[MS-DPPDW-Diff]:

Parallel Data Warehouse Data Portability Overview

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation ("this documentation") for protocols, file formats, data portability, computer languages, and standards as well as overviews of the interaction among each of these technologiessupport. Additionally, overview documents cover inter-protocol relationships and interactions.
- **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 maycan make copies of it in order to develop implementations of the technologies that are described in the Open Specifications this documentation and maycan distribute portions of it in your implementations usingthat use these technologies or in your documentation as necessary to properly document the implementation. You maycan also distribute in your implementation, with or without modification, any schema, IDL'sschemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications. documentation.
- **No Trade Secrets**. Microsoft does not claim any trade secret rights in this documentation.
- **Patents**. Microsoft has patents that maymight cover your implementations of the technologies described in the Open Specifications. documentation. Neither this notice nor Microsoft's delivery of thethis documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specification maySpecifications document might be covered by the Microsoft Open Specifications Promise or the Microsoft Community Promise. If you would prefer a written license, or if the technologies described in the Open Specificationsthis documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **License Programs**. To see all of the protocols in scope under a specific license program and the associated patents, visit the Patent Map.
- **Trademarks**. The names of companies and products contained in this documentation maymight be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit .www.microsoft.com/trademarks.
- **Fictitious Names**. The example companies, organizations, products, domain names, e-mailemail addresses, logos, people, places, and events that are 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 as specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications dodocumentation does 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 documents are intended for use in conjunction with publicly available standardstandards specifications and network programming art, and assumes, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

Support. For questions and support, please contact dochelp@microsoft.com.

Revision Summary

| Date	Revision History	Revision Class	Comments
2/9/2011	0.1	New	Released new document.
7/7/2011	0.1	None	No changes to the meaning, language, or formatting of the technical content.
11/3/2011	0.1	None	No changes to the meaning, language, or formatting of the technical content.
1/19/2012	0.1	None	No changes to the meaning, language, or formatting of the technical content.
2/23/2012	0.1	None	No changes to the meaning, language, or formatting of the technical content.
3/27/2012	0.1	None	No changes to the meaning, language, or formatting of the technical content.
5/24/2012	0.1	None	No changes to the meaning, language, or formatting of the technical content.
6/29/2012	0.1	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	0.1	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	0.1	None	No changes to the meaning, language, or formatting of the technical content.
10/23/2012	0.1	None	No changes to the meaning, language, or formatting of the technical content.
3/26/2013	0.1	None	No changes to the meaning, language, or formatting of the technical content.
6/11/2013	0.1	None	No changes to the meaning, language, or formatting of the technical content.
8/8/2013	0.1	None	No changes to the meaning, language, or formatting of the technical content.
12/5/2013	0.1	None	No changes to the meaning, language, or formatting of the technical content.
2/11/2014	0.1	None	No changes to the meaning, language, or formatting of the technical content.
5/20/2014	0.1	None	No changes to the meaning, language, or formatting of the technical content.
5/10/2016	1.0	None	No changes to the meaning, language, or formatting of the technical content.
8/16/2017	1.0	None	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1 Introduction

The Parallel Data Warehouse Data Portability Overview document provides an overview of the components and methodologies that are used for data portability within the SQL Server Parallel Data Warehouse (PDW) system.

This document provides examples of some of the common user scenarios for data export. It does not restate the details of formats that are used for data portability. These details are described in the specifications for each of the formats that are used by this system.

This document provides an overview of data portability between a Microsoft SQL Server 2008 R2 Parallel Data Warehouse (PDW) database and another ~~vendor's~~vendor's database. The document focuses on exporting data from SQL Server PDW to a SQL Server 2008 R2 database. The BCP data portability document [MS-DPBCP] focuses on portability between a SQL Server 2008 R2 database and another ~~vendor's~~vendor's database.

SQL Server PDW is a highly scalable appliance for enterprise data warehousing that uses massively parallel processing (MPP) to deliver high performance and scalability on Microsoft SQL Server 2008, the Windows Server 2008 operating system, and industry-standard hardware. The appliance ships with software, hardware, and networking components that are architected together to provide high-speed parallel query processing, highly scalable data storage, and high-speed data transfer among the appliance components.

