluciones que desarrollar para vender en este modo, centrándome en la nueva tienda de Office que nos permitirá vender nuestras soluciones a estos usuarios o a empresas que usen SharePoint, estén donde estén.

Alberto Diaz Martin

MVP SharePoint Server

Arquitecto de Soluciones en General de Software

<http://www.gsc.es/>

<http://geeks.ms/blogs/adiazmartin>

@adiazcan

(Entrevista realizada por Juan Carlos González Martín para CompartiMOSS)





Campo para Geolocalización en SharePoint 2013

... el nuevo Field type de SharePoint 2013 preview: SPFieldGeolocation, que nos va a permitir añadir columnas de tipo geolocation, y posicionar puntos, en un mapa de Bing Maps...

Nota: El contenido del presente artículo se basa en la versión "Public preview" de SharePoint 2013, por lo que es posible que en la versión RTM Microsoft pueda introducir cambios en torno a la configuración y uso del campo para Geolocalización en SharePoint 2013.

Resumen

En este artículo hablaremos del nuevo Field type de SharePoint 2013 preview: SPFieldGeolocation, que nos va a permitir añadir columnas de tipo geolocation, y posicionar puntos, en un mapa de Bing Maps.

Artículo

El pasado 16 de Julio, Microsoft nos sorprendió a todos publicando la preview de SharePoint 2013 durante la presentación de la preview de Office 2013.

Entre las muchas mejoras de esta nueva versión de nuestro servidor favorito, contamos con una que, pese a no ser de las más revolucionarias, sí se echaba bastante de menos en el día a día de la versión 2010. Se trata de poder definir columnas de tipo Geolocalización, para así poder ubicar un punto geográfico en Bing Maps.

Sin embargo, por alguna razón desconocida (seguramente porque todavía es una preview), el tipo de campo Geolocation, no podemos crearlo desde la página de añadir un nuevo campo a una lista, o desde la pantalla de creación de una nueva columna de sitio. Tenemos los campos disponibles que muestra la Figura 1.

<p>Name and Type</p> <p>Type a name for this column, and select the type of information you want to store in the column.</p>	<p>Column name:</p> <input type="text"/> <p>The type of information in this column is:</p> <ul style="list-style-type: none"> <input checked="" type="radio"/> Single line of text <input type="radio"/> Multiple lines of text <input type="radio"/> Choice (menu to choose from) <input type="radio"/> Number (1, 1.0, 100) <input type="radio"/> Currency (\$, €, ¥) <input type="radio"/> Date and Time <input type="radio"/> Lookup (information already on this site) <input type="radio"/> Yes/No (checkbox box) <input type="radio"/> Person or Group <input type="radio"/> Hyperlink or Picture <input type="radio"/> Calculated (calculation based on other columns) <input type="radio"/> Task Outcome <input type="radio"/> Full HTML content with formatting and constraints for publishing <input type="radio"/> Image with formatting and constraints for publishing <input type="radio"/> Hyperlink with formatting and constraints for publishing <input type="radio"/> Summary Link data <input type="radio"/> Rich media data for publishing <input type="radio"/> Managed Metadata
---	---

Figura 1: Tipos de campo disponibles desde la interfaz.

Por suerte, tenemos una alternativa para poder añadir un campo de este tipo a una de nuestras listas:

- Lo primero que debemos hacer, si todavía no lo tenemos, es conseguir una Key de Bing Maps. Para ello deberemos acceder al portal de Bing Maps con nuestra cuenta Live ID (<https://www.bingmapsportal.com/>). Una vez autenticados, pincharemos en el enlace "Create or view keys".

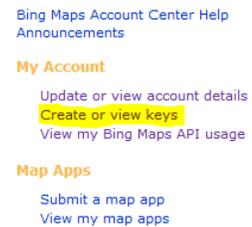


Figura 2: Acceder a la opción para crear una Key válida de Bing Maps.

- Rellenamos el formulario de solicitud de Key.

Figura 3: Rellenar formulario de creación de Key.

- Una vez aceptado el formulario, se nos dará una Key válida para Bing Maps.
- Lo siguiente que debemos hacer es dar de alta esa Key a nivel de la granja de SharePoint. Para ello disponemos de un nuevo comando Power-Shell:





```
PS C:\> Set-SPBingMapsKey -BingKey "12341234123412341234"
```

Figura 4: Estableciendo la Key de Bing Maps en toda la granja de SharePoint.

De igual modo, podemos ver la información de la Key, con otro comando Power-Shell:

```
Administrator: SharePoint 2013 Management Shell
PS C:\> Get-SPBingMapsKey
BingKey           : A1...
Name              : C:\095rkkd0h0h
Type              : SPBingMapsKey
TypeName         : Microsoft.SharePoint.Administration.BingMapsKey
DisplayName       : SPBingMapsKey
Id               : 430000ab-f1af-4a1c-99ab-f1a625609338
Status           : Online
Parent           : SPFarm Name=SharePoint_Config
Version          : 77437
Properties        : {}
Farm             : SPFarm Name=SharePoint_Config
UpgradedPersistedProperties : {}
```

Figura 5: Resultado del comando Get-SPBingMapsKey.

La Key se almacena como una propiedad del SPWeb, por lo que también puede definirse esta clave a nivel de Web, utilizando el modelo de objetos, tanto cliente, como servidor. El siguiente código utiliza el modelo cliente:

```
class Program
{
    static void Main(string[] args)
    {
        SetBingMapsKey();
        Console.WriteLine("Bing Maps set successfully");
    }
    static private void SetBingMapsKey()
    {
        ClientContext context = new ClientContext("<Site Url>");
        Web web = context.Web;
        web.AllProperties["BING_MAPS_KEY"] = "<Valid Bing Maps Key>";
        web.Update();
        context.ExecuteQuery();
    }
}
```

Figura 6: Estableciendo la key de Bing Maps desde el modelo de objetos cliente de SharePoint.

- Una vez establecida la Key de Bing Maps, podemos añadir una columna de tipo Geolocalización, utilizando código de servidor. Lo que haremos será crear una Custom Action a nivel de configuración de lista, que accederá a una página de aplicación personalizada. Esta página recibirá el GUID de la lista, y nos pedirá el nombre del campo Geolocalización a añadir.

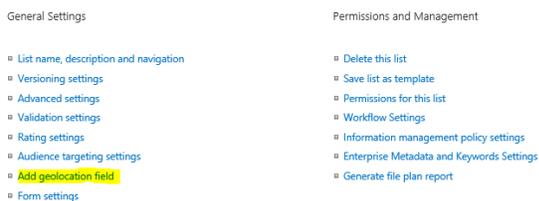


Figura 7: Custom Action a nivel de "General Settings" de lista.

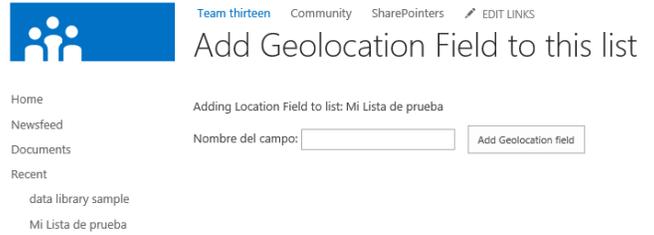


Figura 8: Application page personalizada, para añadir un campo de tipo Geolocation a la lista.

- Para conseguir esto, crearemos un nuevo proyecto de tipo SharePoint desde Visual Studio 2012. El proyecto es del mismo tipo que los de la versión 2010, y será desplegado en modo farm, para poder añadir la página de aplicación. El proyecto se compone de una Feature a nivel de Web, con un Elements.xml, que será el encargado de crear la Custom Action a nivel de Lista. Además, tendremos la página de Aplicación, encargada de añadir el campo a la lista.

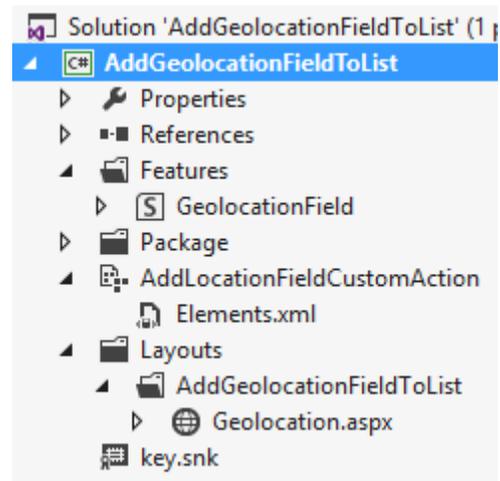


Figura 9: Estructura del proyecto SharePoint de Visual Studio 2012.

- En el fichero Elements.xml, añadiremos la Custom Action:

```
<?xml version="1.0" encoding="utf-8"?>
<Elements xmlns="http://schemas.microsoft.com/sharepoint/">
  <CustomAction
    Id="Custom.Configuration.ListCustomSettings"
    GroupId="GeneralSettings"
    Location="Microsoft.SharePoint.ListEdit"
    RequireSiteAdministrator="FALSE"
    Sequence="100"
    Title="Add geolocation field">
    <UrlAction Url="_layouts/AddGeolocationFieldToList/Geolocation.aspx?List={ListId}"/>
  </CustomAction>
</Elements>
```

Figura 10: Custom Action para añadir una nueva opción en las settings de la lista.





Fijaros como en la UrlAction, llamamos a una página de Aplicación personalizada, pasando como parámetro el GUID de la lista.

- En la página de aplicación, en el markup html, añadimos un campo TextBox para pedir al usuario el nombre del campo, y un botón, para ejecutar la acción.

```
<asp:Content ID="Main" ContentPlaceHolderID="PlaceHolderMain" runat="server">
  <p><asp:Label runat="server" ID="lblListTitle" /></p>
  <p>
    Nombre del campo: <asp:TextBox runat="server" ID="txtFieldName" />
    <asp:Button runat="server" ID="btnCreateField"
      Text="Add Geolocation field" OnClick="btnCreateField_Click" />
  </p>
</asp:Content>

<asp:Content ID="PageTitle" ContentPlaceHolderID="PlaceHolderPageTitle" runat="server">
  Add Geolocation Field
</asp:Content>

<asp:Content ID="PageTitleInTitleArea" ContentPlaceHolderID="PlaceHolderPageTitleInTitleArea" runat="server">
  Add Geolocation Field to this list
</asp:Content>
```

Figura 11: Markup de la página de aplicación.

- En el code-behind de la aplicación, tenemos el código para añadir el campo de tipo Geolocation.

Figura 12: Code-Behind de la página de aplicación, que añade un campo SPFieldGeolocation a la lista.

```
protected void Page_Load(object sender, EventArgs e)
{
    string listId = Request.QueryString["List"];
    if (String.IsNullOrEmpty(listId))
    {
        this.lblListTitle.Text = "List not found";
    }
    else
    {
        SPList list = SPContext.Current.Web.Lists[new Guid(listId)];
        this.lblListTitle.Text = String.Format("Adding Location Field to list: {0}", list.Title);
    }
}

protected void btnCreateField_Click(object sender, EventArgs e)
{
    try
    {
        string fieldName = txtFieldName.Text;
        string listId = Request.QueryString["List"];

        if (String.IsNullOrEmpty(listId))
        {
            this.lblListTitle.Text = "List not found";
        }
        else
        {
            SPList list = SPContext.Current.Web.Lists[new Guid(listId)];
            string fieldAsXml = String.Format("<Field Type='Geolocation' DisplayName='{0}' />", fieldName);

            list.Fields.AddFieldAsXml(fieldAsXml, true, SPAddFieldOptions.Default);
            list.Update();

            this.lblListTitle.Text = "Geolocation Field Added";
        }
    }
    catch (Exception ex)
    {
        this.lblListTitle.Text = ex.ToString();
    }
}
```

Como resultado del código anterior, tenemos un campo de tipo Geolocalización en la lista. Este tipo de campo, muestra un mapa de Bing Maps con la localización introducida.

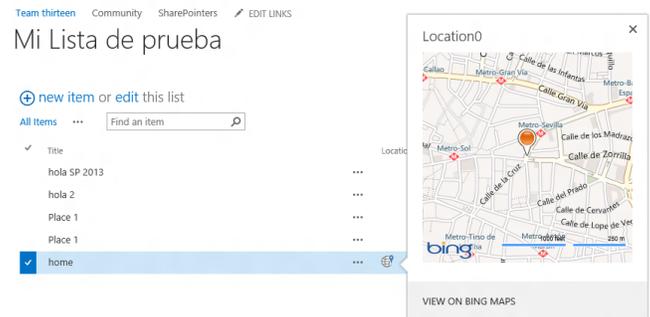


Figura 13: Visualizando un campo de tipo Geolocalización.





- De igual forma, cuando creamos un nuevo elemento de la lista, podemos dar de alta el valor del campo Geolocalización, bien estableciendo la ubicación actual (siempre que el navegador permita HTML 5), o bien definiendo unas coordenadas de latitud y longitud

Location Specify location Or Use my location Clear
Information will be sent to Bing Maps. [Learn More](#)

Specify Location ×

Latitude

Longitude

OK Cancel

Figura 15: Estableciendo la ubicación de un campo de tipo Geolocalización.

- Para acabar, podemos crear una vista de tipo Mapa, donde podemos ver en el mapa de Bing, todos los elementos de la lista, geoposicionados.

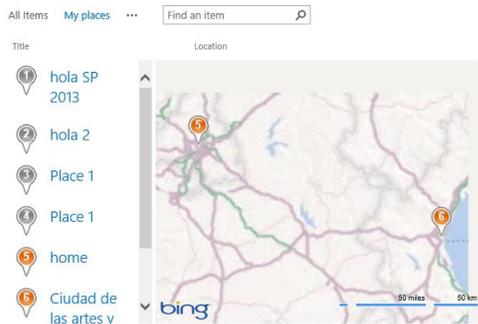


Figura 16: Resultado de la vista de tipo Mapa.

