

## 5 관리 셀의 메타모델

## 5.1 소개

이 절은 자산 관리 셸(AAS)의 정보 메타 모델을 지정합니다. 그렇게 하기 전에, 자산 유형 및 인스턴스 처리의 몇 가지 일반적인 측면이 설명됩니다(하위 절 5.2 유형 및 인스턴스 참조). 하위 조항 5.3에서 합성 i4.0 구성 요소의 처리가 설명됩니다. AAS의 또 다른 매우 중요한 측면은 식별 측면입니다(하위 조항 5.4 참조). 하위절 5.5에서는 이벤트 처리의 측면이 논의됩니다.

자산 관리 셸의 메타 모델에 대한 개요는 하위 조항 5.6에 나와 있습니다. 하위 조항 5.7에서 클래스는 모든 속성과 함께 자세히 설명됩니다.

Administration Shell의 보안 측면에 대한 메타 모델은 7절에 설명되어 있습니다.

UML 다이어그램을 이해하기 위한 범례와 클래스의 테이블 사양은 부록 C와 부록 D에 설명되어 있습니다.

UML 모델의 xmi 표현은 github 프로젝트 admin-shell의 "aas-specs" 리포지토리에서 찾을 수 있습니다.  
io [41]: <https://github.com/admin-shell-io/aas-specs/tree/master/schemas/xmi>

## 5.2 유형 및 인스턴스

### 5.2.1 자산 유형 및 자산 인스턴스의 수명 주기

Industrie 4.0은 공장, 생산 시스템, 장비, 기계, 구성 요소, 생산된 제품 및 원자재, 비즈니스 프로세스 및 주문, 비물질 자산(예: 프로세스, 소프트웨어, 문서, 계획, 지적 재산권 등)으로 구성된 자산에 대한 확장된 이해를 활용합니다. , 표준), 서비스 및 인력 등이 포함됩니다.

RAMI4.0 모델[3]은 IEC 62890에서 파생된 하나의 일반화된 수명 주기 축을 특징으로 합니다. 기본 아이디어는 Industrie 4.0 내의 모든 자산에 대해 가능한 유형과 인스턴스를 구별하는 것입니다. 이를 통해 재료 유형/재료 인스턴스, 제품 유형/제품 인스턴스, 머신 유형/머신 인스턴스 등 모든 요소에 유형/인스턴스 구분을 적용할 수 있습니다. 비즈니스 관련 정보는 RAMI4.0 모델의 '비즈니스' 계층에서 처리됩니다. 비즈니스 계층은 또한 자산 유형/인스턴스와 함께 주문 세부 정보 및 워크플로를 다룹니다.

표 1 자산 유형 및 인스턴스의 수명 주기 단계 및 역할

단계	설명	
유산 유형	개발	아이디어/개념화부터 첫 번째 프로토타입/테스트까지 유효합니다. 자산의 '유형'이 정의되고 구별되는 속성과 기능이 정의되어 구현됩니다. CAD 데이터, 회로도, 임베디드 소프트웨어와 같은 모든(내부) 설계 인공물이 생성되고 자산 유형과 연결됩니다.
	용법 유지	/ 생산 능력을 늘리고 있습니다. 기술 데이터 시트, 마케팅 정보와 같은 자산과 관련된 '외부' 정보가 생성됩니다. 판매 프로세스가 시작됩니다.
유산 사례	생산	자산 유형 정보를 기반으로 자산 인스턴스가 생성/생성됩니다. 생산, 물류, 자격 및 테스트에 대한 특정 정보는 자산 인스턴스와 연관됩니다.
	용법 유지	/ 자산 인스턴스 구매자의 사용 단계입니다. 사용 데이터는 자산 인스턴스와 연결되며 다음과 같을 수 있습니다.

		<p>자산 인스턴스의 제조업체와 같은 다른 가치 사슬 파트너와 공유됩니다.</p> <p>또한 자산 인스턴스의 유지보수, 재설계, 최적화 및 커미셔닝 해제가 포함됩니다. 전체 수명 주기 기록은 자산과 연결되며 문서화를 위해 보관/공유될 수 있습니다.</p>
--	--	--

표 1은 다양한 수명 주기 단계와 이러한 단계에서 자산 유형 및 자산 인스턴스의 역할에 대한 개요를 제공합니다. 가장 중요한 관계는 자산 유형과 자산 인스턴스 간의 관계입니다. 이 관계는 자산 인스턴스의 수명 동안 유지되어야 합니다. 이 관계를 통해 자산 유형에 대한 업데이트를 자동으로 또는 요청 시 자산 인스턴스로 전달할 수 있습니다.

참고: 자산 '유형'과 자산 '인스턴스'의 구분을 위해 이 문서에서는 '자산 종류'라는 용어를 사용합니다.

두 번째 유형의 관계는 자산 유형 및 인스턴스의 수명 주기 내 피드백 루프/정보입니다. 예를 들어 제품 자산의 경우 제품 인스턴스의 사용 및 유지 관리에 대한 정보가 (다음) 제품 유형으로도 제품의 제조를 개선하는 데 사용됩니다.

관계의 세 번째 클래스는 다른 자산 클래스의 자산과의 피드포워드/정보 교환입니다. 예를 들어, 비즈니스 자산에서 정보를 소싱하면 제품의 디자인 측면에 영향을 미칠 수 있습니다. 또는 제품의 디자인이 제조 라인의 디자인에 영향을 미칩니다.

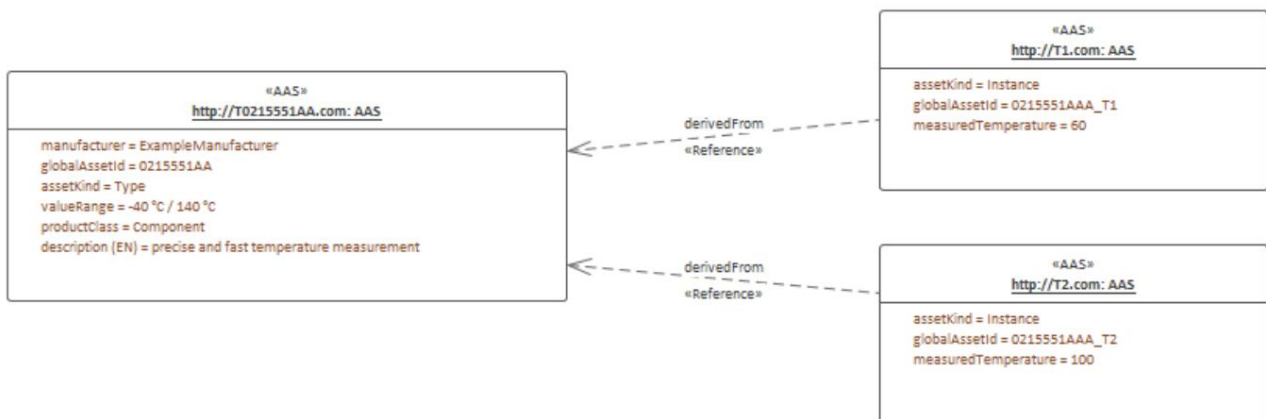
참고: 관계의 두 번째/세 번째 클래스에 대한 설명을 위해 NIST 모델도 제공합니다.

네 번째 유형의 관계는 서로 다른 계층 수준의 자산 사이에 있습니다. 예를 들어, 제조 스테이션과 현재 생산 중인 제품 간의 (동적) 관계가 될 수 있습니다. 물리적, 기능적 또는 안전 계층에서 생산 시스템의 분해일 수도 있습니다. 이러한 관계 유형에 따라 자동화 장비는 지능형 생산 및 자가 학습/최적화 작업을 수행하는 자동화 장치 및 제품의 복잡하고 상호 관련된 그래프로 설명됩니다.

## 5.2.2 자산 유형 및 자산 인스턴스의 예

다음 그림은 자산 유형 및 자산 인스턴스의 처리 및 처리에 대한 예를 제공합니다. 몇 가지 예시적인 정보도 포함됩니다. 자세한 설명은 다음 절에서 이어집니다.

그림 4 여러 AAS로 표시되는 자산의 예시 유형 및 인스턴스



참고: 예제는 이해의 편의를 위해 단순화되었으며 5절에 지정한 대로 메타 모델을 대략적으로 준수합니다. ID 처리도 단순화됩니다. 클래스 이름은 AAS의 고유한 전역 식별자에 해당합니다.

참고: Platform Industrie 4.0의 컨텍스트에서 유형 및 인스턴스는 일반적으로 "자산 유형" 및 "자산 인스턴스"를 나타냅니다. AAS의 유형 또는 인스턴스를 언급할 때 이것은 "AAS 유형" 및 "AAS 인스턴스"로 명시적으로 표시되어 둘 다를 혼동하지 않습니다. AAS 유형은 "AAS 템플릿"이라는 용어와 동의어로 사용됩니다.

참고: 유형 및 인스턴스의 IEC 정의는 2절을 참조하십시오. 이 문서의 범위에서 이러한 정의와 객체 지향의 유형/인스턴스 개념 간에 완전한 동등성은 없습니다.

프로그래밍(OO).

온도 센서의 구체적인 자산 유형과 이 유형의 고유하게 식별 가능한 물리적 온도 센서 2개가 있어야 합니다. 그 의도는 자산 유형과 모든 단일 자산 인스턴스에 대해 별도의 AAS를 제공하는 것입니다.

이 예에서 첫 번째 센서의 고유 ID는 "0215551AA\_T1"이고 두 번째 센서의 고유 ID는 "0215551AA\_T2"입니다. 첫 번째 센서의 AAS는 고유한 URI "http://T1.com/T1"을 갖고 두 번째 센서의 AAS는 고유한 URI "http://T2.com/T2"를 갖습니다. 둘 다의 자산 종류는 "인스턴스"입니다. 이 예는 두 센서의 작동 시간에 측정된 온도가 다르다는 것을 보여줍니다. T1의 경우 60°C, T2의 경우 100°C입니다. 당분간 우리는 AAS "http://T0215551AA.com"이 있는 두 AAS "T1" 및 "T2"의 "파생 출처" 관계를 무시합니다.

참고: 예제에서 HTTP 체계를 사용하더라도 URI는 유효한 URL일 필요가 없으므로 액세스 가능한 콘텐츠를 가리킬 필요가 없습니다.

참고: 물리적 단위는 "measuredTemperature" 요소의 의미 참조로 얻을 수 있습니다. 단순함을 위해 이 예에서는 표시되지 않습니다.

이 두 자산 인스턴스는 공유하는 정보가 많습니다. 자산 유형(이 예에서는 센서 유형) 정보입니다. 이 자산 유형에 대해 자체 AAS가 생성됩니다. 이 AAS의 고유 ID는 "http://T0215551AA.com"이고 센서 유형의 고유 ID는 "0215551AA"입니다. 이 경우 자산 종류는 "인스턴스"가 아닌 "유형"입니다. 이 온도 센서 유형의 모든 인스턴스에 대해 동일한 정보는 ProductClass(="구성 요소"), 제조업체(="ExampleManufacturer") 및 영어 설명 "정확하고 빠른 온도 측정" 및 값 범위 "-40 °C / 140 °C".

이제 두 자산 인스턴스의 두 AAS는 "derivedFrom" 관계 속성을 사용하여 자산 유형 "0215551AA"의 AAS를 참조할 수 있습니다.

참고: "속성"은 UML 의미에서 클래스(인스턴스)의 속성 또는 특성을 나타냅니다.

참고: 일반적으로 특정 자산 유형이 존재하는 경우 해당 자산 인스턴스 이전 시간에 존재합니다.

참고: AAS라는 용어는 AAS 인스턴스라는 용어와 동의어로 사용됩니다. AAS는 AAS 타입을 기반으로 구현될 수 있다. AAS 유형은 이 문서의 범위를 벗어납니다.

참고: 공개 표준화에서 AAS 유형이 표준화될 수 있습니다. 그러나 속성 유형(속성 정의 또는 개념 설명이라고 함) 또는 유형이 지정된 기타 하위 모델 요소와 완전한 하위 모델 유형을 표준화하는 것은 다른 AAS에서 재사용될 수 있기 때문에 훨씬 더 중요합니다.

참고: 사물 인터넷(IoT) 영역에서 자산 인스턴스는 일반적으로 "사물"로 표시되지만 자산 유형은 "제품"으로 표시됩니다.

### 5.2.3 자산 관리 셀 유형 및 인스턴스

이전 절에서는 자산의 유형과 인스턴스에 대해 설명했습니다. 분명히 질문은 AAS 유형과 AAS 유형을 조화시키는 방법에 대한 것입니다. 이 예에서 "globalAssetId" 속성이

**모든 AAS에 대해** "assetKind"와 전역 AAS 식별자 (id, 클래스 이름으로 표시됨)가 존재합니다. 그러나 표준이 없는 경우 "id", "globalised" 및 "kind"의 의미가 모든 AAS에 대해 동일하다는 것이 명확하지 않고 속성 중 어떤 것이 필수이고 어떤 속성에 대해 특정한지 명확하지 않습니다. 자산(유형 또는 인스턴스). 이것은 그림 5에 나와 있습니다.

이것이 이 문서의 목적입니다. 필수 속성을 정의하는 메타 모델의 정의

모든 AAS에 대해 선택 사항입니다. Asset Administration Shell에 대한 Platform Industrie 4.0 메타모델은 5절에 정의되어 있습니다.

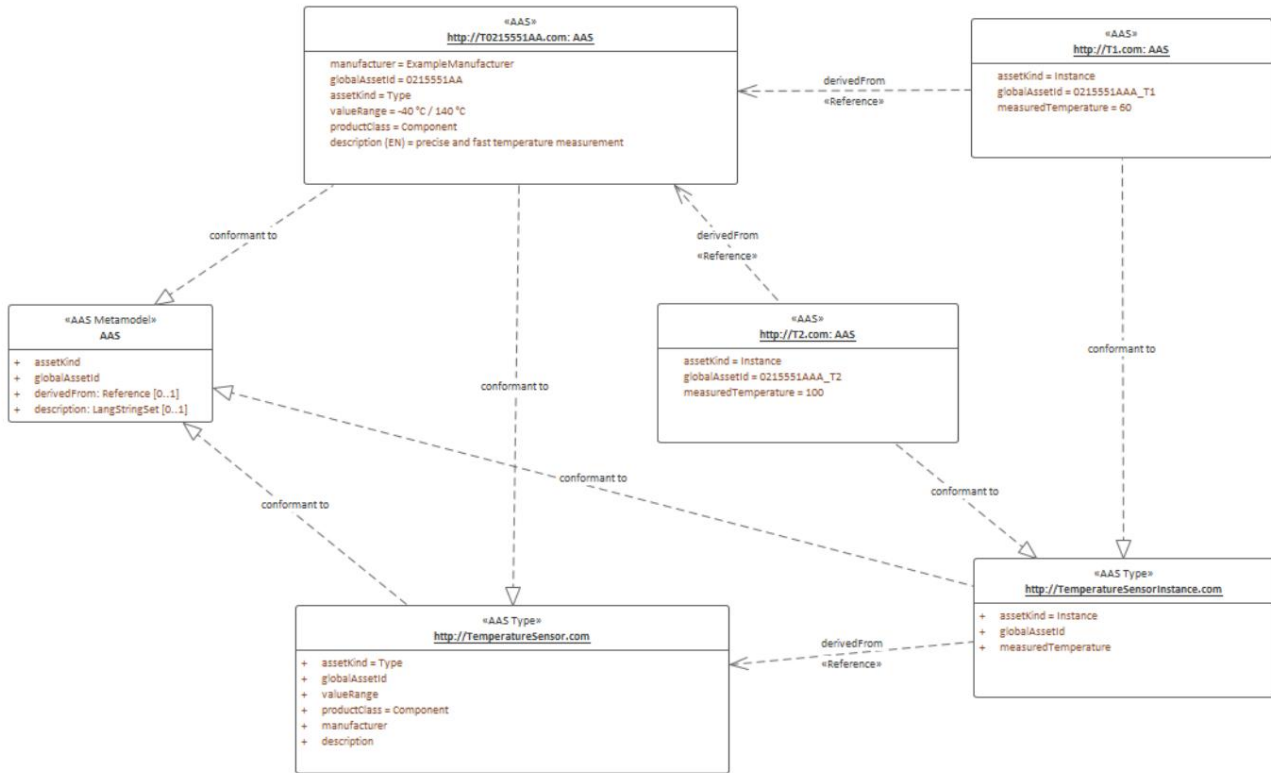
참고: 이 접근 방식은 요구 사항 tAAS-#19가 충족되도록 합니다. 또 다른 접근 방식은 두 개의 메타모델을 정의하는 것이었습니다. 하나는 자산 유형용이고 다른 하나는 자산 인스턴스용입니다. 그러나 하나의 메타 모델을 사용하도록 동기를 부여한 많은 유사점.

참고: 메타모델 자체는 필수 하위 모델을 규정하지 않습니다. 이것은 AAS Type 수준의 하위 모델 처방과 유사한 표준화의 또 다른 단계입니다.

참고: AAS 유형은 이 문서에 정의된 AAS의 메타모델을 기반으로 구현되어야 합니다. 이 메타모델 "AAS 메타모델"이라고 합니다.

참고: AAS(인스턴스)를 정의하기 전에 AAS 유형을 정의해야 하는 것은 아닙니다. AAS 유형을 구현하지 않는 AAS 인스턴스는 이 문서에 정의된 AAS의 메타 모델을 기반으로 구현해야 합니다.

그림 5 AAS의 메타모델, AAS 유형 및 AAS 인스턴스 간의 예시적인 관계



## 5.3 복합 I4.0 구성 요소

5.2.1절에 설명된 것처럼 서로 다른 계층 수준의 자산 간에 관계 클래스가 있습니다. 이 관계 클래스에 의해 자동화 장비는 자동화의 복잡하고 상호 관련된 그래프로 설명됩니다.

장치 및 제품, 지능형 생산 및 자가 학습/최적화 작업을 수행합니다.

복합 I4.0 구성 요소에 대한 세부 정보 및 예는 [12]에서 찾을 수 있습니다.

AAS 메타모델의 다음 모델링 요소를 사용하여 이러한 복합 I4.0 구성 요소를 실현할 수 있습니다.

- RelationshipElement – 자산과 다른 요소 간의 관계를 설명하는 데 사용됩니다.
- 하위 모델 복합 자산은 다른 엔터티와 자산으로 구성됩니다. 이러한 엔터티 및 자산 서로의 관계와 함께 BOM에 지정됩니다.

참고: 이러한 BOM의 구조를 정의하는 하위 모델 템플릿은 AAS에서 미리 정의하지 않습니다. 메타 모델이지만 Entity 요소를 포함하는 것으로 가정합니다.

- 자산 BOM의 일부인 모든 엔터티 (엔티티)에 자체 자산 관리 쉘이 있는 것은 아닙니다. [12]에서 설명한 것처럼 자체 관리 개체는 공동 관리 개체 (Entity/entityType)와 구별됩니다.

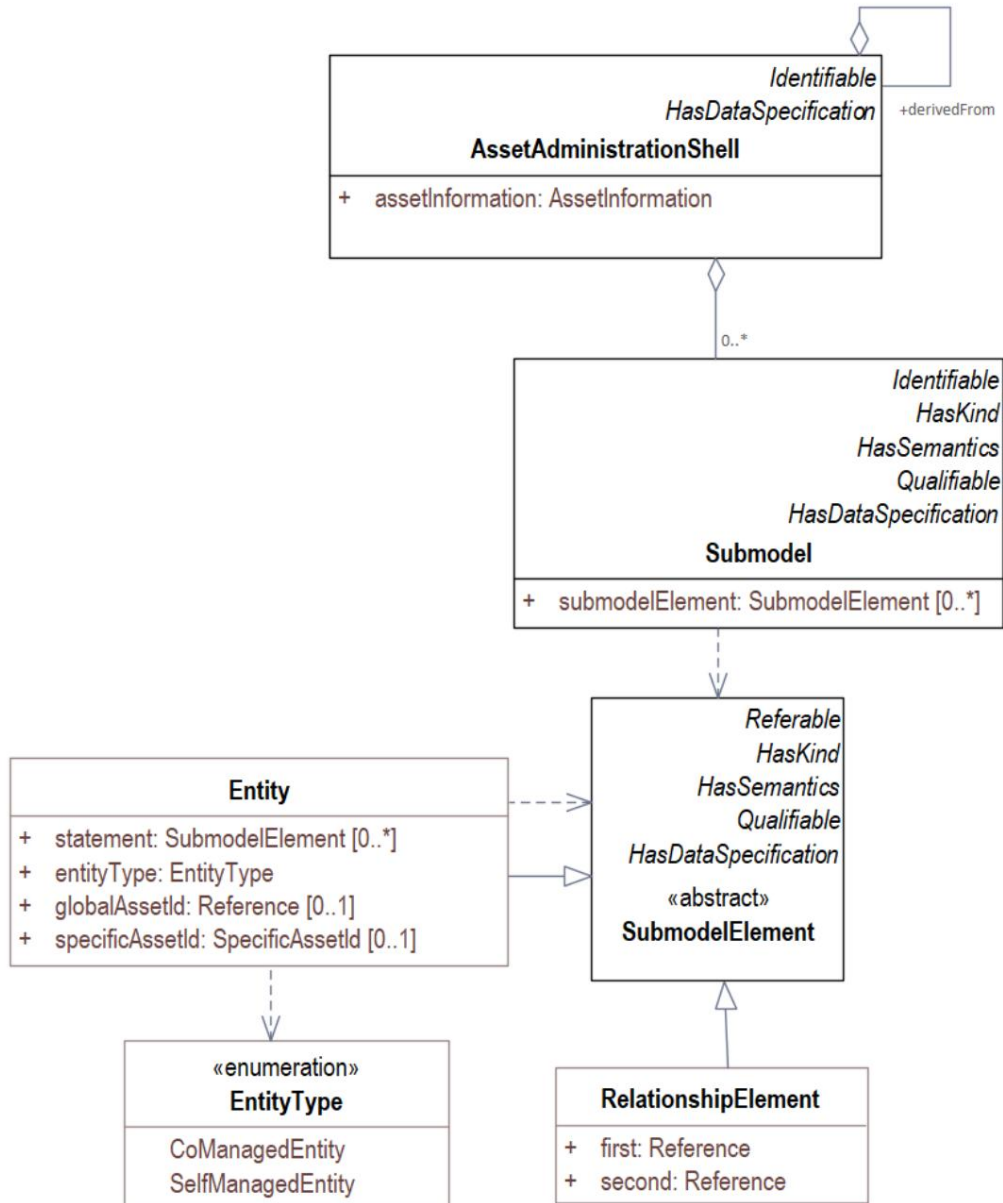
○ 자체 관리 엔터티에는 자체 AAS가 있습니다. 이것이 이 자산에 대한 참조도 지정되는 이유입니다( Entity/globalAssetId 또는 Entity/specificAssetId를 통해). 또한 자산 자체의 AAS에 지정되지 않은 자산에 추가 자산 설명 (Entity/statement) ([15]과 비교)을 추가할 수 있습니다.

I4.0 구성 요소 전용.

- 공동 관리 법인의 경우 별도의 AAS가 없습니다. 그러한 엔터티의 관계 및 속성 설명은 복합 14.0 구성 요소의 AAS 내에서 관리됩니다.

그림 6은 합성 14.0 구성 요소를 설명하는 데 가장 중요한 요소를 포함하는 나중에 소개되는 메타 모델의 추출을 보여줍니다.

그림 6 복합 14.0 구성 요소에 대한 메타 모델에서 추출



## 5.4 요소의 식별

### 5.4.1 개요

스마트 제조 영역 내에서 다양한 요소를 고유하게 식별하려면 [4]에 따라 식별자가 필요합니다. 이러한 이유로 관리 셀에 대한 공식 설명의 기본 요소입니다. 특히 다음과 같은 경우 최소한 신분증이 필요합니다.

- 자산 관리 셀,
- 자산(AssetAdministrationShell/assetInformation/globalAssetId의 값으로),

- 하위 모델 인스턴스 및 하위 모델 템플릿, • ECLASS 또는 IEC CDD와

같은 외부 저장소의 속성 정의/개념 설명

식별은 두 가지 목적으로 수행됩니다.

- (1) 관리 셀의 모든 요소와 그것이 나타내는 자산을 고유하게 구별하기 위해
- (2) 의미론을 관리 셀의 이러한 데이터 및 기능 요소에 바인딩하기 위해 하위 모델 템플릿 및 속성 정의와 같은 외부 정의에 요소를 연결합니다.

## 5.4.2 어떤 식별자가 있습니까?

[4], [20]에는 두 가지 표준 준수 전역 식별 유형이 정의되어 있습니다.

- IRDI - 속성 및 분류에 대한 식별자 체계로서의 ISO29002-5, ISO IEC 6523 및 ISO IEC 11179-6 [20]. 컨소시엄 차원의 사양 또는 국제 표준화 과정에서 생성됩니다. 이를 위해 사용자는 함께 앉아서 그들의 아이디어를 컨소시엄이나 표준화 기구에 제공합니다. ISO, IEC의 속성은 주요 상업적 이익을 보호하는 데 도움이 됩니다.

ECLASS 및 기타 저장소와 같은 저장소를 사용하면 비교적 많은 수의 식별자를 적절하게 짧은 시간에 표준화할 수 있습니다.

- IRI - IRI(Rfc 39871 ) 또는 RFC 3986에 따라 자산, 관리 셀 및 기타(표준화되지는 않았지만 전 세계적으로 고유한) 속성 및 분류의 식별을 위한 URI 및 URL 입니다.

다음도 허용됩니다.

- 사용자 지정 - 제조업체가 관리 셀 내에서 모든 종류의 사내 목적에 사용할 수 있는 UUID/GUID(전역 고유 식별자/범용 고유 식별자<sup>3</sup>)와 같은 내부 사용자 지정 식별자 입니다.

이는 IRI/URI/URL 및 내부 사용자 지정 식별자가 표준화된 정보 및 기능뿐만 아니라 관리 셀 및 4.0 인프라의 제조업체별 정보 및 기능을 나타내고 전달할 수 있음을 의미합니다. 하나의 인프라가 두 가지 목적을 모두 수행할 수 있습니다.

CLSID는 GUID의 URI입니다. 고객별 스키마로 시작합니다. 따라서 Custom은 고객 고유 식별자가 IRDI나 IRI가 아닌 경우에만 사용해야 합니다.

전역 식별자 외에도 정의된 네임스페이스(일반적으로 상위 요소) 내에서만 고유한 식별자도 있습니다. 이러한 식별자를 로컬 식별자라고도 합니다. 예: 하위 모델 내의 속성에는 로컬 식별자가 있습니다.

절대 URI 외에도 상대 URI도 있습니다.

식별에 대한 추가 정보는 DIN SPEC 91406 [43]도 참조하십시오.

## 5.4.3 자산 및 관리 셀의 식별자

스마트 제조 영역의 경우 자산은 식별자(ID)를 통해 전 세계적으로 고유하게 식별되어야 합니다[4][20]. 관리 셀에는 고유한 ID도 있습니다.

