

CS 框架 V2.2-3.0

业务单据窗体(主/从表)快速开发指南

By Jonny Sun at 2012-02-18

(doc v1.0)

《从销售订单学起》	4
1. 销售订单的数据结构	4
2. 销售订单主表(tb_SO)	4
3. 销售订单明细(tb_SOs)	5
一. 业务单据窗体基类介绍	6
1.1 关于业务单据窗体.....	6
1.2 业务窗体的成员（主要字段和方法）	6
二. 业务单据窗体的继承关系.....	8
三. 业务单据窗体逻辑关系图	10
3.1 开发框架 V2.2 逻辑关系图	10
3.2 开发框架 V3.0 逻辑关系图	11
四. 业务单据窗体开发步骤	12
4.1 新建业务窗体	12
4.1.1 在销售模块新建[frmSO]窗体	12
4.1.2 [新建项]对话框.....	13
4.1.3 [Inheritance Picker]对话框	13
4.2 窗体布局设计	14
4.2.1 用户界面介绍.....	14
4.2.2 数据查询页及主表格(GridControl)控件	16
4.2.3 数据编辑页面设计[主/从表输入区域]	19
4.2.4 主表输入组件设计.....	21
4.3 使用代码生成器工具生成 ORM、BLL、DAL 代码	22
4.3.1 参考视频:	22

4.4 BLL 层代码.....	23
4.4.1 BLL 继承关系图.....	23
4.4.2 blIBaseBusiness 类.....	24
4.4.3 blISO 类.....	25
4.5 DAL 层代码.....	26
4.5.1 DAL 继承关系图.....	26
4.5.2 dalBaseBusiness 类.....	27
4.5.3 dalSO 类.....	28
4.6 编写窗体代码.....	28
4.6.1 初始化业务窗体(重写 InitializeForm 方法).....	29
4.6.2 Form.Load 事件调用 InitializeForm 方法.....	29
4.6.3 实现查询功能(DoSearchSummary 方法).....	29
4.6.4 绑定主表输入组件数据源(DoBindingSummaryEditor 方法).....	30
4.6.5 绑定明细表格的数据源(DoBindingDetailGrid 方法).....	30
4.6.6 增加(DoAdd 方法).....	30
4.6.7 删除(DoDelete 方法).....	31
4.6.8 修改(DoEdit 方法).....	31
4.6.9 查看(DoViewContent 方法).....	31
4.6.10 取消(DoCancel 方法).....	32
4.6.11 打印(DoPrint 方法).....	32
4.6.12 审核(DoApproval 方法).....	32
4.6.13 保存(DoSave 方法).....	32
4.6.14 主表数据合法性检查(ValidatingSummaryData 方法).....	33
4.6.15 明细表数据合法性检查(ValidatingDetailData 方法).....	33
4.6.16 显示修改历史记录(DoShowModifyHistory 方法).....	33
4.6.17 其它细节处理.....	33
五. 相关文章及视频参考.....	34
5.1 C/S 框架网视频下载.....	34
5.2 C/S 快速开发框架 - 代码自动生成视频下载.....	34
5.3 C/S 系统快速开发框架 - 生成 BLL 和 DAL.....	34
5.4 C/S 框架业务窗体设计-实现查询功能.....	34

5.5 C/S 系统快速开发框架 - 生成主从表的 ORM.....	35
5.6 C/S 系统快速开发框架 - 新增业务窗体(视频)	35
5.7 C/S 快速开发框架- 新增业务模块(视频下载)	35
六. 意见或建议	36


《从销售订单学起》

本教程以销售订单为例指导用户开发业务窗体，请参考开发框架销售模块的 frmSO 窗体。


1. 销售订单的数据结构

销售订单是标准的主从表结构，主表含有订单号，订单日期，客户资料，货币类型，金额等信息。订单的明细（明细表）含有货品编号，单位，单价，数量，金额等信息。

2. 销售订单主表(tb_SO)

字段名称	字段类型	说明
ISID	Identity	自增字段
SONO 	Varchar(20)	订单编号,流水号
VerNo	Varchar(2)	版本 01 02 顺序往下增加
CustomerCode	Varchar(20)	客户编号
ReceiveDay	Datetime	接单日期
PayType	Varchar(10)	付款方式来源 tb_paytype
CustomerOrderNo	Varchar(20)	客户订单编号
Salesman	Varchar(20)	营销员 tb_Person
Currency	Varchar(4)	币种,默认为公司本位币来源: tb_Currency
Amount	Numeric(18,2)	合计=sum(qty*price)
FinishingStatus	Varchar(10)	完成状态(新增、生产中、成品入仓, 已送货, 已结束)
OrderFinishDay	Datetime	订单实际完成日期
Remark	NVarchar(200)	备注
CreationDate	Datetime	创建日期
CreatedBy	Varchar(20)	创建者
LastUpdateDate	Datetime	最后修改时间
LastUpdatedBy	Varchar(20)	最后修改者
FlagApp	Char(1)	审核标志
AppUser	Varchar(20)	审批人
AppDate	Datetime	审批日期

3. 销售订单明细(tb_SOs)

字段名称	字段类型	说明
ISID 	Identity	自增字段
SONO	Varchar(20)	订单编号
Queue	Decimal(9,2)	订单次序
StockCode	Varchar(20)	货品编码
CustomerOrderNo	Varchar(50)	客户订单号码
ShipDay	Datetime	要求货期
Unit	Varchar(10)	单位
Qty	Int	数量
Price	Numeric(18,2)	订单价格
Amount	Numeric(18,2)	合计金额
Remark	NVarchar(50)	备注
CreationDate	Datetime	创建日期
CreatedBy	Varchar(50)	创建者
LastUpdateDate	Datetime	最后修改时间
LastUpdatedBy	Varchar(50)	最后修改者

SONO 是单据号码, 作为主从表的主键和外键, SONO 由系统按规则自动生成, 格式为: SO+YYYYMM+序列号(四位), 如: SO2012020001, SO 是单号的字头, 序列号四位, 最大数字为 9999, 就是指本月最多能生成 9999 张销售订单。

一. 业务单据窗体基类介绍

1.1 关于业务单据窗体

业务单据英文是 Business Document, 开发框架将业务单据定义为处理业务功能的主从表结构窗体, 如应收/应付帐款(AR,AP), 采购单(PO), 销售单(SO), 出入仓(IN/OUT)等类型的窗体。业务单据通常是主从表结构, 一张主表对应多个明细表, 也有一些业务单据只有主表, 没有严格意义上的划分。

开发框架将业务数据分为数据字典和业务单据, 通常数据字典是指单表数据或基础数据, 如货币资料, 客户, 供应商等。业务数据指应收、应付帐款、采购单、销售单、出入仓等处理的数据, 因此对应数据字典窗体和业务单据窗体。

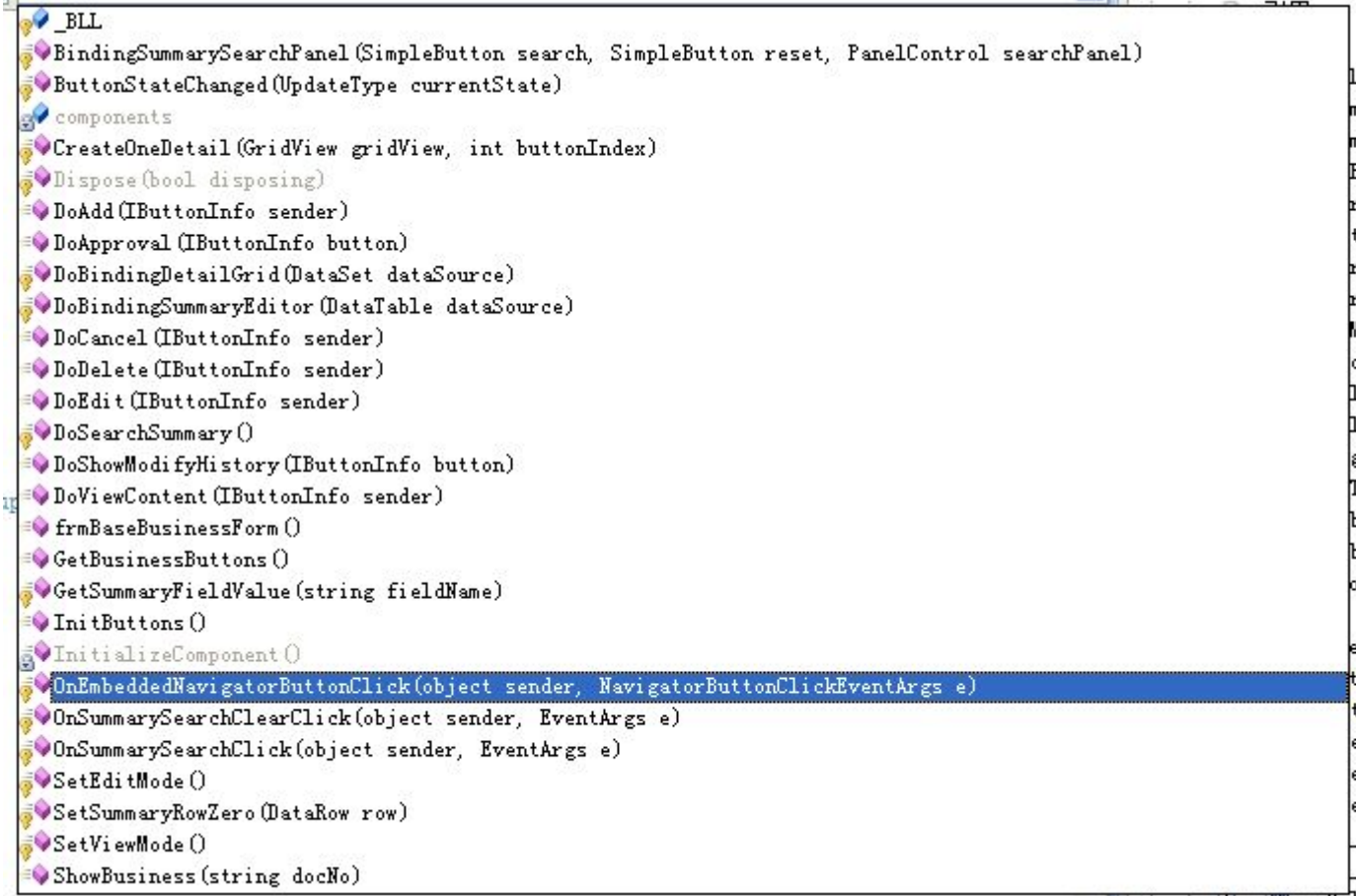
业务窗体继承 CSFramework.Library. frmBaseBusinessForm 类, 下表列出该类主要字段和方法:

1.2 业务窗体的成员 (主要字段和方法)

