Technology Keys for Connected Lighting

ENERGY STAR Webinar
May 26, 2016

Michael Poplawski
Pacific Northwest National Laboratory
What is a Connected Lamp or Luminaire?

- No industry standard
- DOE SSL Program: controllable and intelligent SSL source (capable of using, or consuming data), one or more network interfaces, one or more sensors (i.e. data producers)
- ENERGY STAR Connected Lamp (V2.0, Rev. Feb. 2016): An ENERGY STAR eligible connected lamp includes elements (hardware and software or firmware) or instructions required to enable communication in response to consumer-authorized energy or performance related commands and complies with all requirements for connected lamps in the specification. These elements may reside inside or outside of the base lamp.
What is driving the emergence of Connected Lighting?

- Significant technology trends driving performance improvements and cost reductions
  - Computing
  - Mobile
  - Intelligence (i.e. microcontrollers), network interfaces, and sensors
- Solid-State Lighting
- Emergence of cloud storage, computing
- Focus on systems and data
What can Connected Lighting products do today?

• Dimming
• Dim-to-warm
• White or color tuning
• Notification
• Sensing
• Indoor positioning
• Energy reporting
Where might Connected Lighting go in the (near?) future?
How might connected Lighting Systems change lighting?

**Opportunity**

Enabling intelligent lighting devices with (the right type and amount of) data can result in reduced energy consumption and improved lighting performance.

The collected data may enable other revenue streams that compete with lighting and energy performance.
What can we do to accelerate Connected Lighting?

<table>
<thead>
<tr>
<th>Technology Development</th>
<th>Technology Deployment</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Energy reporting</td>
<td>• Real-world performance</td>
</tr>
<tr>
<td>• Interoperability</td>
<td>• User engagement and education</td>
</tr>
<tr>
<td>• System configuration</td>
<td>• High performance product identification</td>
</tr>
<tr>
<td>• Key new features e.g. “non-energy benefits”</td>
<td></td>
</tr>
<tr>
<td>• Standards and specifications</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Collaborations</th>
<th>Outcomes</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Industry Consortia</td>
<td>• Increased adoption, viable business models</td>
</tr>
<tr>
<td>• Energy Efficiency Programs</td>
<td>• Data-driven energy management</td>
</tr>
<tr>
<td>• Lighting system designers, integrators</td>
<td>• Transactive energy markets</td>
</tr>
</tbody>
</table>
Energy reporting: why?

• Enable new market opportunities
  – Energy billing for devices currently on flat-rate tariffs
  – Pay-for-performance energy efficiency incentives
  – Lower cost, more accurate energy savings validation for service-based business models
  – Self-characterization of available (i.e. marketable) “building energy services”
  – Verified delivery of utility incented energy transactions e.g. peak and other demand response

• Reduce energy consumption
  – Data-driven energy management
  – Transactive energy markets
Data driven performance management

Discovery & Measurement
- Asset data, Remote monitoring
- Baseline performance

Assessment & Simulation
- Analytic engine, What-if scenarios
- Analyze monitored data, Simulate policy scenarios

Policy & Control
- Rules engine, Execution proxies
- Automated deployment and execution of policies

Reporting & Decision Support
- Results, Benefits, Savings
- New baseline performance
Energy reporting: how?

- Identification of major energy data use cases
- Consideration of implementation cost vs. performance trade-offs
- One or more sets of accuracy, precision requirements that meet use case needs
- Standard accuracy classes, test & measurement methods, pass/fail criteria
Interoperability: what?

- **Compatibility**: The ability of two or more devices, applications, networks, or systems to **coexist** in the same physical environment – that is, operate without corrupting, interfering with, or hindering the operation of the other entity.

- **Interoperability**: The ability of two or more devices, applications, networks, or systems to **work together**, and (more specifically) to reliably and securely **exchange and readily use data** with a common shared meaning.

- **Interchangeability**: The ability of two or more devices, applications, networks, or systems to be **physically exchanged** for each other and provide a defined level of identical operation without additional configuration.
Interoperability: what?

OSI Model

7  Data  Application  Network Process to Application
6  Data  Presentation  Data Representation and Encryption
5  Data  Session  Interhost Communication
4  Segments  Transport  End-to-End Connections and Reliability
3  Packets  Network  Path Determination and IP (Logical Addressing)
2  Frames  Data Link  MAC and LLC (Physical addressing)
1  Bits  Physical  Media, Signal, and Binary Transmission

There are many possible levels of interoperability
Interoperability: why?

