关系型数据库存储业务各要素信息，在于**[0010][0011][0012]，中，例如：**

小明提交设计信息，服务器完成设计处理（在关系数据库中生成 Teachers 表，表字段有 TNO、

Birthday、……、BelongGrade、BelongSubject，并建立 BelongGrade 和 BelongSubject 的外键关

系，通过动态编译生成了 Teachers 实体类，并在 class 和 field 上标记冗余设计信息，如外键

关系、主键信息等等，把设计信息按表、列、界面控件属性分别插入入对应冗余表，并生成

一个设计文档 bkd 文件）

[J12]设计处理可对提交的设计信息，在

数据库用表保存~~；(作为冗余表，放在第3权利要求)~~

[J02]提交的设

计信息是根据业务需求所设定的不可缺少的信息，包含完整的数据库建模该业务部分所需内容（表、列、范围、约束）和在客户端中该业务部分的使用界面的属性信息；。

相当于“提交的设计信息是根据业务需求所设定的不可缺少的信息，包含完整的在数据库建模该业务部分所需内容（表、列、范围、约束）和在客户端中使用该业务部分的使用界面的所需属性信息”

[J03]

[K12]

界面控件通过录入业务数据并提交到服务器完成该业务相关功能的接口与服务端程序提供的

业务增删改导出等功能对接，服务端程序通过实体类对由接口传入信息确定的业务表记录进

行增删改导出等处理，实现功能。

(在第一种写法中，对“业务各要素信息

”的出处[J03][K12]进行了说明，本领域的技术人员能清楚查到【J12】，也清楚对业务表记录进行增删改处理实现功能，在数据库中相当于“存储”)

服务端程序包括完全自动化...设计处理功能，

附图1

[J01]在如图 1 所示的开发过程中，开发工具是一套表单集，开发工具负责收集业务信息，形成设计信息并提交自动化业务设计管理系统服务端程序进行设计处理；

~~[J11]使该实体类能在运行时环境中运行，方法包括但不限于“动态编译”。~~

**~~[0014]~~** ~~这样，小明仅用了半小时不到的时间就完成了教职工主表的建模开发，系统自动化完成业务的其他部分，得到一个完整功能的程序模块；~~

小明提交设计信息，服务器完成设计处理（在关系数据库中生成 Teachers 表，表字段有 TNO、

Birthday、……、BelongGrade、BelongSubject，并建立 BelongGrade 和 BelongSubject 的外键关

系，通过动态编译生成了 Teachers 实体类，并在 class 和 field 上标记冗余设计信息，如外键

关系、主键信息等等，把设计信息按表、列、界面控件属性分别插入入对应冗余表，并生成

一个设计文档 bkd 文件），处理后，调用消息机制提示“生成成功”，并提供 bkd 文件下载。

在设计提交后的服务端程序中处理，整个处理过程无需人工干预，这种自动化是完全自动化，属于公知常识。

[J02]提交的设

计信息是根据业务需求所设定的不可缺少的信息，包含完整的数据库建模该业务部分所需内容（表、列、范围、约束）和在客户端中该业务部分的使用界面的属性信息。

[J05]以表约束的方式复用

[J07]以多表连接的 sql 语句方式复用

**[0007]** 本发明的有益效果是，[G01]通过在服务器和客户端建立自动化业务设计管理系统，业

务人员能使用设计工具设计根据业务需求建模的数据库模型，设定相关界面属性，并提交进

行设计处理，系统自动生成表或视图

[J03]把该业务

部分通过一个二维表来表示，各业务要素转化为这个二维表的列、范围、约束。

通过实体类自动化管理业务，[K12]

服务端程序通过实体类对由接口传入信息确定的业务表记录进

行增删改导出等处理，实现功能。

业务界面属性管理，[K06]服务端程序提供的获取业务相关界面属性信息的业务功能存在一个已有界面控件的所有界面属性列表，该列表按所属控件功能、所描述的业务表的列范围的数据类型归类，1 个业务列可和列表上的一个或多个属性存在对应关系，该业务功能按调用控件功能分类获取属于该控件的业务的所

有业务列的所有界面属性，按约定的形式传给客户端调用的界面控件。[K07]已有界面控件的 所有界面属性列表存在方式包括但不限于：1 个或多个数据表存放、1 个或多个服务器文件存放、1 个或多个服务端程序变量赋值。

