



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2015-0136213

(43) 공개일자 2015년12월07일

(51) 국제특허분류(Int. Cl.)

H04L 12/12 (2006.01)

(21) 출원번호 10-2014-0063240

(22) 출원일자 2014년05월26일

심사청구일자 없음

(71) 출원인

전자부품연구원

경기도 성남시 분당구 새나리로 25 (야탑동)

세종대학교산학협력단

서울특별시 광진구 능동로 209 (군자동, 세종대학교)

(72) 발명자

최성찬

서울 동작구 노량진로24길 2, 102동 1306호 (본동, 한강쌍용아파트)

송재승

서울 강서구 강서로 348, 110동 902호 (내발산동, 우장산힐스테이트)

(뒷면에 계속)

(74) 대리인

남충우, 노철호

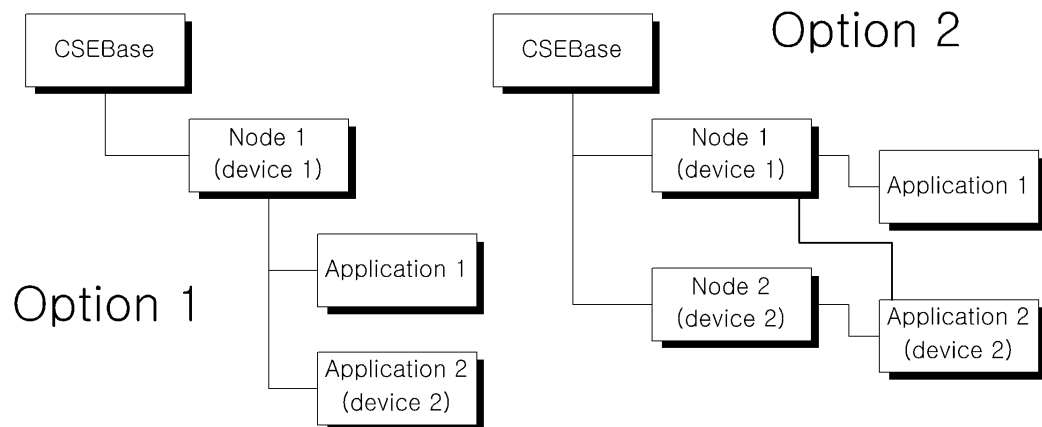
전체 청구항 수 : 총 1 항

(54) 발명의 명칭 IoT 디바이스 페어링 방법

(57) 요약

본 발명의 실시예에 따른 IoT 디바이스 페어링 방법은, IoT 플랫폼 내에서 링크된 디바이스를 나타낼 수 있는 등록 절차를 제공한다. 이를 위해, 새로운 속성 또는 새로운 리소스 유형을 도입하여, 두 개의 리소스들을 링크시킨다. 이에, 페어링 된 디바이스 ID와 함께 측정 데이터를 전송한다.

대표도 - 도3



(72) 발명자

원광호

경기 용인시 수지구 신봉3로 25, 313동 202호 (신봉동, 신봉마을동일하이빌3차)

김재호

경기 용인시 수지구 대지로15번길 60, 507동 1501호 (죽전동, 대지마을현대홈타운3차2단지)

이상신

경기도 용인시 기흥구 죽현로 12, 312동 902호 (보정동 죽현마을 동원로얄듀크)

안일엽

경기 용인시 수지구 탄천상로 7, 104동 1001호 (죽전동, 인현마을힐스테이트7차아파트)

윤재석

경기 용인시 기흥구 기흥로 29, 107동 408호 (구갈동, 한성1차아파트)

송민환

서울 광진구 뚝섬로40길 45, 103동 1402호 (자양동, 삼성아파트)

이 발명을 지원한 국가연구개발사업

과제고유번호 10041262

부처명 미래창조과학부

연구관리전문기관 한국산업기술평가관리원

연구사업명 SW컴퓨팅산업융합원천기술개발사업-차세대플랫폼기술개발사업

연구과제명 사물인터넷 글로벌 생태계 구축 및 서비스 활성화를 위한 개방형 IoT 소프트웨어 플랫폼

개발

기 여 율 1/1

주관기관 전자부품연구원

연구기간 2013.12.01 ~ 2014.11.30

명세서

청구범위

청구항 1

제1 디바이스를 등록하는 단계;

제2 디바이스를 등록하는 단계; 및

상기 제1 디바이스와 상기 제2 디바이스를 페어링하는 단계;를 포함하는 것을 특징으로 하는 디바이스 페어링 방법.

