

第3章

系统规划



本章重点

- (1) 如何将用户的需求具体化、结构化。(★★★★★)
- (2) 如何识别超出项目范围的需求。(★★★)
- (3) 如何识别错误的需求。(★★)
- (4) 需求调研报告的编写方式。(★★★★★)
- (5) 如何绘制业务流程图。(★★)
- (6) 认清软件的价值。(★★★)
- (7) 如何规划软件边界。(★★★)
- (8) 如何规划工作方式。(★★★★★)
- (9) 让用户重复劳动的产生原因。(★)
- (10) 信息孤岛形成的原因,常用处理方式。(★★★★★)



本章内容思维导图



系统规划是根据用户需求规划企业信息化管理体系的过程,主要工作包括:厘清用户需求,对用户的需求进行系统分析,规划未来如何通过信息系统进行企业管理,确定需要哪些软件功能,需要处理哪些数据等。这里叫“系统规划”而不是“软件规划”,是因为这个规划是对相关业务的信息化管理体系的规划,软件只是这个体系的载体。

虽然本书将系统规划与获取需求割裂开来分成两章,其实这两个阶段是紧密交织在一起的,一边进行需求调研一边进行规划分析显然是比较合理的工作方式。当然,由于需求获取阶段往往是针对不同岗位的人分开进行的,而系统规划需要一个综合思考的过程,所以将其作为两个阶段讲述也是有一定道理的。

在系统规划阶段,需要注意的是,不能闭门造车,尽可能跟用户多沟通是系统规划成功的重要保证。在这个阶段会确定软件的走向,决定以后用户如何通过这个软件工作,决定对现在的业务流程需要做什么变更重组,决定这个项目的范围,决定哪些需求可以实现、哪些需求不可以实现,决定整个系统需要处理的信息,决定整个系统需要提供的功能。这一切都与用户的利益息息相关,如果没有用户参与讨论,那么所有的工作成果只能算一厢情愿,虽然不见得是死路一条,但也危机重重。



3.1 需求确定

经过大量的需求调研工作之后,手头上可能有客户提出的大量的、千奇百怪的软件需求。这些需求,有些是技术上可以实现的,有些是技术上不可以实现的;有些是管理上需要的,有些是管理上不需要的;有些是合理的,有些是不合理的;有些是一致的,有些是矛盾的;有些是在信息系统蓝图之内的,有些是在信息系统蓝图之外的;有些必需的需求却没有人提出……

作为需求分析者,如何处理这些需求呢?要牢记一点,“实现用户正确的需求”是软件设计的工作原则。这里“正确”两个字非常重要,用户所提出的需求未必都是正确的,要有一个严格的分析、甄别过程,如果用户要什么就开发什么,还需要需求分析者干什么呢?需求分析人员不能不分析需求,而只做需求的记录员。

3.1.1 认清需求

为了认清用户的需求,先要认清用户。在进行需求调研的时候,会跟企业中各种各样的人员沟通,这些人基本上都是软件未来的用户,他们的技术、知识、性格、职位、工作内容各不相同,但他们一般都有一些非常相似的地方:他们不是做软件的,他们不是搞需求分析的,他们永远不会像你希望的那样去描述需求,他们的需求是用自然语言描述的,是抽象的、概略的、随性的。将这些抽象、概略、随性的用户需求转化成具体、详细、结构化的软件需求,是需求分析的重要目标。

1. 将抽象的需求具体化

在进行需求调研时会发现,用户提出自己的需求时总是不会按照你希望的路子提出来的:有的人因为对信息化管理一无所知,根本不知道你想要什么,只是为了应付领导布置的任务;有的人因为害怕承担责任,说得太具体万一软件做出来不能用怎么办呢;还有的人由于处于管理层级较高的地位,喜欢做抽象的工作布置,习惯了宏大叙事的调子……将用户抽象的需求具体化,是需求分析过程中一项相当困难的事情。



案例：抽象的需求

看看这些典型性抽象需求。

- 我想知道我的下属每天都干了什么。

抽象指数：☆

- 我们的客户经常抱怨我们的售后服务,但我不知道我们哪里做错了,他们究竟在抱怨什么呢?

抽象指数：☆

- 我想知道我们业务员的销售业绩跟绩效工资是不是有关联性,我应该给他们多少提成合适呢?

抽象指数：☆☆

- 我们经常发现仓库丢了东西,但我们查不到原因,我们甚至不知道究竟丢了多少。

抽象指数：☆☆

- 我们车间在交班时乱糟糟的,需要持续太长时间,怎么才能让他们交班快点儿呢?

抽象指数：☆☆☆

- 我们的生产计划老是排不好,经常有些人很闲,有些人很忙。

抽象指数：☆☆☆

- 我们的员工平均薪酬高于同行水平,但离职率并不低。

抽象指数：☆☆☆☆

- 我们的一等品率明显低于同行平均水平。

抽象指数：☆☆☆☆

- 我们要提高订单的准时交单率。

抽象指数：☆☆☆☆☆

- 我们所有的工作都要用软件管起来。

抽象指数：☆☆☆☆☆☆☆☆☆☆

相信在需求调研的过程中一定或多或少遇到过本案例中的类似需求,从严格意义上讲,这些并不能说是需求,只能说是某种问题罢了,提出的人不是在提软件需求,而是在抛出他的问题,他期待你的工作能够解决这些问题。不要把这些抽象需求当成需求,要当成工作目标。用了软件系统后,这些问题解决得越多,工作价值越大,如果不能解决这些问题,往往说明这是个不成功的项目。有了目标,接下来要思考的是,通过什么方法才能够实现这些目标呢?这个思考的过程就是将抽象需求具体化的过程,有的时候可以引导用户思考后去提出自己的方案,有的时候只能由需求人员自己设计方案——也许这是最能体现需求分析者价值的地方。例如,案例中提到的仓库丢东西的问题,具体分析后,可以考虑,如果将仓库所有的物品进行规范编码,所有的入库、出库都需要通过软件系统处理,保证软件系统中的信息实时、正确地反映物料的流动状况,这样至少知道了仓库中“应该”有多少东西,盘点后如果丢了东西完全可以责成仓库管理员负责赔偿,从而极大提高了仓库管理员的管理责任心——那么是不是可以解决仓库丢东西的问题呢?将这种方案跟相关人员讨论决定,顺着这个路子,就可以将这个抽象需求具体化,最终能够落到实处。

2. 将自然语言描述的需求结构化

用户描述需求总是非常随意的,他们使用平时正常沟通的语言(也就是自然语言)描述需求,这种需求的主要特点就是严谨,容易有歧义,这种需求自然是不能直接让开发者处理的,开发者需要的需求是描述明确、精准、没有二义性的。需求分析者作为用户与开发者的沟通桥梁,有义务将用户用自然语言描述的需求结构化。当然,对于需求分析者来说,将需求结构化几乎贯穿于所有工作中,如功能设计、界面设计等。将用户的描述转换成更精准的语言,转换成更接近于IT人使用的语言,这才是需求结构化的第一步。使用这种方式描

述的需求,虽然充满了 IT 术语,但只要稍加解释一般用户也能理解清楚。



案例：将自然语言描述的需求结构化

调研主题：如何给客户发货？

用户描述：我们会根据销售合同整理当天的发货计划(发货计划中一般包括这周需要发货的所有销售合同),跟客户确认是不是需要发货,客户一般都要求发货越快越好,但特殊情况下也会要求拖几天。确定要发货后,先到成品仓库核查产品库存,如果库存足够,就发货,如果不够,就联系客户,如果客户着急,就先发一部分过去。发货前再跟客户确认好收货地址,因为有的时候,客户会要求不同的货物送到不同的地方,虽然合同里写了收货地址,但这个地址经常会变化。

分析：用户在这里描述了根据合同发货的过程,由于是用自然语言描述的,需求就显得很模糊,这种需求需要经过结构化处理。经过进一步询问确认后,需求人员做了结构化的需求描述。

(1) 根据销售合同生成发货计划单,一天生成一个发货计划单,生成规则:交货期在今天以前或者在未来 7 天之内,尚未发货,并且对应商品物料有结存数量的合同。

(2) 一个发货计划单可以包括多个销售合同,一个销售合同可以对应多个发货计划单,发货计划单与销售合同的关系是多对多的关系。

(3) 发货需要发货单,销售合同支持分批发货,销售合同与发货单的关系是一对多的关系。

(4) 一个发货单支持多地址发货。

(5) 需要提供合同发货延期的处理功能。

3. 注意避免理解偏差

所谓的理解偏差,主要是指需求分析者对用户所提需求没有理解到位,用户明明想表达的是这个意思,却被理解成了另外一个意思。这是一个沟通问题,说者觉得自己说得很清楚了,听者也觉得自己听得很清楚了,可偏偏双方就是没有真正理解对方。为了避免理解误差,一般可以从以下这些方面努力。

(1) 提高沟通能力。为了避免理解偏差,需要提高自己的沟通能力,多从对方的立场考虑问题,当对方描述某件事时,要从对方的角度思考这些描述的内涵。同样一句话,不同的人想表达的意思可能完全不一样。举个简单的例子,同样是这句话,“我觉得我的工作太多了”,有的人的意思是“能不能给我减减负”,有的人的意思是“我的工作效率很高”,有的人的意思是“羡慕我吧,我的权力很大啊”,这就要结合说这句话的人的岗位、工作职责、说话时的情境、语气等去理解。

(2) 提高沟通频次。所谓提高沟通频次,就是人们常说的“多沟通”,一是要引导对方多说话,说得多了,有了更多的素材供分析,自然更容易理解这些描述的内涵;二是对不理解的或觉得理解起来有困难的内容,多向对方询问,换成你的表达方式让对方确认是不是这个意思。

(3) 学习对方工作领域的知识。用户有自己的知识领域,需求分析者也有自己的知识领域,前者满脑子的业务术语,后者满脑子的 IT 术语,有时候两者真难沟通。每个人的知识面不同,要想沟通顺畅,两个人的知识面重叠的地方越多越好。因此为了能够跟用户顺利

沟通,需求分析者应该积极学习相关的业务知识,学得越深,学得越宽,就越容易理解用户的描述(做久了需求分析工作的人,几乎无一例外地会成为某些业务领域的行家),对于用户而言,如果要开发信息系统,也应该积极学习 IT 知识,信息系统开发绝不仅仅是软件公司的责任。

当然,不管用户与需求分析者如何努力,理解偏差总是避免不了的,只有程度与范围的不同。



案例：需求调研中的理解偏差

某技术学校需要开发信息管理系统,小王在进行需求调研。这天,他跟学工处管后勤的李老师做了一次需求访谈,李老师说他想知道每天有哪些学生打卡晚了,哪些学生没有打卡,另外需要每周出一个考勤报表,统计本周所有考勤异常的学生。由于这一天李老师的工作很忙,这项需求听起来也很明确,小王没有做更加深入的调研,就去拜访其他用户了。

回来后,小王经过仔细分析,发现李老师的需求并没有那么简单。所谓的学生晚打卡就是上课迟到了,没有打卡就是旷课了,为了知道学生应该在什么时候打卡,在什么地点打卡,就需要知道学生的课表,需要在课表中定义学生每节课应该在什么时候、什么地点打卡,如果课表发生了变化,还需要老师及时调整课表,而学校有规定,调课是要教务处批准的。为了进一步了解排课的问题,小王又到教务处做了调研。

小王规划了学生上课考勤的解决方案:首先,需要一个排课功能,供教务处老师在每学期开学时排好一个学期所有的课程;然后,提供调课申请功能,一旦课表发生变化(如教室、教师等变化),相关老师需要使用这个功能申请调课;每节课开始半小时后,系统进行运算,得到每个班级在该课次有哪些学生迟到,哪些学生旷课;最后,提供考勤记录查询、统计功能。

小王将方案拟好后,兴冲冲地去找李老师讨论这个方案的可行性。可是,李老师问了一个问题就把小王弄蒙了:我是管后勤的,我只管学生有没有按时到宿舍就寝,你为什么跟我说这些呢?

