7.1 일반

AAS는 데이터 액세스의 중심점이므로 여러 역할을 지원하는 세분화된 액세스 제어와 AAS의 개별 노드 또는 하위 모델에 대한 별도의 액세스 제어 정책을 지원할 필요가 있습니다. 접근 제어는 ID 관리를 기반으로 하며 안전한 환경에서만 성공적으로 구현될 수 있습니다. 이러한 측면과 데이터 사용 제어 및 데이터 출처 추적을 지원하는 개념은 앞으로 더 개발될 예정이므로 이 장에서 자세히 설명하지 않습니다. 이 문서에서는 지원되는 액세스 제어 모델에 중점을 둡니다.

7.2 접근 권한 전달

주요 그림(4.2절의 그림 2)을 볼 때 한 가치 사슬 파트너에서 다음 가치 사슬 파트너로 정보를 전송할 때 보안 측면도 고려해야 합니다.

AAS 콘텐츠가 한 파트너에서 다른 파트너로 전달될 때 이는 일반적으로 관련된 파트너(공급자, 통합자, 운영자)의 액세스 제어 도메인 변경, 즉 액세스 제어 정책의 유효성 범위와 관련됩니다.

따라서 공급자가 데이터를 통합자에게 전달하는 예의 경우 다음과 같은 일반적인 단계가 수행됩니다.

- A1-A2 단계: 공급자는 전달할 데이터를 선택하고(10절 참조)
  AASX 패키지의 내용을 결정합니다(8절 참조). • A2-A3 단계: AASX 패키지가 통합자로
  전송됩니다. • A3-A4 단계: 통합자는 패키지를 수신하고 콘텐츠를 자신의 보안 도메인으로 가져옵
  니다.
  이 단계에서 통합자는 자체 보안 도메인의 요구 사항에 따라 액세스 권한을 설정해야 합니다.

ABAC는 역할 기반 액세스를 포함하는 매우 유연한 접근 방식으로, 이 컨텍스트에서 역할을 하나의 속성으로 간주할 수 있습니다. 다른 속성으로는 시간, 자산 위치, 발신 주소 등이 있습니다.

AAS 콘텐츠 자체 외에도 다음 두 가지 이유로 인해 정의된 액세스 권한도 파트너 간에 전송되어야 합니다.

(1) AAS의 정보 요소에 대한 액세스 권한은 각 액세스 제어에서 설정되어야 합니다.
도메인.

(2) 한 파트너는 어떤 액세스 권한을 설정해야 하는지 제안을 통과할 수 있어야 합니다.
AAS에 기술된 자산.

요구 사항 (2)의 예:

로봇 제조업체는 로봇에 대해 기계 설정자, 작업자 및 유지 보수 역할과 같은 역할을 정의해야 한다고 제안합니다. 역할은 로봇을 나타내는 AAS의 속성을 통해 표현되어야 합니다. 그는 또한 이러한 역할에 대한 권한을 제안합니다. 예를 들어 설치자(통합자)는 로봇 프로그램에 대한 쓰기 액세스 권한이 있지만 작업자는 그렇지 않습니다.

위의 예는 액세스 권한 규칙이 하나의 액세스 제어에서 전달되어야 한다는 동기를 부여합니다.
도메인을 다른 사람에게. 액세스 권한 규칙의 전달은 다음과 같은 방법으로 구현됩니다.

- 접근 권한 정의: 세부 접근 권한(예: 읽기, 쓰기, 삭제, 생성, 호출 메서드 등)은 도메인별 하위 모델에서 정의됩니다(AccessControl/
  defaultPermissions 참조).
  및 7.4.4절의 AccessControl/selectablePermissions).
- 정의된 액세스 권한에 기반한 액세스 권한 규칙의 정의. 이는 접근 제어의 일부로 정의됩니다(7.4.4절 참조).

- AAS의 각 정보 요소(객체)에 대한 액세스 권한 규칙의 연결. 이 수단은 AAS 자체의 정보 구조에 의해 실현됩니다(7.4.5절의 PermissionsPerObject 참조).

유효 접근 권한은 접근 권한 규칙에 따라 결정됩니다. AAS의 각 하위 모델 요소에는 각 주제에 대한 액세스 권한을 정의하는 규칙이 있어야 합니다. 주체는 이미 인증된 것으로 간주됩니다.