발명의 설명

기술 분야

[0001] 본 발명은 IoT(Internet of Things) 관련 기술에 관한 것으로, 더욱 상세하게는 IoT 디바이스 관리를 위한 M2M(Machine to Machine) 기법에 관한 것이다.

배경 기술

[0002] 널리 사용되는 IoT/M2M 응용 중 하나는 디바이스(예를 들면, TV, 선풍기, 냉장고 등)의 에너지 소비 측정이다. 디바이스들은 자신의 에너지 사용량을 측정할 수 없기 때문에, 에너지 사용량 측정을 위한 별도의 디바이스가 사용된다.

[0003] 그러나, 현재 IoT/M2M은 디바이스로 두 개의 다른 디바이스를 링크시키기 위한 메커니즘을 지원하지 않는다. 여기서, 링크는 두 개의 디바이스들이 동일한 물리적 디바이스를 나타내도록 하는 것을 의미한다.

발명의 내용

해결하려는 과제

[0004] 본 발명은 상기와 같은 문제점을 해결하기 위하여 안출된 것으로서, 본 발명의 목적은, 여러 디바이스들이 IoT 플랫폼에 등록되어 있는 상황에서, 사용자가 오직 하나의 디바이스로부터 여러 디바이스들의 데이터를 수신할 수 있도록 하는 IoT 디바이스 페어링 방법을 제공함에 있다.

과제의 해결 수단

[0005] 상기 목적을 달성하기 위한 본 발명의 일 실시예에 따른, IoT 디바이스 페어링(가상 연결) 방법은, IoT 플랫폼 내에서 링크된 디바이스를 나타낼 수 있는 등록 절차를 제공한다. 이를 위해, 본 발명의 실시예에 따른 IoT 디바이스 페어링(가상 연결) 방법은, 새로운 속성 또는 새로운 리소스 유형을 도입하여, 두 개의 리소스들을 링크시킨다. 이에, 페어링 된 디바이스 ID와 함께 측정 데이터를 전송한다.

발명의 효과

[0006] 이상 설명한 바와 같이, 본 발명의 실시예들에 따르면, 여러 디바이스들이 IoT 플랫폼에 등록되어 있는 상황에서, 사용자가 오직 하나의 디바이스로부터 여러 디바이스들의 데이터를 수신할 수 있게 된다.

