Document Number: DSP1012

Date: 2009-06-22

Version: 1.0.1

Boot Control Profile

Document Type: Specification

Document Status: DMTF Standard

Document Language: E

Contents

List of Figures

List of Tables

Foreword

The *Boot Control Profile* (DSP1012) was prepared by the Physical Platform Profiles Working Group.

DMTF is a not-for-profit association of industry members dedicated to promoting enterprise and systems management and interoperability.

Acknowledgments

The authors wish to acknowledge the following people.

Editors:

- John Leung – Intel
- David Simpson – IBM
- Khachatur Papanyan – Dell

Contributors:

- Aaron Merkin – IBM
- Jon Hass – Dell
- Khachatur Papanyan – Dell
- Enoch Suen – Dell
- Jeff Hilland – HP
- Hemal Shah – Broadcom
- Christina Shaw – HP
- Perry Vincent – Intel
- Arvind Kumar – Intel

Introduction

The information in this specification should be sufficient for a provider or consumer of this data to unambiguously identify the classes, properties, methods, and values that shall be instantiated and manipulated to represent and manage the boot control configurations of a computer server using the DMTF CIM core and extended model definitions.

The target audience for this specification is implementers who are writing CIM-based providers or consumers of management interfaces representing the components described in this document.

Boot Control Profile

1 Scope

The *Boot Control Profile* describes the classes, associations, properties, and methods used to manage the boot control configurations of a physical or virtual computer system.

2 Normative References

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

2.1 Approved References

DMTF DSP0004, *CIM Infrastructure Specification 2.5*,
http://www.dmtf.org/standards/published_documents/DSP0004_2.5.pdf

DMTF DSP0200, *CIM Operations over HTTP 1.2*,
http://www.dmtf.org/standards/published_documents/DSP0200_1.2.pdf

DMTF DSP1001, *Management Profile Specification Usage Guide 1.0*,
http://www.dmtf.org/standards/published_documents/DSP1001_1.0.pdf

DMTF DSP1033, *Profile Registration Profile 1.0*,
http://www.dmtf.org/standards/published_documents/DSP1033_1.0.pdf

2.2 Other References

ISO/IEC Directives, Part 2, *Rules for the structure and drafting of International Standards*,
http://isotc.iso.org/livelink/livelink.exe?func=ll&objId=4230456&objAction=browse&sort=subtype

BIOS Boot Specification 1.01 (January 11, 1996),
http://www.phoenix.com/NR/rdonlyres/56E38DE2-3E6F-4743-835F-B4A53726ABED/0/specsbbs101.pdf

3 Terms and Definitions

3.1
can
used for statements of possibility and capability, whether material, physical, or causal

3.2
cannot
used for statements of possibility and capability, whether material, physical, or causal

3.3
conditional
used to indicate requirements strictly to be followed, in order to conform to the document when the specified conditions are met

3.4

mandatory

used to indicate requirements strictly to be followed, in order to conform to the document and from which no deviation is permitted

3.5

may

used to indicate a course of action permissible within the limits of the document

3.6

need not

used to indicate a course of action permissible within the limits of the document

3.7

optional

used to indicate a course of action permissible within the limits of the document

3.8

referencing profile

indicates a profile that owns the definition of a class used, but not defined, in this document and can be included in the "Referenced Profiles" table

3.9

shall

used to indicate requirements strictly to be followed, in order to conform to the document and from which no deviation is permitted

3.10

shall not

used to indicate requirements strictly to be followed, in order to conform to the document and from which no deviation is permitted

3.11

should

used to indicate that among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required

3.12

should not

used to indicate that a certain possibility or course of action is deprecated but not prohibited

3.13

unspecified

indicates that this profile does not define any constraints for the referenced CIM element or operation

3.14

Boot Configurable System

an instance of CIM_ComputerSystem whose boot configurations are being managed

3.15

Boot Configuration

a collection of settings that are applied to a boot configurable system during the boot process

3.16

Boot Configuration Representation

the CIM representation of a boot configuration, which consists of an instance of class CIM_BootConfigSetting and, optionally, all of the instances of classes CIM_BootSourceSetting, CIM_BootSettingData and CIM_SettingData that it directly or indirectly aggregates

3.17

Current Boot Configuration

the instance of CIM_BootConfigSetting that was used the last time the managed system was successfully booted

3.18

Default Boot Configuration

the instance of CIM_BootConfigSetting that the computer system manufacturer or a client has informatively tagged as its default boot configuration

3.19

Next Boot Configuration

the instance of CIM_BootConfigSetting that will be used during the next boot of the Boot Configurable System

3.20

Next Single Use Boot Configuration

the instance of CIM_BootConfigSetting that will only be used during the next boot of the Boot Configurable System and then not used again

3.21

Not Next Boot Configuration

an instance of CIM_BootConfigSetting that will not be used during the next boot

3.22

Template Boot Configuration

an existing instance of CIM_BootConfigSetting that is to be used as the template for creating a new boot configuration

4 Symbols and Abbreviated Terms

4.1

BCV

Boot Control Vector. See the *BIOS Boot Specification* for additional information.

4.2

IPL

Initial Program Load. See the *BIOS Boot Specification* for additional information.

4.3

PXE

Preboot Execution Environment. See the *BIOS Boot Specification* for additional information.

5 Synopsis

Profile Name: *Boot Control*

Version: 1.0.1

Organization: DMTF

CIM Schema Version: 2.19

Central Class: CIM_BootService

Scoping Class: CIM_ComputerSystem

The *Boot Control Profile* extends the management capabilities of referencing profiles by adding the capability to represent and manage boot configurations that include boot devices and settings for use during booting.

Table 1 identifies profiles on which this profile has a dependency.

CIM_BootService shall be the Central Class of this profile. The instance of CIM_BootService shall be the Central Instance of this profile.

CIM_ComputerSystem shall be the Scoping Class of this profile. The instance of CIM_ComputerSystem with which the Central Instance is associated through an instance of CIM_HostedService shall be the Scoping Instance of this profile.

Table 1 – Related Profiles

Profile Name	Organization	Version	Relationship
Profile Registration	DMTF	1.0	Mandatory

6 Description

The *Boot Control Profile* describes the elements needed to provide the capability to manage the boot configurations of a computer system.

The profile could manage the following capabilities of a typical computer system:

- A computer system can have one or more boot configurations.
- A computer system can contain a boot configuration that is used during each boot.
- A computer system can contain a single-use boot configuration that is used only during the next boot and then not used again.
- A computer system can contain a current boot configuration that represents the boot configuration successfully used in the last boot.
- A computer system can contain a default boot configuration that is set by the computer system manufacturer or a client.
- A computer system can create new boot configurations.
- A computer system can apply a boot configuration to an active or inactive computer system.

A typical boot configuration could have the following characteristics:

- A boot configuration can contain a boot order that specifies the order in which boot devices are accessed. The boot devices include, but are not limited to, floppy device, CD device, hard disks, network controllers (using the PXE protocol), and BCV devices composed of additional boot sources.
- A boot configuration can contain data that can affect various computer system components during the boot process.
- A boot configuration can contain data that can be passed to the booted image (for example, second-stage boot loader or bootblock) in the form of a boot string.
- Boot devices can be local to the computer system or remote to the computer system.

A boot configuration can be applied when the computer system starts the boot process. The boot process can be started automatically as part of the enablement of the computer system or by a specific request when the computer system is enabled but not booted.

6.1 Class Diagram

Figure 1 represents the class schema for the *Boot Control Profile*. For simplicity, the prefix CIM_ has been removed from the name of the classes.

In Figure 1, CIM_ManagedElement, CIM_LogicalDevice, CIM_SettingData, and CIM_BootSettingData are abstract classes.

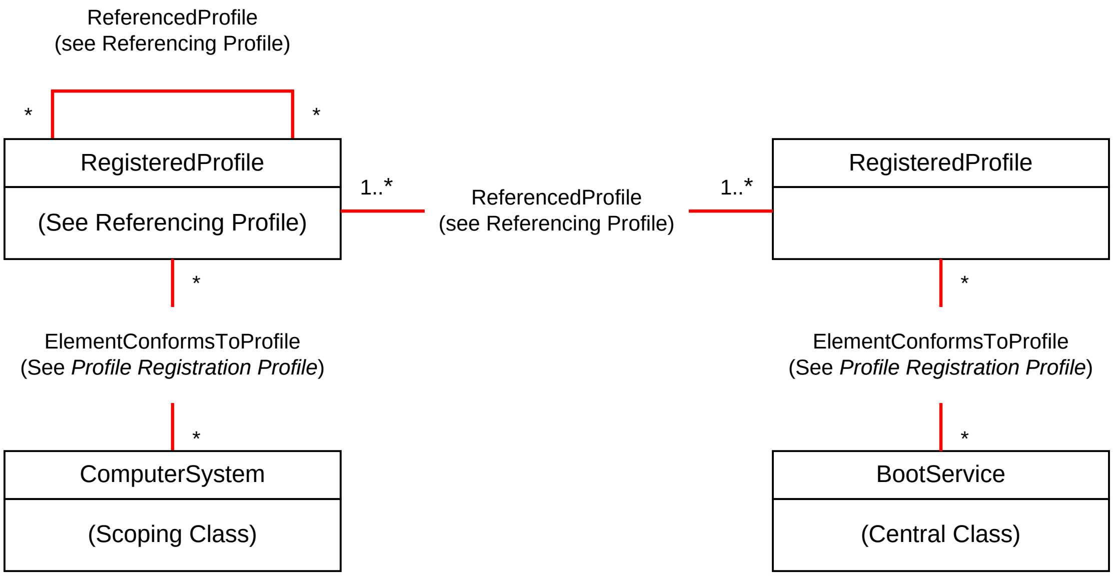


Figure 1 – Boot Control Profile: Class Diagram

A computer system can have multiple boot configurations. Each boot configuration is modeled by a Boot Configuration Representation, which consists of an instance of CIM_BootConfigSetting class and, optionally, all of the instances of classes CIM_BootSourceSetting, CIM_BootSettingData and CIM_SettingData that the instance of CIM_BootConfigSetting aggregates

The usage of each Boot Configuration Representation during the boot process is determined by the IsNext property of the CIM_ElementSettingData association between the Boot Configuration Representation and Boot Configurable System whose boot configuration is being managed.

Each Boot Configuration Representation contains an ordered list of boot sources, which indicate the logical devices to use during the boot process. The boot order is defined by interpreting a property in the CIM_OrderedComponent association between the instance of CIM_BootConfigSetting representing a boot configuration and instances of CIM_BootSourceSetting representing the boot sources.

In some cases a single boot source might, in turn, represent additional ordered boot sources. This set of aggregated boot sources is represented by an instance of CIM_BootConfigSetting, which is associated to the instance of CIM_BootSourceSetting through an instance of CIM_LogicalIdentity.

Settings that apply to a managed element during the boot process are represented by instances of a concrete subclass of the CIM_SettingData class.

Settings that apply to the boot process, itself, are represented by instances of a concrete subclass of the CIM_BootSettingData class.

These settings can apply to either the entire boot configuration or to a specific boot source within a boot configuration. This scoping is determined by traversing the CIM_ConcreteComponent association to either an instance of CIM_BootConfigSetting representing the boot configuration or CIM_BootSourceSetting representing the boot source, respectively.

7 Implementation

This clause contains normative information about the model and the relationship between the model and underlying instrumentation. Normative text for properties is included in this clause. Normative text for methods is contained in clause 8.

7.1 CIM_BootService

At least one instance of the Central Class, CIM_BootService, shall exist.

7.1.1 CIM_BootService.ElementName

ElementName shall be formatted as a free-form string of variable length (pattern “.*”).

7.1.2 Modifying ElementName Is Supported

Subclause 7.1.2 describes conditional behavior. Subclause 7.1.2 describes the CIM elements and behaviors that shall be implemented when the following conditions are met.

Conditional Requirement:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

2) The CIM_BootServiceCapabilities.ElementNameEditSupport property has the value of TRUE.

3) The CIM_BootServiceCapabilities.MaxElementNameLen property has a non-zero value

The implementation shall allow the CIM_BootService.ModifyInstance intrinsic operation to change the value of the ElementName property. The ModifyInstance operation shall enforce the length restriction specified in the MaxElementNameLen property.

7.1.3 Modifying ElementName Is Not Supported

Subclause 7.1.3 describes conditional behavior, Subclause 7.1.3 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement 1:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.
2) The CIM_BootServiceCapabilities. ElementNameEditSupport property has the value of FALSE.

Conditional Requirement 2:

1) An instance of CIM_BootServiceCapabilities is not associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

The implementation shall not allow the CIM_BootService.ModifyInstance intrinsic operation to change the value of the ElementName property.

7.2 CIM_ComputerSystem

An instance of CIM_ComputerSystem shall represent either a Scoping Instance or a Boot Configurable System, or both. The Scoping Instance is used to determine profile conformance. The Boot Configurable System represents a computer system whose boot configurations are being managed.

One Scoping Instance shall exist. Clause 5 describes the process for determining the Scoping Instance from the Central Instance.

Each instance of CIM_ComputerSystem representing a Boot Configurable System shall be associated to the Central Instance through an instance of the CIM_ServiceAffectsElement association. At least one instance of a Boot Configurable System shall exist.

7.3 Representing Boot Service Capabilities

Subclause 7.3 describes optional behavior.

An instance of CIM_BootServiceCapabilities may exist, which represents the capabilities of the boot service.

If an instance of CIM_BootServiceCapabilities is instantiated, then it shall be associated with an instance of CIM_BootService using an instance of CIM_ElementCapabilities.

7.3.1 Representing Implementation Specific Boot Service Capabilities

Subclause 7.3.1 describes optional behavior.

An implementation may identify method-related boot configuration capabilities, other than those explicitly defined in this profile, by setting the BootConfigCapabilities and OtherBootConfigCapabilities property arrays of the CIM_BootServiceCapabilities class.

The additional boot configuration capability shall be identified by setting an entry in the CIM_BootServiceCapabilities.BootConfigCapabilities property array to a value of 1 (Other) for each additional boot configuration capability.

For each entry in the BootConfigCapabilities array property with the value 1 (Other), the corresponding entry in the CIM_BootServiceCapabilities.OtherBootConfigCapabilities array property shall contain a non-NULL, non-empty string that provides a short description of the capability.

7.4 Boot Configurations

An instance of CIM_BootConfigSetting shall represent a boot configuration that may be used during the boot process.

Each Boot Configurable System shall have at least one instance of CIM_BootConfigSetting associated to it through an instance of CIM_ElementSettingData.

7.4.1 CIM_ElementSettingData

An instance of CIM_ElementSettingData shall be used to associate each instance of CIM_BootConfigSetting, representing a boot configuration, to each instance of CIM_ComputerSystem, representing a Boot Configurable System to which the boot configuration applies.

When the CIM_ElementSettingData association is used in this manner, its ManagedElement property shall reference the CIM_ComputerSystem instance and its SettingData property shall reference the CIM_BootConfigSetting instance.

For an instance of CIM_ElementSettingData, the IsNext property shall determine how the associated instance of CIM_BootConfigSetting is used, if at all, during the boot of the Boot Configurable System.

7.4.2 Default Boot Configuration

Subclause 7.4.2 describes optional behavior.

The Default Boot Configuration is the instance of CIM_BootConfigSetting that the computer system manufacturer or a client has informatively tagged as the default configuration for the Boot Configurable System. The Default Boot Configuration does not impact which boot configuration applies during the boot process.

The Default Boot Configuration shall be the instance of CIM_BootConfigSetting that is associated by the instance of CIM_ElementSettingData when the IsDefault property has a value of 1 (Is Default).

For a given Boot Configurable System, at most one Default Boot Configuration shall be associated. The IsDefault property of instances of CIM_ElementSettingData associating the Boot Configurable System to all other Boot Configuration Representations shall have a value of 2 (Is Not Default).

7.4.3 Current Boot Configuration

Subclause 7.4.3 describes optional behavior.

The Current Boot Configuration is the instance of CIM_BootConfigSetting that was used the last time the system represented by the Boot Configurable System was successfully booted.

The Current Boot Configuration shall be the instance of CIM_BootConfigSetting that is associated by the instance of CIM_ElementSettingData when the IsCurrent property has a value of 1 (Is Current).

For a given Boot Configurable System, zero or one Current Boot Configuration shall be associated. The IsCurrent property of instances of CIM_ElementSettingData associating the Boot Configurable System to all other Boot Configuration Representations shall have a value of 2 (Is Not Current).

An implementation may support the Current Boot Configuration when it is able to determine the configuration last used during a successful boot. When an implementation supports the Current Boot Configuration, the Current Boot Configuration shall exist after a successful boot.

7.4.4 Next Boot Configuration

Subclause 7.4.4 describes optional behavior.

NOTE: Successful execution of the ApplyBootConfigSetting() method can independently apply a boot configuration to a Boot Configurable System regardless of the Next Boot Configuration. The requirements in this subclause shall not apply when a Boot Configurable System is booted using the ApplyBootConfigSetting() method.

The Next Boot Configuration is the instance of CIM_BootConfigSetting that shall be used during the next boot of the system represented by the Boot Configurable System, unless there is a Next Single Use Boot Configuration associated to the same Boot Configurable System.

The Next Boot Configuration shall be the instance of CIM_BootConfigSetting that is associated by the instance of CIM_ElementSettingData when the IsNext property has a value of 1 (Is Next).

For a given Boot Configurable System, at most one Next Boot Configuration shall be associated.

7.4.5 Next Single Use Boot Configuration

Subclause 7.4.5 describes optional behavior.

NOTE: Successful execution of the ApplyBootConfigSetting() method can independently apply a boot configuration to a Boot Configurable System regardless of the Next Single Use Boot Configuration. The requirements in this subclause shall not apply when a Boot Configurable System is booted using the ApplyBootConfigSetting() method.

