UPnP
Internet of Things

Dec 2014

Keith Miller – Intel
Wouter van der Beek – Cisco
UPnP Internet of Things Task Force

Overview

- Scope
- Architecture
  - Local components description
  - Sensor Management Bridge
  - Bridging types
  - Bridge component descriptions
  - Cloud components description
  - UDA 2.X for IoT
- SensorManagement Overview
  - Structure, Use Case Example (Aggregation)
- Sensor Management Data Modelling
  - Modelling Approach, Example (refrigerator)
- Security

Scope

UPnP IoT solves:

- Aggregating devices sensor and actuator data in a local network
- Observing and controlling those devices from anywhere agnostic to any platform
- Sharing information on a predefined granularity basis across networks with anyone
- Deciding what, when and with whom to share lies with the owner of the device
- Securing all communication

Using UPnP infrastructure

What is UPnP infrastructure

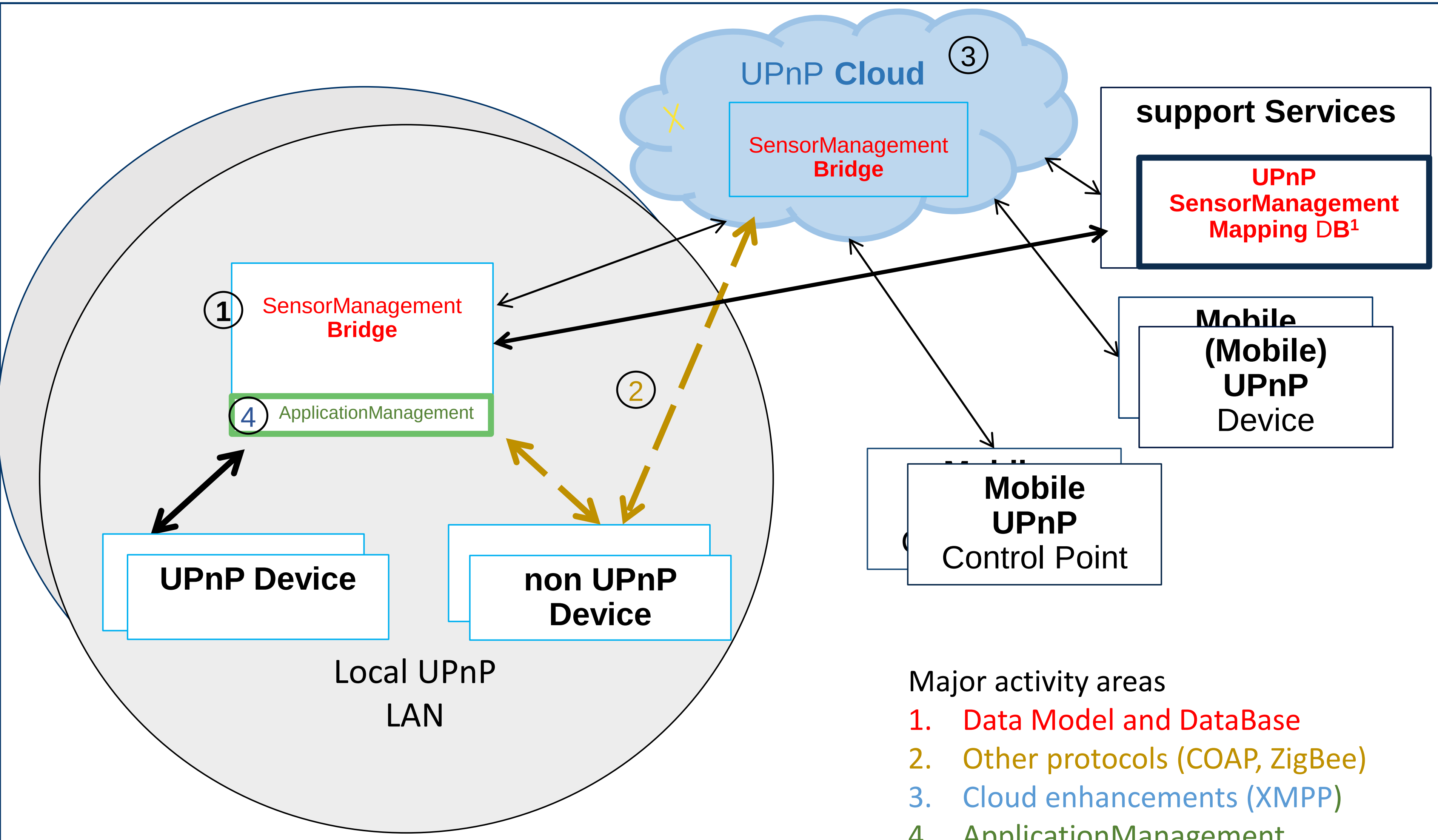
Existing Open Standards

- Billions of deployed devices.
  - Smart TVs, Gateways, Mobile Devices, Game Consoles, PCs
- Existing Device Control Protocols for home automation
  - HVAC, light, security camera, …
  - Sensor, Device, and Energy Management

New Open Standards

- UPnP+
  - UPnP Cloud based on RFC 6120, 6121 [XMPP]
  - IPv6 support
  - ApplicationManagement

