­

**K2NE GmbH  
c/o Launchswiss  
Rathausstr. 14  
6340 Baar, Switzerland**

**Fax: +41-41-560 8513  
Email: k2ne@k2.com  
Web: http://www.k2.com**

K2NE SP Content Broker

Ruben

05 April 2018

HISTORY

|  |  |  |  |
| --- | --- | --- | --- |
| Author | Date | Version | Reason for edit |
| Ruben d’Arco | 2017-12-12 | 0.1 | Initial version |
| Ruben d’Arco | 2018-02-04 | 0.2 | Review after Konstantin worked on this document. |
| Ruben d’Arco | 2018-02-04 | 0.3 | Small update to indicate some service object naming. |
| Ruben d’Arco | 2018-02-07 | 0.4 | Updates after Konstantin. |
| Taher B | 2018-04-05 | 0.5 | Added CheckIn and Checkout Functionality |

Content

[1 Introduction 3](#_Toc510682515)

[2 Methods 3](#_Toc510682516)

[2.1 Special properties 3](#_Toc510682517)

[2.2 List and library functions 3](#_Toc510682518)

[2.2.1 Get Item By Id 3](#_Toc510682519)

[2.2.2 Create Item 4](#_Toc510682520)

[2.2.3 Update item By Id 4](#_Toc510682521)

[2.2.4 Delete Item By Id 4](#_Toc510682522)

[2.2.5 Get Item By Title 4](#_Toc510682523)

[2.2.6 Get Items 5](#_Toc510682524)

[2.3 Document library functions 5](#_Toc510682525)

[2.3.1 Create Document 5](#_Toc510682526)

[2.3.2 Delete document By Id 5](#_Toc510682527)

[2.3.3 Get Document By Id 6](#_Toc510682528)

[2.3.4 Get Documents 6](#_Toc510682529)

[2.3.5 Copy Document By Name 6](#_Toc510682530)

[2.3.6 Move Document By Name 7](#_Toc510682531)

[2.3.7 Rename Document By Id 7](#_Toc510682532)

[2.3.8 Check In document by name 7](#_Toc510682533)

[2.3.9 Check In document by Id 8](#_Toc510682534)

[2.3.10 Check Out document by name 8](#_Toc510682535)

[2.3.11 Check Out document by Id 8](#_Toc510682536)

[2.4 Folder functionality 8](#_Toc510682537)

[2.4.1 Create Folder 8](#_Toc510682538)

[2.4.2 Delete Folder 9](#_Toc510682539)

[2.4.3 Rename Folder 9](#_Toc510682540)

[2.4.4 Move Folder 9](#_Toc510682541)

[2.5 Document set functionality 10](#_Toc510682542)

[2.5.1 Create Document Set 10](#_Toc510682543)

[2.5.2 Delete Document Set 10](#_Toc510682544)

[2.5.3 Update Document SET 10](#_Toc510682545)

[2.5.4 Get Document Set 10](#_Toc510682546)

[2.5.5 Get Document Sets 11](#_Toc510682547)

[2.5.6 Rename document set 11](#_Toc510682548)

[2.6 Permission functionality 11](#_Toc510682549)

[2.6.1 Item functions 11](#_Toc510682550)

[2.6.2 Folder functions 12](#_Toc510682551)

# Introduction

The document aims to describe the functionality that the K2NE SharePoint Content broker has. The document is a description of the interface that this service broker has. With interface, we refer to the Service Objects, Properties and Methods that a service instance will provide. The service instance’s configuration options will also be described.

# Methods

Listed in this chapter are all the methods that the service broker aims to have. Every method provides details of the Input, required and output properties/parameters as well as a short description.

## Special properties

The broker is dynamic, this means that a list/library is interrogated and all fields are provided as output properties for a given method. If a method has these output properties, then this is indicated by ‘\*\* dynamic \*\*’. In case of a document library, the file content is NOT included and will be indicted specifically. The name of the document and link to the document are provided. Dynamic properties are named quite specifically. They will use the internal name within SP for it’s system name, they will use the display name for its display name. Because a list can have multiple properties with the same display name, it means that clashes will occur, in which case the displayname is prefixed by “(internal name)”. This behaviour can also be forced by the service instance configuration property.

The SiteURL property (only provided at input) is a property that is created depending on the service instance configuration. The property depends on the ‘Dynamic’ service instance property.

A \* next to the input property indicates that the property is required.