The Next Single Use Boot Configuration is the instance of CIM_BootConfigSetting that shall only be used during the next boot of the system represented by the Boot Configurable System.

When a Next Boot Configuration is also associated to the Boot Configurable System, the Next Single Use Boot Configuration shall take precedence over the Next Boot Configuration.

Upon a successful usage during a boot, the Next Single Use Boot Configuration shall become a Not Next Boot Configuration.

The Next Single Use Boot Configuration shall be the instance of CIM_BootConfigSetting that is associated by the instance of CIM_ElementSettingData when the IsNext property has a value of 3 (Is Next For Single Use).

For a given Boot Configurable System, there shall be at most one Next Single Use Boot Configuration associated.

7.4.6 Not Next Boot Configuration

The Not Next Boot Configuration is an instance of CIM_BootConfigSetting that will not be used during the next boot.

The Not Next Boot Configuration shall be a CIM_BootConfigSetting whose CIM_ElementSettingData.IsNext property has the value of 2 (Is Not Next).

7.5 Applying the Boot Configuration

The CIM_BootService associated to the Boot Configurable System may support the explicit application of a Boot Configuration Representation through the ApplyBootConfigSetting() method.

NOTE: Successful execution of the ApplyBootConfigSetting() method can independently apply a boot configuration to a Boot Configurable System regardless of the Next Boot Configuration. The requirements in subclause 7.4.4 shall not apply when a Boot Configurable System is booted using the ApplyBootConfigSetting() method.

7.5.1 Apply Boot Configuration Is Supported

Subclause 7.5.1 describes conditional behavior. Subclause 7.5.1 describes the CIM elements and behaviors that shall be implemented when the following conditions are met.

Conditional Requirement:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

2) The CIM_BootServiceCapabilities.BootConfigCapabilities property array contains a value of 3 (Applies Boot Configuration).

The implementation shall support the CIM_BootService.ApplyBootConfigSetting() method.

7.5.2 Apply Boot Configuration Is Not Supported

Subclause 7.5.2 describes conditional behavior. Subclause 7.5.2 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement 1:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

2) The CIM_BootServiceCapabilities.BootConfigCapabilities property array does not contain a value of 3 (Applies Boot Configuration).

Conditional Requirement 2:

1) An instance of CIM_BootServiceCapabilities is not associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

The implementation shall not support the CIM_BootService.ApplyBootConfigSetting() method.

When a Boot Configurable System, that is not associated to a Next Boot Configuration or Next Single Use Boot Configuration, transitions to the Enabled state, then the normal boot process shall be initiated.

7.6 Creating a Boot Configuration

The CIM_BootService may support the client creation of a new boot configuration from an existing boot configuration through the CreateBootConfigSetting() method.

7.6.1 Creating Boot Configuration Is Supported

Subclause 7.6.1 describes conditional behavior. Subclause 7.6.1 describes the CIM elements and behaviors that shall be implemented when the following conditions are met.

Conditional Requirement:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

2) The CIM_BootServiceCapabilities.BootConfigCapabilities property array contains a value of 2 (Creates Boot Configuration).

The implementation shall support the CreateBootConfigSetting() method.

7.6.2 Creating Boot Configuration Is Not Supported

Subclause 7.6.2 describes conditional behavior. Subclause 7.6.2 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement 1:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

2) The CIM_BootServiceCapabilities.BootConfigCapabilities property array does not contain a value of 2 (Creates Boot Configuration).

Conditional Requirement 2:

1) An instance of CIM_BootServiceCapabilities is not associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

When either of the preceding conditions are met, the implementation shall not support the CreateBootConfigSetting() method.

7.7 Deleting a Boot Configuration

Subclause 7.7 describes conditional behavior.

Conditional Requirement: The implementation shall support the client deleting or removing an existing boot configuration through the DeleteInstance() intrinsic operation, when the implementation supports the creation of a new boot configuration.

This conditional behavior shall be determined with the same mechanism used to determine that an implementation supports the creation of a new boot configuration. See subclause 7.6.

7.8 Identifying Boot Sources

Subclause 7.8 describes optional behavior.

An instance of CIM_BootSourceSetting represents a source from which a boot image can be loaded during the boot process.

An instance of CIM_BootSourceSetting shall be associated to one or more instances of CIM_BootConfigSetting.

The CIM_BootSourceSetting class has three boot string properties: BootString, BIOSBootString and StructuredBootString. The BootString and BIOSBootString properties may be supported. The StructuredBootString property should be supported.

7.8.1 CIM_BootServiceCapabilities

When no instance of CIM_BootServiceCapabilities exists, it is not possible to determine, via the CIM_BootServiceCapabilities, which boot string properties are supported.

7.8.1.1 CIM_BootServiceCapabilities.BootStringsSupported

When an instance of CIM_BootServiceCapabilities exists, its BootStringsSupported property array shall contain one or more of the values 1 (BootString), 2 (BIOSBootString) and 3 (StructuredBootString).

The presence of a value in the property array means that the specified boot string in each instance of CIM_BootSourceSettings which are associated to an instance of CIM_BootConfigSetting, which in turn is associated to the CIM_BootService, shall not be NULL.

7.8.2 CIM_BootSourceSetting.ElementName Property

The CIM_BootSourceSetting.ElementName property shall be a character string of variable length (pattern ".*").

The ElementName property shall contain a string that identifies the boot source.

When the CIM_BootSourceSetting.BIOSBootString property is not null, the ElementName property shall match the BIOSBootString property.

7.8.3 CIM_BootSourceSetting.BootString Property

An implementation may support the CIM_BootSourceSetting.BootString property.

7.8.3.1 CIM_BootSourceSetting.BootString Property Is Supported

Subclause 7.8.3.1 describes conditional behavior. Subclause 7.8.3.1 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.
2) The CIM_BootServiceCapabilities.BootStringsSupported property array contains a value of 2 (BootString).

The CIM_BootSourceSetting.BootString property shall contain a character string.

The CIM_BootSourceSetting.BootString property shall contain a string that identifies the boot source. The property may include additional information to be used during the boot process. Examples include a specific address of a bootable partition, flags to request the loading of a kernel debugger, or name of the kernel image.

7.8.3.2 CIM_BootSourceSetting.BootString Property Is Not Supported

Subclause 7.8.3.2 describes conditional behavior. Subclause 7.8.3.2 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement 1:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.
2) The CIM_BootServiceCapabilities.BootStringsSupported property array does not contain a value of 2 (BootString).

Conditional Requirement 2:

1) An instance of CIM_BootServiceCapabilities is not associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

The CIM_BootSourceSetting.BootString property may be NULL.

7.8.4 CIM_BootSourceSetting.BIOSBootString Property

An implementation may support the CIM_BootSourceSetting.BIOSBootString property.

7.8.4.1 CIM_BootSourceSetting.BIOSBootString Property Is Supported

Subclause 7.8.4.1 describes conditional behavior. Subclause 7.8.4.1 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

2) The CIM_BootServiceCapabilities.BootStringsSupported property array contains a value of 3 (BIOSBootString).

The CIM_BootSourceSetting.BIOSBootString property shall contain a character string of variable length (pattern “.*”).

The CIM_BootSourceSetting.BIOSBootString property shall contain a string that identifies the boot source. The property shall match the string used by the BIOS to uniquely name the boot source in its namespace.

For an UEFI BIOS, the BIOSBootString property should match the output of the EFI_DEVICE_PATH_TO_TEXT_PROTOCOL service.

7.8.4.2 CIM_BootSourceSetting.BIOSBootString Property Is Not Supported

Subclause 7.8.4.2 describes conditional behavior. Subclause 7.8.4.2 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement 1:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

2) The CIM_BootServiceCapabilities.BootStringsSupported property array does not contain a value of 3 (BIOSBootString).

Conditional Requirement 2:

1) An instance of CIM_BootServiceCapabilities is not associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

The CIM_BootSourceSetting.BIOSBootString property may be NULL.

7.8.5 CIM_BootSourceSetting.StructuredBootString Property

An implementation should support the CIM_BootSourceSetting.StructuredBootString property.

7.8.5.1 CIM_BootSourceSetting.StructuredBootString Property Is Supported

Subclause 7.8.5.1 describes conditional behavior. Subclause 7.8.5.1 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

2) The CIM_BootServiceCapabilities.BootStringsSupported property array contains a value of 4 (StructuredBootString).

The CIM_BootSourceSetting.StructuredBootString property shall contain a string that identifies the boot source using the following format:

```
"<OrgID>:<identifier>:<index>"
```

The value of <OrgID> shall include a copyrighted, trademarked or otherwise unique name that is owned by the entity creating or defining the CIM_BootSourceSetting, or is a registered ID that is assigned to the entity by a recognized global authority. In addition, <OrgID> shall not contain a colon (:). For DMTF defined instances, the algorithm shall be used with the <OrgID> set to "CIM".

The value of <index> shall be an unsigned integer. When the value of <OrgID> matches "CIM", the value of the <identifier> shall be one of the identifiers listed in Table 2.

Table 2 – Structured Name Identifiers

Identifier	Description
“Unknown”	The boot device type is unknown
"Floppy"	Boot from a floppy device
"Hard-Disk"	Boot from a hard drive device
"CD/DVD"	Boot from a CD or DVD device
"Network"	Boot from a network device
"PCMCIA"	Boot from a PCMCIA device
"BEV"	Boot from a Boot Entry Vector device
"USB"	Boot from a USB device

7.8.5.2 CIM_BootSourceSetting.StructuredBootString Property Is Not Supported

Subclause 7.8.5.2 describes conditional behavior. Subclause 7.8.5.2 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement 1:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

2) The CIM_BootServiceCapabilities.BootStringsSupported property array does not contain a value of 4 (StructuredBootString).

Conditional Requirement 2:

1) An instance of CIM_BootServiceCapabilities is not associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

The CIM_BootSourceSetting.StructuredBootString property may be NULL.

7.8.6 CIM_ConcreteDependency Association

An instance of a concrete subclass of CIM_LogicalDevice may exists, which represents the boot source device.

If such an instance of CIM_LogicalDevice is instantiated, then it shall be associated with an instance of CIM_BootSourceSetting using an instance of CIM_ConcreteDependency.

When the association is used in this manner, its Antecedent property shall reference the instance of a concrete subclass of CIM_LogicalDevice and its Dependent property shall reference the CIM_BootSourceSetting instance.

7.9 Changing the Boot Order

A Boot Configuration Representation may support the client changing the boot order of the boot sources associated to an instance of CIM_BootConfigSetting through the CIM_BootConfigSetting.ChangeBootOrder() method.

7.9.1 Changing Boot Order Is Supported

Subclause 7.9.1 describes conditional behavior. Subclause 7.9.1 describes the CIM elements and behaviors that shall be implemented when either of the following conditions are met.

Conditional Requirement 1:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.
2) The CIM_BootServiceCapabilities.BootConfigCapabilities property array does not contain a value of 6 (Change Boot Order Not Supported).

Conditional Requirement 2:

1) An instance of CIM_BootServiceCapabilities is not associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.

When either of the preceding conditions are met, the implementation shall support the ChangeBootOrder() method.

7.9.2 Changing Boot Order Is Not Supported

Subclause 7.9.2 describes conditional behavior. Subclause 7.9.2 describes the CIM elements and behaviors that shall be implemented when the following conditions are met.

Conditional Requirement:

1) An instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance through an instance of CIM_ElementCapabilities.
2) The CIM_BootServiceCapabilities.BootConfigCapabilities property array contains a value of 6 (Change Boot Order Not Supported).

The implementation shall not support the ChangeBootOrder() method.

7.10 Representing a Set of Aggregated Boot Sources

Subclause 7.9 describes optional behavior.

An instance of CIM_BootSourceSetting may represent an aggregated boot source. An example of an aggregated boot source is a BCV.

When an aggregated boot source is represented, it shall be associated to a representation of the set of aggregated boot sources. The following requirements shall apply.

7.10.1 Aggregated Boot Sources

An instance of CIM_BootSourceSetting shall exist representing an aggregated boot source.

7.10.2 Aggregated Boot Configuration

An instance of CIM_BootConfigSetting shall exist representing the set of aggregated boot sources.

The ElementName property for the instance of CIM_BootConfigSetting representing the set of aggregated boot sources shall match the value of the ElementName property of the instance of CIM_BootSourceSetting that represents the aggregated boot source.

7.10.3 Logical Identity Relationship

An instance of CIM_LogicalIdentity shall associate the instance of CIM_BootSourceSetting with the instance of CIM_BootConfigSetting.

7.10.3.1 CIM_LogicalIdentity.SystemElement

The value of the SystemElement reference shall be the instance of CIM_BootSourceSetting that represents the aggregated boot source.

7.10.3.2 CIM_LogicalIdentity.SameElement

The value of the SameElement reference shall be the instance of CIM_BootConfigSetting that represents the set of aggregated boot sources.

7.11 Boot Order During the Boot Process

Subclause 7.11 describes the CIM elements and behaviors that shall be implemented to define the order or sequence in which the boot sources are used during the boot process.

7.11.1 CIM_OrderedComponent Association

The CIM_OrderedComponent association class shall be used to associate instance of CIM_BootConfigSetting to each instance of CIM_BootSourceSetting representing one of the boot sources in the boot configuration.

When the association is used in this manner, its GroupComponent property shall reference the CIM_BootConfigSetting instance and its PartComponent property shall reference the CIM_BootSourceSetting instance.

7.11.1.1 CIM_OrderedComponent.AssignedSequence Property

When a CIM_BootConfigSetting instance has multiple CIM_BootSourceSetting instances associated to it through instances of the CIM_OrderedComponent association, the value of the CIM_OrderedComponent.AssignedSequence property shall be used to determine the sequence in which the associated CIM_BootSourceSetting instances are used during the boot process.

The value of the AssignedSequence property across instances of CIM_OrderedComponent that reference the same CIM_BootConfigSetting shall be unique when it is not equal to zero.

The boot order shall be interpreted as follows:

- The AssignedSequence properties are compared across instances of CIM_OrderedComponent that reference the same CIM_BootConfigSetting.
- A CIM_BootSourceSetting whose associated CIM_OrderedComponent.AssignedSequence property is equal to zero shall be ignored and not considered part of the boot order.
- The boot order shall proceed from the lowest to the highest non-zero integer value of the AssignedSequence properties.

7.11.2 CIM_BootSourceSetting.FailThroughSupported

The FailThroughSupported property shall describe the behavior of the boot process when the attempt to boot from a boot device represented by an instance of CIM_BootSourceSetting is not successful.

When the FailThroughSupported property has a value of 1 (Is Supported), an unsuccessful boot attempt shall result in continuing through the ordered list for boot sources from which to attempt to boot.

When the FailThroughSupported property has a value of 2 (Is Not Supported), then an unsuccessful boot attempt shall result in the termination of the boot order for the remaining instances of CIM_BootSourceSetting associated to the same instance of CIM_BootConfigSetting.

7.12 Settings to Apply During the Boot Process

Subclause 7.12 describes optional behavior. Subclause 7.12 describes the CIM elements and behaviors that may be implemented to apply settings during the boot process.

During the boot process, settings can be applied to managed elements or the boot process itself. A setting can be applicable to an entire configuration or to a specific boot source.

7.12.1 Settings that Apply to a Managed Element

An instance of a concrete subclass of CIM_SettingData represents a setting that is applied to a managed element during the boot process. The instance shall be associated to either an instance of CIM_BootConfigSetting or an instance of CIM_BootSourceSetting through an instance of CIM_ConcreteComponent.

When a setting to a managed element is applicable to an entire boot configuration, an instance of a concrete subclass of CIM_SettingData shall be associated to the instance of CIM_BootConfigSetting representing the boot configuration through an instance of CIM_ConcreteComponent.

When the CIM_ConcreteComponent association is used in this manner, its GroupComponent property shall reference the CIM_BootConfigSetting instance and its PartComponent property shall reference the CIM_SettingData instance.

When a setting to a managed element is applicable to a specific boot source, an instance of a concrete subclass of CIM_SettingData shall be associated to the instance of CIM_BootSourceSetting representing the boot configuration through an instance of CIM_ConcreteComponent.

When the CIM_ConcreteComponent association is used in this manner, its GroupComponent property shall reference the CIM_BootSourceSetting instance and its PartComponent property shall reference the CIM_SettingData instance.

7.12.2 Settings that Apply to the Boot Process

An instance of a concrete subclass of CIM_BootSettingData represents a setting that is applied during the boot process but does not apply to a managed element. The setting can apply to an entire boot configuration or to a specific boot source.

When an instance of CIM_BootSettingData is instantiated, then it shall be associated with an instance of CIM_BootConfigSetting or CIM_BootSourceSetting using an instance of CIM_ConcreteComponent.

When the CIM_ConcreteComponent association is used in this manner, its GroupComponent property shall reference the CIM_BootConfigSetting or CIM_BootSourceSetting instance and its PartComponent property shall reference the CIM_BootSettingData instance.

When an instance of a concrete subclass of CIM_SettingData is instantiated, then it shall be associated with an instance of CIM_BootConfigSetting or CIM_BootSourceSetting using an instance of CIM_ConcreteComponent.

When the CIM_ConcreteComponent association is used in this manner, its GroupComponent property shall reference the CIM_BootConfigSetting or CIM_BootSourceSetting instance and its PartComponent property shall reference the instance of a concrete subclass of CIM_SettingData.

8 Methods

Clause 8 details the requirements for supporting intrinsic operations and extrinsic methods for the CIM elements defined by this profile.