ID	名称	类型	说明
1	_BLL		业务逻辑层的实例, 由派生的窗体实例化对象。
2	BindingSummarySearchPanel		绑定业务窗体搜索页面上的两组按钮(查询和清空按钮)。
3	ButtonStateChanged		新增或修改时调用方法(触发的事件), 重写此方法可以对窗体的组件进行操作, 通常用于禁用或启用输入框或按钮的状态(Disable/Enable)。
4	CreateOneDetail		创建一条明细记录, 适用于有明细表的业务窗体。此方法由 OnEmbeddedNavigatorButtonClick 事件调用。
5	DoAdd		新增记录, 点【新增】按钮调用的方法, 同时将操作状态设置为 UpdateType. Add 状态。
6	DoApproval		审核单据。开发框架内审核功能有三个字段: FlagApp: CHAR(1), Y/N: Y 表示审核, N 表示未审核。 AppUser: 审核人。 AppDate: 审核日期。
7	DoBindingDetailGrid		绑定明细表格的数据源, 适用于有明细表的业务单据。
8	DoBindingSummaryEditor		绑定业务单据主表对应输入组件的数据源。
9	DoCancel		取消操作, 点【取消】按钮调用的方法, 取消后将操作状态置为 UpdateType.None。
10	DoDelete		删除记录, 点【删除】按钮调用的方法。 开发框架中业务窗体仅删除明细表, 将主表的数字类型的字段清零, 如金额, 数量等关键字段, 保留主表的记录避

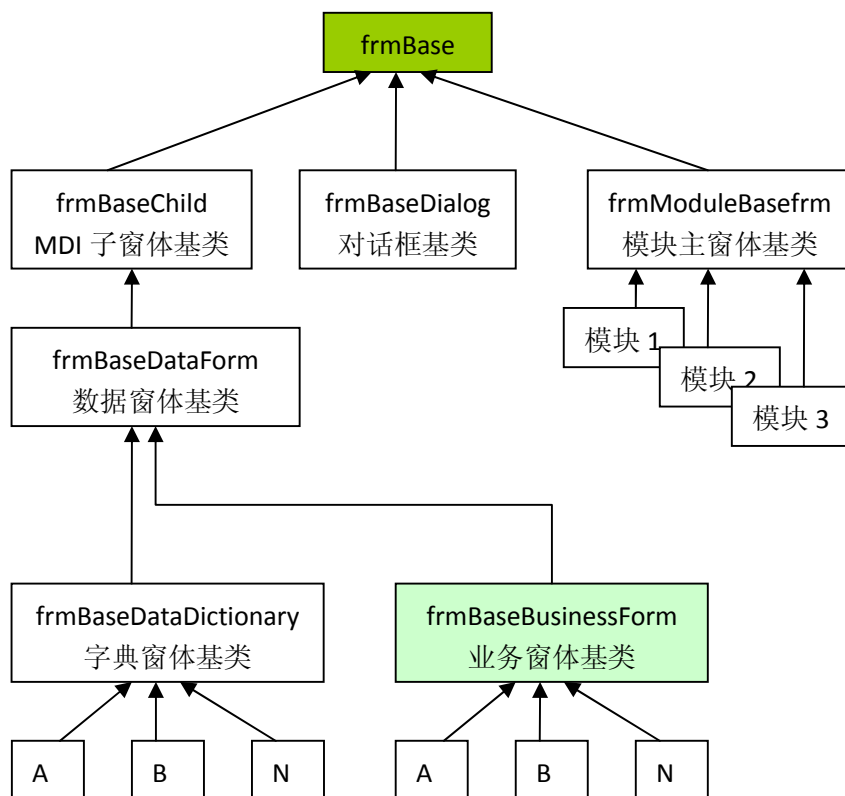
			免断号。
11	DoEdit		修改记录，点【修改】按钮调用的方法。 将操作状态置为 <code>UpdateType.Modify</code> 。
12	DoPrint		打印报表。
13	DoSave		保存数据，点【保存】按钮调用的方法。 保存成功将操作状态置为 <code>UpdateType.None</code> 。
14	DoSearchSummary		模板方法，查找主表的数据。此方法由【查询】按钮的 <code>Click</code> 事件调用。通常业务单据的交易数据很多，当打开窗体时有以下两种方式展示数据： 1.取最近 <code>n</code> 天的数据展示。 2.由用户输入查询条件查询数据。
15	DoShowModifyHistory		显示修改历史记录。
16	GetBusinessButtons		创建业务单据窗体的按钮数组，用于生成主窗体工具条的按钮。
17	GetSummaryFieldValue		获取当前记录指定字段的值。
18	InitButtons		重写的方法，初始化业务窗体的按钮。
19	OnEmbeddedNavigatorButtonClick		明细表格按钮的 <code>Click</code> 事件。由派生窗体的 <code>InitializeForm</code> 方法调用，绑定明细表格按钮的 <code>Click</code> 事件。
20	OnSummarySearchClearClick		【清空】按钮的 <code>Click</code> 事件
21	OnSummarySearchClick		【查询】按钮的 <code>Click</code> 事件
22	SetEditMode		进入修改模式，设置业务窗体相关按钮的状态。
23	SetViewMode		进入查看模式，设置业务窗体相关按钮的状态。
24	ShowBusiness		打开指定单号的业务单据。

