

Manual por:

Criar Web, manuais e recursos para desenvolvimento web

Versão on-line:<http://www.criarweb.com/manuais/7>

Instalação de MySQL em Windows

Um dos pontos mais fortes das páginas em PHP é a possibilidade de usar bases de dados mediante funções de destacada simplicidade e potência. Estas bases de dados podem servir ao nosso site para armazenar conteúdos de uma forma sistemática que nos permita classificá-los, procurá-los e editá-los rápida e facilmente.

Uma base de dados é um conjunto de tabelas nas quais armazenamos diferentes registos (artigos de uma loja virtual, provedores ou clientes de uma empresa, filmes em exibição de um cinema...). Estes registos são catalogados numa função de diferentes parâmetros que os caracterizam e que tem uma utilidade a hora de classifica-los. Assim, por exemplo, os artigos de uma loja virtual poderiam catalogar-se a partir de diferentes campos como pode ser o número de referência, nome do artigo, descrição, preço, provedor...

A base de dados mais difundida com o sistema UNIX-Apache é sem duvida MySQL. Tal e como acontece com o Apache, está disponível uma versão para Windows e pode ser [obtida](#) gratuitamente.

Pô-lo a funcionar não tem muita dificuldade. Uma vez instalado o programa podemos executar as nossas ordens em modo MS-DOS. Para isso, abrimos uma janela MS-DOS e colocamo-nos no directório bin de mysql. Neste directório encontram-se os ficheiros executáveis. Aqui teremos de encontrar um ficheiro chamado mysqld. No caso da versão mais actual durante a redacção deste artigo, este ficheiro chama-se mysqld-shareware. Uma vez executado este ficheiro podemos executar o seguinte: mysql

Uma vez chegados a este ponto veremos como uma mensagem de bem-vinda aparece no nosso ecrã. Neste momento encontramos-nos dentro da base de dados. A partir de aí podemos realizar todo tipo de operações por comandos SQL.

Não vamos entrar numa explicação pormenorizada do funcionamento dessa base de dados pois isso dava-nos para um manual inteiro. Como resumo, aqui propomos as operações mais básicas:

Instrucción	Descripción
Show databases;	Mostra o conjunto da base de dados presente no servidor
Use nome_da_base	Determina a base de dados sobre a qual vamos trabalhar
Create Database nome_da_base;	Cria uma nova bd com o nome especificado
Drop Database nome_da_base;	Elimina a base de dados com o nome especificado
Show tables;	Mostra as tabelas presentes na base de dados actual
Describe nome_da_tabela;	Descreve os campos que compõem a tabela
Drop Table nome_da_tabela;	Apaga a tabela da base de dados
Load Data Local Infile "ficheiro.txt" Into Table nome_da_tabela;	Cria os registos da tabela a partir de um ficheiro de texto no qual separamos por tabulações todos os campos de um mesmo registo
Quit	Sair de MySQL

Para evitarmos ter de editar as nossas tabelas directamente sobre os ficheiros de texto, pode ser muito prático usar qualquer outra base de dados com um editor e exportar a seguir a tabela num ficheiro de texto configurado para deixar tabulações entre cada campo. Isto é possível em Access, por exemplo, clicando com o botão direito sobre a tabela que queremos converter e escolhendo a opção exportar. Uma janela de dialogo aparecerá na qual escolheremos guardar o ficheiro em tipo texto. O passo

seguinte será escolher um formato delimitado por tabulações sem qualificador de texto.

Outra possibilidade que pode ser muito prática e que nos evita trabalhar continuamente com comandos do estilo aos do tempo da carochinha é servir-se de programas em PHP ou Perl já existentes e que se podem obter na rede. O mais popular sem duvida é o phpMyAdmin. Estes tipos de scripts são executados desde um navegador e podem, por tanto, albergados no nosso servidor ou empregues em local para, a partir deles, administrar MySQL de uma forma menos difícil.

Do mesmo modo, dentro do directório bin de MySQL, podemos encontrar uma pequena aplicação chamada MySqlManager. Trata-se de uma interfaz windows, mais agradável a vista e ao uso do que a que obtemos executando o ficheiro mysql. Neste caso, as sentenças SQL devem realizar-se sem o ponto e vírgula final.

Informe de **Miguel Angel Alvarez**
Mail: miguel@desarrolloweb.com

Tipos de dados de MySQL

Depois da fase de design da base de dados, e uma vez que se passou a tabelas, é necessário criar as tabelas correspondentes dentro da base de dados. Para cada campo de cada uma das tabelas, é necessário determinar o tipo de dados que contem, para poder ajustar a estrutura da base de dados, e conseguir um armazenamento com a menor utilização de espaço. Este artigo descreve cada um dos tipos de dados que se podem ter num campo MySQL, para a versão 4.xx.xx.

Os tipos de dados que pode ter um campo, podem-se agrupar em três grandes grupos:

1. Tipos numéricos
2. Tipos de Data
3. Tipos de Cadeia

1 Tipos numéricos:

Existem tipos de dados numéricos, que se podem dividir em dois grandes grupos, os que estão em vírgula flutuante (com decimais) e os que não.

TinyInt: é um número inteiro com ou sem signo. Com signo a margem de valores válidos é desde -128 até 127. Sem signo, a margem de valores é de 0 até 255

Bit ou Bool: um número inteiro que pode ser 0 ou 1.

SmallInt: número inteiro com ou sem signo. Com signo a margem de valores válidos é desde -32768 até 32767. Sem signo, a margem de valores é de 0 até 65535.

MediumInt: número inteiro com ou sem signo. Com signo a margem de valores válidos é desde -8.388.608 até 8.388.607. Sem signo, a margem de valores é de 0 até 16777215.

Integer, Int: número inteiro com ou sem signo. Com signo a margem de valores válidos é desde -2147483648 até 2147483647. Sem signo, a margem de valores é de 0 até 429.496.295

BigInt: número inteiro com ou sem signo. Com signo a margem de valores válidos é desde -9.223.372.036.854.775.808 até 9.223.372.036.854.775.807. Sem signo, a margem de valores é de 0 até 18.446.744.073.709.551.615.

Float: número pequeno em vírgula flutuante de precisão simples. Os valores válidos vão desde -3.402823466E+38 até -1.175494351E-38, 0 até 175494351E-38 até 3.402823466E+38.

xReal, Double: número em vírgula flutuante de dupla precisão. Os valores permitidos vão desde -1.7976931348623157E+308 até -2.2250738585072014E-308, 0 e desde 2.2250738585072014E-308 até 1.7976931348623157E+308

Decimal, Dec, Numeric: Número em vírgula flutuante desempacotado. O número armazena-se como

uma cadeia.

Tipo de Campo	Tamanho de Armazenamento
TINYINT	1 byte
SMALLINT	2 bytes
MEDIUMINT	3 bytes
INT	4 bytes
INTEGER	4 bytes
BIGINT	8 bytes
FLOAT(X)	4 ou 8 bytes
FLOAT	4 bytes
DOUBLE	8 bytes
DOUBLE PRECISION	8 bytes
REAL	8 bytes
DECIMAL(M,D)	M+2 bytes se D > 0, M+1 bytes se D = 0
NUMERIC(M,D)	M+2 bytes se D > 0, M+1 bytes se D = 0

2 Tipos data:

Na hora de armazenar datas, há que ter em conta que MySQL não verifica de uma maneira estrita se uma data é válida ou não. Simplesmente comprova que o mês está compreendido entre 0 e 12 e que o dia está compreendido entre 0 e 31.

Date: tipo data, armazena uma data. A margem de valores vai desde o 1 de Janeiro de 1001 ao 31 de dezembro de 9999. O formato de armazenamento é de ano-mes-dia.

DateTime: Combinação de data e hora. A margem de valores vai desde o 1 ed Janeiro de 1001 às 0 horas, 0 minutos e 0 segundos ao 31 de Dezembro de 9999 às 23 horas, 59 minutos e 59 segundos. O formato de armazenamento é de ano-mes-dia horas:minutos:segundos

TimeStamp: Combinação de data e hora. A margem vai desde o 1 de Janeiro de 1970 ao ano 2037. O formato de armazenamento depende do tamanho do campo:

Tamanho	Formato
14	AnoMesDiaHoraMinutoSegundo aaaammddhhmmss
12	AnoMesDiaHoraMinutoSegundo aammddhhmmss
8	AnoMesDia aaaammdd
6	AnoMesDia aammdd
4	AnoMes aamm
2	Ano aa

Time: armazena uma hora. A margem de horas vai desde -838 horas, 59 minutos e 59 segundos. O formato de armazenamento é 'HH:MM:SS'.

Year: armazena um ano. A margem de valores permitidos vai desde o ano 1901 ao ano 2155. O campo pode ter tamanho dois ou tamanho 4 dependendo de se queremos armazenar o ano com dois ou quatro algarismos.

Tipo de Campo	Tamanho de Armazenamento
DATE	3 bytes
DATETIME	8 bytes
TIMESTAMP	4 bytes
TIME	3 bytes
YEAR	1 byte

3 Tipos de cadeia:

Char(n): armazena uma cadeia de longitude fixa. A cadeia poderá conter desde 0 até 255 caracteres.

VarChar(n): armazena uma cadeia de longitude variável. A cadeia poderá conter desde 0 até 255 caracteres. Dentro dos tipos de cadeia pode-se distinguir dois subtipos, os tipo Text e os tipo Blob (Binary Large Object) A diferença entre um tipo e outro é o tratamento que recebem na hora de ordená-los e compará-los. No tipo text ordena-se sem ter importância as maiúsculas e as minúsculas e no tipo blob ordena-se tendo em conta as maiúsculas e minúsculas.

Os tipos blob utilizam-se para armazenar dados binários como podem ser ficheiros.

TinyText e TinyBlob: Coluna com uma longitude máxima de 255 caracteres.

Blob e Text: um texto com um máximo de 65535 caracteres.

MediumBlob e MediumText: um texto com um máximo de 16.777.215 caracteres.

LongBlob e LongText: um texto com um máximo de caracteres 4.294.967.295. Há que ter em conta que devido aos protocolos de comunicação os pacotes podem ter um máximo de 16 Mb.

Enum: campo que pode ter um único valor de uma lista que se especifica. O tipo Enum aceita até 65535 valores diferentes.

Set: um campo que pode conter nenhum, um ou vários valores de uma lista. A lista pode ter um máximo de 64 valores.

