www.sbVB.com.br

cloud computing and web services

- Version 2013_09_25

- By Sergio Barbosa Villas-Boas
  - www.sbVB.com.br
  - sbvb@sbvb.com.br

Preamble

Preamble

- Academic presentation of sbVB
- Brief History of Everything

What is Cloud Computing ?

- Cloud computing is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet).

- Many services are delivered by the cloud.
  - Software as a service (SaaS)
  - Storage as a service (STaaS)
  - Desktop virtualization
  - others

Purpose of this course

- Study, develop and investigate the skills of "software development for cloud computing"
  - emphasis on web services implemented with java programming

- Course contents
  1. history and analysis of the economic environment of cloud computing
  2. software architecture for cloud, its relations with mobile and business models related to it.
  3. brief review of java
  4. Software development for cloud

web services & soa

Conceiving a new system

Basic recipe

- Determine the main entities (in the database)
- Determine the main use-cases
- Determine the system's actors (User-Groups)
- Determine the use-case – actor authorization table (as data for DOUA).

Some definitions

- “Manage” means CRUD (Create Retrieve, Update, Deleta)

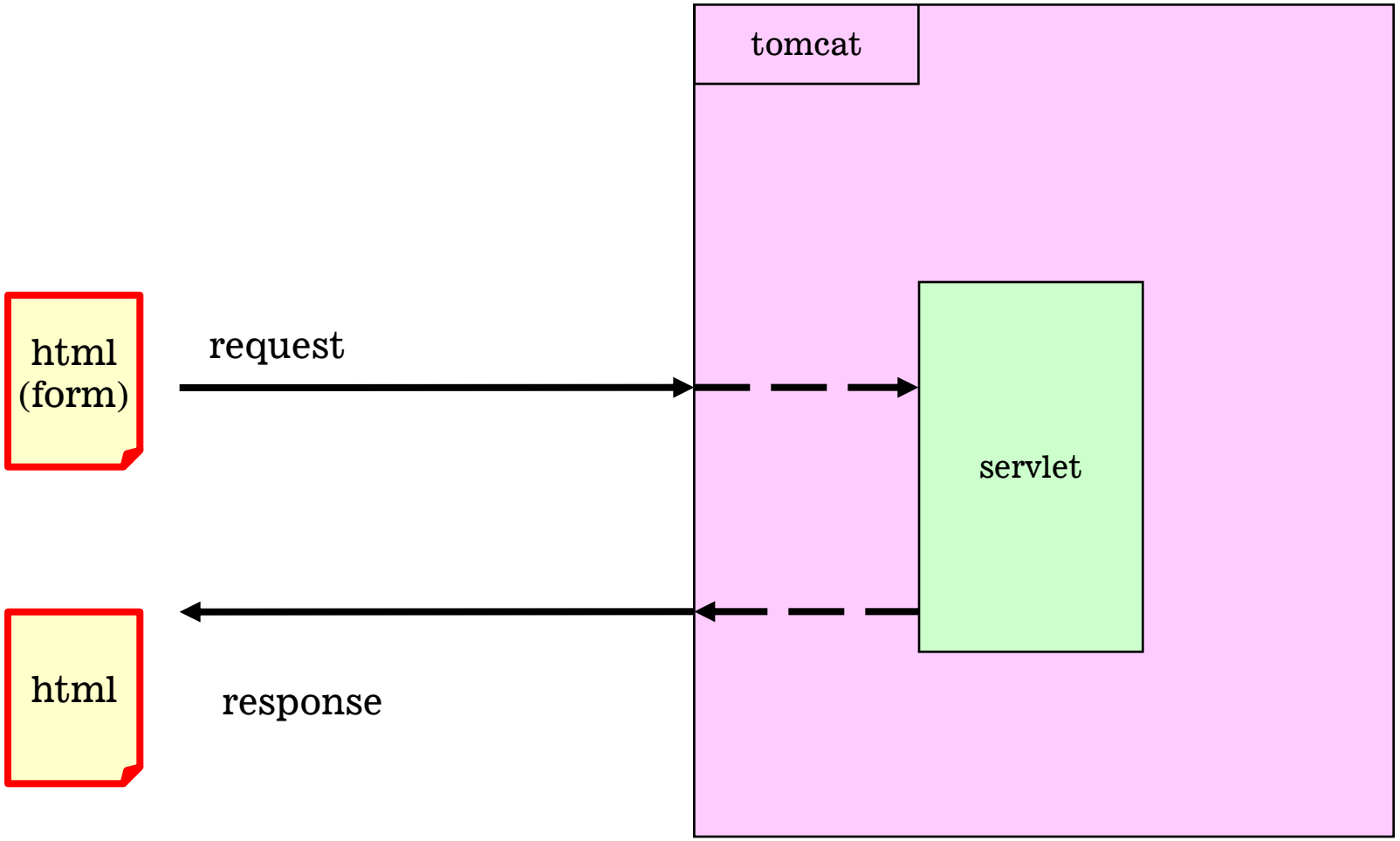
Inovação

Escolha estratégica de arquitetura

- Da mesma forma que a tecnologia, é muito custoso mudar a arquitetura de um sistema web.
  - Portanto é conveniente que se pense nos objetivos de longo prazo do sistema na hora de definir a arquitetura.

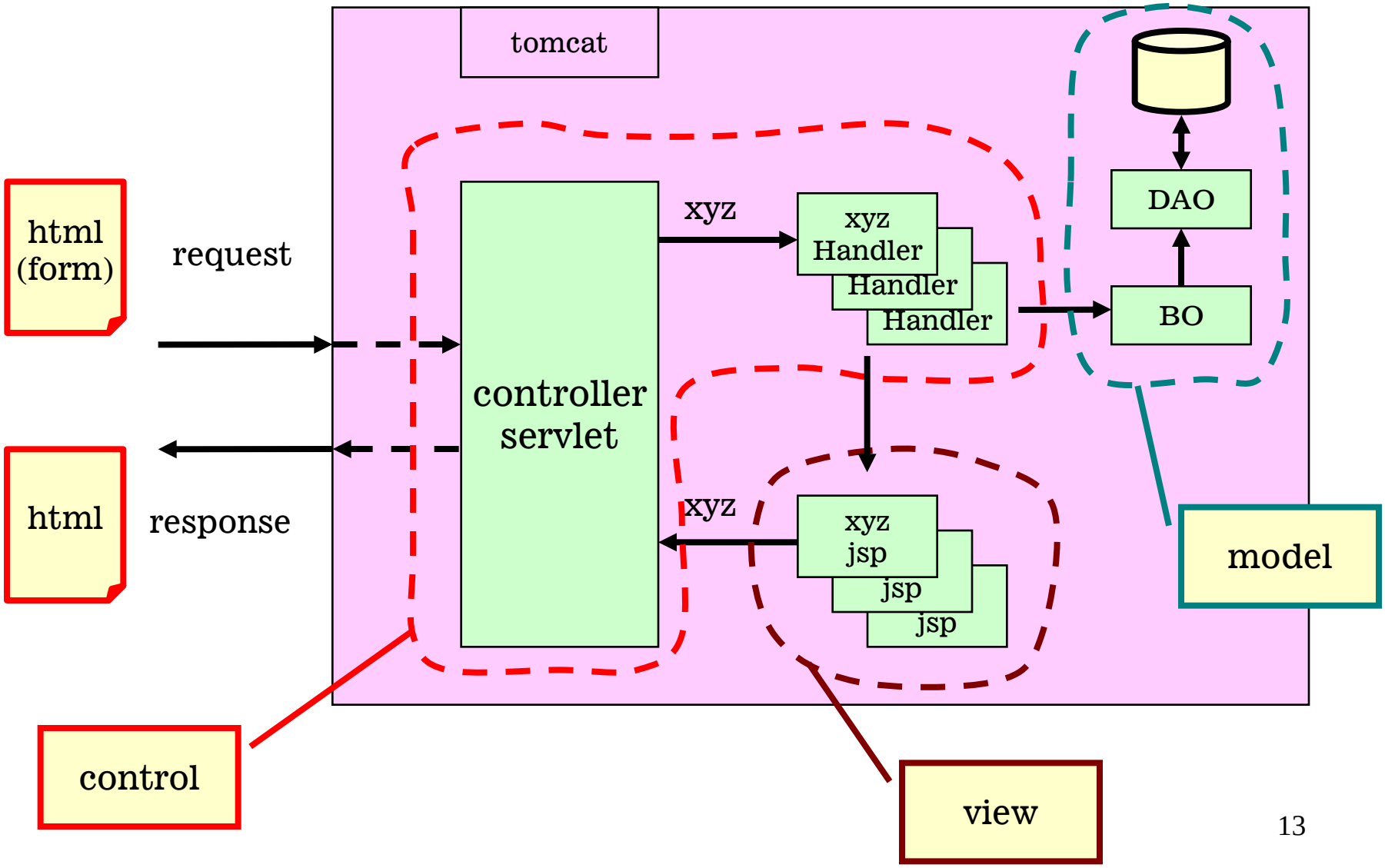
MVC architecture with Tomcat

www.sbVB.com.br



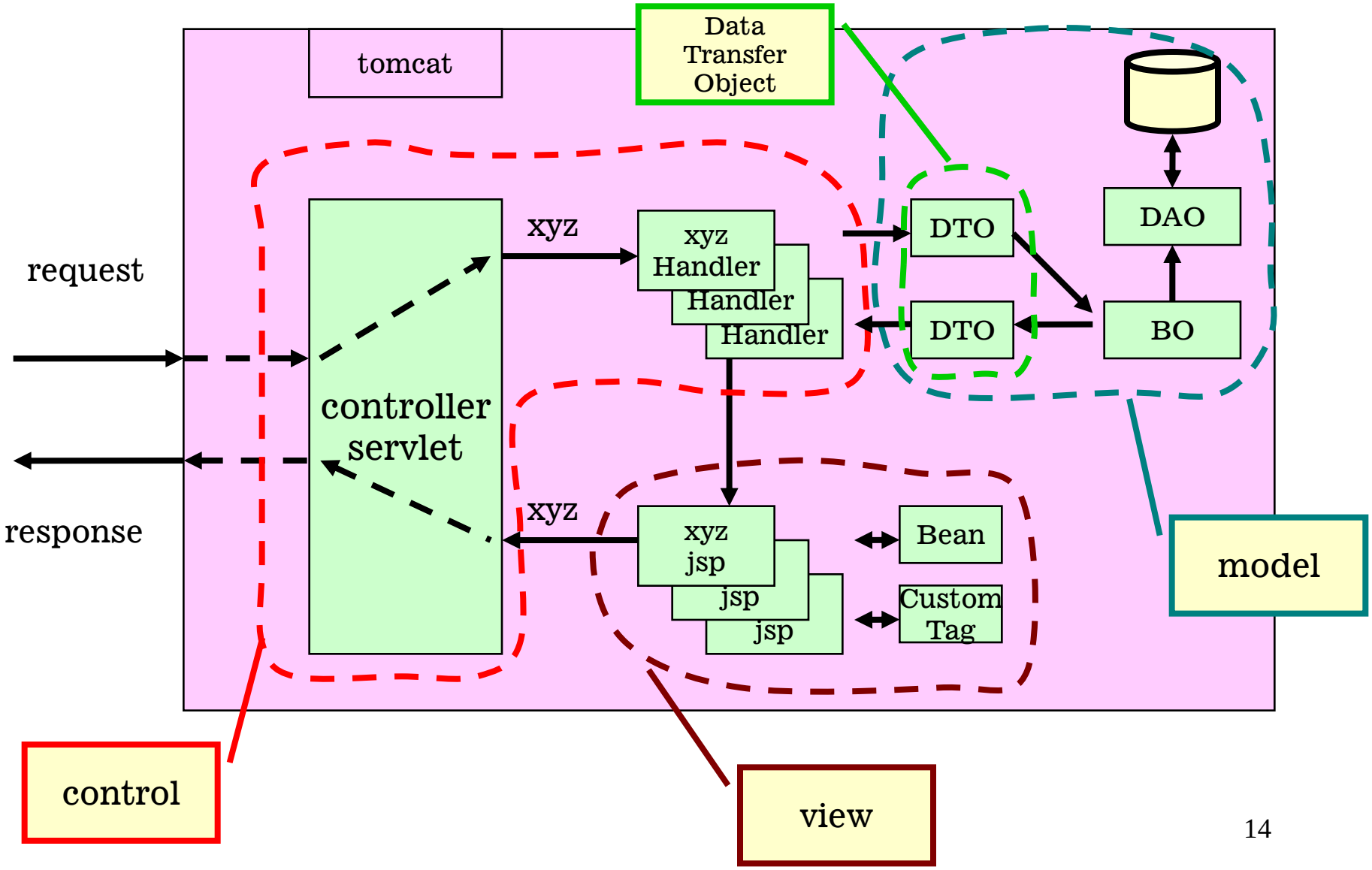
MVC (Model View Controller) architecture with Tomcat

www.sbVB.com.br

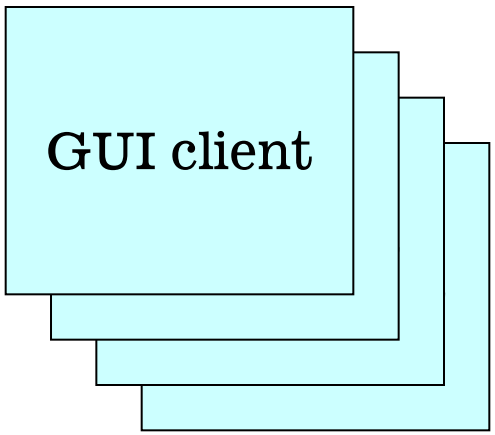


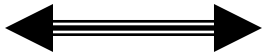
MVC (Model View Controller) architecture with Tomcat (2)

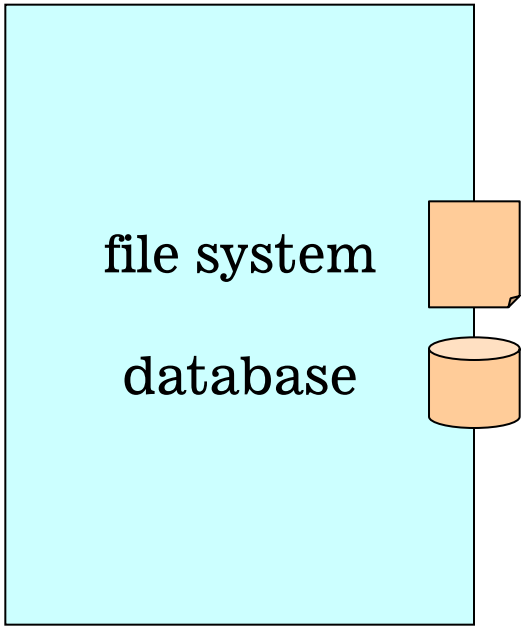
www.sbVB.com.br

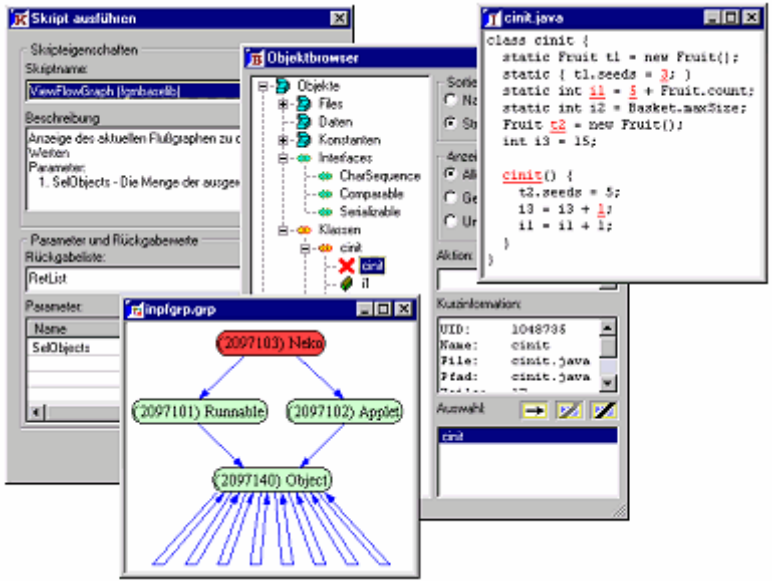


GUI usando datacenter

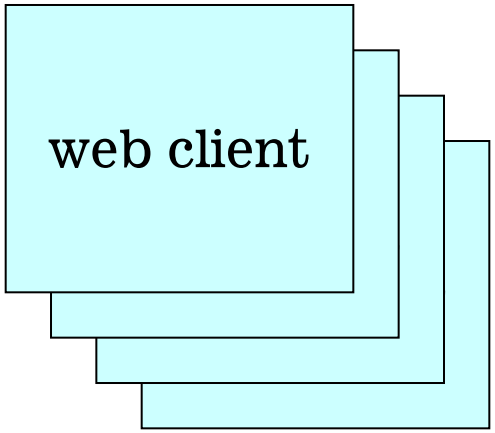


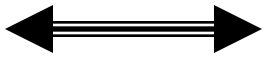


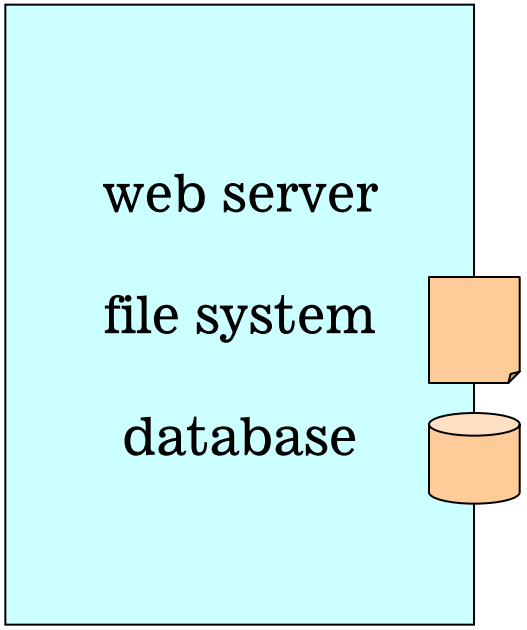


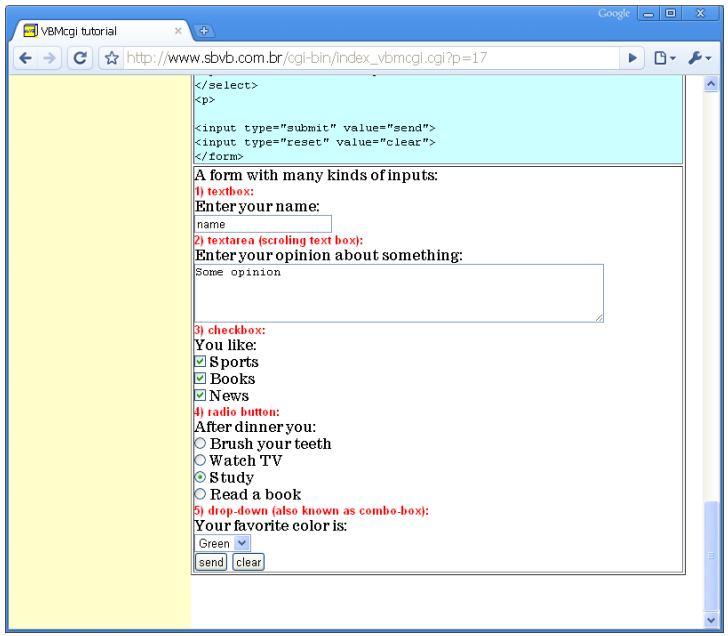


Arquitetura web clássica



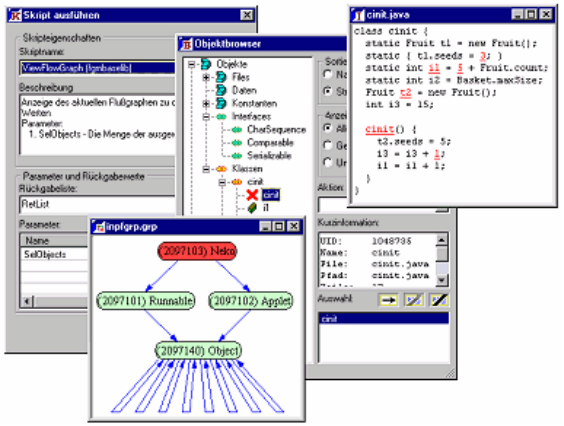


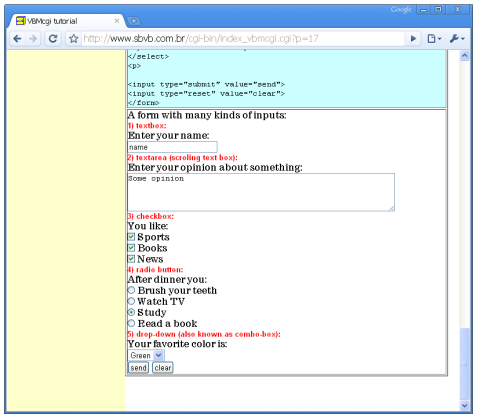




GUI × web

- GUI
  - vantagens
    - cliente mais sofisticado
  - desvantagens
    - mais difícil de desenvolver
    - mais difícil de manter os usuários atualizados (requer ferramenta adicional, desenvolvida a parte).
    - developer tem que resolver problema de concorrência de usuários
- WEB
  - vantagens
    - mais fácil de se desenvolver (desenvolve-se por cima do web server e do web client)
    - uma vez desenvolvido, o mundo inteiro pode ter acesso facilmente
    - implicitamente está resolvido o problema do upgrade do sistema; basta fazer upload da nova versão para o server
  - desvantagens
    - cliente menos sofisticado (imagine o PhotoShop na web . . .)

