

13

*Pocket Edition 2015*

the  
original  
**Hacker**



creado por EUGENIA BAHIT

**jugando con la inteligencia**

STORED PROCEDURES

# Mantenimiento de la base de datos del sistema

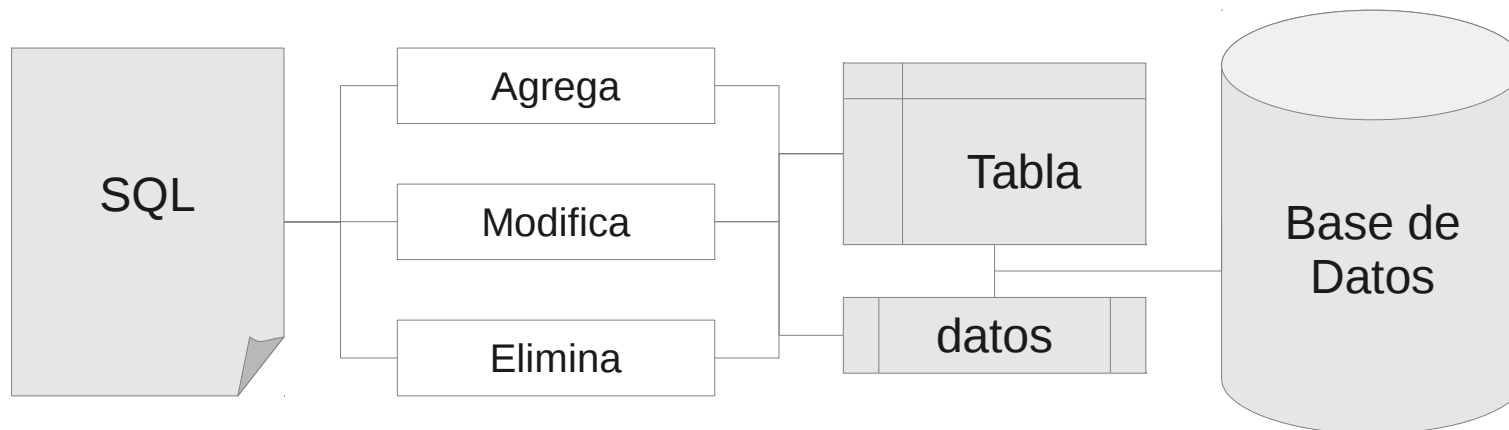
---

*En el desarrollo iterativo e incremental, la base de datos suele modificarse en cada sprint y esto conlleva errores de implementación, sobre todo, en el propio proceso de desarrollo*

Para los desarrolladores, no basta con un simple *pull*.

---

*Los cambios en la base de datos **siempre** deben verse reflejados en un **único** archivo SQL*



El *script* SQL debe correrse siempre luego de cada pull

```
mysql -u root -p DATABASENAME < path/to/myapp/myapp.sql
```

Tener en cuenta que cualquier error arrojado por MySQL tras la ejecución del *script*, implicará que no todas las sentencias se habrán ejecutado

## Utilizar las cláusulas `IF NOT EXISTS` e `IF EXISTS` para crear y eliminar tablas respectivamente

```
CREATE TABLE IF NOT EXISTS table_name (  
    ...  
)
```

```
DROP TABLE IF EXISTS table_name (  
    ...  
)
```

Nunca hacer un `DROP TABLE` “preventivo” antes de crear la tabla. Si se modificará, utilizar `ALTER TABLE` (se explica más adelante)

## Utilizar la cláusula ALTER TABLE dentro de un procedimiento almacenado (*stored procedure*) para evitar fallos

```
DELIMITER //

DROP PROCEDURE IF EXISTS nombre_procedimiento //
CREATE PROCEDURE nombre_procedimiento()
  BEGIN
    CONDICION
    THEN
      ALTER TABLE...;
    END IF;
  END //

DELIMITER ;

CALL nombre_procedimiento();
```

