[MS-MOSSUAN]:
SQL Usage Analytics Protocol 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 Open Specification Promise or the Community Promise. 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 iplg@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		Initial Availability
06/27/2008	1.0	Major	Revised and edited the technical content
12/12/2008	1.01	Editorial	Revised and edited the technical content
03/18/2009	1.02	Editorial	Revised and edited the technical content
07/13/2009	1.03	Major	Changes made for template compliance
08/28/2009	1.04	Editorial	Revised and edited the technical content
11/06/2009	1.05	Editorial	Revised and edited the technical content
02/19/2010	2.0	Editorial	Revised and edited the technical content
03/31/2010	2.01	Editorial	Revised and edited the technical content
04/30/2010	2.02	Editorial	Revised and edited the technical content
06/07/2010	2.03	Editorial	Revised and edited the technical content
06/29/2010	2.04	Editorial	Changed language and formatting in the technical content.
07/23/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
09/27/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
06/10/2011	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
01/20/2012	2.5	Minor	Clarified the meaning of the technical content.
04/11/2012	2.5	No change	No changes to the meaning, language, or formatting of the technical content.
07/16/2012	2.5	No change	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1 Introduction

This document specifies the SQL Usage Analytics Protocol, which is used to collect, access, and analyze data for usage analysis reports about a Web site. This protocol is not an extension to an existing protocol.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in RFC 2119. Sections 1.5 and 1.9 are also normative but cannot contain those terms. All other sections and examples in this specification are informative.

1.1 Glossary

The following terms are defined in [MS-OFCGLOS]:

back-end database server
clickthrough
document
front-end Web server
hit
home page
host header
HTTP referer
result set
return code
site collection
site collection identifier
site identifier
stored procedure
store-relative form
subsite
transaction
Transact-Structured Query Language (T-SQL)
Uniform Resource Identifier (URI)
Uniform Resource Locator (URL)
usage data
user name
Web application
Web application identifier
Web site

The following terms are specific to this document:

shadow resource identifier: A unique identifier for information that is stored in the ANLShadowResource table in the usage shadow data store.

shadow user identifier: A unique identifier for information that is stored in the ANLShadowUser table in the usage shadow data store.

Shared Services Provider (SSP) database: A database that stores service-specific data, such as user information, site usage statistics, business intelligence methods, and calculated audiences.

usage detailed-history period: The number of consecutive days that a protocol server stores detailed usage data.

usage master data store: The location where usage information is stored.

usage shadow data store: The location where usage information can be placed temporarily before it is summarized and moved into a usage master data store.

usage summary-history period: The number of days that a protocol server stores aggregate usage data.

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

References to Microsoft Open Specifications documentation do not include a publishing year because links are to the latest version of the technical documents, which are updated frequently. References to other documents include a publishing year when one is available.

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.

[Iseminger] Microsoft Corporation, "SQL Server 2000 Architecture and XML/Internet Support", Volume 1 of Microsoft SQL Server 2000 Reference Library, Microsoft Press, 2001, ISBN 0-7356-1280-3, http://www.microsoft.com/mspress/books/5001.aspx

[MSDN-TSQL-Ref] Microsoft Corporation, "Transact-SQL Reference", http://msdn.microsoft.com/en-us/library/ms189826(SQL.90).aspx

[MS-TDS] Microsoft Corporation, "Tabular Data Stream Protocol Specification".

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

1.2.2   Informative References

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

1.3   Protocol Overview (Synopsis)

This protocol is used to provide data for usage analysis reports about a **Web site (2)**. The protocol server collects and stores this data, which protocol clients can then request for reports. A usage analysis report can be used to identify which **documents** are visited most and least frequently, and the path that users take when they visit various documents on the Web site (2). Reporting packages such as those that can be developed by using this protocol allow a site administrator to examine these types of usage data, which are typically referred to as Web analytics.

This protocol allows a protocol server to store and report the following usage statistics:

▪ **Hits** and **clickthrough** for documents. The scope of these statistics can include an entire **site collection**, a single Web site (2), or a single **subsite**. For example, if a user visits "Page A" on

the Web site (2), and then follows a link to "Page B" on the same Web site (2), the usage data counts one hit on "Page A," one hit on "Page B," and one clickthrough from "Page A" to "Page B."

- Hits to the **home page** for a Web site (2). This type of hit is reported separately from other hits because the home page is frequently the starting point for users when they access a Web site (2).

- Search requests for a site collection. This type of usage data includes terms that users entered when searching for documents on a Web site (2). For example, if a user searches for "Widget", that activity is collected and stored for use in a report.

- The **user names** of users who access the Web site (2).

The protocol server stores this **usage data** for the number of days specified by a **usage detailed-history period**. After that time period ends, the data is stored at a summary level for the number of days specified in a **usage summary-history period**. After the usage summary-history period ends, the data is deleted to help ensure optimal server performance and space availability.

The following diagram illustrates a typical data flow between the protocol client and the protocol server:
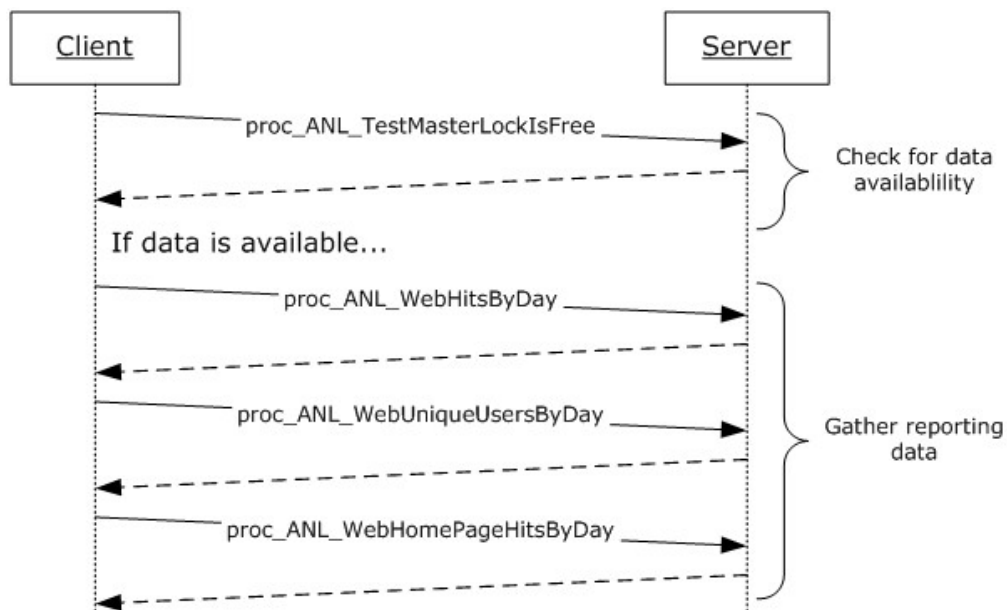
Figure 1: Example data flow for gathering report data

1. The protocol client interacts with the protocol server to check for data availability.

2. Depending on the availability of the data, the protocol client proceeds.

3. If the data is available, the protocol client makes a series of calls to gather the report data.

4. If the data is not available, the protocol client shows a message that instructs the user to try again later.

1.4 Relationship to Other Protocols

The SQL Usage Analytics Protocol uses the **Transact-Structured Query Language (T-SQL)** and the Tabular Data Stream (TDS) Protocol over TCP/IP as shown in the following layering diagram:
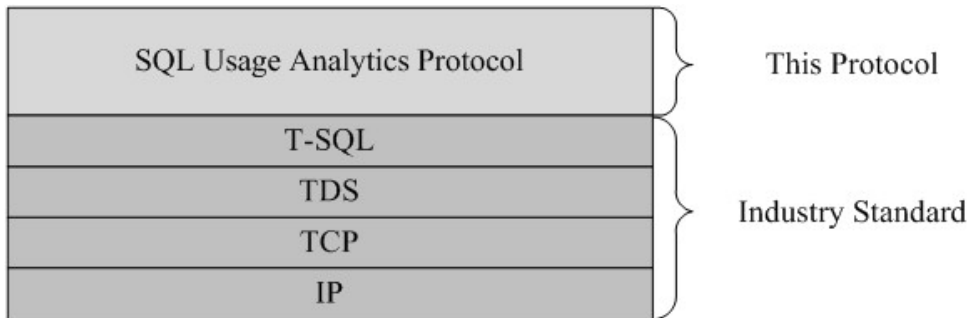

Figure 2: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates between a protocol client and a protocol server that stores the back-end databases for a Web site (2). The protocol client is expected to know the location and connection information for those back-end databases.

This protocol requires a protocol client to have the requisite permission to call the **stored procedures** that are stored on the protocol server.