Esperemos que para la versión definitiva de SharePoint 2013, el campo Geolocalización pueda añadirse a una lista desde la interfaz estándar de SharePoint. Mientras tanto, podemos hacerlo a través de código personalizado.

Luis Máñez

CTO ENCAMINA

MCPD SharePoint 2010

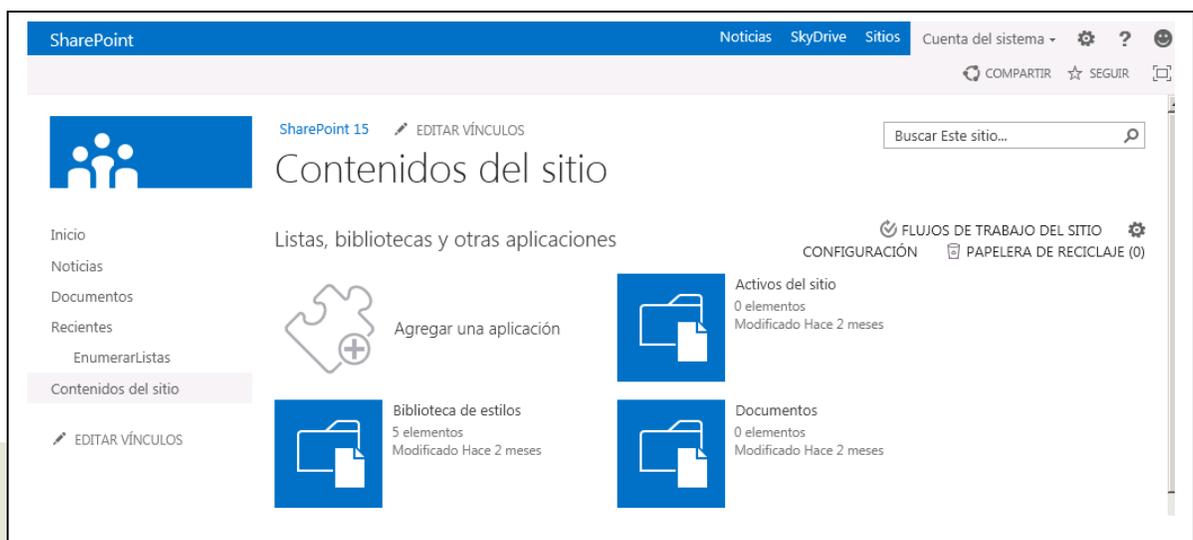
Microsoft Active Professional 2012

<http://geeks.ms/blogs/lmanez/>

<http://twitter.com/luismanez>

Referencias

- How to: Set the Bing Maps key at the web and farm level in SharePoint 2013:
[http://msdn.microsoft.com/en-us/library/jj163283\(v=office.15\).aspx](http://msdn.microsoft.com/en-us/library/jj163283(v=office.15).aspx)
- How to: Add a Geolocation column to a list programmatically in SharePoint 2013:
[http://msdn.microsoft.com/en-us/library/jj164050\(v=office.15\).aspx](http://msdn.microsoft.com/en-us/library/jj164050(v=office.15).aspx)
- Add a list item with the Geolocation field value to a SharePoint list programmatically:
[http://msdn.microsoft.com/en-us/library/jj164050\(v=office.15\).aspx#SP15addgeo_adlistitem](http://msdn.microsoft.com/en-us/library/jj164050(v=office.15).aspx#SP15addgeo_adlistitem)
- SPFieldGeolocation class:
[http://msdn.microsoft.com/en-us/library/jj170145\(v=office.15\)](http://msdn.microsoft.com/en-us/library/jj170145(v=office.15))





Conexión de WebParts de Filtros y Cuadro de Mandos de PerformancePoint mediante programación

Resumen

En este artículo vamos a explicar cómo a través de la programación en Visual Studio podemos conectar dos WebParts de PerformancePoint entre sí (un filtro con su cuadro de mandos correspondiente). La necesidad de la conectividad de estos WebParts surge debido a que realizar este trabajo de forma manual es una tarea muy costosa en cuanto a tiempo necesario. La opción que pensamos que es más productiva es realizarla a través de desarrollo.

¿Qué es PerformancePoint?

PerformancePoint Services es un servicio integrado en SharePoint Server 2010 (en su versión Enterprise) para la administración de rendimiento que permite supervisar y analizar los datos relativos a su empresa. Utilizando la herramienta PerformancePoint Dashboard Designer se pueden crear paneles, cuadros de mandos, informes e indicadores clave de rendimiento (KPI) de una forma relativamente fácil y sencilla para el usuario.

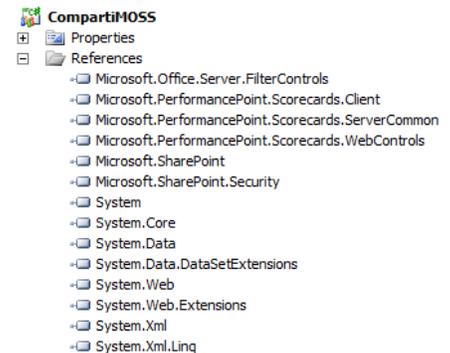
Introducción

Estos paneles, KPI, cuadro de mandos y filtros creados se almacenan en SharePoint a través de Listas en la que se guardan sus valores, cadenas de conexión, etc. Estos datos se visualizan dentro de SharePoint a través de WebParts que podemos posicionar en cualquier página de nuestro sitio web.

El problema surge a la hora de incrustar estos WebParts dentro de las páginas y darle la conectividad a los mismos. No es que sea una tarea muy difícil de realizar sino que es muy costosa en cuanto a tiempo, por lo que si lo hacemos mediante programación la ganancia de tiempo es sustancial. En este artículo se explica cómo conectar un WebPart de Cuadro de Mandos de PerformancePoint con otro WebPart de Filtro de PerformancePoint a través de Visual Studio.

Manos a la obra

Partimos de la base que ya tenemos creado un Cuadro de Mandos y un Filtro mediante PerformancePoint Dashboard Designer. Estos elementos los tenemos incrustados a través de sus WebParts correspondientes dentro de nuestro sitio de SharePoint y queremos conectarlos entre sí. De tal forma que dependiendo del Valor del Filtro en el Cuadro de Mandos se visualicen unos datos u otros (Para poder utilizar los WebParts relativos a Performance Point tenemos que tener activada la característica "Características de sitio de PerformancePoint Services").



Para este ejemplo me voy a crear un Visual WebPart, al que le añadiré un botón donde va a estar el código que conecte estos dos elementos, antes de empezar a introducir el desarrollo hay que añadir varias referencias al proyecto para poder trabajar directamente con estos WebParts. Las referencias son las siguientes:

- Microsoft.Office.Server.FilterControls
- Microsoft.PerformancePoint.Scorecards.Client
- Microsoft.PerformancePoint.Scorecards.ServerCommon
- Microsoft.PerformancePoint.Scorecards.WebControls

... cómo a través de la programación en Visual Studio podemos conectar dos WebParts de PerformancePoint entre sí...



El primer paso para poder llevar a cabo la conectividad, es encontrar la referencia de estos WebParts dentro de la página. Para ello implementaremos una función que dado un nombre y el WebPartManager nos devuelva el WebPart al que está asociado. El WebPartManager es el encargado de administrar todos los controles de elementos web, la funcionalidad y los eventos que se producen dentro de la página web.

```
public static WebPart
GetWebPart(WebPartManager man, string name)
{
    foreach (WebPart wp in man.WebParts)
    {
        if (wp.Title.Contains(name))
        {
            return wp;
        }
    }
    return null;
}
```

A continuación dentro del evento Click del botón nos crearemos dos variables de tipo WebPart, una será el filtro de PerformancePoint y otra será el cuadro de mandos.

```
WebPartManager manager =
WebPartManager.GetCurrentWebPartManager(Page);

WebPart provider =
FunctionPerformancePoint.GetWebPart(manager,
"WFilter");
WebPart consumer =
FunctionPerformancePoint.GetWebPart(manager,
"WCM");
```

Una vez tenemos los dos WebParts ahora nos hace falta llevar a cabo la conexión entre estos dos WebParts. Lo primero que tenemos que hacer es convertir el WebPart al tipo que realmente es, en nuestro caso uno es de tipo de FilterWebPart y el otro es de tipo ScorecardWebPart. (Como curiosidad para saber de qué tipo es un WebPart nos lo descargamos de la página donde lo tengamos desplegados y el fichero .WebPart que nos descargamos lo abrimos con un bloc de notas, en la cabecera del mismo indica la dll que utiliza).

Una vez realizamos este paso, tenemos que sacar de cada WebPart cuáles son los puntos de Conexiones que ofrecen. En el filtro nos hace falta los elementos que él puede enviar y en el Cuadro de Mandos necesitamos saber que elementos pueden recibir datos. Esto traducido a código quedaría de la siguiente forma:

```
FilterWebPart filtroWebPart =
(FilterWebPart)provider;
ScorecardWebPart CuadroMandos =
(ScorecardWebPart)consumer;

ProviderConnectionPointCollection ppp =
Manager.GetProviderConnectionPoints(filtroWebPart);
ConsumerConnectionPointCollection ccc =
manager.GetConsumerConnectionPoints(CuadroMandos);
```

A continuación nos hace falta un elemento que conecte el proveedor con el consumidor de datos, para ello utilizamos un Transformer, que indica que datos vamos a unir de un WebPart y de otro, es decir si yo tengo una Filtro que es de Departamentos y tengo un cuadro de mandos en la que tengo una columna que es Departamentos, hay que implementar algo para llevar a cabo esta conectividad, y eso es lo que se produce con el siguiente código:

```
TransformableBIDataProviderTransformer t = new
TransformableBIDataProviderTransformer();

TransformProviderConsumerRecord tpc = new
TransformProviderConsumerRecord(consumerParameter,
"FilterValues", "Valores de
PerformancePoint", "MemberUniqueName",
"MemberUniqueName", typeof(String), "");
List<TransformProviderConsumerRecord> listTpc =
new List<TransformProviderConsumerRecord>();

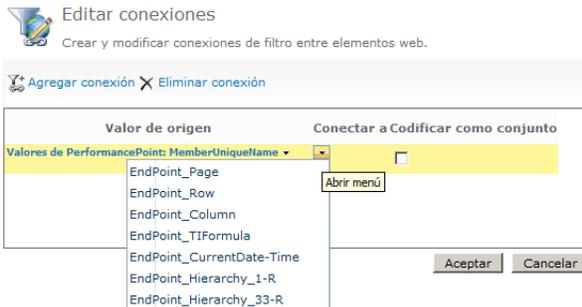
listTpc.Add(tpc);
ProviderConsumerTransformations pct = new
ProviderConsumerTransformations(listTpc);

t.ConfigurationState.ProviderConsumerTransformations = pct;
```

De todo este código puesto, nos tenemos que quedar con lo más importante que es esta línea que es cuando nos creamos el "TransformProviderConsumerRecord" el constructor tiene estos parámetros:

- consumerParameterName: En el cuadro de mando el nombre de la columna con la que se va a realizar el filtro, pero no penséis que va a ser fácil, si le hemos puesto departamento aquí no va a poner departamento, aquí utiliza una nomenclatura interna que tendremos que descifrarla. ¿Cómo? En primer lugar vamos a ver como se realiza la conectividad de estos WebParts de forma manual. En el momento en el que tenemos que conectar los dos WebParts nos saldrá una pantalla como la siguiente:





Lo que nos está indicando es por un lado la parte del Valor Origen (es decir la parte del filtro) y por otro lado nos está pidiendo que lo conectemos con la parte del cuadro de Mandos. Esta parte del desplegable son los valores que nos permite el cuadro de mandos, y que son los que necesitamos poner dentro de nuestro consumerParameterName. Antes de averiguar cómo se obtienen estos valores, voy a descifrar esto en un lenguaje que todos podamos entender: Del desplegable los valores que nos valen son los que contienen la palabra "Hierarchy", ahora bien como saber cuál es nuestra valor, en primer lugar tenemos que saber si es una columna o una fila y esta distinción se hace si finaliza en "Row" o en "Column". Una vez sabido eso aún no sabemos cuál es. En este caso aún tenemos dos tipos de R. ¿Cómo distinguir una u otra? Pues la posición en la que está saliendo es la jerarquía con la que hemos definido en el Cuadro de Mandos el PerformancePoint. En este ejemplo tenemos una agrupación por filas en primer caso por departamento y a continuación por áreas, por lo que si queremos agrupar por departamento tenemos que coger la primera opción.

Ahora bien como hacemos esa distinción utilizando código, la primera opción que buscamos es que dentro del Cuadro de Mandos tuviéramos alguna propiedad pública en la que estuvieran almacenados estos datos, pero cual fue nuestra sorpresa que no encontramos ningún método público que la tuviera almacenados estos valores. Tras buscar en modo de depuración observamos que había una propiedad privada donde estaban almacenados estos datos. Por lo que utilizando una clase "Reflector" como la siguiente fuimos capaces de conseguir estos valores:

```
class ReflectionHelper
{
    public static
    List<BIConsumerParameter>
    GetConsumerParameterRows(ScorecardWebPart
    scorecard)
```

```
{
    Type type =
    typeof(ScorecardWebPart).BaseType;

    FieldInfo parametersField =
    type.GetField("_biConsumerParameters",
    BindingFlags.Instance |
    BindingFlags.NonPublic);

    List<BIConsumerParameter> list =
    parametersField.GetValue(scoreca
    rd) as
    List<BIConsumerParameter>;
    return list;
}
}
```

Ahora bien desde nuestro código lo invocamos de la siguiente manera:

```
List<BIConsumerParameter> valores =
ReflectionHelper.GetConsumerParameterRows(CuadroMandos);
```

Y ahora ya tenemos los valores dentro de una lista, lo único que tenemos que hacer es aplicar la lógica que tenemos en el Cuadro de mandos para seleccionar un valor u otro.

