



iSCSI Boot Firmware Table (iBFT) as Defined in ACPI 3.0b Specification

March 1, 2007 - Version 1.01

Abstract

The iSCSI Boot Firmware (iBF) Table (iBFT) is a block of information that contains various parameters that are useful to the iSCSI Boot process. The iBFT is the mechanism by which iBF parameter values are conveyed to the operating system. The iBF builds and fills in the iBFT. The iBFT is available to the Windows operating system to enable a consistent flow of the boot process.

The current version of this paper is maintained on the Web at:

<http://go.microsoft.com/fwlink/?LinkId=82390>

Contents

1 Requirements	5
2 Conventions	5
IP Address	5
Alignment	5
Endianness	5
Length and Offset Values	5
Heap Array Format	5
Parameter Origins	5
3 iBFT Layout	6
iBFT Sections	6
iBFT Standard Structure Header	6
iBF Table Header	6
Control Structure	7
Structure Type / ID	7
Control Structure Offsets	7
Optional Structure Expansion	7
Structure Alignment in Memory	8
Initiator Structure	8
NIC Structure	8
Target Structure	9
4 References	10
Initiator Structure	10
NIC Structure	10
Target Structure	11

iSCSI BOOT FIRMWARE TABLE (iBFT) SPECIFICATION LICENSE

IMPORTANT—READ CAREFULLY: This Microsoft License Agreement ("Agreement") is a legal agreement between you (either an individual or a single entity) and Microsoft Corporation for the version of the Microsoft specification identified above which you are about to download ("Specification"). **BY DOWNLOADING, COPYING OR OTHERWISE USING THE SPECIFICATION, YOU AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, DO NOT DOWNLOAD, COPY, OR USE THE SPECIFICATION.**

The Specification is owned by Microsoft or its suppliers and is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. **THE SPECIFICATION IS LICENSED, NOT SOLD.**

1. GRANT OF LICENSE.

(a) Provided that you comply with all terms and conditions of this Agreement, Microsoft grants to you the following non-exclusive, worldwide, royalty-free, perpetual, non-transferable, non-sublicensable, limited license under any copyrights or patents that cover the table values described in the Specification and that are owned or licensable by Microsoft without payment of consideration to third parties,

(i) to reproduce copies of the Specification for your and your contractor's internal use for the sole purpose of (1) creating or modifying your firmware and/or BIOS for computing devices ("**Firmware**") so that it writes to memory the appropriate table values in the Specification or (2) creating or modifying your software so that it may read from memory the appropriate table values (the "**Purpose**"),

(ii) to implement the table values in your firmware and/or BIOS,

(iii) to license to third parties directly and indirectly the table values as part of your Firmware (and any related documentation).

The foregoing license is granted only to the extent necessary to accomplish the Purpose and to license and/or distribute your Firmware containing the table values to third parties. The foregoing license shall not extend to any features of your Firmware that (i) are not required to comply with the Specification or (ii) to which there was a practicable alternative to infringing a patent.

(b) Microsoft reserves all other rights it may have in the Specification, its implementation and any intellectual property therein. The furnishing of this document does not give you or any other entity any license to any other Microsoft patents, trademarks, copyrights or other intellectual property rights. Microsoft does not grant to you or any other entity any implied licenses or rights whatsoever under this Agreement. Specifically, this Agreement does not grant any express or implied licenses or rights to any enabling technologies that may be necessary to fully utilize the table or the table values described in the Specification.

2. ADDITIONAL LIMITATIONS AND OBLIGATIONS.

(a) You must implement the table in its entirety (e.g., all fields) and without modification (e.g., byte length, offset, and permissible values as described in the Specification).

(b) Your license rights to the Specification are conditioned upon you not creating, modify, or distributing your Firmware in a way that such creation, modification, or distribution may (a) create, or purport to create, obligations for Microsoft with respect to the Specification (or intellectual property therein) or (b) grant, or purport to grant, to any third party any rights or immunities to Microsoft's intellectual property or proprietary rights in the Specification.