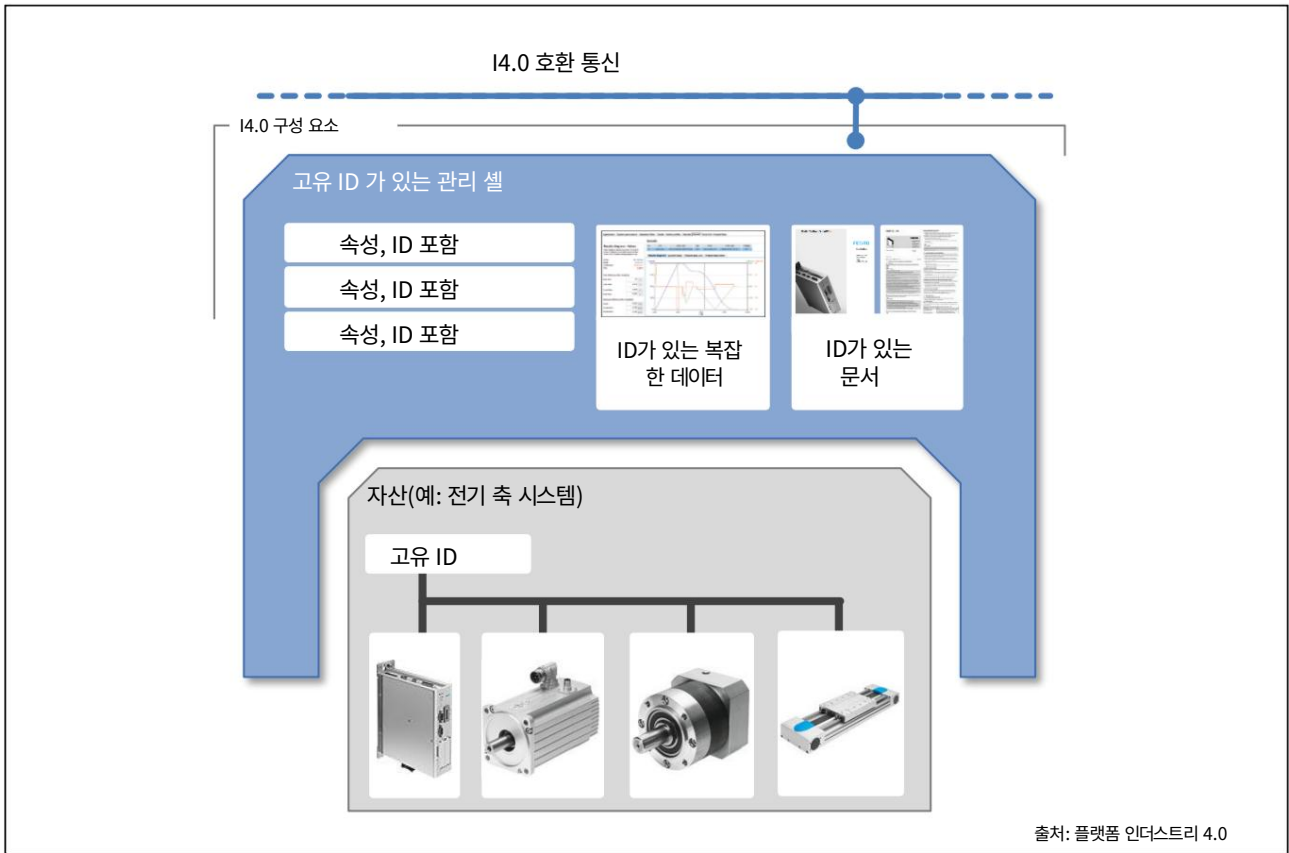
---

<sup>1</sup> <https://tools.ietf.org/html/rfc3987>

<sup>2</sup> <https://tools.ietf.org/html/rfc3986>

<sup>3</sup> [https://en.wikipedia.org/wiki/Universally\\_unique\\_identifier](https://en.wikipedia.org/wiki/Universally_unique_identifier).

그림 7 설명 중인 관리 셀 및 자산의 고유 식별자([4]에서 수정된 그림)



관리 셀은 고유한 자산 ID를 가진 정확히 하나의 자산을 나타냅니다. 배치 기반 생산에서 배치는 자산이 되며 해당 관리 셀에서 설명합니다. 자산 집합이 Administration Shell에 의해 설명되어야 하는 경우 복잡한 자산에 대한 고유 ID를 생성해야 합니다[12].

자산의 ID는 [4][20]에 따른 전역 식별자에 대한 제한 사항을 준수해야 합니다. 자산에 추가 식별 정보, 일련 번호 등이 있는 경우 자산 자체의 고유한 글로벌 식별자와 혼동하지 마십시오 .

#### 5.4.4 어떤 요소에 사용할 식별자

자산 관리 셀을 나타내는 UML 모델의 모든 요소에 모든 식별자를 적용할 수 있는 것은 아닙니다. 따라서 표 2는 "식별 가능" 또는 "HasSemantics"를 구현하는 다양한 엔터티에 대한 다양한 제약 조건 및 권장 사항에 대한 개요를 제공합니다. 5.6절의 메타모델과 관련된 속성 5.7절.

표 2 식별 값이 허용되는 요소

집단 가치 식별	-와 함께 기인하다	허용된 식별자 (권장 또는 일반)	비고
자산 관리 곁대기	ID	IRI(URL)	필수적인 일반적으로 URL이 사용됩니다.

4 이러한 추가 자산 식별자는 AssetInformation/specificAssetIds에 포함됩니다.



집단 가치 식별	~와 함께 기인하다	허용된 식별자 (권장 또는 일반)	비고
	아이디 쇼트	끈	옵션5
종류가 있는 하위 모델 = 템플릿	ID	IRDI, IRI(URI)	필수적인  IRDI, 정의된 하위 모델이 표준화되고 IRDI가 적용된 경우
	아이디 쇼트	끈	추천  일반적으로 Instance 종류의 하위 모델에도 idShort로 사용됩니다.
	의미 ID	IRDI, IRI(URI)	선택 과목  의미론적 ID는 하위 모델의 공식을 설명하는 외부 정보 소스를 참조할 수 있습니다(예: 표준인 경우 PDF).
종류가 있는 하위 모델 = 인스턴스	ID	IRI(URI), 사용자 지정	필수적인
	아이디 쇼트	끈	추천  일반적으로 idShort 또는 영어 짧은 이름 semanticId를 통해 참조되는 하위 모델 템플릿
	의미 ID	IRDI, IRI(URI)	추천  일반적으로 의미 체계는 하위 모델의 의미 체계를 정의하는 외부 표준에 대한 외부 참조입니다.
하위 모델 요소 idShort		끈	필수적인  일반적으로 semanticId를 통해 참조되는 개념 정의의 영어 짧은 이름
	의미 ID	IRDI, <small>아이디</small> (URI), 관습	추천  글로벌 ID를 통해 외부 저장소의 개념 정의 또는 개념 정의에 대한 링크
개념 설명 ID		IRDI, IRI, 커스텀	필수적인  ConceptDescription에는 전역 ID가 있어야 합니다. 개념 설명이 ECLASS 또는 IEC CDD와 같은 외부 사전의 사본인 경우 외부 사전에서 사용된 것과 동일한 전역 ID를 사용할 수 있습니다.
	아이디 쇼트	끈	추천  예: 영어 짧은 이름과 동일
	isCaseOf	IRDI, IRI(URI)	선택 과목  외부 저장소의 개념 정의에 대한 링크

5 참고: 이 사양의 버전 V1.0에서 idShort는 Identifiables에 대해 선택 사항이었습니다. 이것은 V2.0에서 변경되었습니다. idShort는 모든 Referable에 대해 필수로 설정되었습니다. V3.0RC02에서는 idShort가 다시 선택 사항이 되었습니다.

집단 가치 식별	~와 함께 기인하다	허용된 식별자 (권장 또는 일반)	비고
예선	의미 ID	IRDI, 아이디 관습 (URI),	추천  외부 저장소의 한정자 유형 정의에 대한 링크  IRDI, 정의된 한정자 유형이 표준화되고 IRDI가 적용된 경우

#### 5.4.5 새 식별자는 어떻게 생성됩니까?

5.4.3절의 다양한 식별 유형에 따라 다음과 같이 기술할 수 있습니다.

- (a) IRDI는 특정 관리 셀 생성과 관련하여 외부 사양 및 표준화 프로세스에 의해 이미 존재하는 것으로 가정합니다. 이러한 IRDI 식별자를 사용하려면 문서 [4]의 5절을 참조하십시오.
- (b) URI와 URL은 특정 관리 셀을 만들 때 즉석에서 개발자 자신이 쉽게 구성할 수 있습니다. 필요한 것은 예를 들어 회사의 유효한 권한과 도메인(예: admin-shell.io)이 구성되는 방식이 다음 경로를 보장하는지 확인하는 것입니다.  
  
호스트 이름에 추가된 식별자는 이러한 식별자에 대해 의미상 고유한 방식으로 예약됩니다. 이러한 방식으로 각 개발자는 호스트 이름과 선택한 경로를 결합하여 임의의 URI 또는 URL을 만들 수 있습니다. 이 경로는 개발자 조직에서 고유해야 합니다.
- (c) 사용자 지정 식별자는 개발자 스스로도 쉽게 만들 수 있습니다. 필요한 것은 검색할 해당 프로그래밍 기능뿐입니다. 내부 사용자 지정 식별자가 (a) 또는 (b)와 명확하게 구별될 수 있도록 해야 합니다.
- (d) 로컬 식별자는 즉시 생성할 수도 있습니다. 네임스페이스 내에서 고유해야 합니다.

#### 5.4.6 의미 식별자를 위한 매칭 전략

두 요소를 비교할 때 이 두 요소가 의미적으로 어떻게 관련되어 있는지 설명할 수 있는 것으로 간주해야 하는 다른 사용 사례가 있습니다. 이 장에서는 일치하는 의미 식별자를 처리할 때 고려해야 할 측면에 대한 첫 번째 힌트를 제공합니다. 예를 들어 ECLASS에서 IRDI-Path로 가능한 것과 같은 컨텍스트 정보를 포함하는 의미론적 참조는 아직 고려되지 않습니다.

- 정확한 일치(동일한 semanticIds) - DEFAULT
  - o 정확히 일치하는 두 개의 의미론적 ID는 문자열과 동일해야 합니다.
  - o 예: idShort "ManufacturerName" + semanticId 0173-1#02-AAO677#002가 있는 속성 및 idShort "Herstellername" + semanticId 0173-1#02-AAO677#002가 있는 속성은 완전히 동일한 의미 체계를 갖습니다.
- 지능형 매칭(호환되는 semanticId)
  - o 버전 관리 무시
    - 지능형 매칭을 통해 개념 정의의 다른 버전을 매칭할 수 있습니다.  
의미론적 버전 관리만 호환되는 경우, 즉 상위 또는 하위 호환이 가능한 경우 버전이 일치해야 합니다.  
  
예 1: idShort "ManufacturerName" + semanticId 0173-1#02-가 있는 속성 AAO677#002 및 idShort "Herstellername" + semanticId 0173-1#02-가 있는 속성 AAO677#003은 동일한 의미를 갖습니다. 참고: 두 가지 의미론적 ID를 비교하려면 버전 관리에 대한 지식이 있어야 합니다. 예에서 ECLASS의 두 IRDI가 비교됩니다. ECLASS 규칙은 의미 체계가 항상 새 버전에 대해 이전 버전과 호환되도록 합니다. 주요 변경 사항에 대해 새 IRDI가 생성됩니다.
  - o 시맨틱 매핑 고려
    - 지능형 매칭을 통해 기존의 의미 매핑 정보를 고려할 수 있습니다.  
의미 매핑은 하나의 동일한 사전 내에 존재할 수 있지만 다른 사전과 온톨로지 간에도 존재할 수 있습니다.

예: 사전에 있는 배터리의 공칭 용량에 대한 0112/2///61360\_4#AAE530  
ECLASS의 IEC CDD 및 0173-1#02-AAI048#004는 동일한 의미를 갖습니다 6 7 8

o 도메인 지식 고려

- 기계가 읽을 수 있는 형태로 제공되는 지능형 매칭 도메인 지식은 예를 들어 두 개념 정의 간의 "is-a" 관계를 고려할 수 있습니다.

예: 해머 드릴(0173-1#01-ADS698#010) 및 타악기 드릴(0173-1#01-ADS700#010)은 광물성 드릴(0173-1#01-ADN177#005)이므로 광물성 드릴에 대한 요구 또는 제약 조건에 호환됩니다.

## 5.4.7 URI 식별자 생성을 위한 모범 사례

14.0 구성 요소 [17]에 대한 의미론 및 상호 작용에 대한 접근 방식은 다음 구조의 사용을 제안합니다.

(표 3 참조) URI9 , 여기에서 약간 수정되었습니다. 아이디어는 항상 체계에 따라 URI를 구조화하는 것입니다. 다른 요소의. 그러나 이것은 권장 사항일 뿐이며 반드시 사용해야 하는 것은 아닙니다.

표 3 URI에 대한 제안 구조

요소	설명	구문 구성 요소
조직	ID를 발급하는 법인, 행정 단위 또는 회사	t
조직 소단위/ 문서 ID/ 문서 하위 단위	위 조직의 하위 엔터티 또는 위 조직의 릴리스 사양 또는 간행물.	피
하위 모델/도메인 ID	식별자가 속한 자산 또는 관리 셀의 기능적 또는 지식 기반 도메인의 하위 모델입니다.	피
버전	사양 릴리스 또는 식별자 발행에 따른 버전 번호	피
개정	사양 릴리스 또는 식별자 발행에 따른 개정 번호	피
속성/요소-ID 관리 셀의 속성 또는 추가 구조적 요소 ID		피
인스턴스 번호	사양 또는 간행물의 릴리스 내 인스턴스의 개별 번호 지정	피

표에서 구문 구성 요소 "A"는 RFC 3986(URI)의 권한과 RFC 2141의 네임스페이스 식별자를 나타냅니다.  
(항아리); "P"는 RFC 3986(URI)의 경로와 RFC 2141(URN)의 네임스페이스 특정 문자열을 나타냅니다.

<AAS URI> ::= <체계> “.” <권한> [ <경로> ]

<scheme> ::= 유효한 URI 체계

<권한> ::= <기관>

<경로> ::= <서브유닛> <도메인> <릴리스> <요소>

6 참고: 이 예는 기존 의미 매핑을 나타내지 않고 단지 후보일 뿐입니다.

7 시맨틱 매핑 파일은 ECLASS Classic과 ECLASS Advanced 간의 ECLASS에서도 사용됩니다. [https://wiki.eclass.eu/wiki/Transaction\\_Update\\_File](https://wiki.eclass.eu/wiki/Transaction_Update_File)

8 ECLASS에서 시맨틱 매핑에 사용되는 형식입니다. <https://www.eclass.eu/static/eClassXML/3.0/eCl@ssXML/mapping.xsd>

9개의 URL은 URI이기도 합니다.

<하위 단위> ::= [ (“/” | “.”) <조직 하위 단위/문서 ID/문서 하위 단위> ]\*

<도메인> ::= [ (“/” | “.”) <하위 모델 / 도메인-ID>

<릴리스> ::= [ (“/” | “.”) <버전> [ (“/” | “.”) <개정> ]\* ]

<요소> ::= [ (“/” | “.” | “#”) ( <속성/요소-ID> | <인스턴스 번호> )\* ]

이 체계를 사용하여 유효한 URN과 URL을 만들 수 있으며 둘 다 URI입니다. 관리 셀을 사용하는 경우 URL도 선호되며 기능(예: REST 서비스)을 식별자에 바인딩할 수 있습니다.

이러한 식별자의 예는 표 4에 나와 있습니다.

표 4 자산 관리 셀의 URN 및 URL 기반 식별자 예

식별자	설명	속성 클래스	예
관리 셀 ID	아이디 관리 셀	기초	항아리: zvei: SG2: aas: 1: 1: demo11232322  http://www.zvei.de/SG2/aas/1/1/demo11232322
하위 모델 ID(유형) 하위 모델	유형 식별	선택된 하 위 모델 기반, 기타 무료 ~이다	항아리:GMA:7.20:계약 협상:1:1  http://www.vdi.de/gma720/contractnegotiation/1/1
하위 모델 ID (사례)	신원 확인 하위 모델의 인스턴스	무료	항아리:GMA:7.20:계약 협상:1:1#001  http://www.vdi.de/gma720/ contractnegotiation/1/1#001
속성/매개변수 /상태 유형 ID	신원 확인 속성, 매개변수 및 상태 유형	도메인별	항아리: PROFIBUS: PROFIBUS-PA: V3- 02:매개변수:1:1:최대온도  http://www.zvei.de/SG2/aas/1/1/demo11232322/ 최대 온도
속성/매개변수 /status 인스턴스 ID  (메타모델에서 사 용하지 않음)	속성, 매개변수 및 상태 인스 턴스 식별	도메인별	항아리: PROFIBUS: PROFIBUS-PA: V3- 02:매개변수:1:1: MaxTemp#0002  http://www.zvei.de/SG2/aas/1/1/demo11232322/ maxtemp#0002

참고: 표 4의 마지막 행은 완료용으로만 사용됩니다. 메타 모델은 식별자를 예측하지 않습니다.  
속성/매개변수/상태 인스턴스.

## 5.4.8 기존 하위 모델 템플릿을 기반으로 하위 모델 인스턴스 만들기

기존 하위 모델 템플릿을 인스턴스화하려면 예를 들어 Plattform Industrie 4.0의 출판물 통해 하위 모델 템플릿의 공개 사양이 있어야 합니다. 특별한 경우로 제조업체 사양과 같은 비공개 하위 모델 템플릿에서 하위 모델을 인스턴스화하는 것도 가능합니다.

2020년 11월 자산 관리 셀을 위한 처음 두 개의 하위 모델 템플릿이 게시되었습니다. 하나는 명판([52])이고 다른 하나는 일반 기술 데이터([51])입니다. 다른 사람들이 따랐고 따를 것입니다. 등록된 하위 모델 템플릿의 개요는 [60]을 참조하십시오.

각 하위 모델 템플릿에는 의미 참조로 사용할 개념 정의의 식별자가 미리 정의되어 있습니다. 이러한 하위 모델을 인스턴스화하려면 속성 정의에 대한 의미론적 참조가 있는 속성을 생성하고 값을 첨부하기만 하면 됩니다. 하위 모델 요소의 다른 하위 유형도 마찬가지입니다.

템플릿 자체에서 정의할 수 없는 유일한 것은 하위 모델 인스턴스 자체의 고유 ID(하위 모델 템플릿의 ID와 동일하지 않음)와 속성 값 등입니다. 템플릿은 또한 카디널리티를 정의합니다. 요소는 선택 사항인지 여부입니다. 하위 모델 요소 목록의 경우 일반적으로 둘 이상의 요소가 포함됩니다. 템플릿에는 예시적인 요소 템플릿이 포함됩니다. 이 템플릿에서 복사/붙여넣기를 통해 다른 요소를 만들 수 있습니다.

#### 5.4.9 신규 또는 독점 하위 모델을 형성할 수 있습니까?

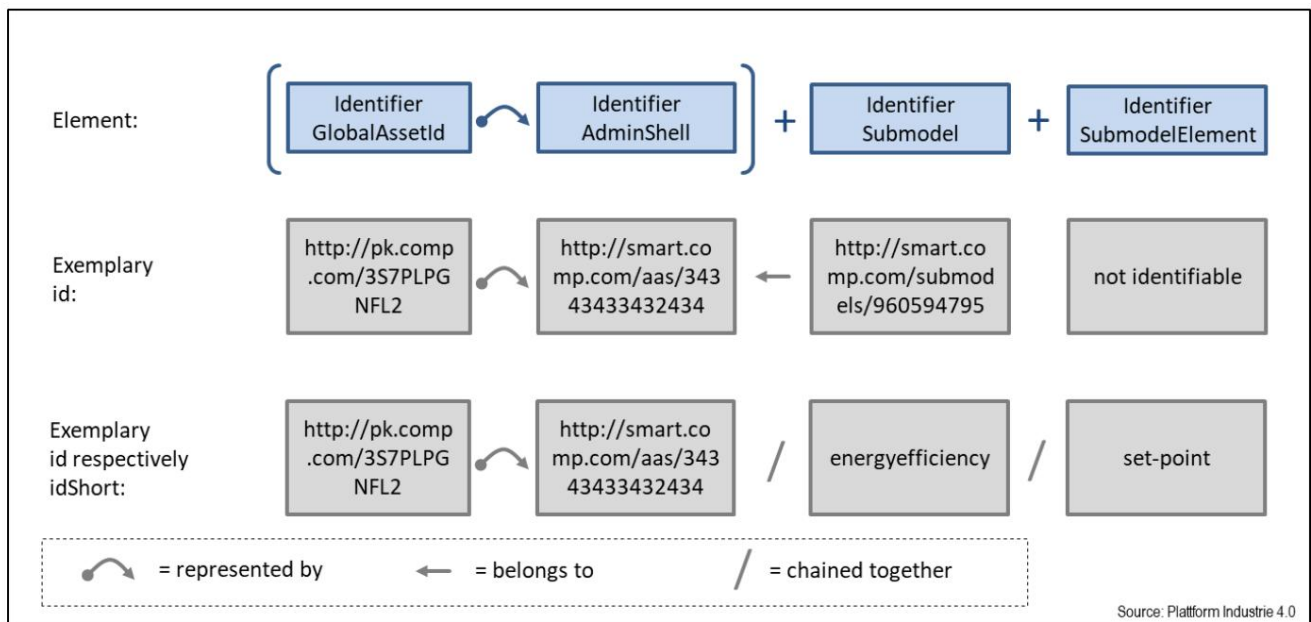
자유 및 독점 하위 모델을 포함하여 가능한 한 많은 하위 모델이 형성되는 것은 Industrie 4.0의 이익입니다(→ [4], "자유 속성 세트"). 자산의 특정 관리 셀에 대해 언제든지 하위 모델을 구성할 수 있습니다. 이를 위해 관리 셀 제공자는 섹션 5.4.5에 따라 하위 모델의 유형 및 인스턴스에 대한 내부 식별자를 구성할 수 있습니다. 모든 I4.0 시스템은 개별적으로 알려지지 않은 하위 모델 및 속성을 무시하고 단순히 "간과"하도록 요청됩니다. 이러한 이유로 항상 관리 셀에 독점 정보, 예를 들어 제조업체별 또는 사용자별 정보, 하위 모델 또는 속성을 저장하는 것이 가능합니다.

참고: 관리 셀의 의도에 따라 소유권 정보도 포함됩니다. 예를 들어, 전사적 식별 체계 또는 전사적 데이터 처리에 필요한 정보에 연결하기 위해, 이를 통해 단일 인프라를 사용하여 표준화된 정보와 독점 정보를 동시에 전송할 수 있습니다. 이것은 또한 새로운 정보 요소의 도입(및 이후의 표준화)을 전달합니다.

#### 5.4.10 식별 가능한 요소에 대한 짧은 ID의 사용

Administration Shell은 전 세계적으로 고유한 식별자의 사용을 크게 촉진합니다. 그러나 어떤 경우에는 이것이 비효율적으로 이어질 수 있습니다. 예를 들어 관리 셀의 일부인 하위 모델의 일부인 속성을 참조할 수 있으며 이들 각각은 전역 식별자로 식별됩니다[4]. 예를 들어, 자원 지향 아키텍처(ROA)를 특징으로 하는 애플리케이션에서 전 세계 고유 자원 로케이터(URL)는 일련의 세그먼트로 구성될 수 있으며, 이는 차례로 전 세계적으로 고유할 필요가 없습니다.

그림 8 예시 식별자 및 idShort의 동기



관리 셀의 API에 의한 요소의 이러한 효율적인 주소 지정을 허용 하기 위해 이러한 종속 요소(→ 5.6)를 참조하기 위해 추상 클래스 Referable에서 상속하는 메타 모델의 클래스 집합에 대해 idShort가 제공됩니다. 그러나 관리 셀의 리소스 주소를 지정하는 외부 시스템

id 또는 idShort (→ 5.7.2) 로 요소에 액세스하기 전에 먼저 해당 의미론, 즉 semanticId 값을 확인해야 합니다.

## 5.5 이벤트

### 5.5.1 개요

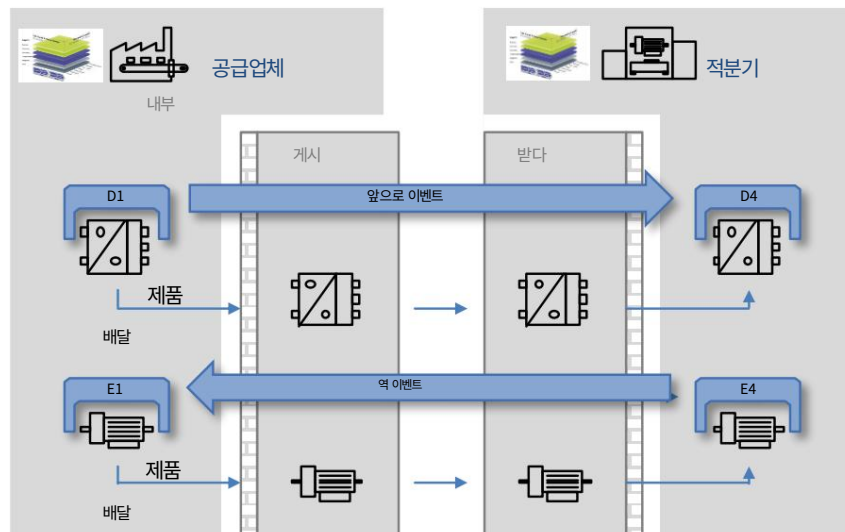
이벤트는 AAS의 매우 다양한 메커니즘입니다. 다음 섹션에서는 먼저 이벤트에 대한 몇 가지 사용 사례를 설명합니다. 요구 사항을 설명하기 위해 다양한 유형의 이벤트가 요약되어 있습니다. 하위 모델 요소

AAS의 이벤트를 선언할 수 있는 "이벤트"가 도입되었습니다. 이벤트 메시지의 일반 형식이 지정됩니다.

### 5.5.2 자산 관리 셀에서 사용되는 이벤트에 대한 간략한 사용 사례

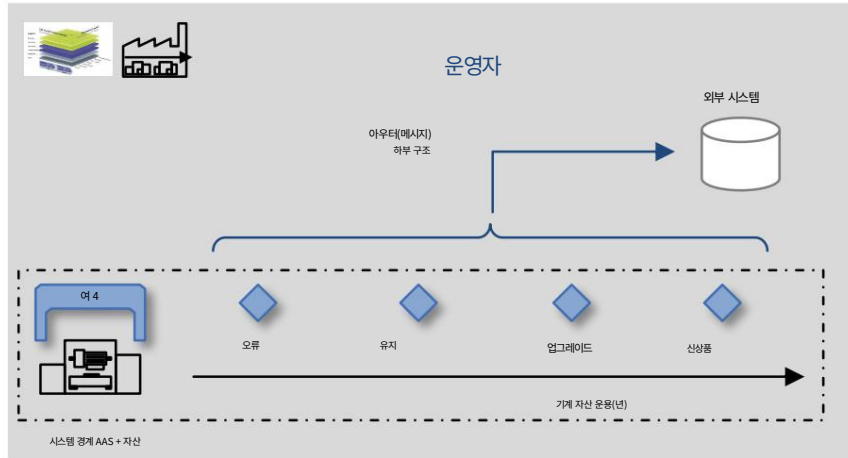
- 통합자가 장치를 구입했습니다. 나중에 장치 공급업체에서 새 펌웨어를 제공합니다. 통합자는 새 펌웨어 제안을 감지하고 적합성("전달 이벤트")을 평가한 후 펌웨어를 업데이트하려고 합니다. 메커니즘은 종속 AAS("D4")가 상위 또는 유형 AAS("D1")의 이벤트를 감지한다는 것입니다. 이는 파생된 From 관계에 의해 설명됩니다.
- 통합자/운영자가 공급업체로부터 구입한 모터를 작동합니다. 작동 중에 상태 모니터링 사고가 발생합니다. 양 당사자는 가용성을 제공하는 비즈니스 모델에 동의합니다. 따라서 공급자는 가치 사슬에서 더 멀리 있는 장치 상태를 모니터링하기를 원합니다("역 이벤트").

그림 9 정방향 및 역방향 이벤트



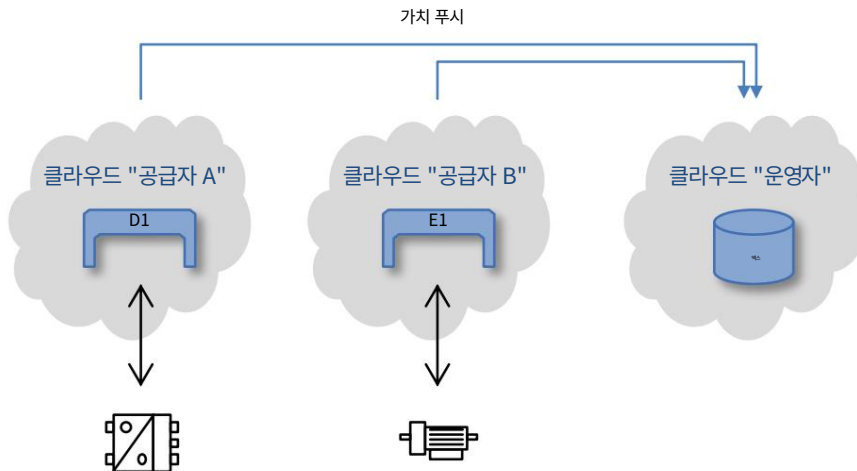
- 운영자는 시간이 지남에 따라 특정 I4.0 구성 요소를 작동하고 있습니다. 때때로 다른 시스템에서 이러한 I4.0 구성 요소가 변경됩니다. 문서화 및 감사를 위해 이 I4.0 구성 요소에 대한 변경 사항을 추적해야 합니다. 이는 시간 경과에 따른 이벤트를 기록하여 달성할 수 있습니다.