UPnP IoT Architecture Overview



Major activity areas

1. Data Model and DataBase
2. Other protocols (COAP, ZigBee)
3. Cloud enhancements (XMPP)
4. ApplicationManagement

Overview Description

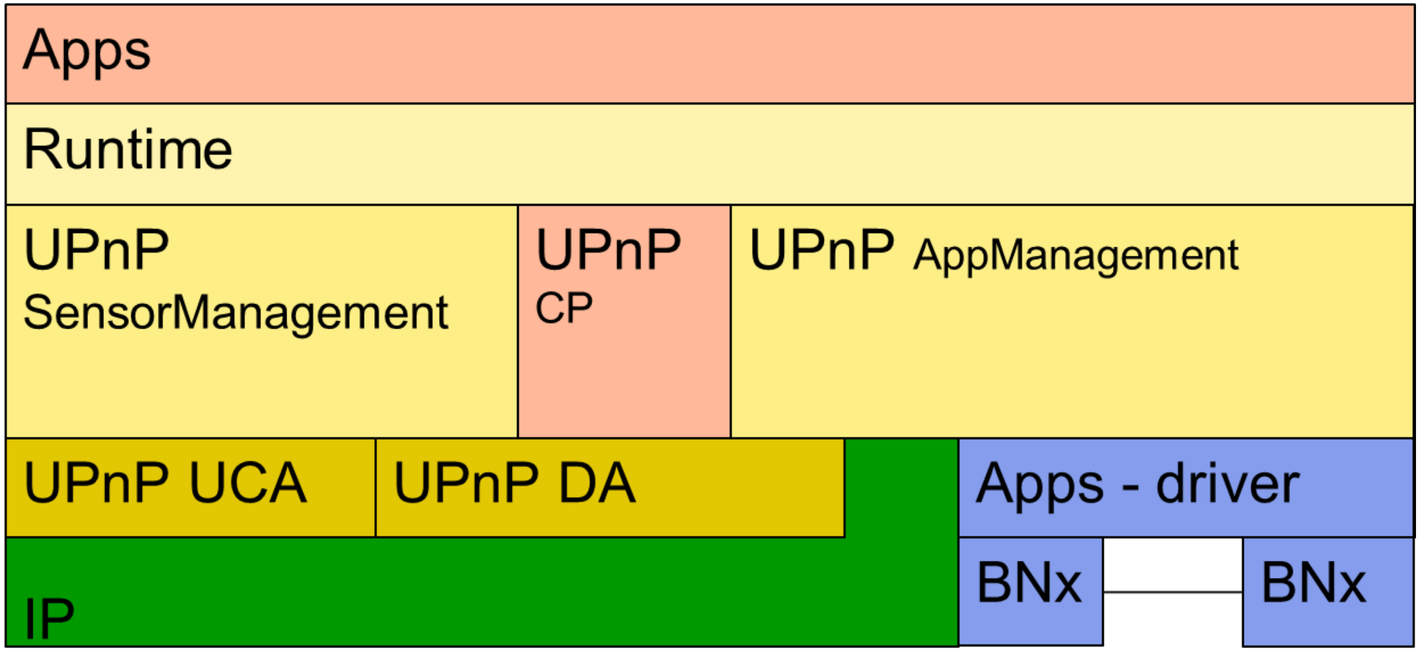
- Multiple local networks are connected to the cloud by means of UPnP Cloud architecture
  - This can include cloud services
- Individual UPnP devices and control points can be connected to the cloud with presence, state, and events shared securely with other local networks
- Ecosystem is easily extended using simple and flexible Data Models
- Data Models can be stored and interacted with via the SensorManagement Database (Service)

Component Overview - local

Bridging between UPnP and non UPnP devices includes

- Devices sensors/actuators supporting IP
  - For example, HTTP, COAP, REST, XMPP, MQTT
- Devices sensors/actuators on non-IP networks
  - For example, sensor hardware bridging between IP and non-IP networks (ZigBee, Z-Wave, ANT+, Bluetooth, etc)
- Runtime conversion Apps
  - ApplicationManagement (DIAL-like) for conversion

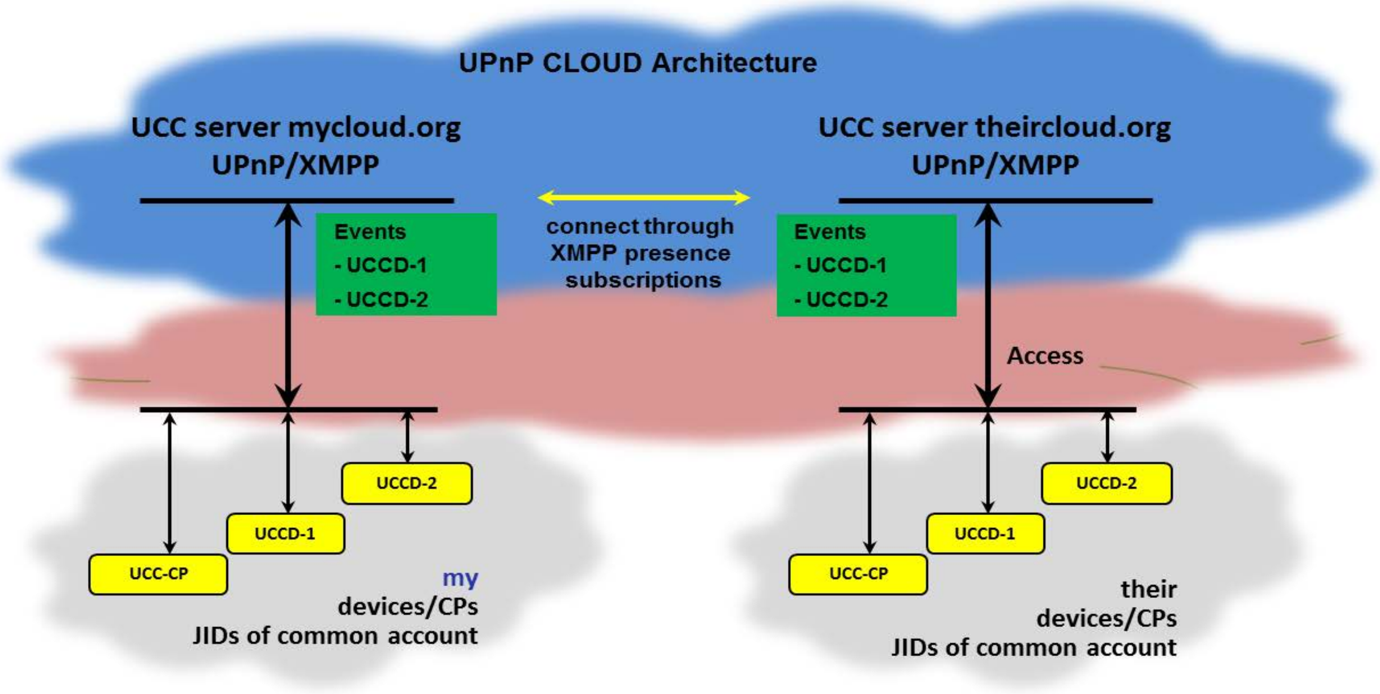
SensorManagement Bridge via Apps



Applications
UPnP DCPs
UPnP infrastructure
Bridged network infrastructure

Component overview - cloud

- UPnP Cloud connects UPnP Devices (UCCD) and Control Points (UCC-CP) as XMPP clients via an XMPP server.