도면의 간단한 설명

- [0007] 도 1은 본 발명이 적용가능한 M2M 아키텍처를 도시한 도면,
 도 2는 본 발명의 실시예에 따른 IoT 디바이스 페어링(가상 연결) 방법의 설명에 제공되는 순서도,
 도 3은 다양한 옵션들에 따른 페어링 결과들을 나타낸 도면, 그리고,
 도 4는 페어링 링크 유형들을 예시한 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0008] 이하에서는 도면을 참조하여 본 발명을 보다 상세하게 설명한다.
- [0009] 도 1은 본 발명이 적용가능한 M2M 아키텍처를 도시한 도면이다. 도 1에 도시된 바와 같이, 네트워크는 다수의 노드들로 구성되는데, 이 노드들은 AE(Application Entity) 및/또는 CSE(Common Service Entity)을 보유한다.
- [0010] 이때, 노드는 적어도 하나의 AE를 보유할 수 있다. 또한, 노드는 CSE를 보유하거나 보유하지 않을 수 있다. CSE를 보유하지 않는 노드는, ADN(Application Dedicated Node : 어플리케이션 전용 노드)으로 지칭한다. 한편, CSE를 보유하는 경우, 1개의 CSE를 보유하는 것이 일반적이겠지만, 반드시 1개로 한정되는 것은 아니다.
- [0011] CSE는 기능 모듈로, 디바이스와 리소스 관리를 포함하여, 네트워크, 보안, 등록/검색, 가입, 과금 등의 다양한 서비스를 제공한다.
- [0012] 노드들은, ADN, ASN(Application Service Node : 어플리케이션 서비스 노드), MN(Middle Node : 중간 노드) 및 IN(Infrastructure Node : 인프라 노드)로 분류된다. MN은 게이트웨이로 기능하는 노드이고, IN은 서버/클라우드 노드로 기능하는 노드로 이해할 수 있다.
- [0013] 이하에서, IoT 디바이스 페어링(가상 연결) 방법에 대해, 바람직한 실시예들을 들어 상세히 설명한다. 한편, 본 발명의 실시예에서는 디바이스와 노드가 혼용되는데, 양자를 동일한 것으로 이해하여도 무방하다.
- [0014] 본 발명의 실시예에 따른 IoT 디바이스 페어링(가상 연결) 방법은, IoT 플랫폼 내에서 링크된 디바이스를 나타낼 수 있는 등록 절차를 제공한다. 이를 위해, 본 발명의 실시예에 따른 IoT 디바이스 페어링(가상 연결) 방법은, 새로운 속성 또는 새로운 리소스 유형을 도입하여, 두 개의 리소스들을 링크시킨다. 이에, 페어링된 디바이스 ID와 함께 측정 데이터를 전송한다.
- [0015] 디바이스 페어링이 이루어지면, 페어링된 다수의 디바이스들 중 하나가 사용자들 또는 다른 디바이스들에 대해, 다수의 디바이스를 대표한다. 이에, IoT/M2M 관리자 또는 사용자는 두 개의 서로 다른 물리적 디바이스들로부터 전달되는 데이터를 수동으로 설정할 필요가 없다. 동일한 디바이스에서 생성된 데이터로 취급되기 때문이다.
- [0016] 도 2는 본 발명의 실시예에 따른 IoT 디바이스 페어링(가상 연결) 방법의 설명에 제공되는 순서도이다.
- [0017] 도 2에 도시된 바와 같이, 먼저, 신규 디바이스인 디바이스-1이, 등록을 위해, 자신의 로컬 IN-CSE에 등록 요청(CREATE)을 전송한다(단계 001).
- [0018] 그러면, 호스팅 IN-CSE은, 디바이스-1이 등록(CREATE)에 대한 권한이 있는지 확인하고, 권한이 있는 경우 디바이스-1에 대한 신규 리소스를 생성한다(단계 002).
- [0019] 이후, 호스팅 IN-CSE는 요청에 대한 응답으로, 디바이스-1이 네트워크에 성공적으로 등록되었음을 안내한다(단계 003). 그리고, 호스팅 IN-CSE는 다바이스들로부터 추가적인 입력을 기다린다(단계 004).
- [0020] 다음, (신규) 디바이스-2는, 등록을 위해, 자신의 로컬 CSE에 등록(CREATE) 요청 메시지를 전송한다(단계 005). 이 메시지는 디바이스 페어링을 위한 메시지이다. 이 메시지는 또한 디바이스-2가 페어링 등록할 타겟인 디바이스-1의 ID를 포함한다.
- [0021] 한편, 디바이스-1의 ID는 다음과 같은 메커니즘으로 획득한다.
- [0022] 1) 디바이스-2에 디바이스-1의 ID를 사전에 제공