类成员截图：



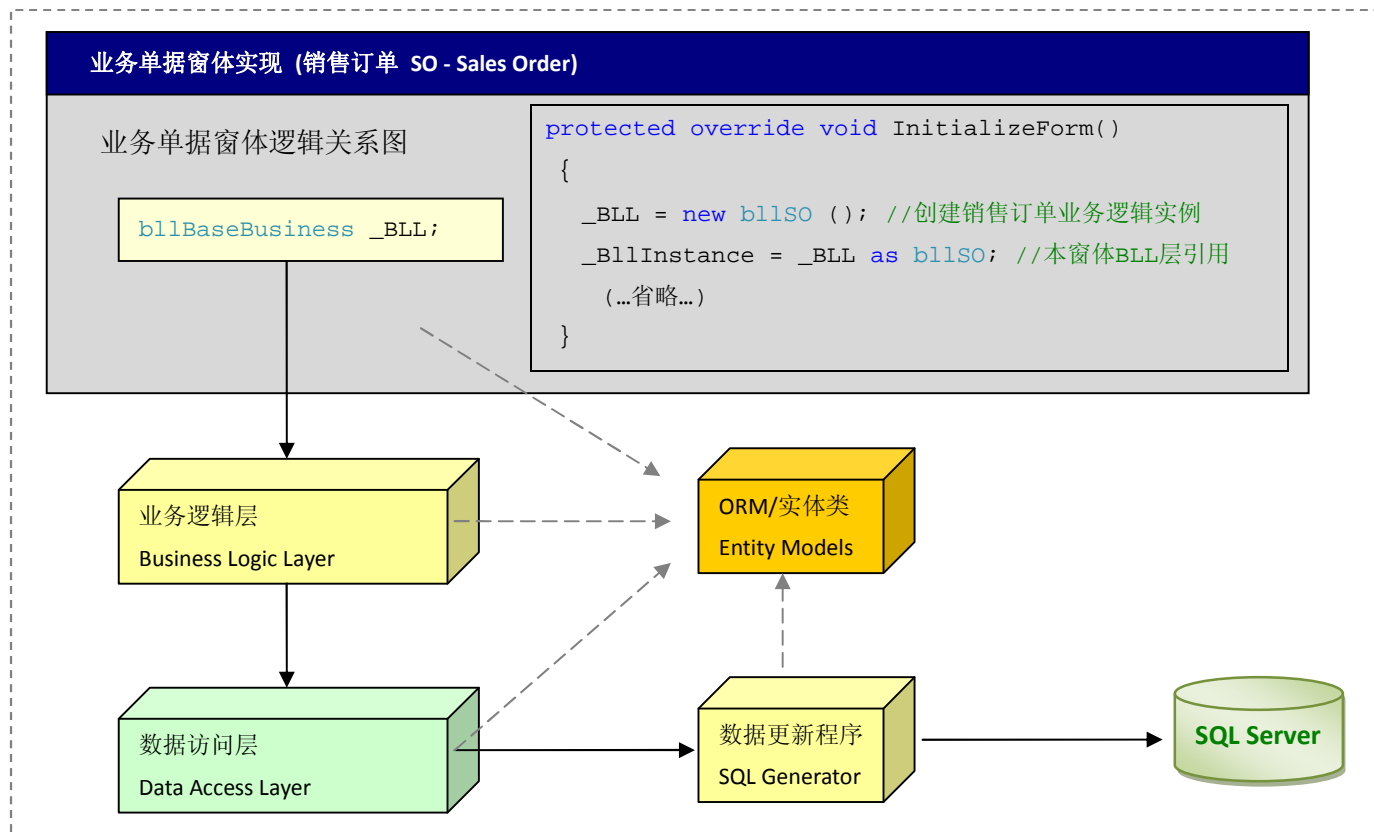
二. 业务单据窗体的继承关系

窗体继承关系图



三. 业务单据窗体逻辑关系图

3.1 开发框架 V2.2 逻辑关系图

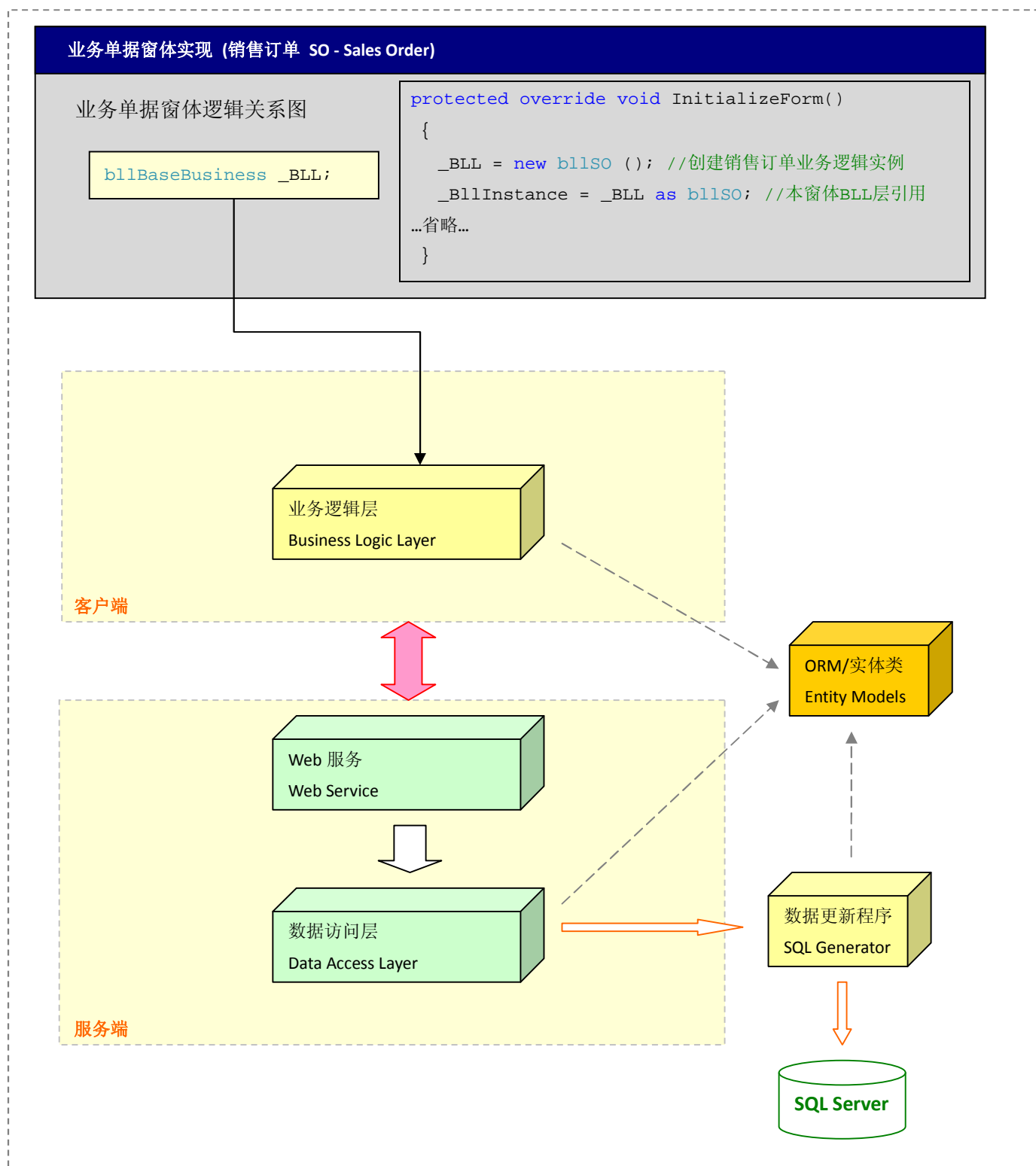


`frmBaseBusinessForm` 类定义了业务逻辑层成员变量 `_BLL`，在销售订单窗体的 `InitializeForm` 方法内实例化 `_BLL` 变量。

```
_BLL = new bllSO (); //销售订单业务逻辑实例
```