- Facilitates competition
- Facilitates collaboration
- Reduces risk
- Enables choice
- Enables integration
- Reduces cost
- Facilitates greater data exchange
- Facilitates more sophisticated automation, and thereby improved performance and user satisfaction
There is an app for that …

... but how many apps, networks, systems do you want to manage?
It’s déjà vu all over again?
Interoperability: why now?

2010

Compatibility
1) Interference with broadcast or communication networks (e.g. FCC, CISPR)
2) Phase control and SSL source issues (e.g. NEMA SSL-7a)

2015

Interoperability
1) 0-10V, DALI
2) ZigBee, EnOcean
3) Connected Lighting Alliance, TALQ, ANSI C137, AllJoyn, many others...

2020?

Interchangeability
1) ANSI bases
2) Electrical, mechanical, thermal interfaces (e.g. Zhaga)
3) ANSI C137?
Leverage, lean on industry consortia and standard development organizations (SDO’s)

Let technology providers and the market pick winners

Characterize and promote maturity (e.g. compliance testing programs, databases)

Identify priorities

Focus: Useable data

Application layer i.e. information models

Start with API’s

Power and energy data

Key non-energy or non-lighting data opportunities?
  - Lighting quality
  - Human factors
  - Non-energy benefits
  - Non-lighting systems
A black box device model

- **Sensors**: Sensor Input 1, Sensor Input 2
- **Actuators**: Actuator Out 1, Actuator Out 2, Actuator Out 3, Comm Out
- **Control**: Input Value 1, Input Value 2, Input Value 3, Input Value 4
- **Output**: Output Value 1, Output Value 2, Output Value 3, Output Value 4
- **Algorithms**: Internal Logic, Local Data Logging, Local Alarms, Local User Interface
- **Local User Interface**: Monitor, Control
Example: LonMark data model

<table>
<thead>
<tr>
<th>NV # (M/O)*</th>
<th>Variable Name</th>
<th>SNVT Name</th>
<th>SNVT Index</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (M)</td>
<td>nviLampValue</td>
<td>SNVT_switch_2</td>
<td>189</td>
<td>Used for scheduled and Occupancy events.</td>
</tr>
<tr>
<td>2 (M)</td>
<td>nviStatReset</td>
<td>SNVT_stat_control</td>
<td>216</td>
<td>Sets/initializes energy, runtime, and error counts.</td>
</tr>
<tr>
<td>3 (M)</td>
<td>nvoLampFb</td>
<td>SNVT_switch_2</td>
<td>189</td>
<td>Feedback of current nviLampValue.</td>
</tr>
<tr>
<td>4(M)</td>
<td>nvoControlData</td>
<td>SNVT_control_data</td>
<td>218</td>
<td>A structured variable describing all current operating values and state of the SLC.</td>
</tr>
<tr>
<td>5(M)</td>
<td>nvoLnStatus</td>
<td>SNVT_faults</td>
<td>217</td>
<td>Latched alarm values are updated only when the condition is asserted active to limit alarm log size and minimize the data sent to the system management software.</td>
</tr>
<tr>
<td>6(M)</td>
<td>nvoVersion</td>
<td>SNVT_version</td>
<td>220</td>
<td>A structure with three fields major.minor.build.</td>
</tr>
<tr>
<td>7(O)</td>
<td>nviTimeNow</td>
<td>SNVT_time_stamp</td>
<td>84</td>
<td>For LCs supporting a HW real-time clock, updates to this variable set the clock time. The profile implementation should display the local time value by updating the value allowing a UI to poll the value to determine the local time base.</td>
</tr>
</tbody>
</table>

