

SEMI E120-0310

共通装置モデル(CEM)の仕様

NOTICE: This translation is a REFERENCE COPY ONLY. If differences should exist between the English version and a translation in any other language, the English version is the official and authoritative version.

免責事項: この SEMI スタンドアードは、投票により作成された英語版が正式なものであり、日本語版は日本の利用者各位の便宜のために作成したものです。万が一英語と日本語とに差異がある場合には英語版記載内容が優先されます。

SEMI スタンドアード日本語翻訳版をご利用にあたっての注釈を本文の末尾に記載しております(「すべきである」「しなければならない」について等)。

本スタンドアードは、global Information & Control Committee で技術的に承認されている。現版は 2009 年 12 月 16 日、global Audits and Reviews Subcommittee にて発行が承認された。2010 年 2 月に www.semi.org で入手可能となる。初版は 2003 年 3 月発行、前版は 2008 年 11 月に発行された。

1 目的

1.1 本仕様の目的は、半導体製造装置の構成および関連要素の汎用のオブジェクトモデルを提供することである。このオブジェクトモデルは、次の方法で使用されることを意図したものである。

- 装置の外部ビューを工場ホストにどのように見せるかを示す装置サプライヤへのガイドとして
- 他の SEMI 装置通信スタンドアードが使用し、拡張する際の基本的なモデルとして
- XML スキーマ等、技術に特徴的なオブジェクトモデルを明確に作成する際の参照先として

2 範囲

2.1 本仕様は、装置メーカーが装置構成および関連要素をモデル化する際に使用される構造に共通なものの一揃えを提供する。

2.2 本仕様は、利用可能な通信インタフェースを介して工場ホストアプリケーションが見る際の装置構造のモデルに適したクラス、属性、関係、その他の詳細事項を定義する。

2.3 本仕様に含まれるモデルは装置の構造の特性を表すが、(現在および将来に渡って)複数の装置通信スタンドアードに必要なものと考えられる。

注意: このスタンドアードはその使用に関連したすべての安全問題を取り扱うことを意図していない。このスタンドアードの使用者は、その責任において、適切な安全および健康上の作業方法を確認し、また使用前に規制上その他の制限への適用性を判断するものである。

3 制限

3.1 本仕様では、装置の「オブジェクト図」(ここでは¹インスタンスを指す)は定義しない。モデルに準拠した定義を作成することは装置サプライヤの責任である。

3.2 本仕様では、オブジェクト/クラスの挙動は定義しない。本仕様のクラスの挙動の定義は、他の装置通信スタンドアードに委ねられる。

3.3 本仕様では、装置と工場ホスト間のメッセージ交換については特に定義しない。共通装置モデルに関連する通信は、他のスタンドアードによって定義されるものと期待される。

3.4 本仕様では、ホストから見た装置にのみ言及する。本文書では、工場の他の部分についてはモデル化していない。

¹ 訳者補足: 一般にオブジェクト図というクラス図も含まれるが、ここでは UML の定義に従ってインスタンス図のことをさす。

4 参照スタンダードおよび文書

4.1 ISO スタンダード²

ISO/IEC 11578:1996 — Information Technology – Open Systems Interconnection – Remote Procedure Call (RPC)

注意：別途指示のない場合、引用されたすべての文書は最新の出版バージョンである。

5 用語

5.1 略語および頭字語

5.1.1 CEM (Common Equipment Model) — 共通装置モデル

5.2 定義

5.2.1 本文書内で使用される UML 用語は、OMG の *UML Specification* の用語集に準拠する。関連情報の R1 項に、その用語集の定義の抜粋を記載する。

5.2.2 本仕様で使用される用語の定義と説明の多くは、SEMI のウェブサイト <http://www.semi.org/> で入手可能な *SEMI Compilation of Terms* に記載されている。

5.2.3 上記の 2 つの情報源の用語に矛盾がある場合は、本仕様では OMG の *UML Specification* の定義が優先する。

6 表記法

6.1 この項では、本文書で守る表記法を定義する。

6.2 オブジェクトモデリング

6.2.1 統一モデリング言語(UML) — 本仕様では、すべてのクラス図および例として提供するオブジェクト図に UML 表記を使用する。本仕様では、他の種類の図は使用されない。