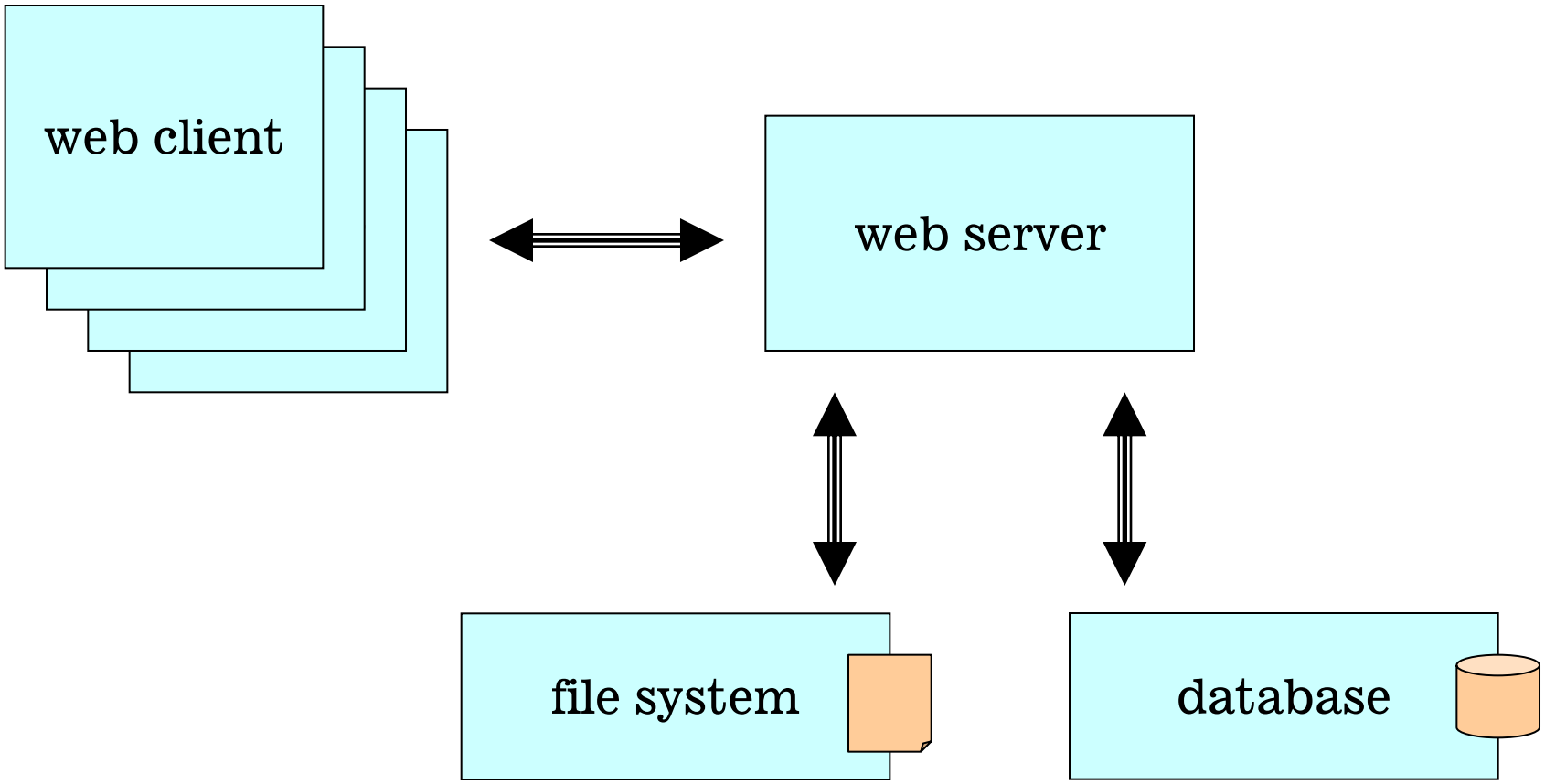




web client side

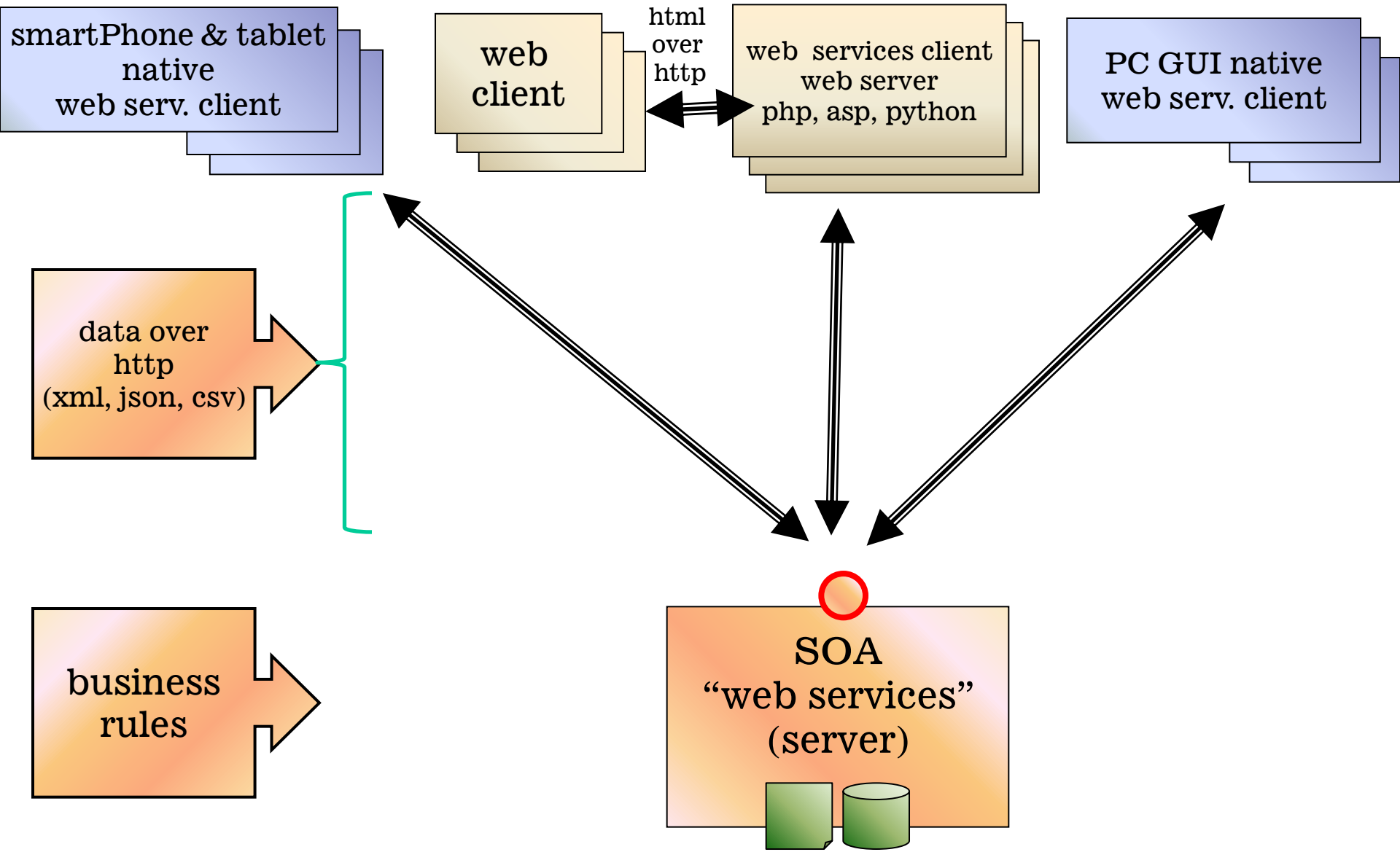
- Ultimamente tem ocorrido um grande aumento da sofisticação do software web client side.
- A “guerra dos browsers” continua
  - Firefox (derivado do Mozilla)
  - Google Chrome
  - Internet Explorer
  - Safari (do Mac)
  - Opera
  - Navegadores de celular
- O javascript torna-se mais sofisticado.
  - Bibliotecas como JQuery permitem bonitos e poderosos efeitos client side, compatível com todos os navegadores.

Separando o file system e o database



SOA-MC
SOA multiple client

www.sbVB.com.br



www.sbVB.com.br

SOA-MC
SOA multiple client

web client

html over http

web services client
web server
php, asp, python

web presentation

xml, csv, etc
over http

smartPhone & tablet
native
web serv. client

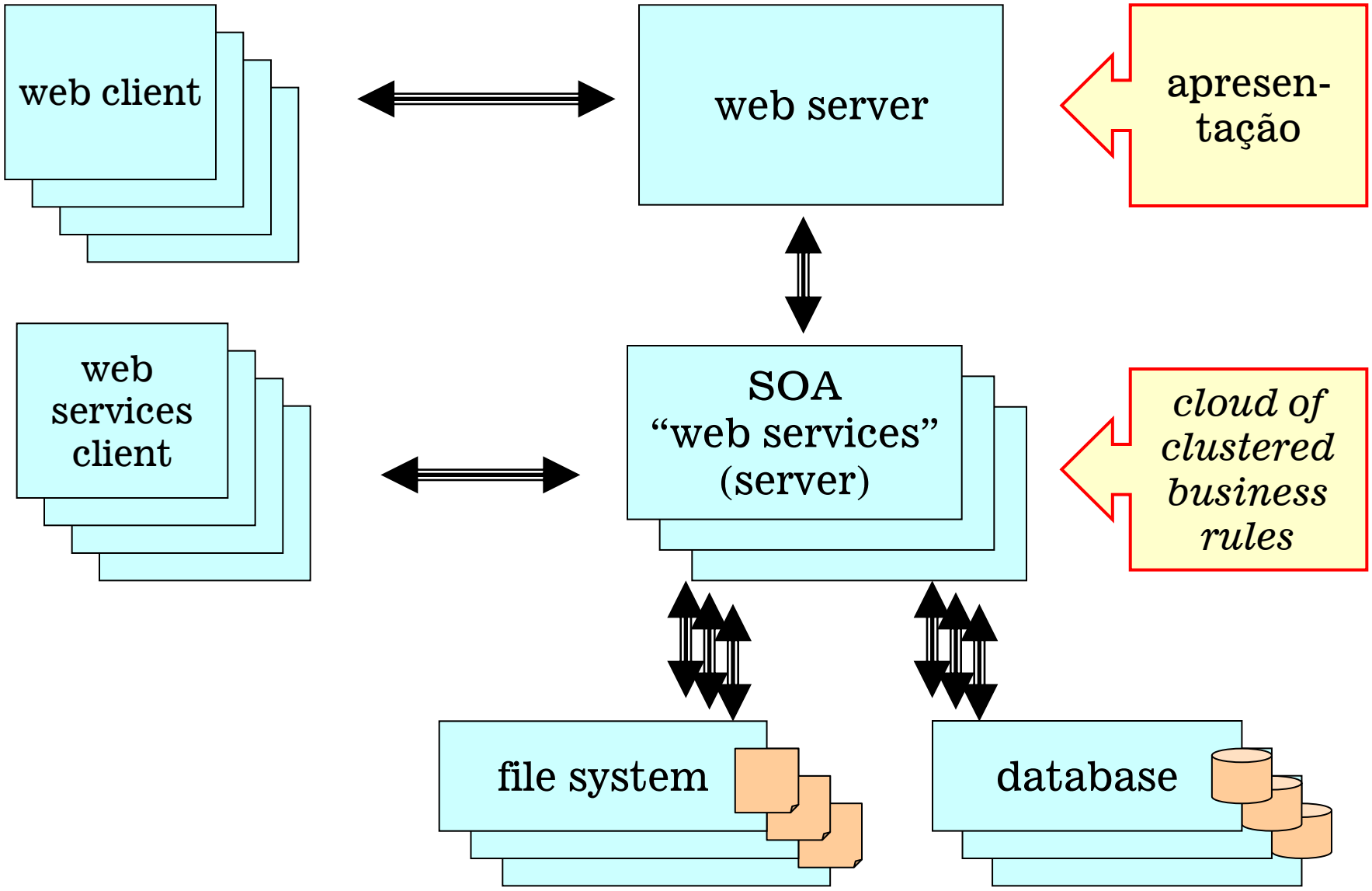
PC GUI native
web serv. client

native smartPhone
& tablet applications

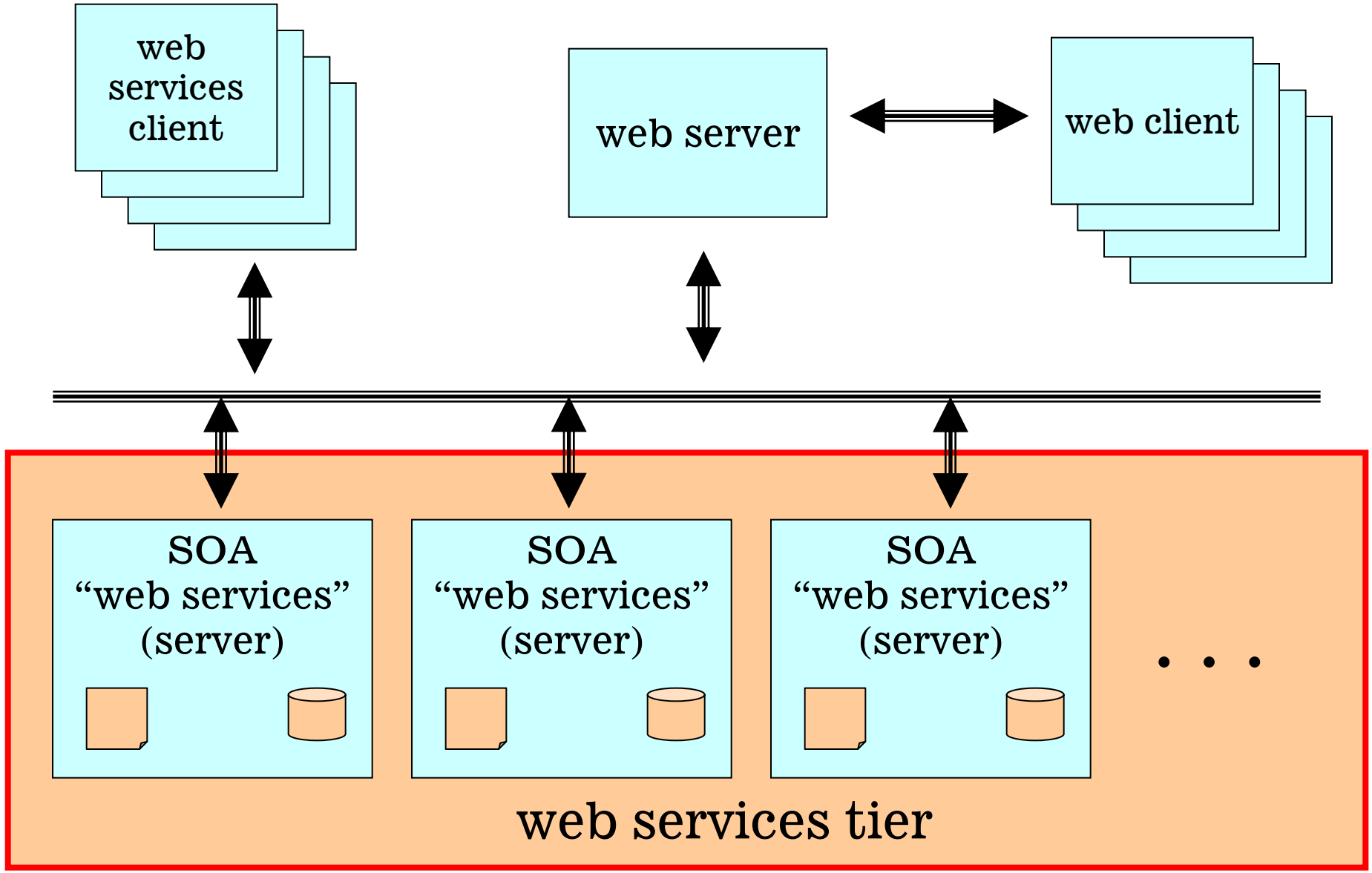
SOA
"web services"
(server)

business
rules

Clusterizando a camada separada de regras de negócio



Barramento de serviços (web services tier)



