

[MS-DSDIFFGRAM]: SharePoint Web Services: DataSet DiffGram Structure Specification

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.mspx>). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplq@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
04/04/2008	0.1	Major	Initial Availability.
06/27/2008	1.0	Editorial	Revised and edited the technical content.
12/12/2008	1.01	Editorial	Revised and edited the technical content.
03/11/2009	1.02	Editorial	Revised and edited the technical content.
04/06/2009	1.03	Editorial	Revised and edited the technical content.
08/07/2009	1.1	Minor	Updated the technical content.
11/06/2009	1.1.2	Editorial	Revised and edited the technical content.
03/05/2010	1.2	Minor	Updated the technical content.

Contents

1 Introduction	4
1.1 Glossary.....	4
1.2 References.....	4
1.2.1 Normative References	4
1.2.2 Informative References	5
1.3 Structure Overview (Synopsis)	5
1.4 Relationship to Protocols and Other Structures	6
1.5 Applicability Statement.....	6
1.6 Versioning and Localization	6
1.7 Vendor-Extensible Fields	6
2 Structures	7
2.1 DiffGram Namespaces	7
2.2 SharePoint DiffGram High-Level Structure	7
2.3 SharePoint DiffGram Schema Element.....	8
2.3.1 DataInstance Element Schema	8
2.3.2 DataTable Element Schema.....	9
2.3.3 Unique Element.....	11
2.4 DiffGram Data Element	11
2.4.1 DataInstance Element.....	12
3 Structure Examples	14
4 Security Considerations	17
5 Appendix A: Product Behavior	18
6 Change Tracking	19
7 Index	21

1 Introduction

This document specifies the Microsoft SharePoint **DataSet DiffGram** structure used to represent the results of a SharePoint Search service Web service call. The structure used by the SharePoint Search service is a subset of the full **DiffGram** structure used by the ADO.NET **DataSet**. The **DiffGram** structure is useful for serializing schema and data for transmission over a network or storage on disk. SharePoint uses the **DiffGram** structure to encapsulate both the schema of the search results, as well as the data that represents the search results.

This document covers only the portion of the **DiffGram** structure that is used by the SharePoint Search service.

1.1 Glossary

The following terms are defined in [\[MS-GLOS\]](#):

Hypertext Transfer Protocol (HTTP)
User Datagram Protocol (UDP)
XML

The following terms are defined in [\[MS-OFCGLOS\]](#):

CDATA section
child element
root element
SOAP (Simple Object Access Protocol)
SOAP envelope
Web service
Web service method
XML attribute
XML document
XML element
XML namespace
XML namespace prefix
XML Schema (1)

The following terms are specific to this document:

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as described in [\[RFC2119\]](#). All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

[MC-ADONETDSSS] Microsoft Corporation, "ADO.NET DataSet Structure Schema", <http://schemas.microsoft.com/2003/07msdata.xsd>

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)", March 2007.

[MS-OFCGLOS] Microsoft Corporation, "[Microsoft Office Master Glossary](#)".

[MS-OFSGLOS] See [MS-OFCGLOS].

[MS-SEARCH] Microsoft Corporation, "[Search Protocol Specification](#)", June 2008.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.ietf.org/rfc/rfc2119.txt>

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., Layman, A., Mendelsohn, N., Nielsen, H. F., Thatte, S., and Winer, D., "Simple Object Access Protocol (SOAP) 1.1", May 2000, <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

[XML10] World Wide Web Consortium, "Extensible Markup Language (XML) 1.0 (Third Edition)", February 2004, <http://www.w3.org/TR/REC-xml>

[XMLNS3] World Wide Web Consortium, "Namespaces in XML 1.0 (Third Edition)", December 2009, <http://www.w3.org/TR/2009/REC-xml-names-20091208/>

[XMLSCHEMA1] Thompson, H.S., Ed., Beech, D., Ed., Maloney, M., Ed., and Mendelsohn, N., Ed., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>

[XMLSCHEMA2] Biron, P.V., Ed. and Malhotra, A., Ed., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/>

[XPath] Clark, J. and DeRose, S., "XML Path Language (XPath), Version 1.0", W3C Recommendation, November 1999, <http://www.w3.org/TR/xpath>

1.2.2 Informative References

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, <http://www.ietf.org/rfc/rfc2616.txt>

1.3 Structure Overview (Synopsis)

The **DataSet** is the part of the .NET Framework that provides in-memory representation of relational data. A **DataSet** consists of a set of named tables. Each table is defined by a collection of named columns with specified data types. A set of columns in the table can also represent a primary key. When a **DataSet** is populated, its tables are filled with rows of data, each of which contains a value for each column. The **DataSet** is a completely in-memory representation of the relational data, and maintains no knowledge of the original source of the data.

