

## 第3章 审计中间表的创建

计算机审计作为信息化环境下一种崭新的审计方式,其主要流程具体如下:

- 审前调查,获取必要和充分的信息;
- 采集数据,全面掌握情况;
- 数据转换、清理和验证;
- 创建审计中间表;
- 多维分析,把握总体、锁定重点;
- 构建个体模型,内外关联、分析数据;
- 延伸落实,审计取证。

从计算机审计的七步流程可以看出,创建审计中间表是其中不可缺少的关键一步。从数据采集、转换、清理、验证到构建审计分析模型进行数据分析,审计中间表在其中起着承上启下的重要作用。只有在深入了解电子数据的特点和规律的前提下,建立相对完整和规范的审计中间表,才能进一步构建审计分析模型进行有效的数据分析。相对完整和规范的审计中间表是构建审计分析模型的基础。

本章将详细讲解审计中间表的概念、特点、创建规范和具体步骤等内容。

### 3.1 审计中间表概述

#### 3.1.1 审计中间表的概念

##### 1. 审计中间表的定义

审计中间表是面向审计分析的数据存储模式(或称目标模式),它是将转换、清理、验证后的源数据按照提高审计分析效率、实现审计目的的要求进一步选择、整合而形成的数据集合。

审计中间表与审计人员从被审计单位及其相关部门和单位采集得来的源数据相比,最大的区别在于,审计中间表是在源数据的基础上,按照审计目标,为了便于审计分析而进行加工整理形成的,可直接用于审计项目数据分析。审计中间表是审计人员进行数据分析的对象、资源和平台,也就是说,审计人员是在审计中间表而非源数据的基础上构建审计分析模型的。由此可见,审计中间表是构建审计分析模型的基础。

##### 2. 审计中间表的分类

从大类上进行区分,审计中间表可划分为基础性中间表和分析性中间表。

###### (1) 基础性中间表

基础性中间表是审计人员结合被审计单位的业务性质和数据结构,根据不同的分析

主题生成的,它是面向审计项目组全体审计人员的。所有基础性中间表的集合就构成了该审计项目的审计数据库,它包括了所有与审计目的相关且为审计人员进行分析所必需的电子数据。基础性中间表,既涵盖了被审计单位本身的数据,也涵盖了与该被审计单位相关联的数据。如在进出口贸易审计中,基础性中间表不仅包括海关的征税、加工贸易、减免税等数据,而且包括码头、船舶公司、外汇管理、税收、电子口岸等方面的数据。审计人员根据各自的审计分工和分析需求,可以从审计数据库中找到各自所需的基础性中间表。

## (2) 分析性中间表

分析性中间表是审计人员在数据分析过程中,在基础性中间表的基础上根据具体的审计目标和分析需求生成的,它是面向审计项目组中特定审计人员的。一个审计项目有其总体目标,而总体目标是由一个个的具体目标组成的,这些具体的目标也可以称为子目标。对子目标的分析,不是面向全体审计中间表,而是其中一部分数据或某些特定的数据,这就需要在基础性中间表中进行查询筛选,或者在基础性中间表的基础上重新组合、关联,生成用于特定分析目的的审计中间表,这就是分析性中间表。可见,创建分析性中间表是审计人员完成具体审计事项、实现具体审计目标的需要。

此外,创建分析性中间表还有另一个用处,那就是提高数据分析效率。在数据分析过程中,审计分析模型在计算机中运算时间的长短与计算机性能及数据量的大小相关。在计算机性能确定的情况下,计算机运算审计分析模型的时间与数据量的大小呈指数级关系,即当数据量增大时,运算所耗费的时间会成倍增加。因此,审计人员可以根据特定的审计内容,在基础性中间表的基础上进一步生成分析性中间表,然后再进行数据分析和运算,从而缩短计算机运算时间,提高数据分析效率。例如,某基础性中间表中包含了被审计单位某项业务 2010—2014 年的数据,而某位审计人员现在只想对其中 2014 年的数据进行深入分析,如果直接在该基础性中间表上进行分析,计算机运算将耗费大量时间,这时审计人员就可从基础性中间表中抽取 2014 年的数据生成分析性中间表,再在分析性中间表上进行数据分析。

