A web service is an instance of a more general notion of a service.

A web service is: A loosely coupled, reusable software component that encapsulates discrete functionality, which may be distributed and programmatically accessed. A web service is a service that is accessed using standard Internet and XML-based protocols.
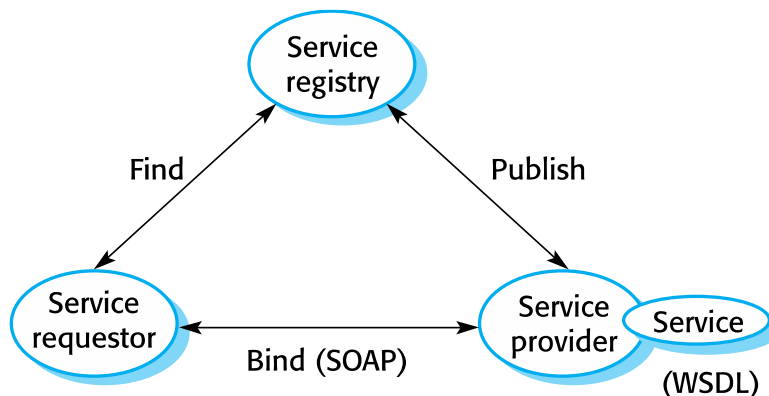
Services are reusable components that are independent (no requires interface) and are loosely coupled.

Benefits of service-oriented (SAO) approach:

- ✧ Services can be offered by any service provider inside or outside of an organisation.
- ✧ The service provider makes information about the service public.
- ✧ Applications can delay the binding of services until they are deployed or until execution.
- ✧ Opportunistic construction of new services is possible.

Service-based applications may be constructed by linking services from various providers using either a standard programming language or a specialized workflow language.

Service-oriented architectures: A means of developing distributed systems where the components are stand-alone services. Services may execute on different computers from different service providers

SOAP: A message exchange standard that supports service communication

WSDL (Web Service Definition Language): This standard allows a service interface and its bindings to be defined
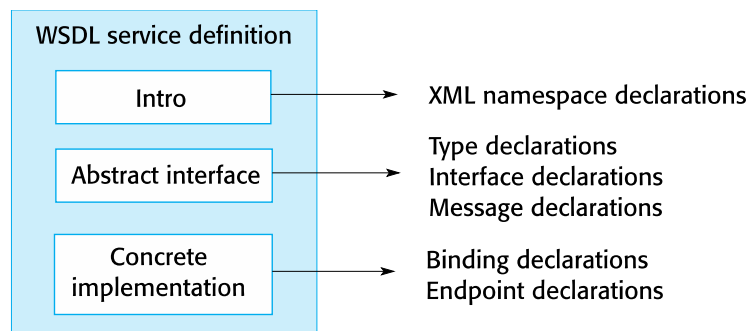
WS-BPEL: A standard for workflow languages used to define service composition

The service interface is defined in a service description expressed in WSDL (Web Service Description Language).

The WSDL specification defines:

- ✧ What operations the service supports and the format of the messages that are sent and received by the service
- ✧ How the service is accessed - that is, the binding maps the abstract interface onto a concrete set of protocols

✧ Where the service is located. This is usually expressed as a URI (Universal Resource Identifier)

```
┌─────────────────────────────┐
│ WSDL service definition      │
│  ┌───────────────────┐       │
│  │       Intro        │───────┼──→  XML namespace declarations
│  └───────────────────┘       │
│  ┌───────────────────┐       │     Type declarations
│  │ Abstract interface │───────┼──→  Interface declarations
│  └───────────────────┘       │     Message declarations
│  ┌───────────────────┐       │
│  │     Concrete       │───────┼──→  Binding declarations
│  │  implementation    │       │     Endpoint declarations
│  └───────────────────┘       │
└─────────────────────────────┘
```

Service engineering. The development of dependable, reusable services

✧ Software development for reuse

Software development with services. The development of dependable software where services are the fundamental components.

✧ Software development with reuse

A service is a loosely-coupled, reusable software component that encapsulates discrete functionality which may be distributed and programmatically accessed. A web service is a service that is accessed using standard Internet and XML-based protocols

Difference between service and a component is services are independent.

✧ Services do not have a 'requires' interface
✧ Services rely on message-based communication with messages expressed in XML.

WSDL specification components:

✧ The 'what' part of a WSDL document, called an interface, specifies what operations the service supports, and defines the format of the messages that are sent and received by the service.

✧ The 'how' part of a WSDL document, called a binding, maps the abstract interface to a concrete set of protocols.

✧ The 'where' part of a WSDL document describes the location of a specific Web service implementation (its endpoint).