- providerParameterName: Es el Nombre del parámetro del filtro aquí es un valor siempre fijo FilterValues.
- providerParameterDisplayNameName: Aquí hay que poner el valor que vemos cuando lo realizamos de forma manual Valores de PerformancePoint.
- ValuesColumnName: Aquí tenemos que indicar el tipo de conectividad que realiza el filtro MemberUniqueName o de tipo de DisplayValue. La diferencia entre elegir uno u otro es que en la primera opción busca por código dentro del cuadro de mandos y la segunda según los valores que se muestran en el filtro.
- DisplaColumnName: Aquí hay que poner el mismo valor que pongamos en la columna anterior.
- Type: Hay que poner el tipo de los datos que vamos a conectar (string, entero).
- providerFormula: En nuestro caso no nos hace falta y por lo tanto lo dejamos en blanco.

Una vez hemos pensado que ya está todo terminado todavía queda un paso más y es pasarle todos estas variables al WebPartConexion de SharePoint, para que la conexión entre los dos WebParts se almacene dentro de la página y listo.





```
SPWebPartManager spManager =  
SPWebPartManager.GetCurrentWebPartManager(Page)  
as SPWebPartManager;  
SPWebPartConnection conn =  
spManager.SPConnectWebParts(provider, ppp[0],  
consumer, ccc[0], t);
```

Conclusión

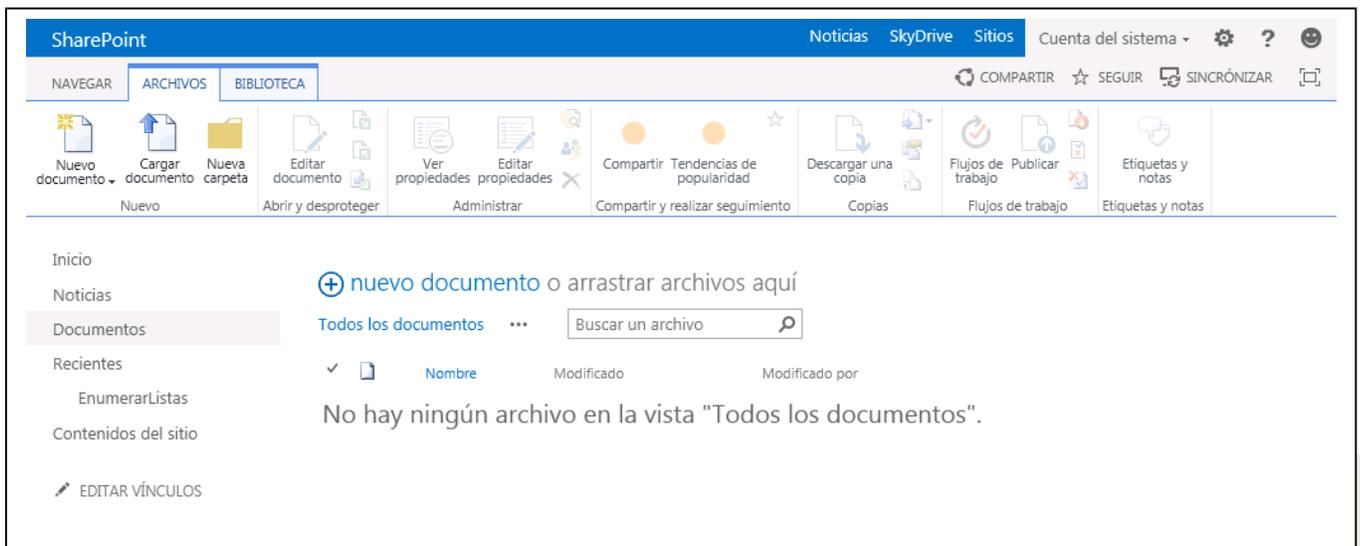
Esta conectividad entre WebParts de PerformancePoint hay que visualizarla para el despliegue de una aplicación a producción y desde el punto de vista de un proyecto bien estructurado. Es decir tenemos una solución en la que desplegamos las columnas de sitio, los tipos de contenido, las listas, las master page, los page layouts, taxonomía, cuadro de mandos de PerformancePoint, etc. Y sabemos que páginas vamos a crear, y que WebParts van a cada una de ellas y de esta forma de realizar los proyectos conseguimos un ahorro sustancial.

El principal problema que tiene PerformancePoint es la cantidad de datos que se muestra, es una herramienta para mostrar datos de cubos de BI y a poco contenido que tengamos estamos enviando por web una cantidad de datos

desbordada para los navegadores, en nuestros entornos de test hemos tenido exploradores con más de 2Gb, con lo que si este contenido (que se traduce a 4 cuadros de Mandos con sus correspondientes filtros) lo tenemos que conectar a través de un navegador ya hemos echado la tarde y parte de la noche.

Este trabajo de Investigación y Desarrollo para averiguar la conectividad entre los Filtros y Cuadro de Mandos de SharePoint la realizamos Fermín Sáez, Paco Martí y un servidor para un desarrollo de un proyecto de BI en la consultora valenciana Encamina S.L.

Adrián Díaz Cervera
MCPD SharePoint 2010
Microsoft Active Professional 2012
addiacer@gmail.com
@AdrianDiaz81





Novedades en Social Enterprise de SharePoint 2013

... En
SharePoint
2013 se ha
hecho un gran
trabajo para
mejorar el área
social de la
plataforma...

Nota: El contenido del presente artículo se basa en la versión "Public preview" de SharePoint 2013, por lo que es posible que en la versión RTM Microsoft pueda introducir cambios en torno a las características "Social Enterprise" en SharePoint 2013.

En SharePoint 2013 se ha hecho un gran trabajo para mejorar el área social de la plataforma. La visión social que ofrecía SharePoint 2010 no era la más adecuada, ya que no estaba alineada con las tendencias actuales de microblogging o seguimiento de actividades. Sobre estas premisas, en SharePoint 2013 se han basado en tres pilares fundamentales del Social Enterprise: Conectar a las personas, Gestionar el conocimiento y Compartir información y colaborar.

Conectando a las personas

Para conectar a las personas es necesario sacarlas de sus silos dándoles la capacidad de comunicarse por su propia voz, conectándolos a sus compañeros o responsables, identificando intereses comunes para, con todo esto, crear redes de personas.

Gestionando el conocimiento

La gestión del conocimiento tiene que conseguir promocionar la información de una forma natural, reduciendo el desorden e incrementando la productividad, mediante el uso de una comunicación informal con wikis y blogs, y creando comunidades para compartir el conocimiento de los proyectos.

El intercambio de información y la colaboración

La información tiene que nacer de las personas y para esto necesitamos podemos encontrarlas y saber quién es quién, para poder interactuar de una forma natural con ellas, comunicándonos, creando conversaciones y colaborando con las personas.

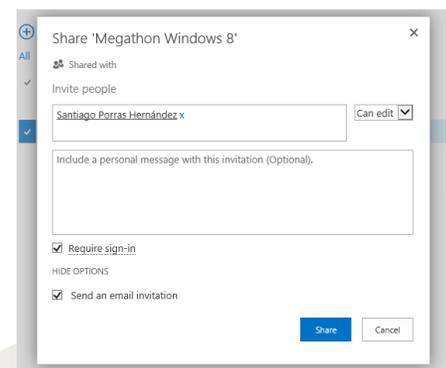
Mejoras en Mis Sitios con microblogging y compartir

En SharePoint 2010, Mis sitios permitían a los usuarios almacenar documentos personales y compartir biblioteca de documentos, además de ofrecer su información profesional y experiencia laboral, etiquetar contenido y comunicarse con otros a través de notas.

Con la nueva versión de SharePoint, Mis sitios continúan proporcionando los beneficios de la versión anterior, sin embargo, la interfaz de usuario ha sido completamente rediseñada y modernizada para ofrecer a los usuarios una experiencia atractiva e intuitiva. Un cambio clave para la interfaz de usuario incluye una experiencia de navegación simplificada y unificada. Además de las nuevas funcionalidades de Microblog y suministro de noticias. Estas permiten a los usuarios participar en conversaciones públicas y mantenerse al día sobre las actividades de los contenidos y las personas en las que están interesadas.

Compartiendo contenido

SharePoint 2013 introduce el concepto de compartir para todas las bibliotecas de documentos. Compartir está basado en la infraestructura de seguridad de SharePoint 2010, simplificando y mejorando la experiencia del usuario. Mediante el uso de esta experiencia simplificada, los usuarios pueden especificar los permisos para un documento específico sin tener que entender el modelo de herencia.

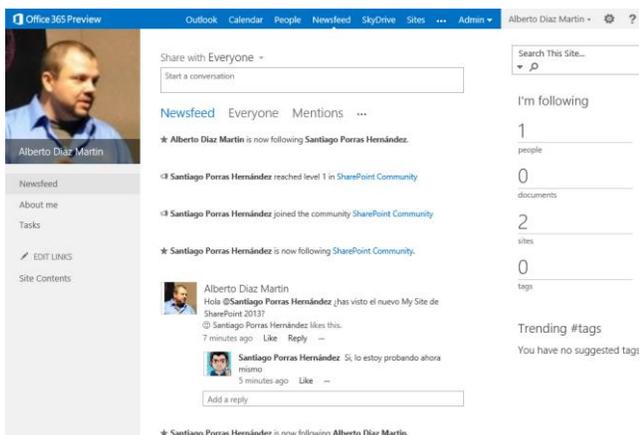




Microblogging y suministro de noticias

El suministro de noticias de Mi sitio sigue ofreciendo una vista agregada de las actividades de los contenidos y las personas que el usuario está siguiendo. Sin embargo, se ha mejorado con funcionalidades de microblogging que permite a los usuarios hacer lo siguiente:

- Participar en las conversaciones a través de comentarios y respuestas.
- Publicar imágenes y enlaces.
- Utilizar las etiquetas (empezando por el símbolo #) para definir las palabras clave que los usuarios puedan seguir y buscar.
- Utilizar menciona (empezando por el símbolo @) para etiquetar los usuarios de mensajes y respuestas.
- Indicar acuerdo con los comentarios y las respuestas haciendo clic en Me gusta.
- Seguir personas, documentos, sitios y etiquetas.



Sitio de comunidad para moderar conversaciones, incentivarlas y reputaciones.

El sitio de comunidad ofrece una experiencia de foro para categorizar y cultivar conversaciones con un grupo amplio de personas a través de las organizaciones en una empresa. Promueve la comunicación abierta y el intercambio de información mediante el fomento de debates entre los usuarios que comparten sus conocimientos y experiencia.

Los debates se organizan en categorías. Los visitantes pueden ver las discusiones y convertirse en miembros si quieren contribuir a esos debates. Los Moderadores gestionan la comunidad mediante el establecimiento de normas, la revisión y el tratamiento mensajes inapropiados, marcando contenido interesante. Los moderadores también pueden asignar insignias de talento a miembros específicos para indicar visualmente que el miembro es reconocido como un tipo específico de contribuyente en el sitio de la comunidad, como un experto o un moderador. Cada sitio contiene información acerca de la comunidad, sus miembros y la reputación del contenido, que los miembros ganan cuando se publica activamente en los debates, y cuando su contenido ha gustado, respondió a, o marcado como mejor respuesta.

Además, cuando tiene varios sitios comunitarios que desea mostrar a los usuarios de su empresa, puede implementar el Portal de la comunidad. El portal de la comunidad es una página de búsqueda de sitios que usan la plantilla Sitio de la Comunidad. Los usuarios pueden visitar el portal de la comunidad para descubrir las comunidades populares y buscar comunidades a las que tal vez desee asociarse.

Conclusiones

Una nueva versión y sigue mejorando el producto. Del lado social, ha sido capaz de adaptarse y encontrar la unión entre la empresa, el contenido y las personas. Cogiendo las mejores funcionalidades de las redes sociales, nos proporciona nuevas capacidades para mejorar la productividad de nuestra empresa con el uso de Seguimientos, menciones y demás. Si a esto le añadimos los sitios de comunidad, nos encontraremos con una experiencia social completa y orientada al sector empresarial, que es donde SharePoint trabaja perfectamente.

Alberto Díaz Martín

MVP SharePoint Server

Arquitecto de Soluciones en General de Software

<http://www.gsc.es/>

<http://geeks.ms/blogs/adiasmartin>

@adiazcan





Explotando los Servicios de Excel desde aplicaciones externas mediante sus servicios web

Una de las principales funcionalidades de SharePoint 2010 son los servicios de Excel. Estos permiten cargar y mostrar libros de Excel en nuestro navegador, integrado en nuestro SharePoint, interactuar con las hojas de cálculo publicadas y todo ello sin necesidad de tener instalado Microsoft Excel en los equipos clientes.

Los servicios de Excel están formados por 3 componentes principales: el acceso Web, el servicio Web y el servicio de cálculo. El acceso Web proporciona la funcionalidad para mostrar las hojas de cálculo en nuestro SharePoint y cobra vida con un Web Part. El servicio web es el componente que permite acceder a los Servicios de Excel a través del servicio web que proporciona lo que permite interactuar. El servicio web permite editar valores de celdas, obtener resultados de cálculos, actualizar conexiones con fuentes de datos externas, etc. desde cualquier aplicación aunque sea fuera del contexto de SharePoint. Por último tenemos el componente del servicio de cálculo. Es el componente menos visible y el que realiza los cálculos en sí de nuestros libros de Excel publicados en nuestro SharePoint.

Aunque en este artículo no entraremos en más detalle, no debemos dejar de comentar 2 funcionalidades importantes de los Servicios de Excel: los funciones definidas por el usuario (UDF: User defined Functions) y las conexiones con fuentes externas. Las UDF son funciones de cálculo publicadas por desarrolladores con la idea de proporcionar a los usuarios de los Servicios de Excel fórmulas de cálculo no existentes en Excel. Esto permite que un programador prepare funciones para cálculo personalizados, se publique en una biblioteca de UDF y que los usuarios desde Excel usen estas funciones. Las conexiones con fuentes externas permiten obtener información desde bases de datos o cualquier otra fuente de datos accesible desde Excel.

