

Understanding the Datasources

In *Cerbo*, dynamic content is accessed using datasources. It is a storage agnostic way to get data from :

- Databases
 - MySQL (using PDO)
- Flat files
 - XML
 - JSON

NB : XML and JSON are not currently implemented.

How to use a DataSource

For the moment you can connect to only one datasource at a time with the following code :

```
\cerbo\kernel\DataSource::getInstance();
```

This method will return a standard **DataSource** object with all the methods you need to fetch, update or delete data.

The **getInstance()** method as no parameters because it load automatically the parameters it needs from **application.ini**.

Configuring application.ini

```
[DATABASE]
DataSourceHandler=DatabaseDataSource
Connection[driver]=mysql
Connection[database]=cerbo
Connection[host]=localhost
Connection[username]=root
Connection[password]=root
```

DataSourceHandler The name of the class you want to use as a Datasource
For exemple : DatabaseDataSource

Connection An array with all the informations the DataSource need to connect to the storage.

Fetching data

To fetch data from the datasource, you have to use the **select** method :

```
$db = \cerbo\kernel\DataSource::getInstance();
$db->select(
    'mytable',
    array(
        'id' => array( '=', $value )
    )
);
```

The prototype of the method is :

```
public function select( $table, $parameters );
```

\$table The name of the table you want to fetch data.

\$parameters An array who describes all the conditions to fetch data.

The \$parameters array

The parameters array is a set of nested arrays.

Multiple conditions

If you have to filter on more than one condition, you will have to use **and** or **or** sub-array :

```
$parameters = array(
    'and' => array(
        'field1' => array( '=', $value1 ),
        'field2' => array( '=', $value2 )
    )
);
```

This will, for exemple, construct an SQL request like :

```
'field1' = '$value1' AND 'field2' = '$value2'
```

Using fetched data

```
$db = \cerbo\kernel\DataSource::getInstance();
$db->select( 'users', null );
while ( ( $user = $db->fetch() ) == true )
{
    echo '<p>' . $user['name'] . '</p>';
}
```