## 范围

OCF规范分为两组文档：

- 核心规范文档：核心规范文档指定框架，以实现物联网使用的OCF规约和生态系统。其中包含OCF核心体系结构，接口，协议和服务。

- 垂直域规范文档：垂直域规范文档指定OCF设备规约，以支持不同垂直市场领域的物联网使用，如智能家居，工业，医疗保健和汽车。 它们还指定了资源定义以启用垂直服务和用例。 这些规范包括ISO / IEC 30118-5：2018，它建立在核心规范中定义的OCF核心架构的接口和网络安全性之上。

本文档是OCF核心规范，它规定了框架和核心体系结构。

## 参考文献

以下文件的全部或部分内容在本文件中作了规范性引用其应用必不可少。 凡是注日期的引用文件，仅引用的版本适用。 未注明日期参考文献，最新版本的参考文件（包括任何修订）适用。

ISO 8601, Data elements and interchange formats – Information interchange –Representation of dates and times, International Standards Organization, December 3, 2004

ISO/IEC DIS 20924, Information Technology – Internet of Things – Vocabulary, June 2018 https://www.iso.org/standard/69470.html

ISO/IEC 30118-2:2018, Information technology – Open Connectivity Foundation (OCF) Specification – Part 2: Security specification https://www.iso.org/standard/74239.html Latest version available at: https://openconnectivity.org/specs/OCF\_Security\_Specification.pdf

ISO/IEC 30118-5:2018, Information technology – Open Connectivity Foundation (OCF) Specification – Part 5: Smart home device specification https://www.iso.org/standard/74242.html Latest version available at: https://openconnectivity.org/specs/OCF\_Device\_Specification.pdf

OCF Easy Wi-Fi Setup, Information technology – Open Connectivity Foundation (OCF) Specification – Part 7: Wi-Fi Easy Setup specification Latest version available at: https://openconnectivity.org/specs/OCF\_Wi- Fi\_Easy\_Setup\_Specification\_v2.0.1.pdf

IETF RFC 768, User Datagram Protocol, August 1980 https://www.rfc-editor.org/info/rfc768

IETF RFC 3339, Date and Time on the Internet: Timestamps, July 2002 https://www.rfc-editor.org/info/rfc3339

IETF RFC 3986, Uniform Resource Identifier (URI): General Syntax, January 2005. https://www.rfc-editor.org/info/rfc3986

IETF RFC 4122, A Universally Unique IDentifier (UUID) URN Namespace, July 2005 https://www.rfc-editor.org/info/rfcfse4122

IETF RFC 4287, The Atom Syndication Format, December 2005, 529 https://www.rfc-editor.org/info/rfc4287

IETF RFC 4941, Privacy Extensions for Stateless Address Autoconfiguration in IPv6, September 2007 <https://www.rfc-editor.org/info/rfc4941>

IETF RFC 5424, The Syslog Protocol, March 2009 https://tools.ietf.org/html/rfc5424IETF RFC 5646, Tags for Identifying Languages, September 535 2009 <https://www.rfc-editor.org/info/rfc5646>

IETF RFC 5988, Web Linking: General Syntax, October 2010 https://www.rfc-editor.org/info/rfc5988 IETF RFC 6347, Datagram Transport Layer Security Version 1.2, January 2012 <https://www.rfc-editor.org/info/rfc6347>

IETF RFC 6434, IPv6 Node Requirements, December 2011 https://www.rfc-editor.org/info/rfc6434

IETF RFC 6573, The Item and Collection Link Relations, April 2012 <https://www.rfc-editor.org/info/rfc6573>

IETF RFC 6690, Constrained RESTful Environments (CoRE) Link Format, August 2012 <https://www.rfc-editor.org/info/rfc6690>

IETF RFC 7049, Concise Binary Object Representation (CBOR), October 2013 <https://www.rfc-editor.org/info/rfc7049>

IETF RFC 7084, Basic Requirements for IPv6 Customer Edge Routers, November 2013 <https://www.rfc-editor.org/info/rfc7084>

