

## C/S 框架开发文档之开发框架与模块

By Jonny Sun at 2010-08-29

(doc v1.0)

### 目 录

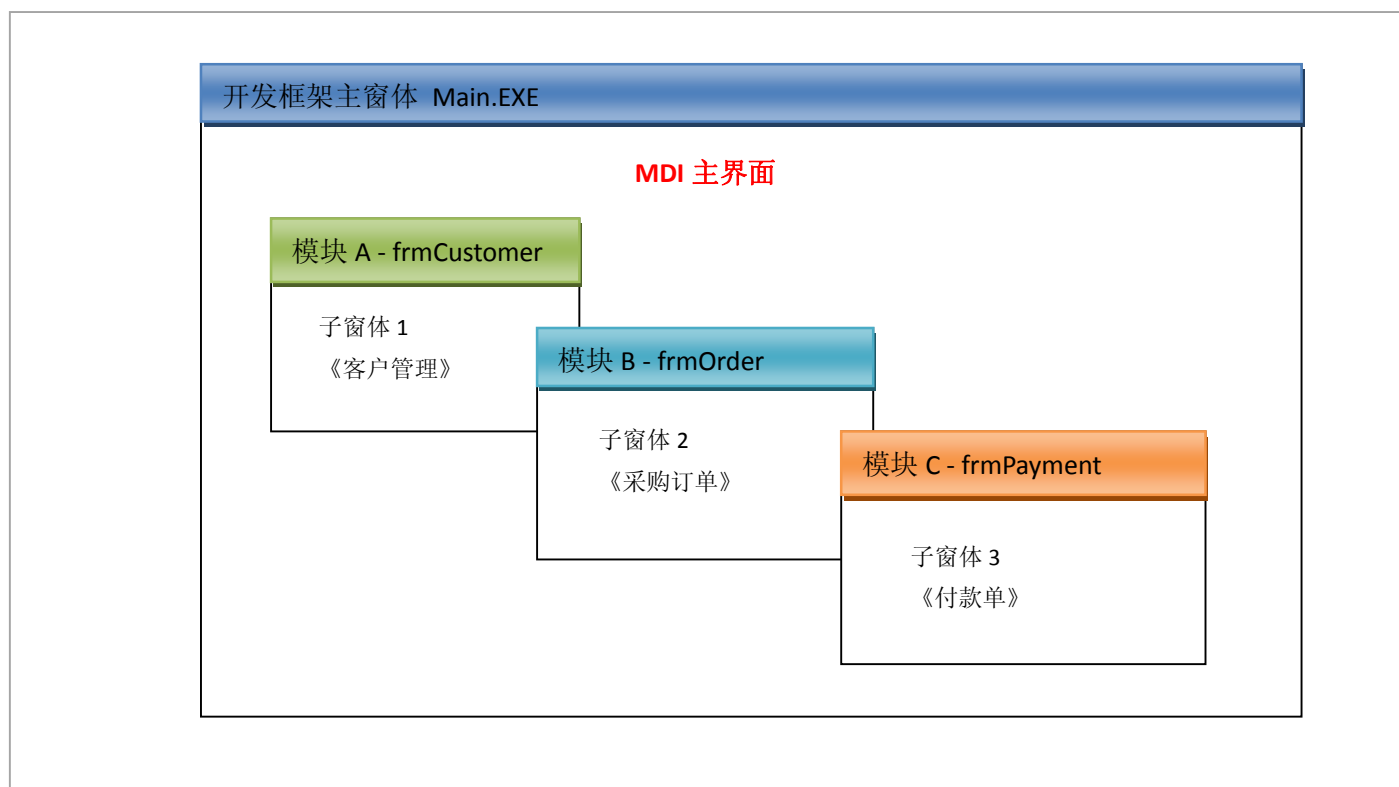
Index

|                                     |    |
|-------------------------------------|----|
| 一. 开发框架介绍.....                      | 2  |
| 1.1 开发框架与 MDI.....                  | 2  |
| 1.2 开发框架与模块.....                    | 2  |
| 二. 开发框架的逻辑组织结构.....                 | 4  |
| 2.1 逻辑组织结构图.....                    | 4  |
| 2.2 业务模块与系统库引用关系.....               | 5  |
| 2.3 客户端模块引用层次(最高层仅引用.Net 程序集).....  | 6  |
| 2.4 服务端模块引用层次(最高层仅引用.Net 程序集).....  | 7  |
| 2.5 高级版 Client/Server 模块的源码共用表..... | 7  |
| 三. 系统库.....                         | 8  |
| 四. 数据字典模块.....                      | 8  |
| 五. 业务模块.....                        | 9  |
| 六. 系统管理模块.....                      | 11 |
| 七. 系统主窗体与模块集成.....                  | 11 |
| 7.1 系统主窗体 (frmMain) .....           | 11 |
| 7.2 主窗体与业务模块组织关系.....               | 15 |
| 7.3 业务模块动态加载.....                   | 16 |
| 7.3.1 简要说明.....                     | 16 |
| 7.3.2 业务模块动态加载部分源码.....             | 16 |
| 八. 提意见或建议.....                      | 18 |

# 一. 开发框架介绍

## 1.1 开发框架与 MDI

开发框架是一套基于多档界面(MDI)的用户解决方案。MDI 全称是多文档界面 (Multiple Document Interface), 主要应用于基于图形用户界面的系统中。其目的是同时打开和显示多个文档, 便于参考和编辑资料。并非所有基于图形用户界面的软件都具有 MDI, 许多软件, 如 MS Windows 下的记事本, 在一个运行中只能打开一个文档, 因此不能被称为具有 MDI 的软件。但是 Windows Explorer 则可以同时打开多个文件窗口 (文档), 因此是 MDI 软件。MDI 可以使用多种方式组织文档, 包括窗口, 标签 (tab), 缓冲区 (buffer) 等。(摘自维基百科)