SQL Server PDW consists of a set of nodes. Each node has specific hardware and software that are architected together to carry out the functions of that node. Data is usually loaded through the node called the Landing Zone. Data is stored on a set of 8 to 10 or more Compute nodes. Queries run in parallel on the Compute nodes. The results are finalized on the Control node, and then returned to the user.
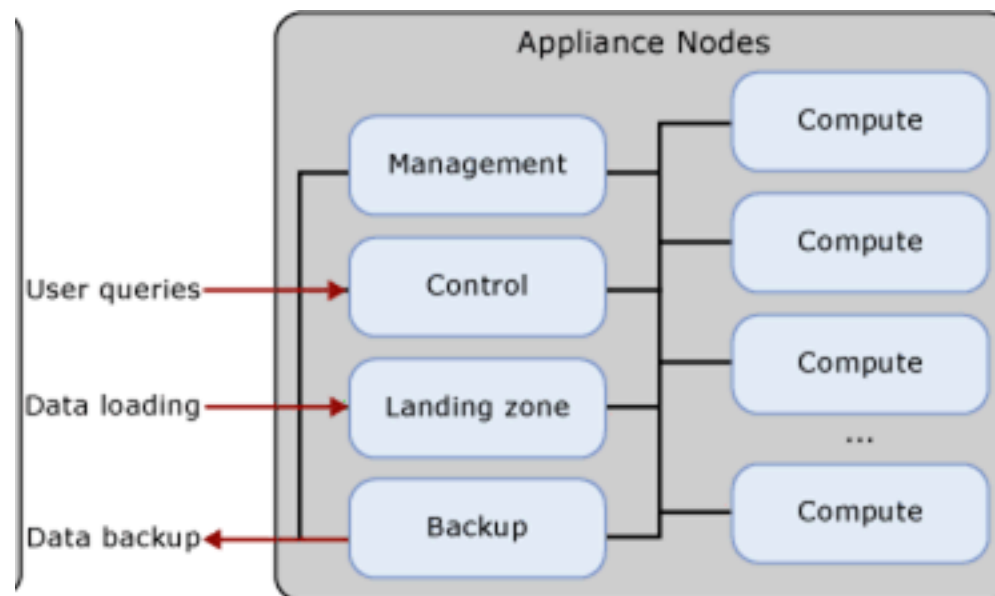
Figure 1: SQL Server PDW appliance nodes

The data export scenario in this document describes how to use SQL Server Integration Services to use the SQL Server PDW parallel data export feature to export data to a SQL Server 2008 R2 database.
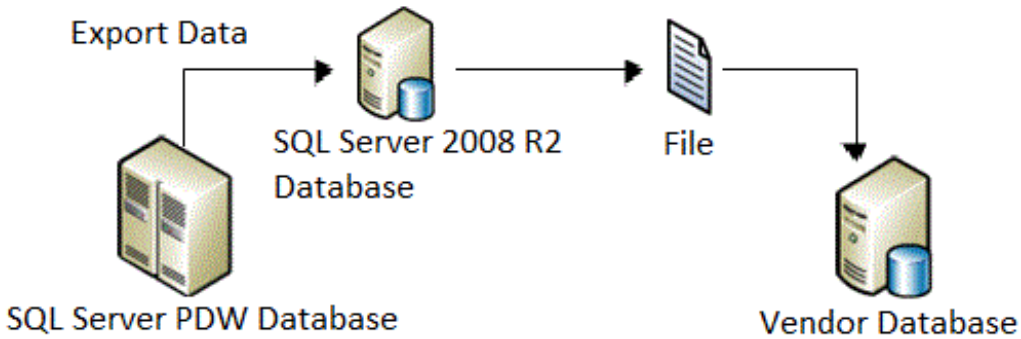
Figure 2: Conceptual overview of SQL Server PDW export data portability

1.1   Glossary

This document uses the following terms:

appliance: An application in which hardware and software are architected together to function as one box. Users can access the box only through the user interface of the application.

