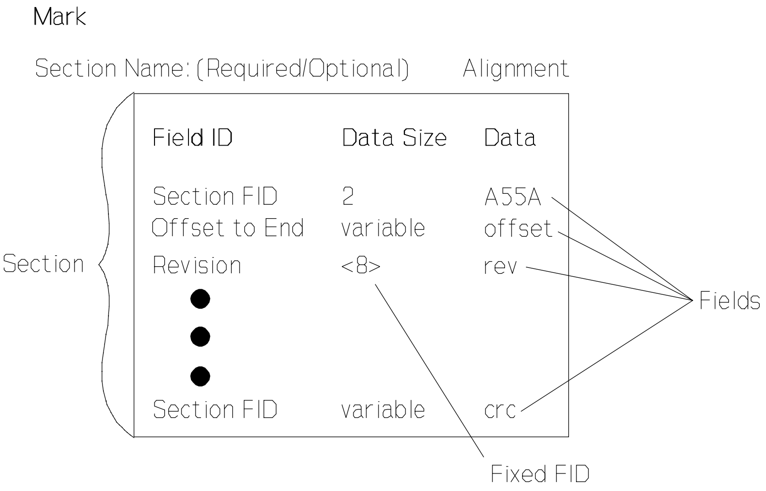
Sections

Section Presentation

In this chapter, each section description has the following parts:

- Mark - a set mark or file mark (not every section has a mark).
- Section identification - a section name followed by "required" or "optional."
- Alignment - describes a particular position, if any, the section must occupy on the media (e.g., logical sector aligned).
- Section description.
- Remarks.

Figure 2-1 shows how a section is presented. The "Field ID" column identifies the FID(s).



Field ID	Data Size	Data
Section FID	2	A55A
Offset to End	variable	offset
Revision	<8>	rev
●		
●		
●		
Section FID	variable	crc

Figure 2-1. Section Description

Figure 2-1 shows that for nearly all sections, the first field identifies the section and an almost identical field marks the end of the section (the same FID is used). The first field is also used to resynchronize parsing. The data for the first field is always 0xA55A. The "offset to end" field contains the offset from the beginning of the following field to the beginning of the last field. Not every section uses this field. Required fields have "(Req.)" following the field name (see "Revision" field above). SIDF calculates the CRC from the beginning of the first field to the beginning of the last field. The CRC is optional and its size value is 0 if no CRC is used.

The following example section reads as follows. First, an engine writes a set mark to the media, then the required session header. The section must be aligned on a physical boundary (see far right). Following the alignment are the section's fields.

SET MARK:

Session Header (Required): (Physical Sector Aligned)

Field ID	**Size**	**Data**
Session Header	2	A55A
Offset to End	variable	offset
	.	.
	.	.
Session Header	variable	crc

The following text presents the sections as they should occur on the media. The exception to this are trailers, which follow their corresponding headers.

Media level

The following describes the fields of the media level sections. The sequence of the sections are shown in "Media level" in Chapter 1.

Blank Space:

Field ID	Size	Data
Blank Space (Req.)	2	0xA55A
Offset to End	variable	offset
n/a	n/a	Fill this area with zeros
Blank Space (Req.)	variable	crc

n/a: not applicable

Remarks

Within a session, any space in the transfer buffer or sector not occupied by data must be padded with the Blank Space section. If there is not enough space for this section, fill the unused space with zeros.

Soft Media Mark: (Physical Sector Aligned)

Field ID	Size	Data
Soft Media Mark(Req.)	2	0xA55A
Mark Type	1	set mark or file mark
Next File Mark	variable	physical address
Previous File Mark	variable	physical address
Next Set Mark	variable	physical address
Previous Set Mark	variable	physical address
Soft Media Mark (Req.)	variable	crc

Remarks

The soft media mark section emulates tape marks (set mark and file mark) on media not supporting these marks. Unlike other types of set and file marks, these marks are linked together and form a singly or doubly linked list that is bounded by the medium (i.e., the list does not span the medium). The medium used dictates the type of list used. Write once media (e.g., WORM) can use only singly linked list because the location of the next mark is not known when the current mark is written. Write many media can use both list types.

Doubly linked lists are more convenient and faster to use than singly linked lists because you only have to advance to the first mark to find the head of the list. For a singly linked list, however, the head of the list is the last mark written. You must read each block, starting from the end of the medium ,until the last mark is found. If singly linked list is used on write many media, the engine must read each block until the last mark is found.

Note: This section occupies one physical sector.

Soft Media Mark is the section header and contains resynchronization data.

Mark Type - Contains a numerical value that defines the mark type. The following values are defined:

Mark Type Value	Mark Type
1	Set mark
2	File mark

Next File Mark - Points to the next file mark.

Previous File Mark - Points to the previous file mark.

Next Set Mark - Points to the next set mark.

Previous Set Mark - Points to the previous set mark.

Soft Media Mark is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Media header: (Required) (Logical Sector Zero)

Field ID	Size	Data
Media header (Req.)	2	0xA55A
Offset to End	variable	offset
Media Type		
Prior Media Type		
♦*Revision Level* (Req.)	<8>	"REV 1.00"
♦*Media Set Create Date And Time*	<4>	date/time
♦*Media Set Label*	variable	label
♦*Media Number* (Req.)	<2>	media number
Alternate Media Label	variable	alternate media label
♦*CRC Type*	variable	CRC type string
♦*Physical Sector Size* (Req.)	variable	physical sector size
♦*Database Location Method*	variable	database location method
Media Mark Type	variable	hard or soft
Media Authentication	variable	authentication
Media header (Req.)	variable	crc

♦ Critical field, repeat in transfer buffer header. A field is critical if its loss adversely affects the restoration process or performance. Thus, they should be repeated in each transfer buffer header.

Remarks

The media header is the first section placed on the media and may be followed by one or more sessions. Since the media header occupies logical sector zero, the engine or device may put any information it needs before this sector (e.g., an ANSI label). SIDF does not care about the data that precedes logical sector zero.

Media header is the section header and contains the resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Media Type identifies the current media's type (e.g., 4mm or 8mm). Each medium within a media set can be different.

Prior Media Type identifies the prior media's type.

Revision Level contains a non-null terminated string that shows the engine's revision level.

Media Set Create Date And Time contains the date and time the backup started. This value is the same for all media in the set.

Media Set Label contains a null-terminated string that names the media set. This string is the same for all media in the set.

Media Number contains a sequence number that indicates the medium's position in the media set.