## 1.2 开发框架与模块

开发框架的逻辑组织结构是由多个模块构成的, 框架的模块是一个标准的类库(Class Library), 存储在一个 DLL 物理文件中。框架把模块分为三大类型: 1. 系统公共库, 2. 系统模块, 3. 功能模块。

公共库是指 Common, Library, Interfaces 等不包涵用户操作界面的 DLL 库, 是开发框架必须依赖的系统库, 缺一不可。

开发框架定义了一个系统模块, 命名 CSFramework.SystemModule。系统模块的主要功能是权限分

配、系统设置、公司资料设置、系统初始化等功能。

系统模块和功能模块是指包涵模块主窗体和 UI 操作功能的窗体，并且能集成到框架主窗体中的模块。系统模块和功能模块是动态加载的，即使丢失模块的 DLL 文件框架仍能运行，但不能进行某些操作，如丢失系统模块则不能分配权限和系统设置等，如丢失功能模块，用户无法进行具体的业务操作，如货币定义，客户管理，采购，销售，盘点等。

当运行程序时，系统模块和业务模块会动态加载并集成到程序的主窗体中，登录界面会动态显示模块的加载进度信息。

如果按业务逻辑细分，功能模块又可分为数据字典模块和业务模块。数据字典模块主要集中管理基础数据操作的窗体，如客户管理，货品定义，货币定义，国家地区，付款方式等，这些功能有个基本特点都是单表操作，把这种类型的操作窗体集成到数据字典模块内。

业务模块涉及到业务操作，有个鲜明特点是业务模块只集成业务单据，如采购单，销售单，送货单，付款单等。业务单据的数据结构大多是主从表的，业务窗体操作比较复杂，所以用户界面相对比数据字典要复杂的多。

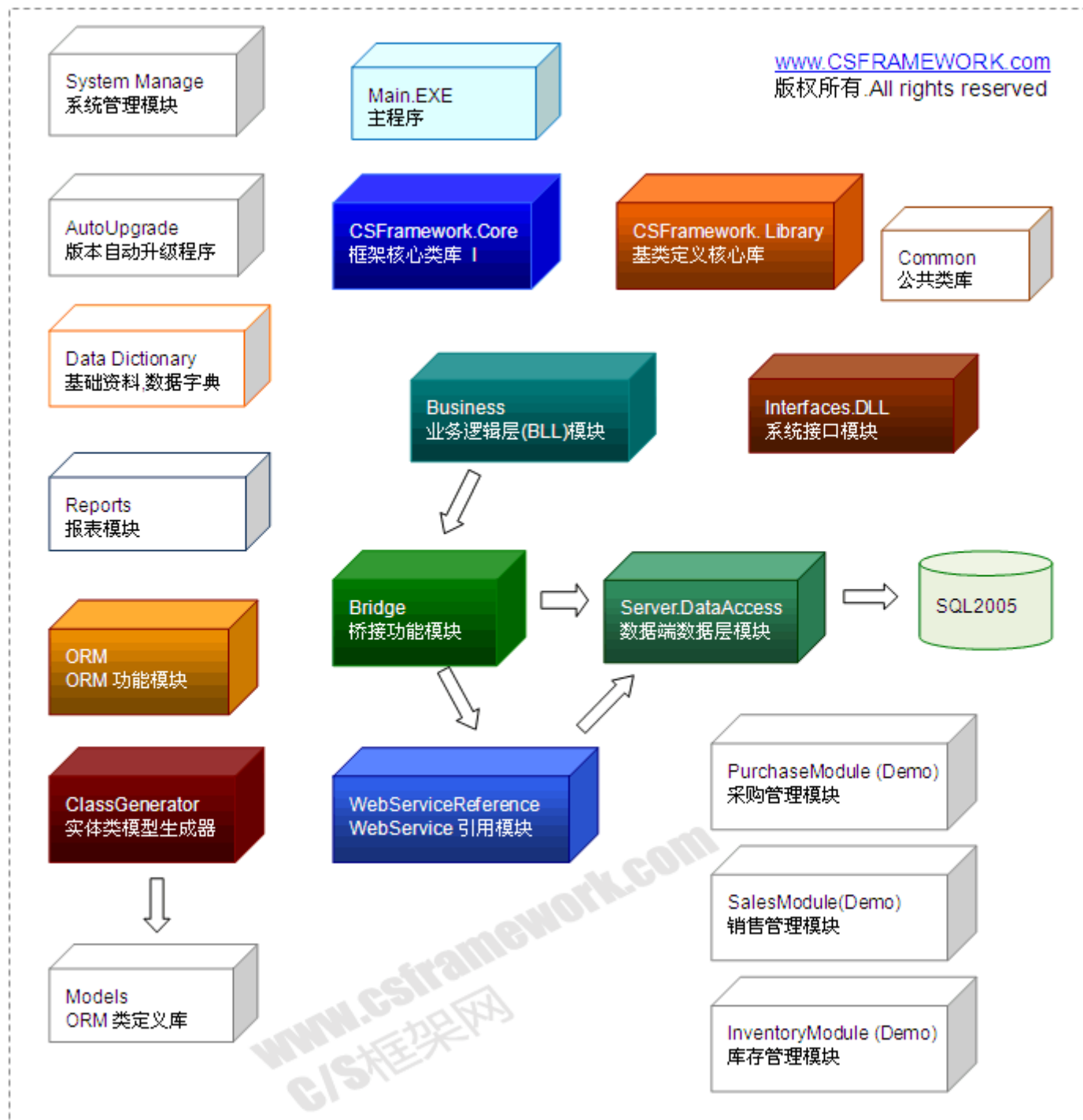
考虑到用户操作良好体验和易学易用，开发框架设计了 1 个系统管理模块，2 个数据字典模块和 3 个业务模块（进/销/存）。另外需要强调的是，以上模块的分类是作者本人基于框架的合理组织而设计的，用户并不需要严格遵守这种设计，可以按照实际需求来设计划分模块。