## 3. 基础性中间表与分析性中间表的联系和区别

分析性中间表是在基础性中间表的基础上生成的,它与基础性中间表的本质区别在于:基础性中间表是面向审计项目组全体审计人员的,而分析性中间表是面向具有不同分析目的的不同审计人员的。

根据不同审计人员的不同分析需求,分析性中间表的表现形式与基础性中间表相比,可能存在以下不同:①数据结构与基础性中间表保持一致,只是在数据量上有所差别(即从中抽取部分记录);②数据结构有所改变,需要从基础性中间表中抽取满足特定分析目的所需的部分字段,或将基础性中间表与其他相关数据表进行关联,从中选择相关字段以满足分析的需要。

### 3.1.2 创建审计中间表的原因

创建审计中间表是一件复杂、耗时、费力的基础性工作,但又是计算机审计七步流程中不可或缺的一个重要环节。俗话说“磨刀不误砍柴工”,只有建立起相对完整和规范的

审计中间表,才能进一步构建审计分析模型进行有效的数据分析,促进审计工作效率的提高,同时保证审计成果的质量和水平。

具体而言,创建审计中间表的原因可归纳为以下四点。

#### 1. 解决因范式分解而造成的信息“分裂”问题

被审计单位在开发其信息系统时,按照数据库设计规范,通常会对数据进行范式分解。范式分解这种规范化过程,常常会将反映某项经济业务信息的数据分解成关系数据库中的多个逻辑表,原本在单个纸质表或文件中同时反映的内容,在电子数据中往往被分别存储在不同的数据库或数据表中。通过建立审计中间表,将反映某项经济业务的主要信息集中到一张表中,审计人员就能够更加清晰地把握被审计单位的各项经济业务类别及其主要信息,数据分析工作也就更容易进行。

#### 2. 解决因数据使用目的不同而造成的“垃圾”数据问题

被审计单位的数据库设计和数据表中的内容安排均是依据其核算、管理的目的和要求进行的,是面向事务的。审计人员是面向分析的。面向事务和面向分析是两种截然不同的要求,在数据库设计中会对数据有不同的处理。审计人员所采集的原始数据一般不可能“天生”就适应审计分析的需求,其中会包含大量与审计目的无关的数据(对审计分析来讲,这些无关的数据就变成了“垃圾”数据)。所以,要利用被审计单位数据库中的数据来实现审计分析,必须对清理、转换、验证后的原始数据按审计目的进行“再加工”,从原始数据中选择出所需要的数据,构成适用于审计分析的审计中间表。

#### 3. 解决因数据结构变化而造成的审计分析模型难以复用的问题

不同的被审计单位有不同的经营特点,其使用的信息系统和数据结构通常存在较大差异,即便是同类的被审计单位,尽管其经营业务类似,但表达经济业务信息的数据表和数据结构也往往不尽相同,甚至是同一被审计单位在不同的历史时期,或是同一被审计单位的不同分支机构之间,也可能存在类似的情况。而审计分析模型是在数据的基础上运行的,数据结构一旦发生变化,就需要对其进行适当修改才能保证正常运行。通过创建审计中间表,有效规范数据结构,这样对不同的被审计单位、不同的历史时期,只要经济业务相同,其审计中间表的数据结构也就相同,审计分析模型的可复用性也就大大提高。