业务逻辑层负责处理窗体的主要事务，如增、删、改、查等，由业务逻辑类调用数据层进行数据查询或更新。

3.2 开发框架 V3.0 逻辑关系图



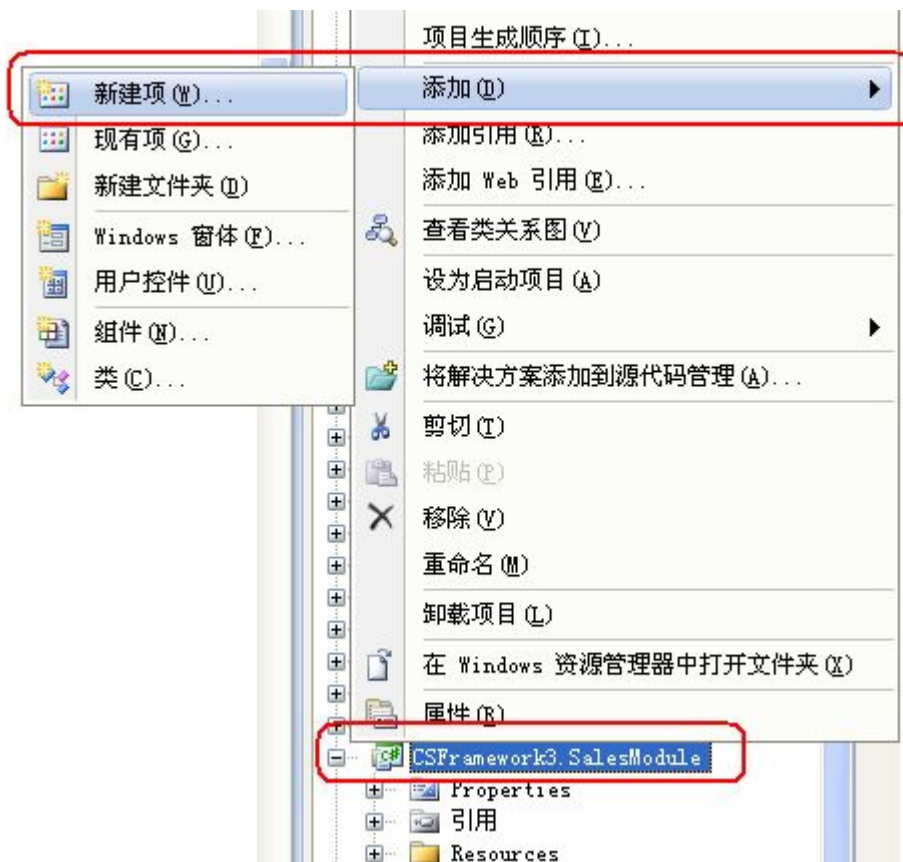
四. 业务单据窗体开发步骤

以销售订单窗体举例说明，窗体名称：frmSO。以下是业务单据窗体开发步骤详解。

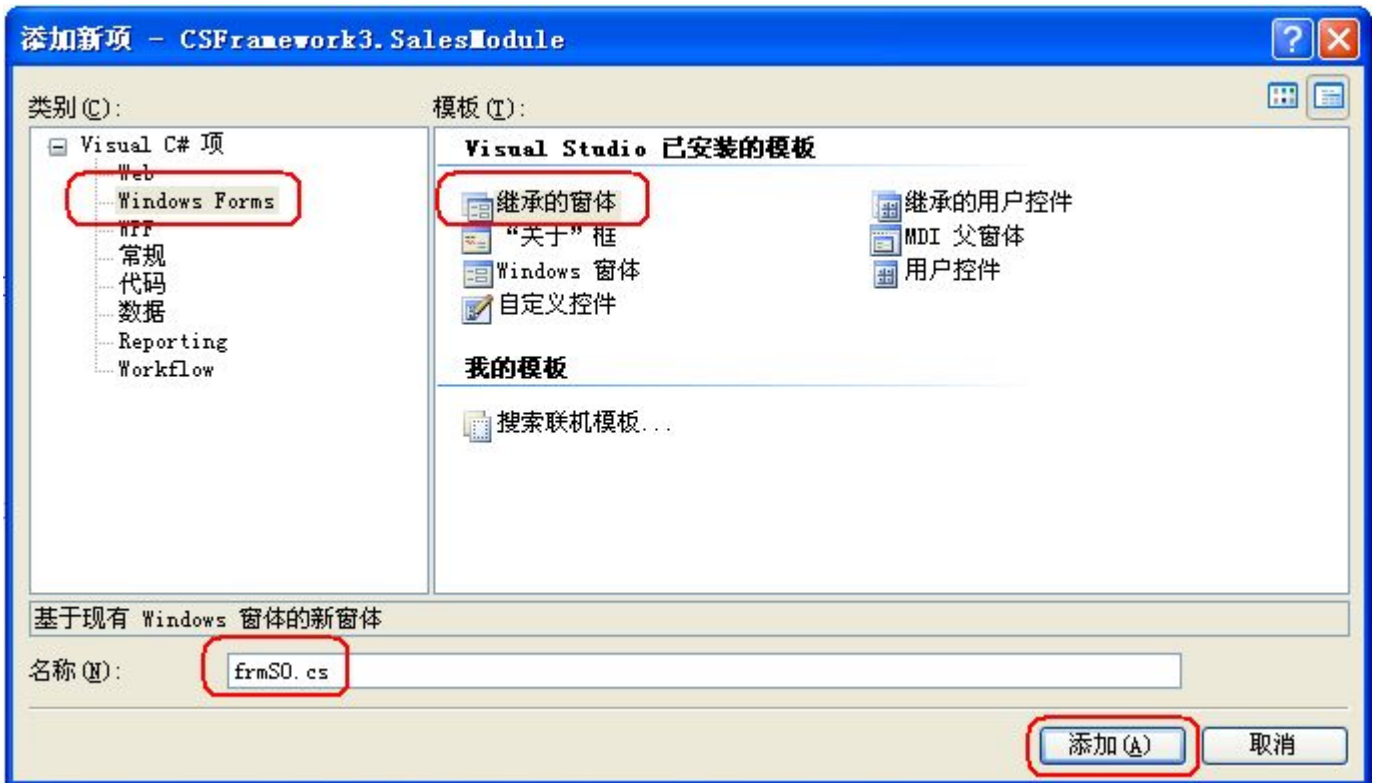
4.1 新建业务窗体

4.1.1 在销售模块新建[frmSO]窗体

在【销售模块】点右键，【Add】→【New Item】→打开【新建项对话框】。

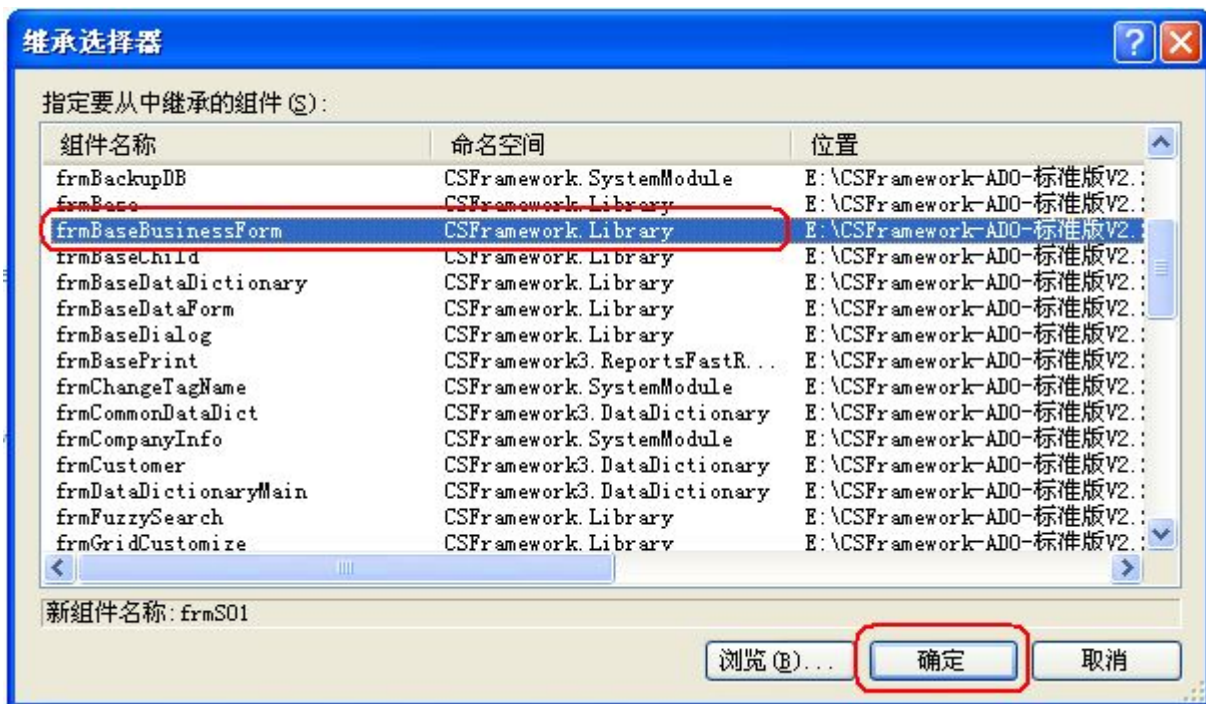


4.1.2 [新建项]对话框



在【Name】输入框输入【frmSO.cs】，最后点【Add】按钮弹出【Inheritance Picker 窗体】。

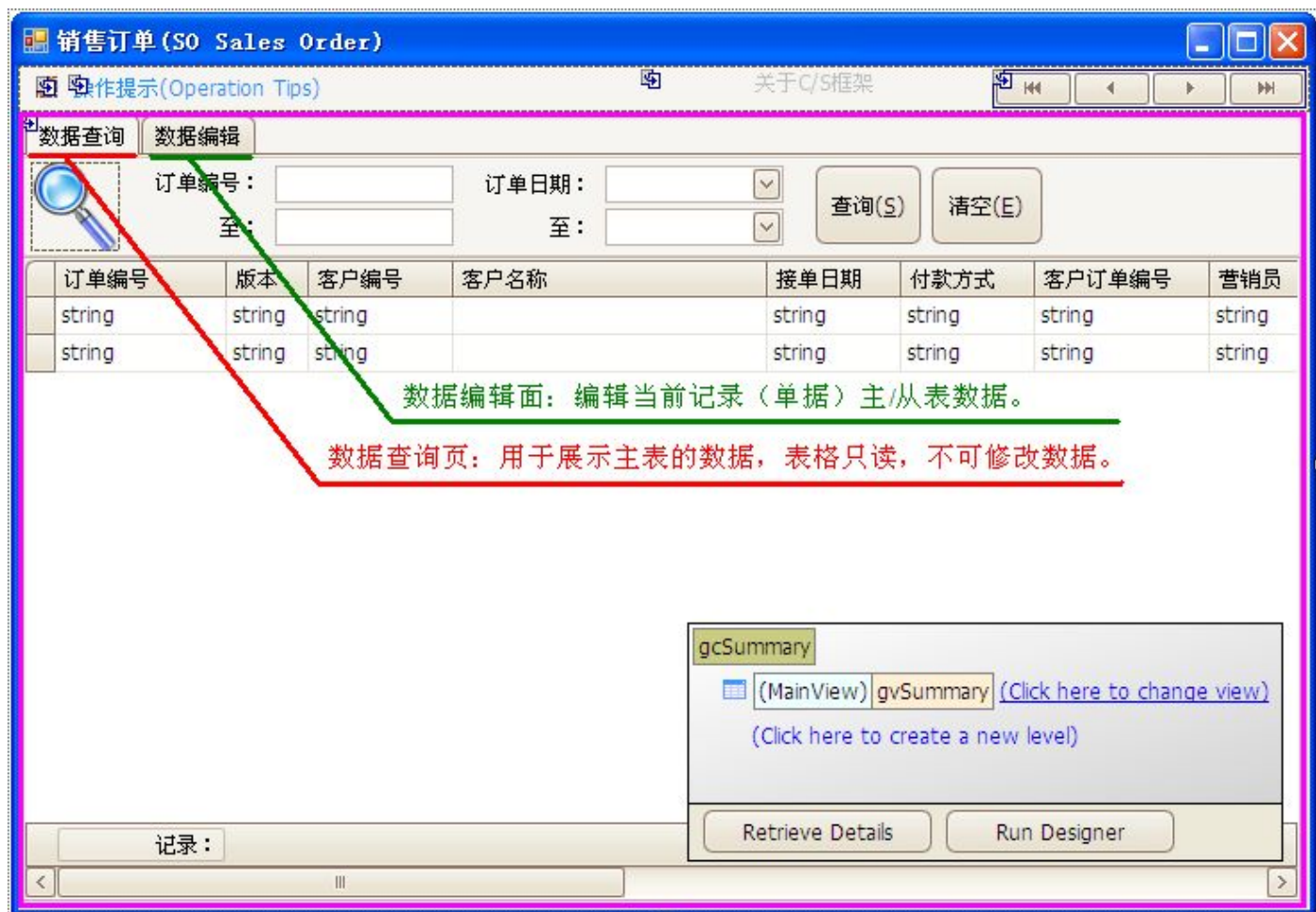
4.1.3 [Inheritance Picker]对话框



选择继承自【frmBaseBusinessForm】，点【OK】按钮，新建窗体的单元文件。

4.2 窗体布局设计

4.2.1 用户界面介绍



销售订单 (SO Sales Order)

操作提示 (Operation Tips) 关于 C/S 框架

数据查询 数据编辑

订单编号: 订单日期: 查询(S) 清空(E) 查询条件

至: 至:

订单编号	版本	客户编号	客户名称	接单日期	付款方式	客户订单编号	营销员
string	string	string		string	string	string	string
string	string	string		string	string	string	string

主表表格: 显示主表查询结果

gcSummary

(MainView) gvSummary (Click here to change view)

(Click here to create a new level)

记录: Retrieve Details Run Designer

数据查询页: 用于展示主表查询结果, 表格只读, 不可修改。

数据编辑页: 编辑表格中当前选择的记录 (单据), 显示详细资料, 用户在数据编辑页修改数据。

销售订单 (SO Sales Order)

操作提示 (Operation Tips) 关于 C/S 框架

数据查询 数据编辑

订单编号: SO20101103 版本: 01 制单人:

客户: 制单日期: 2010-10-29

接单日期: 2010-10-1 客户订单编号: C58765243 审批人:

付款方式: 营销员: 审批日期: 2010-10-29

币种: 金额: 88,888.88

备注: 无备注

业务单据主表编辑区

货品编码	货品名称	客户订单号	要求货期	单价	数量	金额	备注
string	string	string	string	string	string	string	string
string	string	string	string	string	string	string	string

业务单据明细表表格输入组件

gcDetail

(MainView) gvDetail (Click here to change view)

(Click here to create a new level)

Retrieve Details Run Designer

记录: Record 0 of 0

4.2.2 数据查询页及主表格(GridControl)控件

4.2.2.1 设计查询面板

从工具箱拖放 **PanelControl** 组件，Dock 设为 Top，在 PanelControl 内拖放一组查询条件输入框及查询按钮。

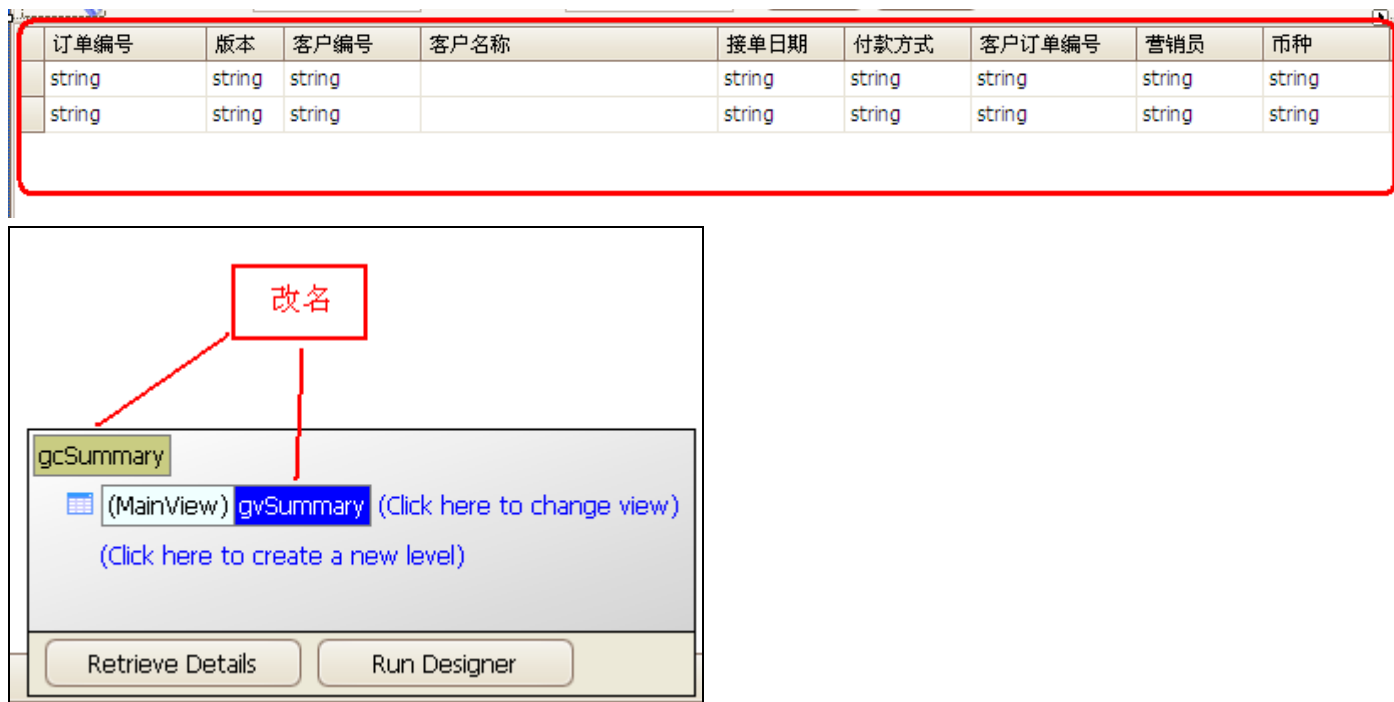
订单编号: 订单日期:

至: 至:

查询(S) 清空(E)