3.1.2 控制需求

所有的项目都是有范围的,控制项目范围是项目管理过程中一项相当重要的工作,这项工作稍有不慎就会导致非常恶劣的后果,轻者导致活干了却没有利润,大家白忙活一场;重者导致工作量无边无际,无法交付验收,导致项目烂尾。对于软件开发项目来说,控制需求是项目范围控制工作的一部分,是相当重要的一部分。无论什么公司,什么部门,什么业务,信息化工作都是无止境的,如果不认真做好需求控制工作,一定会陷入无边无际的需求大海之中,最终把开发团队搞得精疲力竭,而用户却总是不满意,因为他们还有那么多的需求没有实现。

另外,除了那些超出项目范围的需求,还有许多需求属于不值得实现或者技术上不能实现的,也需要需求分析者去识别,从而加以控制。

1. 识别超出项目范围的需求

用户的需求不能漫无边际,所有的需求都应该在项目范围之内,做软件的都知道这一

点,但是,并非所有的用户都知道这一点。为了做好需求控制,需求分析者在进行需求调研与分析的过程中就要时刻将这种观点向用户灌输,只要持之以恒、坚持不懈,无论多么难缠的用户都会逐渐接受你的观点。

接下来要做的是,让用户知道需求边界在什么地方。首先想到的自然是合同,市场行为当然以合同为准,合同要我们干什么我们就干什么,合同没有的我们就干,这话说得是在理,不过在实际工作中往往是另外一个样子。先了解一下项目合同是怎么来的吧。一般来说,项目合同都是销售、售前在客户那边将什么活儿都答应了,然后才签下来的,在这种情况下很难在合同中对项目的范围进行非常明确的规定。因此,先打消通过合同来控制项目范围的想法,即使有些项目的合同签得比较规范,合同中将项目的范围基本确定了,也不可能详细到每个功能的每项需求、每个规则,如果有,这恐怕是个外包开发合同,不大可能是个做信息管理系统的合同。

为了让用户理解需求边界,首先要确定好项目目标,这个目标应该是在项目启动时双方经过讨论达成的共识,后面所有的工作都应该围绕这个目标开展——这件事主要是项目经理的职责,如果这事没做或没做好,项目经理难辞其咎。对于需求分析者来说,要对这个项目的目标了然于心,如果目标不清晰,就要想办法去了解,或者促成相关人员确定好目标。当然,也没有必要把确定目标这件事看得过于严肃刻板,未必就一定要通过规范文件来确定目标,也未必一定要通过非常正式的会议讨论确定目标,关键是大家在内心深处能否达成共识。只要能达成共识,在饭桌边、电梯中的讨论都有用;不能达成共识,写下来签字画押也意义不大。

有了清晰的目标后,只要用户的需求偏离了这个目标,就立即指出来这个需求超出了边界。原则是,宁可在这个阶段的目标实现了以后再设置新目标,也不要不停地修改一个目标,甚至根本不把目标当回事随意践踏。需要注意的是,如果用户的需求超出边界,需要尽早指出来,越早越好,最好在他刚提出来时就果断、坚决地阻断,这样用户就能够体会到你的边界之墙。如果拖拖拉拉,态度暧昧,今天说可以做,明天说不可以做,就会让用户生疑,逐渐就会觉得你这个人不是真心为他服务。这种想法一旦形成,要想再扭转过来就难了,从此以后工作会越来越难做,想砍掉任何一项需求都将是个异常艰巨的任务。许多初学者就是这样把自己与项目一起带入深渊的。



案例：限制超出项目范围的需求

某公司需要开发一款 OA 系统,小王经过需求调研后,将用户的需求做了整理,总结后大概包括以下这些需求点。

- 发布通知公告、新闻。
- 内部信息发布,包括内部消息推送、内部邮件发送、内部论坛。
- 内部通信,包括内部即时通信、文件传输、群聊、发送短信。
- 采购申请审批流程。采购单审批通过后需要推送到现有的 ERP 系统采购模块。
- 物品领用申请审批流程。物品领用单审批通过后需要推送到 ERP 系统库存模块。
- 单据报销审批流程。报销单审批通过后需要推送到现有的财务系统。
- 请假审批流程。
- 用车申请流程。

- 会议室申请流程。
- 公文管理,包括收文、发文管理。
- 工作计划与工作日志。
- 管理客户信息,设置拜访计划,登记拜访记录,登记客户服务日志,录入客户投诉记录。

小王经过分析后,向用户指出:管理客户的需求,明显属于CRM系统的范畴,明显超出了OA项目范围;将信息向ERP、财务系统推送的需求,由于不在这一期的目标中,建议先不考虑,在下一期再考虑如何整合这些不同系统的信息。

2. 识别错误的需求

一般用户对软件并没有太深刻的理解,对信息化管理也没有那么远的预见性,对于自己所提出的需求一旦被实现后,究竟对自己、对同事、对公司的未来有什么影响也是一知半解(不排除有些公司的某些人有这种能力,但毕竟是少数,可遇而不可求)。这样就注定了,用户所提的需求不可避免地带着个人的烙印,这些五花八门的需求可能毫无逻辑性,可能前后矛盾,可能在技术上根本无法实现,可能做出来得不偿失,甚至可能如做梦般虚无缥缈。这些需求,统称为错误的需求。作为开发方,满足客户的需求是义不容辞的责任,但并不包括错误的需求。



案例：识别错误的需求

某公司需要开发一套生产成本管理系统,小王作为需求分析师到装配车间调研。该车间有一个装运工的岗位,主要工作是,将装配线需要的原材料或半成品从仓库领出,运送到装配线上,将装配线完工的成品送到包装车间。装运工的成本如何分摊到生产任务单呢?车间有人提出,需要根据装运工为每个生产任务单的服务时间进行分配,也就是说,假设某个装运工每天的人力成本是80元,一天服务了4个生产任务单,用时分别是1、2、3、2个小时,那么分配到这4个生产任务单的成本应该分别是10、20、30、20元。小王经过分析,得出结论,这是个得不偿失的需求。要知道计算成本本身是有成本的,如果这个成本太大,将成本算得再准也是没有意义的。想想看,车间总共就两个装运工,一个月的人力成本也就几千块钱,各个生产任务单分摊到的这个成本微乎其微,几乎不会对根据成本数据进行的决策有任何影响。可为了如此计算成本,需要记录装运工为每个生产任务单用了多长时间,有时候两个生产任务单的材料一起装运,还要考虑怎么分配,有时候两个装运工共同装运一个生产任务单,还要考虑怎么合并,而车间每天处理的生产任务单少则十几个,多则上百个,想想这是多大的工作量啊,就为了这么点儿人力成本的分摊!很明显,这是个错误的需求。

为了把问题说明得清楚形象一点,这个例子有些极端,但要知道,在实际工作中,类似的不切实际的根本不值得去实施的需求是相当多的。太多根据各种人提出的需求开发出来的功能,最终无人问津,被束之高阁,究其原因,许多都是因为这些需求是得不偿失的需求,开发出来后,到了实施阶段才发现为了实现当初的目标,需要做多少在系统规划阶段根本意想不到的工作(不是指软件开发的工作,而是用户自己需要完成的工作)。或许不能算是意想不到只是没有好好去“想”罢了。最终只好放弃。

3. 识别技术上不能实现的需求

当需求分析者面向用户时,代表的是他身后的整个研发团队,要做好需求分析,需要对自己团队的技术能力有非常清楚的了解,哪些事情能做,哪些事情不能做,哪些事情虽然貌似可以做但代价太大,等等。每个团队都有自己的技术边界,那种无所不能、包治百病的团队只有在神话中才会出现。

理解了这一点,就知道“识别技术上不能实现的需求”是需求分析者的基本素质。用户的需求天马行空无所不包,往往在一个项目里,最终实现的需求只占初始需求的一小部分,那种没有被实现的需求,大部分都是因为超出了项目范围,但确实也有或多或少的一部分是因为开发团队根本实现不了。对于技术上不能实现的需求,要尽早跟用户说清楚,有时候,因为某种原因,可能羞于启齿,那也要想办法让用户知道、感觉到、悟到这个需求是不切实际的,除了放弃别无选择。有时候,也许你会觉得用户的需求很合理,但研发团队就是实现不了,心理上简直无法交代,但无法交代也得交代,要知道,即使团队有引进某种新技术的打算,也很难用到当前的项目中,技术积累并非一朝一夕之功。

3.1.3 挖掘需求

做信息系统,除了实现用户正确的需求外,还有重要的一点需要注意,就是仅实现用户提出来的需求是远远不够的,如果认为确定需求就是用户开价我们还价的过程,就是一个漫天要价就地还钱的过程,那么就大错特错了。

用户不是搞软件的,对未来的信息化管理系统一般不会有太深刻的理解,要想让他们非常有体系地提出要求,然后研发团队实现这些需求,客户的信息化管理体系就建立起来,顺利地运行下去,实际工作中不可能这么简单。在进行需求调研与分析的过程中,要注意不能被用户的需求限死,能引领用户才是高手。对于你的工作来说,满足用户提出的需求并不是目的,建立客户在相关领域的信息化管理体系才是目的。这个目的就决定了,用户提出的需求并不一定需要全部实现,用户没有提出的需求也不一定不实现,一切以能否成功建立信息化管理体系为出发点,为奋斗目标。让用户提需求只是为了实现这个目标的手段之一,有些必要的需求是要挖掘的,无论用户是否提出来,都是要实现的。



案例：挖掘需求

某财务部门需要开发固定资产管理软件,小王是这个项目的需求分析师。用户提出的需求大概是这样的。

- (1) 管理所有固定资产的档案。
- (2) 登记所有固定资产的领用人、存放位置。
- (3) 提供固定资产的折旧费录入功能,系统根据原值、录入的折旧费来计算现值。
- (4) 提供折旧费统计表。

小王经过进一步的沟通,知道用户提出录入折旧费的功能需求,是因为想每月计算固定资产的月折旧费后录入系统,需要提供折旧费统计表的目的是为了根据这个统计表在财务软件中制作折旧的记账凭证。了解清楚后,小王提出如下建议。

(1) 用户提供固定资产折旧的计算规则,由软件系统每个月在某个固定的时间点自动计算折旧额,根本不需要用户自己计算后再录入系统。

(2) 用户提供记账凭证的生成规则,如果财务软件供应商开放接口,本系统可以在生成记账凭证后通过接口推送到财务软件,不需要用户根据折旧报表制作记账凭证。



3.2 整理需求

花了那么大的工夫收集并确定需求之后,自然要做好需求的整理工作。需求整理,不是做会议纪要,不是做工作日志,不是简单地将每个用户所提的需求一条一条写下来,而是一个综合分析的整理过程。通过整理,使得需求更有目的性,更有系统性,更明确,更易理解。需求经过整理后一般会生成需求调研报告与业务流程图,这是后面工作的纲领性文件。

3.2.1 需求调研报告

需求调研报告是经过需求调研并确定需求后必不可少的规范文档,一般包括调研背景、专业术语、项目目标、期待解决的问题、项目范围、双方的约定、相关资料、所有需求点、相关数据、相关系统、注意事项、待定问题等。不同的项目,这个文档的篇幅相差很大,一个小系统,可能寥寥几页就可以了;而对于一个大型系统,可能需要数百页、上千页。需求调研报告的格式也不需要死搬教条,不同公司、不同团队,甚至不同项目都可以根据具体情况设计符合自己要求的格式。当然,为了沟通方便,同一团队建议还是采用相同的文档模板比较好。这里提供一种需求调研报告模板,供读者在工作中参考。



案例：需求调研报告模板

1 引言

- 1.1 编写目的 //为什么要编写本文档
- 1.2 调研背景 //简述调研过程,参与人等
- 1.3 专业术语 //解释本文档中用到的专业术语

.....

