

EZDML 常见问题

<http://www.ezdml.com>

——HUZ 2019 年 10 月

——本文以 EZDML for win 为基础，最后修订日期：20210327，对应 EZDML for win64 版本：3.07。

文档更新记录：

2009.11 表结构设计器 EZDML1.5 新版本发布，比以前改进了很多，因此重新写了个介绍。

2015.10 已经更新到 2.06 版本，决定再次整理重写此文档。

2019.10 到 2.32 版了，再把文档改改吧。

2020.04 跨平台 64 位版本出炉。

2021.03 增加 HTTP 服务和 JDBC 连接问题。

目录

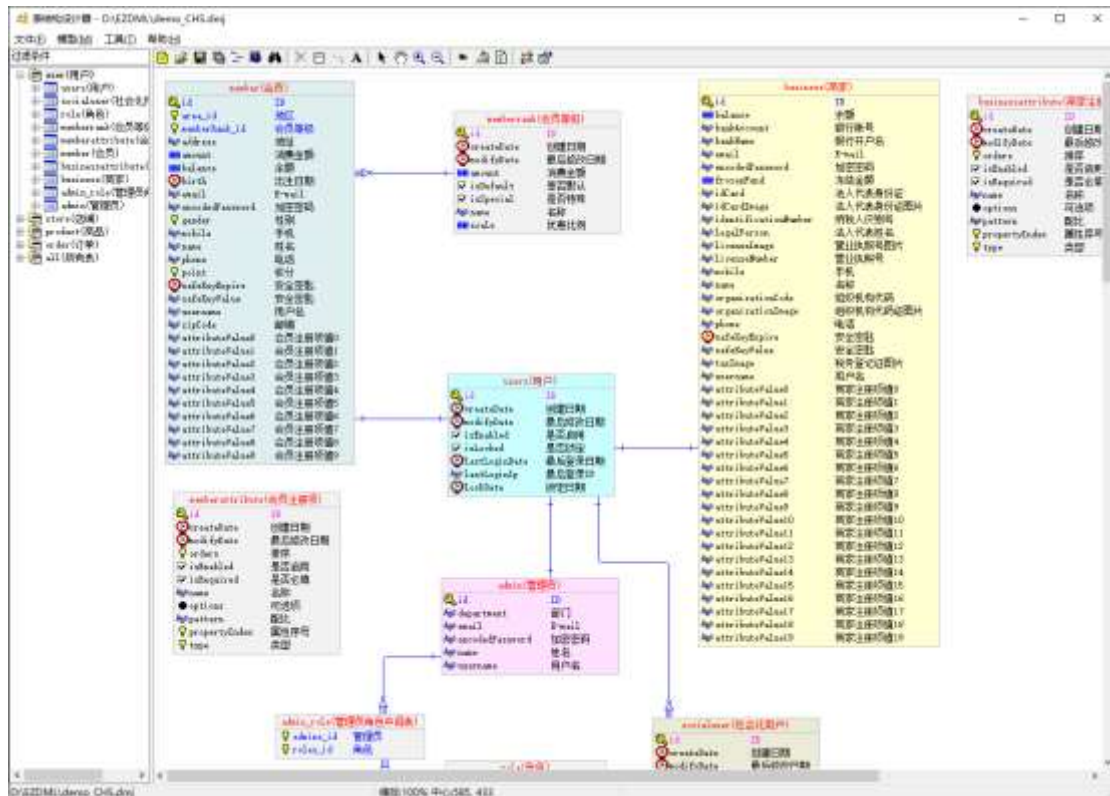
一、 EZDML 是什么东东？	3
二、 有那么多的现成的工具，为什么还要搞这个 EZDML？	3
三、 EZDML 有什么特点？	3
四、 如何快速上手？	4
五、 有绿色版吗？	4
六、 Windows 的 EZDML 有 32 位的和 64 位的，该用哪个好？	4
七、 EZDML 的描述字是什么来的？有什么用？	4
八、 为什么字段的数据类型那么少？	8
九、 默认的物理类型（如 VARCHAR2）不是我要的怎么办？	8
十、 如何定义 CLOB 长字符串类型？	8
十一、 如何定义 BIGINT 长字符串类型？	9
十二、 除了默认类型，我还需要其它的数据类型怎么办？	9
十三、 外键连接只有一对多吗？有没有一对一的？	9
十四、 在模型图中新建或复制粘贴表后，新表的位置是怎么定的？	9
十五、 是否支持多字段的组合索引、组合外键、组合主键、触发器等约束？	9
十六、 同名表有什么讲究？为什么有时会警告相同名称的表对象已经存在？	10
十七、 如何快速复制表名/字段名？	13
十八、 如何复制的方式创建新表？	13
十九、 如何复制字段到另一个表？	13
二十、 字段属性里的“编辑器界面”、“业务逻辑”是干什么用的？	13
二十一、 为什么你的表和字段属性截图有个“自定义”标签？	13
二十二、 那个大小写转换是干什么用的？	13
二十三、 那下面还有个汉字转拼音是干什么用的？	15
二十四、 如何从注释中提取逻辑名？	19
二十五、 如何对多个表批量添加或删除字段？	19
二十六、 如何对多个表或字段批量转换大小写？	19
二十七、 如何对多个表或字段批量进行某种操作？	19
二十八、 PASCAL 和 JAVASCRIPT 脚本有什么用？如何编写脚本？	19

二十九、 为什么有时输入的中文内容是乱码?	19
三十、 为什么导入或生成的 ORACLE 中文表字段是乱码?	20
三十一、 文字对象有什么用?	21
三十二、 重新排列那个是有什么原理?	21
三十三、 为什么表比较多时重新排列和重置连接的效果跟平时不一样?	21
三十四、 缩略图模式有什么用?	22
三十五、 模型图中表的背景颜色是怎么设置出来的?	22
三十六、 如何快速查找表或字段?	24
三十七、 那个 CTRL+F 查找对象又有什么用?	25
三十八、 鼠标键盘操作有没什么秘诀?	25
三十九、 如何在用鼠标矩形拉框选择? 如何只选中表而不选择线?	26
四十、 连不上 ORACLE 怎么办?	26
四十一、 还是连不上数据库怎么办? 或者其它类型的数据库连不上怎么办?	29
四十二、 生成的 SQL 我有一点不喜欢, 想小改一下, 该怎么办?	29
四十三、 生成 SQL 里那个粗略对比模式是干啥用的?	29
四十四、 如何将 DML 文件与旧版本比较?	29
四十五、 临时文件是干什么用的?	30
四十六、 EZDML 有 DMH、DMX、DMJ 三种文件保存格式, 用哪一种比较好?	30
四十七、 如果想重新加载原始的文件, 不想用临时文件要怎么操作?	31
四十八、 EZDML 有哪些配置文件? 如何设置这些配置?	31
四十九、 程序支持哪些命令行参数?	31
五十、 如何复制或导出模型图像?	31
五十一、 导出 WORD 文档格式我不喜欢, 能改不?	32
五十二、 导出的 EXCEL 格式我也想改, 要怎么弄?	32
五十三、 如何配置外部工具?	32
五十四、 字符编码工具是做啥用的?	33
五十五、 EZDML 能导入其它工具 (如 PowerDesigner) 的文件吗?	33
五十六、 HTTP 连接和 JDBC 连接怎么用?	33
五十七、 工具菜单中的 HTTP 服务是作什么用的? 跟 JDBC 服务有什么关系?	36
五十八、 EZDML 收费吗? 需要捐助吗?	37
五十九、 EZDML 是用什么开发的?	37
六十、 EZDML 会开源吗? 会有 MAC 版吗?	37
六十一、 还有其它的问题怎么办?	37

一、EZDML 是什么东东？

EZDML 是一个数据库建表的小软件，可快速的进行数据库表结构设计，建立数据模型。类似大家常用的数据库建模工具如 PowerDesigner、ERWIN、ER Studio 和 Rational Rose 等（还有很多，请原谅我了解的软件较少）的超级精简版。

EZDML 是一个数据模型创建管理的小软件，可快速的进行数据库表结构设计，建立数据模型，支持自定义脚本模板来生成代码文件。



二、有那么多现成的工具，为什么还要搞这个 EZDML？

这个。。也许当时没有现在那么多，没找到喜欢的，也许是嫌 PowerDesigner 太慢太复杂了（你觉得不慢？我现在也觉得不慢，环境不一样了），也许是某天无聊想自己写点东西练练手，时间太久，不太记得了，反正年轻时看别人写的程序都有点不顺眼。要搁现在，我可能就不会再做这种重新发明轮子的事了。

如果你正在找类似的工具，不妨试一下 EZDML。如果你的项目组已经用开了其它的工具有，且用得很好，那可以看看，但没有必要冒险作出转变。

三、EZDML 有什么特点？

有两大特点：

1. 简单。程序本身就很简单，它只需要一个 exe 就可以运行，通过编辑描述字也能很

快创建或修改表结构。

2. 脚本。支持 JAVASCRIPT、PASCAL 脚本，你可以完全控制所有表、字段，想删想改想生成什么都是可以的。
3. 免费——不好意思，这第 3 点后来想起加的。

有人可能要说，你这小 Q&A 破文档都写了几十页，你还好意思说简单？嗯，确实，你有点问倒我了。我一开始也没想到有这么多问题，应该原先是比较简单的，可能后来搞着搞着变复杂了。不过，其实不看文档也基本能用得起来的。简单只是相对的，如果有别的工具比我这个更简单，我就.....我就去抄它的。