8.1 CIM_BootService.CreateBootConfigSetting( )

This method is conditional on the CIM_BootServiceCapabilities.BootConfigCapabilities property array containing a value of 2 (Creates Boot Configuration). For more information, see subclause 7.6.1.

The CreateBootConfigSetting( ) method shall create a clone of an existing Boot Configuration using a Template Boot Configuration and associate the new Boot Configuration to the Boot Configurable System. The method has two input parameters: StartingBootConfig and ScopingComputerSystem. At least one of the two parameters shall be non-null for the method to be successfully invoked.

The input parameter, StartingBootConfig, shall be used to provide a reference to the Template Boot Configuration to use as the template for the new Boot Configuration Representation.

The input parameter, ScopingComputerSystem, shall be used to reference the Boot Configurable System, an existing CIM_ComputerSystem, to which the new CIM_BootConfigSetting instance shall be associated through an instance of CIM_ElementSettingData.

When the StartingBootConfig parameter and the ScopingComputerSystem parameter are both NULL, a return value or an exception shall be returned. When a return value is returned, it shall have a value of 2 (Error Occurred).

When the StartingBootConfig parameter has a NULL value and the ScopingComputerSystem parameter has a non-NULL value, the implementation shall find the Default Boot Configuration associated to the CIM_ComputerSystem instance referenced by the ScopingComputerSystem and use it as the Template Boot Configuration for the new boot configuration. If a Default Boot Configuration is not found, a return value or an exception shall be returned. When a return value is returned, it shall have a value of 2 (Error Occurred).

When the StartingBootConfig parameter has a non-NULL value and the ScopingComputerSystem parameter is NULL, the implementation shall associate the new boot configuration to the Boot Configurable System of the Template Boot Configuration.

Upon successful completion of this method, a new Boot Configuration Representation shall exist and be a replica of the Template Boot Configuration. The new instance of CIM_BootConfigSetting shall be associated to the instance representing the Boot Configurable System through an instance of CIM_ElementSettingData. All properties in the new Boot Configuration Representation and Template Boot Configuration representations are expected to have the same value, except for the key properties, unless otherwise mandated in the requirements below.

- A new instance of CIM_BootConfigSetting shall exist and be referenced by the output NewBootConfig parameter. The new CIM_BootConfigSetting.InstanceID property shall be set to a unique value.
- A new instance of CIM_ElementSettingData shall exist that associates the new CIM_BootConfigSetting to the instance of the Boot Configurable System, which is specified by the ScopingComputerSystem parameter when it is non-NULL or implied by the StartingBootConfig parameter when the ScopingComputerSystem parameter is NULL.
- The CIM_ElementSettingData.IsDefault property shall be set to 2 (Is Not Default). The CIM_ElementSettingData.IsCurrent property shall be set to 2 (Is Not Current). The CIM_ElementSettingData.IsNext property shall be set to 2 (Is Not Next).

- New instances of CIM_BootSourceSetting shall exist, along with instances of CIM_OrderedComponent, when they are present in the boot configuration represented by the Template Boot Configuration. The new instances shall be duplicates of those found in the boot configuration represented by the Template Boot Configuration, except for the key property value.
- New instances of CIM_BootSettingData shall exist when they are present in the boot configuration represented by the Template Boot Configuration. The new instances shall be duplicates of those found in the boot configuration represented by the Template Boot Configuration, except for the key property value.
- New instances of CIM_ConcreteComponent shall exist when they are present in the boot configuration represented by the Template Boot Configuration.
- New instances of CIM_ConcreteDependency shall exist when they are present in the boot configuration represented by the Template Boot Configuration.
- CIM elements that are defined in a Referencing Profile are not copied.

The return code values and parameters for the CreateBootConfigSetting( ) method are specified in Table 3 and Table 4.

No standard messages are defined.

Table 3 – CreateBootConfigSetting( ) Method: Return Code Values

Value	Description
0	Completed with no error
1	Not supported
2	Error occurred
4096	Job started

Table 4 – CreateBootConfigSetting( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN	ScopingComputerSystem	CIM_ComputerSystem REF	Reference to an existing CIM_ComputerSystem instance
IN, REQ	StartingBootConfig	CIM_BootConfigSetting REF	Reference to an existing CIM_BootConfigSetting instance
OUT	NewBootConfig	CIM_BootConfigSetting REF	Reference to the newly created CIM_BootConfigSetting
OUT	Job	CIM_ConcreteJob REF	Reference to a CIM_ConcreteJob returned if job started

8.2 CIM_BootService.ApplyBootConfigSetting( )

This method is conditional on the CIM_BootServiceCapabilities.BootConfigCapabilities property array containing a value of 3 (Applies Boot Configuration). See subclause 7.5.1 for more information.

NOTE: Successful execution of the ApplyBootConfigSetting() method can independently apply a boot configuration to a Boot Configurable System regardless of the Next Boot Configuration. The requirements in subclause 7.4.4 shall not apply when a Boot Configurable System is booted using the ApplyBootConfigSetting() method.

The ApplyBootConfigSetting( ) method shall start the boot process on a specified Boot Configurable System, using the specified boot configuration of the Boot Configurable System. The boot process may be started from a pause in the boot flow or from a reboot of the Boot Configurable System. The method has two input parameters, ScopingComputerSystem and ApplyBootConfig.

The input parameter, ScopingComputerSystem, shall be used to reference the Boot Configurable System, an existing CIM_ComputerSystem with instances CIM_BootConfigSetting associated to it through an instance of CIM_ElementSettingData.

When the ScopingComputerSystem parameter is NULL, the boot configuration shall be applied to each CIM_ComputerSystem which is associated to the instance of CIM_BootConfigSetting referenced by the ApplyBootConfig parameter via an instance of CIM_ElementSettingData.

When the instance of CIM_ComputerSystem referenced by ScopingComputerSystem parameter is not associated to an instance of CIM_BootService, a return value or an exception shall be returned. When a return value is returned, it shall have a value of 2 (Error Occurred).

The input parameter, ApplyBootConfig, shall be used to provide a reference to an instance of CIM_BootConfigSetting associated to the Boot Configurable System for use in the boot process.

When the ApplyBootConfig parameter is NULL, a return value or an exception shall be returned. When a return value is returned, it shall have a value of 2 (Error Occurred).

When the instance of CIM_BootConfigSetting referenced by ApplyBootConfig parameter is not found, a return value or an exception shall be returned. When a return value is returned, it shall have a value of 2 (Error Occurred).

When the instance of CIM_BootConfigSetting referenced by ApplyBootConfig parameter is not associated with the ScopingComputerSystem, a return value or an exception shall be returned. When a return value is returned, it shall have a value of 2 (Error Occurred).

Upon successful completion of this method, the boot process shall have started using the boot configuration referenced by the ApplyBootConfig parameter.

The return code values and parameters for the ApplyBootConfigSetting( ) method are specified in Table 5, respectively.

No standard messages are defined.

Table 5 – ApplyBootConfigSetting( ) Method: Return Code Values

Value	Description
0	Completed with no error
1	Not supported
2	Error occurred
4096	Job started

Table 6 – ApplyBootConfigSetting( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	ScopingComputerSystem	CIM_ComputerSystem REF	Reference to an existing CIM_ComputerSystem instance
IN, REQ	ApplyBootConfig	CIM_BootConfigSetting REF	Reference to an existing CIM_BootConfigSetting instance
OUT	Job	CIM_ConcreteJob REF	Reference to a CIM_ConcreteJob returned if job started

8.3 CIM_BootConfigSetting.ChangeBootOrder( )

The ChangeBootOrder( ) method shall set the order in which the instances of CIM_BootSourceSetting are associated to a CIM_BootConfigSetting instance. The method has one input parameter: Source.

When the ChangeBootOrder() method is not supported, a return value or an exception shall be returned.

The input parameter, Source, is an ordered array of references to CIM_BootSourceSetting instances that defines the new sequence of the CIM_BootSourceSetting instances associated to the instance of CIM_BootConfigSetting. Each CIM_BootSourceSetting instance in the array shall already be associated with this CIM_BootConfigSetting instance through an instance of CIM_OrderedComponent. This parameter is required.

When the Source parameter is NULL, a return value of 2 (Error Occurred) shall be returned.

When any of the CIM_BootSourceSetting instance in the Source array are not associated to the instance of CIM_BootConfigSetting, the implementation shall return a value of 2 (Error Occurred).

Upon successful completion of this method, the value of the AssignedSequence property on each instance of CIM_OrderedComponent shall be updated such that the values are monotonically increasing in correlation with the position of the referenced CIM_BootSourceSetting instance in the Source input parameter. That is, the first position in the array shall have the lowest non-zero value for AssignedSequence. The second position will have the second lowest value, and so on.

Upon successful completion of this method, the value of the AssignedSequence property on each instance of CIM_OrderedComponent, that associates the target CIM_BootConfigSetting instance to a CIM_BootSourceSetting instance that is not present in the input array, shall be assigned a value of 0.

The return code values and parameters for the ChangeBootOrder( ) method are specified in Table 7 and Table 8, respectively.

No standard messages are defined.

Table 7 – ChangeBootOrder( ) Method: Return Code Values

Value	Description
0	Completed with No Error
1	Not Supported
2	Error Occurred
4096	Job Started

Table 8 – ChangeBootOrder( ) Method: Parameters

Qualifiers	Name	Type	Description/Values
IN, REQ	Source[]	CIM_BootSourceSetting REF	An ordered array of references to CIM_BootSourceSetting instances
OUT	Job	CIM_ConcreteJob REF	Reference to a CIM_ConcreteJob Returned if job started.

8.4 Profile Conventions for Operations

Support for operations for each profile class (including associations) is specified in the following subclauses. Each of these subclauses includes a table listing all the operations supported by this profile. Compliant implementations of this profile shall support all these operations.

8.5 CIM_BootService

Compliant implementations of this profile shall support the operations listed in Table 9 for CIM_BootService. Each operation shall be supported as defined in DSP0200.

Table 9 – Operations: CIM_BootService

Operation	Requirement	Messages
GetInstance	Mandatory	None
ModifyInstance	Optional	See subclause 8.5.1.
Associators	Mandatory	None
AssociatorNames	Mandatory	None
References	Mandatory	None
ReferenceNames	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.5.1 CIM_BootService — ModifyInstance Operation

Subclause 8.5.1 details the specific requirements for the ModifyInstance operation applied to an instance of CIM_BootService.

When an instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance and the CIM_BootServiceCapabilities.ElementNameEditSupported property has a value of TRUE, the implementation shall allow the ModifyInstance operation to change the value of the ElementName property of the CIM_BootService instance. The ModifyInstance operation shall enforce the length restriction specified in the MaxElementNameLen property of the CIM_BootServiceCapabilities instance.

When no instance of CIM_BootServiceCapabilities is associated with the CIM_BootService instance, or the ElementNameEditSupported property of the CIM_BootServiceCapabilities has a value of FALSE, the implementation shall not allow the ModifyInstance operation to change the value of the ElementName property of the CIM_BootService instance.

8.6 CIM_BootConfigSetting

Compliant implementations of this profile shall support the operations listed in Table 10 for the CIM_BootConfigSetting class. Each operation shall be supported as defined in DSP0200.

Table 10 – Operations: CIM_BootConfigSetting

Operation	Requirement	Messages
DeleteInstance	Conditional	See subclause 8.6.1.
GetInstance	Mandatory	None
Associators	Mandatory	None
AssociatorNames	Mandatory	None
References	Mandatory	None
ReferenceNames	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.6.1 CIM_BootConfigSetting – DeleteInstance

Subclause 8.6.1 describes conditional behavior.

Conditional requirement: Subclause 7.7 describes the conditions when the DeleteInstance operation shall be supported. Implementations may choose to support the DeleteInstance operation even when the conditions described in subclause 7.7 are not met.

When the DeleteInstance operation is supported for an instance of CIM_BootConfigSetting, upon completion of this operation, the following instances shall be deleted:

- The target instance of CIM_BootConfigSetting shall no longer exist.
- The instance of CIM_ElementSettingData that associated the target CIM_BootConfigSetting to the instance of CIM_ComputerSystem shall no longer exist.
- The instances of CIM_ConcreteComponent, which associate the target instance of CIM_BootConfigSetting to instances of a concrete subclass of CIM_SettingData, shall no longer exist.
- The instances of CIM_ConcreteComponent, which associate the target instance of CIM_BootConfigSetting to instances of a concrete subclass of CIM_BootSettingData, shall no longer exist. The instances of the associated concrete subclass of CIM_BootSettingData shall no longer exist.
- The instances of CIM_OrderedComponent, which associate the target instance of CIM_BootConfigSetting to instances of CIM_BootSourceSetting, shall no longer exist. The instances of the associated CIM_BootSourceSetting shall no longer exist.
- The instances of CIM_ConcreteComponent, which associate instances of a concrete subclass of CIM_SettingData to instances of CIM_BootSourceSetting, which in turn are associated to the target instance of CIM_BootConfigSetting, shall no longer exist.
- The instances of CIM_ConcreteComponent, which associate instances of a concrete subclass of CIM_BootSettingData to instances of CIM_BootSourceSetting, which in turn are associated to the target instance of CIM_BootConfigSetting, shall no longer exist. The instances of the associated concrete subclass of CIM_BootSettingData shall no longer exist.
- The instances of CIM_ConcreteDependency, which associate instances of a concrete subclass of CIM_LogicalDevice to instances of CIM_BootSourceSetting, shall no longer exist.
- The instance of CIM_LogicalIdentity, which associates a deleted instance of CIM_BootSourceSetting to an instance of CIM_BootConfigSetting, shall no longer exist. The associated instance of CIM_BootConfigSetting shall no longer exist. The requirements in this subclause shall be applied recursively to the deleted CIM_BootConfigSetting instance.

8.7 CIM_BootSettingData

Compliant implementations of this profile shall support the operations listed in Table 11 for the CIM_BootSettingData class. Each operation shall be supported as defined in DSP0200.

Table 11 – Operations: CIM_BootSettingData

Operation	Requirement	Messages
GetInstance	Mandatory	None
Associators	Mandatory	None
AssociatorNames	Mandatory	None

Operation	Requirement	Messages
References	Mandatory	None
ReferenceNames	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.8 CIM_BootSourceSetting

Compliant implementations of this profile shall support the operations listed in Table 12 for the CIM_BootSourceSetting class. Each operation shall be supported as defined in DSP0200.

Table 12 – Operations: CIM_BootSourceSetting

Operation	Requirement	Messages
GetInstance	Mandatory	None
Associators	Mandatory	None
AssociatorNames	Mandatory	None
References	Mandatory	None
ReferenceNames	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.9 CIM_ConcreteComponent

Compliant implementations of this profile shall support the operations listed in Table 13 for the CIM_ConcreteComponent class. Each operation shall be supported as defined in DSP0200.

Table 13 – Operations: CIM_ConcreteComponent

Operation	Requirement	Messages
GetInstance	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.10 CIM_ConcreteDependency

Compliant implementations of this profile shall support the operations listed in Table 14 for the CIM_ConcreteDependency class. Each operation shall be supported as defined in DSP0200.

Table 14 – Operations: CIM_ConcreteDependency

Operation	Requirement	Messages
GetInstance	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.11 CIM_ElementCapabilities

Compliant implementations of this profile shall support the operations listed in Table 15 for the CIM_ElementCapabilities class. Each operation shall be supported as defined in DSP0200.

Table 15 – Operations: CIM_ElementCapabilities

Operation	Requirement	Messages
GetInstance	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.12 CIM_ElementSettingData

Compliant implementations of this profile shall support the operations listed in Table 16 for the CIM_ElementSettingData class. Each operation shall be supported as defined in DSP0200.

Table 16 – Operations: CIM_ElementSettingData

Operation	Requirement	Messages
GetInstance	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None
ModifyInstance	Optional	See subclause 8.12.1.

8.12.1 CIM_ElementSettingData – ModifyInstance Operation

The behavior of the ModifyInstance operation varies depending on the property of the association modified.

8.12.1.1 CIM_ElementSettingData.IsDefault Property

When the ModifyInstance operation is used to set the IsDefault property to a value of 1 (Is Default), the ModifyInstance operation shall implement the following behavior.

The behavior described insures that there is at most one instance of CIM_ElementSettingData associated to the Boot Configurable System whose IsDefault property has a value of 1 (Is Default) as specified in subclause 7.4.2, by first finding any existing instance of CIM_ElementSettingData whose IsDefault property already has a value of 1 (Is Default) and modifying the value to 2 (Is Not Default).

- Search for an instance of CIM_ElementSettingData that associates an instance of CIM_BootConfigSetting with the instance of CIM_ComputerSystem, which is referenced by the target instance of CIM_ElementSettingData where the IsDefault property has a value of 1 (Is Default).
- If such an instance of CIM_ElementSettingData is found, the ModifyInstance operation shall set the value of the IsDefault property to 2 (Is Not Default).
- For the target instance of CIM_ElementSettingData, when the IsDefault property already has a value of 1 (Is Default), the ModifyInstance operation shall complete successfully.
- For the target instance of CIM_ElementSettingData, set the value of the IsDefault property to 1 (Is Default).

8.12.1.2 CIM_ElementSettingData.IsNext Property

When the ModifyInstance operation is used to set the IsNext property to a value of 1 (Is Next), the ModifyInstance operation shall implement the following behavior.

The behavior described insures that there is at most one instance of CIM_ElementSettingData associated to the Boot Configurable System whose IsNext property has a value of 1 (Is Next) as specified in subclause 7.4.4, by first finding any existing instance of CIM_ElementSettingData whose IsNext property already has a value of 1 (Is Next) and modifying the value to 2 (Is Not Next).