6.2.1.1 UML クラス図には、明確に定義された意味があり、本仕様の一部である。これらのダイアグラムに含まれている詳細事項については、必ずしも文章で繰り返されるとは限らない。

6.2.2 挙動は未定義 (Behavior Not Defined) — 本仕様では、クラス、その関係、属性を定義する。UML の状態図、アクティビティ図、シーケンス図としても、動作の定義としても挙動を指定することはない。

6.2.3 クラス名 (Name of a Class) — クラス名は大文字で始まるものとする。

6.2.4 抽象クラスおよび具象クラス (Abstract and Concrete Classes) — 各クラスには抽象、具象の区別がある。抽象クラスは、直接実装されることはない。本仕様では、「実装」とは通信インタフェースを経由して工場に姿を示すことを意味する。具象として定義されるすべてのクラスは、直接実装できる。本仕様は、装置の制御システムの実装を定義しようとするものではない。UML クラスダイアグラムでは、抽象クラス名はイタリックで表示される。

6.2.5 クラスの属性の定義 (Class Attribute Definition) — クラスの属性は、次の表 1 に示すように表の形態で定義される。UML クラスダイアグラムの属性の前の「+」の符号は、「public (パブリック)」の属性であることを示す。本仕様で定義されるすべての属性は「public」である。

表 1 属性表のフォーマット

属性名	定義	アクセス	必須	形態
		RO または RW	Y/N/C	下記のリストを参照

6.2.5.1 アクセス — 属性は、装置へのインタフェース経由で設定可能 (ReadWrite すなわち RW) あるいは設定不能 (ReadOnly すなわち RO) のどちらかである。

² International Organization for Standardization, ISO Central Secretariat, 1 rue de Varembé, Case postale 56, CH-1211 Geneva 20, Switzerland. Telephone: 41.22.749.01.11; Fax: 41.22.733.34.30; <http://www.iso.ch>

6.2.5.2 必須 — この属性が必須か否か。Y – 必須, または N – 任意, C – 条件付き。条件付の場合, 属性が要求される条件が定義される。

6.2.5.3 形態 — 属性のデータ型を定義する。共通装置モデル仕様で指定されるデータ型は, 次の各項で説明する。本仕様のデータ型は, 高レベルの定義であり, 必要に応じて個々の技術に沿ってマッピングしなければならない。共通装置モデルに対する拡張は, 本仕様で定義されるデータ型のリストには制限されない。ここで紹介されるもの以外, SEMI の文書でよく使用される形式の実態については, *SEMI Compilation of Terms* の *Form* の定義を参照のこと。

6.2.5.3.1 列挙 — ある属性の型の取り得る範囲として使用される名前で取り扱われる値のリスト。本文書では, これらの値は, 文字列として表現されるが, 異なる実装もあり得る(名前の表す値に対応した整数値等)。

6.2.5.3.2 文字列 — 表記文字による文字列。長さの制限については, 本文書内で特に指定がない限り, 実装技術に委ねられる。

6.2.5.3.3 xxx のリスト — 指定された型の複数の実際値を保持できる情報項目(ここで, xxx はデータ型)。

6.2.6 制約

6.2.6.1 UML の制約は点線で示され, 制約を説明する表現が添えられるのが一般的である。この方法での制約の表記は, UML 図を乱雑に大きくする傾向がある。従って, 本仕様では, 制約は個別のクラスに焦点を絞ったダイアグラムだけに示す。

6.2.7 関連の誘導可能性 (Association Navigability)³

6.2.7.1 UML の関連には「誘導可能性 (Navigability)」という概念がある。関連が参照可能な場合, 関連は関連の反対側にあるクラスのインスタンス(関連先のオブジェクト)に到達する。

6.2.7.2 指定のない場合, 関連は両方向に誘導可能である。矢印が関連のどちらか一方方向にのみ記されている場合, 誘導可能性はその方向にのみ存在する。

6.2.7.3 誘導可能とされる関連の到達先のオブジェクトの参照はすべて可能でなければならない。

6.2.7.4 目的のオブジェクトを参照する方法は, 実装に依存するものであり, 本文書では規定しない。

6.2.7.5 本仕様の概念を実装方法にマッピングするすべての実装者や従属スタンダードは, ソースオブジェクトからの関連付けの対象オブジェクトを参照する手段を指定すること。