Alternate Media Label provides another way of identifying a medium within a media set. Instead of using the media set label and media number, each medium can be given a different name.

CRC Type contains a string identifying the CRC type. For SMS, the data is "SMS."

Physical Sector Size is required and could be calculated by finding the distance from the beginning of the media header to the first session header, however, specifying it here is preferable.

Database Location Method contains a value that describes where the databases are on the media (see "Database Location Information" in Chapter 1). If method five is used, the field's data subfield specifies the method and the location offset (e.g., 100, 1000, etc.). Database location offsets are multiples of logical sectors and aligned on physical sector boundaries.

Media Mark Type indicates if the medium uses soft or hard marks (see Soft Media Marks).

0 File and set marks are preformed by hardware.
1 File and set marks are soft
2 File marks are done by hardware and set marks are two consecutive file marks.

Media Authentication contains a user-created password. The user accessing the media must validate his password against the media authentication before access to the media is granted.

Media header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

SET MARK:

Session Header: (Required) (Physical Sector Aligned)

Field ID	**Size**	**Data**
Session Header (Req.)	2	A55A
Offset to End	variable	offset
•*Session Date And Time*	<4>	date/time
Session Description	variable	description
Software Name	variable	name
Software Type	variable	type
Software Version	variable	version
•*Source Name* (Req.)	variable	name
•*Source Type* (Req.)	variable	type
•*Source Version* (Req.)	variable	version
Session Authentication	variable	authentication
♦*CRC Type*	variable	CRC type string
Maximum Transfer Buffer Size (Req.)	variable	size (in bytes)
Session Header (Req.)	variable	crc

• Fields used to match the transfer buffer to the specified session are flagged with a '•' symbol.

♦ Critical field, repeat in transfer buffer header. A field is critical if its loss adversely affects the restoration process or performance. Thus, they should be repeated in each transfer buffer header.

Remarks

The session header immediately follows the media header and set mark (the **SET MARK** precedes all session headers except the first one in a media set). The session header is uniquely identifiable through the *session date and time*, *session description*, *source name*, *source type* and *source version* fields. The session data follows the session header.

Session Header is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Session Date And Time contains the date and time the session started.

Session Description contains a user created description. The data is a null-terminated string.

Software Name contains the name of the engine servicing the data (e.g., SBackUp).

Software Type contains a null-terminated string that describes the software type.

Software Version contains a null-terminated string that shows the revision level of the engine servicing the source data.

Source Name contains a null-terminated string that uniquely identifies where the data was taken from. The engine creates this string or retrieves it from the source.

Source Type contains a null-terminated string.

Source Version contains a null-terminated string that shows the source's revision level.

Session Authentication contains a user created password that must be validated before any user can use the session.

CRC Type - This field contains a string identifying the CRC type. For SMS, the data is "SMS."

Maximum Transfer Buffer Size contains the maximum transfer buffer size that this session can have. Transfer buffers in the session maybe of any size up to this maximum size.

> **Note:** Transfer buffer sizes must be a multiple of the media physical sector size.

Session Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

FILE MARK:

Session Trailer: (Required) (Physical Sector Aligned)

Field ID	Size	Data
Session Trailer (Req.)	2	0xA55A
Offset To End	variable	offset
•*Session Date And Time* (Req.)	<4>	date/time
Session Description	variable	description
Software Name	variable	software name
Software Type	variable	software type
Software Version	variable	software version
•*Source Name* (Req.)	variable	source name
•*Source Type* (Req.)	variable	source type
•*Source Version* (Req.)	variable	source version
Session Trailer (Req.)	variable	crc

• These fields are used to match the session trailer to the specified session.

Remarks

The session trailer follows the source's file system data and file mark.

Session Trailer is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

The following fields are duplicated in this section to ensure the correct session trailer is found. For a description of these fields, see Session Header.

Session Date And Time
Session Description
Software Name
Software Type
Software Version
Source Name
Source Type
Source Version

Session Trailer is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Session Index: (Required)

Field ID	**Size**	**Data**
Session Index (Req.)	2	0xA55A
Offset To End	variable	offset
•*Session Date And Time* (Req.)	<4>	date/time
Session Description	variable	description
•*Source Name* (Req.)	variable	name
•*Source Type* (Req.)	variable	type
•*Source Version* (Req.)	variable	version
Media Set Open Date And Time	<4>	date/time
Media Set Label	variable	label
	Repeat these fields as necessary	
Media Number	<2>	media number
Media Partition	variable	media partition
The following fields repeat for each Data Set in the session		
Sector Address	variable	sector physical address
Transfer Buffer Offset	variable	media buffer offset
Fully Qualified Path	<1>	TRUE/FALSE
The following fields repeat for each name space the data set occupies.		
Name Space Type	4	name space type
Name Positions (optional)	variable	name positions
Separators Positions (optional)	variable	separators position
Data Set Name	variable	data set name
Session Index (Req.)	variable	crc

• Fields used to match the Session Index to the specified session are flagged with a '•' symbol; these fields are required if the Session Trailer is omitted.

Remarks

The session index records the data sets backed up and their location on the media (for one session only). This section follows the session trailer. The media index must follow the session index.

> **Note:** The session index is encapsulated in transfer buffers to simplify the spanning of the index.

The software fields contain information about the program that serviced the target's data (e.g., SBackup).

The fields following media set label, except the last field, are three groups of nested information. The inner group (the name space information) repeats for each name space the data set exists under. The middle group (data set location) repeats for each data set. The outer group is written at the start and every time the media number or media partition changes.

> **Note:** The first name in the name space list of each data set is the creator’s name space name.

Since the convention for specifying a path varies from one file system to another, SIDF defines the following. The path is broken into two components, the name (or node) and the separator (for DOS these are ‘:’ and ‘\’). The beginning of each component is kept in the *Name Position* array and the *Separator Position* array. For example, a data set with the following path

SYS:LOGIN\LOGIN.EXE

has the following values for its arrays:

Name Positions:	0, 4, 10
Separator Positions:	3, 9, 0

Notice that the array contains the beginning position of each name and separator. Also notice that the last value of the separator position’s array is zero. A value of zero indicates that the last node is a terminal node (a file). If the last node is not a terminal node, as shown below:

VOL1:TOOLS\COMPILERS\

the arrays contains:

Name Position:	0, 5, 11
Separator Position:	4, 10, 20

The data type of *the Name Position*s and *Separator Positions* arrays are words, not bytes. Since their data size descriptors indicate the number of bytes in the array, they must be divided by two to get the correct number of elements in each array.

