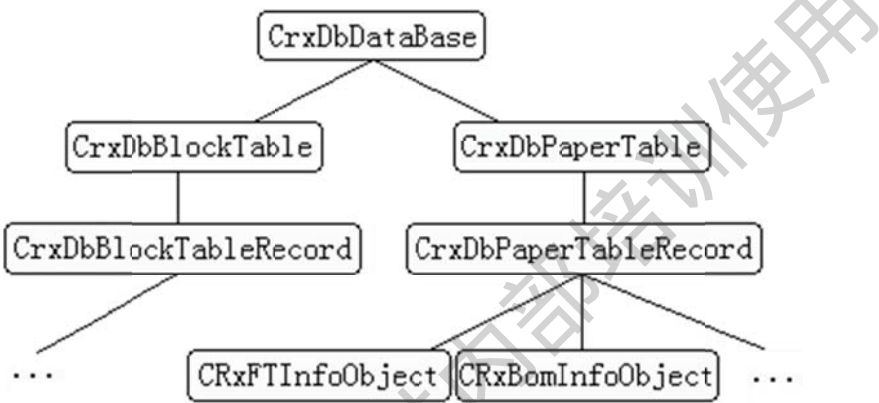


第 10 章 幅面信息管理

工程图纸中通常还包括零件序号、图框、标题栏、参数栏、明细表等内容，并且后续需要进行图纸打印以及产品信息输出。

电子图板可以快速设置图纸尺寸、调入图框、标题栏、参数栏、填写图纸属性信息。电子图板可以快速生成符合标准的各种样式的零件序号、明细表、并且零件序号与明细表可以保持相互关联，极大提高编辑修改的效率。

在 ObjectCRX 中提供的幅面管理的类的层次结构如下图 10.1:



10.1 幅面信息管理类的层次结构

在 2.1.2 章已经详细介绍了电子图板数据库基本结构。CRxDBBlockTable 类和 CRxDBPaperTable 类都是表示数据库中块表的类，其中 CRxDBPaperTable 是表示电子图板 exb 文件块表结构类。CRxDBPaperTableRecord 类表示电子图板 exb 文件块表记录的类。CRxDBPaperTableRecord 包含了对电子图板 exb 文件中特有的图幅信息进行管理的方法和属性。CRxFTInfoObject 是表示图幅中标题栏信息的类，CRxBomInfoObject 是表示图幅中明细表信息的类。

幅面信息中图幅、图框、标题栏、明细表等的信息都是通过 dbpaper.h 头文件中的 CRxDBPaperTableRecord 类及其派生类的对象来进行管理的。

10.1 图幅信息

10.1.1 说明

为一个图纸指定图纸尺寸、图纸比例、图纸方向等参数。

在进行图幅设置时，除了可以指定图纸尺寸、图纸比例、图纸方向外，还可以调入图框和标题栏并设置当前图纸内所绘装配图中的零件序号、明细表风格等。

国家标准规定了 5 种基本图幅，并分别用 A0、A1、A2、A3、A4 表示。电子图板除了设置了这 5 种基本图幅以及相应的图框、标题栏和明细栏外，还允许自定义图幅和图框。

10.1.2 思路

1. CRxDbPaperTableRecord 类

ObjectCRX 中 CRxDbPaperTableRecord 类提供了幅面信息的提取和设置方法。该类声明的成员具体如下：

```
//设置模板
void SetTemplateFile(const CxCHAR* pFilePath,CDraft::Template_For_PaperInfo TempleType);
//获取图纸长宽尺寸
void GetPaperSize(DOUBLE& dPaperWidth,DOUBLE& dPaperHeight);
//获取图纸比例
void GetPaperScale(DOUBLE& dPaperScale);
//设置图纸比例
void SetPaperScale(DOUBLE dPaperScale);
//获取图纸方向
void GetPaperDirection(CDraft::Paper_Direction& eDir);
//设置图纸方向
void SetPaperDirection(CDraft::Paper_Direction eDir);
//获取图框加长系数
void GetPaperLongerQuot(LONG& lLongQuot);
//设置图框加长系数
void SetPaperLongerQuot(LONG lLongQuot);
//获取图幅(A0、A1...)
void GetPaperSizeDescription(CDraft::PaperSizeDescription& eSizeDescription);
//设置图幅(A0、A1...)
void SetPaperSizeDescription(CDraft::PaperSizeDescription eSizeDescription);
//判断尺寸、文字是否随图纸比例变化
void GetIsUpdateDimAndTxtByScale(BOOL& bIsUpdateDT);
//设置尺寸、文字是否随图纸比例变化
void SetIsUpdateDimAndTxtByScale(BOOL bIsUpdateDT);
//获取图框信息
void GetFrameInfo(CRxFTInfoObject* pCrxFramInfoObj);
//设置图框信息
CDraft::ErrorStatus SetFrameInfo(CRxFTInfoObject& CrxFramInfoObj);
//获取标题栏信息
void GetTitleInfo(CRxFTInfoObject* pCrxTitleInfoObj);
//设置标题栏信息
CDraft::ErrorStatus SetTitleInfo(CRxFTInfoObject& CrxTitleInfoObj);
```

```

//获取明细表信息
void GetBomInfo(CRxBomInfoObject* pBomInfoObj);
//设置明细表信息
CDraft::ErrorStatus SetBomInfo(CRxBomInfoObject& CrxBomInfoObj);
//根据文件路径设置图框
CDraft::ErrorStatus SetFrameByFile(const CxCHAR* cfmFilePath);
//根据文件路径设置标题栏
CDraft::ErrorStatus SetTitleByFile(const CxCHAR* chdFilePath);
//生成序号
CDraft::ErrorStatus GenPartNoByCurrentStyle(const CxCHAR* cPartnoTxt, CRxGePoint3d&
ptStart, CRxGePoint3d& ptTurning);

