

HJ

中华人民共和国国家环境保护标准

HJ 719-2014

环境信息系统数据库访问接口规范

Specification for database access interface of environment information system

(发布稿)

本电子版为发布稿。请以中国环境科学出版社出版的正式标准文本为准。

2014-12-25发布

2015-03-01实施

环 境 保 护 部 发 布

目 次

前 言.....	II
1 适用范围.....	1
2 术语和定义.....	1
3 数据库访问接口框架.....	1
4 数据库访问接口方式.....	2
5 环境信息系统数据库访问接口推荐访问方式.....	6
6 数据库访问条件.....	7
附录 A （资料性附录） 触发器方式数据库接口的 SQL 脚本（MICROSOFT SQL SERVER） ..	8
附录 B （资料性附录） 触发器方式数据库接口的 SQL 脚本（ORACLE）	13
附录 C （资料性附录） 时间戳方式数据库接口的 SQL 脚本（MICROSOFT SQL SERVER） .	17
附录 D （资料性附录） 时间戳方式数据库接口的 SQL 脚本（ORACLE）	18
附录 E （资料性附录） 环境信息系统数据库访问示例.....	19

前 言

为贯彻《中华人民共和国环境保护法》，促进环境信息化工作，明确关系型数据库的访问接口，推动环境信息资源建设，制定本标准。

本标准规定了环境信息系统中关系型数据库的访问接口，确定了数据库访问接口的基本架构以及数据库访问接口方式、数据库访问工作流程。

本标准附录 A、附录 B、附录 C、附录 D、附录 E 为资料性附录。

本标准为首次发布。

本标准由环境保护部科技标准司组织制订。

本标准主要起草单位：环境保护部信息中心、深圳市环境保护宣传教育信息中心。

本标准由环境保护部 2014 年 12 月 25 日批准。

本标准自 2015 年 3 月 1 日起实施。

本标准由环境保护部解释。

环境信息系统数据库访问接口规范

1 适用范围

本标准规定了环境信息系统中关系型数据库的访问接口，确定了数据库访问接口的基本架构以及数据库访问接口方式、数据库访问工作流程。

本标准适用于同一局域网内的数据库访问，以及环境信息系统的关系型数据库访问接口的建设。

2 术语和定义

下列术语和定义适用于本标准。

2.1 环境信息 environment information

环境管理、环境科学、环境技术、环境保护产业等与环境保护相关的数据、指令和信号等，以及其相关动态变化信息；包括文字、数字、符号、图形、图像、影像和声音等各种表达形式。

2.2 数据库 database

按照概念结构组织的数据的汇集，它描述这些数据的特征及与数据对应的实体间的关系，并支持一个或多个应用领域。

2.3 数据库适配器 database adapter

按指定要求和标准规范进行数据库访问的功能组件，通过对配置文件的解析和处理，完成数据库访问，实现数据采集和交换。

2.4 数据库访问接口 database access interface

嵌入各环保业务数据库中、专门针对外部应用访问所做的配置。数据库访问接口与适配器相适应，能够实现外部应用对环保业务数据库内容的访问。

2.5 存储过程 stored procedure

一组为了完成特定功能的 SQL (Structured Query Language) 语句集合，经编译后存储在数据库中。

2.6 触发器 trigger

特殊的存储过程，它的执行只由事件触发，如对一个数据库表进行插入、删除、更新操作 (insert, delete, update) 时激活它执行。

2.7 时间戳 timestamp

指明时间上相对于通用时标、精确到秒级的某一时间点的数据项。

3 数据库访问接口框架

在环境信息系统数据管理中，数据库访问接口用于获取同域内的各个业务应用的数据库内容。对各个业务系统提供的数据库访问接口统一规定，便于其他业务应用系统出于数据传输或数据集成的目的进行数据库访问。