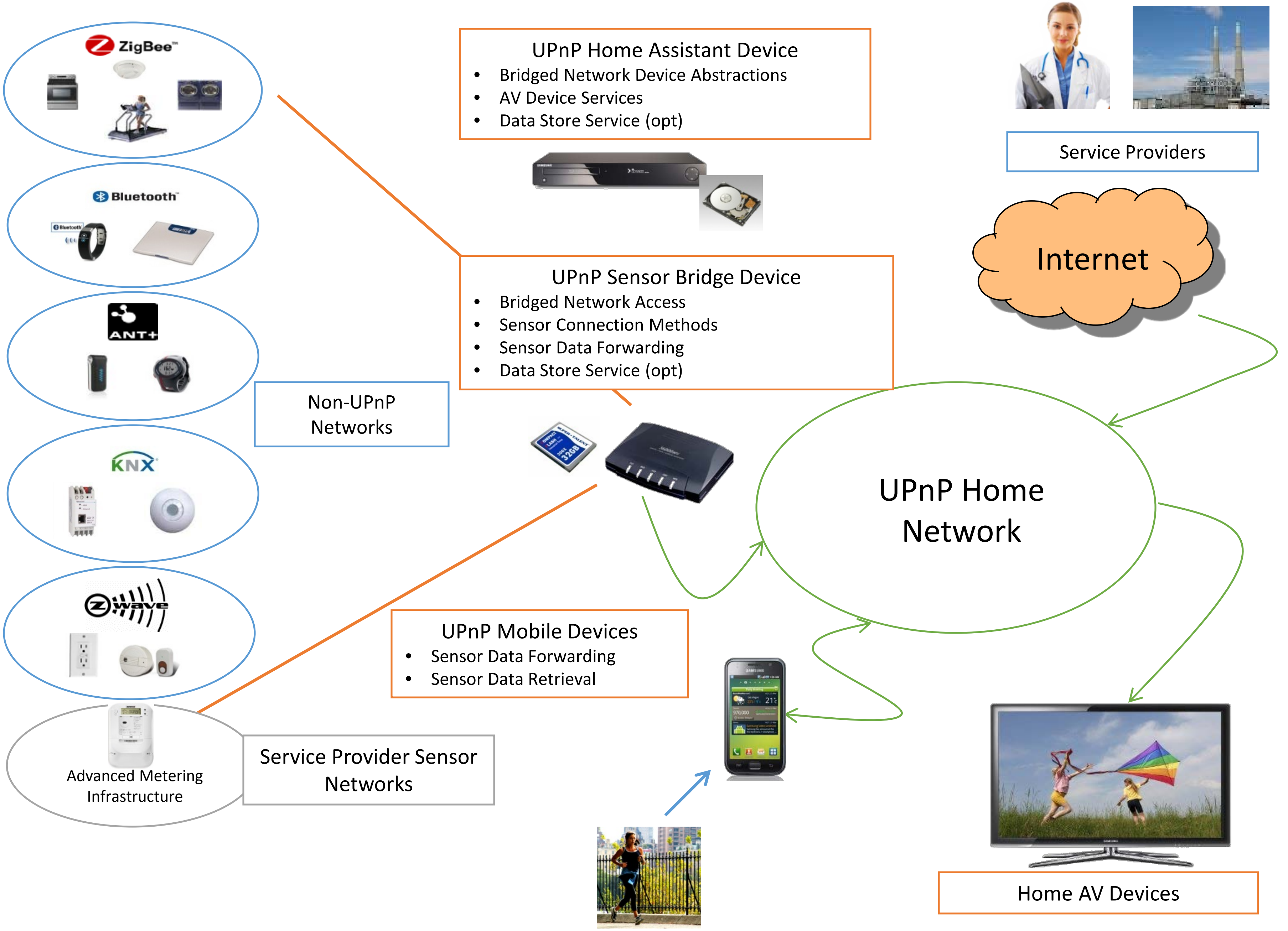


UPnP+ for IoT

- UPnP+ (incl. UDA 2.0) released in September 2014
- UDA 2.X version in development

- UPnP IoT is adding new protocols and architectural elements
  - In particular, existing APIs are being mapped to REST+JSON
    - SensorManagement is already RESTful, by means of SOAP actions
    - Will have a *pure* REST interface
  - CoAP is under consideration as one of the protocols for resource constrained devices.

UPnP Sensor Network Infrastructure



UPnP Home Assistant Device

- Bridged Network Device Abstractions
- AV Device Services
- Data Store Service (opt)

Service Providers

Internet

UPnP Sensor Bridge Device

- Bridged Network Access
- Sensor Connection Methods
- Sensor Data Forwarding
- Data Store Service (opt)