IETF RFC 7159, The JavaScript Object Notation (JSON) Data Interchange Format, March 2014 <https://www.rfc-editor.org/info/rfc7159>

IETF RFC 7252, The Constrained Application Protocol (CoAP), June 2014 <https://www.rfc-editor.org/info/rfc7252>

IETF RFC 7301, Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension, July 2014 <https://www.rfc-editor.org/info/rfc7301>

IETF RFC 7595, Guidelines and Registration Procedures for URI Schemes, June 2015 <https://www.rfc-editor.org/info/rfc7595>

IETF RFC 7641, Observing Resources in the Constrained Application Protocol (CoAP), September 2015 <https://www.rfc-editor.org/info/rfc7641>

IETF RFC 7721, Security and Privacy Considerations for IPv6 Address Generation Mechanisms, March 20016 https://www.rfc-editor.org/info/rfc7721

IETF RFC 7959, Block-Wise Transfers in the Constrained Application Protocol (CoAP), August 2016 <https://www.rfc-editor.org/info/rfc7959>

IETF RFC 8075, Guidelines for Mapping Implementations: HTTP to the Constrained Application Protocol (CoAP), February 2017 <https://www.rfc-editor.org/info/rfc8075>

IETF RFC 8323, CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets, February 2018 https://www.rfc-editor.org/info/rfc8323

IANA IPv6 Multicast Address Space Registry <http://www.iana.org/assignments/ipv6-multicast-addresses/ipv6-multicast-addresses.xhtml>

IANA Media Types Assignment, March 2017 <http://www.iana.org/assignments/media-types/media-types.xhtml>

IANA Link Relations, October 2017 http://www.iana.org/assignments/link-relations/link-relations.xhtml JSON Schema Validation, JSON Schema: interactive and non-interactive validation, January 2013 <http://json-schema.org/draft-04/json-schema-validation.html>

OpenAPI specification, fka Swagger RESTful API Documentation Specification, Version 2.0 <https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md>