Lookup and User columns will not only have a 1 to 1 mapping, there will be additional properties that provide only the value (number/id) and that provide the display value. This is useful as a lookup property in SharePoint is stored as (for example) “3;London”. This value itself will be provided, the ‘3’ value will be provided and the “London” value will be provided in separate properties.

To make method names unique, we will prefix them with the ServiceObject name. This is not reflected in the document and will only impact the system name of the method.

## List and library functions

### Get Item By Id

|  |  |
| --- | --- |
| Method Name: | GetItemById |
| Display name: | Get Item By Id |
| Method type: | Read |
| Description: | Retrieves a single item from a list or library, based on the unique ID of the item. No Document content is ever returned by this method. |
| Input properties: | Id \* |
|  | SiteURL \* |
| Output properties: | \*\* Dynamic \*\* |
|  | FileName - only in case of a document library |
|  | LinkToItem – unique URL to the specific item |

### Create Item

|  |  |
| --- | --- |
| Method Name: | CreateItem |
| Display name: | Create Item |
| Method type: | Create |
| Description: | Creates a single item in a list. |
| Input properties: | \*\* Dynamic \*\* |
|  | SiteURL \* |
|  | Folder – only in case if folders creation is enabled. |
| Output properties: | Id |
|  | LinkToItem – unique URL to the specific item. |

### Update item By Id

|  |  |
| --- | --- |
| Method Name: | UpdateItemById |
| Display name: | Update Item By Id |
| Method type: | Update |
| Description: | Updates the (provided) metadata of a list item or document based on its unique ID. No file content can be changed. |
| Input properties: | \*\* Dynamic \*\* |
|  | SiteURL \* |
|  | Id \* |
| Output properties: | Id |
|  | LinkToItem – unique URL to the specific item. |

### Delete Item By Id

|  |  |
| --- | --- |
| Method Name: | DeleteItemById |
| Display name: | Delete Item By Id |
| Method type: | Delete |
| Description: | Delete a single item from a list or library, based on the unique ID of the item. |
| Input properties: | SiteURL \* |
|  | Id \* |
| Output properties: | None. |

### Get Item By Title

|  |  |
| --- | --- |
| Method Name: | GetItemByTitle |
| Display name: | Get Item By Title/Get Item By Name |
| Method type: | Read |
| Description: | Get item from a list or library, based on the Title |
| Input properties: | SiteURL \* |
|  | Title \* |
|  | Folder – only in case if folders creation enabled |
| Output properties: | \*\* Dynamic \*\* |
|  | LinkToItem – unique URL to the specific item |

### Get Items

|  |  |
| --- | --- |
| Method Name: | GetItems |
| Display name: | Get Items |
| Method type: | List |
| Description: | Get items from a list or library, based on list or library fields. The method can be executed recursively if it has folder creation mode enabled. |
| Input properties: | \*\* Dynamic \*\* |
|  | SiteURL \* |
|  | Folder – only in case if folders creation enabled. |
|  | Recursive – only in case if folders creation enabled. Default set to true. Retrieves the items recursively. |
| Output properties: | \*\* Dynamic \*\* |
|  | FileName - only in case of a document library |
|  | LinkToItem – unique URL to the specific item |
|  | Folder – in which folder the item exists. |

## Document library functions

It might seem that some functionality is missing here, but most of the functions are included in the list item section previously (for example, see update item).

### Create Document

|  |  |
| --- | --- |
| Method Name: | CreateDocument |
| Display name: | Create Document |
| Method type: | Create |
| Description: | Upload a single document with (provided) metadata in a library. |
| Input properties: | \*\* Dynamic \*\* |
|  | SiteURL \* |
|  | File \* |
|  | OverwriteExistingDocument (Yes/No) – Indicates if a document should be overwriten if it already exists, or if it does not exist – create it. |
|  | Folder – only in case if folders creation enabled |
| Output properties: | Id |
|  | FileName |
|  | LinkToItem – unique URL to the specific item. |

### Delete document By Id

|  |  |
| --- | --- |
| Method Name: | DeleteDocumentById |
| Display name: | Delete Document By Id |
| Method type: | Delete |
| Description: | Removes a document from the library by id. |
| Input properties: | Id |
|  | SiteURL \* |
| Output properties: | None. |



### Get Document By Id

|  |  |
| --- | --- |
| Method Name: | GetDocumentById |
| Display name: | Get Document By Id |
| Method type: | Read |
| Description: | Retrieves a single document with metadata from a library, based on the document’s Id |
| Input properties: | SiteURL \* |
|  | Id \* |
|  | Folder – only in case if folders creation enabled |
| Output properties: | \*\* Dynamic \*\* |
|  | File |
|  | FileName |
|  | LinkToItem – unique URL to the specific item |