그림 10 이벤트를 통한 변경 추적



- 운영자는 제조업체 클라우드에 배포되는 다양한 I4.0 구성 요소를 운영하고 있습니다. 운영자는 DIN SPEC 92222에 따라 이러한 구성 요소의 데이터를 통합하기를 원합니다. 따라서 정보는 운영자 클라우드("값 푸시")로 전달되어야 합니다.

그림 11 클라우드 전반의 값 푸시 이벤트



### 5.5.3 이벤트의 입출력 방향

관찰된 모델, 각각의 참조 대상에 대해 이벤트의 입력 및 출력 방향을 구별하는 것이 관련될 수 있습니다.

방향	설명
산출	이벤트가 연결된 참조 대상을 모니터링 하고 있습니다. OPC UA, MQTT 또는 AMQP와 같은 외부 메시지 인프라는 이러한 이벤트를 다른 AAS 및 추가 외부 시스템 및 사용자에게 전송합니다.
입력	각각의 Referable 을 구현하는 소프트웨어 엔터티는 들어오는 이벤트를 처리할 수 있습니다. 이러한 들어오는 이벤트는 OPC UA, MQTT 또는 AMQP와 같은 외부 메시지 인프라에 의해 참조 대상의 소프트웨어 엔터티에 전달됩니다.

### 5.5.4이벤트 유형

위의 사용 사례에 따라 다양한 유형의 이벤트가 가능합니다. 다음 표는 가능한 이벤트 유형에 대한 인상을 제공합니다. 각 이벤트 유형은 semanticId 로 식별되며 특수 페이로드를 특징으로 합니다.

그룹	방향 10 동기/조건	
의 구조적 변화 밖으로	AAS	<ul style="list-style-type: none"> <li>하위 모델, 자산, 하위 모델 요소의 <b>CRUD</b><sup>11</sup> 등등</li> </ul>
	-연	<ul style="list-style-type: none"> <li>상위/유형/ 파생 AAS 에 대한 <b>업데이트 감지</b></li> </ul>
속성 및 종속 속성 업데이트	밖으로	<ul style="list-style-type: none"> <li>SubmodelElements의 값 업데이트</li> <li>타임스탬프 업데이트 및 시계열 업데이트</li> <li>업데이트 이벤트의 명시적 트리거</li> </ul>
AAS 운영	밖으로	<ul style="list-style-type: none"> <li>모니터링(오래 지속되는) <b>실행</b> OperationElement 및 실행 중 이벤트 업데이트</li> </ul>
모니터링, 조건부, 계산된 이벤트	밖으로	<ul style="list-style-type: none"> <li>예를 들어 일부 제한을 무효화하는 경우(예: 예선에서 표현 의미와 함께)</li> </ul>
인프라 이벤트	밖으로	<ul style="list-style-type: none"> <li>해당 참조 대상(AAS, 하위 모델)의 소프트웨어 엔터티 부팅, 종료, 메모리 부족 ...</li> </ul>
리포지토리 이벤트	인/아웃	<ul style="list-style-type: none"> <li>IRD의 의미 변경(관련 개념 정의)</li> </ul>
보안 이벤트	밖으로	<ul style="list-style-type: none"> <li>이벤트 로깅</li> <li>액세스 위반, 적합하지 않은 역할 및 권한, 서비스 거부, ...</li> </ul>
알람 및 이벤트	밖으로	<ul style="list-style-type: none"> <li>분산 제어 시스템(DCS)과 유사한 경보 및 이벤트 관리</li> </ul>

#### 맞춤 이벤트 유형

어떤 경우든 이 이벤트 유형에 대해 독점적이지만 전 세계적으로 고유한 semanticId를 사용하여 사용자 정의 이벤트 유형을 정의할 수 있습니다. 이러한 맞춤형 이벤트는 임의의 조건, 트리거 또는 동작을 기반으로 해당 참조 대상의 소프트웨어 엔터티에서 보내거나 받을 수 있습니다. 그러나 이벤트 메시지의 일반 형식은 이 사양을 준수해야 하지만 페이로드는 완전히 사용자 정의될 수 있습니다.

#### 이벤트 범위

이벤트는 AAS, Submodels 및 SubmodelElements의 Referables 에 대한 observableReference 로 명시될 수 있습니다. 이러한 참조 대상 은 수신 또는 전송될 이벤트의 범위를 정의합니다.

이벤트 첨부...	범위
자산 관리 셸	이 이벤트는 AssetAdministrationShell, AssetInformation, Submodels 와 같은 관리 셸의 모든 논리적 요소를 모니터링/나타냅니다.
하위 모델	이 이벤트는 각 하위 모델 의 모든 논리적 요소 와 모든 논리적 종속을 모니터링/나타냅니다.

<sup>10</sup> 아래 참조

<sup>11</sup> 생성, 검색, 업데이트, 삭제

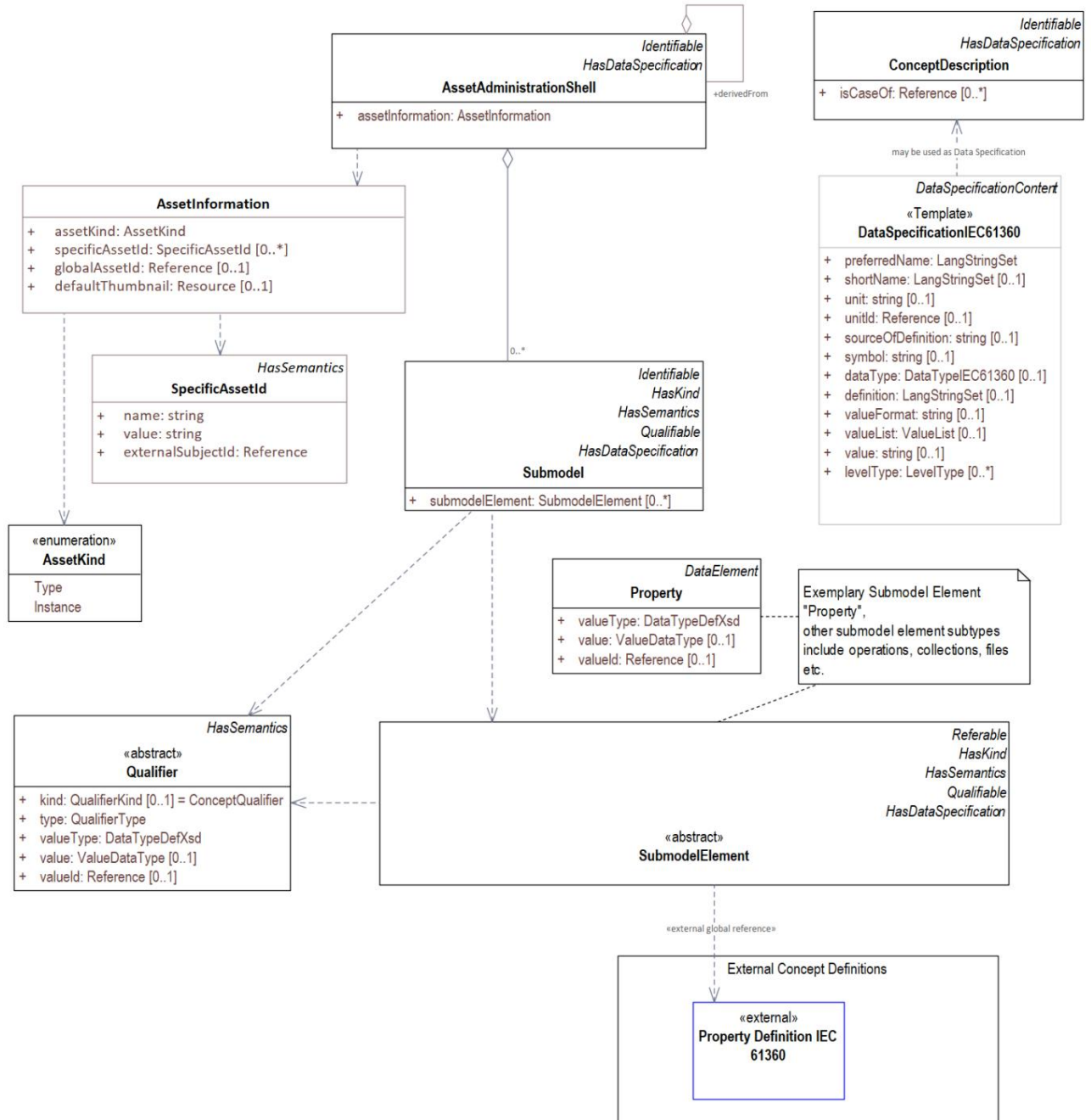


하위 모델 요소 목록 SubmodelElementCollection 및 실재	그리고 이 이벤트는 각 SubmodelElementCollection, SubmodelElementList 또는 Entity 의 모든 논리적 요소 와 모든 논리적 종속 항목(값 또는 명령문 관련)을 모니터링/나타냅니다.
SubmodelElement(가타)	이 이벤트는 단일 원자 SubmodelElement( 예: Blob 또는 파일 의 내용을 포함할 수 있는 데이터 요소 )를 모니터링/나타냅니다.

### 5.6 관리 셸의 개요 메타모델

이 절에서는 AAS(자산 관리 셸) 메타모델의 주요 개념에 대한 개요를 제공합니다(그림 12 참조).

그림 12 자산 관리 셸의 개요 메타 모델



AAS는 정확히 하나의 자산 (AssetAdministrationShell/assetInformation)을 나타냅니다. 자산 유형 및 자산 인스턴스는 "AssetInformation/assetkind" 속성을 설정하여 구별됩니다. 자세한 내용은 5.7.4절을 참조하십시오.

참고: UML 모델링은 "HasSemantics"와 같은 재사용된 개념을 나타내기 위해 소위 추상 클래스를 사용합니다.  
"적격" 등

자산 인스턴스의 AAS의 경우 해당 자산 유형 또는 파생된 다른 자산 인스턴스를 나타내는 AAS에 대한 참조가 추가될 수 있습니다 (AssetAdministrationShell/derivedFrom). 자산 유형의 AAS에 대해서도 마찬가지입니다. 또한 유형은 다른 유형에서 파생될 수 있습니다.

자산은 일반적으로 일련 번호, RFID 코드 등과 같은 여러 다른 식별 속성으로 표시될 수 있습니다. 이러한 외부 식별자는 사용자 정의 이름, 값 및 사용자 도메인(테넌트, 속성 기반 액세스 제어의 주제)

(AssetInformation/specificAssetId). 자세한 내용은 5.7.4절을 참조하십시오. 또한 생산 및 운영 단계에서 글로벌 자산 식별자를 자산 (AssetInformation/globalAssetId) 에 할당해야 합니다 .

AAS, 하위 모델 및 개념 설명은 전역적으로 고유하게 식별 가능해야 합니다 (식별 가능). 예를 들어 속성과 같은 다른 요소는 모델 내에서 참조 가능해야 하므로 로컬 식별자 (Referable 의 idShort) 만 필요합니다 . 식별에 대한 자세한 내용은 5.4절을 참조하십시오. 식별 가능 및 참조 가능 항목에 대한 자세한 내용은 5.7.2.2절 및 5.7.2.3 절을 참조하십시오.

하위 모델은 하위 모델 요소 집합으로 구성됩니다. 하위 모델 요소는 소위 한정자에 의해 한정될 수 있습니다. 자세한 내용은 5.7.2.7절 및 5.7.2.8절을 참조하십시오.

속성, 작업, 목록 등과 같은 하위 모델 요소의 다른 하위 유형이 있습니다. 자세한 내용은 5.7.7절을 참조하십시오. 일반적인 하위 모델 요소는 속성인 개요 그림에 표시됩니다. 속성은 문자열, 날짜 등과 같은 단순 유형의 값을 갖는 데이터 하위 모델 요소입니다. 속성에 대한 자세한 내용은 5.7.7.11절을 참조하십시오.

모든 하위 모델 요소는 잘 정의된 의미를 갖기 위해 의미 론적 정의 ( HasSemantics의 semanticId) 가 필요합니다. 하위 모델 요소는 외부 참조 (예: ECLASS 또는 IEC CDD 속성 정의)에서 제공하는 해당 의미론적 정의를 직접 참조하거나 개념 설명 (ConceptDescription)을 간접적으로 참조할 수 있습니다. 매칭 전략에 대해서는 5.4.6절을 참조하고, 자세한 사항은 5.7.2.6절을 참조하십시오.

개념 설명은 외부 표준의 다른 속성 정의 또는 다른 개념 설명 (ConceptDescription/isCaseOf)에서 파생될 수 있습니다. isCaseOf 는 단순한 텍스트 인 sourceOfDefinition 의 보다 공식적인 정의입니다 .

참고: 이 경우 대부분의 속성은 외부 표준에 정의되어 있으므로 중복됩니다. 이는 preferredName, unit 등과 같은 정보에 대한 속성을 추가하는 유용성에 관한 것입니다. 참조된 하위 모델 요소 정의에 대한 일관성은 해당 도구를 통해 보장되어야 합니다.

개념 설명이 외부 표준의 단순한 복사 또는 수정이 아닌 경우 이 개념 설명을 사용하는 AAS 제공자는 다른 AAS와의 상호 운용성이 보장될 수 없음을 인식해야 합니다.

데이터 사양 템플릿을 사용 하여 (DataSpecification) 요소에 대해 명명된 추가 속성(메타모델에 의해 사전 정의된 속성 외에) 집합을 정의할 수 있습니다. 속성의 개념 설명을 위해 일반적으로 IEC 61360을 따르는 데이터 사양 템플릿이 사용되며 예를 들어 "preferredName" 속성을 제공합니다. 사용할 권장 데이터 사양 템플릿을 나타내기 위해 <<템플릿>>-

종속성이 사용됩니다. 자세한 내용은 5.7.2.9절을 참조하십시오.

IEC 61360 속성 정의용 템플릿과 같은 데이터 사양 템플릿 (DataSpecificationIEC61360 및 DataSpecificationPhysicalUnit) 은 명시적으로 사전 정의되며 플랫폼에서 사용하도록 권장됩니다. 인터스트리 4.0. 자세한 내용은 6절을 참조하십시오. 독점 템플릿을 사용하는 경우 다른 AAS와의 상호 운용성을 보장할 수 없습니다.

속성 및 개념 설명을 포함한 하위 모델 요소 외에도 식별 가능한 다른 요소가 추가 템플릿 (HasDataSpecification)을 사용할 수 있습니다. 데이터 사양 템플릿은 디자인 타임에 선택됩니다. 자세한 내용은 5.7.2.9절을 참조하십시오.

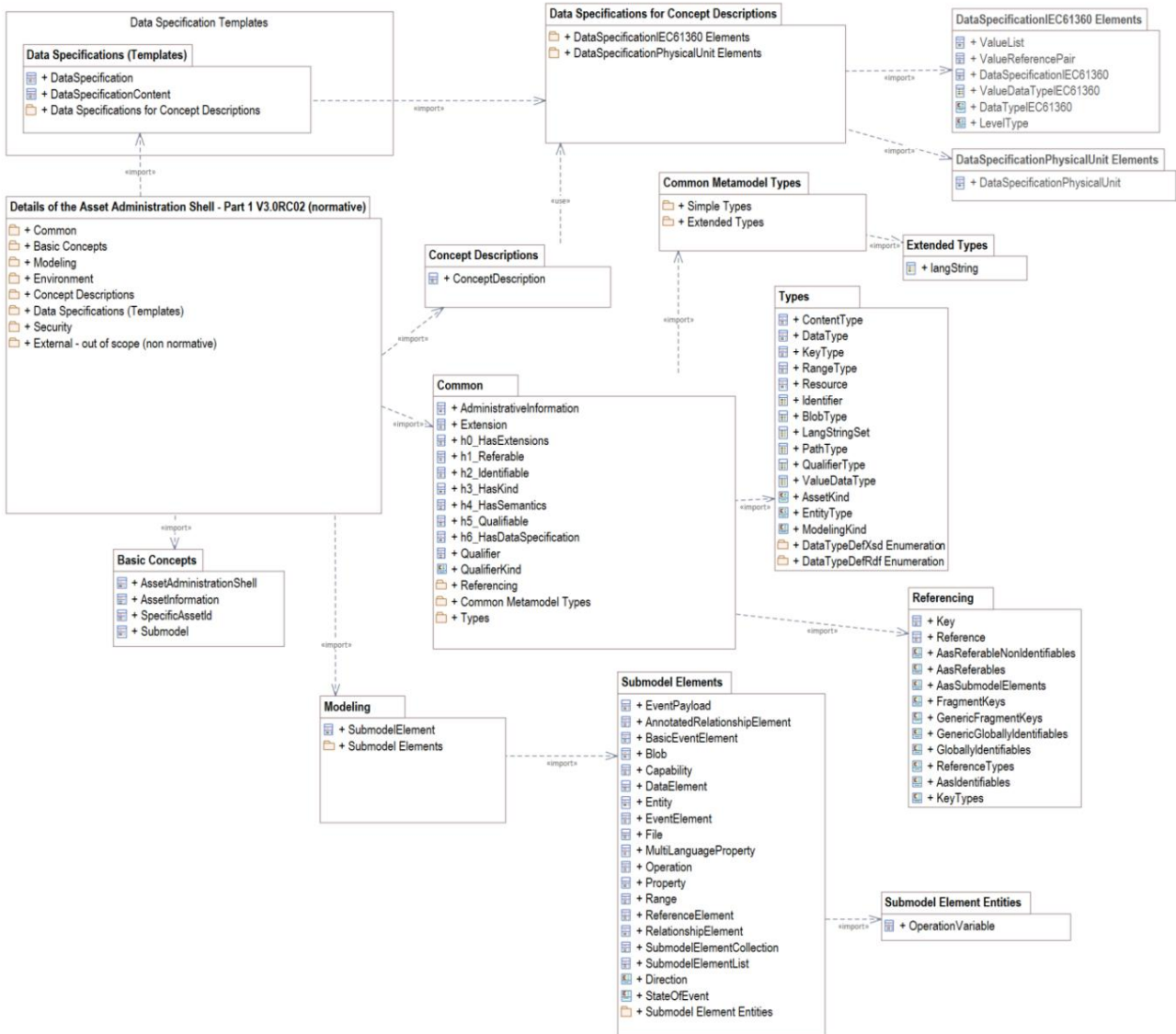
하위 모델 요소와 하위 모델 자체에는 추가 한정자가 있을 수 있습니다 (자격 가능). Qualifying 당 2 개 이상의 한정자가 있을 수 있습니다. 자세한 내용은 5.7.2.7절을 참조하십시오.

모든 AAS에 대해 보안 측면을 고려해야 합니다. 이 문서에서는 액세스 제어 측면에 대해 더 자세히 다룹니다. 인증된 특정 주체가 어떤 개체에 대해 어떤 권한이 있는지 정의하는 소위 액세스 권한 규칙이 정의됩니다. 자세한 내용은 7절을 참조하십시오.

그림 13은 보안을 제외하고 메타모델에 정의된 모든 요소에 대한 완전한 그림을 제공합니다. 보안 부분에 대한 정보는 7항에 있습니다.

참고: 추상 클래스에는 h0\_, h1\_ 등의 번호가 지정되지만(예: h1\_Referable) 이 접두어 없이 별칭이 정의됩니다. 이 이름을 지정하는 이유는 UML 모델링(Enterprise Architect)에 사용되는 도구에서 상속된 클래스에 대한 순서를 정의할 수 없으며 알파벳 순서로 정렬되기 때문입니다. 일부 직렬화의 경우 순서가 중요합니다(예: XML의 경우).

그림 13 메타모델 패키지 개요



## 5.7 메타모델 명세 세부사항: 지정자(규범)

### 5.7.1 소개

이 절에서는 메타모델의 클래스가 자세히 지정됩니다. 부록 B에는 클래스와 관계를 설명하는 데 사용되는 템플릿이 설명되어 있습니다. 부록 D에는 전체 개요를 제공하기 위해 일부 다이어그램이 상속된 모든 속성과 함께 표시됩니다.

사양을 이해하려면 먼저 공통 속성을 이해하는 것이 중요합니다(5.7.2절).

그것들은 다른 클래스의 사양 전체에서 재사용되고("상속") 식별 가능, 한정 가능 등과 같은 중요한 개념을 정의합니다. 그것들은 추상적입니다. 즉, 그러한 클래스의 객체 인스턴스가 없습니다.

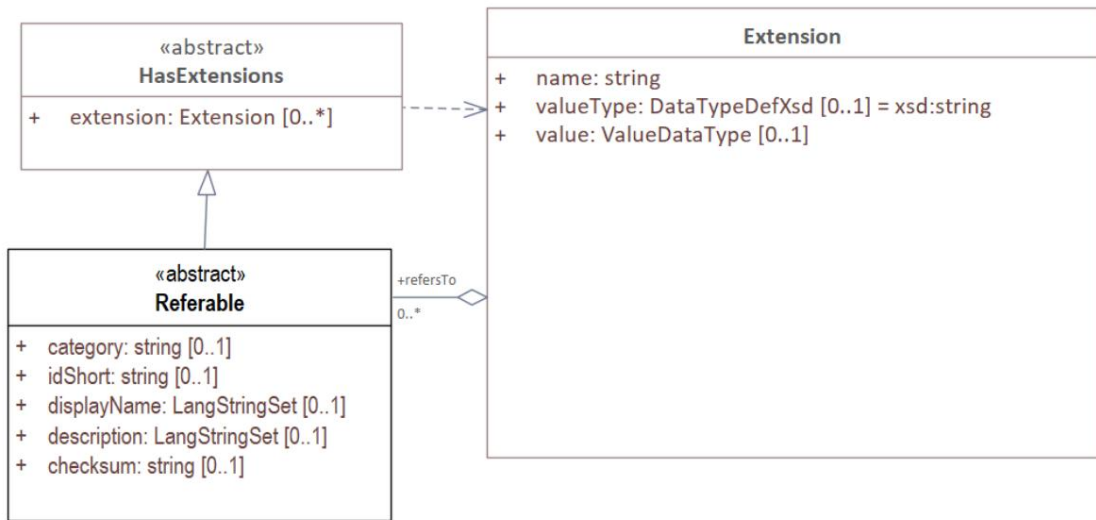
또 다른 중요한 개념은 참조의 개념과 UML 다이어그램 및 테이블에서 참조가 표현되는 방식입니다. 이것은 5.7.9절과 부록 D ii에 설명되어 있습니다.

클래스의 불변성이 아닌 제약 조건은 5.7.12.3절에 명시되어 있습니다.

## 5.7.2 공통 속성

### 5.7.2.1 확장(HasExtensions)

그림 14 HasExtensions의 메타모델



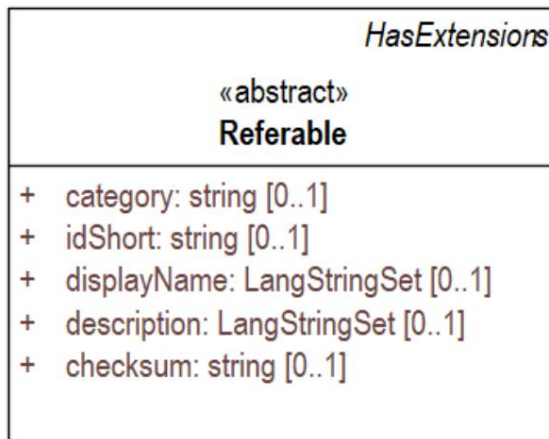
수업:	HasExtensions <<추상>>		
설명:	독점 확장으로 확장할 수 있는 요소입니다. 참고: 확장은 독점적입니다. 즉, 전역 상호 운용성을 지원하지 않습니다.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
확대	요소의 확장입니다.	확대	0..*

수업:	확대		
설명:	요소의 단일 확장입니다.		
다음에서 상속됨:	HasSemantics		
기인하다	설명	유형	카드.
이름	확장의 이름입니다.	끈	1

수업:	확대		
값 유형	확장 값의 유형입니다.  기본값: xs:문자열	DataTypeDefXsd	0..1
값	확장 값	값 데이터 유형	0..1
을 참고하여	요소에 대한 참조 확장자를 가리킵니다.	ModelReference<참조 가능>	0..*

### 5.7.2.2 참조 가능한 속성

그림 15 참조 가능한 메타 모델



메타 모델은 식별 가능하거나 참조 가능하거나 둘 다 없는 요소를 구별합니다.

참조 가능한 요소는 idShort를 통해 참조할 수 있습니다. 참조를 수행하는 방법에 대한 자세한 내용은 절을 참조하십시오. 5.7.9.

메타모델의 모든 요소를 참조할 수 있는 것은 아닙니다. 참조 가능한 속성일 뿐인 요소가 있습니다.

식별 불가능한 참조 대상의 경우 idShort는 이름 공간에서 고유해야 합니다(제약 AASd-022). 이 컨텍스트에서 이름 공간은 다음과 같이 정의됩니다. 요소가 일부이고 참조 가능하거나 식별 가능한 상위 요소는 요소의 이름 공간입니다. 예: 하위 모델은 여기에 포함된 속성의 이름 공간입니다. 하위 모델 요소 컬렉션에 포함된 하위 모델 요소의 이름 공간은 하위 모델 요소 컬렉션입니다.

수업:	참고할 수 있는 <<추상>>		
설명:	idShort에서 참조할 수 있는 요소입니다. 이 ID는 전역적으로 고유하지 않습니다. 이 ID는 요소의 이름 공간 내에서 고유합니다.		
상속:	HasExtensions		
기인하다	설명	유형	카드.
범주	카테고리는 요소의 클래스에 대한 추가 메타 정보를 제공하는 값입니다. 속성의 예상 존재와 제약 조건의 적용 가능성에 영향을 줍니다.  참고: 카테고리는 요소의 의미론적 정의 (HasSemantics)와 동일하지 않습니다. 카테고리 예를 들어 요소가 측정값임을 나타낼 수 있습니다.	끈	0..1

수업:	참고할 수 있는 <<추상>>		
	요소의 의미론적 정의는 측정된 온도임을 나타냅니다.		
아이디 쇼트	<p>식별 가능한 경우 이 속성은 요소. 참조 가능한 경우 이 ID는 해당 이름 공간 내 요소의 식별 문자열입니다.</p> <p><b>제약 조건 AASd -027: 참조 가능한 idShort의 최대 길이는 128자입니다.</b></p> <p>참고: 요소가 속성이고 속성에 IEC61360을 준수 하는 의미 체계 정의 (HasSemantics/semanticId)가 있는 경우 idShort는 일반적으로 영어의 짧은 이름과 동일합니다(사용 가능한 경우).</p>	כן	0..1
displayName 표시	<p>이름입니다. 여러 언어로 제공될 수 있습니다.</p> <p>애플리케이션에서 요청한 언어로 표시 이름이 정의되지 않은 경우 표시 이름은 사용 가능한 경우 다음 순서로 선택됩니다.</p> <ul style="list-style-type: none"> <li>- 요소의 의미를 정의하는 개념 설명의 요청된 언어에서 선호하는 이름</li> <li>- 응용 프로그램에 정의된 기본 언어 목록이 있는 경우 이 순서에 따라 해당 언어의 기본 언어 이름이 선택됩니다. - 요소의 의미를 정의하는 개념 설명의 영어 선호 이름</li> <li>- 개념 설명의 짧은 이름</li> <li>- 요소의 idShort</li> </ul>	랭스트링셋 0..1	
설명	<p>요소에 대한 설명 또는 설명입니다.</p> <p>설명에는 여러 언어로 제공될 수 있습니다.</p> <p>설명이 정의되지 않은 경우 요소의 의미를 정의하는 개념 설명의 정의가 사용됩니다.</p> <p>추가 정보가 제공될 수 있습니다. 예를 들어 요소가 자격이 있고 어떤 한정자 유형이 어떤 컨텍스트에서 또는 어떤 추가 데이터 사양 템플릿이 제공되는지 예상할 수 있는 경우입니다.</p>	랭스트링셋 0..1	
체크섬	<p>Referable(집계된 지식 요소 포함)이 변경되었는지 확인하는 데 사용할 체크섬입니다.</p> <p>체크섬은 사용자의 도구 환경에 의해 계산됩니다.</p> <p>체크섬은 Asset Administration Shell 모델에 대한 의미론적 의미가 없으며 체크섬을 관리하기 위한 Asset Administration Shell 도구에 대한 요구 사항도 없습니다.</p>	כן	0..1

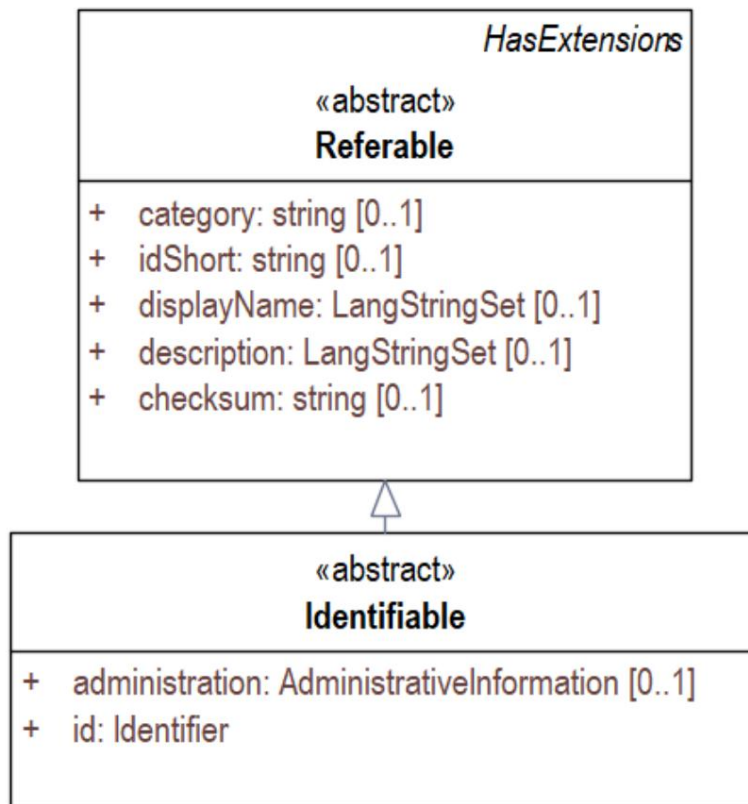
표 5에는 미리 정의된 범주가 설명되어 있습니다.

