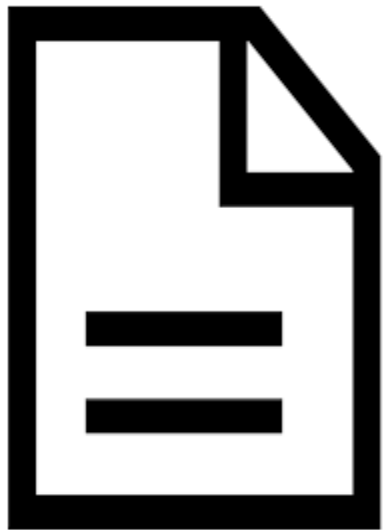


# MINGGU KE 2

## Pengenalan Web Service



## Deskripsi

Materi membahas contoh-contoh implementasi web service yang sudah digunakan oleh masyarakat. Untuk mengenal lebih jauh tentang layanan web service eksplorasi pembahasan tentang jenis web service dan manfaatnya, teknologi web service, arsitektur web servis, operasi web service, keuntungan web service.

## Tujuan Pembelajaran

Dengan mengikuti pembelajaran ini, mahasiswa mampu

1. Mengetahui berbagai layanan web service ditengah-tengah masyarakat
2. Memahami layanan webservice, mampu melatih pembuatan layanan web service
3. Melatih atau mencoba membuat layanan web service XML,SOAP, WSDL



# A. Pengenalan Web Service

Apakah Web Service itu ?

Web Service dengan Website apakah mempunyai kesamaan ?

Web Service ternyata sangat berbeda dengan website. Perbedaan yang paling terlihat adalah Website dibuat untuk memiliki tampilan atau user interface yang bagus sedangkan Web Service tidak memiliki tampilan. Mengapa Web Service tidak memiliki tampilan (user interface) ?

Web Service adalah Sebuah aplikasi lintas platform yang dapat diakses melalui jaringan (intranet dan internet ) dimana dalam aplikasi tersebut menyediakan berbagai fungsi dengan tujuan digunakan untuk interaksi aplikasi satu dengan aplikasi yang lain.

Web service dapat diartikan juga sebuah metode pertukaran data, tanpa memperhatikan dimana sebuah database ditanamkan, dibuat dalam bahasa apa, sebuah aplikasi yang mengkonsumsi data, dan di platform apa sebuah data itu dikonsumsi. Web service mampu menunjang interoperabilitas



# Web Service

Web pada umumnya digunakan untuk melakukan respon dan request yang dilakukan antara client dan server.

## Contoh :

Implementasi web service adalah sistem login seperti yang ada di Kaskus ataupun detik.com

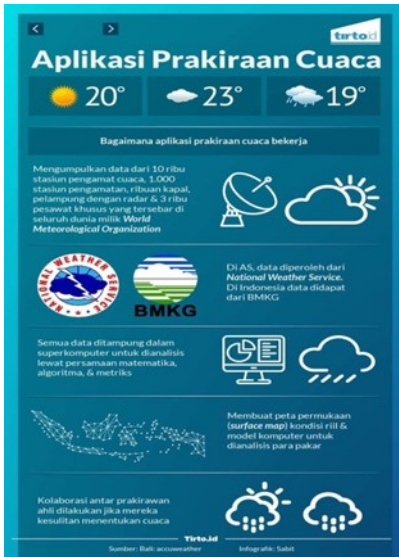
## Jadi apakah Web Service itu?

Web Service adalah aplikasi yang dibuat agar dapat dipanggil atau diakses oleh aplikasi lain melalui internet dengan menggunakan format pertukaran data sebagai format pengiriman pesan.





