



# Deployment (Theory)

Lecture notes

UNIVERSIDAD DE SEVILLA

Welcome to this lecture! Today, we're going to study some topics about deploying your web information systems.

--

Copyright (C) 2018 Universidad de Sevilla

The use of these slides is hereby constrained to the conditions of the TDG Licence, a copy of which you may download from <http://www.tdg-seville.info/License.html>

## What's deployment about?

---



UNIVERSIDAD DE SEVILLA

2

As usual, we start with a question: what's deployment about?

## What's deploying a system?

---



Deploying a system means installing it on a server computer, which more often than not resides on the cloud

Deploying a web information system means installing it on a server computer. Nowadays, that computer's typically a cloud computer, that is, a virtual or a physical computer that is rented from a provider, e.g., Amazon, GoGrid, or OVH, to mention a few examples. Nowadays, it's also quite common that many cloud providers don't rent their computers but their so-called platforms, which are combos that provide a database server and an application server so that you can forget about the hardware or the operating system. Jelastic, CloudFoundry, or CloudBees are well-known platform providers.

## How is it performed?

---



It's now time to ask yourself regarding how a typical deployment's performed. As usual, please, try to produce an answer before peeking at the following slides.

## Step 1: understand configurations

---



Before deploying a system, you must understand the idea of configuration. We talked a little about configurations in the first lesson, but it's time to delve into additional details.

## Step 2: set up a pre-pro configuration

---



Then you have to set up a so-called pre-production configuration, aka pre-pro configuration, which is a clone of your customer's servers.

## Step 3: deploy your database

---



Then you have to deploy your web information system database to the pre-production configuration.

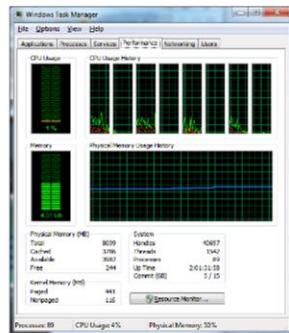
## Step 4: deploy your application

---



And, then, you have to deploy your application to the pre-production configuration.

## Step 5: performance and acceptance



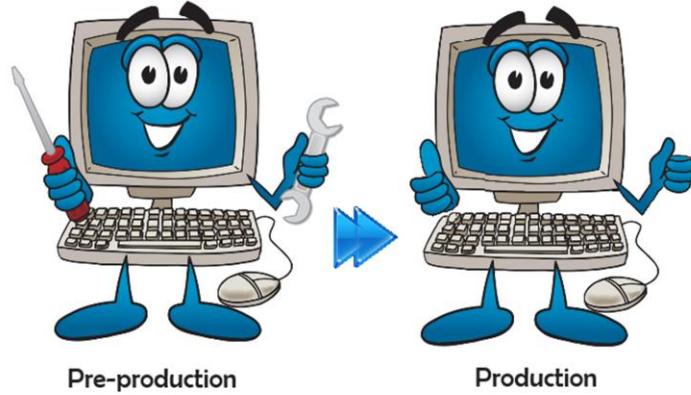
Shall study that in  
Lessons L10 and L11



Once your web information system's been deployed to the pre-production configuration, you can conduct performance and acceptance tests on it. We won't report on these topics in this lesson, since we're going to study them during the next semester.

## Step 6: deploy to production

---



Once the system's been tested, you're sure it performs as expected, and you perfectly know how to install it on the pre-production configuration, it's time to deploy it to the production configuration, that is, to your customer's servers. Yes, it's also time to invoice your customer and earn money! This step should be quite trivial.



We think this is enough to introduce this lesson. The rest's organised as follows: first, we'll report on configurations and will provide a few more details on the one that is commonly referred to as the pre-production configuration, or pre-pro for short; then, we'll report on how to deploy your databases and your applications.



Let's start with a little theory on configurations.

## What's a configuration?

---



It's a computer system that's specifically tailored towards a particular stage in the lifecycle of a web information system

A configuration's a computer system that is specifically tailored towards a particular stage in the lifecycle of a web information system. By computer system, we mean a piece of hardware plus an operating system and a number software tools.

**NOTE:** many people refer to “configurations” as “environments”. Both terms are equivalent, so use the one that you prefer.

## Stages of a typical project



In this slide, we present the most typical stages in the development of a web information system, namely:

- Presales: it deals with the marketing activities required to convince our customers that our company can help them.
- Planning: if the presales stage succeeds, it's then time to plan on a project, which includes eliciting the requirements, planning on a budget, human resources, facilities, calendars, and so on.
- Development: this is a stage with which you're very familiar; it's the stage in which the web information systems are designed, implemented, and tested from a functional point of view.
- Deployment: in this stage, performance and acceptance tests are carried out and, if everything is OK, then the web information system's installed on your customer's servers.
- Maintenance: in this stage, the web information system's evolved and errors that weren't discovered during the previous stages are corrected.

Some people say that the presales stage and the planning stage focus on the “what” (What are our customer’s needs?), whereas development and deployment focus on the “how” (How can we satisfy them?), and maintenance on the “why” (Why didn’t we catch this error? Why didn’t we realise that would be a requirement? Why don’t you implement this new requirement?).

# Organisation of stages

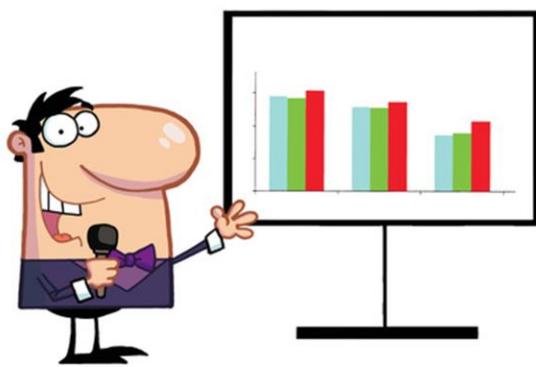
---



Please, don't think of the previous stages as if they were organised linearly. In practice, they overlap and it's very common that the same stage's scheduled several times during a project. What matters in this lesson is that every stage requires a different configuration. In the following slides, we provide an insight.

## The presales stage

---



Presales configuration: office suite (word editor, slide editor, spreadsheet editor, contact manager, calendar manager), messaging utility, and the like.

The presales stage starts with a meeting with a prospective customer and often ends once a project's been sold to him or her. Typical activities involve preparing a proposal, presenting our products and/or services, and the like. A typical presales configuration's a computer with a good office suite (word editor, slide editor, spreadsheet editor, calendar manager, contact manager, and so on), a messaging utility, and similar tools. A presales manager's a person who's related to marketing a company's products and/or services, so there's nothing related to development in the configurations with which they work.

## The planning stage

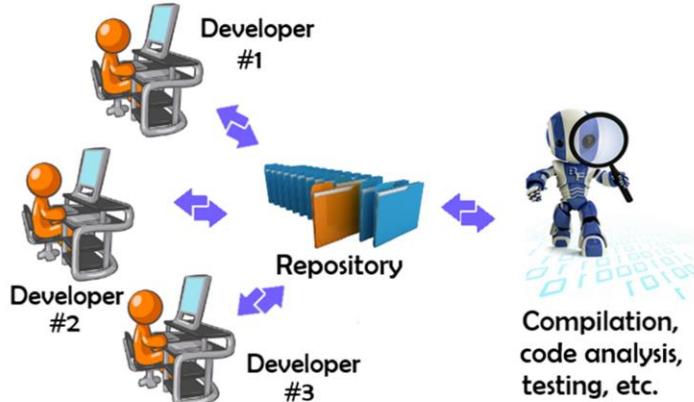
---



Planning configuration: requirements elicitation tool, mock-up design tool, project management tool, office suite, messaging utility, and the like.

The planning stage involves every activity required to understand a customer's needs and prepare a project to satisfy them. Technicians who work on this stage usually require a configuration with a requirements elicitation tool, a tool to design mock-ups, a project management tool, a typical office suite, a messaging utility, and other such tools. This configuration's more related to development, but the focus is on modelling the system that is going to be developed, not on implementing it.