环境信息系统数据库访问接口框架见图 1。

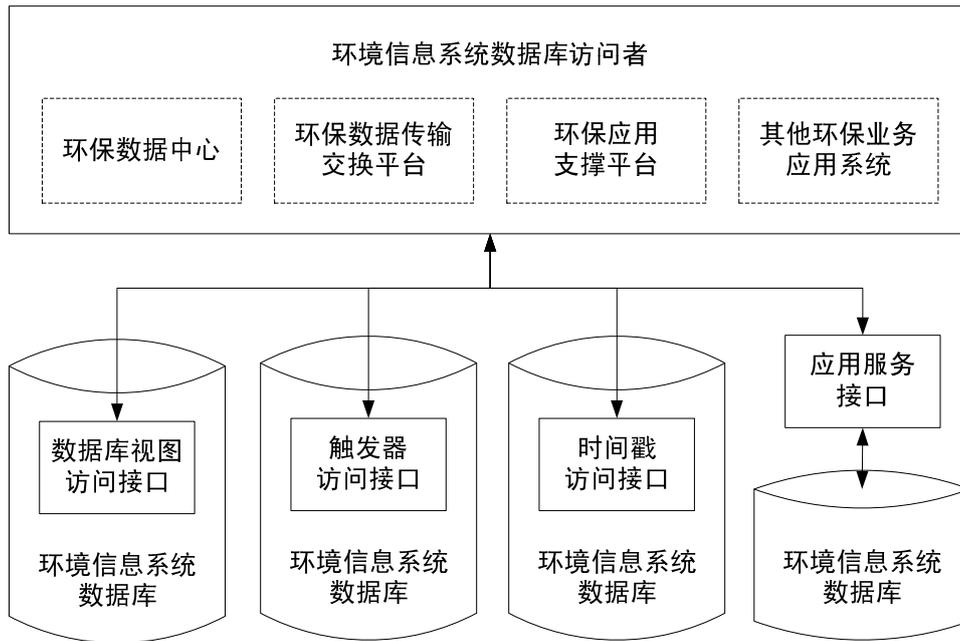


图 1 环境信息系统数据库访问接口框架

环境信息系统数据库访问者通过数据库访问接口访问不同的环境信息系统数据库。数据库访问者与嵌入在环境信息系统数据库中的数据库访问接口进行交互，获取环境业务应用数据库的内容。

4 数据库访问接口方式

4.1 数据库视图方式

在实际业务中，对环境信息系统数据库进行访问，需定制数据与其访问需求有关的数据内容，同时须保证被访问的原始业务数据的安全性。

数据库视图是从一个或几个基本表中根据用户需要而制作的虚拟表，能够为数据提供一定的逻辑独立性，其数据内容可以随着基表的更新而更新。同时，用户对视图不能随意更改和删除，可以保证数据的安全性。

故对环境信息系统的数据库进行全库访问时，可采用数据库视图的访问接口方式。

4.1.1 接口方式描述

根据访问者的访问需要，在环境信息系统数据库中分别创建数据库视图，并对视图赋予相应访问权限。

4.1.2 接口配置方式

使用环境信息系统数据库标准的创建视图语句。

4.1.3 工作流程

数据库视图工作方式见图 2。

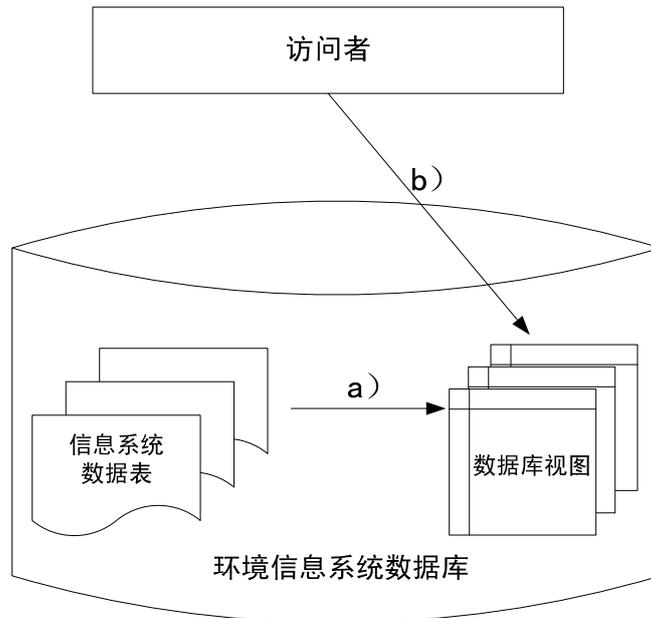


图 2 数据库视图工作方式

- a) 将环境信息系统数据表的数据内容根据访问者需要，通过数据库视图创建命令创建相应的数据库视图；
- b) 访问者根据需要访问数据库视图获取所需要的环境信息系统数据。