Hay algunas características de Excel que no son admitidas en los Servicios de Excel. Entre ellas podemos destacar VBA (Visual Basic for Applications), protección de Information Rights Management (IRM) y controles ActiveX. Las características no compatibles con Excel Services se pueden consultar en la siguiente URL: <http://msdn.microsoft.com/es-es/library/ms496823.aspx>

Configuración de los Servicios de Excel

Tras una instalación básica de SharePoint 2010, los Servicios de Excel no necesitan configuraciones adicionales para poder usarlo. En cualquier caso, en este artículo vamos a revisar la configuración básica de los Servicios de Excel. Posteriormente veremos con un poco más de detalle los servicios webs que nos proporciona los Servicios de Excel y como consumirlos. Lo veremos con un ejemplo y para hacerlo más geek y no salirnos de tecnologías SharePoint, atacaremos el servicio web desde un formulario de InfoPath.

Lo primero que hacemos es asegurarnos que el servicio está arrancado. Para ello, vamos a la "Administración Central" > "Administración de aplicaciones" > "Administrar aplicaciones de servicios" y ahí comprobamos el estado:

Servicios de Excel	Aplicación del servicio web de Servicios de Excel	Iniciado
Servicios de Excel	Proxy de aplicación del servicio web de Servicios de Excel	Iniciado

Configuración global

Define los umbrales del límite, la memoria y el equilibrio de carga. Establezca los tiempos de espera de la conexión de datos y la cuenta de servicio desatendida.

Ubicaciones de archivos de confianza

Define las ubicaciones desde donde se pueden cargar las hojas de cálculo.

Proveedores de datos de confianza

Agregue o quite proveedores de datos que se puedan usar al actualizar las conexiones de datos.

Bibliotecas de conexiones de datos de confianza

Define una Biblioteca de documentos de SharePoint desde donde se pueden cargar las conexiones de datos.

Ensamblados de funciones definidas por el usuario

Registre los ensamblados de código administrado que pueden usarse en las hojas de cálculo.

Si damos sobre "Servicios de Excel" entraremos en la pantalla de configuración de los mismos y donde podemos comprobar parámetros de configuración generales como las conexiones de datos externas, tiempo de sesión, etc. Es importante que aquí revisemos si tenemos definida una ubicación de archivos de confianza definida para más tarde dejar nuestro archivo de Excel en una de estas ubicaciones. Por defecto en SharePoint 2010 está configurado para que se pueda mostrar a través de los Servicios de Excel cualquier archivo Excel alojado en toda la colección de sitios de nuestro SharePoint, pero podríamos restringir a un sitio o una biblioteca como

... Los servicios de Excel están formados por 3 componentes principales: el acceso Web, el servicio Web y el servicio de cálculo...





ubicación de confianza para delimitar los archivos de Excel que se pueden mostrar en los Servicios de Excel.

Otros parámetros que podemos definir desde esta página serán las conexiones de datos de confianza, funciones definidas por el usuario, proveedores de datos, etc.

Escenario de Ejemplo

Antes de ver la publicación del libro Excel, os comento el objetivo del ejemplo que vamos a ver para que tengáis claro por dónde van los tiros. La idea es publicar un formulario de InfoPath de solicitud de vacaciones donde se calcule la diferencia entre la fecha de inicio y la fecha final de la solicitud pero contando sólo los días laborables; descontando los fines de semana y los días festivos. Este cálculo lo podríamos realizar directamente desde InfoPath con fórmulas, pero es más complicado y, en cambio, en Excel existe una fórmula que precisamente realiza lo que buscamos. Por ello, prepararemos una hoja Excel donde tendremos 2 parámetros de entrada (Fecha Inicio y Fecha fin), los días festivos y una celda donde se mostrará el resultado de la fórmula. Este libro de Excel lo publicaremos en los Servicios de Excel de nuestro SharePoint.

Por otro lado, desde el formulario de InfoPath, llamaremos a los servicios web de los Servicios de Excel pasándole como parámetro las dos fechas rellenas en el formulario, recogeremos el resultado de este cálculo realizado en la celda del Excel y lo mostraremos en nuestro formulario de InfoPath.

Con este ejemplo, veremos cómo funcionan los Servicios de Excel, sus servicios web y cómo podemos consumirlos, en este caso, desde InfoPath.

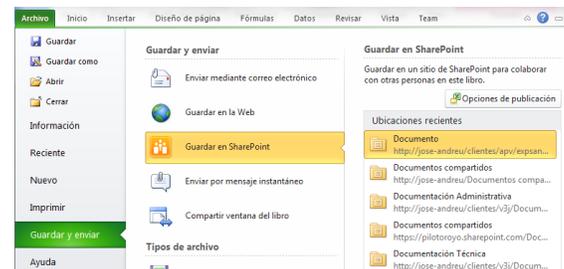
Publicación de un libro Excel en los Servicios de Excel

Creamos un libro de Excel con las celdas para los campos "Fecha Inicio", "Fecha Fin", "Días Laborales" y el listado de "Días Festivos":

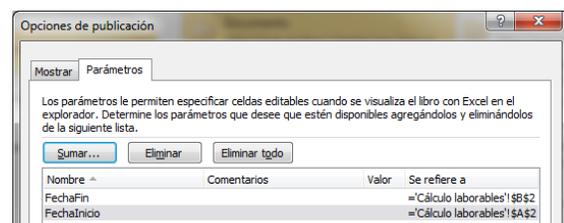
	FechaInicio					
	A	B	C	D	E	F
1	Fecha Inicio	Fecha Fin	Días laborables		Días festivos:	01/01/2010
2			0			06/01/2010
3						22/01/2010
4						18/03/2010
5						19/03/2010
6						02/04/2010
7						06/04/2010
8						01/05/2010
9						15/08/2010
10						09/10/2010

La celda A2 la renombramos a "FechaInicio", la celda B2 a "FechaFin" y la celda C3 a "DíasLaborables". En la columna F guardaremos los días festivos. Y la fórmula que tendremos en nuestra celda "DíasLaborables" quedará de la siguiente manera: `"=DIAS.LAB(FechaInicio;FechaFin;F:F)"`. Esta fórmula nos dirá los días laborables (de lunes a viernes) entre las 2 fechas dadas en las celdas A2 y B2 descontando los días festivos de la columna F. El hecho de renombrar las celdas, además de facilitar la fórmula, es necesario para definir estas celdas como parámetros editables a través de los Servicios de Excel.

Una vez ya tenemos nuestro Excel preparado, vamos a publicarlo en los Servicios Excel de nuestro SharePoint 2010. Para ello, en Archivo'Guardar y enviar'Guardar en SharePoint

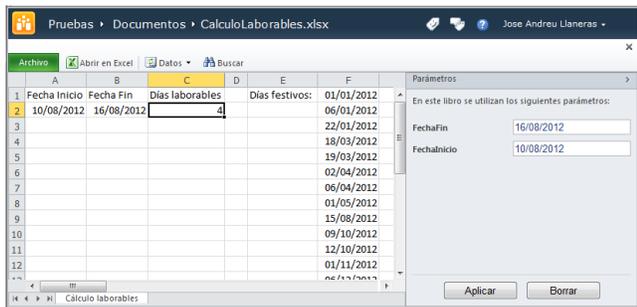


En "Opciones de publicación", se indica qué queremos publicar en los Servicios de Excel y qué parámetros. En este caso publicaremos toda la hoja pero nos podría interesar publicar sólo una gráfica que se basa en fórmulas que no se mostrarían a los usuarios de SharePoint. En parámetros seleccionaremos las 2 celdas de entrada de fechas para que se permitan modificar desde los Servicios de Excel:

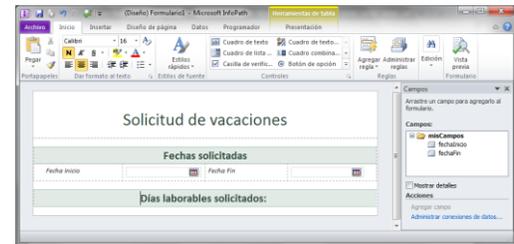


Y usando el botón "Guardar como..." seleccionaremos una biblioteca de documentos de nuestro SharePoint dentro de las Ubicaciones de confianza. Con todo esto, ya podemos aceptar y guardar nuestro libro y acabamos así nuestro trabajo en Excel. Tras publicar nuestro Excel, lo podremos ver en acción en nuestro portal preferido y darle vida dándole valores a los parámetros que hemos definido:





Usaremos la plantilla por defecto de InfoPath 2010 para Biblioteca de formularios de SharePoint. Agregamos dos campos de tipo "Fecha" y los renombramos a "fechaInicio" y "fechaFin" quedando algo parecido a:



Los servicios web de los Servicios de Excel

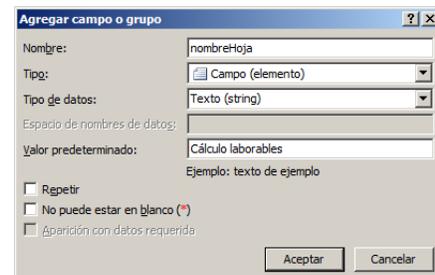
Los Servicios de Excel nos proporcionan unos servicios web para poder interactuar con nuestros libros publicados. Tenemos muchas operaciones, entre ellas, la posibilidad de dar valor a una celda, obtener el valor de otra celda donde se calcula una fórmula, obtener un rango de celdas, etc. A través de la dirección http://{{NombreServidor}}/_vti_bin/ExcelService.aspx podemos acceder a todas las operaciones disponibles. Para probarlos, como hemos comentado antes vamos a consultar nuestro libro de Excel desde un formulario de InfoPath a través de estos servicios web.

Para hacer la posterior llamada al servicio web necesitamos tener como campos (no visibles al usuario) los parámetros que le pasaremos a los métodos del servicio web al hacer la llamada. Por ello, necesitamos agregar estos campos que no mostraremos en el formulario. Para ello, sobre "misCampos" le damos a botón derecho y a "Agregar..." con los siguientes valores:

ExcelService

Las operaciones siguientes son compatibles. Para una definición formal, revise la [descripción de servicios](#).

- [Calculate](#)
- [CalculateA1](#)
- [CalculateWorkbook](#)
- [CancelRequest](#)
- [CloseWorkbook](#)
- [GetApiVersion](#)
- [GetCell](#)
- [GetCellA1](#)
- [GetChartImageUrl](#)
- [GetPublishedItemNames](#)



(Valor predeterminado: Nombre de la hoja de Excel dónde están las celdas de Fecha Inicio y Fecha Fin)

Repetimos la operación para añadir 3 campos más de tipo texto:

- Nombre: nombreFechaInicio Valor predeterminado: FechaInicio.
- Nombre: nombreFechaFin Valor predeterminado: FechaFin.
- Nombre: nombreDiasLaborables Valor predeterminado: DiasLaborables.

Consultemos nuestro Excel desde InfoPath :

Cambiamos de herramienta y vamos a abrir el InfoPath Designer 2010. Aquí crearemos un formulario muy sencillo de ejemplo, con 2 controles de calendario y una celda. Además, agregaremos una conexión de datos con la que nos conectaremos a nuestro Excel publicado en los Servicios de Excel a través de su servicio web y recuperamos el cálculo de los días laborables.

Los valores de estos 3 últimos campos se corresponden con las celdas que renombramos en nuestro Excel al principio del artículo.

Ahora vamos a crear las conexiones de datos al servicio web de los Servicios de Excel y les pasaremos como parámetros los campos que hemos definido. Necesitaremos 3 conexiones, una por cada operación que invocaremos a través del servicio web.

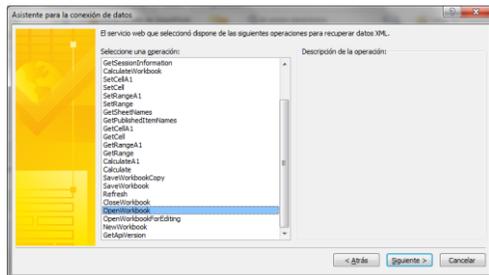




Para ello, en la cinta de "Datos" le damos a "Obtener Datos externos" "desde el servicio web SOAP" y agregaremos la dirección al servicio web:

`http://{NombreServidor}/_vti_bin/ExcelService.asmx`

Veremos todas las operaciones disponibles. La 1ª operación que realizaremos será la de abrir el libro con el que vamos a trabajar (OpenWorkbook); el libro de Excel que publicamos al principio:



Esta operación tiene 3 parámetros, pero sólo obligatorio el parámetro "tns:workbookPath" que es la dirección a nuestro Excel.

Pondremos:

`http://{NombreSitio}/{BibliotecaDocumental}/{archivo}.xlsx`
en mi ejemplo: `http://joseandreu/clientes/pruebas/Documentos/CalculoLaborables.xlsx`

En las siguientes pantallas dejamos desmarcados las opciones de guardar una copia de los datos en la plantilla del formulario y recuperar los datos automáticamente al abrir el formulario y renombramos la conexión a "Abrir libro".

Para introducir valores en un libro que tengamos abierto de Excel Services tenemos 4 métodos en el servicio web: SetCell, SetCellA1, SetRange, y SetRangeA1. Los dos primeros métodos se usan para dar valores a una celda y los dos segundos para dar valores a un rango de celdas. Los métodos que tienen el sufijo "A1" permiten definir la/s celda/s que queremos editar al "estilo Excel" (H3, Hoja 1!A3:C:3). En cambio, los otros 2 métodos se definen las celdas indicándole el número de columna y celda empezando el índice en 0. Como en nuestro ejemplo ya hemos renombrado las celdas podremos identificarlas para darles valor a través de los métodos que tienen el sufijo "A1".

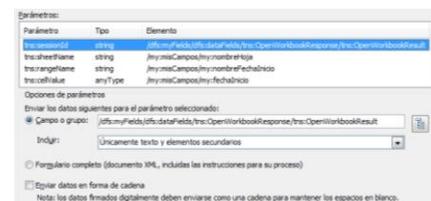
Continuamos ahora con la 2ª conexión, esta vez para enviar a nuestro libro que hemos "abierto" en la 1ª conexión los datos que han rellenado en nuestro formulario de InfoPath (FechaInicio y FechaFin). Necesitaremos una conexión para cada fecha. Añadimos una nueva conexión, pero esta vez seleccionaremos "Enviar formulario" "al servicio web" dentro de "A otras ubicaciones".