## The development stage (I)



The development stage typically consists of a series of short cycles in which developers work individually on a few parts of the system that is being developed and their results are integrated as soon as possible. It requires two configurations that are commonly referred to as the developer's configuration and the integration configuration. We provide additional details in the following slides.

## The development stage (II)

---



Developer's configuration: workbench, UML modelling tool, repository client, database server, application server, office suite, messaging utility, and the like.

The developer's configuration's the one with which you're most familiar. It's a computer with a developer's operating system atop which you've installed a workbench, a UML modelling tool, a repository client, a developer's database, a developer's application server, an office suite, a messaging utility, and other such tools. You must have every tool you need to develop your web information systems in your development configuration.

**NOTE:** please, realise that you've been working with this configuration since the beginning of the course. We provided you with a virtual machine that implements this configuration roughly three months ago.

## The development stage (III)



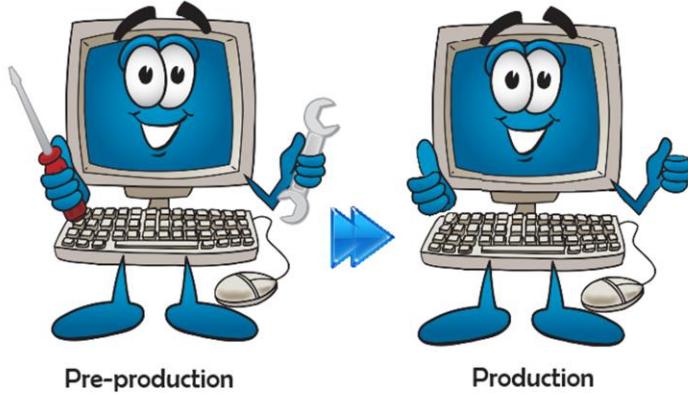
Integration configuration: repository server, database server, integration server (compiler, code analyser, functional tester).

The integration configuration's a computer with a repository server, a database server, and an integration server. Developers commit their projects to the repository server, which merges the individual parts on which they've been working independently; there's little new we can say now about the database server; the integration server's very likely something completely new to you. Typically, they provide a compiler, a code analyser, and a functional tester. Integration servers are typically configured so that they read every project from the repository server every night; they compile them, analyse their code to find bad smells, and execute functional tests to find bugs; finally, they make the results available to the developers so that they can start working on the problems thus found as soon as possible.

**NOTE:** we're sorry, we can't provide you with an integration server; we can provide you only with a repository server by means of ProjETSII.

## The deployment stage (I)

---



The goal of the deployment stage is to install the web information system on which you've been working on your customer's server. In this stage, we have to use two configurations: the pre-production configuration and the production configuration, which we explain in the following slides.

## The deployment stage (II)

---



Pre-production configuration: database server and application server. (It might provide some simple additional tools.)

The pre-production configuration's a clone of your customer's server. It allows you to get familiar with your customer's computing facilities and solve installation problems without interfering with your customer's normal operations; typically, the pre-production configuration's also used to carry out performance and acceptance tests, on which we'll report during the Spring semester. Usually, the pre-production configuration has a database server and an application server; it might have some simple additional tools, but very little else.

## The deployment stage (III)

---



Production configuration: database server  
and application server.

The production configuration's the actual computer on which our web information system will be deployed at your customer's computing facilities. They must be used on-line by your customer's employees and his or her customers, so you'd better finish your deployment as soon as possible. Making a mistake in a production configuration usually has an impact on your customer's business; if this happens, don't expect him or her to be very happy with you!

## The maintenance stage

---



The maintenance stage starts when a web information system's deployed to the production configuration and the customer reports errors that weren't identified in the previous stages or requires us to evolve the system to include new functionality or to adapt it to new requirements. That means that you're likely to find a mix of configurations in this stage; virtually every of the previous configurations might be useful in this stage.



It's now time to provide an insight into the pre-production configuration that you have to use in this subject.

## What's such a configuration?

---

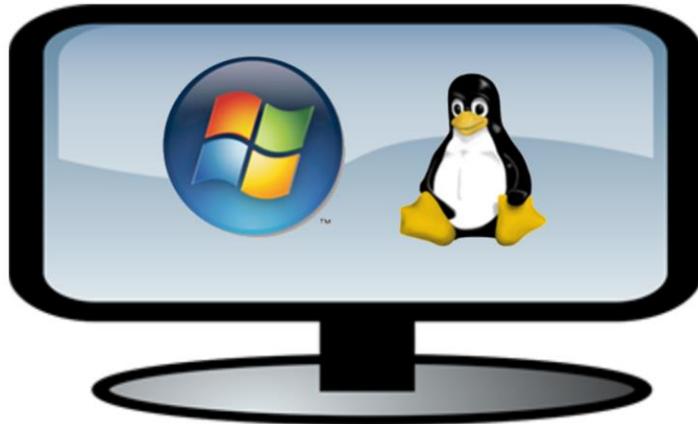


It's a clone of your customer's server  
(with some simple tools, if necessary)

So far, you must know that the pre-production configuration's a clone of your customer's server that you must use to learn how to deploy your web information systems to your customer's actual computing facilities and to conduct performance and acceptance tests. You must also know that it may have some simple tools for debugging purposes, but little else.

# The operating system

---



The pre-production configuration must have a server version of a good operating system. Windows or GNU/Linux are good choices; unfortunately, Apple does not release any server version of their operating systems. Please, recall that you should almost never work as an administrator in your developer's configuration; in the pre-production configuration, however, the only few operations that you need to perform are commonly administrative ones, so you must use your administrator account all the time. Please, do not create any other user accounts except for the non-privileged accounts to run your database server and your application server.

**NOTE:** Windows provides a couple of non-privileged user accounts to run services, namely: "Local Service" and "Network Service"; most servers are installed so that they run on the appropriate user account automatically. GNU/Linux doesn't have a standard user account to run services; please, consult your distribution's documentation to find out which the most appropriate account is or to create a new one.

## Configure the operating system



It's very important that you configure the operating system for maximum performance and security. Please, review the following items:

- a) Have a single administration account; make sure the username's not "administrator", "administrador", "root", or something like that. These username's are far too obvious for hackers.
- b) Remove every unnecessary component, e.g., desktop accessories, mail clients, media players, messaging utilities, and the like; just keep a browser, if necessary.
- c) Remove every unnecessary service, e.g., multi-media services, theme services, DNS servers, and the like.
- d) Configure the firewall for maximum protection; only port 80 should be available to external customers.
- e) Search for other security guidelines that are applicable to your operating system and apply them.

## Install your servers

---



UNIVERSIDAD DE SEVILLA

29

Install a MySQL Server and a Tomcat server. Please, recall that so far you've been using developer versions; it's time to install production versions and to configure them so that they perform optimally and start up automatically when your server restarts.

## And set www.acme.com up

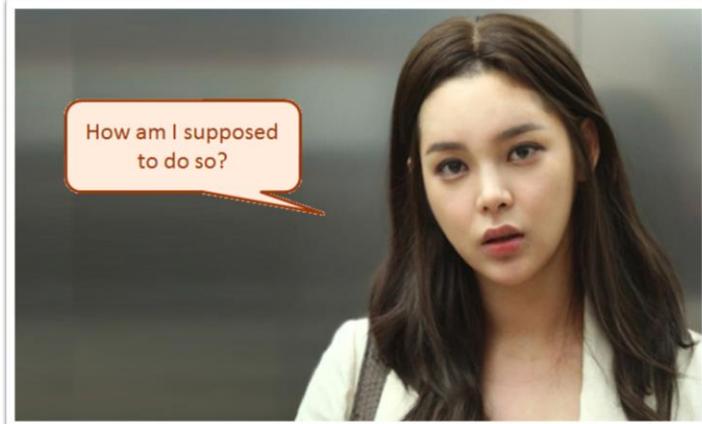
---



Finally, set up your pre-production configuration so that it emulates your customer's internet domain. It's important that you work on your customer's internet domain to make sure that your system works well; otherwise, if you work with localhost or your company's internet domain, there might be some errors in your code that would go unnoticed until the system's deployed to the production configuration. In this subject, we're going to use "www.acme.com" all the time.