1.6 Applicability Statement

This protocol provides usage data that can be used to report historical activity for a Web site (2). This data is intended to be used for usage analysis and statistical reporting activities, specifically for applications where historical and summarized usage information is appropriate. Due to the data-processing, data-locking, and time-lag constraints of this protocol, the data that it provides might not be suitable for real-time accounting applications.

This protocol is intended for use by protocol clients and protocol servers that are both connected by high-bandwidth, low-latency network connections.

1.7 Versioning and Capability Negotiation

This protocol supports the Security Support Provider Interface (SSPI) and SQL Server User Authentication (SQLAUTH) with the protocol server role, as described in [MS-TDS].

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

This protocol uses the Tabular Data Stream (TDS) Protocol, as specified in [MS-TDS], as the transport protocol for calling stored procedures, querying SQL views and SQL tables, and returning **result sets** and **return codes**.

2.2 Common Data Types

The following sections define the common data types that are used in this protocol.

2.2.1 Simple Data Types and Enumerations

No common simple data types or enumerations are defined in this protocol.

2.2.2 Bit Fields and Flag Structures

No common bit field or flag structures are defined in this protocol.

2.2.3 Binary Structures

No common binary structures are defined in this protocol.

2.2.4 Result Sets

This section specifies the result sets that are returned by this protocol for more than one stored procedure.

The T-SQL syntax for each result set is as specified in [MSDN-TSQL-Ref]. In the T-SQL syntax, the variable name is followed by the type of variable, which can optionally have a length value in brackets and can optionally have a default value indicated by an equals sign (=) followed by the default value.

2.2.4.1 Average Hits by Resource Result Set

The **Average Hits by Resource** result set returns one row for each document that received a hit during a usage detailed-history period. The rows in the result set MUST be sorted in descending order based on the **AvgDailyHits** field.

The T-SQL syntax for the result set is as follows:

```
FullUrl          nvarchar(260),
DocName          nvarchar(260),
AvgDailyHits     float;
```

FullUrl: The **URL** of the document that was accessed. The value of this field cannot exceed the maximum character limit of 260 characters. This MUST NOT be NULL.

DocName: The **store-relative form** of the URL of the document that was accessed. The value of this field cannot exceed the maximum character limit of 260 characters.

AvgDailyHits: The total number of hits divided by the number of days in the usage detailed-history period.

2.2.4.2   Big Hits By Day Result Set

The **Big Hits by Day** result set returns one row for each day that usage information was collected during a usage detailed-history period. Each row contains the number of hits that were received during a specified day. The rows in the result set MUST be sorted in ascending order based on the **FullDate** field.

The T-SQL syntax for the result set is as follows:

```
FullDate    datetime,
Hits        bigint;
```

FullDate: A timestamp that is in datetime format and specifies the day when the hits were received. This MUST NOT be NULL. The time component of the datetime value MUST be ignored.

Hits: The number of hits that were received on the day specified by the **FullDate** field. This value MUST be NULL if no hits were received on the specified day.

2.2.4.3   Clickthroughs Result Set

The **Clickthroughs** result set returns one row for each **Uniform Resource Identifier (URI)** that was requested and the number of times that URI was accessed during a usage detailed-history period. The rows in the result set MUST be sorted in descending order based on the **Clickthroughs** field.

The T-SQL syntax for the result set is as follows:

```
TargetFullUrl    nvarchar(260),
Clickthroughs    bigint;
```

TargetFullUrl: The URL for each clickthrough that occurred during the usage detailed-history period. The value of this field cannot exceed the maximum character limit of 260 characters. This MUST NOT be NULL.

Clickthroughs: The total number of requests for the URI specified by the **TargetFullUrl** field.

2.2.4.4   Daily Average DNS Referral Count Result Set

The **Daily Average DNS Host Referral Count** result set returns one row for each **host header** that generated daily referrals during a usage detailed-history period. The rows in the result set MUST be sorted in descending order based on the **AvgDailyReferrals** field.

The T-SQL syntax for the result set is as follows:

```
Referrer            nvarchar(64),
AvgDailyReferrals    float;
```

Referrer: A unique host header that referred hits to the specified site collection. The value of this field cannot exceed the maximum character limit of 64 characters. This MUST NOT be NULL.

AvgDailyReferrals: The average daily referral count for the unique host header specified by the **Referrer** field. This value is calculated by dividing the total number of referrals by the number of days in the usage detailed-history period.

2.2.4.5 Daily Average URL Referral Count Result Set

The **Daily Average URL Referral Count** result set returns one row for each **HTTP referer** that referred hits to a specified site collection during a usage detailed-history period. The rows in the result set MUST be sorted in descending order based on the **AvgDailyReferrals** field.

The T-SQL syntax for the result set is as follows:

```
    Referrer              nvarchar(260),
    AvgDailyReferrals     float;
```

Referrer: The URL of an HTTP referrer that referred hits to the specified site collection. The value of this field cannot exceed the maximum character limit of 260 characters. This MUST NOT be NULL.

AvgDailyReferrals: The average daily referral count of the HTTP referrer that is specified by the **Referrer** field. This value is calculated by dividing the total number of referrals by the number of days in the usage detailed-history period.

2.2.4.6 Hits By Day Result Set

The **Hits By Day** result set returns one row for each day that usage information was collected during a usage detailed-history period. Each row contains the number of hits that were received during a specified day. The rows in the result set MUST be sorted in ascending order based on the **FullDate** field.

The T-SQL syntax for the result set is as follows:

```
    FullDate    datetime,
    Hits        int;
```

FullDate: A timestamp that is in datetime format and specifies the day when the hits were received. This MUST NOT be NULL. The time component of the datetime value MUST be ignored.

Hits: The number of hits that were received on the day specified by the **FullDate** field. This MUST NOT be NULL.

2.2.4.7 Clickthroughs By Resource Result Set

The **Clickthroughs By Resource** result set returns one row for each URL for which usage information was collected during a usage detailed-history period. Each row contains the number of clickthroughs that were received by the document at a specified URL. The rows in the result set MUST be sorted in descending order based on the **Clickthroughs** field.

The T-SQL syntax for the result set is as follows:

```
        TargetFullUrl    nvarchar(260),
        Clickthroughs    bigint;
```

TargetFullUrl: The URL of the document. The value of this field cannot exceed the maximum character limit of 260 characters. This MUST NOT be NULL.

Clickthroughs: The total number of clickthroughs for the document at the URL specified by the **TargetFullUrl** field.

2.2.4.8 Unique Users By Day Result Set

The **Unique Users By Day** result set returns one row for each day during a usage detailed-history period. Each row contains the number of unique users who accessed a site collection, Web site (2), or subsite, depending on the stored procedure that is called, during each day of the usage detailed-history period. A unique user is defined as a user who logs into the system by using a distinct user name. The rows in the result set MUST be sorted in ascending order based on the **FullDate** field.

The T-SQL syntax for the result set is as follows:

```
        FullDate       datetime,
        UniqueUsers    int;
```

FullDate: A timestamp that is in datetime format and specifies the day when the hits were received. This MUST NOT be NULL. The time component of the datetime value MUST be ignored.

UniqueUsers: The number of unique users who accessed a site collection, site, or subsite on the day specified by the **FullDate** field.

2.2.4.9 Historical Unique Users By Day Result Set

The **Historical Unique Users By Day** result set returns one row for each day during a usage summary-history period. Each row contains the number of unique users who accessed a site collection, Web site (2), or subsite, depending on the stored procedure that is called, on each day of the usage summary-history period. A unique user is defined as a user who logs into the system by using a distinct user name. The rows in the result set MUST be sorted in ascending order based on the **FullDate** field.

The T-SQL syntax for the result set is as follows:

```
        FullDate       datetime,
        UniqueUsers    int;
```

FullDate: A timestamp that is in datetime format and specifies the day when the hits were received. This MUST NOT be NULL. The time component of the datetime value MUST be ignored.

UniqueUsers: The number of unique users who accessed a site collection, site, or subsite on the day specified by the **FullDate** field. This MUST NOT be NULL.

2.2.5 Tables and Views

2.2.5.1 ANLShadowHit

The **ANLShadowHit** table stores the day when each hit occurred. The time zone is not specified. The **ANLShadowHit** table also stores references to additional information in the **ANLShadowResource**, **ANLShadowUser**, and **ANLShadowWeb** tables.

The T-SQL syntax for the table is as follows:

```
TABLE ANLShadowHit (
    DayId                   int,
    ShadowResourceId        int,
    ResourceId              int,
    ShadowUserId            int,
    UserId                  int,
    ReferrerResourceId      int,
    ShadowReferrerResourceId  int
);
```

DayId: The identifier of the day that the hit occurred. This value MUST NOT be negative. This value is zero-based and represents the number of calendar days after January 1, 2000. For example, January 1, 2000 is represented by the value "0" (zero). January 2, 2000 is represented by the value "1", and so forth.