Compute node: An appliance node that is the basic unit of scalability and storage. Each Compute node in the SQL Server PDW appliance uses its own user-data and computing resources to perform a portion of each parallel query.

Control node: An appliance node that is the central point of control for processing queries on the SQL Server PDW appliance. The Control node receives the user query, creates a distributed query plan, communicates relevant plan operations and data to Compute nodes, receives Compute node results, performs any necessary aggregation of results, and then returns the query results to the user.

Infiniband network: A high-speed switched fabric network. In SQL Server PDW, Infiniband is used for private communication inside a SQL Server PDW appliance.

Landing Zone node: An appliance node that provides temporary storage and processing for loading data onto the appliance.

Management node: An appliance node that performs multiple functions related to managing the hardware and software in the appliance. This node is the hub for software deployment and servicing, authentication within the appliance (not login authentication), and monitoring system health and performance.

massively parallel processing (MPP): An architecture that distributes the workload across multiple CPUs or machines, resulting in faster performance. In MPP architecture, query processing occurs within multiple instances of a database that each has dedicated CPU, memory, and storage.

parallel data export: A feature of SQL Server PDW that enables hub-and-spoke scenarios by exporting the results of a SQL SELECT statement, in parallel, to an existing table in an SMP database. The parallel data export is initiated with the CREATE REMOTE TABLE SQL statement.

SMP database: A database that is stored in a symmetric multiprocessing (SMP) system. An SMP system is a multi-processor computer architecture in which identical processors share main memory. The architecture allows simultaneous processing to be distributed among the processors. In an SMP architecture, query processing occurs within one physical instance of a database. SQL Server 2008 R2 uses SMP architecture, and its databases can be referred to as SMP databases.

SQL Server PDW appliance: An appliance that runs the SQL Server PDW application. The SQL Server PDW appliance consists of software and at least two racks of servers, storage, and network components that are architected together.

SQL Server PDW database: A user database that is stored in SQL Server PDW.

1.2   References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

[MS-DPBCP] Microsoft Corporation, "Bulk Copy Utility Data Portability Overview".

[MSDN-DIPIS] Microsoft Corporation, "Designing and Implementing Packages (Integration Services)", http://msdn.microsoft.com/en-us/library/ms141091.aspx

2   Data Portability Scenarios

The user can use the SQL Server PDW parallel data export feature to export tables from SQL Server PDW to an existing Microsoft SQL Server 2008 R2 database.

The user can use the Bulk Copy utility (BCP) to import and export a SQL Server 2008 R2 database to and from a vendor database. This process is outside the scope of this document. For more information, see [MS-DPBCP].

2.1   Export Data

This data export scenario describes exporting data from SQL Server PDW by exporting the results of a SELECT statement, in parallel, to a table in an SMP database. This enables hub-and-spoke scenarios.

The SMP database is in an instance of Microsoft SQL Server that is running on a Windows system that is attached to the internal Infiniband network of the SQL Server PDW appliance.

For information about how to export data from a SQL Server 2008 R2 database to a text file that can be imported into another ~~vendor's~~vendor's database, see [MS-DPBCP].

2.1.1   Data Description

Customer data

The customer data is a text file representation of a database object in a ~~vendor's~~vendor's database.

Intended user

The intended user is a vendor who can import data into a vendor database from files that are produced from a SQL Server 2008 R2 database.

2.1.2   Format and Protocol Summary

Integration Services uses the ADO.NET client for SQL Server PDW to connect to SQL Server PDW. This client is available under the redistributable share of the Landing Zone node on the SQL Server PDW appliance.

The following table provides a comprehensive list of the formats and protocols used in this data portability scenario.

Protocol or format name	Description	Reference
ADO.NET client for SQL Server PDW	Connection interface from Integration Services to SQL Server PDW. This connection interface ~~must~~is required to be installed on the same client computer that will be running Integration Services.	[MSDN-DIPIS]

2.1.3   Data Portability Methodology

Exporting data from SQL Server PDW is a four-step process, as follows:

1. Configure an external Windows system to receive parallel data exports. Or, configure an external SQL Server SMP database to receive parallel data exports.