4.2 触发器方式

4.2.1 接口方式描述

触发器方式是指在源数据库的业务应用数据表中添加触发器，监控数据表的新增、修改、删除操作。当操作成功时，触发相应的触发器将该记录的主键等信息插入到指定的访问接口中间表，然后数据库访问者从访问接口中间表中获得业务数据内容的变更情况，从而达到实时的数据库增量访问效果。

4.2.2 接口配置方式

- a) 在被访问的环境信息系统数据库中建立一张访问接口中间表，用于存放相关触发器和临时交换记录。不同的环境信息系统临时交换记录都采用统一的中间表格式，具体包括：
 - T_Oth_Msg
- b) 创建一个对访问接口中间表做操作的存储过程，用于执行相关的触发操作，具体包括：
 - P_QueueMsg
- c) 针对要实时访问的数据表建立新增和修改、删除的触发器，用于触发操作，具体包括：
 - TR_AddTrigger_I
 - TR_UpdateTrigger_U
 - TR_DeleteTrigger_D

由于触发器方式只是对业务应用数据表进行操作监控，且只对访问接口中间表进行操作，因此不会对被监控的业务应用数据表造成任何影响。

中间表的格式内容和存储过程参见附录 A、附录 B。

4.2.3 工作流程

环境信息系统数据库的实时访问是指当业务应用数据内容发生变化（包括新增、更新、删除等操作）时，实时地触发数据库访问者，对业务应用数据库内容进行访问，实时获取业务数据的变化情况。

环境信息系统数据库的实时访问采用触发器方式的数据库访问接口，具体流程见图 3。

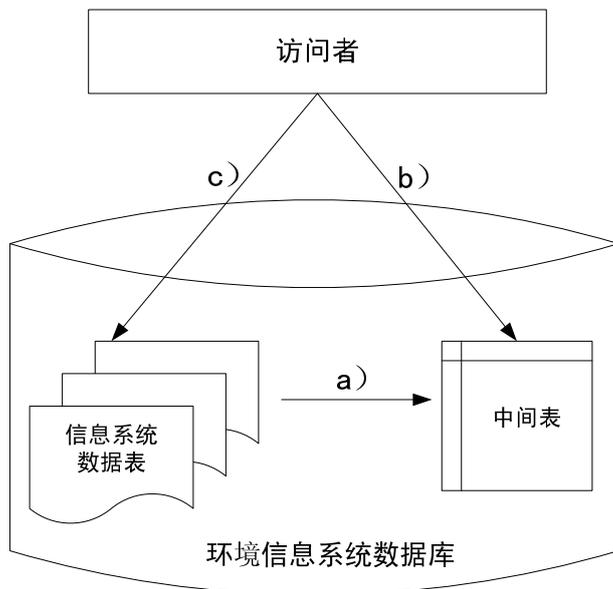


图 3 数据库触发器访问流程

- 信息系统数据表的内容在执行相关操作的时候，通过触发器将受影响的记录直接推送到数据库访问接口的中间表中。如果受影响记录的字段数未超过中间表的字段数，则将受影响记录的所有信息推送；
- 数据库访问者从访问接口中间表中获得业务数据内容的变更情况；
- 当受影响的记录字段数超过中间表的字段数时，仅将受影响记录的主键等唯一标识信息推送到中间表。数据库访问者也可以根据中间表中的数据内容，直接访问业务应用数据表，获得业务应用数据内容。

4.3 时间戳方式

4.3.1 接口方式描述

时间戳方式是指在数据表中配置时间字段，定时轮询数据库是否发生新增或更新的操作，以达到数据增量同步的作用。采用时间戳方式的数据库访问者将上次执行的时间与当前数据表中的时间字段进行比较，甄别在上一时间周期内发生了新增或更新操作的数据库记录，从而达到访问数据的目的。

4.3.2 接口配置方式

- 在被访问的环境信息系统数据表中，创建用于比较的时间戳字段：Create_Time、Update_Time；
- 每个被访问的环境信息系统数据表都必须有两个用于轮询的时间字段作为时间戳。一个用于记录创建时间，另一个用于记录修改时间：Create_Time、Update_Time。