通过以上介绍，开发框架与业务模块是松散耦合的，还可以根据不同的部门部署不同的模块文件，如采购部门，只部署系统库和 PurchaseModule.DLL (采购模块)，财务部门只部署 AccountModule.DLL (财务管理模块)，销售部门只部署 SalesModule.DLL (销售模块)。这样有几个好处：一是方便分配权限，二是简化了客户端程序安装和提高运行速度，三是增强系统的安全性。

## 二. 开发框架的逻辑组织结构

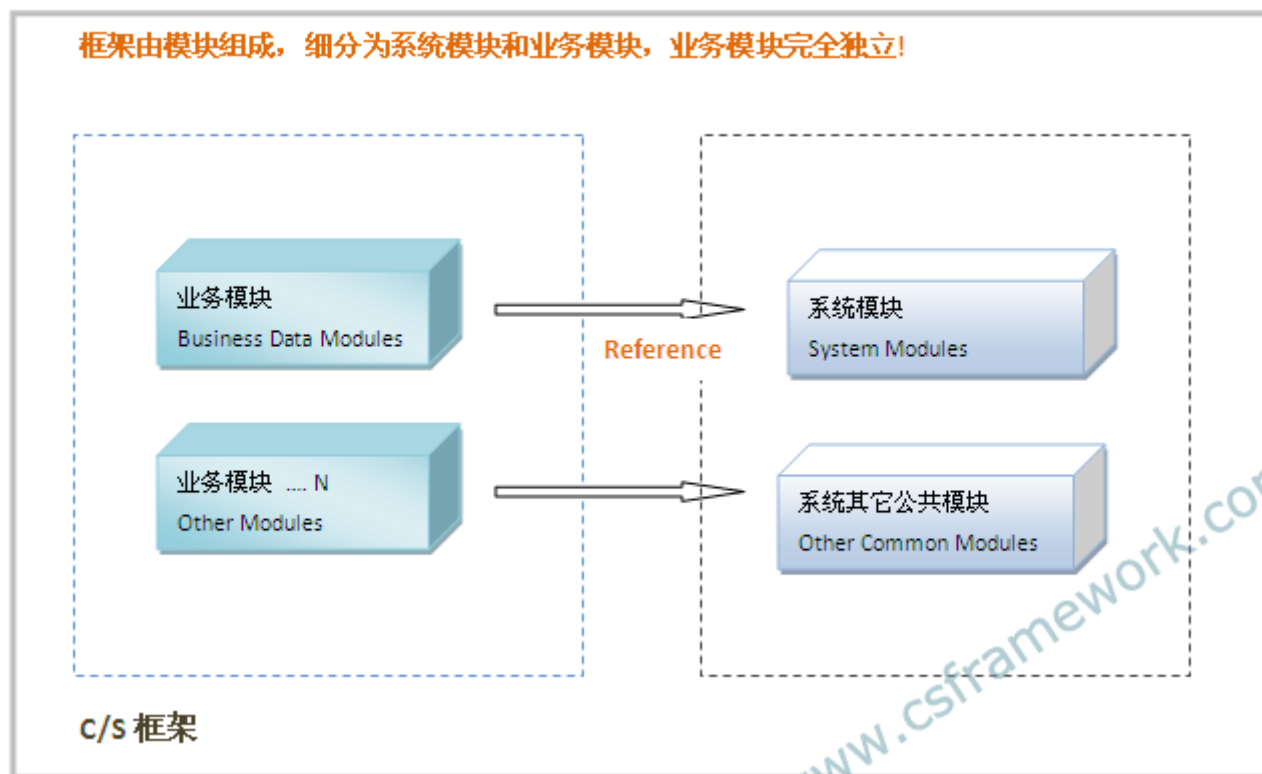
### 2.1 逻辑组织结构图

开发框架由数个模块与系统库构成的, 有系统库、系统模块、数据字典和业务模块。数据字典和业务模块内集成了数个实现了全部功能的开发模板, 作为入门学习资料或者开发时参考用。