Volveremos a introducir la dirección a nuestro servicio web de Excel. Para dar valores a las celdas (FechaInicio, FechaFin) usaremos la operación "SetCellA1" la cual tiene los parámetros: sessionId, sheetName, rangeName y cellValue. El parámetro "sessionId" que identifica la sesión del libro lo obtenemos de la llamada de nuestra primera conexión. Su valor será el campo: `/dfs:myFields/dfs:dataFields/tns:OpenWorkbookResponse/tns:OpenWorkbookResult`.



Al parámetro sheetName le indicaremos el valor de nuestro campo: `/my:misCampos/my:nombreHoja`, al parámetro rangeName: `/my:misCampos/my:nombreFechaInicio` y al parámetro cellValue le indicamos el campo tipo fecha de nuestro formulario: `/my:misCampos/my:fechaInicio`.

Nuestra conexión quedará así:



En la siguiente pantalla renombramos la conexión a "Enviar Fecha Inicio" y desmarcamos la opción de establecer como conexión predeterminada.

Ahora crearemos la 3ª conexión para dar pasar a nuestro libro de Excel la fecha fin, repitiendo los pasos de la 2ª conexión pero con los valores de la fecha fin y la renombraremos a "Enviar Fecha Fin".

Finalmente configuraremos la 4ª conexión para obtener el valor de los días laborables entre las dos fechas calculado en el Excel. Agregamos una conexión para recibir datos a través del servicio web con la operación "GetCellA1". En esta conexión, al ser de obtener valores no podemos definirle los parámetros en función del resultado de otra conexión. Luego le daremos valores mediante reglas al hacer clic sobre el botón que realizará todas las conexiones. En la última pantalla de configuración, desmarcamos la opción de recuperar los datos automáticamente y renombramos la conexión a "Obtener días laborables".

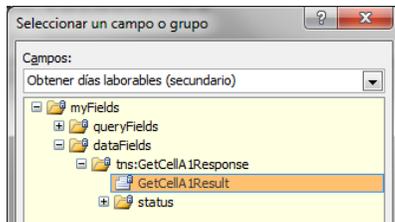




Ya sólo nos queda la última conexión necesaria para cerrar el libro. Será una conexión de envío de datos y usaremos la operación "CloseWorkbook". Tiene el parámetro sessionId, que le daremos el valor /dfs:myFields/dfs:dataFields/tns:OpenWorkbookResponse/tns:OpenWorkbookResult. Renombramos la conexión a "Cerrar libro".

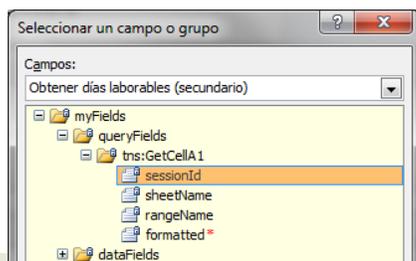
Dejamos ya las conexiones para volver al formulario.

Cuando hemos diseñado el formulario, nos faltaba añadir un botón para ejecutar las llamadas a los métodos de los servicios web y un control para mostrar el resultado de nuestra consulta al libro de los Servicios de Excel. Para mostrar el resultado, agregamos un campo del tipo "Valor calculado". Le pondremos como valor el resultado de la fórmula de la conexión "Obtener días laborables". Le daremos a "Editar fórmula" e Insertamos el campo resultado de la conexión "Obtener días laborables":

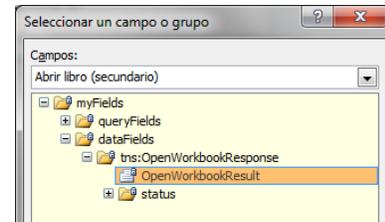


```
xdXDocument:GetDOM("Obtener días laborables")/dfs:myFields/dfs:dataFields/tns:GetCellA1Response/tns:GetCellA1Result
```

Al seleccionar el botón, en la cinta de Office cambiamos la etiqueta para renombrarlo a "Calcular" y vamos a crear las reglas con las que llamaremos a los servicios web de los Servicios de Excel para recuperar el resultado de nuestra fórmula en el Excel. Agregamos una regla con la acción de Consultar de datos con la conexión "Abrir libro" y 2 conexiones para enviar los datos de las fechas del formulario (Enviar fecha Inicio y Enviar fecha Fin). Ahora le daremos valores a los parámetros que de la conexión "Obtener días laborables". Para ello agregamos las acciones de "Establecer el valor de un campo". En campo seleccionamos el SessionID de la conexión:



Y en valor el resultado de la conexión de "Abrir libro":

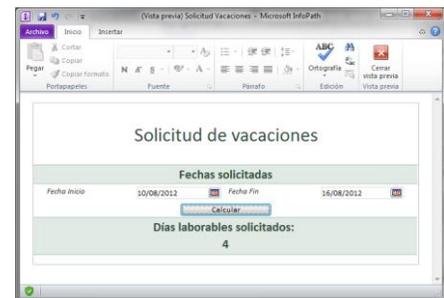


Repetiremos la acción para darle valor al resto de parámetros (sheetName y rangeName) con sus valores respectivos (nombreHoja y nombreDiasLaborables).

Por último agregamos las acciones para llamar a las conexiones de "Obtener días laborables" para recuperar el resultado y "Cerrar libro" para cerrar la sesión.

Como último paso antes de publicar, sólo nos faltará configurar nuestro formulario como "Plena confianza" y firmarlo con un certificado. Lo haremos desde el apartado "Seguridad y confianza" de las opciones del formulario.

Una vez publiquemos el formulario en nuestro SharePoint, al darle al botón "Calcular" hará las llamadas a los servicios de los "Servicios de Excel" pasándole las fechas del formulario y recuperando el resultado de la fórmula de Excel:



Conclusiones

En este artículo, más que enseñar una posible solución para calcular los días laborables entre 2 fechas, hemos querido mostrar cómo podemos usar fórmulas desarrolladas en Excel y poder consultar sus resultados desde una aplicación externa mediante los servicios web de los Servicios Excel.

José Andreu

Responsable desarrollos SharePoint en Aitana

@JoseAndreu

www.mundosharepoint.es

www.aitana.es

jose.andreu@aitana.es





Novedades de Project Server 2013

... las novedades más relevantes de la versión previa de este software...

Junto con la aparición de Windows 8, Microsoft ha actualizado toda su familia de productos, desde el paquete Office hasta Windows Server. Esto incluye SharePoint Server 2013 y su herramienta de gestión de proyectos: Project Server 2013. A continuación hemos querido mencionar las novedades más relevantes de la versión previa de este software.

Novedades en la Arquitectura

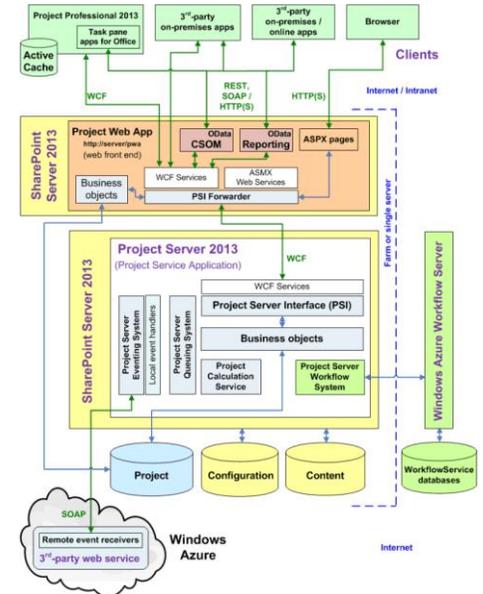
Esta nueva versión de Project Server presenta una serie de cambios en la arquitectura, algunos de ellos motivados por la aparición de Project Server Online, cómo podría ser la reducción de bases de datos. En Project Server 2013 se encuentra una única base de datos a diferencia de las 4 que utilizaba la versión 2010. No obstante, los nombres de las tablas introducen un prefijo que hace referencia a los datos que en ella se almacenan. Estos prefijos son 4 y se corresponden con las antiguas 4 BBDD de Project Server. La clave primaria de las tablas sigue siendo, como en versiones anteriores, un GUID que identifica de forma única a cada registro.

Para los que venimos de Project Server 2010, no hay que preocuparse por las conexiones y consultas que tengamos configuradas desde nuestros Excels publicados con Excel Services. La nueva Base de datos incluye las mismas vistas que la antigua de Reporting.

Más allá de las BBDD, encontramos una mayor relación en el binomio Project Server - SharePoint, que permite la integración del servicio de la aplicación de Project con un site de SharePoint. Esto permite utilizar una lista de tareas de SharePoint como un proyecto empresarial, y tomar el control de este directamente desde la lista, así como actualizar la lista con el cliente Project Professional. Esta característica puede facilitar el acceso de la empresa al uso de Project Server, empezando sólo con gestionar una lista de tareas para más adelante utilizar toda la funcionalidad, o para llevar a cabo pequeños proyectos que no necesitan de un planning más extenso que una serie de tareas.

A diferencia de lo que ocurría con Project Server 2010, la versión 2013 permite compartir una granja de servidores y una infraestructura de SharePoint entre unidades de negocio o clientes distintos. Esto puede ocurrir en un host donde se necesite dar a múltiples clientes la capacidad de manejar sus propios entornos. Por ejemplo, podríamos tener múltiples instancias de Project Server 2013 Preview, una para cada departamento de una organización, y cada una de ellas se ejecuta de manera independiente y aislada de las otras. Ésta funcionalidad de multi-tenancy la aprovecha Project Server de la plataforma que lleva por debajo, SharePoint 2013.

La referencia oficial de Microsoft ofrece el siguiente esquema sobre la arquitectura de Project Server 2013 Preview:



Cabe destacar la mejora en las interfaces de programación, como es el modelo de objetos del lado cliente (CSOM) o las mejoras en la creación de Workflows, aunque todo esto lo trataremos con profundidad más adelante, en la sección dedicada a los desarrolladores.





A nivel de front-end se incluyen aplicaciones de terceros, Project Professional y la Project Web App. Según el fabricante, se han eliminado todas las restricciones de acceso a la PWA, haciendo posible la utilización de toda la funcionalidad de Project Server a través de los navegadores principales (Internet Explorer 8 y superiores, Firefox a partir de la versión 4, Safari a partir de la 5 y Google Chrome).

En la capa intermedia se encuentra la PSI y Business Objects, que de forma conjunta forman toda la lógica de objetos que representa PS2013. Si nos fijamos en el esquema anterior, vemos las interfaces de conexión utilizadas como novedad en esta versión, como CSOM u OData. A excepción de las tablas de Reporting, nunca se debería acceder de forma directa a la BBDD de Project Server, siendo más correcto utilizar la PSI o el modelo de objetos del lado cliente, por ejemplo.

Novedades para profesionales de IT

En Project Server 2013 Preview, los controles administrativos que están más relacionados con IT han sido movidos a la página de configuración del servidor de Project Web App del sitio de Administración Central. Desde ahí el administrador de PMO manejará, por ejemplo, la configuración de la cola de Project Web App, que ahora reside a nivel de la granja de servidores y no a nivel de Web App como ocurría en 2010. Este cambio se ha introducido, además de para hacer más adecuada la página de Server Settings para el administrador de PMO, porque suponen que no todo el mundo debe poder manipular este tipo de configuración, añadido al hecho de que no es un conjunto de configuraciones que se modifiquen muy a menudo.

El objetivo de mover ésta y otras configuraciones a la Administración Central es separar el rol de PMO administrator del de SharePoint IT Pro.

Con respecto a permisos, el nuevo modelo de SharePoint ha sido creado en Project Server 2013, proporcionando a los usuarios una alternativa al modelo tradicional de permisos de Project Server. Una nueva instalación de Project Server 2013 establece por defecto el modelo de permisos de SharePoint, creando grupos que directamente se corresponden con los grupos de seguridad predefinidos en el modelo de permisos de Project. Esto repercute, por ejemplo, en la página de configuración de Project Server App, ya que no mostrará ninguna sección referente a Permisos. Para ello deberíamos seleccionar el modelo 'clásico' de administración de permisos de Project Server (dentro del menú de Project Web App Settings).

El modelo de permisos de Project permite ver los permisos globales y otras configuraciones de seguridad. Como novedad en este aspecto aparece el 'Acceso al servicio de Reporting de Project Server', que permite al usuario acceder al OData service (novedad en el Reporting de Project Server 2013 Preview) para recuperar datos de Project Server.

Cabe mencionar también que se han hecho grandes mejoras sobre la función de sincronización y el 'timer job' sobre la versión previa de Project Server 2013. Estos cambios se han hecho para mejorar el tiempo requerido por las operaciones de sincronización de usuarios del servicio de Active Directory con Project Server Enterprise Resource Pools (ERPs) y los grupos de seguridad de Project Server. Como resultado se obtienen mejoras en el tiempo requerido para la ejecución de esos trabajos de sincronización.

Otra de las nuevas características incluidas en Project Server 2013 Preview es la 'Log Level Manager feature', que nos permitirá rastrear ciertas entidades como proyectos, recursos o tareas. Estas entidades estarán especificadas mediante su GUID a través del Log Level Manager. Una vez configurada la característica, toda actividad relacionada con la entidad en particular que queramos verificar, quedará registrada en los USL Logs de SharePoint, dependiendo del nivel de registro definido.

Sitio de Proyecto

Project Site Feature es una nueva funcionalidad que nos permite tener visibilidad sobre sitios de proyecto de SharePoint 2013 que tengan una lista de tareas en Project Server 2013. Y es que la plantilla de sitio de proyecto de SharePoint 2013 incluye una lista de tareas que estará disponible en Project Server. Esto quiere decir, que existe la posibilidad de importar una lista de tareas de un sitio de SharePoint a Project Web App, que aparecerá en Project Center como un proyecto que añadirá una nueva fila que mostrará la lista de tareas importada.