WS-BPEL

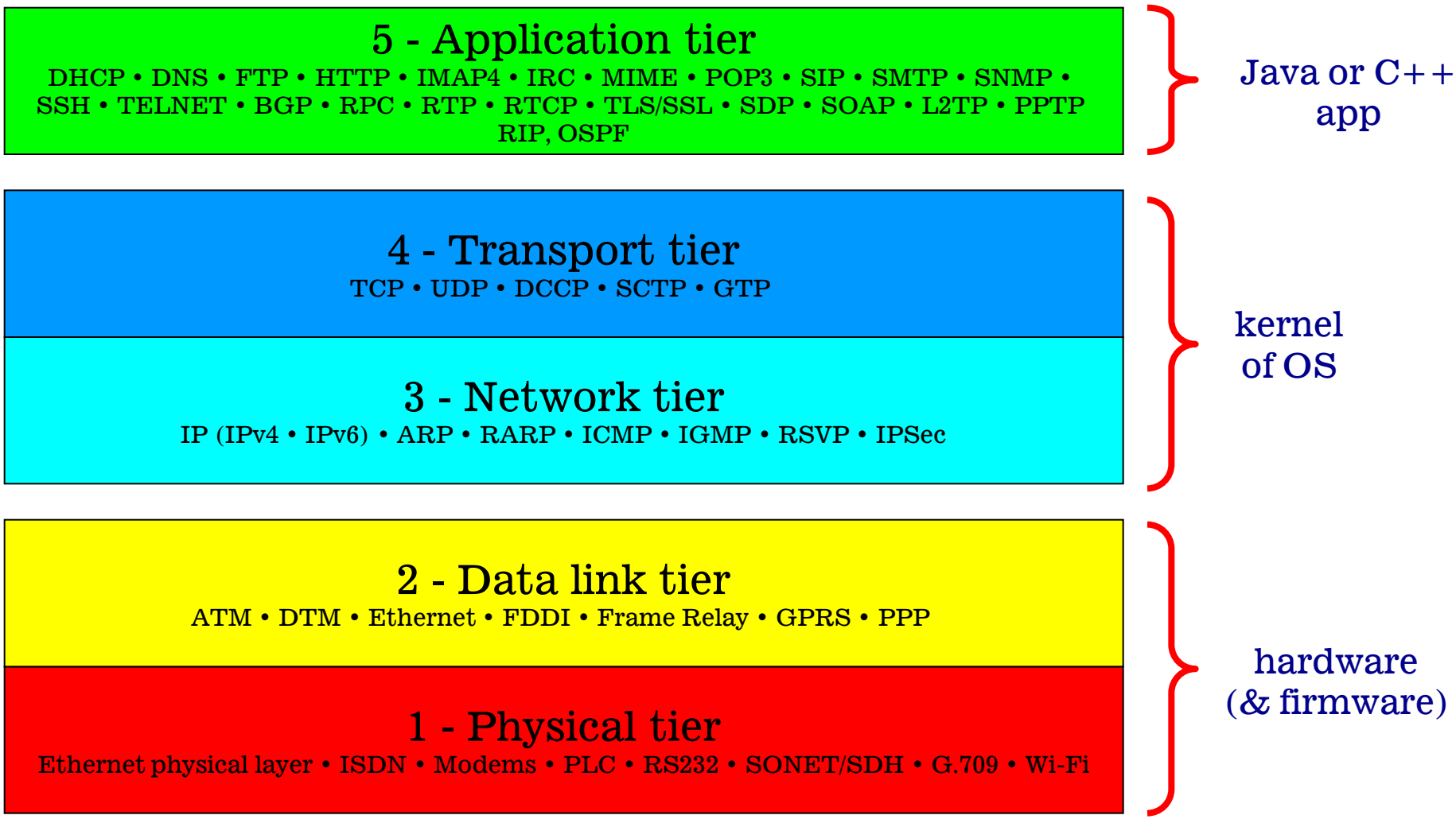
- BPEL = Business Process Execution Language é a versão abreviada de WS-BPEL (Web Services BPEL), proposto pela OASIS (http://www.oasis-open.org/)

- Interações web service (ambas podem ser modeladas por ws-bpel)
  - executável
  - abstrata

BPMN

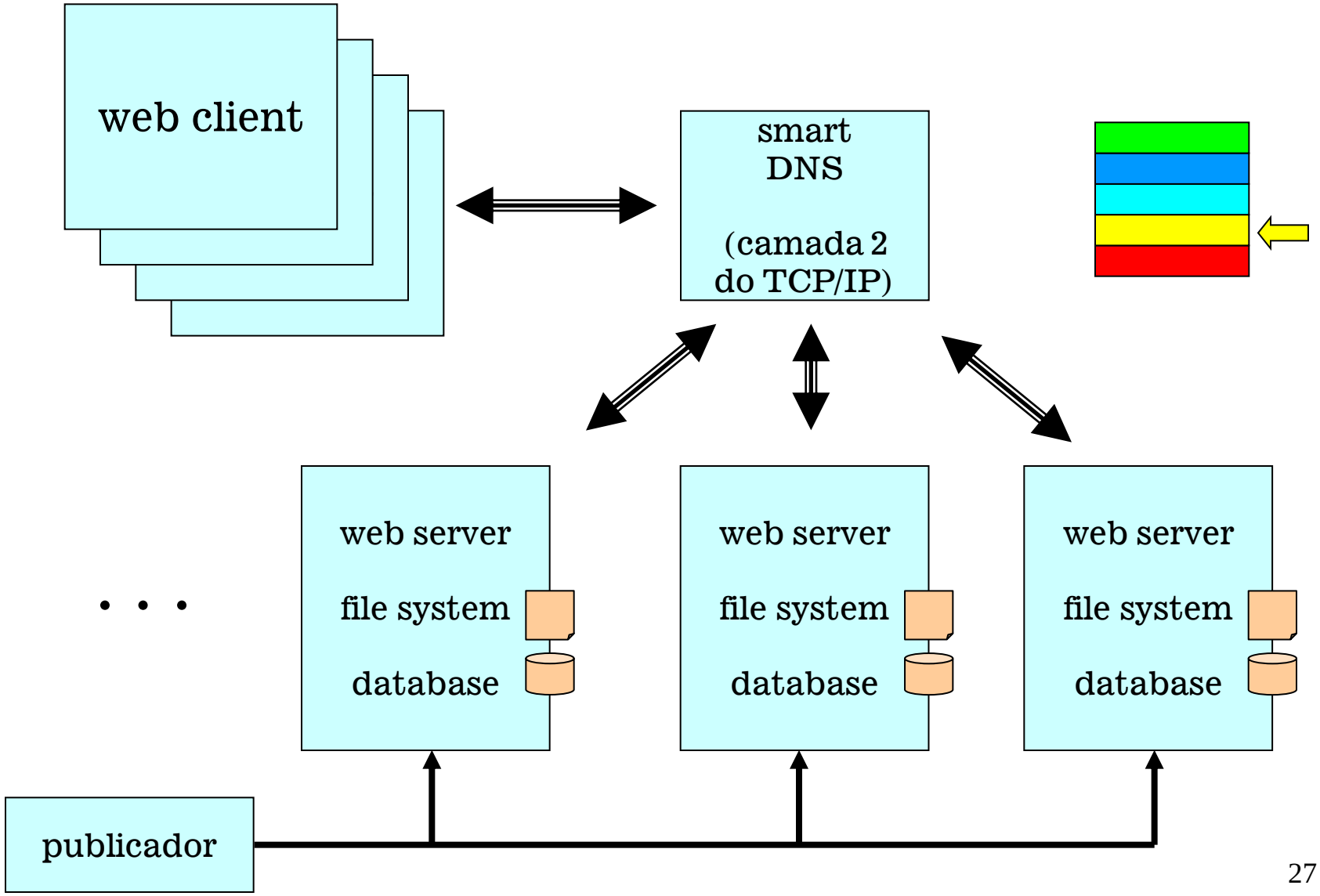
- BPMN = Business Process Modeling Notation
- jBPMN, do jboss
- descrição de workflow

The five layer TCP/IP model (actually used)

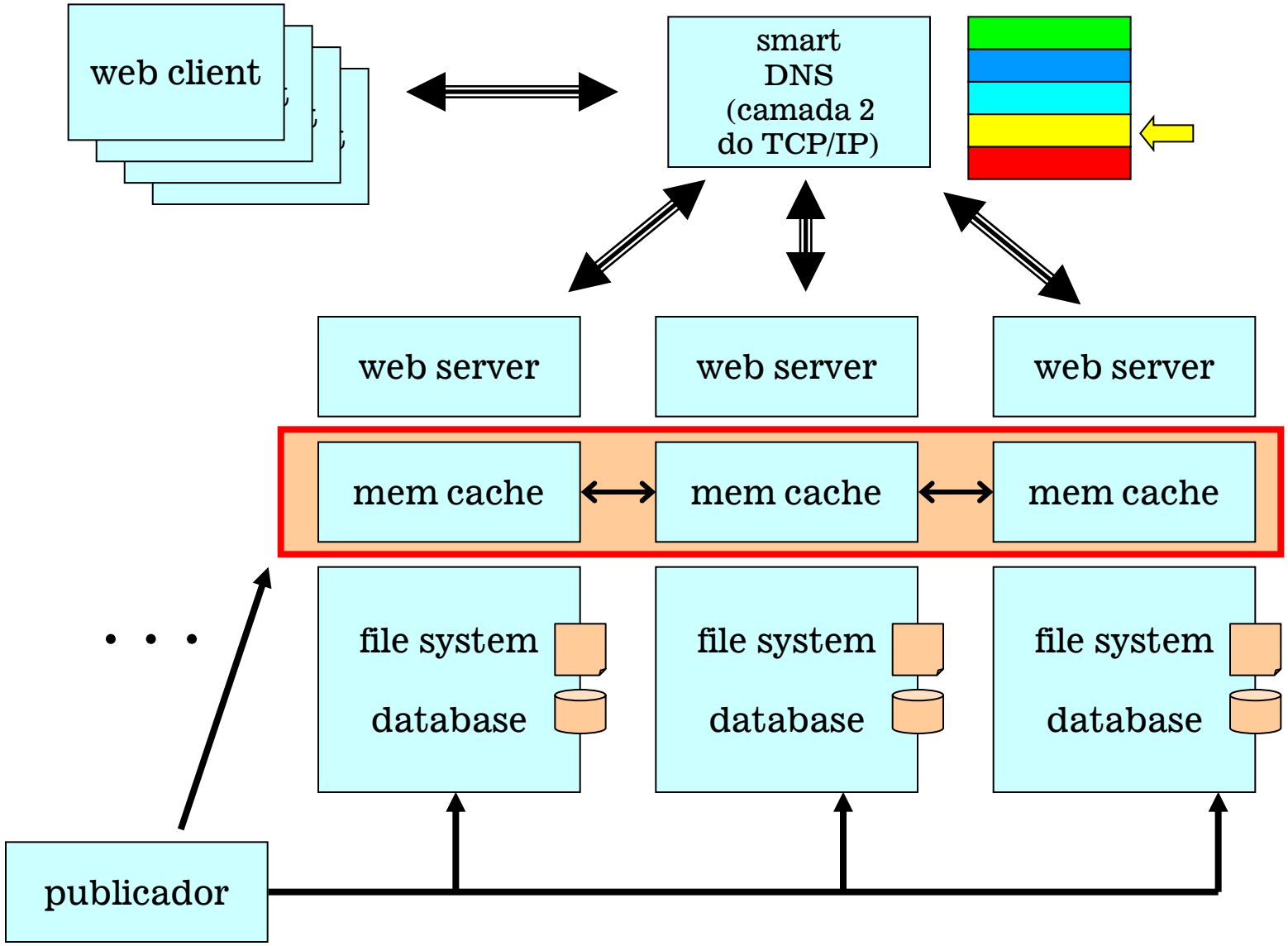


Arquitetura para alto tráfego e apenas leitura

www.sbVB.com.br



Arquitetura para alto tráfego e apenas leitura com mem cache



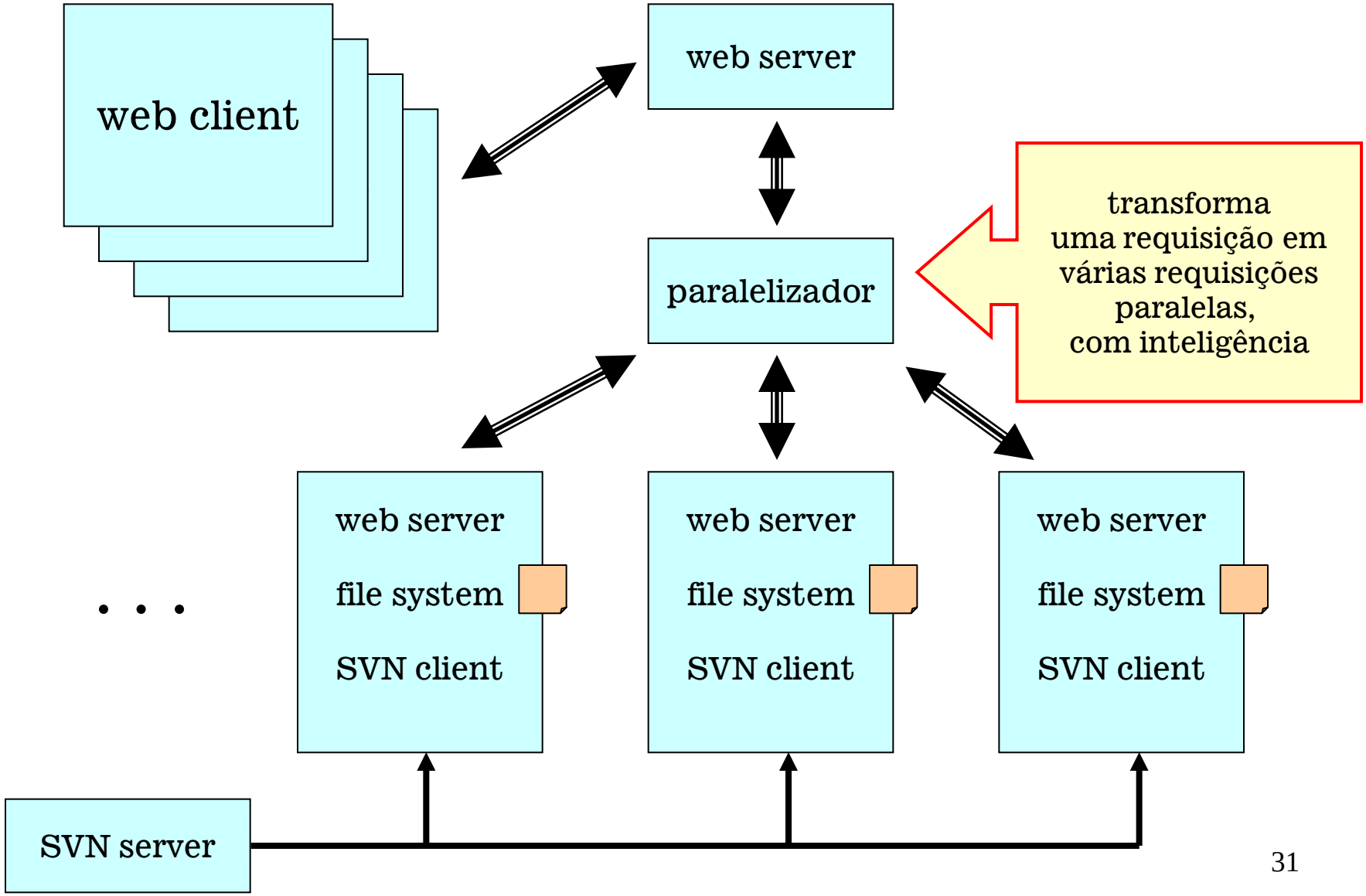
DNS que coloca o datacenter fisicamente perto do cliente

- É o que a google faz

Multiplicando a camada de regras de negócio

- onde implementar as regras de negócio?
  - em stored procedures no banco de dados
  - em código java, no servidor de aplicação (e.g. jboss).

Arquitetura para consulta paralela



SOA, SOAP e Web Services

SOA

- SOA = *Service Oriented Architecture*
- A idéia é “quebrar” um sistema grande em módulos independentes, que se comunicam com protocolo http.
- Pelo fato de ser baseado em http, tudo ocorre pela porta 80, que muitas vezes está liberada no firewall.
  - Existe opção de uso de https para protocolo seguro
- Cada módulo corresponde a um “serviço”, que pode ser definido por uma api.
- Quem usa o serviço precisa apenas entender e usar a api.
  - Os detalhes técnicos internos de como o serviço é implementado, incluindo a tecnologia empregada (e.g. linguagem de programação, sistema operacional) são irrelevantes para quem usa o serviço.

SOAP

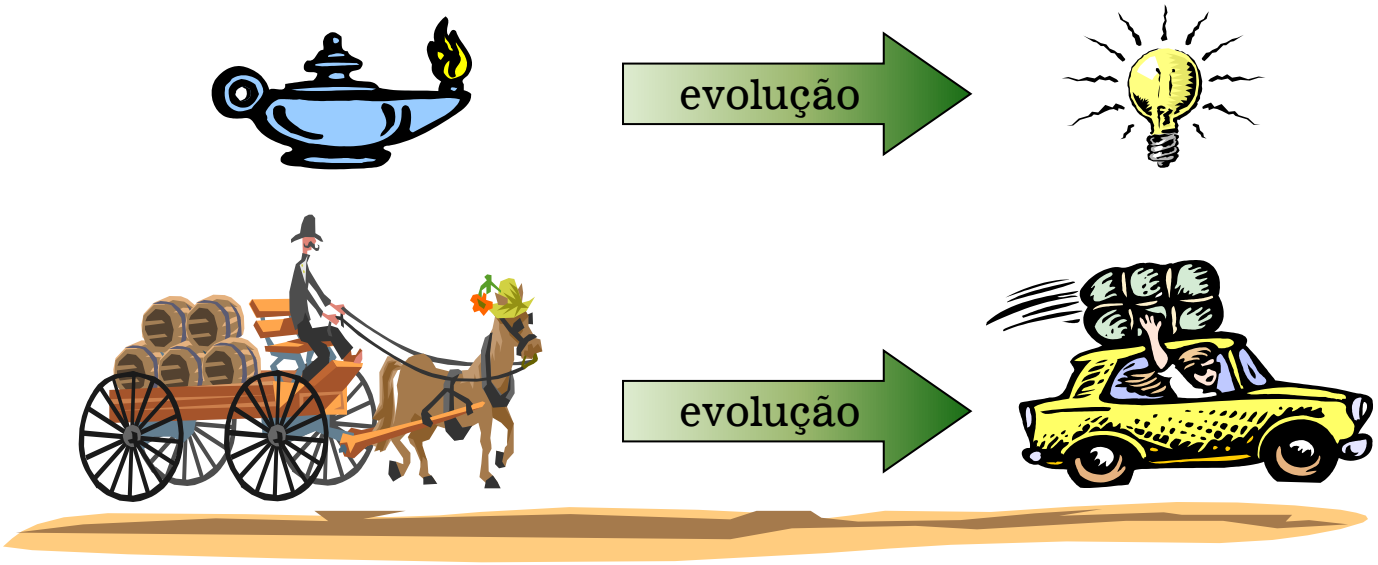
- Originalmente: *Simple Object Access Protocol*

SOA: XML or not XML ?