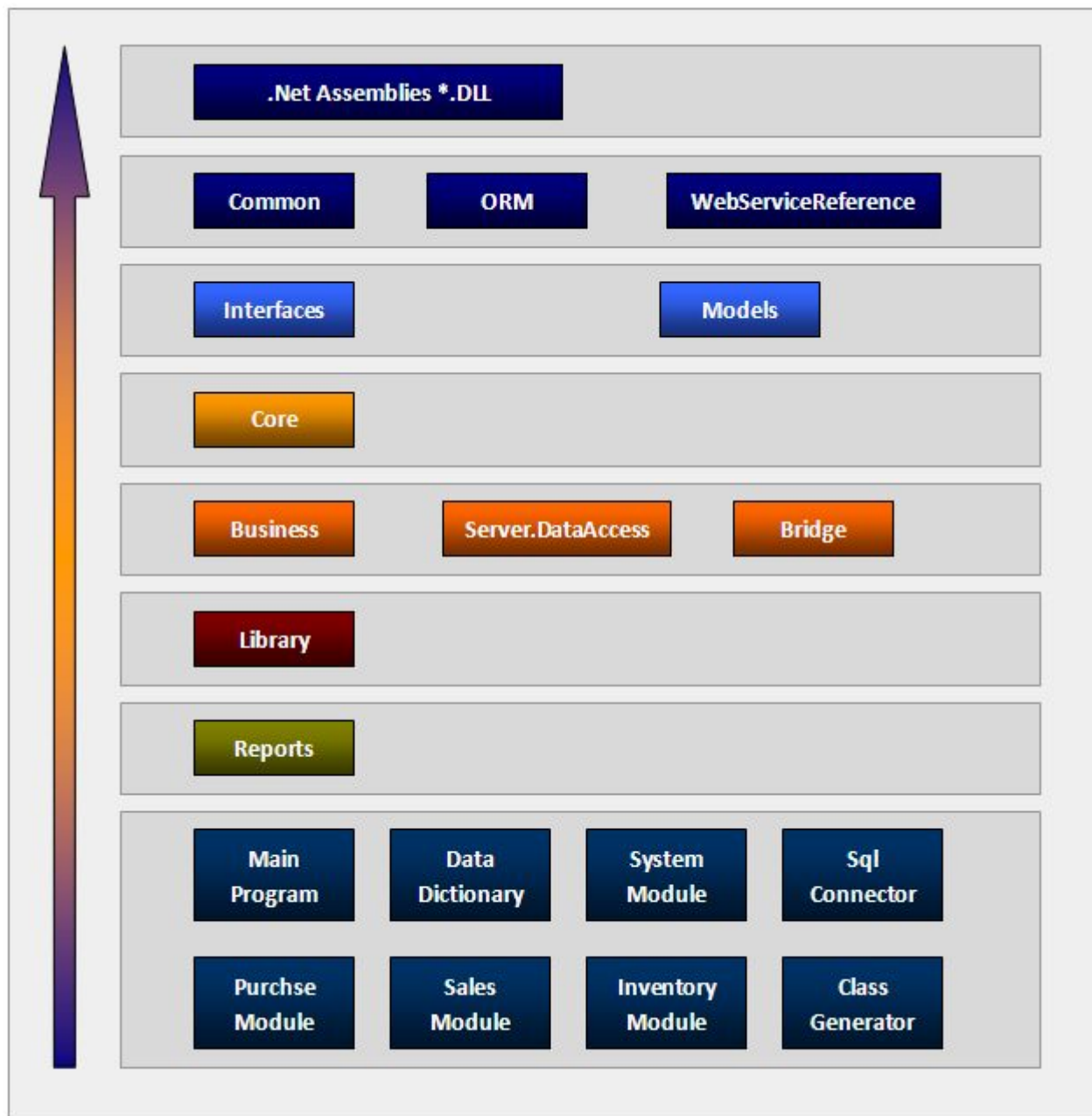


## 2.2 业务模块与系统库引用关系

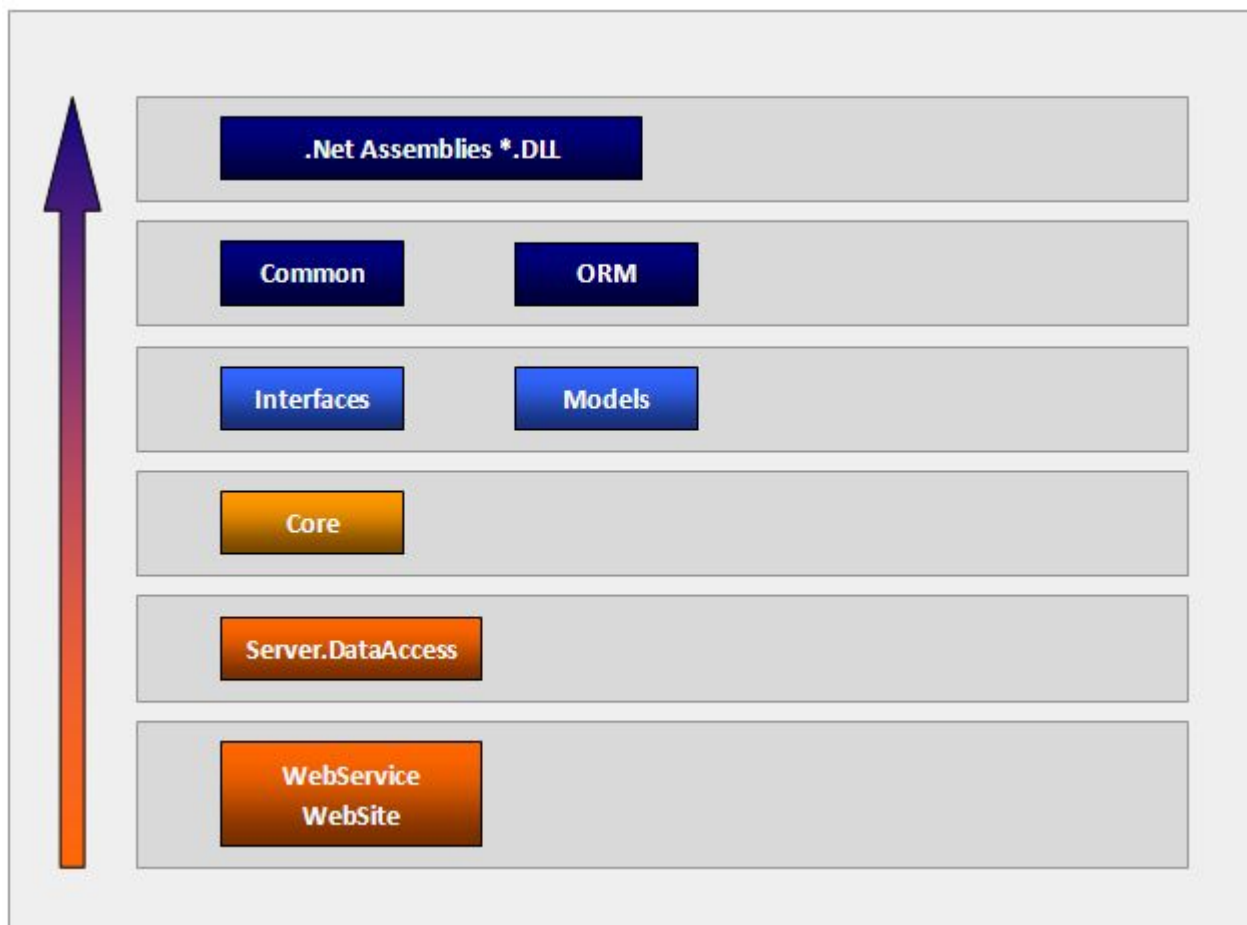
业务模块与系统模块不构成直接引用关系，理论上是完全独立的。按照实际需要系统模块可作为一个较底层的模块提供给其它模块直接引用。系统库（系统公共模块）是开发框架底层的模块，可由其它模块直接引用，但必须遵守引用规则不能双向引用。