ShadowResourceId: A **shadow resource identifier** that MUST be present in the **ANLShadowResource** table. This MUST NOT be NULL.

ResourceId: This MUST be set to NULL and ignored.

ShadowUserId: A **shadow user identifier** that MUST be present in the **ANLShadowUser** table. This MUST NOT be NULL.

UserId: This MUST be set to NULL and ignored.

ReferrerResourceId: This MUST be set to NULL and ignored.

ShadowReferrerResourceId: The shadow resource identifier that is in the **ANLShadowResource** table, or NULL if no shadow resource identifier is present.

2.2.5.2 ANLShadowResource

The **ANLShadowResource** table stores information about a document that received a hit or the HTTP referrer that referred a user to the document, and the identifier of the site collection or Web site (2) that contains the document.

The T-SQL syntax for the table is as follows:

```
TABLE ANLShadowResource (
    ResourceId    int,
    WebAppGuid    uniqueidentifier,
    SiteGuid      uniqueidentifier,
    WebGuid       uniqueidentifier,
    DocName       nvarchar(260),
```

```
    FullUrl      nvarchar(260),
    HostDns      nvarchar(64)
);
```

ResourceId: The unique shadow resource identifier. This MUST NOT be NULL.

WebAppGuid: The **Web application identifier** of the document, or "{0000000-0000-0000-0000-000000000000}" if the value of the **FullUrl** field does not refer to a document. This MUST NOT be NULL.

SiteGuid: The **site collection identifier** of the document, or "{0000000-0000-0000-0000-000000000000}" if the value of the **FullUrl** field does not refer to a document. This MUST NOT be NULL.

WebGuid: The **site identifier** of the document, or "{0000000-0000-0000-0000-000000000000}" if the value of the **FullUrl** field does not refer to a document. This MUST NOT be NULL.

DocName: The store-relative form of the URL for the document.

FullUrl: The URI of the document or the HTTP referer that refers hits to a document.

HostDns: A unique host header value that refers hits to a document.

2.2.5.3   ANLShadowUser

The **ANLShadowUser** table stores information about a user.

The T-SQL syntax for the table is as follows:

```
TABLE ANLShadowUser (
    UserId      int,
    UserName    nvarchar(50)
);
```

UserId: The unique shadow user identifier. This MUST NOT be NULL.

UserName: The user name of a user. This MUST NOT be NULL.

2.2.5.4   ANLShadowWeb

The **ANLShadowWeb** table stores the URI of a Web site (2) or subsite.

The T-SQL syntax for the table is as follows:

```
TABLE ANLShadowWeb (
    WebGuid     uniqueidentifier,
    WebUrl      nvarchar(260)
);
```

WebGuid: The site identifier of the Web site (2) or subsite. This MUST NOT be NULL.

WebUrl: The URI of the Web site (2) or subsite. This MUST NOT be NULL.

2.2.6   XML Structures

No common XML Structures are defined in this protocol.

3   Protocol Details

3.1   SQL Usage Analytics Server Details

3.1.1   Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

This protocol provides data for usage analysis reports about a Web site (2). The protocol server maintains the data in a **usage shadow data store** and a **usage master data store**. The usage shadow data store is used as a staging area, and a protocol client can add data to it or modify the data that it contains. Data is typically added by using bulk insert statements, as specified in [MS-TDS]. The protocol server transfers the data from the usage shadow data store to the usage master data store. The data can then be used for reporting. The protocol server maintains this data for at least the duration of usage summary-history period to ensure that the correct result data is returned.

The transfer of data to the usage master data store begins when the protocol server receives a call from a protocol client to the **proc_ANL_ProcessShadowTables** stored procedure (section 3.1.5.5). This can occur multiple times per day. The protocol server uses the following process to transfer the data:

1. All user information from the usage shadow data store is moved to the usage master data store. Duplicate data is deleted.

2. All site collection, site, and subsite information from the usage shadow data store is moved to the usage master data store. Duplicate data is deleted.

3. All document information from the usage shadow data store is moved to the usage master data store. Duplicate data is deleted.

4. All hit information from the usage shadow data store is moved to the usage master data store. Duplicate data is deleted.

5. Hit data in the usage master data store that is older than the usage detailed-history period is summarized and stored in the usage master data store.

6. Summary data in the usage master data store that is older than the usage summary-history period is deleted from the usage master data store.

7. Hit data in the usage master data store that is older than the usage detailed-history period is deleted from the usage master data store.

The data-processing aspects of this protocol require obtaining exclusive locks for the usage shadow data store and usage master data store before calling any processing procedures. For example, to ensure predictable and reliable behavior, the protocol client calls the **proc_ANL_LockShadowTables** (section 3.1.5.4) and the **proc_ANL_LockMasterTablesForWrite** (section 3.1.5.3) stored procedures before calling the **proc_ANL_ProcessShadowTables** stored procedure (section 3.1.5.5). This call sequence occurs within a single **transaction (2)** to ensure that the locks are released properly when the transaction (2) is committed or rolled back.

Most of the stored procedures that are implemented by this protocol are used to perform usage analysis reporting. This includes the forty stored procedures whose names begin with "proc_ANL_Site" and "proc_ANL_Web". The following list describes the naming conventions that are used for this type of stored procedure:

- **Site or Web**. These terms indicate the scope of the stored procedure. A "Site" stored procedure reports on a Web site (2) and a "Web" stored procedure reports on a site collection, including home-page statistics for the site collection. Examples include the **proc_ANL_SiteAllPages** and **proc_ANL_WebAllPages** stored procedures.

- **Top or All**. These terms indicate the maximum number of rows that the result set can return. A "Top" stored procedure returns as many as five rows and an "All" stored procedure returns as many as 300 rows, although the term "All" is often omitted. Examples include the **proc_ANL_WebTopPages** and **proc_ANL_WebAllPages** stored procedures.

- **Historical or nonhistorical**. These terms indicate the reporting period. A "Historical" stored procedure reports on data in a usage summary-history period of 365 days. All other reporting stored procedures report on a usage detailed-history period of thirty days. Examples include the **proc_ANL_WebHistoricalHitsByDay** and **proc_ANL_WebHitsByDay** stored procedures.

- **ByDay and ByHits**. A "ByDay" stored procedure returns one row for each day in the reporting period. If it is also a "Historical" stored procedure, the reporting period is 365 days; otherwise, it is thirty days. A stored procedure with the term "ByHits" in its name returns the top five records for a "Top" stored procedure or the top 300 records for an "All" stored procedure. Examples include the **proc_ANL_SiteHitsByDay** and **proc_ANL_SiteReferrersByHits** stored procedures.

3.1.2 Timers

An execution timeout timer on the protocol server governs the execution time for requests from a protocol client. The amount of time is specified by a timeout value that is configured on the protocol server for all connections.

3.1.3 Initialization

A connection that uses the underlying protocol layers, as specified in Relationship to Other Protocols (section 1.4), MUST be established before using this protocol, as specified in [MS-TDS].

3.1.4 Higher-Layer Triggered Events

None.

3.1.5 Message Processing Events and Sequencing Rules

All of the stored procedures that are defined in this section are located on the protocol server, which is a **back-end database server**. The T-SQL syntax for each stored procedure and result set, and the variables that they contain, is as specified in [MSDN-TSQL-Ref]. In the T-SQL syntax, the variable name is followed by the type of variable, which can optionally have a length value in brackets and can optionally have a default value indicated by an equal sign (=) followed by the default value. For protocol examples that use the T-SQL syntax, see Protocol Examples (section 4).

3.1.5.1 proc_ANL_GetNextShadowResourceId

The **proc_ANL_GetNextShadowResourceId** stored procedure is called to retrieve a shadow resource identifier that is one greater than the maximum shadow resource identifier that is being

used in the usage shadow data store. The calling process MUST successfully lock the shadow tables by calling the **proc_ANL_LockShadowTables** stored procedure (section 3.1.5.4) before calling this stored procedure.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_GetNextShadowResourceId (
    @nextId          int OUTPUT
);
```

@nextId: The next shadow resource identifier to use. When it is returned from this stored procedure, the value MUST be set to a shadow resource identifier that is one greater than the maximum shadow resource identifier that is present in the system. This value MUST be "1" if there are zero rows in the **ANLShadowResource** table. An arithmetic overflow exception is thrown when this value exceeds 2^31-1.

Return Code Values: An integer that the protocol client MUST ignore.

Result Sets: This MUST NOT return any result sets.

3.1.5.2   proc_ANL_GetNextShadowUserId

The **proc_ANL_GetNextShadowUserId** stored procedure is called to retrieve a shadow user identifier that is one greater than the maximum shadow user identifier that is being used in the usage shadow data store.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_GetNextShadowUserId (
    @nextId          int OUTPUT
);
```

@nextId: The next shadow user identifier to use. When it is returned from this stored procedure, the value MUST be set to a shadow user identifier that is one greater than the maximum shadow user identifier that is present in the system. This value MUST be "1" if there are zero rows in the **ANLShadowUser** table. An arithmetic overflow exception is thrown when this value exceeds 2^31-1.

