

KnowleSys 企业应用模型开发指南

1. 目标

企业应用模型是一个高层规划，该规划的目标是构建一个应用来解决一个特别的企业业务问题。

2. 意义

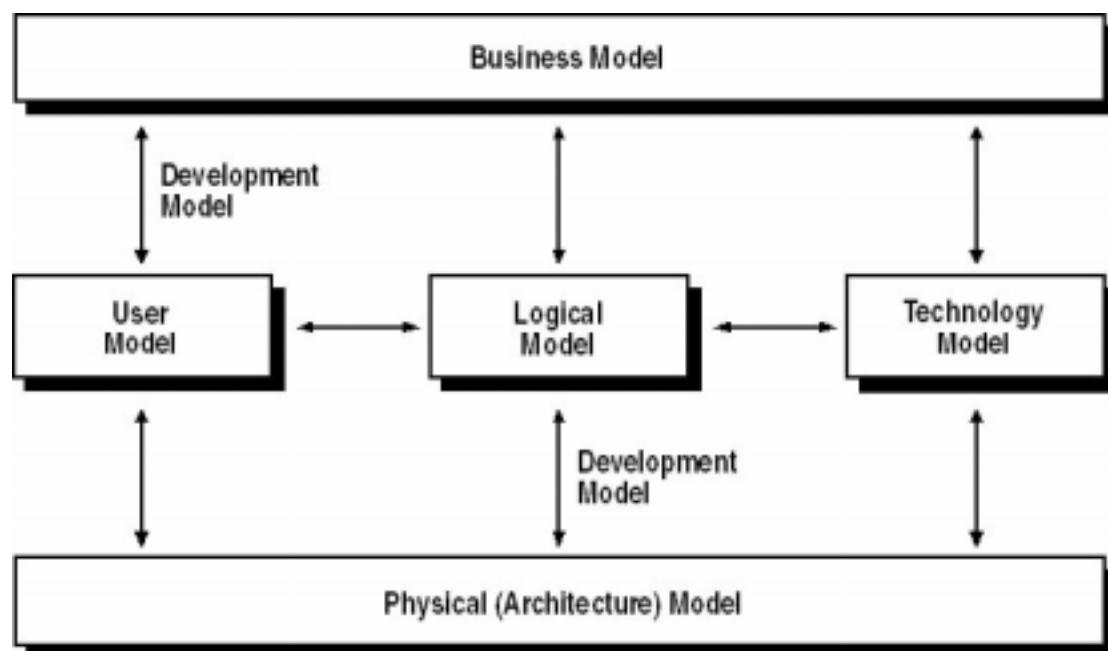
企业应用有三个特点：复杂，面向业务，任务关键。因此一个企业应用的需求将是庞大而复杂的，如果没有一种系统的方法来理解这些复杂而且常常冲突的需求，就很难知道从哪里开始。因此，需要一个直接的方法来减少复杂性，并且通过一种有组织的方式来描绘出众多需求的蓝图，进而设计与创建应用。这就是企业应用模型。

3. 原则

- 与业务目标一致
- 产品观念
- 结构第一
- 基于环境而设计
- 不同的阶段使用不同的语言
- 项目成功要素
- 团队方法
- 每个人都为项目目标做出承诺
- 早期项目演示
- 风险管理
- 基于组件开发
- 变更管理
- 基于客户需求与期望的多产品版本发布
- 灵活的可伸缩的框架

4. 组成

企业应用模型是为实现任何企业应用的所有需求的有序集合，由六个子模型组成：业务模型，用户模型，逻辑模型，技术模型，开发模型与物理模型。其中业务模型是整个模型的入口，规定了需求和目标，通过开发模型的活动设计用户模型，逻辑模型，技术模型，最后通过整个模型的出口物理结构模型来部署应用。企业应用模型提供了对于企业应用的系统化，标准化的分析启示与设计启示。