* M = mandatory, O = optional
## Example: LonMark data model

<table>
<thead>
<tr>
<th>Man. Opt.</th>
<th>SCPT Name</th>
<th>SCPT Index</th>
<th>Associated NVs **</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Man</td>
<td>SCPTcontrolCfg cpControlCfg SNVT_controlCfg</td>
<td>382</td>
<td>Entire Object</td>
<td>Defines many of the operating parameters for the SLC.</td>
</tr>
<tr>
<td>Man</td>
<td>SCPTLimits cpLimits SNVT_fault_limits</td>
<td>383</td>
<td>Entire Object</td>
<td>Used for alarm thresholds as defined in Alarming, below.</td>
</tr>
<tr>
<td>Man</td>
<td>SCPTsceneDef cpSceneTbl structure</td>
<td>384</td>
<td>Entire Object</td>
<td>Defines a table containing between 4 and 12 lighting scenes. Each scene is defined by a scene_number, a level, an unoccupied_scene_number which is used when the controller determines the luminaire light levels should be lowered due to lack of traffic.</td>
</tr>
<tr>
<td>Opt</td>
<td>SCPTgeoLocation cpLocation SNVT_geo_loc</td>
<td>350</td>
<td>Entire Object</td>
<td>Provides tagging for GPS location, and physical asset tagging. Version 13.04 standard type.</td>
</tr>
<tr>
<td>Opt</td>
<td>SCPTlightingGroupMembership cpGroupMember structure</td>
<td>361</td>
<td>nviLampValue</td>
<td>Defines membership in one or more lighting groups</td>
</tr>
<tr>
<td>Opt</td>
<td>SCPTbackupSchedule cpBkUpSchedule enumeration structure</td>
<td>344</td>
<td>Entire Object</td>
<td>Defines a backup schedule to be used if the device determines the segment controller is not updating the one of the control inputs. Using this Cp requires the node to apply only if it has a valid local time value.</td>
</tr>
<tr>
<td>Opt</td>
<td>SCPTpowerProfile cpPowerProfile SNVT_power</td>
<td>381</td>
<td>Entire Object</td>
<td>Defines the nominal power measured at 5 commanded nviLampValues (5% 25% 50% 75% 100%)</td>
</tr>
</tbody>
</table>

* Man = mandatory, Opt = optional

** List of NVs to which this configuration property applies.
Example: AllSeen Alliance data model

5 LampParameters Interface

This chapter defines the LampParameters interface used by the Lighting service framework.

5.1 Interface name

<table>
<thead>
<tr>
<th>Interface name</th>
<th>Version</th>
<th>Secured</th>
<th>Object path</th>
</tr>
</thead>
<tbody>
<tr>
<td>org.allseen.LSF.LampParameters</td>
<td>1</td>
<td>np</td>
<td>/org/allseen/LSF/Lamp</td>
</tr>
</tbody>
</table>

5.2 Properties

<table>
<thead>
<tr>
<th>Property name</th>
<th>Signature</th>
<th>List of values</th>
<th>Writable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Interface version number</td>
</tr>
<tr>
<td>Energy_Use_Milliwatts</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Lamp current energy usage in milliwatts</td>
</tr>
<tr>
<td>Brightness_Lumens</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Lamp current brightness in lumens</td>
</tr>
</tbody>
</table>

5.3 Methods

None.

6 LampDetails Interface

This chapter defines the LampDetails interface used by the Lighting service framework.

6.1 Interface name

<table>
<thead>
<tr>
<th>Interface name</th>
<th>Version</th>
<th>Secured</th>
<th>Object path</th>
</tr>
</thead>
<tbody>
<tr>
<td>org.allseen.LSF.LampDetails</td>
<td>1</td>
<td>np</td>
<td>/org/allseen/LSF/Lamp</td>
</tr>
</tbody>
</table>

6.2 Properties

<table>
<thead>
<tr>
<th>Property name</th>
<th>Signature</th>
<th>List of values</th>
<th>Writable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Interface version number</td>
</tr>
<tr>
<td>Make</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Lamp make</td>
</tr>
<tr>
<td>Model</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Lamp model</td>
</tr>
<tr>
<td>Type</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Type</td>
</tr>
<tr>
<td>LampType</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Lamp type</td>
</tr>
<tr>
<td>LampBeamAngle</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Lamp beam angle</td>
</tr>
<tr>
<td>Dimmable</td>
<td>b</td>
<td>true</td>
<td>false</td>
<td>Can lamp be dimmed</td>
</tr>
<tr>
<td>Color</td>
<td>b</td>
<td>true</td>
<td>false</td>
<td>Color</td>
</tr>
<tr>
<td>VariableColorTemp</td>
<td>b</td>
<td>true</td>
<td>false</td>
<td>Color temp</td>
</tr>
<tr>
<td>HasEffects</td>
<td>b</td>
<td>true</td>
<td>false</td>
<td>Has effects</td>
</tr>
<tr>
<td>MinVoltage</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Minimum voltage</td>
</tr>
<tr>
<td>MaxVoltage</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Maximum voltage</td>
</tr>
<tr>
<td>Wattage</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Wattage</td>
</tr>
<tr>
<td>IncandescentEquivalent</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Incandescent equivalent</td>
</tr>
<tr>
<td>MaxLumens</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Maximum lumens</td>
</tr>
<tr>
<td>MinTemperature</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Minimum temperature</td>
</tr>
<tr>
<td>MaxTemperature</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Maximum temperature</td>
</tr>
<tr>
<td>ColorRenderingIndex</td>
<td>u</td>
<td>Positive integers</td>
<td>No</td>
<td>Color rendering index</td>
</tr>
<tr>
<td>LampID</td>
<td>s</td>
<td>String</td>
<td>No</td>
<td>Lamp ID</td>
</tr>
</tbody>
</table>