时间戳字段的数据类型为业务应用数据库的时间类型，格式必须为：年月日 时分秒（例如：2009-8-20 12: 00: 00）。

每次查询数据表时，对表中所有的数据按时间进行遍历查询，取出最新的记录。这样，如果源数据库中数据过多，进行定时轮询时就会对交换的性能造成影响。

填入时间戳的方式有多种，如通过数据库结构化查询语句（Structure Query Language）填入值、触发器方式自动填入值等。建议采取建立触发器方式，自动获取当前操作时间并填入到对应的字段中。

时间戳触发器编写方式参见附录 C、附录 D。

4.3.3 工作流程

环境信息系统数据库的定时访问是指数据库访问者按照预定的时间周期，定时访问环境信息系统数据库，并根据系统数据表的时间戳，获取上一时间周期里的环境信息系统数据库所发生的变化内容（包括新增、更新、删除等操作）。

环境信息系统数据库的定时访问通常采用时间戳方式的数据库访问接口。具体流程见图 4。

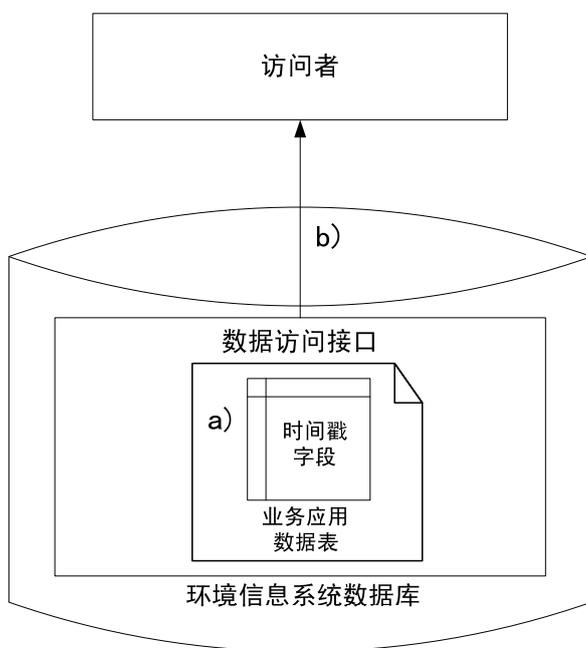


图 4 数据库时间戳定时访问流程

- a) 业务应用数据表在新增、更新的同时，更新时间戳字段的内容；
- b) 数据库访问者定时访问业务应用数据表，根据时间戳字段所记录的新增和更新时间戳，获取上个时间周期里的业务应用数据库的变化内容。