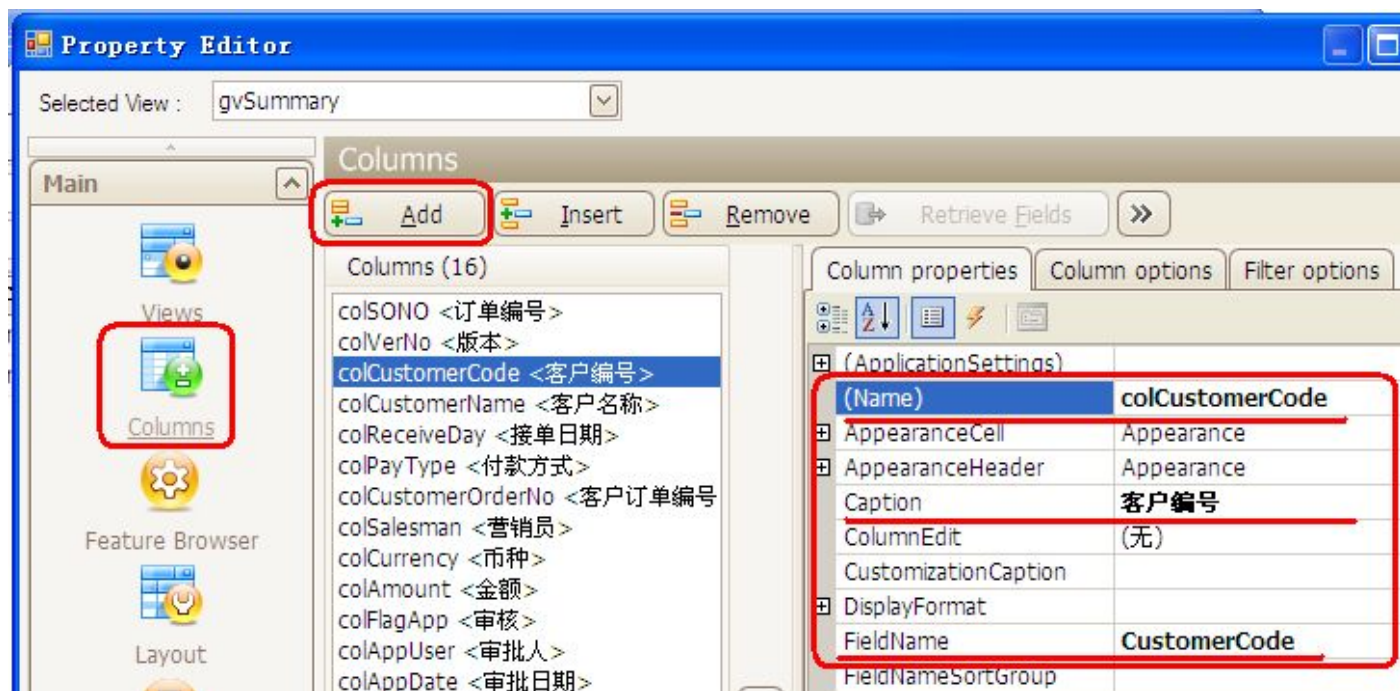
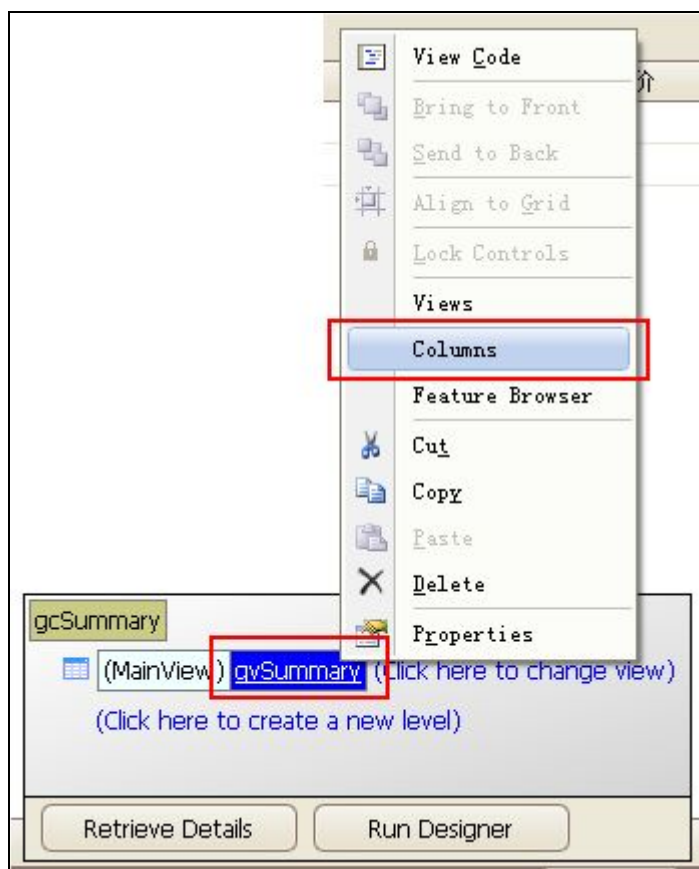
4.2.2.2 拖放主表表格

从工具箱拖放 **PanelControl** 组件，Dock 设为 Client，在 PanelControl 内拖放一个 GridControl 组件，然后改名，如下图：



GridControl 可以从其它窗体数据查询页复制过来，这样不需要设置参数，节省时间。

4.2.2.3 在 GridView(gvSummary)内定义表格的栏位

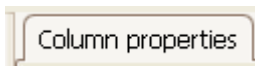




表格的栏位设计页面



新增一个栏位



栏位的属性设置。主要修改(Name),Caption,FieldName 三个属性。

属性名称	说明
(Name)	栏位名称(列名)。建议用 col+ 字段名命名，如：colCustomerCode
Caption	栏位的标题。如：客户编号
FieldName	栏位对应的字段名，用于绑定数据。如字段名：CustomerCode

关闭设计器，表格内定义了以下栏位：

A screenshot of a table design view. The table has 9 columns: 订单编号, 版本, 客户编号, 客户名称, 接单日期, 付款方式, 客户订单编号, 营销员, 币种. The first two rows of data are highlighted in light yellow. The table is enclosed in a red border.

订单编号	版本	客户编号	客户名称	接单日期	付款方式	客户订单编号	营销员	币种
string	string	string		string	string	string	string	string
string	string	string		string	string	string	string	string

4.2.3 数据编辑页面设计[主/从表输入区域]

通常业务单据只有一个明细表，设计效果如下：

销售订单 (SO Sales Order)

操作提示 (Operation Tips) 关于 C/S 框架

数据查询 数据编辑

订单编号: SO20101103 版本: 01 制单人:

客户: 制单日期: 2010-10-29

接单日期: 2010-10-1 客户订单编号: C58765243 审批人:

付款方式: 营销员: 审批日期: 2010-10-29

币种: 金额: 88,888.88

备注: 无备注

业务单据主表编辑区

增加 删除 另存为

货品编码	货品名称	客户订单号	要求货期	单价	数量	金额	备注
string	string	string	string	string	string	string	string
string	string	string	string	string	string	string	string

业务单据明细表表格输入组件

gcDetail

(MainView) gvDetail (Click here to change view)

(Click here to create a new level)

Retrieve Details Run Designer

记录: Record 0 of 0

有些业务单据比较复杂，比如多个明细表的，设计效果如下：

收回委托加工 (Receive DP)

操作提示(Operation Tips)

数据查询 数据编辑

收回加工编号: 加工单位: 期间: ☐ 锁定
加工合同号: 制单人: 核对人: ☐ 作废
加工日期: 制单日期: 核对日期: ☐ 月结
完成日期: 销售人员: 货币:
发票类型: 发票号码: 发票总金额:
委托加工单号: 客户税号: 运费:
备注: 杂费:
本位币金额:
实际总金额:

加工前产品编号	加工前产品名称	查看条码	生产厂商	数量单位	加工前数量	重量单位	加工前
string	string	string			string		
string	string	string			string		

收回加工后产品 收回结余/回卷产品

收回仓库	装船编号	产品编号	生产厂商	查看条码	数量单位	重量单位	单重单位
	string	string		string			
	string	string		string			

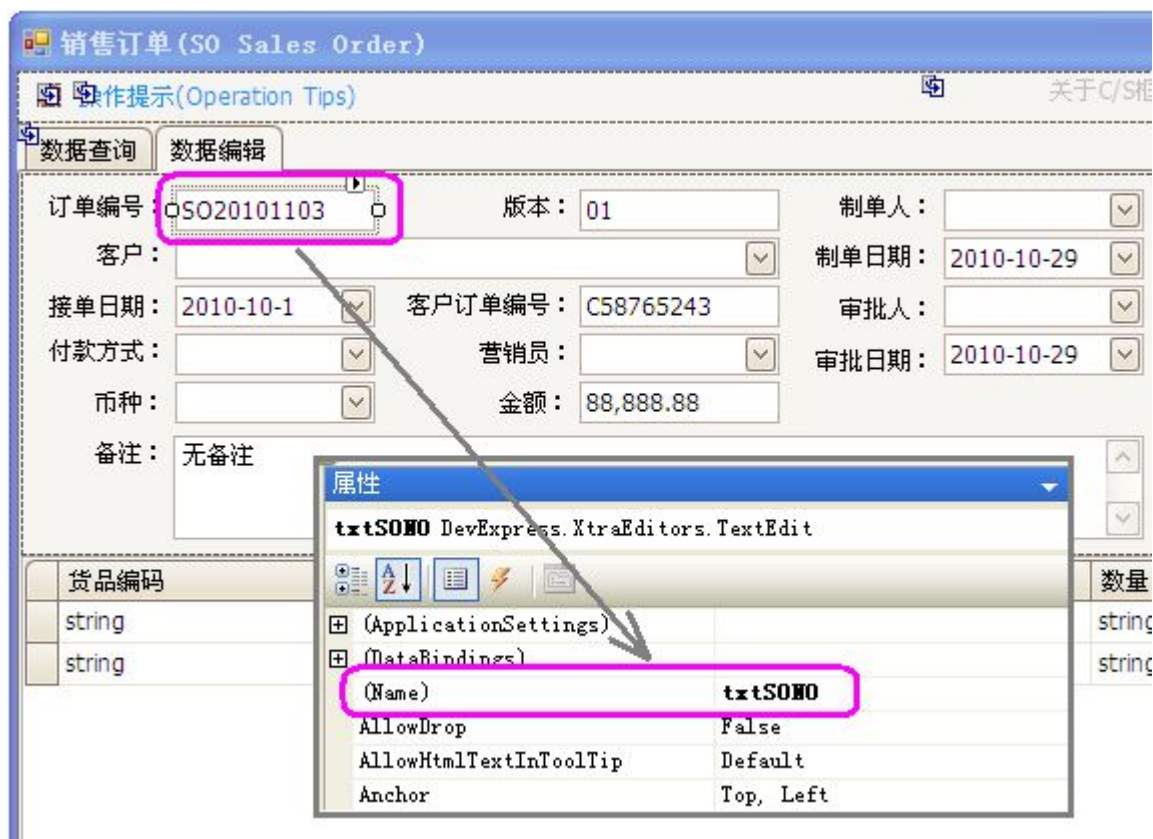
gcDet Retri

gcAfterStock (MainView)

此委托加工单有 3 个明细表，其中两个明细表用 TabPage 收缩到一个区域。

4.2.4 主表输入组件设计

输入框建议以 txt+字段名命名，如 SONO 销售订单字段命名为 txtSONO，CheckBox 用缩写 chk，如 chkAPP 审核。



4.3 使用代码生成器工具生成 ORM、BLL、DAL 代码

代码生成器可生成 ORM 实体及 DAL, BLL 层代码模板。单表如数据字典生成的代码几乎不需要修改即可运行，业务单据功能较复杂，除框架实现的功能外，需要用户扩展方法。SO 功能扩展的方法有 CheckNoExists, GetSummaryByParam, GetReportData。