Session Index is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

For a description of the following fields, see Session Header:

Session Date And Time
Session Description
Source Name
Source Type
Source Version

Target Name Type Ver contains the source's name, type, and version.

Media Set Open Date And Time contain the date and time the media set was created.

Media Set Label contains the media set's label. This is defined in the media header.

Media Number contains the sequence number of the medium that contains the specified data set (see the data set name field). This field is required if the sequence number changes. That is, if the previous data set's media number is the same as the current data set's media number, do not write this field for the current data set.

Media Partition contains the partition number that contains the specified data set. This field is required only if the partition number changes.

Sector Address contains the address of a transfer buffer.

Transfer Buffer Offset contains the offset to a record.

Path Fully Qualified is a boolean value. If set to TRUE, the Data Set Name is fully qualified. If set to FALSE, Data Set Name contains a partial path.

Name Space Type contains the data set's name space type. This field and the following four fields repeat as a group for each name space that exists in the data set's environment. The first entry in this list is the name space that created the data set. The reset of the name spaces follows the order below:

DOS
MAC
OS2
NFS
FTAM

Name Positions contains an array that has the beginning position of each node in the data set name (see the above remarks for more information).

Separators Positions contains an array that has the beginning of each separator in the data set name (see the above remarks for more information).

Data Set Name. If the data set is a parent, then this contains a fully qualified data set name; otherwise, it contains a partial path.

> **Note:** The engine's case sensitivity to the data set's name is dependent upon the name space.

Session Index is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Media Index: (Required)

Field ID	**Size**	**Data**
Media Index (Req.)	2	0xA55A
Offset To End	variable	offset
Previous Media Index	0	
Media Number	<2>	media number
Media Partition	variable	media partition
Sector Address	variable	sector physical address
Number Of Sessions	variable	number of sessions
The following repeats *Number Of Sessions* times		
Session Date And Time (Req.)	4	date/time
Session Description	variable	description
Media Number (Req.)	<2>	media number
Media Partition	variable	media partition
Sector Address (Req.)	variable	sector physical address
Number Of Databases	variable	number of databases
The following repeats *Number Of Databases* times		
Session Date And Time (Req.)	4	date/time
Database Name	variable	database name
Media Number (Req.)	<2>	media number
Media Partition	variable	media partition
Sector Address (Req.)	variable	sector physical address
Media Index (Req.)	variable	crc

Remarks

The media index keeps track of where the session indexes and databases are on a media set. The media index applies to the entire media set and up to the point where it occurs (it may be repeated as many times as necessary). Each media index points to the previous media index in the media set. This section follows the session index.

> **Note:** The media index is encapsulated in transfer buffers when written to the media to simplify the spanning of the index.

For more information about databases, see Chapter 1, "Introduction."

> **Note:** Both *Media Number* and *Media Sector Physical Address* are required if *Previous Media Index* is specified.

Media Index is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Previous Media Index

Media Number and the following two fields contains the location of current media index.

Media Partition (see media number).

Sector Address (see media number).

Number Of Sessions contains the number of sessions this media index references. The following six fields repeat as a group for each session.

Session Date And Time contains the date and time the session started.

Session Description contains the user's description for the session. This information is copied from the session header.

Media Number contains the medium's sequence number that contains the beginning of the session index.

Media Partition contains the partition number that contains the beginning of the session index.

Sector Address contains the sector address that starts the session index.

Number Of Databases contains the number of databases on the medium.

The following repeats Number Of Databases times
Session Date And Time
Database Name
Media Number
Media Partition
Sector Address

Media Index is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

FILE MARK

Media trailer: (Required if the session overflows the medium) (Physical Sector Aligned)

Field ID	**Size**	**Data**
Media trailer (Req.)	2	0xA55A
Close Date And Time	<4>	date/time
Media trailer (Req.)	variable	crc

Remarks

The media trailer is required if the session overflows the medium.

media trailer is the section header and contains resynchronizing data.

Close Date And Time contains the date and time this media trailer was written.

Media trailer is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Transfer Buffer Level

The following describes the fields of the transfer buffer level sections. To understand the role and sequence of these data sections, see "Transfer Buffer Level" in Chapter 1.

Transfer Buffer Header: (Required) (Physical Sector Aligned)

Field ID	Size	Data
Transfer Buffer Header (Req.)	2	0xA55A
Offset To End	variable	offset
Transfer Buffer Size	variable	size (in bytes)
Unused in This Transfer Buffer	variable	unused
Session ID	variable	session ID
•*Session Date And Time* (Req.)	<4>	date and time
Session Description	variable	description
Transfer Buffer CRC	variable	transfer buffer data crc
Transfer Buffer Sequence	variable	buffer sequence
Sector Address	variable	sector physical address
Software Name	variable	software name
Software Type	variable	software type
Software Version	variable	software version
•*Source Name* (Req.)	variable	name
•*Source Type* (Req.)	variable	type
•*Source Version* (Req.)	variable	version
♦*Revision Level*	<8>	revision level string
♦*Media Set Open Date And Time*	<4>	date/time
♦*Media Set Label*	variable	label
♦*Media Number*	<2>	media set number
♦*CRC Type*	variable	CRC type string
♦*Physical Sector Size*	variable	physical sector size
♦*Database Location Method*	variable	database location Info
Transfer Buffer Header (Req.)	variable	crc

• Fields used to match the Session Index to the specified session are flagged with a '•' symbol; these fields are required if the session trailer is omitted or if session interleaving is used.

♦ Critical field. These fields are duplicated here from other sections because they serve to

- Tie together a session's transfer buffers (not all transfer buffers within a session belong to that session [i.e., session interleaving]). If the other sections lost one or more of these fields, the duplicated fields can be used as identifiers to gather all transfer buffers of a session.
- Indicate the state of the session (e.g., physical sector size).
- Increase the restoration performance.

Remarks

Each transfer buffer requires a transfer buffer header.

> **Note:** The minimum transfer buffer size is the medium's physical sector size.

Transfer Buffer Header is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Transfer Buffer Size contains the transfer buffer's size, which must be a multiple of the medium's physical sector size (the session header shows the maximum buffer size). This information can be used to jump from one transfer buffer to the next.

Unused in This Transfer Buffer contains the number of bytes not used at the end of the transfer buffer. That is, the space between the end of the last record/subrecord and the end of the transfer buffer.

Session ID

Session Date And Time contains the session's starting date and time.

Session Description contains the user's description of the session.

Transfer Buffer CRC contains a CRC value for the transfer buffer's data. This does not include the transfer buffer header.