四、 如何快速上手？

说来话长，参见官网《EZDML 快速上手》。

五、 有绿色版吗？

没有。可以自己做，安装后的文件全拷出来就行了。

六、 Windows 的 EZDML 有 32 位的和 64 位的，该用哪个好？

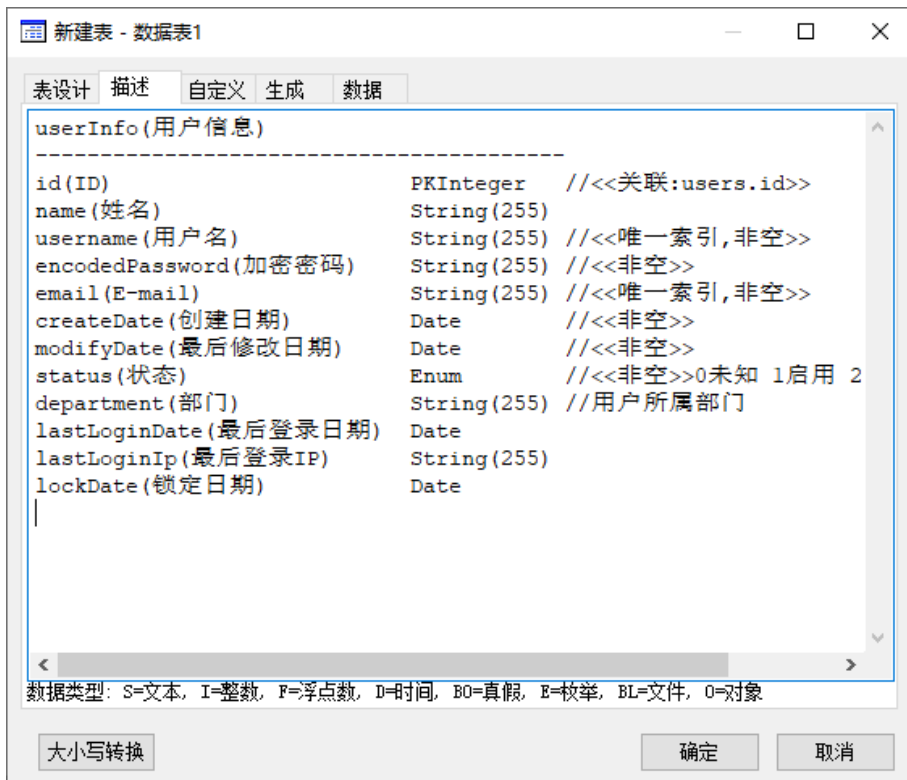
建议用 64 位的，32 位已停止更新。

七、 EZDML 的描述字是什么来的？有什么用？

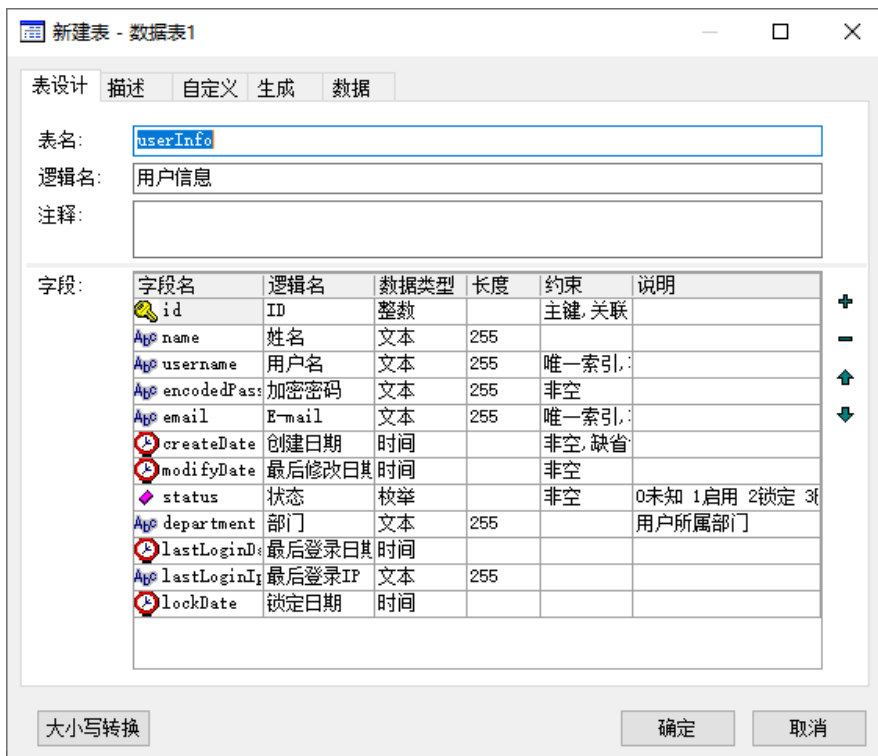
这个强烈安利大家使用。它就是一段用来描述一个表的总体结构的文本，内容很简单，简单到一看就懂。示例如下：

```
userInfo(用户信息)
-----
id(ID)                PKInteger    ///<<关联:users.id>>
name(姓名)            String(255)
username(用户名)     String(255)  ///<<唯一索引,非空>>
encodedPassword(加密密码) String(255)  ///<<非空>>
email(E-mail)        String(255)  ///<<唯一索引,非空>>
createDate(创建日期) Date          ///<<非空>>
modifyDate(最后修改日期) Date          ///<<非空>>
status(状态)         Enum          ///<<非空>>0 未知 1 启用 2 锁定 3 删除
department(部门)     String(255)  //用户所属部门
lastLoginDate(最后登录日期) Date
lastLoginIp(最后登录 IP) String(255)
lockDate(锁定日期)   Date
```

把这段文字复制下来，新建一个表，切到描述字，粘贴进去：



切换到表设计看看：



模型图结果：

userInfo(用户信息)	
id	ID
name	姓名
username	用户名
encodedPassword	加密密码
email	E-mail
createDate	创建日期
modifyDate	最后修改日期
status	状态
department	部门
lastLoginDate	最后登录日期
lastLoginIp	最后登录IP
lockDate	锁定日期

可以看到，表的信息已经基本完备了。

当然了，上面那么整齐划一的描述字是正规军，其实我来操作的话就比较随意，我会直接输入以下内容（实际上我多半会去掉英文直接用中文字段名，但这里演示还是留着吧）：

userInfo 用户信息

```
-----
id PK
name 姓名
username 用户名
encodedPassword 加密密码
email 邮箱
createDate 创建日期 D
modifyDate 最后修改日期 D
status 状态 E //0未知 1启用 2锁定 3删除
department 部门
lastLoginDate 最后登录日期 D
lastLoginIp 最后登录IP
lockDate 锁定日期 D
```

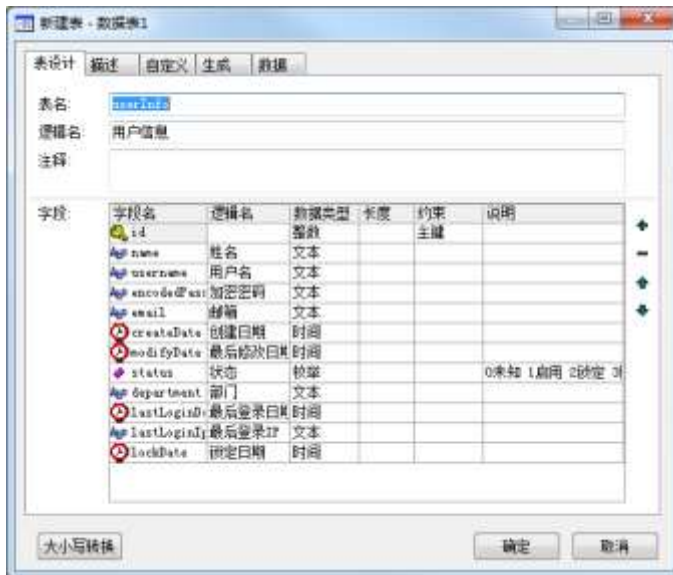
其中 PK 表示主键，主外键默认为整数类型，跟 PKI、PKInteger 是一个意思。如果主键是字符，可以写成 PKS 或 PKString。

同理，外键可以用 FK、FKI、FKInteger、FKS、FKString 表示。

其它字段如果不指定类型，则默认为字符串。大家可能发现了，我创建表经常都不指定字段长度的，因为我懒。



转到表设计看看，效果也是杠杠的：



如果想改变字段的位置、添加删除修改字段，也是 Ctrl+X、Ctrl+V 编辑描述字解决：



最终结果：

userInfo (用户信息)	
🔑 id	
🔑 name	姓名
🔑 username	用户名
🔑 email	邮箱
🔑 lockDate	锁定日期
🔑 encodedPassword	加密密码
🔑 createDate	创建日期
🔑 status	状态
### TestField1	测试添加字段

通过编辑修改描述字，可以轻松实现字段的增删改移和复制粘贴，对键盘侠们来说是非常方便的。我现在创建表和字段基本上是只用描述字，表格只是用来设置约束之类的属性，其它时候都是查看摆设做样子了。当然，如果你不喜欢用描述字，仍旧用表格也是 OK 的。

八、为什么字段的数据类型那么少？

本工具以简单设计为主，所以把常用的数据类型归纳为以下几种了：文本、整数、浮点数、时间、真假、枚举、文件、对象。比如字符串 `VARCHAR`、`VARCHAR2`、`NVARCHAR`、`CLOB`（长度超过数据库限制时会自动转用 `CLOB`），在我看来都是一个类型，只是换了个马甲而已。这样做的好处，主要是跟编写代码比较接近，另一个是跨数据库时比较方便。

大部分情况下我们会经常使用一种基本物理类型，比如常用 `varchar` 对应 `String`，这个 `varchar` 就可以默认预先设置好。偶尔需要区分多个类型时，你可以在选定基本逻辑类型后，在字段的类型名称上做补充，比如基本类型选 `String`，类型名称写 `NCHAR`。

另外还有列表、函数、事件几个类型，是为方便设计生成代码而增加的。

如果你的场景经常要区分数据库的各种物理数据类型，那 `EZDML` 可能不是最好的选择。

九、默认的物理类型（如 `VARCHAR2`）不是我要的怎么办？

如果只是个别字段，可以直接修改字段属性，设置字段的类型名称。类型名称默认为空，为空时使用默认数据类型名称。如果大多数字段都要改，可以在设置中（win32 版是在 INI 中）定义程序默认的物理类型，比如字符串你可以指定默认用 `NVARCHAR2` 类型，[参见官网《EZDML 配置文件》中的 DefaultFieldTypes 相关内容](#)。

十、如何定义 `CLOB` 长字符串类型？

上面说了，`CLOB` 属于字符串，只要定义一个嚣张点的长度，超过普通 `VARCAHR` 的限制即可。简单说，定义为 `String(99999)` 就肯定可以了，没有哪个数据库的 `varchar` 会超过 99999 的。

十一、 如何定义 BIGINT 长字符串类型？

数据类型还是选择整数 (Integer)，对于 SQLSERVER、POSTGRESQL 和 MYSQL，可通过长度定义子类型：

- 小于等于 4: tinyint
- 5、6: smallint
- 7、8: midiumint (MYSQL), integer (PQ)、int (其它)
- 9、10、11: integer (PQ)、int (其它)
- 12 及以上: bigint

——添加于 20210424，以前 ORACLE 用得多了没考虑这个问题 (ORACLE 没有专门的长整数类型)。

十二、 除了默认类型，我还需要其它的数据类型怎么办？

逻辑类型 (如浮点数) 基本上都是有的，但物理数据类型 (如 CURRENCY, DECIMAL) 可能没有。解决办法有两个：

一是在每一个字段的属性中指定类型名称；另一个办法是在设置中 (win32 版是在 INI 中) 增加自定义类型，参见官网《EZDML 配置文件》中的 CustFieldTypes 相关内容。

需要说明的是，自定义类型本质上也是修改字段的类型名称，生成 SQL 时会再替换为最终的类型。

十三、 外键连接只有一对多吗？有没有一对一的？

默认就是一对多，不支持直接设置成一对一。不过，如果你把从表的关联字段加一个唯一索引 (或者设置为主键)，则 EZDML 能自动识别它是一一对一，在画图时会画成一一对一。

十四、 在模型图中新建或复制粘贴表后，新表的位置是怎么定的？

默认就是最后一次鼠标点击的位置。如果你想新的表出现在指定位置，则新建或粘贴前在那个位置用鼠标点一下，左键右键都行。

如果在左边目录树上新建或粘贴表，则位置由程序自动指定。

十五、 是否支持多字段的组合索引、组合外键、组合主键、触发器等约束？

不支持，本工具以简单设计为主，复杂的数据库设计是不支持的 (不过，好像组合主键是支持的，我一开始也不知道，看了版本历史才发现)

作为一个悲催的程序员，混了这么多年也没混上 DBA，很惭愧，说明数据库不是我的专长。EZDML 表面上是数据库表设计工具，可实际上为数据库服务的并不深入，更多是为了搞代码、脚本、文档之类的不务正业的东东。如果你想用它来维护触发器、存储过程、表空间、同义词、分区之类的，那你很可能要失望了，可能你想要的功能没有，你不想要的、没什么暖用的东西却很多。

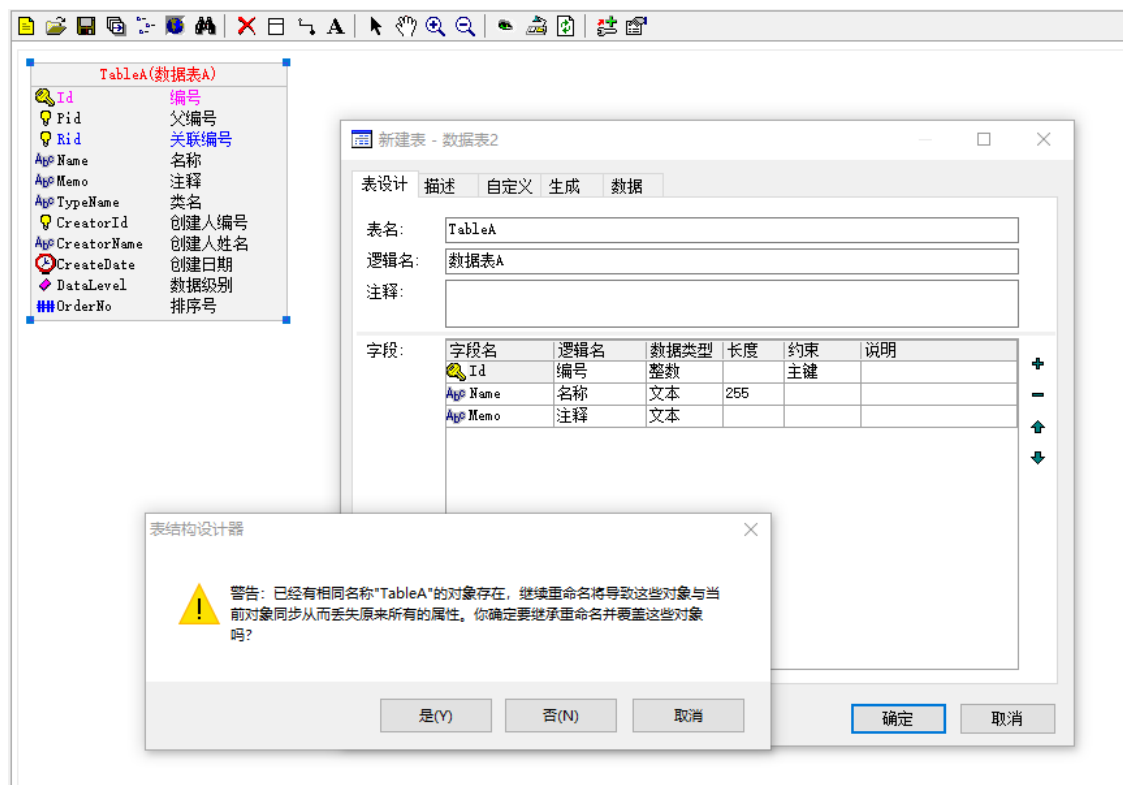
不过话说回来，EZDML 的自定义功能很灵活，如果确实需要，熟悉后你也可以自主增加你需要的内容。