## Keep reading!

---



We assume that installing the operating system or setting up a MySQL or a Tomcat server should not be a problem to you. Read the manuals and it should be like ringing a bell. Unfortunately, emulating an internet domain's not the kind of information you're likely to find in a manual, so we're going to provide you with a guideline.

## Let's hack your configuration

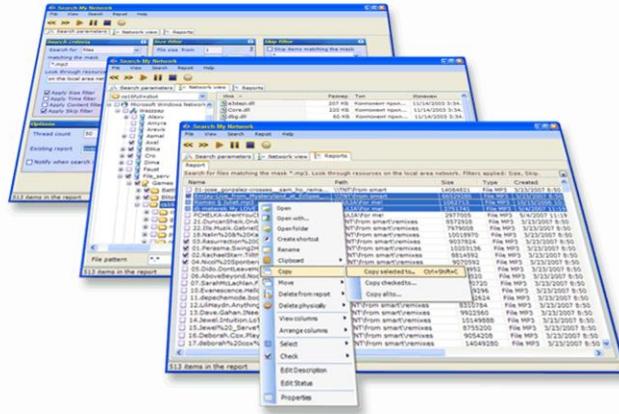
---



Before delving into the details, please, realise that we need to hack your computer's configuration. It's not difficult, but it's very low-level. It's kind of entering the Matrix!

**NOTE:** hacking a system so that it emulates an internet domain's highly dependent on your operating system. We provide you with a few hints if you're using GNU/Linux, but we can't provide a general solution that works with every distribution of this operating system. Please, consult your operating system's documentation or search the Web for information when necessary.

## Set up your hosts file

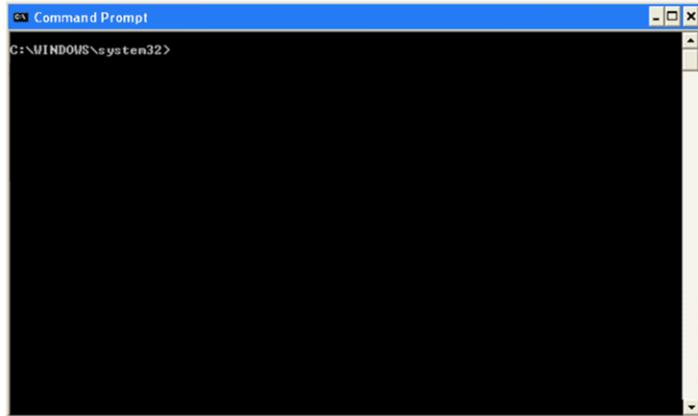


UNIVERSIDAD DE SEVILLA

33

Have you ever heard of a file called “hosts”? It provides your operating system with a local DNS translation table by means of which you can override the records in public DNS servers. Simply put, you can configure your operating system so that every time you make your browser for “www.acme.com” it locally translates this address into “localhost”.

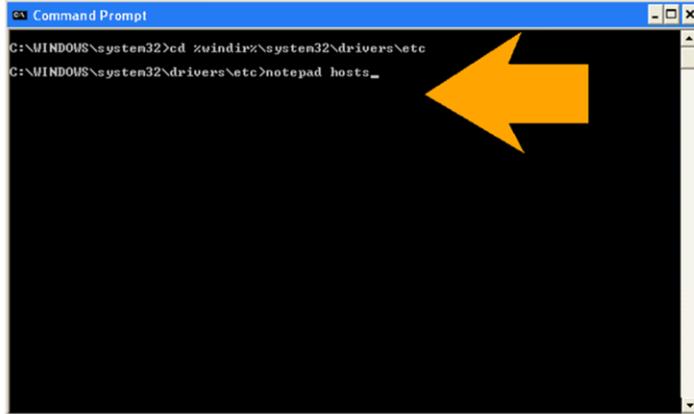
## Open an administrator's shell



To change your “hosts” file, you need to work on an administrator’s shell. Just press Windows+R and type “cmd.exe”, which will result in a shell like the one in this slide.

**NOTE:** from now on, unless otherwise stated, we’ll assume that you’re working on a pre-production configuration in which there’s only an administrator’s account. Please, consult your operating system’s documentation if you’re not using Windows.

## Locate file hosts

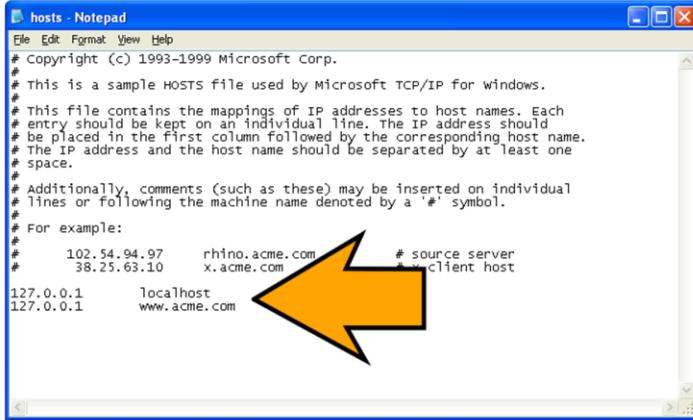


Now locate the “hosts” file and open it in a text editor using the following commands:

```
> cd %windir%\system32\drivers\etc  
> notepad hosts
```

**NOTE:** please, consult your operating system’s documentation to find out where your “hosts” file resides. There’s a document at [http://en.wikipedia.org/wiki/Hosts\\_file](http://en.wikipedia.org/wiki/Hosts_file) with information regarding this file’s path in many operating systems.

## Edit and save it



```
hosts - Notepad
File Edit Format View Help
# Copyright (c) 1993-1999 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#      102.54.94.97    rhino.acme.com      # source server
#              38.25.63.10    x.acme.com        # client host
127.0.0.1      localhost
127.0.0.1      www.acme.com
```

UNIVERSIDAD DE SEVILLA

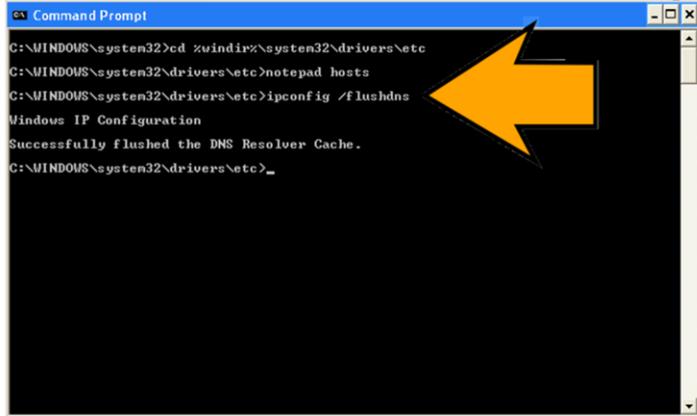
36

This is what the file looks like. It has a lot of comments that start with a hash sign.  
Add the following two lines at the end:

```
127.0.0.1      localhost
127.0.0.1      www.acme.com
```

You should know that “127.0.0.1” is your computer’s loopback address. These two lines define two hosts called “localhost” and “www.acme.com” that have exactly your computer’s loopback address. Save the file when you’re done. “www.acme.com” belongs to your customer, so it must be mapped onto an IP like “216.27.178.28” in public DNS servers. Thanks to the “hosts” file, you can override public DNS records so that when your computer needs to dereference “www.acme.com”, it actually gets IP “127.0.0.1” from the DNS service, that is, your local host.

## Refresh your DNS cache



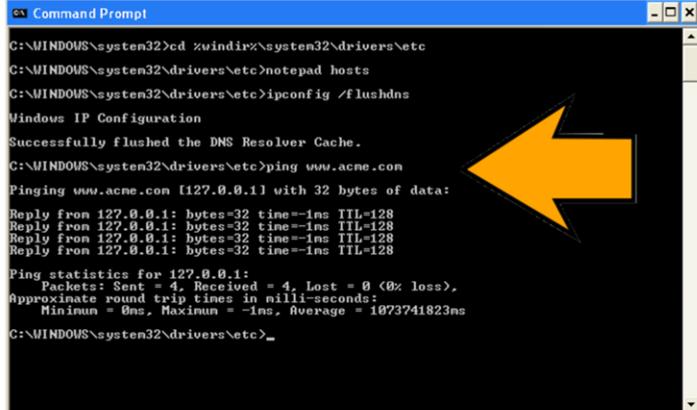
Now you might need to refresh your DNS cache by means of the following command:

> ipconfig /flushdns

This instructs your operating system to reload the “hosts” file.