4.4 应用服务接口方式

专用于应用层面数据库访问，建议采用在线应用服务（webservice）接口方式。

4.4.1 接口方式描述

通过在线应用服务接口对环境信息数据库中数据进行访问，从而获取所需要的信息。

4.4.2 接口配置方式

被访问的环境信息系统根据访问者需要，专门开发一套只用于当前环境信息系统访问的在线应用服务软件。

开发的应用服务软件安装在访问者可访问到的运行环境中。

4.4.3 工作流程

应用服务接口方式见图 5。

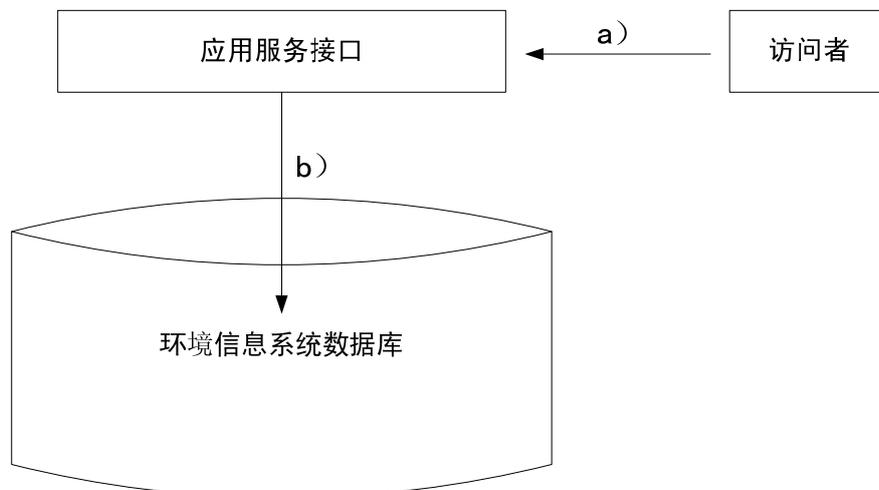


图 5 应用服务接口工作方式

- a) 访问者在访问环境信息系统数据库中数据时，调用在线应用服务接口；
- b) 应用服务接口根据应用者要求，访问信息系统数据，并将访问结果反馈给访问者。

5 环境信息系统数据库访问接口推荐访问方式

环境信息系统数据库访问按照访问的环境信息数据内容可分为全部数据访问和增量数据访问。全部环境信息数据访问建议采用数据库视图的接口方式，增量环境信息数据访问建议采用触发器和时间戳的接口方式。

数据库访问方式参见附录 E。

在应用层面数据库访问方式中，建议采用应用服务接口方式。

不同的环境信息系统采取不同的数据库体系，本规范针对部分主要环境信息系统关系型数据库访问方式进行说明。具体推荐方案见表 1。

表 1 环境信息系统推荐数据库访问方式

序号	类型	环境信息系统		推荐访问方式
1	自动监控类	河流水质自动监测	实时数据	触发器
			历史数据	时间戳
		空气质量自动监测	实时数据	触发器
			历史数据	时间戳
		饮用水源自动监测	实时数据	触发器
			历史数据	时间戳
		污染源在线监控	实时数据	触发器
			历史数据	时间戳
		机动车尾气自动检测	实时数据	触发器
			历史数据	时间戳
		辐射环境监控	实时数据	触发器
			历史数据	时间戳
		危险废弃物在线监控	实时数据	触发器
			历史数据	时间戳

		其他	
2	业务管理类	建设项目管理	应用服务接口
		排污许可证管理	时间戳
		排污收费管理	时间戳
		环保移动执法管理	时间戳
		环境信访投诉管理	应用服务接口
		行政处罚管理	应用服务接口
		公文文件管理	触发器
		其他	
3	统计考核类	环境统计	数据库视图
		全国城市环境综合整治定量考核	数据库视图
		污染源普查	数据库视图
		其他	

6 数据库访问条件

对某一数据库进行访问时，在网络连通的情况下，需具备以下条件：

- a) 获取数据库管理系统（DBMS）类型；
- b) 获取被访问的数据库所在服务器的网络地址；
- c) 获取被访问的数据库的实例名；
- d) 获取数据库的用户名及口令。

附录 A
(资料性附录)

触发器方式数据库接口的 SQL 脚本 (Microsoft SQL Server)

A.1 触发器方式数据库接口的中间表创建脚本

```
CREATE TABLE [dbo].[T_Oth_Msg](
    [PK_MsgId] [numeric](28, 0) IDENTITY(1,1) NOT NULL,
    [MsgQueuedAt] [datetime] NOT NULL,
    [MsgStatus] [varchar](10) COLLATE Chinese_PRC_CI_AS NOT NULL,
    [MsgDeliveredAt] [datetime] NULL,
    [MsgFailedText] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [OperationType] [varchar](10) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam1] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam2] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam3] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam4] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam5] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam6] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam7] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam8] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam9] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
    [MsgParam10] [varchar](255) COLLATE Chinese_PRC_CI_AS NULL,
) ON [PRIMARY]
```

GO

表字段说明见表 A.1。

表 A.1 字段说明表

字段名	字段中文名	应用意义
PK_MsgId	临时记录 id	中间表的主键
MsgQueuedAt	临时记录推送的时间	临时记录从中间表中推送的时间
MsgStatus	临时记录的状态	标识临时记录是否为待推送的信息
MsgDeliveredAt	临时记录推送到目的地的时间	临时记录推送到目的地的时间
MsgFailedText	临时记录推送失败的提示信息	临时记录推送失败的详细信息
OperationType	临时记录的操作类型	临时记录在业务数据表中执行的操作（新增、更新、删除）
MsgParam1	临时记录信息	临时记录的具体信息，来源于存储过程中