- Há opções na definição de web services. Uma das opções é usar ou não XML para transportar as mensagens.
- Se XML é usado, o SOA é SOAP.
- Vantagem de usar XML:
  - É uma forma estruturada de enviar e receber dados, compatível com unicode.
  - Pelo fato de o XML ser estruturado, no caso de manutenção do sistema pode-se acrescentar especificação (novas tags), sem que se perca compatibilidade com serviços já implantados.
- Desvantagem de usar XML:
  - O uso de XML requer parseamento, o que consome potência computacional, isto é, reduz o desempenho.
  - Sem a estruturação do XML, é preciso que se gerencie a manutenção do sistema (acréscimo de especificação)

Ambiente econômico e modelo de negócios

- Mudanças no ambiente podem levar um negócio a tornar-se inconsistente.
  - Exemplo: iluminação a gás tornou-se obsoleta com a iluminação elétrica. Outro: carroça/cavalo como meio de transporte tornou-se obsoleto com o automóvel.

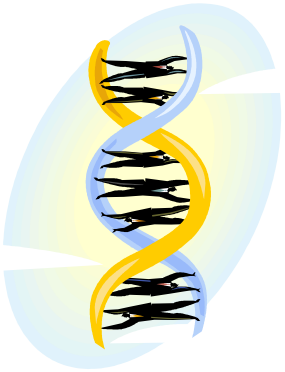
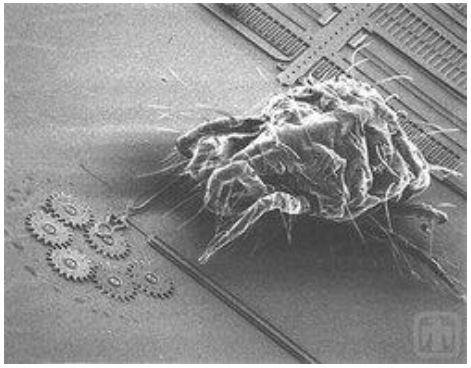


www.sbVB.com.br

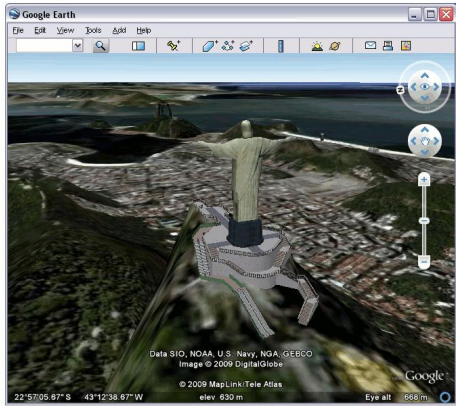
Algumas das inovações que provocam forte impacto no ambiente econômico

- Robótica, TI e segurança da informação, genética, nanotecnologia



- Internet / Web / Google Earth, tablets, smartPhones, novos materiais.





Internet e web ainda são importantes inovações

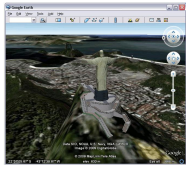
www.sbVB.com.br

- Internet ⊃ web.
- Acesso a web por smartPhone é agora grande público, que aprecia navegar com site desenhado para smartPhone.

Internet

email, ftp, google earth, VOIP, skype, IPTV, telnet, ssh, kazaa, p2p, Instant Messager, video stream, etc.

TortoiseSVN



web

(http://www...)
Navegadores web: Internet Explorer, Firefox, chrome, Safari, navegadores de smartPhone.
Páginas web: Google, submarino, amazon, ebay, etc.

amazon.com

ebay

Submarino

Google

www.sbVB.com.br

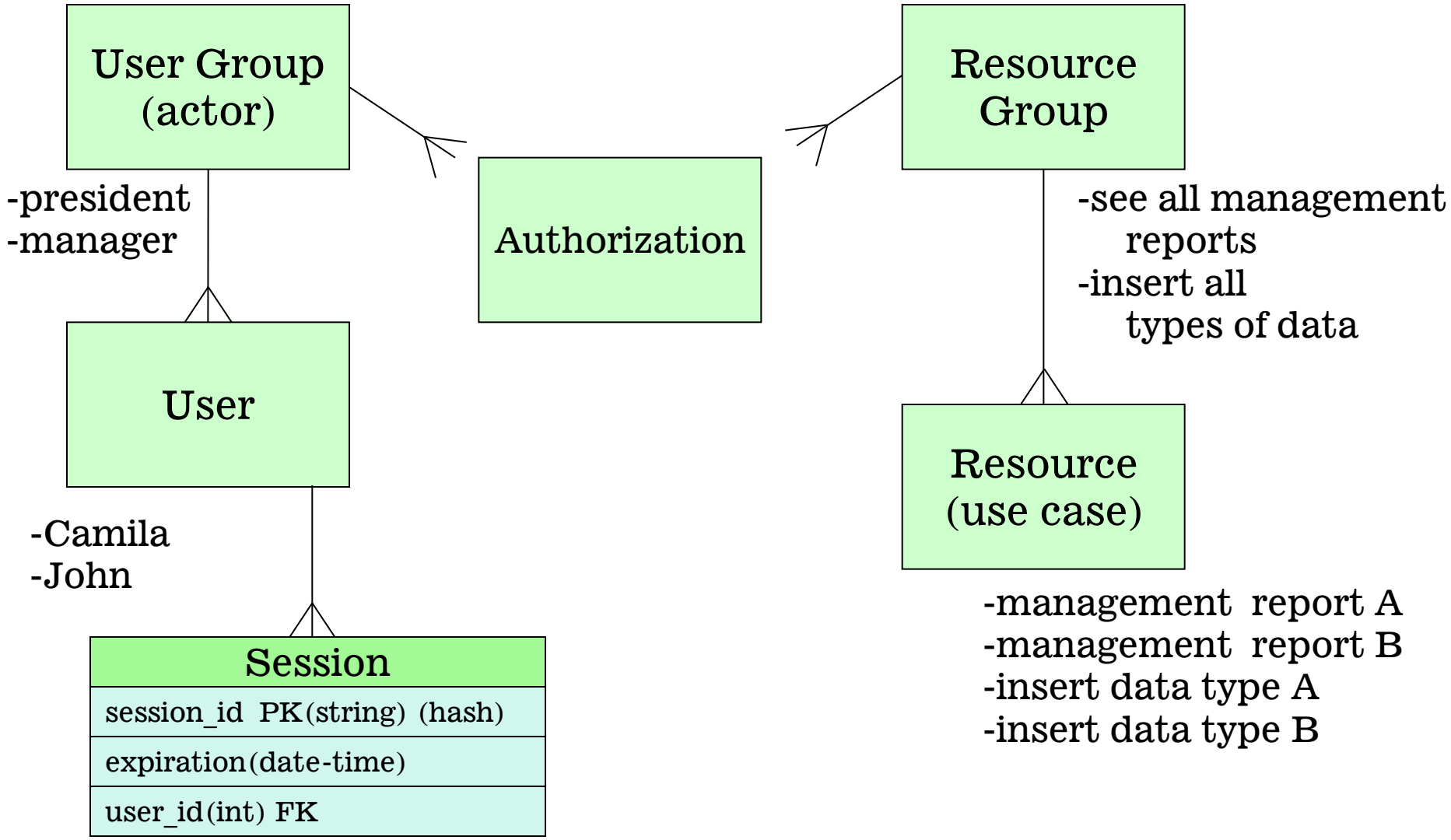
mobile (smartPhones & tablets) são inovações destacadas

- Tablets são importante inovação, considerados como um caso particular de smartPhone.
- Páginas web podem ser otimizadas para visualização via smartPhone / tablet.
- Aplicações nativas para smartPhone produzem melhor experiência para o usuário que acesso a web.

DOUA

Database Orientad Usecase Autorization

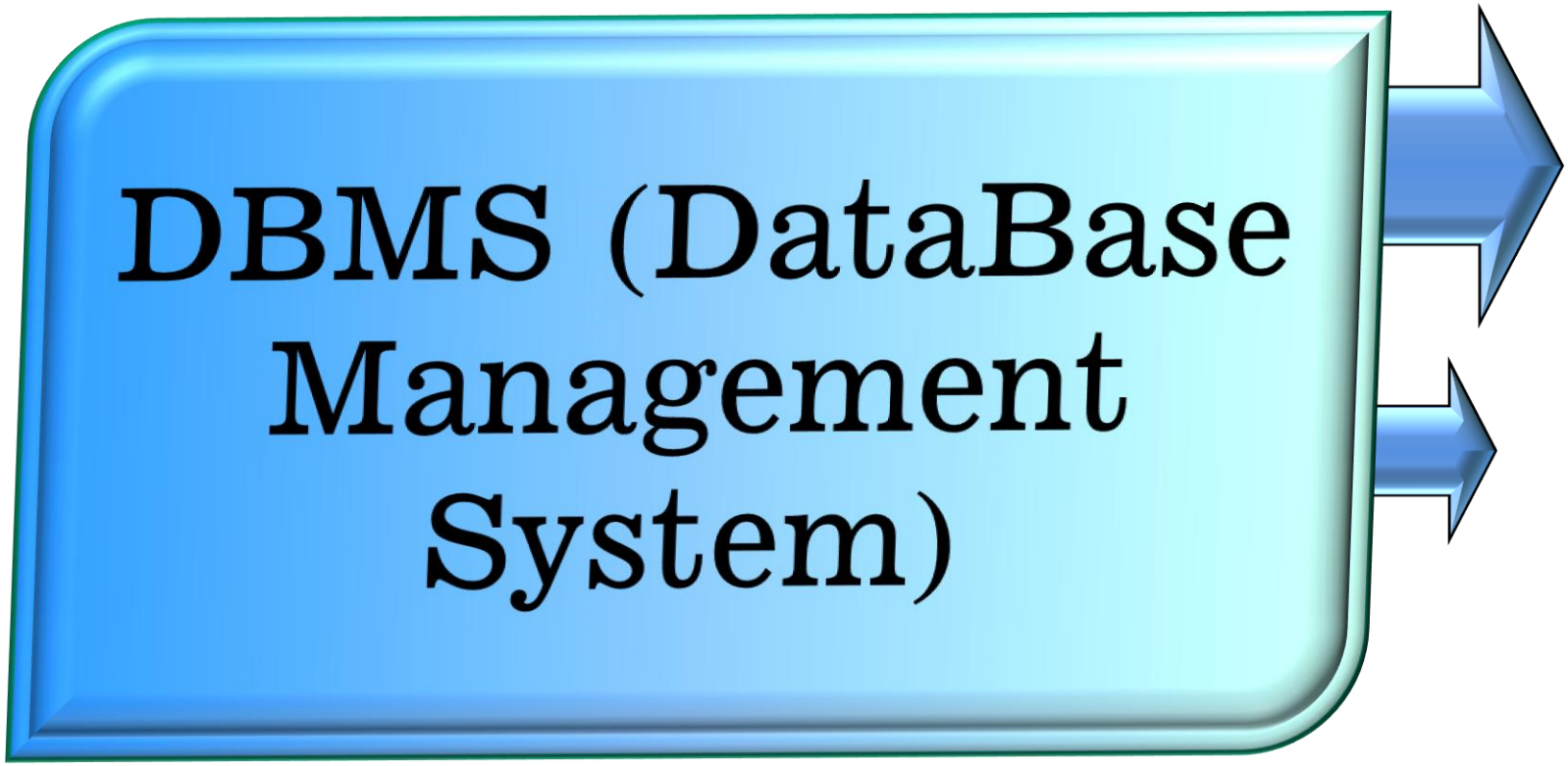
www.sbVB.com.br



SOA-MC and DOUA

- SOA_MC
  - http://www.sbvb.com.br/proposals/SOA-MC-sbvb.com.br.pdf
- DOUA
  - http://www.sbvb.com.br/proposals/DOUA-sbvb.com.br.pdf

DataBase



Database models

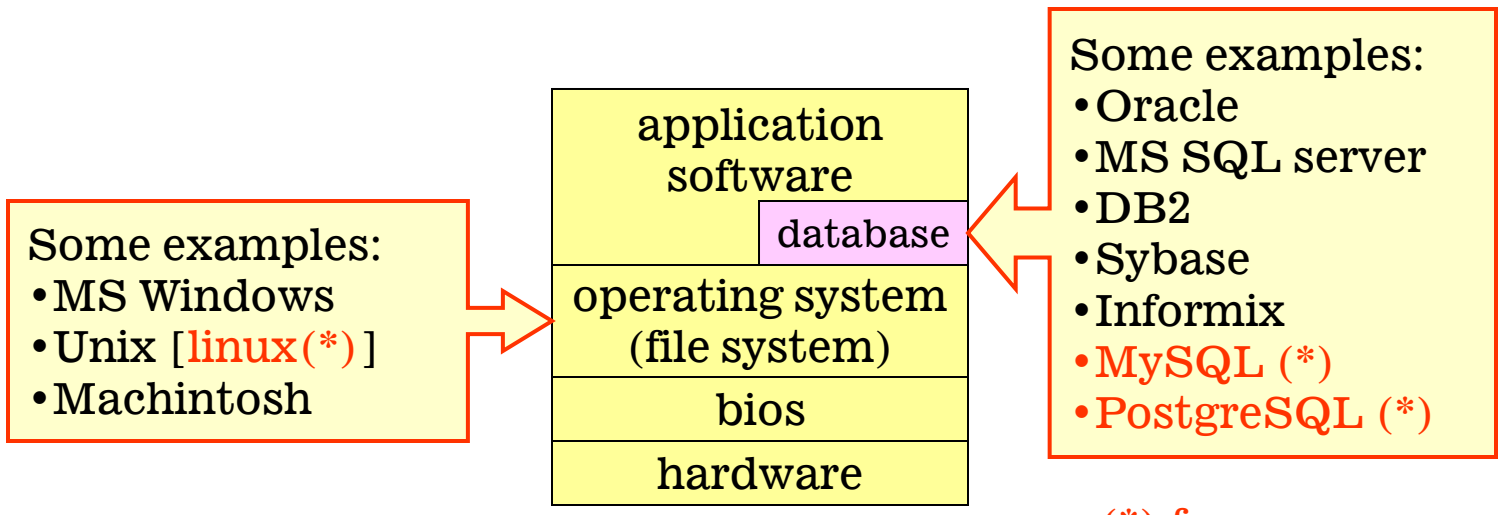
- There exist 5 major models of databases
  - Hierarchical
  - Network
  - Relational (uses PK, FK, SQL)
  - Object-relational
  - Object Oriented

Some examples
- Oracle
- MS SQL server
- DB2
- Sybase
- Informix
- MySQL (*)
- PostgreSQL (*)

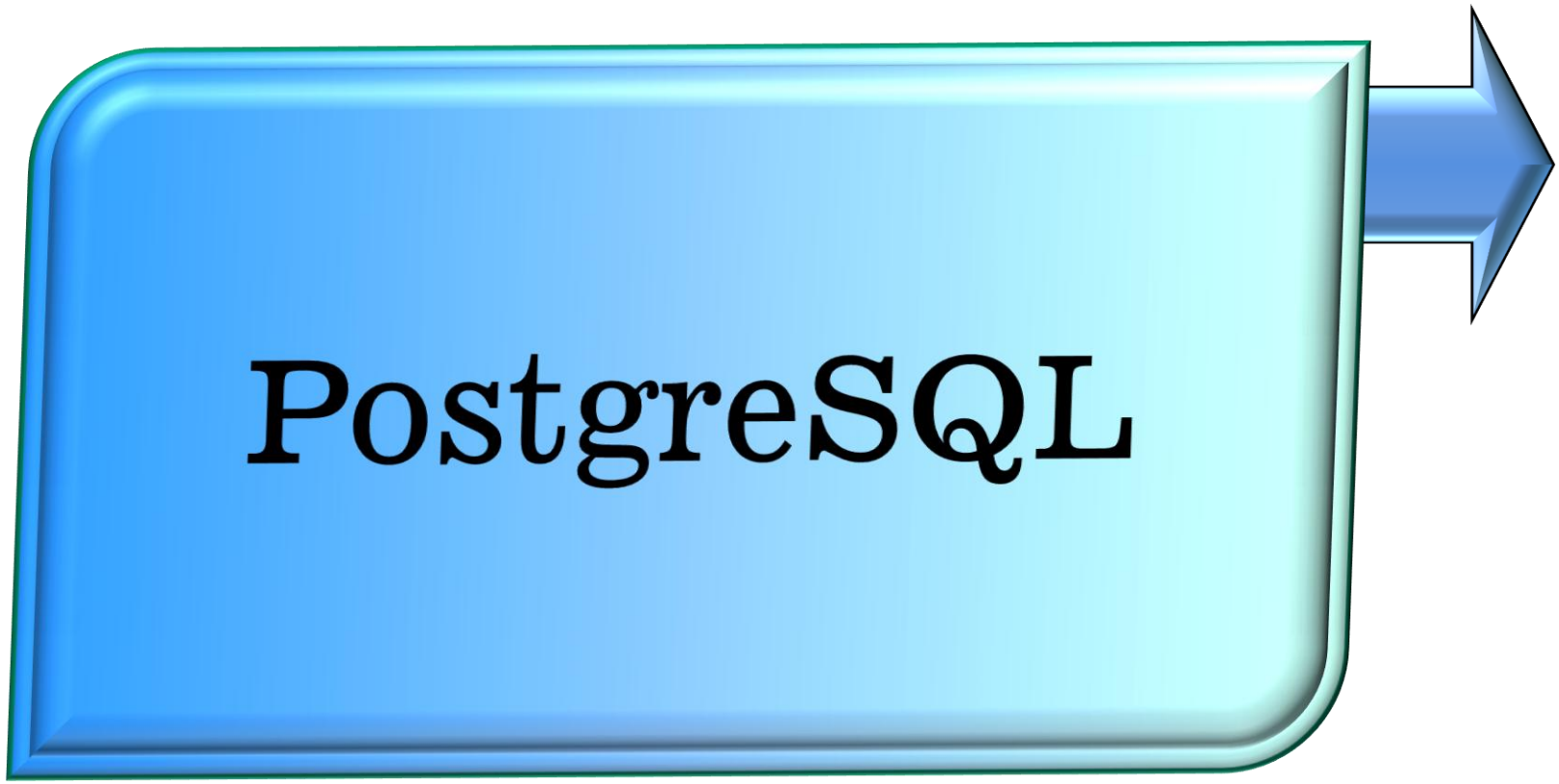
(*) free

What is a database?

- DBMS (DataBase Management System) is a category of software that works as a tier in a software system.
  - The DBMS lies above the operating system (and the file system), and below the user application.
- The Structured Query Language (SQL), is a database query language that was adopted as an industry standard in 1986.
  - SQL is *not* a programming language, in the sense that you can't write a computer application in SQL; it is used to state commands to the DBMS.
- The application software is written in a programming language (such as C++ or Java).



