
**Systèmes d'automatisation et de gestion
technique du bâtiment —**

Partie 5:

Protocole de communication de données

AMENDEMENT 1

iTeh STANDARD PREVIEW

(standards.iteh.ai) *Building automation and control systems —*

Part 5: Data communication protocol

ISO 16484-5:2007/Amd 1:2009

AMENDMENT 1
<https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-7b3255dcdac9/iso-16484-5-2007-amd-1-2009>



PDF – Exonération de responsabilité

Le présent fichier PDF peut contenir des polices de caractères intégrées. Conformément aux conditions de licence d'Adobe, ce fichier peut être imprimé ou visualisé, mais ne doit pas être modifié à moins que l'ordinateur employé à cet effet ne bénéficie d'une licence autorisant l'utilisation de ces polices et que celles-ci y soient installées. Lors du téléchargement de ce fichier, les parties concernées acceptent de fait la responsabilité de ne pas enfreindre les conditions de licence d'Adobe. Le Secrétariat central de l'ISO décline toute responsabilité en la matière.

Adobe est une marque déposée d'Adobe Systems Incorporated.

Les détails relatifs aux produits logiciels utilisés pour la création du présent fichier PDF sont disponibles dans la rubrique General Info du fichier; les paramètres de création PDF ont été optimisés pour l'impression. Toutes les mesures ont été prises pour garantir l'exploitation de ce fichier par les comités membres de l'ISO. Dans le cas peu probable où surviendrait un problème d'utilisation, veuillez en informer le Secrétariat central à l'adresse donnée ci-dessous.

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO 16484-5:2007/Amd 1:2009](https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-7b3255dcdac9/iso-16484-5-2007-amd-1-2009)

<https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-7b3255dcdac9/iso-16484-5-2007-amd-1-2009>



DOCUMENT PROTÉGÉ PAR COPYRIGHT

© ISO 2009

Droits de reproduction réservés. Sauf prescription différente, aucune partie de cette publication ne peut être reproduite ni utilisée sous quelque forme que ce soit et par aucun procédé, électronique ou mécanique, y compris la photocopie et les microfilms, sans l'accord écrit de l'ISO à l'adresse ci-après ou du comité membre de l'ISO dans le pays du demandeur.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Publié en Suisse

Avant-propos

L'ISO (Organisation internationale de normalisation) est une fédération mondiale d'organismes nationaux de normalisation (comités membres de l'ISO). L'élaboration des Normes internationales est en général confiée aux comités techniques de l'ISO. Chaque comité membre intéressé par une étude a le droit de faire partie du comité technique créé à cet effet. Les organisations internationales, gouvernementales et non gouvernementales, en liaison avec l'ISO participent également aux travaux. L'ISO collabore étroitement avec la Commission électrotechnique internationale (CEI) en ce qui concerne la normalisation électrotechnique.

Les Normes internationales sont rédigées conformément aux règles données dans les Directives ISO/CEI, Partie 2.

La tâche principale des comités techniques est d'élaborer les Normes internationales. Les projets de Normes internationales adoptés par les comités techniques sont soumis aux comités membres pour vote. Leur publication comme Normes internationales requiert l'approbation de 75 % au moins des comités membres votants.

L'attention est appelée sur le fait que certains des éléments du présent document peuvent faire l'objet de droits de propriété intellectuelle ou de droits analogues. L'ISO ne saurait être tenue pour responsable de ne pas avoir identifié de tels droits de propriété et averti de leur existence.

L'amendement à l'ISO 16484-5 a été élaboré par le comité technique ISO/TC 205, *Conception de l'environnement intérieur des bâtiments*.

STANDARD PREVIEW
(standards.iteh.ai)
<https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-7b3255dcdac9/iso-16484-5-2007-amd-1-2009>

Introduction

L'objectif du présent Addendum consiste à apporter un certain nombre de modifications indépendantes importantes à la norme BACnet. Les modifications sont récapitulées ci-après.

135-2004a-1, p.1 : Revoir les types d'objet Life Safety Point et Life Safety Zone pour modifier leur comportement lorsqu'ils sont hors service.

135-2004c-1, p.4 . Ajout de l'interface des services Web BACnet/WS.

135-2004d-1, p.39 : Ajouter un nouveau type d'objet Structured View.

135-2004d-2, p.45 : Autoriser l'accusé de réception de notifications d'événements TO-OFFNORMAL non prévues.

135-2004d-3, p.46 : Atténuer les exigences des BIBB Private Transfer et Text Message.

135-2004d-4, p.47 : Exclure les notifications LIFE_SAFETY et BUFFER_READY des BIBB de notifications d'alarme.

135-2004d-5, p.49 : Établir les exigences minimales d'un dispositif BACnet avec une couche application.

135-2004d-6, p.51 : Supprimer l'exigence du BIBB DM-DOB-A des profils de dispositif B-OWS et B-BC.

135-2004d-7, p.52 : Atténuer les valeurs obligatoires des limitations de délai et nouvelles tentatives des APDU lorsque cela est possible, et modifier les valeurs par défaut.

135-2004d-8, p.53 : Corriger l'erreur de traitement EventCount de la machine à états Nœud maître MS/TP.

135-2004d-9, p.54 : Permettre aux routeurs d'utiliser un numéro de réseau local dans Device_Address_Binding.

135-2004d-10, p.55 : Identifier les propriétés inscriptibles sous condition.

135-2004d-11, p.56 : Spécifier des retours d'erreur pour le service AcknowledgeAlarm.

135-2004e-1, p.58 : Ajouter un nouveau type d'objet Load Control.