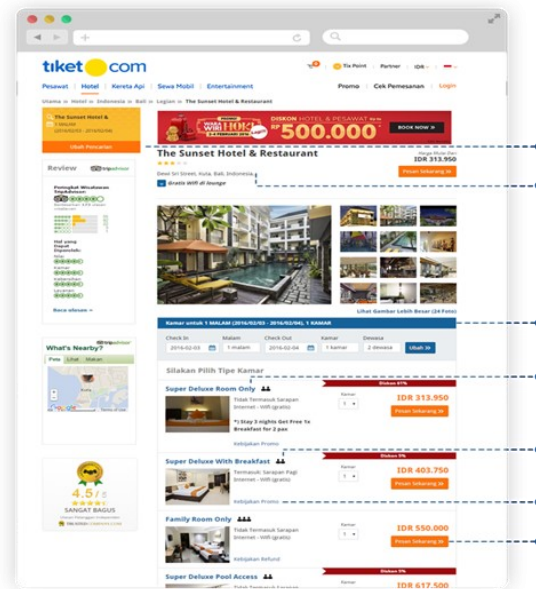
# B. Jenis-jenis Web Service



## 1. Reusable application-components

Jenis web service yang dimanfaatkan secara periodik atau berulang-ulang.

Contoh : layanan konversi mata uang, info cuaca, penterjemahan Bahasa dll.



## 2. Connect Existing Software

Web service dapat dimanfaatkan untuk mengintegrasikan antar berbagai aplikasi yang berbeda.

Contoh : aplikasi traveling diintegrasikan dengan aplikasi pemesanan hotel, wisata, rental kendaraan dll meskipun dengan platform, lokasi dan pemrograman yang berbeda.

# C. Teknologi Pendukung Web Service

Dalam pengoperasian, web service menerapkan lima komponen teknologi yang mendukung kinerjanya antara lain :

1. XML (Extensible Markup Language)
2. SOAP (Simple Object Access Protocol)
3. Web Service Description Language (WSDL)
4. Service Publication and Discovery (UDDI)
5. Common Internet Protocols (HTTP, TCP/IP)

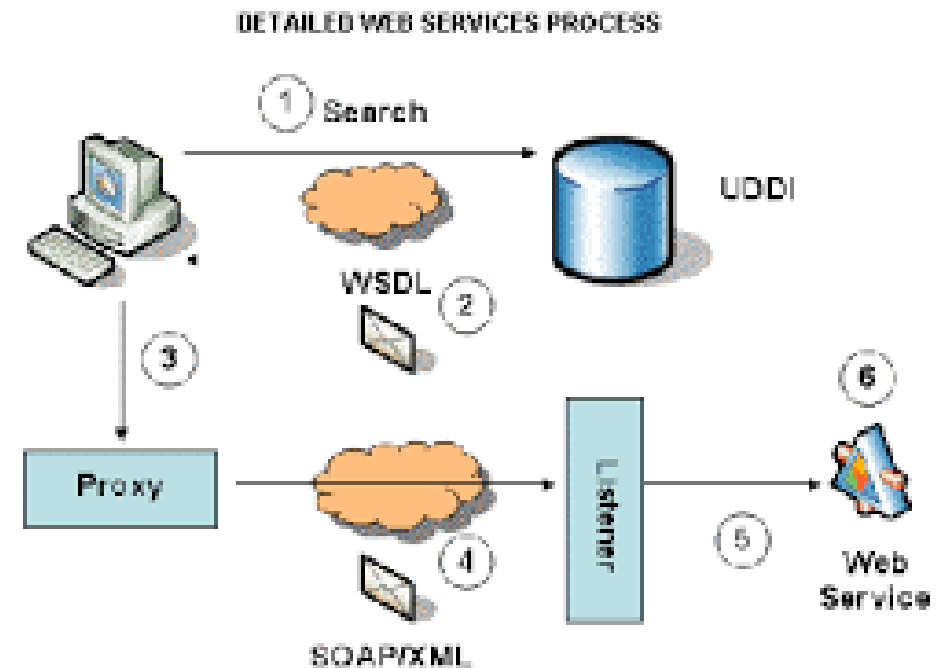


Figure 1: The process flow of a Web service

# Komponen Teknologi, web service

## 1. XML (Extensible Markup Language)

Merupakan standar untuk markup dokumen yang disahkan oleh W3C (World Wide Web Consortium). XML mendefinisikan sintaks yang umum digunakan untuk markup data secara sederhana. XML menyediakan format standar untuk dokumen komputer. Format ini cukup fleksibel untuk disesuaikan dengan domain yang beragam seperti website, pertukaran data elektronik, grafik vektor, hierarki, serialisasi objek, Remote Procedure Call (RPC), dan sistem voice mail (Harold dan Means, 2002). XML merupakan bagian penting bagi pengembang yang ingin membangun web service. XML berbasis teks, sehingga bersifat platform independent. Sifat ini membuat pertukaran data antar platform menjadi mungkin.