PostgreSQL



http://www.postgresql.org/

Brief history of PostgreSQL (early moments)

- Ingres project at UC Berkeley. The project leader, Michael Stonebraker, had left Berkeley to commercialize Ingres in 1982, but eventually returned to academia.
  - After returning to Berkeley in 1985, Stonebraker started a post-Ingres project to address the problems with contemporary database systems that had become increasingly clear during the early 1980s. While they share many of the same ideas, the code bases of PostgreSQL and Ingres started (and remain) completely separated.
- 1986 – the team released a number of papers describing the basis of the system.
- 1988 – the project had a prototype version up and running, called “POSTGRES”.
- 1990 – version 2.
- 1991 – version 3.
- 1993 – version 4.2 (last version as a research project of Berkeley).
- 1995 – Andrew Yu and Jolly Chen release version 5 “Postgre95”, open-source, 100% ANSI C.

Brief history of PostgreSQL (late moments)

- 1996 – Name change from Postgres95 to PostgreSQL, also known as version 6.
- 2000 – version 7 (Foreign keys, SQL92 syntax for joins, etc.)
- 2005 – version 8.0. Native server on Microsoft Windows, savepoints, tablespaces, exception handling in functions, point-in-time recovery.
- 2010 – version 9.0. Built-in binary streaming replication, Hot standby, 64-bit Windows, per-column triggers and conditional trigger execution, exclusion constraints, anonymous code blocks, named parameters, password rules
- 2011 – version 9.1. Synchronous replication, per-column collations, unlogged tables, K-nearest-neighbor indexing, serializable snapshot isolation, writeable common table expressions, SE-Linux integration, extensions, SQL/MED attached tables (Foreign Data Wrappers), triggers on views

Basic operations (using console)

- creation of user “root”

```
[root]$ su postgres
[postgres]$ createuser root
Shall the new user be allowed to create databases? (y/n) y
Shall the new user be allowed to create more new users? (y/n) y
CREATE USER
```

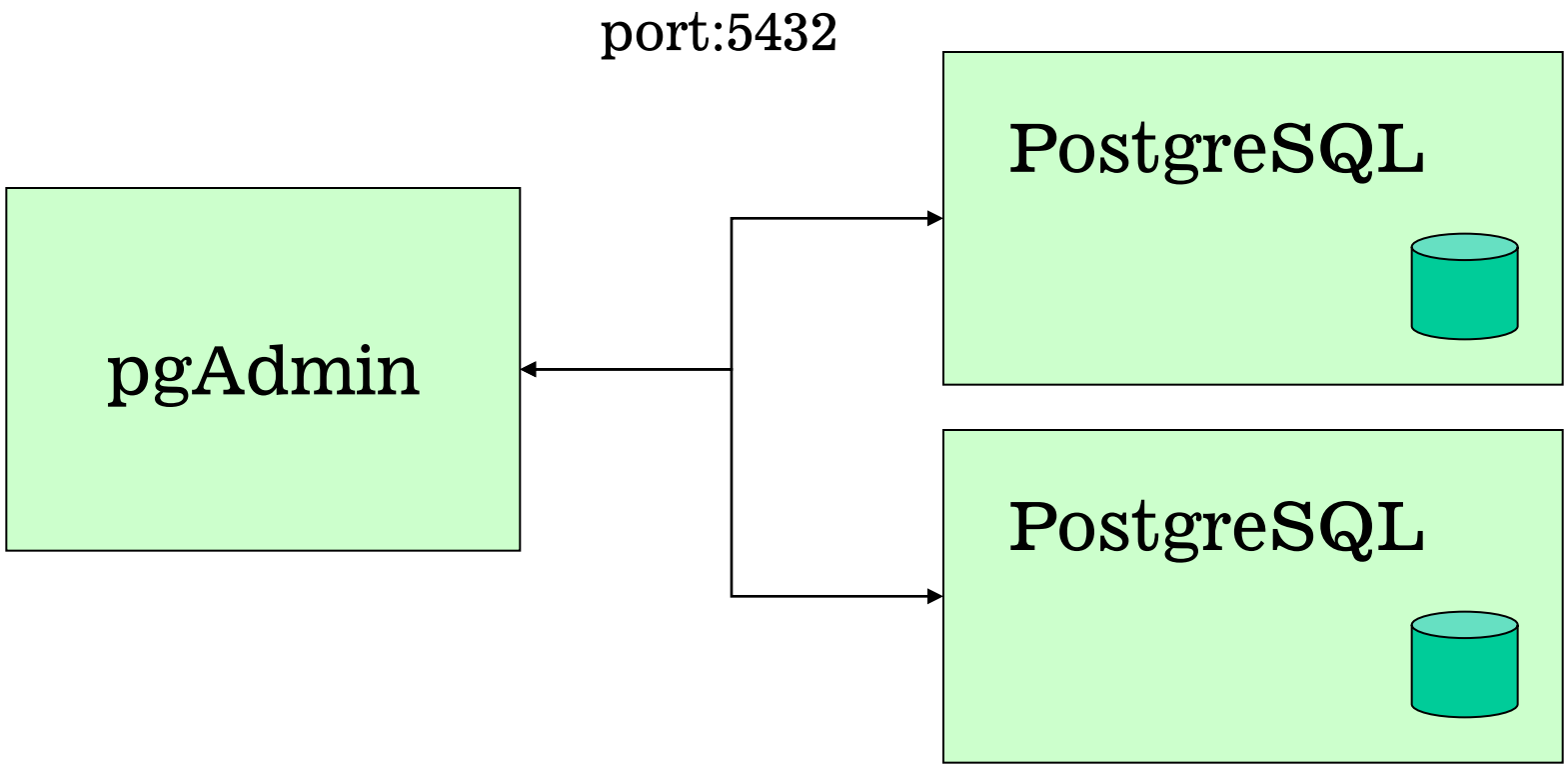
- creation of user “sbvb” (myuser)

```
[root]$
[root]$ createuser –P sbvb
Enter password for new user:
Enter it again:
Shall the new user be allowed to create databases? (y/n) n
Shall the new user be allowed to create more new users? (y/n) n
CREATE USER
```

Basic operations (using console) (2)

- Creation / deletion of a database
  - $ createdb db_sbVB owner=sbvb
  - $ dropdb db_sbVB
- Entering the database console, from user sbvb
  - $ psql db_sbVB
- Enter as user sbvb to database db_bib
  - psql -U sbvb db_bib
- See more details in the chapter “Getting Started” of manual

Administering databases



Since PostgreSQL version 8.x supports unicode!

- Not only data, but also table names or column names can be unicode.
- User application destined to any place in the world can be developed with PostgreSQL, even if it is required to use non-western letters such as Japanese or Chinese.

- 日本語 (Japanese) ok
- 中文 (Chinese) ok
- (any language of the world) ok

SQL

Tables for data

- The data lies in tables

TableName
Col_1 (type)
Col_2 (type)
Col_3 (type)
Col_4 (type)

design view

TableName			
Col_1	Col_2	Col_3	Col_4
data	data	data	data
data	data	data	data
data	data	data	data
data	data	data	data
data	data	data	data

data view

Tables (2)

- Table example

tb_store
Name (string)
Sales (real)
LastReport (date)

design view

tb_store		
Name	Sales	LastReport
Rio de Janeiro	1200.0	26/08/2003
São Paulo	2000.0	20/08/2003
Paraná	1000.0	1/08/2003
Brasília	800.0	1/07/2003

data view

select

- Assuming that is data already stored in the database tables, one can get information about it by using the SQL command “select”.
- The syntax is

```
select column[,column] from table
```

- Your application can send to the database a select command, and it will produce an anonymous table in the database with the answer. Then, you can scan this anonymous table and use it for your purposes. Example:

```
select Name from tb_store
```

(anonymous)
Name
Rio de Janeiro
São Paulo
Paraná
Brasília

```
select Name,Sales from tb_store
```

(anonymous)	
Name	Sales
Rio de Janeiro	1200.0
São Paulo	2000.0
Paraná	1000.0
Brasília	800.0

select syntax

```
select "column1"[,"column2",etc]
  from "tb_tablename"
  [where "condition"];

  [] = optional
```

- The column names that follow the select keyword determine which columns will be returned in the results. You can select as many column names that you'd like, or you can use a "*" to select all columns.
- The table name that follows the keyword from specifies the table that will be queried to retrieve the desired results.
- The where clause (optional) specifies which data values or rows will be returned or displayed, based on the criteria described after the keyword “where”.

insert

- todo

Learning more about SQL

- Links
  - http://www.1keydata.com/
  - http://www.sqlcourse.com/
  - http://www.sqlcourse2.com/
  - http://www.baycongroup.com/tocsql.htm
- PostgreSQL's help

Practice basic SQL functions

- See examples of chapter 2 of manual.
- See that you get familiar with the concepts below
  - Creation / deletion of tables
  - Populate tables with data (insert)
  - Query tables (select)
  - Join among tables (complex queries)
  - Aggregated functions (max, min, like, etc.)
  - Managing data
    - Update / deletion of table's data

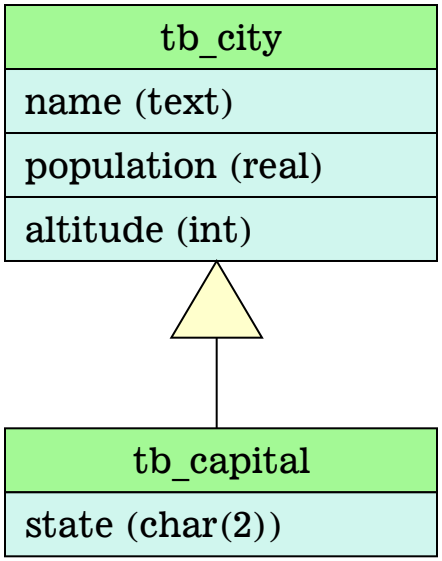
Advanced SQL functions of PostgreSQL

- Views (as alias of a select)
- Foreign keys
- Array

inheritance

inheritance of tables

```
CREATE TABLE tb_city (
name text,
population real,
altitude int -- (in mt)
);

CREATE TABLE tb_capital (
state char(2)
) INHERITS (tb_city);

INSERT INTO tb_city (name, population, altitude)
  VALUES ('Petrópolis', 2000, 1500);
INSERT INTO tb_city (name, population, altitude)
  VALUES ('Teresópolis', 2500, 1800);

INSERT INTO tb_capital (name, population, altitude, state)
  VALUES ('Rio de Janeiro', 8000, 2, 'RJ');
INSERT INTO tb_capital (name, population, altitude, state)
  VALUES ('São Paulo', 16000, 1650, 'SP');
```



Both tables

	name text	population real	altitude integer	state character(2)
1	Rio de Janeiro	8000	2	RJ
2	São Paulo	16000	1650	SP

tb_capital

	name text	population real	altitude integer
1	Petrópolis	2000	1500
2	Teresópolis	2500	1800
3	Rio de Janeiro	8000	2
4	São Paulo	16000	1650

tb_city

MySQL

- mysql –u root –p <rootpwd>
- mysql>CREATE DATABASE db_hello
  - DEFAULT CHARACTER SET utf8
  - DEFAULT COLLATE utf8_general_ci;
- CREATE USER 'sbvb'@'localhost‘
  - IDENTIFIED BY 'sbvbpwd';
- GRANT ALL PRIVILEGES ON *.* TO 'sbvb'@'localhost' WITH GRANT OPTION;

MySQL

- mysql –u sbvb –p <sbvbpwd>
- connect db_hello;

- CREATE TABLE tb_varvalue(
- var VARCHAR(30) NOT NULL,
- PRIMARY KEY(var),
- value VARCHAR(300)
-);

- INSERT INTO tb_varvalue (var, value) VALUES('myvar', 'myvalue');

- select * from tb_varvalue;

Installing tools

Installing mysql to ubuntu

- sudo apt-get install mysql-server
  - Login|pwd:root|rootpwd
- sudo apt-get install mysql-client-5.5

- GRANT ALL PRIVILEGES ON *.* TO 'root'@'localhost';
- FLUSH PRIVILEGES;

- // check mysql status
- mysqladmin -u root -p status

Reset mysql pwd

- // Stop the MySQL Server.
- sudo /etc/init.d/mysql stop

- // Start the mysqld configuration.
- sudo mysqld --skip-grant-tables --skip-networking &

- // Login to MySQL as root.
- mysql -u root mysql

- // Replace YOURNEWPASSWORD with your new password!
- UPDATE user SET Password=PASSWORD('YOURNEWPASSWORD') WHERE User='root'; FLUSH PRIVILEGES; exit;

- // test login to mysql with new pwd
- mysql -u root -p
- <newpwd>

Tools used to develop web services

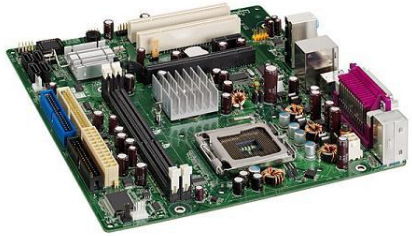
- Operating system
  - Linux Ubuntu 12.04 x86_64 kernel 3.2.0-34-generic
  - could be Windows or Mac
- Java kit ($ java -version)
  - java version "1.7.0"
  - Java(TM) SE Runtime Environment (build 1.7.0-b147)
  - Java HotSpot(TM) 64-Bit Server VM (build 21.0-b17, mixed mode)

Tools used to develop web services

- IDE
  - NetBeans 7.3, with axis2 plugin
- Libraries
  - Axis2 library version 1.6.2
- Database
  - PostgreSQL
  - mysql --version
    - Ver 14.14 Distrib 5.5.28

Software tiers

www.sbVB.com.br



DBMS (MySql)