135-2004f-1, p.71 : Ajouter un nouveau type d'objet Access Door.

Dans le présent Amendement, le texte ajouté aux articles existants de l'ISO 16484-5 est indiqué en *italique*, tandis que les suppressions sont indiquées par du texte ~~barré~~. Du texte en clair est utilisé tout du long lorsque de nouveaux articles entiers sont ajoutés.

Systèmes d'automatisation et de gestion technique du bâtiment —

Partie 5: Protocole de communication de données

AMENDEMENT 1

135-2004a-1. Revoir les types d'objet Life Safety Point et Life Safety Zone pour le fonctionnement hors-service.

Addendum 135-2004a-1

[Modifier le **Tableau 12-18**, p. 194]

iTech STANDARD PREVIEW
(standards.iteh.ai)

Tableau 12-18. Propriétés du type d'objet Life Safety Point

Identifiant de propriété	Type de données de propriété	Code de conformité
...
Present_Value	BACnetLifeSafetyState	R ⁺ R
Tracking_Value	BACnetLifeSafetyState	ØR ⁺
...

[Modifier le paragraphe **12.15.4**, p. 195]

12.15.4 Present_Value

Cette propriété, de type BACnetLifeSafetyState, reflète l'état de l'objet Life Safety Point. La méthode permettant de dériver Present_Value doit être d'ordre local. Present_Value peut verrouiller des valeurs d'état non-NORMAL jusqu'à la réinitialisation. ~~La propriété Present_Value doit être inscriptible lorsque Out_Of_Service est TRUE.~~

[Modifier le paragraphe **12.15.5**, p. 195]

12.15.5 Tracking_Value

Cette propriété ~~optionnelle~~, de type BACnetLifeSafetyState, reflète l'état non verrouillé de l'objet Life Safety Point. La méthode permettant de dériver l'état doit être d'ordre local. Contrairement à Present_Value, qui peut verrouiller des valeurs à l'état non-NORMAL jusqu'à la réinitialisation, Tracking_Value doit continuer à suivre les changements d'état. ~~La propriété Tracking_Value doit être inscriptible lorsque Out_Of_Service est TRUE.~~

[Modifier le paragraphe 12.15.11, p. 196]

12.15.11 Out_Of_Service

La propriété Out_Of_Service, de type BOOLEAN, indique si oui (TRUE) ou non (FALSE) l'entrée ou le processus que représente l'objet n'est pas en service. Cela signifie que les modifications apportées à la propriété Present_Value Tracking_Value sont découplées de l'entrée ou du processus lorsque la valeur de Out_Of_Service est TRUE. En outre, la propriété Reliability et l'état correspondant de l'indicateur FAULT de la propriété Status_Flags doivent être découplés lorsque Out_Of_Service est TRUE. Alors que la propriété Out_Of_Service est TRUE, les propriétés Present_Value Tracking_Value et Reliability peuvent être modifiées en valeurs quelconques afin de simuler des conditions fixes spécifiques ou à des fins d'essai. D'autres fonctions dépendant de l'état des propriétés Present_Value Tracking_Value ou Reliability doivent répondre aux modifications apportées à ces propriétés alors que Out_Of_Service est TRUE, comme si ces changements s'étaient produits au niveau de l'entrée ou du processus.

[Modifier le Tableau 12-19, p. 200]

Tableau 12-19. Propriétés du type d'objet Life Safety Zone

Identifiant de propriété	Type de données de propriété	Code de conformité
...
Present_Value	BACnetLifeSafetyState	R ⁺ R
Tracking_Value	BACnetLifeSafetyState	OR'
...

[Modifier le paragraphe 12.16.4, p. 201]

[ISO 16484-5:2007/Amd 1:2009](https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-7b3255dcdac9/iso-16484-5-2007-amd-1-2009)

12.16.4 Present_Value <https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-7b3255dcdac9/iso-16484-5-2007-amd-1-2009>

Cette propriété, de type BACnetLifeSafetyState, reflète l'état de l'objet Life Safety Zone. La méthode permettant de dériver Present_Value doit être d'ordre local. Present_Value peut verrouiller des valeurs d'état non-NORMAL jusqu'à la réinitialisation. ~~La propriété Present_Value doit être inscriptible lorsque Out_Of_Service est TRUE.~~

[Modifier le paragraphe 12.16.5, p. 201]

12.16.5 Tracking_Value

Cette propriété ~~optionnelle~~, de type BACnetLifeSafetyState, reflète l'état non verrouillé de l'objet Life Safety Zone. La méthode permettant de dériver l'état doit être d'ordre local. Contrairement à Present_Value, qui peut verrouiller des valeurs à l'état non-NORMAL jusqu'à la réinitialisation, Tracking_Value doit continuer à suivre les changements d'état. *La propriété Tracking_Value doit être inscriptible lorsque Out_Of_Service est TRUE.*

[Modifier le paragraphe 12.16.11, p. 202]

12.16.11 Out_Of_Service

La propriété Out_Of_Service, de type BOOLEAN, indique si oui (TRUE) ou non (FALSE) l'entrée ou le processus que représente l'objet n'est pas en service. Cela signifie que les modifications apportées à la propriété Present_Value Tracking_Value sont découplées de l'entrée ou du processus lorsque la valeur de Out_Of_Service est TRUE. En outre, la propriété Reliability et l'état correspondant de l'indicateur FAULT de la propriété Status_Flags doivent être découplés lorsque Out_Of_Service est TRUE. Alors que la propriété Out_Of_Service est TRUE, les propriétés Present_Value Tracking_Value et Reliability

peuvent être modifiées en valeurs quelconques afin de simuler des conditions fixes spécifiques ou à des fins d'essai. D'autres fonctions dépendant de l'état des propriétés `Present_Value`, `Tracking_Value` ou `Reliability` doivent répondre aux modifications apportées à ces propriétés alors que `Out_Of_Service` est TRUE, comme si ces changements s'étaient produits au niveau de l'entrée ou du processus.