6.3 Methods

None.

https://wiki.allseenalliance.org/tsc/connected_lighting
Example: AllSeen Alliance data model

7 LampState Interface

7.1 Interface name

<table>
<thead>
<tr>
<th>Interface name</th>
<th>Version</th>
<th>Secured</th>
<th>Object path</th>
</tr>
</thead>
<tbody>
<tr>
<td>org.allseen.LSF.LampState</td>
<td>1</td>
<td>yes</td>
<td>/org/allseen/LSF/Lamp</td>
</tr>
</tbody>
</table>

7.2 Properties

<table>
<thead>
<tr>
<th>Property name</th>
<th>Signature</th>
<th>List of values</th>
<th>Writable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td></td>
<td>Positive integers</td>
<td>No</td>
<td>Interface version number</td>
</tr>
<tr>
<td>OnOff</td>
<td></td>
<td>True or False</td>
<td>Yes</td>
<td>On or off state of lamp</td>
</tr>
<tr>
<td>Hue</td>
<td></td>
<td>Positive integers</td>
<td>Yes</td>
<td>Hue of lamp</td>
</tr>
<tr>
<td>Saturation</td>
<td></td>
<td>Positive integers</td>
<td>Yes</td>
<td>Saturation of lamp</td>
</tr>
<tr>
<td>ColorTemp</td>
<td></td>
<td>Positive integers</td>
<td>Yes</td>
<td>Color temp of lamp</td>
</tr>
<tr>
<td>Brightness</td>
<td></td>
<td>Positive integers</td>
<td>Yes</td>
<td>Current brightness of lamp</td>
</tr>
</tbody>
</table>

7.3 Methods

The following methods are exposed by a BusObject that implements the org.allseen.LampState interface.

7.3.1 TransitionLampState

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Mandatory</th>
<th>Signature</th>
<th>List of values</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timestamp</td>
<td>Yes</td>
<td></td>
<td>Positive integers</td>
<td>Timespamp (in ms) of when to start the transition</td>
</tr>
<tr>
<td>NewState</td>
<td>Yes</td>
<td>a(ax)</td>
<td>Array of variants</td>
<td>New state of the lamp to transition to</td>
</tr>
<tr>
<td>TransitionPeriod</td>
<td>Yes</td>
<td></td>
<td>Positive integers</td>
<td>Time period (in ms) to transition over to new state</td>
</tr>
</tbody>
</table>

Output

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Mandatory</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LampResponseCode</td>
<td>Yes</td>
<td>The result code of the operation.</td>
</tr>
</tbody>
</table>

Description

Change the state of the lamp at the specified time, between the specified OnOff, Brightness, Hue, Saturation, and ColorTemp values. Pulse for the specified number of times, at the specified duration.