Tipo de campo	Tamanho de Armazenamento
CHAR(n)	n bytes
VARCHAR(n)	n +1 bytes
TINYBLOB, TINYTEXT	Longitude+1 bytes
BLOB, TEXT	Longitude +2 bytes
MEDIUMBLOB, MEDIUMTEXT	Longitude +3 bytes
LOB, LONGTEXT	Longitude +4 bytes
ENUM('value1','value2',...)	1 ó dos bytes dependendo do número de valores
SET('value1','value2',...)	1, 2, 3, 4 ó 8 bytes, dependendo do número de valores

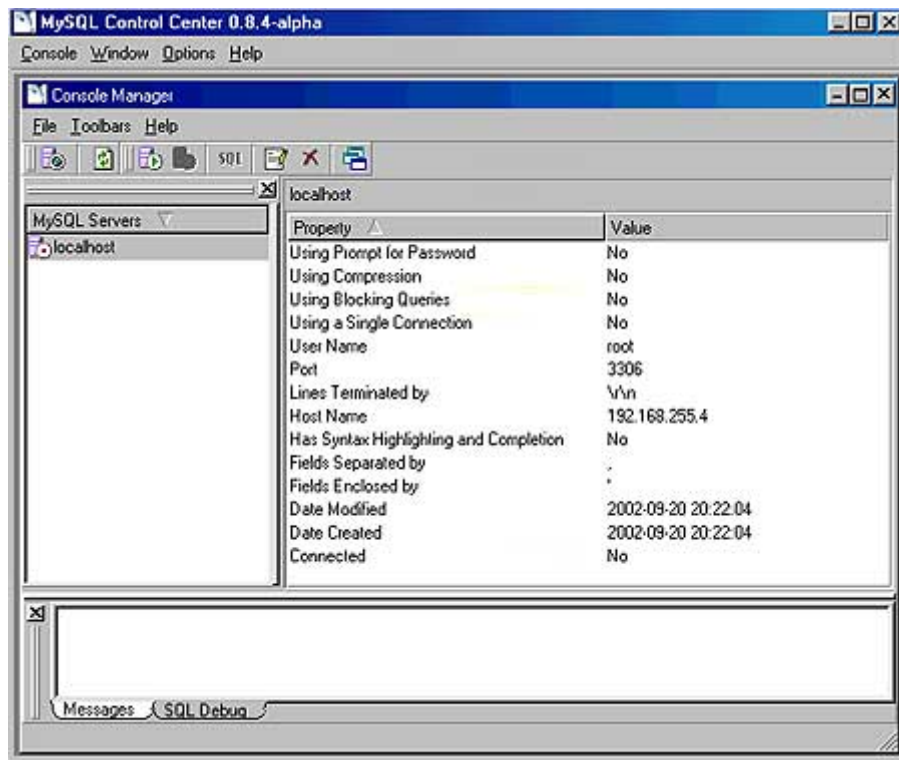
Diferença de armazenamento entre os tipos Char e VarChar

Valor	CHAR(4)	Armazenamento	VARCHAR(4)	Armazenamento
"	"	4 bytes	"	1 byte
'ab'	'ab '	4 bytes	'ab'	3 bytes
'abcd'	'abcd'	4 bytes	'abcd'	
'abcdefgh'	'abcd'	4 bytes	'abcd'	5 bytes

Informe de **Carlos Cuenca**
 Consultoria Informática
 Mail: carlos@desarrolloweb.com

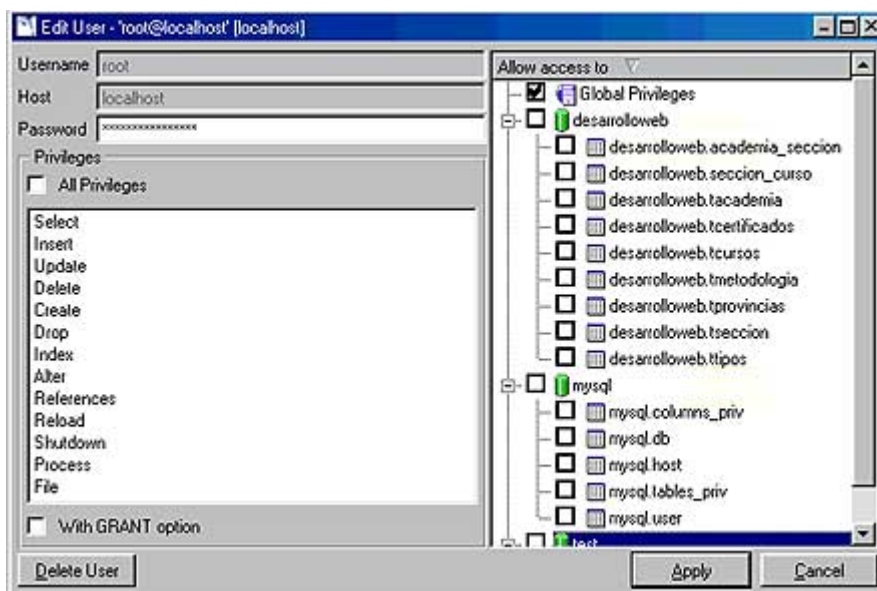
Mysql Control Center

MyCC é uma consola de Administração para MySQL, que permite realizar todas as tarefas de administração e trabalho de MySQL através de um interface gráfico, e que vem a resolver uma das maiores objecções que os usuários tinham de MySQL.



Para utiliza-lo, primeiro terás que fazer download de MySQL.com. Existem actualmente versões disponíveis para Linux e para Windows. Para poder trabalhar com um servidor de MySQL, primeiro terás de o dar de alta através do ícone de Registrar servidor. Aparece uma caixa de diálogo na qual se podem configurar todos os parâmetros de acesso ao servidor, como o nome do servidor, nome do usuário, password, porto etc. Uma vez conectado ao servidor poderá-se aceder às tarefas de administração dos usuários de MySQL, administrador do gestor, e administração das tabelas.

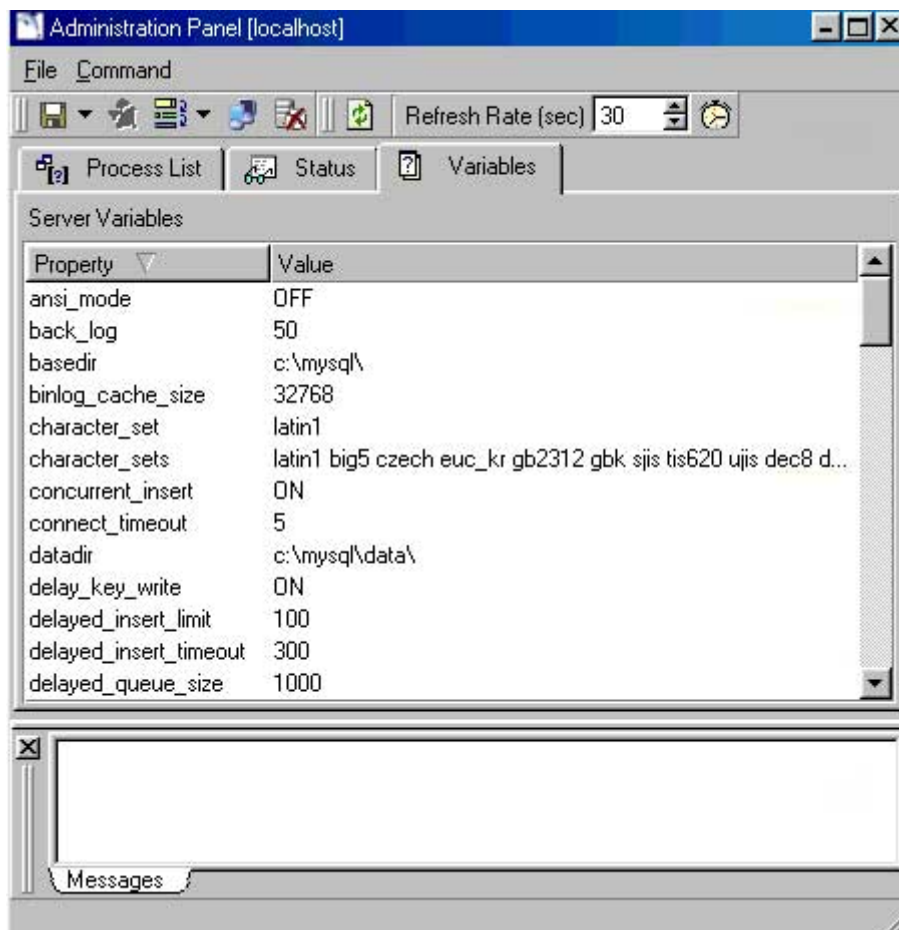
Administrador de Usuários A partir dele pode-se acrescentar, apagar, modificar as propriedades dos usuários de MySQL. Ao fazer click sobre um usuário aparece a janela de configuração do usuário.



Nela podem-se configurar o nome de usuário e a password. Em relação às permissões, permite estabelecer um sistema de permissões geral ou individualizado para cada uma das tabelas. Para além disso permite dar-lhe permissão ao usuário para ceder a outros (GRANT OPTION). Um dos pontos fortes em relação a segurança é que ao definir um usuário, temos de especificar o servidor desde o qual vai aceder, limitando de esta forma, os riscos de acesso não autorizado.

Administração do servidor

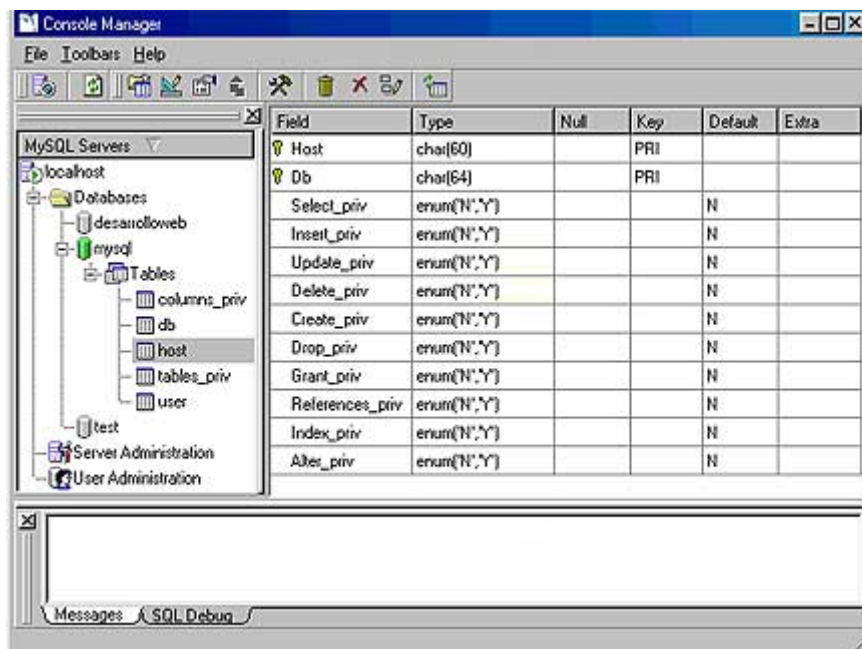
A janela de Administração do servidor consta de três opções nas quais se pode aceder à lista de processos, às variáveis de estado do servidor e às variáveis de configuração.