#### 4. 解决因利用外部关联数据而带来的信息整合问题

在大数据环境下的审计实践中,越来越多的审计项目需要审计人员在采集被审计单位电子数据的同时,还要尽可能地采集其他相关部门、单位的电子数据,以便在拥有充足信息资源的基础上,充分利用内部数据与外部数据间的关联关系来进行分析验证,确定审计重点和查找问题线索。由于这些可以相互补充、验证的信息分散在不同来源、不同格式、不同类型的数据表或数据库中,要将其成功应用于审计分析,就需要将被审计单位的内部数据与其他部门、单位的外部数据相结合,通过反复的转换、验证、清理工作,从中选择所需要的信息并进行整合,最终构成符合审计人员分析需要的审计中间表。

### 3.1.3 审计中间表的特点

审计中间表具有以下四个特点。

### 1. 体现业务特征

审计中间表的创建是为审计人员下一步开展数据分析工作服务的,也就是说,它是与审计业务密切相关的。在对源数据进行选择、整合形成审计中间表的过程中,被审计单位的业务性质、业务类别是其创建的重要依据,因此,审计中间表体现着很强的业务特征。

### 2. 面向分析主题

审计中间表的创建总是针对具体的审计项目进行的,同样的数据库,如果审计的目的不同,创建的中间表也就不同。在创建的过程中,审计人员应该依据审计方案既定的审计目标,进行数据的选择、整合,创建面向主题,即能满足具体审计项目审计目标要求的数据集合——审计中间表。

### 3. 保持相对稳定

审计中间表是在具体的审计项目实施过程中,对采集到的源数据进行转换、清理、验证,并进一步根据审计目标进行选择、整合等操作后创建的数据集合,通常具有较为固定的结构。这里有两种情况:如果是对被审计单位的历史数据进行处理,中间表不仅具有较为固定的结构,其内容也就是数据的值也是固定的,所以在这样的基础上形成的审计中间表一般是不进行更新的;如果是对被审计单位进行实时审计,需要对被审计单位的数据进行在线分析,在被审计单位实时数据上形成的中间表的结构也是相对稳定的,变化的只是数据的值。

### 4. 随着审计分析的深入而变化

从本质上讲审计中间表是为了审计分析的目的而产生的,不同的审计目的要创建不同的审计中间表;即便是同一个审计目的,分析问题的角度不同,也要创建不同的审计中间表;而且随着分析的深入,可能又需要新的数据,要在原有审计中间表的基础上再创建新的审计中间表。从这些意义上讲,审计中间表的创建,尤其是分析性中间表的创建是一个过程,审计目的没达到,这个过程就不会终结,所以审计中间表要随着审计分析的深入而不断变化,产生新的审计中间表。

## 3.2 审计中间表创建规范

此规范中涉及的操作是以 Microsoft Windows 7 专业版操作系统、Microsoft SQL Server 2012 数据库为操作平台。当然审计数据库也可以建立在以 Oracle、IBM DB2 等大型关系数据库为数据库平台的大型、小型计算机或计算机集群之上,这些内容不在本书讨论范围之内。

### 3.2.1 类别规范

由于审计目的、审计内容及重点的不同,审计人员所关注的审计事项也就会有主次、轻重之分。为了便于开展数据分析、提高分析效率,我们应该从类别上对审计中间表加以规范,将所创建的审计中间表进行类别划分。

具体而言,审计中间表可归类定义为如下几个类别。

### 1. 主表

主表主要是根据被审计单位的业务类别来定义的。如果某个审计中间表内存储了有关被审计单位某项业务类别的主要信息,那么就可以把这个审计中间表归类定义为主表。审计人员从一张主表里基本上就能够了解某项业务类别所包含的主要信息要素。

例如,在商业银行审计中,法人贷款业务是商业银行的主要业务类别之一,我们根据其原始数据整理创建的“法人借款凭证表”中包含了借款人编码、借款日期、借款到期日、贷款类别、借款用途、还款方式、借款金额、本凭证累计已还金额、本凭证贷款余额、贷款四级分类和五级分类等主要要素,通过该表我们可以了解到被审计商业银行每笔法人贷款业务的基本情况,因此可以将此表归类定义为主表。该表对审计人员开展法人贷款业务审计是十分重要的。

