

**Chemistry Add-in for**

**Microsoft Word**

**Technical Manual**

**Version 3.0**

Contents

[Introduction 3](#_Toc495521122)

[Storage Model 3](#_Toc495521123)

[Appendix A – Sample code 4](#_Toc495521124)

# Introduction

This document is intended to help users integrate documents produced by the Chemistry Add-in for Microsoft Word into other systems such as SharePoint.

# Storage Model

The machine readable Chemistry is stored as hidden Word Objects called CustomXmlParts, as their name suggests they can be used to store XML (or in our case CML, which is a dialect of XML). When a hidden chemistry object is first created, it is given a Globally Unique Identifier (Guid). This Guid is then stored in the Tag of each visible Content Control which contains Chemistry, this allows us to find the Chemistry data, when an operation on a chemistry zone is carried out.



Sample code to extract the CML is given in Appendix A.

One example of an element of the extracted CML, which could be used to index all chemistry is shown below.

<cml:name id="m1.n3" dictRef="chemspider:Inchikey">SMWDFEZZVXVKRB-UHFFFAOYSA-N</cml:name>

The Inch key[[1]](#footnote-2) is a unique “fingerprint” of the Chemistry, which could be used to index the data.

# Appendix A – Sample code

The following code could very easily be incorporated into a SharePoint ListItemReceiver to extract and index the chemistry when a document is created or updated in a SharePoint library.

File “Program.cs” the main entry point.

using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.IO;

namespace ReadCustomXmlParts

{

    internal class Program

    {

        private static void Main(string[] args)

        {

            string documents = Environment.GetFolderPath(Environment.SpecialFolder.MyDocuments);

            string file = Path.Combine(documents, "Two Chemistry Zones.docx");

            ChemistryReader reader = new ChemistryReader();

            List<string> zones = reader.GetChemistryZones(file);

            Debug.WriteLine($"Found {zones.Count} chemistry zones");

        }

    }

}

File “ChemistryReader.cs” this collects the Chemistry Zones as CML.

using DocumentFormat.OpenXml.Packaging;

using System.Collections.Generic;

using System.IO;

using System.Xml;

namespace ReadCustomXmlParts

{

    public class ChemistryReader

    {

        public List<string> GetChemistryZones(string filename)

        {

            List<string> zones = new List<string>();

            using (WordprocessingDocument wordDoc =

                WordprocessingDocument.Open(filename, false))

            {

                var mainPart = wordDoc.MainDocumentPart;

                foreach (var cxml in mainPart.CustomXmlParts)

                {

                    using (XmlTextReader reader =

                        new XmlTextReader(cxml.GetStream(FileMode.Open, FileAccess.Read)))

                    {

                        reader.MoveToContent();

                        string str = reader.ReadOuterXml();

                        if (str.Contains("cml:cml"))

                        {

                            zones.Add(str);

                        }

                    }

                }

            }

            return zones;

        }

    }

}

1. https://en.wikipedia.org/wiki/International\_Chemical\_Identifier#InChIKey [↑](#footnote-ref-2)