20210416 注：从 3.08 版开始，支持多字段的复合索引了。虽然我自己基本不用这种复杂的索引，但也不反对别人用，有人强烈要求，就加上了，顺便解决之前多字段组合索引导入会变成普通索引的 BUG。

十六、同名表有什么讲究？为什么有时会警告相同名称的表对象已经存在？

为方便设计管理，在一个 DML 模型文件里，同一个名称的所有表（不管是不是在一个模型图里）会认为是同一个，修改完一个的属性后会自动同步到其它同名表（注：界面操作的才会，用脚本改的不会同步），改一个就等于改了全部。于是问题就来了。

假设我们已经有了一个叫“TableA”的表，有 11 个字段；这时我们再新建第二个表，也取名为“TableA”，但只有 3 个字段。请问：创建完第二个表后，TableA 应该有几个字段？嗯，不好回答吧，说 11 个或 3 个都会有人有意见。所以，第二个表保存时程序会警告有同名表存在，你是不是搞错了？



你仍然保存的话，点“是”，新表就会覆盖旧表，最终答案是 3 个字段，有图有真相：



嗯？左边怎么还是 11 个？表与表之间的信任呢？表急，刷新一下就真相了：



这时两个表就结盟了。我们随便选一个 TableA，加一个字段，删一个字段，改一个字



保存，第二个表也会自动同步变化：

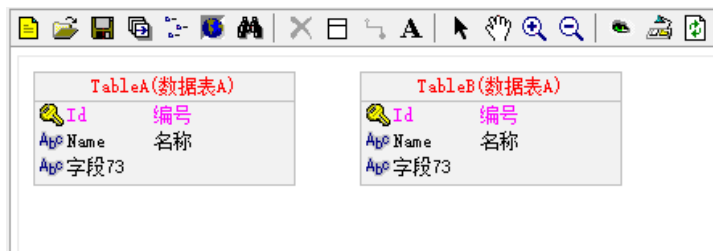
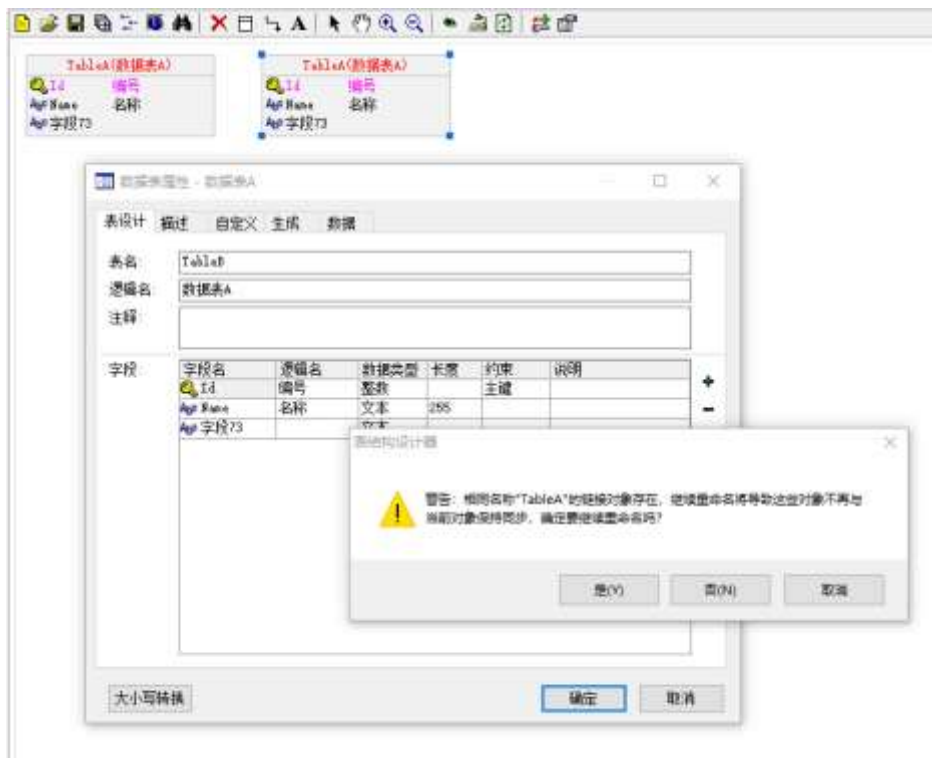


很和谐吧。

需要注意的是，EZDML 同名表在整个文件里都是结盟的，不在同一个模型里也会同步。

不过，又有另一个问题冒出来了，如果这时我把右边那个 TableA 改名为 TableB，会怎么样呢？左边那个会跟着改名吗？

实际上并不会，改完后 A 是 A，B 是 B。但程序会给你一个警告，因为这意味着 B 背弃了盟约，从联盟中独立出来了：



需要说明的是，在界面上操作时会自动同步内容，用脚本修改则并不会，需要手工调一下 SyncTableProps 方法执行同步，或者对所有同名表执行同样的操作。

十七、 如何快速复制表名/字段名？

左边目录树：选中任何对象，按 **CTRL+E** 复制名称（复制高亮内容，显示什么就复制什么，如果有逻辑名，也会一起复制）。

右边模型图：

- 选中整个表，不选中表名和字段名，这时按 **CTRL+E** 会复制出一个 `select` 语句
- 选中表名，按 **CTRL+E** 复制表名（复制高亮内容，显示什么就复制什么，如果有逻辑名，也会一起复制，切换物理/逻辑视图可以控制复制的内容，下同）
- 选中字段，按 **CTRL+E** 复制字段名

十八、 如何复制的方式创建新表？

那个，你可以复制粘贴完后改名（会有同名文件链接断开的警告）；或者，你先粘贴到记事本里（复制的内容其实是 XML 文本），改好名再粘贴回去；或者，你先新开一个 EZDML 粘贴，改名，再复制粘贴回来。

十九、 如何复制字段到另一个表？

有几个办法：

- 在左边目录树中选择一个或多个字段，鼠标右键复制粘贴（不同的文件之间也适用，3.07 版开始支持 **Ctrl+X Ctrl+C Ctrl+V** 键盘操作）
- 在左边目录树中选择一个或多个字段，鼠标拖动到另一个表里
- 在表一的描述字中复制需要的内容，到表二的描述字里粘贴

二十、 字段属性里的“编辑器界面”、“业务逻辑”是干什么用的？

没什么用。本来是隐藏起来了的，想准备给自己设计通用界面和程序用的，但我这么多年也就一直停留在“想”的阶段，没有实践；后来有网友想要，就开放出来了。对我来说，对大部分人来说，它们目前仍然是没暖用的东西。

二十一、 为什么你的表和字段属性截图有个“自定义”标签？

嗯，有一个用脚本自己生成一个属性页的功能，需要改 INI 才能开启（x64 版可以直接设置）。这也是一个对我、对大部分人都没什么暖用的功能，但是我觉得它有意思、有挑战性，就加上了。

二十二、 那个大小写转换是干什么用的？

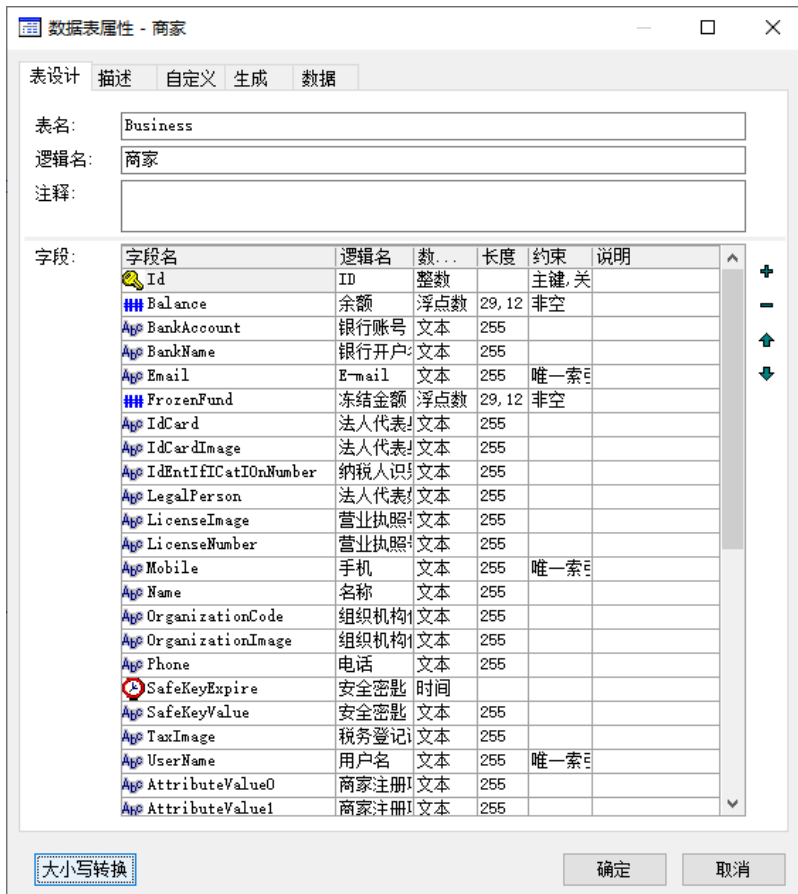
顾名思义就是大小写的互转了。从数据库导进来的表一般都是全大写或不分大小写的，有时比较难分辨，不利于查看。例如商家表生成到 ORACLE 数据库后，再从数据库导进来时，会变成全大写：



很不好看，这时可以点左下方的“大小写转换”按钮，弹出菜单如下：



执行“自动大小写”，系统处理的效果如下：



“自动大小写”的意思是自动对表名和字段名进行首字母大小写分词处理。例如 MEMBERRANK 会处理为 MemberRank。处理过程是根据字典文件 dict.txt 和 MyDict.txt 进行简单的匹配，效果并不能做到 100% 准确，如果对处理结果不满意，可修改字典文件增加你需要的词语。

二十三、那下面还有个汉字转拼音是干什么用的？

比如有这么一个表：



全是中文，生成 C++ 代码是这个样子的：

```
class 会员
{
private:
protected:
    int ID;
    char * 姓名;
    char * 电话;
    Date 出生日期;
    char * 邮箱;
    char * 地址;

public:
    会员(void);
    ~会员(void);

    int GetID(void);
    void SetID(int);
    char * Get 姓名(void);
    void Set 姓名(char *);
    char * Get 电话(void);
    void Set 电话(char *);
    Date Get 出生日期(void);
    void Set 出生日期(Date);
    ...
};
```

这个就比较令人无语了，大家习惯了代码是要用英文的。不过我英文不好，或者想偷懒，

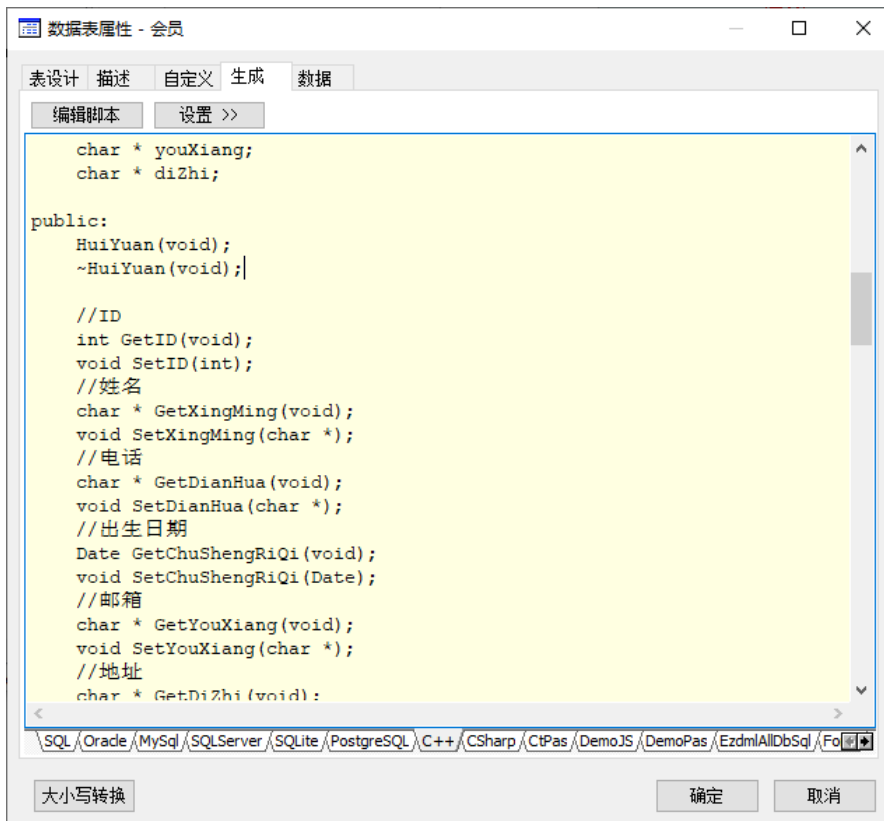
于是我先执行一下“名称转为逻辑名”，将名称复制到逻辑名上：



再执行“汉字转拼音”，将名称转成伪英文：



再来看生成的代码，是不是顺眼多了：



```

/* 会员 */
class HuiYuan
{
private:
protected:
    int ID;
    char * xingMing;
    char * dianHua;
    Date chuShengRiQi;
    char * youXiang;
    char * diZhi;
public:
    HuiYuan(void);
    ~HuiYuan(void);

    //ID
    int GetID(void);
    void SetID(int);
    //姓名
    char * GetXingMing(void);
    void SetXingMing(char *);
    //电话
    char * GetDianHua(void);
    void SetDianHua(char *);
    //出生日期
    Date GetChuShengRiQi(void);
    void SetChuShengRiQi(Date);
    //邮箱
    char * GetYouXiang(void);
    void SetYouXiang(char *);
    //地址
    char * GetDiZhi(void);