Return Code Values: An integer that the protocol client MUST ignore.

Result Sets: This MUST NOT return any result sets.

3.1.5.3   proc_ANL_LockMasterTablesForWrite

The **proc_ANL_LockMasterTablesForWrite** stored procedure is called to obtain an exclusive, read/write lock on the data in the usage master data store. The lock persists for the duration of the current transaction (2). The stored procedure waits indefinitely for the lock.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_LockMasterTablesForWrite ();
```

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
-1	The lock was not acquired.
0	The lock was acquired successfully.

Result Sets: This MUST NOT return any result sets.

3.1.5.4   proc_ANL_LockShadowTables

The **proc_ANL_LockShadowTables** stored procedure is called to obtain an exclusive, read/write lock on the data in the usage shadow data store. The lock persists for the duration of the current transaction (2).

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_LockShadowTables(
    @wait            tinyint
);
```

@wait: An integer that indicates whether to wait to obtain the lock. This parameter MUST be set to one of the values listed in the following table.

Value	Description
0	Don't wait for the lock.
1	Queue up and wait indefinitely for the lock.

Return Code Values: An integer that MUST be one of the values in the following table.

Value	Description
-1	The lock was not acquired.
0	The lock was acquired successfully.

Result Sets: This MUST NOT return any result sets.

3.1.5.5   proc_ANL_ProcessShadowTables

The **proc_ANL_ProcessShadowTables** stored procedure is called to process the data in the usage shadow data store and to insert that data into the usage master data store. This stored procedure can be called multiple times each day. Before calling this procedure, the protocol client MUST obtain shadow and master locks through calls to the **proc_ANL_LockShadowTables** stored procedure (section 3.1.5.4) and the **proc_ANL_LockMasterTablesForWrite** stored procedure (section 3.1.5.3). In addition, sequential calls to the **proc_ANL_LockShadowTables**, **proc_ANL_LockMasterTablesForWrite**, and **proc_ANL_ProcessShadowTables** stored procedures MUST occur within a single transaction (2) to ensure that the locks are released properly when the transaction (2) is committed or rolled back. For more information about this process, see Abstract Data Model (section 3.1.1).

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_ProcessShadowTables (
    @todayDayId            int
);
```

@todayDayId: The identifier of the day for which to process the data. This value MUST be an integer that represents the number of calendar days after January 1, 2000. For example, January 2, 2000 is represented by the value "1".

Return Code Values: The protocol client MUST ignore the return code.

Result Sets: This MUST return eleven instances of the following result set. The protocol client MUST ignore any result sets that are returned by this stored procedure.

3.1.5.5.1 Process Shadow Tables Result Set

The **Process Shadow Tables** result set MUST contain one row.

The T-SQL syntax for the result set is as follows:

```
    column1            int,
```

column1: An integer that the protocol client MUST ignore. This MUST NOT be NULL.

3.1.5.6 proc_ANL_SiteAllPages

The **proc_ANL_SiteAllPages** stored procedure is called to retrieve the top 300 documents in a site collection, based on the average hits per day in a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteAllPages(
    @siteGuid            uniqueidentifier
);
```

@siteGuid: The **site collection identifier** of the site collection that contains the documents.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.6.1   Average Hits by Resource Result Set

The **Average Hits by Resource** result set MUST contain either 300 rows or one row for each document that received a hit during a usage detailed-history period, whichever row count is smaller. For more information, see Average Hits by Resource Result Set (section 2.2.4.1).

3.1.5.7   proc_ANL_SiteClickthroughPagesByHits

The **proc_ANL_SiteClickthroughPagesByHits** stored procedure is called to retrieve the top 300 documents in a site collection, based on the total number of clickthroughs that were received during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteClickthroughPagesByHits(
    @siteGuid            uniqueidentifier,
);
```

@siteGuid: The site collection identifier of the site collection that contains the documents.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.7.1   Clickthroughs By Resource Result Set

The **Clickthroughs By Resource** result set MUST contain up to 300 rows, one row for each document that is in a specified site collection and received a clickthrough during a usage detailed-history period. For more information, see Clickthroughs By Resource Result Set (section 2.2.4.7).

3.1.5.8   proc_ANL_SiteClickthroughWebsByHits

The **proc_ANL_SiteClickthroughWebsByHits** stored procedure is called to retrieve the top 300 URIs in a site collection, based on the total number of clickthroughs for each URI during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteClickthroughWebsByHits(
    @siteGuid            uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that contains the URIs.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.8.1 Clickthroughs Result Set

The **Clickthroughs** result set MUST return up to 300 rows, one row for each URI that is in a specified site collection and received a hit during a usage detailed-history period. For more information, see Clickthroughs Result Set (section 2.2.4.3).

3.1.5.9   proc_ANL_SiteDnsReferrersByHits

The **proc_ANL_SiteDnsReferrersByHits** stored procedure is called to retrieve the top 300 host headers that referred hits to a site collection, based on the average referral count per day in a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteDnsReferrersByHits(
    @siteGuid            uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.9.1   Daily Average DNS Referral Count Result Set

For information about the **Daily Average DNS Referral Count** result set, see section 2.2.4.4.

3.1.5.10   proc_ANL_SiteDnsTopReferrersByHits

The **proc_ANL_SiteDnsTopReferrersByHits** stored procedure is called to retrieve the top five host headers that referred hits to a site collection, based on the average referral count per day in a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteDnsTopReferrersByHits(
    @siteGuid            uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.10.1 Daily Average DNS Referral Count Result Set

For information about the **Daily Average DNS Referral Count** result set, see section 2.2.4.4.

3.1.5.11 proc_ANL_SiteHistoricalHitsByDay

The **proc_ANL_SiteHistoricalHitsByDay** stored procedure is called to retrieve the top 365 documents in a site collection, based on the number of hits received during a usage summary-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteHistoricalHitsByDay (
    @siteGuid            uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.11.1 Hits By Day Result Set

For information about the **Hits By Day** result set, see section 2.2.4.6.

3.1.5.12   proc_ANL_SiteHistoricalUniqueUsersByDay

The **proc_ANL_SiteHistoricalUniqueUsersByDay** stored procedure is called to retrieve the top 365 documents, based on the number of unique users who accessed a site collection during a usage summary-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteHistoricalUniqueUsersByDay (
    @siteGuid            uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that was accessed by the users.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.12.1   Historical Unique Users By Day Result Set

For information about the **Historical Unique Users By Day** result set, see section 2.2.4.9.

3.1.5.13   proc_ANL_SiteHitsByDay

The **proc_ANL_SiteHitsByDay** stored procedure is called to retrieve the top 30 documents in a site collection, based on the number of hits received during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteHitsByDay (
    @siteGuid            uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

Release: July 16, 2012

3.1.5.13.1 Big Hits by Day Result Set

For information about the **Big Hits by Day** result set, see section 2.2.4.2.

3.1.5.14 proc_ANL_SiteReferrersByHits

The **proc_ANL_SiteReferrersByHits** stored procedure is called to retrieve the top 300 HTTP referers that referred hits to a site collection, based on the average referral count per day in a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteReferrersByHits(
    @siteGuid           uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.14.1 Daily Average URL Referral Count Result Set

For information about the **Daily Average URL Referral Count** result set, see section 2.2.4.5.

3.1.5.15 proc_ANL_SiteSummary

The **proc_ANL_SiteSummary** stored procedure is called to retrieve summarized information about usage data and search requests for a site collection during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteSummary(
    @siteGuid           uniqueidentifier,
    @todayDayId         int,
);
```

@siteGuid: The site collection identifier of the site collection for which to obtain the information.

@todayDayId: The identifier of the day to use as a reference for the resulting data. This value MUST be an integer that represents the number of calendar days after January 1, 2000. For example, January 2, 2000 is represented by the value "1".

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.15.1 Site Summary Result Set

The **Site Summary** result set MUST contain six rows. The rows in the result set MUST be sorted in ascending order based on the **MeasureIndex** field. For MeasureIndex values 0 to 3, the term "yesterday" is used in the returned summary metrics it is defined as the day before the value of @*todayDayId*, as determined by the system time of the protocol server. For MeasureIndex values 4, the term "yesterday" refers to one day before the current date as defined by the current database server timestamp.

The T-SQL syntax for the result set is as follows:

```
MeasureIndex            int,
Measure                 varchar(max),
Value                   float;
```

MeasureIndex: The index of the row. This value MUST be "0" (zero), "1", "2", "3", "4", or "5" and it MUST be distinct.