Non-UPnP Networks

UPnP Home Network

UPnP Mobile Devices

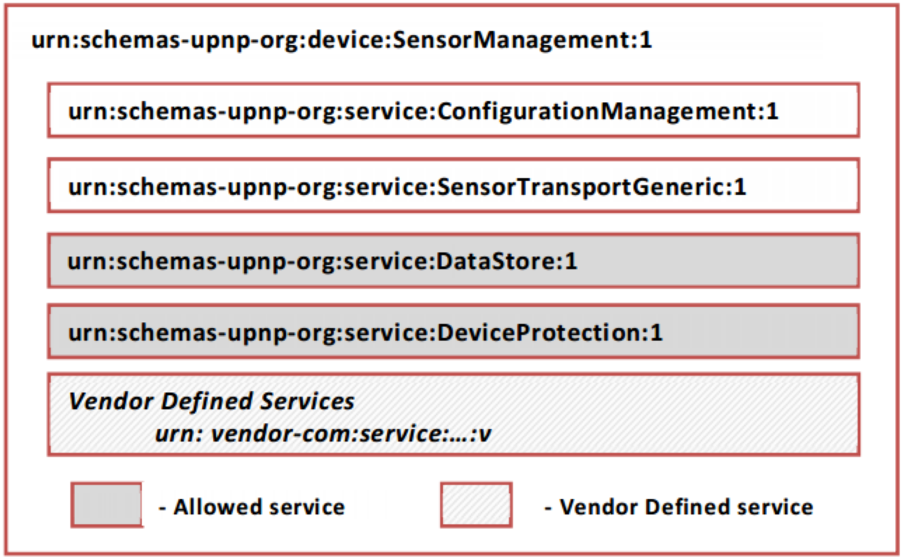
- Sensor Data Forwarding
- Sensor Data Retrieval

Advanced Metering Infrastructure

Service Provider Sensor Networks

Home AV Devices

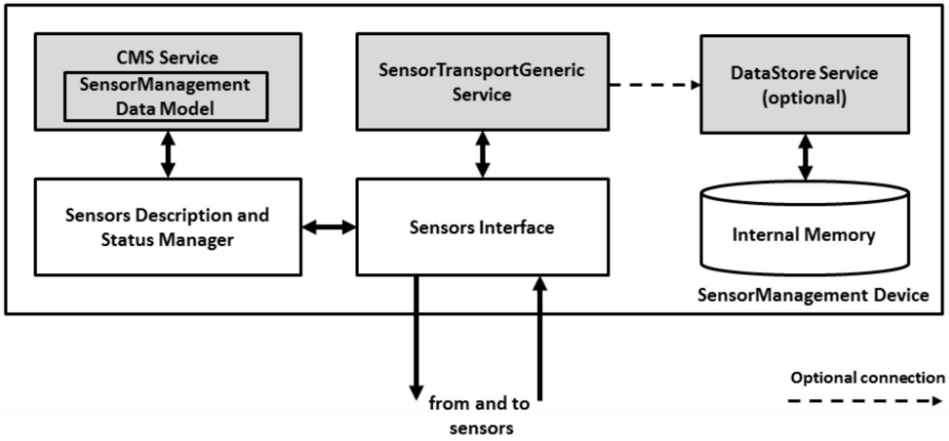
SensorManagement Overview


urn:schemas-upnp-org:device:SensorManagement:1
urn:schemas-upnp-org:service:ConfigurationManagement:1
urn:schemas-upnp-org:service:SensorTransportGeneric:1
urn:schemas-upnp-org:service:DataStore:1
urn:schemas-upnp-org:service:DeviceProtection:1
Vendor Defined Services
urn: vendor-com:service:…:v
- Allowed service
- Vendor Defined service

SensorManagement is a UPnP Device

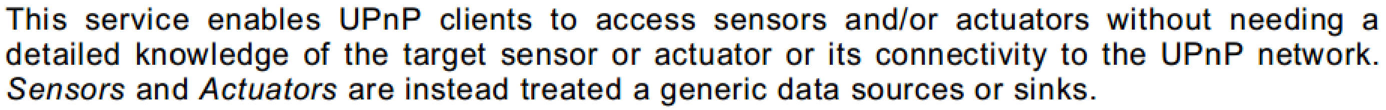
- 2 Mandatory Services
  - ConfigurationManagement
    SensorTransportGeneric
- 2 Optional Services
  - DataStore
  - DeviceProtection

Interfaces look like this ->

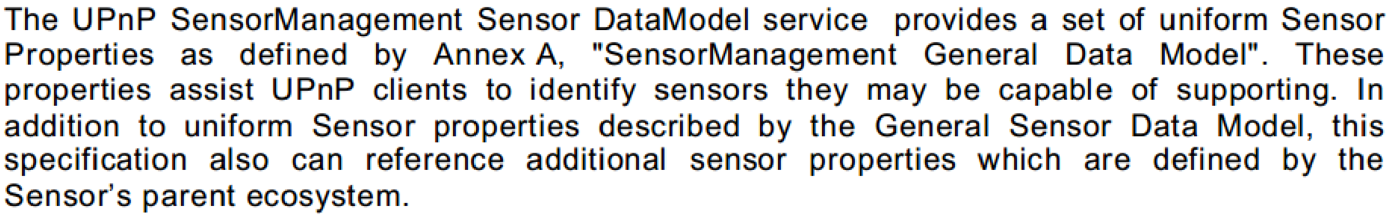

CMS Service
SensorManagement Data Model
SensorTransportGeneric Service
DataStore Service (optional)
Sensors Description and Status Manager
Sensors Interface
Internal Memory
SensorManagement Device
from and to sensors
Optional connection

SensorManagement Services

ConfigurationManagement (with specific Sensor DataModel)

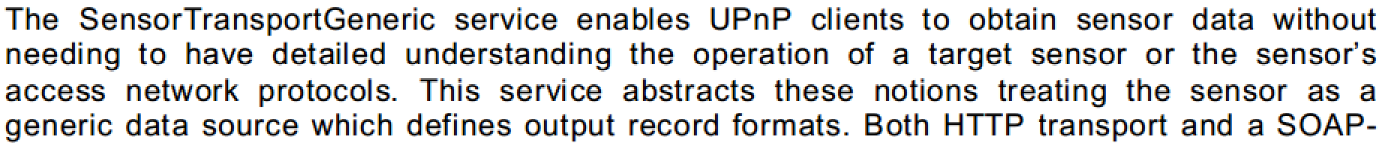


This service enables UPnP clients to access sensors and/or actuators without needing a detailed knowledge of the target sensor or actuator or its connectivity to the UPnP network. *Sensors* and *Actuators* are instead treated a generic data sources or sinks.