## 2.3 客户端模块引用层次(最高层仅引用.Net 程序集)



## 2.4 服务端模块引用层次(最高层仅引用.Net 程序集)



## 2.5 高级版 Client/Server 模块的源码共用表

(此文档仅适合高级版)

高级版是标准的 C/S (Client/Server) 构架, 所以有两个解决方案。因某些公共模块如 Common, Interface, Model 等必须同时给 Client/Server 两个解决方案引用, 从开发角度讲绝对不允许复制两份相同的源码, 所以客户端和服务端必须共用一些模块的源码, 下图列了清单以供参考:

| ID | 目录                               | C/S 源码共用 | 说明              |
|----|----------------------------------|----------|-----------------|
| 1  | CSFramework3.Bridge              | 不共用      | 桥接功能模块          |
| 2  | CSFramework3.Business            | 不共用      | 业务逻辑层模块         |
| 3  | CSFramework3.DataDictionary      | 不共用      | 数据字典模块          |
| 4  | CSFramework3.Interfaces          | 共用       | 系统接口模块          |
| 5  | CSFramework3.InventoryModule     | 不共用      | 库存管理模块 (DEMO)   |
| 6  | CSFramework3.Main                | 不共用      | 主程序             |
| 7  | CSFramework3.Models              | 共用       | ORM, 实体类定义模块    |
| 8  | CSFramework3.PurchaseModule      | 不共用      | 采购模块 (DEMO)     |
| 9  | CSFramework3.Reports             | 不共用      | 报表模块            |
| 10 | CSFramework3.Server.DataAccess   | 共用       | 数据访问层模块         |
| 11 | CSFramework3.SalesModule         | 不共用      | 销售模块 (DEMO)     |
| 12 | CSFramework3.WebServiceReference | 不共用      | WebService 引用模块 |
| 13 | CSFramework.Common               | 共用       | 公共模块            |
| 14 | CSFramework.Core                 | 共用       | 开发框架核心模块        |
| 15 | CSFramework.Library              | 不共用      | 基类库模块           |
| 16 | CSFramework.ORM                  | 共用       | ORM 类库公共模块      |
| 17 | CSFramework.SystemModule         | 不共用      | 系统管理模块          |
| 18 | CSFramework.Tools.ClassGenerator | 不共用      | 类生成器实用工具        |
| 19 | CSFramework.Tools.SqlConnector   | 不共用      | SQL 数据库服务器连接工具  |

## 三. 系统库

公共库是指开发框架内的公共模块, 如 Common, Library, Interfaces 等不包涵用户操作界面的 DLL 库, 是开发框架必须依赖的系统库, 缺一不可。

## 四. 数据字典模块

开发框架定义了一个数据字典模块, CSFramework3.DataDictionary。





### 产品资料

产品/物料数据定义



产品(分类)



### 客户资料

客户资料管理



员工资料



### 公共字典

公共字典数据定义



### 销售员资料

销售员资料定义

## 五. 业务模块

开发框架定义了三个业务模块作为学习参考，分别是 CSFramework3. PurchaseModule、CSFramework3. SalesModule 和 CSFramework3. InventoryModule。