6.3 モデルの拡張

6.3.1 CEM のすべての具象クラスは拡張できる。拡張は, 関連付けによっておこなう。「関連付けによる拡張」において, 新しい機能は関連付けによってオリジナルに関連する別のクラス内に格納される。CEM の場合, 「*Extension*」クラスは, 必要な関連付けと共にモデルに追加された(¶ 8.5.3.4 を参照)。

6.3.2 *Extension* クラスは, *Nameable* が複数の *Extension* を持つことができるように, *Nameable* に関連付けられる。*Extension* クラスに, 属性やサービスがなく, それ自体は拡張 CEM クラスに何も追加しない。意味のある拡張をおこなうには, 新しい機能を含む *Extension* のサブクラスを作成しなければならない。

6.3.3 従って, 選択された具象 CEM クラスに対する拡張を作成するためには, 追加クラスによる新しいクラスがデザインされる。例えば, 新しい属性を格納することもある。この新しいクラスは, *Extension* のサブクラスとしてデザインされる。任意の *Nameable* は, その任意の *Extension* に対する関連付けを誘導可能であることを忘れないこと。従って, *Extension* の新しいサブクラスは, 任意の CEM クラスに関連付けることができる。その図示については, 次の図 1 を参照すること。最終ステップは, その新しい *Extension* サブクラスを, 拡張する特定の CEM クラスに関連付けることである。

6.3.4 ほとんどのサブクラスは, 特定の CEM クラスについて動作するようにデザインされる。CEM モデルでは, 拡張が意図した CEM クラスとマッチングすることを強制しない。従って, *Extension* クラスの仕様には, 許される使用方法を記述しなければならない。

³ 訳者補足: Navigability の訳語「誘導可能性」はオブジェクト指向関連の業界で現在, 必ずしも定着した訳語ではない。

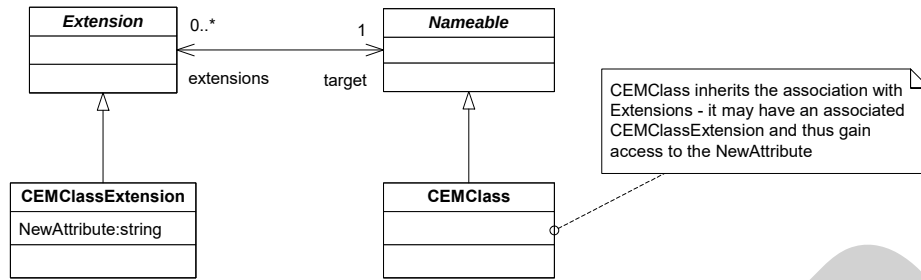


図 1
CEM モデルの拡張

6.3.5 CEM の範囲は、装置に関する静的な情報に限定される。しかし、その制限は、拡張には適用されない。従って、拡張には静的あるいは動的な属性および状態モデルやその他の複雑な構造へのリンクを含むことができる。サプライヤによって追加される可能性がある拡張の例は、*Equipment* オブジェクトに適用される「manufacturedDate」属性および「serviceTagNumber」属性を含むクラスである。

7 背景

7.1 本仕様は装置に共通した特質を単一の仕様書として文書化するもので、他のスタンダードが参照、使用できることを意図するものである。ここで述べた他のスタンダードで、共通装置モデルを拡張し、そのスタンダードに固有の特質を付加するものと考えられる。このように、複数のスタンダードの集合体としての完全モデルに伴い、唯一の整合の取れた装置のオブジェクトモデルの核心部分を本仕様で構成するよう、作成したものである。

7.2 モデル作成の道筋 (Approach to Model Description)

7.2.1 クラスモデルとインスタンスモデル (Class Model vs. Instance Model)

7.2.1.1 ある特定の装置を明細に示すモデルは、「インスタンスモデル」と見なされる。個々の装置のインスタンスモデルには、その機能に基づく独自の特質がある。唯一のモデルでは、そのようなインスタンスのすべてを直接記述することは不可能である。その代わるものとして、本仕様で使うクラスモデルは、装置構造および、複数のクラス間でどういう関係を持たなければならないかを記述するために使用されるクラスを定義する。装置のインスタンスモデルは、CEM クラスモデルを使って、装置サプライヤが提供しなければならない。⁴

7.2.2 モデルの拡張性 (Model Extensibility)