**Contoh Dokumen XML :** `<?xml version="1.0" encoding="utf-8" ?>`

```
<karyawan instansi="UBSI">
  <nama>Rachmat Hidayat, M.Kom</nama>
  <nik> </nik>
  <jabatan>Staf Akademi</jabatan>
  <status>Tetap</status>
  <alamat>Tangerang</alamat>
</karyawan>
```



eXtensible  
M Markup  
L Language



# Komponen Teknologi, web service

## 2. SOAP (Simple Object Access Protocol)

adalah sebuah protokol berbasis XML dengan komunikasi bergaya RPC (*Remote Procedure Call*) yang menyediakan cara standar untuk memaketkan pesan (Chappell dan Jewell, 2002). SOAP mendefinisikan cara dalam pembungkusan (encapsulation) dan pertukaran (exchange) pesan.

Dokumen XML yang terdiri dari komponen:

1. **SOAP Envelope**, mengidentifikasi dokumen XML sebagai SOAP message
2. **Elemen Header** (opsional)
3. **Elemen Body**, berisi informasi pemanggilan dan responsnya
4. **Elemen Fault** (opsional), berisi error yang terjadi saat pemanggilan diproses

**Contoh Dokumen XML :**

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap=" envelope" soap:encodingStyle=" encoding">
  <soap:Header>....</soap:Header>
  <soap:Body>
    <soap:Fault>.....</soap:Fault>
  </soap:Body>
</soap:Envelope>
```



Simple  
Object  
Access  
Protocol

# Komponen Teknologi, web service

## 3. *Web Service Description Language (WSDL)*

adalah sebuah dokumen dalam format XML yang isinya menjelaskan informasi detail sebuah Web Service. Di dalam WSDL dijelaskan method-method yang tersedia dalam web service, parameter apa saja yang diperlukan untuk memanggil sebuah method, dan hasil atau tipe data yang dikembalikan oleh method yang dipanggil.

### Format WSDL

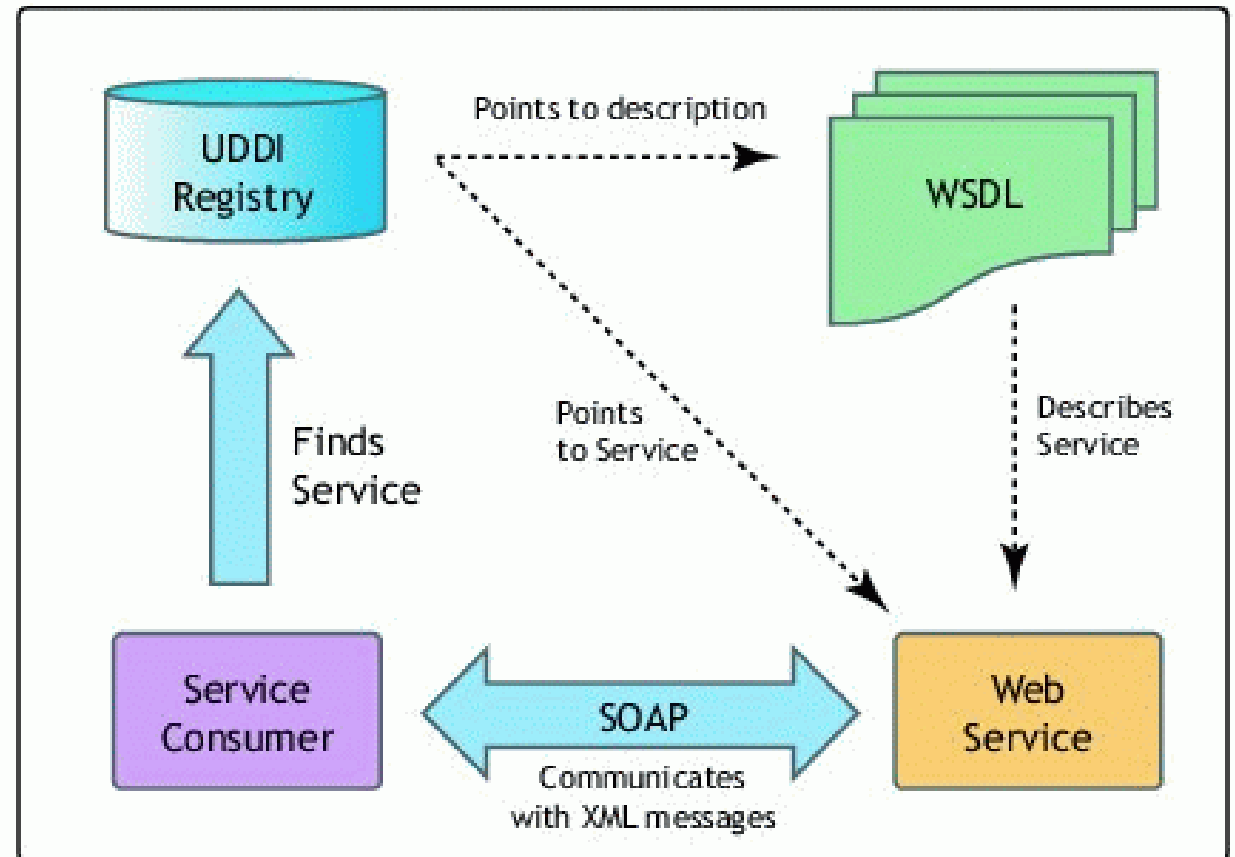
1. **Elemen <PortType>**  
Mendefinisikan namaweb service, operasi-operasi yang dilakukan, dan message terlibat.
2. **Elemen <message>**  
Mendefinisikan message yang digunakan oleh web service.
3. **Elemen <binding>**  
Mendefinisikan protocol komunikasi yang digunakan.
4. **Elemen <type>**  
Mendefinisikan tipe-tipe data yang digunakan.