		的参数值
MsgParam2	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam3	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam4	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam5	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam6	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam7	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam8	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam9	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam10	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值

A.2 触发器方式数据库接口的存储过程脚本

```

create proc [dbo].[P_Trigger_业务应用数据表名]
    @OperationType    varchar(10),
    @MsgParam1        varchar(255) = NULL,
    @MsgParam2        varchar(255) = NULL,
    @MsgParam3        varchar(255) = NULL,
    @MsgParam4        varchar(255) = NULL,
    @MsgParam5        varchar(255) = NULL,
    @MsgParam6        varchar(255) = NULL,
    @MsgParam7        varchar(255) = NULL,
    @MsgParam8        varchar(255) = NULL,
    @MsgParam9        varchar(255) = NULL,
    @MsgParam10       varchar(255) = NULL
as
BEGIN
    declare @ErrorCode int, @ErrorMessage varchar(255), @NumRows int
    select @ErrorCode = 0
    insert    T_Oth_Msg    (    MsgQueuedAt,    MsgStatus,    MsgDeliveredAt,

```

```

MsgFailedText,OperationType, MsgParam1, MsgParam2, MsgParam3, MsgParam4, MsgParam5,
MsgParam6, MsgParam7, MsgParam8, MsgParam9, MsgParam10 ) select getdate(), 'NEW', NULL,
NULL,@OperationType, @MsgParam1, @MsgParam2, @MsgParam3, @MsgParam4,
@MsgParam5, @MsgParam6, @MsgParam7, @MsgParam8, @MsgParam9, @MsgParam10
select @ErrorCode = @@error, @NumRows = @@rowcount return @ErrorCode
END

```

GO

存储过程参数设置说明见表 A.2。

表 A.2 存储过程参数表

参数名	参数含义
MsgQueuedAt	临时记录从中间表中推送的时间
MsgStatus	标识临时记录是否为待推送的信息
MsgDeliveredAt	临时记录推送到目的地的时间
MsgFailedText	临时记录推送失败的详细信息
OperationType	临时记录在业务数据表中执行的操作（新增、更新、删除）
MsgParam1	业务数据表中受影响记录的具体信息
MsgParam2	业务数据表中受影响记录的具体信息
MsgParam3	业务数据表中受影响记录的具体信息
MsgParam4	业务数据表中受影响记录的具体信息
MsgParam5	业务数据表中受影响记录的具体信息
MsgParam6	业务数据表中受影响记录的具体信息
MsgParam7	业务数据表中受影响记录的具体信息
MsgParam8	业务数据表中受影响记录的具体信息
MsgParam9	业务数据表中受影响记录的具体信息
MsgParam10	业务数据表中受影响记录的具体信息

A.3 触发器方式数据库接口的触发器脚本

```

CREATE TRIGGER TR_业务应用数据表名_I ON [dbo].[业务应用数据表名] FOR insert AS
BEGIN
declare [表字段 1, 表字段 2,表字段 3...]
declare cursor_insert cursor for select "表主键" from inserted
open cursor_insert
fetch cursor_insert into @表字段 1, @表字段 2, @表字段 3 ...
while @@fetch_status=0
BEGIN
exec P_Trigger_业务应用数据表名@OperationType='insert', @MsgParam1= @表字段

```

```

1, @MsgParam2= @表字段 2, @MsgParam3= @表字段 3 ...
    fetch cursor_insert into @表字段 1, @表字段 2, @表字段 3 ...
END
close cursor_insert
deallocate cursor_insert
END

GO

CREATE TRIGGER TR_业务应用数据表名_U ON [dbo].[表名] FOR update AS
BEGIN
    declare [表字段 1, 表字段 2,表字段 3...]
    declare cursor_insert cursor for select "表主键" from inserted
    open cursor_insert
    fetch cursor_insert into @表字段 1, @表字段 2, @表字段 3 ...
    while @@fetch_status=0
    BEGIN
        exec P_Trigger_业务应用数据表名 @OperationType ='update' , @MsgParam1= @表字
段 1, @MsgParam2= @表字段 2, @MsgParam3= @表字段 3 ...
        fetch cursor_insert into @表字段 1, @表字段 2, @表字段 3 ...
    END
    close cursor_insert
    deallocate cursor_insert
END

GO

CREATE TRIGGER TR_业务应用数据表名_D ON [dbo].[表名] FOR delete AS
BEGIN
    declare [表字段 1,表字段 2,表字段 3...]
    declare cursor_insert cursor for select "表主键" from deleted
    open cursor_insert
    fetch cursor_insert into @表字段 1, @表字段 2, @表字段 3 ...
    while @@fetch_status=0
    BEGIN
        exec P_Trigger_业务应用数据表名 @OperationTypeName = 'delete', @MsgParam1= @表字
段 1, @MsgParam2= @表字段 2, @MsgParam3= @表字段 3 ...
        fetch cursor_insert into @表字段 1, @表字段 2, @表字段 3 ...
    END