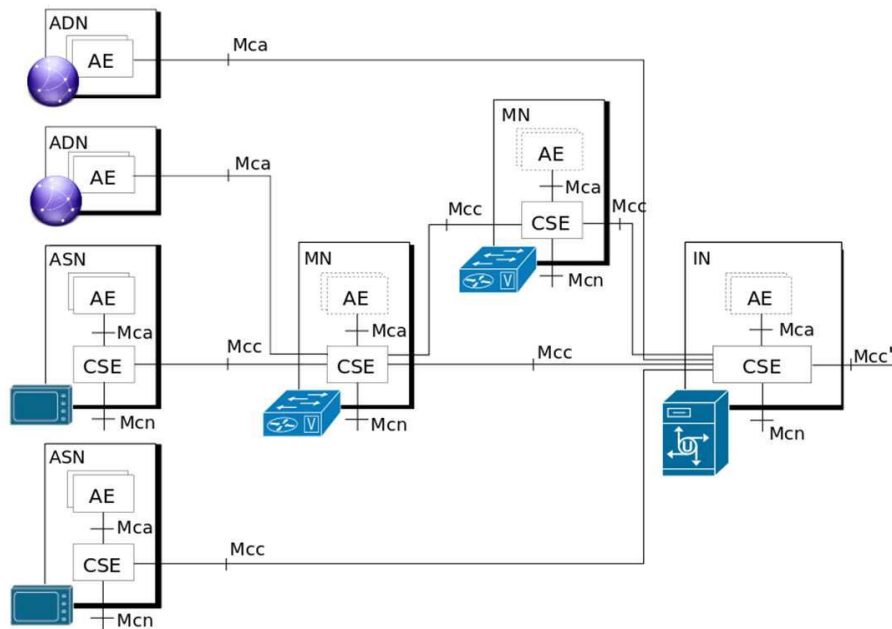
- [0023] 2) 디바이스-1과 디바이스-2 간에 별도의 교환 메커니즘을 통해 획득
- [0024] 예를 들어, 디바이스-2가 페어링을 위해 필요한 디바이스 ID를 획득하기 위해, 디바이스-1에 요청 메시지 전송하여, 디바이스-1의 ID 획득
- [0025] 이후, 호스팅 IN-CSE은, 페어링 등록 요청이 있는지 확인한다(단계 006). 요청이 있으면, 호스팅 IN-CSE은, 디바이스-2의 신규 어플리케이션/리소스를 생성하는 대신, 디바이스-2의 어플리케이션/리소스를 디바이스-1의 추가 어플리케이션/리소스로 추가한다.
- [0026] 호스팅 IN-CSE는 이전에 어플리케이션/리소스가 알려진 모든 IN-CSE들에게 어플리케이션/리소스 업데이트를 안내한다.
- [0027] 한편, 단계 006는 다음과 같이 구현될 수 있으며, 각각에 대한 페어링 결과를 도 3에 도시하였다.
- [0028] 옵션 1 : 디바이스-2의 어플리케이션/리소스를 디바이스-1의 어플리케이션/리소스 내에 추가 어플리케이션/리소스로 추가(도 3의 좌측)
- [0029] 옵션 2 : 디바이스-2에 대한 별도의 어플리케이션/리소스를 생성하고, 디바이스-1로의 링크를 생성(도 3의 우측)
- [0030] 도 4에는 페어링 링크 유형을 예시하였다. 디바이스 페어링을 위한 추가 링크는 속성이나 리소스로 도입가능하다.
- [0031] 이후, 데이터 전송에는 페어링 된 디바이스 ID와 함께 데이터 전송이 이루어진다. 그리고, 위 옵션 2의 경우(별도의 리소스를 생성하고, 페어링 된 디바이스로 링크시키는 경우), 데이터는 디바이스-2로부터 데이터를 요청한 디바이스로 전달된다. 한편, IN-CSE의 기능으로, 메시지의 소스 ID를 페어링 된 디바이스 ID로 교체하는 것도 가능하다.
- [0032] 또한, 이상에서는 본 발명의 바람직한 실시예에 대하여 도시하고 설명하였지만, 본 발명은 상술한 특정의 실시예에 한정되지 아니하며, 청구범위에서 청구하는 본 발명의 요지를 벗어남이 없이 당해 발명이 속하는 기술분야에서 통상의 지식을 가진자에 의해 다양한 변형실시가 가능한 것은 물론이고, 이러한 변형실시들은 본 발명의 기술적 사상이나 전망으로부터 개별적으로 이해되어져서는 안될 것이다.