(c) The foregoing license is applicable only to the version of the Specification which you are about to download. It does not apply to any additional versions of or extensions to the Specification.

(d) Without prejudice to any other rights, Microsoft may terminate this Agreement if you fail to comply with the terms and conditions of this Agreement. In such event you must destroy all copies of the Specification and must not further distribute the table values.

3. **INTELLECTUAL PROPERTY RIGHTS.** All ownership, title and intellectual property rights in and to the Specification, and any copies you are permitted to make herein, are owned by Microsoft or its suppliers.

© 2007 Microsoft Corporation. All rights reserved. By using or providing feedback on these materials, you agree to the attached license agreement.

4. **FEEDBACK.** You have no obligation to give Microsoft any suggestions, comments or other feedback ("Feedback") relating to this Specification. However, any Feedback you voluntarily provide may be used in Microsoft products and related specifications or other documentation (collectively, "Microsoft Offerings") which in turn may be relied upon by other third parties to develop their own products. Accordingly, if You do give Microsoft Feedback on any version of this Specification or the Microsoft Offerings to which it applies, you agree: (a) Microsoft may freely use, reproduce, license, distribute, and otherwise commercialize Your Feedback in any Microsoft Offering; (b) you also grant third parties, without charge, only those patent rights necessary to enable other Products to use or interface with any specific parts of a Microsoft product that incorporate Your Feedback; and (c) you will not give Microsoft any Feedback (i) that you have reason to believe is subject to any patent, copyright or other intellectual property claim or right of any third party; or (ii) subject to license terms which seek to require any Microsoft Offering incorporating or derived from such Feedback, or other Microsoft intellectual property, to be licensed to or otherwise shared with any third party.

5. **DISCLAIMER OF WARRANTIES.** To the maximum extent permitted by applicable law, Microsoft and its suppliers provide the Specification (and all intellectual property therein) and any (if any) support services related to the Specification ("Support Services") **AS IS AND WITH ALL FAULTS**, and hereby disclaim all warranties and conditions, either express, implied or statutory, including, but not limited to, any (if any) implied warranties or conditions of merchantability, of fitness for a particular purpose, of lack of viruses, of accuracy or completeness of responses, of results, and of lack of negligence or lack of workmanlike effort, all with regard to the Specification, any intellectual property therein and the provision of or failure to provide Support Services. **ALSO, THERE IS NO WARRANTY OR CONDITION OF TITLE, QUIET ENJOYMENT, QUIET POSSESSION, CORRESPONDENCE TO DESCRIPTION OR NON-INFRINGEMENT, WITH REGARD TO THE SPECIFICATION AND ANY INTELLECTUAL PROPERTY THEREIN. THE ENTIRE RISK AS TO THE QUALITY OF OR ARISING OUT OF USE OR PERFORMANCE OF THE SPECIFICATION, ANY INTELLECTUAL PROPERTY THEREIN, AND SUPPORT SERVICES, IF ANY, REMAINS WITH YOU.**

6. **EXCLUSION OF INCIDENTAL, CONSEQUENTIAL AND CERTAIN OTHER DAMAGES.** To the maximum extent permitted by applicable law, in no event shall Microsoft or its suppliers be liable for any special, incidental, indirect, or consequential damages whatsoever (including, but not limited to, damages for loss of profits or confidential or other information, for business interruption, for personal injury, for loss of privacy, for failure to meet any duty including of good faith or of reasonable care, for negligence, and for any other pecuniary or other loss whatsoever) arising out of or in any way related to the use of or inability to use the SPECIFICATION, ANY INTELLECTUAL PROPERTY THEREIN, the provision of or failure to provide Support Services, or otherwise under or in connection with any provision of this AGREEMENT, even in the event of the fault, tort (including negligence), strict liability, breach of contract or breach of warranty of Microsoft or any supplier, and even if Microsoft or any supplier has been advised of the possibility of such damages.