7.2.2.1 共通装置モデルは、(決められた制約のもとで)拡張が可能なモデルを意図している。¶ 6.3 は、モデルの拡張に使っても良い方法について説明する。

7.2.2.2 従って、装置のインスタンスモデルは、次の項目に基づいて装置サプライヤが構築しなければならない。

- 本文書で定義されるクラス
- 本モデルに対する拡張で、他の SEMI スタンダードで定義されるものに関するもの
- 標準化モデルへの拡張で、装置の固有な特質に対応するために装置サプライヤが行ったもの

7.2.3 外からの見え方 — 装置設計にはあらず (External View – Not Equipment Design)

7.2.3.1 本文書が定義するモデルを装置サプライヤが装置の設計のベースとして使用するという風には期待するものではない。CEM は、透明性の高い意思疎通を促すために構造を提供するように意図したものである。曖昧性を排除して装置を表し工場ホストの求めに呼応した見え方を提供する必要がある。

⁴ 訳者補足:ここでいわれているインスタンスモデルは本来、具象モデル (Concrete Model) のことを意味するものと思われる。

7.2.4 通信は定義せず (Communication Not Defined)

7.2.4.1 本仕様は、共通装置モデル (の全体または一部) を装置と工場ホスト間でやり取りする仕方の定義については、他のスタンダードに委ねる。モデルには多くの使用方法があり、意図した使い方の脈絡に合わせて、通信の詳細が最良な方法で定義されることが望まれる。

7.2.5 挙動は定義せず (Behavior Not Define)

7.2.5.1 モデルの単純化のため本文書では CEM でのクラスの挙動は定義しない。クラスの挙動は機能と関連するので、装置インタフェースによって提供される機能に関連する。この機能と挙動については、他の SEMI スタンダードの定義に委ねる。

7.2.6 適用性 (Applicability)

7.2.6.1 CEM は、工場内の半導体に関連するすべての装置に対応使用と考えられているものの、その通りにはならない可能性が指摘されている。CEM は、製造装置 (すなわち、材料の表面に作用を及ぼす装置で単独に存在する環境監視装置等ではない) のみをカバーする。半導体工場の装置の大多数が、本仕様に準拠できるはずである。

7.3 CEM の使用方法 (Uses for the CEM)

7.3.1 CEM を理解するには、モデルの期待される使用方法について理解する必要がある。可能なすべての使用方法を予期することはできないが、他のスタンダードが盛り込みそうな使用方法を以下に示す。

7.3.2 **メタデータを使用した装置の自己説明 (Equipment Self-Description Using Metadata)** — 半導体製造装置を工場に統合するための重要な考慮事項は、装置の構造と能力を簡単に理解でき、その内容を機能として動作するソフトウェアに対して簡単に伝達できることである。従来の半導体通信技術では、まず装置サプライヤの通信マニュアルを読み、それから工場ソフトウェアを構成、コーディングしその文書に合致させて始めて通信が成立する。残念ながら、マニュアルに不具合があったり、古くなっていることがあまりに多い。装置がそのインスタンスモデル (およびインスタンスモデルに基づく拡張クラスモデル) をホストに提供できれば、より完全かつ正確になると思われる。現在のハードウェア構成に基づいてリアルタイムでモデルを作成できれば、さらに正確である。この自己説明データは、「メタデータ」としても知られている。

7.3.3 **装置に対するデータの関連付け (Relation of Data to the Equipment)** — 装置から収集したデータは、装置のどの部分に関連しているかが明確であり、関連データのどの項目がコンテキスト情報を提供するかが明確であれば、より大きな意味を持つ。インスタンスモデルは、データコレクションプランを定義する上での参考ともでき、実際に収集されるデータが反映されることになる。

7.3.4 **パラメータの関連付けおよび装置への設定 (Relation of Parameters & Setting to the Equipment)** — 使えるパラメータおよび設定すべての効果を理解した場合のみ、装置を正しくかつ思った通りに制御できる。これらは装置および影響が現れる測定可能な値と関連付けられるはずである。この情報は、インスタンスモデルによって提供できる。

7.3.5 **装置の能力の理解 (Understanding Equipment Capabilities)** — 装置のプロセス能力は、搭載したモジュールおよびそれらのモジュールの特性と状態に依存する。状況によっては、装置のハードウェア構成が日常、変わることもあり得る。