```
<definitions>  
  
<types> ... </types>  
  
<message> ... </message>  
  
<portType> ... </portType>  
  
<binding> ... </binding>  
  
</definitions>
```

## Komponen Teknologi, web service

### 4. *Service Publication and Discovery (UDDI), Universal Description, Discovery and Integration (UDDI)*

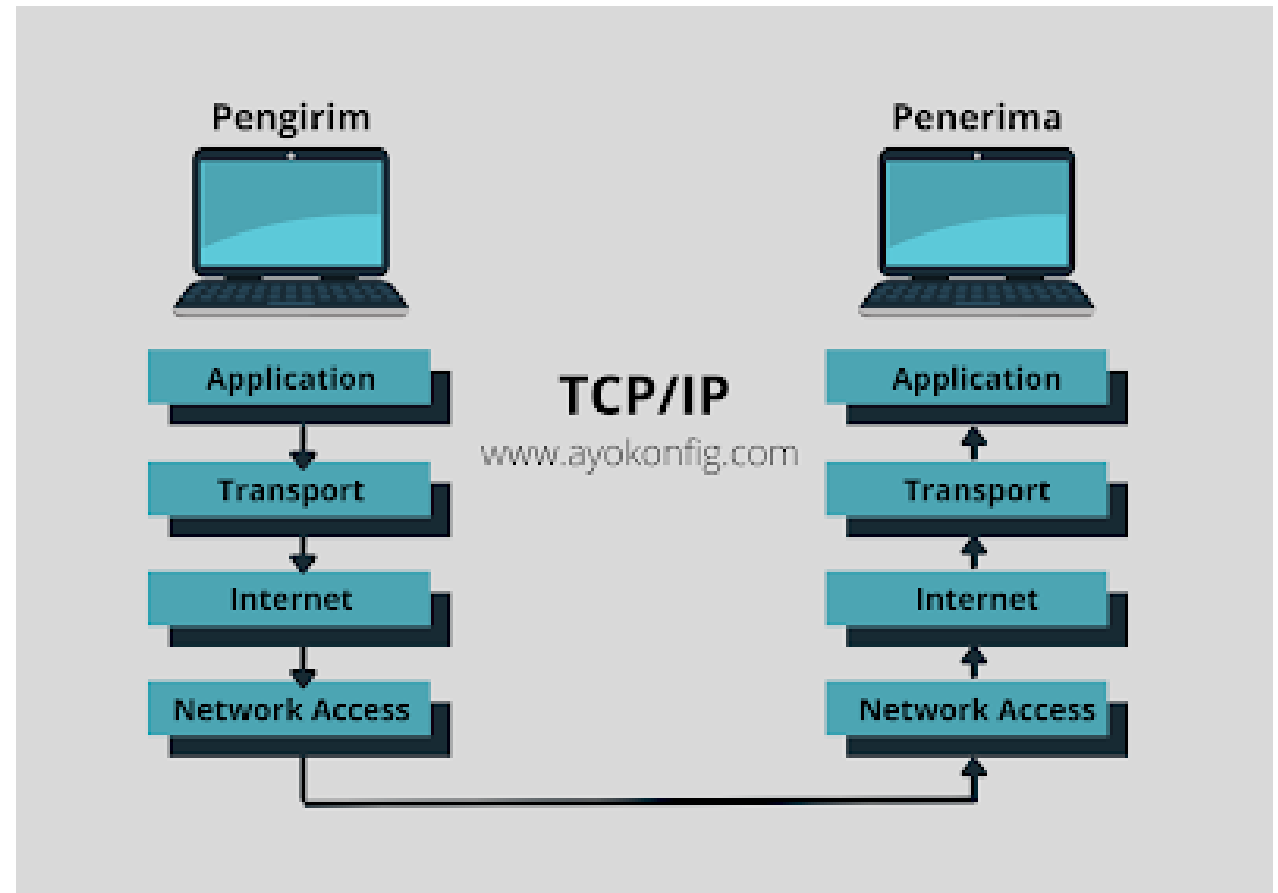
adalah suatu directory service yang digunakan untuk mendaftarkan dan mencari Web Service.



# Komponen Teknologi, web service

## 5. Common Internet Protocols (HTTP, TCP/IP)

Dalam hal ini, HTTP maupun TCP/IP berperan sebagai common internet protocol yang berfungsi sebagai transport layer.



## D. Arsitektur Web Service

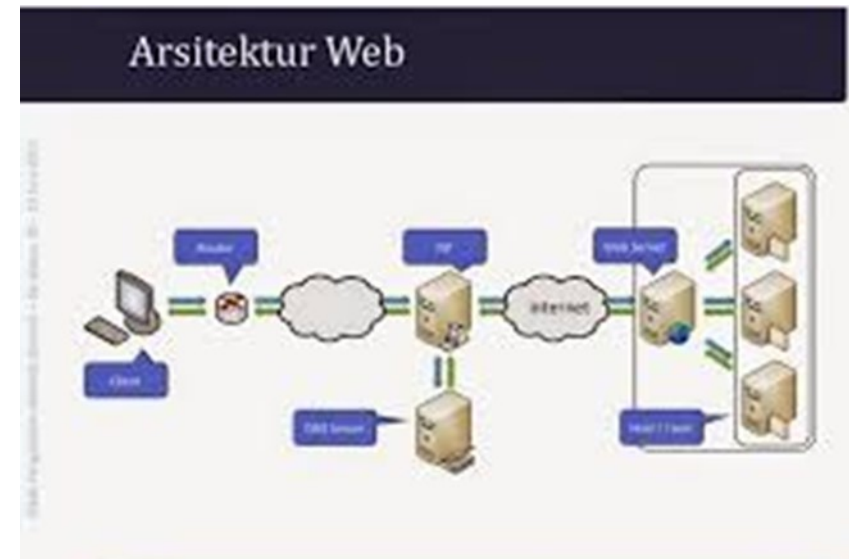
## Arsitektur mendeskripsikan struktur

Menurut (Bassetal.1998), Arsitektur dari sistem perangkat lunak (software) terdiri dari strukturnya, dekomposisinya ke dalam komponen-komponen, dan *interface* dan hubungannya.

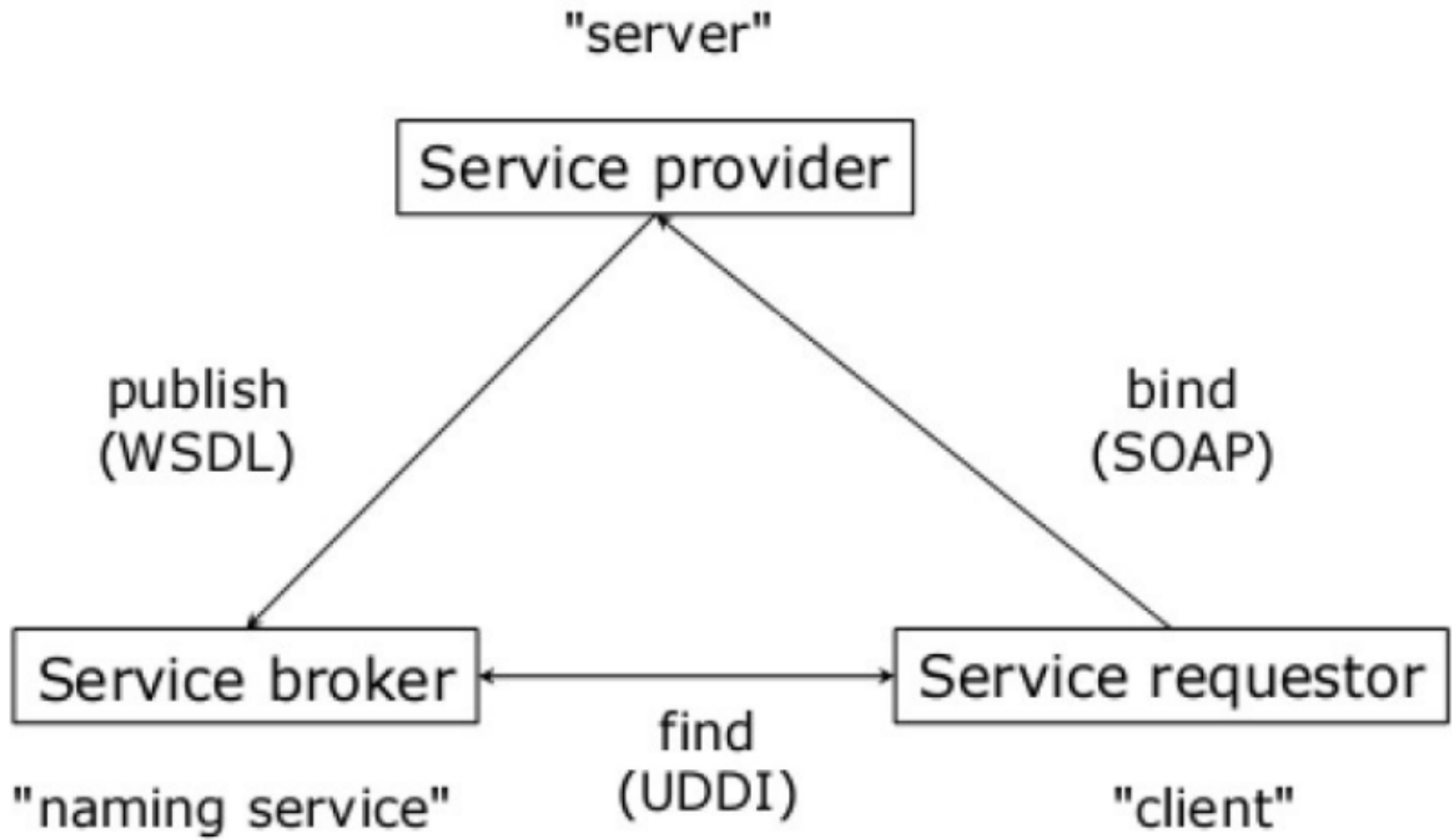
Arsitektur **menjelaskan aspek statis** dan **dinamis** dari sistem software sehingga dapat diperkirakan rancangan bangunan (*building design*) dan diagram alir (*flow chart*) bagi suatu produk software.