```

END

close cursor_insert

deallocate cursor_insert

END

触发器参数设置说明：

触发器中使用到的参数与存储过程、中间表中使用的字段含义一致。

触发器方式的数据库访问接口由中间表、存储过程、触发器之间相互配合完成，具体流程为：

由触发器对业务数据表进行实时监控。当业务数据表有变化时，对应的触发器对发生的变化进行捕获，并调用相应的存储过程，将发生的变化信息存储到中间表中。

附录 B
(资料性附录)

触发器方式数据库接口的 SQL 脚本 (Oracle)

B.1 触发器方式数据库接口的中间表创建脚本

```
drop table "T_Oth_Msg" cascade constraints;
create table "T_Oth_Msg" (
    "PK_MsgId"          bigint not null,
    "MsgQueuedAt"       datetime,
    "MsgStatus"         varchar(10),
    "MsgDeliveredAt"   datetime NULL,
    "MsgFailedText"    varchar(255) NULL,
    "OperationType"    varchar(10) NULL,
    "MsgParam1"        varchar(255) NULL,
    "MsgParam2"        varchar(255) NULL,
    "MsgParam3"        varchar(255) NULL,
    "MsgParam4"        varchar(255) NULL,
    "MsgParam5"        varchar(255) NULL,
    "MsgParam6"        varchar(255) NULL,
    "MsgParam7"        varchar(255) NULL,
    "MsgParam8"        varchar(255) NULL,
    "MsgParam9"        varchar(255) NULL,
    "MsgParam10"       varchar(255) NULL,
    constraint PK_T_OTH_MSG primary key ("PK_MsgId")
);
create sequence seq_t_oth_msg minvalue 1000 maxvalue 99999999 start with 1000 increment
by 1 nocache;
```

表字段说明见表 B.1。

表 B.1 字段说明表

字段名	字段中文名	应用意义
PK_MsgId	临时记录 id	中间表的主键
MsgQueuedAt	临时记录推送的时间	临时记录从中间表中推送的时间
MsgStatus	临时记录的状态	标识临时记录是否为待推送的信息
MsgDeliveredAt	临时记录推送到目的地的时间	临时记录推送到目的地的时间
MsgFailedText	临时记录推送失败的提示信息	临时记录推送失败的详细信息
OperationType	临时记录的操作类型	临时记录在业务数据表中执行的操作 (新

		增、更新、删除)
MsgParam1	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam2	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam3	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam4	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam5	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam6	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam7	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam8	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam9	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值
MsgParam10	临时记录信息	临时记录的具体信息，来源于存储过程中的参数值

B.2 触发器方式数据库接口的存储过程脚本

```

create or replace procedure P_QueueMsg_业务应用数据表名 (
    OperateType in varchar2,
    MsgParam1 in varchar2,
    MsgParam2 in varchar2,
    MsgParam3 in varchar2,
    MsgParam4 in varchar2,
    MsgParam5 in varchar2,
    MsgParam6 in varchar2,
    MsgParam7 in varchar2,
    MsgParam8 in varchar2,
    MsgParam9 in varchar2,
    MsgParam10 in varchar2
)
as