8 共通装置モデルの定義

8.1 この項では、共通装置モデルについて記述する。

8.2 概要

8.2.1 共通装置モデルの作成は、モデルをシステムの視点で眺めることから始まり、属性と関係を考えながらクラスの詳細な記述へと移る。挙動については記述しない。すなわち、クラスについて状態モデルやメソッドは記述しないということである。

8.2.2 作成する 2 つのシステム視点は同じモデルのものであることに注意することが重要である。大きくて複雑な図にならないように分けてあるので、理解しやすくなる。

8.2.3 モデルは、クラスとその関係のみを示す。実際の装置には、装置の視点でのインタフェースから見た装置の詳細を定義する装置毎に対応インスタンスモデルがある。従って、本文書の共通装置モデルでは、装置のインスタンスモデルを構築する規則を提供する。

8.2.4 共通装置モデルは、2つのシステムレベルの視点で表現される。1つ目は、どのクラスがどのクラスから特質(属性や関係等)を継承するのかを示す継承階層構造である。2つ目の視点は装置の全般的な構造を示すクラス間の関係を表現するものである。これらの視点は、モデルの重要な側面を強調するために、一般的に使われる用語で記述する。

8.2.5 これらの2つの視点に登場するもののすべては、後の項で繰り返し登場するが、そこではモデル内の個々のクラスを取り扱う。個々のクラスの視点は、対象となるクラスを他のクラスに対する関係によって示す。この視点では、その他の詳細事項は示されない。

8.3 継承階層(Inheritance Hierarchy)