2 概述

- 2.1 项目目标 //希望对企业管理改善达成的目标
- 2.2 期待解决的问题 //希望通过本项目解决的管理问题
- 2.3 项目范围 //本项目的工作边界
- 2.4 双方约定 //澄清双方理解上可能产生冲突的地方

.....

3 相关资料

- 3.1 组织结构
- 3.2 用户名单
- 3.3 重要业务规则

.....

4 需求

- //整理所有需求,这是本文档的核心内容
- 4.1 财务部 //可以以业务领域为维度,也可以以软件功能为维度
- 4.2 计划部

.....

5 数据 //整理本系统需要处理的所有数据

5.1 销售合同

5.2 采购单

.....

6 相关系统 //可能与本项目有关系的其他软件系统

6.1 系统 A

6.2 系统 B

.....

7 其他

7.1 注意事项 //注意点

7.2 待定问题 //没有定论,还需要继续讨论的问题

.....

下面根据上述案例介绍一些重要章节的撰写方法。

1. 项目目标

项目目标指软件开发、实施完成后希望对企业管理改善所达成的目标,也就是说,这个项目的目的是什么,如给企业带来什么价值,给企业解决哪些问题等。不过在大部分情况下,无论是甲方还是乙方,这个目标并没有那么清晰,不清晰的原因主要是因为目标是否实现很难评判。一个难以评判的目标自然很难清晰,清晰了也没有意义,无法证明目标已经实现了。看看常见的一些貌似可以用来作为目标的量化指标,如投资回报率、资金周转率、库存周转率、客户投诉率、坏账率等,只要管理规范一点儿,这些指标都是可以算出来的,但谁又能说得清楚这些指标跟项目之间有什么关系呢?不能说资金周转率的提高都是财务软件的功劳吧?不能说库存周转率的提高都是库存管理软件的功劳吧?对于软件公司而言,也不会正式文档中把这些指标当成目标,销售在推销时可以拿这些指标举例,到了项目真正启动时,没有几个项目经理敢拿这些指标当项目目标,因为对于项目组而言,这些指标太不可控,起决定作用的因素太多了,软件的作用只占一小部分而已。

考虑到这些原因,财务指标类的项目目标能不写就不写,应该写那些能实现的、可以看得见摸得着的目标,这种目标对项目组才有指导意义,才能给甲乙双方以努力的方向。



案例：某销售管理系统的“项目目标”

对销售部的拜访、签单、发货、回款一系列的工作流程进行信息化管理。

业务员可以通过手机随时汇报拜访情况,领导可以通过本系统监管业务员的拜访工作。

客户可以通过网页远程下单,也可以通过电话、传真下单,每个订单要管理对应的业务员、产品、数量、单价、质量要求、交期要求、交货地点要求。

对所有产品使用条码管理,发货时使用扫描枪扫描条码发货。

发货需要根据订单的要求发货,如果客户对发货要求有变更,可以随时进行变更。

对每一批装运的货物要落实到责任人,送货人送货后需要获得客户签字的送货单,否则需要赔偿。

允许客户赊账,但不同的客户允许赊账的信用额度不一样,系统需要每个月根据客户的历史交易、回款情况自动计算信用额度。

没有按要求回款的,到了账期发送催款提醒(邮件、短信)给业务员。

每个月给每个客户生成往来对账单,自动发送邮件给客户。

计算业务员、大区经理的销售绩效。

2. 期待解决的问题

期待解决的问题,指希望通过本系统解决当前存在的哪些问题,这可以看作是项目目标的一部分。有些项目期待解决的问题非常明确,如“为了解决信息孤岛的问题需要开发一个信息集成平台”,也有些项目就非常模糊虚妄,如“为了解决管理效率低下的问题需要上ERP”。明确的问题,可以写下来当成项目目标的一部分,给双方的努力提供方向。



案例：某库存管理系统“期待解决的问题”

账实不符：现在仓库里面究竟有多少东西除了仓库保管员没有人搞得清楚,账本根本不准,有时候生产计划员为了编排生产计划,不得不跑到仓库亲自核查某种关键物料的库存数量。

丢东西：有时候,某种物料明明记得上次刚买了足够的数量,在生产中明显没有用到那么多,但在仓库中就是找不到了,找仓库保管员,仓库保管员坚持说仓库中应该还有,可能被压在什么东西下面了,过一段时间应该就会出现,而最终往往不会出现。

堆放混乱：货物在仓库中堆放混乱,除了仓库保管员,没人知道怎么才能找到需要的东西,有些体积小的物料,要找到简直跟大海捞针一样。

货架利用率低：现在的货架利用率很低,为了防止找不到东西,都不敢在货架上堆放太多的东西,只能横七竖八地堆在地上。

收货太慢：供应商送货时,仓库保管员不清楚什么应该收,什么不应该收,每次都要找采购、生产、计划部门的一大堆人沟通,有时候需要处理大半天才能把一车货收好,供应商意见很大。

3. 项目范围

项目范围确定这个项目的工作边界。如何描述项目范围,不同的项目要求是不一样的,一般会由销售人员对客户的承诺、合同的要求、本公司跟客户的关系、双方项目经理的工作方式、客户相关人员的工作习惯等各方面的因素共同决定。可以考虑从两种维度来阐述项目范围,一是功能性的范围,二是业务性的范围。所谓功能性的范围,是指软件会提供哪些功能,如物料出入库、存货成本计算、智能调度等;所谓业务性范围,是指信息系统会管理企业的哪些业务,如采购部采购合同管理、销售部销售合同管理、计划部生产计划管理等。有些功能具有很强的业务关联性,如某成本运算的功能,一定是成本会计在成本核算中用的;有些功能跟业务关联性不强,如工作日志登记,所有业务领域的人员都可以使用。不同的项目,根据项目的特点可以从不同的维度编写项目范围描述,可以从功能的维度,也可以从业务领域的维度,也可以混用。对于那些比较通用的系统,由于功能有很强的通用性,跟业务领域关联性不大,可以以功能为维度描述项目范围,如常见的OA系统;而对于那些业务性比较强的系统,由于主要强调的是涉及企业哪些业务领域,可以以业务为维度展开描述,如

销售管理系统。

由于撰写项目范围的目的是双方确认这个项目要完成哪些工作,事关重大,因此,应该描写得清楚明白,少用“尽量、争取、等等、可能”之类充满变数的、可以进行各种解释的词语。例如,写成“包括销售管理、采购管理、库存管理、客户管理、财务管理 5 个模块”,而不是写成“包括销售管理、采购管理、库存管理等模块”。

还要注意,项目范围的描述只是对这个项目的工作所进行的概略性描写,不需要长篇大论,毕竟,整个需求调研报告都可以看作是对项目范围的界定。



案例：某销售管理系统的“项目范围”

本系统所管理的业务范围包括：潜在客户拜访,客户下单,销售合同签订,销售发货,跟踪服务,市场促销。(根据这个描述可知,销售部办公人员的请假、考勤等方面的需求不在这个项目的范围之内。)

本系统包括以下管理功能：潜在客户信息,客户档案,业务员基本信息,拜访日志,客户订单,销售合同,发货单,退货单,服务日志,投诉记录,客户建议,市场促销活动。(根据这个描述可知,对销售发票、客户回款的管理不在这个项目的范围之内。)

4. 双方约定

双方约定指将某些容易引起双方误解,容易起争执的事情提前阐述清楚,免得到项目后期扯皮。约定的内容可以包括各方面的事情,如项目范围上的,技术上的,服务上的,功能上的,等等,只要觉得这个事情可能会因为双方的理解偏差而影响到项目的验收、上线,就应该写个约定,对方如果不同意就可以提前讨论,免得到项目验收时再来做这方面的事情,那就麻烦了。



案例：某销售管理系统的“双方约定”

手机端只支持 Android 系统的手机(操作系统版本为 Android 8.0 或以上),不支持其他操作系统的手机(包括 iOS 手机)。

地图使用的是某第三方提供的免费地图,对地图的处理会受限于地图接口(详见该公司提供的接口说明文档)。

市场促销活动,只能进行活动的录入、维护,不能分析某次促销活动对销售的影响。

服务日志只能在本系统录入,或从 Excel 导入,不能从 QQ 中导入聊天记录。

本系统不管理销售发票与销售回款,因此也不能出具客户往来对账单。

所有的开发基于某平台,本平台只支持两种标准界面风格(“红色火焰”与“绿色树林”),新风格的开发不在本项目范围之内。

5. 相关资料

相关资料是指在需求调研过程中收集的、经过整理分析的、对后阶段的工作影响较大的资料,常见的如公司的组织结构、人员名单、重要的业务规则等。注意这里强调是经过整理的资料,而不是将用户提供的所有原始资料都贴在这里——不要将需求调研报告搞成客户单据电子档案室。

组织结构、人员名单这些资料,一般在需求调研时都是需要收集整理的,客户可能会以

各种方式提供这些资料,口述的、手写的、Excel 表格的、Word 文档的,什么都有可能。由于这些资料对系统的初始化、未来在系统中的工作分配、用户权限管理等方面会有很大的影响,因此,团队最好设计一些规范的文档格式,用以整理这些资料。例如,如何画组织结构图,如何表达每个人应该有哪些权限等,既可以促使工作规范化,又可以给沟通带来方便。

有些对后面的程序算法可能会产生重大影响的、复杂的、核心的业务规则,也可以在这里进行整理(有些复杂的业务规则,可能需要几页甚至几十页才能说清楚),如某标准成本的计算要求,某智能排单的算法等。当然,这里的整理只是将用户的描述或者提供的文档整理得更清楚、明确、规范一些,还远没有到功能设计甚至算法设计的地步。

6. 需求

需求部分整理用户所提出的各种需求。需求调研,目的就是获得用户的需求,这部分自然是调研报告的核心。需求的描述可以从各种维度去描写,不同的项目可以采用不同的方式,最常见的也有两种维度,一种是按照业务领域的维度,另一种是按照功能模块的维度。例如,某企业的信息管理系统,可以按照部门描述,从公司管理层,到各个部门,到各个岗位,依次整理,总经理需要什么,原材料仓库需要什么,成品仓库需要什么,一车间需要什么,二车间需要什么,等等;也可以按照功能模块描述,OA 需要什么,库存管理需要什么,计划管理需要什么,知识管理需要什么,生产管理需要什么,等等。当然,如果项目够复杂,也可以交叉描述,每个部门下面描述对每个功能模块的要求,或每个功能模块下面描述各部门的要求等,这个没有一定之规,看项目情况。需求调研时,用户提出的需求中,有的需求是超出范围的,有的需求是技术无法实现的,有的需求是不合逻辑的,有的需求是自相矛盾的,在整理需求调研报告时,都要跟相关人员沟通确认好,实在没有定论的可以先放到“待定问题”中,在“需求”这部分整理的需求应该都是确定的、双方有共识的。



案例：某车间调度的“需求”

4 需求

4.1 总经理

4.2 财务部

.....