표 5 값이 있는 요소의 범주

범주:	적용 가능한 예	설명:
끊임없는	재산 참조요소	CONSTANT 범주의 요소는 시간이 지나도 변하지 않는 값을 가진 요소입니다.  ECLASS에서 이러한 종류의 범주에는 "코딩된 값" 범주가 있습니다.
매개변수 속성	MultiLanguage속성 범위 하위 모델 요소 컬렉션	PARAMETER 범주의 요소는 한 번 설정되고 일반적으로 시간이 지나도 변경되지 않는 요소입니다.  예를 들어 구성 매개변수의 경우입니다.
변하기 쉬운	재산 하위 모델 요소 목록	VARIABLE 범주의 요소는 런타임 중에 계산되는 요소입니다. 즉, 해당 값은 런타임 값입니다.

### 5.7.2.3 식별 가능한 속성

그림 16 식별 가능한 메타모델



식별 가능한 요소는 전역 고유 식별자 (식별자)로 참조할 수 있습니다. idShort 는 식별 가능 항목에 대해 고유하지 않기 때문에 식별 가능 항목을 참조하려면 글로벌 ID (식별 가능/id) 만 사용해야 합니다.

식별 정보에는 버전 등과 같은 관리 정보가 있을 수 있습니다.

식별할 수 없는 참조 가능한 요소는 참조할 수 있지만 그렇게 하려면 요소의 컨텍스트가 필요합니다. 식별할 수 없는 참조 가능한 항목에는 컨텍스트에서만 고유한 짧은 식별자 (idShort) 가 있습니다.

이름 공간.

식별에 대한 정보는 5.4절에서 확인할 수 있습니다. 5.4.4 절에서 어떤 유형의 식별자를 찾을 수 있는지에 대한 제약 및 권장 사항.

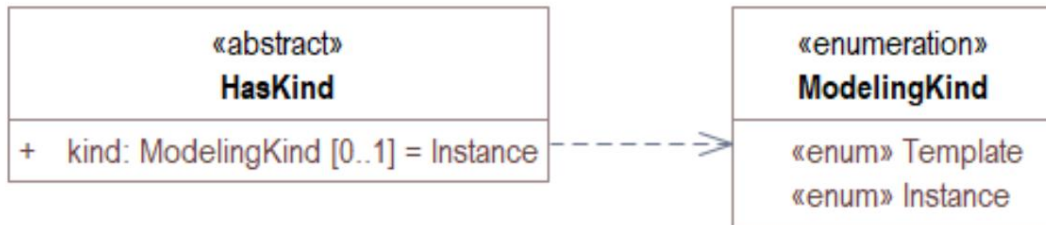
식별자의 예는 5.4.3절 자산 및 관리 셀에 대한 식별자에서 찾을 수 있습니다.

지원되는 식별자 유형에 대한 정보는 5.4.4절을 참조하십시오.

수업:	식별 가능한 <<abstract>>		
설명:	전역적으로 고유한 식별자가 있는 요소입니다.		
다음에서 상속됨:	참조 가능		
기인하다	설명	유형	카드.
관리 식별 가능한 관리	정보 요소.  참고: 버전 번호와 같은 일부 관리 정보는 식별의 일부가 되어야 할 수 있습니다.	관리정보 0..1	
ID	요소의 전역적으로 고유한 식별입니다.	식별자	1

### 5.7.2.4 모델 요소 속성의 템플릿 또는 인스턴스(HasKind)

그림 17 HasKind의 메타모델



수업:	HasKind		
설명:	종류가 있는 요소는 템플릿 또는 사례.  요소의 기본값은 인스턴스를 나타내는 것입니다.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
친절한	요소의 종류: 유형 또는 인스턴스.  기본값 = 인스턴스	모델링종류	0..1

종류 열거는 요소가 템플릿 또는 인스턴스 종류인지 여부를 나타내는 데 사용됩니다.

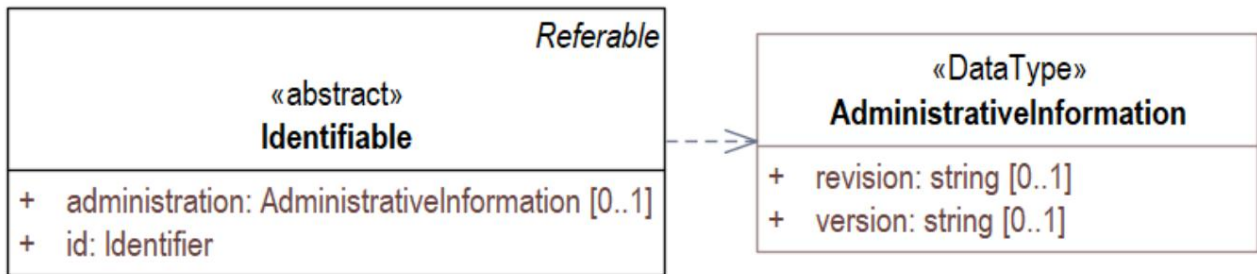
열거:	모델링종류
설명:	요소가 템플릿인지 인스턴스인지를 나타내는 열거형입니다.



열거:	모델링종류
세트:	--
정확한	설명
주형	<p>템플릿의 모든 인스턴스가 공유하는 공통 속성을 지정하는 소프트웨어 요소입니다.</p> <p>[출처: IEC TR 62390:2005-01, 3.1.25] 수정됨</p>
사례	<p>특정 템플릿의 구체적이고 명확하게 식별 가능한 구성 요소입니다.</p> <p>참고: 템플릿의 개별 엔터티가 됩니다(예: 특정 속성 값을 정의하여 장치 모델. 참고: 객체 지향 뷰에서 인스턴스는 객체를 나타냅니다. 템플릿(클래스).</p> <p>[출처: IEC 62890:2016, 3.1.16 65/617/CDV] 수정됨</p>

### 5.7.2.5 관리 정보 속성

그림 18 행정정보 메타모델



모든 식별 가능 에는 관리 정보가 있을 수 있습니다. 관리 정보에는 예를 들어

- 요소의 버전에 대한 정보 • 누가 요소를 만들었거나 마지막으로 변경했는지에 대한 정보
- 요소에 텍스트가 포함된 경우 번역할 수 있는 언어에 대한 정보 또한 마스터 또는 기본 언어를 정의할 수 있습니다.

AAS 메타모델의 첫 번째 버전에서는 관리 정보용으로 버전 정보만 정의됩니다. 이후 버전에서는 추가 속성이 추가될 수 있습니다.

버전은 원칙적으로 IEC 62832에 따른 version\_identifier 에 해당 하지만 개념 식별자(IEC TS 62832-1)에만 사용되는 것이 아니라 식별 가능한 모든 요소에 대해 사용됩니다. 버전 및 개정판은 함께 IEC 62832에 따른 버전 번호에 해당합니다.

AdministrativeInformation 은 템플릿 (HasDataSpecification) 의 사용을 허용 하지만 관리 정보에 대한 이 메타모델 버전에는 미리 정의된 템플릿이 없습니다.

참고: semanticId는 같지만 관리 정보가 다른 두 개의 하위 모델(특히 다른 버전), 다른 ID (Submodel/id)를 가져야 합니다. idShort 또는 숫자 식별자는 일반적으로 변경되지 않습니다. 즉, 동일합니다. 다른 식별 가능 항목 (식별 가능/id)도 마찬가지입니다.

참고: 버전이 다른 하위 모델은 식별자가 다르기 때문에 AAS는 semanticId는 같지만 버전이 다른 두 개의 하위 모델, 즉 다른 관리 메타정보를 가질 수 있습니다.

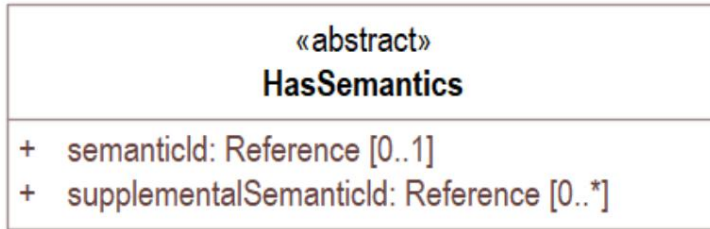
참고: 버전 번호와 같은 일부 관리 정보는 식별의 일부가 되어야 할 수 있습니다. 이것은 IRDI를 사용하여 개념 설명에 대한 식별자를 처리하는 것과 유사합니다. ECLASS에서 IRDI 0173 -1#02-AO677#002에는 버전 정보 #002가 포함되어 있습니다.

참고: 요소의 버전 관리 또는 기타 관리 정보 추가 프로세스는 외부 버전에서 수행됩니다.  
AAS 자체가 아닌 구성 관리 소프트웨어.

수업:	관리 정보 <<데이터 유형>>		
설명:	버전 정보와 같은 요소에 대한 관리 메타 정보입니다.  <u>제약 조건 AASd-005: AdministrativeInformation/version</u> 이 지정되지 않은 경우 행정정보/개정 은 불특정다. 즉, 개정판에는 버전이 필요합니다. 버전이 없으면 개정판도 없습니다. 수정은 선택 사항입니다.		
다음에서 상속됨:	HasData 사양		
기인하다	설명	유형	카드.
버전	요소의 버전입니다.	끈	0..1
개정	요소의 개정입니다.	끈	0..1

### 5.7.2.6 시맨틱 참조 속성(HasSemantics)

그림 19 의미론적 참조의 메타모델(HasSemantics)

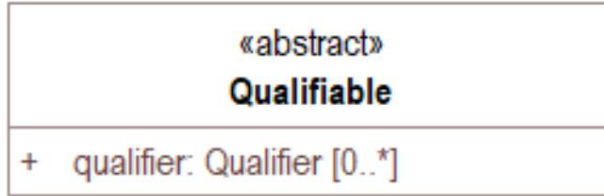


수업:	HasSemantics <<추상>>		
설명:	의미론적 정의와 추가 의미론적 정의를 가질 수 있는 요소.  <u>제약 AASd-118: 정의된 추가 의미 ID (HasSemantics/supplementalSemanticId) 가 있는 경우 기본 의미 ID (HasSemantics/semanticId)도 있어야 합니다.</u>		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
의미 ID	<p>의미론적 정의의 식별자 요소. 시맨틱 ID 또는 메인이라고도 합니다. 요소의 시맨틱 ID입니다.</p> <p style="background-color: #e0f0ff; padding: 2px;">전역 참조를 사용하는 것이 좋습니다.</p>	참조	0..1
SupplementalSemanticId 요소	<p>추가 의미론적 정의의 식별자입니다. 그것이라고</p> <p>요소의 보충 의미 ID입니다.</p>	참조	0..*

수업:	HasSemantics <<추상>>		
	전역 참조를 사용하는 것이 좋습니다.		

### 5.7.2.7 적격 속성

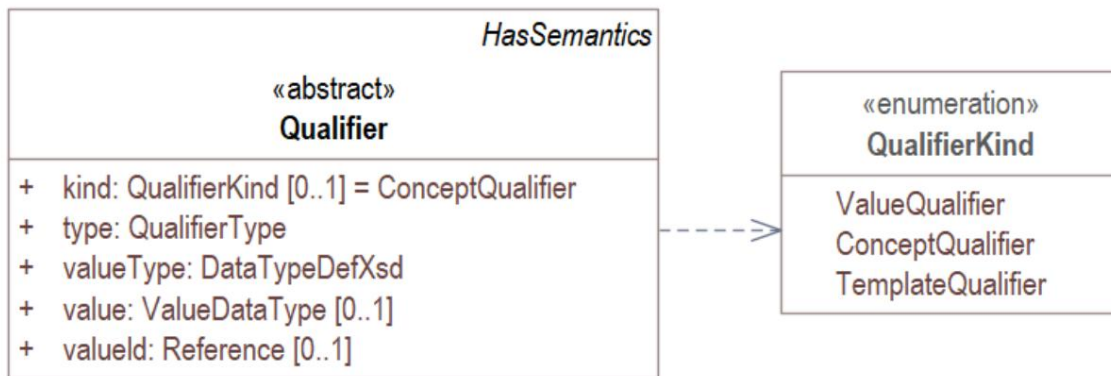
그림 20 자격의 메타모델



수업:	적격 <<abstract>>		
설명:	규정 가능한 요소의 값은 하나 이상의 규정자로 추가 규정될 수 있습니다.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
예선	적격 요소의 추가 적격성.	예선	0..*

### 5.7.2.8 한정자 속성

그림 21 한정자의 메타모델



한정 요소의 경우 추가 한정자가 정의될 수 있습니다. 다른 한정자 외에도 레벨 유형 최소값, 최대값, 일반 값 및 공칭 값을 정의하는 레벨 한정자는 IEC 62569-1에서 찾을 수 있습니다.

예를 들어 표현식 의미론 및 표현식 논리와 같은 추가 한정자 유형은 DIN SPEC 92000에 정의되어 있습니다.

수업:	예선
설명:	한정자는 요소 값을 작성하는 추가 명령문을 만드는 유형-값-쌍입니다.

수업:	예선		
	<p>제약 조건 AASd-006: 한정자 의 값 과 valueId 가 둘 다 있는 경우 값은 참조된 코딩된 값의 값과 동일해야 합니다.</p> <p>한정자/valueId.</p> <p>제약 AASd-020: 한정자/값의 값은 한정자/값 유형에 정의된 데이터 유형과 일치해야 합니다 .</p>		
상속:	HasSemantics		
기인하다	설명	유형	카드.
전절한	<p>한정자 종류는 요소에 적용되는 한정자의 종류를 설명합니다.</p> <p>기본값: ConceptQualifier</p>	한정자종류	0..1
유형	한정자 유형 은 요소에 적용되는 한정자의 유형을 설명합니다.	한정자 유형	1
값 유형	한정자 값의 데이터 유형입니다.	DataTypeDefXsd 1	
값	한정자 값은 한정자의 값입니다.	값 데이터 유형 0..1	
값 ID	<p>코딩된 값의 전역 고유 ID에 대한 참조입니다.</p> <p>전역 참조를 사용하는 것이 좋습니다.</p>	참조	0..1

한정자 에 대한 semanticId 를 추가하는 것이 좋습니다 .

열거:	한정자종류
설명:	한정자 종류에 대한 열거입니다.
세트:	--
정확한	설명
값 한정자	<p>요소의 값을 한정하고 런타임 중에 변경될 수 있습니다.</p> <p>값 한정자는 종류가 "인스턴스"인 요소에만 적용할 수 있습니다.</p>
ConceptQualifier	<p>요소가 참조하는 의미론적 정의를 한정합니다.</p> <p>(HasSemantics/semanticId)</p>
템플릿 한정자	<p>개념 수준에서 특정 하위 모델 내의 요소를 한정합니다.</p> <p>템플릿 한정자는 종류가 "템플릿"인 요소에만 적용할 수 있습니다.</p>

## 5.7.2.9 데이터 사양 속성에 사용되는 템플릿(HasDataSpecification)

그림 22 HasDataSpecification의 메타모델

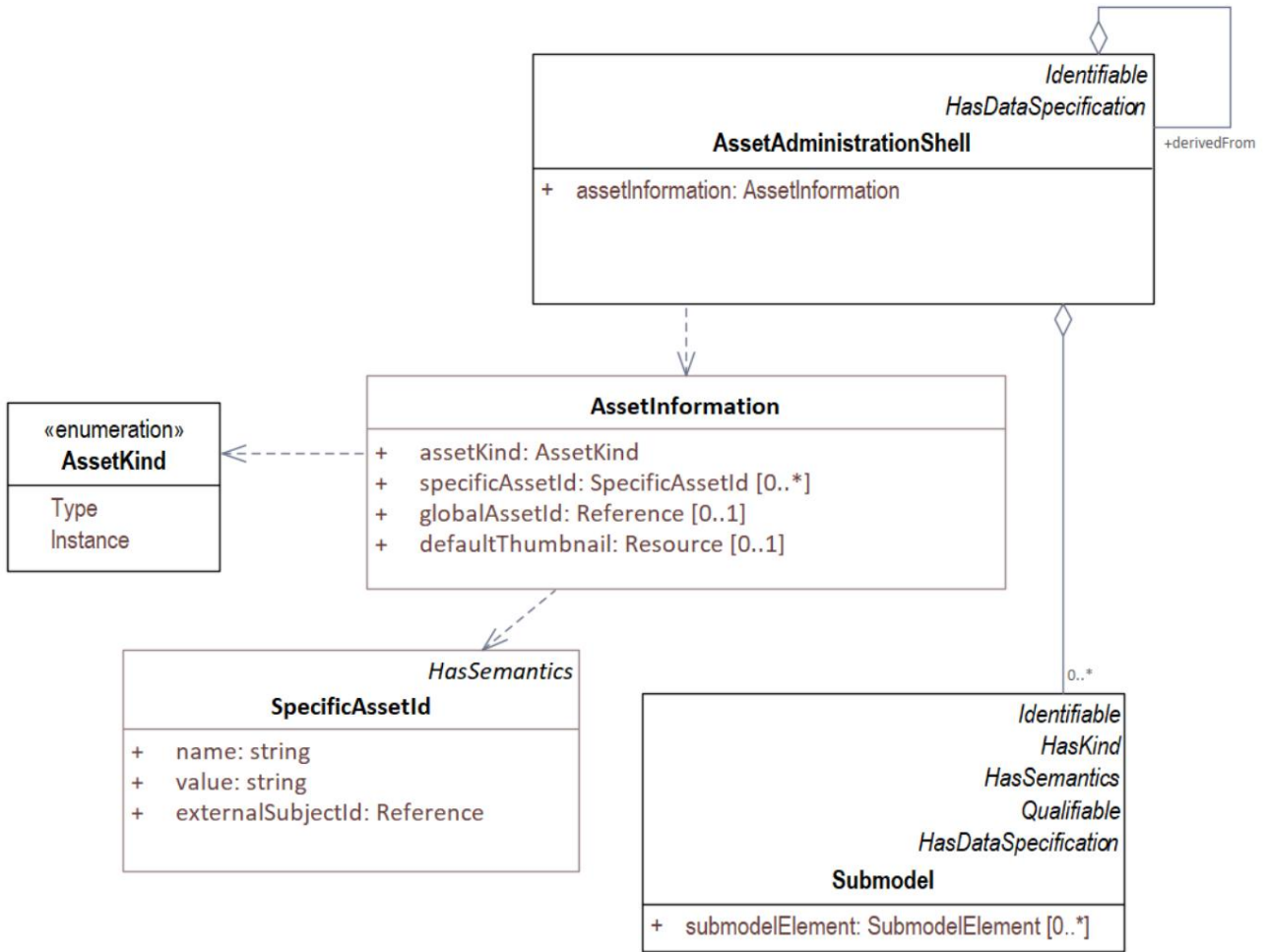


<b>수업:</b>	HasDataSpecification <<추상>>		
<b>설명:</b>	데이터 사양 템플릿을 사용하여 확장할 수 있는 요소입니다. 데이터 사양 템플릿은 요소가 가질 수 있거나 가질 수 있는 추가 속성의 명명된 집합을 정의합니다. 사용된 데이터 사양은 전역 ID로 명시적으로 지정됩니다.		
<b>다음에서 상속됨:</b>	--		
<b>기인하다</b>	<b>설명</b>	<b>유형</b>	<b>카드.</b>
데이터 사양	<p>데이터에 대한 전역 참조 요소에서 사용하는 사양 템플릿입니다.</p> <p style="background-color: #D9E1F2;">이것은 글로벌 참조입니다.</p>	참조	0..*

데이터 사양에 대한 자세한 내용은 6절을 참조하십시오.

### 5.7.3 자산 관리 셸 속성

그림 23 메타모델 AssetAdministrationShell



관리 셸은 식별 가능에서 상속되기 때문에 고유하게 식별 가능 합니다.

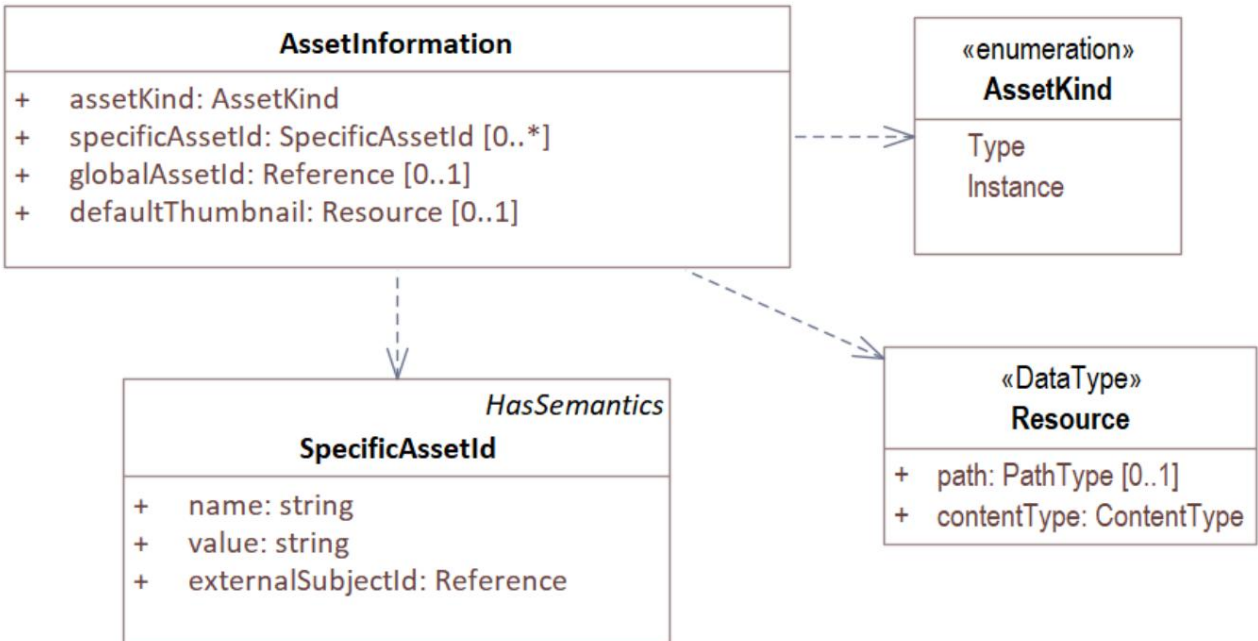
파생 된 From 특성은 서로 파생된 두 자산 관리 셸 간의 관계를 설정하는 데 사용됩니다. 파생된 From 개념에 대한 자세한 내용은 5.2절 유형 및 인스턴스를 참조하십시오.

수업:	자산 관리 셸		
설명:	자산 관리 셸.		
다음에서 상속됨:	식별 가능 HasData 사양		
기인하다	설명	유형	카드.
로부터 나온다	AAS에 대한 참조는 AAS에서 파생되었습니다.	ModelReference<AssetAdministrationShell>	0..1
assetInformation AAS에 대한 메타 정보는 유산 대표.	자산정보		1
하위 모델	의 하위 모델에 대한 참조 AAS.	ModelReference<하위 모델>	0..*

<b>수업:</b>	자산 관리 셀		
	<p>하위 모델은 AAS가 나타내는 자산의 측면에 대한 설명입니다.</p> <p>AAS의 자산은 일반적으로 하나 이상의 하위 모델로 설명됩니다.</p> <p>일시적으로 AAS에 하위 모델이 할당되지 않을 수 있습니다.</p>		

### 5.7.4 자산 정보 속성

그림 24 자산정보의 메타모델



<b>수업:</b>	자산정보		
<b>설명:</b>	<p>AssetInformation에는 AAS로 표현되는 자산의 메타 데이터를 식별하는 정보가 정의되어 있습니다.</p> <p>자산은 자산 유형 또는 자산 인스턴스를 나타낼 수 있습니다.</p> <p>자산에는 전역적으로 고유한 식별자와 필요한 경우 추가 도메인별(독점) 식별자가 있습니다. 그러나 안정화된/일정한 글로벌 자산 식별자가 아직 존재하지 않는 라이프사이클의 첫 번째 단계의 코너 케이스를 지원하기 위해 해당 속성 "globalAssetId"는 선택 사항입니다.</p>		
<b>다음에서 상속됨:</b>	--		
<b>기인하다</b>	<b>설명</b>	<b>유형</b>	<b>카드.</b>
자산종류	자산 이 "유형" 또는 "인스턴스" 유형 인지 나타냅니다.	자산 종류	1
글로벌 자산 ID	AAS가 나타내는 자산의 글로벌 식별자입니다. 참조		0..1

수업:	자산정보		
	<p>이 속성은 자산의 수명 주기에서 파트너를 통해 AAS를 교환하는 즉시 필요합니다.</p> <p>수명 주기의 첫 번째 단계에서 자산에는 아직 전역 ID가 없지만 이미 내부 식별자가 있을 수 있습니다. 내부 식별자는 다음을 통해 모델링됩니다.</p> <p>"특정 자산 ID".</p> <p>이것은 글로벌 참조입니다.</p>		
특정 자산 ID	추가 도메인 특정, 일반적으로 일련 번호 등과 같은 자산에 대한 독점 식별자입니다.	특정 자산 ID	0..*
defaultThumbnail 자산	이 나타내는 자산의 축소판 관리 셀. 기본값으로 사용됩니다.	자원	0..1

참고: 이 자산 정보 외에도 추가 정보가 포함된 식별 하위 모델이 있을 수 있습니다. 특정 자산 ID는 주로 자산에 대한 AAS 검색을 지원하는 목적으로 사용됩니다.

참고: specificAssetIds의 키는 고유할 필요가 없습니다.

참고: 단일 specificAssetIds에 대한 SemanticId는 고유할 필요가 없습니다.