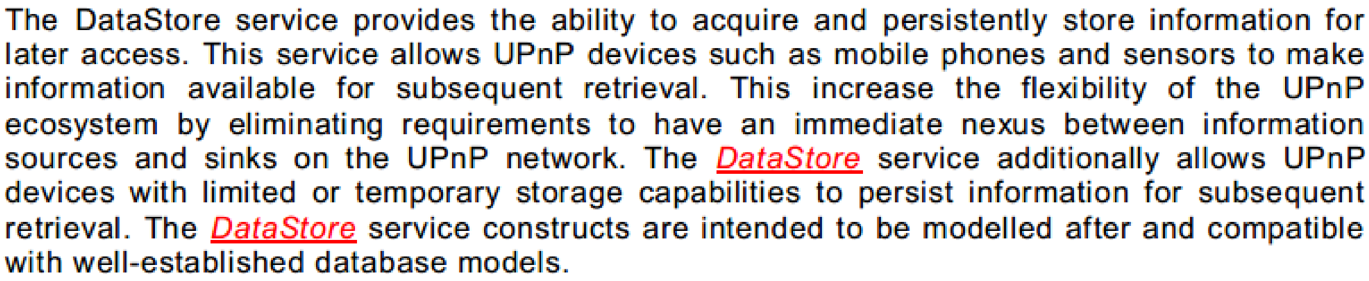
The UPnP SensorManagement Sensor DataModel service provides a set of uniform Sensor Properties as defined by Annex A, "SensorManagement General Data Model". These properties assist UPnP clients to identify sensors they may be capable of supporting. In addition to uniform Sensor properties described by the General Sensor Data Model, this specification also can reference additional sensor properties which are defined by the Sensor's parent ecosystem.

TransportGeneric Service



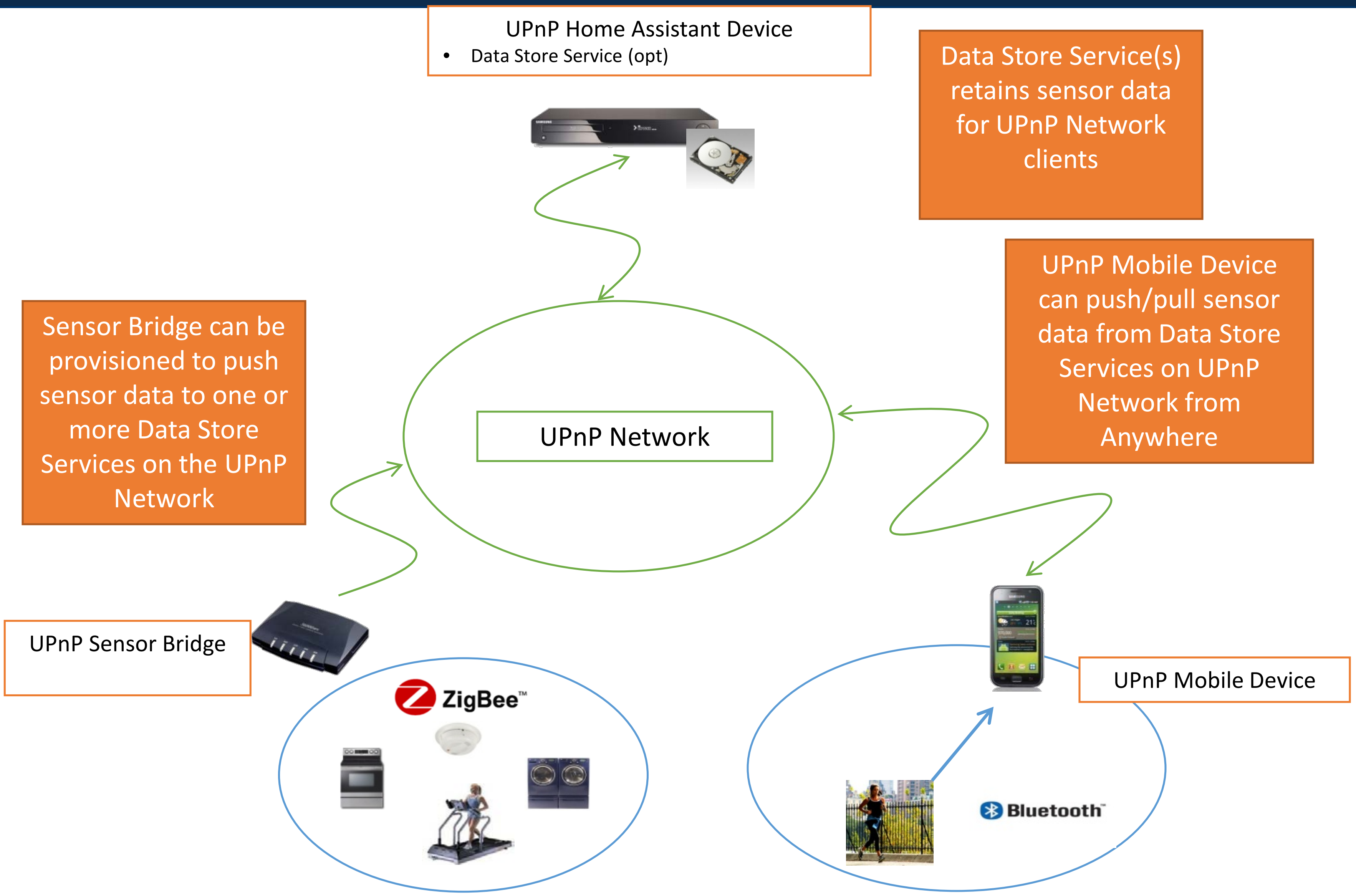
The SensorTransportGeneric service enables UPnP clients to obtain sensor data without needing to have detailed understanding the operation of a target sensor or the sensor's access network protocols. This service abstracts these notions treating the sensor as a generic data source which defines output record formats. Both HTTP transport and a SOAP-

DataStore Service



The DataStore service provides the ability to acquire and persistently store information for later access. This service allows UPnP devices such as mobile phones and sensors to make information available for subsequent retrieval. This increase the flexibility of the UPnP ecosystem by eliminating requirements to have an immediate nexus between information sources and sinks on the UPnP network. The *DataStore* service additionally allows UPnP devices with limited or temporary storage capabilities to persist information for subsequent retrieval. The *DataStore* service constructs are intended to be modelled after and compatible with well-established database models.