[Modifier l'Annexe C, p. 459]

```
LIFE-SAFETY-POINT ::= SEQUENCE {
...
present-value          [85] BACnetLifeSafetyState,
tracking-value         [164] BACnetLifeSafetyState OPTIONAL,
description            [28]  CharacterString OPTIONAL,
...
}
```

[Modifier l'Annexe C, p. 460]

```
LIFE-SAFETY-ZONE ::= SEQUENCE {
...
present-value          [85]  BACnetLifeSafetyState,
tracking-value         [164] BACnetLifeSafetyState OPTIONAL,
description            [28]  CharacterString OPTIONAL,
...
}
```

iTeh STANDARD PREVIEW
(standards.iteh.ai)

[ISO 16484-5:2007/Amd 1:2009](https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-135-2004c-1)

[https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-](https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-135-2004c-1)

135-2004c-1. Ajout de l'interface des services Web BACnet/WS

Argumentaire

Les « services Web » apparaissent comme la technologie prédominante pour l'intégration d'une grande variété d'informations d'entreprise. Le présent addendum définit un moyen normalisé permettant d'utiliser les services Web de manière à intégrer des données d'installation issues de sources diverses, y compris les réseaux BACnet, à une multitude d'applications d'entreprise.

Addendum 135-2004c-1

[Ajouter la nouvelle Annexe N]

ANNEXE N – INTERFACE DES SERVICES WEB BACnet/WS (NORMATIVE)

(La présente annexe fait partie de la présente Norme et est requise pour son utilisation.)

La présente annexe définit un modèle de données et une interface de service Web permettant d'intégrer des données d'installation de sources différentes à une variété d'applications de gestion d'entreprise. Le modèle de données et les services d'accès sont génériques et peuvent être utilisés pour modéliser et accéder aux données de source quelconque, que le serveur détienne les données en local ou qu'il agisse en tant que passerelle vers d'autres protocoles normalisés ou propriétaires.

Les implémentations des services décrits dans la présente norme doivent être conformes à Web Services Interoperability Organization (WS-I) *Basic Profile 1.0*, qui spécifie l'utilisation du protocole *SOAP 1.1 (Simple Object Access Protocol sur Hypertext Transfer Protocol -- HTTP/1.1 (RFC2616)* et code les

données de transport à l'aide de *Extensible Markup Language (XML) 1.0 (Second Edition)*, qui utilise les types de données et les représentations lexicales et canoniques définies par le World Wide Web Consortium XML Schema.

Les clients peuvent déterminer la version de la norme BACnet/WS implémentée par un serveur en demandant une valeur numérique spécifique définie au paragraphe N.9. La valeur numérique de la version décrite dans le présent document est 1.

La présente norme utilise trois noms de type de données différents. Les noms de type de données commençant par une lettre minuscule, tels que « string » et « nonNegativeInteger », désignent des types de données définis par la norme XML Schema. Les noms commençant par une lettre majuscule, tels que « Real » ou « Multistate » font référence aux types de valeur définis au paragraphe N.8.9. Les noms de type de données utilisés dans une « signature type reliant au langage » (typical language binding signature) sont arbitraires et n'ont qu'une vocation illustrative.

N.1 Modèle de données

Les structures et méthodes de données servant à stocker les informations dans un serveur BACnet/WS sont d'ordre local. Toutefois, pour échanger ces informations à l'aide des services Web, la présente norme établit un ensemble minimal d'exigences pour la structuration et l'association des données échangées avec un serveur BACnet/WS.

Le nœud est l'élément de données primitif fondamental du modèle de données BACnet/WS. Les nœuds sont organisés en hiérarchie dans le modèle de données. Le nœud situé en haut de la hiérarchie est le nœud racine. Un nœud racine a des enfants, mais pas de parent. Tous les autres nœuds ont un seul parent et peuvent, en option, avoir des enfants. L'état visible du réseau d'un nœud est représenté comme un ensemble d'attributs.

Tout nœud peut comporter une valeur. Les types possibles pour une valeur de nœud se limitent aux types de données primitifs « String », « OctetString », « Real », « Integer », « Multistate », « Boolean », « Date », « Time », « DateTime » et « Duration ». Les nœuds comportant une valeur peuvent également avoir d'autres attributs liés à cette valeur, tels que minimum, writable, etc.

Un attribut représente un seul aspect ou une seule qualité d'un nœud, telle que sa valeur ou son inscriptibilité. Chaque nœud expose un ensemble d'attributs. Certains d'entre eux sont obligatoires pour tous les nœuds, et d'autres sont requis sous condition en fonction de la valeur d'autres attributs. Certains attributs sont localisables et peuvent retourner différentes valeurs en fonction d'une option d'une requête de service. Les attributs sont décrits plus en détail à l'Article N.8.

Les attributs peuvent eux-mêmes avoir des attributs qui définissent un seul aspect ou une seule qualité de l'attribut original. La présente norme prend en charge cette récurrence syntaxique mais ne définit pas ni n'exige que les attributs normalisés aient eux-mêmes des attributs à ce moment-là. Les serveurs peuvent fournir des attributs propriétaires pour tout nœud ou attribut à n'importe quel niveau de la hiérarchie.