하위 모델 요소에 이러한 규칙이 없으면 포함된 요소에 대한 테이블을 자동으로 사용합니다("위에서 상속"). 최상위 개체는 AAS 자체입니다. 즉, AAS는 상속을 위한 시작점입니다. .

이전에 표시된 대로 주체 식별, 규칙 정의 및 권한은 수신 당사자가 다른 액세스 제어 도메인에 있을 수 있기 때문에 다를 수 있습니다. 수신 당사자가 A3-A4 단계에서 액세스 권한을 설정할 때 전달된 액세스 정의(권한 및 액세스 권한 규칙)를 액세스 제어 도메인의 기존 정의에 병합해야 합니다.

[19] 예제와 속성 액세스 제어 및 일반적인 액세스 제어에 대한 추가 배경 정보를 찾을 수 있습니다. 클래스와 그 속성은 7.4절에 정의되어 있습니다.

7.3 관리 셸 wrt 보안의 개요 메타 모델

액세스 제어와 관련된 보안 관련 속성은 AAS 정보 모델의 일부입니다. 액세스 제어의 목적은 무단 액세스로부터 시스템 리소스(여기서는 AAS 콘텐츠)를 보호하는 것입니다. 보호 조치는 액세스 제어 전용 보안 도메인에 의해 유효성 범위가 정의되는 액세스 제어 정책에 지정됩니다.

참고: I4.0 시스템에서 액세스 제어를 구현하려면 ID 관리, 디지털 인증서 관리, 인증 및 액세스 제어 시행과 같은 전용 인프라 서비스의 지원이 필요합니다.

액세스 제어에 적용되는 기본 개념은 ABAC(속성 기반 액세스 제어)의 개념입니다. 일반적으로 ABAC 요청 흐름은 [22]에 설명되어 있습니다. 원래 ABAC는 데이터 흐름 모델과 OASIS XACML(eXtensible Access Control Markup Language) 사양의 언어 모델에 의존합니다[54].

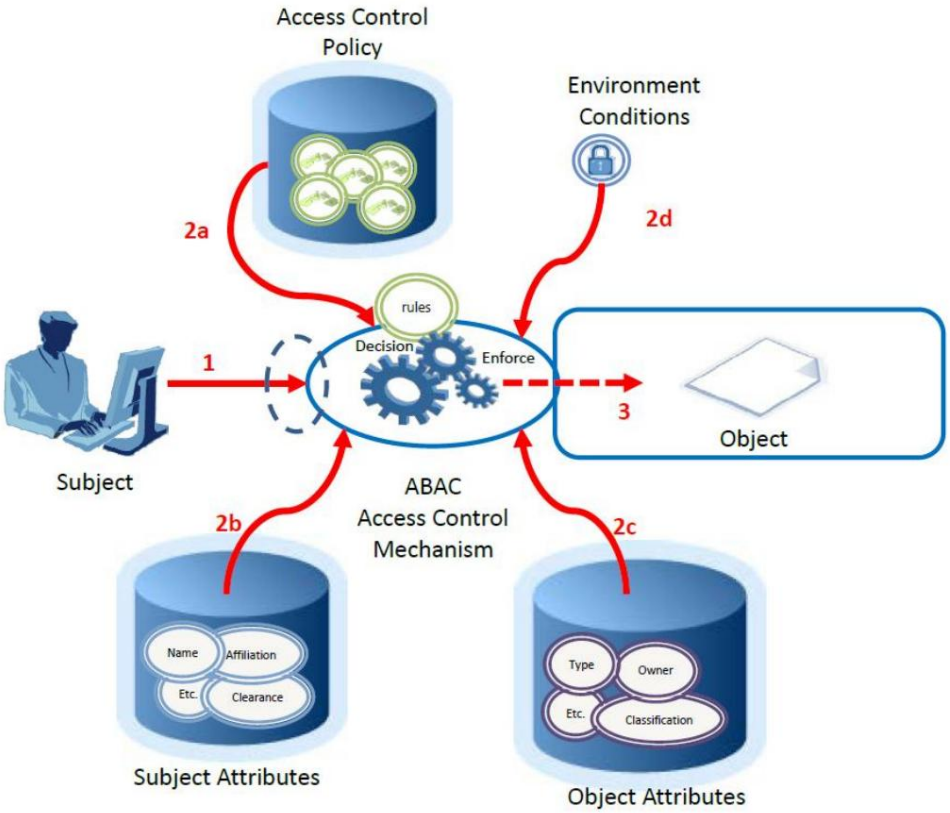
OASIS XACML에는 다음과 같은 개념이 포함됩니다.