Typical UPnP Sensor Use Case



UPnP Home Assistant Device

- Data Store Service (opt)

Data Store Service(s) retains sensor data for UPnP Network clients

UPnP Mobile Device can push/pull sensor data from Data Store Services on UPnP Network from Anywhere

Sensor Bridge can be provisioned to push sensor data to one or more Data Store Services on the UPnP Network

UPnP Network

UPnP Sensor Bridge

UPnP Mobile Device

SensorManagement Data Model

- An IoT Sensor is defined as a set of SensorURNs
- Generic SensorURNs can be used by multiple devices
    - Standard SensorURNs
- Defining a set of sample devices that use those SensorURNs
    - Standard SensorTypes
- Manufacturers can create their own SensorTypes and keep interoperability
    - Just have to use standard SensorURNs
- SensorTypes and SensorURNs are like “interfaces”

Naming conventions

- Sensor URNs (DataItems)
  - List of UPnP defined sensors/actuators (features).
  - Generic list that every device can use
  - Units are defined
- List of standard modelled devices
  - Containing:
    - Mandatory SensorURN (features)
    - Optional SensorURN (features)
    - Vendor defined extensions
- Where a sensor is located

Sources of Models

- Member companies – vendor specific models
- Some popular home devices and bridges –
  - HUE, StriimLight, WeMo, ..
- Other SDOs
  - ongoing evaluation based on IPR and accessibility
- Short list of Generic Models and Features
  - UPnP IoT Data Model Task Force

Sensor Management

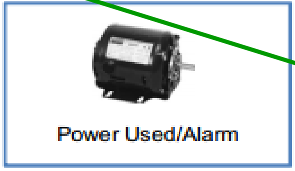
- Reuses ConfigurationManagement Service
  - Difference is: modelling of the nodes itself
  - Model described in Annex A.
- Tree list of nodes
- Node describes functionality/behaviour
  - Reference to other node
  - Collection of sensors
  - DataItem
    - Can be an real world sensor/actuator

Sensor Management

Nodes can be:

- Created
  - CreateInstance()
- Read
  - GetValues()
- Updated
  - SetValues()
- Deleted
  - DeleteInstance()
- Notified
  - Alarming Feature: UPnP state variable event including the node & value of the node

DataModel Refrigerator



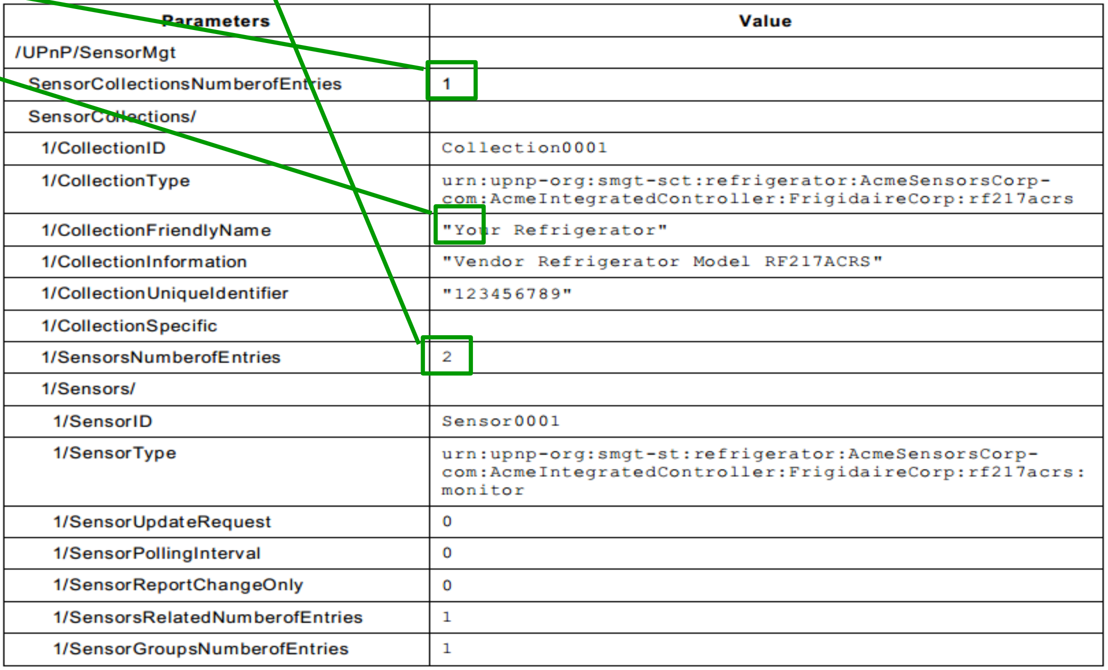
Sensor 1 - Status

AccumulatedPowerUsed	(kW-h, Cumulative)
FreezerTemp	(degC, Average)
GroceryTemp	(degC, Average)
VegtableTemp	(degC, Average)
DoorOpenAlarm	("Door Id", Timeout)
PowerFaultAlarm	(0|1)
StatusInterval	(s)

Sensor 2 - Control

FreezerTempSetting	(degC - Current, LowLimit, HighLimit)
GroceryTempSetting	(degC - Current, LowLimit, HighLimit)
VegtableTempSetting	(degC - Current, LowLimit, HighLimit)