Transfer Buffer Sequence contains the sequence number of the transfer buffer relative to the first transfer buffer of the current session.

Sector Address contains a sector address that is relative to the session header.

The following fields, except the trailer field, are repeated from the media and session header as a precaution against a catastrophic failure. The information shows the context of the transfer buffer within the media and session. For a description of each field, see their respective sections.

Session header information:
Software Name
Software Type
Software Version
Source Name

Note: The engine's case sensitivity to the sources name is dependent upon the name space type.

media header information:
Revision Level
Media Set Open Date And Time
Media Set Label
Media Number
CRC Type
Physical Sector Size
Database Location Method (see "Database Location Information" in Chapter 1 for more information).

Transfer Buffer Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Data Record Header: (Required)

Field ID	**Size**	**Data**
Data Record (Req.)	2	0xA55A
Archive Date And Time	<4>	date/time
Record Size (Req.)	variable	size
Data Record (Req.)	variable	crc

Remarks

Every data set information and data set data pair requires a record header. See "Transfer Buffer Level" and "Data Set Level" in Chapter 1 for more information.

If the record cannot fit into a transfer buffer, the remaining data is put into a subrecord and placed into the next transfer buffer. This buffer has the following: a transfer buffer header, followed by a subrecord header, and then the remaining data.

If the record header cannot fit into the transfer buffer, start a new transfer buffer.

Data Record is the section header and contains resynchronization data.

Archive Date And Time records date and time the data set was backed up.

Record Size contains the size of the record data (the data consists of the data set information and data set data). It does not include the size of the subrecord. This size information can be used to skip from one record to the next.

Data Record is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Data Set Information(1.3): (Required)

Field ID	**Size**	**Data**
Directory Information (Req.)	2	0xA55A
Offset To End	variable	
	(Scan Information Follows)	
Attributes	variable	attributes
Creator ID	<4>	creator ID
Creator Name Space Number	<4>	name space number
Primary Data Stream Size	<4>	primary data stream size
Total Streams Data Size	<4>	total streams data size
Modified Flag	<1>	modified flag
Deleted Flag	<1>	deleted flag
Parent Flag	<1>	parent flag
Access Date And Time	<4>	access date and time
Create Date And Time	<4>	create date and time
Modified Date And Time	<4>	modified date and time
Archived Date And Time	<4>	archived date and time
Other Information	variable	other information
	(Data Set Name list follows)	
Data Set Name (Req.)	variable	(Data Set Name Space List Information) UINT16 *Data Set Name Size* UINT8 *Name Space Count* UINT8 *Key Info Size* UINT8 *Key Info[Key Info Size]*
		(Name Space Information for Data Set) The following repeat "Name Space Count" times UINT32 *Name Space Type* UINT32 *Reserved* UINT8 *Count* UINT16 *Name Position[Count]* UINT16 *Separator Position[Count]* UINT16 *Name Length* UINT8 *Name[Name Length + 1]*
Directory Information (Req.)	variable	crc

Remarks

This section is required for each data set. The information contained in this section is common to most targets.

Directory Information is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Attributes contain a bit map of the data set's attributes (e.g., read only, hidden, etc.).

Creator ID. This is defined by the engine servicing the data set.

Creator Name Space Number contains the name space number that created the data set. The name space numbers are:

Name Space	Name Space Number
DOS	0x0
MAC	0x1
UNIX	0x2
OS/2	0x3
WINDOWS	0x4

Primary Data Stream Size - This is the size, in bytes, of the data stream and does not represent the size of the data set. For example, the primary data streams for DOS and Macintosh are the file and data fork, respectively.

Total Streams Data Size contains the total number of physical blocks occupied by all data streams. DOS has one data stream, Macintosh files have two streams (data fork and resource fork), and FTAM may have three streams.

Modified Flag

Deleted Flag

Parent Flag is TRUE if the data is a parent (e.g., directories and subdirectories are parents, but files are not). All data sets scanned, that are not parents, are children of the last parent scanned. All parents do not have to have children.

Access Date And Time contains the last time the data set was accessed. The engine's access should not alter this date.

Create Date And Time contains the data set's creation date.

Modified Date And Time contains the data set's last modified date and time.

Archived Date And Time contains the data set's last archive date and time value.

Other Information - Contains developer-specific information. This field is not currently used.

Data Set Names is a list of the data set's name as it appears under each supported name space. The first entry is the data set's name as it appears under the name space that created it.

Data Set Name Size defines the actual memory space used by the list from *Data Set Name Size* through the end of the data set name list inclusively.

Name Space Count contains the number of name spaces in the list.

Key Info Size defines the size of *Key Information* in bytes.

Key Info contains key encryption information and is variable in length. This key is used to encrypt and decrypt the data set.

Note: Key Info is not used currently.

Name Space Type - Specifies the data set's name space type (see name space number on pg. 2-24, 2-27, 2-40)

Count: Specifies the size of the *Name Positions* and *Separator Positions* array.

Name Positions and **Separator Positions** - For more information about these fields, see Session Index (pg. 2-12).

Name Length - Contains the length of *Name*.

Name - Contains the data set's name.

Directory Information is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Data Set Information(1.4A): (Required)

Field ID	Size	Data
Directory Information (Req.)	2	0xA55A
Offset To End	variable	
	(Scan Information Follows)	
Attributes	variable	attributes
Creator ID	<4>	creator ID
Creator Name Space Number	<4>	name space type
Primary Data Stream Size	<4>	primary data stream
Total Streams Data Size	<4>	total stream data size
Modified Flag	<1>	modified flag
Deleted Flag	<1>	deleted flag
Parent Flag	<1>	parent flag
Access Date And Time	<4>	access date and time
Create Date And Time	<4>	create date and time
Modified Date And Time	<4>	modified date and time
Archived Date And Time	<4>	archived date and time
Other Information	variable	other information
	(Data Set Name list follows)	
	(Name Space Information for Data Set) The following fields repeat for each Data Set Name of the current Data Set.	
Name Space Type	4	name space type
Name Positions (optional)	variable	name positions
Separators Positions (optional)	variable	separators position
Data Set Name	variable	data set name
Directory Information (Req.)	variable	crc

Remarks

This section is required for each data set. The information contained in this section is common to most targets.

Directory Information is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Attributes contain a bit map of the data set's attributes (e.g., read only, hidden, etc.).

Creator ID. This is defined by the engine servicing the data set.