- 정책 관리 지점(PAP): 정책 집합을 생성하는 시스템 개체입니다.
- 정책 결정 지점(PDP): 해당 정책을 평가하고 권한 부여 결정을 내리는 시스템 엔터티입니다.
- PEP(Policy Enforcement Point): 의사결정을 통해 접근통제를 수행하는 시스템 개체 요청 및 승인 결정 시행.
- 정책 정보 포인트(PIP): 속성 값의 소스 역할을 하는 시스템 엔터티.

일반적인 요청 흐름은 그림 65에 나와 있습니다.

- 주체가 객체(1)에 대한 액세스를 요청하고 있습니다. AAS의 맥락에서 객체는 일반적으로 자산에 연결된 하위 모델 또는 속성 또는 기타 하위 모델 요소.
- AAS의 구현된 액세스 제어 메커니즘은 주체 속성(2b), 객체 속성(2c) 및 환경 조건(2d)에 대해 충족되어야 하는 제약 조건을 포함하는 액세스 권한 규칙(2a)을 평가합니다.
- 평가 후 객체(3)에 대한 결정이 내려지고 시행됩니다. 즉, 하위 모델 요소에 대한 액세스가 허용되거나 거부됩니다.

그림 65 속성 기반 접근 제어 [22]



참고: ABAC 컨텍스트의 속성은 메타모델에 정의된 요소의 속성과 다릅니다.

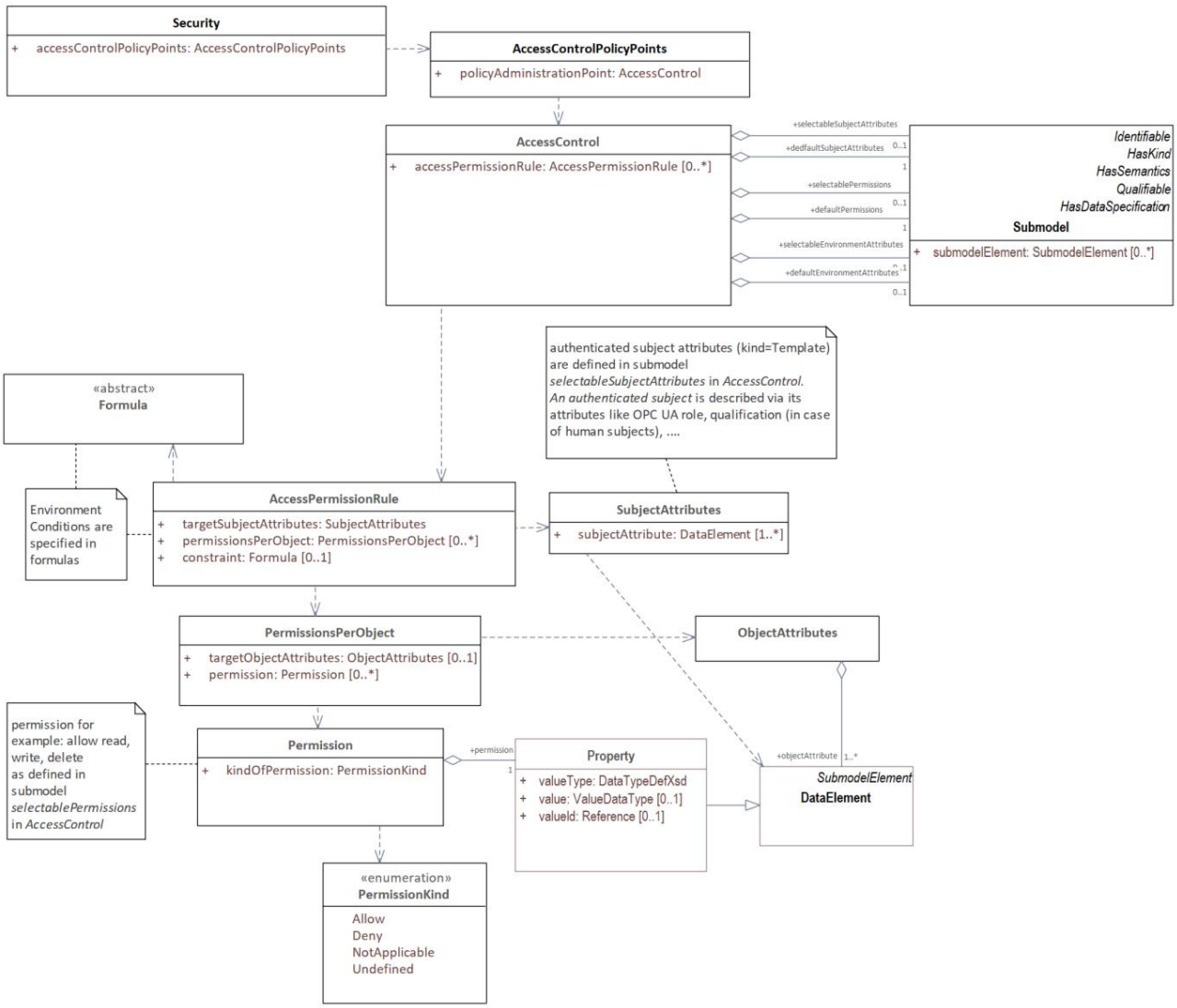
그림 66에는 AAS wrt 보안 측면의 정보 모델에 대한 개요가 나와 있습니다.