Un chemin est une chaîne de caractères permettant d'identifier un nœud ou un attribut d'un nœud. La hiérarchie des nœuds se retrouve dans un chemin sous la forme d'une hiérarchie d'identifiants organisés comme une série délimitée, semblable à l'organisation des identifiants d'une URL (Uniform Resource Locator) pour le World Wide Web. Un chemin tel que « /East Wing/AHU #5/Discharge Temp » identifie un nœud, et un chemin tel que « /East Wing/AHU #5/Discharge Temp:InAlarm » identifie l'attribut InAlarm de ce nœud. Les chemins sont décrits plus en détail à l'Article N.2.

Pour permettre un nombre arbitraire d'arrangements logiques de nœuds, un seul nœud peut logiquement apparaître à plusieurs endroits de la hiérarchie grâce à l'utilisation d'un nœud de référence. Les nœuds de référence peuvent être utilisés pour créer d'autres arrangements logiques de nœuds dans la mesure où les enfants d'un nœud de référence peuvent être différents de ceux de son nœud référent. Les nœuds de référence sont décrits plus en détail à l'Article N.4.

L'arrangement des nœuds de données en hiérarchies et la dénomination de ces nœuds sont généralement d'ordre local. Toutefois, la présente norme définit également un certain nombre de nœuds normalisés avec des noms et des emplacements normalisés permettant aux clients d'obtenir des informations de base sur le serveur lui-même. Les nœuds normalisés sont décrits plus en détail l'Article N.9.

N.2 Chemins

Un chemin est une chaîne de caractères permettant d'identifier un nœud ou un attribut spécifique. La hiérarchie des nœuds apparaît dans un chemin sous la forme d'une hiérarchie d'identifiants de nœuds disposés en une série délimitée séparée par des barres obliques (« / »). De même, la hiérarchie des attributs apparaît dans un chemin sous la forme d'une hiérarchie d'identifiants d'attributs disposés en une série délimitée séparée par des deux-points (« : »).

Certains services acceptent un chemin d'attribut optionnel à la fin d'un chemin de nœuds. Si un chemin d'attribut n'est pas spécifié pour ces services, l'attribut Value est supposé. Le chemin d'attribut est séparé du chemin de nœud par une virgule.

La forme concaténée du chemin est la suivante :

```
[/node-identifiant[/node-identifiant]...][:attribute-identifiant[:attribute-identifiant]...]
```

où les crochets indiquent le caractère optionnel et "..." indiquent la répétition de l'élément précédent.

Exemples : "/aaa" "/aaa/bbb" "/aaa/bbb/ccc:Description" "/aaa/bbb/ccc:Description:.foo"

iTeh STANDARD PREVIEW

Tous les identifiants sont sensibles à la casse et doivent être de longueur non zéro. Les identifiants ne sont pas localisables et ne sont pas concernés par les options de service « locale » ou « canonical ». Un chemin sans identifiant de nœud ("") désigne la racine de la hiérarchie, et « :attribute-identifiant » est la syntaxe permettant d'accéder aux attributs du nœud racine.

Seuls des caractères imprimables peuvent être utilisés pour construire des identifiants de chemin et, restriction supplémentaire, tous les caractères équivalents aux caractères de contrôle ANSI X3.4 (inférieurs à X'20') ne sont pas autorisés, ni les caractères équivalents aux caractères ANSI X3.4 suivants :

```
/ \ : ; | < > * ? " [ ] { }
```

Les identifiants de nœud commençant par un point (« . ») et les identifiants d'attribut ne commençant pas par un point (« . ») sont réservés par l'ASHRAE. Cette restriction sépare les identifiants de nœud et d'attribut qui sont définis par la présente norme de ceux définis par le serveur, sans doute en fonction de l'entrée de l'utilisateur. Les identifiants de nœud définis par le serveur ne doivent pas commencer par un point, si bien que « /aaa.first-floor » est incorrect tandis que « /aaa/first-floor » est correct. Inversement, tous les identifiants définis par un serveur doivent commencer par un point, si bien que « /aaa:MyNewAttribute » est incorrect alors que « /aaa.:MyNewAttribute » est correct. Cette asymétrie se base sur l'usage courant escompté où la plupart des identifiants de nœuds sont définis par le serveur et la plupart des attributs sont normalisés, l'utilisation des points faisant plutôt exception.

Les espaces sont autorisés et sont chargés de sens dans les identifiants ; il convient cependant que les identifiants ne commencent pas ni ne se terminent par des espaces.

N.3 Points normalisés

La plupart des protocoles d'automatisation du bâtiment, normalisés et propriétaires, organisent les données en « points » qui ont des « valeurs ». Outre leurs valeurs, les points contiennent souvent des données telles que la « description de point » ou « le point est en alarme ». Mais ces données peuvent être nommées, structurées et/ou accessibles de différentes manières dans divers protocoles.

Pour garantir qu'un client de service Web peut extraire des données sans connaître ces détails de dénomination et de méthode d'accès, la présente norme définit des « points normalisés ». Cela signifie

que les attributs courants des points disponibles dans la majorité des modèles de données du bâtiment sont exposés à l'aide d'un ensemble de noms commun.

Dans ce modèle de données, les nœuds avec un NodeType (voir N.8.5) « Point » doivent avoir une valeur et un ensemble commun d'attributs pouvant servir au mappage avec ces données issues d'autres protocoles. Certaines données peuvent ne pas être disponibles dans certains protocoles, auquel cas l'attribut normalisé est absent ou n'a pas de valeur par défaut raisonnable.