www.csframework.com

C/S框架网版权所有



**销售订单**

销售订单管理。在这里输入特别提示信息。



**销售发票**

销售发票管理。在这里输入特别提示信息。



**销售收款单**

销售收款管理。在这里输入特别提示信息。

- 1. 销售月报表(Demo)
- 2. 销售汇总表(按客户,销售员)
- 3. 销售分析报表

**C/S 框架網** (C#, Asp.Net)  
www.csframework.com

适合企业快速开发  
数据库应用软件

\*\*\* 注: 暂时只实现了销售模块的《销售订单》窗体的功能。 \*\*\*

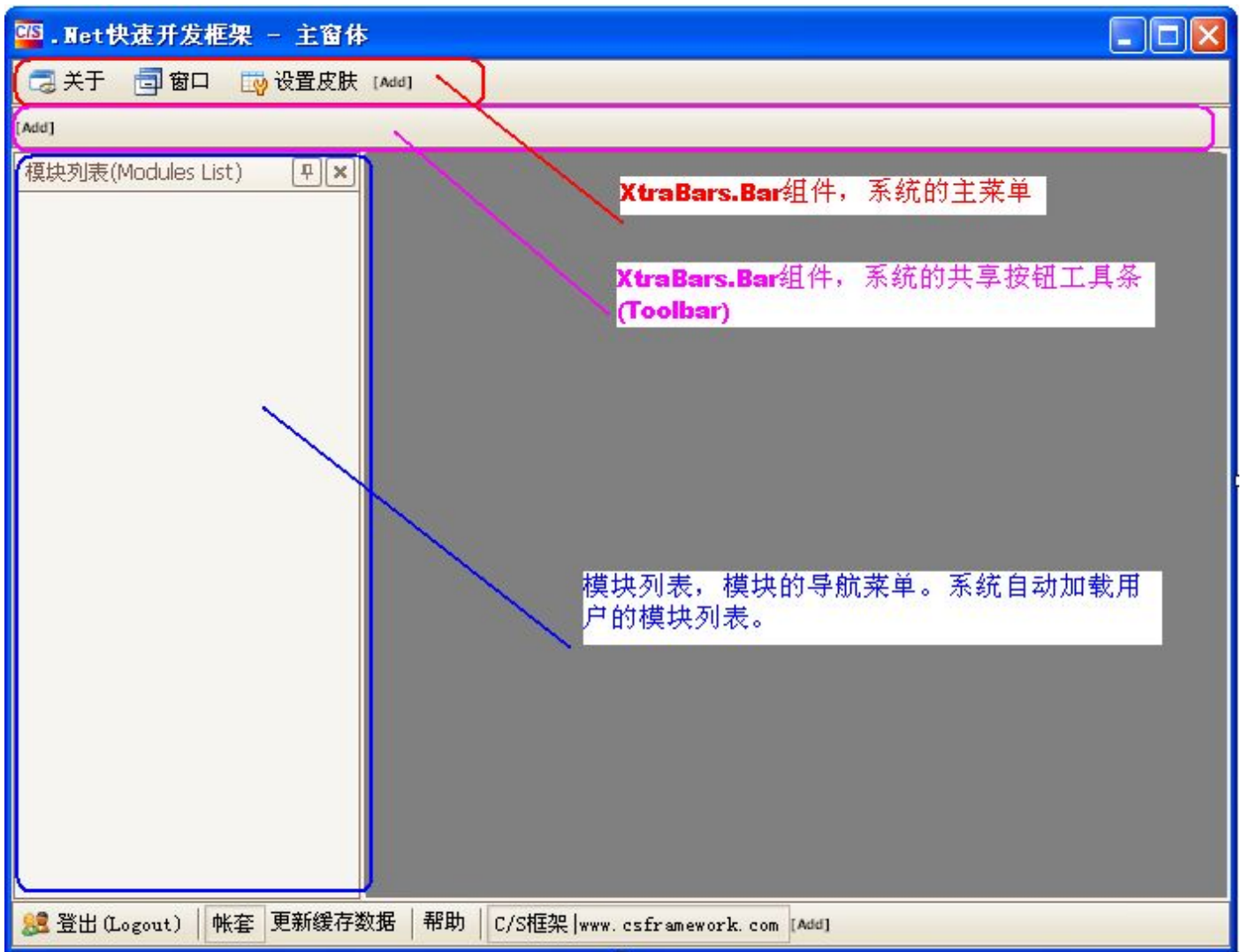
## 六. 系统管理模块



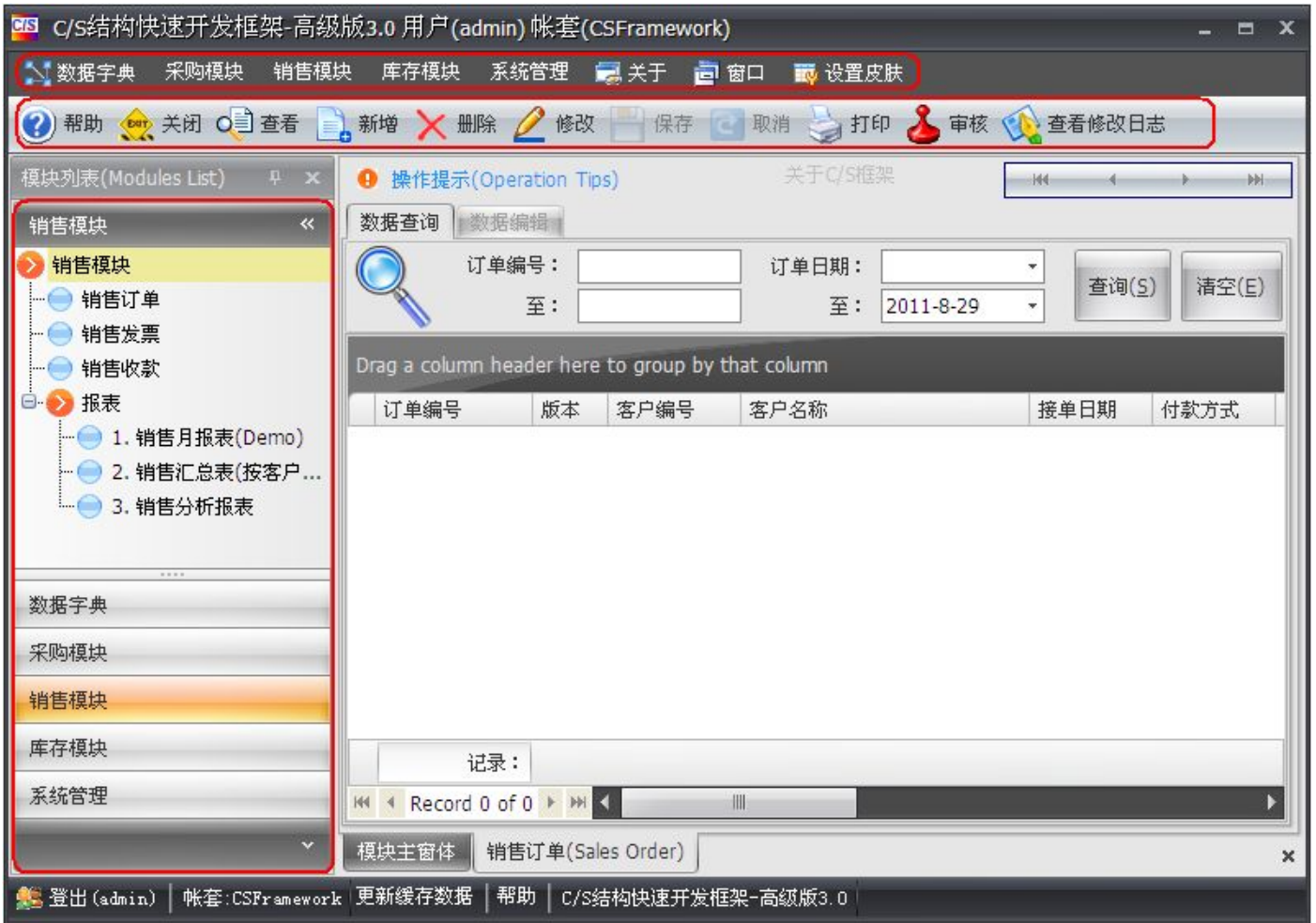