ABAC 컨텍스트의 개체는 일반적으로 하위 모델 또는 하위 모델 요소에 해당합니다. 개체 속성은 다시 하위 모델 요소로 모델링됩니다.

주제 속성은 외부 PIP(정책 정보 포인트)를 통해 액세스하거나 AAS의 특수 하위 모델 내에서 속성으로 정의됩니다. 일반적인 주제 속성은 역할입니다. 역할은 ABAC가 역할 기반 접근 제어로 적용될 때 정의되는 유일한 주제 속성입니다.

선택적으로 환경 조건을 정의할 수 있습니다. 역할 기반 액세스 제어에서는 환경 조건이 정의되지 않습니다. 환경 조건은 공식 제약 조건을 통해 표현할 수 있습니다. 그렇게 하려면 필요한 값을 AAS의 하위 모델 내에서 데이터에 대한 참조 또는 속성으로 정의해야 합니다.

그림 66 접근 제어를 위한 메타모델 개요



액세스 제어 정책(예: 액세스 권한 규칙 측면에서)을 통해 AAS 내에서 어떤 주체가 어떤 객체35에 액세스할 수 있는지가 정의됩니다 . 주체가 이미 인증된 것으로 가정합니다.

개체는 참조 가능한 모든 요소가 될 수 있습니다. 즉, 하위 모델 및 개념 설명과 같은 식별 가능한 요소를 포함합니다. 보다 일반적으로 인증된 주체가 객체에 대한 액세스를 허용 또는 거부하는지 여부를 지정할 수 있으므로 "액세스"는 AAS 요소에 대해 지정된 권한 중 하나일 수 있습니다. 선택할 수 있는 권한은 AAS의 메타모델에 의해 정의되지 않습니다. **선택 가능한 권한은 하위 모델** (AccessControl/selectablePermissions) 을 통해 정의됩니다 . 속성 (AccessControl/selectableSubjectAttributes). 기본 주제 속성 및 기본 권한은 AAS 소유자가 덮어쓰지 않는 경우 사용됩니다. 권한에 관해서는 사용된 인증된 주제 속성이 하위 모델 AccessControl/selectableSubjectAttributes에 정의되어 있습니다. 동일하게 유지 ~을 위한 주제

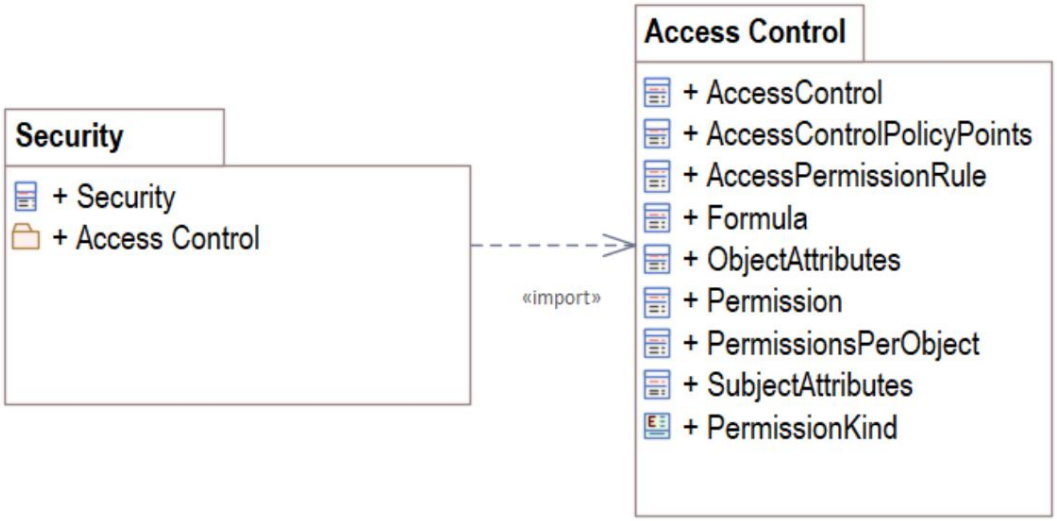
접근 권한이 더 제한될 수 있습니다. 예를 들어, 정책 규칙은 "유지 관리 엔지니어" 역할(더 정확하게는 "역할 = '유지 관리 엔지니어'"라는 주체 속성을 가진 인증된 주체)이 기계(자산) 실행되지 않습니다. "상태" 속성을 기반으로 하는 이 액세스 규칙의 형식적인 표현은 5.7.2.7절의 그림 73을 참조하십시오.