Desde esta janela para além de poder fazer um Ping ao servidor, para-lo, ou guardar o conteúdo das variáveis de estado. Desde a opção de processos pode-se ver o estado de cada um dos processos ativos, vendo as suas propriedades e com a opção de para-los através de um Kill. Desde a opção de estado, pode-se ver o conteúdo de cada uma das variáveis de estado, e pode-se personalizar para mostrar as que interessarem. Desde a opção de variáveis pode-se aceder ao valor de cada uma das variáveis de configuração do servidor MySQL.

Janela de Base de Dados

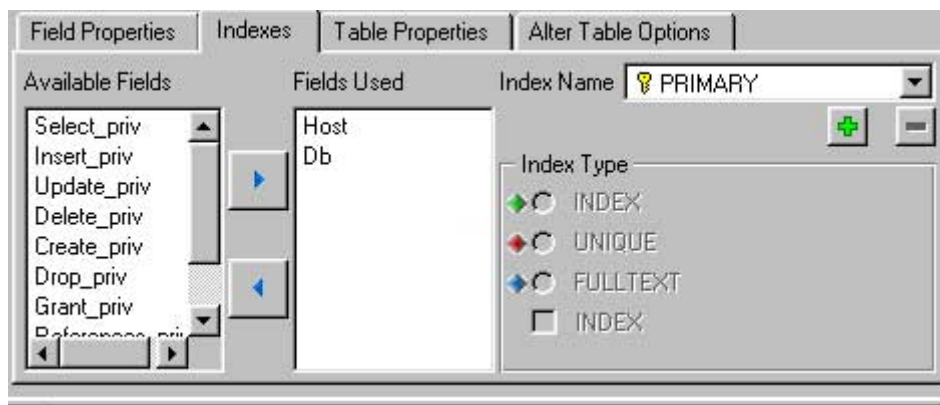
Ao seleccionar uma base de dados dentro de uma selecção de bases de dados, aparece a janela de base de dados, onde podemos ver informação relacionada com a base de dados, como por exemplo o número de tabelas, o número de consultas por segundo, o tempo de funcionamento, etc. Ao seleccionar uma base de dados, pode-se ver os nomes das tabelas e o número de registos que contem. As tabelas podem-se eliminar, vaziar e renomear. Quando se selecciona uma tabela, podem-se ver os nomes dos campos, os tipos de dados e os parâmetros opcionais de cada um deles.



Esta janela permite criar novos campos, elimina-los ou modificar as suas propriedades. Mediante o ícone de ferramentas, pode-se analisar a tabela, optimiza-la e repara-la. Ao fazer duplo click sobre uma tabela mostra-se na janela os dados que contem. Fazendo click sobre qualquer um dos campos, acede-se a estrutura. Para cada campo pode-se escolher o tipo de dados, o tamanho, e se é chave ou não e o valor pré-determinado.



Dentro da tabela, existe uma opção para configurar os índices que tem.



Desde essa janela podemos seleccionar os campos que contem, e também o tipo de índice que se deseja criar. Desde a opção de propriedades da tabela acede-se aos atributos, como por exemplo o tipo de tabela, o modo de escrita, o tamanho máximo de cada fila, etc.

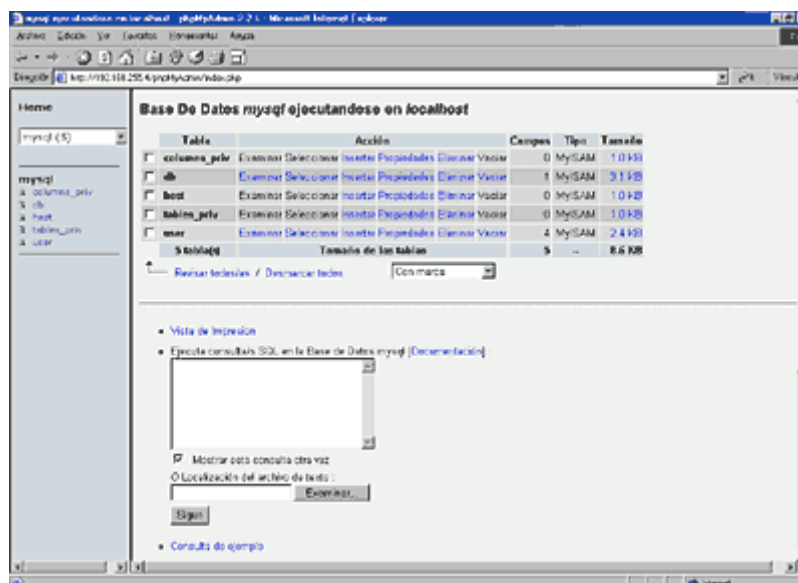
Field Properties	Indexes	Table Properties	Alter Table Options
Table Name	host	Table Type	MyISAM
Comment		Row Format	DEFAULT
Min Rows		Pack Keys	1
Max Rows	100	Delay Key Write	[DEFAULT]
Avg. Row Length		Checksum	[DEFAULT]
Auto Increment			

Em fim, como podes ver esta é a consola de acesso ao servidor MySQL de que estávamos a espera. O produto distribui-se como software livre, por tanto, tanto o download como a sua utilização ilimitada é gratuita. Está disponível as últimas versões Linux e Windows para além de muitíssima informação no seguinte endereço: www.mysql.com/products/mysqlcc/.

Informe de **Carlos Cuenca**
 Consultoria Informática
 Mail: carlos@desarrolloweb.com

phpMyAdmin

phpMyAdmin é um programa de distribuição livre em PHP, criado por uma comunidade sem animo de lucro, que só trabalha no projecto por amor a arte. É uma ferramenta muito completa que permite aceder a todas as funções típicas da base de dados MySQL através de uma interface web muito intuitiva.



A aplicação em si não é mais do que um conjunto de ficheiros escritos em PHP, os quais podemos copiar no nosso servidor web, de modo que, quando acedemos a esses ficheiros vemos umas páginas onde podemos encontrar as bases de dados às que temos acesso no nosso servidor de bases de dados e todas as suas tabelas. A ferramenta permite-nos criar tabelas, inserir dados nas tabelas existentes, navegar pelos registos das tabelas, editá-los e apagá-los, apagar tabelas e um grande etc, inclusive executar sentenças SQL e fazer um backup da base de dados.

Página de phpMyAdmin

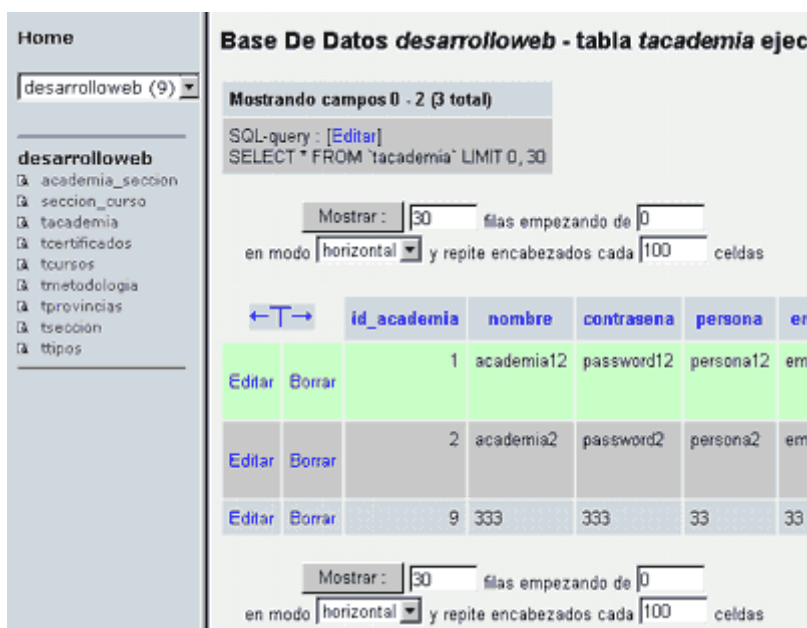
A página de inicio do projecto é <http://www.phpmyadmin.net/>. Daí podemos fazer download dos ficheiros da última versão da aplicação, que posteriormente devemos colocar no nosso servidor web. Também podemos encontrar o phpMyAdmin na rede, como por exemplo Sourceforge.net, que é um site que tem montes de projectos "Open Source".

Há várias versões disponíveis, mas é recomendável escolher a que nos aconselhem como a última versão estável (The last stable version). Na hora de escrever este artigo era a 2.2.6. De modo que, se

o nosso sistema +e Windows, optaremos pelo ficheiro phpMyAdmin-2.2.6-php.zip.

Este ficheiro é para a versão 4 de PHP, mas também temos a possibilidade de obter os ficheiros que tem compatibilidade com a versão 3 de PHP, para aqueles que não tem o motor de PHP mais potente.

A página de inicio do programa também nos oferece a possibilidade de ver uma demo online, ainda que avisam que o servidor onde se encontra pode estar caído.
<http://www.phpmyadmin.net/phpMyAdmin/>



A instalar phpMyAdmin

Uma vez obtida a última versão temos de a descomprimir, para obter os ficheiros PHP (a ferramenta) e coloca-los dentro do directório de publicação do nosso servidor web.

Nota: recordamos que phpMyAdmin é um projecto escrito em PHP, por conseguinte necessitaremos colocar os ficheiros num servidor web que permita programação de páginas PHP. Deveremos aceder à ferramenta através do endereço do servidor web, seguido do directório no que temos os ficheiros. Por exemplo, se o nosso servidor é o PWS e colocámos os ficheiros dentro do directório de publicação (Normalmente C:\Inetpub\wwwroot), no subdirectório phpMyAdmin, devemos escrever algo similar a <http://localhost/phpMyAdmin>. Se tivéssemos instalado um servidor Apache os colocaríamos na pasta que tivéssemos indicado como "documentRoot", que costuma ser htdocs.

O primeiro que podemos ler é o ficheiro da documentação, que encontramos junto aos ficheiros de phpMyAdmin. Explica dados gerais do programa, como os seus requerimentos, instruções de instalação, configuração, perguntas frequentes, etc.

Posteriormente, tal e como explica a documentação, temos de editar o ficheiro config.inc.php para mudar os valores de host da base de dados (maquina que tem instalado o MySQL) e o usuário e password com o qual nos vamos conectar. Podemos configurar muitos aspectos na ferramenta, ainda que agora só vou comentar os que acho que são essenciais para fazer que funcione. Na documentação temos um ponto dedicado por completo a especificar o sentido de cada variável.