## 七. 系统主窗体与模块集成

### 7.1 系统主窗体 (frmMain)

系统主窗体(frmMain)是一个 MDI 父窗体界面，实际上是各模块的窗器。主窗体只负责集成各模块的主窗体和模块菜单，主窗体的开发界面截图：



运行程序时动态加载模块, 主窗体容器已集成了各模块菜单和入口。

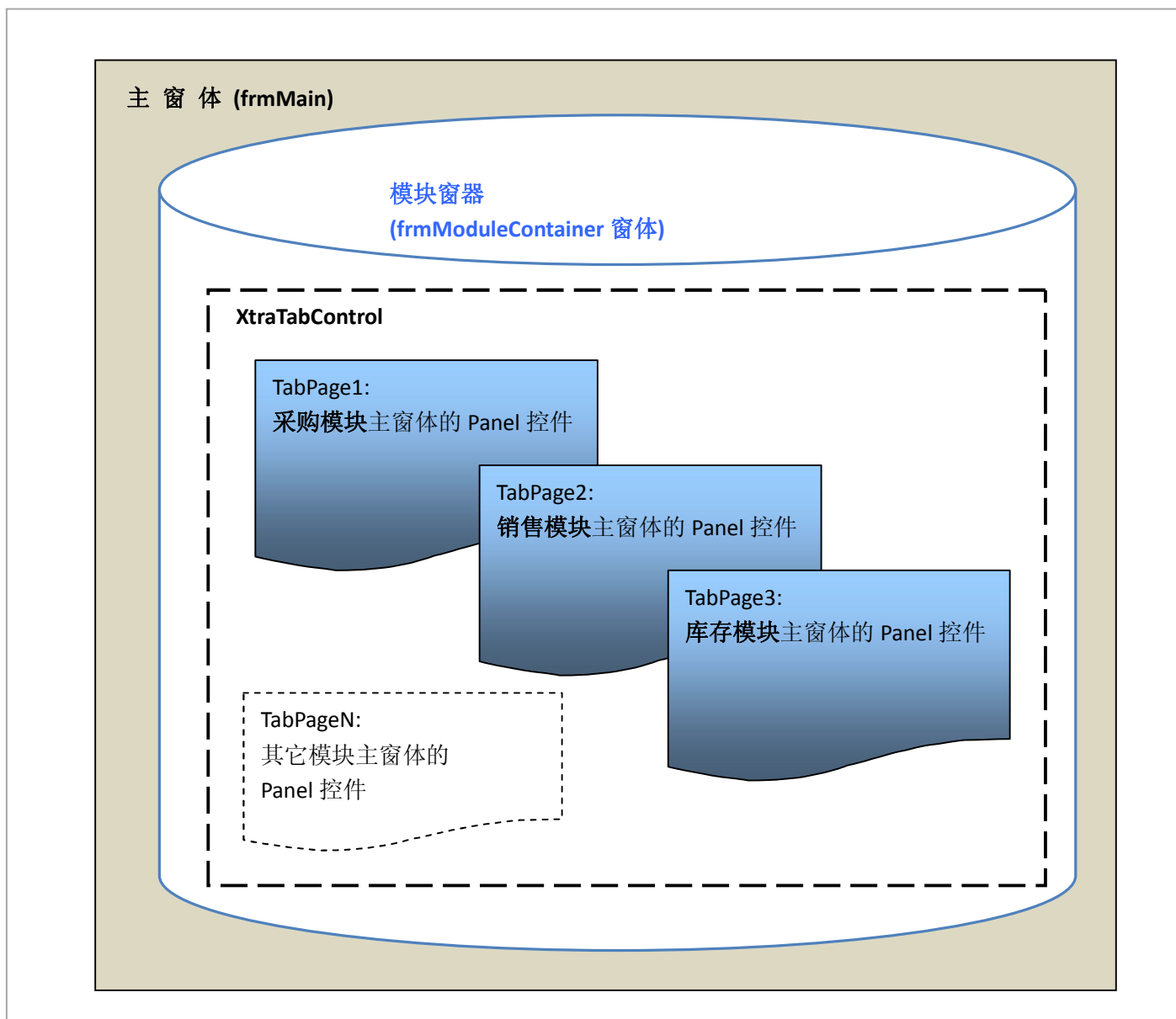






## 7.2 主窗体与业务模块组织关系

主窗体组合一个模块容器窗体 `frmModuleContainer`，在初始化主窗体第一时间实例化并最大化显示。模块容器窗体的主要功能是存放动态加载的模块主窗体上的 `Panel`。模块容器窗体内只有一个 `XtraTabControl` 组件，其每个 `TabPage` 页嵌套一个模块主窗体上的 `Panel`。

