**摘要:** 本文以我主持开发的某公司生产经营管理系统为例，探讨了基于构件的软件开发问题。该系统是一个集原料采购、生产管理、物流管控等七大功能于一体的综合信息系统，在该系统的开发过程中，我担任系统架构师角色，主要负责需求分析、系统建模和方案设计三个方面的工作。 本文首先简要分析了CORBA、EJB、COM/DCOM三种构件技术的特点，然后着重论述了采用构件技术进行软件开发的过程。在构件的获取阶段，我们采用了三种构件获取方式来解决用户提出的3类不同的需求；在构件的开发阶段，我们统一采用一个查询构件进行了封装，以实现将同一功能的不同表现封装到一个独立的构件中；在构件组装阶段，采用了3种构件组装方式完成了构件的组装。最终项目顺利上线并运行稳定，获得用户一致好评。

2013年5月，我所在公司承接了某大型粮食加工企业某公司生产经营综合管理系统的开发工作。该系统是某公司在国家粮食案例政策的指导下，结合自身经营管理需求提出建设的。其目的是对内加强管理，对外提升服务，以实现提升品牌形象、保护消费者利益的战略目标。系统整体上分为两个部分，一是经营管理WEB平台，二是手机APP应用。系统采用了基于服务的层次架构，共分为三层。其中用户界面层使用Extjs、senchatouch和phonegap框架实现，业务服务层使用.Net平台实现，数据层使用IBM DB2 V9.5。该系统于2013年5月开始建设，2014年6月上线运行，历时一年。在系统的开发过程中，我担任系统架构师角色，负责需求的获取和分析、系统建模和总体方案设计工作。在系统的实现过程中，我和我的团队使用了基于构件的软件开发技术，收到了很好的效果。  
基于构件的软件开发是在面向对象技术的基础上发展起来的，它是软件危机问题日益突出形势下的产物，它有效地解决了软件系统复杂度、成本 、质量、效率等难以控制的问题，受到业界的广泛推崇。当前主流的软件构件技术有三大流派，分别是OMG的CORBA、SUN的EJB和微软的COM技术。上述三种技术中，CORBA实现的是最为漂亮的，它分为对象请示代理、公共对象服务和公共设施三个层次，其特点是大而全，互操作性和开放性特别好，其缺点是庞大且复杂，相关技术和标准更新缓慢。相比之下，EJB的发展要更加迅速，它是在JAVA语言基础上建立的服务器端组件模型，具有优秀的跨平台性。EJB框架提供了远程访问、安全、持久化和生命周期等多种支持分布式计算的服务，目前JAVA和CORBA有融合的趋势。最后是COM技术，它是微软公司的独家产品，基于windows平台，功能强大、效率很高，且有一系列相应的开发工具支持，其最大的弱点是跨平台性较差，很多人认为CORBA将比COM技术走地更远。基于构件的软件开发技术能有效地简化设计、提高效率、保证质量，某公司生产经营管理系统的开发项目时间紧、任务重，我们使用基于构件的软件开发技术解决了软件开发中的各种问题，取得了很好的效果。  
具体在实践过程中，我们开展了模块划分、构件标识、构件获取、构件组装与测试、构件管理等活动，这里重点谈一下“构件获取”、“构件开发”和“构件组装”三个方面的问题。  
一、构件获取  
在本系统的开发过程中，用户提出的需求从实现方式上可分为三类。第一类需要修改当前的系统来实现，比如用户提出，需要将某公司当前在用的两套生产控制系统的支行状态集成到目标系统中；第二类是直接使用我公司现成的构件来实现。比如用户登录、角色权限管理、日志记录、数据库访问等基本功能；第三类需要集成第三方的服务来实现，比如用户要求APP软件需要具有消息推送、GPS定位功能，可通过集成第三方运营商的产品来实现，无需另行开发。针对上述三种类型的需求，我们采用了三种构件获取方式。首先是通过改造现有系统获得构件，我们与生产控制系统的开发商联系，请他们使用COM技术对软件的工作状态进行封装，并开放接口。其次是使用构件库中现成的构件，我公司在长期的软件项目建设过程中，积累了大量可重用的软件构件，如文件序列化、数据库连接等模块，这些构件可直接在新项目中使用。最后是集成第三方的构件，针对用户提出的消息推送和GPS定位功能，我们经过对比和测试，选择了个百度的产品进行集成。此外我们还根据需求，重新开发了一些功能构件，以支撑用户的需求。  
二、构件开发  
构件的优势体现在其粗粒度的重用性，因此在构件的设计过程中，应尽可能将同一功能的不同表现封装到一个独立的构件中，以保持其高内聚、低耦合的特性，本系统的构件设计就很好地遵循了这一原则。以数据查询构件为例，整个系统中用户需要查询的数据多种多样，对分页显示的要求不尽相同，数据返回格式也不完全一致，可能是XML，也可能是JSON，针对所有这些查询需求，我们统一用一个查询构件进行了封装，开发人员只需要构造好SQL语句，再配合一些特定的参数，就能得到自己想要的结果，这样最大限度地保证了构件的可重用性和重用粒度。为了实现这一目标，有时还需要用到一些经典的设计模式。比如我们使用的数据库连接构件，就用到了抽象工作方法，构件可根据配置文件的内容，在运行过程中建立对不同数据库、不同方式的连接，很好地体现了构件的优势。  
三、构件的组装  
不同的构件类型，需要采用不同的组装方式。在本系统的开发过程中，我们用到了以下三种方式。首先是DCOM构件的组装，采用了远程调用方式，该方式主要用于对生产控制系统构件的组装。其实现比较复杂，需要在控制系统上部署DCOM服务，还得在WEB平台的IIS上配置相应的权限。其次是构件库中构件的组装，该方式比较简单，直接引用构件库中的产品文件，可能是DLL格式，或者是C＃的源码文件。最后是SOA服务调用组装，主要针对第三方的软件服务，其实现也很简单，使用基于http的webservice访问即可。需要注意的是在这种组装方式中，由于数据需要经过第三方的软件平台进行传输，需要采用一定的数据加密措施，以提高系统的安全性。  
由于使用了基于构件的软件开发方式，某公司生产经营管理系统的开发工作进行地十分顺利，系统按期上线，并得到了用户的肯定。该项目的成功让我认识到好的软件设计思想和技术在软件开发过程中的作用和价值，坚定了我对基于构件的软件开发技术的信心。从软件行业的发展来看，从汇编语言、基于过程的软件开发、面向对象的软件开发，直到现在基于构件、面向服务的软件开发，我们可以认识到软件元素在两个维度上的进化趋势，即内部功能越来越强大、全面，对外的接口却越来越简单、标准，终有一日各领域的软件必将能在一个统一的标准下进行无缝的组装，届时面向协作的软件开发、基于职能的软件开发等新技术都可能会出现，上层应用功能的实现也将变地异常简单，计算机软件可能会变地无所不在，数字化生活、智能地球等等现在还比较超前的概念，将在那里得以实现。我想这一天值得每一位软件从业人员为之努力、期待。