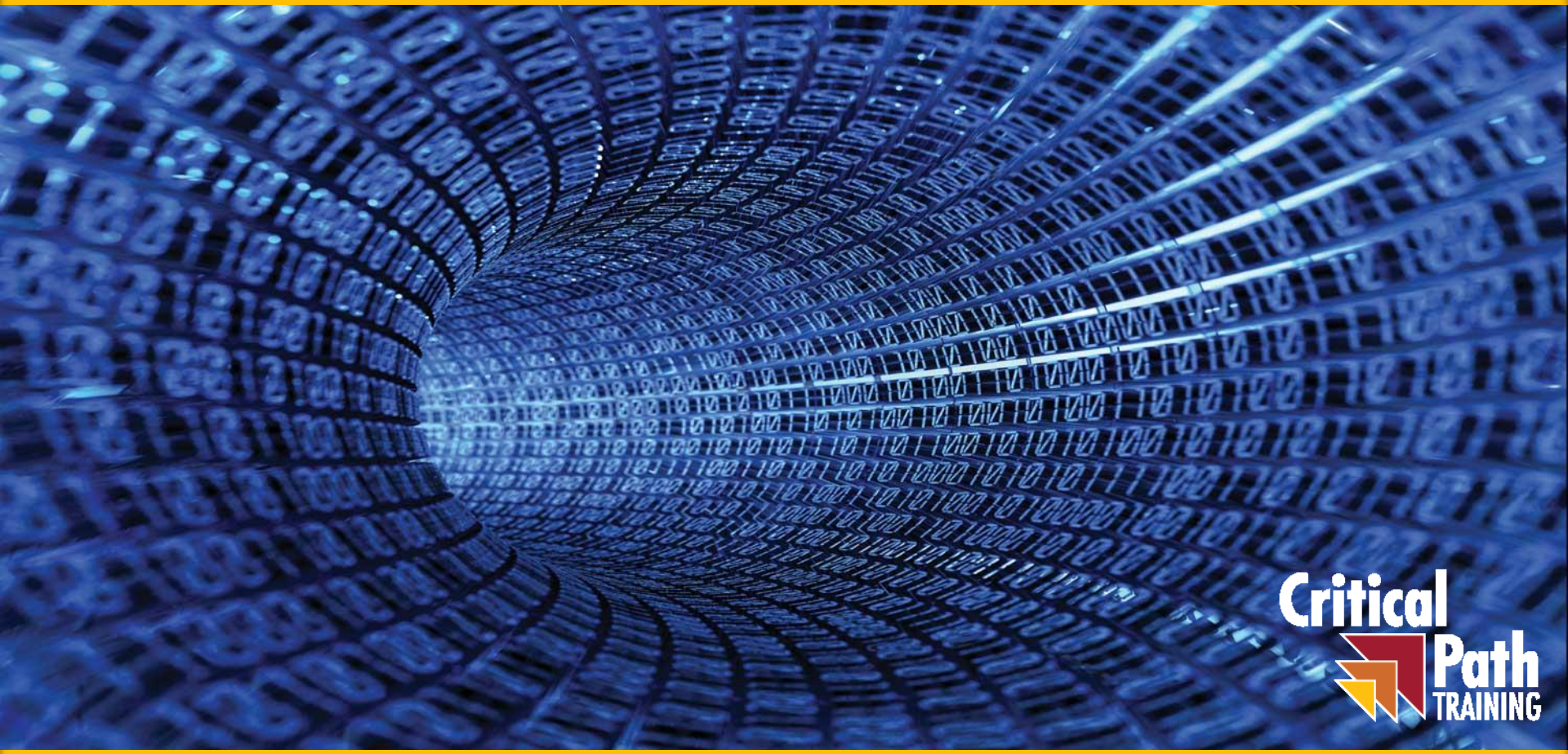
Enterprise Content Management



Agenda

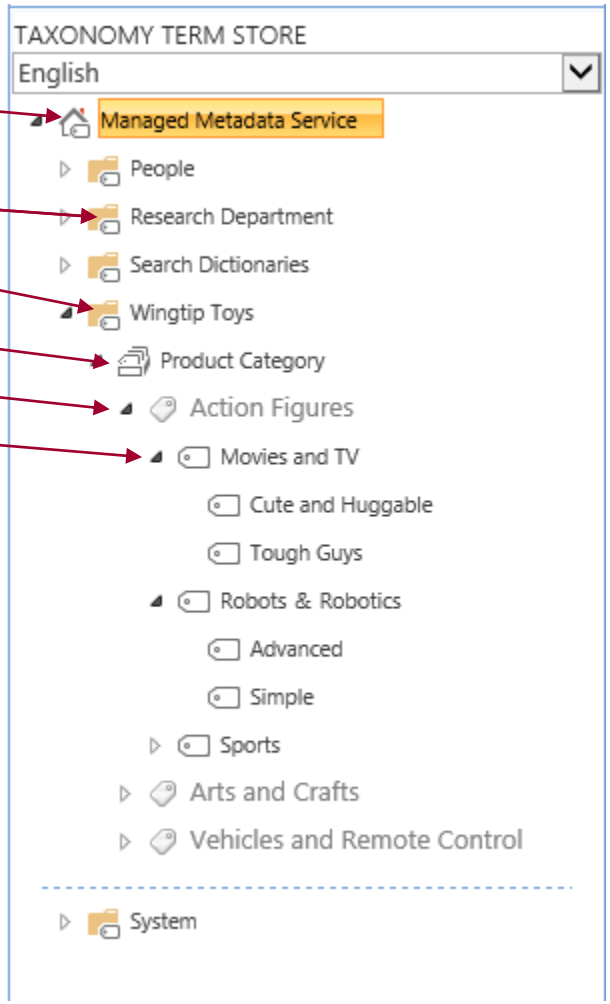
- Managed Metadata
- Enterprise Content Types
- Document Sets
- Records Management
- eDiscovery
- Programming with Managed Metadata

Managed Metadata Service

- Term Store Management
  - Default Keyword Store
  - Shared Enterprise Term Store
  - User Profile Service Term Store
- Enterprise Content Types
  - Syndication of Content Types
  - Content Type Publishing (Push Down)

Understanding Terms and Term Sets

- Term Store
  - Term Group
    - Term Set
      - **Term**
        - Term
      - **Term**



Managed Metadata Improvements

- Metadata leveraged in various ways throughout SharePoint 2013
- New pages introduced so not everyone has to use Term Store Manager to modify taxonomies
  - Permissions for groups
    - SharePoint 2010 allowed read
    - SharePoint 2013 supports read/write
- Numerous features based on taxonomy targeting WCM scenarios
- Ability to flag a term set's "intended use"
- Taxonomy API exposed via CSOM

Metadata Manager

- Provides UI for managing term sets and terms
  - Import of term sets and terms
  - Manage custom properties
  - Translations & synonyms
- Manage term set / term languages
- Submission policy (open / closed)
  - Open means users can submit terms to the term store (when adding / editing items)
  - Regardless of the policy, users can always submit keywords

Creating a Taxonomy

- Steps to creating a taxonomy
  - Create a new group
  - Create a new term set
  - Create top-level terms
  - Create hierarchy of child terms