Creator Name Space Number contains the number of the name space that created the data set. The name space numbers are:

Name Space	Name Space Number
DOS	0x0
MAC	0x1
UNIX	0x2
OS/2	0x3
WINDOWS	0x4

Primary Data Stream Size - This is the size, in bytes, of the data stream and does not represent the size of the data set. For example, the primary data streams for DOS and Macintosh are the file and data fork, respectively.

Total Streams Data Size contains the number of physical blocks occupied by all data streams. DOS has one data stream, Macintosh files have two streams (data fork and resource fork), and FTAM has three streams.

Modified Flag

Deleted Flag

Parent Flag is TRUE if the data is a parent (e.g., directories and subdirectories are parents, but files are not). All data sets scanned, which are not parents, are children of the last parent scanned. All parents do not have to have children.

Access Date And Time contains the last time the data set was accessed. The engine's access should not alter this date.

Create Date And Time contains the data set's creation date.

Modified Date And Time contains the data set's last modified date and time.

Archived Date And Time contains the data set's last archive date and time value.

Other Information - Contains developer-specific information. This field is not currently used.

Name Space Type contains the data set's name space type. This field and the following four fields repeat as a group for each name space that exists in the data set's environment. The first entry in this list is the name space that created the data set. The reset of the name spaces follows the order below:

DOS
MAC
OS2
NFS
FTAM

Name Positions contains a UINT16 array that has the beginning position of each node in the data set name (for more information about these fields, see Session Index [pg. 2-12]).

Separators Positions contains a UINT16 array that has the beginning of each separator in the data set name (for more information about these fields, see Session Index [pg. 2-12]).

Data Set Name. If the data set is a parent, then this contains a fully qualified data set name; otherwise, it contains a partial path.

> **Note:** The engine's case sensitivity to the data set's name is dependent upon the name space.

Directory Information is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Data Subrecord Header: (Required if a record overflow a transfer buffer)

Field ID	Size	Data
Data SubRecord (Req.)	2	0xA55A
Subrecord Size (Req.)	variable	size
Data SubRecord (Req.)	variable	crc

Remarks

The subrecord header is used when the data set information and data set data cannot fit into one record or when a record cannot fit into a transfer buffer. Repeat the subrecord as often as needed.

Data SubRecord is the section header and contains resynchronization data.

Subrecord Size - Contains the size of the data following the subrecord header. This size can be used to jump around the data.

Data SubRecord is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Data Set Level

The data set level defines how an engine must format the target’s data. For more information about the role of the data set level, see "Data Set Level" in Chapter 1.

Data Set Data

Volume Header: (Required if backing up a volume)

Field ID	Size	Data
Volume Header (Req.)	2	0xA55A
Volume Header (Req.)	variable	crc

Remarks

The volume header begins a new volume. The sections that follow the volume header section are

- Full path
- Macintosh characteristics (if applicable)
- NFS characteristics (if applicable)
- Extended attributes
- Trustees
- Volume restriction
- Volume trailer

The volume's data (e.g., bindery, directories, and files) follows the volume trailer. If other name spaces exist, place them between the full path and extended attributes sections.

Volume Header is the section header and contains resynchronization data.

Volume Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Directory Header: (Required if backing up a directory)

Field ID	Size	Data
Directory Header (Req.)	2	0xA55A
Backup Options		backup options
Directory Header (Req.)	variable	crc

Remarks

The directory header begins a new directory. The sections that follow the header are

- Full path
- Data set characteristics
- Macintosh characteristics (if applicable)
- NFS characteristics (if applicable)
- Extended attributes
- Trustees
- Directory trailer

Following the directory trailer are the contents of the directory (i.e., subdirectories and files). If other name spaces exist, place them between the full path and extended attributes sections.

Directory Header is the section header and contains resynchronization data.

Backup Options when the file was backed up, the following information was purposefully excluded from data set data. (extended attribute for dir only).

0x0800	Extended Attributes Excluded
0x1000	Data Stream Excluded

Directory Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Transaction Set Header: (Required if backing up a transaction set)

Field ID	Size	Data
Transaction Set Header (Req.)	2	0xA55A
Offset to End	variable	offset set to end
Transaction Set Type	variable	transaction set type
Transaction Set Name	variable	transaction set name
Transaction Set Header (Req.)	variable	crc

Remarks

This section is used to group data sets (e.g., files) that must be serviced as a set or not at all (e.g., files belonging to one database). Following the header is a transaction set full path section that lists the full path of each file in the database (see Figure 1-7). Following this section is one or more transaction data set(s). Each transaction data set consists of the following sections:

- Transaction set full paths (this section repeats for each transaction set).
- Transaction set file header
- Macintosh characteristics (if applicable)
- NFS characteristics (if applicable)
- Extended attributes
- Trustees
- Data stream header
- File data (not a section)
- Data stream trailer
- Transaction set file trailer

Following the last transaction data set is the transaction set trailer. Everything encapsulated by the transaction set header and trailer, inclusive, is one data set (see "Data Set Level" in Chapter 1 for more information).

Transaction Set Header is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Transaction Set Type. This is defined by the engine servicing the data.

Transaction Set Name. This is defined by the engine servicing the data set.

Transaction Set Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

File Header: (Required if backing up a file)

Field ID	**Size**	**Data**
File Header (Req.)	2	0xA55A
Data Stream Number	variable	data stream number
Data Stream Type	variable	data stream type
Data Stream Compressed Type	variable	data stream compressed size
Backup Options		backup options
File Header (Req.)	variable	crc

Remarks

The file header begins a new file. The sections that follow the header are

- Full path
- Data set characteristics
- Macintosh characteristics (if applicable)
- NFS characteristics (if applicable)
- Extended attributes
- Trustees
- Data stream header
- Data set data (this is not a section, but the data set data, e.g., DOS file or Macintosh data fork)
- Data stream trailer
- If the data set is a Macintosh file, the sections below follow:
    - Data stream header
    - Data set data (resource fork)
    - Data stream trailer
- File Trailer

File Header is the section header and contains resynchronization data.

The following three fields repeat for each nonzero data stream type.

Data Stream Type contains the data stream type as defined below:

0	Clear data
1	Sparse data stream

Data Stream Number contains the data set's data stream number. The following are defined:

0	Data file (e.g., DOS executable and text files or Macintosh's data fork)
1	Macintosh resource fork
2	FTAM fork

Data Stream Compressed Type contains the type of compression used on the data stream. This field exist if the data stream type is compressed. Only one type has been defined so far

0x1	Novell Compression Version 1.0

Backup Options when the file was backed up, the following information was purposefully excluded from data set data. (extended attribute for dir only).