### 2. 附表

定义附表,是相对于主表而言的。对于主表反映的业务类别来说,除了在主表中反映的主要信息之外,可能在其他审计中间表中还存储着相关的信息,这些相关信息能够使审计人员更加全面地了解此项业务类别,存储这些相关信息的表就可以归类定义为附表。

沿用前文所述的例子,我们将“法人借款凭证表”定义为主表,该表反映的是每笔法人贷款业务的主要要素信息。除此之外,每笔法人贷款业务还涉及了许多其他相关信息,如借款人的相关信息、担保合同的信息以及担保人的相关信息等,因此,针对法人借款凭证表这张主表而言,我们可以将相关的“法人基本情况表”“担保合同表”“保证人基本情况表”等审计中间表归类定义为附表。

### 3. 代码表

在数据库的设计中,为了减少数据冗余,通常会把一些具有相对稳定性的数据代码化,在数据表中以代码的形式存放,用代码来表示相应的复杂的内容。因此,在数据库中就会定义相关表,来解释说明代码的具体含义,这些表就可以归类定义为代码表。

如“法人借款凭证表”中的客户、信贷机构、贷款类别、贷款性质、专项贷款类别、币种、借款还款方式、还款周期、保证形式、担保方式、贷款四级分类、贷款五级分类等,都采用了适当设计的代码来表示。为便于理解和使用,数据库中相应设计了一些表来解释各个代码的含义,如“信贷机构代码表”“贷款类别代码表”“贷款性质代码表”“专项贷款类别代码表”“币种牌价代码表”“借款还款方式代码表”“还款周期代码表”“保证形式代码表”“担保方式代码表”“贷款四级分类代码表”和“贷款五级分类代码表”。这些表经过整理就构成了审计中间表中的代码表。

### 4. 补充表

在审计数据库中还可能存在这样一种表,它与某一项具体业务类别的联系不是十分紧密,不太适合归类到某一具体业务类别中,在这种情况下,我们就可以将该表定义为补充表,作为审计数据库中的补充信息。

例如,某会计核算系统中的“销账登记簿”记录了会计核算中发生的每一笔错账。显然,该表无法归类到某一具体业务类别中,但其中的数据又与审计工作有关,我们就可将其归类定义为补充表。

## 5. 分析表

上述四类审计中间表主要是针对基础性中间表而言的。在具体审计项目中,审计人员往往需要结合一定的审计思路和审计方法,对某项业务类别进行进一步的深入分析。在此分析过程中,审计人员可能需对基础性中间表进行再加工,如只保留基础性中间表中的部分记录、部分字段或需要与其他相关表再进行关联,从而形成面向部分审计人员、面向特定分析需求的一些分析性中间表。在审计数据库中保存的这些在分析过程中形成的审计中间表,我们就将其归类定义为分析表。

### 3.2.2 格式规范

#### 1. 定义字段

在定义审计中间表的字段时,需要考虑三个要素:字段名称、字段类型和字段长度。

##### (1) 字段名称

字段名称反映了字段存储内容所要表达的具体经济含义或管理、控制信息,因此在定义字段名称时,应使其能够真实、准确地体现该字段存储数据的含义。

##### (2) 字段类型

字段类型指定了字段所要表达内容所使用的表达格式,字段类型的选择应与字段所要表达的内容的格式相一致。

例如,“发生额”字段所表达的内容是指发生某一笔经济业务时,此笔业务所涉及的金额,金额是数量值,是可用于数值计算的,因此我们在定义“发生额”字段类型时,就应该将其定义为货币型字段,而不能将其定义为字符型字段。相反,有些代码字段,如“机构编码”字段的一个值为“10310010123”,在表现形式上是一些数字的组合,但这样的字段并不用于数值计算,它实际上是一些数字字符的组合,通常需要进行一些字符串的截取操作,因此,在定义此类字段时,就应将其定义为字符型字段,而非数值型字段。此外,一些表示日期的字段,有些系统在设计时可能将其定义为字符型,而我们在定义这种字段的类型时,应从审计角度出发,将其定义为日期型,这样便于审计人员进行一些有关日期的函数运算。