수업:	리소스 <<데이터 유형>>		
설명:	리소스는 파일(로케이터)에 대한 주소를 나타냅니다. 같은 절대 또는 상대 경로를 나타낼 수 있는 URI입니다.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
길	<p>리소스의 경로 및 이름(파일 확장명 포함).</p> <p>경로는 절대적이거나 상대적일 수 있습니다.</p>	경로 유형	1
컨텐츠 타입	<p>파일 콘텐츠의 콘텐츠 유형입니다.</p> <p>콘텐츠 유형은 파일이 가질 수 있는 파일 확장자를 나타냅니다.</p>	컨텐츠 타입	0..1

열거:	자산 종류
설명:	자산이 유형 자산인지 인스턴스 자산인지를 나타내는 열거형입니다.
세트:	--
정확한	설명
유형	<p>유형의 모든 인스턴스가 공유하는 공통 속성을 지정하는 하드웨어 또는 소프트웨어 요소</p> <p>[출처: IEC TR 62390:2005-01, 3.1.25]</p>
사례	특정 유형의 구체적이고 명확하게 식별 가능한 구성 요소



<b>열거:</b>	자산 종류
	<p>참고 1: 특정 속성 값을 정의하여 장치와 같은 유형의 개별 개체가 됩니다.</p> <p>참고 2: 객체 지향 보기에서 인스턴스는 객체를 나타냅니다. (유형의) 클래스.</p> <p>[출처: IEC 62890:2016, 3.1.16] 65/617/CDV</p>

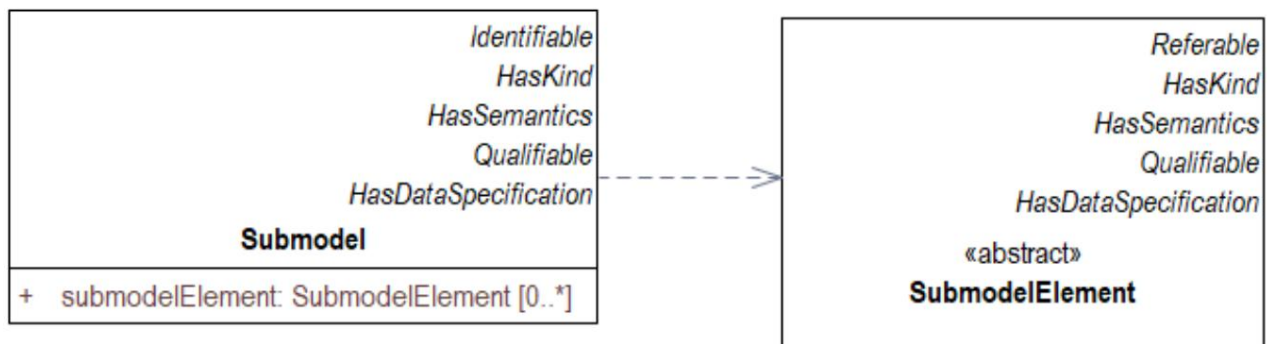
유형 및 인스턴스에 대한 자세한 내용은 5.2절을 참조하십시오.

<b>수업:</b>	특정 자산 ID		
<b>설명:</b>	특정 자산 ID는 자산의 일반 보조 식별 속성을 설명합니다. 특정 자산 ID가 반드시 전역적으로 고유한 것은 아닙니다.		
<b>다음에서 상속됨:</b>	HasSemantics		
<b>기인하다</b>	<b>설명</b>	<b>유형</b>	<b>카드.</b>
이름	식별자의 이름	끈	1
값	해당 이름이 있는 특정 자산 식별자의 값입니다.	끈	1
외부 주제 ID	(외부) 대상 특정 자산 ID에 속하거나 의미가 있습니다.  이것은 글로벌 참조입니다.	참조	1

주제 개념에 대한 자세한 내용은 ABAC(속성 기반 액세스 제어)에 대한 7절을 참조하십시오.

### 5.7.5하위 모델 속성

그림 25 서브모델의 메타모델



수업:	하위 모델		
설명:	하위 모델은 AAS가 나타내는 자산의 특정 측면을 정의합니다. 하위 모델은 관리 셸의 디지털 표현과 기술 기능을 구별 가능한 부분으로 구조화하는 데 사용됩니다. 각 하위 모델은 잘 정의된 영역 또는 주제를 나타냅니다. 하위 모델은 표준화되어 하위 모델 템플릿이 될 수 있습니다.		
다음에서 상속됨:	식별 가능 하스킨드; HasSemantics; 적격; HasData 사양		
기인하다	설명	유형	카드.
submodelElement 하위	모델은 0개 이상으로 구성됩니다. 하위 모델 요소.	하위 모델 요소 0..*	

하위 모델은 SubmodelElementCollection처럼 처리됩니다. 차이점은 식별 가능하고 미리 정의되어 있다는 것입니다.  
AAS 내의 정보 단위.

하위 모델에 대한 semanticId 를 추가하는 것이 좋습니다 .

하위 모델은 규정될 수 있습니다 (적격 가능).

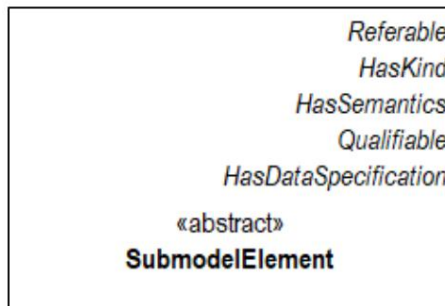
하위 모델 요소는 자격을 갖춘 요소입니다. 즉, 각각에 대해 하나 이상의 한정자가 정의될 수 있습니다.

하위 모델 및 하위 모델 요소에는 데이터 사양 템플릿이 정의되어 있을 수도 있습니다. 지금까지 하위 모델 및 하위 모델 요소에 대한 표준화된 데이터 사양 템플릿이 정의되어 있지 않습니다.

하위 모델이 종류=템플릿 ( HasKind 에 의해 상속된 모델링 종류 )인 경우 하위 모델 내의 하위 모델 요소는 하위 모델 요소 템플릿을 표시합니다. 하위 모델 이 종류=인스턴스 인 경우 하위 모델 요소는 하위 모델 요소 인스턴스를 나타냅니다.

### 5.7.6 하위 모델 요소 속성

그림 26 하위 모델 요소의 메타 모델



하위 모델 요소는 한정 가능한 요소입니다. 즉, 각각에 대해 하나 이상의 한정자가 정의될 수 있습니다.

하위 모델 요소에는 해당 요소에 대해 정의된 데이터 사양 템플릿이 있을 수도 있습니다. 예를 들어 사용 가능한 해당 개념 설명이 없는 경우 속성 개념 정의 단위 및 속성 개념 정의 단위와 같은 일부 속성을 미러링하도록 템플릿을 정의할 수 있습니다. 그렇지 않으면 속성 정의만 있습니다.

속성에 사용할 수 있는 semanticId 에 의해 참조됨 : 속성 조치는 다른 방식으로 온라인에서 실현되어야 하며 오프라인에서는 사용할 수 없습니다.

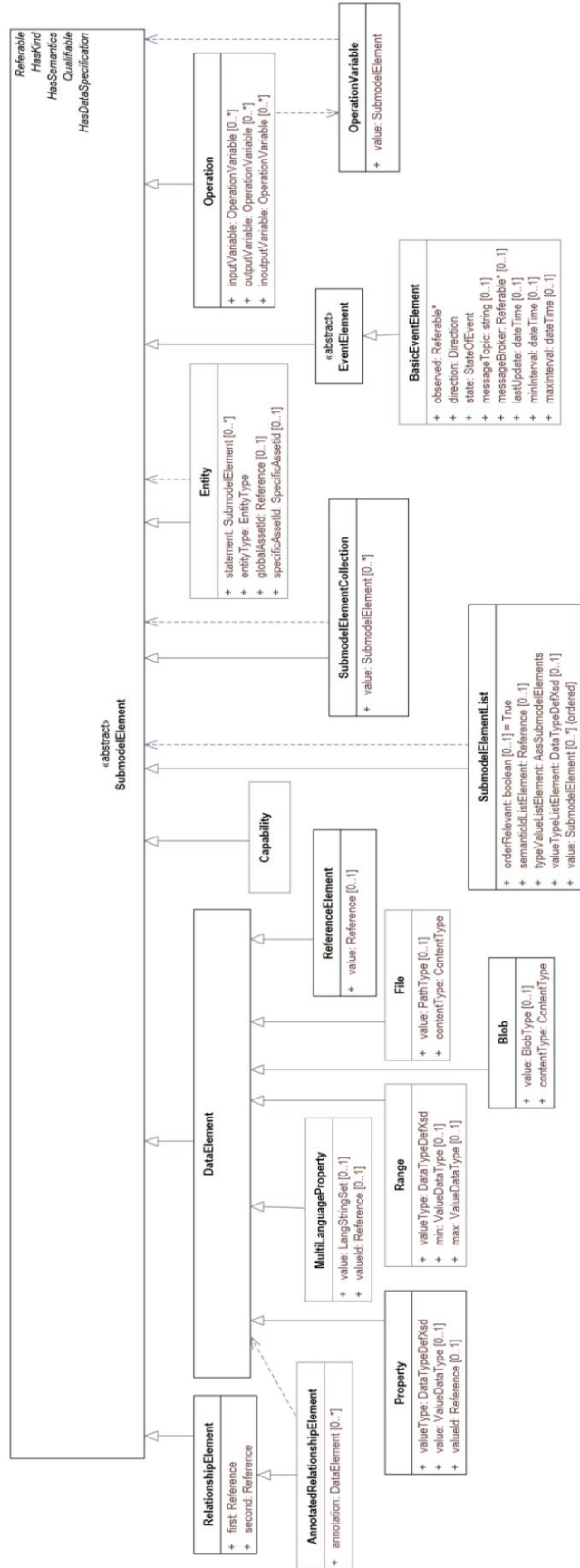
하위 모델이 종류=템플릿 인 경우 하위 모델 내의 하위 모델 요소가 표시됩니다.

하위 모델 요소 유형. 하위 모델이 종류=인스턴스 인 경우 하위 모델 요소는 하위 모델 요소 인스턴스를 나타냅니다.

수업:	하위 모델 요소 <<추상>>		
설명: <del>하위 모델 요소는</del>	설명 및 차별화에 적합한 요소입니다. 자산.  SemanticId 를 SubmodelElement 에 추가하는 것이 좋습니다 .		
다음에서 상속됨:	참조 가능; 하스킨드; HasSemantics; 적격; HasData 사양		
기인하다	설명	유형	카드.

### 5.7.7 하위 모델 요소 유형의 개요

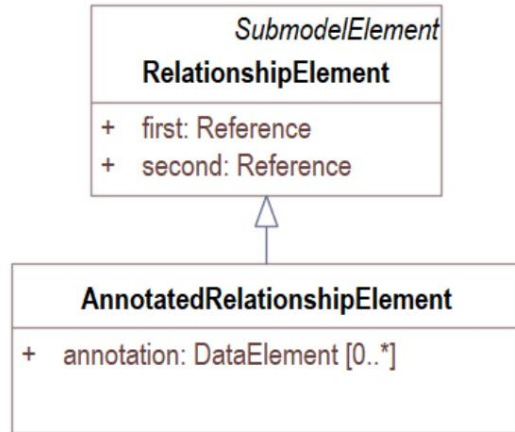
그림 27 하위 모델 요소 하위 유형에 대한 메타모델 개요



하위 모델 요소에는 자산에 대한 모델을 설명하는 데 필요한 데이터 속성과 작업, 이벤트 및 기타 요소가 포함됩니다(그림 27 참조).

### 5.7.7.1 주석이 달린 관계 요소 속성

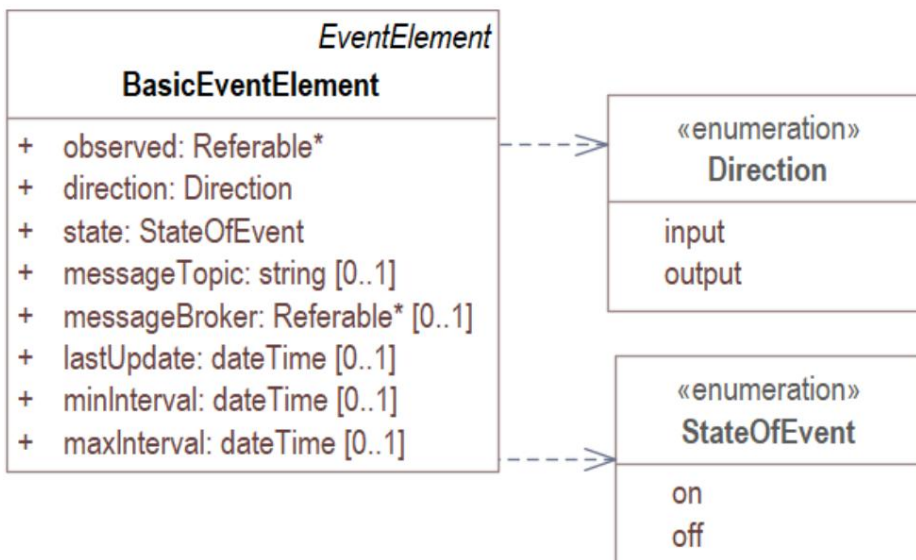
그림 28 주석이 달린 관계 요소의 메타모델



<b>수업:</b>	주석이 달린 관계 요소		
<b>설명:</b> 주석이 달린 관계	요소는 주석을 달 수 있는 관계 요소입니다. 추가 데이터 요소가 있습니다.		
<b>다음에서 상속됨:</b>	RelationshipElement		
<b>기인하다</b>	<b>설명</b>	<b>유형</b>	<b>카드.</b>
주석	두 요소 간의 관계를 유지하는 주석을 나타내는 데이터 요소입니다.	데이터 요소	0..*

### 5.7.7.2 기본 이벤트 요소 속성

그림 29 기본 이벤트 요소의 메타모델



수업:	기본 이벤트 요소		
설명:	기본 이벤트 요소입니다.		
다음에서 상속됨:	이벤트 요소		
기인하다	설명	유형	카드.
관찰	이벤트 범위를 정의하는 Referable에 대한 참조입니다. AAS, Submodel 또는 SubmodelElement일 수 있습니다.  관찰 중인 참조 가능 항목(예: 데이터 요소 또는 하위 모델)에 대한 참조입니다.	ModelReference<참조 가능> 1	
방향	행사의 방향.  { 입력, 출력 }일 수 있습니다.	방향	1
상태	이벤트 상태.  { 켜기, 끄기 }일 수 있습니다.	상태 이벤트	1
messageTopic 각 통신 채널에 이벤트를 스케줄링하기 위한 외부 메시지 기반 구조에 대한 정보입니다.		끈	0..1
messageBroker 정보, 외부 메시지 인프라는 EventElement에 대한 메시지를 처리해야 합니다. 하위 모델을 나타냅니다.  SubmodelElementList, SubmodelElementCollection 또는 Entity( 메시지에 대한 독점 사양을 설명하는 DataElements 포함 )  브로커.  참고: OPC UA 또는 MQTT 또는 AMQP와 같은 다양한 메시지 인프라의 경우 이 독점 사양은 각각의 하위 모델을 사용하여 표준화할 수 있습니다.		ModelReference<참조 가능> 0..1	
마지막 업데이트	마지막 이벤트가 수신(입력 방향)되거나 전송(출력 방향)된 시간 (UTC)의 타임스탬프입니다.	날짜 시간	0..1
최소 간격	입력 방향, 최대 빈도에 대한 보고의 경우 해당 Referable 뒤에 있는 소프트웨어 엔터티는 입력 이벤트를 처리할 수 있습니다.  출력 이벤트의 경우 이 이벤트를 외부 인프라로 출력하는 최대 빈도를 지정합니다.  지정되지 않을 수 있습니다. 즉, 최소 간격이 없습니다.	날짜 시간	0..1
최대 간격	입력 방향: 해당 없음.	날짜 시간	0..1

수업:	기본 이벤트 요소		
	출력 방향의 경우: 시간의 최대 간격에 대해 각각의 Referable은 이벤트에 대한 다른 트리거 조건이 충족되지 않은 경우에도 이벤트 상태의 업데이트를 전송해야 합니다.		
	지정되지 않을 수 있습니다. 즉, 최대 간격이 없습니다.		

열거:	방향
설명:	방향
세트:	--
정확한	설명
입력	방향을 입력합니다.
산출	출력 방향.

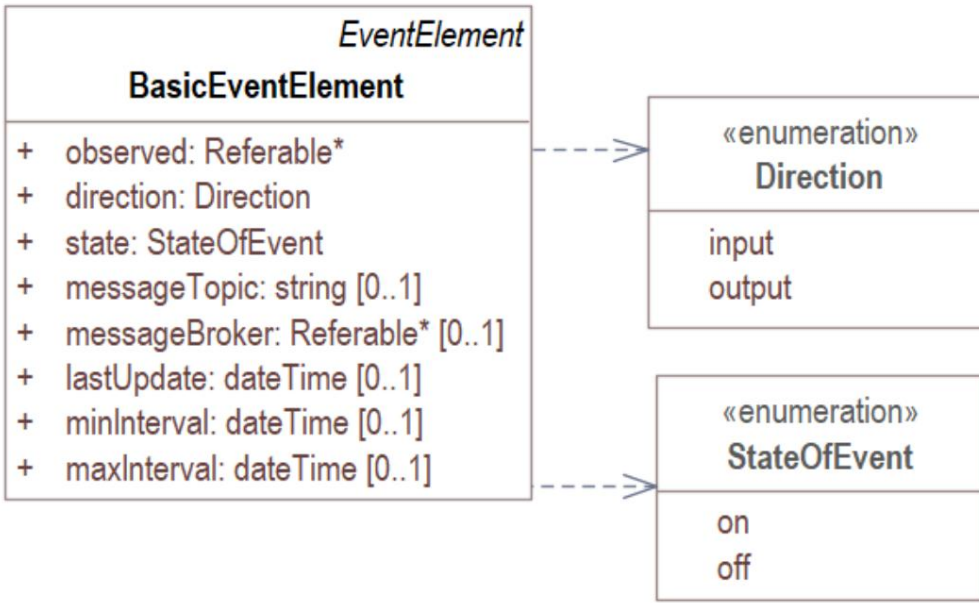
열거:	상태 이벤트
설명:	이벤트 상태
세트:	--
정확한	설명
~에	이벤트가 진행 중입니다.
끄다	이벤트가 꺼져 있습니다.

AAS에서 보내거나 받는 이벤트는 항상 일반 형식을 따릅니다. 예외: Industrie 4.0 상호 작용 패턴 과정에서 교환되는 이벤트.

다음에서는 이러한 이벤트의 페이로드가 지정됩니다.

참고: 페이로드는 AASX 패키지 형식을 통해 교환되는 정보 모델의 일부가 아니지만 활성 자산 관리 셀.

그림 30 메타모델 이벤트 페이로드



수업:	이벤트 페이로드		
설명:	보내거나 받은 이벤트 인스턴스의 필수 정보를 정의합니다.		
다음에서 상속됨:	-		
기인하다	설명	유형	카드.
원천	AAS, Submodel, SubmodelElements를 포함한 소스 이벤트 요소에 대한 참조.	ModelReference<참조 가능> 1	
sourceSemanticId	사용 가능한 경우 소스 이벤트 요소의 semanticId  전역 참조를 사용하는 것이 좋습니다.	참조	0..1
observableReference 이벤트 범위	범위를 정의하는 참조 대상에 대한 참조입니다.  AssetAdministrationShell, Submodel 또는 SubmodelElement일 수 있습니다.	ModelReference<참조 가능> 1	
observableSemanticId 참조할 수 있는 semanticId	사용 가능한 경우 이벤트 범위를 정의합니다.  전역 참조를 사용하는 것이 좋습니다.	참조	0..1

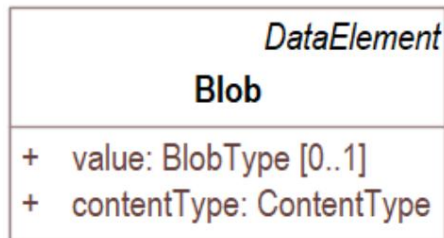


수업:	이벤트 페이로드		
주제	이벤트를 해당 통신 채널로 스케줄링하기 위한 외부 메시지 인프라에 대한 정보입니다.	끈	0..1
주제 ID	생성을 시작한 주제.  이것은 글로벌 참조입니다.	참조	0..1
타임스탬프	이 이벤트가 트리거된 UTC의 타임스탬프입니다.	날짜 시간 스탬프	1
유효 탑재량	이벤트별 페이로드.	끈	0..1

주제 개념에 대한 자세한 내용은 ABAC(속성 기반 액세스 제어)에 대한 7절을 참조하십시오.

### 5.7.7.3 얼룩 속성

그림 31 Blob의 메타모델

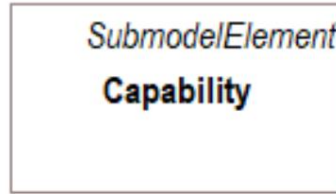


콘텐츠 유형에 대한 정보는 "파일" 하위 모델 요소에 대한 5.7.7.8절을 참조하십시오.

수업:	얼룩		
설명:	BLOB은 소스 코드와 함께 포함된 파일을 나타내는 데이터 요소입니다. 값 속성에서.		
다음에서 상속됨:	DataElement		
기인하다	설명	유형	카드.
값	Blob 데이터 요소의 BLOB 인스턴스 값입니다.  참고: 파일 속성과 달리 파일 콘텐츠는 Blob 데이터 요소에 값으로 직접 저장됩니다.	얼룩 유형	0..1
콘텐츠 타입	BLOB 콘텐츠의 콘텐츠 유형입니다.  콘텐츠 유형(MIME 유형)은 파일이 가질 수 있는 파일 확장자를 나타냅니다.  유효한 값은 "application/json", "application/xls", "image/jpg"와 같은 콘텐츠 유형입니다.  허용되는 값은 RFC2046과 같이 정의됩니다.	콘텐츠 유형 1	

### 5.7.7.4 능력 속성

그림 32 기능의 메타모델



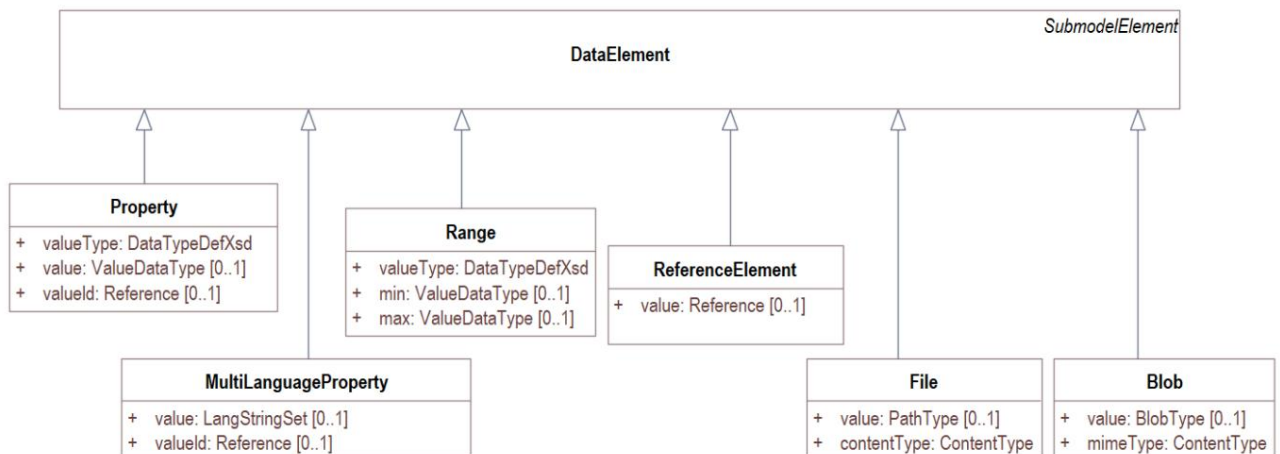
수업:	능력		
설명:	기능은 구현과 무관한 잠재력에 대한 설명입니다. 물리적 또는 가상 세계에서 특정 효과를 달성하기 위한 자산.		
상속:	SubmodelElement		
기인하다	설명	유형	카드.

참고: 기능의 semanticId는 일반적으로 온톨로지입니다. 따라서 능력에 대한 추론이 가능합니다.

기능 개념을 적용하는 방법과 기능을 구현하는 하나 이상의 기술에 매핑하는 방법에 대한 정보와 예는 [36]을 참조하십시오. 매핑은 해당 의미 체계가 있는 관계 요소를 통해 수행됩니다. 기술은 일반적으로 속성 또는 작업입니다. 더 복잡한 경우 매핑은 컬렉션 또는 완전한 하위 모델이 될 수도 있습니다.

### 5.7.7.5 데이터 요소 및 데이터 요소 유형의 개요

그림 33 데이터 요소의 메타모델



데이터 요소는 다른 하위 모델 요소로 더 이상 구성되지 않는 하위 모델 요소입니다.

데이터 요소는 범위 데이터 요소와 같이 값 또는 미리 정의된 수의 값이 있는 하위 모델 요소입니다.

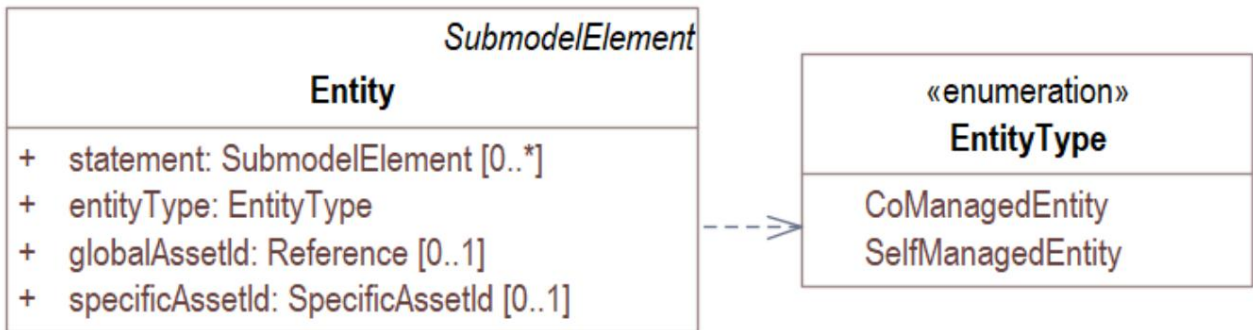
통제된 값은 의미가 외부 소스에서 주어진 값입니다("ISO/TS 29002-10:2009(E)" 참조).

값 유형은 데이터 요소의 하위 유형에 따라 다릅니다. 데이터 요소에는 속성, 파일 처리 및 참조 요소가 포함됩니다(그림 33 참조).

수업:	데이터 요소 <<추상>>		
설명:	<p>데이터 요소는 다른 요소로 더 이상 구성되지 않는 하위 모델 요소입니다. 하위 모델 요소.</p> <p>데이터 요소는 값이 있는 하위 모델 요소입니다. 값 유형은 데이터 요소의 하위 유형에 따라 다릅니다.</p> <p>제약 조건 AASd-090: 데이터 요소 범주 (Referable에 의해 상속됨) 는 CONSTANT, PARAMETER 또는 VARIABLE 값 중 하나여야 합니다. 기본값: 변수</p>		
상속:	SubmodelElement		
기인하다	설명	유형	카드.

### 5.7.7.6 엔티티 속성

그림 34 엔티티의 메타모델

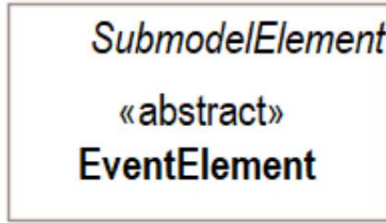


수업:	실재		
설명:	엔터티는 엔터티를 모델링하는 데 사용되는 하위 모델 요소입니다. 제약 조건 AASd-014: Entity/entityType 이 "SelfManagedEntity" 로 설정된 경우 Entity 의 속성 globalAssetId 또는 specificAssetId 가 설정되어야 합니다. 그렇지 않으면 존재하지 않습니다.		
다음에서 상속됨:	하위 모델 요소		
기인하다	설명	유형	카드.
성명	일반적으로 한정된 값을 사용하여 하위 모델 요소 집합으로 엔터티에 적용할 수 있는 문을 설명합니다.	하위 모델 요소 0..*	
엔터티 유형	엔터티가 공동 관리 엔터티인지 자체 관리 엔터티인지 설명합니다.	엔터티 유형	1
글로벌 자산 ID	엔터티가 나타내는 자산의 전역 식별자입니다.  이것은 글로벌 참조입니다.	참조	0..1
특정 자산 ID	자산 관리 셀이 나타내는 자산의 보조 식별자를 나타내는 특정 자산 ID에 대한 참조입니다.	특정 자산 ID	0..1

열거:	엔터티 유형
설명:	엔터티가 자체 관리 엔터티인지 공동 관리 엔터티인지를 나타내는 열거형입니다.
세트:	--
정확한	설명
공동 관리 엔터티	공동 관리 법인의 경우 별도의 AAS가 없습니다. 공동 관리 엔터티는 자체 관리 엔터티의 일부여야 합니다.
자체 관리 엔터티	자체 관리 엔터티에는 자체 AAS가 있지만 복합 자체 관리 엔터티의 BOM에 포함될 수 있습니다.  14.0 구성 요소의 자산은 정의에 따라 자체 관리되는 엔터티입니다.