Measure: The descriptive name of the measure that is contained in the row. This value MUST be one of the values listed in the following table, and it MUST be equal to the value that corresponds to the **MeasureIndex** value from the result set row. The field size MUST be fewer than or equal to $2^{31} - 1$ bytes of data or $2^{30} - 1$ bytes for Unicode.

MeasureIndex	Measure
0	"Average hits over past 30 days"
1	"Distinct users over past 30 days"
2	"Distinct users yesterday"
3	"Requests yesterday"
4	"Queries yesterday"
5	"Queries over past 30 days"

Value: The value of the measure that is contained in the row. This value MUST be one of the values listed in the following table, and it MUST be equal to the value that corresponds to value of the **MeasureIndex** from the result set row.

MeasureIndex	Value
0	The total number of hits for the site collection divided by 30 or the number of days in the usage detailed-history period, whichever is smaller. The value is "0" (zero) if there

MeasureIndex	Value
	are no days in the usage detailed-history period.
1	The number of unique users who accessed the site collection during the usage detailed-history period or the past 30 days, whichever is smaller.
2	The number of unique users who accessed the site collection yesterday, which is defined as the day before the value of **@todayDayId**. Time references are based on the system time of the protocol server.
3	The number of hits that were received by the site collection yesterday, which is defined as the day before the value of **@todayDayId**. Time references are based on the system time of the protocol server.
4	The number of search requests that were made yesterday, which is defined as the day before the current date, as defined by the current database server timestamp, by all users for the site collection. Time references are based on the system time of the protocol server.
5	The number of search requests that were made by all users for the site collection for the 30 days prior to the current date, as defined by the current database server timestamp,

3.1.5.16   proc_ANL_SiteTopClickthroughPagesByHits

The **proc_ANL_SiteTopClickthroughPagesByHits** stored procedure is called to retrieve the top five URLs in a site collection, based on the total number of clickthroughs received during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteTopClickthroughPagesByHits(
    @siteGuid               uniqueidentifier,
);
```

@siteGuid: The site collection identifier of the site collection that contains the documents.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.16.1   Clickthroughs By Resource Result Set

The **Clickthroughs By Resource** result set MUST contain five rows or one row for each URL that received a clickthrough in the specified site collection, whichever is smaller. For more information, see Clickthroughs By Resource Result Set (section 2.2.4.7).

3.1.5.17   proc_ANL_SiteTopClickthroughWebsByHits

The **proc_ANL_SiteTopClickthroughWebsByHits** stored procedure is called to retrieve the top five URIs in a site collection, based on the total number of hits for each URI by a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteTopClickthroughWebsByHits(
    @siteGuid           uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that contains the URIs.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.17.1   Clickthroughs Result Set

The **Clickthroughs** result set MUST return up to five rows, one row for each URI in the site collection that received a hit. For more information, see Clickthroughs Result Set (section 2.2.4.3).

3.1.5.18   proc_ANL_SiteTopPages

The **proc_ANL_SiteTopPages** stored procedure is called to retrieve the top five documents in a site collection, based on the average number of hits per day during the usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteTopPages(
    @siteGuid           uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that contains the documents.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.18.1   Average Hits by Resource Result Set

The **Average Hits by Resource** result set MUST contain five rows or one row for each document that received a hit during a usage detailed-history period, whichever is smaller. For more information, see Average Hits by Resource Result Set (section 2.2.4.1).

3.1.5.19   proc_ANL_SiteTopReferrersByHits

The **proc_ANL_SiteTopReferrersByHits** stored procedure is called to retrieve the top five HTTP referrers that referred hits to a site collection, based on the average referral count per day in a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteTopReferrersByHits(
    @siteGuid             uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.19.1   Daily Average URL Referral Count Result Set

For information about the **Daily Average URL Referral Count** result set, see section 2.2.4.5.

3.1.5.20   proc_ANL_SiteTopUsersByHits

The **proc_ANL_SiteTopUsersByHits** stored procedure is called to retrieve the top five users of a site collection, based on the average number of hits per day during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteTopUsersByHits (
    @siteGuid             uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.20.1   Site Top Users Result Set

The **Site Top Users** result set MUST contain five rows or one row for each user who accessed a document, whichever is smaller. The rows in the result set MUST be sorted in descending order based on the **AvgHits** field.

The T-SQL syntax for the result set is as follows:

```
    UserName              nvarchar(50),
    AvgHits               float;
```

UserName: The user name of the user who accessed a document in the site collection. This MUST NOT be NULL.

AvgHits: The total number of hits divided by the number of days in the usage detailed-history period.

3.1.5.21   proc_ANL_SiteUniqueUsersByDay

The **proc_ANL_SiteUniqueUsersByDay** stored procedure is called to retrieve the top 30 documents, based on the number of unique users who accessed a site collection during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
  PROCEDURE proc_ANL_SiteUniqueUsersByDay (
      @siteGuid              uniqueidentifier
  );
```

@siteGuid: The site collection identifier of the site collection that was accessed.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.21.1   Unique Users By Day Result Set

For information about the **Unique Users By Day** result set, see section 2.2.4.8.

3.1.5.22   proc_ANL_SiteUsersByHits

The **proc_ANL_SiteUsersByHits** stored procedure is called to retrieve the top 300 unique users who accessed a document from a site collection during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_SiteUsersByHits (
    @siteGuid            uniqueidentifier
);
```

@siteGuid: The site collection identifier of the site collection that was accessed.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.22.1   Site Users Result Set

The **Site Users** result set MUST contain 300 rows or one row for each user who accessed a document within a site collection during a usage detailed-history period, whichever is smaller. Each row in the result set contains the average number of hits per day that was generated by each user. There MUST be only one row for each user who logs into the system by using a distinct user name. The rows in the result set MUST be sorted in descending order based on the **AvgHits** field.

The T-SQL syntax for the result set is as follows:

```
    UserName             nvarchar(50),
    AvgHits              float;
```

UserName: The user name of the user who accessed a document in the specified site collection. This MUST NOT be NULL.

AvgHits: The total number of hits divided by the number of days in the usage detailed-history period.

3.1.5.23   proc_ANL_TestMasterLockIsFree

The **proc_ANL_TestMasterLockIsFree** stored procedure is called to determine whether the data in the usage master data store can be read. This stored procedure attempts to obtain an exclusive read lock on the data store.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_TestMasterLockIsFree ();
```

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The data cannot be read. Another caller has already obtained an exclusive read lock on the data store.
1	The data can be read because an exclusive read lock was obtained.

Result Sets: This MUST NOT return any result sets.

3.1.5.24   proc_ANL_WebAllPages

The **proc_ANL_WebAllPages** stored procedure is called to retrieve the top 300 documents in a Web site (2) or subsite, based on the average hits per day in a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebAllPages(
    @webGuid            uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that contains the documents.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.24.1   Average Hits by Resource Result Set

The **Average Hits by Resource** result set MUST contain 300 rows or one row for each document that received a hit during a usage detailed-history period, whichever is smaller. For more information, see Average Hits by Resource Result Set (section 2.2.4.1).

3.1.5.25   proc_ANL_WebClickthroughPagesByHits

The **proc_ANL_WebClickthroughPagesByHits** stored procedure is called to retrieve the top 300 documents in a Web site (2) or subsite, based on the total number of clickthroughs received during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

Release: July 16, 2012

```
PROCEDURE proc_ANL_WebClickthroughPagesByHits(
    @webGuid            uniqueidentifier,
);
```

@webGuid: The site identifier of the site or subsite that contains the documents.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.25.1   Clickthroughs By Resource Result Set

The **Clickthroughs By Resource** result set MUST contain 300 rows or one row for each document that received a clickthrough in the specified Web site (2) or subsite, whichever is smaller. For more information, see Clickthroughs By Resource Result Set (section 2.2.4.7).

3.1.5.26   proc_ANL_WebClickthroughWebsByHits

The **proc_ANL_WebClickthroughWebsByHits** stored procedure is called to retrieve the top 300 URIs in a Web site (2) or subsite, based on the total number of hits for each URI.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebClickthroughWebsByHits(
    @webGuid            uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that contains the URIs.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.26.1   Clickthroughs Result Set

The **Clickthroughs** result set MUST contain 300 rows or one row for each of the top URIs in a Web site (2) or subsite, whichever is smaller. The rows in the result set MUST be sorted in descending

order based on the **Clickthroughs** field. For more information, see Clickthroughs Result Set (section 2.2.4.3).

3.1.5.27   proc_ANL_WebDnsReferrersByHits

The **proc_ANL_WebDnsReferrersByHits** stored procedure is called to retrieve the top 300 host headers that referred hits to a Web site (2) or subsite, based on the average referral count per day during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebDnsReferrersByHits(
    @webGuid           uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that received the hits**.**

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.27.1   Daily Average DNS Referral Count Result Set

For information about the **Daily Average DNS Referral Count** result set, see section 2.2.4.4.

