



CPI Schema

This section gives an example of a Cisco Programmatic Interface (CPI) schema.

```
<?xml version="1.0" encoding="UTF-8" ?>
- <xs:schema targetNamespace="http://www.cisco.com/cpi_10/schema"
  xmlns="http://www.cisco.com/cpi_10/schema"
  xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- The following elements define the cisco extensions for the content of the
  filter element in a <get-config> request. They allow the client to specify the format
  of the response and to select subsets of the entire configuration to be included.-->
  - <xs:element name="config-format-text-block">
    - <xs:annotation>
      <xs:documentation>If this element appears in the filter, then the client is
      requesting that the response data be sent in config command block
      format.</xs:documentation>
    </xs:annotation>
    - <xs:complexType>
      - <xs:sequence>
        <xs:element ref="text-filter-spec" minOccurs="0" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  - <xs:element name="config-format-text-cmd">
    + <xs:complexType>
      - <xs:sequence>
        <xs:element ref="text-filter-spec" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  - <xs:element name="config-format-xml">

  - <xs:annotation>
    <xs:documentation>When this element appears in the filter of a get-config
    request, the results are to be returned in E-DI XML format. The content of this
    element is treated as a filter.</xs:documentation>
  </xs:annotation>
  - <xs:complexType>
    - <xs:complexContent>
      <xs:extension base="xs:anyType" />
    </xs:complexContent>
  </xs:complexType>
  </xs:element>
  - <!-- These elements are used in the filter of a <get> to specify operational data to
  return.-->
  - <xs:element name="oper-data-format-text-block">
    - <xs:complexType>
```

```

- <xs:sequence>
  <xs:element name="show" type="xs:string" maxOccurs="unbounded" />
</xs:sequence>
</xs:complexType>
</xs:element>
- <xs:element name="oper-data-format-xml">
  - <xs:complexType>
    - <xs:sequence>
      <xs:any />
    </xs:sequence>
  </xs:complexType>
</xs:element>
- <!-- When config-format-text format is specified, the following describes the content
of the data element in the response -->
  - <xs:element name="cli-config-data">
    - <xs:complexType>
      - <xs:sequence>
        - <xs:element name="cmd" type="xs:string" maxOccurs="unbounded">
          - <xs:annotation>
            <xs:documentation>Content is a command. May be multiple
              lines.</xs:documentation>
          </xs:annotation>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
- <xs:element name="cli-config-data-block" type="xs:string">
  - <xs:annotation>
    <xs:documentation>The content of this element is the device configuration as
it would be sent to a terminal session. It contains embedded newline characters that
must be preserved as they represent the boundaries between the individual command
lines</xs:documentation>
  </xs:annotation>
</xs:element>
- <xs:element name="text-filter-spec">
  - <xs:annotation>
    <xs:documentation>If this element is included in the config-format-text
element, then the content is treated as if the string was appended to the "show
running-config" command line.</xs:documentation>
  </xs:annotation>
</xs:element>
- <xs:element name="cli-oper-data-block">
  - <xs:complexType>
    - <xs:annotation>
      <xs:documentation>This element is included in the response to get
operation. Content of this element is the operational data in text
format.</xs:documentation>
    </xs:annotation>
    - <xs:sequence>
      - <xs:element name="item" maxOccurs="unbounded">
        - <xs:complexType>
          - <xs:sequence>
            <xs:element name="show" />
            <xs:element name="response" />
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
- <xs:element name="xml-oper-data">
  - <xs:complexType>
    - <xs:annotation>

```

```

        <xs:documentation>This element is included in the response to      get
operation. Content of this element is the operational data      in xml
format.</xs:documentation>
      </xs:annotation>
    - <xs:sequence>
      <xs:any />
    </xs:sequence>
  </xs:complexType>
</xs:element>
- <xs:element name="xml-config-data">

  - <xs:complexType>
    - <xs:annotation>
      <xs:documentation>This element is included in the response to      get-config
and get operations. Content of this element is the      configuration data in xml
format.</xs:documentation>
    </xs:annotation>
    - <xs:sequence>
      <xs:any />
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