使得业务人员一次设计提交就可。。。建模或维护，**[0007]** 本发明的有益效果是，[G01]通过在服务器和客户端建立自动化业务设计管理系统，业

务人员能使用设计工具设计根据业务需求建模的数据库模型，设定相关界面属性，并提交进

行设计处理，系统自动生成表或视图，生成设计文档，建立相关的实体类，完成该业务的程

序模块，为该业务的使用作好准备，有效地减少对开发人员的依赖，[G02]业务人员使用业务

模块时，客户端程序自动根据使用的业务功能调用相关的界面控件向服务端程序发出请求并

获取该业务数据和相关界面属性信息，完成界面的显示，操作业务模块时，界面控件会提起

处理具体业务记录的请求，服务端程序根据请求完成相应处理，并返回结果，达到控制业务

的效果如同开发人员开发的效果，并有很强的适应性和可复用性，大大提高了开发效率，可

随时维护修改，使开发一个程序模块可以用“天”，甚至以“小时”为单位计算，极大节省

了企业的软件开发成本开销。

**[0003]** [C01]软件开发商如果使用计算机辅助软件工程（CASE）工具可以较大地加快开发进度。

[C02]CASE 工具由许多部分组成,一般按软件开发的不同阶段分为上层 CASE 和下层 CASE 产品。

[C03]上层 CASE 工具自动进行应用的计划、设计和分析,帮助用户定义需求,产生需求说明,并可

完成与应用开发相关的所有计划工作。[C04]下层 CASE 工具自动进行应用系统的编程、测试

和维护工作。[C05]除非下层 CASE 和上层 CASE 工具的供应商提供统一界面,否则用户必须编写

或重新将所有信息从上层 CASE 工具转换到下层 CASE 工具。[C06]CASE 工具主要包括：画图工

具，报告生成工具，数据词典、数据库管理系统和规格说明检查工具，代码生成工具和文档

资料生成工具等。[C07]目前 CASE 的标准是 UML。[C08]这些工具集成在统一的 CASE 环境中，

就可以通过一个公共接口，实现工具之间数据的可传递性，连接系统开发和维护过程中各个

步骤，最后，在统一的软、硬件平台上实现系统的全部开发工作。

[J01]开发工具是一套表单集，开发工具负责收集业务信

息，形成设计信息并提交自动化业务设计管理系统服务端程序进行设计处理。多表单的一般存在分步表单，公知常识

【附图2】（分步产生的部分设计信息）请求到达服务端程序

[K03]界面控件获取自身界面属性信息的接口与服务端程

序提供的获取业务相关界面属性信息的业务功能对接，获取界面属性信息，按照约定的形式，供界面控件生成相应界面。

由于界面控件作为客户端程序是展现，因此分步的表单的设计信息必然只能存放在服务端程序，不可能放在界面控件中

。[K07]已有界面控件的

所有界面属性列表存在方式包括但不限于：1 个或多个数据表存放、1 个或多个服务器文件存

放、1 个或多个服务端程序变量赋值。

[J02]提交的设

计信息是根据业务需求所设定的不可缺少的信息，包含完整的数据库建模该业务部分所需内

容（表、列、范围、约束）和在客户端中该业务部分的使用界面的属性信息。[J03]把该业务

部分通过一个二维表来表示，各业务要素转化为这个二维表的列、范围、约束。

（可以提交成多个表，可以部分提交）

通过界面控件操作管理业务

[G02]业务人员使用业务

模块时，客户端程序自动根据使用的业务功能调用相关的界面控件向服务端程序发出请求并

获取该业务数据和相关界面属性信息，完成界面的显示，操作业务模块时，界面控件会提起

处理具体业务记录的请求，服务端程序根据请求完成相应处理，并返回结果，达到控制业务

的效果如同开发人员开发的效果，

后端复杂业务数据模型的绑定和控制管理

[C05]除非下层 CASE 和上层 CASE 工具的供应商提供统一界面,否则用户必须编写

或重新将所有信息从上层 CASE 工具转换到下层 CASE 工具。[C06]CASE 工具主要包括：画图工

具，报告生成工具，数据词典、数据库管理系统和规格说明检查工具，代码生成工具和文档

资料生成工具等。[C07]目前 CASE 的标准是 UML。[

[C08]这些工具集成在统一的 CASE 环境中，

就可以通过一个公共接口，实现工具之间数据的可传递性，连接系统开发和维护过程中各个