In addition to storing data in the rows and columns of the **DataSet**, applications can attach additional data to the entire **DataSet**, or to particular tables or columns, via extended properties. Extended properties are name-value pairs that are exposed to consumers of the **DataSet**, but not interpreted by the **DataSet** in any way. For details on how extended properties are used in the context of a SharePoint search, please see [\[MS-SEARCH\]](#).

In various scenarios, it is necessary to transfer a **DataSet** across application boundaries. This is usually accomplished by serializing the **DataSet** into a format suitable for transmission over the serialization substrate. Common patterns include returning a **DataSet** from a **Web service method** and taking **DataSets** as input parameters to Web service methods.

The **DiffGram** structure is an **XML** serialized form of a **DataSet** that can be used in these scenarios. Any **DataSet** instance can be serialized into a **DiffGram** that can be transmitted over a service interface or written to persistent storage. The **DiffGram** structure encapsulates all of the information required to re-create the in-memory **DataSet** in the exact state it was in at the time it was serialized. This includes the schema information that defines the structure of the data in the **DataSet** and the actual values of the data. The **DiffGram** also contains serialized representations of any extended properties that have been defined on the tables and/or columns.

1.4 Relationship to Protocols and Other Structures

The **DiffGram** structure is used by the ADO.NET Framework as a serialization format for the contents of **DataSets**. Whenever a **DataSet** is returned from or received by a Web service method, the **DiffGram** structure is used as the default serialization format. When used this way, the **DiffGram** can be wrapped in other data structures (for example, as specified in [\[SOAP1.1\]](#), Section 4) that encapsulate other parts of the **Web service** call.

The services that exchange **DataSets** can use a variety of network protocols and encodings to transfer **DiffGrams**. For example, one Web service can choose to use a plain-text encoding of a **DiffGram** within a **SOAP envelope**, transmitted using **Hypertext Transfer Protocol (HTTP)** as specified in [\[RFC2616\]](#). Another can choose a binary encoding for the SOAP envelope containing the **DiffGram** and transmit it via **User Datagram Protocol (UDP)**. The network protocols and encodings that can be used to transmit **DiffGrams** are not covered in this document.

1.5 Applicability Statement

The **DiffGram** structure can be used whenever a serialized representation of a **DataSet** is needed. More generally, the **DiffGram** can be used whenever it is necessary to serialize relational data. This document specifies the serialization of the relational data as used by the SharePoint Search service, but does not cover the general case of **DataSet** serialization.

1.6 Versioning and Localization

None.

1.7 Vendor-Extensible Fields

None.

2 Structures

The SharePoint **DiffGram** is an **XML document** that encapsulates the following information:

- An **XML Schema (1)** that specifies the structure of the data in a **DataSet**
- The data in the **DataSet**
- Extended properties associated with tables and columns

The following sections provide details on the particular representation used to capture this information.

2.1 DiffGram Namespaces

The XML that comprises a SharePoint **DiffGram** MUST include required **XML elements** and **XML attributes** as specified in the following sections of this document. These XML elements and XML attributes are defined in various **XML namespaces**. The following table lists these XML namespaces and specifies the **XML namespace prefixes** commonly associated with them. Producers of SharePoint **DiffGrams** MUST ensure that the XML refers to these namespaces by using the mechanisms that are specified in [\[XMLNS3\]](#), but they SHOULD use the prefixes shown in the table below. For clarity, when XML elements and attributes from these namespaces are referenced in following sections of this document, their fully-qualified names are used.

Description	Namespace URI	Commonly used Prefix	Reference
XML Schema elements and attributes	http://www.w3.org/2001/XMLSchema	xs	[XMLSCHEMA1] [XMLSCHEMA2]
DiffGram elements and attributes	urn:schemas-microsoft-com:xml-diffgram-v1	diffgr	This namespace is internal to the DiffGram structure and is specified in DiffGram Data Element .
DataSet specific annotations	urn:schemas-microsoft-com:xml-msdata	msdata	[MC-ADONETDSSS]
DataSet extended properties	urn:schemas-microsoft-com:xml-msprop	msprop	User and application-specific information SHOULD be annotated on the DataSet schema with extended properties. The extended properties are defined in this namespace.

