分离业务逻辑服务层

商业公司的业务同质化很高，市场如战场，谁能快一步应变，谁能给客户提供个性化，谁就得到了业务，谁就能生存。特别是象中国这样各地的经济、文化、政治极其不平均的国家，中央与地方的差异鸿沟巨大，地方特色必然需要。

但是站在总公司的管理角度上来考虑，当然是希望业务流程越规范越好，新花样总是意味着管理上的潜在危险。而对于总部信息技术部门的角度来看，个性化的新花样则是开发工作量的剧增、无止无尽的新需求。

管理与市场、领导与客户、全局与局部、总公司与分公司之间，这个思路方向性的矛盾是现实存在，而且不可避免的。

当然，最后项目还是要按上级的管理意图来实施，于是我们得到了一个全国一致的系统，一个唯一可用的UI，一本统一印发的操作手册。对于常规的业务，按着系统的要求操作就可以满足需要了。但是这个系统不再给机会进行业务、技术的创新，一切新的想法只能做为新需求向上级提出，然后由情绪恶劣的IT人员在程序里加入一个个 “if(strncmp(deptno, “BEJ”, 3) == 0)…” 这样丑陋到暴的代码来完成流程定制化。

其实要在同一个系统里同时满足总分公司双方的诉求，并不是不可能，就是把应用系统分为两层：业务逻辑与UI层，业务逻辑层是对业务逻辑的原子化，以实时服务的形式提供。在此业务逻辑服务的基础上，构建界面展现。

这里的关键点在于业务逻辑服务的提供，不仅可以是标准UI对其的进程内调用，也同时需要能够通过webserivce等协议提供进程外服务。对于标准流程，可以由总部来做“典型实现”，而对于有特别需要的分部，则可以方便在业务服务的基础构建业务系统的其他前端外延。

这样，从管理的角度上来讲，业务数据的进出都是通过标准服务来进行的，业务数据质量、业务一致性、合规性都可以通过统一的业务逻辑来得到保证，而分支机构则有机会为不同的业务开发不同的前端接口，以根据市场要求灵活创新。

这种模式特别适合于有外部合作机构的公司采用，如金融企业、电子商务企业等。如果业务系统只有一个UI的IO方式，一切数据进出都要通过操作员来进行，那么与其他合作方进行自动化的数据对接就很难实现，一但有这样的需求，只能由交由掌握了底层逻辑的总公司IT从头开发。

现在我们的系统已经上线，可惜的是，系统采用了最简单最直觉的思路开发，几乎要把业务逻辑写在界面里了。事已至此，也不可能有什么改变了，这里只是我有过的想法分享出来，作为讨论。