N.4 Nœuds de référence

Un nœud désignant un autre nœud de la hiérarchie est appelé « nœud de référence ». *Le nœud auquel il fait référence est son « nœud référent »*. Un nœud de référence fait apparaître la plupart des attributs de son « nœud référent », y compris son type, si bien que la plupart du temps, le nœud de référence est indissociable de son nœud référent. L'utilisation des nœuds de référence permet aux données d'un nœud d'apparaître à plusieurs emplacements d'une hiérarchie.

Plusieurs hiérarchies peuvent être prises en charge sur un serveur. La découverte automatisée de ces hiérarchies peut s'opérer en commençant à la racine, ou à tout autre point de départ, et en utilisant l'attribut Children pour énumérer les nœuds disponibles de manière structurée. La présence de deux ou plusieurs chemins dans différentes hiérarchies peut signifier des relations différentes pour un seul objet. Pour représenter cela et faire en sorte que les doubles apparents d'un objet puissent être distingués, un nœud peut désigner un autre nœud de la hiérarchie. La question de savoir quel nœud est le nœud référent et quel nœud est le nœud de référence est arbitraire et d'ordre local. Plusieurs nœuds de référence peuvent désigner le même nœud référent, ou se connecter en guirlande pour aboutir à un référent qu'ils ont tous en commun qui n'est pas un nœud de référence. Il doit y avoir au moins un nœud référent qui n'est pas un nœud de référence, dans la mesure où il est interdit de créer une boucle de références.

Une distinction visible du réseau entre un nœud de référence et son nœud référent se fait dans la présence d'un attribut Reference dans le nœud de référence. Cet attribut contient un chemin vers le nœud référent. L'attribut Reference est présent dans un nœud si et seulement si ce nœud est un nœud de référence.

ISO 16484-5:2007/Amd.1:2009
<https://standards.iteh.ai/catalog/standards/sist/c3067e44-713c-4e42-8d7d-7b3255dcdac9/iso-16484-5-2007-amd-1-2009>

Dans la plupart des cas, la distinction entre un nœud et un nœud de référence n'est pas nécessaire. Lorsque le client a besoin de faire la distinction, il peut vérifier la présence d'un attribut Reference et agir en conséquence. Un client peut également déterminer, pour un nœud donné, si des nœuds de référence le désignent. Cela peut se faire avec l'attribut Aliases.

À l'exception des attributs Children, Aliases, Attributes et Reference, tout attribut lu à partir du nœud de référence aura la même valeur que s'il est lu à partir du nœud référent. Cela s'explique par le fait que lorsque des références sont utilisées pour créer différentes relations entre des nœuds, ceux-ci ne sont pas fondamentalement modifiés par cette association. Par conséquent, seuls les attributs participant à l'expression des relations entre nœuds, à savoir Children, Aliases, Attributes et Reference, sont censés être différents selon le chemin emprunté pour accéder au nœud. Le nœud Attributes ne change que pour refléter la présence ou l'absence des attributs Children, Aliases ou Reference. Sinon, le contenu de l'attribut Attributes est inchangé.

Un nœud de référence peut désigner un autre nœud de référence, sans pour autant être autorisé à se désigner lui-même, ni à créer une boucle de références.

Par exemple, les chemins « /Geographic/East Wing/Air Handler 5/Discharge Temp » et « /Cooling/Chiller Manager/Air Handler 5/Terminal Box 345-A » expriment deux relations différentes pour Air Handler 5. Si la relation géographique a été modélisée en premier, pour la relation de la distribution de refroidissement, le nœud identifié par « /Cooling/Chiller Manager/Air Handler 5 » serait un nœud de référence avec l'attribut Reference contenant le chemin « /Geographic/East Wing/Air Handler 5 ».

N.5 Localisation

BACnet/WS prend en charge la création de produits spécifiquement conçus pour des régions particulières du monde. La désignation d'une langue naturelle, associée à un ensemble de pratiques de notation, tels que les formats de date et de nombre, est appelée « paramètre régional ». Un serveur BACnet/WS peut prendre en charge plusieurs paramètres régionaux en même temps, et plusieurs attributs d'un nœud sont accessibles pour différents paramètres régionaux (voir paragraphes N.11.4, N.11.5 et N.11.6). Par exemple, dans un serveur prenant en charge plusieurs paramètres régionaux, l'attribut DisplayName peut être utilisé pour obtenir le nom de présentation d'interface utilisateur du nœud dans plusieurs langues. La spécification d'un paramètre régional dans un service permet également au client de demander des dates, heures et nombres dans un format compatible avec ce paramètre.

N.6 Sécurité

BACnet/WS ne définit pas son propre mécanisme d'authentification ; la présente norme spécifie à la place l'utilisation de méthodes d'authentification de services Web de niveau inférieur définies par d'autres normes. Certains serveurs peuvent ne pas prendre en charge ni requérir l'authentification. D'autres peuvent assurer l'authentification au moyen d'un nom d'utilisateur et d'un mot de passe simples en utilisant l'authentification HTTP Basic (définie par la section 2 de HTTP Authentication : Basic and Digest Access Authentication) sécurisée via une connexion SSL (Secure Sockets Layer, défini par SSL Protocol Version 3.0) ou TLS (Transport Layer Security, défini par TLS Protocol Version 1.0). Certains serveurs peuvent être sécurisés via des certificats de clé publique ou des options plus avancées actuellement en cours de développement ou à définir.