各子模型的需求如下：

子模型	需求
业务	业务目标 开发成本 投资回报 所需资源 时间限制 安全与维护 现存投资 业务规则与策略 业务流程
用户	用户界面 易用性需求 培训和文档 应用支持 用户桌面配置和网络连接
逻辑	逻辑应用结构 对象和数据模型 业务对象和服务 接口定义
技术	可重用组件开发 开发工具 开发平台 系统和数据库技术 群集，池，消息技术
开发	开发小组 开发过程 项目管理 源代码控制 测试应用里程碑与可交付物
物理	物理应用结构 组件的分布与连接 其他子模型的迭代输入的最终产品

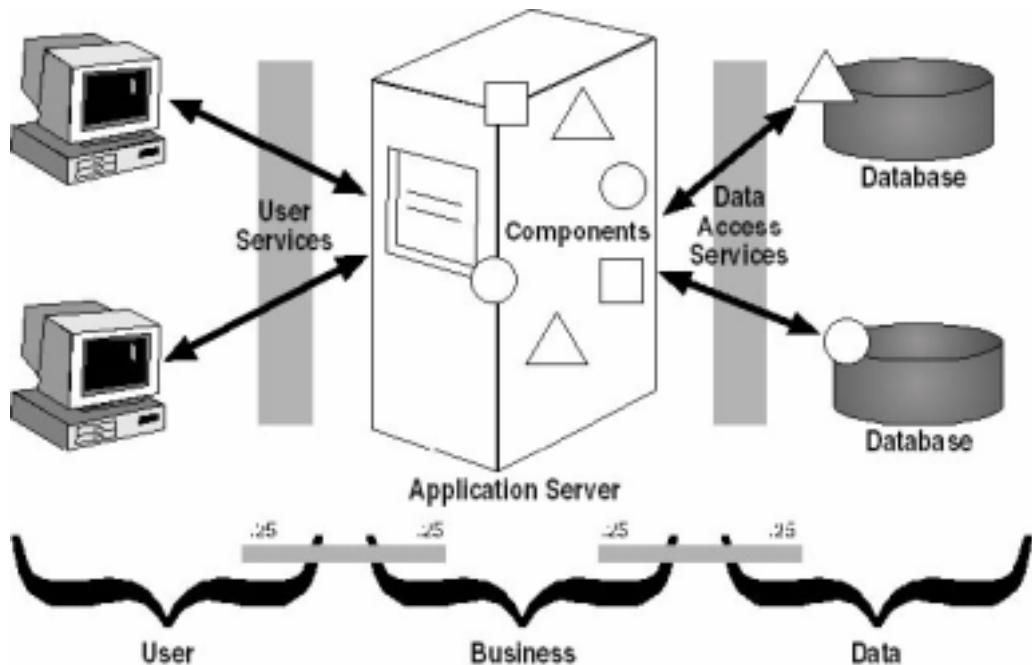
5. 建立企业应用模型

子模型	必须回答的问题
业务模型	<p>这个项目的业务需求是什么？</p> <p>它必须提供什么业务目标和特性？</p> <p>什么层次的投资将获取最佳的回报？</p> <p>它必须以什么样的速度交付？</p> <p>它的部署成本有多贵？</p> <p>它必须支持什么平台？</p> <p>并发存取用户有多少？</p> <p>安全性有多重要？</p> <p>它必须有多可靠？</p> <p>什么时候这个发布版本将被替代或升级？</p> <p>新的业务策略或用户需求需要被新的升级程序实现的速度怎么样？</p>
用户模型	<p>谁是用户？他们的技能，经验与使用用例或使用场景情况如何？</p> <p>用户对于任务序列，可用性，培训，可响应性，性能以及与外部应用与数据的互操作性的需求是怎么样的？</p> <p>用户能够容忍系统的异常使用方式与不方便吗？</p> <p>客户是否会因为用户界面不够简单，可靠与有效而不购买产品？</p> <p>用户需要多少特性文档与帮助？他们有耐心读文档吗？</p> <p>用户需要产品支持吗？他们愿意为此而付钱吗？</p> <p>在任一时刻有多少用户在使用该应用？</p> <p>用户的计算机性能怎么样？网络连接速度怎么样？</p> <p>需要什么来保证适当的安全性？</p>
逻辑模型	<p>业务实体有哪些？</p> <p>业务规则有哪些？</p> <p>何时一个客户将获得一个购买折扣？</p> <p>何时需要重新进货？</p> <p>州外订单的税率是多少？</p>
技术模型	<p>桌面，服务器与数据库各需要什么样的操作系统特性？</p> <p>需要支持什么网络协议？</p> <p>将使用什么安全技术？</p> <p>如何使用技术来支持高可扩展性？</p> <p>对于跨组件与跨数据库的事务，如何维护数据库的完整性？</p> <p>如何处理大量异步的请求？</p> <p>使用什么技术来存取历史遗留系统？</p> <p>UI 窗体如何展示？</p> <p>使用什么技术来存取数据和远程数据库？</p> <p>使用什么技术来复制桌面数据或跨数据库服务器数据？</p> <p>考虑到业务的需求和可用技术的能力，产品将被如何构建？</p> <p>使用什么工具来帮助创建，调试，集成与部署应用？</p>