5.7.7.7 이벤트 속성

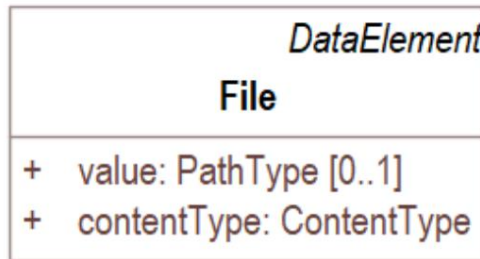
그림 35 이벤트의 메타모델



수업:	이벤트 요소 <<추상>>		
설명:	이벤트 요소입니다.		
다음에서 상속됨:	하위 모델 요소		
기인하다	설명	유형	카드.

5.7.7.8 파일 속성

그림 36 파일 하위 모델 요소의 메타 모델



미디어 유형(MIME 유형 및 콘텐츠 유형도 포함) [...]은 인터넷에서 전송되는 파일 형식 및 형식 콘텐츠에 대한 두 부분으로 된 식별자입니다. IANA(Internet Assigned Numbers Authority)는 이러한 분류의 표준화 및 출판을 위한 공식 기관입니다. 미디어 유형은 원래 Request for

1996년 11월의 RFC 2045는 MIME(Multipurpose Internet Mail Extensions) 사양의 일부로 전자 메일 메시지 콘텐츠 및 첨부 파일의 유형을 나타냅니다.<sup>12</sup>

수업:	파일		
설명:	파일은 파일(데이터)에 대한 주소를 나타내는 데이터 요소입니다. 값은 절대 또는 상대 경로를 나타낼 수 있는 URI입니다.		
다음에서 상속됨:	DataElement		
기인하다	설명	유형	카드.
값	파일의 경로 및 이름(파일 확장명 포함). 경로는 절대적이거나 상대적일 수 있습니다.	경로 유형	0..1
콘텐츠 타입	파일 콘텐츠의 콘텐츠 유형입니다.	콘텐츠 타입	1

<sup>12</sup> Wikipedia.org, 날짜: 2018-04-09

수업:	파일		
	콘텐츠 유형은 파일이 가질 수 있는 파일 확장자를 나타냅니다.		

AASX 패키지 형식으로 AAS 사양을 교환할 때 추가 외부 파일을 처리하려면 AASX(자산 관리 셸 패키지 파일 형식)에 대한 8.3 조항을 참조하십시오. 파일이 AAS와 독립적으로 존재하는 경우 절대 경로가 사용됩니다. 파일이 AAS의 직렬화된 패키지의 일부인 경우 패키지 루트에 상대적인 상대 경로를 사용해야 합니다.

### 5.7.7.9 다중 언어 속성 속성

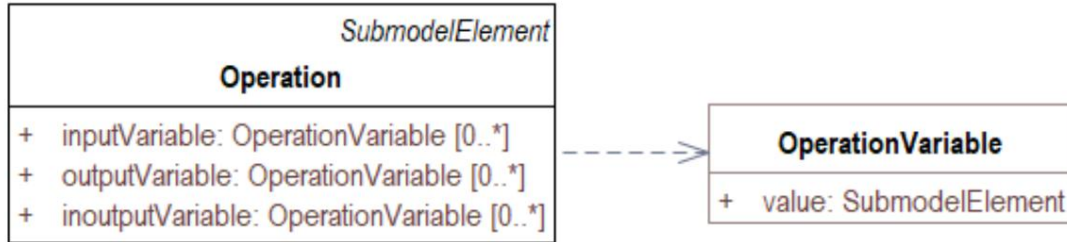
그림 37 다국어 속성의 메타모델



수업:	MultiLanguage속성		
설명:	속성은 다국어 값이 있는 데이터 요소입니다. <u>제한 조건 AASd-012: MultiLanguageProperty/value 및 MultiLanguageProperty/valueId 가 모두 있는 경우 특정 언어의 각 문자열에 대해 의미는 MultiLanguageProperty/valueId 에 지정된 것과 동일해야 합니다.</u>		
다음에서 상속됨:	DataElement		
기인하다	설명	유형	카드.
값	속성 인스턴스의 값입니다.	랭스트리셋	0..1
값 ID	코딩된 값의 전역 고유 ID에 대한 참조입니다.  전역 참조를 사용하는 것이 좋습니다.	참조	0..1

### 5.7.7.10 작업 속성

그림 38 작업의 메타모델



수업:	작업		
설명:	작업은 입력 및 출력 변수가 있는 하위 모델 요소입니다.		
다음에서 상속됨:	하위 모델 요소		
기인하다	설명	유형	카드.
입력변수	작업의 입력 매개변수입니다.	조작변수	0..*
outputVariable	작업의 출력 매개변수입니다.	조작변수	0..*
inoutVariable	작업의 입력 및 출력 매개변수	조작변수	0..*

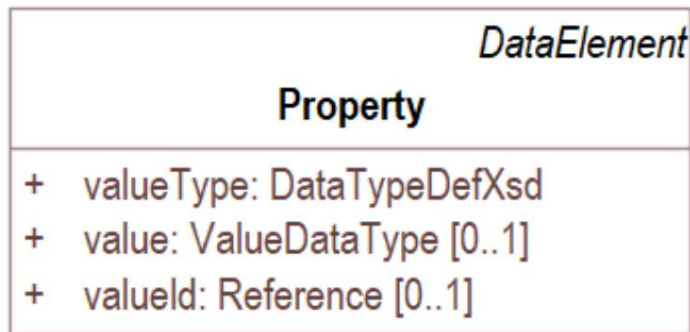
수업:	조작변수		
설명:	연산 변수의 값은 입력으로 사용되는 하위 모델 요소입니다. 및/또는 연산의 출력 변수.		
다음에서 상속됨:			
기인하다	설명	유형	카드.
값	하위 모델 요소를 통해 작업의 인수 또는 결과를 설명합니다.	하위 모델 요소	1

참고: 작업은 일반적으로 절차 측면에서 구성 요소의 동작을 지정합니다. 따라서 작업은 절차 기반 상호 작용을 통해 서비스 사양을 가능하게 합니다[32].

참고: OperationVariable은 기본 값인 카디널리티(옵션/필수)를 추가하기 위해 향후 확장을 활성화하기 위해 별도의 클래스로 도입되었습니다.

## 5.7.7.11 속성 속성

그림 39 속성의 메타모델

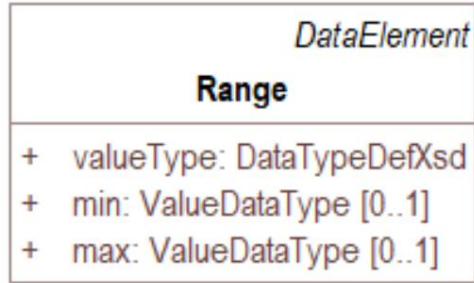


<b>수업:</b>	재산		
<b>설명:</b> 속성은 단일 값을 갖는 데이터 요소입니다.	<p>제약 조건 AASd-007: 속성/값 및 속성/valueId 가 모두 있는 경우 속성/값의 값은 속성/값 Id 에서 참조된 코딩된 값의 값과 동일해야 합니다.</p>		
<b>다음에서 상속됨:</b>	DataElement		
<b>기인하다</b>	<b>설명</b>	<b>유형</b>	<b>카드.</b>
값 유형	값의 데이터 유형	DataTypeDefXsd	1
값	속성 인스턴스의 값입니다.	값 데이터 유형	0..1
값 ID	코딩된 값의 전역 고유 ID에 대한 참조입니다.	참조	0..1
	전역 참조를 사용하는 것이 좋습니다.		



5.7.7.12 범위 속성

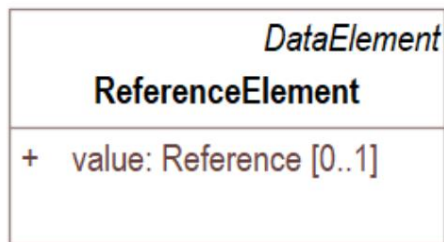
그림 40 범위의 메타모델



수업:	범위		
설명:	범위 데이터 요소는 최소 및 최대 범위를 정의하는 데이터 요소입니다.		
다음에서 상속됨:	DataElement		
기인하다	설명	유형	카드.
값 유형	최소 및 최대 데이터 유형	DataTypeDefXsd	1
분	범위의 최소값입니다. 최소값이 누락된 경우 값은 음의 무한대로 간주됩니다.	값 데이터 유형	0..1
최대	범위의 최대값입니다. 최대값이 누락된 경우 값은 양의 무한대로 간주됩니다.	값 데이터 유형	0..1

5.7.7.13 참조 요소 속성

그림 41 참조 요소의 메타모델



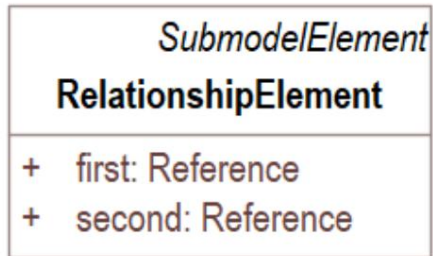
수업:	참조요소		
설명:	참조 요소는 동일하거나 다른 AAS 내의 다른 요소에 대한 논리적 참조 또는 외부 개체 또는 엔티티에 대한 참조를 정의하는 데이터 요소입니다.		
다음에서 상속됨:	DataElement		
기인하다	설명	유형	카드.
값	외부 개체 또는 엔티티에 대한 전역 참조 또는 동일한 내의 다른 요소에 대한 논리적 참조	참조 0..1	

수업:	참조요소		
	또는 다른 AAS(즉, 참조 대상에 대한 모델 참조).		

참조에 대한 자세한 내용은 5.7.9절을 참조하십시오.

#### 5.7.7.14 관계 요소 속성

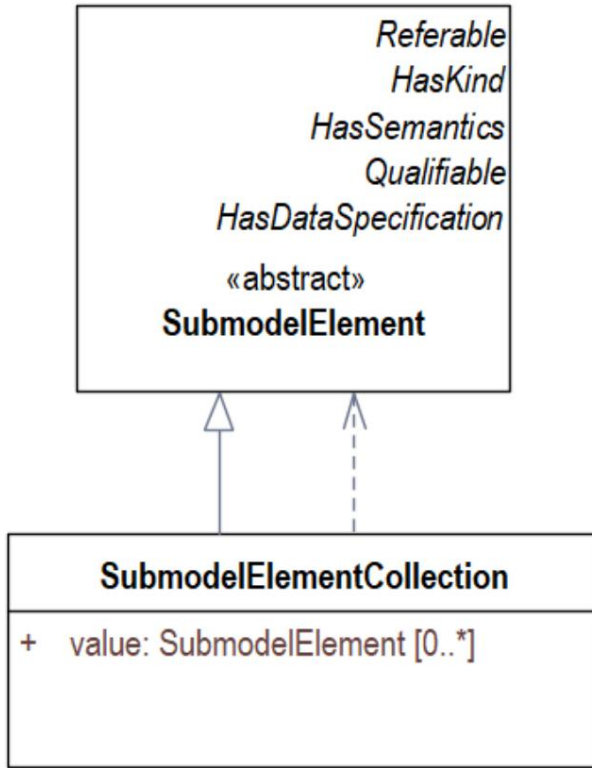
그림 42 관계 요소의 메타모델



수업:	관계요소		
설명:	관계 요소는 두 요소 간의 관계를 정의하는 데 사용됩니다. 참조 가능(모델 참조) 또는 외부(글로벌 참조).		
상속 에서:	하위 모델 요소		
기인하다	설명	유형	카드.
첫 번째	주체의 역할을 하는 관계의 첫 번째 요소에 대한 참조입니다.	참조	1
초	개체의 역할을 하는 관계의 두 번째 요소에 대한 참조입니다.	참조	1

5.7.7.15 하위 모델 요소 컬렉션 속성

그림 43 하위 모델 요소 컬렉션의 메타 모델



하위 모델 요소 컬렉션은 구조체 내에서 고유한 이름을 가진 고정 속성 집합이 있는 엔터티에 사용됩니다. 컬렉션 내의 각 속성에는 명확하게 정의된 의미 체계가 있어야 합니다. 구조체의 속성은 값이 있는 모든 하위 모델 요소가 될 수 있습니다.

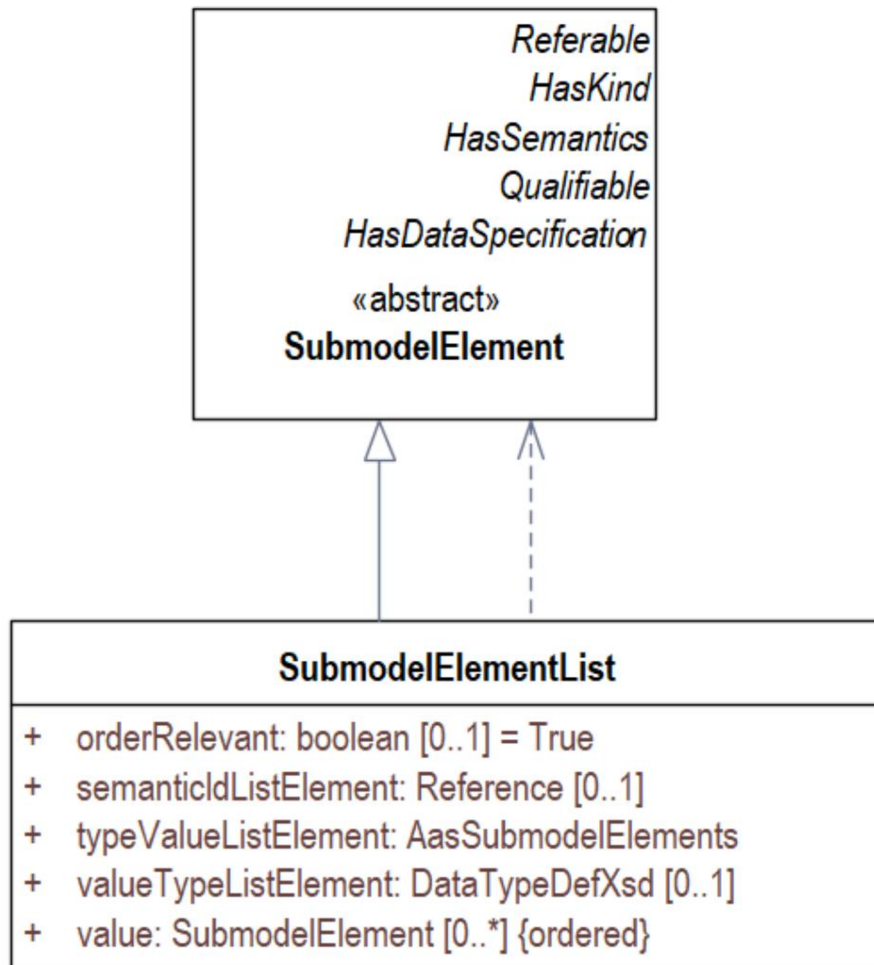
하위 모델 요소 구조체의 다른 요소가 다른 semanticId를 가질 필요는 없습니다. 그러나 이러한 경우 SubmodelElementList의 사용을 고려해야 합니다.

예: 자산의 경우 일반적으로 작동 지침, 안전 지침 등과 같은 다양한 문서를 사용할 수 있습니다. 각 단일 문서에는 제목, 버전, 작성자 등과 같은 사전 정의된 속성 집합이 있습니다. 이들은 논리적으로 문서에 속합니다. 따라서 단일 문서는 SubmodelElementCollection으로 표시됩니다. 모든 문서 세트는 SubmodelElementList로 표시됩니다. 따라서 이 예제에는 SubmodelElementCollections의 SubmodelElementList가 있습니다.

<b>수업:</b>	하위 모델 요소 컬렉션		
<b>설명:</b>	컬렉션은 일종의 구조체입니다. 즉, 여러 요소의 논리적 캡슐화입니다. 명명된 값. 고정된 수의 하위 모델 요소가 있습니다.		
<b>상속:</b>	SubmodelElement		
<b>기인하다</b>	<b>설명</b>	<b>유형</b>	<b>카드.</b>
값	컬렉션에 포함된 하위 모델 요소입니다. 하위 모델 요소		0..*

## 5.7.7.16 하위 모델 요소 목록 속성

그림 44 하위 모델 요소 목록의 메타 모델



하위 모델 요소 목록은 집합(예: 중복이 없는 정렬되지 않은 컬렉션), 정렬된 목록(예: 중복을 포함할 수 있는 정렬된 컬렉션), 가방(예: 중복을 포함할 수 있는 정렬되지 않은 컬렉션) 및 정렬된 집합(예: 중복이 없는 정렬된 컬렉션)에 사용됩니다. 하위 모델 요소의 정렬된 컬렉션을 통해 실현됩니다.

하위 모델 요소 목록은 다차원 배열을 만드는 데도 사용됩니다. 속성 값이 있는 2차원 배열 목록[3][5]의 경우 다음과 같이 실현됩니다. 첫 번째 하위 모델 요소 목록에는 3개의 SubmodelElementList 요소가 포함됩니다. 이러한 3개의 SubmodelElementList 각각에는 5개의 단일 Property 요소가 포함됩니다. 포함된 속성의 semanticId는 첫 번째 목록의 모든 목록에 대해 동일합니다. 즉, semanticIdListElement는 첫 번째 목록에 포함된 세 목록 모두에 대해 동일합니다. 세 개의 semanticId

포함된 목록은 해당 목록이 나타내는 차원에 따라 다릅니다. 배열에 복잡한 값이 있는 경우 SubmodelElementCollection이 리프 목록의 값으로 사용됩니다.

마찬가지로 3개의 열이 있는 테이블을 나타낼 수 있습니다. 이 경우 3개의 SubmodelElementList가 있는 SubmodelElementCollection이 포함됩니다. 이 경우 semanticId 및 semanticIdListElement 세 개의 열이 다르기 때문입니다.

의미론적 ID에 대한 5.4.6절 일치 전략을 참조하십시오.

수업:	하위 모델 요소 목록		
설명:	<p>하위 모델 요소 목록은 하위 모델 요소의 정렬된 목록입니다.</p> <p>번호 매기기는 영(0)으로 시작합니다.</p> <p><u>제약 조건 AASd-107:</u> SubmodelElementList 는 semanticId와 동일해야 합니다. SubmodelElementList/semanticIdListElement.</p> <p><u>제약 조건 AASd-114:</u> 두 개의 첫 번째 수준 자식 요소가 SubmodelElementList 에는 semanticId 가 있으면 동일해야 합니다.</p> <p><u>제약 조건 AASd-115:</u> SubmodelElementList의 첫 번째 수준 자식 요소 가 semanticId 를 지정하지 않으면 값이 SubmodelElementList /semanticIdListElement 와 동일한 것으로 간주됩니다 .</p> <p><u>제약 조건 AASd-108:</u> a의 모든 첫 번째 수준 자식 요소 SubmodelElementList 는 SubmodelElementList/typeValueListElement 에 지정 된 것과 동일한 하위 모델 요소 유형을 가져야 합니다.</p> <p><u>제약 조건 AASd-109:</u> SubmodelElementList/typeValueListElement인 경우 속성 또는 범위와 같은 SubmodelElementList/valueTypeListElement SubmodelElementList 의 모든 첫 번째 수준 하위 요소가 설정되어야 합니다. SubmodelElementList/valueTypeListElement 에 지정된 값 유형을 가져야 합니다 .</p>		
다음에서 상속됨:	하위 모델 요소		
기인하다	설명	유형	카드.
주문 관련	<p>목록의 순서가 관련이 있는지 여부를 정의합니다. 만약에 orderRelevant = False인 경우 목록은 다음과 같습니다. 세트 또는 가방을 나타냅니다.</p> <p>기본값: 참</p>	부울	0..1
값	목록에 포함된 하위 모델 요소입니다. 하위 모델 요소 0..*		
semanticIdListElement	<p>시맨틱 ID 하위 모델 요소 일치하는 목록에 포함되어 있습니다.</p> <p>전역 참조를 사용하는 것이 좋습니다.</p>	참조	0..1
유형 값 목록 요소	목록에 포함된 하위 모델 요소의 하위 모델 요소 유형입니다.	하위 모델요소E 요소	1
값 유형 목록 요소	목록에 포함된 하위 모델 요소의 값 유형입니다.	DataTypeDefXsd	0..1

## 5.7.8 개념 설명 속성

그림 45 개념 설명의 메타모델



<b>수업:</b>	개념설명		
<b>설명:</b>	<p>의미 설명을 가질 수 있는 속성 또는 기타 요소의 의미는 개념 설명에 의해 정의됩니다.</p> <p>개념에 대한 설명은 표준화된 스키마(데이터 사양 템플릿으로 실현)를 따라야 합니다.</p> <p><b>제약 AASd-051:</b> ConceptDescription 은 VALUE, PROPERTY, REFERENCE, DOCUMENT, CAPABILITY, RELATIONSHIP, COLLECTION, FUNCTION, EVENT, ENTITY, APPLICATION_CLASS, QUALIFIER, VIEW 범주 중 하나를 가져야 합니다. 기본값: 속성.</p>		
<b>다음에서 상속됨:</b>	식별 가능 HasData 사양		
<b>기인하다</b>	<b>설명</b>	<b>유형</b>	<b>카드.</b>
isCaseOf	<p>외부 정의에 대한 참조는 개념과 호환되거나 파생되었습니다.</p> <p style="background-color: #D9E1F2; padding: 5px;">전역 참조를 사용하는 것이 좋습니다.</p> <p style="background-color: #D9E1F2; padding: 5px;">참고: ISO 13584-32 &amp;의 is-case-of 관계와 비교 IEC EN 61360</p>	참조	0..*

다른 유형의 하위 모델 요소에는 의미 체계를 설명하기 위해 다른 속성이 필요합니다. 이것이 개념 설명에 연관된 데이터 사양 템플릿이 하나 이상 있는 이유입니다. 이 템플릿 내에서 의미 체계를 정의하는 데 필요한 속성이 정의됩니다.

사용할 사전 정의된 데이터 사양 템플릿에 대해서는 5.7.12.3절을 참조하십시오.

## 5.7.9 환경

그림 46 환경에 대한 메타모델

Environment	
+	assetAdministrationShell: AssetAdministrationShell [0..*]
+	submodel: Submodel [0..*]
+	conceptDescription: ConceptDescription [0..*]

참고 wrt 파일 교환: 포함된 요소가 분할되는 파일 수에 독립적인 환경은 정확히 하나입니다. 파일이 분할되면 두 개의 다른 파일에 동일한 식별자를 가진 요소가 없어야 합니다.

수업:	환경		
설명:	서로 다른 식별 항목 집합을 위한 컨테이너입니다.		
다음에서 상속됨:	참조		
기인하다	설명	유형	카드.
assetAdministrationShell 자산 관리 셀		AssetAdministrationShell 0..*	
하위 모델	하위 모델	하위 모델	0..*
개념설명	개념 설명	개념설명	0..*

## 5.7.10 자산 관리 셀에서 참조

## 5.7.10.1 개요

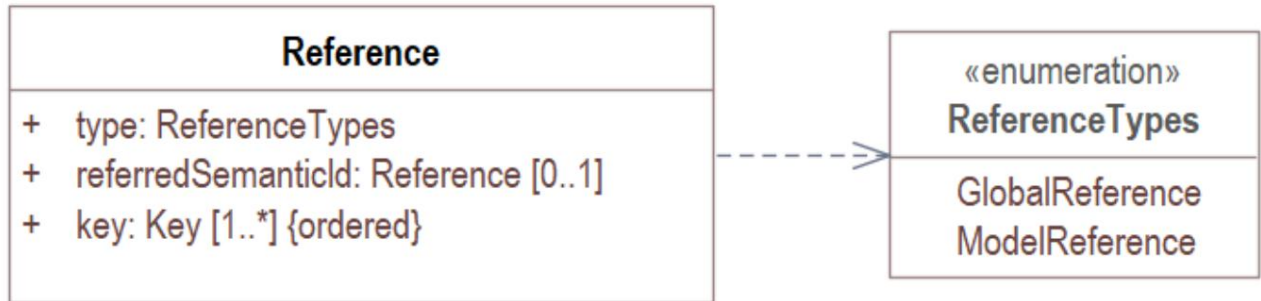
지금까지는 외부 개체 또는 엔터티에 대한 참조(전역 참조)와 동일하거나 다른 자산 관리 셀의 모델 요소에 대한 참조(모델 참조)의 두 가지 참조가 구분됩니다. 모델 참조는 AssetAdministrationShell의 하위 모델 등과 같은 메타 모델 고유 관계에도 사용됩니다(표기는 부록 C 참조).

전역 참조는 고유 식별자입니다. 식별자는 예를 들어 IIRDI 경로 등을 나타내는 다른 식별자의 연결일 수 있습니다.

참고: 참조는 로케이터와 혼동되어서는 안 됩니다. URL(URL = Uniform Resource Locator)도 식별자로 사용할 수 있으며 액세스할 수 있는 리소스를 반드시 설명하지는 않습니다.

5.7.10.2 참조 속성

그림 47 참조 메타모델



수업:	참조		
설명:	동일하거나 다른 AAS의 모델 요소 또는 외부 참조 실재. 참조는 순서가 지정된 키 목록입니다.  모델 참조는 순서가 지정된 키 목록으로, 각 키는 요소를 참조합니다. 예를 들어 전체 키 목록은 경로에 연결되어 요소에 고유한 액세스 권한을 부여할 수 있습니다.  전역 참조는 외부 엔터티에 대한 참조입니다.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
유형	참조의 유형입니다.  참조가 글로벌 참조인지 모델 참조인지를 나타냅니다.	참조 유형	1
referredSemanticId	참조된 모델 요소의 SemanticId (참조/유형=ModelReference). 전역 참조의 경우 일반적으로 의미론적 ID가 없습니다.  전역 참조를 사용하는 것이 좋습니다.	참조	0..1
키 <<주문>>	이름 공간의 고유 참조입니다.	열쇠	1..*

5.7.10.3 주요 속성

그림 48에는 키 유형의 논리적 모델이 나와 있습니다. 컨텍스트에 따라 다른 열거형이 파생될 수 있습니다. 참조 컨텍스트에서 열거는 "KeyTypes"입니다.