- Search for an instance of CIM_ElementSettingData that associates an instance of CIM_BootConfigSetting with the instance of CIM_ComputerSystem, which is referenced by the target instance of CIM_ElementSettingData where the IsNext property has a value of 1 (Is Next).
- If such an instance of CIM_ElementSettingData is found, the ModifyInstance operation shall set the value of the IsNext property to 2 (Is Not Next).
- For the target instance of CIM_ElementSettingData, when the IsNext property already has a value of 1 (Is Next), the ModifyInstance operation shall complete successfully.
- For the target instance of CIM_ElementSettingData, when the IsNext property has a value other than 1 (Is Next), set the value of the IsNext property to 1 (Is Next).

When the ModifyInstance operation is used to set the IsNext property to a value of 3 (Is Next For Single Use), the ModifyInstance operation shall implement the following behavior.

The behavior described insures that there is at most one instance of CIM_ElementSettingData associated to the Boot Configurable System whose IsNext property has a value of 3 (Is Next For Single Use) as specified in subclause 7.4.5, by first finding any existing instance of CIM_ElementSettingData whose IsNext property already has a value of 3 (Is Next For Single Use) and modifying the value to 2 (Is Not Next).

- For the target instance of CIM_ElementSettingData, when the IsNext property has a value of 1 (Is Next), the ModifyInstance operation shall fail.
- Search for an instance of CIM_ElementSettingData that associates an instance of CIM_BootConfigSetting with the instance of CIM_ComputerSystem referenced by the target instance of CIM_ElementSettingData where the IsNext property has a value of 3 (Is Next For Single Use).
- If such an instance of CIM_ElementSettingData is found, the ModifyInstance operation shall set the value of the IsNext property to 2 (Is Not Next).
- For the target instance of CIM_ElementSettingData, when the IsNext property already has a value of 3 (Is Next For Single Use), the ModifyInstance operation shall complete successfully.
- For the target instance of CIM_ElementSettingData, when the IsNext property has a value neither 1 (Is Next) nor 3 (Is Next For Single Use), set the value of the IsNext property to 3 (Is Next For Single Use).

8.12.1.3 CIM_ElementSettingData.IsCurrent Property

The ModifyInstance operation shall not be used to set the IsCurrent property.

8.13 CIM_BootServiceCapabilities

Compliant implementations of this profile shall support the operations listed in Table 17 for the CIM_BootServiceCapabilities class. Each operation shall be supported as defined in DSP0200.

Table 17 – Operations: CIM_BootServiceCapabilities

Operation	Requirement	Messages
GetInstance	Mandatory	None
Associators	Mandatory	None
AssociatorNames	Mandatory	None
References	Mandatory	None
ReferenceNames	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.14 CIM_HostedService

Compliant implementations of this profile shall support the operations listed in Table 18 for the CIM_HostedService class. Each operation shall be supported as defined in DSP0200.

Table 18 – Operations: CIM_HostedService

Operation	Requirement	Messages
GetInstance	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.15 CIM_LogicalIdentity

Compliant implementations of this profile shall support the operations listed in Table 19 for the CIM_LogicalIdentity class. Each operation shall be supported as defined in DSP0200.

Table 19 – Operations: CIM_LogicalIdentity

Operation	Requirement	Messages
GetInstance	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.16 CIM_OrderedComponent

Compliant implementations of this profile shall support the operations listed in Table 20 for the CIM_OrderedComponent class. Each operation shall be supported as defined in DSP0200.

Table 20 – Operations: CIM_OrderedComponent

Operation	Requirement	Messages
GetInstance	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

8.17 CIM_ServiceAffectsElement

Compliant implementations of this profile shall support the operations listed in Table 21 for the CIM_ServiceAffectsElement class. Each operation shall be supported as defined in DSP0200.

Table 21 – Operations: CIM_ServiceAffectsElement

Operation	Requirement	Messages
GetInstance	Mandatory	None
EnumerateInstances	Mandatory	None
EnumerateInstanceNames	Mandatory	None

9 Use Cases

Clause 9 contains object diagrams and use cases for the *Boot Control Profile*.

9.1 Advertising the Profile Conformance

The object diagram in Figure 2 shows how instances of CIM_RegisteredProfile are used to identify the version of the *Boot Control Profile* with which an instance of CIM_BootService and its associated instances are conformant. An instance of CIM_RegisteredProfile exists for each profile that is instrumented in the system. One instance of CIM_RegisteredProfile identifies the DMTF *Base Server Profile*, version 1.0.0. The other instance identifies the DMTF *Boot Control Profile*, version 1.0.0. The Central Instance is the CIM_BootService. The Scoping Instance is the CIM_ComputerSystem instance.

This instance of CIM_ComputerSystem is conformant with the *Base Server Profile* version 1.0.0 as indicated by the CIM_ElementConformsToProfile association to the CIM_RegisteredProfile instance.

This instance of CIM_BootService is conformant with the *Boot Control Profile* version 1.0.0 as indicated by the CIM_ElementConformsToProfile association to the CIM_RegisteredProfile instance.

The CIM_ReferencedProfile relationship between *BaseServer* and *BootControl* places the CIM_BootService instance within the scope of *BaseServer.*

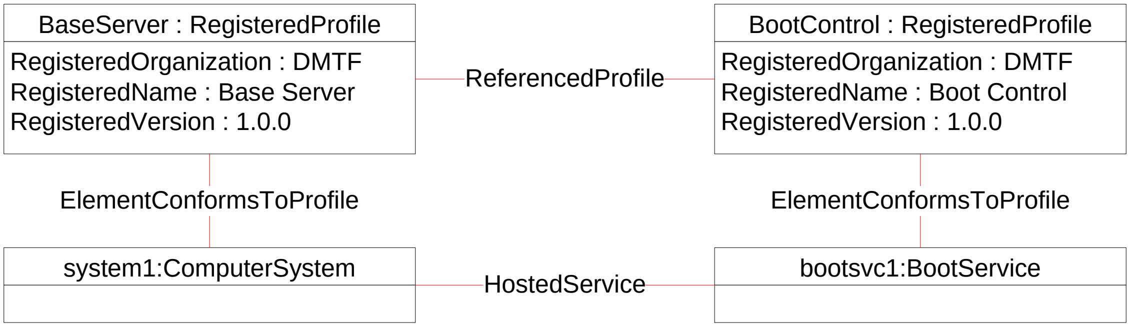

BaseServer : RegisteredProfile
RegisteredOrganization : DMTF
RegisteredName : Base Server
RegisteredVersion : 1.0.0
ReferencedProfile
BootControl : RegisteredProfile
RegisteredOrganization : DMTF
RegisteredName : Boot Control
RegisteredVersion : 1.0.0
ElementConformsToProfile
ElementConformsToProfile
system1:ComputerSystem
HostedService
bootsvc1:BootService

Figure 2 – Registered Profile

9.2 Object Diagram for a Monolithic Server

Figure 3 shows the CIM instances required to control the boot configuration for a single, monolithic server, *system1*. *System1* hosts the boot service, *bootsvc1*, which is used to control the boot configuration, *bootcfgsetting1*, for *system1. System1* is also identified as the Boot Configurable System through the CIM_ServiceAffectsElement association*.* The capabilities of *bootsvc1* are defined by *bootsvccap1*.

The boot configuration, *bootcfgsetting1,* has one boot source, *bootsrcsetting1*.

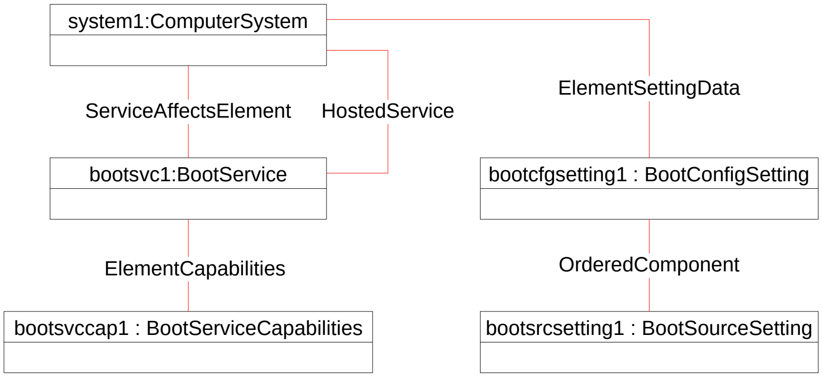


Figure 3 – Monolithic Server Object Diagram

9.3 Object Diagram for a Monolithic Server with Service Processor

Figure 4 shows the CIM instances required to control the boot configuration for a single, monolithic server, *system1*, with an attached service processor, *sp1*. The boot service, *bootsvc1*, is hosted by the service processor and is responsible for managing the boot configuration, *bootcfgsetting1*, for *system1*.

Optionally, the service processor may host another boot configuration service, *bootsvc2*, to control its own boot configuration, *bootcfgsetting2*.

The capabilities of *bootsvc1* and bootsrv2 are defined by *bootsvccap1* and *bootsvccap2* respectively.

Each boot configuration *(bootcfgsetting1, bootcfgsetting2)* has one boot source (*bootsrcsetting1*, *bootsrcsetting2),* respectively.

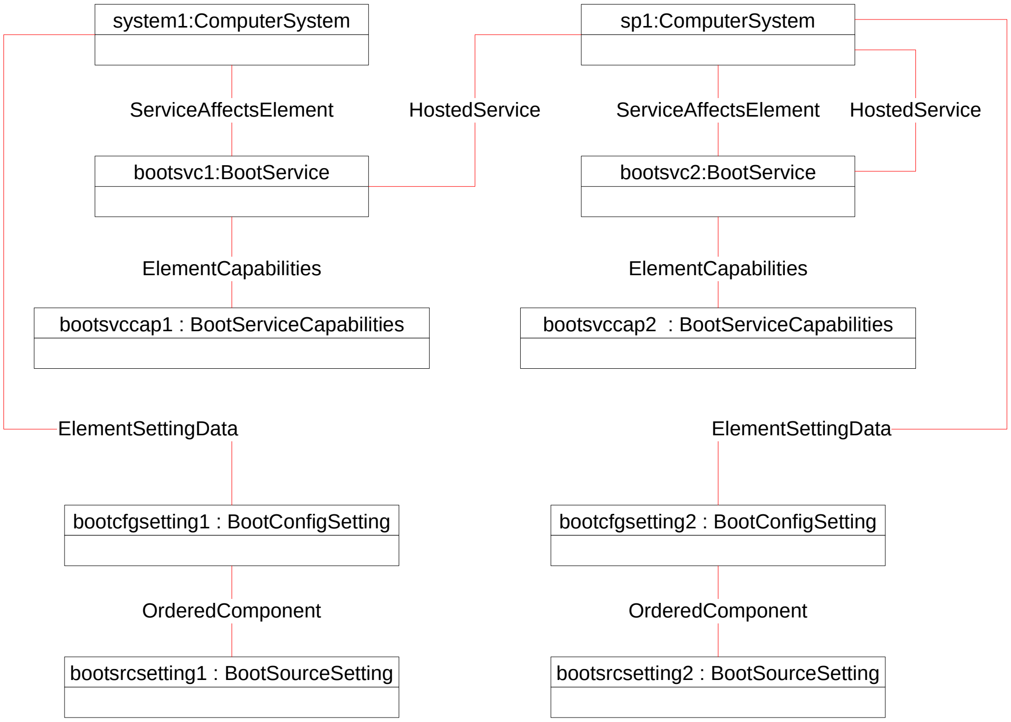


Figure 4 – Monolithic Server with Service Processor Object Diagram

9.4 Object Diagram for a Modular System

Figure 5 shows the CIM instances required to control the boot configuration for a modular system. The boot service, *bootsvc1*, is hosted by the chassis manager, chassismgr1, and is responsible for managing the boot configuration for two blade systems, *system1* and *system2*. *System1* and *system2* each have one boot configuration, *bootcfgsetting1* and *bootcfgsetting2* respectively.

Optionally, the chassis manager may host another boot configuration service, *bootsvc2*, to control its own boot configuration, *bootcfgsetting3.*

The capabilities of *bootsvc1* and bootsrv2 are defined by *bootsvccap1* and *bootsvccap2* respectively.

Each boot configuration *(bootcfgsetting1, bootcfgsetting2, bootcfgsetting3)* has one boot source (*bootsrcsetting1*, *bootsrcsetting2, bootsrcsetting3*), respectively.

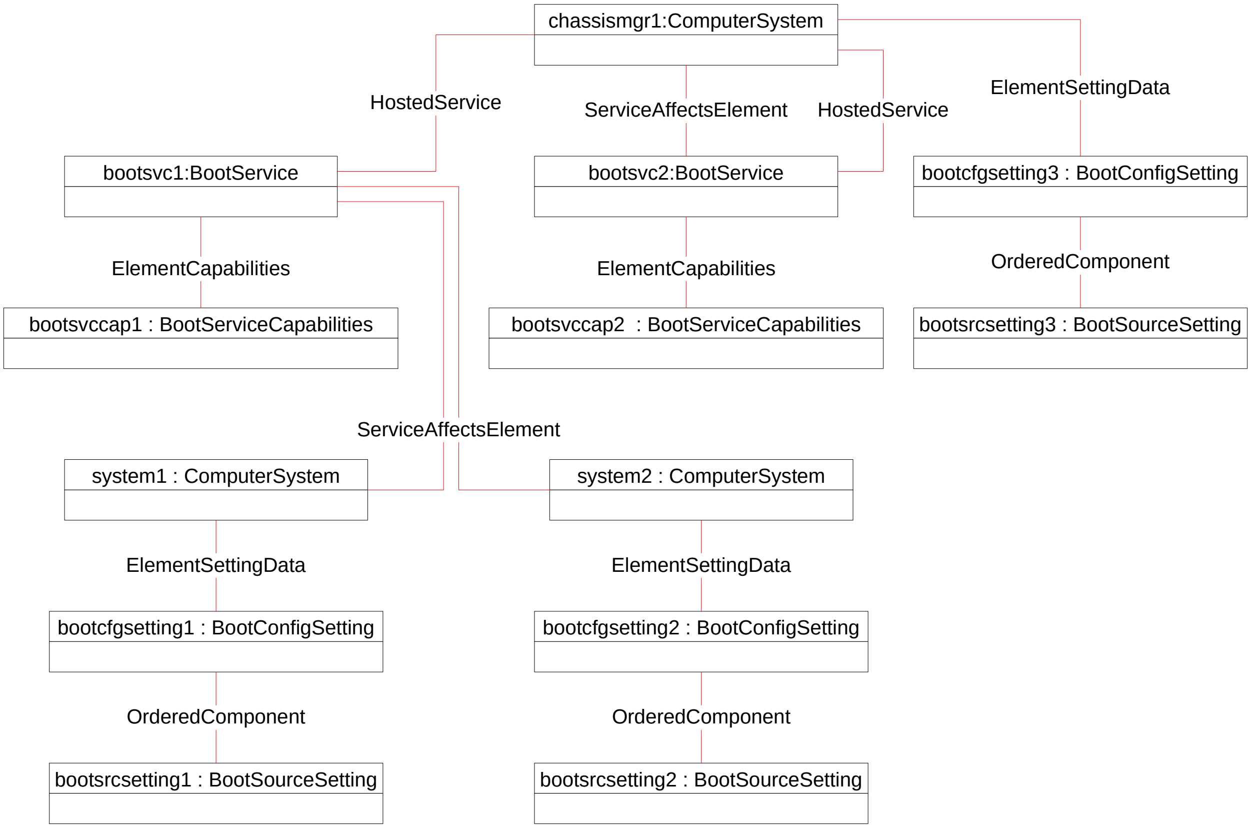
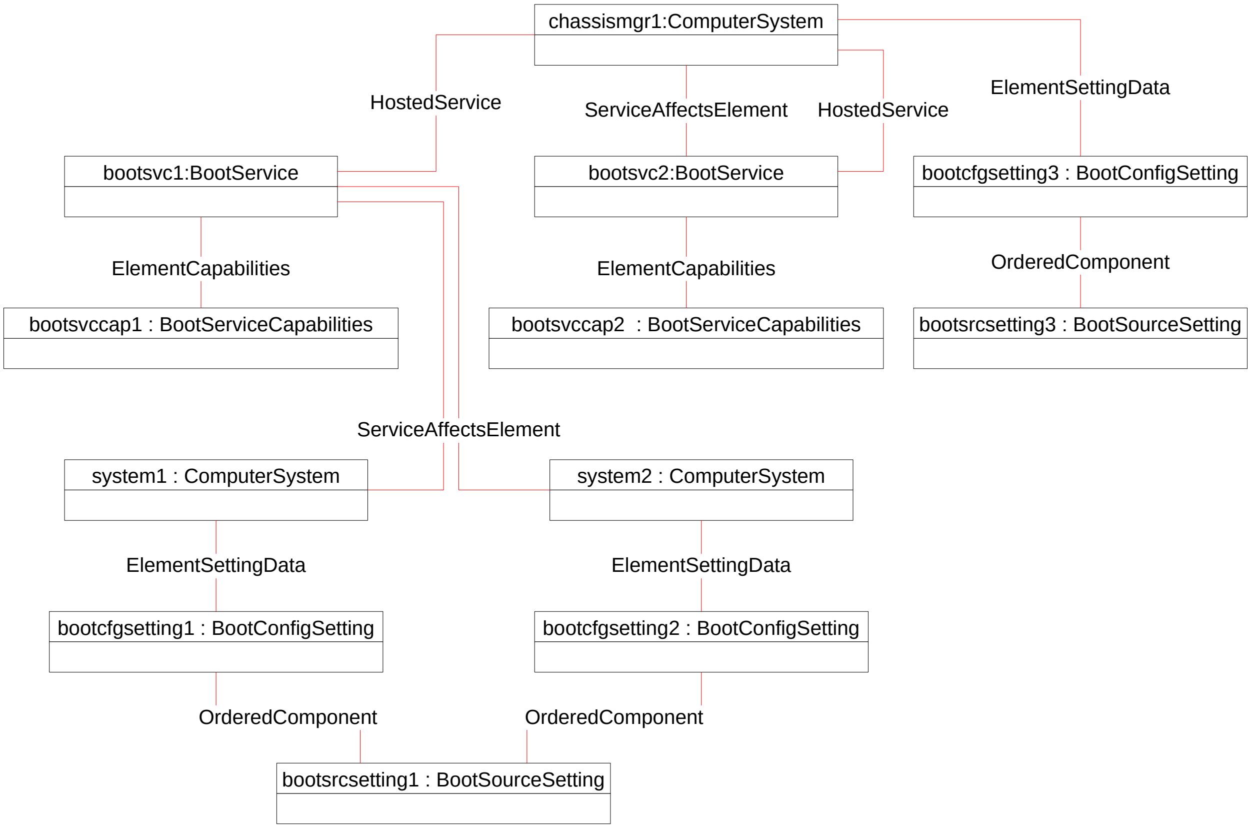

chassismgr1:ComputerSystem
HostedService
ServiceAffectsElement
HostedService
ElementSettingData
bootsvc1:BootService
bootsvc2:BootService
bootcfgsetting3 : BootConfigSetting
ElementCapabilities
ElementCapabilities
OrderedComponent
bootsvccap1 : BootServiceCapabilities
bootsvccap2 : BootServiceCapabilities
bootsrcsetting3 : BootSourceSetting
ServiceAffectsElement
system1 : ComputerSystem
system2 : ComputerSystem
ElementSettingData
ElementSettingData
bootcfgsetting1 : BootConfigSetting
bootcfgsetting2 : BootConfigSetting
OrderedComponent
OrderedComponent
bootsrcsetting1 : BootSourceSetting
bootsrcsetting2 : BootSourceSetting

Figure 5 – Modular System Object Diagram

Alternative representation of modular blade system boot sources is shown in the Figure 6. The blade computer systems, in this example, have the same boot sources, and thus, the representation of boot sources can be optimized by instantiating only one CIM_BootSourceSetting shared between the respective CIM_BootConfigSetting instances. This optimization is especially useful when modular system contains many blade computer systems with the similar boot sources. Each blade boot configuration *(bootcfgsetting1, bootcfgsetting2)* has one boot source (*bootsrcsetting1*).



