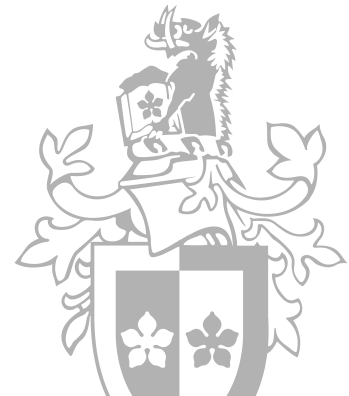


# COS30041 Creating Secure and Scalable Software

## Lecture 09 Web Services



SWIN  
BUR  
\* NE \*

SWINBURNE  
UNIVERSITY OF  
TECHNOLOGY

Commonwealth of Australia  
*Copyright Act 1968*

**Notice for paragraph 135ZXA (a) of the *Copyright Act 1968***

### **Warning**

This material has been reproduced and communicated to you by or on behalf of Swinburne University of Technology under Part VB of the *Copyright Act 1968* (the Act).

The material in this communication may be subject to copyright under the Act. Any further reproduction or communication of this material by you may be the subject of copyright protection under the Act.

Do not remove this notice.

# Learning Objectives

- After studying the lecture material, you will be able to
  - Understand and describe what web services is
  - Understand and discuss two types of web services
    - Traditional “Big” WS
    - RESTful WS
  - Understand and discuss when to use “Big” WS and when to use RESTful WS

# Pre-requisite

- Object Oriented Programming
- Some experiences on XML would be an advantage

# Outline

- Web Services
- Two types of Web Services
  - Traditional “Big” Web Services
  - RESTful Web Services

# Web Services

- A way to build and integrate software applications running on different platforms and framework via HTTP
- Different software components can be integrated together via HTTP, even though they are implemented in different programming languages
  - Similar to CORBA (Common Object Request Broker Architecture)

# Two Types of Web Services

## Traditional “Big”

- Aim for system integration
  - Vs CORBA
- Use WSDL (Web Services Description Language)
- Use SOAP (simple object access protocol), on top of HTTP
- Using XML for values
- No standardized naming for methods

## RESTful

## (REpresentational State Transfer)

- An architecture style
- Use WADL (Web Application Description Language)
- Use HTTP
- Use XML / JSON for values
- Standardized HTTP verbs to name methods
  - POST / GET / PUT / DELETE
  - Create / Review / Update / Delete

# Suitability

## “BIG”

- Integrate Enterprise Applications
- Address advanced QoS requirements
- Easy to support the WS-\* set of protocols
  - Standards for security, reliability, ...
- Easy to interoperate with other WS-\* conforming clients and servers

## RESTful

- Integrate over the web
- Easy to write apps to address the constraints of the REST style
- Flexible for clients to choose all or parts of the RESTful web services to consumes