**NOTE:** please, consult your operating system’s documentation to find out how the DNS host cache is refreshed. Unfortunately, there’s not a standard way to flush the cache in Unix-based operating systems. You may try the hints that are available at <https://help.dreamhost.com/hc/en-us/articles/214981288-Flushing-your-DNS-cache-in-Mac-OS-X-and-Linux>.

## Ping www.acme.com



```
ps Command Prompt
C:\WINDOWS\system32>cd %windir%\system32\drivers\etc
C:\WINDOWS\system32\drivers\etc>notepad hosts
C:\WINDOWS\system32\drivers\etc>ipconfig /flushdns
Windows IP Configuration
Successfully flushed the DNS Resolver Cache.
C:\WINDOWS\system32\drivers\etc>ping www.acme.com
Pinging www.acme.com [127.0.0.1] with 32 bytes of data:
Reply from 127.0.0.1: bytes=32 time=-1ms TTL=128

Ping statistics for 127.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = -1ms, Average = 1073741823ms
C:\WINDOWS\system32\drivers\etc>
```

UNIVERSIDAD DE SEVILLA

38

You may try immediately your new internet domain. Enter the following command:

> ping www.acme.com

You should get replies from “127.0.0.1”, which indicates that requests to “www.acme.com” are being routed to your computer. That’s enough so far.

**NOTE:** some systems are configured not to reply to ping requests. Please, consult your operating system’s and your firewall’s documentation if you can’t ping “www.acme.com”.



Let's now report on how to deploy your databases.

## What's deploying a database?

---

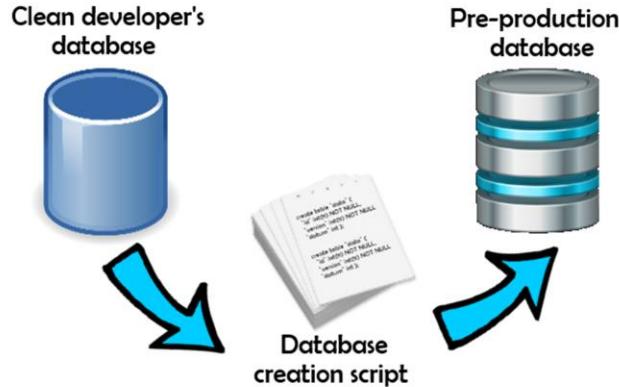


It's dumping your developer's database so that it can be replicated in the pre- or the production configuration

Deploying a database means dumping your developer's database so that it can be replicated in the pre- or the production configuration.

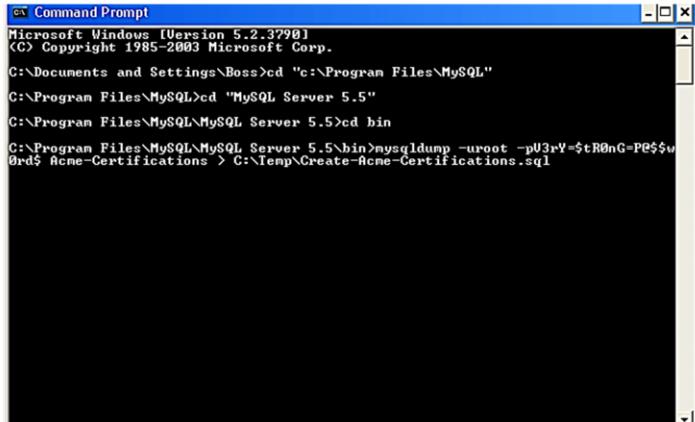
## The process

---



This is the process that we need to implement: first, you must make sure that you have a clean developer's database, that is, a database that doesn't have any inappropriate data; then, we'll use a tool to dump it, that is, to create a SQL script that will allow us to recreate it on another configuration. Note that the script must keep only the data that is absolutely necessary for your system to work well; this typically includes an administrator's account information and little else.

## Dumping a database



The screenshot shows a Windows Command Prompt window titled "Command Prompt". The window displays the following command sequence:

```
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.

C:\Documents and Settings\Boss>cd "c:\Program Files\MySQL"
C:\Program Files\MySQL>cd "MySQL Server 5.5"
C:\Program Files\MySQL\MySQL Server 5.5>cd bin
C:\Program Files\MySQL\MySQL Server 5.5\bin>mysqldump -uroot -pV3rY=$tR0nG=P@$$w0rd$ Acme-Certifications > C:\Temp\Create-Acme-Certifications.sql
```

Dumping a database isn't difficult at all: just open a user's shell on your developer's configuration and execute the following commands to change the current directory to MySQL's bin directory:

```
> cd "c:\Program Files\MySQL"
> cd "MySQL Server 5.5"
> cd "bin"
```

Now, execute the following command to dump your database:

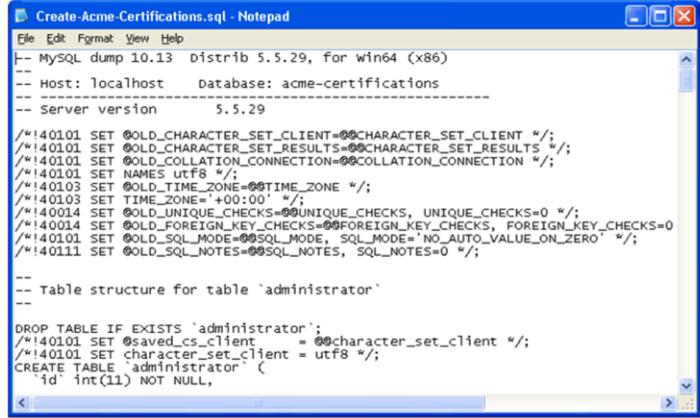
```
> mysqldump -u<root> -p<pass> <database-name> > <script-file.sql>
```

Where `<root>` and `<pass>` refer to the database root credentials, `<database-name>` to the name of the database you wish to dump, and `<script-file.sql>` to the file in which you wish to store the script. In this example, as usual, we're working with the database of project "Acme Certifications".

**NOTE:** please, note that you don't need to open an administrator's shell to execute the previous commands; note, however, that you need to log in to MySQL as the root.

**NOTE:** in the configuration that we provide, the root credentials are "`root"/"V3rY=$tR0nG=P@$$w0rd$"`".

# The database creation script



A screenshot of a Windows Notepad window titled "Create-Acme-Certifications.sql - Notepad". The window contains a large block of MySQL SQL code. The code includes various MySQL dump header information, such as the version (5.5.29), host (localhost), database (acme-certifications), and server version (5.5.29). It also contains numerous SET statements for character sets and collations, including utf8 and utf8mb4. The script then moves on to creating a table named "administrator" with an "id" column defined as int(11) NOT NULL.

```
-- MySQL dump 10.13  Distrib 5.5.29, for win64 (x86)
-- Host: localhost      Database: acme-certifications
-- 
-- Server version      5.5.29

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

-- 
-- Table structure for table `administrator`
--

DROP TABLE IF EXISTS `administrator`;
/*!40101 SET @saved_cs_client     = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `administrator` (
  `id` int(11) NOT NULL,
```

Please, take a look at the script. It should look more or less like in this slide. It includes a lot of SQL statements to create the data tables, the attributes, the foreign keys, and so on.

## This is a good question!

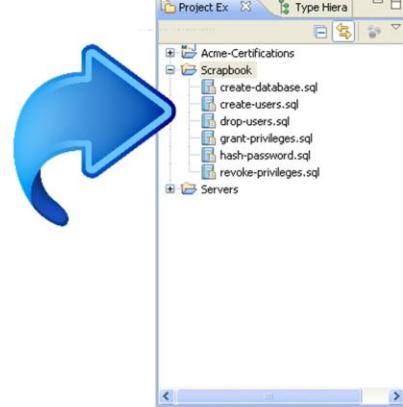
---



Where are the statements to create the database, the users, or to assign privileges?