4. x 二车间

4. x. 1 车间主任

4. x. 2 车间调度

(1) 需要获得计划部的生产计划,有了新的生产计划,可以发送提醒短信到手机。

(2) 需要有图形化的界面用以监控当前车间各个机器的运行状态,是在工作中还是闲置中,工作中的机器需要显示当前处理的任务单。

(3) 需要系统智能计算生成最优的生产任务排单建议,调度人员可以根据排单建议调整后下生产任务,同一个生产计划单可以下多个生产任务单,是一对多的关系。(生产任务的排单规则参见“相关资料”。)

(4) 需要根据生产任务单打印生产任务卡,需要用条码打印卡号,任务卡格式详见报表格式文档。

(5) 当某生产任务超过预定完成时间 4 个小时还没有完成时,系统需要发送提醒给调

度人员(可以发送提醒短信到手机)。

(6) 机器处于维修、保养状态时,系统中需要有显目的标识。

(7) 某生产任务单领料失败时,要及时提醒调度人员(不需要发短信,在系统中通过内部消息提醒)。

(8) 需要一个统计分析表,分析最近一个月所有没有按计划交期要求完成的生产任务的产生原因,包括4种原因:材料原因、机器原因、生产能力原因、任务安排原因。

材料原因的判断规则:……

机器原因的判断规则:……

生产能力原因的判断规则:……

任务安排原因的判断规则:……

4. x. 3 车间统计员

……

7. 数据

这里整理本系统需要处理的所有业务数据。现在还没有到设计规范的数据字典的阶段,只要整理好本系统需要处理哪些数据项就足够了。由于我们的目的是做信息管理系统,很多情况下,数据对确定项目范围有至关重要的影响。例如,假如所管理的数据中没有任何员工的相关信息,那么所有管理员工档案相关的功能明显都不应该在本项目的范围之内。



案例:某采购管理系统的“数据”

5 数据

5.1 供应商

供应商代号、名称、组织机构代码证号、邮编、地址、联系人、联系电话、邮箱、传真

5.2 采购订单

订单头:采购订单号、供应商、下单日期、交货日期、采购员、要求说明

订单行:采购物品、规格、数量、单价、备注

……

8. 相关系统

相关系统指跟本系统有关系的其他软件系统,一般包括:本系统需要跟它协作共同完成某些事情的,例如,现在需要开发采购管理软件,但采购单的审批过程是在OA系统中完成的;本系统需要通过接口跟它进行数据交互的,例如,现在需要开发考勤分析软件,根据班次、打卡记录分析考勤状况,但打卡数据需要从一卡通系统中获得;本系统需要依附其上的,例如,某公司采购了一套ERP软件并正常使用,现在需要围绕它开发报表管理系统。

9. 待定问题

待定问题是指在需求调研过程中,双方没有达成一致意见,或者到目前为止还没有讨论出结论,或者已经知道需要处理但还没有着手处理的问题,等等。需求调研报告这个文档应该是在需求调研启动不久就开始撰写的,在调研过程中,只要有这方面的问题,就可以在这里记录整理,一旦经过讨论确认,就从这里清除,到最后交稿时,自然就不应该存在待定问题了。



案例：某考勤分析系统的“待定问题”

获取一卡通打卡数据时,是从一卡通系统的数据库中直接读取数据,还是由一卡通系统通过接口将数据推送过来?

考勤分析的计算过程,是在上班后实时计算,还是在夜里定时计算?

允许员工提前多久打卡,滞后多久打卡算迟到,滞后多久打卡算旷工?

是否要指定员工的打卡地点,还是只要在公司内打卡都算?

3.2.2 业务流程图

企业做信息化管理系统,一个重要的目的就是规范企业的工作流程,按照规范工作可能不是效率最高的工作方式,但一定是最好控制的工作方式,企业规模越大,就越难控制,规范工作流程的要求就越强烈。业务流程图是促使工作规范化的重要工具。

需求调研后画出业务流程图,一者可以发现当前企业中有哪些不规范的地方,将来可以在这些方面多下功夫,从而起到事半功倍的效果;二者可以让自己对企业的工作了解得更加深刻。

画流程图的方式各种各样,工具很多,要求各异,每个团队都可以制定自己的流程图规范——需要强调的是,规范不是死搬教条,适合自己的才是最好的。这里介绍一种画业务流程图的规范,供读者在工作过程中参考。



案例：业务流程图

某车间装配业务流程图如图 3-1 所示。

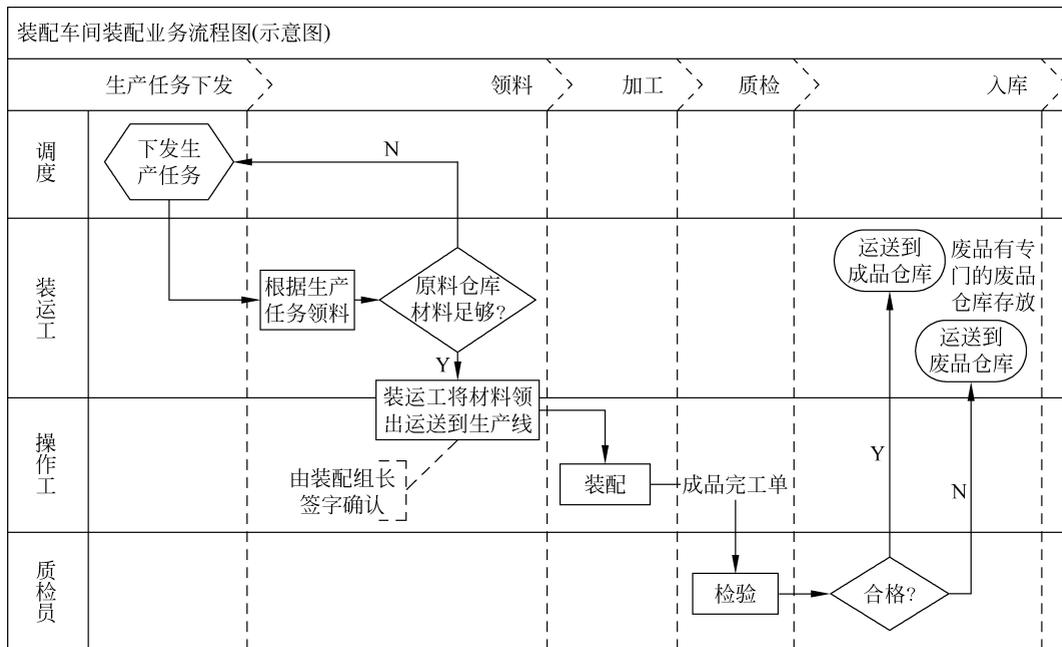
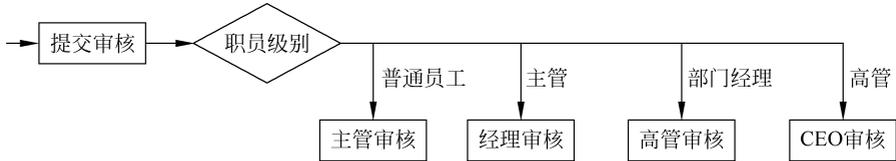


图 3-1 装配车间装配业务流程图

案例中的图形用法介绍如表 3-1 所示。

表 3-1 流程图各元素表达的意义

图形	意义
行	每一横道代表一种职能角色,可以是某一部门,某一班组,某一岗位等,横道中的内容代表这个职能角色负责的工作
列	每一竖道代表流程中的某一大步骤,这只是一个大概的划分,目的是为了让这个图更容易被理解,不是必需的
	表示流程启动
	表示流程结束
	表示某种操作、动作、行动
	表示条件判断,意味着在这里出现流程分支,如果分支超过两个,可以用这种方式表示: 

画图规范介绍如下。

(1) 为了使读者容易阅读,流程图的总体流向趋势是从左上到右下。

(2) 连线方向允许向右、向下、向上,但不允许向左(除了条件判断表示返回外)。如图 3-2 所示的连线方式是不允许的,因为线段 A→B 的方向是由右向左的。

(3) 为了使读者容易阅读,连接线之间应尽量避免出现交叉点,如果不得已,交叉点越少越好。

(4) 使用折线连接各图形,不得使用斜线,如图 3-3 所示。

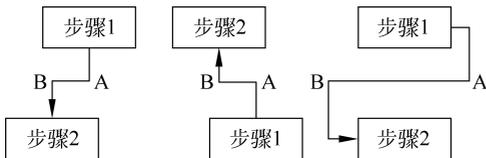


图 3-2 流程图中不允许出现的连线方式

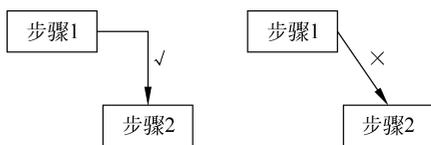


图 3-3 流程图中不允许使用斜线

(5) 有时候,为了强调物料、单据的流转,可以直接将在这个步骤流转的物料、单据写在连接线上,表示这些物品从这个步骤流转到另外一个步骤。

(6) 如果有需要补充说明的内容,可以直接在相应工作步骤旁边使用文字说明,或者使用备注框。

3.3 系统蓝图设计



在进行软件策划之前,要做一件非常重要的事情:规划如何建立相关业务的信息化管理体系——这个过程称为系统蓝图设计。管理软件是为管理服务的,每家企业的管理都是

一整套体系,怎么管理资金,怎么管理生产资源,怎么管理人力资源,怎么获得材料,怎么生产产品,怎么销售产品等,种种领域盘根错节,相互影响,它可能完善,可能残缺,可能精密,可能粗放,可能历史悠久,可能出生不久,但它一定不会简单,它非常复杂。如果把这个体系比作一个生命体的话,你所做的工作就是在这个生命体中植入某种组织器官,这个器官不但不能妨碍这个生命体的运作,而且还要让它运转得更快、更好、更强,要让这个器官与生命体融为一体,克服最初的排斥反应,让创口逐渐愈合,最终成为这个生命体的一部分——这就是做管理软件需要完成的工作。