1157

1158 **Figure 6 – Modular System Object Diagram**

1159 9.5 PXE Boot Source

1160 Figure 7 shows an instance diagram for a boot configuration, *bootcfgsetting1*, which has two boot sources
1161 associated to it, *bootsrcsetting1* and *bootsrcsetting2*, which are both network ports.

1162 A network port can support various protocols. Both *bootsrcsetting1* and *bootsrcsetting2* designate the
1163 PXE protocol in their BIOSBootString property. The two CIM_ConcreteDependency associations to
1164 instances of CIM_NetworkPort are *netport2* and *netport1, respectively*.

1165 The AssignedSequence property values on the OrderedComponent associations indicate that the boot
1166 order is *bootsrcsetting1* followed by *bootsrcsetting2*.

1167 On *bootsrcsetting1*, the FailThroughSupported property value of 1 (Is Supported) indicates that if the
1168 *bootsrcsetting1* fails or times out, the boot process should proceed to *bootsrcsetting2* on *netport1*.

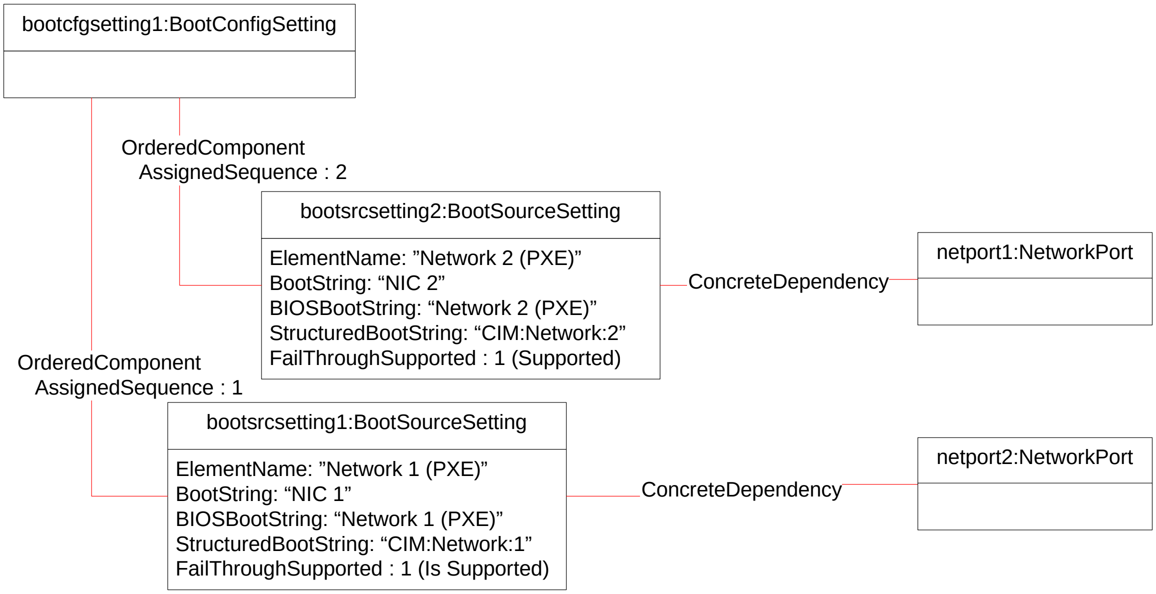

bootcfgsetting1:BootConfigSetting
OrderedComponent
AssignedSequence : 2
bootsrcsetting2:BootSourceSetting
ElementName: ”Network 2 (PXE)”
BootString: “NIC 2”
BIOSBootString: “Network 2 (PXE)”
StructuredBootString: “CIM:Network:2”
FailThroughSupported : 1 (Supported)
ConcreteDependency
netport1:NetworkPort
OrderedComponent
AssignedSequence : 1
bootsrcsetting1:BootSourceSetting
ElementName: ”Network 1 (PXE)”
BootString: “NIC 1”
BIOSBootString: “Network 1 (PXE)”
StructuredBootString: “CIM:Network:1”
FailThroughSupported : 1 (Is Supported)
ConcreteDependency
netport2:NetworkPort

Figure 7 – PXE Boot Sources Object Diagram

9.6 Disk Boot Source

Figure 8 shows an instance diagram for a boot configuration, *bootcfgsetting1*, which has a single boot source associated to it, *bootsrcsetting1*, which is a disk device.

The CIM_BootSourceSetting.ElementName property identifies "Hard Disk 1" as the boot source, which matches the BIOSBootString property. The BootString property contains the string "C:", which could be interpreted by the boot process to assign the hard drive the letter "C". The CIM_ConcreteDependency association relates *bootsrcsetting1* to a CIM_LogicalDisk (*disk1*).

Because there is only one boot source, the value of the CIM_BootSourceSetting.FailThroughSupported is not meaningful.

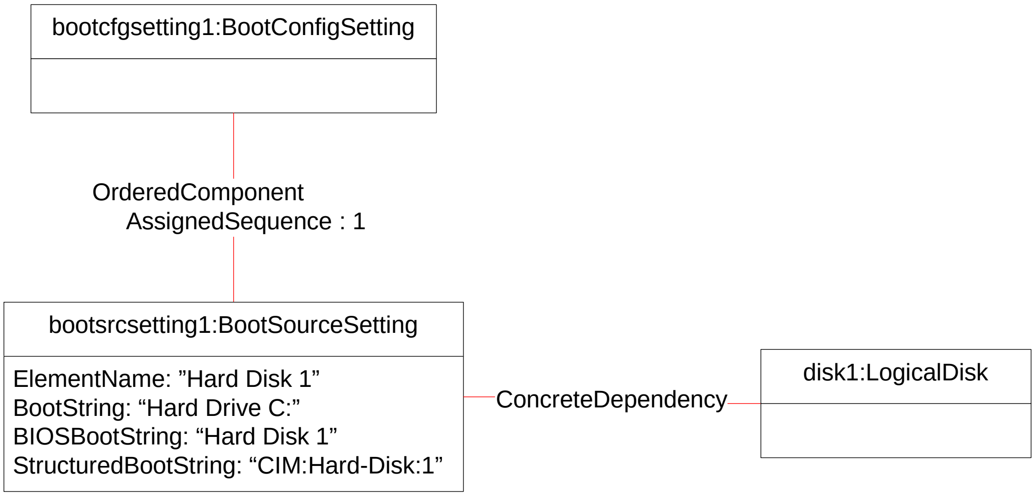

bootcfgsetting1:BootConfigSetting
OrderedComponent
AssignedSequence : 1
bootsrcsetting1:BootSourceSetting
ElementName: ”Hard Disk 1”
BootString: “Hard Drive C:”
BIOSBootString: “Hard Disk 1”
StructuredBootString: “CIM:Hard-Disk:1”
ConcreteDependency
disk1:LogicalDisk

Figure 8 – Booting from Disk

9.7 Local CDROM and Floppy Boot Sources

Figure 9 shows an instance diagram for a boot configuration, *bootcfgsetting1*, which has two boot sources associated to it, *bootsrcsetting1* and *bootsrcsetting2*. *Bootsrcsetting1* is a CD-ROM device; *bootsrcsetting2* is a floppy drive.

The AssignedSequence property of the CIM_OrderedComponent associations is set such that booting is attempted from the CD-ROM drive first and then the floppy drive.

The BootString property for the CD-ROM drive, *bootsrcsetting1*, contains the string "F:", which could be interpreted by the boot process to assign the floppy drive the letter "F". The CIM_ConcreteDependency association relates *bootsrcsetting1* to a CIM_CDROMDrive (*cdrom1*).

The BootString property for the floppy drive, *bootsrcsetting2*, contains the string "A:", which could be interpreted by the boot process to assign the floppy drive the letter "A". The CIM_ConcreteDependency association relates *bootsrcsetting2* to a CIM_DisketteDrive (*floppymedia1*).

On *bootsrcsetting1*, the value of the FailThroughSupported property set to 1 (Is Supported) specifies that if the *bootsrcsetting1,* the CD-ROM device, fails or times out, then the boot process should proceed to *bootsrcsetting2*, the floppy device.

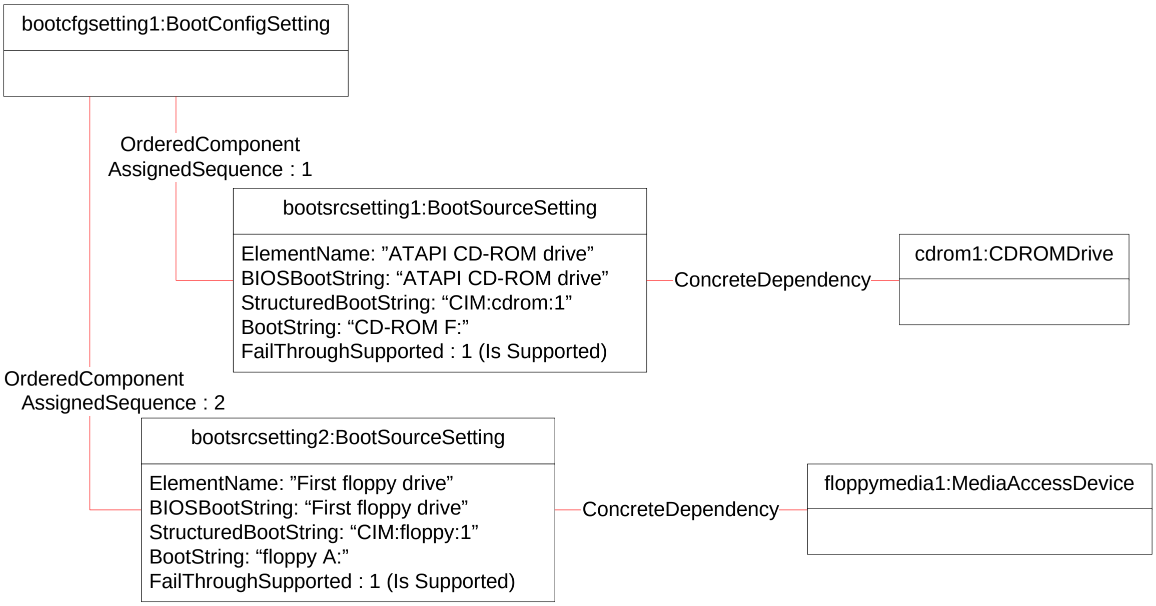

bootcfgsetting1:BootConfigSetting
OrderedComponent
AssignedSequence : 1
bootsrcsetting1:BootSourceSetting
ElementName: ”ATAPI CD-ROM drive”
BIOSBootString: “ATAPI CD-ROM drive”
StructuredBootString: “CIM:cdrom:1”
BootString: “CD-ROM F:”
FailThroughSupported : 1 (Is Supported)
ConcreteDependency
cdrom1:CDROMDrive
OrderedComponent
AssignedSequence : 2
bootsrcsetting2:BootSourceSetting
ElementName: ”First floppy drive”
BIOSBootString: “First floppy drive”
StructuredBootString: “CIM:floppy:1”
BootString: “floppy A:”
FailThroughSupported : 1 (Is Supported)
ConcreteDependency
floppymedia1:MediaAccessDevice

Figure 9 – Booting from CDROM and Floppy

9.8 Representing IPL and Boot Control Vector (BCV) Lists

Figure 10 shows an instance diagram for a boot configuration, *bootcfgsetting1*, composed of an IPL and BCV list of boot devices.

To represent the IPL list, *bootcfgsetting1* has three boot sources associated to it, *bootsrcsetting1*, *bootsrcsetting2*, and *bootsrcsetting3. Bootsrcsetting1* is a CD-ROM device. *Bootsrcsetting2* is a floppy drive. *Bootsrcsetting3* is a BCV device (boot control vector).

The AssignedSequence property of the CIM_OrderedComponent associations is set such that booting is attempted from the CD-ROM drive first and then the BCV device. Booting from the floppy device is not attempted because the AssignedSequence property is set to 0. The

CIM_BootConfigSetting.FailThroughSupported property value of 1 (Is Supported) specifies that the boot process should proceed to the second boot source if the first boot source fails or times out.

In the diagram, the BCV device is a SCSI controller that may have multiple bootable SCSI devices attached to it. This relationship is represented by an instance of CIM_LogicalIdentity between *bootsrcsetting3* and an instance of CIM_BootConfigSetting, *bootcfgsetting20*.

The boot configuration, *bootcfgsetting20*, has two boot sources associated to it, *bootsrcsetting21* and *bootsrcsetting22.* Both boot sources are hard disk devices.

The AssignedSequence property of the CIM_OrderedComponent associations is set such that booting is attempted from *bootsrcsetting21* first and from *bootsrcsetting22*.

On *bootsrcsetting21*, the FailThroughSupported property value of 2 (Is Not Supported) specifies that if the *bootsrcsetting21,* "CIM:Hard-Disk:2", fails or times out, then the boot process should terminate the boot order for *bootconfigsetting20*.

In total, this use case describes a source boot order that proceeds from *bootsrcsetting1* to *bootsrcsetting21*. Bootsrcsetting2 will never be used because of its AssignedSequence value of 0 and bootsrcsetting22 will never be used because of the FailThroughSupported value on bootsrcsetting21.