7. **LIMITATION OF LIABILITY AND REMEDIES.** Notwithstanding any damages that you might incur for any reason whatsoever (including, without limitation, all damages referenced above and all direct or general damages), the entire liability of Microsoft and any of its suppliers under any provision of this Agreement and your exclusive remedy for all of the foregoing shall be limited to the greater of the amount actually paid by you for the Specification or U.S.\$5.00. The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails its essential purpose.

8. **APPLICABLE LAW.** If you acquired this Specification in the United States, this Agreement is governed by the laws of the State of Washington. In respect of any dispute which may arise hereunder, you consent to the jurisdiction of the state and federal courts sitting in King County, Washington.

9. **ENTIRE AGREEMENT.** This Agreement is the entire agreement between you and Microsoft relating to the Specification supersedes all prior or contemporaneous oral or written communications, proposals and representations with respect to the Specification or any other subject matter covered by this Agreement. To the extent the terms of any Microsoft policies or programs for Support Services conflict with the terms of this Agreement, the terms of this Agreement shall control.

1 Requirements

1. The various entries are compliant with the corresponding specification.
2. The table is compatible with an ACPI table format.
3. Support both IPV4 and IPV6 conventions.
4. Support various firmware packaging types:
 - System ROM
 - Adapter ROM
 - Network Boot Program (NBP)
5. The iBFT must remain as compact as possible because it can be located in low memory which is a scarce resource. For example, in an NBP implementation it is possible that between 512k and 640k are the EBDA, UNDI Stack, UNDI, PXE, and the NBP itself.

2 Conventions

IP Address

An example IPV4 address (192.168.70.50) stored in an IPV6 field:

```
IPv4-mapped IPv6 address
UCHAR exampleIP[16] = { 0x00, 0x00, 0x00, 0x00,
                        0x00, 0x00, 0x00, 0x00,
                        0x00, 0x00, 0xff, 0xff,
                        0xc0, 0xA8, 0x46, 0x23 };
```

An IP Address that is not present or not specified shall be all zeros.

See [\[Addr-Arch\]](#), [\[ipv4/6\]](#).

Alignment

Bytes (8-bit entries) are aligned on a byte boundary.

Words (16-bit entries) are aligned on an even byte boundary.

Endianness

All 2-byte entries are little endian "Words" (16-bit quantities) unless otherwise noted.

All 4-byte entries are little endian "DWords" (32-bit quantities) unless otherwise noted.

Length and Offset Values

Length	Offset	Description
0	0	Entry does not exist
0	Non-zero	Empty entry exists

Heap Array Format

A Heap Array entry is a collection of bytes. The total count of bytes of an entry is reflected in the corresponding Length field.

All array items stored in the Heap area will be followed by a separate NULL (a byte with a value of zero). This terminating NULL is not counted as part of the array length.

Parameter Origins

The iBF may obtain the necessary parameters using implementation specific methods. Some examples include all information from DHCP, all information from local non-volatile storage

(NVRAM), some information from DHCP and some from NVRAM, some information from network services such as iSNS.

See [\[iscsi-boot\]](#).

3 iBFT Layout

A compactness approach has been taken in defining the iBFT. Variable length fields are referenced by an offset mechanism.

iBFT Sections

The iBFT is built by using a variety of sections.

Section	Byte Length	Byte Offset	Description
Header	48	0	Primary Header
Control	variable	48	Extended Header
Initiator	variable		Initiator Description
NIC	variable		NIC Description
Target	variable		Target Description
Heap	variable		Storage area for variable length values. String and blob Entries within the iBFT will point into this Heap area.

iBFT Standard Structure Header

Field	Byte Length	Byte Offset	Description
Structure ID	1	0	Structure ID
Version	1	1	Structure Version
Length	2	2	Structure Length
Index	1	4	Index
Flags	1	5	Structure Type Specific

iBF Table Header

Field	Byte Length	Byte Offset	Description
Signature	4	0	'iBFT' Signature for the iSCSI Boot Firmware Table
Length	4	4	Length in bytes of the entire iBFT, including the signature
Revision	1	8	Revision = 1
Checksum	1	9	Entire table must sum to zero
OEMID	6	10	OEM ID. All unused trailing bytes must be zero. A list of OEM IDs is maintained by the ACPI SIG. Contact the ACPI SIG Secretary for OEMID registration.
OEM Table ID	8	16	For the iBFT the Table ID is the Manufacturer's Model ID. All unused trailing bytes must be zero.
Reserved	24	24	Reserved