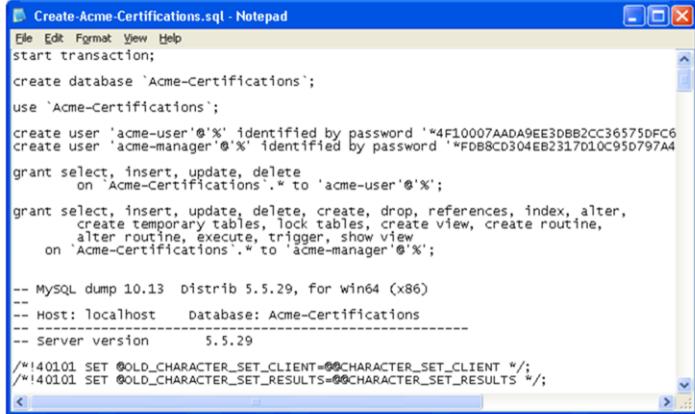
This script is incomplete, isn't it? This is a good question! If you examine the previous script carefully, you'll easily discover that it's incomplete: it includes every sentence that is required to create the structure that is required to persist your objects, but it doesn't include a single sentence to create the database itself, the users, or to assign privileges to them.

## Here are a few useful scripts!



In our workspace template, there's a project called "scrapbook" that provides SQL templates to create databases, to create users, to drop them, to grant privileges to them, to revoke them, or to hash their passwords. We're pretty sure that you're familiar with these scripts since we've used them a lot of times in multiple problems in the previous lessons.

# Completing the script



```
CREATE-Acme-Certifications.sql - Notepad
File Edit Format View Help
start transaction;
create database `Acme-Certifications`;
use `Acme-Certifications`;
create user 'acme-user'@'%' identified by password '4F10007AADA9EE3DBB2CC36575DFC6';
create user 'acme-manager'@'%' identified by password 'FDB8CD304EB2317D10C95D797A4';
grant select, insert, update, delete
on `Acme-Certifications`.* to 'acme-user'@'%';
grant select, insert, update, delete, create, drop, references, index, alter,
create temporary tables, lock tables, create view, create routine,
alter routine, execute, trigger, show view
on `Acme-Certifications`.* to 'acme-manager'@'%';
-- MySQL dump 10.13 Distrib 5.5.29, for win64 (x86)
-- Host: localhost      Database: Acme-Certifications
-- Server version      5.5.29
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
```

You have to edit the script that is generated by “mysqldump” to add the sentences required to create the users, the database, and to grant the users the appropriate privileges on the database. The result should be more or less like in this slide. Please, note that the initial sentence’s “start transaction”, so that no changes are committed to the database if a problem occurs; it’s not shown in the slide, but you need to add a “commit” sentence at the end of the file to close the transaction.

**NOTE:** the MySQL version that we’re using doesn’t support transactions regarding the Schema Manipulation Language; that is, if a problem occurs while loading a table with data, then the transaction is aborted, which means that the data entered is erased, but not the schema of the database. Please, keep however the “start transaction; ... commit;” block.

## This is a good question, too!

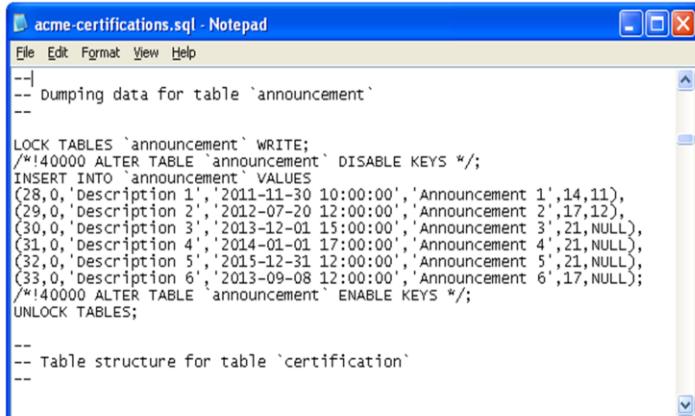
---



The script creates  
spurious data,  
doesn't it?

The script creates spurious data, doesn't it? That is a good question, too! If you examine the previous script carefully, you'll easily discover that it includes a lot of insert statements

## Purge the script



```
File Edit Format View Help
--|
-- Dumping data for table `announcement`
--

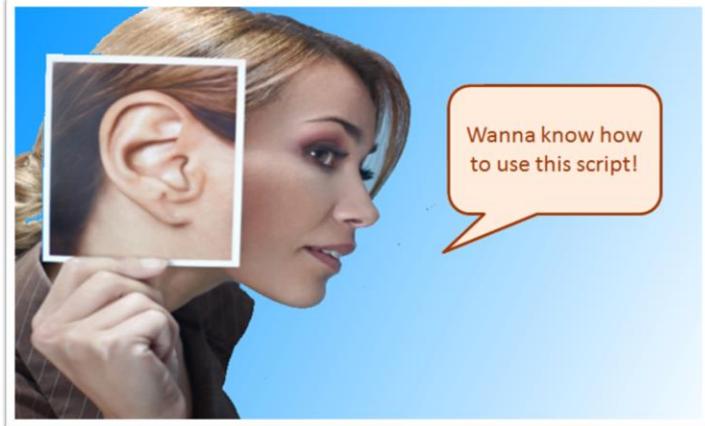
LOCK TABLES `announcement` WRITE;
/*!40000 ALTER TABLE `announcement` DISABLE KEYS */;
INSERT INTO `announcement` VALUES
(28,0,'Description 1','2011-11-30 10:00:00','Announcement 1',14,11),
(29,0,'Description 2','2012-07-20 12:00:00','Announcement 2',17,12),
(30,0,'Description 3','2013-12-01 15:00:00','Announcement 3',21,NULL),
(31,0,'Description 4','2014-01-01 17:00:00','Announcement 4',21,NULL),
(32,0,'Description 5','2015-12-31 12:00:00','Announcement 5',21,NULL),
(33,0,'Description 6','2013-09-08 12:00:00','Announcement 6',17,NULL);
/*!40000 ALTER TABLE `announcement` ENABLE KEYS */;
UNLOCK TABLES;

--|
-- Table structure for table `certification`
--|
```

Very likely, most of these data are just sample data that don't need to be stored in the pre- or the production configuration. It's important that you purge these data, that is, that you remove the statements that insert them into the database. You must keep only data that is actually necessary in the pre- or the production configuration, e.g., the administrator account, information about your customer's company, a list of provinces and zip codes, and the like.

## Nice to hear that!

---

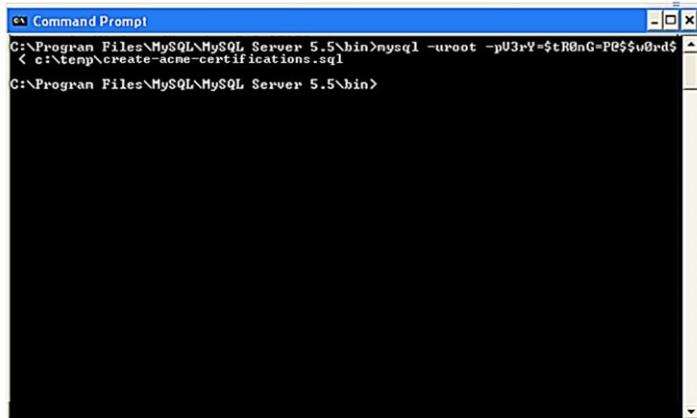


UNIVERSIDAD DE SEVILLA

49

Creating the script wasn't very difficult, was it? It's now time to learn how you can use it in your pre-production configuration.

## Executing the script



You just need to open an administrator's shell on your pre-production configuration, change the working directory to MySQL's bin directory and execute the following command:

```
> mysql -u<root> -p<pass> < <script-file.sql>
```

Where <root> and <pass> refer to the database root credentials and <script-file.sql> refers to the file that provides the database creation script. If everything goes well, then you shouldn't see any messages on the screen; if there are any problems, please, interpret the error messages and correct them.