2. Update host names in DNS.

3. Run the CREATE REMOTE TABLE statement.

4. Use BCP [MS-DPBCP] to export the data from the SQL Server SMP database to a file that can be imported into another vendor's database.

Step 1a: Configure an external Windows system to receive parallel data exports

The parallel data export feature exports data from the SQL Server PDW appliance to an external SQL Server SMP database that is running on a Windows system. ~~To use~~Use of this feature~~,~~ requires a Windows system that will receive the exported data ~~must~~to be provided. To do this, follow these steps:

1. Purchase a Windows system and rack it close enough to the appliance that it can be connected to the appliance Infiniband network.

2. Purchase Infiniband cables and an Infiniband network adapter from your appliance hardware vendor. We recommend purchasing a network adapter with two ports for fault tolerance when receiving the exported data. A two-port network adapter is recommended but is not a requirement.

3. Install the Infiniband network adapter on the Windows system.

4. Connect the Infiniband network adapter to the main Infiniband switch in the Control rack by using Infiniband cables.

5. Install and configure the appropriate Windows driver for the Infiniband network adapter.

    Infiniband drivers for Windows are developed by the OpenFabrics Alliance, an industry consortium of Infiniband vendors. The correct driver ~~may~~might have been distributed with the Infiniband adapter. If not, the driver can be downloaded from www.openfabrics.org.

6. Configure the IP address for each port on the adapter. SMP systems are required to use static IP addresses from a range of addresses reserved for this purpose. Configure the first port according to the following parameters:

    -   IP network address: 172.16.132.x

    -   IP subnet mask: 255.255.128.0

    -   IP host range: 1-254

    For Infiniband network adapters with two ports, configure the second port according to the following parameters:

    -   IP network address: 172.16.132.x

    -   IP subnet mask: 255.255.128.0

    -   IP host range: 1-254

7. If a two-port adapter is used or multiple external Windows systems are connected to an appliance, assign each system a different host number within each IP subnet.

Step 1b: Configure an external SQL Server SMP database to receive parallel data exports

The parallel data export feature exports data from the SQL Server PDW appliance to an external SQL Server SMP database that is running on a Windows system. After the external Windows system is configured to receive parallel data exports, the next step is to install and configure SQL Server Enterprise Edition on the Windows system.

To configure an external SQL Server SMP database to receive parallel data exports, follow these steps:

1. Install Microsoft SQL Server 2008 Enterprise Edition or SQL Server 2008 R2 Enterprise Edition on the Windows system. SQL Server 2008 R2 is an SMP system.

2. Configure SQL Server to accept TCP/IP connections on a fixed TCP port. This configuration is disabled by default and ~~must~~is required to be enabled to allow SQL Server PDW to connect to the SQL Server SMP database.

3. Disable Windows Firewall. Or, configure the TCP port of the SQL Server SMP database so that it will work with Windows Firewall enabled.

4. Configure SQL Server to allow SQL Server Authentication mode. The parallel data export always uses SQL Server accounts for authentication.

5. Choose a SQL Server account on the SQL Server SMP database (or create a new account) to be used for authentication. Grant this account privileges to create, drop, and insert data into tables in the destination database for the parallel data export operation.

Step 2: Update host names in DNS

The CREATE REMOTE TABLE statement, used for parallel data exports, specifies the destination server by using either the IP address or the IP name of the Windows system that is running the SQL Server SMP database. To use the IP name, add entries for successful name resolution to the DNS server on the SQL Server PDW Management node. To do this, follow these steps:

1. Log on to the SQL Server PDW Management node.

2. Open the DNS Manager. This is located under **Administrative Tools** on the **Start** menu.

3. Use the DNS Manager to add the IP name.

Step 3: Run the CREATE REMOTE TABLE statement

The CREATE REMOTE TABLE statement selects data from a SQL Server PDW database and copies that data to a new table in a SQL Server 2008 or SQL Server 2008 R2 SMP database. The SMP database does not reside on a server in the SQL Server PDW appliance. This provides the flexibility to handle scenarios that require SQL Server functionality.