## 术语、定义和缩写

### 3.1术语和定义

出于本文档的目的，以下给出的术语和定义适用。ISO和IEC在以下地址维护用于标准化的术语数据库：

- ISO在线浏览平台：https://www.iso.org/obp。

- IEC Electropedia：可从http://www.electropedia.org/获取。

#### 3.1.1 Alert

设备（3.1.10）通过专门的资源类型（3.1.32）提供的信息，提供有关可能采取措施的潜在问题或设备（3.1.10）状态的详细信息。

#### 3.1.2 Atomic Measurement

一种设计模式，确保客户端（3.1.3）只能原子地访问链接资源（3.1.28）的属性（3.1.31），即单个组。

#### 3.1.3 Client

访问服务器上的资源（3.1.28）的逻辑实体（3.1.38）。

#### 3.1.4 Collection

包含零个或多个链接（3.1.18）的资源（3.1.28）。

#### 3.1.5 Common Properties

为所有资源（3.1.28）指定的属性（3.1.31）。

#### 3.1.6 Composite Device

建模为多种设备类型（3.1.11）的设备（3.1.10）; 将每个组件设备类型（3.1.11）作为集合公开（3.1.4）。

#### 3.1.7 Configuration Source

云或服务网络或包含向设备（3.1.10）提供相关配置信息的本地只读文件。

#### 3.1.8 Core Resources

这些资源（3.1.28）在本文档中定义。

#### 3.1.9 Default OCF Interface

一种OCF接口（3.1.15），用于在请求中省略OCF接口（3.1.15）时生成响应。

#### 3.1.10 Device

一个或多个角色的逻辑实体，例如Client（3.1.3），Server（3.1.38）。

注1：在Platform（3.1.27）上可存在多个设备（3.1.10）。

#### 3.1.11 Device Type

拥有唯一命名的定义，指示设备（3.1.10）支持的最小资源类型集（3.1.32）。

注1：设备类型（3.1.11）提供设备（3.1.10）的提示，例如灯或风扇，以便在资源（3.1.28）发现期间使用。

#### 3.1.12 Discoverable Resource

“/ oic / res”中列出的资源（3.1.28）。

#### 3.1.13 OCF Endpoint

参与OCF协议的实体，进一步被识别为给定传输协议簇的请求和响应消息的源或目的地。

注1：传输协议簇的示例是基于IPv6的CoAP over UDP。

#### 3.1.14 Framework

 本文档中定义的一组相关功能和交互的集合，可实现包括物联网在内的各种网络设备的互操作性。

#### 3.1.15 OCF Interface

根据IETF RFC 6690和OCF定义的接口描述，提供资源（3.1.28）的视图和允许的响应。

#### 3.1.16 Introspection

确定设备（3.1.10）的托管资源（3.1.28）功能的机制。

#### 3.1.17 Introspection Device Data (IDD)

描述构成设备（3.1.10）的资源（3.1.28）的每个实现方法的有效负载的数据。

#### 3.1.18 Links

根据IETF RFC 5988扩展类型的Web链接。

#### 3.1.19 Non-Discoverable Resource

未在“/ oic / res”中列出的资源（3.1.28）。资源可以通过链接（3.1.18）访问，该链接由另一个资源传送。例如，在Collection（3.1.4）中链接的资源不必在“/ oic / res”中列出，因为遍历Collection会发现设备上实现的资源。

#### 3.1.20 Non-OCF Device

不符合OCF要求的设备（3.1.10）。

#### 3.1.21 Notification

使客户端（3.1.3）了解资源（3.1.28）中状态变化的机制。

#### 3.1.22 Observe

通过发送RETRIEVE操作来监视资源（3.1.28）的行为，该操作由托管资源的服务器（3.1.38）缓存并在对该资源的每次更改时重新处理。

#### 3.1.23 OpenAPI 2.0

 本文档中使用的资源（3.1.28）和Introspection Device Data（3.1.17）定义如同OpenAPI规范中所定义。

#### 3.1.24 Parameter

提供有关由链接的目标URI引用的资源（3.1.28）的元数据的元素。

#### 3.1.25 Partial UPDATE

对资源（3.1.28）的UPDATE操作，包括通过应用于资源类型（3.1.32）的OCF接口（3.1.15）可见的属性（3.1.31）的子集。

#### 3.1.26 Physical Device

设备（3.1.10）展现的物理内容。

#### 3.1.27 Platform

包含一个或多个设备（3.1.10）的物理设备（3.1.26）。

#### 3.1.28 Resource

表示框架（3.1.14）建模和展现的实体。

#### 3.1.29 Resource Directory

资源（3.1.28）的一组描述，其中实际资源（3.1.28）保存在托管资源目录（3.1.29）的设备（3.1.10）外部的服务器（3.1.38）上，允许执行查找这些资源（3.1.28）