**NOTE:** in the configuration that we provide, the root credentials are "root"/"V3rY=\$tR0nG=P@\$\$w0rd\$".

## A few hints

---

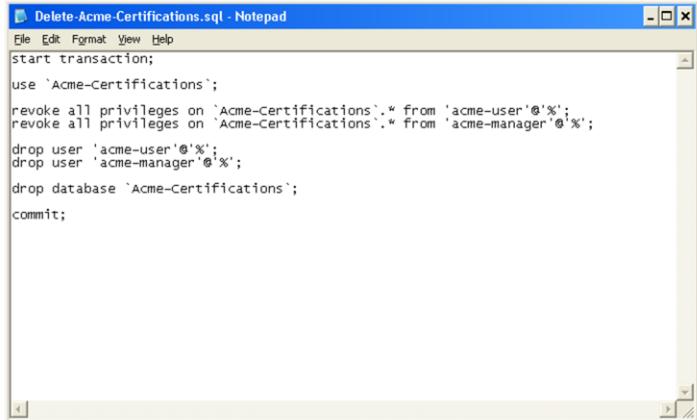


UNIVERSIDAD DE SEVILLA

51

Before concluding this section, we'd like to provide you with a few hints. Not having the MySQL Workbench in the pre-production configuration makes everything a little more complex since you need to use textual commands all the time; but don't worry: you just need to know a few common commands to be productive with the MySQL command-line interface in your pre-production configuration.

## Hint #1: how to drop everything?

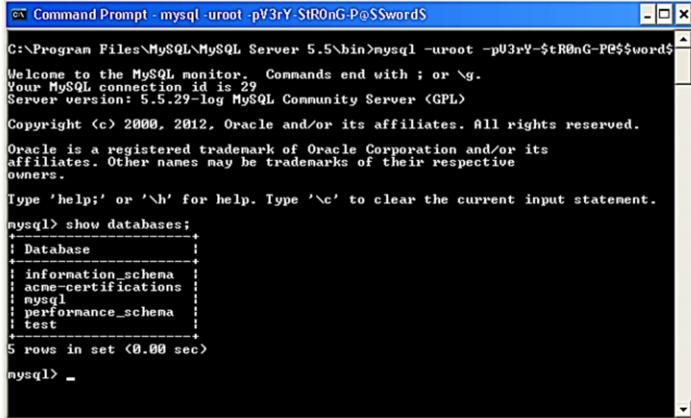


```
File Edit Format View Help
Delete-Acme-Certifications.sql - Notepad
start transaction;
use `Acme-Certifications`;
revoke all privileges on `Acme-Certifications`.* from 'acme-user'@'%';
revoke all privileges on `Acme-Certifications`.* from 'acme-manager'@'%';
drop user 'acme-user'@'%';
drop user 'acme-manager'@'%';
drop database `Acme-Certifications`;
commit;
```

The first hint's that you should also create a script to completely remove your database from the server. It's very common that you make mistakes and that you have to deploy your database several times before you command the procedure. Every time you do that, you need to remove your database from the server. In this slide, we show a typical script to remove every trace of your database from a configuration: first you have to revoke your users' privileges, then you have to drop them, and, finally, you have to drop the database. Note that the sentences are enclosed into a "start transaction; ... commit;" block so as to execute them within the context of a transaction.

**NOTE:** due to a problem with the MySQL version that we're using, these sentences are not actually executed in a transaction. Please, keep however the "start transaction; ... commit;" block.

## Hint #2: listing databases



```
mysql -uroot -pV3rY-$tR0nG-P@$$w0rd$S
C:\Program Files\MySQL\MySQL Server 5.5\bin>mysql -uroot -pV3rY-$tR0nG-P@$$w0rd$S
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 29
Server version: 5.5.29-log MySQL Community Server (GPL)

Copyright (c) 2000, 2012, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| acne-certifications |
| mysql              |
| performance_schema |
| test               |
+--------------------+
5 rows in set (0.00 sec)

mysql>
```

UNIVERSIDAD DE SEVILLA

53

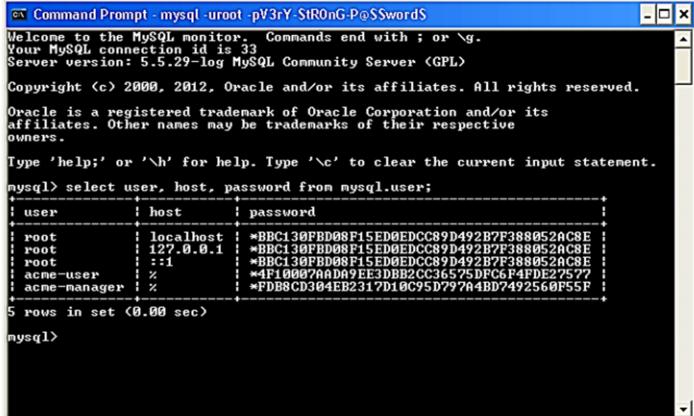
The second hint's regarding the command you need to know in order to list the databases that are available at your server. First you have to open a MySQL shell by means of the following command:

```
> mysql -u<root> -p<pass>
```

Where <root> and <pass> refer to the database root credentials. There are a few interesting MySQL commands that you can run in this shell. The first one is “show databases”, which reports on the names of the databases that are available at the server, including the system databases “information\_schema”, “mysql”, “performance\_schema”, and “test”.

**NOTE:** in the configuration that we provide, the root credentials are “root”/”V3rY=\$tR0nG=P@\$\$w0rd\$”.

## Hint #3: listing users



```
mysql> select user, host, password from mysql.user;
+-----+-----+-----+
| user | host | password |
+-----+-----+-----+
| root | localhost | *BBC130FBD08F15ED0EDCC89D492B7F388952AC8E |
| root | 127.0.0.1 | *BBC130FBD08F15ED0EDCC89D492B7F388952AC8E |
| root | ::1 | *BBC130FBD08F15ED0EDCC89D492B7F388952AC8E |
| acne-user | % | *4F100079ADAE9E3DBB2CC36575DFC6F4FDE2757? |
| acne-manager | % | *FDB8CD304EB2317D10C95D797A4BD7492560F55F |
+-----+-----+-----+
5 rows in set (0.00 sec)

mysql>
```

UNIVERSIDAD DE SEVILLA

54

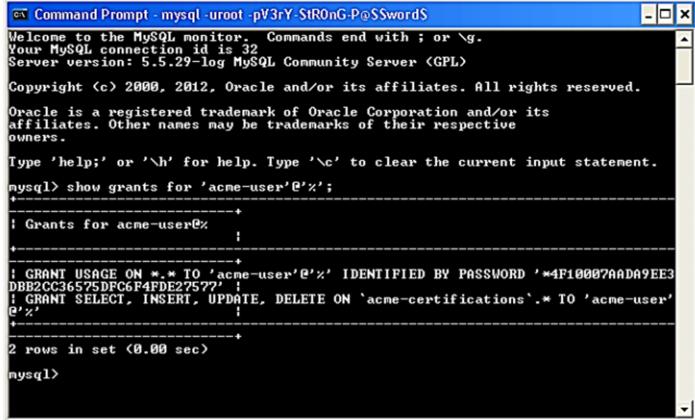
You can also use the following command to list the users that are registered in the server:

```
> select user, host, password from mysql.user;
```

It lists their usernames, the computers from which they are allowed to connect to the server, and the hashes of their passwords. Please, recall that no serious application should store passwords in clear text.