Pour une spécification simple et une meilleure interopérabilité, les serveurs doivent revendiquer la prise en charge d'un ou des deux mécanismes d'authentification et d'autorisation suivants : « None » ; « HTTP Basic through SSL or TLS ».

Outre l'authentification, certaines formes d'autorisation peuvent également intervenir avant que les services Web définis par la présente norme soient invoqués. Par exemple, certains environnements d'hébergement de services Web (par exemple, les serveurs d'application) peuvent être configurés pour limiter l'accès des utilisateurs à certains services en fonction du chemin HTTP ou de la méthode SOAP.

Le contenu et le format des erreurs retournées de la part de ces méthodes d'authentification et d'autorisation de niveau inférieur varient et ne sont pas spécifiés par la présente norme dans la mesure où les services qui y sont définis n'ont jamais été invoqués.

Lorsqu'une demande de service Web passe avec succès à travers les niveaux inférieurs, et que les services définis par la présente norme sont invoqués, d'autres opérations d'authentification et d'autorisation peuvent être réalisées par ces services et le contenu et le format résultant de ces opérations sont définis de manière exhaustive par la présente norme. La configuration des règles d'authentification et d'autorisation, à tout niveau, est d'ordre local.

N.7 Sessions

Les services Web définis par la présente norme sont sans état et n'établissent pas de session entre les clients et les serveurs. Il n'est pas nécessaire que des informations soient conservées sur le serveur d'une invocation de service à l'autre. Les options de service telles que « locale » pouvant être conservées dans une session sur le serveur sont détenues par le client dans une chaîne d'options de service fournie au serveur pour chaque invocation.

N.8 Attributs

Un nœud est exposé aux services Web comme un ensemble d'attributs nommés. Il existe deux formes d'attributs : ceux dont le type de données est primitif, et ceux formant un tableau de types de données primitifs. Seul l'attribut Value est inscriptible avec les services définis par la présente norme.

Alors que certains attributs sont spécifiés comme optionnels, la présence de ces attributs sur un nœud donné n'est pas censée changer de manière dynamique. Les clients peuvent supposer que l'ensemble d'attributs disponibles restera relativement stable lors du fonctionnement et ne sera amené à être modifié qu'en cas de reconfiguration ou de reprogrammation du serveur et non au cours du fonctionnement normal. Par exemple, même si la valeur par défaut de l'attribut InAlarm est « false », cet attribut n'est pas censé être absent lorsque le nœud n'est pas en alarme et est censé être présent uniquement lorsque le nœud est en alarme. En général, si un attribut peut avoir une valeur qui est différente de sa valeur par défaut pendant le fonctionnement normal, il convient que l'attribut soit présent à tout moment.

Le serveur peut fournir des attributs propriétaires pour n'importe quel nœud ou attribut à n'importe quel emplacement dans la hiérarchie du modèle de données. Les attributs propriétaires doivent commencer par un point ('.') afin de pouvoir se distinguer des attributs standard. Le type de données et l'ensemble des valeurs possibles de ces attributs ne sont pas définis par la présente norme.

N.8.1 Attributs primitifs

Le type de données d'un attribut primitif de la présente norme est défini à l'aide de son nom de type de données XML Schema, tel que « boolean », « nonNegativeInteger » et « double ». Voir l'Article N.10 pour des détails sur la manière dont ces attributs sont codés pour une utilisation dans les services Web.

Le type de données de certains attributs, tels que Value et Minimum, dépend de la valeur de l'attribut ValueType. Le paragraphe N.8.9 fournit une description plus détaillée.

N.8.2 Attributs énumérés

Certains attributs primitifs sont des énumérations. Les attributs énumérés sont de type de données XML Schema « string », mais l'ensemble des valeurs autorisées est défini par la présente norme. En outre, certains attributs énumérés sont localisables (voir N.5). Dans ce cas, l'ensemble de valeurs non localisé est défini par la présente norme, mais les chaînes de caractères localisées sont d'ordre local.

N.8.3 Attributs de tableau

Il s'agit d'attributs qui contiennent un tableau de valeurs primitives. Chaque élément du tableau a le même type de données primitif. Le contenu d'un attribut de tableau est accessible en tant que tableau d'éléments séparés ou en tant que concaténation de tous les éléments.

Le type de données d'un élément de tableau de la présente norme est défini à l'aide de son nom de type de données XML Schema, tel que « boolean », « nonNegativeInteger » et « double ». Voir l'Article N.10 pour des détails sur la manière dont ces attributs sont codés pour une utilisation dans les services Web.

Lorsque les attributs de tableau sont accessibles avec un service retournant un tableau, tel que `getArray`, les éléments de tableau sont retournés sous forme de chaînes de caractères individuelles. Toutefois, lorsqu'ils sont accessibles avec un service retournant une seule chaîne, telle que `getValue`, les valeurs de tableau sont concaténées en une seule chaîne en séparant les éléments par un point-virgule « ; », par exemple, « high;medium;low ». Les valeurs des éléments de tableau individuels ne doivent pas contenir de point-virgule.

Le serveur doit conserver l'ordre des éléments d'un attribut de tableau. Les clients de services tels que `getArrayRange` peuvent, par conséquent, être dépendants de ce comportement pour lire le tableau, élément après élément.

N.8.4 Récapitulatif sur les attributs

Certains attributs sont toujours requis, et d'autres sous certaines conditions, en fonction des critères mis en évidence dans le tableau suivant. Le type de données auquel il est fait référence dans le tableau est un nom de type de données XML Schema. Voir l'Article N.10 pour des informations sur le codage des services Web. Les attributs non répertoriés comme Localisables ne sont jamais affectés par l'option de service « locale » (voir paragraphe N.11.4) et sont toujours codés sous leur forme canonique non localisée (voir paragraphe N.11.6).