3.3.1 进行价值分析

所谓价值分析,不是进行这个项目的可行性分析,而是要分析这个系统将来会给客户、用户带来什么,如果软件给客户带来的不是利益,而是麻烦,那么可能一开始就注定了这将是一个失败的项目。作为需求分析者,可能并不需要去分析可提高多少利润率,降低多少积压资金,加快多少库存周转率等,但是,要时刻记住一点,要让自己的软件系统成为管理的好帮手,改善管理是软件的价值所在,如果做不到这一点你的工作将没有任何意义。

要分析这个软件系统会对以后的管理带来什么价值,会对管理工作有什么影响,管理方式会因之而做出什么变更。虽然一开始不大可能考虑到每个管理细节,但至少要对相关业务未来的信息化管理蓝图做到胸有成竹,实现这个信息化管理体系是目的,通过这个信息化体系改善管理工作是其价值所在,以后的一切工作,都应该向这个目标努力。要实现你在这里的价值,就要有大局观,要习惯于系统性地分析问题,那种把业务割裂开来,做一块是一块,只顾这边不顾那边的工作方式,是一叶障目不见森林,风险相当大,有可能带来无穷无尽的麻烦,导致不断返工、变更、推翻、重建,让客户逐渐失去信心,最终不得不悲哀出局。虽然貌似有好多问题应该是项目经理考虑的,但是,作为软件设计者,你所设计的每一个操作、每一个功能都会用在管理中,会对管理过程产生影响,每个用户的工作方式都会因之而发生变化,如果不去分析软件给管理、给每个用户带来什么价值,显然是做不好设计的。

当然,有一点要非常清醒,信息系统会给管理方带来价值,给整个组织带来收益,但并不指给每个人都带来收益,对某些岗位、人员来说,可能不仅不会带来收益,反而会带来损害。做价值分析,要搞清楚哪些岗位、哪些人的利益会受到损害。每个组织都有一些既得利益者,没有人愿意放弃既得利益。

3.3.2 规划软件边界

规划软件边界也就是说规划好软件在这个公司做什么,不做什么。无论多么强大的软件系统,都不可能处理管理过程中所发生的所有事情,总有一部分你的软件可以处理,一部分你的软件不可以处理(一般情况下,真正软件能处理的工作只占整个管理工作的一小部分)。例如,准备设计一款库存管理软件用在仓库管理中,该软件可以处理物料出入库,却管理不了仓库的清洁卫生工作。也有些事情,虽然软件系统可以处理,但因为种种原因管理者未必愿意使用你的软件处理。例如,这款库存管理软件,可以对物料所在的仓库位置进行精确管理,但管理方认为仓库并不大,东西存放在什么地方一目了然,并不需要这方面的管理。



案例：规划软件边界

某公司的原材料仓库包括三种岗位——管理员、仓库会计与库工。他们进行如下工作。

- 管理员接收供应商送货,验货。
- 管理员通知检验员进行原材料检验。
- 管理员安排库工卸货。
- 管理员核对送货单,制作原材料验收单。
- 管理员安排库工将货物送到仓库指定位置。
- 仓库会计根据验收单登记仓库材料出入库明细账。
- 管理员接收车间、部门领料单,根据领料单将货物移交给领料人。
- 仓库会计根据领料单登记仓库材料出入库明细账。
- 管理员安排库工进行货物整理,如合并包装、移动位置等。
- 管理员安排库工做好仓库清洁工作。
- 管理员接收车间、部门的退料,存放到相应位置,制作车间退料单。
- 仓库会计根据车间退料单登记仓库材料出入库明细账。
- 管理员发现有质量问题的原料,通知采购部联系供应商退货。
- 管理员制作原材料退货单,供应商取走退回材料。
- 仓库会计根据原材料退货单登记仓库材料出入库明细账。
- 管理员每周统计结存数量低于安全库存的原料,报送给采购部。
- 仓库会计每月制作报表给财务,统计仓库结存成本、车间部门领用成本。
- 财务人员每月进行仓库盘点,如果盘点误差超过一定额度需要追究管理员责任。
- 仓库会计根据盘点结果进行账目调整。
- 仓库所有人员每周开一次例会,总结一周工作的情况。

小王在详细调研后进行了工作分析,觉得这个仓库的工作可以分成4大类:入库、出库、报告、日常管理。供应商送货、验收、检验、登记入库、车间退料等,都属于入库的内容;部门车间领料、退货给供应商、登记出库都属于出库的内容;制作月报给财务、制作采购建议给采购部,属于报告的内容;其他盘点、整理、清洁卫生等属于日常管理的工作。经过分析讨论后,小王跟仓库管理方一起确定了软件的边界,通过软件处理的工作大概包括货物入库登记、打印验收单、材料领用登记、打印领料单、退货登记、打印退货单、车间材料退回登记、打印退料单、盘点表打印、盘点结果登记、采购建议生成、月报生成。有些工作,软件是无法处理的,如清洁卫生、货物整理等;有些工作,虽然软件可以处理,但管理方并不觉得有这个必要,如通知供应商退货、通知质检部检验、货物的位置管理等。

入库部分的业务示意流程如图3-4所示。

图中,矩形表示规划中通过软件完成的事项,其他的事项不通过软件。可以看出,真正通过软件处理的事项只占入库过程中很少的一部分。

还有些事项,是通过别的软件处理的。一家公司一般不可能只用一种软件,很有可能还有其他的软件在使用中,你设计的软件可能需要跟其他软件进行数据交换,这时候也要规划好你的软件与其他软件的边界,这些软件之间如何分工,在什么情况下需要从其他软件中获取数据,什么情况下需要向其他软件推送数据。

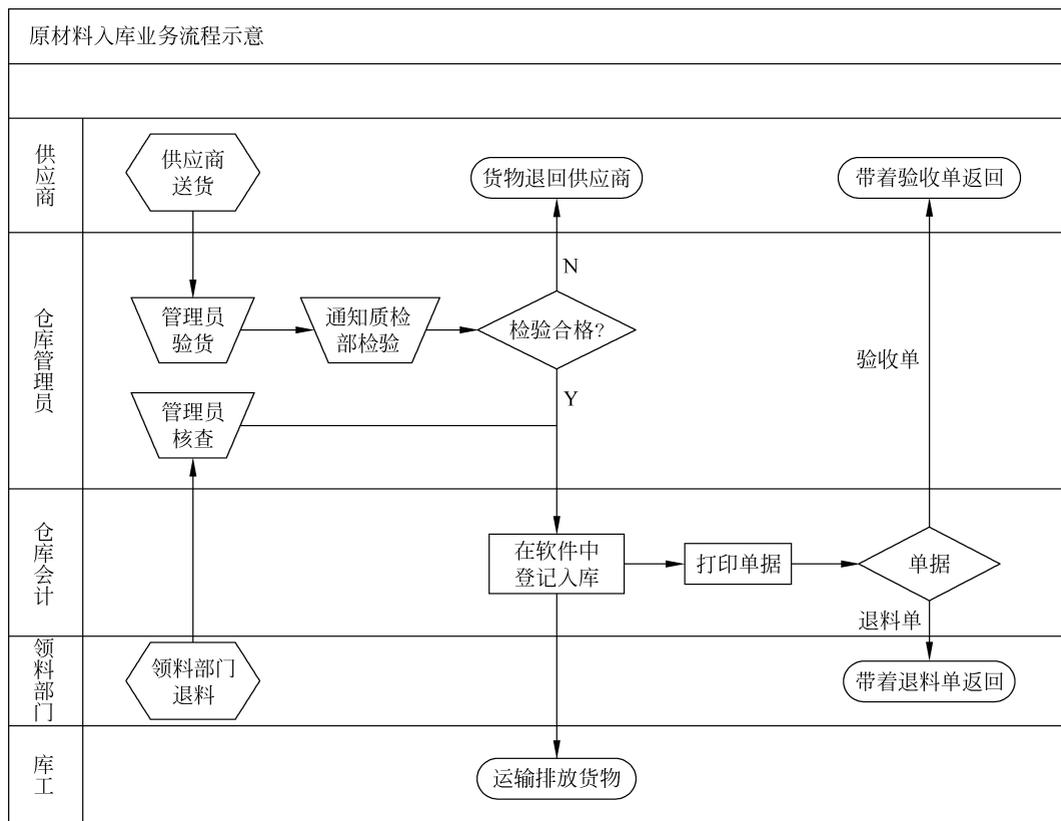


图 3-4 原材料入库业务流程示意



案例：存在多系统的软件边界

某公司使用一款 ERP 软件进行企业管理,现在需要开发 OA 软件。那么,对于 OA 软件的需求分析者来说,需要规划好,什么事项需要在 OA 系统中处理,什么不需要在 OA 系统中处理(或许还有些事项,一部分在 OA 系统中处理,一部分不在 OA 系统中处理);这款 OA 系统跟 ERP 系统的边界在哪里,跟办公相关的事项是不是有些会在 ERP 中处理,有什么跟办公相关的事项不需要经过这两个软件系统,这两个系统需要进行什么数据交互。例如,在 OA 中有一个采购申请审批流程,在审批完成后,采购申请计划是否要推送到 ERP 系统中进行采购计划安排,在 ERP 系统中采购入库时,是否需要将入库消息推送给 OA 系统中的采购申请者。

3.3.3 规划工作方式

规划工作方式也就是规划各岗位人员使用软件后的具体工作过程,这是确定软件边界后更细化的工作。需要规划相关岗位的职员在使用软件之后应该如何工作,围绕软件系统的具体工作步骤是什么,这是个相当重要也相当困难的事情。一个企业的岗位有很多,每个岗位都有大量不同的工作,岗位跟岗位之间的工作相互关联,相互依赖,而每个岗位的工作一般都是不可分割的整体,有其系统性与连续性,要把这些岗位的所有工作都搞清楚,还要

根据信息化系统做好新工作方式的设计,这事想起来就头痛。为了在有限的时间完成这件事情,需要抓住重点,那些跟这个信息化系统无关的岗位就不管了。例如,这次做的是销售管理系统,那么管后勤的岗位估计跟这个系统没有太大关系,先不管,真的遇到时再说。

规划工作方式主要包括规划各个岗位在什么地点使用软件系统,在什么时间使用软件系统,由什么事件触发使用软件系统,使用软件的工作场景等。

1. 使用软件的地点

规划使用软件的地点,就是规划工作人员在什么地方使用软件,大部分情况下可以理解为安装软件的计算机应该摆放在什么地方——当然,由于移动办公的日益普及,这句话有待商榷。这种规划针对不同的岗位有不同的要求,有些岗位,如办公室文员,天天坐在计算机前面,软件系统的使用地点问题根本不是事儿;而有些岗位,如很多车间操作工,软件系统的地点问题就是个天大的问题,处理不好会严重影响工作效率,计算机(或其他终端,如条码扫描枪、POS机等)是摆放在车间办公室中需要时去使用一下,还是摆放在操作台上可以随时使用,还是配置掌上电脑随身携带等,这些都需要根据岗位的工作特点预先做好规划。例如,车间中一些负责收取完工产品的人员,可以在每次收取产品完成后,到车间办公室的计算机中做登记;对于超市收银员,POS机除了摆放在工作台上别无选择;而有些巡查岗位,需要一边巡查一边记录巡查结果,可以考虑使用移动设备(如掌上电脑、无线扫描枪等)随时记录、上传,实现使用地点的自由性。