**NOTE:** In MySQL's parlance, a host is a computer and "%" means any host.

## Hint #4: listing privileges



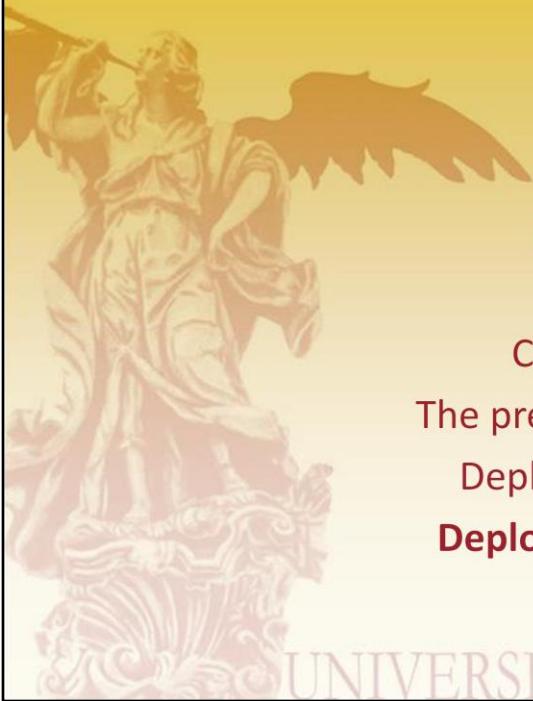
```
mysql> show grants for 'acme-user'@'%';
+-----+
| Grants for acme-user@% |
+-----+
| GRANT USAGE ON *.* TO 'acme-user'@'%' IDENTIFIED BY PASSWORD '*4F10007AADAA9EE3DBB2CC36575DFCF4FDE275??'; |
| GRANT SELECT, INSERT, UPDATE, DELETE ON `acme-certifications`.* TO 'acme-user'@'%' |
+-----+
2 rows in set (0.00 sec)

mysql>
```

Finally, you can use the following command to list the privileges (aka grants) that a user has:

> `show grants for <username>@<host>`

where `<username>@<host>` refers to the username and the host in which we're interested.



## Roadmap

Configurations

The pre-pro configuration

Deploying databases

**Deploying applications**

UNIVERSIDAD DE SEVILLA

Finally, it's time to learn how to deploy your application.

## What's deploying an application?

---

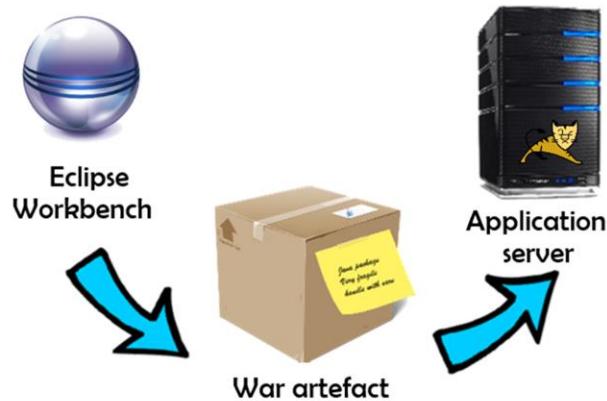


It's packaging your application in a war artefact so that it can be run on a pre- or a production configuration

Deploying an application means packaging it in a war artefact so that it can be run on a pre- or a production configuration. A war artefact packages images, scripts, styles, views, classes (the binary code, not the source code), configuration files, and the components on which your system relies. Simply put, it includes almost everything in your project except for the source code of your Java classes and the database.

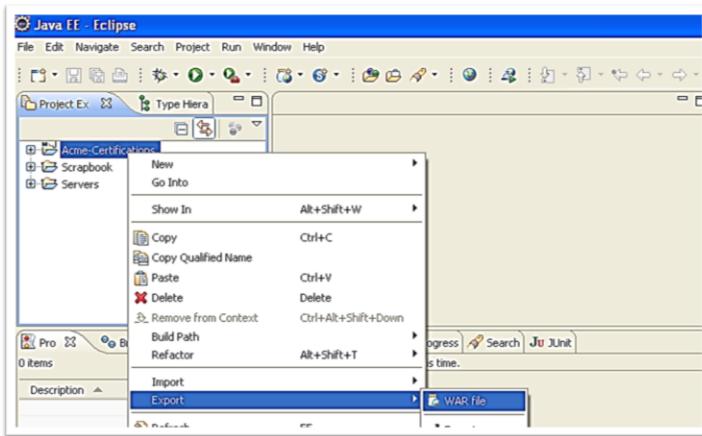
## The process

---



This is the process that we're going to follow: first, we'll use the Eclipse workbench on our developer's configuration to create a war artefact and we'll then upload it to Tomcat using Tomcat's management application.

## Creating a war artefact (I)

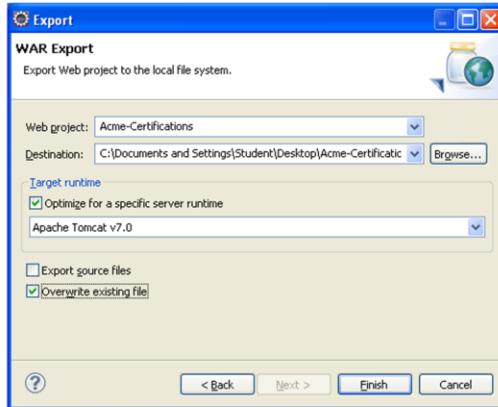


UNIVERSIDAD DE SEVILLA

59

Creating a war artefact is the simplest task! Just right click your project on the project explorer, then select “Export”, and then select “WAR file”.

## Creating a war artefact (II)



Eclipse will show this dialog in which you only need to select the destination file and click “Finish” to get your war artefact. We however recommend that you should also check the “Optimise for a specific server runtime” option, since this instructs Eclipse to generate a war artefact that is specifically tailored to our specific application server, if possible. Note that you can also export your source code into the war artefact but, as we mentioned earlier, this doesn’t usually make a lot of sense. So you’d better leave this option unchecked.

## Enter the app manager



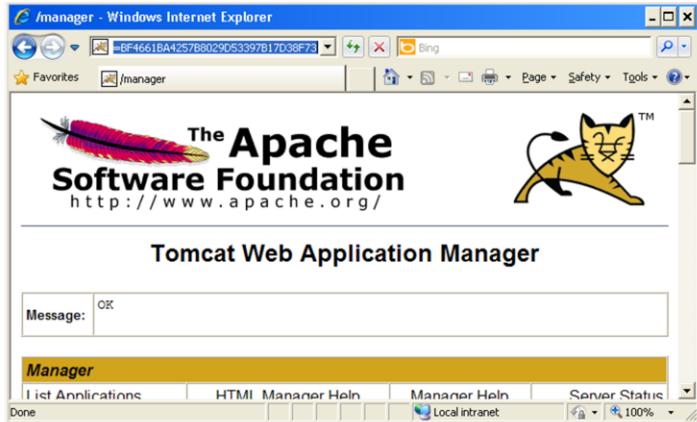
UNIVERSIDAD DE SEVILLA

61

To start configuring the server, open a browser and make it for "http://localhost/manager". "manager" is a pre-defined application context in which Tomcat's service installs a simple application manager that will help us deploy our sample system. An authentication window will popup. Just key in Tomcat's administrator's credentials and press the "OK" button.

**NOTE:** in the virtual machine that we provide, the default credentials are "admin"/"T0mC@t=Adm1n1\$trat0R".

## This is what it looks like



UNIVERSIDAD DE SEVILLA

62

This is what the application manager looks like. It's quite a simple web application.

## Scroll to see the applications

The screenshot shows the 'Applications' section of the Tomcat Manager interface. It lists two application contexts: 'Welcome to Tomcat' and 'Tomcat Documentation'. Both contexts are running, have 0 sessions, and were last deployed at least 30 minutes ago. The 'Welcome to Tomcat' context is highlighted in green.

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	<button>Start</button> <button>Stop</button> <button>Reload</button> <button>Undeploy</button>  <button>Expire sessions</button> with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	<button>Start</button> <button>Stop</button> <button>Reload</button> <button>Undeploy</button>  <button>Expire sessions</button> with idle ≥ 30 minutes

UNIVERSIDAD DE SEVILLA

63

Scroll a little down to see the applications that are installed by default: a welcome application, documentation, examples, and the manager itself.

**NOTE:** the vocabulary is a little confusing here since “application contexts” are referred to as “paths” in the application manager. (See the header of the first column).