DEMO
Working with Managed Metadata

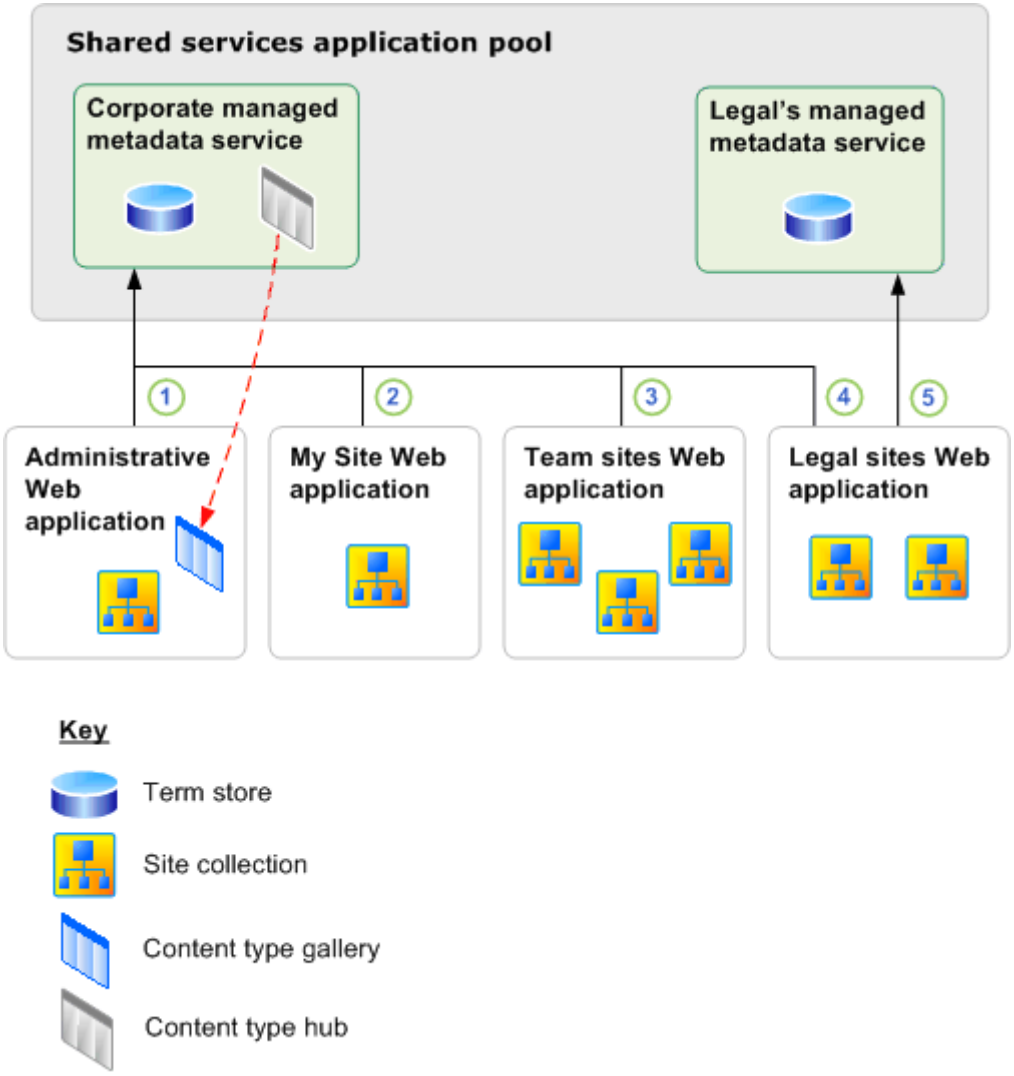
Agenda

- ✓ Managed Metadata
- ➢ Enterprise Content Types
- Document Sets
- Records Management
- eDiscovery
- Programming with Managed Metadata

Enterprise Content Types

- Central Store of Content Types
- Published to Subscribed Site Collections
- Republished when Source is Updated
- In Subscriber Site
  - Cannot Edit Source Content Type
  - Create New Content Types that Inherit

Enterprise Content Types




DEMO
Enterprise Content Type Publishing

Agenda

- ✓ Managed Metadata
- ✓ Enterprise Content Types
- ➢ Document Sets
- Records Management
- eDiscovery
- Programming with Managed Metadata

Document Sets

- SharePoint Server 2007 treated documents as atomic & not linked to other documents
- SharePoint Server 2010 introduced document sets, or a collection of documents, spreadsheets, presentations, etc. that make up a single work product
- Metadata exists on individual items and the set as a whole

Document Sets

- Each document set has:
  - List of available content types allowed within it
  - Default content automatically added to the set
- Can create shared columns (defined in document set's content type) that are pushed down across all content in set
- Welcome page acts as the homepage
  - Customizable Web Part Page displaying the document set's properties