```

```
//出生日期
Date GetChuShengRiQi(void);
void SetChuShengRiQi(Date);
...
};
```

另外，脚本中也可以用 `ChnToPY` 这个函数直接转汉字为拼音。

二十四、 如何从注释中提取逻辑名？

有几种办法：

1. 从数据库导入时，勾选“注释转为逻辑名”
2. 双击表，弹出表属性对话框，点左下角的“大小写转换”，选择“注释转为逻辑名”
3. 在左边目录对上，选择一个或多个表（选字段也是可以的），然后在右键菜单中执行“大小写转换|注释转为逻辑名”。
以上 1-3，请注意提取的算法是我随便写的，可能不一定合你意。
4. 最高境界：自己写脚本处理，参见《EZDML 脚本参考》

二十五、 如何对多个表批量添加或删除字段？

有两种方式，推荐第 1 种：

1. 在左边目录树中选择多个表，右键菜单执行批量添加或删除字段；
2. 使用脚本：在模型图中右键执行脚本，编写脚本完成批量添加或删除字段，参见《EZDML 脚本参考》。

二十六、 如何对多个表或字段批量转换大小写？

请参考上一问题的答案。

二十七、 如何对多个表或字段批量进行某种操作？

请考虑用一下脚本，用过几次后，你可能就不需要问这类问题了，参见《EZDML 脚本参考》。

二十八、 PASCAL 和 JAVASCRIPT 脚本有什么用？如何编写脚本？

说来话长，简单地说，它们可以让你彻底掌控模型中的一切。参见《EZDML 脚本参考》。

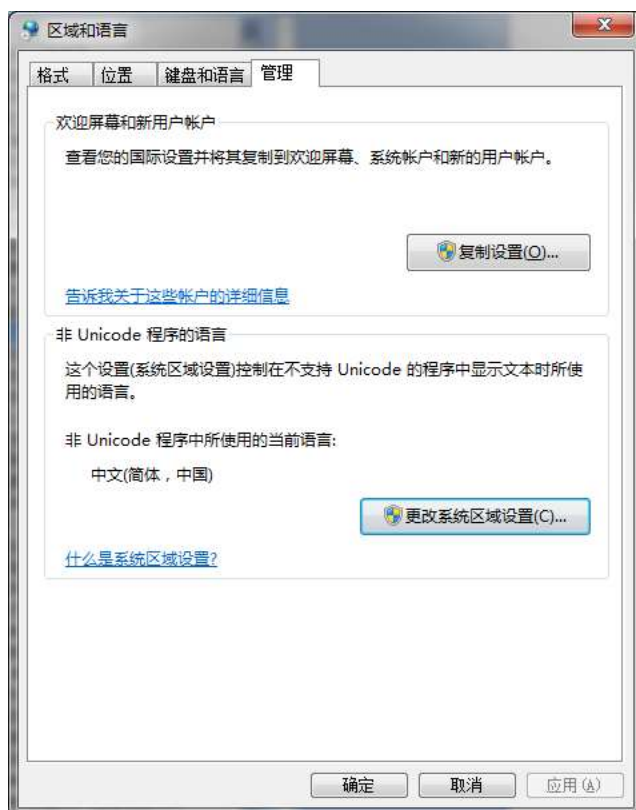
二十九、 为什么有时输入的中文内容是乱码？

这问题应该是在旧的 win32 上会出现，新版 x64 应该没这个了，升级即可。

如果非要 win32，解决办法一般有以下一些：

- 如果是英文或其它非中文的系统，请在控制面板中添加中文语言的支持

- 在区域和语言里设置非 Unicode 程序的语言为中文（WIN7 和 WIN10 都有这个设置的）



- 如果有多个输入法，可切换一下输入法试试
- 任务栏输入法图标如果是英文（EN），说明是英文键盘，请切换为中文键盘（输入法），设置中文键盘（输入法）为首选
- 如果是 WIN10 系统，可试着移除英文键盘或美式键盘，只留下中文键盘（输入法），WIN10 的中文输入法是已经包含了英文输入的
- 如果语言列表里有多个，确保中文为首选，或者干脆删除其它语言
- 尽量还是放弃 win32，改用 win64 吧

三十、为什么导入或生成的 ORACLE 中文表字段是乱码？

这是因为数据库的字符集和 ORACLE 客户端（OCI）的字符集不匹配导致的，一般需要修改客户端的字符集（OCI 就是麻烦，JDBC 就没这个烦恼）。这方面网上有很多文章说明了，以下几个 SQL 有助于你查询数据库字符集：

数据库：

```
select *
  from nls_database_parameters
 where PARAMETER in ('NLS_LANGUAGE', 'NLS_TERRITORY', 'NLS_CHARACTERSET')
```

客户端：

```
select *
  from nls_instance_parameters
```

```
where PARAMETER in ('NLS_LANGUAGE', 'NLS_TERRITORY', 'NLS_CHARACTERSET')
```

当前事务:

```
select *  
  from nls_session_parameters  
 where PARAMETER in ('NLS_LANGUAGE', 'NLS_TERRITORY', 'NLS_CHARACTERSET')
```

或:

```
select userenv('language') from dual
```

字符集编码改数据库的不太现实，一般都是改客户端的。

修改客户端字符集的最简单的办法，是设置环境变量 NLS_LANG，其组成规则为 NLS_LANGUAGE_NLS_TERRITORY.NLS_CHARACTERSET，如：

```
NLS_LANG=SIMPLIFIED CHINESE_CHINA.ZHS16GBK
```

注：3.08 版后可以在设置里直接修改 NLS_LANG，程序启动时自动将其写入环境变量。

Windows 下也可以改注册表（你最好是熟悉 oracle 注册表机制）：

```
HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_Oracle_homeXXX\NLS_LANG
```

三十一、 文字对象有什么用？

你觉得有什么用就有什么用，可以装饰一下吧，当成表的注释；对我没什么用。这东西应该是以前遗留下来的，请注意它是以前以表的形式保存的，只是它的类型 TypeName 等于“TEXT”。

另外，如果你有一些额外的 SQL 非要在生成数据库时加上，也可以用文字对象，勾选“内容为 SQL”，把 SQL 写在里面即可。

三十二、 重新排列那个是有什么原理？

作用很明显，表很多的话能自动布局是方便很多。

原先布局的原理就是随机选一个表，为它找一个位置，方法是从左到右从上到下，然后再为跟它相连表找位置。

最近我改了算法，先找到连线最多的表（有多个则按随机顺序取），把它放在中间；然后找跟它相连的连线最多的表（有多个则按随机顺序取），全放在它的附近；然后为这些表叠代地找相连子表放在附近。改进算法后的效果好了很多。

20200111：改进了自动布线算法，尽量避免被表对象遮挡，以及跟其它线重叠。

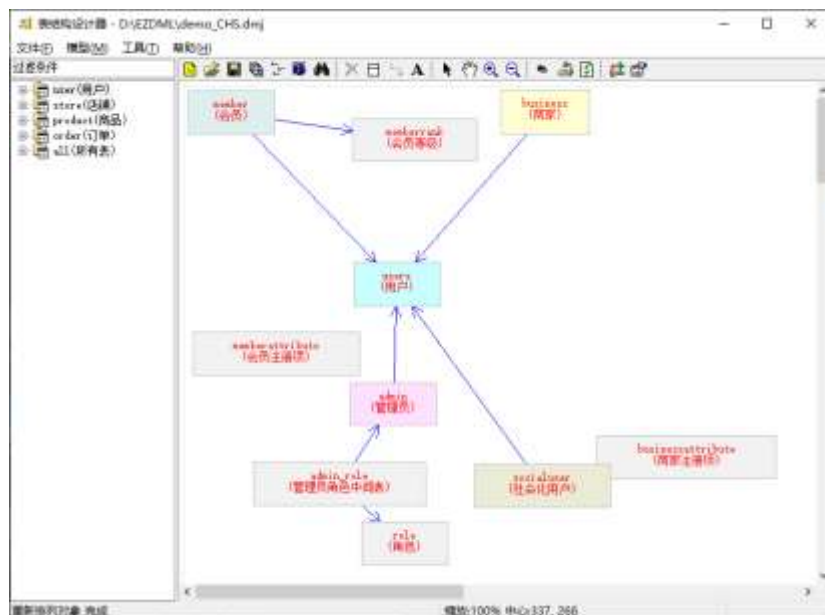
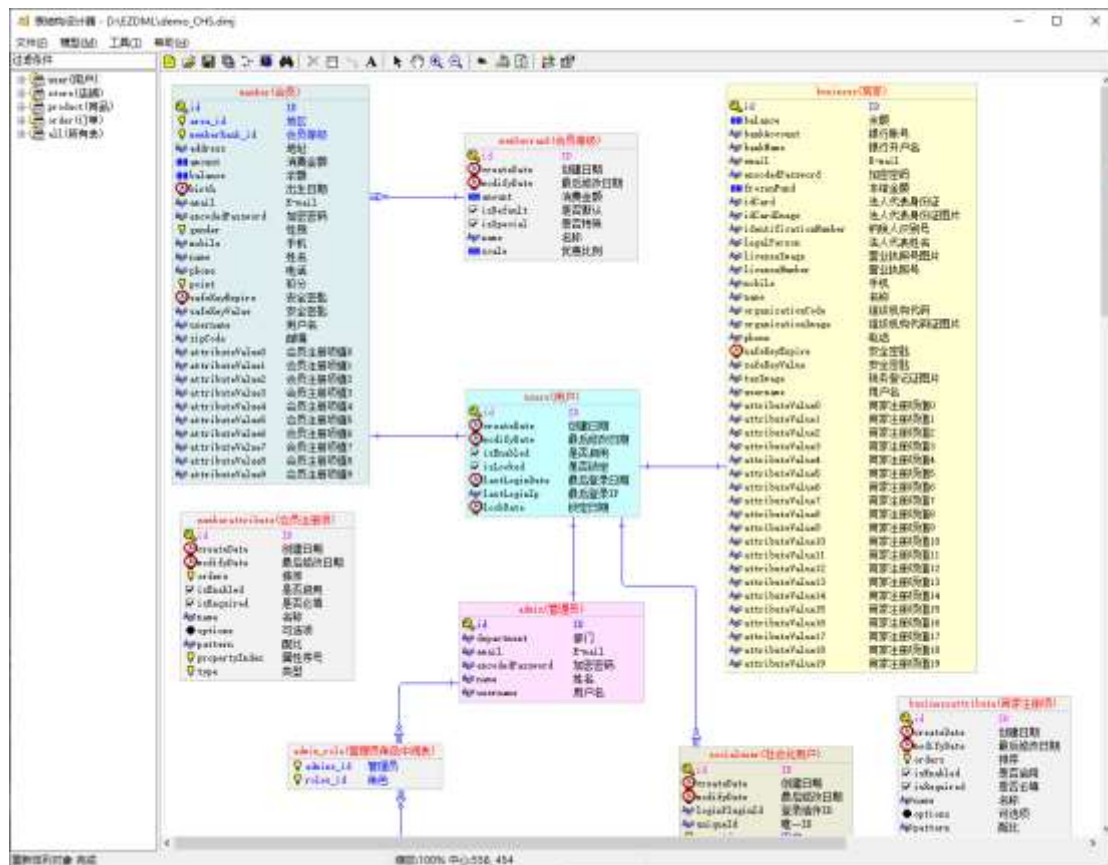
三十三、 为什么表比较多时重新排列和重置连接的效果跟平时不一样？