另外,由于物联网的普及,越来越多的软件系统通过感应设备直接采集数据(这些设备的主要功能是对外界环境或信号进行监测采集,如温度、湿度、气压等),发出控制信息,对这些物联网感应设备的安装地点的规划也是个不容忽视的问题。很多现代化车间都使用了RFID技术进行数据采集:给货物贴上RFID标签,当货物经过某感应器附近时,信号会被自动接收,从而实时记录物流信息。可想而知,在这种情况下,需要仔细规划好货物的移动路线,根据货物移动路线中的关键地点(如车间出入口)设计感应器的安装位置,因为每一次信号的抓取就预示着某种工序的开始或结束,或者预示着某种管理责任的转移等。

2. 使用软件系统的时间

对于使用软件系统时间的规划,不同的功能有不同的要求。有些功能是刚上班用的,如考勤打卡功能;有些功能是下班用的,如每日工作汇报功能;还有些功能是每周、每月、每季度、每年用的,如各种期间报表。当然,除了一些定期自动执行的功能(如每月最后一天的12:00自动生成某种月报),一般情况下,软件功能的使用跟具体时间点并没有直接的关系,这里最重要的关注点是,对软件的运算压力提前做好预案。例如,某学校的一卡通系统,需要处理学生的三种刷卡数据,上课考勤刷卡、用餐刷卡、超市购物刷卡,考虑到软件使用时间的问题,很明显,这三种刷卡方式对软件性能的要求差别很大。上课刷卡,成千上万的学生在上课前几分钟同时刷卡,软件压力很大;而用餐刷卡,持续时间较长,压力相对要小些;在超市购物的刷卡,一般没有这种井喷式的刷卡过程,所以软件压力的问题可以忽视。

3. 使用软件系统的触发事件

规划使用软件系统的触发事件,就是要规划所有岗位的人员,什么事情发生时,需要使用软件来处理,处理不同的事情,需要使用软件的什么功能。例如,仓库中,当供应商送货检验合格时,触发仓库管理员使用库存管理系统中的入库功能;当车间有人领料时,触发仓库

管理员使用库存管理系统中的出库功能。有些触发事件很简单,例如,上班考勤刷卡,员工上班到公司自然就触发他使用刷卡功能;有些触发事件就很复杂,例如,计划人员为了完成一份生产周计划,需要用到软件中大量的功能,当需要知道车间生产情况时,需要用到生产进度分析功能,当需要知道原材料库存情况时,需要用到库存查询功能,当需要知道生产能力情况时,需要用到机器负载分析功能等。



案例：使用软件系统的触发事件

装配车间有个岗位叫配货员,主要工作如下:每过一两个小时到车间巡查一遍,安排装运工将已经完工的产品用手推车运送到成品仓库,并记录送货信息;巡查时如果发现装配线上原材料不足了,就安排装运工将原材料送到装配线,并记录发料信息;每天早晨到原材料仓库、半成品仓库领料,将当天装配线需要的原材料领到车间,存放在车间的材料中转区,有时候有特殊情况,也会偶尔去原材料仓库、半成品仓库补领材料;每周会对材料中转区进行一次盘点,将多领的原材料退回原材料仓库,将多领的半成品退回半成品仓库。

根据配货员的工作特点,使用软件系统的主要触发事件规划如下。

(1) 事件 1: 每天早晨刚上班。

【处理方式】通过系统查询今天的生产任务,分析今天需要领用的材料数量,在系统中填写原材料申领单、半成品申领单。

(2) 事件 2: 发现装配线原材料不足(自己巡查发现,或者装配组长通知)。

【处理方式】通过系统查询该装配线的生产任务,根据后面的加工任务准备原材料,在系统中录入班组材料发放单。

(3) 事件 3: 装配组长通知某种材料多发。

【处理方式】通过系统确认该材料是否确实多发,如果确实多发了,就在系统中录入班组退料单,否则通知技术部门核查。

(4) 事件 4: 发现车间中转区某种原材料或半成品不足。

【处理方式】通过系统查询今天的生产任务,分析今天需要补领的材料数量,在系统中录入原材料申领单、半成品申领单。

(5) 事件 5: 每周对车间材料中转区进行盘点后。

【处理方式】如果发现某种原材料或半成品多领了,就在系统中录入退料申请单;如果少领了,就录入原材料申领单、半成品申领单。

(6) 事件 6: 装运工将完工的成品装上手推车,准备运送到成品仓库之前。

【处理方式】在系统中登记生产任务完成情况,录入成品入库申请单。

(7) 事件 7: 每天下班前。

【处理方式】在系统中生成报表,统计当天每个装配线的生产任务完成与领料情况,发送邮件到车间主任,并抄送每个装配组长。

(8) 事件 8: 装配组长对生产任务统计有疑问。

【处理方式】在系统中查询该装配组的任务完成统计记录,与收货时给装配组的签字记录核对,找到问题所在。

4. 使用软件系统的工作场景

仅规划使用软件系统的触发事件当然是不够的,还要继续规划使用软件系统处理问题

的过程,也就是说使用软件系统的工作场景是怎样的。同样的一件事,原来是怎么处理的,现在该怎么处理,经历哪些步骤;在处理过程中,人需要做什么,系统需要做什么,人跟系统怎么进行信息交互等。有些工作,使用系统处理的场景非常简单。例如,办公室文员使用软件给全公司员工发布一个公告,无非就是打开发布公告功能,录入公告内容,选择接收人员,然后发布。但有些工作,使用系统处理的场景确实非常复杂,需要认真做好规划。例如,成本经理做下一年的成本预算,需要通过软件系统检索生产、库存、成本、财务、计划、技术等各种相关信息,进行各种成本数据计算。对于这种工作毋宁说规划工作场景,还不如说规划能提供哪些信息由用户自己自由发挥——当然,这个例子有些极端,绝大部分的工作,软件需求人员还是应该有能力做好工作场景规划的。



案例：使用软件系统工作的场景

某成品仓库有大量的立体货架,货物都存放在货架上,具体存放的位置并没有进行正式的管理,管理员只是对货架进行了大概的区分,某些货架存放这个系列的产品,某些货架存放那个系列的产品。现在假设,有一批成品入库,全部放置在某一货架上。然而,发货是根据销售合同进行的,一般不会整批同时发货,而是一部分接着一部分不定量、不定期出库的。随着类似的入库、出库的过程不断进行,会出现越来越多的零星的货架空间,这些零星空间增多自然会影响货物的存放。仓库管理员每隔一段时间会进行一次整理,通过移动货物将这些零星空间合并成大空间,好让新的货物可以入库上架。这个整理货架的工作过程是这样的:当管理员发现货架上的零星货物太多后,会将这些零星货物合并到新位置(但需要在相应系列产品应该存放的货架上),由于并没有用账目记录每个货架上的货物存放情况,这个整理过程还是比较随意的。

现在管理方需要开发一套仓库管理系统,其中有明确的关于货架管理的需求,需求人员经过分析与沟通,规划了货架整理的工作场景。

- (1) 仓管员调用软件中的货架分析功能。
- (2) 系统给出货架整理的建议方式(根据货架的大小、货物的长宽高、发货频率等进行计算)。
- (3) 仓管员调整系统整理建议,确认货架整理开始。
- (4) 仓管员移动货物,持掌上电脑扫描条码,先扫描货架条码,再扫描货物包装条码,持续这个过程直到货物移动完成。
- (5) 系统记录货物移动结果。
- (6) 仓管员在计算机端查看刚才货物移动的结果以及货架的货物分布示意图。
- (7) 系统总结这次货架整理的报告,如果这次移动跟货架整理建议不符,系统会发出提示,仓管员可以选择以货物实际移动结果为准,或者重新移动货物。
- (8) 仓管员确认这次货架整理完成。



3.4 几个注意事项

系统规划阶段的工作会直接或间接影响以后几乎所有的工作,一个不小心就会导致无穷无尽的麻烦。这里有几个注意事项,希望读者在工作过程中注意。

3.4.1 警惕利益受损者

根据处理业务规模的大小,程度的深浅,在推行信息化管理的过程中或多或少都要对企业的管理方式、工作流程进行重新设计,这就是所谓的“流程重组”。虽然以定制开发为主的解决方案,对流程重组的要求不如套装产品那么严格,但是要知道,流程重组是躲避不了的。流程重组是一次管理变革,而变革总是会遭到大部分人反对的,可能这个变革未必会损害他们的利益,这些人只是因为见识、习惯、思维等方面的原因反对这种变化,因为改变习惯总是痛苦的;另外,变革总会损害一部分人的利益——可能是经济利益受损,可能是工作负荷增加,可能是工作职权变化,甚至可能有失业的危险。

认清利益受损者,是系统规划阶段的一项重要工作,要对在信息系统建设的过程中来自这部分人的阻力有足够的心理准备,触动利益比触动灵魂还难,虽然这部分人是少数,但他们带来的阻力比单纯反对变革的阻力要大得多。

3.4.2 避免重复劳动

有两类工作最容易影响士气,一是重复劳动,二是返工(或许返工也是重复劳动的一种)。同样一件事,做一遍新鲜,做两遍无聊,做三遍厌恶,做四遍就要崩溃了。重复劳动,会严重影响用户体验,从而给信息系统的推行带来阻碍,因为人并非机器,不可避免地会把情绪带入工作。在进行工作方式规划时,要注意尽量避免给用户带来重复劳动(无论这种重复劳动来自于你的系统还是别的系统),重复劳动可能会导致软件根本无法上线投入使用,或者即使能勉强上线也不会有好结果。轻者天天有人发牢骚,导致你的软件或团队在客户那边声名扫地;重者消极怠工,拒绝使用,导致软件下线,最终不得不宣告项目失败。

对于一般用户来说,使用软件的大部分工作量都来自于数据录入(用户使用软件,无非就是调用功能、输入数据、软件系统加工处理、展现信息这几个过程,后两个过程是计算机完成的,需要耗时很少),因此重复劳动最大的可能性来自于数据的重复录入。如果规划、设计不合理,稍有不慎就会给用户带来重复录入数据的工作。

数据重复录入,可能是某些数据,明明在这个岗位或者这个工作步骤录入了,到另外一个岗位或者另外一个工作步骤,这些数据或者其中的一部分还需要重新录入一遍。



案例：直接重复录入数据

有甲、乙两个车间,在工序上是前后道,即甲车间的产品是乙车间的原料,甲车间的每批产品加工完成后,甲车间的记录员会登记该批成品移出本车间的时间,而乙车间的记录员又会登记接收该批产品的时间,事实上,这两个时间是一样的——这是典型的数据重复录入。