Document Sets – Templates, Versioning

- Document Templates:
  - Admins can provide users document templates for creating new items for the work product
  - Templates created with Visual Studio
- Document Set Versioning:
  - Set owners can capture the state of the set at different points in the lifecycle
  - Ability to see point-in-time history of the set
  - Set owners can restore to a previous version of the set
- Workflows
  - Special OOTB activities for working with document sets


DEMO
Document Sets

Agenda

- ✓ Managed Metadata
- ✓ Enterprise Content Types
- ✓ Document Sets
- ➢ Records Management
- eDiscovery
- Programming with Managed Metadata

In Place Records Management

- SharePoint Server 2007 introduced records management functionality tied to Records Center site template
- SharePoint Server 2010 moved this to a Feature for use in any site template:
  In Place Records Management
- Define who can & can’t declare records

Content Organizer

- Site owners configure rules for new content
- Content authors create content, let the organizer move it to the desired location
- Automatically create subfolders in libraries for lots of content (ie: Press Releases 2008/2009)
- Enabled with site collection scoped Feature
- Managed from each site’s Site Settings page

Unique Document ID Service

- New site collection Feature: Document ID Service
- Adds unique ID for all documents throughout the site collection
- Documents can be retrieved regardless of the current of future location based on their unique ID
- Activating the feature adds 3 columns to the Document and Document Set content types
  - Changes pushed down to all derived content types
  - IDs assigned to existing items as a delayed process
- Adds a Web Part to help find by ID vs. URL: Find By Document ID