开发框架仅能生成 BLL, DAL 代码模板，实际上用户需要跟据业务需求自行扩展代码。通过修改代码生成器的模板部分，用户可以生成适合当前项目的代码。

代码生成步骤：ORM->DAL->BLL。

4.3.1 生成 ORM

4.3.2 生成 DAL

4.3.3 生成 BLL

生成后将代码加入解决方案，尝试编译。

4.3.1 参考视频：

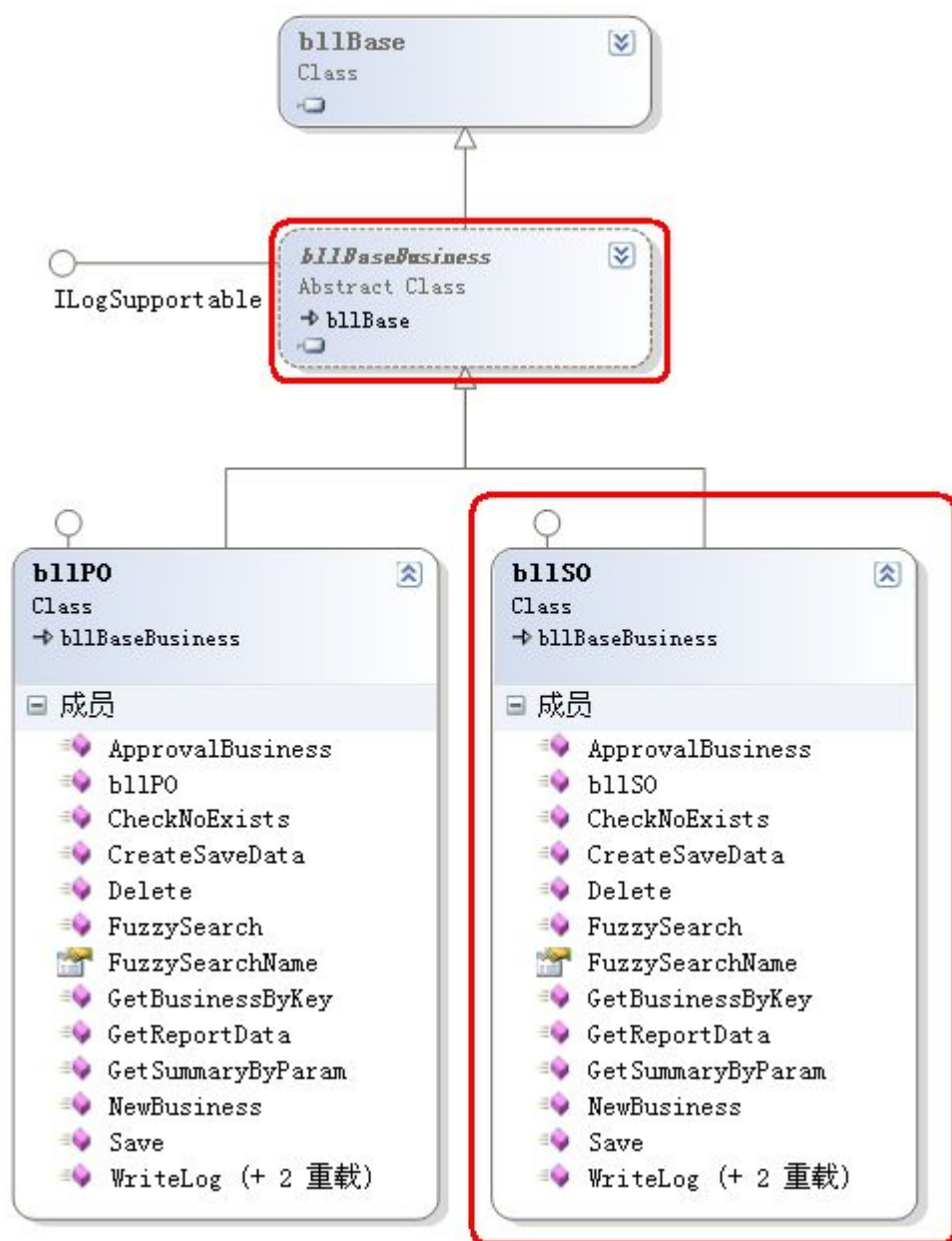
[C/S 快速开发框架 - 代码自动生成视频下载](#)

[CS 系统快速开发框架 - 生成 BLL 和 DAL](#)

4.4 BLL 层代码

BLL, Business Logic Layer 业务逻辑层。关于 [C/S 结构开发框架中 BLL 层](http://www.csframework.com/archive/1/arc-1-20110414-1332.htm)请参考:
<http://www.csframework.com/archive/1/arc-1-20110414-1332.htm>

4.4.1 BLL 继承关系图







如上图所示，**bllBase** 是所有业务层的基类，**bllBaseBusiness** 是业务单据的业务逻辑层基类，实现 **ILogSupportable** 接口，该接口用于处理修改日志功能。

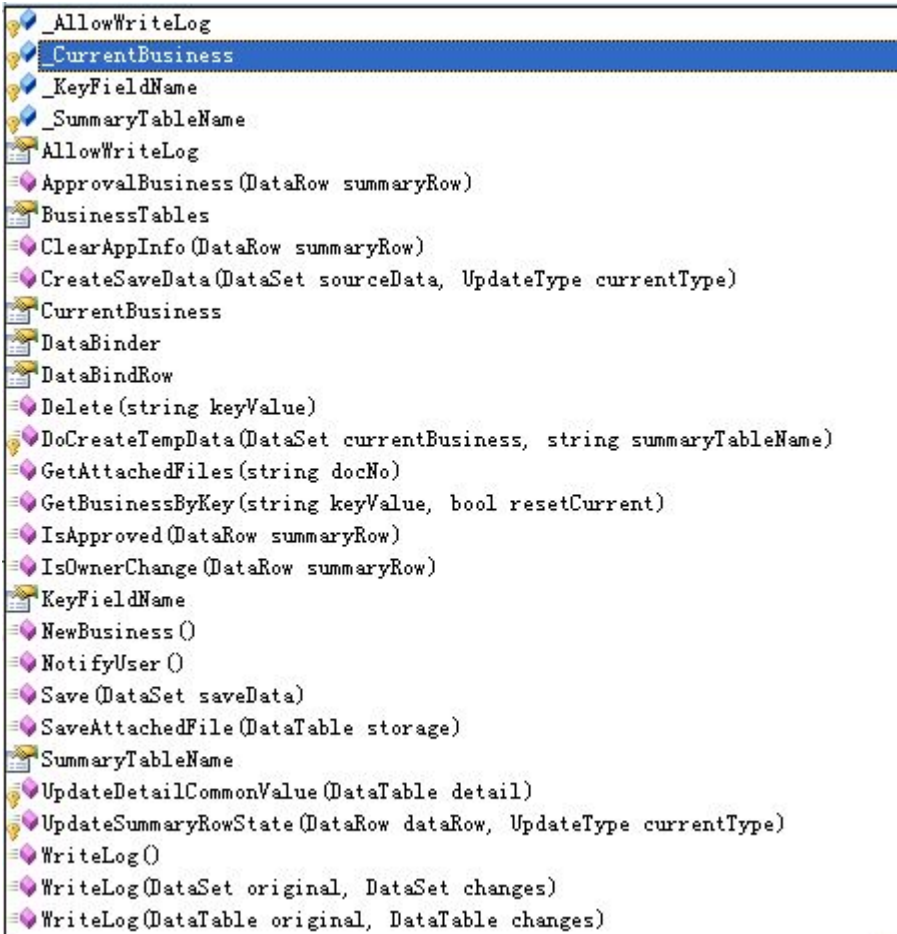
所有业务单据的逻辑层从 `bllBaseBusiness` 派生, 如 `bllPO`, `bllSO`。

4.4.2 `bllBaseBusiness` 类

`bllBaseBusiness` 类是业务单据的业务逻辑层基类, 实现了业务单据所需的公共方法, 虚方法便于派生类重写功能。
`bllBaseBusiness` 类说明:

ID	名称	类型	说明
1	<code>AllowWriteLog</code>		标记允许写入修改日志。
2	<code>ApprovalBusiness</code>		审核功能的方法。
3	<code>BusinessTables</code>		对应当前业务单据的数据。 <code>DataTableCollection</code> 类型, 第 1 个表 (Index 为 0) 为主表, 其它为明细表。
4	<code>ClearAppInfo</code>		清空审核功能相关字段的数据。
5	<code>CreateSaveData</code>		创建用于保存的临时数据。
6	<code>CurrentBusiness</code>		当前业务单据的数据。
7	<code>DataBinder</code>		对应当前单据的主表数据 (<code>DataTable</code>), 该表只有一条记录, 用于绑定[数据修改页面]输入组件的数据源。
8	<code>DataBindRow</code>		<code>DataRow</code> 类型, 对应 <code>DataBinder</code> 的第一条记录。
9	<code>Delete</code>		删除单据。
10	<code>DoCreateTempData</code>		创建用于保存的临时数据。该方法由 <code>CreateSaveData</code> 方法调用。
11	<code>GetAttachedFiles</code>		获取单据附件数据。
12	<code>GetBusinessByKey</code>		下载指定单号的业务数据。
13	<code>IsApproved</code>		检查业务单据是否审核。
14	<code>IsOwnerChange</code>		比较当前登录用户与制单人是否一致。
15	<code>KeyFieldName</code>		业务单据的主键字段名。
16	<code>SummaryTableName</code>		业务单据的主表名称。
17	<code>NewBusiness</code>		新增业务数据。
18	<code>NotifyUser</code>		当单据被修改发送手机短信或 Email 给制单人。
19	<code>Save</code>		保存业务数据。
20	<code>SaveAttachedFile</code>		保存附件。
21	<code>UpdateDetailCommonValue</code>		更新业务单据明细表的公共字段数据。
22	<code>UpdateSummaryRowState</code>		更新记录状态。
23	<code>WriteLog</code>		保存修改日志。

类成员截图:



4.4.3 bllSO 类

SO, Sales Order 销售订单, 继承 bllBaseBusiness 基类。

bllSO 类说明:

ID	名称	类型	说明
1	ApprovalBusiness	重载的方法	重载的方法, 用于审核单据。
2	CheckNoExists	重载的方法	检查单据号码是否存在。
3	CreateSaveData	重载的方法	重载的方法, 创建用于保存的临时数据。
4	Delete	重载的方法	重载的方法, 删除单据。
5	FuzzySearch	重载的方法	产品资料模糊查询。实现 IfuzzySearchSupportable 接口, 在 frmFuzzySearch 窗体中使用。
6	FuzzySearchName	模糊查询功能的名称	模糊查询功能的名称, 如: 产品资料查询。在 frmFuzzySearch 窗体的标题显示。
7	GetReportData	重载的方法	获取报表数据。
8	GetBusinessByKey	重载的方法	重载的方法, 下载指定单号的业务数据。
9	GetSummaryByParam	重载的方法	查询销售订单的主表数据, 查询结果在主表格中显示。
10	NewBusiness	重载的方法	重载的方法, 新增业务数据。
11	Save	重载的方法	重载的方法, 保存数据。
12	WriteLog	重载的方法	重载的方法, 保存修改日志。