```

2. 图幅信息管理

电子图板使用 CRxDBPaperTableRecord 类的对象对图幅信息的管理从功能上可分为两类：获取图幅信息；设置图幅信息。具体使用的方法如下：

提取方法：GetPaperScale、GetPaperDirection、GetPaperLongerQuot、GetPaperSizeDescription；GetIsUpdateDimAndTxtByScale。

设置方法：SetPaperScale、SetPaperDirection、SetPaperLongerQuot、SetPaperSizeDescription；SetIsUpdateDimAndTxtByScale。

注意：由于设置图纸方向、图纸加长系数及图幅规格均可导致图框的变化，调用这三个函数将导致当前图框丢失。因此这三个函数调用后需再次设置图框。

10.1.3 步骤

1. ObjectCRX 对图幅信息进行管理时候，要首先获得电子图板当前的图形数据库，方法如下：

```
CRxDBDatabase *pDb = crxdbHostApplicationServices()->workingDatabase();
```

2. 然后获得指向图纸块表的指针，代码如下：

```
CRxDBPaperTable* pTable;
es = pDb->GetPaperTable(pTable, CRxDB::kForRead);
```

3. 然后获得指向图纸块表记录的指针，代码如下：

```
CRxDBPaperTableRecord* pRecord;
pTable->getAt(_T("Model"), pRecord, CRxDB::kForRead);
```

4. 获得块表记录后，就可以进行幅面信息的管理。

注册一个GetBom的命令，获得明细表的内容后，然后在命令栏里显示出来。详细的代码如下：

```

void GetBomInfo()
{
    // 获得当前图形数据库

```

```
CRxDBDatabase *pDb = crxdbHostApplicationServices()->workingDatabase();

// 打开当前图形模型空间块表记录
CDraft::ErrorStatus es = CDraft::eOk;
//
CRxDBPaperTable* pTable;
es = pDb->GetPaperTable(pTable, CRxDB::kForRead);
if (es != CDraft::eOk)
    return ;
CRxDBPaperTableRecord* pRecord;

pTable->getAt(_T("Model"), pRecord, CRxDB::kForRead);
//如果记录不为空，则读取明细表内容
if (pRecord != NULL)
{
    //pRecord->SetTemplateFile()
    CString strHeader, strItem, strContent, strBom;
    strHeader = _T("\n");
    CRxBomInfoObject* pBomInfo = new CRxBomInfoObject;
    pBomInfo->Init();
    pRecord->GetBomInfo(pBomInfo);
    int iCol = pBomInfo->GetColCount();
    int iRow = pBomInfo->GetRowCount();
    for (int j=0; j<iRow; j++)
    {
        for (int i=0; i<iCol; i++)
        {
            if (j == 0)
            {
                CxCHAR* cHeader;
                pBomInfo->GetBomHeaderAt(i, cHeader);
                strItem = cHeader;
                strHeader = strHeader + strItem + _T("; ");
            }
            CxCHAR* cItem;
            pBomInfo->GetBomItemAt(j, i, cItem);
            strItem = cItem;
            strContent = strContent + strItem + _T("; ");
        }
        strContent += _T("\n");
    }
}
```

```
    }  
    strBom = strHeader + _T("\n") + strContent;  
    crxutPrintf(strBom);  
    delete pBomInfo;  
    pBomInfo = NULL;  
}  
else  
{  
    acutPrintf(_T("error"));  
}  
}
```