步骤，最后，在统一的软、硬件平台上实现系统的全部开发工作。

[D01]本发明和 CASE 工具设计理念相似。[D02]本发明采用可视化设计工具，消息机制，

设计处理的数据库操作，代码生成，设计文档生成下载等，提供和 CASE 工具一般的高效率和

高可靠性。[D03]但实施方式和 CASE 工具有明显不同：CASE 工具是属于计算机辅助软件工程

技术领域，是面向开发人员服务的，不存在客户界面，单个工具完成的是软件生命周期的一

部分，工具间不一定有关联性，很难形成一个完整性系统；本发明系统主要面向的是事企业

中的业务人员，在实施中充分考虑到易用性和规范性，注重客户界面，业务适应性和复用性

强，针对业务的开发使用维护形成一个独立的整体。

[J03]把该业务

部分通过一个二维表来表示，各业务要素转化为这个二维表的列、范围、约束。

[K06]服务端程序提

供的获取业务相关界面属性信息的业务功能存在一个已有界面控件的所有界面属性列表，该

列表按所属控件功能、所描述的业务表的列范围的数据类型归类，1 个业务列可和列表上的

一个或多个属性存在对应关系，该业务功能按调用控件功能分类获取属于该控件的业务的所

有业务列的所有界面属性，按约定的形式传给客户端调用的界面控件。

[K12] 界面控件通过录入业务数据并提交到服务器完成该业务相关功能的接口与服务端程序提供的

业务增删改导出等功能对接，服务端程序通过实体类对由接口传入信息确定的业务表记录进

行增删改导出等处理，实现功能。

数据库中复杂业务数据的完整性和一致性

小明在客户端设计工具建模设定信息有：表（表名：教师表；

表英文名：Teachers），列和范围（【列名：教师编号，列英文名：TNO，数据类型：字符串，

长度：6，界面控件属性：新增修改控件必填该字段&新增修改控件使用输入控件填入&数据

展示用表格控件&表格控件该列显示单元格长度 150px&表格控件该列排序在前&搜索控件该

字段的选择判断条件为模糊查询】，【列名：出生年月，列英文名：Birthday，数据类型：日

期，界面控件属性：日期控件必填该字段&数据展示用表格控件&表格控件该列显示单元格长

度 100px&表格控件该列排序在前&搜索控件该字段的选择判断条件为等于值】，……诸如姓

名、手机号、地址等等基本信息字段），约束（【因为教师与级组是多对一关系，所有设定

教师表的所属级组字段与级组表的主键为外键约束关系，列名：所属级组，列英文名：

BelongGrade，新增修改搜索界面控件使用单选下拉框，新增修改界面控件该字段必填】；【同

理设定教师表的所属科组字段与科组表的主键为外键约束关系】；【TNO 设定为唯一约束】）。

小明提交设计信息，服务器完成设计处理（在关系数据库中生成 Teachers 表，表字段有 TNO、

Birthday、……、BelongGrade、BelongSubject，并建立 BelongGrade 和 BelongSubject 的外键关

系，通过动态编译生成了 Teachers 实体类，并在 class 和 field 上标记冗余设计信息，如外键

关系、主键信息等等，把设计信息按表、列、界面控件属性分别插入入对应冗余表，并生成一个设计文档 bkd 文件），处理后，调用消息机制提示“生成成功”，并提供 bkd 文件下载。

[N01]小明把生成好的“教师表”放在左侧菜单的基础数据菜单下，形成新一个子菜单“教师

表”。这样点击子菜单“教师表”，就可以开始使用该业务模块。

[G02]业务人员使用业务

模块时，客户端程序自动根据使用的业务功能调用相关的界面控件向服务端程序发出请求并

获取该业务数据和相关界面属性信息，完成界面的显示，操作业务模块时，界面控件会提起

处理具体业务记录的请求，服务端程序根据请求完成相应处理，并返回结果，达到控制业务

的效果如同开发人员开发的效果，并有很强的适应性和可复用性，大大提高了开发效率，可

随时维护修改，使开发一个程序模块可以用“天”，甚至以“小时”为单位计算，极大节省

了企业的软件开发成本开销。

复杂是相对的中文词语，论述复杂必定存在论述者的默认的一条基线，基线之上是复杂，基线之下是简单。在专利文件中，天然存在这种基线，使得发明前的技术和发明后的技术的比对，能够得出结果是否复杂。这是公知常识。