그림 48 참조 키에 대한 논리적 모델

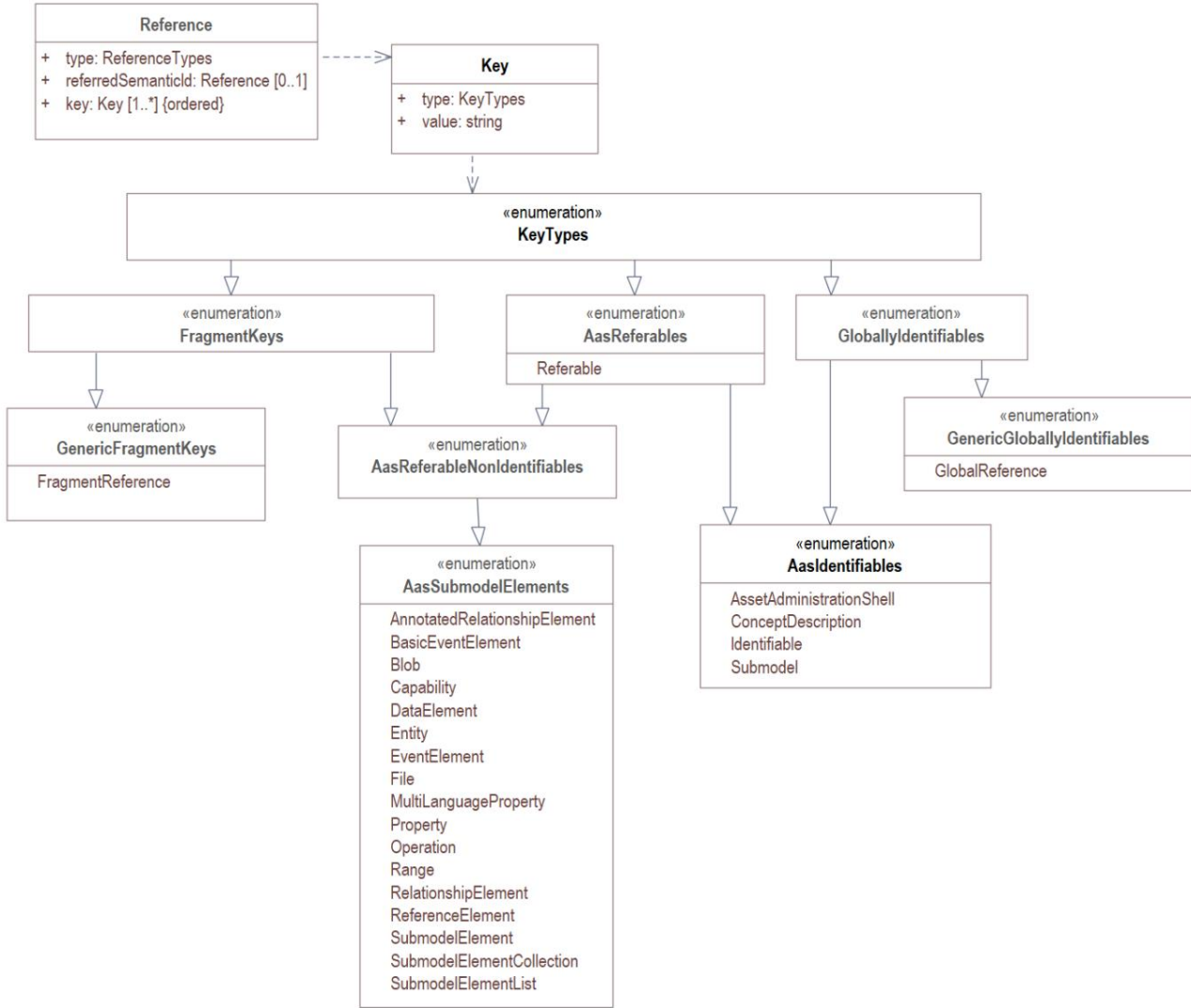
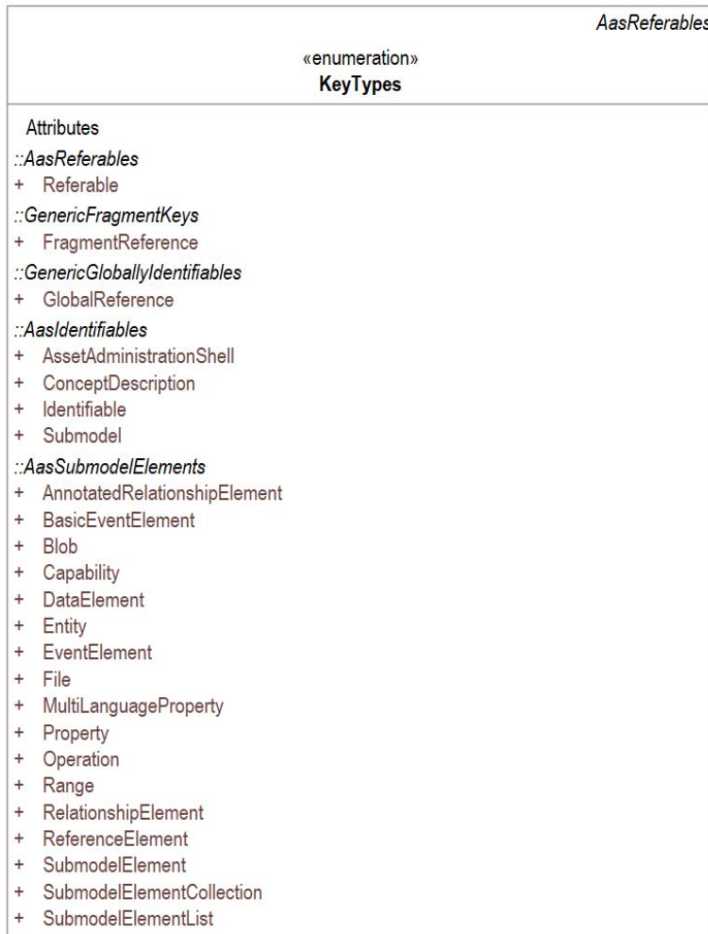


그림 49 KeyTypes 열거의 메타모델



수업:	열쇠		
설명: 키는 ID로 요소에 대한 참조입니다.			
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
유형	<p>참조되는 엔티티의 종류를 나타냅니다.</p> <p>type = FragmentId 인 경우 키는 이 키 앞에 오는 키로 지정된 상위 요소 내의 책갈피 또는 유사한 로컬 식별자를 나타냅니다.</p> <p>다른 모든 경우에 키는 동일하거나 다른 AAS의 모델 요소를 참조합니다. 모델 요소의 이름이 명시적으로 나열됩니다.</p>	중요 요소들	1
값	키 값(예: IRDI 또는 URI)	끈	1

FragmentId 를 키 유형으로 사용하는 예 : 자산 관리 셸 aasx 패키지의 일부인 파일 내의 요소에 대한 참조.

열거:	키 유형
설명:	키 내 다른 키 값 유형의 열거입니다.
세트:	FragmentKeys, 전역적으로 식별 가능한 항목
정확한	설명
참조 가능	참조 가능  참고: 참조 가능은 추상적입니다. 즉, 키가 "참조 가능"을 사용하는 경우 참조는 자산 관리 셀, 속성 등이 될 수 있습니다.
조각 참조	기본의 하위 부분에 대한 책갈피 또는 유사한 로컬 식별자 자원
글로벌 참조	글로벌 참조
자산 관리 셀	자산 관리 셀
개념설명	개념 설명
식별 가능	식별 가능  참고: 식별 가능은 추상적입니다. 즉, 키가 "식별 가능"을 사용하는 경우 자산 관리 셀, 하위 모델 또는 개념 설명이 참조될 수 있습니다.
하위 모델	하위 모델
AnnotatedRelationshipElement 주석이 달린 관계 요소	
기본 이벤트 요소	기본 이벤트 요소
얼룩	얼룩
능력	능력
데이터 요소	데이터 요소.  참고: 데이터 요소는 추상적입니다. 즉, 키가 "DataElement"를 사용하는 경우 참조는 속성, 파일 등이 될 수 있습니다.
실재	실재
이벤트 요소	이벤트  참고: 이벤트 요소는 추상적입니다.
파일	파일
MultiLanguage속성	여러 언어로 제공할 수 있는 값을 가진 속성
작업	작업
재산	재산
범위	최소 및 최대 범위
참조요소	참조
관계요소	관계
하위 모델 요소	하위 모델 요소

열거:	키 유형
설명:	키 내 다른 키 값 유형의 열거입니다.
세트:	FragmentKeys, 전역적으로 식별 가능한 항목
정확한	설명
	참고: 하위 모델 요소는 추상적입니다. 즉, 키가 "SubmodelElement"를 사용하는 경우 참조는 Property, SubmodelElementList, Operation 등이 될 수 있습니다.
하위 모델 요소 컬렉션	하위 모델 요소의 구조
하위 모델 요소 목록	하위 모델 요소 목록

열거:	프래그먼트 키
설명:	키 내 다른 조각 키 값 유형의 열거입니다.
세트:	AASReferableNonIdentifiabiles, GenericFragmentKeys
정확한	설명
조각 참조	기본 하위 부분에 대한 책갈피 또는 유사한 로컬 식별자 자원
AnnotatedRelationshipElement 주석	달린 관계 요소
기본 이벤트 요소	기본 이벤트 요소
열록	열록
능력	능력
데이터 요소	데이터 요소. 참고: 데이터 요소는 추상적입니다. 즉, 키가 "DataElement"를 사용하는 경우 참조는 속성, 파일 등이 될 수 있습니다.
실재	실재
이벤트 요소	이벤트 참고: 이벤트 요소는 추상적입니다.
파일	파일
MultiLanguage속성	여러 언어로 제공할 수 있는 값을 가진 속성
작업	작업
재산	재산
범위	최소 및 최대 범위
참조요소	참조
관계요소	관계
하위 모델 요소	하위 모델 요소

열거:	프래그먼트 키
설명:	키 내 다른 조각 키 값 유형의 열거입니다.
세트:	AASReferableNonIdentifiables, GenericFragmentKeys
정확한	설명
	참고: 하위 모델 요소는 추상적입니다. 즉, 키가 "SubmodelElement"를 사용하는 경우 참조는 Property, SubmodelElementList, Operation 등이 될 수 있습니다.
하위 모델 요소 컬렉션	하위 모델 요소의 구조
하위 모델 요소 목록	하위 모델 요소 목록

열거:	전 세계적으로 식별 가능
설명:	키 내 다른 키 값 유형의 열거입니다.
세트:	AAS식별 가능 항목, GenericGloballyIdentifiables
정확한	설명
글로벌 참조	글로벌 참조
자산 관리 셀	자산 관리 셀
개념설명	개념 설명
식별 가능	식별 가능 참고: 식별 가능은 추상적입니다. 즉, 키가 "식별 가능"을 사용하는 경우 자산 관리 셀, 하위 모델 또는 개념 설명이 참조될 수 있습니다.
하위 모델	하위 모델

열거:	AasReferableNonIdentifiables
설명:	키 내 다른 조각 키 값 유형의 열거입니다.
세트:	Aas서브모델요소
정확한	설명
AnnotatedRelationshipElement 주석	이 달린 관계 요소
기본 이벤트 요소	기본 이벤트 요소
얼룩	얼룩
능력	능력
데이터 요소	데이터 요소. 참고: 데이터 요소는 추상적입니다. 즉, 키가 "DataElement"를 사용하는 경우 참조는 속성, 파일 등이 될 수 있습니다.
실재	실재
이벤트 요소	이벤트

열거:	AasReferableNonIdentifiables
설명:	키 내 다른 조각 키 값 유형의 열거입니다.
세트:	Aas서브모델요소
정확한	설명
	참고: 이벤트 요소는 추상적입니다.
파일	파일
MultiLanguage속성	여러 언어로 제공할 수 있는 값을 가진 속성
작업	작업
재산	재산
범위	최소 및 최대 범위
참조요소	참조
관계요소	관계
하위 모델 요소	하위 모델 요소  참고: 하위 모델 요소는 추상적입니다. 즉, 키가 "SubmodelElement"를 사용하는 경우 참조는 Property, SubmodelElementList, Operation 등이 될 수 있습니다.
하위 모델 요소 컬렉션	하위 모델 요소의 구조
하위 모델 요소 목록	하위 모델 요소 목록

열거:	Aas서브모델요소
설명:	키 내 다른 조각 키 값 유형의 열거입니다.
세트:	--
정확한	설명
AnnotatedRelationshipElement 주석	이 달린 관계 요소
기본 이벤트 요소	기본 이벤트 요소
얼룩	얼룩
능력	능력
데이터 요소	데이터 요소.  참고: 데이터 요소는 추상적입니다. 즉, 키가 "DataElement"를 사용하는 경우 참조는 속성, 파일 등이 될 수 있습니다.
실재	실재
이벤트 요소	이벤트  참고: 이벤트 요소는 추상적입니다.

열거:	Aas서브모델요소
설명:	키 내 다른 조각 키 값 유형의 열거입니다.
세트:	--
정확한	설명
파일	파일
MultiLanguage속성	여러 언어로 제공할 수 있는 값을 가진 속성
작업	작업
재산	재산
범위	최소 및 최대 범위
참조요소	참조
관계요소	관계
하위 모델 요소	하위 모델 요소  참고: 하위 모델 요소는 추상적입니다. 즉, 키가 "SubmodelElement"를 사용하는 경우 참조는 Property, SubmodelElementList, Operation 등이 될 수 있습니다.
하위 모델 요소 컬렉션	하위 모델 요소의 구조
하위 모델 요소 목록	하위 모델 요소 목록

열거:	AasReferables
설명:	참조 가능한 열거
세트:	AasReferableNonIdentifiables, AasIdentifiables
정확한	설명
참조 가능	참조 가능  참고: 참조 가능은 추상적입니다. 즉 키가 "참조 가능"을 사용하는 경우 참조는 자산 관리 셀, 속성 등이 될 수 있습니다.
자산 관리 셀	자산 관리 셀
개념설명	개념 설명
식별 가능	식별 가능  참고: 식별 가능은 추상적입니다. 즉 키가 "식별 가능"을 사용하는 경우 자산 관리 셀, 하위 모델 또는 개념 설명이 참조될 수 있습니다.
하위 모델	하위 모델
AnnotatedRelationshipElement 주석	이 달린 관계 요소
기본 이벤트 요소	기본 이벤트 요소
얼룩	얼룩
능력	능력

열거:	AasReferables
설명:	참조 가능한 열거
세트:	AasReferableNonIdentifiabes, AasIdentifiabes
정확한	설명
데이터 요소	데이터 요소. 참고: 데이터 요소는 추상적입니다. 즉, 키가 "DataElement"를 사용하는 경우 참조는 속성, 파일 등이 될 수 있습니다.
실재	실재
이벤트 요소	이벤트 참고: 이벤트 요소는 추상적입니다.
파일	파일
MultiLanguage속성	여러 언어로 제공할 수 있는 값을 가진 속성
작업	작업
재산	재산
범위	최소 및 최대 범위
참조요소	참조
관계요소	관계
하위 모델 요소	하위 모델 요소 참고: 하위 모델 요소는 추상적입니다. 즉, 키가 "SubmodelElement"를 사용하는 경우 참조는 Property, SubmodelElementList, Operation 등이 될 수 있습니다.
하위 모델 요소 컬렉션	하위 모델 요소의 구조
하위 모델 요소 목록	하위 모델 요소 목록

열거:	일반 조각 키
설명:	키 내 다른 조각 키 값 유형의 열거입니다.
세트:	--
정확한	설명
조각 참조	기본의 하위 부분에 대한 책갈피 또는 유사한 로컬 식별자 자원

열거:	Aas식별 가능
설명:	키 내 다른 키 값 유형의 열거입니다.
세트:	--
정확한	설명
자산 관리 셀	자산 관리 셀
개념설명	개념 설명



열거:	Aas식별 가능
설명:	키 내 다른 키 값 유형의 열거입니다.
세트:	--
정확한	설명
식별 가능	식별 가능  참고: 식별 가능은 추상적입니다. 즉 키가 "식별 가능"을 사용하는 경우 자산 관리 셸, 하위 모델 또는 개념 설명이 참조될 수 있습니다.
하위 모델	하위 모델

열거:	GenericGloballyIdentifiables
설명:	키 내 다른 키 값 유형의 열거입니다.
세트:	--
정확한	설명
글로벌 참조	글로벌 참조

#### 5.7.10.4 제약

제약 AASd-121: 참조의 경우 참조/키의 첫 번째 키 유형은 GloballyIdentifiables 중 하나여야 합니다.

제약 AASd-122: 전역 참조, 즉 참조/유형 = GlobalReference가 있는 참조의 경우 참조/키의 첫 번째 키 유형은 GenericGloballyIdentifiables 중 하나여야 합니다.

제약 조건 AASd-123: 모델 참조, 즉 참조/유형 = ModelReference가 있는 참조의 경우 유형 Reference/keys의 첫 번째 키 중 하나는 AasIdentifiables 중 하나입니다.

제약 AASd-124: 전역 참조, 즉 참조/유형 = GlobalReference가 있는 참조의 경우 참조/키의 마지막 키는 GenericGloballyIdentifiables 또는 GenericFragmentKeys 중 하나여야 합니다.

제약 AASd-125: 모델 참조의 경우, 즉 Reference/type = ModelReference가 있는 참조, Reference/keys에 둘 이상의 키가 있는 Reference/keys의 첫 번째 키 다음에 오는 키 유형은 FragmentKeys 중 하나여야 합니다.

참고: 제약 조건 AASd-125는 최단 경로가 사용되도록 합니다.

제약 조건 AASd-126: 모델 참조, 즉 참조/유형 = ModelReference가 있는 참조의 경우 참조/키에 둘 이상의 키가 있는 참조 키 체인의 마지막 키 유형은 GenericFragmentKeys 중 하나이거나 키가 전혀 없어야 합니다. GenericFragmentKeys의 값입니다.

제약 조건 AASd-127: 모델 참조, 즉 참조/유형 = ModelReference가 있는 참조의 경우 참조/키에 둘 이상의 키가 있는 경우 유형이 FragmentReference인 키는 파일 또는 Blob 유형의 키가 앞에 와야 합니다. 다른 모든 AAS 조각, 즉 AasSubmodelElements의 유형 값은 조각을 지원하지 않습니다.

참고: 지원되는 조각 종류는 콘텐츠 유형과 참조를 통해 참조되는 해당 리소스에 대해 허용되는 조각 식별자 사양에 따라 다릅니다.

제약 조건 AASd-128: 모델 참조, 즉 Reference /type = ModelReference가 있는 참조의 경우 Key / type=SubmodelElementList 가 있는 키가 앞에 오는 키의 키/값은 하위 모델 요소 목록의 배열에서 위치를 나타내는 정수입니다..

유효한 참조의 예: \_\_\_\_\_

(하위 모델)http://example.com/aas/1/1/1234859590

(GlobalReference)http://example.com/specification.html

유효한 전역 참조의 예: \_\_\_\_\_

(GlobalReference)http://example.com/ressource

(GlobalReference)0173-1#02-EXA123#001

(GlobalReference)http://example.com/specification.html (FragmentReference)힌트

참고: (GlobalReference)http://example.com/specification.html (FragmentReference)Hints는 경로를 나타냅니다.  
조각 식별자 http://example.com/specification.html#Hints

유효한 모델 참조의 예: \_\_\_\_\_

(자산관리)http://example.com/aas/1/0/12348

(하위 모델)http://example.com/aas/1/1/1234859590

(서브모델)http://example.com/aas/1/1/1234859590, (파일)사양

(개념설명)0173-1#02-BAA120#008

(하위 모델)http://example.com/aas/1/1/1234859590, (SubmodelElementList)문서,  
(SubmodelElementCollection)0, (MultiLanguageProperty)제목

(서브모델)http://example.com/aas/1/1/1234859590, (SubmodelElementCollection)매뉴얼,  
(MultiLanguageProperty)제목

참고: (SubmodelElementCollection)0, (MultiLanguageProperty)Title은 의미 및 내용 면에서 (SubmodelElementCollection)Manual,  
(MultiLanguageProperty)Title과 동일할 수 있습니다. 차이점은 첫 번째 하위 모델에서는 둘 이상의 문서가 허용되므로 하위 모델 요소 목록이 정의된다는 것  
입니다. 목록의 요소에는 번호가 매겨집니다. 그러나 이 경우에도 표시 이름을 정의할 수 있습니다. 따라서 SubmodelElementCollection의 표시 이름은  
"Users Manual"과 같이 두 기반 모두에서 동일해야 합니다.

(서브모델)http://example.com/aas/1/1/1234859590, (파일)사양, (FragmentReference)힌트

파일에 절대 경로(상대 경로가 아님)를 사용하는 값이 있다고 가정합니다.  
http://example.com/specification.html 그런 다음 첫 번째 참조는 전역 참조(GlobalReference)http://example.com/specification.html,  
(FragmentReference)힌트와 동일한 정보를 가리킵니다.

(하위 모델)http://example.com/aas/1/1/1234859590, (Blob)사양, (FragmentReference)힌트

잘못된 모델 참조의 예: \_\_\_\_\_

(글로벌 참조)http://example.com/aas/1/1/1234859590

(속성)0173-1#02-BAA120#008

(서브모델)http://example.com/aas/1/1/1234859590, (EventElement)이벤트, (FragmentReference)댓글

(AssetAdministrationShell)http://example.com/aas/1/0/12348, (서브모델)http://example.com/aas/1/1/1234859590, (속성)은도

AssetAdministrationShell 과 Submodel 은 모두 전역적으로 식별할 수 있으므로 유효한 모델 참조가 아닙니다 .

### 5.7.10.5 참조 매칭 알고리즘

5.4.6절에서 의미 식별자에 대한 일치 전략이 논의되었습니다. 이 절에서는 참조 개념을 기반으로 한 일치 전략에 대해 더 자세히 설명하므로 다른 종류의 식별 요소도 다룹니다.

예를 들어, 9.2.3절에 정의된 참조의 문자열 직렬화는 이해를 돕기 위해 사용됩니다.

정확히 일치:

- 전역 참조 A는 모든 키의 모든 값이 동일한 경우 전역 참조 B와 일치합니다. 참고: 조각 값이 전역 참조 값과 동일하여 다른 것을 참조할 가능성은 없다고 가정합니다.
- 모델 참조 A는 모든 키의 모든 값이 동일한 경우 모델 참조 B와 일치합니다.  
참고: 조각 키가 항상 고유하기 때문에 키 유형을 무시할 수 있습니다(예: 상위 모델의 모든 하위 모델 요소 또는 하위 모델 요소 목록 또는 컬렉션의 모든 하위 모델 요소). • 전역 참조 A는 모델 참조 B와 일치하고 모든 키의 모든 값이 동일한 경우 그 반대의 경우도 마찬가지입니다.

참고: AAS의 식별 항목은 전역적으로 고유하므로 모델 참조는 전역 참조의 특수한 경우입니다. 유일한 차이점은 AAS 요소에 대해 사전 정의된 키 유형의 처리입니다. 예를 들어 IRDI 경로 등에 대해 다른 키 유형을 미리 정의할 수 있지만 지금까지는 일반 키 유형만 지원됩니다.

참고: 모든 키 값이 키 체인의 올바른 순서를 따르지만 키 유형이 동일하지 않은 경우 최소한 참조에 버그가 있으므로 경고를 발생해야 합니다.

### 5.7.11 템플릿, 상속, 한정자 및 범주

언뜻 보기에 데이터 사양 템플릿, 확장, 상속, 한정자 및 범주의 개념 간에 겹치는 부분이 있는 것 같습니다. 이 절에서는 공통점과 차이점이 설명되고 모범 사례에 대한 힌트가 제공됩니다.

일반적으로 상속에 의한 메타모델의 확장이 예상됩니다. 대안으로 템플릿을 사용할 수도 있습니다.

- 확장을 사용하여 독점 및/또는 임시 정보를 요소에 추가할 수 있습니다. 확장은 상호 운용성을 지원하지 않습니다. 표준에서 누락된 속성에 대한 해결 방법으로 사용할 수 있습니다.  
이 경우 동일한 확장이 특정 클래스의 모든 요소(예: 속성)에 연결됩니다.  
그러나 일반적으로 확장은 매우 임의적인 방식으로 첨부될 수 있습니다. 속성은 키 값 쌍(이 경우 "이름"이라는 키)으로 사전 정의된 방식으로 정의됩니다.
- 확장과 대조되는 템플릿은 템플릿에 동의하는 파트너 간의 상호 운용성을 가능하게 하는 것을 목표로 합니다. 템플릿은 클래스의 다른 인스턴스가 다른 스키마를 따르고 스키마에 대한 템플릿이 메타모델의 디자인 타임에 알려지지 않은 경우에만 사용해야 합니다. 전체 메타모델이 아직 충분히 안정적이지 않거나 도구가 템플릿을 지원하지만 완전한 메타모델은 지원하지 않는 경우 템플릿을 사용할 수도 있습니다. 일반적으로 동일한 범주를 가진 특정 클래스의 모든 인스턴스는 템플릿을 준수하는 동일한 속성 값을 제공합니다. 확장과 달리 템플릿의 속성에는 말하는 이름이 있습니다.
- 단: 표준화되지 않은 독점 데이터 사양 템플릿을 사용하는 경우 상호 운용성 보장할 수 없으므로 가능한 한 피해야 합니다.
- 클래스의 모든 인스턴스가 동일한 스키마를 따르는 경우 상속 및/또는 범주를 사용해야 합니다.
- 클래스의 모든 인스턴스가 동일한 스키마를 따르지만 해당 범주에 따라 제약 조건이 다른 경우 범주를 사용할 수 있습니다. 이러한 제약 조건은 선택적 속성이 다음에 대해 필수임을 지정할 수 있습니다.

이 범주(예: 물리적 가치를 나타내는 속성에 필수 단위).

상속을 통해 동일한 것을 실현하면 생략해야 할 다중 상속이 발생합니다<sup>13</sup>.

- 요소의 구조와 의미 체계가 한정자와 무관하게 동일한 경우 한정자가 사용됩니다. 요소에 대한 값의 품질이나 의미만 다릅니다.

## 5.7.12 기본 및 단순 데이터 유형

### 5.7.12.1 미리 정의된 단순 데이터 유형

자산 관리 셸의 메타모델은 XSD(XML 스키마 정의)<sup>14</sup>에 정의된 대로 다음과 같은 기본 데이터 유형을 사용합니다. 이들의 정의는 이 문서의 범위를 벗어납니다.

xsd 유형의 의미와 형식은 <https://www.w3.org/XML/Schema>에 지정되어 있습니다. 단순 유형 "langString"은 RDF(Resource Description Framework)<sup>15</sup>에 지정되어 있습니다.

소스 기본 데이터 유형	xsd 문자열	값 범위 문자열	샘플 값
		든 유니코드 문자열은 아님	"안녕 월드", "굿모닝 월드", "콘니차하"
xsd	부울 바이	참, 거짓	허위 사실
xsd	트	-128...+127(8비트)	-1, 0, 127
rdf	langString	언어 태그가 있는 문자열	"안녕"@en, "안녕"@de. 이것은 RDF/Turtle 구문으로 작성되었으며 "Hello" 및 "Hello"만 실제 값입니다.

간단한 데이터 유형은 소문자로 시작합니다.

이러한 유형 외에도 다양한 유형(즉, 기본 데이터 유형 "ValueDataType" 사용)이 있는 데이터 요소의 특수 처리가 있습니다. 그것들은 5.7.12.2절과 5.7.12.3절에서 논의됩니다.

### 5.7.12.2 원시 데이터 유형

특정 데이터 사양 템플릿에 사용되는 유형은 데이터 사양의 해당 절에 나열되어 있습니다.

표 6은 메타모델에 사용된 Primitives를 나열합니다. 기본 데이터 유형은 대문자로 시작합니다.

표 6 메타모델에서 사용되는 기본 데이터 유형

원어	설명	값 예
얼룩 유형	바이트 그룹 파일 내용을 나타내기 위해(바이너리 및 비바이너리)	<pre>&lt;?xml 버전="1.0" 인코딩="UTF-8"?&gt; &lt;schema elementFormDefault="qualified" targetNamespace="http://www.admin-shell.io/aas/2/0" xmlns="http://www.w3.org/2001/XMLSchema" xmlns:aas="http://www.admin-shell.io/aas/2/0" xmlns:abac="http://www.admin-shell.io/aas/abac/2/0" xmlns:IEC61360="http://www.admin-shell.io/IEC61360/2/0"&gt;   &lt;import 네임스페이스="http://www.admin-shell.io/aas/abac/2/0" schemaLocation="AAS_ABAC.xsd"/&gt;</pre>

<sup>13</sup> 예외: 이 사양에서는 다중 상속이 사용되지만 추상에서 상속하는 경우에만 클래스.

<sup>14</sup> <https://www.w3.org/XML/Core/>, 이전 <https://www.w3.org/XML/Schema>

<sup>15</sup> 참조: <https://www.w3.org/TR/rdf11-concepts/>

원어	설명	값 예
		<p>MZ • _____ yy____, _____ @ _____            _____ € _____ ° _____'            Í! ,_LÍ! 이 프로그램은 다음에서 실행할 수 없습니다.            에서            모드.\$ _____ PE__L__Rö\^ _____ to_</p>
식별자	끈	<p>https://cust/123456            0173-1#02-BAA120#008</p>
랭스트링셋	<p>langString 유형의 요소 배열</p> <p>langString은 RDF 데이터 유형입니다.            langString은 언어 코드로 태그가 지정된 문자열 값입니다.</p> <p>이것이 실현되는 방법은 기술의 직렬화 규칙에 따라 다릅니다.</p>	<p>XML에서:</p> <p>&lt;aas:langString lang="EN"&gt;다국어 값입니다.            영어&lt;/aas:langString&gt;            &lt;aas:langString lang="DE"&gt; 독일어로 된 다국어 값입니다. &lt;/aas:langString&gt;</p> <p>rdf에서:</p> <p>"이것은 영어로 된 다국어 값입니다."@en ;            "이것은 독일어로 된 다국어 값입니다."@de</p> <p>JSON에서:</p> <pre>{   "언어": "English",   "text": "이것은 영어로 된 다국어 값입니다." } {   "언어": "DE",   "text": "이것은 독일어로 된 다국어 값입니다." }</pre>
콘텐츠 타입	<p>끈</p> <p>RFC2046과 같은 모든 콘텐츠 유형.</p> <p>미디어 유형(MIME 유형 및 콘텐츠 유형도 포함) [...]은 인터넷에서 전송되는 파일 형식 및 형식 콘텐츠에 대한 두 부분 식별자입니다.            IANA(Internet Assigned Numbers Authority)는 이러한 분류의 표준화 및 출판을 위한 공식 기관입니다. 미디어 유형은 원래 Request for</p>	<p>신청서/pdf            이미지/jpeg</p>

원어	설명	값 예
	<p>덧글 2045(1996년 11월)는 MIME(Multipurpose Internet Mail Extensions) 사양의 일부로 전자 메일 메시지의 콘텐츠 및 첨부 파일의 유형을 나타냅니다.<sup>16</sup></p>	
경로 유형	<p>끈</p> <p>다음에 준수하는 모든 문자열 RFC808917, "파일" URI 체계 (상대 및 절대 파일 경로의 경우)</p>	<p>./사양.pdf</p> <p>파일:c:/local/Specification.pdf</p> <p>파일://host.example.com/path/to/file</p>
한정자 유형	끈	<p>표현의미</p> <p>라이프 사이클 품질</p>
값 데이터 유형	<p>DataTypeDefXsd 를 통해 지정된 모든 xsd 원자 유형</p>	<p>"이것은 문자열 값입니다"</p> <p>10</p> <p>1.5</p> <p>2020-04-01</p> <p>진실</p>

### 5.7.12.3 하위 모델 요소 값 유형에 대한 열거

열거형은 기본 데이터 유형입니다. 대부분의 열거는 사용되는 클래스의 컨텍스트에서 정의됩니다. 이 절에서는 하위 모델 요소 값 유형<sup>18</sup>에 대한 열거가 정의됩니다.