7.4 Signals

<table>
<thead>
<tr>
<th>Signal name</th>
<th>Parameter name</th>
<th>Mandatory</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LampStateChanged</td>
<td>LampID</td>
<td>Yes</td>
<td>A way to notify a listener (e.g., lamp controller) that the lamp state has changed.</td>
</tr>
</tbody>
</table>
Example OCF/OIC data models:

```yaml
#%RAML 0.8
title: OICIlluminanceSensor
version: v1.1.0-20160229
documentation:
  - title: © 2016 Open Interconnect Consortium, Inc. All rights reserved.
    content: |
      Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
      1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
      2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE OPEN INTERCONNECT CONSORTIUM, INC. "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED. IN NO EVENT SHALL THE OPEN INTERCONNECT CONSORTIUM, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

schemas:
- Illuminance: !include oic.r.sensor.illuminance.json
traits:
- interface: queryParameters:
  if: enum: ["oic.if.s","oic.if.baseline"]

/IlluminanceSensorResURI:
description: |
This resource describes an illuminance sensor
illuminance is a float and represents the sensed luminous flux per unit area in lux.

displayName: Illuminance Sensor
is: [ interface ] # valid for all methods

get:
responses:
  200:
    body:
      application/json:
        schema: Illuminance
        example: |
          { "rt": "oic.r.sensor.illuminance", "id": "unique_example_id", "illuminance": 450 }
```

```yaml
#%RAML 0.8
title: OICMotionSensor
version: v1.1.0-20160229
documentation:
  - title: © 2016 Open Interconnect Consortium, Inc. All rights reserved.
    content: |
      Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
      1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
      2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE OPEN INTERCONNECT CONSORTIUM, INC. "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE OR WARRANTIES OF NON-INFRINGEMENT, ARE DISCLAIMED. IN NO EVENT SHALL THE OPEN INTERCONNECT CONSORTIUM, INC. OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

schemas:
- Motion: !include oic.r.sensor.motion.json
traits:
- interface: queryParameters:
  if: enum: ["oic.if.s","oic.if.baseline"]

/MotionResURI:
description: |
This resource describes whether motion has been sensed or not.
The value is a boolean.
A value of True means that motion has been sensed.
A value of False means that motion not been sensed.

displayName: Motion Sensor
is: [ interface ] # valid for all methods

get:
responses:
  200:
    body:
      application/json:
        schema: Motion
        example: |
          { "rt": "oic.r.sensor.motion", "id": "unique_example_id", "value": true }
The path to mature interoperability standards

1) Goal definition
2) Specification
3) Specification Verification
4) Broad Manufacturer Adoption
5) Compliance Testing Program
6) Open Standardization
7) Broad Market Adoption

Mature Standard

multiple iterations
## 11.4. Start Time: All Lamps

<table>
<thead>
<tr>
<th>Lamp Type</th>
<th>ENERGY STAR Requirements</th>
<th>Methods of Measurement and/or Reference Documents</th>
<th>Supplemental Testing Guidance</th>
</tr>
</thead>
</table>
| Non-Connected Lamps        | Reported value of time for lamp to remain continuously illuminated shall be within 750 milliseconds of application of electrical power. | Measurement: [ENERGY STAR Start Time Test](#)  
The reported value shall be the average of measured unit values tested, rounded to the nearest millisecond. |
| Connected Lamps            | Reported value of time for lamp to remain continuously illuminated shall be within 1 second of application of electrical power. |                                                                                                                                                      |                                                                                                                                                      |

## 11.7. Standby Power Consumption: All Lamps

<table>
<thead>
<tr>
<th>Source Type</th>
<th>ENERGY STAR Requirements</th>
<th>Methods of Measurement and/or Reference Documents</th>
<th>Supplemental Testing Guidance</th>
</tr>
</thead>
</table>
| All Source Types          | Lamps without integral controls shall not draw power in the off mode.  
Exception: Lamps with integral controls (e.g., motion sensors, photosensors, wireless control, standby mode, or connected functionality) shall consume no more than 0.5 watt in standby mode or network mode.  
Standby power (if applicable) shall be reported for equipment (outside of the lamp) required for connectivity (e.g., gateways, hubs, and network controllers, excluding equipment typically found in the home such as a Wi-Fi router). | IEC 62301 Edition 2.0 2011-01 Household electrical appliances - Measurement of standby power  
U.S. Department of Energy Conservation Test Procedures for Compact Fluorescent Lamps (once final)  
Laboratory test results shall detail off-state power consumption to the tenth of a watt.  
This applies to lamps that may have wireless controllability but may not meet all connected criteria as identified in the specification definition for connected lamp and [Section 12.7 Connected Product Criteria](#).  
If required for connectivity, the lamp manufacturer shall specify one set of representative equipment (outside of the lamp) for which standby power shall be reported. |
ENERGY STAR Lamp V2.0 Specifications

12.6. Products with Connected Functionality – Optional

<table>
<thead>
<tr>
<th>Source Type</th>