\$cfgPmaAbsoluteUri

Devemos atribuí-lo ao path completo necessário para aceder a phpMyAdmin. Por exemplo: <http://localhost/phpMyAdmin> ou <http://www.meudominio.com/phpMyAdmin>

\$cfgServers[\$i]['host'] string

O nome do host da base de dados. Por exemplo localhost, se o computador onde está instalada a base de dados e onde estamos a instalar phpMyAdmin é o mesmo. Também poderia ser o endereço IP do computador ao que nos ligamos.

\$cfgServers[\$i]['user'] string

\$cfgServers[\$i]['password'] string

O par usuário/password que deve utilizar phpMyAdmin para conectar-se com o servidor MySQL.

Com isto, já podemos aceder a phpMyAdmin e trabalhar com a nossa base de dados mais facilmente, que se torna muito de agradecer tendo em conta que, no caso de não ter esta ferramenta ou outra parecido, a outra opção seria utilizar a linguagem SQL, e no caso de que a base de dados esteja alojada em remoto na internet, não o poderíamos fazer, a não ser com acesso por TELNET ao servidor da base de dados.

Informe de **Carlos Cuenca**
Consultoria Informática
Mail: carlos@desarrolloweb.com

Instalar o driver ODBC para conectar com uma base de dados MySQL

Instalar este driver serve para que desde um sistema Microsoft Windows possamos aceder a uma base de dados MySQL. As aplicações são várias, por exemplo podemos utiliza-lo para criar um DSN associado a uma base de dados MySQL, de modo que as nossas páginas ASP poderiam aceder à base de dados. Outra aplicação é aceder desde Access a base de dados MySQL e exportar ou importar dados (migrar os dados de Access a MySQL ou vice-versa), inclusive para criar um back-end da nossa base de dados MySQL em interface Access.

O primeiro que temos de fazer no computador que tem o sistema Windows XX e Access é actualizar a versão 6 de Microsoft jet: <http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q239114&>

A seguir, temos de fazer download da última versão de Myodbc da página de MySQL:
<http://www.mysql.com/downloads/api-myodbc-2.50.html>

Com tudo isto feito, instalamos a actualização de Microsoft jet, e instalamos os drivers ODBC de MySQL. Quando pergunta no ecrã de "Data Sources" faz click em "Close" para terminar.

Com o driver ODBC instalado, acede ao painel de controlo de ODBC de 32 bits (Início->Configuração->Painel de Controlo->Fontes de dados ODBC 32 bits). Neste momento, terás de escolher se queres utilizar o driver para um usuário (DSN de usuário), ou para qualquer usuário do computador (DSN de Sistema). Uma vez escolhida uma opção, faz click no botão "Add" para acrescentar uma nova fonte de dados e a seguir, selecciona o driver de MySQL. Veremos o seguinte ecrã:.

Terás de preencher os seguintes campos:

Windows DSN name: Nome da fonte de dados que estará disponível desde Windows.

Mysql host (name or IP): Nome ou endereço IP do computador onde se encontra instalado o servidor MySQL

Mysql Database Name: Nome da base de dados com a que se trabalhará desde a fonte de dados.

User: Nome de usuário com o que se acederá ao servidor de bases de dados.

Password: palavra-chave do usuário.

Port: Serve para especificar o porto no que se encontra o servidor MySQL, há que pôr um valor no caso de que não se esteja a utilizar o pré-definido, que é o 3306.

Uma vez que estejam estas opções configuradas, fazemos click em "OK" para fechar as janelas e finalizar o processo.

Informe de **Carlos Cuenca**
Consultoria Informática
Mail: carlos@desarrolloweb.com

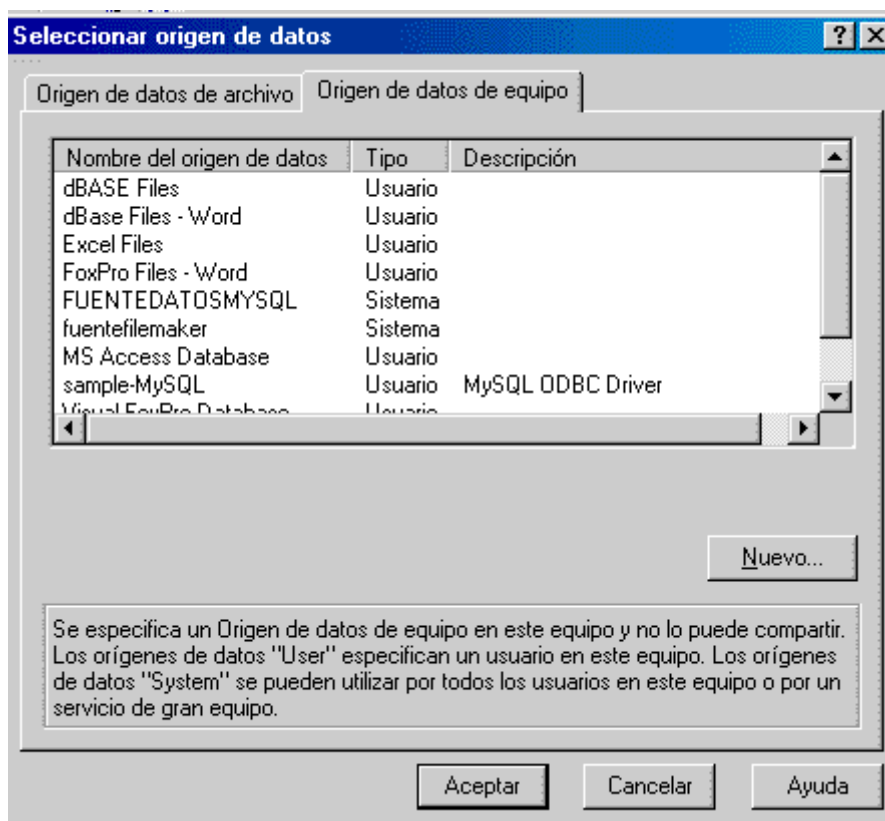
Exportar dados de MySQL a Microsoft Access

Migrar dados da uma base de dados a outra é algo ao que muitos de nós tivemos de confrontar em algum momento. A continuação explicamos como recuperar informação armazenada num servidor de dados MySQL a uma base Access.

Referência: Para realizar esta tarefa é necessário que tenhamos obtido o driver ODBC e o tenhamos instalado no nosso sistema Windows. Esta tarefa pode-se conhecer num artigo de CriarWeb.com: [Instalar o driver ODBC para](#)

MySQL.

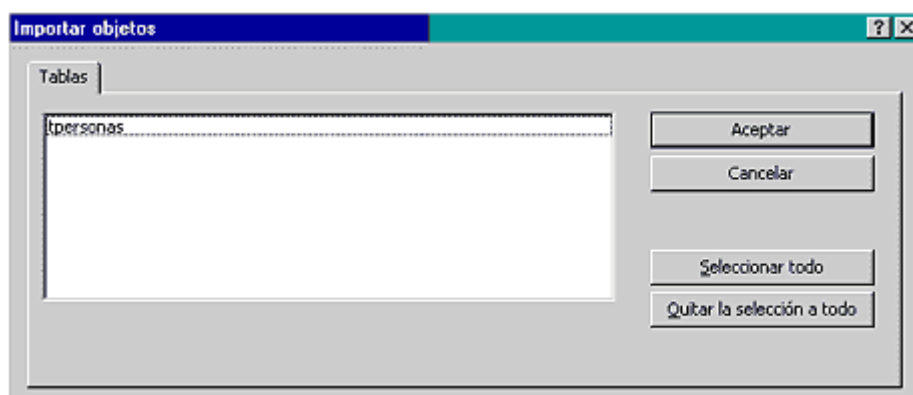
Para importar uma tabela de MySQL a Microsoft Access, desde Access, e com a base de dados na que desejamos importar os dados aberta, seleccionar o menu Ficheiro->Obter dados Externos->Importar. No ecrã de importar dados, na opção tipo do ficheiro seleccionar ODBC databases().



Seleccionar origem de dados, e dentro de esta, o nome da fonte de dados que criámos anteriormente. Uma vez seleccionada, e fizeste click em "Aceitar", aparecerá um ecrã de configuração do driver e podes configurar algumas opções de configuração que aparecem no driver ODBC, se não desejas marcar nenhuma, click sobre "OK".

Nota: é possível que, em algum caso, os dados da base nos sistemas MySQL e Access não sejam totalmente compatíveis e se produza alguma anomalia ao exporta-los. Realmente é só uma hipótese que pensamos, ainda que nas provas que realizámos não vimos nenhum tipo de problema. Claro que os campos com os que trabalhamos não eram estranhos.

Aparecerá uma janela onde pergunta que tabela de MySQL queremos exportar a Access:



Seleccionar a tabela, e faz click sobre "Aceitar"

Nota: se estamos a exportar dados a um servidor de bases de dados alojado nalgum provedor de hosting, temos de ter em conta que estes não sempre incluem acesso remoto ao servidor da base de dados, ou requer um aviso explícito por parte do cliente para a sua configuração.

Referência: se desejamos uma migração de dados no sentido oposto, isto é, desde Access a MySQL, será tratado noutro artigo em CriarWeb que explica o processo minuciosamente: [Exportar dados de Access a MySQL.](#)

Informe de **Carlos Cuenca**
Consultoria Informática
Mail: carlos@desarrolloweb.com

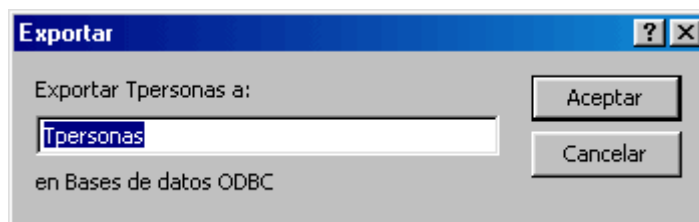
Exportar dados de Access a MySQL

Não é estranho que tenhamos começado a fazer os nossos primeiros sites utilizando uma base de dados simples como Access. Também não é nada estranho que, chegado o momento, passemos a coisas mais serias e nos passemos a um servidor de dados como MySQL. Aqui mostramos uma maneira bastante prática de migrar os dados de uma base Access a MySQL.

Referência: Para realizar esta tarefa é necessário que tenhamos obtido o driver ODBC e o tenhamos instalado no nosso sistema Windows. Esta tarefa pode-se conhecer num artigo de CriarWeb.com: [Instalar o driver ODBC para MySQL](#).