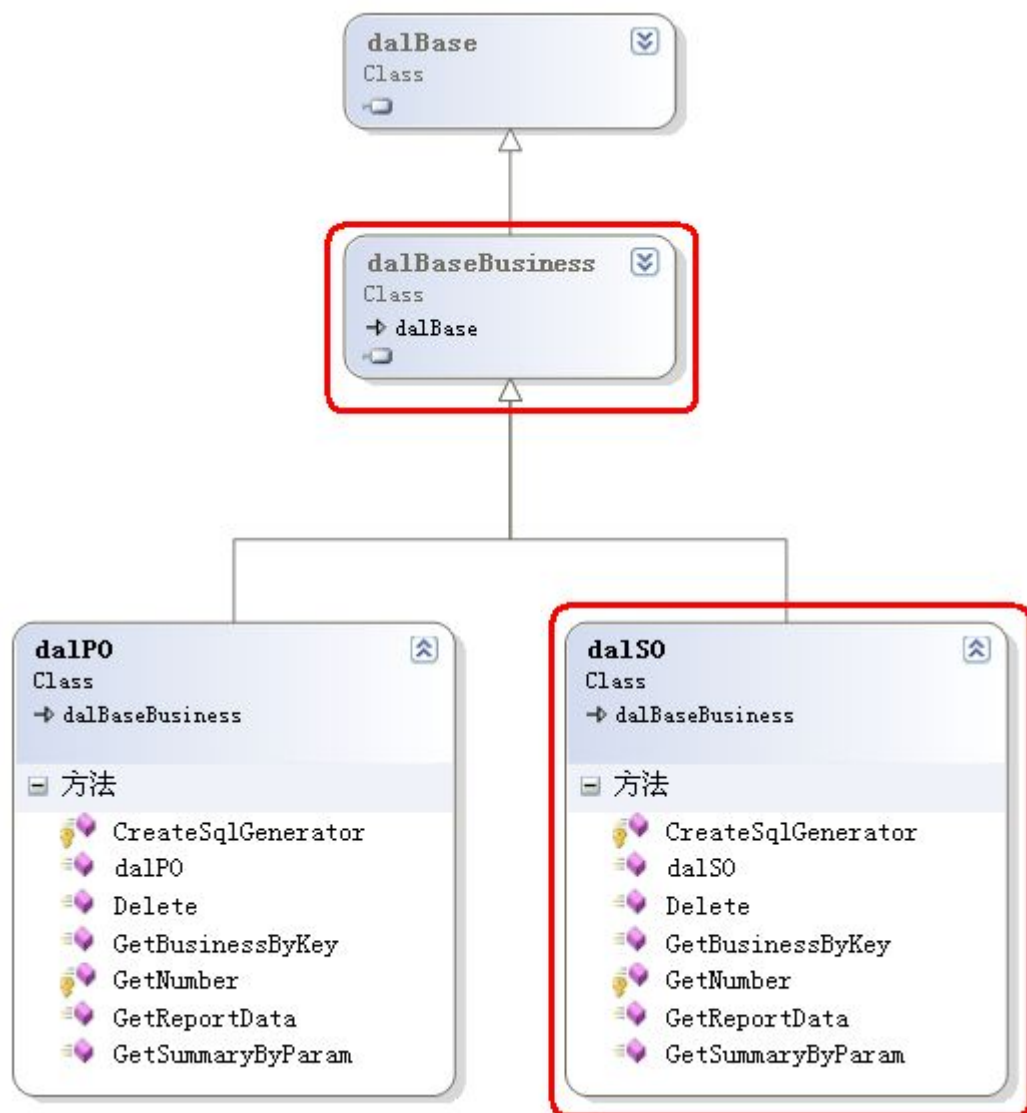
类成员截图:

```
ApprovalBusiness(DataRow summaryRow)
bllSO()
CheckNoExists(string keyValue)
CreateSaveData(DataSet currentBusiness, UpdateType currentType)
Delete(string keyValue)
FuzzySearch(string content)
FuzzySearchName
GetBusinessByKey(string keyValue, bool resetCurrent)
GetReportData(string DocNoFrom, string DocNoTo, DateTime DateFrom, DateTime DateTo)
GetSummaryByParam(string docNoFrom, string docNoTo, DateTime docDateFrom, DateTime docDateTo)
NewBusiness()
Save(DataSet saveData)
WriteLog()
WriteLog(DataSet original, DataSet changes)
WriteLog(DataTable original, DataTable changes)
```

4.5 DAL 层代码

DAL, Data Access Layer 数据访问层。关于 [C/S 开发框架中数据访问层\(DAL\)](http://www.csframework.com/archive/1/arc-1-20110414-1333.htm)的作用请参考:
<http://www.csframework.com/archive/1/arc-1-20110414-1333.htm>

4.5.1 DAL 继承关系图



上图所示，dalBase 是数据访问层的基类，dalBaseBusiness 是业务单据数据层的基类，业务单据的 DAL 从 dalBaseBusiness 类继承，如 dalPO，dalSO。

4.5.2 dalBaseBusiness 类

ID	名称	类型	说明
1	_SummaryKeyName		主表主键字段名。
2	_SummaryTableName		主表表名。
3	_UpdateSummaryKeyMode		业务单据号码更新类型(自动生成流水号或 GUID)。
4	ApprovalBusiness		单据审核/批准功能。。
5	CheckNoExists		检查业务数据是否存在。
6	CreateSqlGenerator		虚方法，根据表名获取该表的 ORM 及对应的 SQL 命令生成器。
7	GetNumber		获取单据流水号码。
8	Update		保存数据，对应 BLL.Save 方法
9	UpdateDetailKey		更新明细表的外键。
10	UpdateSummaryKey		更新主表主键。

类成员截图:

```
SummaryKeyName
SummaryTableName
UpdateSummaryKeyMode
ApprovalBusiness(string keyValue, string flagApp, string appUser, DateTime appDate)
CheckNoExists(string keyValue)
CreateSqlGenerator(string tableName)
dalBaseBusiness(Loginer loginer)
GetNumber(SqlTransaction tran)
Update(DataSet data)
UpdateDetailKey(DataTable detail, string foreignFieldName, string foreignKeyValue)
UpdateSummaryKey(SqlTransaction tran, DataTable summary, UpdateKeyMode model, string keyFieldName,
```

4.5.3 dalSO 类

类成员列表

ID	名称	类型	说明
1	CreateSqlGenerator	⇒	重载的方法，根据表名获取该表的 ORM 及对应的 SQL 命令生成器。
2	Delete	⇒	删除单据，开发框架的模板窗体仅删除明细表，将主表数字类型的字段清零，如金额，数量等关键字段，保留主表的记录避免断号。
3	GetBusinessByKey	⇒	获取一张业务单据的数据。
4	GetNumber	⇒	获取单据流水号码。
5	GetReportData	⇒	根据查询条件获取报表数据。
6	GetSummaryByParam	⇒	查询销售订单的主表数据。

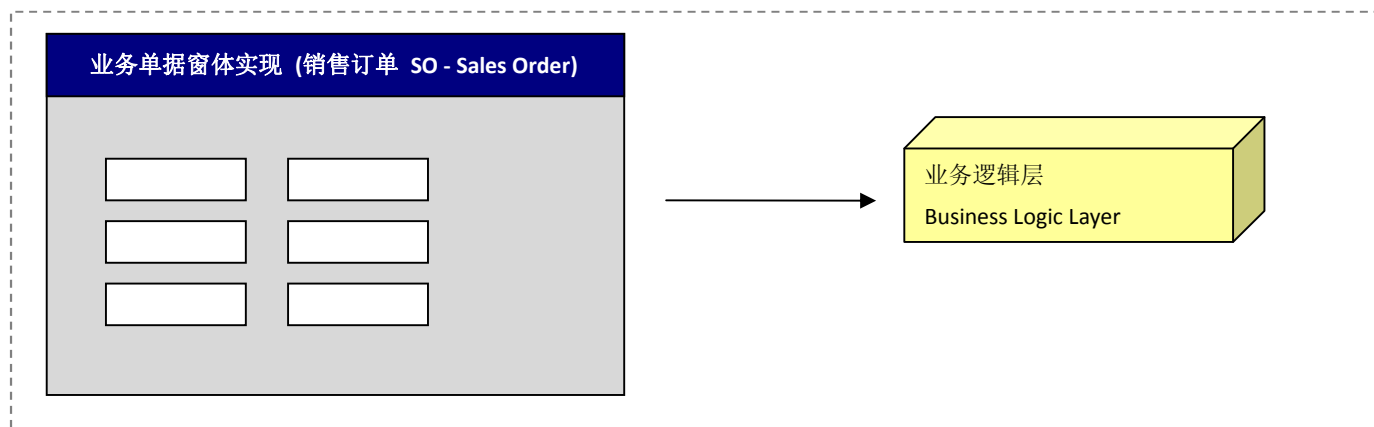
类成员截图:

```
CreateSqlGenerator(string tableName)
dalSO(Loginer loginer)
Delete(string docNo)
GetBusinessByKey(string docNo)
GetNumber(SqlTransaction tran)
GetReportData(string DocNoFrom, string DocNoTo, DateTime DateFrom, DateTime DateTo)
GetSummaryByParam(string docNoFrom, string docNoTo, DateTime docDateFrom, DateTime docDateTo)
```

4.6 编写窗体代码

设计好窗体上的组件及界面布局，然后使用工具生成 ORM，BLL，DAL 层的代码，最后手工编写窗体内的代码。开发框架的数据窗体模板就两种，为了快速开发我们可以直接引用 frmSO,frmPO 窗体的代码。

窗体与业务逻辑层的引用关系:



开发框架将用户界面与业务逻辑代码分离出来的，所以任何与业务有关的操作建议封装到业务层，大大减少用户界面的代码量，使界面与业务代码松散耦合，用户界面仅保留操作本窗体组件的代码及引用业务层的代码。

4.6.1 初始化业务窗体(重写 InitializeForm 方法)

InitializeForm()是 frmBaseDataForm 窗体中定义的虚方法，派生的窗体必须在 Form.Load 事件中调用。

InitializeForm 方法主要功能是：

1. 实例化业务层_BLL 变量。
2. 调用公共方法设置表格样式。
3. 绑定主表表格及明细表表格的事件。
4. 绑定数据参照组件(LookUpEdit)的数据源。
5. 加载一个空的业务对象，用于测试网络环境及初始化数据。

❗ 请参考 frmSO. InitializeForm 方法

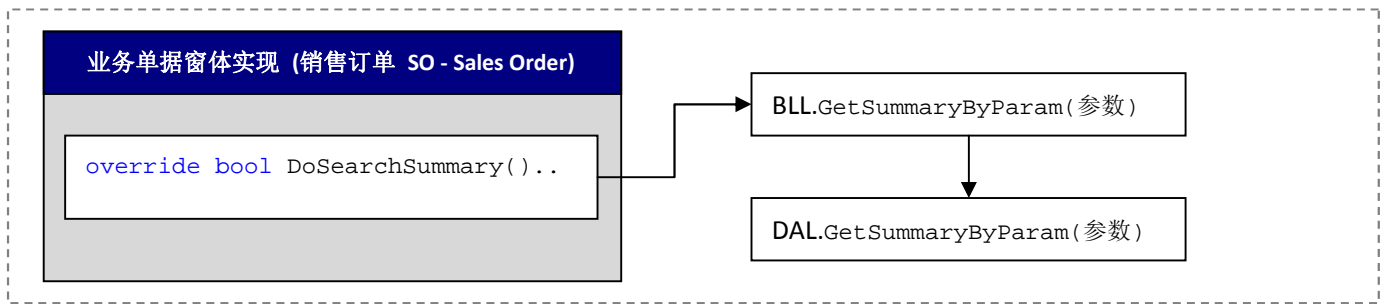
4.6.2 Form.Load 事件调用 InitializeForm 方法

```
private void frmSO_Load(object sender, EventArgs e)
{
    this.InitializeForm(); //初始化主窗体
}
```