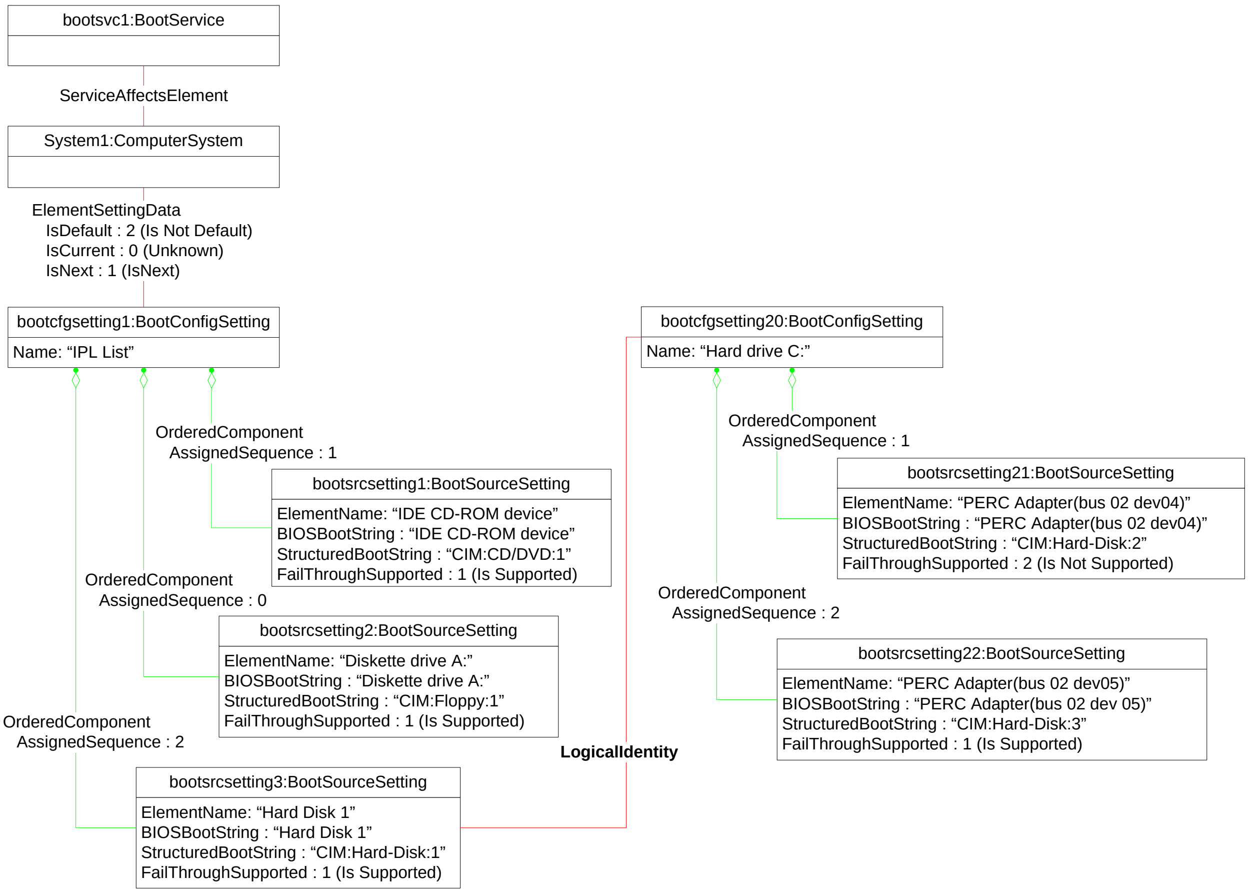

bootsvc1:BootService
ServiceAffectsElement
System1:ComputerSystem
ElementSettingData
IsDefault : 2 (Is Not Default)
IsCurrent : 0 (Unknown)
IsNext : 1 (IsNext)
bootcfgsetting1:BootConfigSetting
Name: “IPL List”
bootcfgsetting20:BootConfigSetting
Name: “Hard drive C:”
OrderedComponent
AssignedSequence : 1
bootsrcsetting1:BootSourceSetting
ElementName: “IDE CD-ROM device”
BIOSBootString : “IDE CD-ROM device”
StructuredBootString : “CIM:CD/DVD:1”
FailThroughSupported : 1 (Is Supported)
OrderedComponent
AssignedSequence : 0
bootsrcsetting2:BootSourceSetting
ElementName: “Diskette drive A:”
BIOSBootString : “Diskette drive A:”
StructuredBootString : “CIM:Floppy:1”
FailThroughSupported : 1 (Is Supported)
OrderedComponent
AssignedSequence : 2
bootsrcsetting3:BootSourceSetting
ElementName: “Hard Disk 1”
BIOSBootString : “Hard Disk 1”
StructuredBootString : “CIM:Hard-Disk:1”
FailThroughSupported : 1 (Is Supported)
LogicalIdentity
OrderedComponent
AssignedSequence : 1
bootsrcsetting21:BootSourceSetting
ElementName: “PERC Adapter(bus 02 dev04)”
BIOSBootString : “PERC Adapter(bus 02 dev04)”
StructuredBootString : “CIM:Hard-Disk:2”
FailThroughSupported : 2 (Is Not Supported)
OrderedComponent
AssignedSequence : 2
bootsrcsetting22:BootSourceSetting
ElementName: “PERC Adapter(bus 02 dev05)”
BIOSBootString : “PERC Adapter(bus 02 dev 05)”
StructuredBootString : “CIM:Hard-Disk:3”
FailThroughSupported : 1 (Is Supported)

Figure 10 – Booting from IPL and BCV Devices

9.9 Representing Settings and Boot Settings

Figure 11 shows an instance diagram for a boot configuration, *bootcfgsetting1*, which has settings that need to be applied to the computer system during the boot process. Two example concrete subclasses of CIM_SettingData, *settingdata1* and *settingdata2*, apply to concrete subclasses of CIM_LogicalDevice, *keyboard1* and *display1*. The instance of an example concrete subclass of CIM_BootSettingData is *bootsettingdata1*.

Being associated to the instance of CIM_BootConfigSetting, the settings apply to the entire boot process that uses *bootcfgsetting1*. Note that any of these settings could be associated to an instance of CIM_BootSourceSetting, which would reduce the scope of the settings to just the specified boot source.

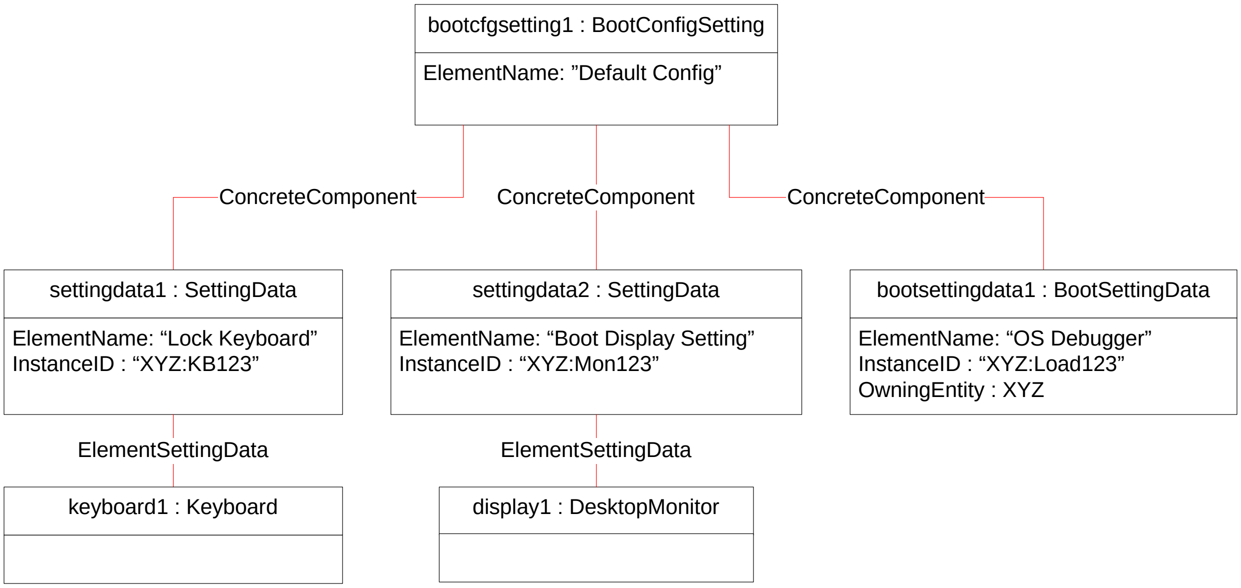

bootcfgsetting1 : BootConfigSetting
ElementName: ”Default Config”
ConcreteComponent
ConcreteComponent
ConcreteComponent
settingdata1 : SettingData
ElementName: “Lock Keyboard”
InstanceID : “XYZ:KB123”
settingdata2 : SettingData
ElementName: “Boot Display Setting”
InstanceID : “XYZ:Mon123”
bootsettingdata1 : BootSettingData
ElementName: “OS Debugger”
InstanceID : “XYZ:Load123”
OwningEntity : XYZ
ElementSettingData
ElementSettingData
keyboard1 : Keyboard
display1 : DesktopMonitor

Figure 11 – Setting Data and Boot Setting Data

9.10 Representing the Default Boot Configuration for a Computer System

Figure 12 shows an instance diagram for a Boot Configurable System, *system1*. *System1* has a single boot configuration, *bootcfgsetting2*. This boot configuration is a Default Boot Configuration, because the value of the ElementSettingData.IsDefault property is set to 1 (Is Default). There are no Next Boot Configuration or Current Boot Configuration.

Bootcfgsetting2 is associated with two instances of CIM_BootSourceSetting (*bootsrcsetting1* and *bootsrcsetting2*), through instances of CIM_OrderedComponent. The respective CIM_OrderedComponent.AssignedSequence properties designate the order in which the boot process should use the boot sources (bootsrcsetting1 followed by bootsrcsetting2).

On *bootsrcsetting1*, the FailThroughSupported property value of 1 (Is Supported) specifies that if, during the boot of *bootsrcsetting1,* the hard disk fails or times out, then the boot process should proceed to *bootsrcsetting2*, the network port using PXE.

When the system represented by *system1* is enabled, the boot process will not be initiated because there is no Next Boot Configuration for the boot process to use. The system, *system1*, will be in an enabled, but not booted, state. One could manually boot the system from this state by applying an existing boot configuration (see subclause 9.14).

System1 would initiate the boot process if the Default Boot Configuration were also the Next Boot Configuration (see subclause 9.11) or a new boot configuration is created as the Next Boot Configuration (see subclause 9.13).

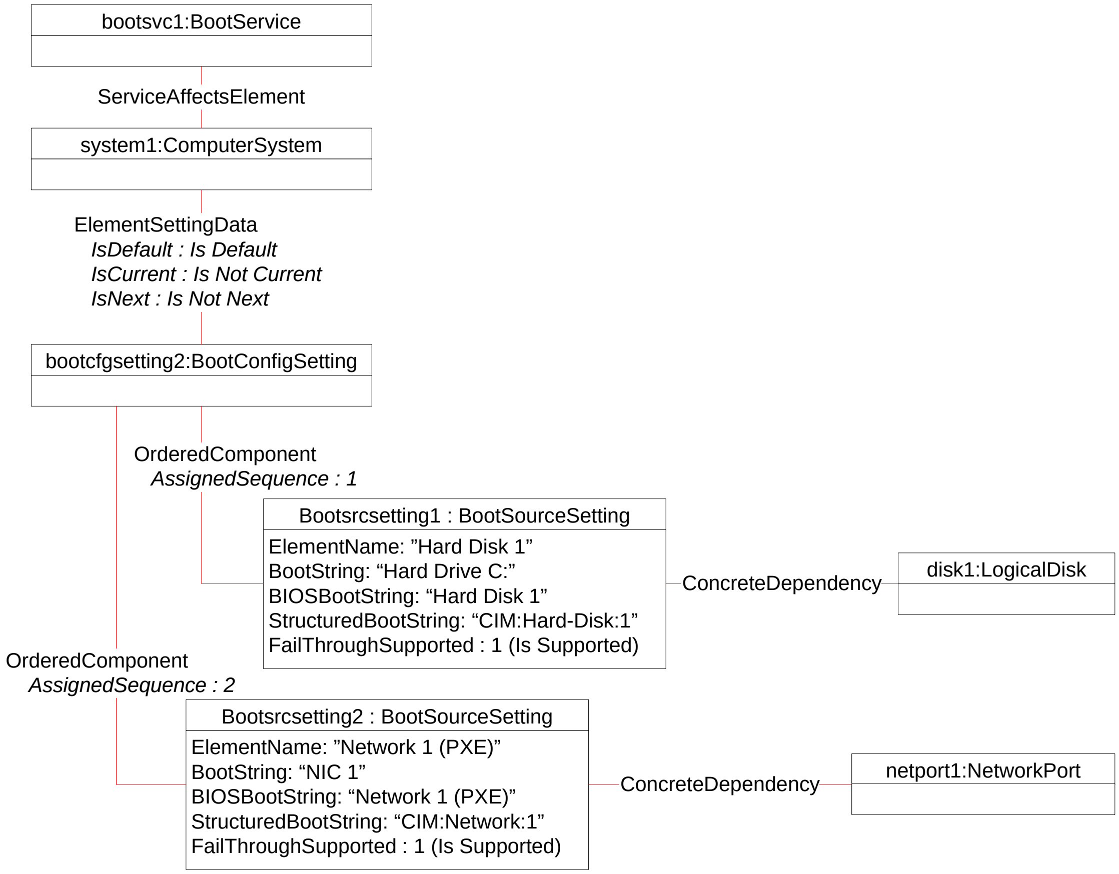

bootsvc1:BootService
ServiceAffectsElement
system1:ComputerSystem
ElementSettingData
IsDefault : Is Default
IsCurrent : Is Not Current
IsNext : Is Not Next
bootcfgsetting2:BootConfigSetting
OrderedComponent
AssignedSequence : 1
Bootsrcsetting1 : BootSourceSetting
ElementName: ”Hard Disk 1”
BootString: “Hard Drive C:”
BIOSBootString: “Hard Disk 1”
StructuredBootString: “CIM:Hard-Disk:1”
FailThroughSupported : 1 (Is Supported)
ConcreteDependency
disk1:LogicalDisk
OrderedComponent
AssignedSequence : 2
Bootsrcsetting2 : BootSourceSetting
ElementName: ”Network 1 (PXE)”
BootString: “NIC 1”
BIOSBootString: “Network 1 (PXE)”
StructuredBootString: “CIM:Network:1”
FailThroughSupported : 1 (Is Supported)
ConcreteDependency
netport1:NetworkPort

Figure 12 – Default Boot Configuration Object Diagram

9.11 Representing the Next Boot Configuration for a Computer System

Figure 13 shows an instance diagram for a Boot Configurable System, system1. *System1* has a single boot configuration, *bootcfgsetting2*. This boot configuration is a Default Boot Configuration, because the value of the ElementSettingData.IsDefault property is set to 1 (Is Default). This boot configuration is also the Next Boot Configuration, because the value of the ElementSettingData.IsNext property is set to 1 (Is Next).

Bootcfgsetting2 is associated with two instances of CIM_BootSourceSetting (*bootsrcsetting1* and *bootsrcsetting2*), through instances of CIM_OrderedComponent. The respective CIM_OrderedComponent.AssignedSequence properties designate the order in which the boot process should use the boot sources (bootsrcsetting1 followed by bootsrcsetting2).

On *bootsrcsetting1*, the FailThroughSupported property value of 1 (Is Supported) specifies that if the *bootsrcsetting1,* the hard disk fails or times out during the boot process, then the boot process should proceed to *bootsrcsetting2*, the network port using PXE.

When the system represented by *system1* is enabled, the boot process will find a Next Boot Configuration, *bootcfgsetting2 and proceed to use it to boot*.

When the system represented by *system1* is an enabled, but not booted, state. The BootService.ApplyBootConfigSetting() method can be invoked referencing *system1* as the BootConfigurableSystem parameter.

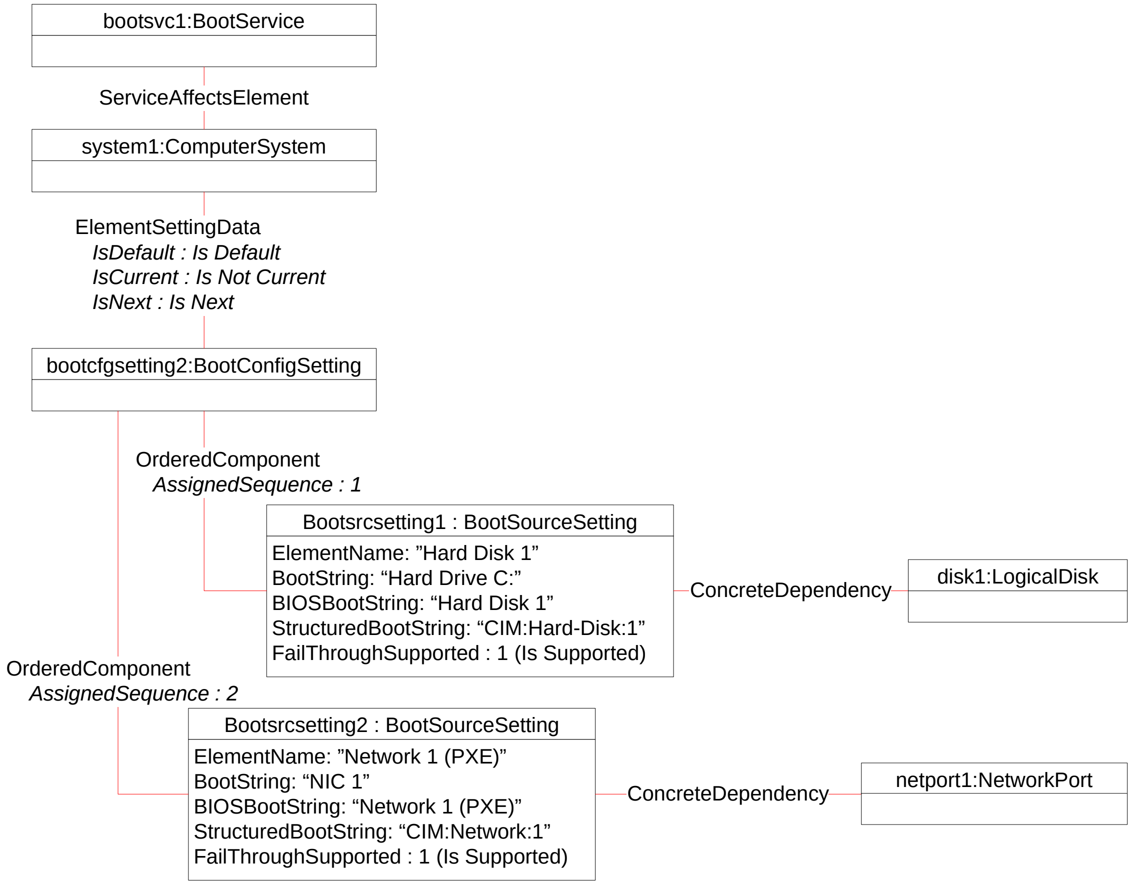

bootsvc1:BootService
ServiceAffectsElement
system1:ComputerSystem
ElementSettingData
IsDefault : Is Default
IsCurrent : Is Not Current
IsNext : Is Next
bootcfgsetting2:BootConfigSetting
OrderedComponent
AssignedSequence : 1
Bootsrcsetting1 : BootSourceSetting
ElementName: "Hard Disk 1"
BootString: "Hard Drive C:"
BIOSBootString: "Hard Disk 1"
StructuredBootString: "CIM:Hard-Disk:1"
FailThroughSupported : 1 (Is Supported)
ConcreteDependency
disk1:LogicalDisk
OrderedComponent
AssignedSequence : 2
Bootsrcsetting2 : BootSourceSetting
ElementName: "Network 1 (PXE)"
BootString: "NIC 1"
BIOSBootString: "Network 1 (PXE)"
StructuredBootString: "CIM:Network:1"
FailThroughSupported : 1 (Is Supported)
ConcreteDependency
netport1:NetworkPort

Figure 13 – Next Boot Configuration Object Diagram

9.12 Representing the Current Boot Configuration for a Booted Computer System

Figure 14 shows an instance diagram for the Boot Configurable System, *system1*, described in the previous use case (see Figure 13) after it has been successfully booted.