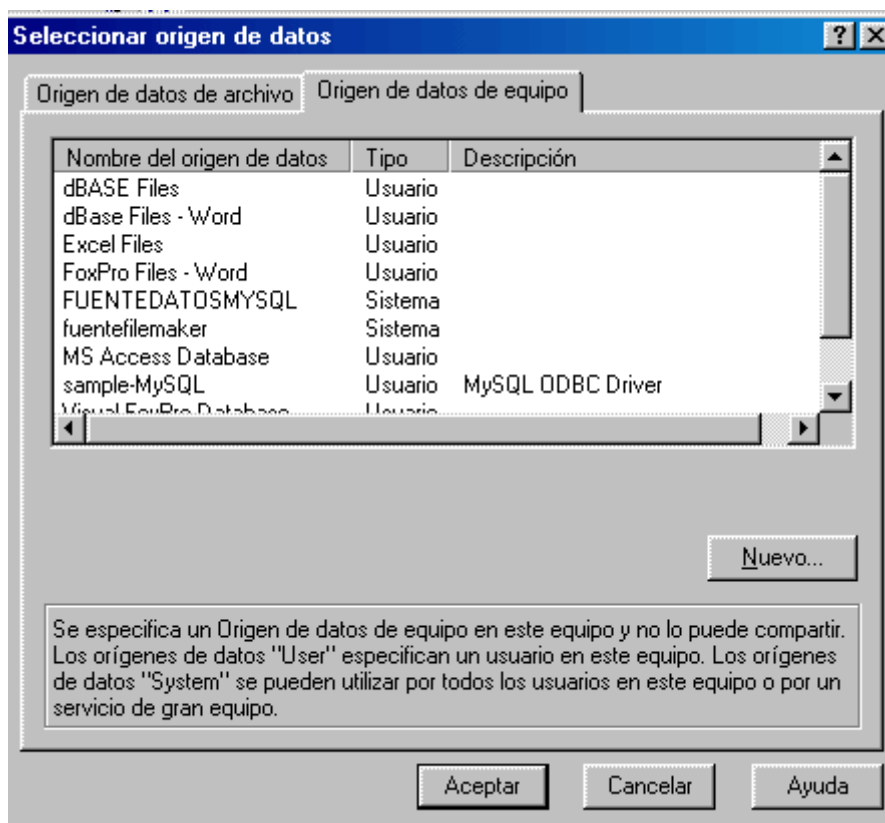
Para exportar uma tabela a MySQL, há que abrir a base de dados e seleccionar a tabela. Depois, fazer click sobre Ficheiro->Exportar. Na janela de exportar, na opção de guardar como tipo, seleccionar ODBC databases().

Feito isto, aparece uma janela que nos pergunta o nome que lhe queremos dar a tabela MySQL, pré-definidamente aparece o mesmo.



Faz click sobre "Aceptar", e aparecerá uma janela na que se pede que selecciones a origem de dados ODBC:

Nota: É possível que, em algum caso, os dados da base nos sistemas MySQL e Access não sejam totalmente compatíveis e se produza alguma anomalia ao exporta-los. Realmente é só uma hipótese que pensamos, ainda que nas provas que realizámos não vimos nenhum tipo de problema. Claro que os campos com os que trabalhamos não eram estranhos.



Seleccionar origem de dados, e dentro, o nome da fonte de dados que criámos anteriormente. Uma vez seleccionado, fazer click em "Aceitar", aparece um ecrã de configuração do driver que nos permitirá, se queremos, configurar algumas opções sobre o driver. Se não queremos configurar nada, faz click sobre "OK" e os dados começarão a exportar-se.

Nota: se estamos a exportar dados a um servidor de bases de dados alojado nalgum provedor de hosting, temos d éter em conta que estes não sempre incluem acesso remoto ao servidor da base de dados, ou requer um aviso explícito por parte do cliente para a sua configuração.

Referência: se desejamos uma migração de dados no sentido oposto, isto é, desde MySQL a Access, foi tratado noutra artigo em CriarWeb que explica o processo minuciosamente: [Exportar dados de MySQL a Microsoft Access](#).

Informe de **Carlos Cuenca**
Consultoria Informática
Mail: carlos@desarrolloweb.com

Administrar e actualizar uma base de dados MySQL desde Access

Um dos maiores problemas dos que adoece actualmente MySQL é não ter um sistema gráfico que satisfaça a maioria dos usuários. Existem magníficos projectos através de páginas web, como phpMyAdmin, mas muitas vezes há gente que gostaria de ter algo parecido a um Access, possibilidade de incluir formulários para a entrada de dados, ou de informes sobre os dados armazenados dentro de uma tabela.

Dado que não existe hoje em dia nenhuma ferramenta parecida a Access para trabalhar com MySQL utilizando um modo gráfico de Access. Poderás utilizar os formulários, consultas e informes de Access com os dados dos ficheiros de MySQL.

Desta forma, o encarregado de actualizar os dados duma página, poderá trabalhar desde Access, com a comodidade dos formulários, os menus dinâmicos etc, e os dados serão enviados automaticamente a MySQL.


```

mysql> connect access
Connection id: 27
Current database: access

mysql> select nombrep,apellido1,apellido2,telefono,direccion from tpersonas1 where id=304;
+-----+-----+-----+-----+-----+
| nombrep | apellido1 | apellido2 | telefono | direccion |
+-----+-----+-----+-----+-----+
| sadffsdfsd | sdfsd | sdfsd | 324324234 | sdafadsad |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql>

```

Imagem da consola de MySQL

Datos Personales:

nombrep: apellido1: apellido2: sexo:

Dirección de envío:

Direccion: CP:

Localidad: Número: Provincia: Piso:

Otros datos de contacto:

Telefono: fechaNacimiento:

Registros: 1 de 1

Formulário de Access que acede a uma base de dados MySQL

Nota: Duas imagens que se diferenciam um mundo. Com as duas pode-se administrar uma base de dados MySQL, uma corresponde a consola MySQL (a ferramenta mais básica) e a outra a um formulário com Access, que sem dúvida oferece uma interface muito mais agradável.

Ainda por cima, para a geração de formulários e informas pode-se utilizar os assistentes... tudo uma poupança de tempo!!!

Referência: Para realizar esta tarefa é necessário que tenhamos obtido o driver ODBC e o tenhamos instalado no nosso sistema Windows. Esta tarefa pode-se conhecer num artigo de CriarWeb.com: [Instalar o driver ODBC para MySQL](#).

Agora, uma vez instalado o driver MyODBC, o primeiro que há que fazer é criar uma base de dados em branco desde a qual se irão vincular as tabelas. Uma vez criada, faz-se click sobre a opção de criar nova tabela. Aparecerá a seguinte janela na que se seleccionará criar nova tabela vinculada:

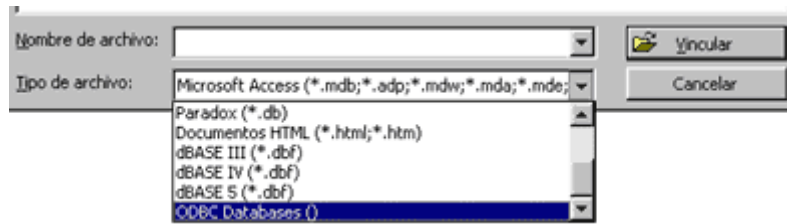
Nueva tabla

Este asistente crea tablas vinculadas a un archivo externo en la base de datos actual.

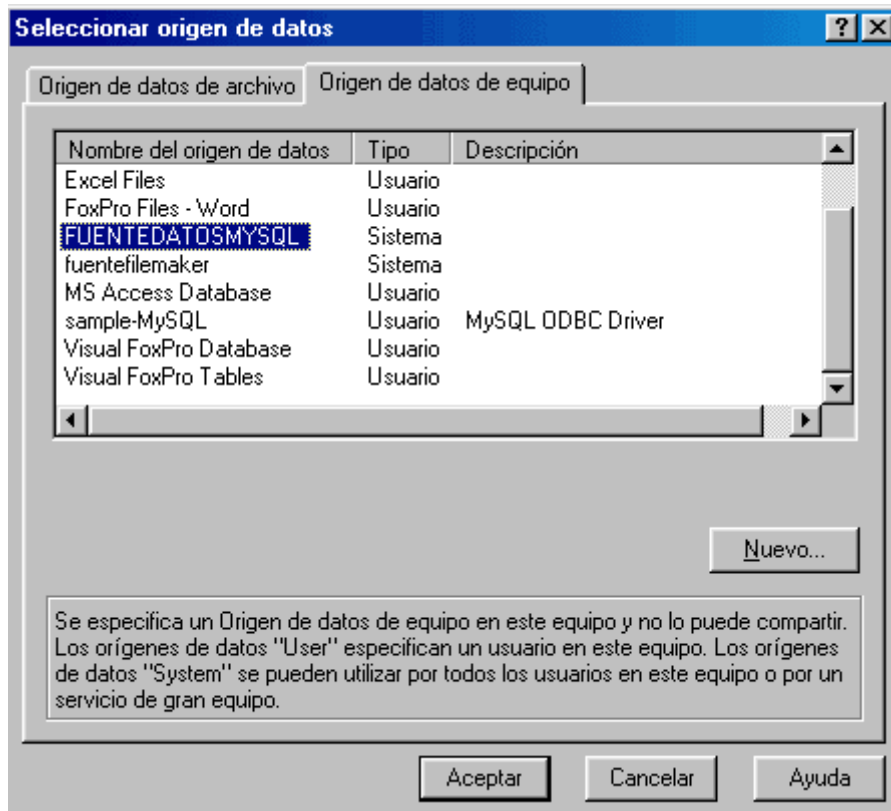
- Vista Hoja de datos
- Vista Diseño
- Asistente para tablas
- Importar tabla
- Vincular tabla**

Aceptar Cancelar

Aparece a janela de vincular uma tabela de uma base da dados, e na parte inferior seleccionamos o tipo de ficheiro: ODBC Databases().