Web service memiliki 3(tiga) entitas dalam arsitekturnya, yaitu:

1. **Service Requester** (peminta layanan)  
merupakan aplikasi yang bertindak sebagai klien dari Web Service yang mencari dan memulai interaksi terhadap layanan yang disediakan
2. **Service Provider/Broker** (penyedia layanan)  
merupakan pemilik Web Service yang berfungsi menyediakan kumpulan operasi dari Web Service.
3. **Service Registry** (daftar layanan)  
merupakan tempat dimana Service provider mempublikasikan layanannya. Pada arsitektur Web Service, Service registry bersifat optional.







# Bagaimana Web Service Beroperasi?

## Sisi Server

- ☐ Membuat fungsi utama/core function
- ☐ Membuat service wrapper berupa XML-RPC atau SOAP
- ☐ Membuat deskripsi service berupa WSDL atau instruksi integrasi XML-RPC (memuat semua method public, argumen dan return valuenya); plus dokumentasi yang human readable
- ☐ Deploy (rilis) service
- ☐ Daftarkan service tersebut melalui UDDI agar discoverable

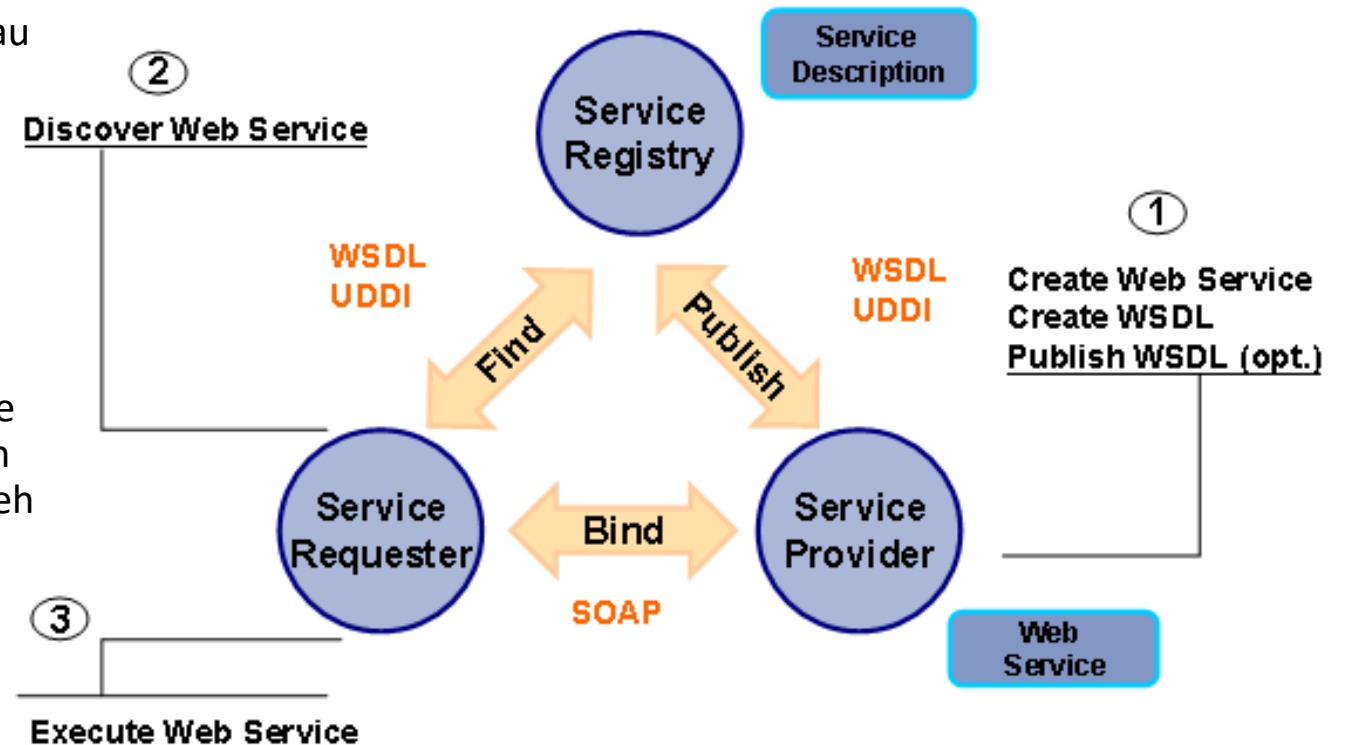
## Sisi Client

- ☐ Mencari service melalui UDDI
- ☐ Mengambil service description file berupa WSDL atau instruksi XML-RPC
