摘要:

2017年7月，我参与某互联公司自主研发全国运营的网约车出行平台。该平台主要是为了解决客户在线预约网约车。包含用户叫车、自动派单、司机接单、行程结算、车辆管理等功能模块。我在该项目中担任系统架构师设计师一职，主要负责整个系统的架构设计。本文以该项目为例，主要论述基于构件的软件开发过程。通过xx方法来识别构件达xx目的；通过xx方法来定义构件达xx目的；通过xx方法来复用构件达xx目的；。事实证明，使用这些技术手段使得项目整体能够克服项目中遇到的各种问题。最终项目得以顺利完成，取得预期目标，获得用户好评。

正文:

2017年7月，我参与某互联公司自主研发全国运营的网约车出行平台。我公司致力于网络出行市场服务是首批获得当地“网约车出行牌照”, 并且先后获得四川、河南、海南、山东等各地的合法运营牌照背景下。以构建一个合法、合规安全的网约车出行环境为使命。以解决用户打车难、提升用户的出行幸福感，拉动了产业链的发展，构建了一个完整的出行生态链的为目标，构建一个全国性的网约车出行平台。 网约车出行平台包括乘客端、司机端、后台管理系统三部分组成。乘客端供乘客查询车辆、发布订单、支付车费、评论司机；司机端供司机车辆信息认证、出车接单、乘客接送、车费提现等；管理系统主要是提供系统报表查询、规则配置、乘客管理、司机管理、分公司管理、账务管理等。本平台提供了专车、快车、出租车业务，乘客可以通过自身的需求来选择不同类型业务出行，司机需要上传自身拥有的运营车辆等证照信息到平台审核，只有当平台审核通过后才能正常的再平台上合法运营。我在该项目中担任系统架构师设计师一职，主要负责整个系统的架构设计。

基于构件的开发技术是在面向对象的基础上发展起来的， 能够有效的降低系统的复杂性，缩短项目的开发周期。现在主流的构建有3中种微软提出的COM、SUN公司提出的EJB和OMG提出的Cobra。COM 是微软的一套软件组件的二进制接口标准。支持跨编程语言的的进程通讯、动态对象创建等。是多项微软技术与框架的基础。EJB 是用来构筑企业级应用的服务端组件。CORBA是OMG组织订制的一种标准的面向对象的程序体系规范，或者说CORBA体系是对象管理组织解决分布式环境中，硬件和软件系统互联二题出的一种解决方案。它的职责是为应用开发提供一个公共框架， 定制工业指南和对象管理规范，加快对象技术的发展。结合项目实际，我们历经了创建构件仓库、创建构件、提取构件、标识构件、构件修改和构件组装等活动。下面就运用构件技术进行软件开发过程中的提取构件、构建修改、做具体的论述。

1、创建构件

创建构件有三种方法，第一种公共构建直接将他们上传到构件仓库中如：Spring、HIbernate等，第二种其他公司或者服务提供的构件，然后将他集成到系统中封装成一个独立的构件例如：接入某平台的人脸识别接口然后封装为独立的构件。第三种项目所特有的流程算法等将他们封装为一个独立的构件然后上传的构建仓库中，如：我们自己定义的工具类，通用函数库等。构件的不断迭代支持的能力逐步提升，开发效率提高降低了复杂度，在开发的过程中如果我们的构建不能满足需求我们就需要修改构件或者产生新的构件

2、构件修改

在理想状态下构件是能够直接复用构件库中已有的构建，但是大多数情况下，必须对构件进行修改才能适应新需求（修改构件的目的）。为了减少构件修改的工作量，要求开发人员尽量使用构建的功能，行为和接口合计更为抽象化，通用化和参数化。这样，服用者即可以通过对已实参的选取来调整构件的功能或行为，如果这种调整依然不能拿满足性的系统，复用者必须借助设计信息和文档来修构件。构件库中若无修改使用的构建，则需要按照新需求开发构建并存入构件库中 （举例1个修改构件的例子）为了方便记录日志，我们增加了日志组件，能够记录操作日志。

3、构件组装

功能的组装，过程中容易引起版本的冲突。

在项目中主要用到三种构建组装技术，功能组装、数据组装、面向对象的组装。基于功能的组装技术，基于功能的组装技术采用子程序调用和参数传递的方式将构件组装起来，它要求库中的构建以子程序/函数/过程的形式出现过，并且接口说明必须清晰。当使用这种组装技术开发时，开发人员首先要对新的功能进行分解，将系统分解为高内聚、低耦合的功能模块。然后对应提取构件，进行适应性修改，在挂接在上述功能分解框架中。基于数据的组装技术手相根据当前软件的核心数据设计出一个框架，然后根据框架中各个节点的需求提取构件并进行适应性修改，再讲构件逐个分配到框架中的适当位置。此后构件的组装方式依然是传动的子程序调用与参数传递的方式，这种组装技术也要求库中以子程序形式出现，但它所以来的软件设计方法不再是功能分解，而是面向数据的设计方法，例如，Jackon 系统的开发。面向对象的组装技术。由于分装和基础层的特征，面向对象方法比其他方法更加适合支持软件复用，在面向对象的开发中，如果从类库中检索出来的基类能够完全满足性的系统的需求则可以直接应用，否则必须是基类为父类，生成相应的之类，以满足新系统的需求。

经过全体成员的不懈努力。在2018年2月，先后在四川、重庆、河南、贵州、海南等城市全国开展内测。2018年6月，全国正式发布运营。上线1年多程序一直稳定可靠运行。无较大线上生产事故、系统平稳运行。（省略150字）。上线以来得到了多个地区交通部门的点名表扬和和上万用户的好评。也为我们后续的开发、迭代、运维奠定了一个良好的基础。

项目上线至今运行1年多进入产品优化迭代阶段一直运行稳定运行，无较大生产事故。基于构件软件架构能能够以架构组装为蓝图，可以复用软件架构为组装的模块，支持组装式的复用，大大的提高了生产效率和软件质量。展望：在系统的迭代过程中由于一些构建的组装会导致构件的依赖冲突或者构件的接口过于冗余，这个使我们的一个权衡点，对于构建的组装和修改的选择我们需要权衡让所有的构件都是一个美、小、巧的软件作品。