0x0800	Extended Attributes Excluded
0x1000	Data Stream Excluded

File Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Transaction Set File Header: (Required if backing up a transaction set file)

Field ID	**Size**	**Data**
Transaction Set File Header (Req.)	2	0xA55A
Offset to End	variable	offset to end
TS Name Positions	variable	name positions
TS Name Separator Position	variable	separator positions
TS Name Space Type	variable	name space type
TS Path Name	variable	full path name
Transaction Set File Header (Req.)	variable	crc

Remarks

The transaction set data and the transaction set file trailer follow this header section.

Transaction Set File Header is the header for this section and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

For information on TS Name Positions, TS Name Separator Position, and TS Name Space Type see "Session Index" (pg. 2-12).

TS Path Name contains the full path (from the volume to the data set's name [inclusive]) as it appears under the name space that created it.

Transaction Set File Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Bindery Header: (Required if backing up the bindery)

Field ID	Size	Data
Bindery Header (Req.)	2	0xA55A
Bindery Header (Req.)	variable	crc

Remarks

This section begins a new bindery. The sections following the bindery header are

- Data stream header
- Data for NET$OBJ.SYS (not a section)
- Data stream trailer
- Data stream header
- Data for NET$PROP.SYS (not a section)
- Data stream trailer
- Data stream header
- Data for NET$VAL.SYS (not a section)
- Data stream trailer
- Bindery trailer

The bindery section is used for NetWare v3.11 and below. The order of the bindery files are important.

If other name spaces exists, place them between the full path and extended attributes sections.

Bindery Header is the header for this section and contains resynchronization data.

Bindery Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Full Paths: (Required)

Field ID	**Size**	**Data**
Full Paths (Req.)	2	0xA55A
Offset to End	variable	offset
Path Is Fully Qualified	<1>	TRUE/FALSE
	The following repeats for each name space used	
Name Space Type	4	name space type
Name Positions	variable	name positions
Separators Positions	variable	separators position
Data Set Name	variable	data set name
Full Paths (Req.)	variable	crc

Remarks

This section contains the data set's full path (as it appears under every name space on the target) from the root of the file server to the data set inclusive. The first entry in the path list is the path under the data set's creator name space (e.g., if it is a DOS data set, then the DOS path is always the first entry).

Full Paths is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Path Is Fully Qualified. If this field is TRUE, the path is fully qualified. If set to FALSE, Path Name contains a partial path. This is used for UNIX paths or to speed up checks that make sure the path is a fully qualified path (i.e., there is no need to check the path if it is fully qualified). This flag is necessary for UNIX paths, because it helps to simplify the functions that check the paths. This field and the following two fields are repeated as a group for each name space.

> **Note:** A file name can contain a colon ":".

Path Name Space Type contains the name space type of the full path (see name space number on pgs. 2-24, 2-27, 2-40).

Name Positions is an array of integers and contains the beginning position of each node in path name (see Session Index [pg. 2-12] for more information). To get the number of elements in the array, divide the data size by two.

Separator Positions is an array of integers and contains the beginning position of each separator in path name (see Session Index [pg. 2-12] for more information). To get the number of elements in the array, divide the data size by two.

Path Name contains null-terminated string that represents the fully qualified path of the data set.

> **Note:** Case sensitivity to the path name is dependent upon the name space type and must be enforced by the engine where needed.

Full Paths is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Transaction Set Full Path: (Required for each name space)

Field ID	Size	Data
(The field below repeats for each file)		
Transaction Set Requested	variable	creator name space type
(The fields below repeats for each name space)		
TS Name Positions (Req)	variable	name positions
TS Separator Positions (Req.)	variable	separator positions
TS Name Space Type	variable	name space type
TS Path	variable	path name

Remarks

This section follows the transaction set header. Notice that although this is a section, it does not begin with a resynchronizing field and its first field does not match its last field. This section has two levels of groupings. The innermost group repeats for each name space. The outer group repeats for each file in the database.

For example, a database exists on a file service with DOS and Macintosh name spaces (DOS is the creator name space), is in directory Sys :Database, and consists of two files "a" and "b," this section contains the following:

Requested name space type:	DOS
Name positions:	0, 4, 12
Separator positions:	3, 11, 0
Name space type:	DOS
Path:	SYS:DATABASE\A
Name positions:	0, 5, 13
Separator position:	3, 12, 0
Name space type:	Macintosh
Path:	SYS::DATABASE:A

Requested name space type:	DOS
Name positions:	0, 4, 12
Separator positions:	3, 11, 0
Name space type:	DOS
Path:	SYS:DATABASE\B
Name positions:	0, 5, 13
Separator positions:	3, 12, 0
Name space type:	Macintosh
Path:	SYS::DATABASE:B

Transaction Set Requested contains the name space type that created the transaction set. This field is repeated for each file.

For information about TS Name Positions, TS Separator Positions, and TS Name Space Type see Session Index. These fields are repeated as a group for each name space.

TS Path is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Data Set Characteristics: (Required)

Field ID	**Size**	**Data**
Data Set Characteristics Header (Req.)	2	0xA55A
Offset to End	8	offset
The following repeats as many times as needed		
Data Set Characteristic Field	Size	Data
Data Set Characteristics Header (Req.)	variable	crc

Remarks

Data Set Characteristics Header is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Data Set Characteristic Field contains one or more of the following:

Characteristics	Size
Access Date	<2 (bytes)>
Access Time	<2 (bytes)>
Archive Date And Time	<4 (bytes)>
Archiver ID	variable[1]
Archiver Name[2]	variable
Audit Read/Write (C000)	110000bb[3]
Creation Date And Time	<4 (bytes)>
Directory (10)	1100000b[4,3]
Directory Space Restrictions	4 (bytes)
Execute Only (8)	1100000b[3]
Hidden (2)	1100000b[3]
Indexed	1100000b[3]
Inherited Rights Mask	<2 (bytes)>
Inhibit Ren/Del/Copy (E0000)	11000bbb[3]
Low/Mid/Hi Search (700)	11000bbb[3]
Modifier ID	variable[1, 7]

[1] This field's data is an array of bytes that has the following format: a four byte ID, a two byte type, followed by a NULL terminated name string. The engine must ensure that case sensitivity is enforced where needed.

[2] Archiver Name replaces Archiver ID under NetWare v4.0. It is a fully distinguishable unicode name.

[3] Bit pattern

[4] The absence of this characteristics indicates that the data set is a file.