- ☐ Membuat klien XML-RPC atau SOAP (dapat berupa fungsi lokal atau pesan XML untuk dikirim → berdasarkan WSDLnya)
- ☐ Memanggil remote service tersebut

# E. Operasi-operasi Web Service

Secara umum, web service memiliki 3 (tiga) operasi yang terlibat di dalamnya, yaitu :

1. **Publish/Unpublish**  
Menerbitkan/menghapus layanan ke dalam atau dari registrasi
2. **Find**  
Service requestor mencari dan menemukan layanan yang dibutuhkan.
3. **Bind**  
Service requestor setelah menemukan layanan yang dicarinya, kemudian melakukan binding ke service provider untuk melakukan interaksi dan mengakses layanan/service yang disediakan oleh service provider.



## F. Keuntungan Menggunakan Web Service

1. *Web Service* menyediakan interoperabilitas antar berbagai aplikasi perangkat lunak yang berjalan pada platform yang berbeda.
2. *Web Service* menggunakan standard dan protocol yang bersifat terbuka.
3. *Web service* mengijinkan penggunaan kembali service dan komponen didalam suatu infrastruktur.
4. *Web service* dapat secara bebas digunakan (loosely coupled) dengan demikian memudahkan suatu pendekatan terdistribusi ke peintegrasian aplikasi