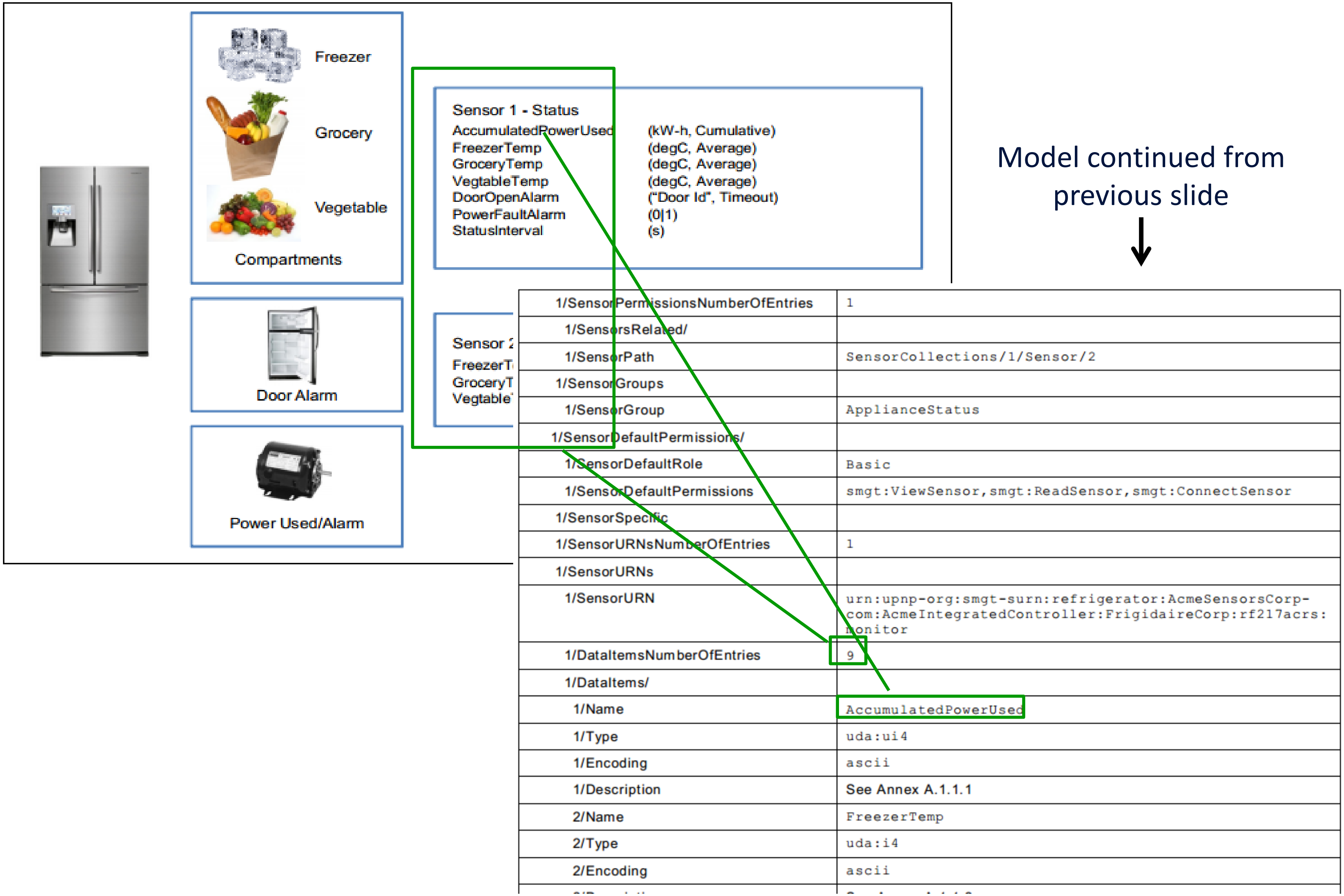
Features are named collection of sensors/actuators



Parameters	Value
/UPnP/SensorMgt	
SensorCollectionsNumberofEntries	1
SensorCollections/	
1/CollectionID	Collection0001
1/CollectionType	urn:upnp-org:smgt-sct:refrigerator:AcmeSensorsCorp-com:AcmeIntegratedController:FrigidaireCorp:rf217acrs
1/CollectionFriendlyName	"Your Refrigerator"
1/CollectionInformation	"Vendor Refrigerator Model RF217ACRS"
1/CollectionUniqueIdentifier	"123456789"
1/CollectionSpecific	
1/SensorsNumberofEntries	2
1/Sensors/	
1/SensorID	Sensor0001
1/SensorType	urn:upnp-org:smgt-st:refrigerator:AcmeSensorsCorp-com:AcmeIntegratedController:FrigidaireCorp:rf217acrs:monitor
1/SensorUpdateRequest	0
1/SensorPollingInterval	0
1/SensorReportChangeOnly	0
1/SensorsRelatedNumberofEntries	1
1/SensorGroupsNumberofEntries	1

Refrigerator is a modelled device – can be generic or specific

DataModel Refrigerator (Cont)



Sensor 1 - Status

AccumulatedPowerUsed	(kW-h, Cumulative)
FreezerTemp	(degC, Average)
GroceryTemp	(degC, Average)
VegtableTemp	(degC, Average)
DoorOpenAlarm	("Door Id", Timeout)
PowerFaultAlarm	(0|1)
StatusInterval	(s)

Sensor 2
FreezerT
GroceryT
Vegtable

Model continued from previous slide

1/SensorPermissionsNumberOfEntries	1
1/SensorsRelated/	
1/SensorPath	SensorCollections/1/Sensor/2
1/SensorGroups	
1/SensorGroup	ApplianceStatus
1/SensorDefaultPermissions/	
1/SensorDefaultRole	Basic
1/SensorDefaultPermissions	smgt:ViewSensor,smgt:ReadSensor,smgt:ConnectSensor
1/SensorSpecific	
1/SensorURNsNumberOfEntries	1
1/SensorURNs	
1/SensorURN	urn:upnp-org:smgt-surn:refrigerator:AcmeSensorsCorp-com:AcmeIntegratedController:FrigidaireCorp:rf217acrs:monitor
1/DataItemsNumberOfEntries	9
1/DataItems/	
1/Name	AccumulatedPowerUsed
1/Type	uda:ui4
1/Encoding	ascii
1/Description	See Annex A.1.1.1
2/Name	FreezerTemp
2/Type	uda:i4
2/Encoding	ascii
2/Description	See Annex A.1.1.2

UPnP+ security

UPnP+ adds security for:

•In home by means of UPnP Device Protection

Access to the home is designed from ground up to include security which is incorporated in XMPP.