注 1：睡眠服务器（3.1.38）或选择不直接对多播请求进行监听/响应的服务器（3.1.38）可以使用此功能。

#### 3.1.30 Resource Interface

 资源（3.1.28）允许请求的资格。

#### 3.1.31 Property

资源（3.1.28）的重要参数（3.1.24），包括通过资源公开的元数据。

#### 3.1.32 Resource Type

属性类（3.1.31）的唯一命名定义以及该类支持的交互操作。

 注1：每个资源（3.1.28）都有一个Property（3.1.31）“rt”，其值是资源的唯一名称。

#### 3.1.33 Scene

一个静态实体，用于存储资源（3.1.28）集合（3.1.4）的一组已定义的Property（3.1.31）值。

 注1：一个场景（3.1.33）是一组资源（3.1.28）的预定设置，每个资源都具有必须改变的属性（3.1.31）的预定值。

#### 3.1.34 Scene Collection

包含可能的场景值枚举（3.1.36）和当前场景值的集合（3.1.4）

#### 3.1.35 Scene Member

资源（3.1.28），包含场景值（3.1.36）到资源中属性（3.1.31）的值的映射。

#### 3.1.36 Scene Value

 一个场景（3.1.33）枚举器，表示资源（3.1.28）可以处于的状态。

#### 3.1.37 Secure OCF Endpoint

 具有安全连接的OCF端点（3.1.13）（例如，CoAPS）。

#### 3.1.38 Server

提供资源（3.1.28）状态信息并促进与其资源的远程交互（3.1.28）的设备（3.1.10）。

注1：可以实现服务器（3.1.38）以将非OCF设备（3.1.20）资源暴露给客户端（3.1.3）（见5.6）

#### 3.1.39 Unsecure OCF Endpoint

 具有不安全连接的OCF端点（）（例如，CoAP）

#### 3.1.40 Vertical Resource Type

垂直域规范中的资源类型（3.1.32）。

注1：垂直资源类型（3.1.40）的例子是“oic.r.switch.binary”。

### 3.2缩写术语

#### 3.2.1 ACL

Access Control List

#### 3.2.2 BLE

Bluetooth Low Energy

#### 3.2.3 CBOR

Concise Binary Object Representation

#### 3.2.4 CoAP

Constrained Application Protocol

#### 3.2.5 CoAPS

Secure Constrained Application Protocol

#### 3.2.6 DTLS

Datagram Transport Layer Security

#### 3.2.7 EXI

Efficient XML Interchange

#### 3.2.8 IP

Internet Protocol

#### 3.2.9 IRI

Internationalized Resource Identifiers

#### 3.2.10 ISP

Internet Service Provider

#### 3.2.11 JSON

JavaScript Object Notation

#### 3.2.12 mDNS

Multicast Domain Name Service

#### 3.2.13 MTU

Maximum Transmission Unit

#### 3.2.14 NAT

Network Address Translation

#### 3.2.15 OCF

Open Connectivity Foundation

#### 3.2.16 REST

Representational State Transfer

#### 3.2.17 RESTful

REST-compliant Web services

#### 3.2.18 UDP

User Datagram Protocol

#### 3.2.19 URI

Uniform Resource Identifier

#### 3.2.20 URN

Uniform Resource Name

#### 3.2.21 UTC

Coordinated Universal Time

#### 3.2.22 UUID

Universal Unique Identifier

#### 3.2.23 XML

Extensible Markup Language

## 4文件约定和组织

### 4.1约定

 在本文档中，使用大写的每个单词的第一个字母打印多个术语，条件，机制，序列，参数，事件，状态或类似术语，其余的小写（例如，网络架构）。这些单词的任何小写用法都具有正常的技术英语含义。

### 4.2表示法

 在本文档中，功能按需要，推荐，允许或过时描述如下：

 要求（或应当或必须）（M）。

  - 应实施这些基本功能以符合Core Architecture。短语“不得”和“禁止”表示禁止的行为，即如果执行则表示行为 实施不符合要求。

 推荐（或应该）（S）。

  - 这些功能增加了Core Architecture支持的功能，应该实现。

 推荐的功能利用了Core Architecture的功能，通常不会增加复杂性。请注意，对于合规性测试，如果实施了推荐功能，则应满足指定要求以符合这些准则。一些推荐的功能可能会成为未来的要求。

词组“不应该”表示允许但不推荐的行为。

 允许（可以或允许）（O）。

  - Core Architecture既不要求也不推荐这些功能，但如果实现该功能，则应满足指定的要求以符合这些指南。

 已过时。

  - 尽管本文档中仍介绍了这些功能，但除了向后兼容性之外，不应实现这些功能。操作期间出现不推荐使用的功能 符合当前文档的实现对实现的操作没有影响，并且不会产生任何错误条件。向后兼容性可能要求实现一个功能并按指定的功能运行，但它永远不会被使用符合本文档的实现。