Modifier Name[5]	variable
Modify Date And Time	<4 (bytes)>
Needs Archive (20)	1100000[3]
Other Name Space Characteristics	variable[6]
Owner ID	variable[7]
Owner Name[8]	variable
Purge (10000)	1100000b[3]
Read Only (1)	1100000b[3]
Shareable (80)	1100000b[3]
System (4)	1100000b[3]
Transactional (1000)	1100000b[3]

Note: The number following the characteristic shows the flag position for NetWare.

Data Set Characteristics Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Note: Since case sensitivity is dependent upon the name space type, the engine must enforce this where needed.

[5] Modifier Name replaces Modifier ID under NetWare v4.0. It is a fully distinguishable unicode name.

[6] FTAM information follows this characteristic.

[7] This field's data is an array of bytes that has the following format: a four byte ID, a two byte Type, followed by a NULL terminated name string.

[8] Owner Name replaces Owner ID under NetWare v4.0. It is a fully distinguishable unicode name.

Macintosh Characteristics: (Optional)

Field ID	**Size**	**Data**
Mac Characteristics (Req.)	2	0xA55A
Offset to End	variable	offset
Finder Information	<32>	finder info
Pro Dos Information	6	pro dos info
Directory Rights Mask	4	dir rights mask
Mac Characteristics (Req.)	variable	crc

Remarks

Mac Characteristics is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Finder Information contains 32 bytes of Macintosh finder information as shown below (see *Inside Macintosh*[9] for more information):

4 bytes	File Type
4 bytes	Engine Signature
2 bytes	Finder Flags
4 bytes	Icon location within a folder
2 bytes	Folder Number (not used with HFS)
16 bytes	Reserved

Pro Dos Information contains Pro-DOS data for the Apple II OS as shown below (see *Inside Appletalk*[9] for more information):

1 byte	File Type
1 byte	unused
2 bytes	Aux Type
2 bytes	unused

Directory Rights Mask contains NetWare specific data.

Mac Characteristics is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Pro Dos Information and *Directory Rights Mask* are NetWare specific.

Note: This section has not been finalized.

[9] Contact Apple Computer, Inc. for information about this document.

NFS Characteristics: (Optional)

Field ID	**Size**	**Data**
NFS Characteristics (Req.)	2	0xA55A
Offset to End	variable	offset
File Access Mode	<4>	access mode
Group Owner ID	<4>	owner id
R Device	<4>	r device
Number Of Links	<4>	number of links
Linked Flag	<1>	linked flag
First Creator Flag	<1>	first creator flag
ACS Flags	<4>	acs flags
User ID	<4>	user id
My Flags	<1>	my flags
Hard Link Paths	<1>	TRUE (1)
Hard Link Paths Section This group of fields (i.e., the following group and the HL Terminator field) repeat until "Hard Link Paths" is encountered.		
Path Information. This group of fields repeats until "HL Terminator" is found.		
HL Path Name	variable	Path
HL User ID	variable	id, type, name
HL Group Owner ID	<4>	Owner ID
HL File Access Mode	<4>	Access Mode
HL Terminator	0	
Hard Link Paths	<1>	FALSE (0)
NFS Characteristics (Req.)	variable	crc

Remarks

Hard Link Paths - Indicates the beginning and end of the hard link paths. Set it to TRUE to mark the beginning and to FALSE to mark the end.

The NFS Characteristics section contains two embedded groups of fields. The innermost group defines a hard link. This group consists of the HL Path Name, HL User ID, HL Group Owner ID, HL File Access Mode, and HL Terminator fields. The HL Terminator field indicates the end of one hard link path.

The outer group consists of one or more hard links. The Hard Link Paths field terminates this group. The example below shows how the hard links are recorded.

```
file:   /work/PhoneList
links:  /projects/description -> PhoneList
        /home/sam/list -> PhoneList
```

Hard Link 1:	HL Path Name:	projects
	User ID:	0x1L, 0x1, Supervisor
	Group Owner ID:	1
	File Access Mode:	0xFFFE
	HL Path Name:	description
	User ID:	0x1L, 0x1, Supervisor
	Group Owner ID:	0
	File Access Mode:	0xFFFE
	HL Terminator	
Hard Link 2:	HL Path Name:	home
	User ID:	0x1L, 0x1, Supervisor
	Group Owner ID:	1
	File Access Mode:	0xFFFE
	HL Path Name:	sam
	User ID:	0x1L, 0x1, Supervisor
	Group Owner ID:	1
	File Access Mode:	0xFFFE
	HL Path Name:	list
	User ID:	0x1L, 0x1, Supervisor
	Group Owner ID:	1
	File Access Mode:	0xFFFE
	HL Terminator	
	Hard Link Paths	

NFS Characteristics is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

File Access Mode contains the UNIX file access mode permissions.

Group Owner ID contains the file’s group ID (UNIX format).

Number Of Links contains the number of hard links.

R Device contains the raw device number.

Linked Flag contains NetWare NFS flags.

First Creator Flag contains the NetWare NFS first creator flags.

User ID contains the file owner’s ID (UNIX format).

ACS Flags contains access control flags for NetWare.

My Flags contains miscellaneous flags for NetWare NFS.

Hard Link Paths is set to TRUE if it marks the beginning of the hard link paths. It is set to FALSE if it marks the end of the hard link paths.

The following fields (i.e, the following group and the HL Terminator field) repeat until "Hard Link Paths" is encountered.

Path Information. This group of fields repeat until "HL Terminator" is found.

HL Path Name contains a full path to the file.

HL User ID contains the NetWare ID of the user that owns the first node in HL Path Name. For example, if the path is "/home/phonelist", the user’s id for home is contained in HL User ID.

HL Group Owner ID contains the NetWare NFS group owner’s ID.

HL File Access Mode contain’s the file’s UNIX access mode.

HL Terminator signals the end of one hard link path.

Hard Link Paths is set to FALSE if it marks the end of the hard link paths. It is set to TRUE if it marks the beginning of the hard link paths.

NFS Characteristic is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Data Stream Header: (Required)

Field ID	**Size**	**Data**
Data Stream Header (Req.)	2	0xA55A
Data Stream Name	variable	name
Data Stream Number	variable	data stream number
Data Stream Type	variable	data stream type
Data Stream Size (Req.)	variable	size
Data Stream Compressed Type	variable	data stream compressed size
Data Stream Expanded Size	variable	data set expanded size
Sparse Data Block Size[10]	variable	file block size
Sparse Data Stream Map[10]	variable	sparse bit map
Data Stream Header (Req.)	variable	crc

Remarks

The data stream header is a multipurpose header used to encapsulate data forks, resource forks, and sparse files. This section replaces an older version of the data stream header. However, the older section should still be supported. That section contained the following fields (the data sizes and data are the same as above):

Data Stream Header
Data Stream Name
Data Stream Size
Data Stream Header

Note: To know when the data stream header and trailer are used, see the bindery header and the file header sections.