数据重复录入,也有可能不是这么直接。可能在某个岗位或某个工作步骤录入的数据从表面上看不像重复录入,但仔细分析下,发现这些数据跟其他数据有很强的逻辑依赖性,完全可以通过某种运算计算获得,这也算是数据重复录入的一种。



案例：间接重复录入数据

某公司人力资源部使用的人力资源管理系统中,有一个功能“员工基本信息管理”,管理员工的基础信息,包括工号、姓名、身份证、手机、住址、入职日期等。另外还有一个“年假管

理”功能,管理员工每年应休的年假天数,已使用的年假天数,剩余年假天数。应休年假天数由用户在每年年初录入系统,已使用的年假天数从请假功能中抽取,两者的差额为剩余年假天数。

仔细分析下来,发现这里就有个数据重复录入的过程。应休年假天数跟员工在本公司的服务年限相关,计算的规则是,从员工入职那一天算起,到当前年份的1月1日,为员工服务年限,服务年限在1~3年一个级别,3~5年一个级别,等等。根据这个规则,完全可以根据员工的入职日期计算出来员工的应休年假天数,而不需要用户录入。

数据重复录入往往源于三个方面:一是因为没有站在用户的角度规划与设计系统,上面的两个案例都属于这个方面;二是因为软件需求的变更与追加,在软件实施过程中发现缺少某数据,只好追加这种数据录入的功能,虽然这种数据跟用户在其他功能模块中录入的某些数据重复,但由于数据录入与处理的场景、规则不同,对于开发者来说,开发个新的录入功能比从其他功能模块通过一大堆规则抽取数据要简单得多,于是就产生了数据重复录入的问题;三是因为没有处理好软件与软件之间的关系——详见3.4.3节。

3.4.3 处理好软件关系

对于软件系统内部的事情,比较容易控制,因为功能也好,数据也好,都是统一设计的,你自己的软件做什么,不做什么,都可以统一规划。但千万不要忘记一点,大部分情况下,客户使用的软件不止一家,你们只是这个客户的若干软件供应商之一,所有的软件都是这个客户的信息化管理体系中不可或缺的组成部分。要规划你的软件如何跟这些已经存在的软件通力合作,共同为客户信息化管理体系的构建提供支持——这个就有点儿麻烦了,因为有太多的事情超出了控制范围。如何处理好软件之间的关系,是系统规划阶段的难题之一。

有些软件跟你的软件是完全不相干的领域,一般不需要考虑跟它们的关系。例如,你做OA系统,别人做工艺CAD系统,前者主要在于通知公告、沟通交流、办公流程的处理,而后者主要在于产品设计图纸的制作,从业务的角度来看,这两者之间的数据并没有什么关联关系,你在做OA系统时可以无视那个CAD系统。当然,说没有关联关系也是相对的,这两个系统中,至少有某些操作者有可能是相同的,管理者如果要分析某工艺员在这个企业中的综合状况,可能就会用到这种关系了。

有些软件跟你的软件在业务范围上有明显的交叉领域,这就不能不引起警惕,一定要做好这方面的处理预案。例如,你做MES系统,别人做了ERP系统,有些MES系统向外延伸得也像个ERP系统了,有些ERP系统向车间深入得也像个MES系统了,这两者之间的工作交叉的地方就会非常多,如生产单管理、生产任务调度、加工汇报、生产过程监控等,都是非常有可能交叉的地方。这种交叉关系处理不好,很容易产生两大恶果,一是给用户带来重复劳动,二是形成企业的信息孤岛。

业务范围有交叉的这些软件,往往需要输入许多相同的数据,而它们又各自为政,在这种情况下,重复录入数据不可避免。



案例：在不同的系统中重复录入数据

某财务部门使用两款软件,一是财务软件,一是预算软件。这两个软件系统是独立的,某些费用发生时,用户不得不在财务系统中登记一次,在预算系统中再登记一次,前者用于

生成记账凭证,后者用于内部预算管理。

随着企业采购的软件越来越多,这种业务交叉的范围也会越来越宽,给用户带来重复劳动的可能性自然也越来越大,而重复劳动的工作量也会越来越大。笔者曾经遇到过一些客户,有些数据,同一个用户需要在四五个系统中都登记一遍。对于管理者来说,这些软件都各有侧重点,但对某个具体的操作者来说,这些侧重点跟他没有任何关系,他只知道他要在不同的软件中录入相同的数据,一遍又一遍。



案例：在多个系统中重复录入数据

某车间使用软件“生产管理系统”进行生产任务管理,每天记录人员都会把生产完成情况、完工产品质量检测信息录入这个软件系统中。后来,公司为了提高产品质量,推行质量管理,设计了一套新的质量管理体系,又上了一款“质量管理系统”,管理方要求记录员每天在系统中录入质量检测的相关信息,但是由于生产管理系统中的质量信息需要用来计算加工人员的计件工资,也不能不录入。再后来,公司为了提高服务水平,维系好客户关系,又上了一款“客户关系管理系统”,为了让客户可以查看自己订单的加工进度及质量检测结果,管理方要求记录员在每个订单完成后,在这个系统中录入完成日期,以及相关的质量检测结果。虽然这三个软件系统对质量信息的要求并不完全一样,但有些质量基本信息,如检验等级等,录入员不得不在三个系统中重复录入。

企业中的不同软件往往是由不同的软件供应商提供的,每款软件所管理的数据都有它特定的数据结构、编码规范、约束要求等,这就导致不同软件所管理数据的关联关系很难被处理好。例如,这个系统中员工编号是流水号,那个系统中员工编号是工号,从软件的角度,很难建立这两个系统中员工信息之间的关系,因为它们的编码规则完全不一样,软件根本无法识别一个系统中的某个员工跟另外一个系统中的某个员工是不是同一个员工。

可以这么说,一家企业中的所有数据从根本上来讲都是有一定的业务关联关系的,就看从什么角度看这些数据。例如,仓库出库记录跟生产过程中的质量检验记录,貌似是两种独立的数据,但如果从生产单的角度来看,这都是围绕某一生产任务产生的数据,它们之间不但有关系,而且关系很紧密。还有些数据,当前看上去没有关系,但在未来当某事件发生时,这种建立关系的要求就会产生。例如,企业中的库存数据、机器数据、物料工艺数据,这些数据看上去都可能是独立无关的,但一旦要求进行生产计划编排运算,建立这些数据的关联关系就成为必不可少的工作。

在业务上有关联关系的数据,在信息系统中难以建立关联关系,这样就形成了所谓的“信息孤岛”。

3.4.4 避免信息孤岛

1. 什么是信息孤岛

“信息孤岛”几个字用得比较形象,顾名思义,就是信息被分割成许多独立块,块与块之间缺少有效的联系手段,犹如海洋中孤零零的岛屿,这些岛屿之间没有桥梁,没有飞机,甚至船都没有。当然,一般来说,一家企业内部,信息不管如何被分割,总能找到一些方式进行信息块之间的数据沟通,“信息孤岛”几个字略嫌夸张,用“信息割据”几个字恐怕更能反映信息

分割的状况。每一款软件系统所管理的数据都有一套自己的规则,都是一个割据的“信息诸侯”,这些诸侯对外各有自己的边境线,对内各有自己的规章制度,缺少一个中央政府的统一领导,有时候看起来同文同种的样子,但交流起来就是不那么顺利。如果想把这些系统整合成一个系统,那将是个非常痛苦的过程。看看春秋战国时的那些诸侯王国,每个国家使用自己的车轨、自己的度量衡体系、自己的货币,国家跟国家之间的沟通非常麻烦,后来秦国花了几十年,杀了无数人,终于由秦始皇统一了六国,然后统一货币,统一度量衡,书同文车同轨,就使得沟通变得容易了。

信息孤岛包括两种,一种是物理信息孤岛,一种是逻辑信息孤岛。

物理信息孤岛:数据被存放在不同的物理地点,相互之间被完全隔开,除了用 U 盘复制数据外几乎找不到别的通信方式。由于现在网络发达,这种情况很少出现,出现的原因往往都是管理者从安全的角度着想有意识地隔开这些数据。例如,人力资源部上了一款工资管理软件,为了确保工资保密,就在人力资源部办公室部署了硬件环境,服务器就安装在人力资源部经理办公桌边,这个服务器跟公司的局域网完全隔开,只有人力资源部的几个员工才能访问。

逻辑信息孤岛:从物理层面来看,连接没有任何障碍,孤岛的形成纯粹是由数据的产生过程、加工过程、存储格式、数据结构引起的——这种情况占了绝大部分。可以这么说,是不是信息孤岛,跟数据存储的物理位置几乎没有任何关系,处理好了,哪怕是跨国界存放的数据都不是信息孤岛;处理不好,同一磁盘中的数据,同一数据库中的数据,甚至同一表中的数据都有可能成为信息孤岛。

2. 形成信息孤岛的原因

形成逻辑信息孤岛的原因主要有以下几种。

1) 人为因素

由于企业使用不同供应商提供的软件,这些供应商从各自的利益出发,可能采用一些方法控制别人对自己所管理数据的访问,如访问权限控制、数据加密等;或者即使不是有意进行控制,但由于本身平台的特性,数据格式的特性,开发工具的特性等,也可能导致其他人很难访问、处理他们的数据。例如,某工艺 CAD 软件,采用特殊的数据存储格式,没有软件供应商的帮助,其他人很难分析这些数据。

2) 编码差异

有些明明是完全相同的数据,在这个软件系统中采用这种编码方式,在另外一个软件系统中又采用另外一种编码方式,例如前面提到的员工编号的例子,站在人的角度,一眼就可以看出这个系统中的某某跟另外一个系统中的某某是同一员工,可从软件的角度却很难确定。

3) 缺少关联字段

有些明明是业务上有关联的数据,在两个软件系统中就是找不到关联方式,因为可能某些软件系统在设计时就没有考虑这种关联要求。例如,库存管理系统中管理出入库记录,生产单管理系统中管理生产单信息,但库存管理系统中管理的出库记录并没有登记是为哪个生产单而出库的,从业务的角度,材料发给哪个生产单是非常明确的关联要求,但现在缺少这种关联字段。

4) 数据结构差异

有些在业务上相同或相似的数据,在不同的软件系统中采用了完全不同的数据结构。

例如,人力资源管理系统中,用树来表达企业的组织方式,但是在项目管理系统中,却是用图来表达企业的组织方式的。

3. 处理信息孤岛的方式

信息孤岛的存在既影响了企业信息的综合分析,又严重制约着企业信息化管理的发展。处理信息孤岛,一般有以下几种方式。

1) 数据接口

软件系统之间通过接口进行数据沟通,这是最常见的解决信息孤岛问题的方式。一般的过程是,当某事件发生时,数据发送方通过一定的规范格式将数据抛出,接收方根据一定的规则解析数据,转换成符合自己系统要求的数据,存储下来,然后给发送方提示信息。例如,前面提到的某财务部的财务系统与预算系统,当某些费用发生,用户在财务系统登记时,财务系统按照一定的格式将这笔费用数据抛给预算系统,预算系统通过接口程序接收下来,保存到数据库,这样既能够保证两个系统的数据一致,又能够减少用户的重复劳动。各个系统之间本来各干各的,你是你的数据,我是我的数据,因为有了接口,这些孤岛就有了连接通道,这相当于在岛跟岛之间开辟了航线,以后数据可以通过这些航线往返。不过,通过接口交换数据的方式还是有很多麻烦的:①通过接口交换的数据一般只能局限在小范围之内;②一般系统中的数据都是在不断更新的,通过接口很难保证接收方数据的实时性;③扔出去的数据也可能会变化,扔出去简单,但要保证两个系统的数据同步更新却非常麻烦。