java

myservice

axis2

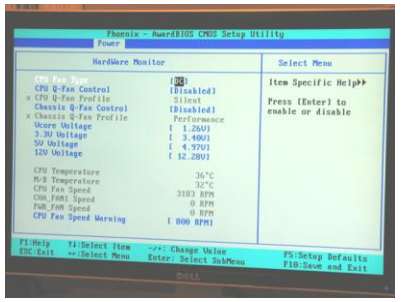
tomcat

JVM

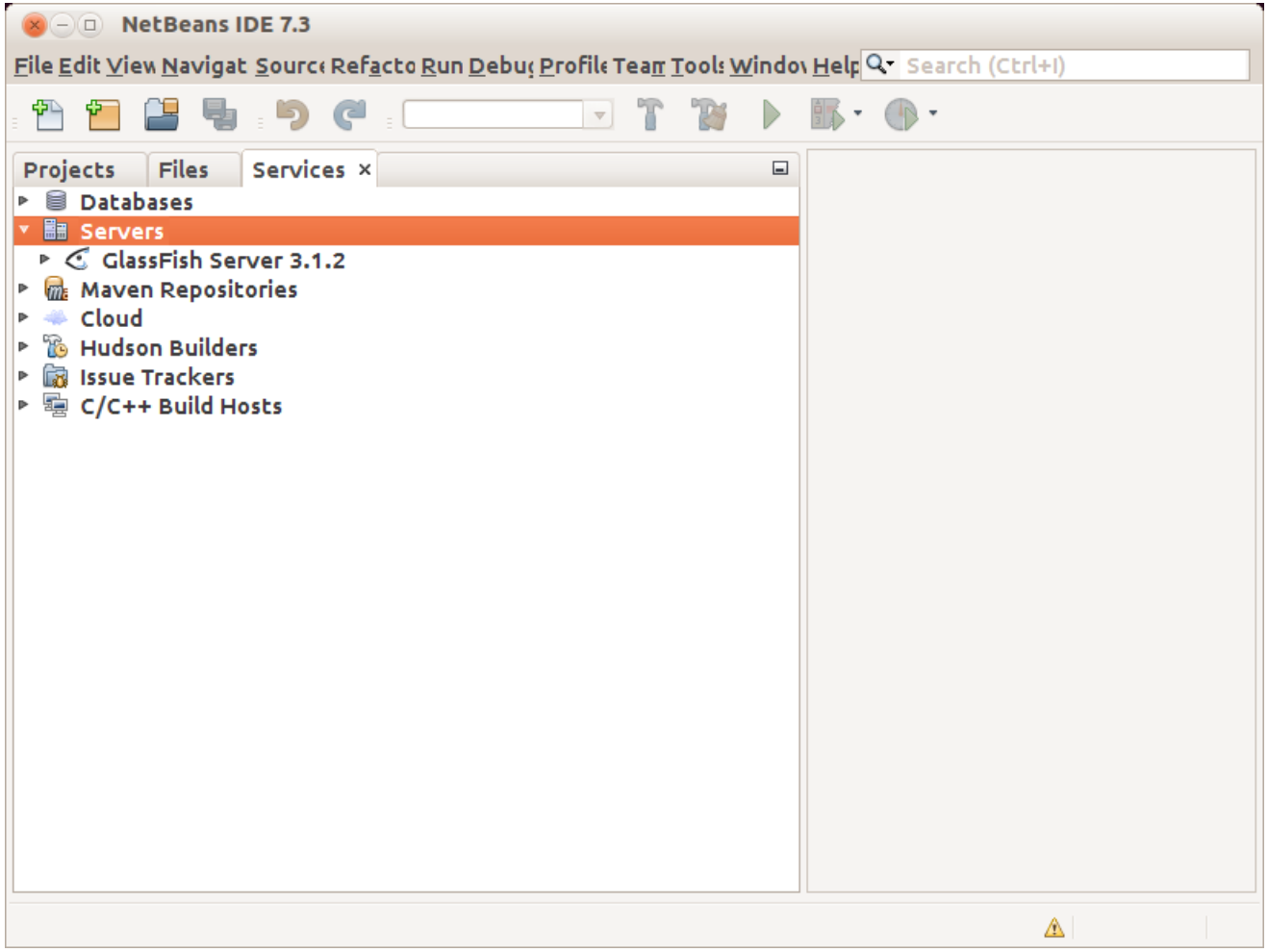
OS + file system

BIOS

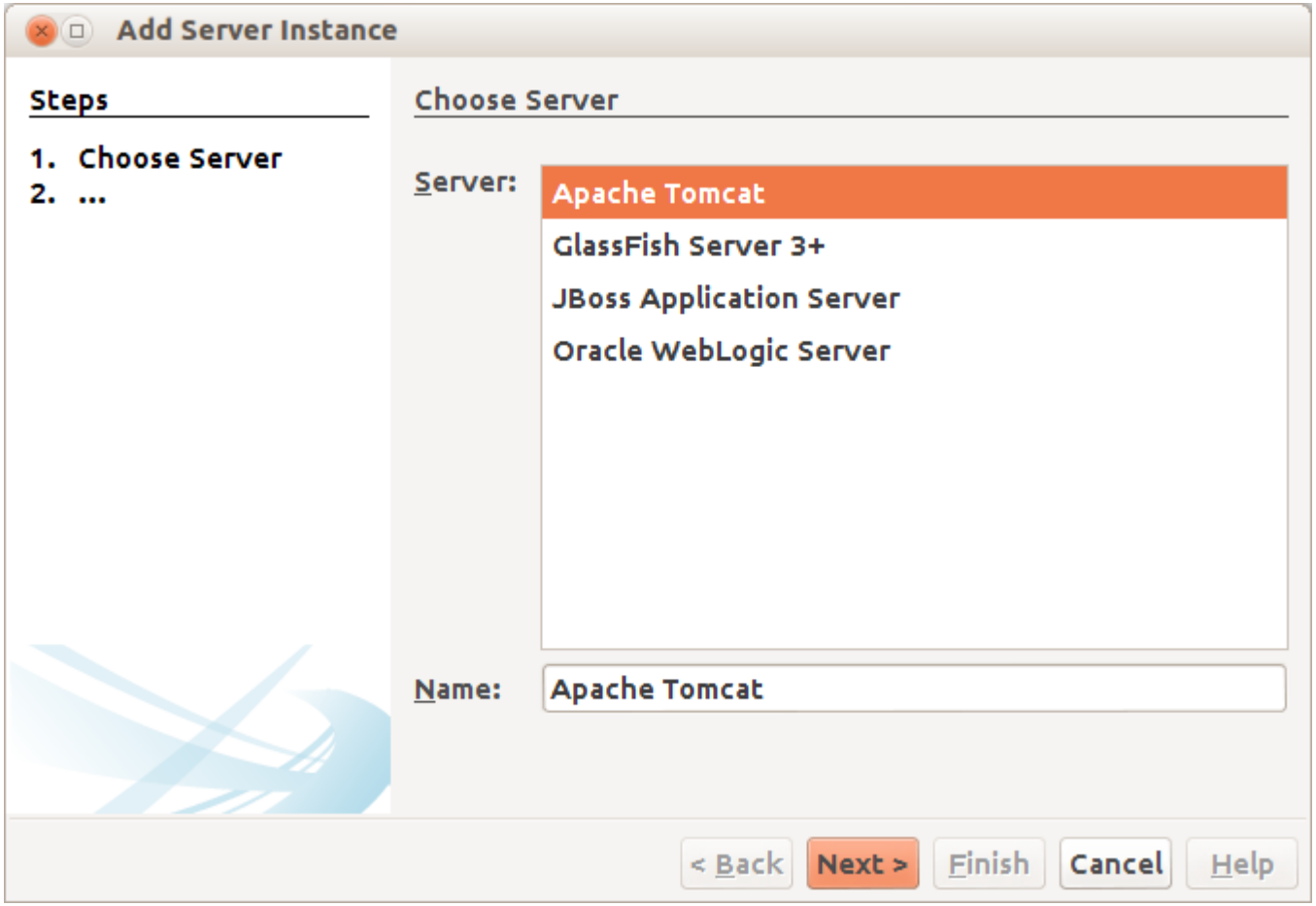
hardware



Netbeans with Tomcat

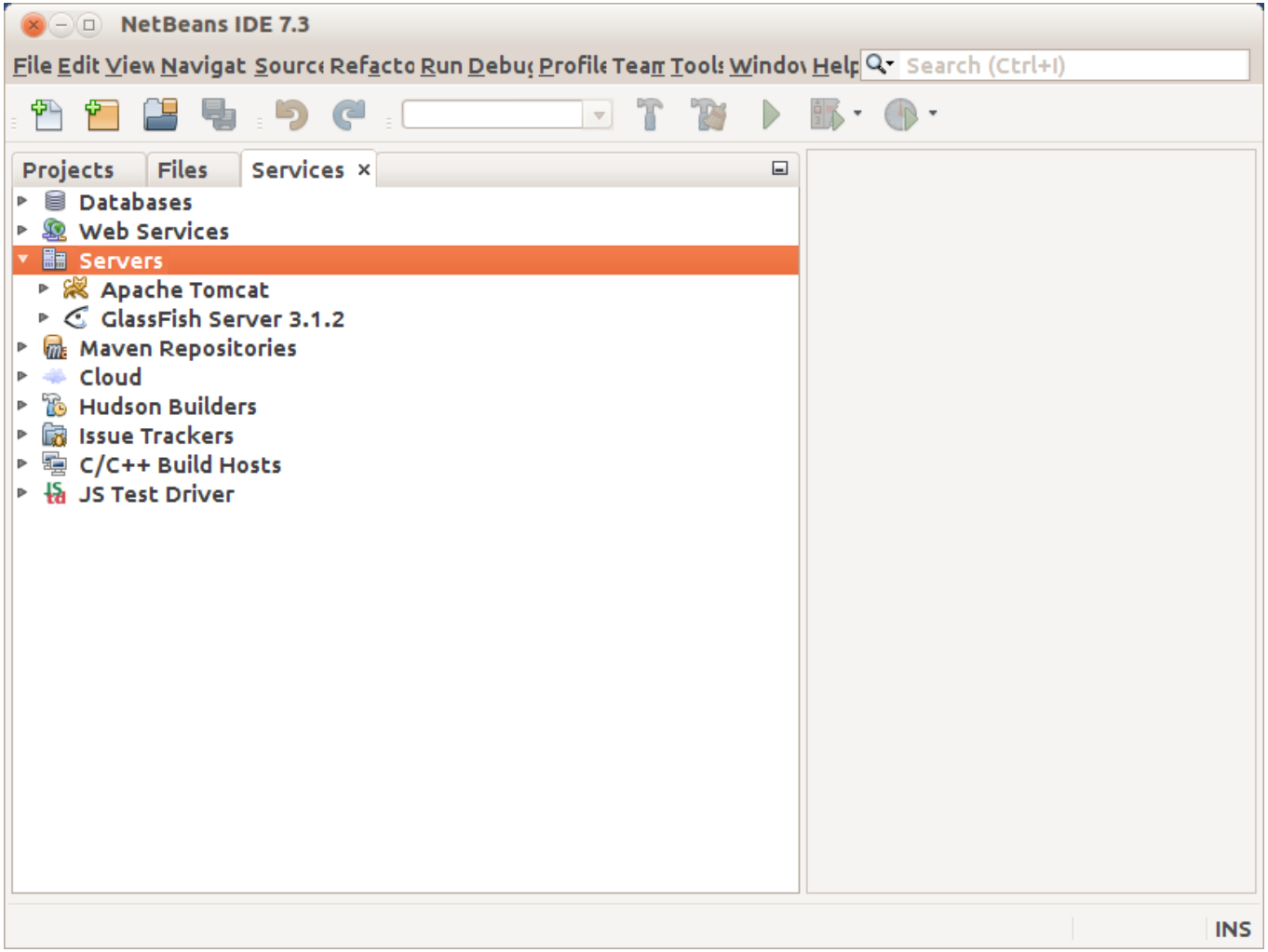


right click on servers, choose add server


Add Server Instance
Steps
1. Choose Server
2. ...
Choose Server
Server:
Apache Tomcat
GlassFish Server 3+
JBoss Application Server
Oracle WebLogic Server
Name:
Apache Tomcat
< Back
Next >
Finish
Cancel
Help

Tomcat

- Setup NetBeans
  - http://netbeans.org/kb/docs/websvc/gs-axis.html
- Setup Tomcat 7.0.39
  - Download apache-tomcat-7.0.39.zip and expand it to a folder
  - cd /home/sbvb/app/apache-tomcat-7.0.39/bin
  - chmod 755 *.sh
  - ./startup.sh
    - Look for “catalina_home” with “find / -name catalina -print”. In my computer it is “/usr/share/maven-repo/org/apache/tomcat/catalina”
  - edit conf/tomcat-users.xml; add lines below
    - <role rolename=“manager-gui"/>
    - <user username="sbvb" password="sbvb_pwd" roles="manager-gui"/>
  - login|pwd = sbvb|sbvb_pwd
  - /home/sbvb/app/apache-tomcat-7.0.39/bin/startup.sh
  - /home/sbvb/app/apache-tomcat-7.0.39/bin/shutdown.sh



Login|pwd: sbvb|sbvbpwd

Make tomcat scripts executable

- Go to folder
- <tomcat>/bin
- Write
- $ chmod 755 *.sh

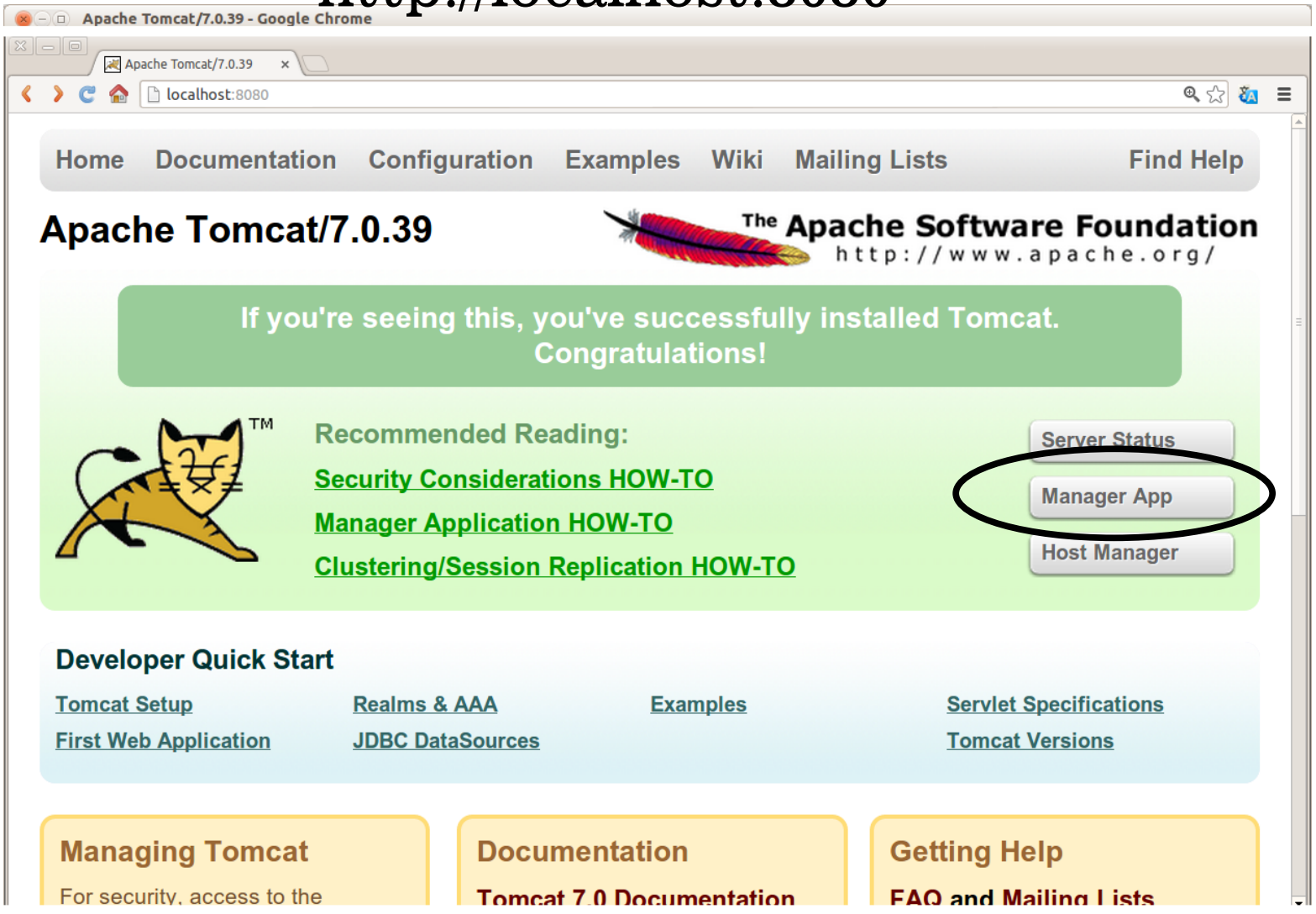
Add manager-gui

- Add gui manager role.
  - In <tomcat>/conf/tomcat-users.xml
- <user password="sbvbpwd" roles="admin,manager-gui" username="sbvb"/>

Check Tomcat is working

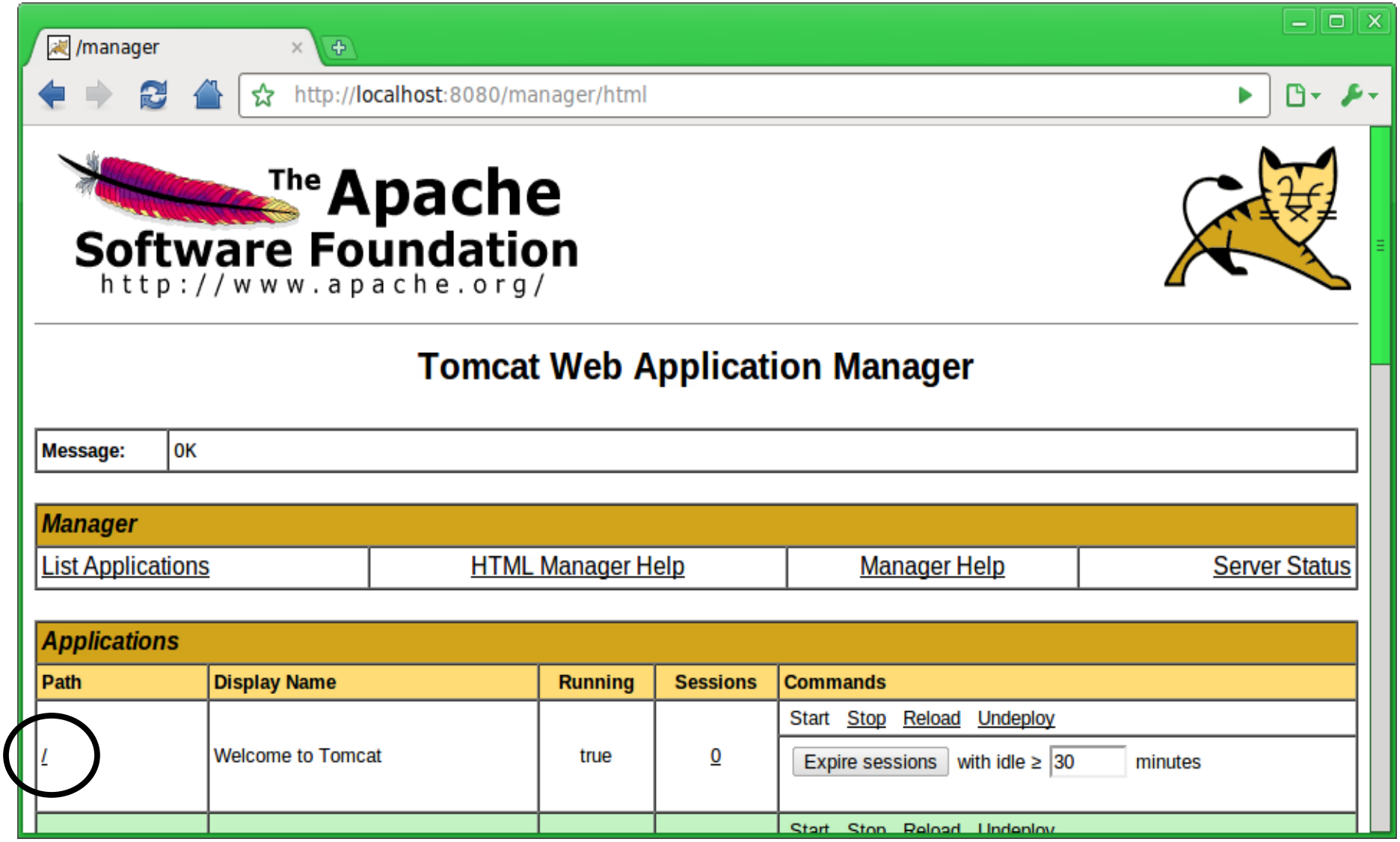
www.sbVB.com.br

http://localhost:8080



www.sbVB.com.br

Check Tomcat is working

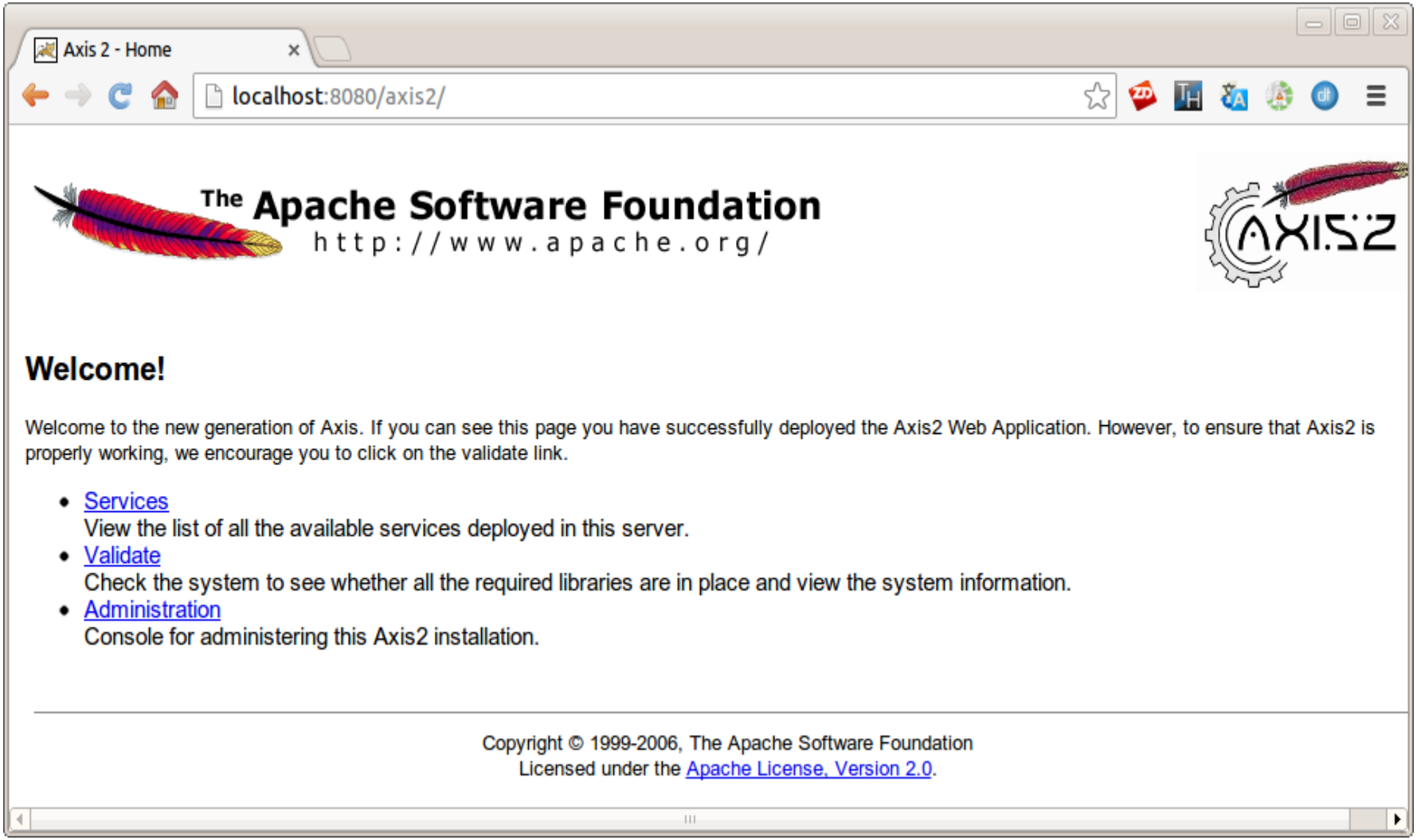


Axis2

- Download axis2.war (axis2-1.6.2-war.zip)
  - http://axis.apache.org/axis2/java/core/
    - Download releases, war distribuition
  - Copy axis2.war to <tomcat>/webapps
- Restart tomcat
- axis2 should appear as an application of tomcat. Click on it.