2.2 SharePoint DiffGram High-Level Structure

A valid SharePoint **DiffGram** MUST conform to the following rules:

- The SharePoint **DiffGram** MUST have a **root element**, hereafter referred to as the **Root** element.
- The **Root** element MUST have two **child elements**:

- The first child element, hereafter referred to as the [SharePoint DiffGram Schema element](#), MUST be a **Schema** element as defined by [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#) and MUST contain a valid XML Schema.
- The second child element, hereafter referred to as the [SharePoint DiffGram Data element](#), MUST be a **DiffGram** element as defined in the namespace `urn:schemas-microsoft-com:xml-diffgram-v1`.

The sections that follow define the **DiffGram Schema** element and the **DiffGram Data** element in more detail. At a basic level, the purpose of these elements can be explained as follows:

- The SharePoint **DiffGram Schema** element defines the XML Schema for the data representation in the SharePoint **DiffGram Data** element's content. The XML representation of the data in the **DiffGram Data** element's content MUST conform to the XML Schema defined in the **DiffGram Schema** element.
- The SharePoint **DiffGram Data** element encapsulates the values of the data in the **DataSet**.

2.3 SharePoint DiffGram Schema Element

The **Schema** element in a SharePoint **DiffGram** MUST contain an XML Schema, as specified in [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#), that defines an XML representation for the data in the **DataSet**. The **Schema** element is a representation of the shape of the **DataSet** that will be used for serialization purposes, with the tables, columns, and primary keys of the **DataSet** represented as anonymous complex types and unique elements, subject to various constraints specified in the following sections.

2.3.1 DataInstance Element Schema

The **DataInstance** schema MUST define exactly one element that will encapsulate the representation of all data in the **DataSet**. This element is referred to as the **DiffGram DataInstance** element. In addition to being a valid XML Schema, this schema MUST conform to the following rules:

- The **DataInstance** element MUST be defined using an anonymous complex type. The complex type MUST be defined as a choice of element content with zero minimum occurrence and unbounded maximum occurrence. Each element in the choice MUST conform to the rules of the element definition specified in section [2.3.2](#).
 - The **DataInstance** element definition MUST contain zero or more **DataTable** element definitions in the choice element. The **DataTable** element definitions are immediate children of the choice.
- The `http://www.w3.org/2001/XMLSchema:>` element that defines the **DataInstance** element MUST have the `urn:schemas-microsoft-com:xml-msdata:IsDataSet` attribute set to true.
- The **DataInstance** element MUST contain zero, one, or more [Unique](#) elements. The **Unique** element's definitions are defined immediately after the complex type element that defines a choice of **DataTable** elements.
- The **DataInstance** element complex type definition MUST NOT contain `http://www.w3.org/2001/XMLSchema:>` attribute definitions.
- The `http://www.w3.org/2001/XMLSchema:>` element that defines the **DataInstance** element MAY contain the `urn:schemas-microsoft-com:xml-msdata:UseCurrentLocale` attribute. If this attribute is present, the value of the attribute MUST be set to true.

- The **http://www.w3.org/2001/XMLSchema:>** element that defines the **DataInstance** element MAY have the **msdata:DataSetName** attribute. If this attribute is specified, then the name of the **DataSet** is equal to the value of the attribute. If this attribute is not specified, then the name of the **DataSet** is equal to the name of the **DataInstance** element.
- Each extended property MUST appear as an attribute in the declaration of the **DataInstance** element (for properties that apply to the whole **DataSet**). The attribute name MUST be the name of the extended property, and the attribute's namespace MUST be `urn:schemas-microsoft-com:xml-msprop`. The attribute's value MUST be the value of the extended property.
- Any attributes with a namespace other than `urn:schemas-microsoft-com:xml-msprop` are not extended properties.