Locating the iBFT

The iBFT can be located by the Low RAM Method. Scan for the table header signature in system memory between 512K and 1024K. The scan MUST be done starting at the lower address scanning forward to the higher address. When using the Low RAM Method, the table header must be aligned on a 16-byte boundary.

Control Structure

Field	Byte Length	Byte Offset	Description
Structure ID	1	48 (0)	Structure ID = Control
Version	1	49 (1)	Structure Version = 1
Length	2	50 (2)	Structure Length >= 18
Index	1	52 (4)	Index = 0
Flags	1	53 (5)	<p>Bit 0 : Boot Failover Flag</p> <p>0=Connect to all targets specified in the Target structures.</p> <p>1 = Connect to only one target by attempting connection to targets specified in the Target structures one by one.</p> <p>If Boot Failover flag is set, then the iSCSI initiator will only connect to a single target regardless of the number of Target structures. The initiator will try to connect to targets based on the value in the Index field in the Target structure starting with the Target with the lowest Index field value.</p> <p>If Boot Failover flag is not set then the iSCSI initiator will connect to all targets as specified in all of the Target structures.</p> <p>All other bits are reserved and must be 0.</p>
Structure Offsets			
Extensions	2	54 (6)	Optional. If unused must be zero. If used, must point to an Extensions Structure with a standard Structure header.
Initiator Offset	2	56 (8)	
NIC 0 Offset	2	58 (10)	
Target 0 Offset	2	60 (12)	
NIC 1 Offset	2	62 (14)	
Target 1 Offset	2	64 (16)	
Optional: Structure Exp.			

Structure Type / ID

Structure Type/ID:

- 0 = Reserved
- 1 = Control
- 2 = Initiator
- 3 = NIC
- 4 = Target
- 5 = Extensions

Control Structure Offsets

Unused Offsets should be set to zero. For example, if NIC 1 and Target 1 are not used, then the offset values should be zero and no NIC 1 and Target 1 Structures are required to be present.

Optional Structure Expansion

The Control Structure can be expanded beyond the default minimum size. The Structure Length is used to compute the number of additional Optional Structure Offsets. The type of the structure is determined by reading the corresponding Structure ID.

The Structures that are added, if any, do not need sequential Index values. The Index values may be sparse. For example, a single Structure Expansion entry may be added for NIC Index = 5.

Structure Alignment in Memory

Each Structure, if present, must be aligned on an 8-byte boundary.

Initiator Structure

Field	Byte Length	Byte Offset	Description
Structure ID	1	0	Structure ID = Initiator
Version	1	1	Structure Version = 1
Length	2	2	Structure Length = 74
Index	1	4	Index = 0
Flags	1	5	Bit 0 : Block Valid Flag 0 = no, 1=yes Bit 1 : Firmware Boot Selected Flag 0 = no, 1 = yes All other bits are reserved and must be 0.
iSNS Server	16	6	IP Address
SLP Server	16	22	IP Address
Primary Radius Server	16	38	IP Address
Secondary Radius Server	16	54	IP Address
Initiator Name Length	2	70	Heap Entry Length
Initiator Name Offset	2	72	Offset from the beginning of the iBFT