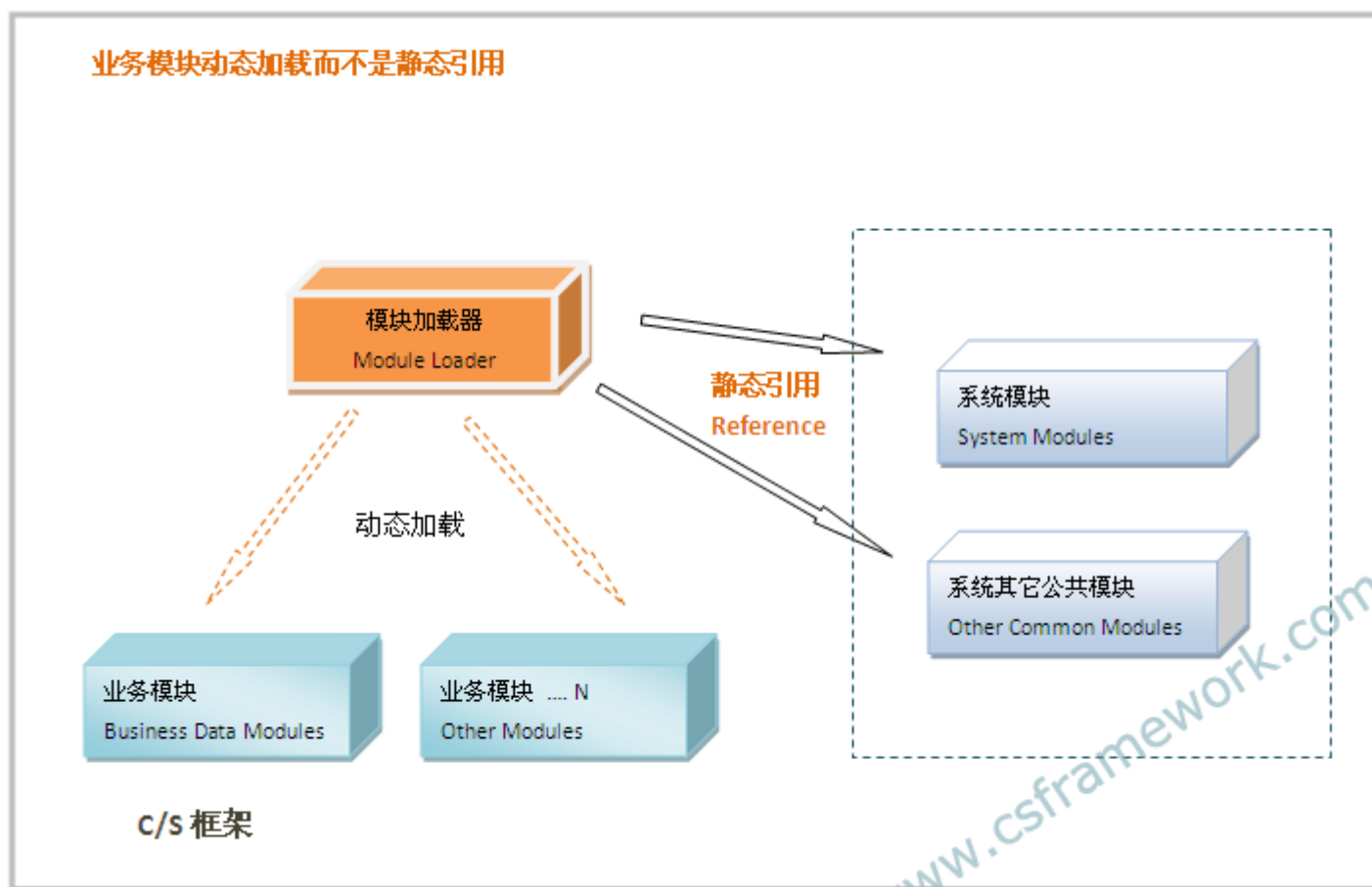


## 7.3 业务模块动态加载

### 7.3.1 简要说明

系统模块和业务模块是动态加载的,即使丢失模块的 DLL 文件框架仍能运行,但不能进行某些操作,如丢失系统模块则不能分配权限和系统设置等,如丢失功能模块,用户无法进行具体的业务操作,如货币定义,客户管理,采购,销售,盘点等。

当运行程序时,系统模块和业务模块会动态加载并集成到程序的主窗体中,登录界面会动态显示模块的加载进度信息。



### 7.3.2 业务模块动态加载部分源码

主窗体定义一个 `frmModuleContainer` 类型的窗体,在初始化主窗体时第一时间实例化并最大化。

```
/// <summary>
/// 系统主窗体(MDI主窗体)
/// </summary>
```



```
public partial class frmMain : frmBase, IMdiForm
{
    //模块主界面容器
    private frmModuleContainer _ModuleContainer;
}
```

动态加载模块时逐一将模块主窗体上的 Panel 嵌套在模块窗体的 TabControl 内。

```
//获取模块主窗体的Panel容器
Control container = _ModuleMainForm.GetContainer();

XtraTabPage page = new XtraTabPage();
page.Tag = _ModuleMainForm; //模块窗体
page.Text = _ModuleMainForm.ModuleName; //取模块名称 // this.GetCurrentModuleName();
page.Controls.Add(container); //嵌套在主窗体的 TabControl 内
xtraTabControl.TabPages.Add(page); //frmModuleContainer.xtraTabControl 组件
```

## 八. 提意见或建议

QQ: 23404761,1980854898

Email: [jonnysun@csframework.com](mailto:jonnysun@csframework.com)

C/S 框架网

[www.csframework.com](http://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!