



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

## API Reference

Create Profile Extension Schema

# Create Profile Extension Schema

## Important

**Prerequisites:** You need to **enable** profiles in UCS.

**POST /metadata/profiles/extensions**

## Contents

- **1 Create Profile Extension Schema**
  - 1.1 Description
  - 1.2 Operation
  - 1.3 Response
  - 1.4 Example

## Description

Creates a profile extension schema resource, which is mandatory to query and use profile extension schemas.

- [Read more about Extensions.](#)

## Operation

### Create Profile Extensions Schema

ID	CV.WS.PROF.META.4		
Method	POST		
URL	/metadata/profiles/extensions		
Parameter	Type	Mandatory	Description
<b>Body:</b> The schema to create.<ref>This is the <a href="#">Profile Extension Schema</a> to create.</ref>			
name	string	yes	The unique, case-insensitive name of the extension. The name must start with a letter, and can be followed with letters, numbers, or underscores.  The name is restricted to a maximum of 26 characters (Maximum RDBMS shared limit on creation of index).
type	token <ul style="list-style-type: none"><li>• "single-valued"</li><li>• "multi-valued"</li></ul>	yes	Extensions come in the following forms: <ul style="list-style-type: none"><li>• Single-valued: a single extension record is associated with a given customer.</li><li>• Multi-valued: multiple extension records can be associated with a given customer.</li></ul>
attributes	<a href="#">Attribute Schema</a> []	no	The array of <i>zero or more</i> attributes. Empty when type="classifier".
unique	string[]	no	Lists attributes which

ID	CV.WS.PROF.META.4		
			<p>are unique in the scope of a given customer. The customer profile cannot include several extension records with identical values for these attributes. For example, let's consider a contact extension with the attributes "phone_num", "ext", "time_of_day". To ensure that a given customer does not have two contact extensions with the same phone number, set the following in the extension schema:</p> <pre>unique = ["phone_num", "ext"]</pre>

&lt;references /&gt;

## Response

The Context Management Service API answers with HTTP codes for every request. The following table shows the correct response for a successful request. See [HTTP Response Codes and Errors](#) for further details on the possible codes that this operation can return.

Response	
HTTP code	201
HTTP message	Created
Header	<p>Location: <b><code>\${base_uri}/profiles/extensions/\${extension-name}</code></b></p> <p>where</p> <ul style="list-style-type: none"><li>• <code>\${extension-name}</code> is the name of the extension provided in the request's entity body.</li><li>• <code>\${base_uri}</code> is the base URI of Context Services.</li></ul>
Body	<pre>{ "name": "<b><code>\${extension-name}</code></b>" }</pre> <p>where:</p> <ul style="list-style-type: none"><li>• <code>\${extension-name}</code> is the name of the created extension.</li></ul>

### Example

#### Operation

```
POST http://ucsserver.mycompany.com:8080/cms/profiles/extensions
{
  "name": "Address",
  "type": "single-valued",
  "attributes": [
    {"name": "AddressType", "type": "integer", "default": 0},
    {"name": "Address", "type": "string", "length": 256},
    {"name": "City", "type": "string", "length": 32},
    {"name": "County", "type": "string", "length": 32},
    {"name": "PostCode", "type": "string", "length": 10},
    {"name": "Country", "type": "string", "length": 32}
  ]
}
```

#### Result

```
201 Created
Content-Location http://ucsserver.mycompany.com:8080/cms/profiles/extensions/Address
{"name": "Address"}
```