NIC Structure

Field	Byte Length	Byte Offset	Description
Structure ID	1	0	Structure ID = NIC
Version	1	1	Structure Version = 1
Length	2	2	Structure Length = 102
Index	1	4	Index = 0 for NIC 0 Index = 1 for NIC 1 ... Index = n for NIC n
Flags	1	5	Bit 0 : Block Valid Flag 0 = no, 1=yes Bit 1 : Firmware Boot Selected Flag 0 = no, 1 = yes Bit 2 : Global / Link Local 0 = Link Local, 1 = Global All other bits are reserved and must be 0
IP Address	16	6	IP Address
Subnet Mask Prefix	1	22	The mask prefix length. For example, 255.255.255.0 has a prefix length of 24
Origin	1	23	See [origin]
Gateway	16	24	IP Address
Primary DNS	16	40	IP Address
Secondary DNS	16	56	IP Address
DHCP	16	72	IP Address
VLAN	2	88	VLAN
MAC Address	6	90	MAC Address

iSCSI Boot Firmware Table (iBFT) as Defined in ACPI 3.0b Specification - 9

Field	Byte Length	Byte Offset	Description
PCI Bus/Dev/Func	2	96	Bus = 8 bits Device = 5 bits Function = 3 bits
Host Name Length	2	98	Heap Entry Length
Host Name Offset	2	100	Offset from the beginning of the iBFT In a DHCP scenario this can be the name stored as Option 12 host-name.

Host name if supplied must be in one of the following formats:

- <host name>
- <host name>.
- <host name>.<domain>

Target Structure

Field	Byte Length	Byte Offset	Description
Structure ID	1	0	Structure ID = Target
Version	1	1	Structure Version = 1
Length	2	2	Structure Length = 54
Index	1	4	Index = 0 for Target 0 Index = 1 for Target 1 ... Index = n for Target n
Flags	1	5	Bit 0 : Block Valid Flag 0 = no, 1=yes Bit 1 : Firmware Boot Selected Flag 0 = no, 1 = yes Bit 2 : Use Radius CHAP 0 = no, 1 = yes Bit 3 : Use Radius rCHAP 0 = no, 1 = yes All other bits are reserved and must be 0
Target IP Address	16	6	IP Address
Target IP Socket	2	22	Likely 3260
Target Boot LUN	8	24	See [iscsi] Little Endian Quad Word
CHAP Type	1	32	0 = No CHAP 1 = CHAP 2 = Mutual CHAP
NIC Association	1	33	NIC Index
Target Name Length	2	34	Heap Entry Length
Target Name Offset	2	36	Offset from the beginning of the iBFT
CHAP Name Length	2	38	Heap Entry Length
CHAP Name Offset	2	40	Offset from the beginning of the iBFT
CHAP Secret Length	2	42	Heap Entry Length
CHAP Secret Offset	2	44	Offset from the beginning of the iBFT
Reverse CHAP Name Length	2	46	Heap Entry Length
Reverse CHAP Name Offset	2	48	Offset from the beginning of the iBFT
Reverse CHAP Secret Length	2	50	Heap Entry Length
Reverse CHAP Secret Offset	2	52	Offset from the beginning of the iBFT

CHAP – The Name/Password the Initiator sends to the Target.

Reverse CHAP – The Name/Password the Target sends to the Initiator.

4 References

[Addr-Arch] IP Version 6 Addressing Architecture

<http://www.ietf.org/rfc/rfc2373.txt>

[ipv4/6] See “2.1 Addresses”

<http://www.ietf.org/rfc/rfc2765.txt>

[iscsi] Internet Small Computer Systems Interface (iSCSI) RFC 3720

<http://www.ietf.org/rfc/rfc3720.txt>

[iscsi-boot] Bootstrapping Clients...

<http://www.ietf.org/rfc/rfc4173.txt>

[origin] Origin

<http://msdn2.microsoft.com/en-us/library/aa366281.aspx>

Initiator Structure

Field	Byte Length	Byte Offset	Description
Structure ID	1	0	Structure ID = Initiator
Version	1	1	Structure Version = 1
Length	2	2	Structure Length = 74
Index	1	4	Index = 0
Flags	1	5	Bit 0 : Block Valid Flag 0 = no, 1=yes Bit 1 : Firmware Boot Selected Flag 0 = no, 1 = yes
iSNS Server	16	6	IP Address
SLP Server	16	22	IP Address
Primary Radius Server	16	38	IP Address
Secondary Radius Server	16	54	IP Address
Initiator Name Length	2	70	Heap Entry Length
Initiator Name Offset	2	72	Offset from the beginning of the iBFT