35 "객체"라는 용어는 더 일반적이고 미래에는 예를 들어 클래스의 속성과 같은 다른 객체도 요소 외에 포함될 수 있기 때문에 사용됩니다.

개체 속성은 다른 방식으로 처리됩니다. 초점에 있는 개체의 모든 속성이 개체 속성의 역할을 추가로 인수할 수 있다고 가정합니다. 따라서 기본 또는 선택 가능한 개체 속성에 대한 특별한 하위 모델이 없습니다.

그림 67은 메타모델의 보안 문제에 대해 정의된 모든 요소에 대한 개요를 제공합니다.

그림 67 보안 개요 패키지



7.4 메타모델 사양 세부사항: 지정자

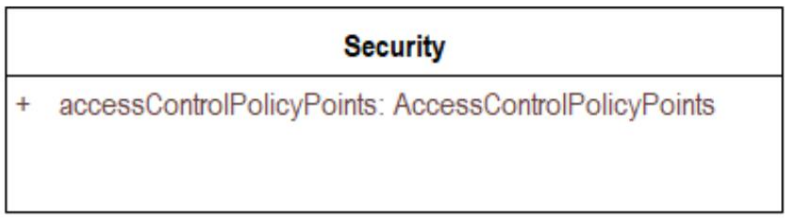
7.4.1 소개

이 절에서는 보안과 관련된 메타 모델의 클래스를 자세히 지정합니다. 이는 5.7절에 설명된 메타모델의 확장입니다.

확장을 이해하려면 기본 및 공통 추상 클래스를 이해해야 합니다(특히 5.7.2절, 5.7.9절 및 5.7.10.4절 참조).

7.4.2 보안 속성

그림 68 AAS의 보안속성 메타모델



수업:	보안		
설명:	AAS의 보안 관련 정보를 담는 컨테이너.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
accessControlPolicyPoints	AAS의 액세스 제어 정책 지점.	AccessControlPolicyPoi NTS	1