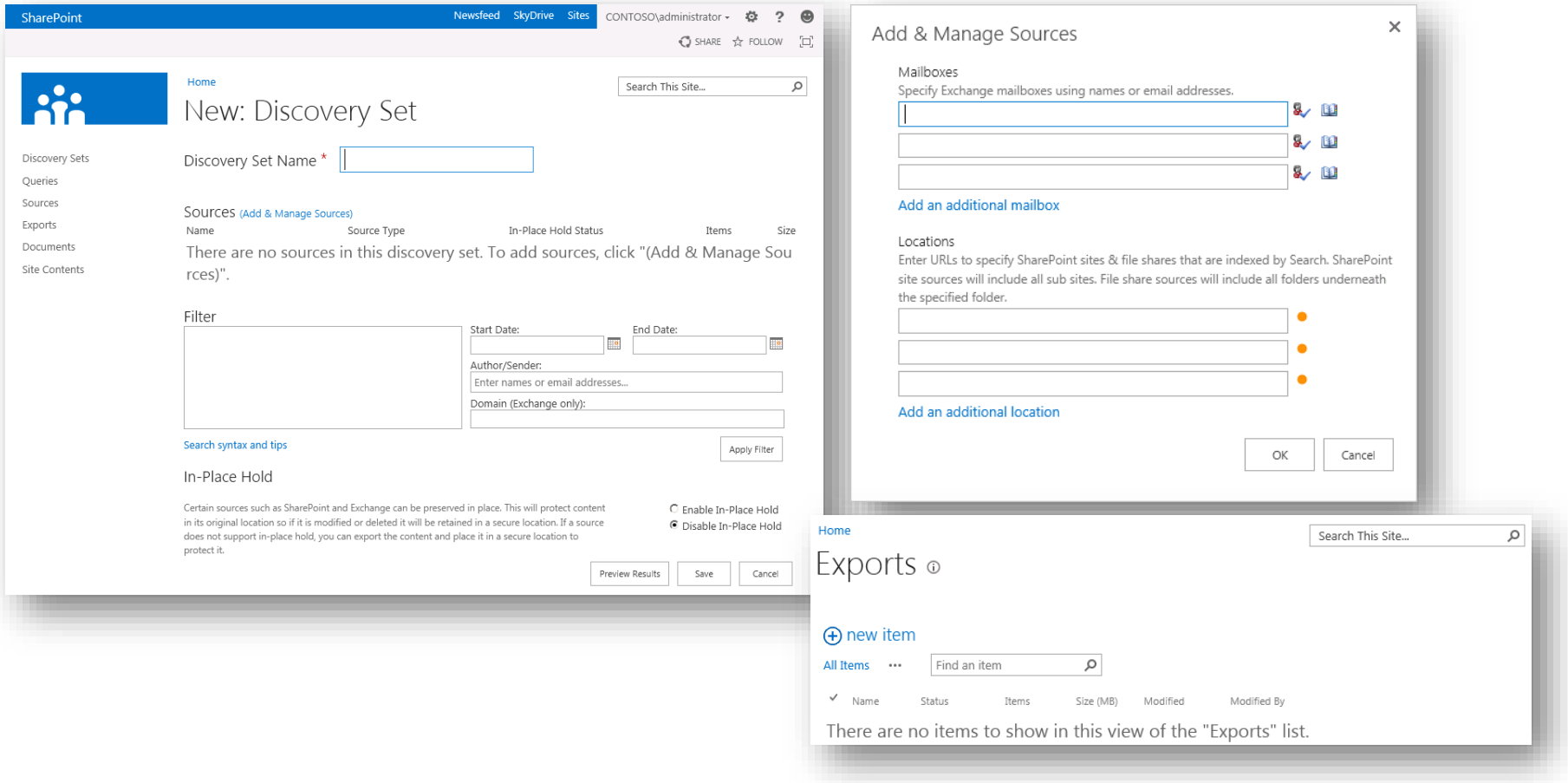
Agenda

- ✓ Managed Metadata
- ✓ Enterprise Content Types
- ✓ Document Sets
- ✓ Records Management
- ➢ eDiscovery
- Programming with Managed Metadata

eDiscovery

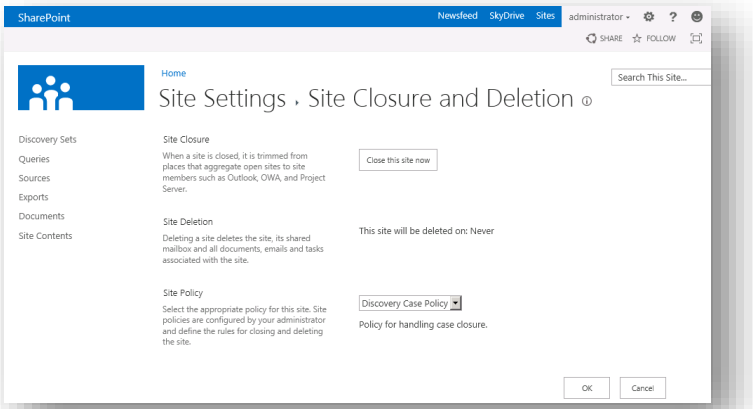
- eDiscovery is the process of finding content relevant to a specific topic
  - For example: legal action / litigation
- Companies also employ internally
  - Ensure they are in compliance
  - Ensure the appropriate policies are in place
  - Ensure they understand their IT & content sources