## Tidy your server up!

The screenshot shows the Tomcat Manager interface in a Windows Internet Explorer window. The 'Applications' section displays a single entry:

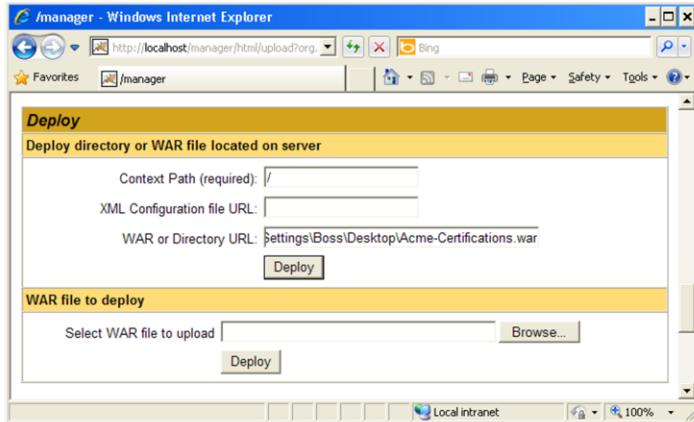
Path	Version	Display Name	Running	Sessions	Commands
/manager	None specified	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

The 'Deploy' section contains a form for deploying a WAR file:

Deploy directory or WAR file located on server  
Context Path (required):

The first thing you must do is to tidy your server a little up. Please, undeploy every default application, but the application manager.

# Deploy your war artefact



UNIVERSIDAD DE SEVILLA

65

We're ready to deploy our war artefact. Scroll down until you get to a section entitled "Deploy". There are two subsections, and you have to use the first one: "Deploy directory or war file located on server". You just need to provide an application context (a path in the manager's parlance) and the path to the war artefact in your file system; please, ignore the XML configuration file. The application context should be "/", since we wish our sample application to be available from the root context of our web domain. Note that the subsection entitled "WAR file to deploy" also allows you to deploy a war artefact and it offers a typical "Choose file" button by means of which you can easily locate your war artefact in the file system; unfortunately, if you deploy your artefacts using this choice, you won't be able to indicate the application context to which you wish to deploy them. They'll be deployed to a context whose name's the name of your war artefact.

# Is everything OK?

The screenshot shows the 'Applications' section of the Tomcat Manager interface. It lists two applications:

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Acme-Certifications	true	0	<a href="#">Start</a> <a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Undeploy</a>  <a href="#">Expire sessions</a> with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	1	<a href="#">Start</a> <a href="#">Stop</a> <a href="#">Reload</a> <a href="#">Undeploy</a>  <a href="#">Expire sessions</a> with idle ≥ 30 minutes

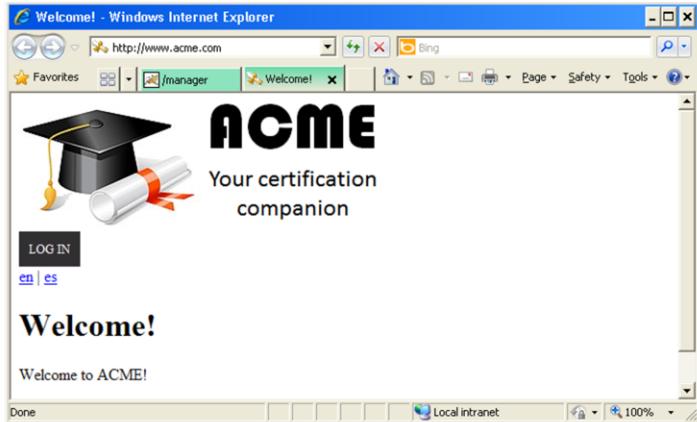
The application at path '/' has 0 sessions and the application at path '/manager' has 1 session.

UNIVERSIDAD DE SEVILLA

66

If everything's OK, then you should see your application in the “Applications” section. There you can stop, reload, start, or undeploy it whenever necessary.

Try <http://www.acme.com> now!

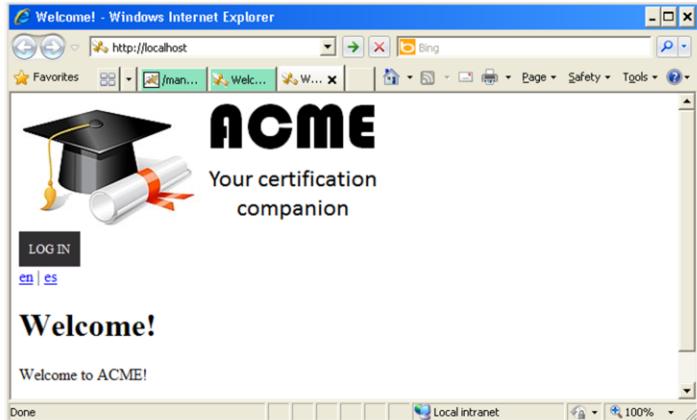


UNIVERSIDAD DE SEVILLA

67

To check that the application works, just make your browser for "http://www.acme.com" and your application should show up in a few seconds. Note that "www.acme.com" works locally because we previously hacked our pre-production configuration so that it overrides the public DNS record for this internet domain; on other computers, "www.acme.com" will show you customer's current web site.

## Try <http://localhost>, too



UNIVERSIDAD DE SEVILLA

68

Try making your browser for “<http://localhost>”, too. The result should be the same. Please, recall that both “localhost” and “[www.acme.com](http://www.acme.com)” redirect to your computer’s IP address.

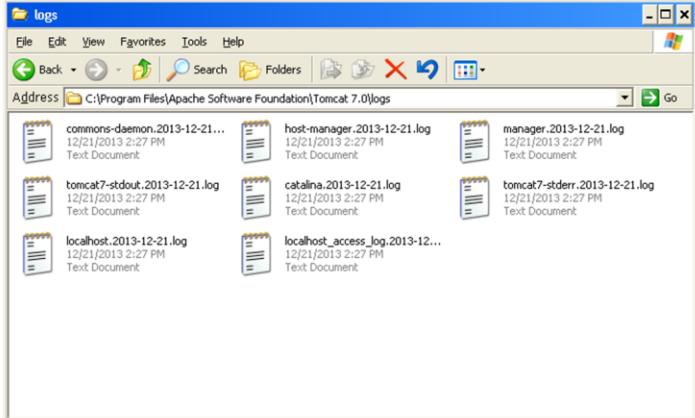
## Ooohhh, sorry!

---



Doesn't it work? We're sorry. In that case you need to check every step and repeat them until you find the problem that's preventing you from deploying and using your web information system.

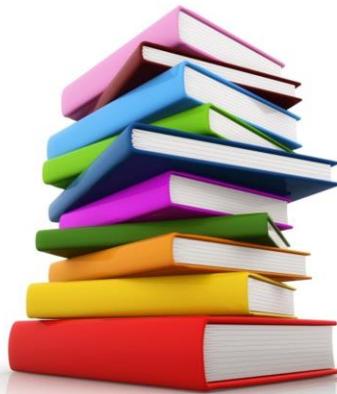
## A good hint: check Tomcat's logs



In case of trouble, there's very little you can do to debug your system. We suggest that you should take a look at Tomcat's logs. Please, search for the log files that correspond to the day on which you're working and analyse them. The logs typically show exceptions that may give you a hint regarding what the problem is.

# Bibliography

---



UNIVERSIDAD DE SEVILLA

71

Our recommendation is that you should take a look at the following books regarding MySQL and Apache Tomcat:

MySQL, 5th edition

P. Dubois

Addison-Wesley, 2013

Apache Tomcat 7 essentials

Tanuj Khare

Packt Publishing, 2012

This bibliography is available in electronic format for our students at the USE's virtual library. If you don't know how to have access to the USE's virtual library, please, ask our librarians for help.

Time for questions, please

---



Time for questions, please. Note that we won't update the slides with the questions that are posed in the lectures. Please, attend them and take notes.

## The next lecture



- We need (coerced) volunteers
- Volunteer collaboration is strongly advised
- Produce a solution and a presentation
- Rehearse your presentation at home!
- Each presentation is allocated  $100/N$  min
- Presentations must account for feedback

The next lecture is a problem lecture. We need some volunteers, who are expected to collaborate to produce a solution and a presentation. Please, rehearse your presentation at home taking into account that you have up to  $100/N$  minutes per problem, including feedback, where  $N$  denotes the number of problems.



Thanks for attending this lecture! See you next day!