2.3.2 DataTable Element Schema

The **DataInstance** schema MUST define zero or more **DataTable** elements as children of the **DataInstance** element. A **DataTable** element MUST be defined using an anonymous complex type, and conform to the following rules:

- A **DataTable** element complex type definition MUST NOT contain attribute definitions.
- A **DataTable** element definition MUST allow only element content.
- The content model of a **DataTable** element definition MUST be specified as a sequence of elements, referred to as **DataColumn** elements, each of which represents a column in the table.
 - A **DataTable** element definition MUST define zero or more **DataColumn** element definitions that are the immediate children of the sequence.
 - The type of a **DataColumn** element definition MUST be specified as one of the following valid `http://www.w3.org/2001/XMLSchema:types`. If the type is a string, then the type MAY be a simple type restriction of the facets as defined by [XMLSCHEMA2](#) section 3.2.1.1, length, minlength, and maxlength. All other types MUST NOT be restricted.
 - string
 - Boolean
 - unsignedByte
 - base64Binary
 - double
 - float
 - short
 - int
 - long
 - byte
 - integer
 - decimal

- date
 - time
 - unsignedShort
 - unsignedInt
 - unsignedLong
 - dateTime
- The minimum occurrence of a **DataColumn** element definition MUST be zero or one and the maximum occurrence MUST be one.
 - If the type of the **DataColumn** element definition is a string, then all data in the DataColumn instance MUST be interpreted as **CDATA section**, even if it contains data that could be interpreted as XML.
 - Each extended property MUST appear as an attribute in the declaration of the **DataColumn** element (for properties that apply to columns). The attribute name MUST be the name of the extended property and the attribute's namespace MUST be urn:schemas-microsoft-com:xml-msprop. The attribute's value MUST be the value of the extended property.
 - Any attributes with a namespace other than urn:schemas-microsoft-com:xml-msprop are not extended properties.
- Each extended property MUST appear as an attribute in the declaration of the **DataTable** element (for properties that apply to tables). The attribute name MUST be the name of the extended property and the attribute's namespace MUST be urn:schemas-microsoft-com:xml-msprop. The attribute's value MUST be the value of the extended property.
 - Any attributes with a namespace other than urn:schemas-microsoft-com:xml-msprop are not extended properties.

The following is a sample **Schema** element from a **DiffGram**:

```
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata"
xmlns:msprop="urn:schemas-microsoft-com:xml-msprop">
  <xs:element name="SalesDS" msdata:IsDataSet="true">
    <xs:complexType>
      <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element name="Customers" msprop:ExtProp1="USA">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="CustId" type="xs:int" minOccurs="0"/>
              <xs:element name="CustName" type="xs:string" minOccurs="0"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:choice>
    </xs:complexType>
    <xs:unique name="Constraint2" msdata:PrimaryKey="true">
      <xs:selector xpath="./Customers" />
      <xs:field xpath="CustId" />
    </xs:unique>
  </xs:element>
```

```
</xs:schema>
```

In this sample, the element **SalesDS** defines the **DataInstance** element. It can contain **Customers** elements, which are known as **DataTable** elements, corresponding to rows in the Customers table in the **DataSet**. The columns of the Customers table (CustId and CustName), which are known as **DataColumn** elements, are represented as child elements of the **Customers** element. The primary key of the **DataTable** element is specified in the **Unique** element. The selector defines which **DataTable** element to reference. The field element defines what **DataColumn** element is part of the primary key. An extended property on the Customers table, **ExtProp1** is manifested as an attribute on the Customers **DataTable** element declaration.

2.3.3 Unique Element

A **Unique** element is specified as an XML element per [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#) specifications. A **Unique** element MUST follow these rules:

- The element MUST have an `msdata:PrimaryKey` where the value is equal to true.
- The **selector xpath** attribute value MUST use `'./data-table'` XPath-abbreviated syntax described in section 2.5 of the [\[XPATH\]](#) specification where `data-table` is the name of a **DataTable** element.
- A **field xpath** attribute value MUST use the `'data-column'` XPath-abbreviated syntax where `data-column` is the name of a **DataColumn** element. The following is an example of a **Unique** element:

```
<xs:unique name="Constraint2" msdata:PrimaryKey="true">
  <xs:selector xpath="./Customers" />
  <xs:field xpath="CustId" />
</xs:unique>
```

2.4 DiffGram Data Element

Conceptually, the **DiffGram Data** element encapsulates the XML representation of the data in the **DataSet**. The **DiffGram Data** element MUST have one child element:

- A [DataInstance](#) element, which MUST conform to the **DataInstance** element definition specified in the [DiffGram Schema](#) element.