부호의 설명

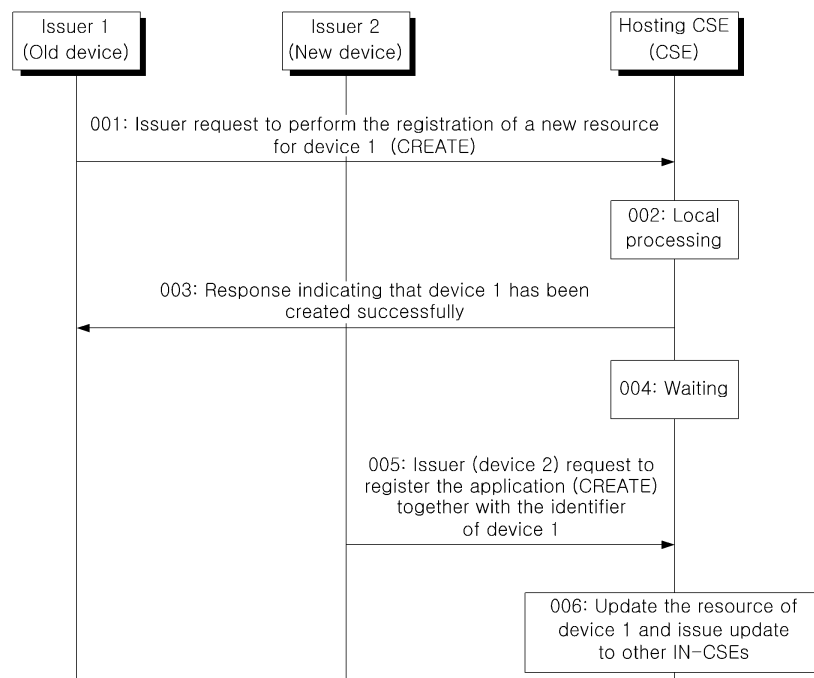
- [0033] AE : Application Entity
- CSE : Common Service Entity
- ADN : Application Dedicated Node
- ASN : Application Service Node
- MN : Middle Node
- IN : Infrastructure Node

도면

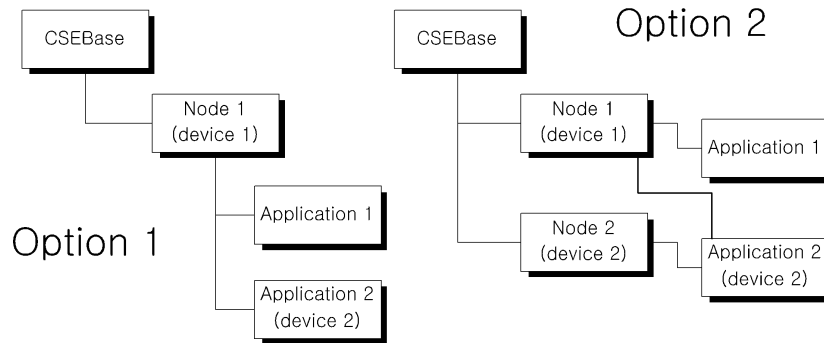
도면1



도면2



도면3



도면4

Linked Resource Type (link destination)	Linking Resource Type (link origin)	Linking method
accessControlPolicy	Several (node, application, remoteCSE, container...)	Attribute named "accessControlPolicyID"
node	CSEBase	Attribute named "nodeID" OR Child resource of type "node"
CSEBase or remoteCSE	node	Attribute named "hostedCSEID" OR parent resource of type CSEBase
a parent resource of any resourceType	a child resource of any resourceType	Attribute named "parentID"
A paired resource of any resourceType	A pairing resource of any resourceType	Attribute named "pairedID"