The boot configuration, *bootcfgsetting1*, is now the Current Boot Configuration, because the value of the ElementSettingData.IsCurrent property is set to 1 (Is Current). *Bootcfgsetting1* is still concurrently the Default Boot Configuration and the Next Boot Configuration.

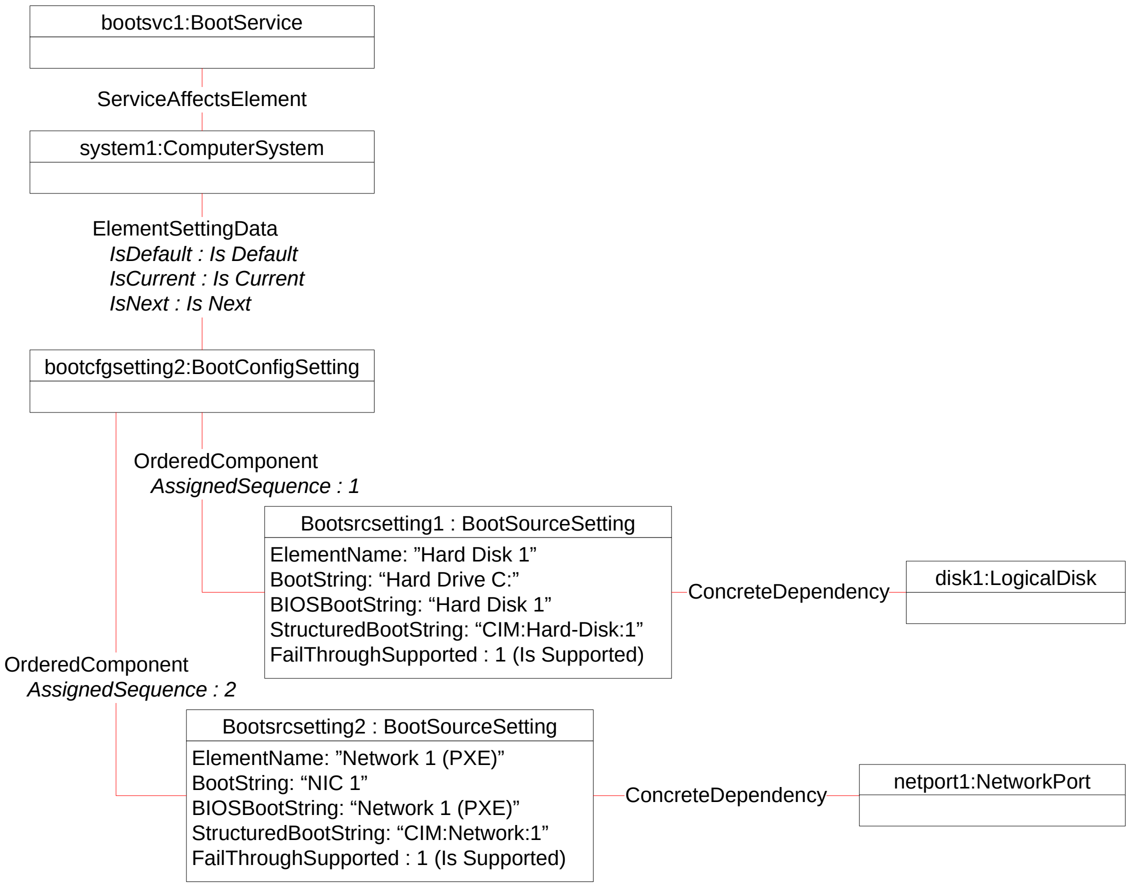

bootsvc1:BootService
ServiceAffectsElement
system1:ComputerSystem
ElementSettingData
IsDefault : Is Default
IsCurrent : Is Current
IsNext : Is Next
bootcfgsetting2:BootConfigSetting
OrderedComponent
AssignedSequence : 1
Bootsrcsetting1 : BootSourceSetting
ElementName: "Hard Disk 1"
BootString: "Hard Drive C:"
BIOSBootString: "Hard Disk 1"
StructuredBootString: "CIM:Hard-Disk:1"
FailThroughSupported : 1 (Is Supported)
ConcreteDependency
disk1:LogicalDisk
OrderedComponent
AssignedSequence : 2
Bootsrcsetting2 : BootSourceSetting
ElementName: "Network 1 (PXE)"
BootString: "NIC 1"
BIOSBootString: "Network 1 (PXE)"
StructuredBootString: "CIM:Network:1"
FailThroughSupported : 1 (Is Supported)
ConcreteDependency
netport1:NetworkPort

Figure 14 – Boot Configuration for a Booted System Object Diagram

9.13 Create a New Boot Configuration

Referencing the object diagram in Figure 12, a client could create a new boot configuration as follows:

1) From the Boot Configurable System, *system1*, find the instance of CIM_BootService that manages the boot configurable system by traversing the CIM_ServiceAffectsElement association.

2) Verify that the CreateBootConfigSetting() method is supported (see subclause 9.27). If not, a new boot configuration cannot be created.

3) Find an existing instance of CIM_BootConfigSetting to use as the template. For this use case, *bootcfgsetting2* is the only template configuration available.

4) Create the new boot configuration, *bootcfgsetting4,* by invoking the CIM_BootService.CreateBootConfigSetting( ) method. The ScopingComputerSystem parameter is set to system1 and the StartingBootConfig parameter is set to bootcfgsetting2.

Figure 15 shows the instance diagram after the CreateBootConfigSetting() method has been invoked and successfully completed on the computer system, *system1*, shown in Figure 14. The new boot configuration, *bootcfgsetting4*, is associated to *system1* through a new instance of CIM_ElementSettingData.

In the new instance of CIM_ElementSettingData, the IsDefault property is set to 2 (Is Not Default); the IsCurrent property is set to 2 (Is Not Current); and the IsNext property is set to 2 (Is Not Next).

Bootcfgsetting4 is associated through instances of CIM_OrderedComponent to two instances of CIM_BootSourceSetting (*bootsrcsetting3* and *bootsrcsetting4*), which are copies of *bootsrcsetting1* and *bootsrcsetting2*, respectively.

The instance of CIM_NetworkPort is not copied. CIM_NetworkPort is a concrete subclass of CIM_LogicalDevice, which is not part of the *Boot Control Profile*. However, an instance of CIM_ConcreteDependency has been created that associates the instance of CIM_NetworkPort to the new instance of CIM_BootSourceSetting (*bootsrcsetting4*).

CIM_LogicalDisk has been elided from the object diagram to make the diagram less cluttered, but the instance of CIM_LogicalDisk is also not copied. An instance of CIM_ConcreteDependency is created that associates the existing instance of CIM_LogicalDisk to the new instance of CIM_BootSourceSetting (*bootsrcsetting3*).

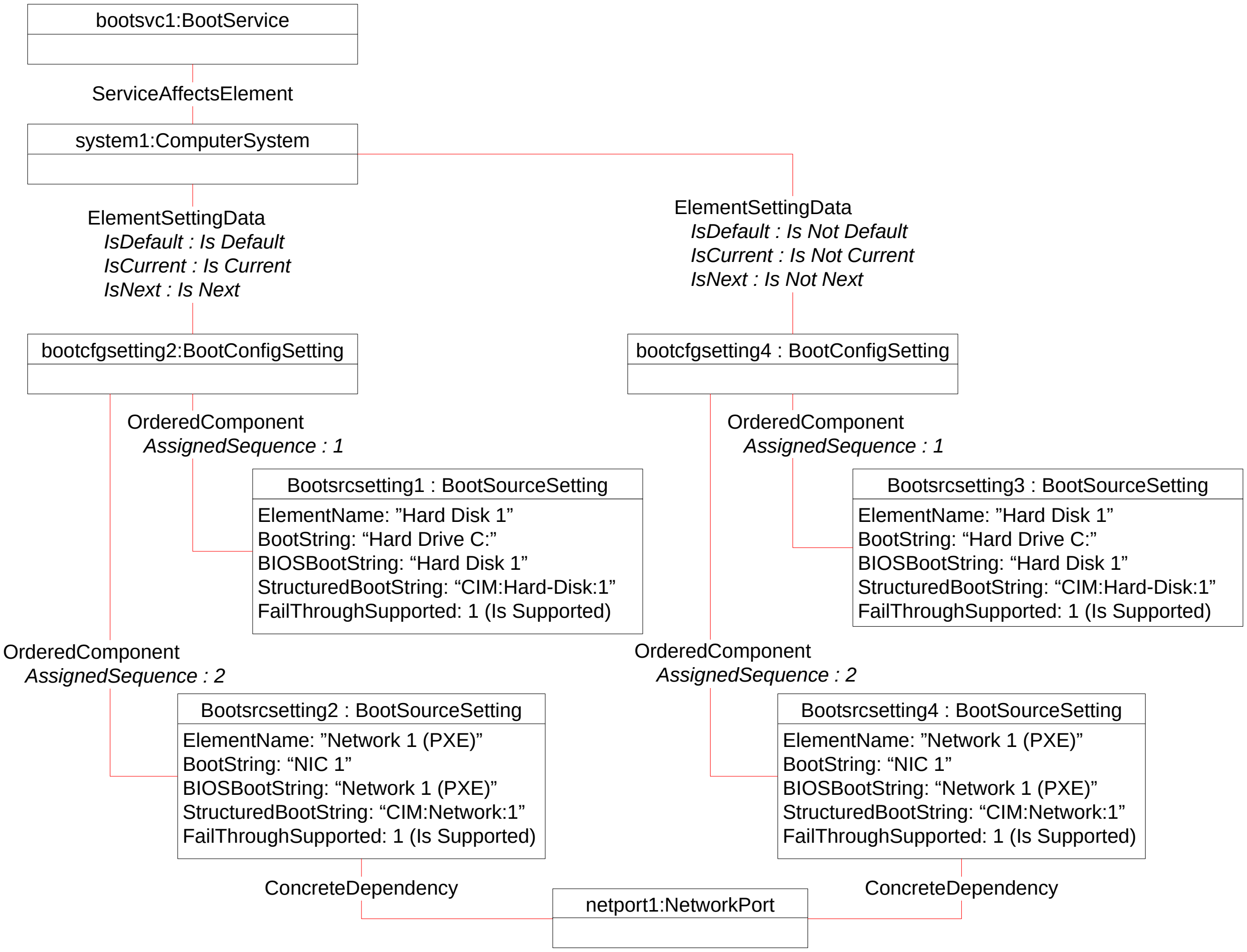

bootsvc1:BootService
ServiceAffectsElement
system1:ComputerSystem
ElementSettingData
IsDefault : Is Default
IsCurrent : Is Current
IsNext : Is Next
ElementSettingData
IsDefault : Is Not Default
IsCurrent : Is Not Current
IsNext : Is Not Next
bootcfgsetting2:BootConfigSetting
bootcfgsetting4 : BootConfigSetting
OrderedComponent
AssignedSequence : 1
OrderedComponent
AssignedSequence : 1
Bootsrcsetting1 : BootSourceSetting
ElementName: "Hard Disk 1"
BootString: "Hard Drive C:"
BIOSBootString: "Hard Disk 1"
StructuredBootString: "CIM:Hard-Disk:1"
FailThroughSupported: 1 (Is Supported)
Bootsrcsetting3 : BootSourceSetting
ElementName: "Hard Disk 1"
BootString: "Hard Drive C:"
BIOSBootString: "Hard Disk 1"
StructuredBootString: "CIM:Hard-Disk:1"
FailThroughSupported: 1 (Is Supported)
OrderedComponent
AssignedSequence : 2
OrderedComponent
AssignedSequence : 2
Bootsrcsetting2 : BootSourceSetting
ElementName: "Network 1 (PXE)"
BootString: "NIC 1"
BIOSBootString: "Network 1 (PXE)"
StructuredBootString: "CIM:Network:1"
FailThroughSupported: 1 (Is Supported)
Bootsrcsetting4 : BootSourceSetting
ElementName: "Network 1 (PXE)"
BootString: "NIC 1"
BIOSBootString: "Network 1 (PXE)"
StructuredBootString: "CIM:Network:1"
FailThroughSupported: 1 (Is Supported)
ConcreteDependency
netport1:NetworkPort
ConcreteDependency

Figure 15 – System with New CIM_BootConfigSetting

9.14 Apply an Existing Boot Configuration

Referencing the object diagram in Figure 12, a client could apply a boot configuration as follows:

1) Find the instance of CIM_BootService for the boot configurable system as outlined in subclause 9.15.

2) Verify that the ApplyBootConfigSetting() method is supported (see subclause 9.28). If not, a boot configuration cannot be applied.

3) Find the existing instances of CIM_BootConfigSetting for *system1* (see subclause 9.18). In this example, this results in *bootcfgsetting2*. Pick one of them to use as the boot configuration to apply.

4) Apply the selected boot configuration, *bootcfgsetting2*, by invoking the CIM_BootService.ApplyBootConfigSetting() method. The ScopingComputerSystem parameter is set to *system1* and the BootConfigSetting parameter is set to *bootcfgsetting2*.

The ApplyBootConfigSetting() method will boot *system1* by applying the boot configuration specified in *bootcfgsetting2*. If *system1* is currently booted, an implementation has the option of rejecting the ApplyBootConfigSetting() request or of rebooting the system.

9.15 Find the Boot Service for a Computer System

A client can find the boot service for a given computer system as follows:

1) For the instance of CIM_ComputerSystem, representing the given computer system, select the referencing instance of CIM_BootService, representing the boot control service for the server, through the CIM_ServiceAffectsElement association.

9.16 Find the Boot Configuration for a Computer System

A client can find the boot configurations for a computer system as follows:

1) From the instance of CIM_ComputerSystem, enumerate the CIM_ElementSettingData associations with CIM_BootConfigSetting as the SettingData reference.

9.17 Find the Boot Service Capabilities for a Computer System

A client can find the boot service capabilities for a computer system as follows:

1) Find the boot service for the computer system as specified in subclause 9.15.

2) Select the instance of CIM_BootServiceCapabilities through the CIM_ElementCapabilities association.

9.18 Find the Current Boot Configuration for a Computer System

A client can find the current boot configuration for a computer system as follows:

1) From the instance of CIM_ComputerSystem, enumerate the CIM_ElementSettingData associations with CIM_BootConfigSetting as the SettingData reference.

2) Find the instance of CIM_ElementSettingData whose IsCurrent property is set to 1 (Is Current).

3) The CIM_BootConfigSetting instance referenced by this association instance represents the current boot configuration.

9.19 Find the Default Boot Configuration for a Computer System

A client can find the default boot configuration for a computer system as follows:

1) From the instance of CIM_ComputerSystem, enumerate the CIM_ElementSettingData associations with CIM_BootConfigSetting as the SettingData reference.

2) Find the instance of CIM_ElementSettingData whose IsDefault property is set to 1 (Is Default).

3) The CIM_BootConfigSetting instance referenced by this association instance represents the default boot configuration.

9.20 Find the Boot Configuration that Will Be Used during the Next Reboot for a Computer System

A client can find the boot configuration that will be used during a computer system's next reboot as follows:

1) For the instance of CIM_ComputerSystem, enumerate the CIM_ElementSettingData associations with CIM_BootConfigSetting as the SettingData reference.

2) Find the CIM_ElementSettingData instances for the instance whose IsNext property is set to 3 (Is Next For Single Use). The CIM_BootConfigSetting instance referenced by this association instance represents the next boot configuration.

3) If no instance is found, find the instance of CIM_ElementSettingData whose IsNext property is set to 1 (Is Next). The CIM_BootConfigSetting instance referenced by this association instance represents the next boot configuration.

9.21 Make a Boot Configuration Applicable for Subsequent Reboots

A client can make a boot configuration apply to a computer system for subsequent reboots as follows:

1) From the instance of CIM_ComputerSystem, find the CIM_BootConfigSetting of interest as outlined in subclauses 9.9 through 9.12.

2) On the instance of the CIM_ElementSettingData association that associates the instance of CIM_ComputerSystem to the instance of CIM_BootConfigSetting, use the intrinsic ModifyInstance() to change the IsNext property to 1 (Is Next).

Note that this boot configuration applies for all subsequence reboots, unless it is overridden by a Next Single Use Boot Configuration that is associated to the CIM_ComputerSystem of interest.

9.22 Make a Boot Configuration Applicable for the Next Reboot Only

A client can make a boot configuration apply to a computer system for only the next reboot as follows:

1) From the instance of CIM_ComputerSystem, find the CIM_BootConfigSetting of interest as outlined in subclauses 9.9 through 9.12.

2) On the instance of the CIM_ElementSettingData association that associates the instance of CIM_ComputerSystem to the instance of CIM_BootConfigSetting, use the intrinsic ModifyInstance() to change the IsNext property to 3 (Is Next For Single Use).

The behavior of this property after the next boot is specified in subclause 7.4.5.

9.23 Determine If the Computer System Supports PXE Boot

A client can determine if the computer system supports PXE boot as follows:

1) For the instance of CIM_ComputerSystem enumerate its instances of CIM_BootConfigSetting as outline in subclause 9.18.

2) For each instance of CIM_BootConfigSetting, enumerate the instances of CIM_BootSourceSetting.