开发模型	<p>谁将负责整个开发问题的哪个部分？ 子模型的任务（如逻辑对象设计，可用性测试，数据库设计，编程与测试）以什么样的顺序进行？这些过程如何被量化？ 有多少开发人员投入到这个项目中，他们的技能如何？ 什么样的特别活动是需要被安排的？以优化开发效率和最终交付的质量？ 项目该如何管理以及如何与其它项目协调一致？ 软件重用和组件化的重要性如何？（这个问题的回答影响到协调，标准的重要性，设计所花费的时间，以及目标通用化结构的价值） 过程中的工作如何共享，以及如何促使其他开发人员与测试人员使用？ 行政部门如何被管理？ 如何部署更新升级？</p>
物理模型	<p>何被使用所有的物理资源如计算机，网络带宽与协议，数据库，组件，操作系统，后端办公服务，第三方特性等来满足业务的整体需求（如可扩展性和健壮性）？ 如何把现存的部署移植到需要的部署上去？ 如何最有效率地存取资源（例如：本地计算机与因特网）？ 在运行环境中如何满足可用性和吞吐量性能的需求？包括局域网和广域网的低速存取时间，短暂的网络服务器连接断开，以及因特网服务器无法使用。</p>

5. 规划企业应用结构

应用结构是应用实现后的物理结构的一个概念视图。每个应用都有三个不同的层次：用户层，业务层，数据层。每个应用都包括展示代码，业务规则处理代码，数据处理代码，数据存储代码。应用结构展示了这些代码如何被打包而组成一个产品的。多层次应用结构的物理视图：



多层次应用结构各层次的功能与可采用的技术

层	子层	功能与技术
用户层	用户交互界面	为用户展示信息（功能，数据） 接受用户信息（指令，数据） Presentation : WindowsForm , Web ; Actions
	用户服务	为应用的界面提供用户逻辑 努力把信息视图从用户界面中隐藏与隔离开来 ActionObject , TaskProxyObject
业务层	业务服务	控制操作序列 强制实现业务规则与数据规则 规则的集中点，依赖于下层，基于组件 供所有应用存取 TaskObject , BusinessObject , COM+
数据层	数据存取服务	提供操作数据的最低可见性的细节 实际的数据存取操作：定义，创建，读取，更新，删除 ADO , DBExpress , ODBC , BDE
	数据存储	持久存放数据 关系数据库：Oracle , DB2 , MSSQL , InterBase , MySQL 文件数据库：Access , Paradox , XML , DataFile 内存数据库 存储系统：MS Exchange Server , Lotus Notes

多层应用结构的逻辑视图：