3.1.5.28   proc_ANL_WebDnsTopReferrersByHits

The **proc_ANL_WebDnsTopReferrersByHits** stored procedure is called to retrieve the top five host headers that referred hits to a Web site (2) or subsite, based on the average referral count per day during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebDnsTopReferrersByHits(
    @webGuid           uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.

Value	Description
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.28.1   Daily Average DNS Referral Count Result Set

For information about the **Daily Average DNS Referral Count** result set, see section 2.2.4.4.

3.1.5.29   proc_ANL_WebHistoricalHitsByDay

The **proc_ANL_WebHistoricalHitsByDay** stored procedure is called to retrieve the top 365 documents in a Web site (2) or subsite, based on the number of hits received during a usage summary-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebHistoricalHitsByDay (
    @webGuid            uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.29.1   Hits By Day Result Set

For information about the **Hits By Day** result set, see section 2.2.4.6.

3.1.5.30   proc_ANL_WebHistoricalUniqueUsersByDay

The **proc_ANL_WebHistoricalUniqueUsersByDay** stored procedure is called to retrieve the top 365 documents, based on the number of unique users who accessed a site or subsite during a usage summary-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebHistoricalUniqueUsersByDay (
    @webGuid            uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that was accessed by the users.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.30.1 Historical Unique Users By Day Result Set

For information about the **Historical Unique Users By Day** result set, see section 2.2.4.9.

3.1.5.31 proc_ANL_WebHitsByDay

The **proc_ANL_WebHitsByDay** stored procedure is called to retrieve the top 30 documents, based on the number of hits that a Web site (2) or subsite received during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebHitsByDay (
    @webGuid            uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.31.1 Big Hits by Day Result Set

The **Big Hits by Day** result set MUST contain one row for each day in a usage detailed-history period. For more information, see Big Hits by Day Result Set (section 2.2.4.2).

3.1.5.32 proc_ANL_WebHomePageClickthroughPages

The **proc_ANL_WebHomePageClickthroughPages** stored procedure is called to retrieve the top 300 URLs in a Web site (2) or subsite, based on the average number of hits per day and by the number of unique users that performed a clickthrough during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebHomePageClickthroughPages(
    @webGuid            uniqueidentifier,
);
```

@webGuid: The site identifier of the site or subsite that contains the documents.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.32.1   Clickthrough Summary by Resource Result Set

The **Clickthrough Summary by Resource** result set MUST contain 300 rows or one row for each URL that received a hit by a user in the Web site (2) or subsite, whichever is smaller. Each row contains the average number of hits and the total number of unique users that performed a clickthrough during a usage detailed-history period. A unique user is defined as a user who logs into the system by using a distinct user name. The rows in the result set MUST be sorted in descending order first by the **AvgDailyClickthroughs** field and then by the **Clickthroughs** field.

The T-SQL syntax for the result set is as follows:

```
TargetFullUrl               nvarchar(260),
AvgDailyClickthroughs       float,
Clickthroughs               bigint;
```

TargetFullUrl: The URL of the document. This MUST NOT be NULL.

AvgDailyClickthroughs: The total number of clickthroughs divided by the number of days in the usage detailed-history period.

Clickthroughs: The total number of unique users that performed a clickthrough during the usage detailed-history period.

3.1.5.33   proc_ANL_WebHomePageHistoricalHitsByDay

The **proc_ANL_WebHomePageHistoricalHitsByDay** stored procedure is called to retrieve the top 365 documents, based on the number of hits that the home page for a Web site (2) or subsite received during a usage summary-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebHomePageHistoricalHitsByDay (
    @webGuid            uniqueidentifier
```

```
    );
```

@webGuid: The site identifier of the site or subsite that contains the home page.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.33.1   Hits By Day Result Set

For information about the **Hits By Day** result set, see section 2.2.4.6.

3.1.5.34   proc_ANL_WebHomePageHistoricalUniqueUsersByDay

The **proc_ANL_WebHomePageHistoricalUniqueUsersByDay** stored procedure is called to retrieve the top 365 documents, based on the number of unique users that accessed the home page of a Web site (2) or subsite during a usage summary-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebHomePageHistoricalUniqueUsersByDay (
    @webGuid           uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that contains the home page.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.34.1   Historical Unique Users By Day Result Set

For information about the **Historical Unique Users By Day** result set, see section 2.2.4.9.

3.1.5.35   proc_ANL_WebHomePageHitsByDay

The **proc_ANL_WebHomePageHitsByDay** stored procedure is called to retrieve the top 30 documents, based on the number of hits that the home page for a Web site (2) or subsite received during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebHomePageHitsByDay (
    @webGuid              uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that contains the home page.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.35.1   Big Hits by Day Result Set

For information about the **Big Hits by Day** result set, see section 2.2.4.2.

3.1.5.36   proc_ANL_WebHomePageReferringPages

The **proc_ANL_WebHomePageReferringPages** stored procedure is called to retrieve the average number of referrals per day and the average number of unique users per day for each URI that accessed the home page of a Web site (2). The averages are calculated only for the days when usage information was collected. The result set that is returned MUST contain the top 300 URIs with the highest average number of referrals per day. If there are fewer than 300 URIs, all of the URIs are returned.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebHomePageReferringPages (
    @webGuid              uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that contains the home page.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.

Value	Description
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.36.1 Referring Pages Result Set

The **Referring Pages** result set returns one row for each URI that referred hits to the home page of a Web site (2). Each row contains the average number of referrals per day and the average number of unique users per day for each URI. A unique user is defined as a user who logs into the system by using a distinct user name. The rows in the result set MUST be sorted in descending order first by the **AvgDailyReferrals** field and then by the **AvgDailyUniqueUsersReferred** field.

The T-SQL syntax for the result set is as follows:

```
    Referrer                     nvarchar(260),
    AvgDailyReferrals            float,
    AvgDailyUniqueUsersReferred  float;
```

Referrer: The URI that referred the hits.

AvgDailyReferrals: The average number of hits to the home page from the originating URI per day.

AvgDailyUniqueUsersReferred: The average number of unique users who accessed the home page from the originating URI per day.

3.1.5.37 proc_ANL_WebHomePageUniqueUsersByDay

The **proc_ANL_WebHomePageUniqueUsersByDay** stored procedure is called to retrieve the top 30 documents, based on the number of hits by a unique user that the home page of a Web site (2) or subsite received during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
  PROCEDURE proc_ANL_WebHomePageUniqueUsersByDay (
      @webGuid            uniqueidentifier
  );
```

@webGuid: The site identifier of the site or subsite that contains the home page.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.37.1   Unique Users By Day Result Set

For information about the **Unique Users By Day** result set, see section 2.2.4.8.

3.1.5.38   proc_ANL_WebReferrersByHits

The **proc_ANL_WebReferrersByHits** stored procedure is called to retrieve the top 300 HTTP referers that referred hits to a Web site (2) or subsite, based on the average referral count per day during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebReferrersByHits(
    @webGuid            uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.38.1   Daily Average URL Referral Count Result Set

For information about the **Daily Average URL Referral Count** result set, see section 2.2.4.5.

3.1.5.39   proc_ANL_WebSummary

The **proc_ANL_WebSummary** stored procedure is called to retrieve summarized information about usage data for a Web site (2) or subsite during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebSummary(
    @webGuid            uniqueidentifier,
    @todayDayId         int,
);
```

@webGuid: The site identifier of the site or subsite to which the usage data applies.

@todayDayId: The day to use as a reference for the resulting data. This value MUST be an integer that represents the number of calendar days after January 1, 2000. For example, January 2, 2000 is represented by the value "1".

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.39.1 Web Summary Result Set

The **Web Summary** result set MUST contain four rows. The rows in the result set MUST be sorted in ascending order based on the **MeasureIndex** field.

The T-SQL syntax for the result set is as follows:

```
MeasureIndex            int,
Measure                 varchar(max),
Value                   float;
```

MeasureIndex: The index of the row. This MUST be "0" (zero), "1", "2", or "3" and MUST be distinct.

Measure: The descriptive name of the measure that is contained in the row. This value MUST be one of the values listed in the following table, and it MUST be equal to the value that corresponds to the **MeasureIndex** value from the result set row. The field size MUST be fewer than or equal to $2^{31} - 1$ bytes of data or $2^{30} - 1$ bytes for Unicode.

MeasureIndex	Measure
0	"Average hits over past 30 days"
1	"Distinct users over past 30 days"
2	"Distinct users yesterday"
3	"Requests yesterday"

Value: The value of the measure that is contained in the row. This value MUST be one of the values listed in the following table, and it MUST be equal to the value that corresponds to the **MeasureIndex** value from the result set row.