Esta nueva funcionalidad es ideal para organizaciones que quieran tener visibilidad de proyectos de poco peso en Project Server, permitiendo estar al corriente del estado del mismo, acceder a información, documentos, etc.

Trabajar con Tareas

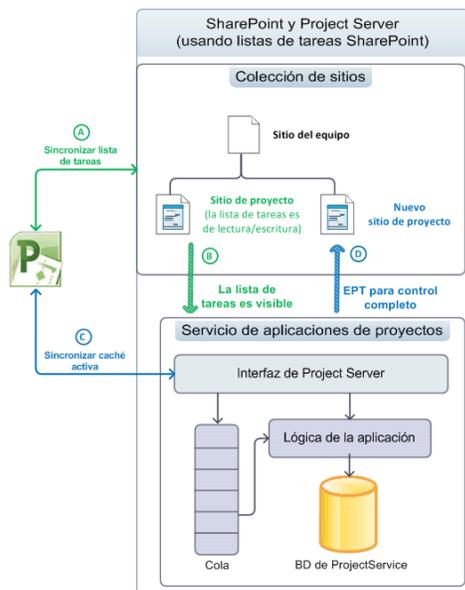
La nueva característica 'My Tasks' permite a los usuarios ver un conjunto de sus tareas de SharePoint, de Project Server y de Exchange/Outlook en una misma localización: la página 'My Tasks' del sitio personal del usuario en SharePoint. Además de navegar por diferentes listas, sitios





y colecciones de sitios para ver o editar tanto tareas de SharePoint como de Project Server, el usuario sólo tendrá que ir a su página personal de 'Mis Tareas' para consultar una única enumeración de cosas que debe hacer.

Esto se consigue mediante un nuevo servicio llamado "Work Management Service", que combina y sincroniza todas las tareas de usuario en su Site personal. Dicho servicio es el responsable de agregar tareas de SharePoint a través de la Búsqueda Empresarial, y tareas de Project Server mediante la aplicación de servicio de Project.



Acceso móvil

Otra de las novedades en Project Server 2013 es el acceso a información de proyecto mediante tu móvil. Ahora se ha incluido un sitio web móvil que permite a los miembros del equipo de trabajo y a los jefes de proyecto ver el estado del proyecto mediante un vistazo en un dispositivo móvil.

Las características para móviles son usadas principalmente por dos tipos de usuarios de Project Server: los 'Team Members', que pueden ver el estado de sus tareas o interactuar con documentos y otra información almacenada en el sitio de proyecto, y los Jefes de Proyecto, que pueden administrar proyectos, estar al día sobre la información de sus recursos, tareas, etc., así como saber el estado del proyecto con un solo vistazo.

Los integrantes del Equipo de proyecto también podrán editar y enviar el estado de sus tareas a través de sus teléfonos mediante Microsoft Exchange. La sincronización entre Exchange Server y SharePoint Server 2013 (en concreto, la página 'My Tasks' mencionada anteriormente) se lleva a cabo mediante el 'Work Management Service'.

Novedades para desarrolladores

Desde el punto de vista de un desarrollador, Project Server 2013 Preview introduce muchas novedades enfocadas a facilitar y agilizar la extensibilidad del producto, cuando este no cubre nuestras expectativas de forma out-of-the-box. PS2013 trabaja sobre la misma base utilizada por sus antecesores Project Server 2007 y Project Server 2010, aunque añadiendo mejoras significativas. Se ha simplificado la API de conexión con los datos de Project Server, o lo que es lo mismo, la Project Server Interface (PSI). Para ello, adopta un modelo de objetos del lado cliente (CSOM), añadiendo librerías de JavaScript y .NET Framework 4.

Apps de Project

Siguiendo la moda de los últimos tiempos, Project Server 2013 permite el desarrollo de Apps. Éstas, además, se pueden crear utilizando la interfaz de usuario Metro, a fin de obtener una experiencia final lo más integrada posible con los nuevos productos de Microsoft. La unión con la suite Office 2013 sigue encabezada por Project Professional 2013, que amplía las capacidades de organización y planificación, aunque también se puede interconectar Project Server 2013 con Word, Excel y Outlook. Y no sólo eso, sino que aprovecha otros elementos del paquete Office, como los Office Art Objects para mejorar la creación de informes. En cualquiera de estos casos, es posible desarrollar Apps tanto para la parte servidor como para la parte cliente.

Una App se basa en un paquete cerrado que realiza ciertas funcionalidades y que es fácilmente desplegable en distintos entornos. Esto hace que sea altamente reutilizable, y permite que se puedan comprar (y vender) en la Office Store, añadiendo así funcionalidades específicas a Project Server 2013 de forma rápida.

Cabe destacar las llamadas Task Pane Apps, que permiten mostrar información del proyecto mediante programación JavaScript. Estas aplicaciones se pueden incorporar a Project Standard 2013 o Project Professional 2013 para aumentar la funcionalidad de los mismos, permitiendo así extender la aplicación de escritorio. Es también posible utilizar estas aplicaciones en la Project Web App (PWA), así como utilizar aplicaciones diseñadas para otros programas del paquete Office 2013.

Integración con SharePoint Server 2013

Project Server 2013 se mantiene sobre la infraestructura de SharePoint. Es por ello que comparte con él ciertas características, y delega en él ciertas funcionalidades, cómo la autenticación, la seguridad mediante los grupos de SharePoint o la sincronización de tareas con listas.





También hace uso del software de edición SharePoint Designer 2013, des del que se puede programar un workflow de forma sencilla para gestionar el ciclo de vida de un proyecto. A diferencia de versiones anteriores, dónde era expresamente necesario, en la mayoría de los casos, realizar un desarrollo a medida mediante Visual Studio, en PS2013 encontramos la facilidad de poder montar el flujo de trabajo de Project Server sobre el modelo de flujo de SharePoint. Además, el propio SharePoint Designer se encarga de conceder los permisos necesarios para que el workflow se invoque de forma correcta si el usuario tiene autorización para ello, y ejecutarse bajo credenciales de otros usuarios sin necesidad de utilizar impersonalización. No hay que olvidar que cuando SharePoint Designer 2013 se nos quede pequeño para la realización de workflows complejos, éstos se podrán seguir desarrollando mediante Visual Studio 2012, cómo se hacía en versiones anteriores de Project Server.

A nivel de plataforma, los flujos de trabajo de SharePoint 2013, y por ello los de Project Server 2013 están contruidos sobre Workflow Foundation 4, y precisan del motor de flujos de trabajo de Windows Azure.

Modelo de objetos del lado cliente (CSOM: Client-side object model)

Cómo hemos comentado anteriormente, PS2013 se encuentra disponible tanto en su modalidad on-premise como online. El acceso por código a esta segunda, es necesario que se realice mediante CSOM, que está construido sobre el CSOM de SharePoint.

Utilizar este modelo de objetos añade múltiples ventajas para el desarrollador, que se traducen, principalmente, en facilidad de programación. En este sentido, podemos entender el CSOM cómo un adaptador a la interface de Project Server (PSI), que nos permite acceder a sus funcionalidades sin necesidad de una llamada específica a ella, sino mediante métodos del modelo. Aunque internamente el modelo llame a la PSI, el acceso a él se implementa como un único servicio de Windows Communication Foundation (WCF), ahorrándonos tener que hacer llamadas a cada uno de los servicios de la PSI. Este modelo, además, evita tener que recurrir a distintas GUID para recuperar los datos, sino que devuelve directamente

los datos por su nombre. También se facilita el acceso a los mismos a través de la clase ProjectContext.

La infraestructura de este modelo permite la utilización de múltiples librerías e interfaces, cómo por ejemplo Microsoft .Net, JavaScript, REST, LINQ, Silverlight y Windows Phone 7.

Capturar eventos

El sistema de eventos de Project Server permite la implementación de Event Handlers en local o Event Receivers en remoto, siendo esta segunda la única opción cuando se trabaja con una instancia de Project Server Online. La creación y administración de los mismos no se realiza des de Project Server, sino que es necesario acceder a la Administración Central de SharePoint. El proceso es parecido al que se realizaba en PS2010, aunque las opciones disponibles son mayores.

Acceso a los datos de Reporting

Uno de los cambios más significativos, si no el mayor, en Reporting de Project Server 2013 es el uso del servicio OData para el acceso a la Project Service Database, siendo esta la única forma de acceder a ella en caso de utilizar Project Server Online. Para las instalaciones on-premise se puede acceder también de forma directa. Internamente, el servicio OData utiliza llamadas SQL, devolviendo los datos en formato XML o JSON.

Como sucedía con versiones anteriores de Project Server, es posible realizar informes de forma más o menos sencilla utilizando la característica PowerPivot de Excel 2013. La cadena de consulta que se almacenará en el fichero .odc se realizará en el formato OData.

Arnau Roca

MCP Project Server 2010

Reme Martínez

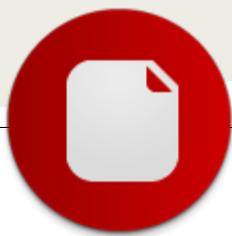
Especialista PMO sobre Project Server 2010

Marc Bàguena

MCP en Project Server 2010

MCITP PRO SharePoint 2010





Portal en SharePoint para Ceibal Impulsando el aprendizaje de 500.000 niños uruguayos

... El programa OLPC (One Laptop Per Child), creado por Nicholas Negroponte, tiene como objetivo entregar a cada niño una computadora de bajo costo conectada a Internet...

Introducción

El programa OLPC (One Laptop Per Child), creado por Nicholas Negroponte, tiene como objetivo entregar a cada niño una computadora de bajo costo conectada a Internet.

Este programa mundial está siendo actualmente implementado en 42 países con 2,4 millones de usuarios (<http://one.laptop.org>). Uruguay ha sido el primer país del mundo en adoptar este programa en forma completa, y posee actualmente más de 500.000 laptops XO en las manos de niños que asisten a escuelas públicas, incluyendo aquellos en las peores condiciones socio-económicas. Todos los esfuerzos de OLPC-UY son coordinados por la organización denominada Centro Ceibal (www.ceibal.org.uy) que fuera creada por el presidente Dr. Tabaré Vázquez en su mandato entre los años 2000 y 2005 (y que continúa dependiendo de Presidencia).

Un poco de historia o "la historia"

Todas estas laptops XO, así como las computadoras que los docentes utilizan, tienen acceso a este portal educativo (www.ceibal.edu.uy) que contiene todos los recursos educacionales necesarios para que los docentes puedan enseñar, y los alumnos aprender, desde sofisticados materiales pedagógicos especialmente confeccionados hasta simples imágenes e información general sobre cómo cuidar de las laptops, sólo por brindar algunos ejemplos.

Ceibal ya contaba con una solución para su portal educativo. Estaba desarrollado en ASP.NET que funcionó correctamente en los primeros años luego de la creación de la institución. Sin embargo, no fue diseñada para hacer frente a la creciente demanda, por lo que se necesitaba una nueva solución que le permitiera crecer, no sólo en número de usuarios accediendo al portal sino también en agregar nuevas funcionalidades y características, todo dentro de una misma plataforma tecnológica.



Asimismo, la nueva solución debía brindar a los usuarios una experiencia de uso similar a la anterior, de forma de evitar cambios que impactaran negativamente en el uso del portal educativo. Esto resultaba muy importante debido a la amplia y diversa audiencia del mismo (desde niños pequeños a maestros experimentados, pasando por poblaciones rurales poco familiarizadas con la tecnología en general) por lo que cambiar de un portal a otro debía hacerse de la forma más transparente para el usuario final desde el punto de vista de su experiencia de usuario.

Detrás de escena, también existía preocupación sobre la migración de todo el contenido que ya estaba presente en el portal anterior hacia el nuevo portal, manteniendo en todo lo posible el material original.

El análisis

Luego de analizar pros y contras de re-diseñar el portal en tecnología ASP.NET o de re-implementarlo en SharePoint, concluimos que ir por una solución nueva en SharePoint era la mejor opción, pero el cliente no podía esperar a terminar completamente el nuevo portal (aun cuando el tiempo total estimado de desarrollo era de un par de meses). Necesitaban urgentemente un portal de back-up que pudiera estar funcionando en pocas semanas, en el caso que el portal vigente en ese momento, dejase de funcionar. En otras palabras, necesitaban ya un nuevo portal en el que pudieran confiar.





La solución fue entonces dividida en dos iteraciones: una primera versión del portal (ya sobre plataforma SharePoint pero con funcionalidades reducidas) que pudiera estar funcionando en semanas; y una segunda versión: el portal educativo completo (como una extensión de la primera versión).

¿Cómo se construyó la solución definitiva?

Para el desarrollo del proyecto se planteó como plataforma base el contar con SharePoint Server 2010 Enterprise y SQL Server 2008 R2. Para el desarrollo se trabajó sobre Microsoft .NET Framework 3.5 con Visual Studio 2010.

Tres temas importantes a resolver:

1. Migración del contenido actual del sitio.
2. Creación de nuevos recursos educativos.
3. Mantener diseño actual.

Para la migración, se evaluaron distintos enfoques:

1. Utilizar una herramienta ya existente de migración automática.
2. Migración manual.
3. Implementar una solución que permita mover el contenido al nuevo portal.

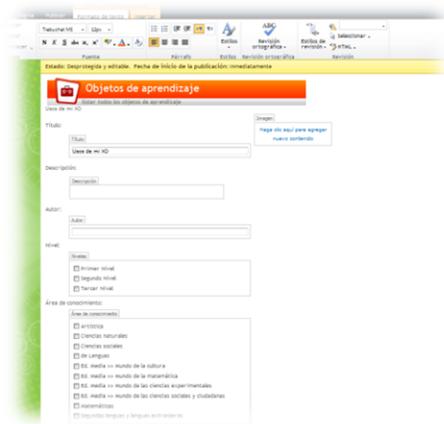
Luego de tomarnos el tiempo de evaluar seriamente la opción 1 y 2, y concluir que no eran viables, nos decidimos por la opción 3. El principal problema con el que nos encontramos fue que por la forma en que se guardaba el recurso educativo en la base de datos debíamos analizar el contenido para detectar que había en él, por ejemplo: imágenes, video, contenidos flash, referencias a otros contenidos, vínculos, texto, etc. La estrategia en este punto, fue ir implementando distintas versiones que fueran afinando el resultado final hasta llegar a un estado con la mínima cantidad de errores. Se desarrolló también una herramienta para detectar referencias (vínculos) dentro de los recursos, que ayudó a ir refinando la herramienta de migración.