속성 및 기타 값의 값 유형을 정의하는 데 사용되는 미리 정의된 유형은 XSD(XML 스키마 정의)<sup>19</sup>의 이름과 의미를 사용합니다. 또한 RDF(Resource Description Framework)<sup>20</sup>에 정의된 의미 체계를 가진 "langString" 유형이 사용됩니다. "langString"은 언어 코드로 태그가 지정된 문자열 값입니다.

RDF21 은 다음 xsd 데이터 유형을 사용하지 않을 것을 권장합니다. 이것이 허용된 데이터에서 제외되는 이유입니다.

- XSD BuildIn 목록 유형은 지원되지 않습니다(ENTITIES, IDREFS 및 NMTOKENS).
- XSD 문자열 BuildIn 유형은 지원되지 않습니다(normalizedString, 토큰, 언어, NCName, ENTITY, ID, IDREF).
- NOTATION, QName과 같은 XSD 기본 유형은 지원되지 않습니다.

<sup>16</sup> Wikipedia.org, 날짜: 2018-04-09

<sup>17</sup> <https://datatracker.ietf.org/doc/html/rfc8089>

<sup>18</sup> 예: 속성/값 유형

<sup>19</sup> <https://www.w3.org/XML/Schema> 참조

<sup>20</sup> 참조: <https://www.w3.org/TR/rdf11-concepts/>

<sup>21</sup> <https://www.w3.org/TR/rdf11-concepts/#xsd-datatypes> 참조

참고: HTML 및 XMLLiteral과 같은 RDF 유형은 지원되지 않습니다.

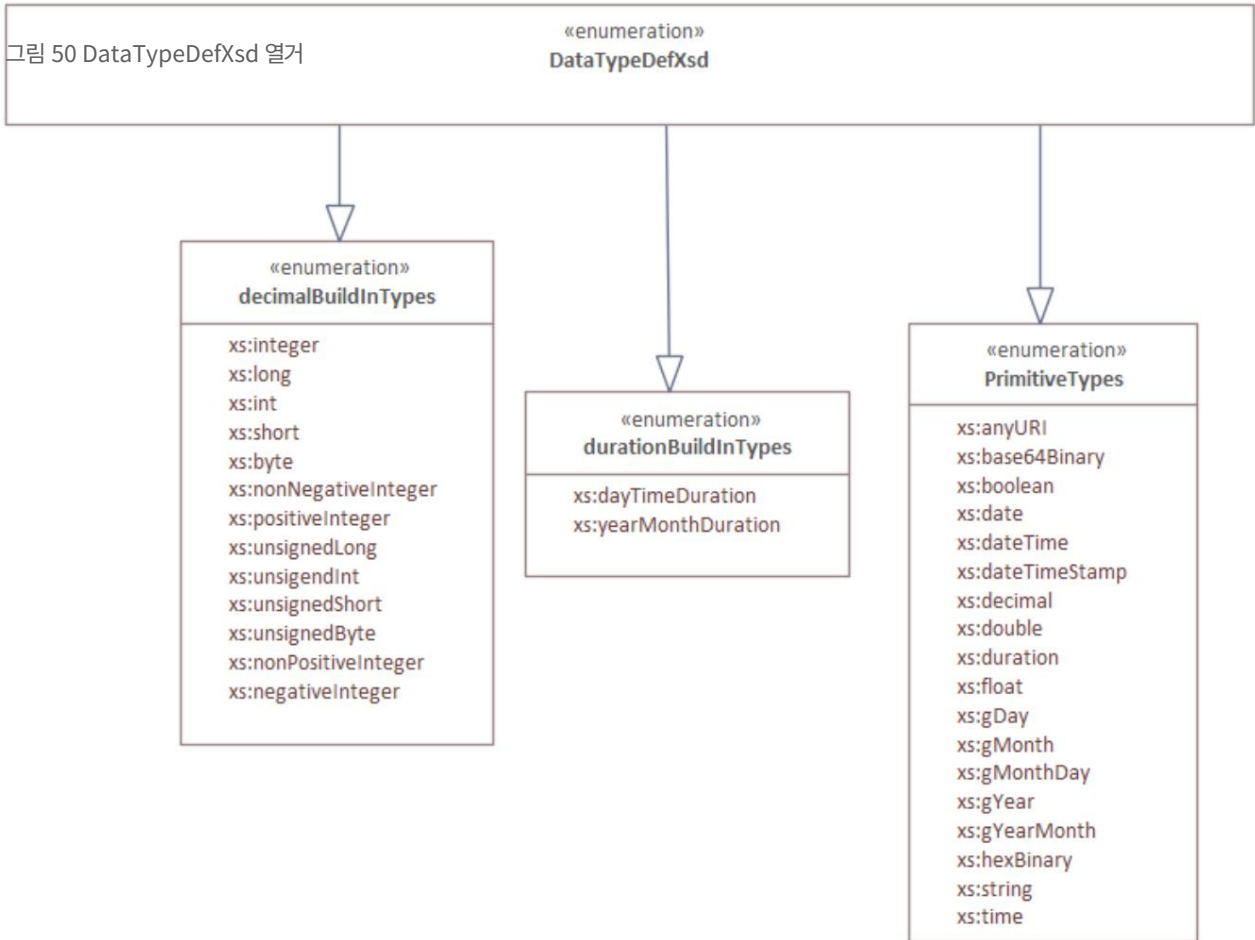
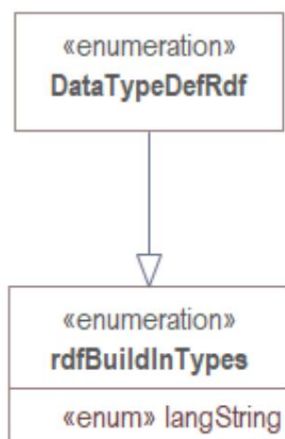


그림 51 DefTypeDefRdf 열거



다른 데이터 유형의 값과 값 범위는 표 7에 나와 있습니다. 왼쪽 열 "데이터 유형"은 하위 모델 요소 값에 사용할 수 있는 데이터 유형을 보여줍니다. 데이터 유형은 W3C XML 스키마(<https://www.w3.org/TR/xmlschema-2/#built-in-datatypes> 및 <https://www.w3.org/TR/xmlschema-2/>)에 따라 정의됩니다. /#내장 파생). "값 범위"는 이 데이터 유형에 대해 가능한 데이터 값 범위를 추가로 설명합니다. 오른쪽 열에는 해당 데이터 유형의 값에 대한 관련 예가 있습니다.

표 7 예제가 있는 데이터 유형2

	데이터 유형	값 범위	샘플 값
핵심 유형	xs:문자열	문자열(하지만 모두 유니코드 문자열)	"Hello world", "Good morning world", "콘 니차하"
	xs:부울	허위 사실	허위 사실
	xs:십진수	임의 정밀도 십진수	-1.23, 126789672374892739424.543233, +100000.00, 210
	xs:정수	임의 크기의 정수	-1, 0, 126789675432332938792837429837429837429, +100000
IEEE 부동 소수 점 숫자	xs:더블	64비트 부동 소수점 숫자 ±Inf, ±0, NaN	-1.0, +0.0, -0.0, 234.567e8, -INF, NaN
	xs:플로트	32비트 부동 소수점 숫자 ±Inf, ±0, NaN	-1.0, +0.0, -0.0, 234.567e8, -INF, NaN
시간과 데이터	xs:날짜	없는 날짜(yyyy-mm-dd) 또는 시간대	"2000-01-01", "2000-01-01Z", "2000-01-01+12:05"
	xs:시간	타임스 (hh:mm:ss.sss...) 시간대 포함 또는 제외	"14:23:00", "14:23:00.527634Z", "14:23:00+03:00"
	xs:날짜 시간	또는 포함된 날짜 및 시간 시간대 없이	"2000-01-01T14: 23: 00", "2000-01-01T14: 23: 00.66372+14: 00 "
	xs:날짜타임스탬프	날짜와 시간 필요한 시간대	"2000-01-01T14: 23: 00.66372+14: 00"
반복 및 부분 날짜	xs:g년	그레고리우스 달력 년도	"2000", "2000+03:00"
	xs:g월	그레고리력 달 달력	"--04", "--04+03:00"
	xs:g데이	그레고리력 일 달력	"---04", "---04+03:00"
	xs:g년월	그레고리력 연도 및 월	"2000-01", "2000-01+03:00"
	xs:gMonthDay	그레고리력 월 달력 과 일	"--01-01", "--01-01+03:00"
	xs:지속 시간	기간	"P30D", "-P1Y2M3DT1H", "PT1H5M0S"
	xs:yearMonthDuration	기간(월 및 연도만) 의 시각	"P10M", "P5Y2M"
	xs:dayTimeDuration	시간 기간(일, 시, 분, 초만 해당)	"P30D", "P1DT5H", "PT1H5M0S"
	xs:바이트	-128...+127(8비트)	-1, 0, 127

22 짧은 설명과 함께 RDF 호환 XSD 유형 목록을 참조하십시오 . [https://www.w3.org/TR/rdf11-concepts/#xsd 데이터 유형](https://www.w3.org/TR/rdf11-concepts/#xsd-데이터-유형).  
[https://openmanufacturingplatform.github.io/sds-bamm-aspect-meta-model/bamm 사양/v1.0.0/datatypes.html](https://openmanufacturingplatform.github.io/sds-bamm-aspect-meta-model/bamm-사양/v1.0.0/datatypes.html) 의 예



제한된 범위 정수	xs:짧은	-32768...+32767(16비트)	-1, 0, 32767
	xs:int	2147483648...+2147483647(32비트)	-1, 0, 2147483647
	xs:긴	-9223372036854775808...+9223372036854775807(64비트)	-1, 0, 9223372036854775807
	xs:unsignedByte	0...255(8비트) 0...	0, 1, 255
	xs:unsignedShort	65535(16비트) 0...	0, 1, 65535
	xs:unsignedInt	4294967295(32비트)	0, 1, 4294967295
	xs:unsignedLong	0...18446744073709551615(64비트)	0, 1, 18446744073709551615
	xs:positive정수	정수 >0	1, 7345683746578364857368475638745
	xs:nonNegativeInteger 정수 ≥0	정수 ≥0	0, 1, 7345683746578364857368475638745
	xs:음수 정수	정수 <0	-1, 23487263847628376482736487263847
xs:nonPositiveInteger	정수 ≤0	-1, 0, 93845837498573987498798987394	
인코딩 바이너리 데이터	xs:16진 바이너리	16진수로 인코딩된 이진 데이터	"6b756d6f77617368657265"
	xs:base64바이너리	Base64 인코딩 바이너리 데이터	"a3Vtb3dhc2hlcmU="
기타 예외 유형	xs: 모든 URI	절대 또는 상대적 URI 및 IRI	"http://customer.com/demo/aas/1/1/1234859590", "urn:example:company:1.0.0"
	rdf:langString	언어 태그가 있는 문자열	"안녕"@en, "안녕"@de. 이것은 RDF/Turtle 구문으로 작성되었으며 "Hello" 및 "Hello"만 실제 값입니다.

열거:	DataTypeDefXsd
설명:	모든 xsd anySimpleType을 나열하는 열거 자세한 내용은 <a href="https://www.w3.org/TR/rdf11">https://www.w3.org/TR/rdf11</a> 를 참조하십시오. 개념/#xsd-데이터 유형
세트:	DecimalBuildInTypes, durationBuildInTypes, PrimitiveTypes
정확한	설명
xs:정수	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#integer">https://www.w3.org/TR/xmlschema11-2/#integer</a>
xs:긴	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#long">https://www.w3.org/TR/xmlschema11-2/#long</a>
xs:int	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#int">https://www.w3.org/TR/xmlschema11-2/#int</a>
xs:짧은	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#short">https://www.w3.org/TR/xmlschema11-2/#short</a>
xs:바이트	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#byte">https://www.w3.org/TR/xmlschema11-2/#byte</a>

열거:	DataTypeDefXsd	
xs:비음수 정수	보다:	<a href="https://www.w3.org/TR/xmlschema11-2/#음수가_아닌_정수">https://www.w3.org/TR/xmlschema11-2/#음수가 아닌 정수</a>
xs:positive정수	보다:	<a href="https://www.w3.org/TR/xmlschema11-2/#positive정수">https://www.w3.org/TR/xmlschema11-2/#positive정수</a>
xs:unsignedLong	보다:	<a href="https://www.w3.org/TR/xmlschema11-2/#unsignedLong">https://www.w3.org/TR/xmlschema11-2/#unsignedLong</a>
xs:unsignedInt	참조:	<a href="https://www.w3.org/TR/xmlschema11-2/#unsignedInt">https://www.w3.org/TR/xmlschema11-2/#unsignedInt</a>
xs:unsignedShort	보다:	<a href="https://www.w3.org/TR/xmlschema11-2/#unsigned짧음">https://www.w3.org/TR/xmlschema11-2/#unsigned짧음</a>
xs:unsignedByte	보다:	<a href="https://www.w3.org/TR/xmlschema11-2/#unsigned짧음">https://www.w3.org/TR/xmlschema11-2/#unsigned짧음</a>
xs:nonPositiveInteger	보다:	<a href="https://www.w3.org/TR/xmlschema11-2/#nonPositiveInteger">https://www.w3.org/TR/xmlschema11-2/#nonPositiveInteger</a>
xs:음수 정수	보다:	<a href="https://www.w3.org/TR/xmlschema11-2/#음수_정수">https://www.w3.org/TR/xmlschema11-2/#음수 정수</a>
xs:주간 지속 시간	보다:	<a href="https://www.w3.org/TR/xmlschema11-2/#dayTimeDuration">https://www.w3.org/TR/xmlschema11-2/#dayTimeDuration</a>
xs:년월지속기간	보다:	<a href="https://www.w3.org/TR/xmlschema11-2/#yearMonthDuration">https://www.w3.org/TR/xmlschema11-2/#yearMonthDuration</a>
xs: 모든 URI	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#anyURI">https://www.w3.org/TR/xmlschema-2/#anyURI</a>
xs:base64바이너리	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#base64Binary">https://www.w3.org/TR/xmlschema-2/#base64Binary</a>
xs:부울	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#boolean">https://www.w3.org/TR/xmlschema-2/#boolean</a>
xs:날짜	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#date">https://www.w3.org/TR/xmlschema-2/#date</a>
xs:날짜 시간	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#dateTime">https://www.w3.org/TR/xmlschema-2/#dateTime</a>
xs:십진수	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#decimal">https://www.w3.org/TR/xmlschema-2/#decimal</a>
xs:더블	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#double">https://www.w3.org/TR/xmlschema-2/#double</a>
xs:지속 시간	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#duration">https://www.w3.org/TR/xmlschema-2/#duration</a>
xs:플로트	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#float">https://www.w3.org/TR/xmlschema-2/#float</a>
xs:g데이	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#gDay">https://www.w3.org/TR/xmlschema-2/#gDay</a>
xs:g월	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#gMonth">https://www.w3.org/TR/xmlschema-2/#gMonth</a>
xs:gMonthDay	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#gMonthDay">https://www.w3.org/TR/xmlschema-2/#gMonthDay</a>
xs:g년	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#gYear">https://www.w3.org/TR/xmlschema-2/#gYear</a>
xs:g년월	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#gYearMonth">https://www.w3.org/TR/xmlschema-2/#gYearMonth</a>
xs:16진 바이너리	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#hexBinary">https://www.w3.org/TR/xmlschema-2/#hexBinary</a>
xs:문자열	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#string">https://www.w3.org/TR/xmlschema-2/#string</a>
xs:시간	참조:	<a href="https://www.w3.org/TR/xmlschema-2/#time">https://www.w3.org/TR/xmlschema-2/#time</a>

열거:

10진수빌드인타입

설명:	십진수 유형의 모든 xsd 빌드를 나열하는 열거
세트:	--
정확한	설명
xs:정수	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#integer">https://www.w3.org/TR/xmlschema11-2/#integer</a>
xs:긴	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#long">https://www.w3.org/TR/xmlschema11-2/#long</a>
xs:int	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#int">https://www.w3.org/TR/xmlschema11-2/#int</a>
xs:짧은	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#short">https://www.w3.org/TR/xmlschema11-2/#short</a>
xs:바이트	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#byte">https://www.w3.org/TR/xmlschema11-2/#byte</a>
xs:비음수 정수	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#nonNegativeInteger">https://www.w3.org/TR/xmlschema11-2/#nonNegativeInteger</a>
xs:positive정수	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#positiveInteger">https://www.w3.org/TR/xmlschema11-2/#positiveInteger</a>
xs:unsignedLong	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#unsignedLong">https://www.w3.org/TR/xmlschema11-2/#unsignedLong</a>
xs:unsignedInt	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#unsignedInt">https://www.w3.org/TR/xmlschema11-2/#unsignedInt</a>
xs:unsignedShort	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#unsignedShort">https://www.w3.org/TR/xmlschema11-2/#unsignedShort</a>
xs:unsignedByte	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#unsignedShort">https://www.w3.org/TR/xmlschema11-2/#unsignedShort</a>
xs:nonPositiveInteger	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#nonPositiveInteger">https://www.w3.org/TR/xmlschema11-2/#nonPositiveInteger</a>
xs:음수 정수	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#negativeInteger">https://www.w3.org/TR/xmlschema11-2/#negativeInteger</a>

열거:	지속 시간 빌드 인 유형
설명:	기간과 관련하여 모든 xsd 빌드 인 유형을 나열하는 열거
세트:	--
정확한	설명
xs:주간 지속 시간	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#dayTimeDuration">https://www.w3.org/TR/xmlschema11-2/#dayTimeDuration</a>
xs:년월지속기간	참조: <a href="https://www.w3.org/TR/xmlschema11-2/#yearMonthDuration">https://www.w3.org/TR/xmlschema11-2/#yearMonthDuration</a>

열거:	기본 유형
설명:	모든 xsd 기본 유형을 나열하는 열거
세트:	--
정확한	설명
xs: 모든 URI	참조: <a href="https://www.w3.org/TR/xmlschema-2/#anyURI">https://www.w3.org/TR/xmlschema-2/#anyURI</a>
xs:base64바이너리	참조: <a href="https://www.w3.org/TR/xmlschema-2/#base64Binary">https://www.w3.org/TR/xmlschema-2/#base64Binary</a>
xs:부울	참조: <a href="https://www.w3.org/TR/xmlschema-2/#boolean">https://www.w3.org/TR/xmlschema-2/#boolean</a>
xs:날짜	참조: <a href="https://www.w3.org/TR/xmlschema-2/#date">https://www.w3.org/TR/xmlschema-2/#date</a>
xs:날짜 시간	참조: <a href="https://www.w3.org/TR/xmlschema-2/#dateTime">https://www.w3.org/TR/xmlschema-2/#dateTime</a>
xs:십진수	참조: <a href="https://www.w3.org/TR/xmlschema-2/#decimal">https://www.w3.org/TR/xmlschema-2/#decimal</a>
xs:더블	참조: <a href="https://www.w3.org/TR/xmlschema-2/#double">https://www.w3.org/TR/xmlschema-2/#double</a>
xs:지속 시간	참조: <a href="https://www.w3.org/TR/xmlschema-2/#duration">https://www.w3.org/TR/xmlschema-2/#duration</a>
xs:플로트	참조: <a href="https://www.w3.org/TR/xmlschema-2/#float">https://www.w3.org/TR/xmlschema-2/#float</a>

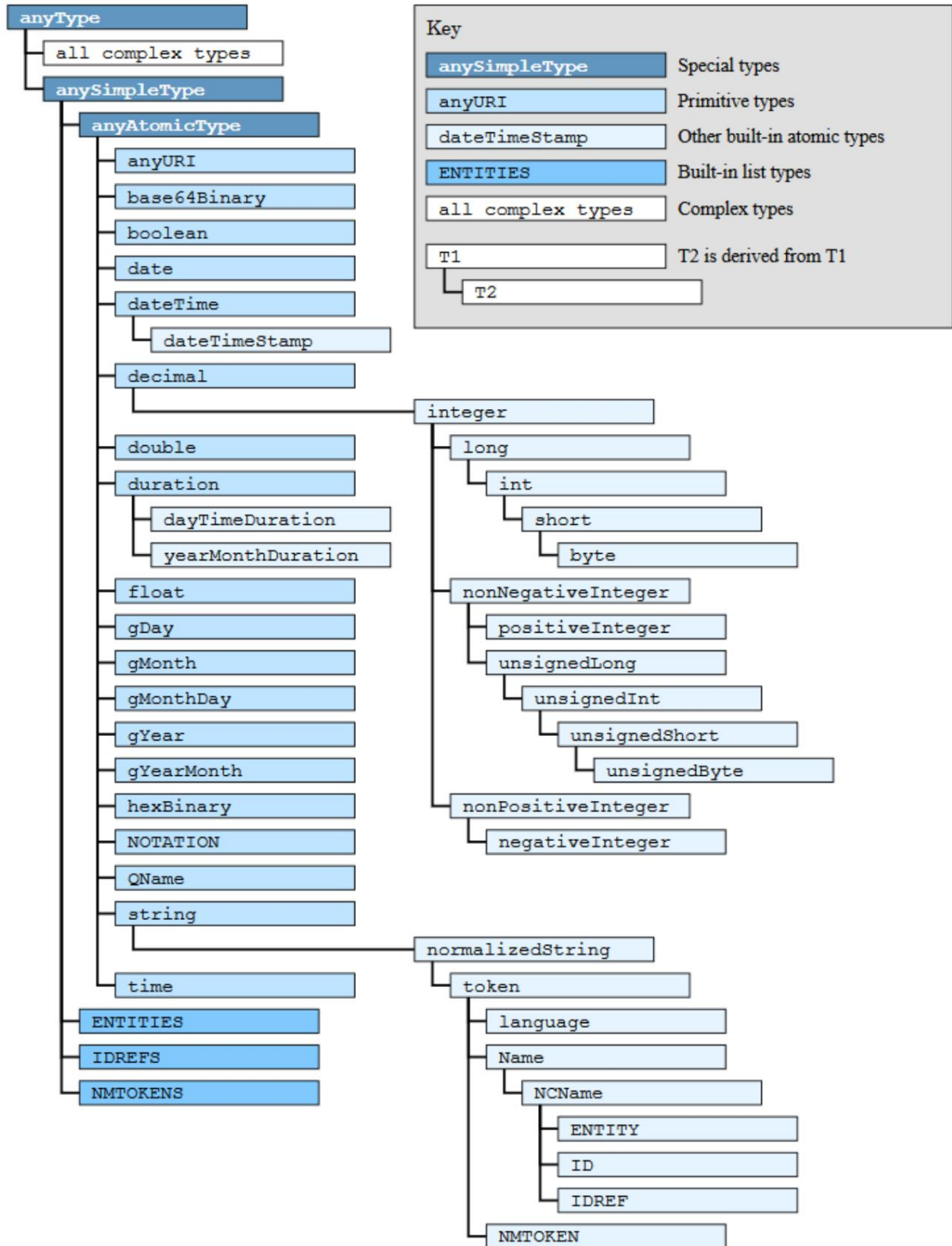
열거:	기본 유형
xs:g데이	참조: <a href="https://www.w3.org/TR/xmlschema-2/#gDay">https://www.w3.org/TR/xmlschema-2/#gDay</a>
xs:g월	참조: <a href="https://www.w3.org/TR/xmlschema-2/#gMonth">https://www.w3.org/TR/xmlschema-2/#gMonth</a>
xs:gMonthDay	참조: <a href="https://www.w3.org/TR/xmlschema-2/#gMonthDay">https://www.w3.org/TR/xmlschema-2/#gMonthDay</a>
xs:g년	참조: <a href="https://www.w3.org/TR/xmlschema-2/#gYear">https://www.w3.org/TR/xmlschema-2/#gYear</a>
xs:g년월	참조: <a href="https://www.w3.org/TR/xmlschema-2/#gYearMonth">https://www.w3.org/TR/xmlschema-2/#gYearMonth</a>
xs:16진 바이너리	참조: <a href="https://www.w3.org/TR/xmlschema-2/#hexBinary">https://www.w3.org/TR/xmlschema-2/#hexBinary</a>
xs:문자열	참조: <a href="https://www.w3.org/TR/xmlschema-2/#string">https://www.w3.org/TR/xmlschema-2/#string</a>
xs:시간	참조: <a href="https://www.w3.org/TR/xmlschema-2/#time">https://www.w3.org/TR/xmlschema-2/#time</a>

열거:	DataTypeDefRdf
설명:	모든 RDF 유형을 나열하는 열거
세트:	--
정확한	설명
rdf:langString	언어 태그가 있는 문자열

RDF에는 IETF BCP 4723 언어 태그가 필요합니다. 즉, ISO 639-1을 준수하는 "de"와 같은 로케일에 대한 간단한 두 글자 언어 태그는 물론 국가 코드, 방언 등에 대한 "de-DE"와 같은 확장자도 허용됩니다. 영어(영국) 및 영어(미국)의 경우 "en-US" 또는 "en-GB". IETF 언어 태그는 ISO 639, ISO 3166 및 ISO 15924를 참조합니다.

<sup>23</sup> <https://tools.ietf.org/rfc/bcp/bcp47.txt> 참조

그림 52 XML 스키마 정의 1.1(XSD)의 기본 제공 유형



### 5.7.13 교차 제약과 불변

#### 5.7.13.1 소개

이 절에서는 단일 클래스에 할당할 수 없는, 즉 클래스 불변성이 아닌 제약 조건이 문서화됩니다.

클래스 불변은 항상 클래스의 모든 인스턴스에 대해 true여야 하는 제약 조건입니다.

### 5.7.13.2 참조 대상 및 식별 대상에 대한 제약 조건

제약 AASd-002: 참조 가능 항목의 idShort는 문자, 숫자, 밑줄("\_")만 포함해야 합니다. 편지로 의무적으로 시작합니다. 즉, [a-zA-Z][a-zA-Z0-9\_]+.

제약 AASd-117: SubmodelElementList와 같지 않은 식별 불가능한 참조 가능 항목의 idShort가 지정되어야 합니다(즉, idShort는 SubmodelElementList 및 모든 식별 가능 항목을 제외한 모든 참조 가능 항목에 대해 필수입니다).

제약 AASd-120: SubmodelElementList 내의 하위 모델 요소의 idShort는 지정되지 않아야 합니다.

제약 AASd-022: 식별 불가능한 참조 대상의 idShort는 해당 네임스페이스에서 고유해야 합니다.

제약 AASd-003: 참조 가능 항목의 idShort는 대소 문자를 구분하여 일치해야 합니다.

### 5.7.13.3 예선에 대한 제약

제약 조건 AASd-021: 모든 한정자는 동일한 한정자/유형을 가진 하나의 한정자만 가질 수 있습니다.

제약 AASd-119: Qualifier/qualifier의 Qualifier/kind 값이 TemplateQualifier와 같은 한정된 요소가 "hasKind"에서 상속되는 경우 한정된 요소는 템플릿 종류 (HasKind/kind = "Template")여야 합니다.

### 5.7.13.4 확장에 대한 제약

제약 조건 AASd-077: HasExtensions 내의 확장 이름은 고유해야 합니다.

### 5.7.13.5 자산 관련 정보에 대한 제약

제약 조건 AASd-116: "globalAssetId" (대소문자 구분 안 함)는 예약된 키입니다. SpecificAssetId/name에 대한 값으로 사용되는 경우 SpecificAssetId/value는 AssetInformation/globalAssetId와 동일해야 합니다.

### 5.7.13.6 유형에 대한 제약 조건

제약 조건 AASd-100: 데이터 유형이 "문자열"인 속성은 비워둘 수 없습니다.