7.4.3 접근 제어 정책 포인트 속성

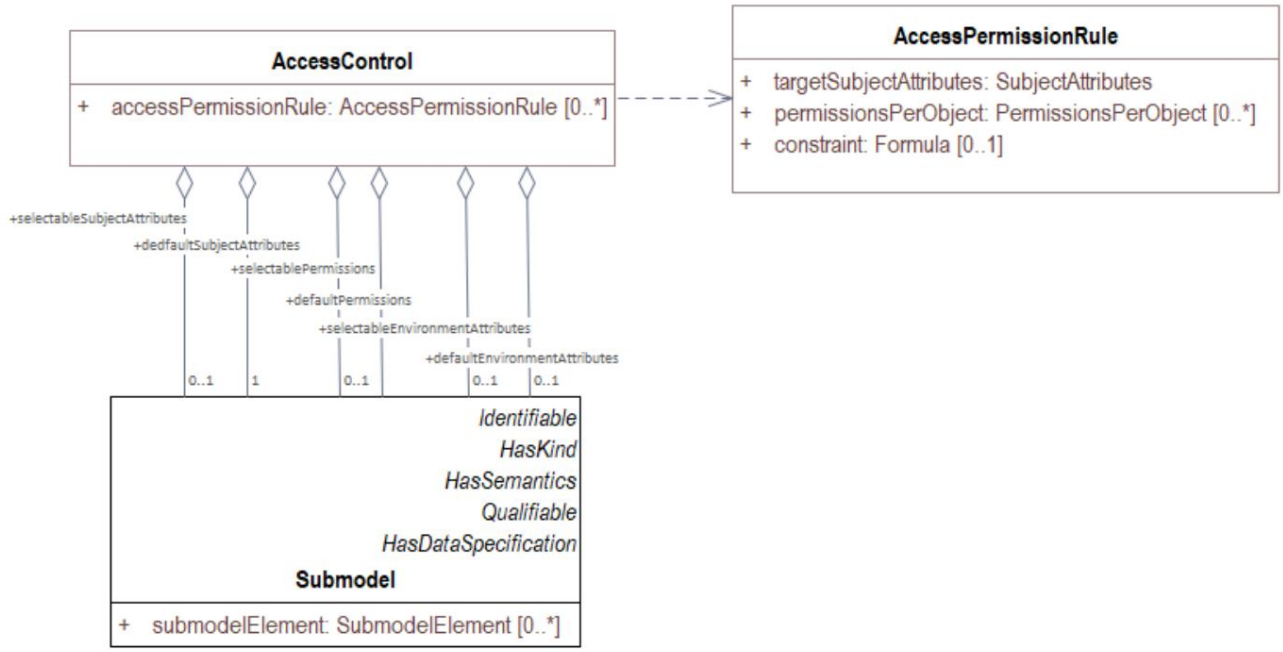
그림 69 액세스 제어 정책 포인트의 메타 모델

수업:	AccessControlPolicyPoints		
설명:	액세스 제어 정책 지점에 대한 컨테이너입니다.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
policyAdministrationPoint	AAS의 액세스 제어 관리 정책 지점입니다.	액세스 제어	1

Policy Administration point의 정의는 [22]에서 가져왔다. PAP는 정책 관리를 담당하며 정책 자체에 대한 액세스 제어도 포함합니다. 액세스 제어 결정을 평가하기 위해 정책이 PDP에 배포됩니다.