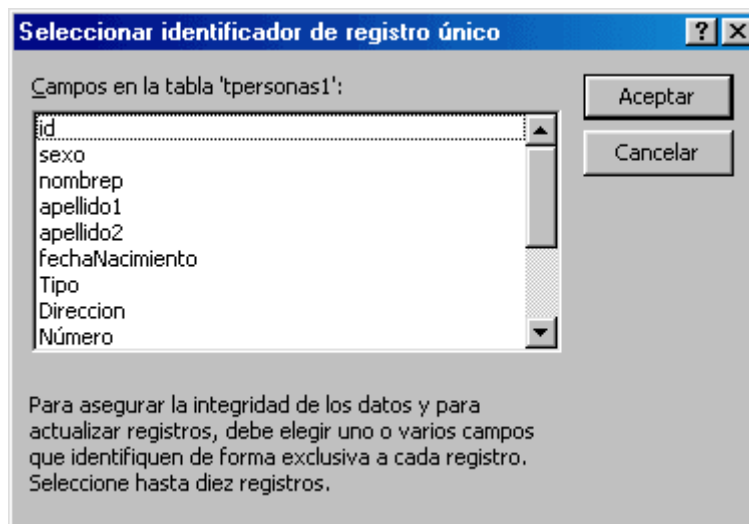


Ao fazer click sobre vincular, aparece uma janela para seleccionar uma origem de dados, seleccionamos a fonte de dados que criámos na primeira parte do artigo:



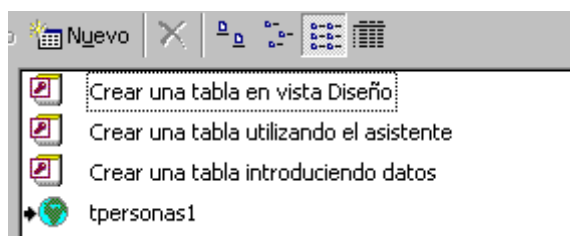
A seguir, depois de fazer click em "Aceitar", aparece a janela de configuração da fonte de dados ODBC de MySQL. Como já está configurada, fazemos click em "OK", e aparecerá a janela na que se podem escolher entre as tabelas que contem a base de dados para a qual configuramos a fonte de dados ODBC.

Seleccionamos uma tabela, e a seguir aparecerá uma janela onde devemos especificar, até um máximo de dez, os campos que formam parte da chave.



Seleccionamos os campos, e fazemos click en "Aceitar".

Aparecerá uma nova tabela dentro da base de dados de Access. Esta tabela, não existe fisicamente dentro do ficheiro de Access, por tanto, todas as modificações que se realizam sobre a mesma, enviam-se através de ODBC ao ficheiro MySQL.



A partir de agora, poderão-se criar formulários, consultas e informes sobre esta tabela tal e como se faz normalmente desde Access.

Nota: Alguns provedores de hosting não incluem no seu pack básico o acesso remoto ao servidor de base de dados, ou requer de um aviso explícito por parte do cliente para a sua configuração.

Informe de **Carlos Cuenca**
 Consultoria Informática
 Mail: carlos@desarrolloweb.com

Connector/J

Mysql Connector é um driver criado por Mysql AB que te permitirá trabalhar com Mysql desde programas escritos em Java. A diferença em relação a outros drivers, é que este é de livre distribuição, e tem um bom rendimento.

MySQL Connector/J é um driver nativo de Java que converte as chamadas geradas por JDBC no protocolo de rede que utiliza a base de dados Mysql. Permite trabalhar com a linguagem de programação Java e desta forma construir programas que interactivam com Mysql.

O MySQL Connector/J é um driver JDBC tipo IV e contém todas as características de JDBC para utilizar Mysql.

No desenvolvimento das últimas versões incrementou-se bastante a velocidade do driver, ganhando tanto em rapidez como em eficácia. O driver suporta resultados de dados "streaming" o que permite ao usuário recolher um grande número de filas sem a necessidade de utilizar um buffer de memória. O driver implementa um protocolo de packs grandes que permitem enviar filas e campos BLOBs de até 2 GigaBytes. Na versão de desenvolvimento do mesmo implementam-se as novas características acrescentadas no API 3.0 de JDBC.

O driver foi testado para trabalhar com as seguintes ferramentas de bases de dados e servidores de aplicações:

- Apache
- Jboss
- BEA Weblogic
- IBM VisualAge for Java
- IBM WebSphere Application Server 4.0
- Forte for Java
- Oracle SQL/J
- TableGen

Download

Para instalar o driver há que fazer download da página de Mysql [Connector/J](#)

Informe de **Carlos Cuenca**

Consultoria Informática

Mail: carlos@desarrolloweb.com

Cópias de segurança em MySQL

O gestor de Bases de dados MySQL incluem varias ferramentas para a realização de copias de segurança da base de dados. Através delas poderemos por a salvo os nossos dados, para que, no hipotético caso de que se percam, poder recupera-los.

Na hora de fazer uma cópia de segurança, o primeiro que há que ter em conta é a integridade dos dados que se estão a guardar. Em todos os casos é necessário que haja integridade nos dados de uma tabela, com isto quero dizer que todos os dados da tabela deverão estar escritos na tabela, isto pode parecer um pouco estranho, mas tal e como acontece com todos os gestores de bases de dados, MySQL dispõem de diversas "cachês" nas que se armazenam dados temporalmente com o objetivo de melhorar o rendimento, de forma que por exemplo, uma vez feita uma modificação numa tabela, pode ser que os dados não se guardem imediatamente em disco, até que termine, por exemplo, uma consulta que se estava a executar. Por isso, é necessário "forçar" MySQL a escrever todos os dados em disco, mediante a sentença "Flush Tables".

Para além disto, é necessário que não se escreva nas tabelas enquanto se está a fazer a cópia de segurança da base de dados. Isto faz-se através do comando "lock tables", seguido do nome da tabela. Pode haver bases de dados nas que seja necessário bloquear todas as tabelas ao mesmo tempo antes de fazer a cópia de segurança.

Existem várias opções para realizar a cópia de segurança de uma base de dados MySQL.

- Em primeiro lugar, podíamos utilizar alguma ferramenta comercial que gira todo o processo, isto tem como vantagens a simplicidade do método, e como inconveniente, que não costuma ser grátis, sendo necessário pagar uma licença pela sua utilização. Estas ferramentas não se analisarão neste artigo.
- Em segundo lugar, e a partir da versão 3.23.25 e posteriores, existe a possibilidade de realizar uma cópia de segurança através da sentença sql "backup table".
- Como terceira opção, é possível realizar cópias de segurança através das ferramentas que nos proporciona o próprio gestor de bases de dados, como podem ser mysqldump ou mysqlhotcopy.

Informe de **Carlos Cuenca**

Consultoria Informática

Mail: carlos@desarrolloweb.com

Backup MySQL mediante sentenças SQL

Veremos como realizar uma cópia de segurança da tabela e a sua recuperação a partir de sentenças SQL.

Backup dos dados

Este comando permite-nos fazer uma cópia dos ficheiros das tabelas das quais queremos fazer um backup, actualmente só funcionam com tabelas do tipo MyIsam, e copia tanto os ficheiros .frm que contêm a definição da tabela, como os ficheiros .myd, que contêm os dados.

Antes de executar-se, guarda todas as alterações que possam existir em memória da tabela, de forma que fique de uma maneira consistente. Do mesmo modo, durante a execução do comando, bloqueia a tabela sobre a que se está a fazer a cópia de segurança para que os dados sejam consistentes na tabela. Há que ter em conta que este comando bloqueia uma das tabelas, consoante vai fazendo a cópia de segurança. Isto poderia provocar uma inconsistência de dados a nível de base de dados, se é necessário fazer um backup de todas as tabelas num instante, sendo então necessário a utilização do comando "lock tables" para bloquear todas as tabelas antes de começar a cópia de segurança.

Ao finalizar o comando retorna uma tabela, que contem os dados:

Coluna Valores

Table Nome da tabela
Op Sempre põe "backup"
Msg_type Pode conter status, error, info ou warning.
Msg_text Mensagem descritivo do resultado da operação

Para realizar uma cópia de uma tabela chamada pedidos, à pasta Backups:

```
BACKUP TABLE pedidos TO `/backups/`
```

Recuperação de dados

Para complementar a sentença "Backup table", Mysql dispõem da sentença "Restore table", que permite restaurar uma tabela a partir de uma cópia de segurança que se realizou com o comando "Backup Table".

Esta opção só se pode utilizar se a tabela que se pretende restaurar não se encontra na base de dados, pois em caso afirmativo mostrará uma mensagem de erro. Igual que o comando de Backup, esta opção só está disponível para as tabelas de tipo MyIsam, e a partir da versão 3.23.25.

Como o comando de backup não copia os ficheiros de índices, o comando para recuperar os dados, volta a indexar todos os campos que contem índices, criando os ficheiros correspondentes. Ao igual que o comando para fazer cópias de segurança, invocar o comando "restore table" retorna uma tabela, com um registo por cada tabela sobre a que se fez a base de dados, e que contem a seguinte informação:

Coluna Valores

Table Nome da tabela
Op Sempre põe "restore"
Msg_type Pode conter status, error, info ou warning.
Msg_text Mensagem descritivo do resultado da operação

Para restaurar uma tabela chamada pedidos da pasta Backups à base de dados:

```
RESTORE TABLE pedidos FROM `/backups/`
```

Informe de **Carlos Cuenca**
Consultoria Informática
Mail: carlos@desarrolloweb.com

Backup MySQL com mysqldump

Este comando permite fazer a cópia de segurança de uma ou várias bases de dados. Também permite que estas cópias de segurança se possam restaurar em diversos tipos de gestores de bases de dados,

sem a necessidade de que se trate de um gestor de MySQL. Isto consegue-se criando uns ficheiros, que contêm todas as sentenças SQL necessárias para poder restaurar a tabela, que contem desde a sentença de criação da tabela, até uma sentença insert por cada um dos registos que formam parte da mesma.

O comando dispõe de uma ampla variedade de opções que nos permitirá realizar a cópia da forma mais conveniente para o propósito da mesma. Para poder restaurar a cópia de segurança, bastará com executar todas as sentenças SQL que se encontram dentro do ficheiro, bem desde a linha de comandos de MySQL, ou desde o menu de criação de sentenças SQL de qualquer entorno gráfico como pode ser o MySQL Control Center.

As limitações da restauração dependerão das opções que se especificaram na hora de fazer a cópia de segurança, por exemplo, se incluirmos a opção `-add-drop-table` ao fazer a cópia de segurança, poderão-se restaurar tabelas que existem actualmente no servidor (apagando-as primeiro). Pelo qual é necessário estudar primeiro os procedimentos que se utilizarão tanto na cópia como na restauração, para que tudo saia correcto!

Algumas das opções que tem são:

--add-locks

Adiciona LOCK TABLES antes, e UNLOCK TABLE depois da cópia de cada tabela.

--add-drop-table

Adiciona um drop table antes de cada sentença create.

-A, --all-databases

Copia todas as bases de dados. Isto é o mesmo que utilizar `-databases` seleccionando todas.

-a, --all

Inclui todas as opções de criação específicas de MySQL.

--allow-keywords

Permite a criação de nomes de colunas que são palavras chave, isto realiza-se pondo como prefixo a cada nome de coluna, o nome da tabela.