MeasureIndex	Value
0	The total number of hits for the site or subsite divided by the number of days in the usage detailed-history period or "0" (zero) if there are no days in the usage detailed-history period.

MeasureIndex	Value
1	The number of unique users who accessed the site or subsite during the usage detailed-history period.
2	The number of unique users who accessed the site or subsite the day before the value of **@todayDayId**.
3	The number of hits that were received by the site or subsite the day before the value of **@todayDayId**.

3.1.5.40   proc_ANL_WebTopClickthroughPagesByHits

The **proc_ANL_WebTopClickthroughPagesByHits** stored procedure is called to retrieve the top five documents in a Web site (2) or subsite, based on the total number of clickthroughs during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebTopClickthroughPagesByHits(
    @webGuid          uniqueidentifier,
);
```

@webGuid: The site identifier of the site or subsite that contains the documents.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.40.1   Clickthroughs By Resource Result Set

The **Clickthroughs By Resource** result set MUST contain five rows or one row for each document that received a clickthrough in the specified site or subsite, whichever is smaller. For more information, see Clickthroughs By Resource Result Set (section 2.2.4.7).

3.1.5.41   proc_ANL_WebTopClickthroughWebsByHits

The **proc_ANL_WebTopClickthroughWebsByHits** stored procedure is called to retrieve the top five URIs in a Web site (2) or subsite, based on the total number of hits for each URI during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebTopClickthroughWebsByHits(
    @webGuid          uniqueidentifier
```

```
   );
```

@webGuid: The site identifier of the site or subsite that contains the URIs.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.41.1   Clickthroughs Result Set

The **Clickthroughs** result set MUST contain a maximum of five rows for the top five URIs in a Web site (2) or subsite, based on the total number of hits for each URI. For more information, see Clickthroughs Result Set (section 2.2.4.3).

3.1.5.42   proc_ANL_WebTopPages

The **proc_ANL_WebTopPages** stored procedure is called to retrieve the top five documents in a Web site (2) or subsite, based on the average number of hits per day during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
  PROCEDURE proc_ANL_WebTopPages(
      @webGuid              uniqueidentifier
  );
```

@webGuid: The site identifier of the site or subsite that contains the documents.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.42.1   Average Hits by Resource Result Set

The **Average Hits by Resource** result set MUST contain five rows or one row for each document that received a hit during a usage detailed-history period, whichever is smaller. For more information, see Average Hits by Resource Result Set (section 2.2.4.1).

3.1.5.43   proc_ANL_WebTopReferrersByHits

The **proc_ANL_WebTopReferrersByHits** stored procedure is called to retrieve the top five HTTP referers that referred hits to a Web site (2) or subsite, based on the average referral count per day during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebTopReferrersByHits(
    @webGuid              uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.43.1   Daily Average URL Referral Count Result Set

For information about the **Daily Average URL Referral Count** result set, see section 2.2.4.5.

3.1.5.44   proc_ANL_WebTopUsersByHits

The **proc_ANL_WebTopUsersByHits** stored procedure is called to retrieve the top five users of a Web site (2) or subsite, based on the average number of hits per day during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebTopUsersByHits (
    @webGuid              uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that received the hits.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.44.1 Web Top Users Result Set

The **Web Top Users** result set MUST contain five rows or one row for each user who accessed a document on a Web site (2) or subsite, whichever is smaller. The rows in the result set MUST be sorted in descending order based on the **AvgHits** field.

The T-SQL syntax for the result set is as follows:

```
        UserName              nvarchar(50),
        AvgHits               float;
```

UserName: The user name of the user who accessed a document on the site or subsite. This MUST NOT be NULL.

AvgHits: The total number of hits divided by the number of days in the usage detailed-history period.

3.1.5.45 proc_ANL_WebUniqueUsersByDay

The **proc_ANL_WebUniqueUsersByDay** stored procedure is called to retrieve the top 30 documents, based on the number of unique users who accessed a Web site (2) or subsite during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
  PROCEDURE proc_ANL_WebUniqueUsersByDay (
      @webGuid              uniqueidentifier
  );
```

@webGuid: The site identifier of the site or subsite that was accessed by the users.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.45.1 Unique Users By Day Result Set

For information about the **Unique Users By Day** result set, see section 2.2.4.8.

Release: July 16, 2012

3.1.5.46   proc_ANL_WebUsersByHits

The **proc_ANL_WebUsersByHits** stored procedure is called to retrieve the top 300 users who accessed a document from a Web site (2) or subsite during a usage detailed-history period.

The T-SQL syntax for the stored procedure is as follows:

```
PROCEDURE proc_ANL_WebUsersByHits (
    @webGuid            uniqueidentifier
);
```

@webGuid: The site identifier of the site or subsite that was accessed by the users.

Return Code Values: An integer that MUST be one of the values listed in the following table.

Value	Description
0	The result set is not available at this time. The data store is currently locked by another process.
1	Execution was successful.

Result Sets: This MUST return the following result set if the return code is "1". This MUST NOT return any result sets if execution is not successful.

3.1.5.46.1   Web Users Result Set

The **Web Users** result set MUST contain 300 rows or one row for each user who accessed a document on a Web site (2) or subsite, whichever is smaller. The rows in the result set MUST be sorted in descending order based on the **AvgHits** field.

The T-SQL syntax for the result set is as follows:

```
    UserName            nvarchar(50),
    AvgHits             float;
```

UserName: The user name of the user who has accessed a document on the specified site or subsite. This MUST NOT be NULL.

AvgHits: The total number of hits divided by the number of days in the usage detailed-history period.

3.1.6   Timer Events

None.

3.1.7   Other Local Events

None.

4   Protocol Examples

4.1   Importing Usage Data

This example demonstrates requests that can be made by a protocol client to add usage data to a usage shadow data store. The usage data was collected on February 7, 2008 for the **Web application (1)** that has the Web application identifier "{9AB2CE9E-F2D7-4459-92EB-D75BF83F6C5A}".

1. The protocol client creates a new transaction (2).

   After creating the transaction (2), the protocol client sets locks to prevent other protocol clients from accessing the data during processing.

2. The protocol client calls the **proc_ANL_LockShadowTables** stored procedure (section 3.1.5.4), waiting indefinitely for a lock, by using the following T-SQL syntax:

   ```
   exec proc_ANL_LockShadowTables @wait=1
   ```

3. The return code is "0" (zero). Therefore, the protocol client proceeds.

4. The protocol client calls the **proc_ANL_GetNextShadowResourceId** stored procedure (section 3.1.5.1) by using the following T-SQL syntax:

   ```
   exec proc_ANL_GetNextShadowResourceId @nextId OUTPUT
   ```

5. The protocol client calls the **proc_ANL_GetNextShadowUserId** stored procedure (section 3.1.5.2) by using the following T-SQL syntax:

   ```
   exec proc_ANL_GetNextShadowUserId @nextId OUTPUT
   ```

6. The protocol client calculates that February 7, 2008 is 2,959 days after January 1, 2000. The protocol client bulk inserts the collected hit data into the **ANLShadowHit** table (section 2.2.5.1). It uses "2959" as the value of the **DayId** column, "{9AB2CE9E-F2D7-4459-92EB-D75BF83F6C5A}" as the value of the **WebAppGuid** column, and sequential integers starting with the values of "*@nextId*" that were obtained in step 4 for the **ShadowResourceId** column and step 5 for the **ShadowUserId** column.

7. The protocol client bulk inserts the collected hit data into the **ANLShadowResource** table (section 2.2.5.2). It uses the **ResourceId** column to define relationships with the **ShadowResourceId** column in the **ANLShadowHit** table.

8. The protocol client bulk inserts the collected hit data into the **ANLShadowUser** table (section 2.2.5.3). It uses the **UserId** column to define relationships with the **ShadowUserId** column in the **ANLShadowHit** table.

9. The protocol client bulk inserts the collected hit data into the **ANLShadowWeb** table (section 2.2.5.4). It uses the **WebGuid** column to define relationships with the **WebGuid** column in the **ANLShadowHit** table.

10. The protocol client commits the transaction (2).

All of the interaction between the protocol client and the protocol occurs within the same transaction (2). Therefore, any locks that are obtained during this interaction are released automatically when the transaction (2) is committed or rolled back.