This high-level structure is defined by the following XML Schema:

```
<?xml version="1.0" standalone="yes"?>
<xs:schema targetNamespace="urn:schemas-microsoft-com:xml-diffgram-v1"
  xmlns="urn:schemas-microsoft-com:xml-diffgram-v1"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1"
  attributeFormDefault="qualified"
  elementFormDefault="qualified">
  <xs:attribute name="id" type="xs:string"/>
  <xs:element name="diffgram">
    <xs:complexType>
      <xs:sequence minOccurs="0" maxOccurs="1">
        <xs:any namespace="##other" processContents="lax" minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

```
</xs:complexType>
</xs:element>
</xs:schema>
```

However, it is not possible to write a single schema for the **DiffGram Data** element that properly defines the **DataInstance** element, because the schema of the **DataInstance** element is defined by the **DiffGram Schema** element (and varies per **DataSet**). Therefore, the schema specified previously in this section allows any content in the place of the **DataInstance** element. Producers and consumers of **DiffGrams** MUST ensure that the **DataInstance** element complies with the schema defined in the SharePoint DiffGram Schema element, as specified in section [2.3.1](#).

The following section defines the parts of the **DiffGram Data** element in more detail.

2.4.1 DataInstance Element

The **DataInstance** element MAY contain one child element named **DocumentElement**. The **DocumentElement** MUST contain zero or more first-level child elements, referred to as "row elements". If the **DocumentElement** is not present, then the **DataInstance** element MUST contain zero or more row elements. Each row element MUST conform to a **DataTable** element definition, as specified in section [2.3.2](#).

A row element MUST contain zero or more first-level child elements, referred to as "column elements". Each column element MUST conform to a **DataColumn** element definition. A column element's content represents a data value. A NULL data value is represented by the absence of a column element in a row element, or by using the <http://www.w3.org/2001/XMLSchema:nil> attribute.

The following is an example of a **DataInstance** element that can appear in a **DiffGram Data** element:

```
<SalesDS>
  <Customers diffgr:id="Customers1" msdata:rowOrder="0" >
    <CustId>1</CustId>
    <CustName>C1</CustName>
  </Customers>
  <Customers diffgr:id="Customers2" msdata:rowOrder="1">
    <CustId>2</CustId>
    <CustName>C2</CustName>
  </Customers>
  <Customers diffgr:id="Customers3" msdata:rowOrder="2" >
    <CustId>3</CustId>
    <CustName>C3</CustName>
  </Customers>
</SalesDS>
```

The following rules apply to the row elements within the **DataInstance** element:

- Each row element MUST have an **urn:schemas-microsoft-com:xml-diffgram-v1:id** attribute. The string value of this attribute acts as the row identifier within the scope of the **DiffGram Data** element. The **DataInstance** element MUST NOT have two row elements with the same value for the **urn:schemas-microsoft-com:xml-diffgram-v1:id** attribute.
- Each row element MUST have an **urn:schemas-microsoft-com:xml-msdata:rowOrder** attribute whose value MUST be an integer that specifies the 0-based ordinal position of the row element. The **DataInstance** element MUST NOT have two row elements with the same value for

the **urn:schemas-microsoft-com:xml-msdata:rowOrder** attribute. The value for the **urn:schemas-microsoft-com:xml-msdata:rowOrder** attribute MUST be less than the count of the rows in the **DataInstance** element.

- Each row element MAY have an **urn:schemas-microsoft-com:xml-msdata:hasChanges** attribute. If this attribute is specified, the value MUST be "inserted", "modified", or "decent".

3 Structure Examples

Consider a SharePoint Search with the keywords "Cool Bikes". The results consist of a **DataSet** containing one **DataTable** named "RelevantResults", which contains the results of the search. This **DataSet** is serialized into an XML document that consists of one **DataTable** element with the name "RelevantResults", with a number of child **DataColumn** elements. Note that while the type of the **HitHighlightedProperties DataColumn** is a string, the contents appear to be XML. However, they are serialized by the **DataSet** as a string, not XML, and are not encoded as XML.

The following XML is the **DiffGram** structure that encapsulates this search result:

```
<?xml version="1.0" encoding="utf-8" ?>
<DataSet xmlns="http://tempuri.org/">
  <xs:schema id="Results" xmlns=""
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:msdata="urn:schemas-microsoft-com:xml-msdata"
    xmlns:msprop="urn:schemas-microsoft-com:xml-msprop">
    <xs:element name="Results" msdata:IsDataSet="true"
      msdata:UseCurrentLocale="true" msprop:QueryTerms="Cool Bikes;"
      msprop:IgnoredNoiseWords="" msprop:Keyword="" msprop:ElapsedTime="938"
      msprop:Definition="" msprop:SpellingSuggestion="">
      <xs:complexType>
        <xs:choice minOccurs="0" maxOccurs="unbounded">
          <xs:element name="RelevantResults" msprop:TotalRows="175"
            msprop:IsTotalRowsExact="False">
            <xs:complexType>
              <xs:sequence>
                <xs:element name="WorkId" type="xs:long" minOccurs="0" />
                <xs:element name="Rank" type="xs:long" minOccurs="0" />
                <xs:element name="Title" type="xs:string" minOccurs="0" />
                <xs:element name="Author" type="xs:string" minOccurs="0" />
                <xs:element name="Size" type="xs:long" minOccurs="0" />
                <xs:element name="Path" type="xs:string" minOccurs="0" />
                <xs:element name="Description" type="xs:string" minOccurs="0" />
                <xs:element name="Write" type="xs:dateTime" minOccurs="0" />
                <xs:element name="SiteName" type="xs:string" minOccurs="0" />
                <xs:element name="CollapsingStatus" type="xs:long" minOccurs="0" />
                <xs:element name="HitHighlightedSummary" type="xs:string"
                  minOccurs="0" />
                <xs:element name="HitHighlightedProperties" type="xs:string"
                  minOccurs="0" />
                <xs:element name="ContentClass" type="xs:string" minOccurs="0" />
                <xs:element name="IsDocument" type="xs:long" minOccurs="0" />
                <xs:element name="PictureThumbnailURL" type="xs:string"
                  minOccurs="0" />
              </xs:sequence>
            </xs:complexType>
          </xs:element>
        </xs:choice>
      </xs:complexType>
    </xs:element>
  </xs:schema>
  <diffgr:diffgram xmlns:msdata="urn:schemas-microsoft-com:xml-msdata"
    xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1">
    <Results xmlns="">
      <RelevantResults diffgr:id="RelevantResults1" msdata:rowOrder="0">
        <WorkId>1321891</WorkId>
        <Rank>822</Rank>
```

```

<Title>New Metro Sport Equipment Bikes</Title>
<Author>Ms.Kim Abercrombie</Author>
<Size>8276480</Size>
<Path>file://PublicShare/BikesConference/postshow/NewModels.ppt</Path>
<Description>Metro Sport Equipment Bikes is introducing Bikes for this
model year - this slide deck shows the new models and options</Description>
<Write>2006-10-06T14:46:27.7529559-07:00</Write>
<SiteName>file://PublicShare/BikesConference</SiteName>
<CollapsingStatus>0</CollapsingStatus>
<HitHighlightedSummary>Metro Sport Equipment Bikes is introducing Bikes
for this model year - this slide deck shows the new models and
options</HitHighlightedSummary>
<HitHighlightedProperties>
  <HHTitle>
    Bike Retailers - Always ready to ride
  </HHTitle>
  <HHUrl>
    file://PublicShare/BikesConference/postshow/NewModels.ppt
  </HHUrl>
</HitHighlightedProperties>
<IsDocument>1</IsDocument>
</RelevantResults>
<RelevantResults diffgr:id="RelevantResults2" msdata:rowOrder="1">
  <WorkId>26116233</WorkId>
  <Rank>793</Rank>
  <Title>How to care for BB Ball Bearings</Title>
  <Author>Mr.GustavoAchong</Author>
  <Size>50004</Size>
  <Path>http://bikewiki/Parts/Wiki Pages/BB_Ball_Bearings.aspx</Path>
  <Write>2008-04-01T22:00:46-07:00</Write>
  <SiteName>http://bikewiki/Parts</SiteName>
  <CollapsingStatus>0</CollapsingStatus>
  <HitHighlightedSummary>
    BB Ball Bearings are known for being hard to take care of, but actually
they aren't too bad if you follow these simple directions.
  </HitHighlightedSummary>
  <HitHighlightedProperties>
    <HHTitle>How to care for BB Ball Bearings</HHTitle>
    <HHUrl>http://bikewiki/Parts/Wiki Pages/BB_Ball_Bearings.aspx</HHUrl>
  </HitHighlightedProperties>
  <ContentClass>STS_ListItem_WebPageLibrary</ContentClass>
  <IsDocument>1</IsDocument>
</RelevantResults>
<RelevantResults diffgr:id="RelevantResults3" msdata:rowOrder="2">
  <WorkId>5522013</WorkId>
  <Rank>714</Rank>
  <Title>014 PPS Build</Title>
  <Author>Mr. Samuel N. Agcaoili</Author>
  <Size>253623</Size>
  <Path>http://sharepoint/sites/PerformanceBikes/Build Changes.docx</Path>
  <Write>2008-02-18T15:03:43-08:00</Write>
  <SiteName>http://sharepoint/sites/PerformanceBikes/</SiteName>
  <CollapsingStatus>0</CollapsingStatus>
  <HitHighlightedSummary>
    No Summary available
  </HitHighlightedSummary>
  <HitHighlightedProperties>
    <HHTitle>014 PPS Build </HHTitle>
    <HHUrl>http://sharepoint/sites/PerformanceBikes/Build

```