Tableau N-1. Récapitulatif des attributs

Identifiant d'attribut	Type de données	Tableau	Énuméré	Localisable	Présence
"NodeType"	string	Non	Oui	Non	Obligatoire
"NodeSubtype"	string	Non	Non	Oui	Facultative
"DisplayName"	string	Non	Non	Oui	Facultative
"Description"	string	Non	Non	Oui	Facultative
"ValueType"	string	Non	Oui	Non	Obligatoire
"Value"	(varie – voir N.8.9)	Non	Non	Oui	Obligatoire si ValueType n'est pas « None »
"Units"	string	Non	Oui	Oui	Obligatoire si ValueType est « Real » ou « Integer »
"Writable"	boolean	Non	Non	Non	Obligatoire si ValueType n'est pas « None »
"InAlarm"	boolean	Non	Non	Non	Facultative
"Minimum"	(varie – voir N.8.9)	Non	Non	Oui	Facultative
"Maximum"	(varie – voir N.8.9)	Non	Non	Oui	Facultative
"Resolution"	(varie – voir N.8.9)	Non	Non	Oui	Facultative
"MinimumLength"	nonNegativeInteger	Non	Non	Non	Facultative et uniquement si ValueType est « String »
"MaximumLength"	nonNegativeInteger	Non	Non	Non	Facultative et uniquement si ValueType est « String »
"IsMultiLine"	boolean	Non	Non	Non	Facultative
"Attributes"	string	Oui	Non	Non	Obligatoire
"WritableValues"	string	Oui	Non	Oui	Obligatoire si ValueType est « Multistate » ou « Boolean » et si Writable est true
"PossibleValues"	string	Oui	Non	Oui	Obligatoire si ValueType est « Multistate » ou « Boolean »
"Overridden"	boolean	Non	Non	Non	Facultative
"ValueAge"	double (secondes)	Non	Non	Oui	Facultative
"Aliases"	string	Oui	Non	Non	Obligatoire si des nœuds de référence font référence à ce nœud (voir N.4)
"Children"	string	Oui	Non	Non	Facultative
"Reference"	string	Non	Non	Non	Présent si et seulement si le nœud est un nœud de référence (voir l'Article N.4)
"HasHistory"	boolean	Non	Non	Non	Obligatoire si ValueType n'est pas « None »
"SinglyWritableLocales"	string	Oui	Non	Non	Présent si et seulement si ValueType est « String » et Writable est true
"HasDynamicChildren"	boolean	Non	Non	Non	Facultative

N.8.5 NodeType

Cet attribut obligatoire indique la classification générale d'un nœud. Il sert d'indication à l'application client sur le contenu d'un nœud et il n'est pas prévu pour acheminer une définition exacte. La liste des valeurs de cet attribut n'est pas extensible. Une classification plus précise est fournie par l'attribut NodeSubtype. Les valeurs admises pour cet attribut sont les suivantes :

```
{ "Unknown", "System", "Network", "Device", "Functional", "Organizational", "Area",
  "Equipment", "Point", "Collection", "Property", "Other" }
```

Le type « Unknown » peut être utilisé pour les données originaires d'une autre source et pour lesquelles aucun type d'information n'est connu. Le type « System » peut être utilisé pour désigner un système mécanique dans son intégralité. Le type « Network » peut servir à représenter un réseau de communications, et le type « Device » pourrait être utilisé pour représenter un dispositif physique sur ce réseau. Le type « Functional » peut être utilisé pour représenter un seul composant système tel qu'un module de commande ou un composant logique tel qu'un bloc fonctionnel. Le type « Organizational » a pour but de représenter les concepts d'entreprise tels que les départements ou le personnel. Le type « Area » représente un concept géographique tel qu'un campus, un bâtiment, un étage, etc. Un « Point » représente un seul point de données, une entrée ou une sortie physique d'un dispositif de commande ou de surveillance, ou un paramètre de calcul logiciel ou de configuration. Un type « Equipment » peut servir à représenter un seul élément d'un équipement pouvant être un ensemble de « Points ». Un type « Collection » est un simple conteneur générique servant à regrouper des éléments tels qu'un ensemble de références à toutes les valeurs de température ambiante relevées dans un bâtiment. Le type « Property » a pour but de modéliser les données faisant logiquement partie d'un nœud parent. Le type « Other » sert à tout ce qui ne rentre pas dans les catégories précédentes.

N.8.6 NodeSubtype

Cet attribut optionnel est une chaîne de caractères imprimables dont le contenu n'est pas limité. Il fournit une classification plus spécifique du nœud. Par exemple, lorsque l'attribut NodeType a une valeur « Area », le nœud NodeSubtype pourrait avoir une valeur telle que « Campus », « Building » ou « Floor ». Cet attribut peut être localisé, éventuellement en retournant différentes valeurs appropriées aux paramètres régionaux lorsqu'une option de service « locale » est spécifiée.

N.8.7 DisplayName

Cet attribut obligatoire est une chaîne de caractères imprimables dont le contenu n'est pas limité. Il sert à fournir un nom ou un titre descriptif court (10-30 caractères) qui s'affiche dans les interfaces utilisateur pour les humains. Il convient qu'il soit localisé, si la localisation est prise en charge, éventuellement en retournant différentes valeurs appropriées aux paramètres régionaux lorsqu'une option de service « locale » est spécifiée. Un client peut extraire cet attribut dans n'importe quel paramètre régional pris en charge par le serveur pour la création d'affichages multilingues. Il n'est pas nécessaire que les valeurs des attributs DisplayName soient uniques parmi les nœuds apparentés.