案例：通过数据接口保持数据同步

某公司使用着两个信息系统,一是ERP系统,二是OA系统。这两个系统是不同的软件供应商开发的,双方之间是完全独立的,没有任何数据交换。很多用户在工作过程中需要使用两个系统,如销售人员需要在ERP系统中录入销售订单,在OA系统中请假,等等。使用一段时间后,用户就觉得非常麻烦,因为需要在两个系统中切换,来回登录,对于系统管理员来说,来了新员工需要在两个系统中同时建立用户,也是个麻烦事。于是,该公司想做一些整合工作。为了简化这项工作,只做用户同步与单点登录。需求是,在一个系统中建立用户后,在另外一个系统中可以自动生成相同的用户。另外,用户可以在OA系统中单击ERP系统链接,如果当前用户的用户名、密码通过了ERP系统的验证,就可以直接进入ERP系统,而不需要再在ERP系统中输入用户名、密码登录。同理,在ERP系统中,也可以单击OA系统的链接进入OA系统,而不需要在OA系统中再输入一遍用户名与密码。

由于系统用户信息的维护功能在两个系统中是独立的,为了实现上述需求,需要保持两个系统中用户信息的同步。在ERP系统中创建、修改、删除用户时,需要将相关数据通过OA系统的接口推送到OA系统;在OA系统中创建、修改、删除用户时,又需要将相关数据通过ERP系统的接口推送到ERP系统。而在这两个系统中,有很多功能牵涉用户系统信息的变化,如用户修改密码、管理员重置密码、从员工档案自动生成用户等。

这项工作比大部分人的直觉要困难得多。

2) 统一编码

对不同系统中保存的相同业务信息,特别是实体型的信息(如部门、员工、物料、客户、供

应商等),进行统一编码,也是在解决信息孤岛问题时使用得比较多的方式。使用这种方式,虽然各个系统中的数据还是各存各的,该重复录入的还是要重复录入,孤岛之间也没有开辟直接航线,但有了一个最大的好处,就是这些关键数据变得一致了,只要采用一些非常简单的方式就可以进行不同程度的信息整合。例如,可以导出到 Excel 或 Access 之类的软件中,经过简单处理后进行综合分析。这相当于在各个岛之间推行标准化,说相同的语言,吃相同的食品,等等。

3) 整合平台

为了解决信息孤岛的问题,也可以在孤岛之外建立信息整合平台——这其实也是一种软件,不过它存在的目的是整合信息,消除信息孤岛。整合平台最基本的功能就是通过各种手段从其他软件系统中获取数据,给业务上相关的数据建立关联关系,在一个统一平台上处理、展示。对于整合平台来说,最麻烦的事情就是如何从其他软件中获取数据了,如果有别的软件供应商支持,可以采用数据接口的方式,由其他软件系统将需要整合的数据推送过来,或者按照一定的规则从对方系统中直接读取数据。但有时候并不是所有的软件供应商都会配合的,或者他们觉得利益受损,或者有些软件系统根本没人维护,供应商早就不见了,这时候可以在研究对方的数据结构后直接从数据库中提取数据。这相当于在大洋中建立了某种商业中心,所有的孤岛都跟这个商业中心有业务往来,有了这个中心后,所有的孤岛都不再孤单了,如图 3-5 所示。

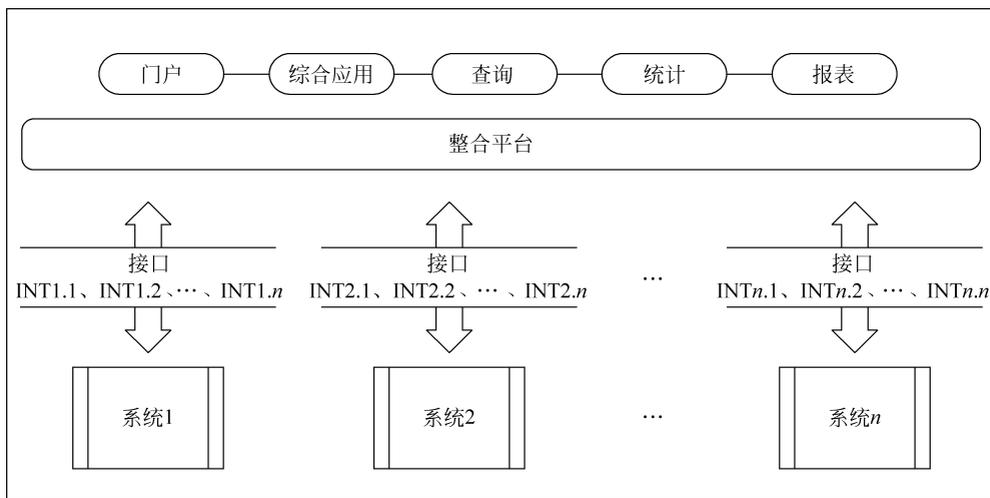


图 3-5 用整合平台处理信息孤岛

4) 综合解决方案

无论采用前面的什么方案,信息孤岛的问题都不能从根本上得到解决,只能治标不能治本,而且,随着业务的发展、管理的变化,这种问题只会越来越严重,最终会严重阻碍企业信息化管理的发展。往往到了这个时候,就会推动管理层开始研究是不是该采用某种综合的信息化解决方案(如常见的 ERP 系统)。所谓综合解决方案,就是推行某种信息化管理系统,用以对全企业的管理信息进行综合管理,对信息化管理体系进行统一策划,信息统一建模,操作统一设计,数据采用统一规范。这种方案将企业的信息化管理系统建成了一个有机体,极大提高了信息系统的运转效率,极大提高了信息系统给管理提升带来的收益。这相当

于将大洋填平,移山填海,不再有孤岛,大洋变成了一马平川。

当然,综合解决方案的前景是美好的,这不容怀疑,道路也是曲折的,这更不容怀疑。要想将综合解决方案建成,并真正投入使用,是一个相当艰巨的任务,需要开发或引入一个庞大的软件系统,需要梳理改变企业的管理流程,需要改变员工的工作方式,需要克服来自各方面的阻力,需要处理以前信息孤岛的遗留问题,需要应对各种异常情况,需要耗时少则几个月,多则几年。由于工程的复杂性,花费了大量的资金与人力而最终失败的案例屡见不鲜。



思考题

1. 学校需要开发一款管理学生档案信息的软件。对于学生基本信息的编辑权限,客户提出了这样的需求:学生的基本信息由班主任录入,如果班主任请假,领导又催得急的话,由学工处王老师处理。请用正确的方式重新描述本需求。

【提示】 描述需求的语言应该是明确的、精练的、没有歧义的。

2. 假设需要开发一款软件用于学校宿舍的床位分配,根据你的想法提出关于床位分配的需求。

【提示】 床位分配可能跟学生的院系、年级、专业、性别等相关,注意可能出现异常情况,如某院系床位不够只能跟其他院系合并。

3. 根据学校图书馆借书、还书的管理要求,画出业务流程图。

【提示】 想想自己空着双手,带着借书证来到图书馆,然后从图书馆拿了几本书出来,读完之后又还回去了,这个过程经过了哪些步骤。

4. 假设你到新华书店去买了两本书。描述一下到收银台结算时的工作场景。

【提示】 主要看这段时间内,你自己、收银员、计算机分别干了啥,经历了哪些步骤。

5. 观察在学习、生活中使用到的一些软件,请举一个信息孤岛的例子,并说明(或猜想)其形成的原因,有什么解决方法。

【提示】 只要仔细观察,你会发现信息孤岛遍地都是,如果两个系统有相同数据,基本上就可以断定这里存在信息孤岛了。



案例分析

1. 某负责垃圾处理的单位,因为所辖区域内偷倒垃圾的现象比较严重,准备在关键的交通路口安装大批高清监控摄像头,智能识别垃圾车,对垃圾车进行跟踪定位,建立一个垃圾运输车智能监管系统。以下是用户用自己的语言提出的关于软件系统的需求,请对这些需求进行分析:哪些是需求,哪些不是需求;哪些是清晰的,哪些是模糊的;哪些是合理的,哪些是不合理的。并使用结构化的语言重新描述用户需求(如果没有确定的需求,写明“待确认”字样)。

我们要一个垃圾运输车智能管理系统,而不仅仅是一个识别系统。

全方位地对垃圾运输车进出本区域进行管理,对垃圾运输车辆进行建档管理。

监控中心,就两个人在看,怎样对几十或上百辆的运输车进行管理。

分析偷倒车辆都是些什么车辆,哪些车辆的偷倒概率较大。

怎样判断偷倒车辆有没有证照,没有证照的车辆怎么识别和管理。

大部分的垃圾运输车是本地区自己固有的运输车,对车辆的编号、车型、电话、使用单位、驾驶员进行管理。

怎么对自己固有的运输车和外来运输车进行区别管理。

对本地区的固有运输车,在确定运输车的起运点和倾倒点的情况下,怎么对运输车的轨迹和它是否偏离固定轨迹进行管理。能否画出运输车的运行轨迹,能否判断车辆是否走了不正确的路。

装和没装垃圾的车辆如何区别。

如果地上发现一堆偷倒的垃圾,能不能将经过该地的垃圾运输车辆都找出来。

需要充分利用原有的摄像头,因预算原因,新装摄像头不能超过 50 个,应该装在偷盗垃圾比较严重的区域。

在地图上,标明所有已建档垃圾车的位置。

做一个 APP,能够运行这套管理系统。

发现可疑垃圾车要在大屏幕、电脑、手机 App 上有报警信息。

对 1600 多家企业的工业垃圾进行环保督查,监控工业垃圾的处理。

2. 某工厂在加工产品时,每一批产品在加工过程中都有一张对应的流转卡跟随产品流动,用于跟踪、记录每道工序的完成情况。假设你在做需求调研时拿到了这张卡(表 3-2),从中看到了哪些管理思路?画出相关业务流程图。为了实现信息化管理,需要设计哪些软件功能?针对这些功能,规划相关的工作方式(使用地点、使用时间、触发事件、工作场景)。

表 3-2 产品加工流转卡

任务单号	TO20199210	计划数量	2500	发卡人	李娜	批号	B1909001					
工艺路线	压铸→切边→去毛刺→砂打→精磨→外观检查→包装											
产品	车间	工序	操作工填写			检验员填写						
			日期	姓名	数量	不合格数量	A	B	C	D	E	F
齿轮箱 (CLX032)	压铸 车间	压铸										
		切边										
		去毛刺										
		砂打										
	精磨											
	成品 车间	外观检查										
包装												

不良分类: A—变形; B—螺纹不良; C—尺寸问题; D—粗糙; E—气孔; F—其他