8.3.1 図2の継承階層の視点では、共通装置モデルのスーパークラス/サブクラスの関係を示す。

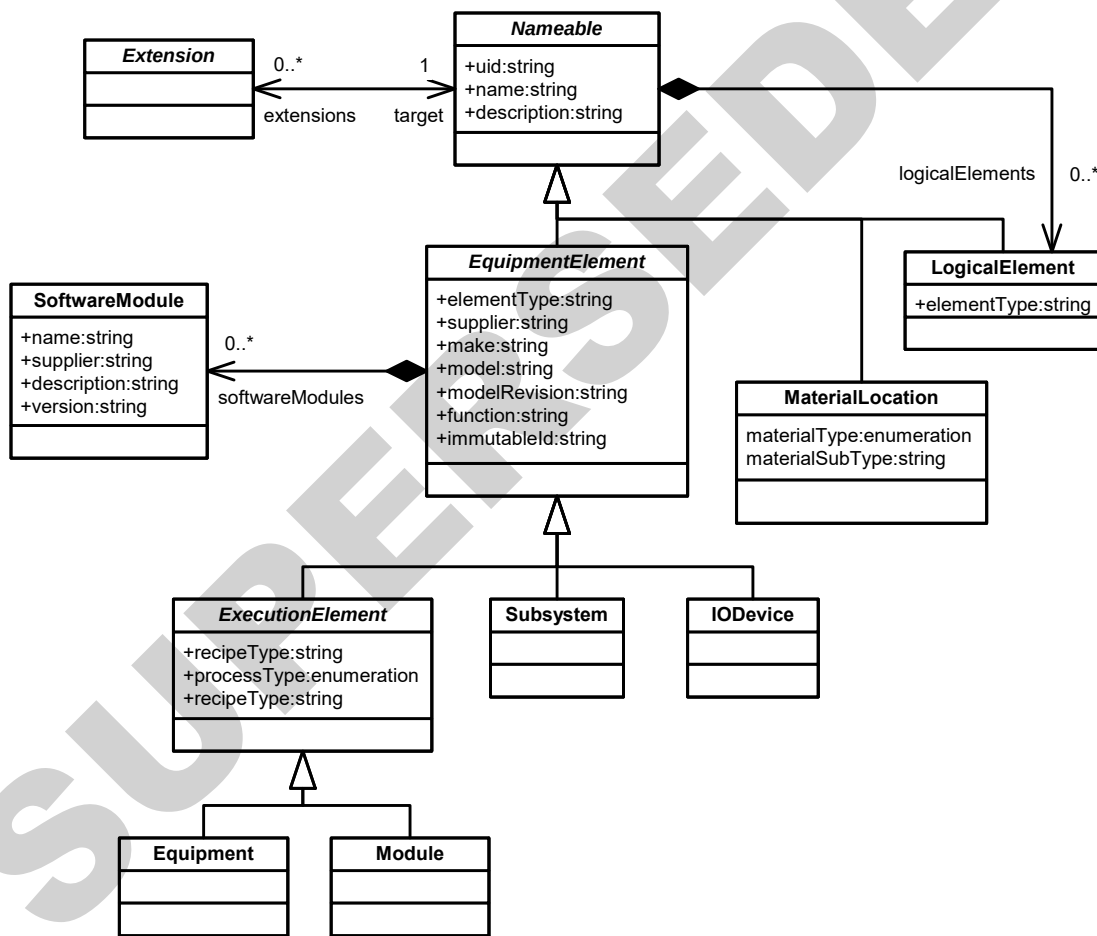


図2
継承階層

8.3.2 継承階層視点についての特記事項:

- *Nameable* は、継承階層の最上位にあり、共通装置モデル内のすべてのクラスに対する共通のアクセス方法を提供する。

- *Equipment* および *Module* は材料を処理することができるが、*Subsystem* はできない。
- サブクラスはそのスーパークラスの属性を継承することに注意すること。本仕様では、サブクラスによっては、追加属性が定義されていないものがある。他のスタンダードにより、このモデルは拡張され、このモデル上に装置挙動がマッピングされる際に、属性、関係、状態モデル等が追加されるものと期待される。

8.4 関係視点 (Relationship View)

8.4.1 関係についての視点を図 3 に示す。この視点は望まれる装置構造を示す。関係視点の特記事項は次の通りである。

- 例えば、*IODevice* が装置構造のほとんどのレベルにも直接属することができるようにするために、構造内に柔軟性を持たせている。
- *Subsystem*, *Module*, *Equipment* は、*MaterialLocation* (すなわち、材料を保持する) を持つことができる。
- 装置—装置の集約により他の装置の代わりに働く装置がそれら装置を示すことができる。

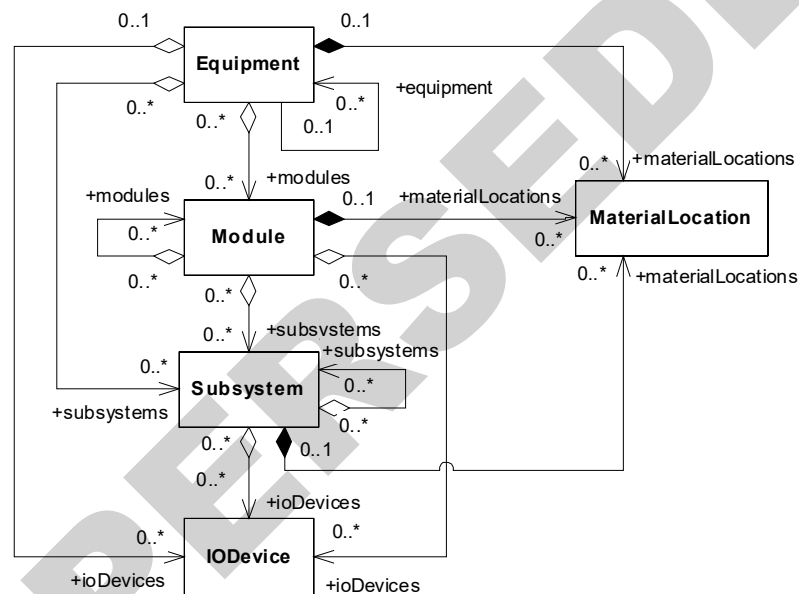


図 3
関係図

8.5 クラスの定義 (Class Definitions)

8.5.1 クラスの定義は、対象とするクラスの簡単な説明、そのクラスとすべての関連クラスを示すダイアグラム (継承と関係の両方の立場から)、属性を定義する表、クラスを完全に定義するために必要なその他の説明文で構成する。

8.5.2 区別するために、抽象クラスは、具象クラスとは別に取り扱う。

8.5.3 抽象クラス (Abstract Classes)