由于重新排列的算法写得不好，运行效率较差，因此，模型中的表超过一定数量（默认 500 个）时，EZDML 将会采用旧的简化算法快速处理。

一般人并不会在一个系统里用很多表。但如果你不是一般人，如果你在表很多时仍然想要“高级”的排列布局算法，那你可以修改 INI 配置（x64 版直接改设置），将 HugeModeTableCount=500 的这个默认限制数提高。

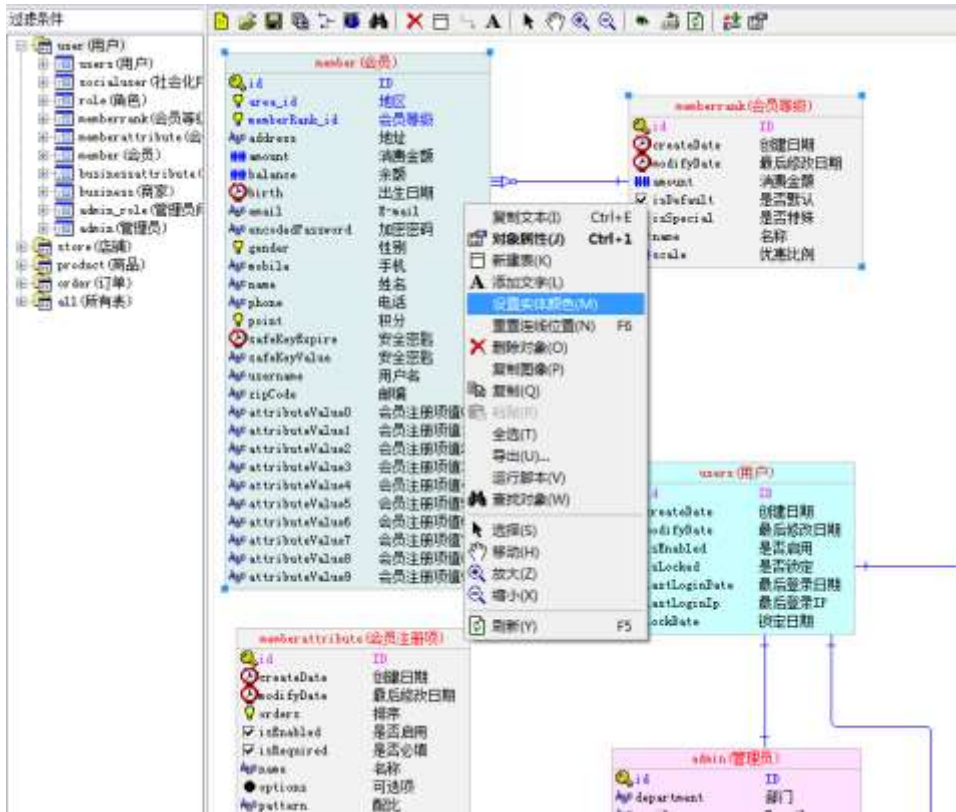
三十四、 缩略图模式有什么用？

可能表多的时候方便查看吧。还是那句话，你觉得有什么用就有什么用。截两个图直观对比一下：

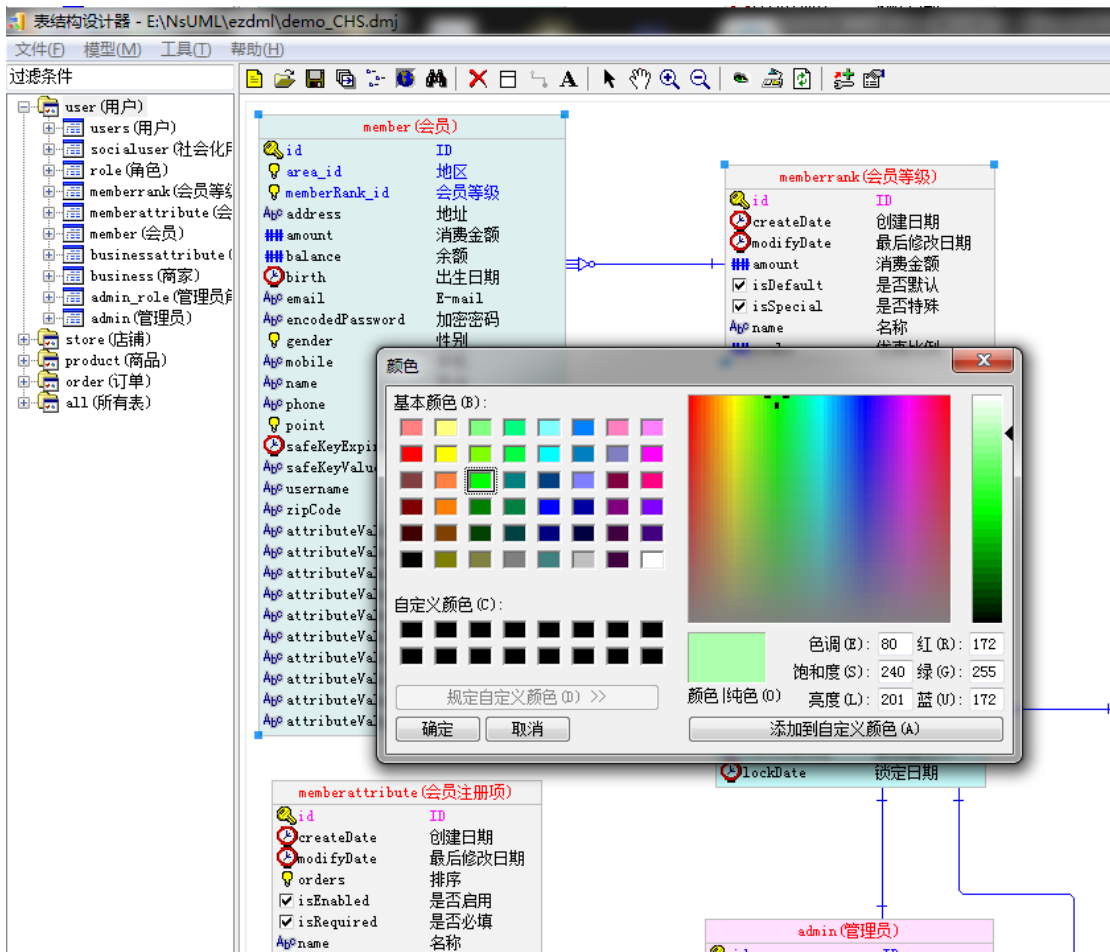


三十五、 模型图中表的背景颜色是怎么设置出来的？

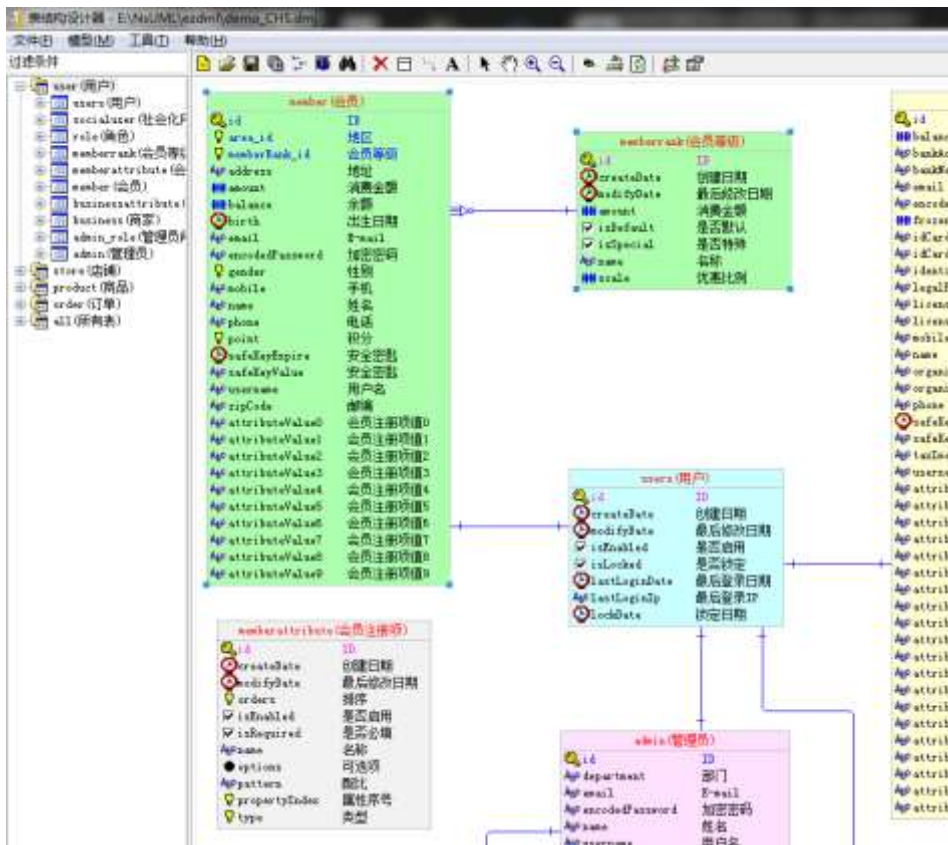
选择一个或多个表，鼠标右键弹出菜单，执行“设置实体颜色”命令：



建议选择一个相对浅一点的颜色：

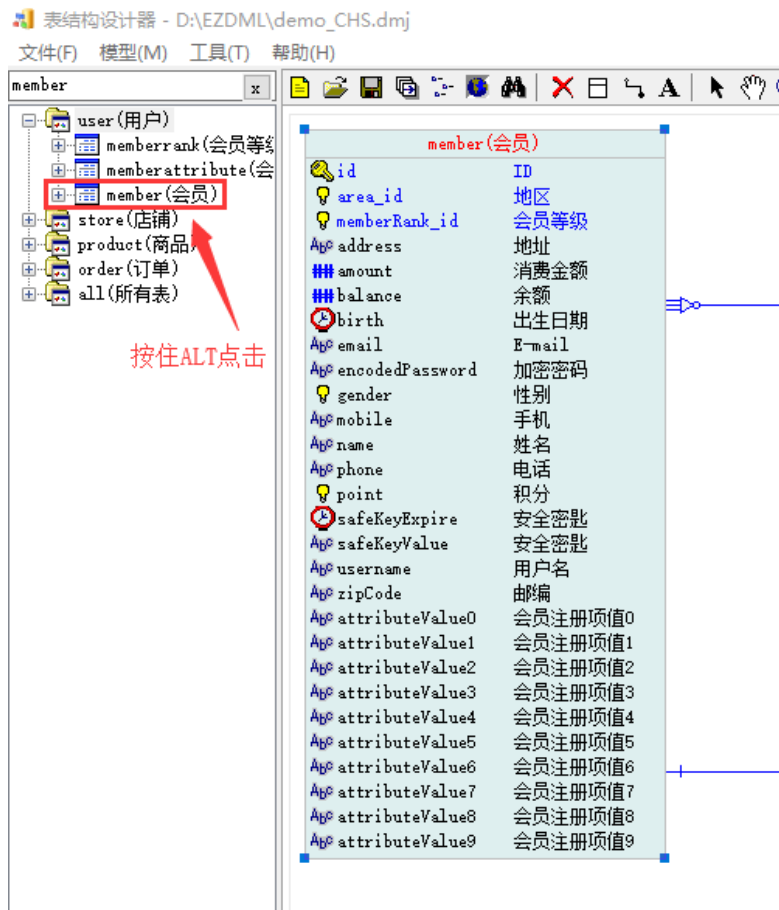


效果:



三十六、 如何快速查找表或字段？

在左上角查找框中输入表名或字段名（按 ALT+D 可快速切换焦点到查找框），找到对应表后，按住 ALT 键用鼠标点它（2020 年的新版本有右键定位菜单，不麻烦的话可以用它），就能快速跳到对应的图形上。



三十七、那个 CTRL+F 查找对象又有何用？

没什么用，之前是用它来查找对象，有快速查找功能之后，基本上就废了。

不过这功能写得不错，自我感觉良好，能查找注释，多线程，支持模糊查找，空格分隔关键词，花了我不少时间，总之就是用了高精尖的技术做了一个没用的东西，舍不得删，就留着了。

三十八、鼠标键盘操作有没什么秘诀？

模型图中可以进行平移、缩小、放大、选择对象、拖动等操作。模型图中可以用键盘鼠标进行以下操作：

- 操作前，请确认输入焦点在模型图上，可用鼠标点击模型图获取焦点
- 按加减号放大缩小，上下左右方向键平移
- 鼠标滚轮默认是上下滚动，按住 SHIFT 键则左右滚动，按住 CTRL 键则放大缩小
- 默认为选择模式，这时可用鼠标左键选择和拖动对象，拖动空白处则平移视图（2.4版后），按住 CTRL 键拉框选择，按 CTRL+A 则选择全部；选中对象后可用鼠标左键拖动其到所需位置
- 任何时候按住 ALT 键，即可用鼠标左键拖动视图进行平移操作，此时也可以双击鼠标左键进行快捷缩放（双击一次“显示全图”，再次双击则“缩放至 100%”）