Un attribut DisplayName peut être différent de l'identifiant de chemin utilisé pour accéder au nœud. Par exemple, pour le nœud identifié par le chemin « /Building 12/Room 225 », le DisplayName pourrait être « Bob's Office » dans un paramètre régional et « Bureau de Bob » dans un autre, ou juste « Room 225 » dans tous les paramètres régionaux.

N.8.8 Description

Cet attribut optionnel est une chaîne de caractères imprimables dont le contenu n'est pas limité. Cet attribut peut être localisé, éventuellement en retournant différentes valeurs appropriées aux paramètres régionaux lorsqu'une option de service « locale » est spécifiée.

N.8.9 ValueType

Cet attribut obligatoire indique le type de données de l'attribut Value et des attributs limitant l'attribut Value. Si le nœud n'a pas de valeur, cet attribut doit toujours avoir la valeur « None ». La liste des valeurs de cet attribut n'est pas extensible. Les valeurs admises pour cet attribut sont les suivantes :

["None", "String", "OctetString", "Real", "Integer", "Multistate", "Boolean", "Date", "Time", "DateTime",
"Duration"]

Le type « None » est utilisé lorsque le nœud n'a pas de valeur. Le type « String » est utilisé pour les nœuds dont les valeurs de chaîne de caractères ont pour but d'être lisibles par les humains. Un type « OctetString » est utilisé pour contenir des données binaires arbitraires qui ne sont généralement pas lisibles par les humains. Un type « Real » est une valeur à virgule flottante, par exemple 75,6. Un type « Integer » concerne les valeurs exprimées sous forme de chiffres, par exemple 1234. Un type « Multistate » est une valeur représentant un choix parmi un ensemble d'états nommés, par exemple,

{"high", "medium", "low"}. Un type « Boolean » est un choix entre deux états nommés exactement, tels que « on » et « off », l'un étant considéré comme true et l'autre comme false. Un type « Date » sert à représenter les valeurs correspondant à des dates calendaires. Un type « Time » est utilisé pour représenter une heure du jour. Un type « DateTime » est utilisé pour représenter un moment exact dans le temps, avec spécification d'une date et d'une heure. Un type « Duration » représente une durée, telle que « 5 secondes ».

La représentation de tous les types de valeur autres que « None » et « OctetString » peut être affectée par l'option de service « locale » si le serveur prend en charge la localisation d'un ou de plusieurs paramètres régionaux. Voir l'Article N.5 et le paragraphe N.11.4.

L'effet de cet attribut sur le type de données Value et des attributs apparentés est récapitulé dans le tableau suivant. Les types de données référencés dans le tableau sont des noms de type de données XML Schema. Voir l'Article N.10 pour des informations sur le codage des services Web. Les attributs dont le type de données est répertorié sous la forme n/a dans le tableau ne doivent pas être présents dans le nœud.

Tableau N-2. Effet de l'attribut ValueType

Valeur de l'attribut ValueType	Type de données de l'attribut Value	Type de données de l'attribut Minimum	Type de données de l'attribut Maximum	Type de données de l'attribut Resolution
"None"	n/a	n/a	n/a	n/a
"String"	string	n/a	n/a	n/a
"OctetString"	base64Binary	n/a	n/a	n/a
"Real"	double	double	double	double
"Integer"	integer	integer	integer	integer
"Multistate"	string	n/a	n/a	n/a
"Boolean"	boolean	n/a	n/a	n/a
"Date"	date	date	date	integer (jours)
"Time"	time	time	time	double (secondes)
"DateTime"	dateTime	dateTime	dateTime	double (secondes)
"Duration"	double (secondes)	double (secondes)	double (secondes)	double (secondes)

N.8.10 Value

Cet attribut optionnel représente la valeur du nœud. Le type de données de cet attribut est indiqué par l'attribut ValueType. L'attribut Value est présent si et seulement si la valeur de l'attribut ValueType n'est pas « None ». Lorsque l'attribut ValueType du nœud est « String » ou « Multistate », les valeurs de cet attribut peut être localisé en fonction de l'option de service « locale ». Voir N.11.4.

N.8.11 Units

Cet attribut optionnel définit les unités d'ingénierie de l'attribut Value du nœud. Si l'attribut ValueType est « Real » ou « Integer », cet attribut doit être présent mais peut avoir la valeur « no-units ». Cet attribut peut en option être présent pour d'autres valeurs de l'attribut ValueType.

La valeur de cet attribut est disponible sous deux formes. Si l'option de service « canonical » est false, la valeur de cet attribut est une chaîne dont le contenu n'est pas limité et peut être approprié au paramètre régional demandé. Si l'option de service « canonical » est true, la valeur de cet attribut se limite à être exactement égale à l'un des identifiants d'énumération, tels que « degrees-Celsius », « inches-of-water », etc., qui sont définis par la production ASN.1 pour BACnetEngineeringUnits à l'Article 21.

Cet attribut est extensible pour prendre en charge d'autres unités que celles définies par la présente norme. Dans le cas où l'unité de la valeur du nœud ne correspond pas à l'une des unités définies dans la présente norme, la valeur retournée pour cet attribut lorsque l'option de service « canonical » doit être « other », et la valeur retournée lorsque l'option de service « canonical » est false doit être une chaîne dont le contenu n'est pas limité et peut être appropriée au paramètre régional demandé.