10.1.4 效果

1. 首先在电子图板中新建一个文件，然后调入图框 A4E-F-Divided(CHS)，然后调入标题栏 GB-A(CHS)，最后填写明细表如下图：

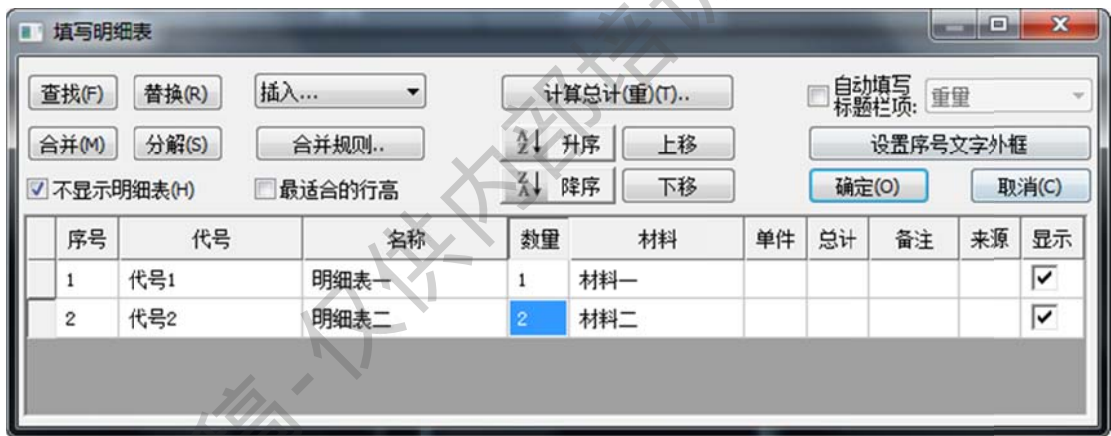


图 10.2 明细表内容

2. 编译链接程序，启动电子图板 2011，加载生成的 CRX 文件。然后执行 GetBom 命令，效果如下图：

```
命令: getbom  
启动执行命令: "GetBom"  
序号; 代号; 名称; 数量; 材料; 单件; 总计; 备注; 来源;  
1; 代号1; 明细表一; ; 材料一; ; ; ;  
2; 带好; 明细表二; ; 材料二; ; ; ;
```

图 10.3 GetBom 命令执行结果

10.1.5 小结

学习本节的内容之后，读者需要掌握下面的知识点：

- 理解幅面概念。
- 理解电子图板幅面管理中类的层次结构。
- 掌握类 CRxDbPaperTableRecord 的对象的运用。
- 掌握图幅信息获取和设置的编程方法。

10.2 图框、标题栏、明细表

10.2.1 说明

电子图板可以快速设置调入图框、标题栏等；电子图板可以快速生成符合标准的各种样式的零件序号、明细表、并且零件序号与明细表可以保持相互关联；电子图板的上述功能极大提高编辑修改的效率。ObjectCRX 中也提供了相应的方法通过编程来实现上述功能。

10.2.2 思路

1. 图框、标题栏、明细表

通过 10.1.2 节的内容我们可以看到 ObjectCRX 中 CRxDbPaperTableRecord 类也提供了对图框、标题栏、明细表信息进行管理的方法。从功能上可以分为以下三类：提取属性信息、反写属性信息、调入文件(限图框、标题栏、参数栏)。具体的方法分类明细如下：

提取方法：GetFrameInfo、GetTitleInfo、GetBomInfo

反写方法：SetFrameInfo、SetTitleInfo、SetBomInfo

调入方法：SetFrameByFile、SetTitleByFile

上述方法的调用方式如下：

- (1) 提取方法中，需要传入类对象 CrxFrameInfoObj，比如：
GetFrameInfo(CrxFrameInfoObj& pFrameInfoObj)、GetBomInfo(CrxBomInfoObj& pBomInfoObj)。所传入类对象包含所要提取的信息
- (2) 反写方法中，需要指定包含属性信息的类对象，比如：
SetFrameInfo(CrxFrameInfoObj pFrameInfoObj)、SetBomInfo(CrxBomInfoObj pBomInfoObj)。
- (3) 调入方法中，前两者需要指定调入文件位置，如果为空，则表示清除现有图框、标题栏；后者除指定文件位置外，还需指定参数栏在图纸中的定位点；由于调入图框要受到图纸幅面的限制，并非任意的图框文件均能调入到当前 Paper 中，如果不匹配，会弹出提示框。
- (4) 传入/传出类对象简述
图框、标题栏对应的类对象主要包含了一行数据，方法就是这一行数据的读写；
明细表对应的类对象主要包含了一张数据表，方法就是对这张数据表的读写