7.4.4 접근 제어 속성

그림 70 접근통제 메타모델

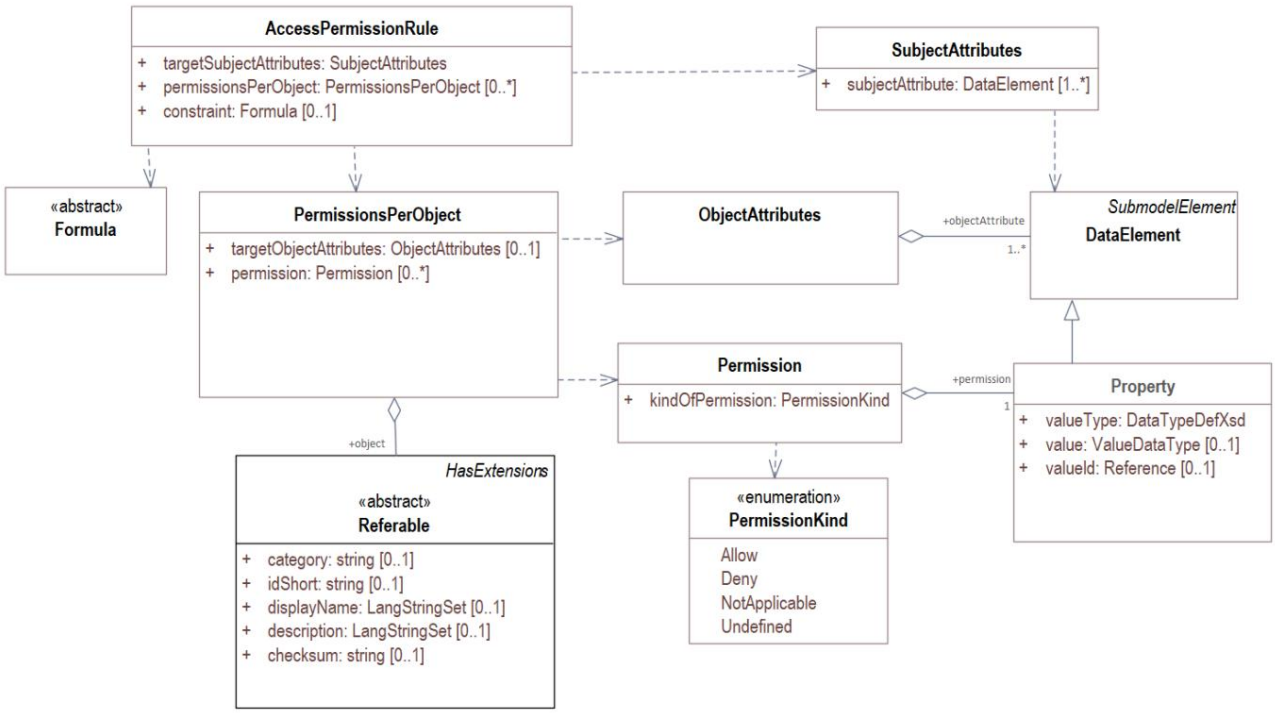


수업:	액세스 제어		
설명:	액세스 제어는 로컬 액세스 제어 정책 관리 지점을 정의합니다. 접근 제어는 접근 권한 규칙을 정의하는 주요 작업이 있습니다.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
액세스 권한 규칙	AAS의 액세스 요소에 대해 (이미 인증된) 주체에게 할당된 권한을 설명하는 AAS의 액세스 권한 규칙.	액세스 권한 규칙 0..*	
selectableSubject속성	AAS에 대해 구성된 인증된 주제를 정의하는 하위 모델에 대한 참조입니다. 접근 권한 규칙에 따라 선택하여 주체에 권한을 할당할 수 있습니다. 기본값: 하위 모델에 대한 참조 참조 -을 통해 defaultSubject속성.	ModelReference<하위 모델>	0..1
defaultSubject속성	액세스 권한 규칙을 설명하는 데 사용할 수 있는 AAS에 대한 기본 주체의 속성을 정의하는 하위 모델에 대한 참조입니다. 하위 모델은 종류=템플릿입니다.	ModelReference<하위 모델>	1

수업:	액세스 제어		
선택 가능한 권한	주제에 할당할 수 있는 권한을 정의하는 하위 모델에 대한 참조입니다. 기본값: 하위 모델에 대한 참조 defaultPermissions 를 통해 참조 됨	ModelReference<하위 모델>	0..1
기본 권한	AAS에 대한 기본 권한을 정의하는 하위 모델에 대한 참조입니다.	ModelReference<하위 모델>	1
selectableEnvironmentAttributes AAS에	대해 정의된 권한 규칙을 통해 액세스할 수 있는 환경 속성, 즉 자산 자체를 설명하지 않는 속성을 정의하는 하위 모델에 대한 참조입니다. 기본값: 참조된 하위 모델에 대한 참조 ~을 통해 기본 환경 속성	ModelReference<하위 모델>	0..1
기본 환경 속성	기본 환경 속성, 즉 자산 자체를 설명하지 않는 속성을 정의하는 하위 모델에 대한 참조입니다. 하위 모델은 종류=템플릿입니다. 동일한 유형에서 이러한 환경 속성의 값은 다음과 같아야 합니다. 액세스 권한 규칙을 평가할 때 액세스할 수 있습니다. 이것은 정책 정보 포인트로 구현됩니다.	ModelReference<하위 모델>	0..1

7.4.5 접근 권한 규칙 속성

그림 71 접근 권한 규칙의 메타 모델



수업:	액세스 권한 규칙		
설명:	개체 집합(참조 가능한 요소)에 대해 인증된 주체별 액세스 권한을 정의하는 테이블입니다.		
다음에서 상속됨:			
기인하다	설명	유형	카드.
targetSubject속성	이 규칙에 정의된 권한을 얻기 위해 액세스하는 주체가 충족해야 하는 대상 주체 속성입니다.	주제 속성	1
권한별 개체	액세스 권한 규칙 내에서 개체당 권한을 정의하는 개체-권한 쌍의 집합입니다.	개체별 권한 CT	0..*
강제	해야 하는 제약 액세스할 수 있도록 true로 검증 권한 규칙이 유지됩니다.	공식	0..1

수업:	권한별 개체		
설명:	지정된 개체에 대한 액세스 권한을 정의하는 테이블입니다. 개체는 AAS에서 참조 가능한 요소입니다. 또한 권한이 적용되는 개체의 종류를 추가로 지정하는 개체 속성을 정의할 수 있습니다.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.