## Utilizar IF NOT EXISTS como condición del *stored procedure* para agregar campos

```
DELIMITER //

DROP PROCEDURE IF EXISTS agregar_columna //
CREATE PROCEDURE agregar_columna()
  BEGIN
    IF NOT EXISTS (
      SELECT NULL
      FROM INFORMATION_SCHEMA.COLUMNS
      WHERE table_name = 'tabla'
      AND table_schema = DATABASE()
      AND column_name = 'columna_nueva'
    ) THEN
      ALTER TABLE tabla ADD COLUMN columna_nueva INT(3);
    END IF;
  END //

DELIMITER ;

CALL agregar_columna();
```

## Utilizar IF EXISTS como condición del *stored procedure* para eliminar campos y/o modificarlos

```
DELIMITER //

DROP PROCEDURE IF EXISTS eliminar_columna //
CREATE PROCEDURE eliminar_columna()
  BEGIN
    IF EXISTS (
      SELECT NULL
      FROM INFORMATION_SCHEMA.COLUMNS
      WHERE table_name = 'tabla'
      AND table_schema = DATABASE()
      AND column_name = 'columna'
    ) THEN
      ALTER TABLE tabla DROP COLUMN columna;
    END IF;
  END //

DELIMITER ;

CALL eliminar_columna();
```



## Error «Thread stack overrun» al ejecutar un procedimiento almacenado **SOLUCIÓN**

Localizar el archivo de configuración de MySQL (generalmente llamado my.cnf) y comentar la variable thread\_stack colocando un punto y coma delante:

```
; thread_stack = 128K
```

En Debian GNU/Linux y distribuciones derivadas (como Ubuntu), este archivo podrá encontrarse en:

```
/etc/mysql/my.cnf
```

Reiniciar MySQL antes de volver a intentarlo:

```
Service mysql restart
```

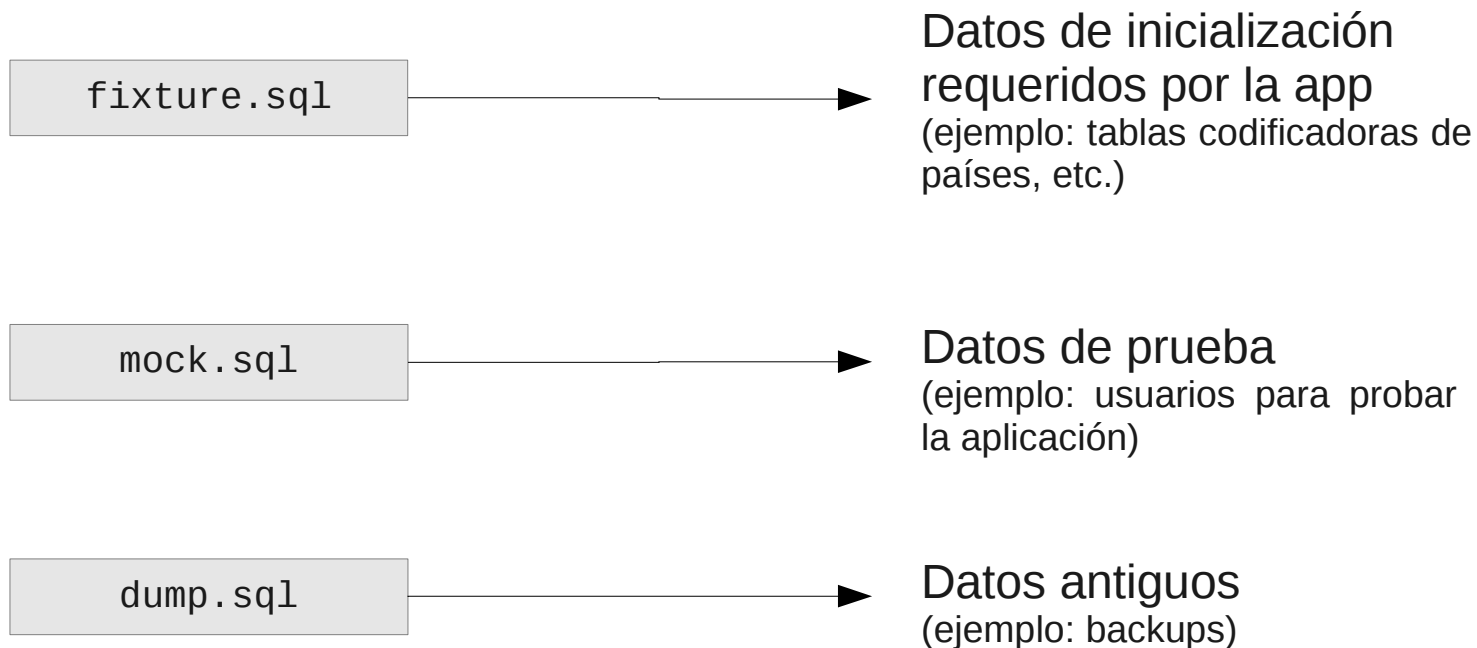
## Utilizar la cláusula IGNORE en todo INSERT que se haga por defecto

```
INSERT IGNORE INTO tabla
  (campo_id_pk, campo2, campo3)
VALUES
  (1, 'dato 2.1', 'dato 3.1'),
  (2, 'dato 2.2', 'dato 3.2'),
  (3, 'dato 2.3', 'dato 3.3')
;
```