注意：由于 DWG 文件中没有图框、标题栏、明细表的概念，提取这些信息时需要借助模板。模

板有两种，位置模板和块模板，具体格式参考 CaxaInfo 组件的规定。按模板提取信息时，在调用方法 GetFrameInfo、GetTitleInfo、GetBomInfo 之前，需调用方法 SetTemplateFile 指定模板文件及其类型。反写信息时，首先判断是否已指定模板文件。如果已指定块模板，则反写图框标题栏时反写对应属性的属性值，反写明细表信息时，首先找到指定的行，再根据列的位置找到对应块的属性，反写其属性值；如果已指定位置模板，则按位置推断出指定位置的文本，如果该位置存在文本，则先将其清除，再写入新值。需要指出的是，由于位置模板中没有指定图框位置信息，目前不支持按位置模板提取、反写图框信息。

2. 序号

方法包括，在指定位置生成数字为 n 的序号；生成序号为 n 的明细表行

生成方法：GenPartNoByCurrentStyle

注意：当调用 GenPartNoByCurrentStyle 生成序号后，明细表实际上已经生成，只是将明细表项中的“是否显示”项置为“否”；当调用 GenBomItem 生成明细表项时，如果已存在该项，且为“不显示”，则将其置为“显示”，其他情况则不做任何处理。

10.2.3 步骤

编写 SetTitleValue()函数，设置标题栏上的图纸编号，具体过程如下：

1. 首先获得图纸块表记录，详细过程参见 10.1.3 节内容。
2. 获得标题栏信息，代码如下：

```
CRxFTInfoObject* pTitleInfo = new CRxFTInfoObject;
pTitleInfo->Init();
pRecord->GetTitleInfo(pTitleInfo);
```

需要声明一个标题栏信息类对象指针，然后通过 CRxDbPaperTableRecord 类对象的 GetTitleInfo 方法，使刚声明标题栏信息类对象指针指向当前图纸块表记录的标题栏信息。

3. 设置标题栏上的图纸编号，代码如下：

```
pTitleInfo->SetAttributeValue(_T("测试XX"), i);
```

通过标题栏信息类 CRxFTInfoObject 的 CRxFTInfoObject 方法，可以设置当前标题栏的属性值。

4. 最后注册一个 SetTitle 命令，与函数关联，函数详细代码如下：

```
void SetTitleValue()
{
    CRxDbDatabase *pDb = crxdbHostApplicationServices()->workingDatabase();
    CDraft::ErrorStatus es = CDraft::eOk;
    CRxDbPaperTable* pTable;
```

```
es = pDb->GetPaperTable(pTable, CRxDB::kForRead);
if (es != CDraft::eOk)
    return ;
CRxDBPaperTableRecord* pRecord;
pTable->getAt(_T("Model"), pRecord, CRxDB::kForRead);
if (pRecord != NULL)
{
    CString strAttName;
    CRxFTInfoObject* pTitleInfo = new CRxFTInfoObject;
    pTitleInfo->Init();
    pRecord->GetTitleInfo(pTitleInfo);
    int iCount = pTitleInfo->GetAttributesCount();
    for (int i=0; i<iCount; i++)
    {
        CxCHAR* cAttName;
        pTitleInfo->GetAttributeName(cAttName, i);
        strAttName = cAttName;
        if (strAttName.Compare(_T("图纸编号")) == 0)
        {
            //设置图纸编号
            pTitleInfo->SetAttributeValue(_T("测试XX"), i);
            break;
        }
    }
    for (int i=0; i<iCount; i++)
    {
        CxCHAR* cAttName;
        CxCHAR* cAttValue;
        pTitleInfo->GetAttributeName(cAttName, i);
        pTitleInfo->GetAttributeValue(cAttValue, cAttName);
    }
    es = pRecord->SetTitleInfo(*pTitleInfo);
    if (es == CDraft::eOk)
    {
        //设置成功
        crxutPrintf(_T("Has Set Title"));
    }
    if (pTitleInfo != NULL)
    {
        delete pTitleInfo;
    }
}
```



```

        pTitleInfo = NULL;
    }
}
}

```

10.2.4 效果

(1) 编译链接程序, 启动电子图板 2011, 加载生成的 CRX 文件。打开 10.1.4 节新建文件, 执行 SetTitle 命令, 效果如下:



图 10.4 SetTitle 命令执行效果

10.2.5 小结

学习本节的内容之后，读者需要掌握下面的知识点：

- 掌握图框、标题栏、明细表的提取、反写、调入方法。
- 掌握序号的生成和带序号明细表行生成的方法。