- 在对象上双击鼠标左键可查看编辑其属性：选中表则显示表属性，如果同时选中了字段，则显示字段属性（注意：如果视图缩得较小，双击时会先执行“缩放至 100%”，如仍要显示属性，可按住 CTRL 键）；如果什么都没选中，则进行快捷缩放（双击一次“显示全图”，再次双击则“缩放至 100%”）
- 按 R 复原视图，按 F 为缩放至最佳大小；如果选中了对象，则这两个操作会围绕选中对象进行
- 按 Z 进入放大模式，按 X 进入缩小模式，按 A 进入缩放模式（从左向右拉框为放大，从右向左为全图），按 S 恢复选择模式
- 按 F6 重置连接，按 CTRL+E 复制选中对象的文字，CTRL+I 显示对象属性

按 F4 可以快速切换全图和复原；如果选中了对象，则复原视图时会将中心切到选中对象。这个 F4 是全局的，可以在左边树中使用，不需要获取焦点，所以特别说明一下。

左边树形列表中，可按住 SHIFT、CTRL 键并用鼠标单击来选择多个表或字段进行复制、粘贴；直接拖动可以进行排序。

按 Alt+D 可将输入焦点切换到左上方的“过滤条件”中，输入内容可查找并列符合条件的对象，按回车切到目录树，按 ESC 退出。

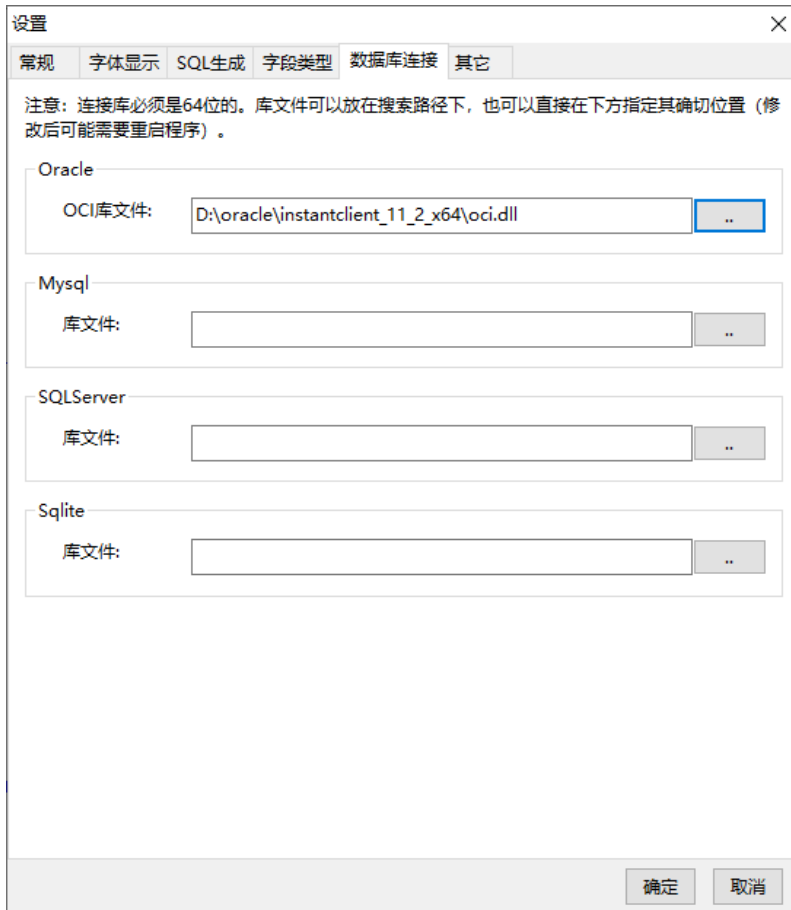
三十九、 如何在用鼠标矩形拉框选择？如何只选中表而不选择线？

新版本默认拖动空白处是平移视图，按住 CTRL 键可拉框选择，拉框开始后，按住鼠标左键不放，松开 CTRL 键，再按住 SHIFT 键就可以只选择表不选线了。

四十、 连不上 ORACLE 怎么办？

旧的 Win32 版需要 32 位的 ORACLE 客户端，mac64/linux64/win64 版则需要 64 位的。

目前基本上都是用 64 位的 ORACLE，因此使用 64 位的 EZDML 一般都应该 OK 的，系统会自动搜索 OCI 库，如果找不到 OCI 库，可在设置里指定一下。



以下回复只对 EZDML for win32 有效，如果你非要用旧的 win32 版可以看看：

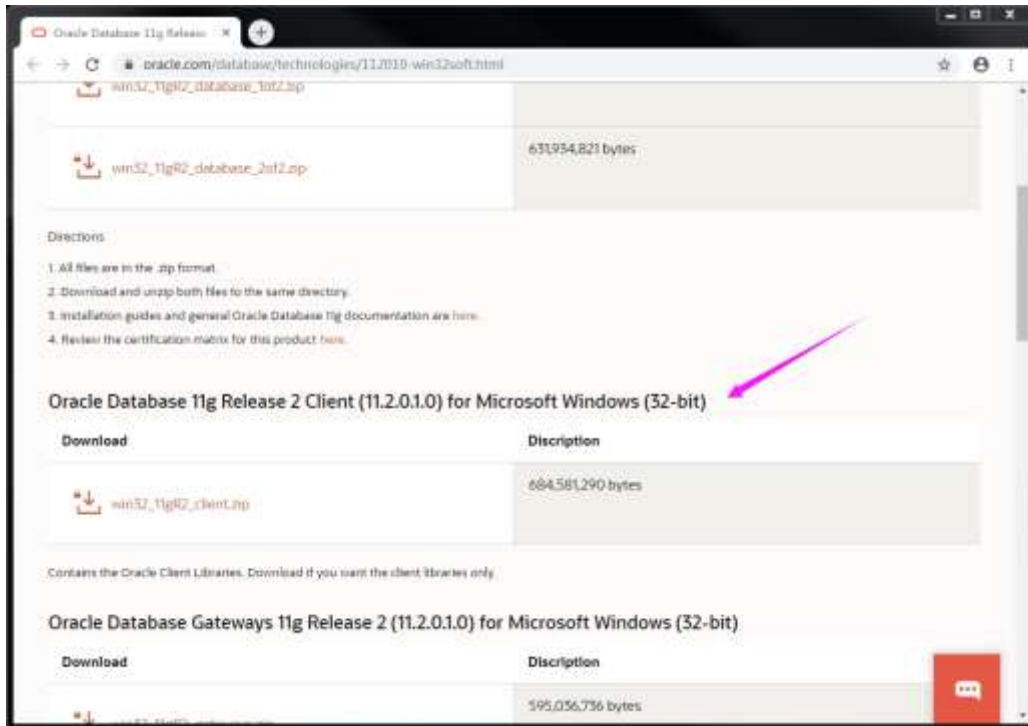
一般来说，连不上最大的可能就是 ORACLE 客户端（oci.dll 及相关文件）的版本不对。EZDML 需要 32 位的 oci.dll 及相应的支持文件，而目前大部分 oracle 安装程序都是 64 位，因此需要单独下载安装。两个办法：

A. 下载 32 位 windows 的 oracle client 并安装（必须 32 位的，数据库自带的 oci 一般是 64 位的，所以不能直接用，要另行安装）；

B. 下载 32 位 windows 的 oracle instantclient，并按说明修改 ezdml.INI 指定 OCIDLL 路径。

另外很重要的一点，是 oracle client 版本号要跟数据库服务匹配，否则可能会有莫名其妙的错误。比如 11GR2 的 32 位的 oracle client 是这个：

<https://www.oracle.com/database/technologies/112010-win32soft.html>



oracle client 包含功能较全，所以比较大，一般有几百 MB，需要安装、配置，相对麻烦。如果不想那么麻烦，可使用 oracle instant client，绿色环保，一般只有几十 MB。

32 位的 instant client 目前在这个位置可以找到：

<https://www.oracle.com/technetwork/cn/topics/winsoft-095945-zhs.html>



注意选择正确的版本号。

四十一、 还是连不上数据库怎么办？或者其它类型的数据库连不上怎么办？

那个，内事不决问百度，外事不决。。自己想办法。

我自己平时常用的是 ORACLE，其它除了 ODBC（win32 版的 SQLSERVER 本质上也是 ODBC，x64 版已经没有 ODBC 了）就是通过 EZDML 主目录那几个 DLL 访问的了。肯定都是试过的，但每个人的环境有区别，只能说在我机上测试通过了。

四十二、 生成的 SQL 我有一点不喜欢，想小改一下，该怎么办？

编写 PASCAL 脚本接管全局事件，在生成表、字段 SQL 的地方，程序会将当前表、字段、数据库比对的信息、默认生成的 SQL 传给你，你可以为所欲为，之后再新的结果返回即可。参见《EZDML 脚本参考》。

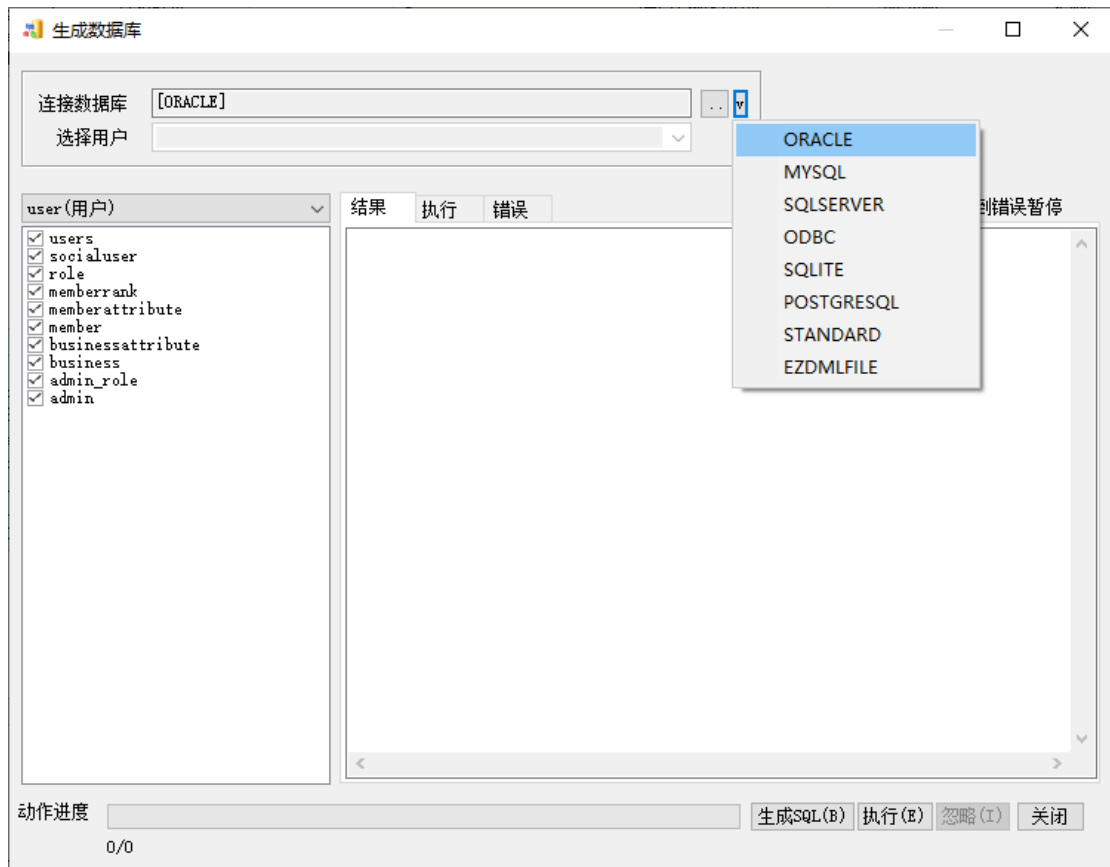
四十三、 生成 SQL 里那个粗略对比模式是干啥用的？

在与数据库模型进行对比时，我们经常需要判断两个表或两个字段是否相同，如果相同就不生成相应的 alter table 之类的 DDL 修改 SQL。正常情况下是要求精确匹配，包括名称、类型名、长度、精度、缺省值、注释、是否可为空、是否主外键、是否有索引等，任何一个属性不同都可能产生 DDL SQL。

但有时候我们只想把缺少的表和字段补上，或者想比较一下大体变化，这种精确对比并不是必须，反而会导致生成大量的不想要的 SQL，造成干扰。这时可选择粗略对比模式，它只比较名称和逻辑类型，两个相同就通过对比不再处理了。这种情况下，基本上只有你新添加的表和字段才会生成 SQL。