-c, --complete-insert

Utiliza inserts incluindo os nomes de coluna em cada sentença (incrementa bastante o tamanho do ficheiro)

-C, --compress

Comprime a informação entre o cliente e o servidor, se ambos suportam compressão.

-B, --databases

Para copiar várias bases de dados. Neste caso, não se especificam tabelas. O nome dos argumentos refere-se aos nomes das bases de dados. Incluirá-se `USE db_name` na saída antes de cada base de dados.

--delayed

Insere as filas com o comando INSERT DELAYED.

-e, --extended-insert

Utiliza a sintaxe de INSERT multilinha. (proporciona sentenças de insert mais compactas e rápidas)

-#, --debug[=option_string]

Utilização do modo de depuração.

--help

Mostra mensagem de ajuda e termina.

--fields-terminated-by=...

--fields-enclosed-by=...

--fields-optionally-enclosed-by=...

--fields-escaped-by=...

--lines-terminated-by=...

Estas opções utilizam-se com a opção `-T` e têm o mesmo significado que a correspondente cláusula

LOAD DATA INFILE.

-F, --flush-logs

Escreve em disco todos os logs antes de começar com a cópia.

-f, --force,

Continua ainda que se produza um erro de SQL durante a cópia.

-h, --host=..

Copia os dados do servidor de MySQL especificado. O servidor pré-definido é localhost.

-l, --lock-tables.

Bloqueia todas as tabelas antes de começar com a cópia. As tabelas bloqueiam-se com READ LOCAL para permitir inserts simultâneos em caso das tabelas MyISAM. Quando se realiza a cópia de múltiplas bases de dados. As tabelas em diferentes bases de dados copiarão-se em estados completamente diferentes.

-K, --disable-keys

Incluirá-se na saída `/*!40000 ALTER TABLE tb_name DISABLE KEYS */;e /*!40000 ALTER TABLE tb_name ENABLE KEYS */;` Isto fará que o carregamento de dados num servidor MySQL 4.0 se realize mais rápido devido a que os índices se criarão depois de que todos os dados tenham sido restaurados.

-n, --no-create-db

Não se incluirá na saída `CREATE DATABASE /*!32312 IF NOT EXISTS*/ db_name;` Esta linha inclui-se se a opção `--databases` ou `--all-databases` foi seleccionada.

-t, --no-create-info

Não incluirá a informação de criação da tabela (sentença `CREATE TABLE`).

-d, --no-data

Não incluirá nenhuma informação sobre os registos da tabela. Esta opção serve para criar uma cópia só da estrutura da base de dados.

--opt

O mesmo que `--quick --add-drop-table --add-locks --extended-insert --lock-tables`. Esta opção deveria permitir realizar a cópia de segurança da base de dados da forma mais rápida e efetiva.

-pyour_pass, --password[=your_pass]

Password utilizada quando se conecta com o servidor. Se não se especifica `=your_pass`, `mysqldump` perguntará a password.

-P, --port=...

Porto utilizado pelas ligações TCP/IP.

--protocol=(TCP | SOCKET | PIPE | MEMORY)

Especifica o protocolo de conexão que se utilizará.

-q, --quick

Não armazena no buffer a sentença, copia-a diretamente na saída. Utiliza `mysql_use_result()` para realiza-lo.

-Q, --quote-names

Põe entre aspas as tabelas e nomes de coluna.

-r, --result-file=...

Redireciona a saída ao ficheiro especificado. Esta opção deveria utilizar-se em MSDOS, porque previne a conversão de nova linha `'\n'` em nova linha e return `'\n\r'`.

--single-transaction

Utiliza o comando `BEGIN` antes de realizar a cópia desde o servidor. É muito útil com as tables InnoDB e o nível de transação `READ_COMMITTED`, porque deste modo realizará-se a copia de segurança de um estado consistente sem a necessidade de bloquear as aplicações. Consultar o manual para mais detalhes.

-S /path/to/socket, --socket=/path/to/socket

O ficheiro de sockets que se especifica ao conectar ao localhost (que é o host pré-definido).

--tables

sobrescreve a opção --databases (-B).

-T, --tab=path-to-some-directory

Cria um ficheiro table_name.sql, que contem a sentença de criação de SQL, e um ficheiro table_name.txt, que contem os dados de cada tabela. O formato do ficheiro '.txt' realiza-se com as opções --fields-xxx y --lines--xxx options.

Nota: Esta opção só funciona se o comando mysqldump se executa na mesma máquina que o demónio (daemon) mysqld, o usuário deverá ter licença para criar e escrever o ficheiro no path especificado.

-u nombre_usuario, --user=nombre_usuario

O nome de usuário que se utilizará quando se conecte com o servidor, o valor pré-definido é o do usuário actual.

-v, --verbose

Vai mostrando informação sobre as acções que se vão realizando (mais lento)

-w, --where='cláusula where'

Serve para realizar a cópia de determinados registos.

-X, --xml

Realiza a cópia de segurança num documento XML.

-x, --first-slave

Bloqueia todas as tabelas de todas as bases de dados.

Exemplo:

Para realizar a cópia de segurança da base de dados minhabase ao ficheiro copia_seguranca.sql

```
mysqldump --opt minhabase > copia_seguranca.sql
```

Informe de **Carlos Cuenca**

Consultoria Informática

Mail: carlos@desarrolloweb.com

Subir uma base de dados ao servidor de Internet

Para além dos ficheiros da página, devemos subir a base de dados com a que temos de trabalhar. As bases de dados com as que trabalha PHP são muito variadas e em vários casos podemos utilizar uma ou outra, e por tanto o modo de subir a base de dados também pode variar.

É muito normal que o nosso provedor de hosting ofereça junto com PHP a base de dados MySQL, assim que os apontamentos para subir essa base de dados ao servidor de este artigo vão encaminhados a oferecer soluções para essa base de dados.

A base de dados MySQL não se pode subir por FTP, como se fazia com os ficheiros do código PHP. Para a subir teremos que utilizar outros mecanismos. Vou distinguir entre três casos diferentes nos que poderíamos encontrar-nos neste momento:

1. A base de dados que pretendemos subir está vazia. Só criámos as tabelas, mas não introduzimos dados ou simplesmente introduzimos algum dado como experiência.
2. A base de dados que queremos subir está completa e é uma base de dados MySQL. Neste caso teremos de criar a base de dados em local com toda a informação dentro e obviamente, queremos que esteja disponível essa informação em remoto.
3. A base de dados está completa (como no caso anterior), mas não é uma base de dados MySQL. Neste caso estaríamos a fazer uma migração da base de dados de um sistema gestor a outro.

Veremos os três casos por separado mais à frente, mas antes disso, vamos mostrar umas ferramentas que nos serão muito úteis para a administração de qualquer base de dados remota.

As ferramentas em concreto são as seguintes:

- **PhpMyAdmin.** Uma aplicação criada em PHP que podemos instalar no nosso espaço de alojamento para administrar a base de dados.
- **Mysql Control Center (a partir de agora MyCC).** Uma aplicação Windows que permite conectar-se a diversas bases de dados MySQL, que se estejam em local ou em remoto.
- **Access.** Também permite administrar uma base de dados MySQL conectada em local ou em remoto. Neste caso utiliza-se uma interface que muitos já conhecem, como é a do Access, para administrar uma base de dados que nada tem a ver com o mencionado programa.

Nos três casos o que nos permite realizar o software de administração são tarefas, sobre as bases de dados, de todo tipo, como podem ser criar tabelas, modificá-las, inserir dados, apagá-los, editá-los. Modificar ou apagar tabelas ou campos das mesmas, etc.

A eleição de uma ferramenta ou de outra passa pelos recursos que nos permitam utilizar no nosso provedor. Basicamente, o que nos pode levar a escolher uma opção ou outra é se nos permitem ou não conectar de maneira remota à base de dados MySQL. Conheço alojamentos onde se permite essa conexão e alojamentos onde não.

Se não é permitido conectar-nos remotamente escolheríamos o PhpMyAdmin, pois é uma aplicação PHP que se conecta em local e ao que se pode aceder através de uma página web, e isso está permitido pelos provedores, inclusive há muitos que já têm instalado esse software para administrar as bases de dados.

No caso de que efetivamente possamos conectar-nos remotamente com a base de dados, a nossa escolha será MyCC ou Access, que são aplicações Windows muito mais potentes e rápidas que as que utilizam interface web, como o PhpMyAdmin. É preferível usar MyCC porque está especialmente feito para conectar-se e operar com bases de dados MySQL.

Informe de **Miguel Angel Alvarez**

Mail: miguel@desarrolloweb.com

Subir uma base de dados MySQL vazia ao servidor

É normal que tenhamos uma base de dados para o nosso projecto do 0, definindo às diferentes entidades do nosso modelo de dados, os seus campos e os seus tipos.

Neste caso o mais provável é que a base de dados esteja vazia, ou bem contenha dados que introduzimos para verificar o correcto funcionamento da base. Obviamente não queremos conservar estes dados quando a aplicação esteja a funcionar no servidor...

A opção mais interessante podia ser criar outra vez as tabelas que temos em local na base de dados remota. Para o fazer temos duas hipóteses.

a) Se temos poucas tabelas e bastante simples

Podemos criá-las em remoto com alguma ferramenta como PhpMyAdmin ou MyCC

b) Se temos muitas tabelas e/ou são muito complicadas

Neste caso recomendamos fazer um backup da estrutura em local e restaura-la em remoto. Isto evitará-nos ter que voltar a criar todas as tabelas e definir todos os seus campos e tipos. Pode ser um pouco mais complicado mas sem dúvida poupará-nos tempo.

Para fazer backup da estrutura em local podemos utilizar alguma ferramenta como PhpMyAdmin, ou utilizar o comando `mysqldump` desde a linha de comandos de MS-DOS.

- [Insert textfiles into table](#)
- View dump (schema) of table

☒ Structure only
 ☐ Add 'drop table'

☐ Structure and data
 ☒ Send
 ☐ Complete inserts

☐ CSV data
 terminated by :

Ferramenta de backup de PhpMyAdmin. Está marcada a opção de extrair só a estrutura das tabelas. Se marcamos também a opção "Send", o nosso navegador criará um ficheiro de texto. Se não o fizermos só se visualizará.