Device Protection

- Inside the home UPnP specified device protection as a backwards compatible mechanism
- When using device protection unsecured control points still can use the device
- However the functionality is then restricted to “open” actions
- Most actions are profiled so data can be read, but not modified
  - Example: an unsecure control point can browse AV-CDS content, but cannot delete or add content

Device Protection (2)

- Uses TLS with self generated certificates
  - no trust authority involved
- Certificate identification is translated to a user role
  - e.g. admin, super user, regular user, guest,…
- DCPs define user roles they distinguish and the actions each role has access to
- Secured control points therefore use HTTPS for
  - device and service description downloads
  - invocation of actions allowed by their user role
- Hence this communication is secure
  - network traffic can still be observed when unsecured mode is used
- Any control point, including unsecured ones, can still register for events
  - e.g. see what state the device is in

Device Protection (3)

- WPS based authentication
- Other scenarios described

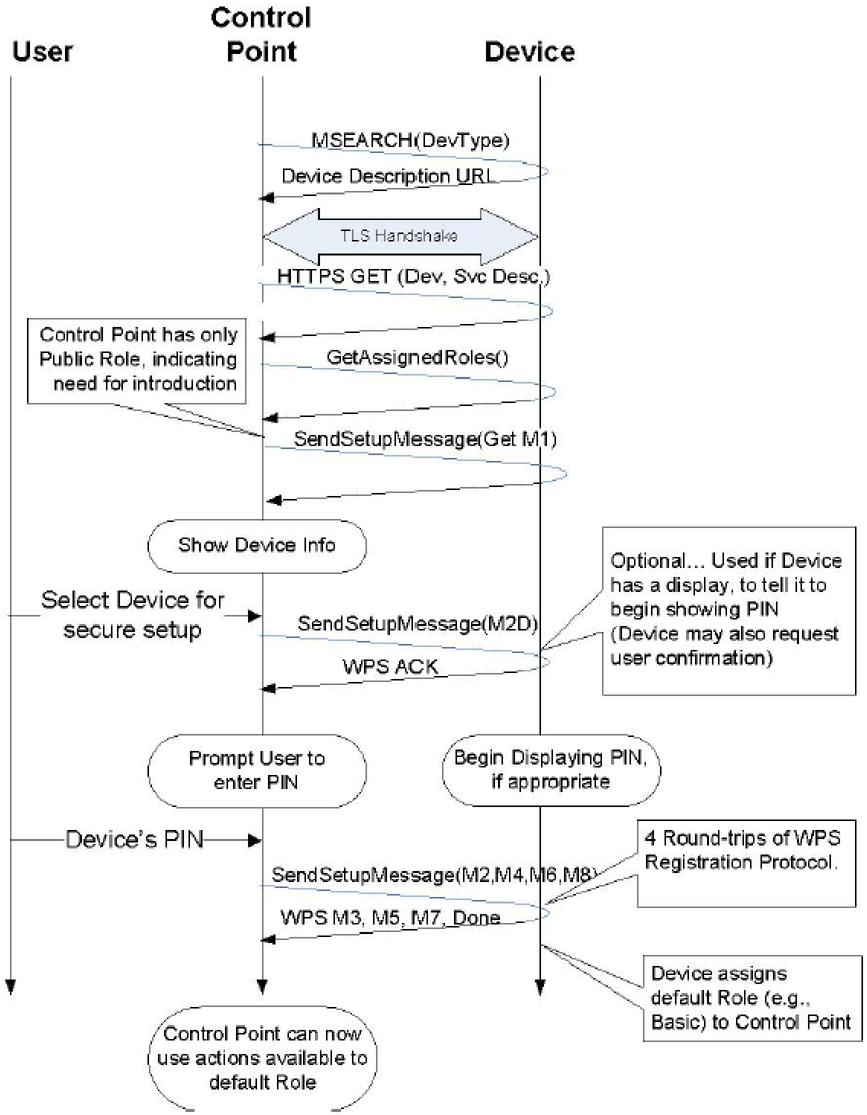

User
Control Point
Device
MSEARCH(DevType)
Device Description URL
TLS Handshake
HTTPS GET (Dev, Svc Desc.)
Control Point has only Public Role, indicating need for introduction
GetAssignedRoles()
SendSetupMessage(Get M1)
Show Device Info
Optional… Used if Device has a display, to tell it to begin showing PIN (Device may also request user confirmation)
Select Device for secure setup
SendSetupMessage(M2D)
WPS ACK
Prompt User to enter PIN
Begin Displaying PIN, if appropriate
Device's PIN
4 Round-trips of WPS Registration Protocol.
SendSetupMessage(M2,M4,M6,M8)
WPS M3, M5, M7, Done
Device assigns default Role (e.g., Basic) to Control Point
Control Point can now use actions available to default Role

Figure 3-1: Default WPS-based Introduction.

XMPP security

Remote Access is using XMPP as transport mechanism

Using UPnP Cloud means that one needs to log in to XMPP by having an account (JID)

•XMPP is using

- SASL for authentication
- TLS for encrypting the link

•UPnP specifications are silent about how you register the device/control point to your account

Cloud security

- Using UPnP cloud also enables the local network to be more secure

- Share information by means of the cloud:

  No need to exchange WiFi passwords with visitors

- Create a virtual room, where you can share the TV
- Invite a visitor to that room to use the TV to display pictures
- The visitor can use a guest WiFi network or the 3/4G network on his mobile phone

Cloud source code

- https://github.com/upnpforum
- UPnP Cloud Device Applications
  - Sample desktop applications implementing UPnP Cloud Architecture (UCA). The repository contains the implementation of the following UPnP devices: DimmableLight, MediaServer, MediaRenderer and a light bulb modelled as a SensorManagement device.
- UPnP Cloud Controller Application for Android
  - Sample Android application capable of controlling several types of network devices connected using UPnP protocol for both local (UDA) and cloud devices (UCA).

UPnP SM

FORUM

www.upnp.org

For the interconnected lifestyle