四十四、 如何将 DML 文件与旧版本比较？

有时我们想比较一下新版本与旧版本模型之间的差异。正常情况下，我们是和数据库比较生成差异 SQL。但是，EZDML 也提供了与另一个 DML 文件比较的功能，就是下图这个小箭头的下拉菜单里了：



先选择一个数据库类型，再选择最下面那个 EZDMLFILE，选择要对比的文件，生成 SQL，即可。

四十五、临时文件是干什么用的？

以前，EZDML 保存的是 XML 格式，关闭时总要问是否要保存，另外打开 XML 文件速度也很慢（二进制的格式快，但显然不方便维护）；后来嫌烦，就直接在关闭时保存一个二进制临时文件到源文件旁边，下次打开时就直接加载这个临时文件，速度杠杠的，世界也清静了。再后来，觉得临时文件放在源文件旁边不太妥，就将它移到系统临时文件目录了，同时增加了定时保存功能，将保存历史记录也保存到了临时文件夹。所以，你可以在临时文件夹中找到历史文件和最近一次退出时的文件。

请注意，这些临时文件和历史文件是不会主动删除的，需要的话你得手工删除。

四十六、EZDML 有 DMH、DMX、DMJ 三种文件保存格式，用哪一种比较好？

推荐用 DMJ 格式，它其实是 JSON 文本，可用记事本打开来修改。

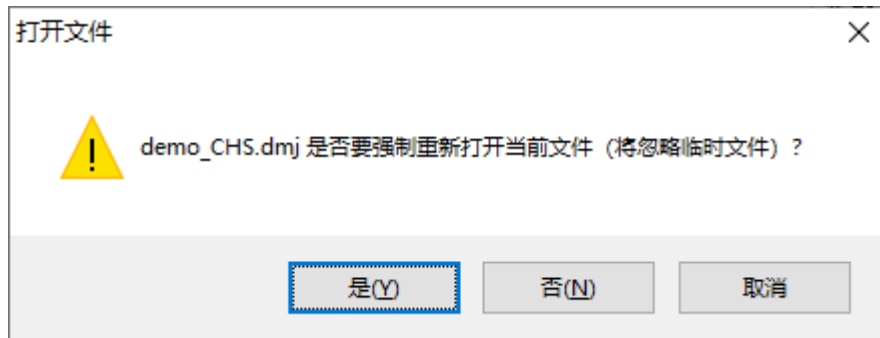
DMH 是二进制格式，速度快但不好维护。

DMX 是 XML 格式，以前没有 JSON 的时候用它。

注：所有数据文件（dmh dmj dmj），EZDML2.4 版（2020 年 1 月 27 日）以前是 GBK 编码，之后是 UTF8 编码。

四十七、 如果想重新加载原始的文件，不想用临时文件要怎么操作？

可以按 ALT+1 重新打开文件（文件菜单里有这个命令），第一次打开文件时是用临时文件的，而已经打开的情况下再次重新打开，程序就会提示是否要跳过临时文件了：



你也可以在 INI 配置中永久禁用临时文件，参见《EZDML 配置文件》。

四十八、 EZDML 有哪些配置文件？如何设置这些配置？

说来话长，参见官网《EZDML 配置文件》。

四十九、 程序支持哪些命令行参数？

支持传一到两个文件名作为启动参数：

- 如果传一个参数，则参数可以是一个 DML 模型文件名（dmx、dmh 或 dmj）由程序直接打开，也可以是一个脚本文件名（js 或 pas）来加载并执行
- 如果传两个参数，则参数 1 应为一个 DML 模型文件名（dmx、dmh 或 dmj），参数二为脚本文件名（js 或 pas），程序将在打开参数一模型文件后再加载执行参数二的脚本

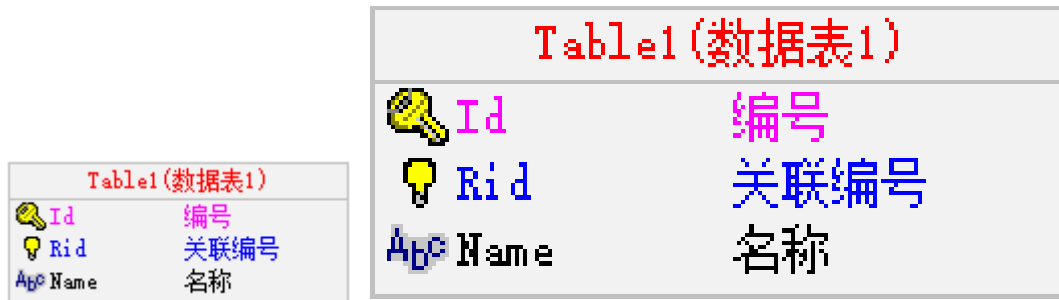
五十、 如何复制或导出模型图像？

选择要复制或导出的对象（不选择则默认全部），右键菜单中选择“复制图像”，可以将图像复制到剪贴板（默认为 PNG 图格式，如果需要 WMF 格式，可按住 SHIFT 键）；右键菜单选择“导出”，然后选择保存类型为 BMP、JPG、PNG 或 WMF 格式（注：x64 版不支持 WMF 格式）。

注：发现 WINDOWS7/10 自带画笔直接打开 WMF 文件是残缺的，不知出了什么问题，以前不是这样的；用我自己写的工具能打开；好消息是目前 WORD 仍支持粘贴 WMF 格式。

矢量图在放大时是有优势的，我们可以简单作个对比。

下面左边是直接复制图像（PNG 格式）后粘贴，右图是粘贴后拉大：



1 比 1 乍一看没什么，放大一下就露出马脚了。

下面左边是按住 SHIFT 键复制图像（WMF 格式）后粘贴，右图是粘贴后拉大：



果然吧，没有对比就没有伤害。

五十一、导出 WORD 文档格式我不喜欢，能改不？

理论上是可以的，WORD、HTML、MHT 都可以改，实际上行不行还得看情况。WORD 文档其实是 MHT（MIME）格式，是由脚本生成的。

Templates 目录下有下 export_doc_html.ps_ 的脚本文件，它可以生成 HTML 和 MHT 两种格式，导出 WORD 是用了其中的 MHT 格式。你只要看懂了它，基本上就可以随意改了。也不难看懂的，信我没错，除了 <% %> 里的脚本，剩下的全是 HTML。原始代码是我新建一个 WORD 文档后另存 HTML 搞出来的。

五十二、导出的 EXCEL 格式我也想改，要怎么弄？

—(win32 版)改不了，这个是硬核编程出来的。要不你考虑下改 HTML 输出 TABLE 吧？或者自己写一个自定义工具脚本，流程我都帮你想好了，用脚本把模型输出到 JSON 文件里—(allModels.saveToFile(application.exeName+'.tmp.dmj')，一句代码的事)，再调一个你自己写的 EXE 程序读这个 JSON，剩下的事你自己看着办了。—

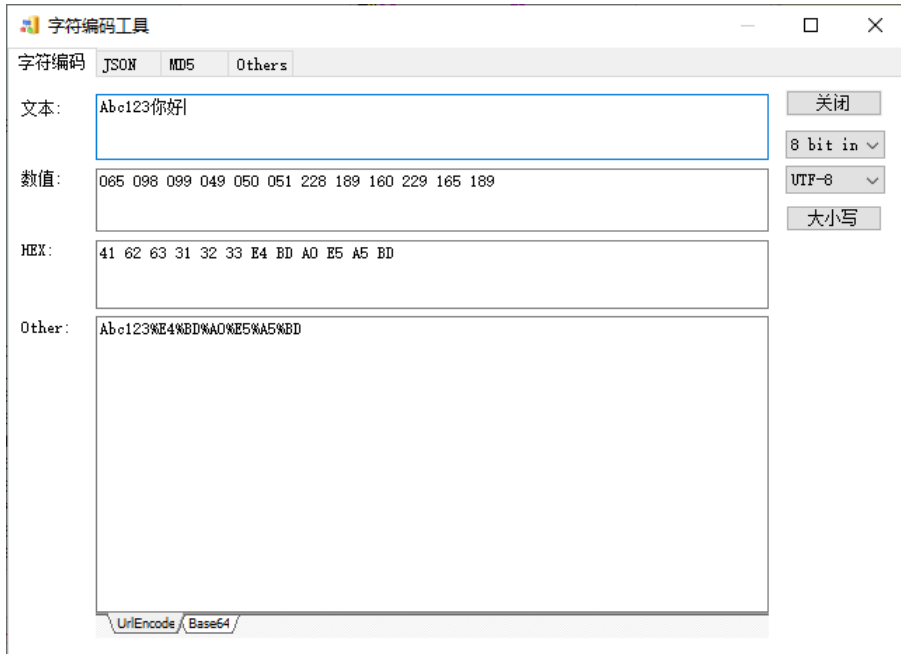
mac64/linux64/win64 下 EXCEL 也变成脚本生成 html 了，修改 export_xls_LANG.ps_脚本即可。

五十三、如何配置外部工具？

往那个自定义工具目录塞文件、塞程序、塞脚本就是了，参见《EZDML 配置文件》。

五十四、 字符编码工具是做啥用的？

是我自己常用的一些编码转换工具，包含几种编码的相互转换、JSON 的格式化、MD5 计算、随机数生成等功能。本来是独立的小程序，因为要用的时候不好找，而 EZDML 是经常用的，干脆就绑一块了。



五十五、 EZDML 能导入其它工具（如 PowerDesigner）的文件吗？

不能，目前唯一能做的是导入现有数据库表结构。当然，有脚本工具在，理论上你可以写脚本导入任何东西。就像某人曾经说的，给客户一个 PL/SQL dev 就行了，想干啥都可以自己到数据库里去弄，还要啥自行车呢。

EZDML 只是个小工具，完全无法与其它大型工具相提并论。如果你用其它工具用得很好，没必要使用本工具；如果你的项目已经用开了一种工具，奉劝你不要随便冒险作出改变（除非你们项目组所有成员本来早就已经看它不顺眼了）；如果你觉得其它工具不太好用，或者还没有固定使用哪种工具，或者有一些功能是你正需要的，可以试试 EZDML。

还是要补上那句话，使用 EZDML 出现任何后果本人均不负责。

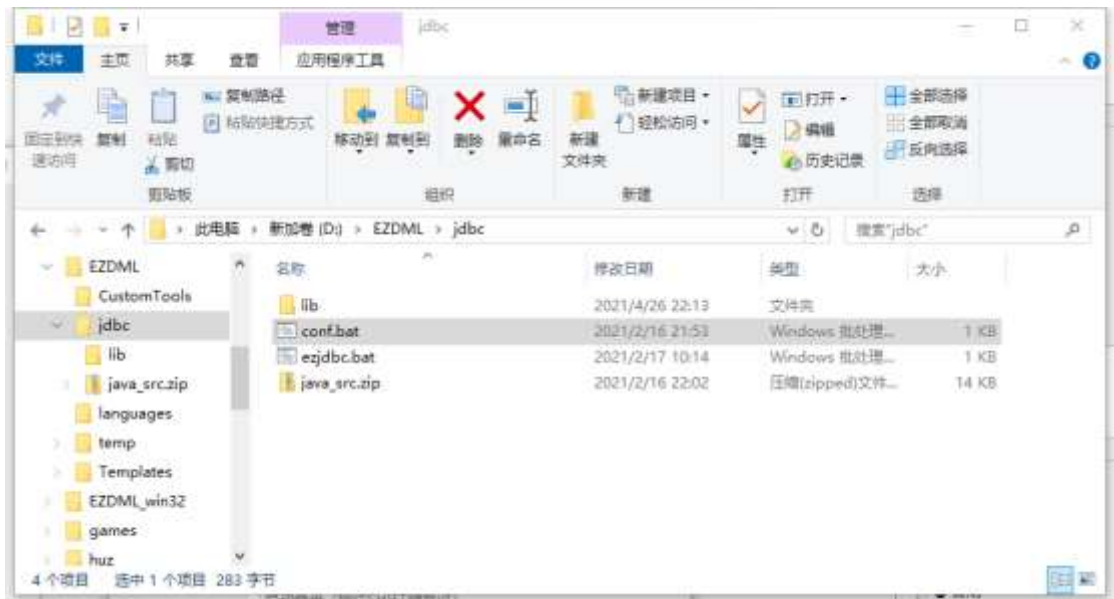
嗯，最新的版本可以导入 pdm (XML) 格式了。

五十六、 HTTP 连接和 JDBC 连接怎么用？

EZDML 原生只支持有限几种数据库（甚至这有限几种的驱动程序也不能保存在所有环境下连接成功），为了兼容支持其它的数据库，最新的版本增加了 HTTP 连接，就是只要有一个 HTTP 服务，实现有限的几个请求 URL 返回 JSON 结果，告诉 EZDML 有哪些表、哪些字段，以及哪些索引约束主外键，如何执行 SQL，就够了。