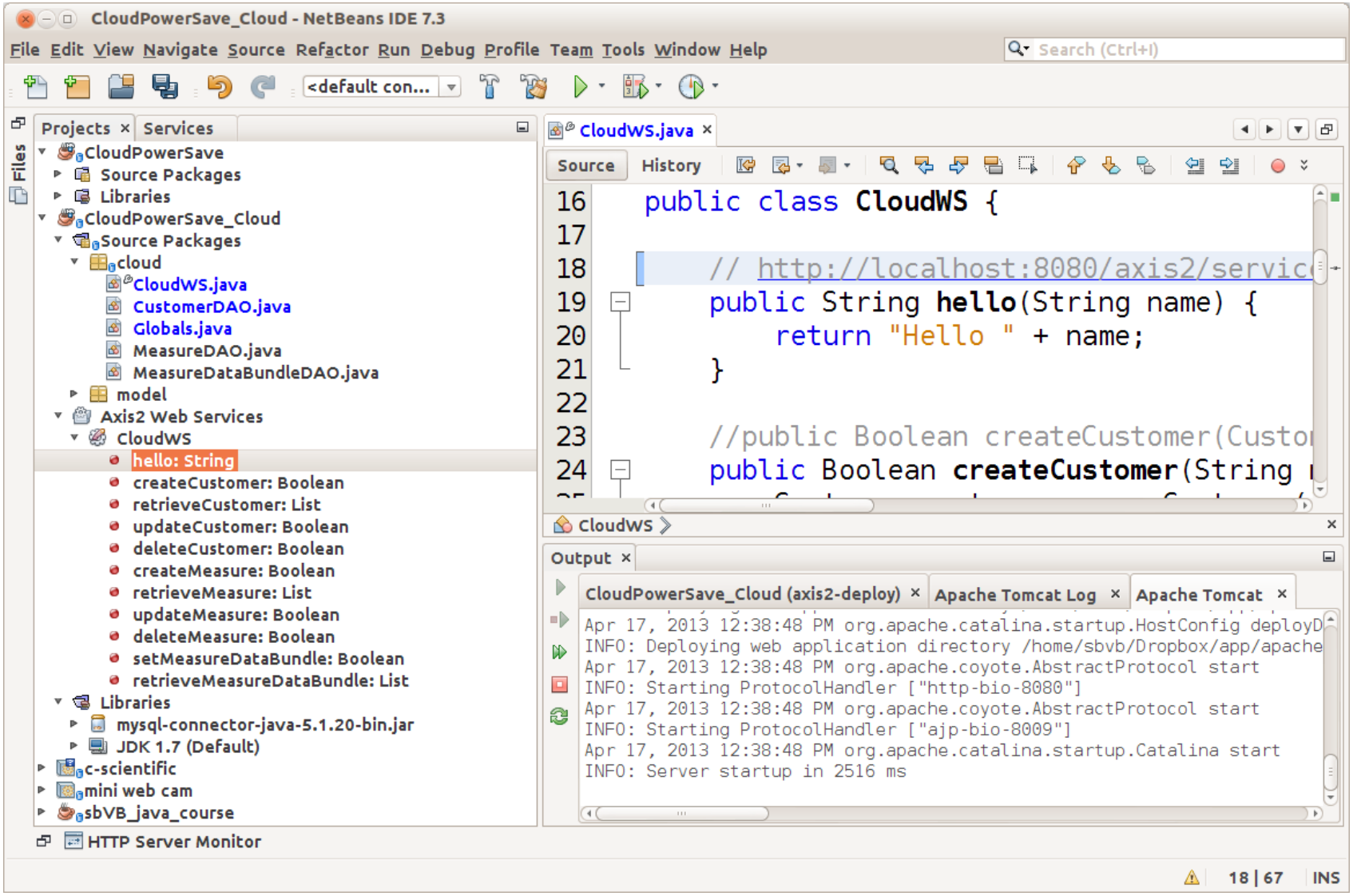
www.sbVB.com.br

axis2 services under tomcat



www.sbVB.com.br

Test web service on browser



Definitions

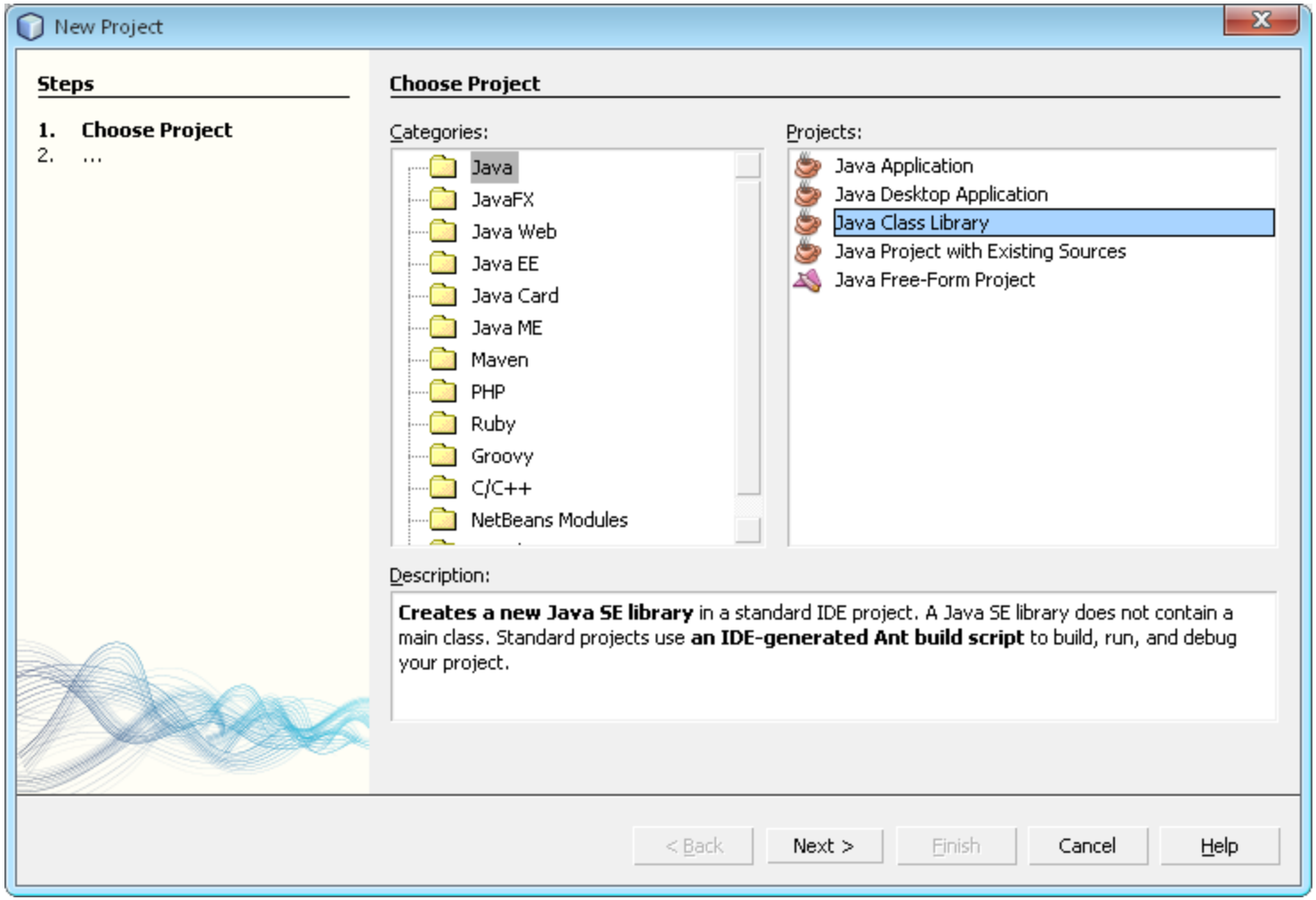
- POJO = *Plain Old Java Object*
  - The name is used to emphasize that a given object is an ordinary Java Object, not an Enterprise JavaBean, nor a DTO.

References

- http://ws.apache.org/axis2/1_2/pojoguide.html
- http://www.agileskills2.org/DWSAA/v20pub/FourtySixPages.pdf
- http://netbeans.org/kb/docs/websvc/gs-axis.html

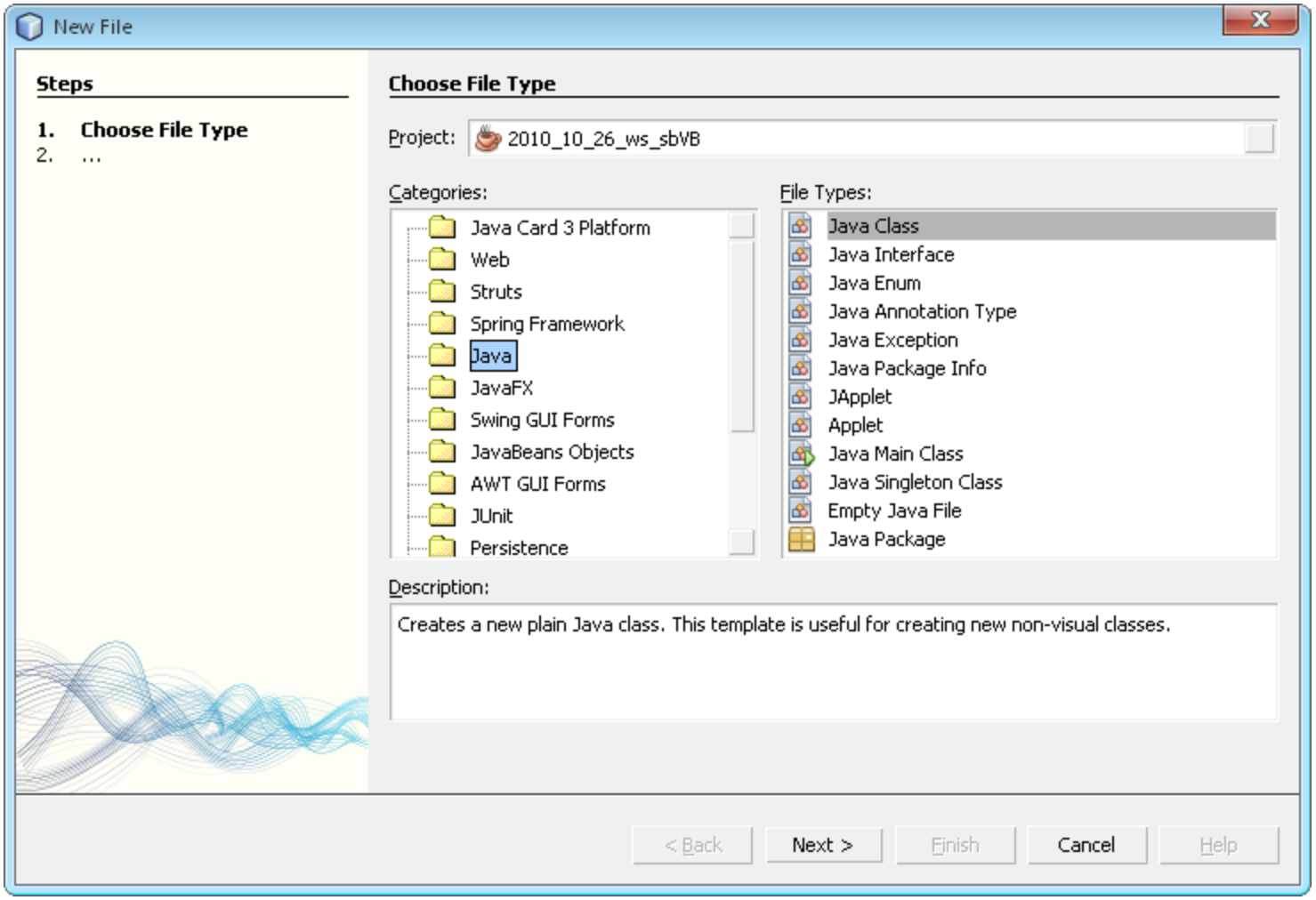
www.sbVB.com.br

New project for Web services in NetBeans



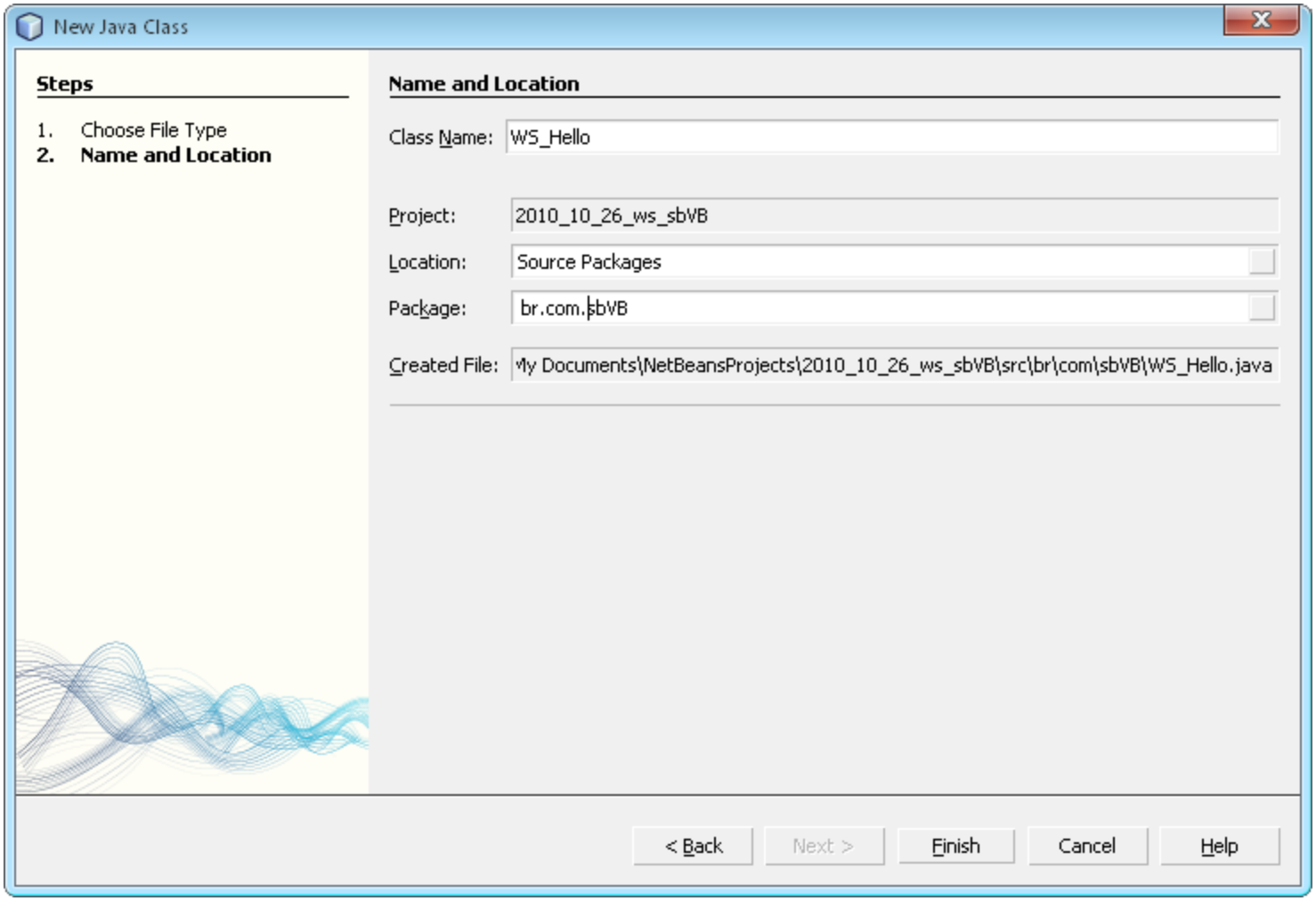
www.sbVB.com.br

Create some java class



www.sbVB.com.br

WS_Hello class



Code of HelloClass

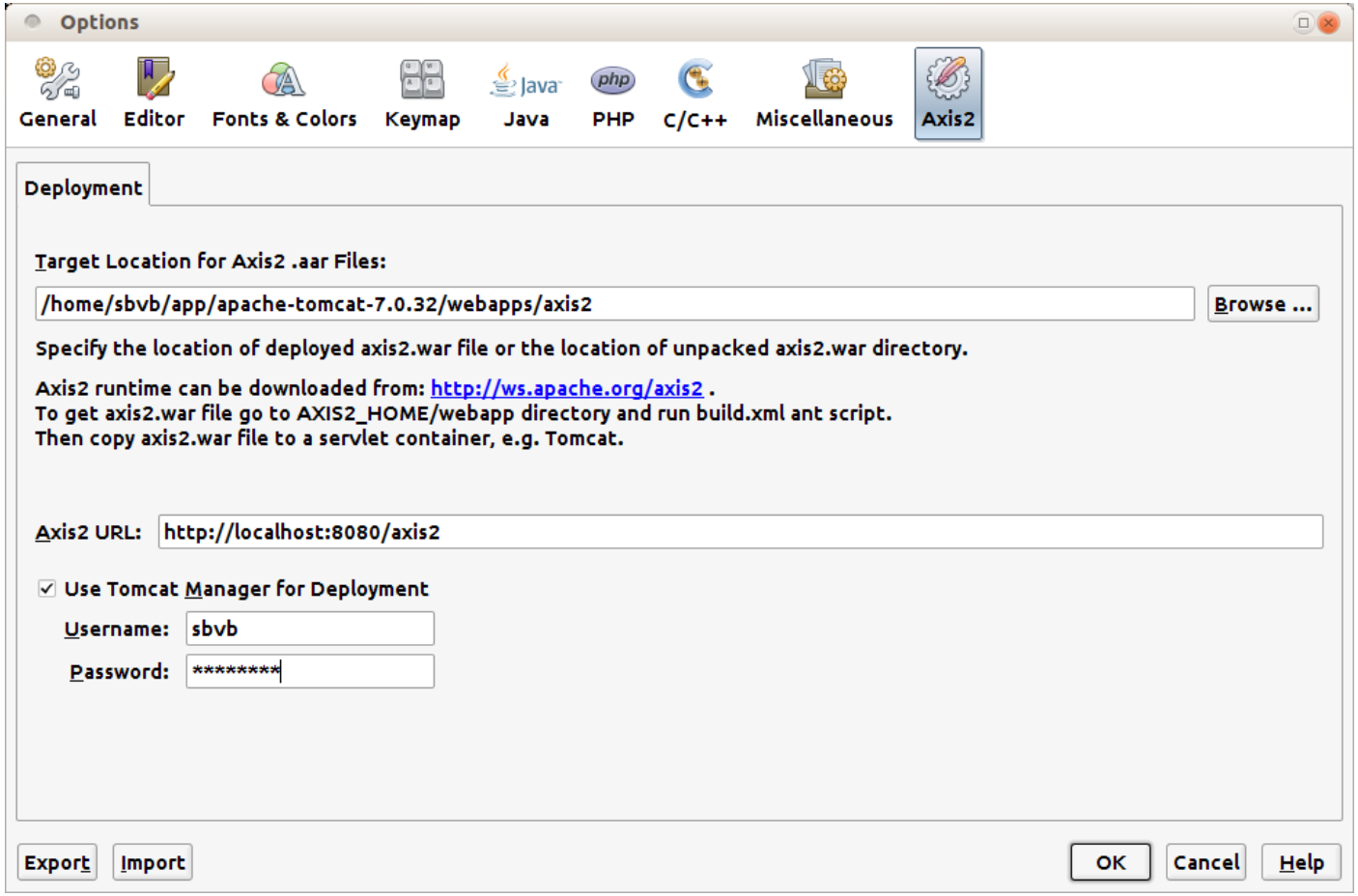
```
package br.com.sbVB;

public class HelloClass {
    public String plus_a(String in) {
        return in + "a";
    }
}
```