Para la creación de nuevos recursos educativos:

Los recursos educativos son publicaciones de una página con descripción y características del tema a tratar en el recurso. Existen varios tipos de recursos dentro del portal, por ejemplo objetos de aprendizaje, experiencias docentes, artículos, libros digitales, etc. Las diferencias entre los tipos de recursos, era la forma de presentación y las columnas de contenido.

¿Cómo los diseñamos en SharePoint?, Creamos diferentes tipos de contenido para representar los diferentes tipos de recursos. Definimos un tipo base que tiene las columnas comunes y utilizamos la herencia de tipo de contenido para crear los tipos específicos. Para cada tipo de contenido creamos un diseño de página a medida que muestra el contenido según se mostraba en el sitio anterior. También para cada tipo de recurso creamos un subsitio dentro de la colección de sitios, para manejar los permisos a nivel de sitio y tener una biblioteca de páginas para cada tipo. Para crear estos recursos, se implementó una WebPart que permite seleccionar el tipo de recurso a crear y resuelve la creación de la página con el diseño de página correspondiente y el sitio donde debe de almacenarse. También habilita los tipos de recursos que puede crear cada usuario.

Cada biblioteca de páginas dentro de los subsitios, cuenta con un flujo de aprobación de publicación y un manejador de eventos. El manejador de eventos lo utilizamos para generar contenido específico para las XO a partir de un archivo que adjuntan los docentes cuando crean los recursos.



Para la autenticación de los usuarios:

Uno de los requerimientos del portal es dar la posibilidad de comentar los recursos educativos que crean los docentes y es necesario que los usuarios estén registrados en el sitio para poder realizarlo. Para esto se configuró la autenticación por formularios y se desarrollaron WebParts específicas para:

- Registrar un nuevo usuario
- Recuperar contraseña de un nuevo usuario
- Modificar los datos de un usuario





A su vez, se desarrolló una WebPart para la creación de nuevos comentarios y la visualización de los mismos. Esta WebPart se dispuso en cada uno de los diseños de página de los recursos (páginas). La misma se asocia a una lista de comentarios, donde cada ítem de la lista corresponde a un comentario y se relaciona al contenido mediante el ID de la página. Esta WebPart además de controlar que los usuarios estén autenticados en el portal para ingresar un nuevo comentario, cuenta con un diccionario de 'palabras prohibidas' contra el cual chequea y filtra el contenido de los comentarios.

Para presentar el contenido con diferentes presentaciones:

Debido que el contenido de la portada del sitio varía periódicamente, se desarrolló una WebPart para facilitar esta tarea al personal encargado de mantener el sitio. Dicha WebPart permite seleccionar la lista de dónde obtener el contenido a mostrar y el aspecto a tomar o sea la forma que muestra la información (por ejemplo, 2, 3 o 4 columnas). Para esto se definió un template de lista con las columnas necesarias y mediante la configuración de la WebPart se permite variar el aspecto que toma.

Otras características implementadas:

Como cualquier sitio de publicación, tuvimos que modificar el diseño de SharePoint. Si bien el diseño que teníamos que aplicar ya estaba definido (tenía que tomar el mismo que tenía el portal), tuvimos que definir la página maestra, crear los diferentes diseños de página, hojas de estilo y algunos JavaScripts para adaptarnos a todos los requerimientos del portal.

También hubo que personalizar la búsqueda, mediante el desarrollo de una WebPart de búsqueda que hereda de las WebParts de búsqueda existentes, así creamos una página de búsqueda especial para el portal.

Otra de las personalizaciones realizadas, fue utilizar la WebPart de consulta de contenido para implementar listados de recursos, modificando sus propiedades para mostrar información de distintos orígenes.

Conclusión

Si bien la primera versión tenía el objetivo de back-up, debió ser puesta en producción inmediatamente, pues el tráfico se duplicó y pasó de 8.000 visitas diarias a más de 20.000 (debido a que en Uruguay las clases comienzan en Marzo) y la solución vigente en ese momento no resultaba confiable.

La versión completa del portal ya está en producción desde Mayo 2012 extendiendo las funcionalidades de la primera.

Si bien los beneficios de la adopción de SharePoint se verán en los próximos meses, ya las primeras versiones del portal están en funcionamiento, sirviendo a más de 20.000 niños y docentes por día, con la transición requerida por el cliente.

Finalmente, el destacable el time-to-market que pudimos lograr con la primera versión, mediante la utilización de múltiples características out-of-the-box de SharePoint, y la solución en dos fases permitió resolver dos problemas que a priori parecían opuestos: entregar rápidamente un nuevo portal educativo (urgente) pero también proveer una solución de clase mundial que acompañe el crecimiento de la institución (importante).

De esta forma es como en este momento más de medio millón de niños y docentes de primaria y secundaria a lo largo de todo el país (no sólo en zonas urbanas) están utilizando SharePoint para e-learning.

Leticia Jawad

Coordinadora de Proyectos SharePoint

Jorge Corral

Preventa Técnica

Arkano Software



2013





Compartir aplicaciones de servicio entre granjas SharePoint 2010

... Una gran característica de SharePoint 2010 es la posibilidad de compartir las aplicaciones de servicio de una granja de servidores con otra granja de servidores...

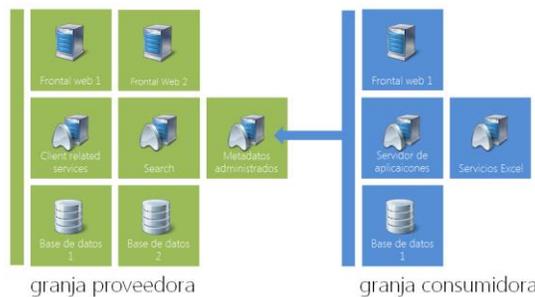
Una gran característica de SharePoint 2010 es la posibilidad de compartir las aplicaciones de servicio de una granja de servidores con otra granja de servidores.

Esto nos permite tener entornos totalmente independientes pero que compartan servicios como los metadatos, las búsquedas, los perfiles, etc. Así es posible optimizar los recursos y máquinas de la empresa y evitar redundancia.

Es posible compartir las siguientes aplicaciones de servicio:

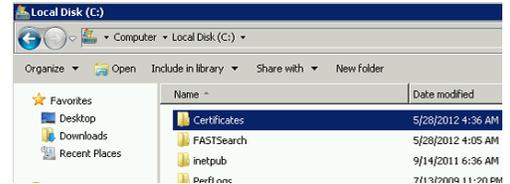
- Servicio de Perfiles de usuario.
- Servicio de Metadatos Administrados.
- Servicio de Web Analytics.
- Búsquedas.
- Servicio de almacenamiento seguro.
- Conectividad a Datos Empresariales (BCS).

Vamos a verlo con un ejemplo. Vamos a suponer dos granjas SharePoint, las llamaremos "Granja proveedora" y "Granja consumidora". La granja proveedora va a compartir su servicio de metadatos y la granja consumidora conectará al servicio para utilizarlo:

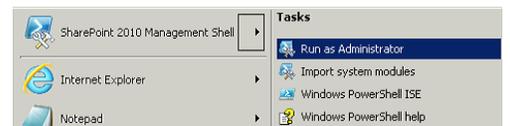


- Para que ambas granjas confíen entre sí, deben proporcionarse una serie de certificados entre ellas. La granja consumidora debe proporcionar dos certificados de confianza a la granja proveedora: un certificado raíz y un certificado de servicio de token de seguridad (STS). La granja proveedora también debe proporcionar un certificado raíz a la granja consumidora. Al intercambiar certificados, cada granja de servidores reconoce que puede confiar en la otra.

- El primer paso consiste en crear un directorio en un servidor de cada una de las granjas. Para el ejemplo hemos usado C:\Certificates:



- Y también es necesario ejecutar el PowerShell de Administración de SharePoint en ambas granjas. Es necesario ejecutarlo como administrador:



- En la granja CONSUMIDORA ejecutaremos el siguiente script PowerShell para exportar el certificado raíz:

```
$rootCert = (Get-SPCertificateAuthority).RootCertificate
$rootCert.Export("Cert") | Set-Content
c:\certificates\consumer-root.cer -encoding byte
```

- También ejecutaremos la siguiente secuencia PowerShell para exportar el certificado de servicio de token de seguridad (STS):

```
$stsCert = (Get-SPSecurityTokenServiceConfig).LocalLoginProvider.SigningCertificate
$stsCert.Export("Cert") | Set-Content
c:\certificates\consumer-sts.cer -encoding byte
```

Podemos comprobar el resultado mirando las variables \$rootCert y \$stsCert:





```

Administrator: SharePoint 2010 Management Shell
PS C:\> $rootCert = (Get-SPCertificateAuthority).RootCertificate
PS C:\> $rootCert.Export("Cert") | Set-Content c:\certificates\consumer-root.cer -encoding byte
PS C:\> $stsCert = (Get-SPSecurityTokenServiceConfig).LocalLoginProvider.SigningCertificate
PS C:\> $stsCert.Export("Cert") | Set-Content c:\certificates\consumer-sts.cer -encoding byte
PS C:\> $rootCert

Thumbprint                               Subject
-----
2F251D55ED062705C0D1C06ECD70D43324D08A734 CN=SharePoint Root Authority, OU=S...

PS C:\> $stsCert

Thumbprint                               Subject
-----
E6B0F3620E4EB0707C2C3ACB83785EFBE3D764CF CN=SharePoint Security Token Servi...

PS C:\> _
    
```

- A continuación, vamos a la granja PROVEEDORA y ejecutamos estas líneas para exportar el certificado raíz:

```

$rootCert = (Get-SPCertificateAuthority).RootCertificate
$rootCert.Export("Cert") | Set-Content c:\certificates\provider-root.cer -encoding byte
    
```

- Mediante la variable \$rootCert podemos comprobar el resultado:

```

Administrator: SharePoint 2010 Management Shell
PS C:\> $rootCert = (Get-SPCertificateAuthority).RootCertificate
PS C:\> $rootCert.Export("Cert") | Set-Content c:\certificates\provider-root.cer -encoding byte
PS C:\> $rootCert

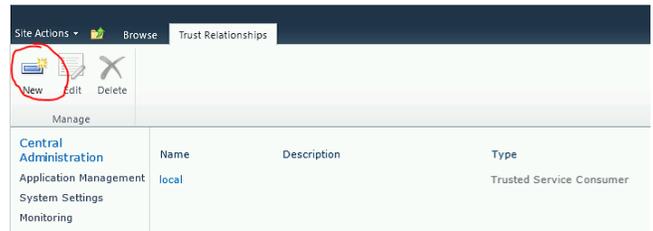
Thumbprint                               Subject
-----
5A6D614C4AD505CF5E298925AD0411A8E0C6B4A0 CN=SharePoint Root Authority, OU=S...

PS C:\> _
    
```

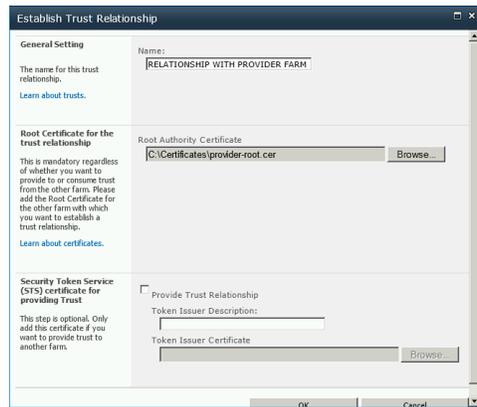
- El siguiente paso consiste en copiar los certificados de C:\Certificates de cada granja en la otra, de forma que la granja proveedora tenga los dos certificados de la consumidora y la consumidora tenga el certificado de la proveedora. A continuación, en la granja CONSUMIDORA, abrimos Administración Central ' Security ' Manage trust:



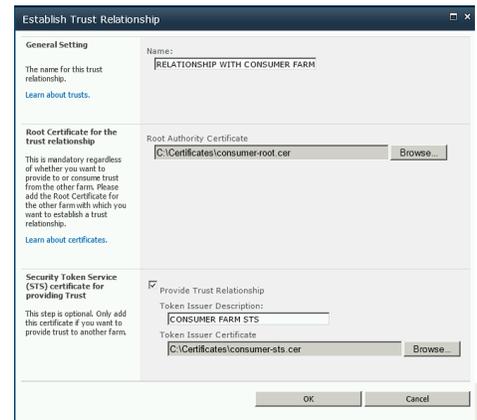
- Entraremos en la sección de relaciones de confianza, donde le especificaremos que debe confiar en la granja proveedora. Hacemos clic en "New":



- Se abrirá el formulario para establecer la relación de confianza. Escribimos un nombre para la relación y buscamos el certificado raíz que acabamos de traer desde la granja proveedora. La sección para certificado STS no debe marcarse. Debe quedar así:



- Nos desplazamos a la granja PROVEEDORA y hacemos lo mismo con los certificados de la consumidora. Esta vez debemos añadir el certificado raíz y el STS:



- Llegados a este punto, según las guías de TechNet, las granjas ya están preparadas para que una comparta los servicios y otra los consuma. Sin embargo, si intentamos realizar esta conexión obtendremos el siguiente error:





"Unable to connect to the specified address. Verify the URL you entered and contact the service administrator for more details."

- Si miramos los log, será más fácil entender el problema:

"An exception occurred when calling SPTopologyWebServiceApplicationProxy.EnumerateSharedServiceApplications on service https://SERVERNAME:32844/Topology/topology.svc : System.ServiceModel.Security.SecurityAccessDeniedException: Access is denied."

