



**Australian Government**

**Department of Education, Employment and Workplace Relations**

# **ICAPRG427A Use XML effectively**

**Release: 1**

## ICAPRG427A Use XML effectively

### Modification History

Release	Comments
Release 1	This Unit first released with <i>ICAll Information and Communications Technology Training Package version 1.0</i>

### Unit Descriptor

This unit describes the performance outcomes, skills and knowledge required to create eXtensible markup language (XML) documents, web service applications, and client applications to consume web services.

### Application of the Unit

This unit applies to programmers who build Windows or web-based applications that access data in XML format. Others may be programmers who develop and consume web services.

These individuals may work as web developer, web programmer, and application programmer.

### Licensing/Regulatory Information

No licensing, legislative, regulatory or certification requirements apply to this unit at the time of endorsement but users should confirm requirements with the relevant federal, state or territory authority.

### Pre-Requisites

Not applicable.

### Employability Skills Information

This unit contains employability skills.

## Elements and Performance Criteria Pre-Content

Element	Performance Criteria
<i>Elements describe the essential outcomes of a unit of competency.</i>	<i>Performance criteria describe the performance needed to demonstrate achievement of the element. Where bold italicised text is used, further information is detailed in the required skills and knowledge section and the range statement. Assessment of performance is to be consistent with the evidence guide.</i>

## Elements and Performance Criteria

1. Represent structured data with XML	1.1 Gather information by analysing data, documents and problem domains 1.2 Write well-formed XML document by incorporating proper <b><i>structure</i></b> and <b><i>syntax</i></b> 1.3 Validate XML document using <b><i>validation tools</i></b> 1.4 Format XML document using <b><i>styling tools</i></b>
2. Access and manipulate XML document	2.1 Create new XML document using an <b><i>XML parser application programming interface</i></b> (API) 2.2 Access and traverse elements and attributes using an XML parser API 2.3 Modify elements and attributes using an XML parser API 2.4 Delete elements and attributes using an XML parser API 2.5 Transform XML document into data object using an XML parser API 2.6 Transform data object into XML document using an XML parser API
3. Create service-oriented application using XML	3.1 Define web services architecture and <b><i>platform elements</i></b> 3.2 Develop and deploy web service applications using a language 3.3 Develop a client application to consume web service using a <b><i>language</i></b>

## Required Skills and Knowledge

*This section describes the skills and knowledge required for this unit.*

### Required skills

- analytical skills to:
  - analyse and comprehend interdependencies between data fields
  - examine and define structure requirements for data
- communication skills to:
  - liaise with developers on programming matters
  - seek requirements and information from business and technical experts
- literacy skills to:
  - read and interpret program specifications developed by business and technical experts
  - read and interpret technical documentation
  - write a basic documentation
- numeracy skills to identify order and position of element in a sequence, such as first and last
- problem-solving skills to apply basic debugging techniques in the context of software and application development
- research skills to:
  - locate and interrogate complex and varied sources of information
  - source information from available sources
- technical skills to:
  - develop a small scale application
  - perform basic operations of a computer system.

### Required knowledge

- basic knowledge of:
  - hierarchy and tree structure
  - programming concept and language
  - world wide web (WWW)
  - data modelling
  - hypertext markup language (HTML)
  - information system features
  - source characteristics.

## Evidence Guide

*The evidence guide provides advice on assessment and must be read in conjunction with the performance criteria, required skills and knowledge, range statement and the Assessment Guidelines for the Training Package.*

<b>Overview of assessment</b>	
<b>Critical aspects for assessment and evidence required to demonstrate competency in this unit</b>	<p>Evidence of the ability to:</p> <ul style="list-style-type: none"> <li>• produce, validate, transform, format and transport XML data</li> <li>• exchange data between different applications and different platforms by creating and consuming web services.</li> </ul>
<b>Context of and specific resources for assessment</b>	<p>Assessment must ensure access to:</p> <ul style="list-style-type: none"> <li>• tools in accessing and developing a data source, such as integrated development environment (IDE), text editor and web browser</li> <li>• specific requirements, including client and functionality requirements</li> <li>• web server and software that host services to deploy and test web service application</li> <li>• libraries to develop web service and client applications, such as SOAP libraries</li> <li>• appropriate learning and assessment support when required</li> <li>• modified equipment for people with special needs.</li> </ul>
<b>Method of assessment</b>	<p>A range of assessment methods should be used to assess practical skills and knowledge. The following examples are appropriate for this unit:</p> <ul style="list-style-type: none"> <li>• evaluation of ability to: <ul style="list-style-type: none"> <li>• develop XML applications to generate a report in HTML or portable document format (PDF) file by applying certain presentation style using styling tools</li> <li>• develop web or stand-alone applications that access and store information in XML data format</li> <li>• develop and consume web service applications to solve interoperability problem by writing client code in different platform</li> </ul> </li> <li>• verbal or written questioning to assess candidate's knowledge of XML.</li> </ul>
<b>Guidance information for assessment</b>	<p>Holistic assessment with other units relevant to the industry sector, workplace and job role is recommended, where appropriate.</p> <p>Assessment processes and techniques must be culturally</p>

	<p>appropriate, and suitable to the communication skill level, language, literacy and numeracy capacity of the candidate and the work being performed.</p> <p>Indigenous people and other people from a non-English speaking background may need additional support.</p> <p>In cases where practical assessment is used it should be combined with targeted questioning to assess required knowledge.</p>
--	---

## Range Statement

*The range statement relates to the unit of competency as a whole. It allows for different work environments and situations that may affect performance. Bold italicised wording, if used in the performance criteria, is detailed below. Essential operating conditions that may be present with training and assessment (depending on the work situation, needs of the candidate, accessibility of the item, and local industry and regional contexts) may also be included.*

<b>Structure</b> must include:	<ul style="list-style-type: none"> <li>• child element</li> <li>• root element</li> <li>• sub-child element.</li> </ul>
<b>Syntax</b> must include these rules:	<ul style="list-style-type: none"> <li>• case sensitive</li> <li>• closing tag</li> <li>• entity reference</li> <li>• proper nesting</li> <li>• quoted attribute values</li> <li>• single-root element.</li> </ul>
<b>Validation tools</b> may include:	<ul style="list-style-type: none"> <li>• document type definition (DTD)</li> <li>• validating parsers</li> <li>• XML schema</li> <li>• XSD schema validator.</li> </ul>
<b>Styling tools</b> may include:	<ul style="list-style-type: none"> <li>• cascading style sheets (CSS)</li> <li>• eXtensible stylesheet language (XSL)</li> <li>• eXtensible stylesheet language transformations (XSLT).</li> </ul>
<b>XML parser application programming interface</b> may include:	<ul style="list-style-type: none"> <li>• data binding</li> <li>• document object model (DOM)</li> <li>• language-specific API, such as XML API in .NET, JAXP in Java, and E4X in ActionScript</li> <li>• simple API for XML (SAX)</li> <li>• XMLPULL.</li> </ul>
<b>Platform elements</b> may include:	<ul style="list-style-type: none"> <li>• simple object access protocol (SOAP)</li> <li>• universal description, discovery and integration (UDDI)</li> <li>• web services description language (WSDL).</li> </ul>
<b>Language</b> may include:	<ul style="list-style-type: none"> <li>• Java</li> <li>• Net languages</li> <li>• PHP</li> <li>• Python.</li> </ul>

## **Unit Sector(s)**

Programming and software development