##### (3) 字段长度

字段长度是指该字段在计算机中存储时占用的存储空间。

需要注意的是,我们定义字段的长度应大于或等于该字段中值的最大长度。例如,我们定义某字符型字段的长度为30个字符,而在存储数据时该字段某值的实际长度为40个字符,那么在实际存储时,该值的后10个字符就会丢失,不能被正确存储。

如果某字段类型为数值型,我们还需要根据具体情况定义小数点后的位数。在一般情况下,数值型字段定义小数点后2位即可,但有时可能要定义的长一些,比如该字段表示汇率或利率时,小数点后位数保留可能需要超过2位以保持较高的精度。

因此,在定义审计中间表的字段名称、字段类型和字段长度时,最好的做法就是参考被审计单位提供的数据库字典中对应字段的类型和长度来定义,特别是分布在不同表中的含义相同的字段的名称、类型和长度应定义一致。在定义中间表的过程中,如需对原始数据的类型进行转换,事前应根据审计需要周密考虑,科学设计。

2. 定义主键

在创建审计中间表时,我们还需要为每张审计中间表定义该表的主键,以作为每条记录在表中的唯一性标识。定义主键,可以避免在创建审计中间表的过程中由于表与表之间进行连接所造成的存在重复记录的情况。因此,为审计中间表定义主键,也是对审计中间表进行验证的一种方法。

一般情况下,应根据被审计单位提供的数据字典中相关表的主键,来定义我们所创建的审计中间表的主键。

3. 建立关系

数据库关系图是一种可视化的表间关系结构。在为审计中间表定义主键后,还可建立相关表与相关表之间的关系图,这里主要是指建立主表与附表、代码表之间的关系图(如图 3-1 所示)。这样既可以验证数据之间的约束关系,又便于审计人员了解数据之间的关联关系,从而更好地利用数据。

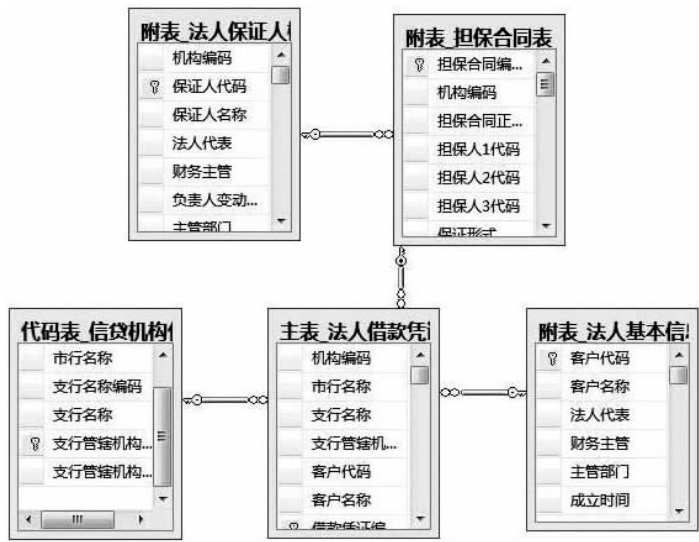


图 3-1 主表与附表、代码表的关系图

3.2.3 名称规范

1. 审计数据库命名

通常被审计单位有若干不同的信息系统,如会计核算系统、业务管理系统等,它们常常被集成于一套软硬件及数据库平台(如 ERP 系统)上,现在的系统大多如此;但较早开发,现在仍在运行的系统,有时则会采取不同的软硬件平台和数据库系统。在第二种情况下,从被审计单位采集不同业务系统的数据时,由于这些数据存储于不同的数据库中,就需要针对不同的业务系统建立若干个审计数据库,通过对审计数据库的命名来区分不同的业务系统。