```
Changes.docx</HHUrl>  
  </HitHighlightedProperties>  
  <ContentClass>STS_ListItem_DocumentLibrary</ContentClass>  
  <IsDocument>1</IsDocument>  
</RelevantResults>  
</Results>  
</diffgr:diffgram>  
</DataSet>
```

4 Security Considerations

None.

5 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products and technologies:

- Microsoft® Office SharePoint® Server 2007
- 2007 Microsoft® Office system
- Windows® SharePoint Services 3.0

Exceptions, if any, are noted below. Unless otherwise specified, any statement of optional behavior in this specification prescribed using the terms SHOULD or SHOULD NOT implies the aforementioned Microsoft products' behavior is in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that these Microsoft products do not follow the prescription.

6 Change Tracking

This section identifies changes made to [MS-DSDIFFGRAM] protocol documentation between November 2009 and March 2010 releases. Changes are classed as major, minor, or editorial.

Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- An extensive rewrite, addition, or deletion of major portions of content.
- A protocol is deprecated.
- The removal of a document from the documentation set.
- Changes made for template compliance.

Minor changes do not affect protocol interoperability or implementation. Examples are updates to fix technical accuracy or ambiguity at the sentence, paragraph, or table level.

Editorial changes apply to grammatical, formatting, and style issues.

No changes means that the document is identical to its last release.

Major and minor changes can be described further using the following revision types:

- New content added.
- Content update.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.
- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- New content added for template compliance.
- Content updated for template compliance.

- Content removed for template compliance.
- Obsolete document removed.

Editorial changes always have the revision type "Editorially updated."

Some important terms used in revision type descriptions are defined as follows:

Protocol syntax refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.

Protocol revision refers to changes made to a protocol that affect the bits that are sent over the wire.

Changes are listed in the following table. If you need further information, please contact protocol@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Revision Type
1.2.1 Normative References	Replaced the reference to [XMLNS] with a reference to [XMLNS3], which is the updated version that supersedes [XMLNS].	N	Content updated due to protocol revision.

7 Index

A

[Applicability](#) 6

C

[Change tracking](#) 19

D

[DataInstance element](#) 12

[DataInstance element schema](#) 8

[DataTable element schema](#) 9

[DiffGram Data element](#) 11

[DiffGram namespaces](#) 7

E

Element schema

[DataInstance](#) 8

[DataTable](#) 9

Examples

[overview](#) 14

F

Fields

[vendor-extensible](#) 6

G

[Glossary](#) 4

H

[high-level structure](#) 7

I

[Informative references](#) 5

[Introduction](#) 4

L

[Localization](#) 6

N

[Namespaces](#) 7

[Normative references](#) 4

O

[Overview](#) 5

P

[Product behavior](#) 18

R

[References](#) 4

[informative](#) 5

[normative](#) 4

[Relationship to other protocols](#) 6

S

Security

[overview](#) 17

[SharePoint DiffGram high-level structure](#) 7

[SharePoint DiffGram Schema element](#) 8

Structures

[high-level](#) 7

[overview](#) 7

T

[Tracking changes](#) 19

U

[Unique element](#) 11

V

[Vendor-extensible fields](#) 6

[Versioning](#) 6