新版同时提供了一个 JDBC 的 HTTP 服务，它是 JAVA 代码写的（嗯，只是个 DEMO，你可以用自己熟悉的语言写），只要你的数据库能通过 JDBC 连上，基本上就可以了。使用步骤大概如下：

1. 打开 EZDMLjdbc 目录，编辑 conf.bat/sh 文件设置连接参数，如：



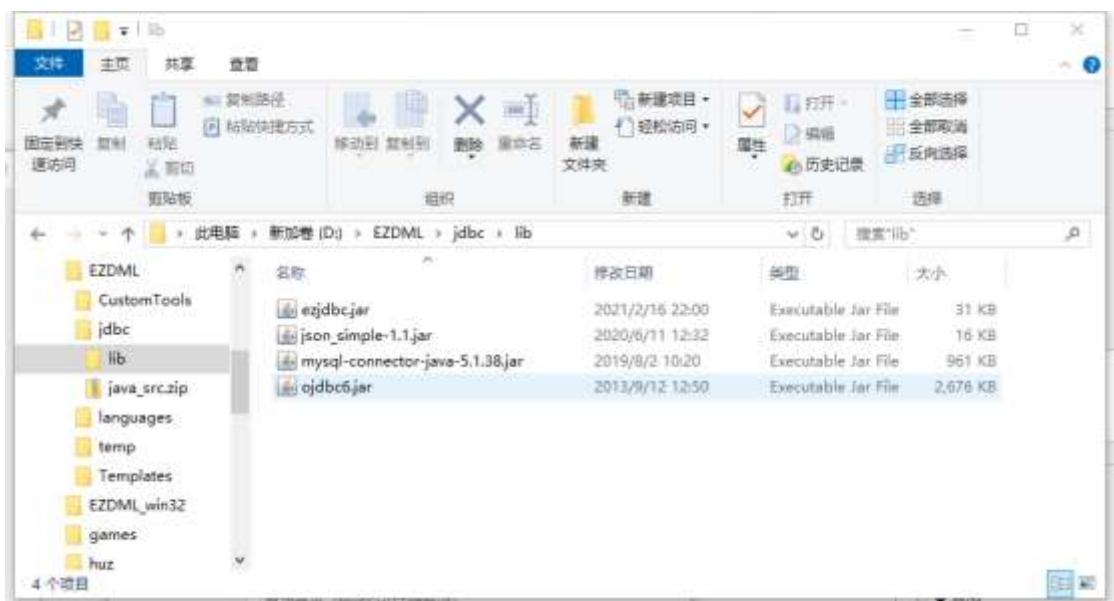
```

set java_bin=java.exe
set jdbc_driver=com.mysql.jdbc.Driver
set jdbc_url=jdbc:mysql://192.168.0.2:3306/test?useUnicode=true^^&characterEncoding=UTF-8
set jdbc_username=root
set jdbc_password=abcd
set jdbc_engineType=
set http_password=1234
set http_port=8083

```

如 java.exe 不在搜索 PATH 上，请正确设置 java.exe 的全路径。

2. 将连接 JDBC 数据库所需要的 JAR 包放入 lib 子目录（嗯，这里已经有 mysql5 和 oracle 的两个 jar 了，其它数据库需要自己添加 jar）：



3. 运行 ezjdbc.bat/sh 连接数据库并启动 HTTP 服务（当然，从 EZDML 工具菜单启动也是一样的效果）：

```
C:\Windows\system32\cmd.exe
EzdmJdbcHttpServer parameters:
jdbc.driver=com.mysql.jdbc.Driver
jdbc.url=jdbc:mysql://192.168.1.100:3306/?useUnicode=true&characterEncoding=UTF-8
jdbc.username=root
http.port=8083
CLASSPATH=.\lib\ezjdbc.jar;.\lib\json_simple-1.1.jar;.\lib\mysql-connector-java-5.1.38.jar;.\lib\ojdbc6.jar
edit conf.bat to change parameters
-----
starting java...
2021-05-16 17:45:39 EzdmJdbcHttpServer V20210216 by buzz@163.com, launching...
2021-05-16 17:45:39 Loading driver class... com.mysql.jdbc.Driver
2021-05-16 17:45:39 Connecting database... root/jdbc:mysql://192.168.1.100:3306/?useUnicode=true&characterEncoding=UTF-8
2021-05-16 17:45:40 Connect database success.
2021-05-16 17:45:40 EzdmJdbcHttpServer starting... binding port: 8083 engineType: MYSQL
2021-05-16 17:45:40 EzdmJdbcHttpServer started at http://localhost:8083/ezdm/ (PASSWORD AUTH ENABLED)
```

4. 打开 EZDML，通过 HTTP 连接 JDBC 服务（用户名无所谓，但密码要跟 conf 中指定的一样）：

连接数据库

连接类型: HTTP

数据源: http://localhost:8083/ezdm/

用户名: aaa

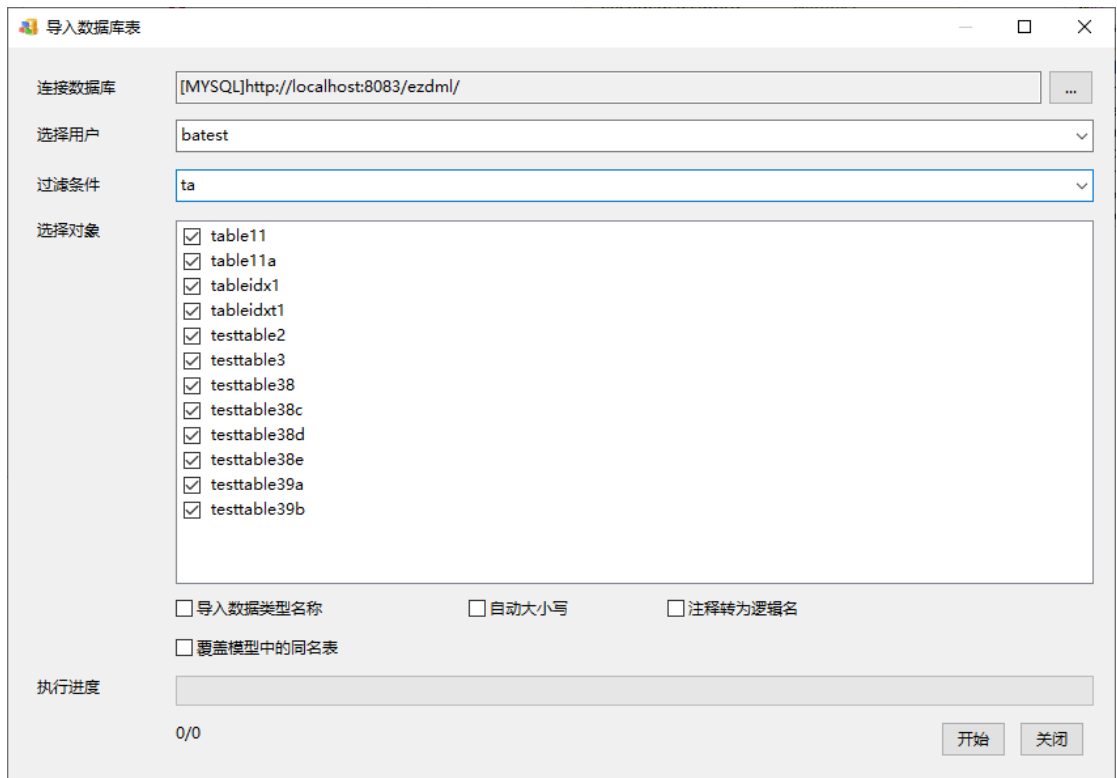
密码: ****

记住密码

自动登录 (按住SHIFT键跳过)

确定 取消

5. 连接成功即可导入生成数据库：



安装包中有附上 JDBC HTTP 服务的源码，从源码 EzHttpServer.java 可以看出 HTTP 服务大概需要实现以下几个接口：

- connect——连接，会传密码过来校验，成功时需要返回一个 token（后续所有请求都会附上这个 token），并告知数据库类型
- disconnect——断开连接
- ExecSql——执行 SQL
- OpenTable——通过 SQL 打开数据表，返回数据（注意格式转换，BLOB 字段需要用 HEX 编码）
- GetObjInfos——获取表的所有信息（表名可能带点号指定 OWNER，返回的 JSON 格式与 EZDML 的 dmj 文件格式是基本一样的）
- ObjectExists——判断指定名称的对象（表、索引、主外键约束等，名称可能带点号指定 OWNER）是否存在
- GenCustomSql——如果 connect 连接时返回了 NeedGenCustomSql，则 EZDML 在每次生成 SQL 时都会调这个方法

请求执行成功时，返回 JSON 结果中 resultCode 为 0；执行失败时 resultCode 为 -1，并在 errorMsg 中返回错误信息内容。

五十七、 工具菜单中的 HTTP 服务是作什么用的？跟 JDBC 服务有什么关系？

那个跟 JDBC 没有任何关系。它的作用嘛就像是连接代理，比如说你的电脑 A 没有装驱动连不上数据库，但旁边有一台电脑 B 能连上，那你可以旁边电脑 B 上安装 EZDML 连接数据库并运行 HTTP 服务作为代理，然后你的电脑 A 上就可以通过连接电脑 B 的 HTTP 服务来访问数据库了。

五十八、 EZDML 收费吗？需要捐助吗？

EZDML 是免费的。使用本软件导致任何后果作者均不负责。

暂时也不需要捐助。之前有网友非要表示捐助了一下，在此表示感谢！

五十九、 EZDML 是用什么开发的？

最开始是 Delphi，现在是 Lazarus，开发语言是 Pascal。

六十、 EZDML 会开源吗？有 MAC 版吗？

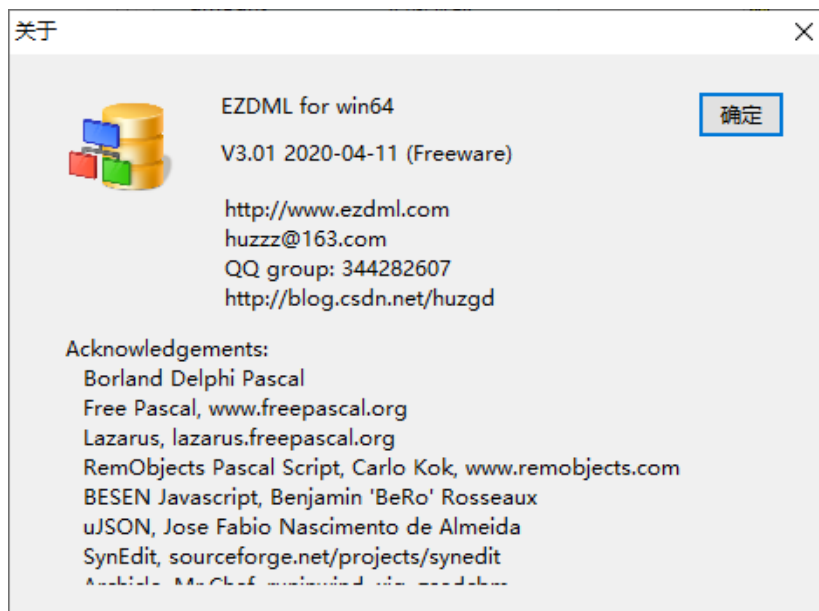
暂时不会开源。

EZDML 本来也只是自己用，有空就改改，没空就放放，并不追求什么效果。我是个自私的人，对自己没好处的事是不会干的。写这个工具更多是满足自己的一种兴趣和成就感，而不是为谁谁服务的说法，没有那么高尚。这么多年我也没写什么文档，这次写这么多算是恶补一把了。我自己也经常用开源的东西，我很理解大家有希望开源的想法，也不排除以后会有这可能，但目前只能说：抱歉！

新版本已经移植到 MAC 或 Linux，欢迎使用。

六十一、 还有其它的问题怎么办？

About 里有联系方式：



Q 群仅用于交流问题，平时没什么问题时是比较安静的，入群没有欢迎仪式，不需要打招呼和自我介绍，想出群也不用客气走就是了，发广告或不合适的主题的话很可能被禁言或踢出。

还有个 CSDN 博客（<http://blog.csdn.net/huzgd>），由于之前 CSDN 非得要关注微信公众号发消息才允许登录（认证了手机都登录不了），我懒得关注他们的微信，所以有段时间没

更新那个博客了。本来想抽点时间自己在网站上建个博客的，这两天又试了下，CSDN 又可以登录了，那还是用 CSDN 吧。

邮件不常看，未及时回复时请谅解。

作者只是一个普通俗人，有时忙有时心情不好，有时可能有些问题不喜欢或不愿意或其它原因解决不了的，也请见谅。

未完待续~后续有其他典型问题的话会陆续更新此文档。

此文档及相关资料的解释权归作者所有，转载请注明来源。