### Get Documents

|  |  |
| --- | --- |
| Method Name: | GetDocuments |
| Display name: | Get Documents (warning) |
| Method type: | List |
| Description: | Get documents from a library, based on library fields. |
| Input properties: | \*\* Dynamic \*\* |
|  | SiteURL \* |
|  | Folder – only in case if folders creation enabled |
|  | Recursive – only in case if folders creation enabled. Defaults to true, retrieves all documents recursively. |
| Output properties: | \*\* Dynamic \*\* |
|  | File – the content of the files. |
|  | FileName |
|  | LinkToItem – unique URL to the specific item |

### Copy Document By Name

|  |  |
| --- | --- |
| Method Name: | CopyDocumentByName |
| Display name: | Copy Document By Name |
| Method type: | Execute |
| Description: | Copy a single document from one library to another based on it’s Name. |
| Input properties: | SiteURL \* |
|  | Destination SiteURL \* |
|  | Destination Library \* |
|  | Destination Folder |
|  | OverwriteExistingDocument (Yes/No) |
|  | Folder - only in case if folders creation enabled. This is the folder where the existing document is stored. |
|  | FileName \* |
| Output properties: | FileName |
|  | LinkToItem - unique URL to the newly created item. |

### Move Document By Name

|  |  |
| --- | --- |
| Method Name: | MoveDocumentByName |
| Display name: | Move Document By Name |
| Method type: | Execute |
| Description: | Move a single document from one library to another based on it’s Name. |
| Input properties: | SiteURL \* |
|  | Destination SiteURL \* |
|  | Destination Library \* |
|  | Destination Folder |
|  | OverwriteExistingDocument (Yes/No) |
|  | Folder - only in case if folders creation enabled. This is the folder where the existing document is stored. |
|  | FileName \* |
| Output properties: | FileName |
|  | LinkToItem - unique URL to the specific item. The new location. |

### Rename Document By Id

|  |  |
| --- | --- |
| Method Name: | RenameDocumentById |
| Display name: | Rename Document By Id |
| Method type: | Execute |
| Description: | Rename a single document in a library based on it’s Name. |
| Input properties: | SiteURL \* |
|  | NewFileName \* |
|  | Id \* |
| Output properties: | LinkToItem - unique URL to the specific item. |
|  | Id |

### Check In document by name

|  |  |
| --- | --- |
| Method Name: | CheckInDocumentByName |
| Display name: | Check In Document By Name |
| Method type: | Execute |
| Description: | Check In a single document in a library based on its Name. |
| Input properties: | SiteURL \* |
|  | FileName \* |
|  | FolderName |
|  | CheckInComments |
|  | RetainCheckout |
|  | UseCheckedInVersion |
| Output properties: | LinkToItem - unique URL to the specific item. |
|  | Id |

### Check In document by Id

|  |  |
| --- | --- |
| Method Name: | CheckInDocumentById |
| Display name: | Check In Document By Id |
| Method type: | Execute |
| Description: | Check In a single document in a library based on its Id. |
| Input properties: | SiteURL \* |
|  | Id \* |
|  | FolderName |
|  | CheckInComments |
|  | RetainCheckout |
|  | UseCheckedInVersion |
| Output properties: | LinkToItem - unique URL to the specific item. |
|  | Id |

### Check Out document by name

|  |  |
| --- | --- |
| Method Name: | CheckOutDocumentByName |
| Display name: | Check Out Document By Name |
| Method type: | Execute |
| Description: | Check Out a single document in a library based on its Name. |
| Input properties: | SiteURL \* |
|  | FileName \* |
|  | FolderName |
|  | UseCheckedOutVersion |
|  |  |
| Output properties: | LinkToItem - unique URL to the specific item. |
|  | Id |

### Check Out document by Id

|  |  |
| --- | --- |
| Method Name: | CheckOutDocumentById |
| Display name: | Check Out Document By Id |
| Method type: | Execute |
| Description: | Check Out a single document in a library based on its Id. |
| Input properties: | SiteURL \* |
|  | FileName \* |
|  | FolderName |
|  | UseCheckedOutVersion |
|  |  |
| Output properties: | LinkToItem - unique URL to the specific item. |
|  | Id |

## Folder functionality

Whenever a list or library has the option to create folders, these methods will be added to it’s Service Object.

