PHP开发中编码问题探讨

在php开发中，编码问题是一个比较困扰的问题。虽说问题不算严重，但是一旦碰到中文乱码的情况，那也是比较头疼的。

由于web开发中涉及到数据库，前端页面还有文件。这些都需要指定编码。所以说保持这几个地方的编码一致一般情况就能避免中文乱码情况，不管是在数据库存储还是在前端页面显示，都能显示正常的中文。

因此，这里有个比较关键的地方就是统一编码。一般情况下，将数据库存储编码置为utf-8。那么在往数据库里存的数据都是utf-8。当然要保证所有的中文来源都是utf-8那就需要对数据进行转码。

这里就有两个函数登场了：mb\_detect\_encoding()和mb\_convert\_encoding()。

string mb\_detect\_encoding ( string $str [, [mixed](http://php.net/manual/zh/language.pseudo-types.php#language.types.mixed) $encoding\_list = mb\_detect\_order() [, bool $strict = false ]] )

检测字符串str的编码。该函数的返回值是检测到的str的编码，当没有检测到编码的时候将返回false。

该函数的第二个参数$encoding\_list用来指定一个字符编码列表，当没有指定该参数的话将默认使用detect\_order。这个列表可以是以逗号分隔的字符串，也可以使一个数组。

$encode =mb\_detect\_encoding($v,array('GB2312','ASCII','GBK','UTF-8','BIG5','JIS','eucjp-win','sjis-win','EUC-JP'));

如果$v的值的编码是gb2312就会返回GB2312。如果是utf8就返回utf-8。

然而有时候会将utf-8误判为cp936。这样会导致转码错误。所以这就需要第三个参数

$strict 指定了是否严格地检测编码。 默认是 FALSE。

$encode =mb\_detect\_encoding($v,array('GB2312','ASCII','GBK','UTF-8','BIG5','JIS','eucjp-win','sjis-win','EUC-JP'),true);

这样就能比较准确的检测出编码，从而为mb\_convert\_encoding()转换提供一个比较准确的编码。

string mb\_convert\_encoding ( string str, string to\_encoding [, mixed from\_encoding] )

转换字符的编码。

通常要和mb\_detect\_encoding一块儿使用

$encode = mb\_detect\_encoding($v, array('GB2312','ASCII','GBK','UTF-8','BIG5','JIS', 'eucjp-win','sjis-win','EUC-JP'),true);

$str = mb\_convert\_encoding($v, 'UTF-8', $encode) //其中$encode就是我们检测出来的编码

这样就能将多数编码的字符串转化成utf-8编码了。从而在做进一步的处理。