수업:	권한별 개체		
물체	권한이 할당될 요소입니다.	ModelReference<참조 가능> 1	
targetObjectAttributes 액세스가 가능하도록 충족되어야 하는 대상 개체 속성 접근하는 주체에 권한이 적용됩니다.		개체 속성	0..1
허가	개체에 할당된 권한입니다. 액세스 권한 규칙에 지정된 대로 모든 주체에 대한 권한이 유지됩니다.	허가	0..*

수업:	개체 속성		
설명: 개체 속성을 정의하는 데이터 요소 집합입니다. 이러한 속성은 다음을 참조해야 합니다. 기존 하위 모델 내의 데이터 요소.			
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
objectAttribute 추가	데이터 요소에 대한 참조 개체를 분류합니다.	ModelReference<DataElement> 1..*	

수업:	허가		
설명:	단일 권한에 대한 설명입니다.		
다음에서 상속됨:	--		
기인하다	설명	유형	카드.
허가	권한의 의미를 정의하는 속성에 대한 참조입니다.	ModelReference<속성>	1
kindOfPermission 권한	종류에 대한 설명입니다. 가능한 종류의 허가에는 허가 거부도 포함됩니다. 값: • 허용하다 • 부인하다 • 해당 없음 • 한정되지 않은	권한 종류	1

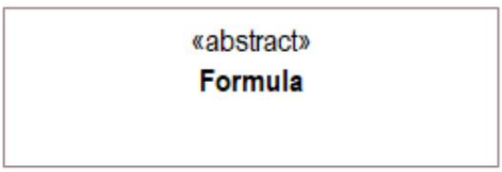
수업:	주제 속성
설명:	특정 주제를 추가로 분류하는 데이터 요소 집합입니다.
다음에서 상속됨:	--

수업:	주제 속성		
기인하다	설명	유형	카드.
subjectAttribute 특정	주제를 추가로 분류하는 데이터 요소입니다.	데이터 요소	1..*

열거:	권한 종류
설명:	주체에 대한 권한 할당에 부여되는 권한 종류의 열거입니다.
세트:	--
정확한	설명
허용하다	주제에 부여된 권한을 허용합니다.
부인하다	주체에게 주어진 권한을 명시적으로 거부합니다.
해당 없음	권한이 주제에 적용되지 않습니다.
한정되지 않은	권한이 허용되는지, 대상에 적용되지 않거나 거부되는지 여부는 정의되지 않습니다.

7.4.6수식 속성

그림 72 수식의 메타모델



수식은 논리식에 사용되는 참조 가능 항목에 따라 달라질 수 있습니다.

참조된 요소의 값은 수식에서 true로 평가될 수 있도록 액세스 가능해야 합니다.
또는 그것이 사용되는 해당 논리 표현식에서 false입니다.

그림 73에는 "상태" 속성에 따른 수식의 예가 나와 있습니다. 그러나 지금까지 AAS에 대해 공식 언어가 정의되지 않았으므로 예제는 규칙을 공식화하는 데 필요한 속성 값을 설명하기 위해 xsd 혼합 콘텐츠 유형과 속성을 사용하는 가능성 중 하나를 보여줍니다.

참고: 이 예시적인 메커니즘을 사용하면 지금까지는 복잡한 개체(예: 하위 모델 요소 컬렉션 또는 관계 요소)를 포함하는 공식을 공식화할 수 없습니다. 데이터 요소
또는 기타 요소로 제한됩니다.
사용 가능하고 정의된 문자열로 직렬화가 있습니다.

그림 73 "Machine Status not Running" 공식 예(비규범)

```
<aas: 공식>
  <aas:DependsOn>
    <카>
      <Key type="Submodel">https://myShell/Machine</Key>
      <Key type="Property">상태</Key> </Keys>
  </aas:dependsOn> ⊨ 실행 중
</ aas: 공식>
```

수업:	수식 <<추상>>		
설명: 수식은 논리 표현식으로 제약 조건을 설명하는 데 사용됩니다.			
다음에서 상속됨:			
기인하다	설명	유형	카드.

7.4.7 교차 제약과 불변

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.

제약 조건 AASs-010: 허가/허가 에 언급된 속성 은 "CONSTANT" 범주를 가져야 합니다.

제약 조건 AASs-011: 권한/허가 에서 참조된 속성은 "AccessControl" 의 "selectablePermissions" 속성 내에서 참조되는 하위 모델의 일부여야 합니다 .

제약 AASs-015: SubjectAttributes/subjectAttributes 의 모든 데이터 요소는 "AccessControl" 의 "selectableSubjectAttributes" 속성 내에서 참조되는 하위 모델의 일부여야 합니다 .