O que temos de fazer neste caso é um backup da estrutura da base de dados, isto é, os "create tables" ou sentenças SQL para criar as tabelas. Seriam bastantes sentenças deste género.

```
# -----
#
# Table structure for table 'comentario'
#
```

```
CREATE TABLE comentario (
  id_comentario int(5) unsigned NOT NULL auto_increment,
  id_artigo int(4) DEFAULT '0' NOT NULL,
  comentario text NOT NULL,
  data int(14) unsigned DEFAULT '0' NOT NULL,
  revisado tinyint(1) DEFAULT '0' NOT NULL,
  nome_comentario varchar(100) DEFAULT 'Nome não especificado' NOT NULL,
  email_comentario varchar(100) DEFAULT 'Email sem especificar' NOT NULL,
  tipo tinyint(1) unsigned DEFAULT '1' NOT NULL,
  PRIMARY KEY (id_comentario)
);
```

Para restaurar estas sentenças temos opções tanto dentro de PhpMyAdmin como de MyCC. Em ambos casos o que temos de fazer é executar estas sentenças no servidor MySQL remoto. Em PhpMyAdmin temos um campo para introduzir sentenças SQL e outro para seleccionar um ficheiro de texto com todas as sentenças SQL, para executa-las uma detrás da outra. No MyCC temos um botão que nos permite abrir uma consola onde introduzir uma ou várias sentenças SQL e executa-las.

Para restaurar a tabela desde o backup composto por sentenças SQL

- [Print view](#)
- Run SQL query/queries on database [wwwdesarrolloweb.com](#) [Documentation]:

or Location of the textfile:

Introduzir uma ou várias sentenças SQL

Indicar um ficheiro com sentenças SQL

- [Query by Example](#)
- View dump (schema) of database

☒ Structure only
 ☐ Add 'drop table'

☐ Structure and data
 ☐ Send
 ☐ Complete inserts

Backup da estrutura das tabelas

A estrutura e os dados das tabelas

Para obter o backup da base de dados

Botão para introduzir sentenças SQL em MyCC



Recordamos que isto só nos servirá para subir a estrutura da base de dados e não os dados que contenham. Se desejamos subir também a informação da base de dados temos de utilizar outras estratégias que relataremos proximamente.

Informe de **Miguel Angel Alvarez**
Mail: miguel@desarrolloweb.com

Subir uma base de dados MySQL com a estrutura e os dados

Se a base de dados que queremos subir está cheia de informação e desejamos que se conserve uma vez subida a base de dados a remoto, temos de realizar um backup da base de dados e restaura-lo em remoto.

Nota: Estas recomendações pensadas para subir uma base de dados MySQL que esteja em local a uma base de dados MySQL que tenhamos contratado em remoto. Se a base origem não é MySQL estaríamos a falar de uma migração de bases de dados, mas isto será visto num [artigo mais a frente](#).

Neste caso o processo seria muito parecido ao de [subir uma base de dados vazia](#), relatado anteriormente, excepto que agora temos de extrair não só a estrutura da base de dados, mas também os registos que contém.

Para fazer isto podemos utilizar mysqldump ou PhpMyAdmin, seleccionando a opção que indica que o backup contem a estrutura e os dados (Structure and data).

A estrutura e os dados vêm num ficheiro de texto com uma serie de sentenças SQL para criar as tabelas e os insert necessários para introduzir cada um dos dados.

Para restaurar a base de dados, vamos seguir os mesmos passos como para o caso em que a base de dados está vazia, com a ajuda de uma instalação de PhpMyAdmin em remoto ou MyCC, que se conecte a base de dados contratada no servidor de Internet.

Se temos problemas para subir o ficheiro de backup da base de dados é possível que o nosso provedor de alojamento nos possa ajudar a subir o ficheiro e restaura-lo. Como o provedor dispõe dos serviços nas suas próprias instalações, tem muitas mais possibilidades para trabalhar com as bases de dados do que nós, sem temor a que as lentas comunicações pela internet nos dêem erros na restauração dos dados.

Se o nosso provedor não pode ajudar-nos, de certeza que dispõem ou nos pode indicar algum mecanismo para realizar a tarefa sem erros. Pode acontecer que algum provedor nos diga que se encarrega de tudo mas que nos exige um pagamento pelas horas de trabalho da pessoa que vai restaurar o backup da base de dados. Se não dispõe de facilidades nem sequer nisto, provavelmente seja melhor ir pedindo-lhe o dinheiro de volta porque o seu serviço quase de certeza não vai ser de qualidade.

Informe de **Miguel Angel Alvarez**
Mail: miguel@desarrolloweb.com

Migrar uma base de dados a MySQL

O último caso no que nos podemos encontrar na hora de subir uma base de dados ao nosso provedor de alojamento é que a base de dados que tenhamos criada em local vá ser utilizada em remoto. Em remoto supomos sempre que vamos utilizar a base de dados MySQL. Em local poderíamos dispor de uma base de dados Access, SQL Server ou de outro sistema de base de dados.

O processo da migração pode ser bastante complexo e como há tantas bases de dados diferentes, vai ser complicado dar uma "receita" que funcione em todos os casos. Para além da dificuldade de transferir informação entre os dois sistemas gestores de bases de dados, também influirá muito na complexidade do problema o tipo de dados das tabelas que estamos a utilizar. Por exemplo, as datas, os campos numéricos com decimais ou os booleanos podem dar problemas ao passar de um sistema a outro porque podem armazenar-se de maneiras diferentes ou nos caso dos números, com uma

precisão diferente.

Recomendações para migrar de Access a MySQL

Se a nossa base de dados anterior estava construída em Access a migração será fácil graças a que MySQL dispõe dum driver ODBC para sistemas Windows, que nos permite conectar Access com o próprio MySQL e passar informação facilmente.

Devemos indicar que se desejarmos fazer uma exportação desde Access em local a MySQL em remoto pode haver problemas porque não todos os servidores permitem as conexões em remoto com a base de dados. Se não temos disponível uma conexão em remoto com o nosso servidor de bases de dados vamos ter que mudar de estratégia.

A ideia neste último caso é instalar MySQL em local e realizar a migração desde Access em local a MySQL em local e depois ,fazer um backup da base de dados local e subi-la a remoto, tal e qual como se relatou antes.

Recomendações para migrar desde SQL Server a MySQL

A verdade é que nunca passei por este problema, mas Access também nos pode ajudar neste caso. Access permite seleccionar uma base de dados SQL Server e trabalhar desde a própria interface de Access. A ideia é que Access também permite trabalhar com MySQL e possivelmente fazendo uma pontes entre estes dois sistemas gestores podemos exportar dados desde SQL Server a MySQL.

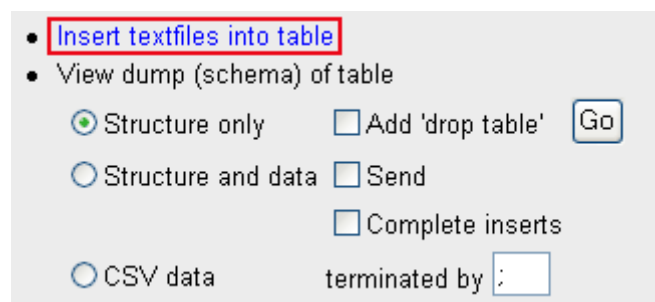
O certo é que utilizando o próprio Access de ponte poderíamos realizar o trabalho. Primeiro exportando de SQL Server a Access e depois de Access a MySQL.

Outras bases de dados e outras técnicas

Se a base de dados origem dispõem de um driver ODBC não haverá (em teoria) nenhum problema para conecta-la com Access, de maneira similar a como se conecta com MySQL. Poderíamos utilizar Access para exportar os dados, porque desde ali poderia-se aceder aos dois sistemas gestores de bases de dados

Se não temos Access, ou a base de dados original não tem driver ODBC, ou ainda se não funciona correctamente o processo e não sabemos como resolver o problema, outra possibilidade é exportar os dados a ficheiros de texto, separados por vírgulas ou algo parecido. Muitas bases de dados têm ferramentas para exportar os dados das tabelas a ficheiros de texto, os quais podem ser introduzidos no nosso sistema gestor destino (MySQL) com a ajuda de alguma ferramenta tal como PhpMyAdmin.

Para isso, na página de propriedades da tabela encontraremos uma opção para fazer o backup da tabela e para introduzir ficheiros de texto dentro de uma tabela (Insert textfiles into table)



- **Insert textfiles into table**
- View dump (schema) of table
 - ☒ Structure only
 - ☐ Structure and data
 - ☐ CSV data
- ☐ Add 'drop table'
- ☐ Send
- ☐ Complete inserts
- terminated by
-

Acedendo a este link poderemos ver um formulário onde introduzir as características do ficheiro de texto, como o carácter utilizado como separador de campos, ou o terminador de linhas, etc, junto com o próprio ficheiro com os dados, e PhpMyAdmin encarregar-se-á de todo esse trabalho de incluir eses dados na tabela.

Location of the textfile	<input type="text"/>	<input type="button" value="Examinar..."/>
Replace table data with file	<input type="checkbox"/> Replace	The contents of the file replaces the contents of the selected table for rows with identical primary or unique key.
Fields terminated by	<input type="text" value=";"/>	The terminator of the fields.
Fields enclosed by	<input type="text" value=""/> <input type="checkbox"/> OPTIONALLY	Often quotation marks. OPTIONALLY means that only char and varchar fields are enclosed by the "enclosed by"-character.
Fields escaped by	<input type="text" value="\\"/>	Optional. Controls how to write or read special characters.
Lines terminated by	<input type="text" value="\n"/>	Carriage return: \r Linefeed: \n
Column names	<input type="text"/>	If you wish to load only some of a table's columns, specify a comma separated field list.
[Documentation]		
<input type="button" value="Submit"/> <input type="button" value="Reset"/>		

Obviamente, é necessário ter a tabela criada em remoto para que se possa introduzir os dados do ficheiro de texto.

Mudança de um formato de dados a outro

Toda migração tem de ter em conta muito especialmente, como já se disse, as formas que tenham cada base de dados de guardar a informação. Temos que contar sempre com a possibilidade de transformar alguns dados como podem ser os campos booleanos, datas, campos memo (texto com longitude indeterminada), etc, que podem armazenar-se de maneira diferente em cada um dos sistemas gestores, origem e destino.

Nalguns casos possivelmente tenhamos de realizar algum script que realize as mudanças necessárias nos dados. Por exemplo para localizar os valores booleanos guardados como true/false e muda-los a valores inteiros 0 / 1, que é como se guarda em MySQL. As datas também podem sofrer alterações de formato, por exemplo, em Access aparece com o formato (dd/mm/aaaa), e em MySQL o formato é (aaaa-mm-dd). PHP pode Ajudar-nos na tarefa de fazer este script mas também Visual Basic Script para Access pode fazer estas tarefas complexas e a própria linguagem SQL, à base de sentenças dirigidas contra a base de dados.

Informe de **Miguel Angel Alvarez**
Mail: miguel@desarrolloweb.com

Todos los [direitos de reprodução e difusão](#) reservados

[Voltar](#)