 有条件地允许（CA）。

  - 定义或行为取决于条件。如果满足指定的条件，则允许定义或行为，否则不允许。

 有条件地要求（CR）。

  - 定义或行为取决于条件。如果满足指定的条件，则需要定义或行为。否则，除非明确定义为不允许，否则将允许定义或行为作为默认值。

 字面意思的字符串用“双引号”括起来。

 强调的单词以斜体印刷。

 在本文档中包含的所有属性和资源定义表中，“强制性”列表示详细的项目必须实施;将项目包含在与CRUDN操作关联的资源有效负载中的强制取决于该操作的适用模式。

### 4.3数据类型

 使用从IETF RFC 7159中定义的JSON值派生的数据类型定义资源。但是，资源可以使用JSON模式验证中定义的验证关键字来重载JSON定义的值以指定JSON值的特定子集。

在其他验证关键字中，JSON模式验证中的第7节定义了一个“格式”关键字，其中包含许多格式属性，如“uri”和“date-time”，以及一个“pattern”关键字。可用于验证字符串的正则表达式。 本节定义了可用于描述OCF资源的模式。 模式名称可用于文档文本可以出现JSON格式名称的位置。 实际的JSON模式应使用JSON类型和模式。

对于表1中定义的所有行，JSON类型为字符串。

表1 - 其他OCF类型

|  |  |  |
| --- | --- | --- |
| 模式名 | 模式 | 说明 |
| “csv” | <none> | 逗号分隔的列表，其值为字符串中的值。 csv中的值类型由使用csv的属性描述。 例如整数的csv。  注意csv被认为已弃用且是字符串数组应该用来代替新的资源 |
| “date” | ^([0 -9]{4})-(1[0-2]|0[1-9])-(3 [0-1]|2[0 - 9]|1[0-9]|0[1 -9])$ | 全日期格式模式参照IETF RFC 3339 |
| " duration " | ^(P(?!$)([0-9]+Y)?([0-9]+M)?([0- 9 ]+W)? ([0-9]+D)?((T(?=[0- 9]+[HMS])([0-9]+H)?([0-9]+M)?([0- 9 ]+S)?)?))$|^(P[0-9]+W)$|^(P[0- 9]{4})-(1[0-2]|0[1-9])-(3[0-1]|2[0- 9]|1[0-9]|0[1 -9])T(2[0-3]|1[0-9]|0[1- 9]):([0-5][0-9]):([0-5][0-9])$|^(P[0- 9]{4})(1[0-2]|0[1-9])(3[0-1]|2[0-9]|1[0- 9]|0[1-9])T(2[0-3]|1[0-9]|0[1-9])([0- 5][0 -9])([0-5][0-9])$ | 表示ISO 8601中定义格式的持续时间的字符串。  Al lowable格式是：P [n] Y [n] M [n] DT [n] H [n] M [n] S，P [n] W，P [n]的Y [n]的-M[n]的-DT[0-23] H [0-59]：M[O-  59]：S和P [n] W，P [n] Y [n] M [n] DT [0-23] H [0-59] M [0-59] S上。 P是强制性的，其他元素是可选的，时间  必须遵循T. |
| "int64” | ^0|(-?[1-9][0-9]{0,18})$ | 如果字符串实例包含范围为[ - （2 \*\* 63），（2 \*\* 63）-1]的整数，则该字符串实例有效获得此属性 |
| "language -tag " | ^[A-Za-z]{1,8}(-[A-Za-z0-9]{1,8})\*$ | 根据IETF RFC 5646子句格式化的IETF语言标签2.1 |
| “uint64 " | ^0|([1 -9][0-9]{0,19})$ | 如果字符串实例包含范围为[0，（2 \*\* 64）-1的整数，则该字符串实例对此属性有效 |
| " uuid " | ^[a-fA-F0-9]{8}-[a-fA-F0-9]{4}-[a-fAF0 -9]{4}-[a-fA-F0 -9]{4}-[a-fA-F0- 9]{12}$ | 根据IETF RFC 4122第3条的格式化的UUID字符串表示。 |