3) For each CIM_BootSourceSetting, inspect the BootString, BIOSBootString, or StructuredBootString property to determine if PXE is supported.

9.24 Find the Boot Order for a Computer System for the Next Reboot

This use case references the object diagram in Figure 10, which represents a complex boot order.

A client can find the boot order for the next reboot of a computer system as follows:

1) From the instance of CIM_ComputerSystem, *system1,* find the CIM_BootConfigSetting that will be used during the next reboot, *bootcfgsetting1* (see subclause 3)).

2) Determine the boot order for *bootcfgsetting1* by enumerating the CIM_OrderedComponent associations with *bootcfgsetting1* as the GroupComponent reference. The results in this example would be *bootsrcsetting1*, *bootsrcsetting2* and *bootsrcsetting3*.

3) Use the CIM_OrderedComponent.AssignedSequence property to determine the boot order. The boot order in this example will be bootsrcsetting1 followed by bootsrcsetting3. The boot source represented by bootsrcsettin2 will be ignored because its associated AssignedSequence property value is 0.

4) For each boot source, determine whether any it contains additional boot sources by checking for a CIM_LogicalIdentity association to an instance of CIM_BootConfigSetting; in this example, *bootcfgsetting20*, and repeat steps in this subclause recursively to find the boot order of the associated boot sources.

9.25 Change the Boot Order for a Computer System

This use case references the object diagram in Figure 13.

A client can change the boot order for a computer system as follows:

1) Find the boot configuration of interest from the set of boot configurations for the computer system as outlined in subclause 9.18.

2) Find the set of boot sources for the boot configuration by following the OrderedComponent associations from the selected boot configuration representation (bootcfgsetting2) to all instances of CIM_BootSourceSetting. In this example, this results in bootsrcsetting1 and bootsrcsetting2.

3) Determine the desired boot order.

4) Create an array of CIM_BootSourceSetting references. Assign the existing boot sources to the array in the new order. For instance, if one wanted to reverse the boot order in this example, the array would contain bootsrcsetting2 at index 0 and bootsrcsetting1 at index 1.

5) Invoke the ChangeBootOrder() method on the selected instance of CIM_BootConfigSetting. The Source parameter is set to the array created above.

NOTE: The order of each boot configuration must be changed independently. Thus if the computer system has a complex boot structure, such as that illustrated in Figure 10, changing the boot order for the system may require changing the boot order for multiple CIM_BootConfigSetting instances.

9.26 Determine Whether BootService.ElementName Is Modifiable

A client can determine whether the ElementName can be modified as follows:

1) Find the CIM_BootServiceCapabilities instance associated with the CIM_BootService instance through the CIM_ElementCapabilities association.

2) If a CIM_BootConfigCapabilities instance cannot be found, then the CIM_BootService.ElementName property cannot be modified.

3) Query the value of the CIM_BootServiceCapabilities.ElementNameEditSupported.

4) If the value is TRUE, the CIM_BootService.ElementName property can be modified

5) If the value of ElementNameEditSupported has a value of FALSE, then the CIM_BootService.ElementName property cannot be modified.

9.27 Determine Whether a New Boot Configuration Can Be Created

A client can determine whether a new boot configuration can be created as follows:

1) Find the CIM_BootServiceCapabilities instance that is associated with the CIM_BootService instance through the CIM_ElementCapabilities association.

2) Query the value of the CIM_BootServiceCapabilities.BootConfigCapabilities property array. If the array contains the value 2 (Creates Boot Configuration), the client's ability to create a boot configuration is supported.

3) If the BootConfigCapabilities property array does not contain the value 2 (Creates Boot Configuration), or there is not an instance of CIM_BootServiceCapabilities associated with the CIM_BootService instance, a boot configuration cannot be created.

9.28 Determine Whether a Boot Configuration Can Be Applied

A client can determine whether a boot configuration can be manually applied to the boot configurable systems as follows:

1) Find the CIM_BootServiceCapabilities instance that is associated with the CIM_BootService instance through the CIM_ElementCapabilities association.

2) Query the value of the CIM_BootServiceCapabilities.BootConfigCapabilities property array. If the array contains the value 3 (Applies Boot Configuration), the client's ability to manually apply a boot configuration is supported.

3) If the BootConfigCapabilities property array does not contain the value 3 (Applies Boot Configuration), or there is not an instance of CIM_BootServiceCapabilities associated with the CIM_BootService instance, a boot configuration cannot be manually applied.

10 CIM Elements

Table 22 shows the instances of CIM Elements for this profile. Instances of the following CIM Elements shall be implemented as described in Table 22. Clauses 7 (“Implementation”) and 8 (“Methods”) may impose additional requirements on these elements.

Table 22 CIM Elements – Boot Control Profile

Element Name	Requirement	Description
CIM_RegisteredProfile	Mandatory	See subclause 10.1.
CIM_BootService	Mandatory	See subclause 10.2.
CIM_BootServiceCapabilities	Optional	See subclause 10.3.
CIM_BootConfigSetting	Mandatory	See subclause 10.4.
CIM_BootSettingData	Optional	See subclause 10.5.
CIM_BootSourceSetting	Mandatory	See subclause 10.6.
CIM_ConcreteComponent	Optional	See subclause 10.7.
CIM_ConcreteDependency	Optional	See subclause 10.8.
CIM_ElementCapabilities	Optional	See subclause 10.9.
CIM_ElementSettingData	Mandatory	See subclause 10.10.
CIM_HostedService	Mandatory	See subclause 10.11.
CIM_LogicalIdentity	Conditional	See subclause 10.12.
CIM_OrderedComponent	Mandatory	See subclause 10.13.
CIM_ServiceAffectsElement	Mandatory	See subclause 10.14.

10.1 CIM_RegisteredProfile

CIM_RegisteredProfile identifies the *Boot Control Profile* in order for a client to determine whether an instance of CIM_ComputerSystem is conformant with this profile. The CIM_RegisteredProfile class is defined by the *Profile Registration Profile*. With the exception of the mandatory values specified for the properties below, the behavior of the CIM_RegisteredProfile instance is per the *Profile Registration Profile*. Table 23 contains the requirements for elements of this class.

Table 23 – Class: CIM_RegisteredProfile

Elements	Requirement	Notes
RegisteredName	Mandatory	This property shall have a value of "Boot Control".
RegisteredVersion	Mandatory	This property shall have a value of "1.0.1".
RegisteredOrganization	Mandatory	This property shall have a value of 2 (DMTF).

NOTE: Previous versions of this document included the suffix 'Profile' for the RegisteredName value. If implementations querying for RegisteredName value find the suffix 'Profile', they should ignore the suffix, with any surrounding white spaces, before any comparison is done with the value as specified in this document.

10.2 CIM_BootService

The CIM_BootService class represents the ability to view and control the boot settings of a computer system. Table 24 contains the requirements for elements of this class.

Table 24 – Class: CIM_BootService

Elements	Requirement	Notes
CreationClassName	Mandatory	**Key**
Name	Mandatory	**Key**
SystemCreationClassName	Mandatory	**Key**
SystemName	Mandatory	**Key**
ElementName	Mandatory	See subclause 7.1.1.
CreateBootConfigSetting( )	Conditional	See subclause 8.1.
ApplyBootConfigSetting()	Conditional	See subclause 8.2.

10.3 CIM_BootServiceCapabilities

Support of the CIM_BootServiceCapabilities class is optional.

When supported, CIM_BootServiceCapabilities is used to indicate the capabilities of the boot service. Table 25 contains the requirements for elements of this class.

Table 25 – Class: CIM_BootServiceCapabilities

Elements	Requirement	Notes
InstanceID	Mandatory	**Key**
ElementName	Mandatory	This property shall be a character string of variable length (pattern “.*”).
ElementNameEditSupported	Mandatory	See subclause 7.1.2
BootConfigCapabilities	Mandatory	See subclauses 7.5, 7.6, and 7.7.
OtherBootConfigCapabilities	Conditional	See subclause 7.3.1.
BootStringsSupported	Optional	See subclause 7.8.

10.4 CIM_BootConfigSetting

The CIM_BootConfigSetting class represents a boot configuration of a computer system. Table 26 contains the requirements for elements of this class.

Table 26 – Class: CIM_BootConfigSetting

Elements	Requirement	Notes
InstanceID	Mandatory	**Key**
ElementName	Mandatory	This property shall be a character string of variable length (pattern “.*”).
ChangeBootOrder( )	Conditional	See subclause 7.9 and 8.3.

10.5 CIM_BootSettingData

Support of the CIM_BootSettingData class is optional.

The CIM_BootSettingData class represents the settings that apply during booting of a computer system. Table 27 contains the requirements for elements of this class.

For each property added in a concrete subclass of CIM_BootSettingData, there shall be a Description qualifier that contains a string which describes the setting. When the range of the setting is bounded and discrete, the Values and ValueMap qualifiers should contain the values and name of the values, respectively, which are applicable for the setting.

Table 27 – Class: CIM_BootSettingData

Elements	Requirement	Notes
InstanceID	Mandatory	**Key**
ElementName	Mandatory	This property shall be a character string of variable length (pattern “.*”).
OwningEntity	Mandatory	None

10.6 CIM_BootSourceSetting

Support of the CIM_BootSourceSetting class is optional.

The CIM_BootSourceSetting class represents a boot source, from which booting is attempted during the boot process. Table 28 contains the requirements for elements of this class.

Table 28 – Class: CIM_BootSourceSetting

Elements	Requirement	Notes
InstanceID	Mandatory	**Key**
ElementName	Mandatory	See subclause 7.8.2.
BootString	Conditional	See subclause 7.8.3.
BIOSBootString	Conditional	See subclause 7.8.4.
StructuredBootString	Conditional	See subclause 7.8.5.
FailThroughSupported	Mandatory	See subclause 7.11.2.

10.7 CIM_ConcreteComponent

Subclause 10.7 describes optional behavior.

10.7.1 Relating CIM_BootConfigSetting to a Concrete Subclass of CIM_SettingData

When supported, the CIM_ConcreteComponent association is used to relate an instance of a concrete subclass of CIM_SettingData to a CIM_BootConfigSetting instance. Table 29 contains the requirements for elements of this class.

Table 29 – Class: CIM_ConcreteComponent – Use 1

Elements	Requirement	Notes
GroupComponent	Mandatory	This property shall be a reference to an instance of the CIM_BootConfigSetting class. See subclause 7.12. Cardinality is "*".
PartComponent	Mandatory	This property shall be a reference to an instance of a concrete subclass of the CIM_SettingData class. See subclause 7.12. Cardinality is "*".

10.7.2 Relating CIM_BootConfigSetting to a Concrete Subclass of CIM_BootSettingData

When supported, the CIM_ConcreteComponent association is used to relate an instance of a concrete subclass of CIM_BootSettingData to a CIM_BootConfigSetting instance. Table 30 contains the requirements for elements of this class.

Table 30 – Class: CIM_ConcreteComponent – Use 2

Elements	Requirement	Notes
GroupComponent	Mandatory	This property shall be a reference to an instance of the CIM_BootConfigSetting class. See subclause 7.12. Cardinality is "0..1".
PartComponent	Mandatory	This property shall be a reference to an instance of a concrete subclass of the CIM_BootSettingData class. See subclause 7.12. Cardinality is "*".

10.7.3 Relating CIM_BootSourceSetting to a Concrete Subclass of CIM_SettingData

When supported, the CIM_ConcreteComponent association is used to relate an instance of a concrete subclass of CIM_SettingData to a CIM_BootSourceSetting instance. Table 31 contains the requirements for elements of this class.

Table 31 – Class: CIM_ConcreteComponent – Use 3

Elements	Requirement	Notes
GroupComponent	Mandatory	This property shall be a reference to an instance of the CIM_BootSourceSetting class. See subclause 7.12. Cardinality is "*".
PartComponent	Mandatory	This property shall be a reference to an instance of a concrete subclass of the CIM_SettingData class. See subclause 7.12. Cardinality is "*".

10.7.4 Relating CIM_BootSourceSetting to a Concrete Subclass of CIM_BootSettingData

When supported, the CIM_ConcreteComponent association is used to relate an instance a concrete subclass of CIM_BootSettingData to a CIM_BootSourceSetting instance. Table 32 contains the requirements for elements of this class.

Table 32 – Class: CIM_ConcreteComponent – Use 4

Elements	Requirement	Notes
GroupComponent	Mandatory	This property shall be a reference to an instance of the CIM_BootSourceSetting class. See subclause 7.12. Cardinality is "0..1".
PartComponent	Mandatory	This property shall be a reference to an instance of a concrete subclass of the CIM_BootSettingData class. See subclause 7.12. Cardinality is "*".

10.8 CIM_ConcreteDependency

Subclause 10.8 describes optional behavior.

When supported, the CIM_ConcreteDependency association is used to relate the dependency of a CIM_BootSourceSetting instance on an instance of a concrete subclass of CIM_LogicalDevice. Table 33 contains the requirements for elements of this class.

Table 33 – Class: CIM_ConcreteDependency

Elements	Requirement	Notes
Antecedent	Mandatory	This property shall be a reference to an instance of a concrete subclass of the CIM_LogicalDevice class. See subclause 7.8.5.2. Cardinality is "*".
Dependent	Mandatory	This property shall be a reference to an instance of the CIM_BootSourceSetting. See subclause 7.8.5.2. Cardinality is "*".

10.9 CIM_ElementCapabilities

Subclause 10.9 describes optional behavior.

When supported, the CIM_ElementCapabilities association is used to relate an instance of CIM_BootServiceCapabilities with an instance of CIM_BootService. Table 34 contains the requirements for elements of this class.

Table 34 – Class: CIM_ElementCapabilities

Elements	Requirement	Notes
ManagedElement	Mandatory	This property shall be a reference to an instance of CIM_BootService. See subclause 7.1. Cardinality is "1..*".
Capabilities	Mandatory	This property shall be a reference to an instance of CIM_BootServiceCapabilities. See subclause 7.1. Cardinality is "0..1".

10.10 CIM_ElementSettingData

The CIM_ElementSettingData association is used to relate the CIM_BootConfigSetting instance to the CIM_ComputerSystem instance to which it applies. Table 35 contains the requirements for elements of this class.

Table 35 – Class: CIM_ElementSettingData

Elements	Requirement	Notes
ManagedElement	Mandatory	This property shall be a reference to an instance of the CIM_ComputerSystem class. See subclause 7.4.1. Cardinality is "0..1".
SettingData	Mandatory	This property shall be a reference to an instance of the CIM_BootConfigSetting class. See subclause 7.4.1. Cardinality is "*".
IsDefault	Mandatory	See subclause 7.3.
IsCurrent	Mandatory	See subclause 7.3.
IsNext	Mandatory	See subclause 7.3.

10.11 CIM_HostedService

The CIM_HostedService association is used to relate the CIM_BootService to the CIM_ComputerSystem on which it is hosted. Table 36 contains the requirements for elements of this class.

Table 36 – Class: CIM_HostedService

Elements	Requirement	Notes
Antecedent	Mandatory	This property shall be a reference to the scoping instance of the CIM_ComputerSystem class. See subclause 5. Cardinality is "1".
Dependent	Mandatory	This property shall be a reference to an instance of the CIM_BootService. See subclause 5. Cardinality is "*".

10.12 CIM_LogicalIdentity

Support of the CIM_LogicalIdentity association is conditional.

Conditional Requirement: The support is required if instances of CIM_BootSourceSetting are used to represent aggregated boot sources; see subclause 7.9.

When supported, CIM_LogicalIdentity is used to associate an instance of CIM_BootSourceSetting with an instance of CIM_BootConfigSetting. Table 37 contains the requirements for elements of this class.

Table 37 – Class: CIM_LogicalIdentity

Elements	Requirement	Notes
SystemElement	Mandatory	This property shall be a reference to an instance of the CIM_BootSourceSetting class. See subclause 7.10.3. Cardinality is "0..1"
SameElement	Mandatory	This property shall be a reference to an instance of the CIM_BootConfigSetting class. See subclause 7.10.3. Cardinality is "0..1"

10.13 CIM_OrderedComponent

Support of the CIM_OrderedComponent association is conditional.

Conditional Requirement: The support is required if the CIM_BootSourceSetting instances are used to represent boot sources; see 7.11.1.

When supported, the CIM_OrderedComponent association is used to indicate the order in which CIM_BootSourceSetting instances should be attempted for a CIM_BootConfigSetting instance. Table 38 contains the requirements for elements of this class.

Table 38 – Class: CIM_OrderedComponent

Elements	Requirement	Notes
GroupComponent	Mandatory	This property shall be a reference to an instance of the CIM_BootConfigSetting class. See subclause 7.11.1. Cardinality is "1..*".
PartComponent	Mandatory	This property shall be a reference to an instance of the CIM_BootSourceSetting class. See subclause 7.11.1. Cardinality is "1..*".
AssignedSequence	Mandatory	See subclause 7.11.1.1.

10.14 CIM_ServiceAffectsElement

The CIM_ServiceAffectsElement association is used to associate the CIM_BootService instance with a CIM_ComputerSystem instance that it affects. Table 39 contains the requirements for elements of this class.

Table 39 – Class: CIM_ServiceAffectsElement

Elements	Requirement	Notes
AffectingElement	Mandatory	This property shall be a reference to an instance of the CIM_BootService class. See subclause 7.2. Cardinality is "0..1".
AffectedElement	Mandatory	This property shall be a reference to an instance of the CIM_ComputerSystem class. See subclause 7.2. Cardinality is "1..*".
ElementEffects	Mandatory	Matches 5 (Manages)

ANNEX A
(informative)

Change Log

Version	Date	Description
1.0.0a	10/10/2006	Preliminary Standard
1.0.0	11/03/2008	Final Standard
1.0.1	06/22/2009	DMTF Standard Release