*Utilizar IGNORE y agregar las ID (claves primarias) evitará registros duplicados y dependencias insatisfechas*

Procurar colocar todos los “INSERT” al final del *script* o en su defecto, en un archivo `fixture.sql`

## Sobre la inserción de datos: archivos SQL de destino



¿Te gustaría aprender más sobre buenas prácticas de programación y bases de datos?

Documentación oficial de MySQL:

<http://dev.mysql.com/doc/>

Mis papers en PDF sobre MySQL:

<http://library.originalhacker.org/search/mysql>

Ayuda y soporte gratuito:

<http://forums.mysql.com>

<http://stackoverflow.org>

**Curso profesional de Ingeniería de Software (a distancia):**

<http://cursos.eugeniabahit.com/curso-3>

# Eugenia Bahit

GLAMP Hacker & programadora eXtrema



**Especializada en seguridad informática y desarrollo de Software mediante Ingeniería Inversa de código en Python y PHP.**

Miembro de **Free Software Foundation** presidida por *Richard Stallman* (creador del proyecto GNU y el Software Libre) y **The Linux Foundation** presidida por *Linus Torvalds* (creador del kernel Linux).

Creadora de los proyectos **Europio Engine, Jack The Stripper, python-printr y Enhancement CLI for PHP.**

Fundadora de las revistas **The Original Hacker y Hackers & Developers Magazine.**

Dedicada actualmente a la docencia e investigación y a la **Formación Profesional de Hackers y Programadores**

WEB PERSONAL:  
[www.eugeniabahit.com](http://www.eugeniabahit.com)

TWITTER:  
[@eugeniabahit](https://twitter.com/eugeniabahit)

REPOSITORIOS:  
<http://repo.eugeniabahit.com>

# Licencia Libre

## Creative Commons Atribución CompartirIgual

---

### Eres libre de:

#### Compartir

copiar y distribuir este documento por cualquier medio y formato

#### Modificar

crear documentos derivados y distribuirlos

### Pero:

#### Hazlo con respeto hacia al autor

Manteniendo la leyenda de copyright que ves al pie de cada página (cuando hagas cambios, indica que eres el responsable de esos cambios)

#### No quites la libertad a otros

Cuando distribuyas este documento y/o tus modificaciones, deberás mantener la licencia original