审计数据库的命名可参照被审计单位业务系统的名称。如在商业银行审计中,某被审计商业银行设有信贷综合管理信息系统、会计核算综合应用系统、人事信息管理系统等

信息系统,则可以相应地以“信贷综合管理信息系统”“会计核算综合应用系统”“人事信息管理系统”来对要建立的审计数据库进行命名。这样命名的优点在于:一方面,审计数据库的使用者明白自己使用的审计数据的来源;另一方面,通过业务系统的名称基本上可以判断审计数据库中存储数据的基本内容。

## 2. 审计中间表命名

通常从被审计单位采集的原始数据表的数量比较多,特别是在商业银行、大型企业集团等大型审计项目中,采集到的数据表往往达到上百张,甚至更多。在整理形成审计中间表后,审计中间表的数量相应也较多。虽然在数据库 SQL Server 2012 对象资源管理器的数据库表对象列表中,数据表的排序方式是按照表名的字母顺序排列的,但要一下子从众多的数据表中找到审计分析所需要的表并非易事,往往需要从头到尾将大部分的表名浏览一遍,才能找到所需要的数据表,可谓既费时又费力。因此,我们在创建审计中间表时需要建立一定的命名规则,对审计中间表的名称进行一定的规范,这样既方便审计人员从数据库查询分析器的对象浏览器的列表中快速查找到所需数据表,又方便审计人员理解数据表储存数据的内容。

为了能够全面地反映一张表的属性,在命名审计中间表时需要考虑以下因素。

### (1) 表的类别

审计中间表的命名应该体现该表所属的类别,如主表、附表、代码表、补充表或分析表。

### (2) 业务类别或业务品种

由于审计中间表是体现业务特征、面向分析主题的,对其命名还应体现该表存储信息所反映的业务类别。这样在对某项业务类别进行分析时,就能从众多的审计中间表中快速定位到针对该业务类别创建的审计中间表。

此外,对于某些业务类别而言,其中还包含多种具体的业务品种,如储蓄存款业务类别中包括活期储蓄存款、零存整取储蓄存款、整存整取储蓄存款、定活两便储蓄存款等业务品种,在命名与此类业务相关的审计中间表时,还需要体现该表所反映的业务品种。

### (3) 表的具体内容

审计中间表的命名还应该体现该表所存储的各业务类别或业务品种的具体内容,如“法人借款凭证表”存储的就是法人贷款业务类别的每笔借款凭证的主要信息,而“零存整取储蓄存款账户信息表”“整存整取储蓄存款账户信息表”这样的表名,则说明这两张表存储的是储蓄存款业务类别中的零存整取和整存整取这两个业务品种的账户的基本信息。

### (4) 数据所属时点或时期

由于在被审计单位的电子数据中,有些是属于时点或时期状态的数据,也就是说,随着时间的变化,数据表中的内容是在不断变化的,而我们采集到的数据只能反映此类数据在某一时刻或某一时期的状态。因此,在命名存储此类数据的审计中间表时,表名中应考虑包含所采集数据的时点或时期。例如,可用“年月日”的形式来表示时点数据,用“起始年月日—结束年月日”的形式来表示时期数据。

### (5) 表的创建者

通常,在创建基础性中间表时,我们利用数据库管理员的身份登录,其拥有者就缺省定义为 DBO(Database Owner)。审计人员进行具体数据分析时,通过个人账号登录服

务器,其账号对审计数据库中的这些基础性中间表只拥有查询权限。与此同时,个人账号还拥有创建新表(指分析性中间表)、视图或存储过程的权限。审计人员创建的表、视图或存储过程的拥有者被定义为用户登录服务器的账号。因此,在审计数据库的过程中,可以通过对象浏览器中审计中间表的拥有者来区分此表的创建者。

总结上述审计中间表的命名规则,我们可按照如下格式来命名审计中间表:

表类别\_业务类别[业务品种]\_表内容[\_时点(或时期)]

[ ]中的内容根据具体情况可以省略。

例如,存储时点为2014年12月31日的对公活期存款账户基本信息的审计中间表可命名为:附表\_对公活期存款\_账户信息表\_20141231,存储2014年1月1日到2014年12月31日对公活期存款账户流水信息的审计中间表可命名为:主表\_对公活期存款\_账户明细表\_20140101\_20141231,而某位审计人员创建的反映某企业贴现情况的分析性中间表则可命名为:分析表\_法人贷款\_某企业贴现情况。

### 3.2.4 存储规范

#### 1. 数据库所在硬盘分区的选择

为了保证数据库在计算机进行重新安装操作系统等操作时不受影响,通常情况下,不要将数据库建立在操作系统所在的硬盘分区上,而是应将其建立在操作系统所在硬盘分区之外的分区上。这样可以避免在重新安装操作系统时,将操作系统所在硬盘分区格式化,而又忘记将数据库备份,误将数据库清除,造成数据丢失。

因此,在建立数据库时,应注意数据库所在硬盘分区的选择。具体操作如下。

(1) 利用数据库管理员身份打开 Microsoft SQL Server Management Studio,在对象资源管理器的“数据库”节点上单击鼠标右键,在弹出的菜单中选择“新建数据库(N)...”命令(如图3-2所示),弹出如图3-3所示的“新建数据库”窗口。



图 3-2 新建数据库

(2) 为新建的数据库命名,在新建数据库窗口中的“数据库名称(N):”文本框内输入要建立数据库的名称,假设输入“演示数据库”,则在下方“数据库文件(F):”列表中,数据库主文件和日志文件的逻辑名称自动被填入“演示数据库”和“演示数据库\_log”,如图3-3

所示。



图 3-3 为新建数据库命名

(3) 拖动图 3-3 中“数据库文件(F)”列表下方滚动条,出现“路径”栏,鼠标左键单击所显示路径后面的 [...] 按钮,弹出“定位文件夹”窗口,如图 3-4 所示,然后可以根据需要选择数据库主文件和日志文件的存放路径。

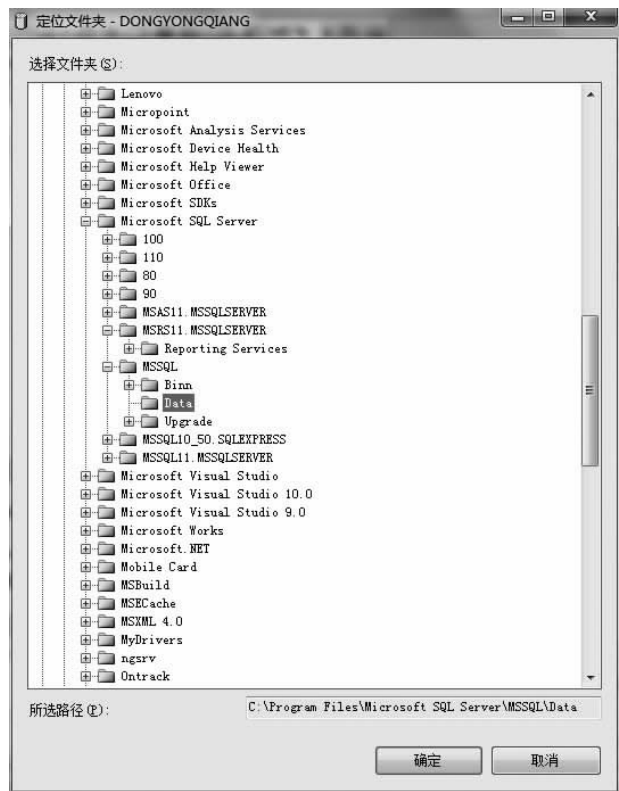


图 3-4 指定数据库文件存储路径