NIC Structure

Field	Byte Length	Byte Offset	Description
Structure ID	1	0	Structure ID = NIC
Version	1	1	Structure Version = 1
Length	2	2	Structure Length = 102
Index	1	4	Index = 0 for NIC 0 Index = 1 for NIC 1 ... Index = n for NIC n
Flags	1	5	Bit 0 : Block Valid Flag 0 = no, 1=yes Bit 1 : Firmware Boot Selected Flag 0 = no, 1 = yes Bit 2 : Global / Link Local 0 = Link Local, 1 = Global
IP Address	16	6	IP Address

Field	Byte Length	Byte Offset	Description
Subnet Mask Prefix	1	22	The mask prefix length. For example, 255.255.255.0 has a prefix length of 24
Origin	1	23	See [origin]
Gateway	16	24	IP Address
Primary DNS	16	40	IP Address
Secondary DNS	16	56	IP Address
DHCP	16	72	IP Address
VLAN	2	88	VLAN
MAC Address	6	90	MAC Address
PCI Bus/Dev/Func	2	96	Bus = 8 bits Device = 5 bits Function = 3 bits
Host Name Length	2	98	Heap Entry Length
Host Name Offset	2	100	Offset from the beginning of the iBFT In a DHCP scenario this can be the name stored as Option 12 host-name.

Host name if supplied must be in one of the following formats:

- <host name>
- <host name>.
- <host name>.<domain>

Target Structure

Field	Byte Length	Byte Offset	Description
Structure ID	1	0	Structure ID = Target
Version	1	1	Structure Version = 1
Length	2	2	Structure Length = 54
Index	1	4	Index = 0 for Target 0 Index = 1 for Target 1 ... Index = n for Target n
Flags	1	5	Bit 0 : Block Valid Flag 0 = no, 1=yes Bit 1 : Firmware Boot Selected Flag 0 = no, 1 = yes Bit 2 : Use Radius CHAP 0 = no, 1 = yes Bit 3 : Use Radius rCHAP 0 = no, 1 = yes
Target IP Address	16	6	IP Address
Target IP Socket	2	22	Likely 3260
Target Boot LUN	8	24	See [iscsi] Little Endian Quad Word
CHAP Type	1	32	0 = No CHAP 1 = CHAP 2 = Mutual CHAP
NIC Association	1	33	NIC Index
Target Name Length	2	34	Heap Entry Length
Target Name Offset	2	36	Offset from the beginning of the iBFT
CHAP Name Length	2	38	Heap Entry Length
CHAP Name Offset	2	40	Offset from the beginning of the iBFT

Field	Byte Length	Byte Offset	Description
CHAP Secret Length	2	42	Heap Entry Length
CHAP Secret Offset	2	44	Offset from the beginning of the iBFT
Reverse CHAP Name Length	2	46	Heap Entry Length
Reverse CHAP Name Offset	2	48	Offset from the beginning of the iBFT
Reverse CHAP Secret Length	2	50	Heap Entry Length
Reverse CHAP Secret Offset	2	52	Offset from the beginning of the iBFT

CHAP – The Name/Password the Initiator sends to the Target.

Reverse CHAP – The Name/Password the Target sends to the Initiator.

5 References

[Addr-Arch] IP Version 6 Addressing Architecture

<http://www.ietf.org/rfc/rfc2373.txt>

[ipv4/6] See “2.1 Addresses”

<http://www.ietf.org/rfc/rfc2765.txt>

[iscsi] Internet Small Computer Systems Interface (iSCSI) RFC 3720

<http://www.ietf.org/rfc/rfc3720.txt>

[iscsi-boot] Bootstrapping Clients...

<http://www.ietf.org/rfc/rfc4173.txt>

[origin] Origin

<http://msdn2.microsoft.com/en-us/library/aa366281.aspx>