Add axis2 plugin to netbeans 7.x

- In Netbens menu:
  - Tools->Plugins ->- the "Settings" tab
  - Add
    - name = Dynamic Development UC update center
    - url = http://deadlock.netbeans.org/hudson/job/nbms-and-javadoc/lastStableBuild/artifact/nbbuild/nbms/updates.xml.gz
- Now you can go to the “Available Plugins” tab and the “Reload Catalog” button
- Now, you'll find the “Axis2 Support” plugin in the list, and can install it.
- Restart Netbeans
- Configure target location for Axis2.aar files
  - <tomcat>/webapps/axis2/WEB-INF/services

Configure folder of tomcat in Netbeans plugin axis2

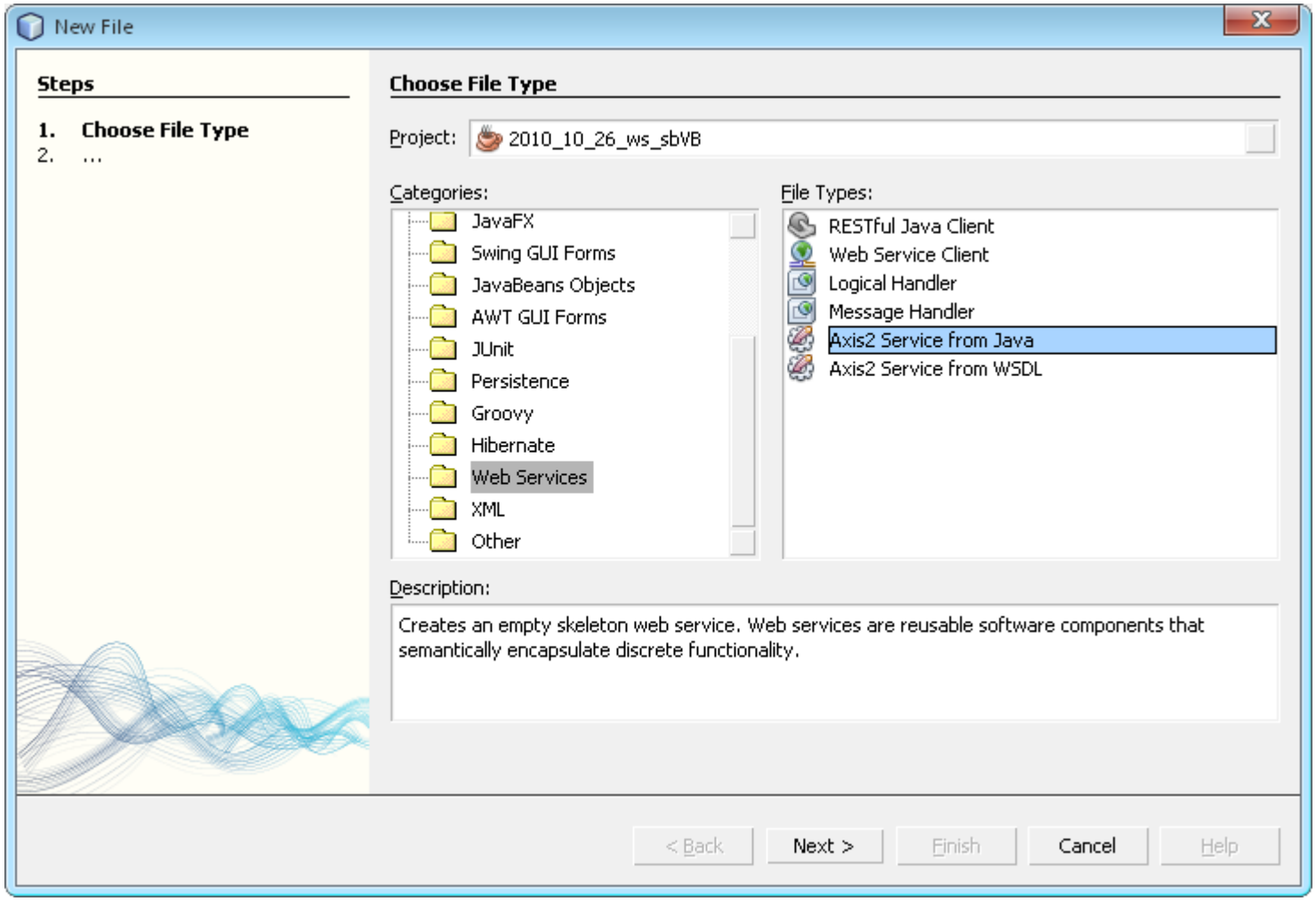
Menu: tools->options



www.sbVB.com.br

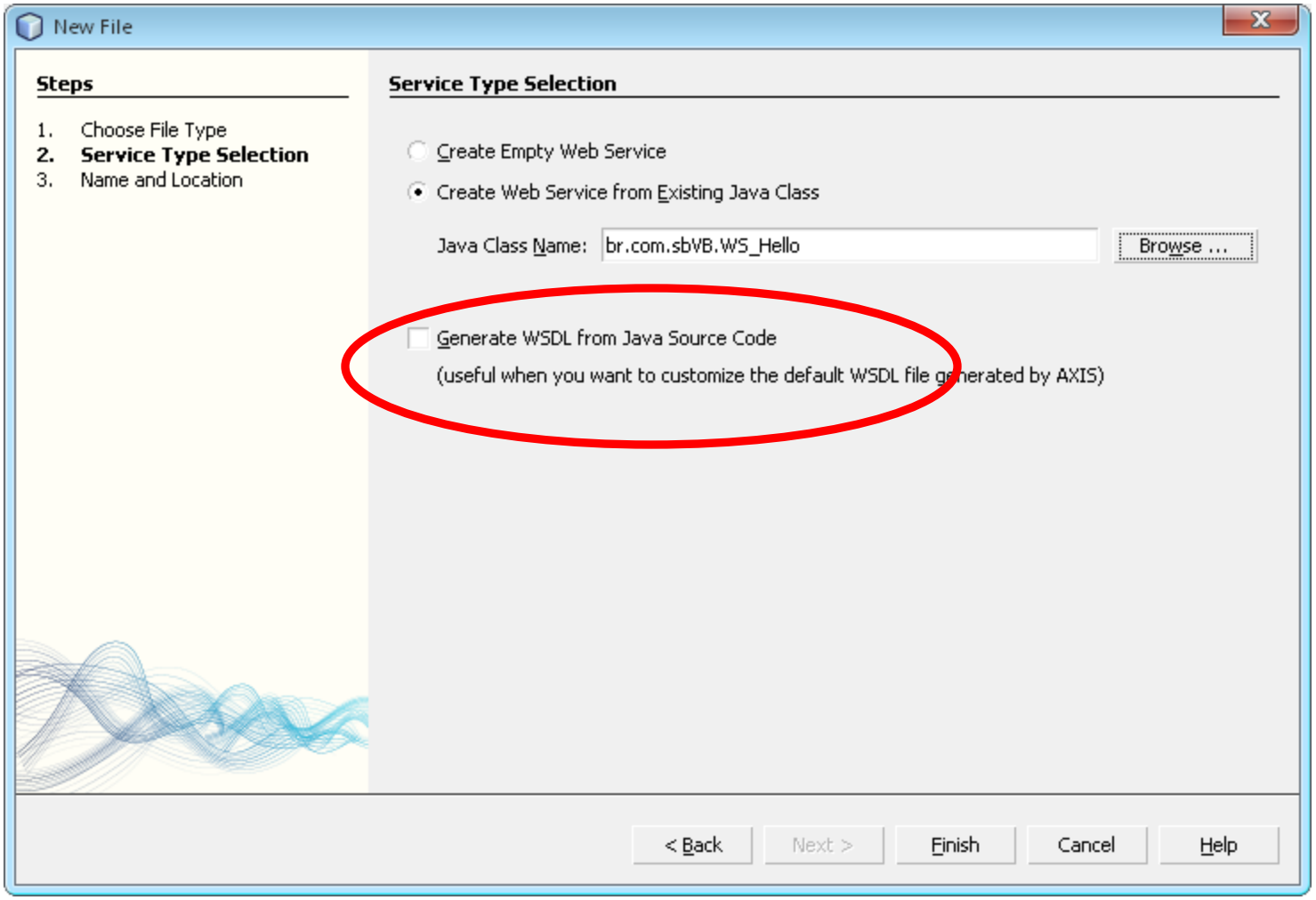
Using Netbeans with axis2plugin, add Web Services to project

Menu: File->New File



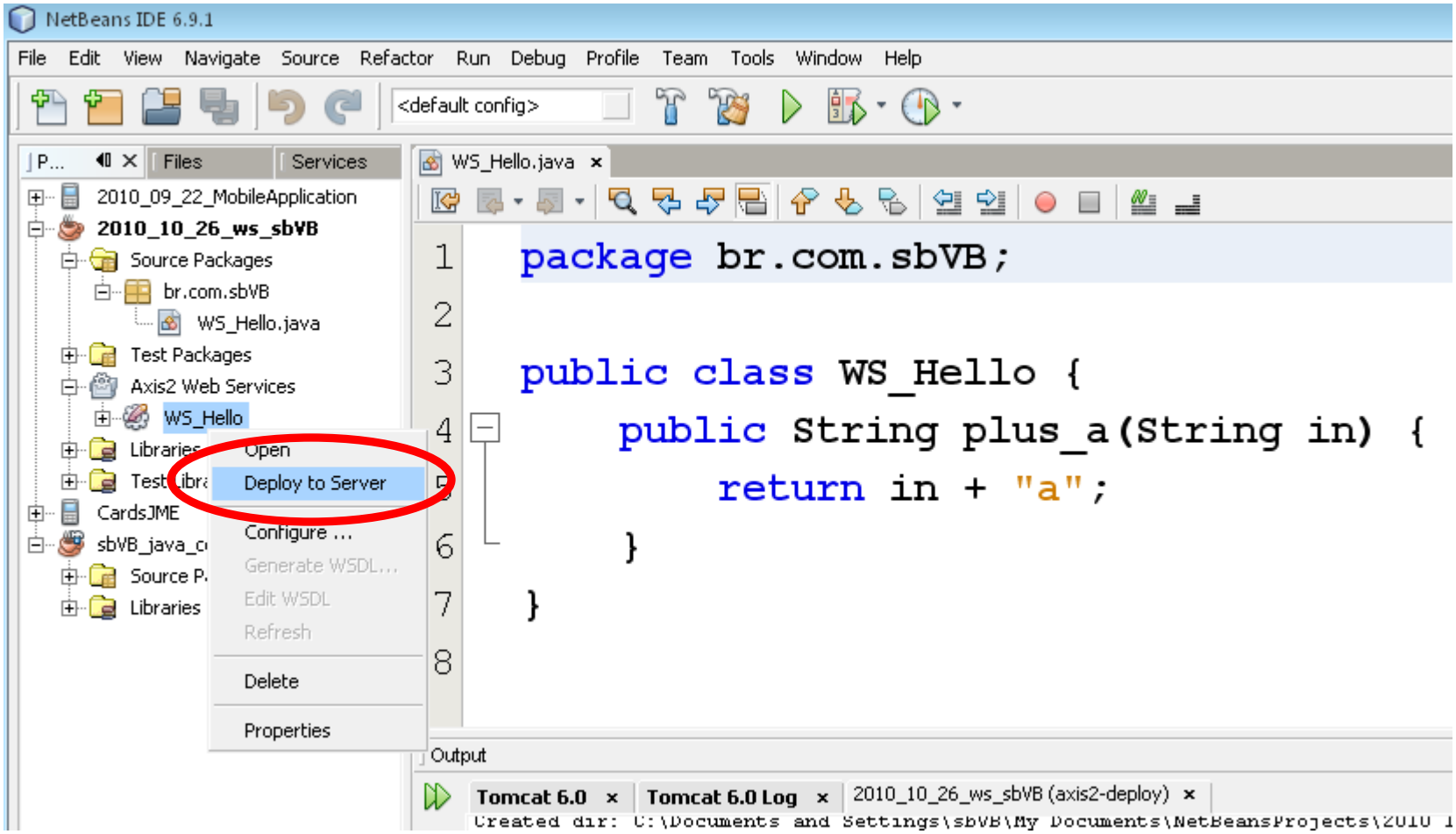
www.sbVB.com.br

Don't mark that check box !

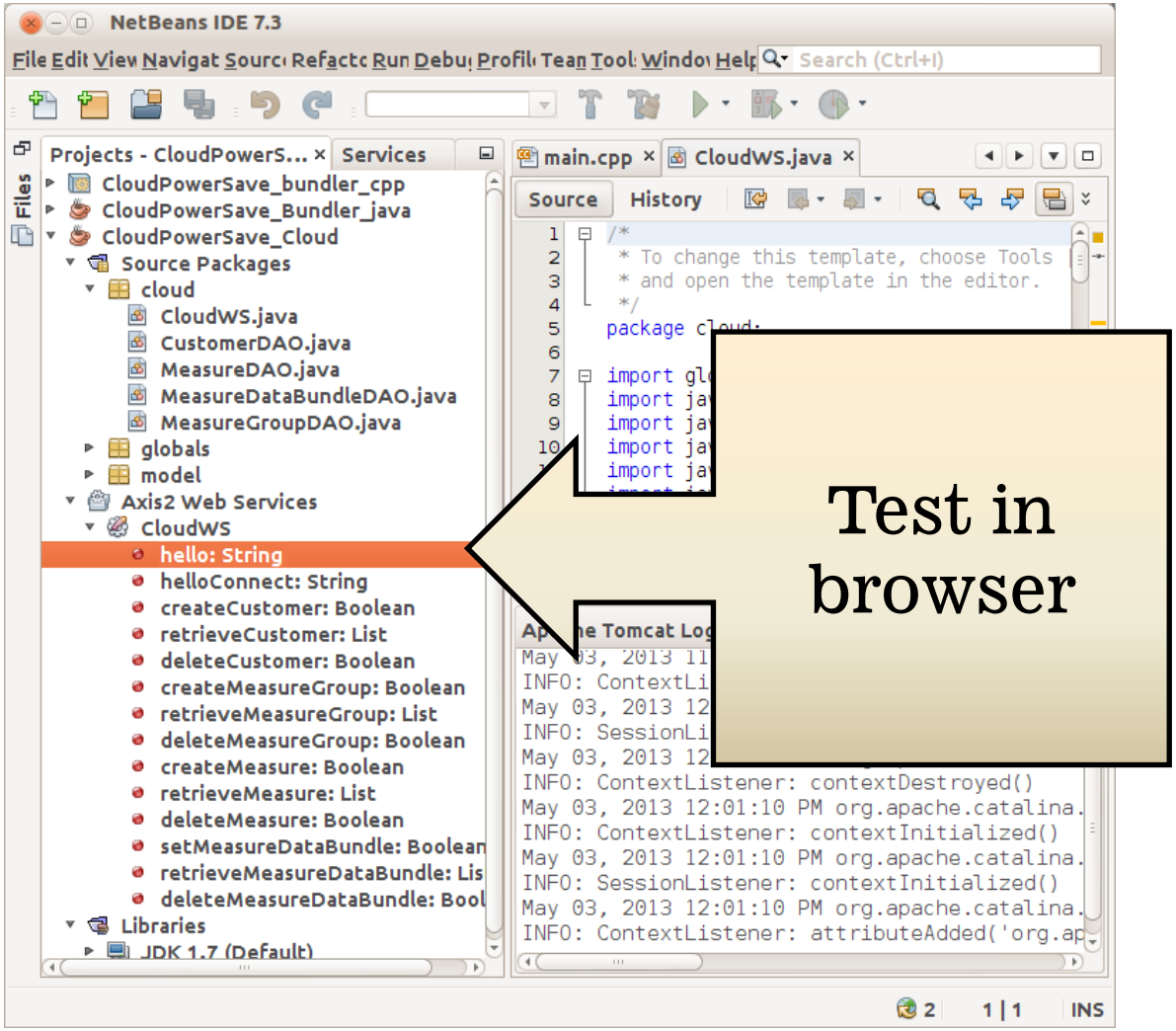


www.sbVB.com.br

Right-click and deploy



Test in browser



Accessing the web services from the web browser

- http://localhost:8080/axis2/services/Version/getVersion

```
<ns:getVersionResponse
xmlns:ns="http://axisversion.sample">
<ns:return>Hi - the Axis2 version is
1.6.2</ns:return>
</ns:getVersionResponse>
```

Hello WS plus_a

- http://localhost:8080/axis2/services/HelloClass/plus_a?in=banan

```
<ns:plus_aResponse xmlns:ns="http://sbVB.com.br">
<ns:return>banana</ns:return>
</ns:plus_aResponse>
```

Folder of services

- /home/sbvb/app/apache-tomcat-7.0.32/webapps/axis2/WEB-INF/services

- // add custom jars to folder below
- /home/sbvb/app/apache-tomcat-7.0.32/webapps/axis2/WEB-INF/lib

Amazon cloud

Basics

- EC2 (Elastic Computing Cloud)
- AMI (Amazon Machine Image)
- EBS (Elastic Block Storage)
- EBS-backed AMI × store-backed instance

Some web pages

- Home page of service
  - http://aws.amazon.com/
- Console to control
  - https://console.aws.amazon.com/ec2/
- Resource center
  - http://aws.amazon.com/developertools/
- Resource center
  - http://aws.amazon.com/developertools/