## G. Kerugian Menggunakan Web Service

1. Karakteristik standard Web Service saat ini masih dalam tahap perkembangan awal dibandingkan open standard komputer terdistribusi yang lebih matang seperti CORBA. Ini nampaknya akan merupakan suatu kerugian yang temporer ketika kebanyakan vendor sudah merasa terikat dengan standard OASIS untuk menerapkan Mutu dari aspek service dari produk mereka. *Web Service* menggunakan standard dan protocol yang bersifat terbuka.
2. Web Service dapat saja memiliki performance/kinerja yang lemah dibandingkan dengan pendekatan komputasi terdistribusi lain seperti RMI, CORBA, atau DCOM. Ini merupakan suatu trade-off yang umum ketika memilih format yang text-based.



## Case Study: Implementasi Web Service Menggunakan NuSOAP

Gambaran umum case study sebagai berikut:

1. Sumber case study dari salah satu artikel yang dapat di pelajari melalui link: [Implementasi Web Service Dengan SOAP Menggunakan NuSOAP \(Bag 1\) - Tutorial PHP \(rosihanari.net\)](https://rosihanari.net/Implementasi-Web-Service-Dengan-SOAP-Menggunakan-NuSOAP-Bag-1-Tutorial-PHP)
2. Konsep case study tersebut membutuhkan dua komputer, artinya untuk mendukung penyelesaian di atas minimal dua mahasiswa berpasangan dengan menjadikan laptop sebagai client dan server.
3. Karena case study berbasis client server maka konsep koneksi antar dua computer mahasiswa wajib membuka Kembali mata kuliah Jaringan Komputer pada semester 4
4. Mahasiswa mempelajari dan mempraktikan berdasarkan sumber referensi tersebut pada point tersebut.
5. Simulasi hasil praktik dapat di lihat melalui video pada link : <https://tinyurl.com/Vidukasi-TWS>
6. Mahasiswa mendokumentasikan jika menemukan debug pada saat mempraktikan. Dokumentasi debugging dapat menggunakan format sebagai berikut

## Case Study: Implementasi Web Service Menggunakan NuSOAP

Programmer	Temuan Debug	Perbaikan Coding	Hasil/tampilan
NIM Nama	Mahasiswa dapat screen shoot debug	Mahasiswa menuliskan code sebagai solusi dari penanganan debug yang ditemukan	Mhs dapat SS tampilan output
NIM Nama	Mahasiswa dapat screen shoot debug	Mahasiswa menuliskan code sebagai solusi dari penanganan debug yang ditemukan	Mhs dapat SS tampilan output
dst	dst	dst	dst

**Penilaian project minggu ke 2 mengacu pada Rencana  
Tugas Mahasiswa (RTM)**





Terima kasih!