说明书的背景技术[0002]表明了传统的业务开发流程中，业务人员作用仅仅在于提供需求，开发人员在此开发流程中起主导作用。

说明书的[0006][0007]表明了涉案专利让业务人员在创新的开发方法中起到主导的作用，而且达到控制业务的效果如同开发人员开发的效果。

这种用户操作下建立的业务的固有性质是复杂的，可以从上述的对比中直接得出。

从技术层面上看，说明书[0010][0011][0012]表明，对业务人员，不仅需要认知需求，而且还要操作开发工具落实业务的细节，从传统的单纯使用业务变成了对业务的开发使用维护，毫无疑义地这种业务是复杂的。

从技术层面上看，说明书[0003][0004][0010][0011][0012][0013]表明，对开发人员，从传统的依据获取的需求进行针对性的编程开发，变成了要满足不定的需求而预置自动化算法的编程开发，业务的实现代码不仅完全出自开发人员，而且开发人员还要开发设计工具，数据和模型的处理等等方便业务人员的种种手段，毫无疑义地这种业务是复杂的。

 复杂是使用本专利技术进行开发的业务的固有属性。也就是说业务这个软件模块是复杂的

业务可被复用形成高复杂度的业务，

[F01]本发明所要解决的技术问题是，提出一种事企业业务人员能自行开发使用维护满

足自身业务需求的程序模块，高效可复用的自动化业务设计管理系统。

[G02]业务人员使用业务

模块时，客户端程序自动根据使用的业务功能调用相关的界面控件向服务端程序发出请求并

获取该业务数据和相关界面属性信息，完成界面的显示，操作业务模块时，界面控件会提起

处理具体业务记录的请求，服务端程序根据请求完成相应处理，并返回结果，达到控制业务

的效果如同开发人员开发的效果，并有很强的适应性和可复用性，大大提高了开发效率，可

随时维护修改，使开发一个程序模块可以用“天”，甚至以“小时”为单位计算，极大节省

了企业的软件开发成本开销。

业务复用是在原业务的基础上构建新的业务逻辑，这个是公知常识。

[G02]业务人员使用业务

模块时，客户端程序自动根据使用的业务功能调用相关的界面控件向服务端程序发出请求并

获取该业务数据和相关界面属性信息，完成界面的显示，操作业务模块时，界面控件会提起

处理具体业务记录的请求，服务端程序根据请求完成相应处理，并返回结果，达到控制业务

的效果如同开发人员开发的效果，并有很强的适应性和可复用性，大大提高了开发效率，可

随时维护修改，使开发一个程序模块可以用“天”，甚至以“小时”为单位计算，极大节省

了企业的软件开发成本开销。

（表明业务是软件）

复杂度是软件的固有属性，这个是公知常识。

软件复杂度也就是算法复杂度，公知常识。业务是程序。整个技术特征的文本也意味着涉案专利的复杂度属于程序复杂度，也就是算法复杂度，我司给出证据29《数据结构——c语言描述》的1.4.1算法及其性能指标的原文“当一个算法直接使用计算机程序设计语言描述时，该算法便成为程序。”，证明算法和程序的复杂度关系。

业务复用是在原业务的基础上构建新的业务逻辑，因此算法复杂度绝对会变高。

所述不定数的范围是1到数据库能支持的最大值

[J02]提交的设

计信息是根据业务需求所设定的不可缺少的信息，包含完整的数据库建模该业务部分所需内 容（表、列、范围、约束）和在客户端中该业务部分的使用界面的属性信息。[J03]把该业务

部分通过一个二维表来表示，各业务要素转化为这个二维表的列、范围、约束。

（如果建模成1个表，那么该业务部分就是全部，如果建模成多个表，那么各个业务部分组成业务整体。列也是如此。）

。[J05]以表约束的方式复用：通过当前设计的表的某列与复用的

表（已设计处理过生成的数据库的表）的某列设定外键约束关系进行复用，体现业务间的逻

辑关系，由设计工具提供设定步骤，被复用的表可为主键表，也可为外键表。

表的复用引入的新表，表中必定有列，本领域技术人员直接毫无疑义地明白，持续地复用会产生数据库能支持的最大表数值，同时业务要素也能达到数据库支持的的最大列值