<th>ENERGY STAR Requirements</th>
<th>Methods of Measurement and/or Reference Documents</th>
<th>Supplemental Testing Guidance</th>
</tr>
</thead>
</table>
| All source types | Product must continue to comply with the applicable product safety standards – the addition of the functionality shall not override existing safety protections and functions.  
Product must comply with Section 11.7 Standby Power Consumption.  
Power consumption (if applicable) shall be reported for equipment (outside of the lamp) required for connectivity (e.g., gateways, hubs, and network controllers, excluding equipment typically found in the home such as a Wi-Fi router). | Measurement: None | Test Requirements: Connected products without color tuning capabilities shall be tested at full power for all applicable requirements. Connected products with color tuning capabilities shall be tested under the conditions specified under Section 5.1.  
Compliance with connected functionality requirements, in Sections 12.7-12.12, shall be demonstrated through examination of product and/or product documentation.  
If required for connectivity, the lamp manufacturer shall specify one set of representative equipment (outside of the lamp) for which power consumption shall be reported. |

12.7. Connected Product Criteria:

To be recognized as connected, a “connected lamp” shall include elements (hardware and software or firmware) or instructions required to enable communication in response to consumer-authorized energy or performance related commands (e.g., instructions for downloading a mobile application, Bluetooth syncing guidance) and shall meet the requirements in Sections 12.8-12.12. These elements may reside inside or outside of the base lamp. For example, a “base lamp” may connect wirelessly via a home gateway or network controller to a cloud service that implements energy estimation functions.

The specific design and implementation of the connected lamp is at the manufacturer’s discretion provided it is interoperable with other devices via open communications protocol and enables economical, consumer-authorized third party access to the functionalities provided for in sections 12.9, 12.10 and 12.11. Capabilities of system controller and connected protocol shall be reported as applicable.
12.8. **Open Access**

The product shall enable connectivity by one of following means:

1. Open-standards communications from the lamp, or
2. Open-standards communications from an external controller, included with the product or available separately.
3. Where no suitable open standards communications method exists (e.g., an IP interface), an available and documented communication method must be used. In these cases, a manufacturer-specific method to implement the functions in sections 12.9, 12.10, and 12.11 shall be published for use with the product.

To enable interconnection with the product; an interface specification, Application Programming Interface (API) or similar documentation shall be made available to interested parties that enables sections 12.9, 12.10 and 12.11 connected functionality, and includes accuracy, units and measurement or estimation interval for Energy Consumption Reporting.

12.9. **Energy Consumption Reporting**

The lamp, or the gateway device or cloud service connected to it, shall be capable of interconnecting with consumer authorized entities to communicate data representative of its interval energy consumption. It is recommended that data be reported in watt-hours for intervals of 15 minutes; however, representative data may also be reported in alternate units and intervals as specified in the product manufacturer’s interface specification or API. If the lamp does not provide power consumption directly in watts, the manufacturer shall make available a method for estimating power consumption, in watts, from the representative data that is provided by the lamp.

12.10. **Operational Status Reporting**

At a minimum, the lamp, or the gateway device or cloud service connected to it, shall be capable of providing the following information to energy management systems and other consumer authorized devices, services or applications via a communication link: operational status (e.g., on/off).
12.11. Remote Management

The product shall be capable of receiving and responding to energy management system or other consumer authorized remote requests, via devices, services or applications, similar to hard-wired consumer controllable functions.

12.12. Information to Consumers

If additional devices, services, and/or infrastructure are required to activate the product’s connected capabilities, prominent labels, or other forms of consumer notifications shall be displayed at the point of purchase and in the product literature. (e.g., “This product has Z-wave control capability and requires interconnection with a Z-wave controller to enable local lighting control.”)
Future Connected Lighting Specification Considerations

• Energy reporting accuracy determined according to industry standard (perhaps ANSI C137.XX)
• User interface requirements (e.g. industry standard white or color picker)
• Compliance with a qualified or industry standard Data/Information Model for energy data/information
• Reporting of other data/information (e.g. occupancy, ambient lighting, environmental conditions)
• Compliance with a qualified or industry standard Data/Information Model for other data/information
• Certified compliance with industry standard or industry consortia interoperability specification
Questions?
michael.poplawski@pnnl.gov