8.5.3.1 *Nameable* — 装置階層における、各構成要素のルートクラスである。このクラスは、装置構造の構成要素を一意的に識別できるようにするものである。*Nameable* は、*EquipmentElement*, *LogicalElement* および *MaterialLocation* の親 (基底) クラス (スーパークラス) である (図 4 を参照)。3 つの属性、*uid*, *name*, *description* を持つ (表 2 を参照)。Nameable はクラス *Extension* と結びつく。Nameable は、*LogicalElements* の複合的な集約単位を有することもある。これにより、*Equipment*, *Module*, *Subsystem*, *IODevice*, *MaterialLocation* などの実在す

るサブクラスである *Nameable* のいかなるインスタンスも *LogicalElements* を格納することが可能となる。*Extension* の使用については¶ 6.3 を参照。

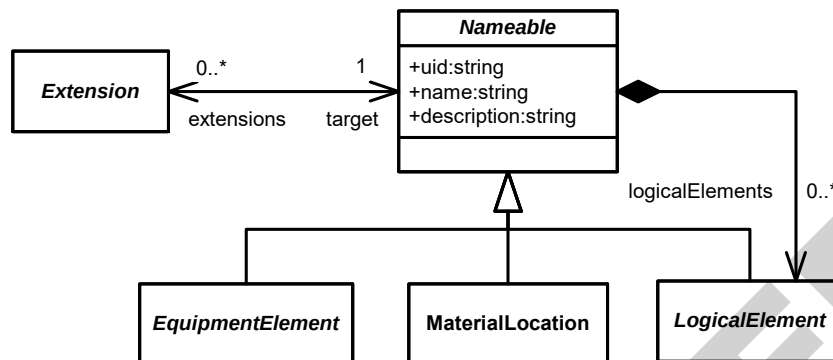


図 4
Nameable クラス

表 2 Nameable の属性

属性名	説明	アクセス	必須	フォーム
<i>uid</i>	一意かつ不変的に <i>Nameable</i> を指定する指定子。 <i>uid</i> が「 <i>uuid</i> 」フォーマットの値である場合、全世界的にユニークでなければならない。 <i>uuid</i> フォーマットを使用するときは、ISO/IEC 11578 に定義されるフォーマットと規則に適合しなければならない。 <i>uuid</i> の定義については ISO/IEC 11578:1996 の § 4 を参照のこと。 <i>uid</i> が <i>uuid</i> としてフォーマットされない場合、装置の範囲内だけでユニークであることが要求される。	RO	Y	String
<i>name</i>	人間が読んで理解ができる <i>Nameable</i> の名前。 <i>Name</i> は <i>Nameable</i> が装置構造において行う役割もしくは機能を表す。	RO	Y	String – 次の名前付け規則を参照
<i>description</i>	人間が読んで理解ができる <i>Nameable</i> の説明。この属性の内容は、人間がこの <i>Nameable</i> の目的を理解するのに十分な情報を提供しなければならない。	RO	Y	String

8.5.3.1.1 *uid* — *uuid* フォーマットの値が *uid* に使用されるとき、*uid* の値は全世界的にユニークということになる。この *Nameable* に関連するデータのトラッキングを促進するために、*uid* の値はこのオブジェクトに対して時間の経過とともに不変であることが望ましい。このオブジェクトが表わす物理アイテムが交換されたときに、この値は変わらなければならない。*uid* が *uuid* フォーマットの値ではない場合、この属性は装置の CEM モデルの範囲でユニークであり、ファクトリー全体でユニークではない。さらに、このオブジェクトが表わす物理アイテムが交換されたときに、この値は変わってはならない。

8.5.3.1.2 *name* — 各 *Nameable* の名前付けの要求条件は次の通りである。

- *Names* は、英数字、スペース、ハイフン、アンダースコアで構成しなければならない。名前は英字で始まり、冒頭や末尾にスペース文字があってはならない。
- *name* 属性の値は、同じ集約(オブジェクト)の構成要素部品間では一意性を持たなければならない⁵。この制限は、複数のオブジェクトによって集約(共有)されている *Nameable* についても適用される。
- *Nameable* が、ある SEMI スタandard に記述されたクラスや変数を表現する場合、使用される *name* はそのStandard で指定されたスペルや大文字小文字のスタイルに一致しなければならない。一意性の規則に従うこ

⁵ 集約のメンバ(すなわちダイヤモンドの反対側にあるクラス/オブジェクト)は「集約の構成部品」または「構成要素」と呼ばれる。集約関係のダイヤモンド側のクラス/オブジェクトが「集約する」ということである。