The sparse data stream fields are used only for sparse data sets. The data set's sparseness is represented as follows. The data set is divided into blocks of *Sparse Data Block Size* bytes. Each block is represented by a corresponding bit in Sparse Data Stream Map (e.g., bit 0 represents block 0). If a block has data, its corresponding bit is set to 1 and the block is written out to the medium. If the block has no data, its corresponding bit is set to 0, and no data is written out to the medium.

Data Stream Header is the section header and contains resynchronization data.

[10] This field is only used for sparse data streams.

Data Stream Name contains a string that describes the data stream. This field is used for NetWare's bindery and for backwards compatibility. This field should not exist if *Data Stream Type* is used.

> **Note:** Since case sensitivity is dependent upon the name space type, the engine must enforce it where needed.

Data Stream Number contains the data set's data stream number. The following are defined:

0	Data file (e.g., DOS executable and text files or Macintosh's data fork)
1	Macintosh resource fork
2	FTAM fork

Data Stream Type contains the data stream type as defined below:

0	Clear data
1	Sparse data stream

Data Stream Size contains the size of the data stream only. For DOS this is the file size and for Macintosh and FTAM this is the data fork.
Data Stream Compressed Type contains the type of compression used on the data stream. This field exist if the data stream type is compressed. Only one type has been defined so far

0x1	Novell Compression Version 1.0

Data Stream Expanded Size is the size of the data stream before being compressed. This field exists if the data stream type is compressed.

Sparse Data Block Size contains the physical size block size. This field exists if data stream type is sparse data.

Sparse Data Stream Map contains a bit map of blocks containing data. See the above remarks for more details. This field exists if data stream type is sparse data.

Data Stream Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Data Stream Trailer: (Required)

Field ID	**Size**	**Data**
Data Stream Trailer (Req.)	2	0xA55A
Data Stream Is Invalid	0	
Data Stream CRC	variable	crc
Data Stream Trailer (Req.)	variable	crc

Remarks

The data stream trailer immediately follows the data stream.

Data Stream Trailer is the section header and contains resynchronization data.

Data Stream Is Invalid. This field is optional and is only present if the data stream was found to be invalid during the backup process.

Data Stream CRC contains a CRC value for data stream. This CRC uses the same polynomial as defined in the media header.

Data Stream Trailer is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Extended Attributes:

Field ID	Size	Data
Extended Attribute (Req.)	2	0xA55A
Offset To End	variable	offset
The following repeats for each extended attribute		
EAAccess	variable	access
EAKey	variable	key name
EAValue	variable	value
Extended Attribute (Req.)	variable	crc

Remarks

The order of the Access, Ac, and EAValue fields must follow the order shown above.

Extended Attribute is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Access contains the extended attribute access value. This field and the following two fields are repeated as a group for each extended attribute.

Ac contains the extended attribute key value.

EAValue contains the extended attribute value.

Extended Attribute is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Trustees: (Required, if it exists)

Field ID	Size	Data
Trustee Header (Req.)	2	0xA55A
Offset To End	variable	offset
	The following repeats for each trustee	
Trustee ID	variable	id, type, name
Trustee Name	variable	name
Trustee	2	data
Trustee Header (Req.)	variable	crc

Remarks

The trustee's ID and trustee must be backed up as a pair.

Trustee Header is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

The following three fields are repeated as a group for each trustee. One or more fields may not appear in the group as noted.

Trustee ID contains the trustee's ID (this field is not used for NetWare v4.0).

Trustee Name. This field is used only under NetWare v4.0 and is a fully distinguishable unicode name.

Trustee contains the trustee value.

Trustee Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Bindery Trailer: (Required if backing up the bindery)

Field ID	**Size**	**Data**
Bindery Trailer (Req.)	2	0xA55A
Bindery Trailer (Req.)	variable	crc

Remarks

Bindery Trailer is the section header and contains resynchronization data.

Bindery Trailer is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Transaction Set File Trailer: (Required if backing up a transaction set file)

Field ID	**Size**	**Data**
TS File Trailer (Req.)	2	0xA55A
TS File Trailer (Req.)	variable	crc

Remarks

Transaction Set File Trailer is the section header and contains resynchronization data.

Transaction Set File Trailer is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

File Trailer: (Required if backing up a file)

Field ID	Size	Data
File Trailer (Req.)	2	0xA55A
File Trailer Header (Req.)	variable	crc

Remarks

File Trailer is the section header and contains resynchronization data.

File Trailer is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Volume Restriction: (Required)

Field ID	Size	Data
Volume Restriction Header (Req.)	2	0xA55A
Offset To End	variable	offset
Volume Name (Req.)	variable	name
	The following repeats for each trustee	
Trustee ID	variable	id, type, name
Trustee Name	variable	name
Trustee Volume Restriction	variable	volume restriction
Volume Restriction Header (Req.)	2	crc

Remarks

Volume Restriction Header is the section header and contains resynchronization data.

Offset to End contains the offset from the beginning of the following field to the beginning of the last field.

Volume Name contains a NULL-terminated string.

The following three fields are repeated as a group for each trustee. One or more fields may not appear in the group as noted.

Trustee ID. This field is not used for NetWare v4.0.

Trustee Name. This field is used only for NetWare v4.0 and is a fully distinguishable unicode name.

Trustee Volume Restriction contains the volume restrictions.

Volume Restriction Header is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Transaction Set Trailer: (Required if backing up a transaction set)

Field ID	**Size**	**Data**
TS Trailer (Req.)	2	0xA55A
TS (Req.)	variable	crc

Remarks

Transaction Set Trailer is the section header and contains resynchronization data.

Transaction Set Trailer is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Directory Trailer: (Required if backing up a directory)

Field ID	**Size**	**Data**
Directory Trailer	2	0xA55A
Directory Trailer	variable	crc

Remarks

Directory Trailer is the section header and contains resynchronization data.

Directory Trailer is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.

Volume Trailer: (Required if backing up a volume)

Field ID	Size	Data
Volume Trailer (Req.)	2	0xA55A
Volume Trailer (Req.)	variable	crc

Remarks

Volume Trailer is the section header and contains resynchronization data.

Volume Trailer is the section trailer and contains a CRC value for all fields except for this trailer. Size is zero if a CRC was not used.