4.2 Processing Usage Data

This example describes requests that can be made by a log-processing, protocol client to process all of the data for February 7, 2008. The protocol client uses this protocol to initiate data processing on the protocol server, but it first sets locks to prevent other protocol clients from accessing the data during processing. The following diagram shows the data flow for this example.
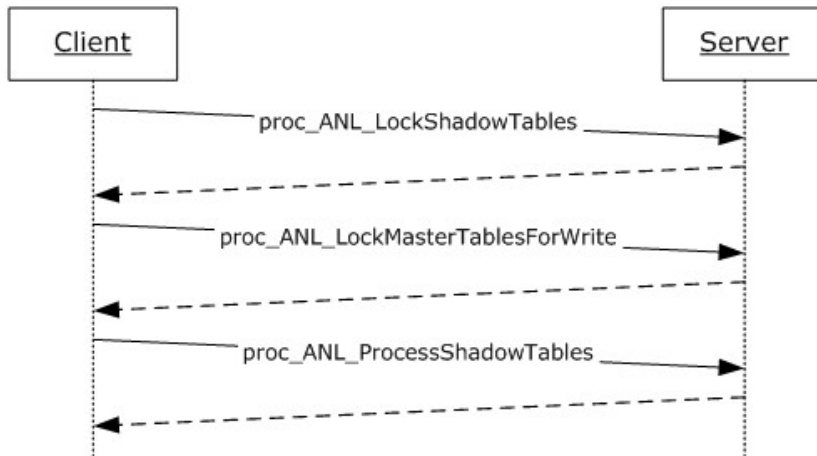
Figure 3: Data flow for processing usage data

1. The protocol client calls the **proc_ANL_LockShadowTables** stored procedure (section 3.1.5.4), without waiting for a lock, by using the following T-SQL syntax:

```
exec proc_ANL_LockShadowTables @wait=0
```

2. The return code is "0" (zero). Therefore, the protocol client proceeds.

3. The protocol client calls the **proc_ANL_LockMasterTablesForWrite** stored procedure (section 3.1.5.3) by using the following T-SQL syntax:

```
exec proc_ANL_LockMasterTablesForWrite
```

4. The return code is "0" (zero). Therefore, the protocol client proceeds.

5. The protocol client calculates that February 7, 2008 is 2,959 days after January 1, 2000. The protocol client calls the **proc_ANL_ProcessShadowTables** stored procedure (section 3.1.5.5) and specifies the date value by using the following T-SQL syntax:

```
exec proc_ANL_ProcessShadowTables @todayDayId=2959
```

6. The protocol client ignores the return code**.**

All of the interaction between the protocol client and the protocol occurs within the same transaction (2). Therefore, the locks that are obtained in steps 3 and 4 are in effect for the call in steps 5 and 6. These locks are released automatically when the transaction (2) is committed or rolled back.

4.3   Reporting Data Is Not Available

This example describes requests that can be made by a protocol client for usage reporting. The protocol client reports usage summary data for a specific site as of February 6, 2008. However, the data is unavailable for reporting because a third-party protocol client has called the **proc_ANL_LockMasterTablesForWrite** stored procedure (section 3.1.5.3). The following diagram shows the data flow for this example.
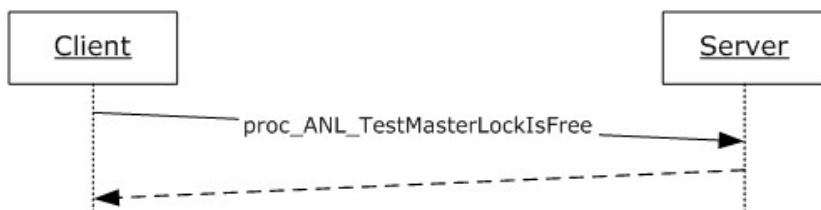
Figure 4: Data flow if reporting data is not available

1. The protocol client calls the **proc_ANL_TestMasterLockIsFree** stored procedure (section 3.1.5.23).

2. The return code is "0" (zero). Therefore, the protocol client displays a message indicating that the reporting data is not available.

The protocol client does not proceed to call other stored procedures of this protocol.

4.4   Reporting Site Usage Summary

This example describes requests that can be made by a protocol client for usage reporting. The protocol client reports usage summary data for the site that has the site identifier "95688941-D67C-4B88-B9EF-F31576C62E53" as of February 6, 2008. The following diagram shows the data flow for this example.
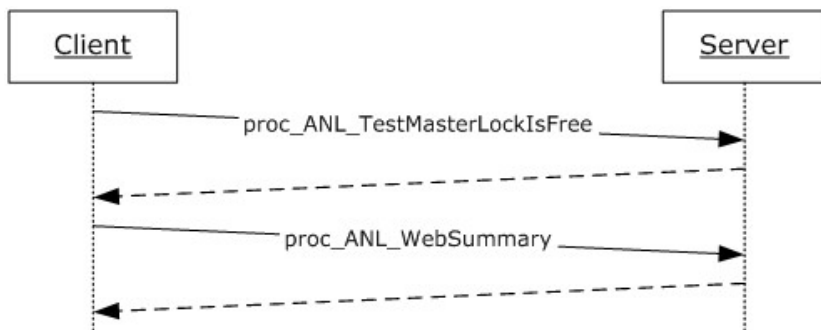
Figure 5: Data flow for reporting site-usage summary data

1. The protocol client calls the **proc_ANL_TestMasterLockIsFree** stored procedure (section 3.1.5.23).

2. The return code is "1". Therefore, the protocol client proceeds.

3. The protocol client calculates that February 6, 2008 is 2,958 days after January 1, 2000. The protocol client calls the **proc_ANL_WebSummary** stored procedure (section 3.1.5.39) and specifies the site identifier and the date value by using the following T-SQL syntax:

```
exec proc_ANL_WebSummary @webGuid='95688941-D67C-4B88-B9EF-
F31576C62E53',@todayDayId=2958
```

4. The return code is "1". Therefore, the protocol client displays the **Web Summary** result set (section 3.1.5.39.1).

4.5 Reporting Site Usage, Multiple Result Sets

This example describes requests that can be made by a protocol client for usage reporting. The protocol client reports all of the usage data that is exposed by this protocol and applies to the site that has the site identifier "95688941-D67C-4B88-B9EF-F31576C62E53". The following diagram shows a possible data flow for this example.
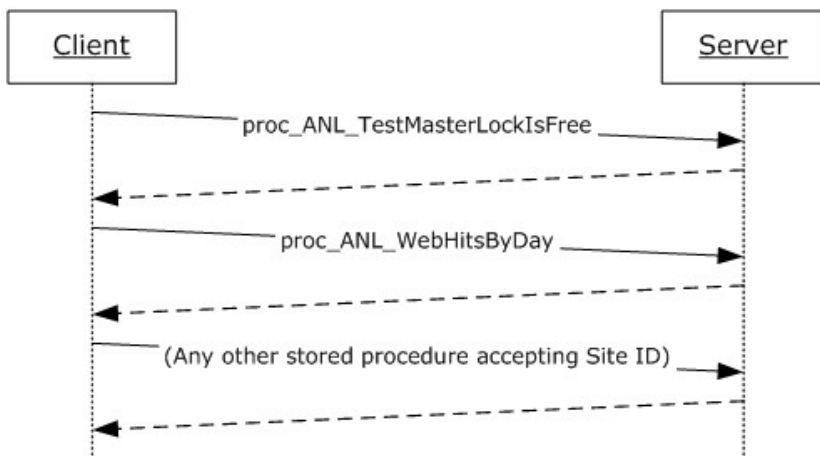
Figure 6: Data flow for complex reporting that involves multiple result sets

1. The protocol client calls the **proc_ANL_TestMasterLockIsFree** stored procedure (section 3.1.5.23).

2. The return code is "1". Therefore, the protocol client proceeds.

3. The protocol client calls the **proc_ANL_WebHitsByDay** stored procedure (section 3.1.5.31) and specifies the site identifier by using the following T-SQL syntax:

```
exec proc_ANL_WebHitsByDay @webGuid='95688941-D67C-4B88-B9EF-F31576C62E53'
```

4. The return code is "1". Therefore, the protocol client displays the **Big Hits by Day** result set (section 2.2.4.2).

5. The protocol client continues to call any stored procedure of this protocol that uses a *@webGuid* parameter by following the pattern demonstrated in steps 3 and 4.

5 Security

5.1 Security Considerations for Implementers

Security for this protocol is controlled by the access rights for the back-end databases on the protocol server, which are negotiated through the TDS Protocol, as described in [MS-TDS].

The database access account that is used by the **front-end Web server** can access the appropriate **Shared Services Provider (SSP) database** on the back-end database server. If the account does not have sufficient access rights, access is denied when attempting to set up the TDS connection to the Shared Services Provider (SSP) database or when calling the stored procedures.

Interactions with SQL are susceptible to tampering and other forms of security risks. Implementers are advised to sanitize input parameters for stored procedures before invoking the stored procedure.

5.2 Index of Security Parameters

None.

6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft® Office SharePoint® Server 2007

- Microsoft® SQL Server® 2005

- Microsoft® SQL Server® 2008

- Microsoft® SQL Server® 2008 R2

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

7 Change Tracking

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

8 Index