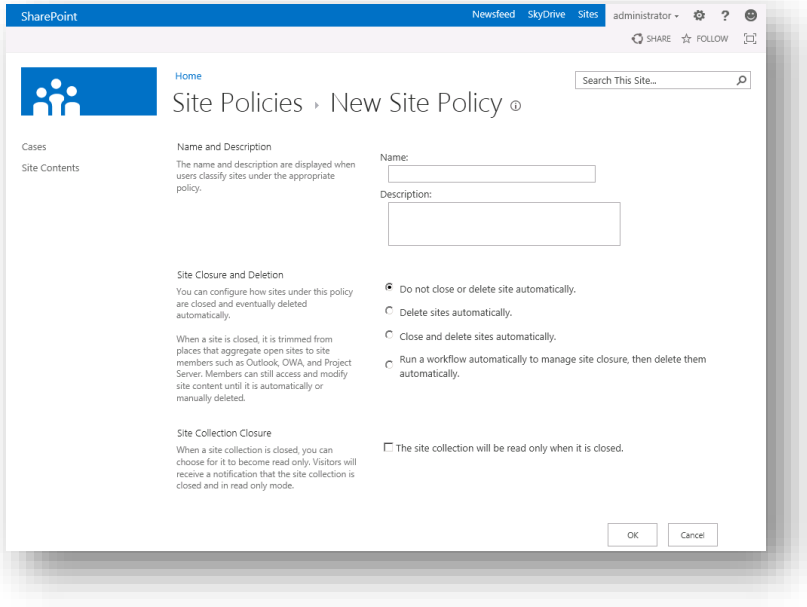
Add, Manage & Export Discovery Sets



Site Based Compliance & Preservation

- Compliance officers create policies, which define:
  - The retention policy for the entire site and the Site Mailbox, if one is associated with the site
  - What causes a project to be closed
  - When a project should expire





Agenda

- ✓ Managed Metadata
- ✓ Enterprise Content Types
- ✓ Document Sets
- ✓ Records Management
- ✓ eDiscovery
- ➢ Programming with Managed Metadata

Managed Metadata CSOM

- SharePoint 2013 CSOM has support for taxonomy
- Add references to:
  - `Microsoft.SharePoint.Client.dll`
  - `Microsoft.SharePoint.Client.Runtime.dll`
  - `Microsoft.SharePoint.Client.Taxonomy.dll`

Using the Local Termset Groups

```
static TermGroup GetSiteCollectionTermGroup(ClientContext clientContext, Site siteCollection)
{
  TaxonomySession taxonomySession = TaxonomySession.GetTaxonomySession(clientContext);
  taxonomySession.UpdateCache();

  clientContext.Load(taxonomySession, ts => ts.TermStores);
  clientContext.ExecuteQuery();

  TermStore termStore = taxonomySession.TermStores.FirstOrDefault<TermStore>();
  Guid localTermStoreID = termStore.Id;
  TermGroup termGroup = termStore.GetSiteCollectionGroup(siteCollection, true);
  clientContext.Load(termGroup);
  clientContext.Load(termGroup.TermSets);
  clientContext.ExecuteQuery();
  return termGroup;
}
```

Creating a Termset

```
static TermSet CreateTermset(ClientContext clientContext, TermGroup termGroup, string termSetName) {

  // delete termset if it already exists
  foreach (TermSet termset in termGroup.TermSets) {
    if (termset.Name.Equals(termSetName)) {
      termset.DeleteObject();
      termGroup.TermStore.CommitAll();
      clientContext.ExecuteQuery();
    }
  }

  Guid termSetId = Guid.NewGuid();
  TermSet newTermSet = termGroup.CreateTermSet(termSetName, termSetId, 1033);
  newTermSet.IsOpenForTermCreation = true;
  termGroup.TermStore.CommitAll();
  clientContext.Load(newTermSet);
  clientContext.ExecuteQuery();

  return newTermSet;
}
```

Creating Terms

```
static void CreateTerms(ClientContext clientContext, TermSet termSet) {

  Term termEurope = termSet.CreateTerm("Europe", 1033, Guid.NewGuid());
  termEurope.CreateTerm("United Kingdon", 1033, Guid.NewGuid());
  termEurope.CreateTerm("France", 1033, Guid.NewGuid());
  termEurope.CreateTerm("Spain", 1033, Guid.NewGuid());
  termEurope.CreateTerm("Germany", 1033, Guid.NewGuid());

  Term termNorthAmerica = termSet.CreateTerm("North America", 1033, Guid.NewGuid());
  termNorthAmerica.CreateTerm("Canada", 1033, Guid.NewGuid());
  termNorthAmerica.CreateTerm("United States", 1033, Guid.NewGuid());
  termNorthAmerica.CreateTerm("Mexico", 1033, Guid.NewGuid());

  clientContext.ExecuteQuery();
  termSet.TermStore.CommitAll();

}
```

Summary

- ✓ Managed Metadata
- ✓ Enterprise Content Types
- ✓ Document Sets
- ✓ Records Management
- ✓ eDiscovery
- ✓ Programming with Managed Metadata