- Para que este problema no ocurra, es necesario que la granja consumidora tenga permisos en la aplicación de servicio de Topología de la granja proveedora. Para ello, en la granja CONSUMIDORA ejecutamos:

```
(Get-SPFarm).Id
```

- Esto nos devolverá el guid de la granja consumidora:

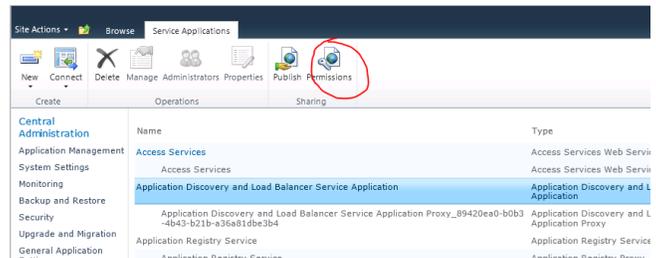
```
Administrator: SharePoint 2010 Management Shell
PS C:\> (Get-SPFarm).Id
Guid
513d22cd-1c24-41a4-a4fa-a7092bf87115
```

- Ahora, en la granja PROVEEDORA ejecutamos el siguiente script, donde <farmid> es el guid que acabamos de obtener de la granja consumidora:

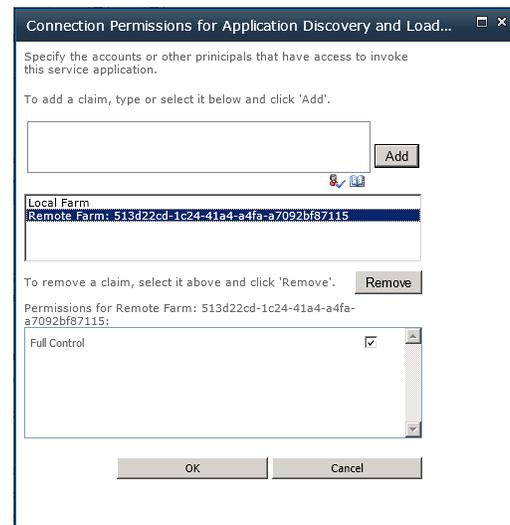
```
$security = Get-SPTopologyServiceApplication | Get-SPServiceApplicationSecurity
$claimProvider = (Get-SPClaimProvider System).ClaimProvider
$principal = New-SPClaimsPrincipal -ClaimType http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid -ClaimProvider $claimProvider -ClaimValue <farmid>
Grant-SPObjctSecurity -Identity $security -Principal $principal -Rights "Full Control"
Get-SPTopologyServiceApplication | Set-SPServiceApplicationSecurity -ObjectSecurity $security
```

```
Administrator: SharePoint 2010 Management Shell
PS C:\> $security = Get-SPTopologyServiceApplication | Get-SPServiceApplicationSecurity
PS C:\> $claimProvider = (Get-SPClaimProvider System).ClaimProvider
PS C:\> $principal = New-SPClaimsPrincipal -ClaimType http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid -ClaimProvider $claimProvider -ClaimValue 513d22cd-1c24-41a4-a4fa-a7092bf87115
PS C:\> Grant-SPObjctSecurity -Identity $security -Principal $principal -Rights "Full Control"
PS C:\> Get-SPTopologyServiceApplication | Set-SPServiceApplicationSecurity -ObjectSecurity $security
PS C:\> _
```

- Para asegurarnos que ha funcionado, podemos ir a la Administración Central ' Manage Service Applications, marcar la aplicación de servicio de topología (a.k.a. Application Discovery and Load Balance) y hacer clic en "Permissions":

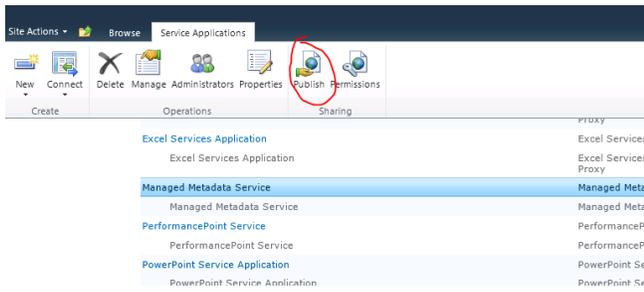


- Veremos que la aplicación tiene permisos para la granja remota:

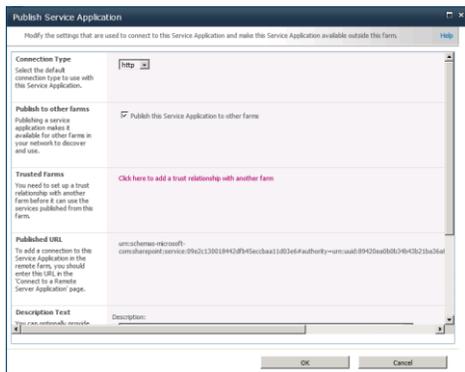


- Este paso adicional para dar permisos al servicio de topología no está documentado en Technet y ha sido posible explicarlo gracias a Spence Harbar, de harbar.net (<http://harbar.net/archive/2010/05/03/service-application-federation-with-sharepoint-2010.aspx>). Ya están ambas granjas preparadas. El último paso es publicar la aplicación de servicio de metadatos en la granja proveedora y conectar a él desde la granja consumidora.
- Para ello, en la granja PROVEEDORA, iremos a Administración Central ' Manage Service Applications. Seleccionamos la aplicación de servicio de metadatos y hacemos clic en el botón "Publish":

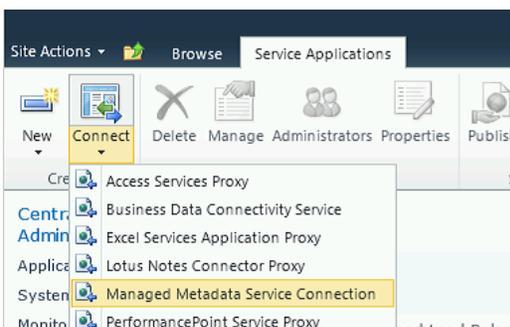




- Se abrirá el formulario. Simplemente debemos marcar el check para publicar a otras granjas y asegurarnos que copiamos el enlace "Published URL", ya que es el enlace que debemos darle a la granja consumidora. En la sección "Trusted Farms" no debemos hacer nada, ya que lo hemos configurado anteriormente:



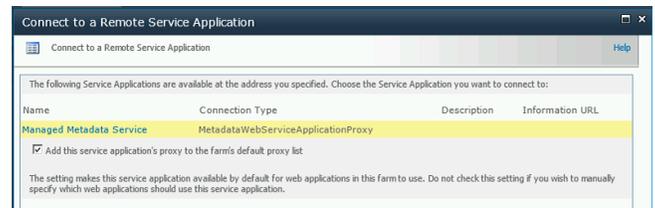
- Y ya por fin, en la granja CONSUMIDORA, en la Administración Central 'Manage Service Applications', desplegamos las opciones del botón "Connect" y especificamos que queremos conectar a un servicio de Metadatos Administrados:



- En el formulario, pegamos la URL de la aplicación de servicio de la granja proveedora y pulsamos OK:



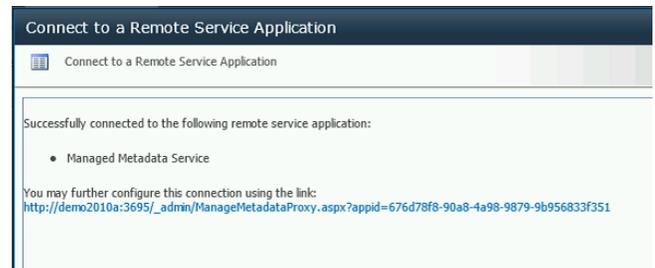
- Si todo ha ido bien, veremos la siguiente ventana, en la que marcaremos el check y pulsaremos OK:



A continuación, es posible especificar un nombre para la conexión. Personalmente, me gusta dejar el que nos pone, es bastante claro. Pulsamos OK:



- Y por último, nos mostrará un mensaje informando que todo ha ido bien:

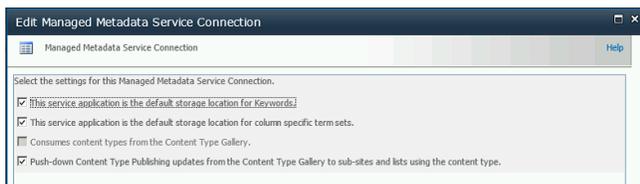


- La conexión de cada aplicación de servicio tiene unas propiedades. Es necesario configurar estas propiedades. Para ello, marcamos la conexión que acabamos de crear y le damos al botón "Properties":





- No nos asustemos, dependiendo del servicio que sea veremos unas opciones distintas (o incluso no veremos ninguna). En el caso de la conexión al servicio de metadatos, las opciones serán:



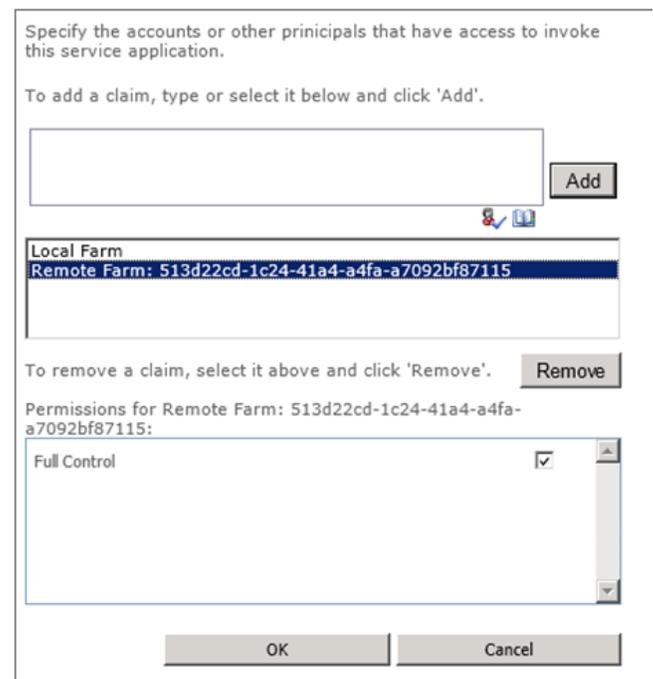
- Si vemos esta ventana, es que todo ha quedado perfectamente configurado.
- En el caso de obtener un error en la ventana de propiedades, o de tener errores de permisos al realizar la conexión, es posible que no hayamos dado los permisos mencionados anteriormente al servicio de topología. Si los hemos dado y seguimos con problemas, es posible otorgar permisos directamente a la aplicación de servicio en concreto que queremos compartir. Para ello, en la granja PROVEEDORA debemos ejecutar lo siguiente:

```

$security = Get-SPTopologyServiceApplication | Get-SPServiceApplicationSecurity
$claimProvider = (Get-SPClaimProvider System).ClaimProvider
$principal = New-SPClaimsPrincipal -ClaimType http://schemas.microsoft.com/sharepoint/2009/08/claims/farmid -ClaimProvider $claimProvider -ClaimValue b50f9a39-cfa9-453e-9d65-6da6a0e400bd
Grant-SPOjectSecurity -Identity $security -Principal $principal -Rights "Full Control"
Get-SPServiceApplication |?{$_.TypeName -eq "Managed Metadata Service"} | Set-SPServiceApplicationSecurity -ObjectSecurity $security
    
```

Es importante fijarse que en el script se busca la aplicación por su nombre (en este caso "Managed Metadata Service"). Si tiene otro nombre, o es otra, debemos cambiar este texto.

- Si todo ha ido bien, veremos que la aplicación de servicio de metadatos tiene permisos otorgados a la granja consumidora:



Miguel Tabera Pacheco

SharePoint Architect en Beezy & Spenta
 www.sinsharepointnohayparaiso.com
 @migueltabera



SharePoint Communities



¿Desea colaborar con CompartiMOSS?

La subsistencia del magazine depende de los aportes en contenido de todos. Por ser una revista dedicada a información sobre SharePoint en español, todo el contenido deberá ser directamente relacionado con Microsoft SharePoint Services (WSS) y/o Microsoft Office SharePoint Server (MOSS) y escrito en castellano. No hay limitaciones sobre el tipo de artículo o contenido, lo mismo que sobre el tipo de versión.

Si desea publicar algo, por favor, utilice uno de los siguientes formatos:

- Artículos de fondo: tratan sobre un tema en profundidad. Normalmente entre 2000 y 3000 palabras y alrededor de 4 o 5 figuras. El tema puede ser puramente técnico, tanto de programación como sobre infraestructura, o sobre implementación o utilización.
- Artículos cortos: Máximo 1000 palabras y 1 o 2 figuras. Describen rápidamente una aplicación especial de SharePoint, o explica algún punto poco conocido o tratado. Experiencias de aplicación de SharePoint en empresas o instituciones puede ser un tipo de artículo ideal en esta categoría.
- Ideas, tips y trucos: Algunos cientos de palabras máximo. Experiencias sobre la utilización de SharePoint, problemas encontrados y como solucionarlos, ideas y trucos de utilización, etc.

Los formatos son para darle una idea sobre cómo organizar su información, y son una manera para que los editores le den forma al magazine, pero no son obligatorios. Los artículos deben ser enviados en formato Word (.doc o .docx) y las figuras por separado en un formato de alta resolución (.tif), todo comprimido en un archivo (.zip o .rar) con el nombre del autor y del artículo.

Si desea escribir un artículo de fondo o corto, preferiblemente envíe una proposición antes de escribirlo, indicando el tema, aproximada longitud y número de figuras. De esta manera evitaremos temas repetidos y permitirá planear el contenido de una forma efectiva.

Envíe sus proposiciones, artículos, ideas y comentarios a la siguiente dirección:

fabiani@siderys.com.uy
jgonzalez@gruposodercan.es
gustavo@gavd.net
compartimoss@hotmail.com