### Create Folder

|  |  |
| --- | --- |
| Method Name: | CreateFolder |
| Display name: | Create Folder |
| Method type: | Execute |
| Description: | Create a folder in a list or library based on a folder name. |
| Input properties: | SiteURL \* |
|  | Folder \* - Full path of the folder to be created. Folders will be created recursively. Path is always the fill path within the document library (should start with “/”) |
| Output properties: | LinkToItem - unique URL to the specific item. |

### Delete Folder

|  |  |
| --- | --- |
| Method Name: | DeleteFolder |
| Display name: | Delete Folder |
| Method type: | Execute |
| Description: | Delete a folder in a list or library based on a folder name. |
| Input properties: | SiteURL \* |
|  | Folder \* |
|  | Recursive \* - Indicate if the folders should be deleted recursively. Meaning that if I have “/Folder1/2/file.docx” in my library and I pass “/Folder1/2/” to the Folder property and set this (recursive) to true, that the file.docx will be deleted. If I passed “/Folder1”. |
| Output properties: | None. |

### Rename Folder

|  |  |
| --- | --- |
| Method Name: | RenameFolder |
| Display name: | Rename Folder |
| Method type: | Execute |
| Description: | Rename a folder in a list or library based on a folder name. |
| Input properties: | SiteURL \* |
|  | Folder \* - full path to the folder, example “/folder1/folder2/folder3/” |
|  | Destination Folder \* - may not contain slashes. Example “Folder4” to make the above into “/folder1/folder2/Folder4/” |
| Output properties: | LinkToItem - unique URL to the specific item. |

### Move Folder

|  |  |
| --- | --- |
| Method Name: | MoveFolder |
| Display name: | Move Folder |
| Method type: | Execute |
| Description: | Move a folder from one list or library to another, based on a folder name. Method throws an exception if the destination folder already exists. |
| Input properties: | SiteURL \* |
|  | Folder \* |
|  | Destination SiteURL – can be left empty to indicate a move within the same list/library. |
|  | Destination List/Library – can be left empty to indicate a move within the same list/library. |
|  | Destination Folder |
| Output properties: | LinkToItem - unique URL to the specific item. |

## Document set functionality

Document set functionality should be very similar to folders, as technically the underlying constructs are the same (a document set is technically a folder).

The methods are only added if a list of library has document sets enabled.

### Create Document Set

|  |  |
| --- | --- |
| Method Name: | CreateDocumentSet |
| Display name: | Create Document Set |
| Method type: | Create |
| Description: | Create document set |
| Input properties: | SiteURL \* |
|  | \*\* Dynamic \*\* |
| Output properties: | Id |

### Delete Document Set

|  |  |
| --- | --- |
| Method Name: | DeleteDocumentSetById |
| Display name: | Delete Document Set By Id |
| Method type: | Delete |
| Description: | Delete document set |
| Input properties: | SiteURL \* |
|  | Id \* |
| Output properties: | None. |

### Update Document SET

|  |  |
| --- | --- |
| Method Name: | UpdateDocumentSetById |
| Display name: | Update Document Set By Id |
| Method type: | Update |
| Description: | Update document set metadata by Id |
| Input properties: | SiteURL \* |
|  | \*\* Dynamic \*\* |
|  | Id \* |
| Output properties: | Id |

### Get Document Set

|  |  |
| --- | --- |
| Method Name: | GetDocumentSetById |
| Display name: | Get Document Set By Id |
| Method type: | Read |
| Description: | Get document set metadata by id |
| Input properties: | SiteURL \* |
|  | Id \* |
| Output properties: | \*\* Dynamic \*\* |

### Get Document Sets

|  |  |
| --- | --- |
| Method Name: | GetDocumentSets |
| Display name: | Get Document Sets |
| Method type: | List |
| Description: | Retrieve all document sets provided. |
| Input properties: | SiteURL \* |
|  | Id \* |
|  | Folder – You can specify a folder to only get the document sets from that folder. |
|  | Recursive – Defaults to true, retrieve the list of document sets in a recursive manner. |
| Output properties: | \*\* Dynamic \*\* |

### Rename document set

|  |  |
| --- | --- |
| Method Name: | RenameDocumentSetById |
| Display name: | Rename Document Set By Id |
| Method type: | Read |
| Description: | Rename the document set. |
| Input properties: | SiteURL \* |
|  | Id \* |
|  | DocumentSetName – new name of the document set. |
| Output properties: | LinkToItem |

## Permission functionality

### Item functions

#### Break Item Inheritance

|  |  |
| --- | --- |
| Method Name: | BreakItemInheritanceById |
| Display name: | Break Item Inheritance By Id |
| Method type: | Execute |
| Description: | Stop inheritance permissions on a list or library item by Id |
| Input properties: | SiteURL \* |
|  | Id \* |
| Output properties: | None. |