4.6.3 实现查询功能(DoSearchSummary 方法)

首先实现查询功能显示主表的数据，非常简单，在窗体内重载 DoSearchSummary 方法，然后调用 BLL 层的 GetSummaryByParam 方法，该方法的参数跟据查询条件定义，通常查询单据号码，日期范围及客户编号。

流程如下：



❗ 请参考 frmSO. DoSearchSummary 方法

4.6.4 绑定主表输入组件数据源(DoBindingSummaryEditor 方法)

实现 DoSearchSummary 方法后可查询数据, 现在我们需要展示业务单据的详细资料。销售订单只有一个从表, 用一个表格展示明细数据。主表的资料用输入框展示。重写 DoBindingSummaryEditor 方法绑定输入框的数据源。

❗ 请参考 frmSO. DoBindingSummaryEditor 方法

4.6.5 绑定明细表格的数据源(DoBindingDetailGrid 方法)

重写 DoBindingDetailGrid 方法绑定销售订单明细表的数据源。DoBindingDetailGrid 是 frmBaseBusinessForm 窗体中定义的虚方法。

❗ 请参考 frmSO. DoBindingDetailGrid 方法

4.6.6 增加(DoAdd 方法)

销售订单不需要直接实现 DoAdd 方法, 在 frmBaseBusinessForm 窗体中该方法调用业务层的虚方法, 所以我们只需要实现派生类 bllSO 中相关方法就行。

销售订单的新增功能处理流程:

1. 调用业务层的 GetBusinessByKey 方法, 下载一个空的业务单据 DataSet。
2. 调用业务层的 NewBusiness 方法, 新增一条记录并初始化数据, 如单据日期, 制单人等预设值。
3. 绑定主表, 明细表的数据源。
4. 将状态改为 UpdateType.Add。
5. 显示详细资料页面。

❗ 请参考 frmBaseBusinessForm. DoBindingDetailGrid 方法
❗ 及 bllSO. GetBusinessByKey, NewBusiness

4.6.7 删除(DoDelete 方法)

销售订单不需要直接实现 DoDelete 方法, 在 frmBaseBusinessForm 窗体中该方法调用业务层的虚方法, 所以我们只需要实现派生类 bIISO 中相关方法就行。

销售订单删除功能处理流程:

1. 调用 BLL 的 IsApproved 方法检查单据是否已经审核, 已审核的单据禁止删除。
2. 调用 BLL 的 Delete 方法删除销售订单的明细表及将主表的金额等字段的数据置零。
3. 如后台数据删除成功, 将主表格当前记录的金额等字段的数据置零。

❗ 请参考 frmBaseBusinessForm. DoDelete 方法
❗ 及 bIISO. IsApproved, Delete

4.6.8 修改(DoEdit 方法)

销售订单不需要直接实现 DoEdit 方法, 在 frmBaseBusinessForm 窗体中该方法调用业务层的虚方法, 所以我们只需要实现派生类 bIISO 中相关方法就行。

销售订单的修改功能处理流程:

1. 调用 BLL 的 IsOwnerChange 方法检查单据的制单人是否为当前用户, 不能修改他人创建的单据。
2. 调用 BLL 的 IsApproved 方法检查单据是否已经审核, 已审核的单据禁止删除。
3. 调用 DoViewContent 方法显示当前单据的详细资料。
4. 将状态改为 UpdateType.Modify。
5. 显示详细资料页面。

❗ 请参考 frmBaseBusinessForm. DoEdit 方法
❗ 及 bIISO. IsApproved, IsOwnerChange

4.6.9 查看(DoViewContent 方法)

销售订单不需要直接实现 DoViewContent 方法, 在 frmBaseBusinessForm 窗体中该方法调用业务层的虚方法, 所以我们只需要实现派生类 bIISO 中相关方法就行。

销售订单的查看功能处理流程:

1. 调用业务层的 GetBusinessByKey 方法下载业务单据数据集 DataSet, 参数: 当前单据号码。
2. 绑定主表, 明细表的数据源。
3. 将状态改为 UpdateType.None。
4. 显示详细资料页面。

❗ 请参考 frmBaseBusinessForm. DoViewContent 方法
❗ 及 bIISO. GetBusinessByKey

4.6.10 取消(DoCancel 方法)

销售订单不需要直接实现 DoCancel 方法, 在 frmBaseBusinessForm 窗体中已实现具体功能。

销售订单的取消功能处理流程:

1. 调用 DoViewContent 方法重新显示当前单据。
2. 将状态改为 UpdateType.None。

! 请参考 frmBaseBusinessForm. DoCancel 方法

4.6.11 打印(DoPrint 方法)

打开打印窗体。

! 请参考 frmSO. DoPrint 方法

4.6.12 审核(DoApproval 方法)

销售订单不需要直接实现 DoApproval 方法, 在 frmBaseBusinessForm 窗体中该方法调用业务层的虚方法, 所以我们只需要实现派生类 bllSO 中相关方法就行。

销售订单的审核功能处理流程:

1. 调用 BLL 的 IsApproved 方法检查单据是否已经审核, 已审核的单据不能重复审核。
2. 调用 BLL 的 ApprovalBusiness 方法审核单据。

! 请参考 frmBaseBusinessForm. DoApproval 方法及 bllSO. ApprovalBusiness

4.6.13 保存(DoSave 方法)

销售订单的保存功能处理流程:

1. 更新最后获得焦点输入框的数据及主表的汇总金额。
2. 主、从表的数据合法性检查。
3. 保存本次修改日志记录。
4. 创建用于保存的临时数据集。bllBaseBusiness 类的 _CurrentBusiness 成员是当前正在处理的业务数据, 通常在保存前会做些特殊处理, 如设置单号, 某些日期及单据的关联数据, 为了不影响原始数据, 我们在保存前复制一个副本用于保存, 即使保存失败也不影响当前正在处理的业务数据。
5. 调用 BLL.Save 方法。
6. 如保存成功, 前台相关处理。
7. 将状态改为 UpdateType.None。

! 请参考 frmSO. DoSave 方法及 bllSO. Save

4.6.14 主表数据合法性检查(ValidatingSummaryData 方法)

! 请参考 frmSO. ValidatingSummaryData 方法

4.6.15 明细表数据合法性检查(ValidatingDetailData 方法)

! 请参考 frmSO. ValidatingDetailData 方法

4.6.16 显示修改历史记录(DoShowModifyHistory 方法)

打开 frmModifyLog 窗体显示修改历史记录。

! 请参考 frmSO. DoShowModifyHistory 方法

4.6.17 其它细节处理

ID	名称	类型	说明
1	ButtonAuthorized	⇒	frmBaseChild 窗体中定义的虚方法。派生类通过重写该方法可以检查按钮的权限并隐藏工具栏的按钮。
2	ButtonStateChanged	⇒	frmBaseDataForm 窗体中定义的虚方法。当按钮状态发生变化时调用，可以重写此事件禁用某些输入组件。
3	CreateOneDetail	⇒	<p>创建一条明细记录，适用于有明细表的业务窗体。此方法由 OnEmbeddedNavigatorButtonClick 事件调用。</p> <p>! 明细表格的导航按钮新增了三个自定义按钮，如下图：</p>  <p>分别是新增，插入，删除。在 InitializeForm 方法中调用 DevStyle.SetGridControlLayout(gcDetail, true) 方法创建 3 个按钮，但必须将第 2 个参数 allowEdit 为 true。</p> <p>CreateOneDetail 方法的 buttonIndex 参数对应按钮的序号，分别为 0:Add,1:Insert,2:Delete，在枚举类型 DetailButtons 中定义。</p>

			插入记录功能需要定义 Queue 字段，带符号的数字类型，用于排序。
4	OnCellValueChanged	⇒	明细表格的 OnCellValueChanged 事件，当用户修改明细表单元格的数据时触发该事件。可以在此事件内计算总金额（金额=单价*数量）
5	OnStockCode_ButtonClick	⇒	<p>明细表的货品栏位支持选择功能，将栏位设置为 ButtonEdit 类型。</p>  <p>运行效果图如下：</p>  <p>点 ButtonEdit 的按钮触发 OnStockCode_ButtonClick 事件。 在 InitializeForm 方法中绑定 OnStockCode_ButtonClick 事件。</p>
6	OnStockCode_Validating	⇒	<p>用户可以选择或输入货品编码，当用户输入货品编码后离开当前单元格时触发 OnStockCode_Validating 事件。</p>  <p>在 InitializeForm 方法中绑定 OnStockCode_Validating 事件。</p>
7	SearchStockCallBack	⇒	跟据货品编号获取货品详细资料，该方法由 OnStockCode_ButtonClick 和 OnStockCode_Validating 事件调用。
8	SetDetailEditorsAccessable	⇒	在 frmBaseDataForm 窗体中定义的虚方法，用于控制明细页面上的控件可被编辑，同时禁用或启用明细表三个自定义按钮。
9	UpdateDetailAmount	⇒	自动计算明细表的金额合计，在 OnCellValueChanged 事件中调用。
10	UpdateTotalAmount	⇒	<p>自动计算主表的总金额。 主表的总金额=SUM(明细表的金额)。 通常在保存前会重新计算主表的总金额。</p>

五. 相关文章及视频参考

5.1 C/S 框架网视频下载

<http://www.csframework.com/category-list-video.htm>

5.2 C/S 快速开发框架 - 代码自动生成视频下载

<http://www.csframework.com/archive/6/arc-6-20110407-1306.htm>

5.3 C/S 系统快速开发框架 - 生成 BLL 和 DAL

<http://www.csframework.com/archive/6/arc-6-20110507-1425.htm>

5.4 C/S 框架业务窗体设计-实现查询功能

<http://www.csframework.com/archive/6/arc-6-20110507-1426.htm>

5.5 [C/S 系统快速开发框架 - 生成主从表的 ORM](#)

<http://www.csframework.com/archive/6/arc-6-20110507-1424.htm>

5.6 [C/S 系统快速开发框架 - 新增业务窗体\(视频\)](#)

<http://www.csframework.com/archive/6/arc-6-20110507-1423.htm>

5.7 [C/S 快速开发框架- 新增业务模块\(视频下载\)](#)

<http://www.csframework.com/archive/6/arc-6-20110412-1325.htm>

注意: 开发框架内提供了几个数据模板窗体,
数据字典窗体请参考《客户管理》和《产品管理》
业务单据窗体请参考采购模块的《采购订单》及销售模块的《销售订单》。

六. 意见或建议

QQ: 23404761

Email: jonnysun@csframework.com

C/S 框架网

www.csframework.com

Feedback 意见反馈 – 您的意见是我们进步的根源!

<http://www.csframework.com/feedback.htm>

Feedback

意见反馈 – 您的建议是我们进步根源!

www.csframework.com

C/S框架网版权所有