The following is an example of the syntax for the CREATE REMOTE TABLE statement.

```
CREATE REMOTE TABLE [ database_name . [ schema_name ] . | schema_name. ] table_name
    AT ('<connection_string>')
    [ WITH ( BATCH_SIZE = batch_size ) ]
    AS SELECT select_criteria
}
[;]
<connection_string> ::=
    Data Source = [tcp:] { IP_address | hostname } [ \ instance_name ]    [, port ]; User ID
= user_name ;Password = password;
```

The following table describes the arguments that are used in the CREATE REMOTE TABLE statement.

Argument	Description
database_name	The SQL Server 2008 or SQL Server 2008 R2 database in which to create the remote table. The default is the default database for the user login on the destination SQL Server instance.
schema_name	The schema for the new table. The default is the default schema for the user login on the destination SQL Server instance.
table_name	The name of the new table. The remote table is created as a heap. It does not have check constraints or triggers. The

Argument	Description
	collation of the remote table columns is the same as the collation of the source table columns. This collation applies to columns of type **char**, **nchar**, **varchar**, and **nvarchar**.
connection_string	A character string that specifies the data source, user ID, and password parameters for connecting to the remote server and database. The connection string is a semicolon-delimited list of key/value pairs. Keywords are not case-sensitive. Spaces between key/value pairs are ignored. However, values ~~may~~might be case-sensitive, depending on the data source.
tcp	The TCP protocol. This is the default and the only supported protocol for this release.
IP_address	The IPv4 address of the remote server. IPv6 addresses are not supported.
hostname	The name of the remote server. The server ~~must~~is required to be remote and therefore cannot be specified as (local).
instance_name	The name of the remote database instance. The maximum number of characters is 16.
port	The port number from 0 to 65535 for the remote server connection.
user_name	A valid SQL Server 2008 R2 user name. The maximum number of characters is 128.
password	A valid SQL Server 2008 R2 password. The maximum number of characters is 128.
batch_size	The maximum number of rows per batch. SQL Server PDW sends rows in batches to the destination server. *Batch_size* is a positive integer that is greater than or equal to 0. The default is 0.
select_criteria	The query predicate that specifies which data will populate the new remote table. The TOP and ORDER BY clauses are not supported.

The following example creates a remote table named MyOrdersTable on the OrderReporting database and the Orders schema. The OrderReporting database is on a server named SQLA. Port 8080 is used for the connection. The connection to the server is made with the credentials of the user whose user name is David and whose password is e4n8@3.

```
CREATE REMOTE TABLE OrderReporting.Orders.MyOrdersTable
AT ( 'Data Source = SQLA, 8080; User ID = David; Password = e4n8@3;' )
AS SELECT <select_criteria>;
```

Step 4: Use BCP to export the data from the SQL Server SMP database to a file that can be imported into another vendor's database

To do this, see [MS-DPBCP].

2.1.3.1 Preconditions

The SMP database cannot reside on a server that is within the SQL Server PDW appliance.

The destination server ~~must~~is required to be a non-appliance server. The CREATE REMOTE TABLE statement cannot be used to copy data from one appliance to another.

The remote SMP database server destination ~~must~~is required to be accessible to all nodes on the appliance's Infiniband network. Creating tables on database servers that are external to the appliance's Infiniband network is not supported.

2.1.3.2 Versioning

None.

2.1.3.3 Error Handling

If copying data to the remote database fails when the CREATE REMOTE TABLE statement is run, SQL Server PDW aborts the operation, logs an error, and attempts to delete the remote table. SQL Server PDW does not guarantee a successful cleanup of the new table. If SQL Server PDW cannot delete the table, it retries for several minutes. If the retries are unsuccessful, the rollback record becomes inactive.

2.1.3.4 Coherency Requirements

There are no special coherency requirements.

2.1.3.5 Additional Considerations

There are no additional considerations.

3 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

4   Index