#### Reset Item Inheritance

|  |  |
| --- | --- |
| Method Name: | ResetItemInheritanceById |
| Display name: | Reset Item Inheritance By Id |
| Method type: | Execute |
| Description: | Reset inheritance permissions on a list or library item by Id |
| Input properties: | SiteURL \* |
|  | Id \* |
| Output properties: | None. |

#### Add Item Permission

|  |  |
| --- | --- |
| Method Name: | AddItemPermissionById |
| Display name: | Add Item Permission By Id |
| Method type: | Execute |
| Description: | Add user or group permissions to a list or library item by Id. Both User and groups can be provided. One of them is mandatory. |
| Input properties: | SiteURL \* |
|  | Id \* |
|  | Users - user logins separated by semicolon |
|  | Groups – group names separated by semicolon |
|  | Permission\* - Role Definition Name. |
| Output properties: | None. |

#### Remove Item Permission

|  |  |
| --- | --- |
| Method Name: | RemoveItemPermissionById |
| Display name: | Remove Item Permission By Id |
| Method type: | Execute |
| Description: | Remove user or group permissions to a list or library item by Id. Both User and groups can be provided. One of them is mandatory. Function always executes, so if the group or user did not exist, it will not cause an error but just continue. |
| Input properties: | SiteURL \* |
|  | Id \* |
|  | Users - user logins separated by semicolon |
|  | Groups – group names separated by semicolon |
| Output properties: | None. |

#### Get Item Permissions

|  |  |
| --- | --- |
| Method Name: | GetItemPermissionById |
| Display name: | Get Item Permission By Id |
| Method type: | Execute |
| Description: | Get user or group permissions to a list or library item by user or group id |
| Input properties: | SiteURL \* |
|  | Id \* |
|  | UserOrGroup \* – user’s or group’s login |
| Output properties: | Permissions – permissions separated by semicolon |

### Folder functions

Whenever a list or libraries has folders, the following set of methods will be added.

#### BreaK Folder Inheritance

|  |  |
| --- | --- |
| Method Name: | BreakFolderInheritance |
| Display name: | Break Folder Inheritance |
| Method type: | Execute |
| Description: | Stop inheritance permissions on a list or library folder by folder name |
| Input properties: | SiteURL \* |
|  | Folder \* |
| Output properties: | None. |

#### Reset Folder Inheritance

|  |  |
| --- | --- |
| Method Name: | ResetFolderInheritance |
| Display name: | Reset Folder Inheritance |
| Method type: | Execute |
| Description: | Reset inheritance permissions on a list or library folder by folder name |
| Input properties: | SiteURL \* |
|  | Folder \* |
| Output properties: | None. |

#### Add Folder PermissioN By Name

|  |  |
| --- | --- |
| Method Name: | AddFolderPermissionByName |
| Display name: | Add Folder Permission By Name |
| Method type: | Execute |
| Description: | Add user or group permissions to a list or library folder by name |
| Input properties: | SiteURL \* |
|  | Folder \* |
|  | SpecifyUsers - user logins separated by semicolon |
|  | SpecifyGroups – group names separated by semicolon |
|  | SpecifyPermissions\* - role definition names separated by semicolon |
| Output properties: | None. |

#### Remove Folder Permission By Name

|  |  |
| --- | --- |
| Method Name: | RemoveFolderPermissionByName |
| Display name: | Remove Folder PermissionByName |
| Method type: | Execute |
| Description: | Remove user or group permissions to a list or library folder by name |
| Input properties: | SiteURL \* |
|  | Folder \* |
|  | SpecifyUsers - user logins separated by semicolon |
|  | SpecifyGroups – group names separated by semicolon |
| Output properties: | None. |

#### Get Folder Permissions

|  |  |
| --- | --- |
| Method Name: | GetFolderPermission |
| Display name: | Get Folder Permission |
| Method type: | Execute |
| Description: | Get user or group permissions to a list or library folder by user or group id |
| Input properties: | SiteURL \* |
|  | List/Library \* |
|  | Folder \* |
|  | UserOrGroup – user’s or group’s login |
| Output properties: | Permissions – permissions separated by semicolon |

RUBEN NOTES:

* SharePoint has SharePoint groups. You can add AD Users and AD groups to a sharepoint group.
* A sharepoint group has certain permissions levels. Custom permission levels can be created too.
* In the above methods, I don’t know what we’re removing/adding.