```

```

begin
    insert into T_Oth_Msg ( PK_MsgId,MsgQueuedAt, MsgStatus, MsgDeliveredAt,
MsgFailedText,OperationType, MsgParam1, MsgParam2, MsgParam3, MsgParam4, MsgParam5,
MsgParam6, MsgParam7, MsgParam8, MsgParam9, MsgParam10 )
        values (seq_t_oth_msg.nextval,sysdate, 'NEW', NULL, NULL,OperationType, MsgParam1,
MsgParam2, MsgParam3, MsgParam4, MsgParam5, MsgParam6, MsgParam7, MsgParam8,
MsgParam9, MsgParam10);
end P_Trigger_业务应用数据表名;

```

存储过程参数设置说明见表 B.2。

表 B.2 存储过程参数设置说明

参数名	参数含义
MsgQueuedAt	临时记录从中间表中推送的时间
MsgStatus	标识临时记录是否为待推送的信息
MsgDeliveredAt	临时记录推送到目的地的时间
MsgFailedText	临时记录推送失败的详细信息
OperationType	临时记录在业务数据表中执行的操作（新增、更新、删除）
MsgParam1	业务数据表中受影响记录的具体信息
MsgParam2	业务数据表中受影响记录的具体信息
MsgParam3	业务数据表中受影响记录的具体信息
MsgParam4	业务数据表中受影响记录的具体信息
MsgParam5	业务数据表中受影响记录的具体信息
MsgParam6	业务数据表中受影响记录的具体信息
MsgParam7	业务数据表中受影响记录的具体信息
MsgParam8	业务数据表中受影响记录的具体信息
MsgParam9	业务数据表中受影响记录的具体信息
MsgParam10	业务数据表中受影响记录的具体信息

B.3 触发器方式数据库接口的触发器脚本

```

create or replace trigger 触发器名
    before [insert|update|delete] on 业务应用数据表名
    for each row
begin
    P_QueueMsg_业务应用数据表名 ([insert|update|delete],表字段 1,表字段 2,...);
End

```

触发器参数设置说明：

触发器中使用到的参数与存储过程、中间表中使用的字段含义一致。

触发器方式的数据库访问接口由中间表、存储过程、触发器之间相互配合完成，具体流程为：

由触发器对业务数据表进行实时监控。当业务数据表有变化时，对应的触发器对发生的变化进行捕获，并调用相应的存储过程，将发生的变化信息存储到中间表中。

附录 C
(资料性附录)

时间戳方式数据库接口的 SQL 脚本 (Microsoft SQL Server)

时间戳方式数据库接口的触发器脚本。

```
CREATE TRIGGER 触发器名  
ON 表名  
FOR [INSERT | UPDATE ]  
AS Update 表名 set [create_time | update_time ] =getdate();
```

触发器参数设置说明见表 C.1。

表 C.1 触发器参数表

参数名	参数中文名	应用意义
create_time	业务数据的创建时间	业务数据表中的字段, 用于记录表中数据的创建时间
update_time	业务数据的更新时间	业务数据表中的字段, 用于记录表中数据的更新时间

时间戳方式数据库接口的触发器工作流程为:

触发器对业务数据表进行实时监控, 当业务数据表有变化时, 触发该触发器对发生变化的时间进行记录, 并插入到对应的业务数据表时间字段中。

附录 D
(资料性附录)

时间戳方式数据库接口的 SQL 脚本 (Oracle)

时间戳方式数据库接口的触发器脚本。

```
create or replace trigger 触发器名  
  before [insert|update] on 业务应用数据表名  
  for each row  
begin  
  as update 表名 set [create_time|update_time] = sysdate;  
end
```

触发器参数设置说明见表 D.1。

表 D.1 触发器参数表

参数名	参数中文名	应用意义
create_time	业务数据的创建时间	业务数据表中的字段, 用于记录表中数据的创建时间
update_time	业务数据的更新时间	业务数据表中的字段, 用于记录表中数据的更新时间

时间戳方式数据库接口的触发器工作流程为:

触发器对业务数据表进行实时监控, 当业务数据表有变化时, 触发该触发器对发生变化的时间进行记录, 并插入到对应的业务数据表时间字段中。

附录 E
(资料性附录)

环境信息系统数据库访问示例

数据传输与交换平台系统访问国控重点污染源自动监控系统数据库

E.1 基本情况

系统名称：国控重点污染源自动监控系统

访问者：数据传输与交换平台

数据库访问接口：视图、时间戳

E.2 访问方式

数据库访问方式见图 E.1。

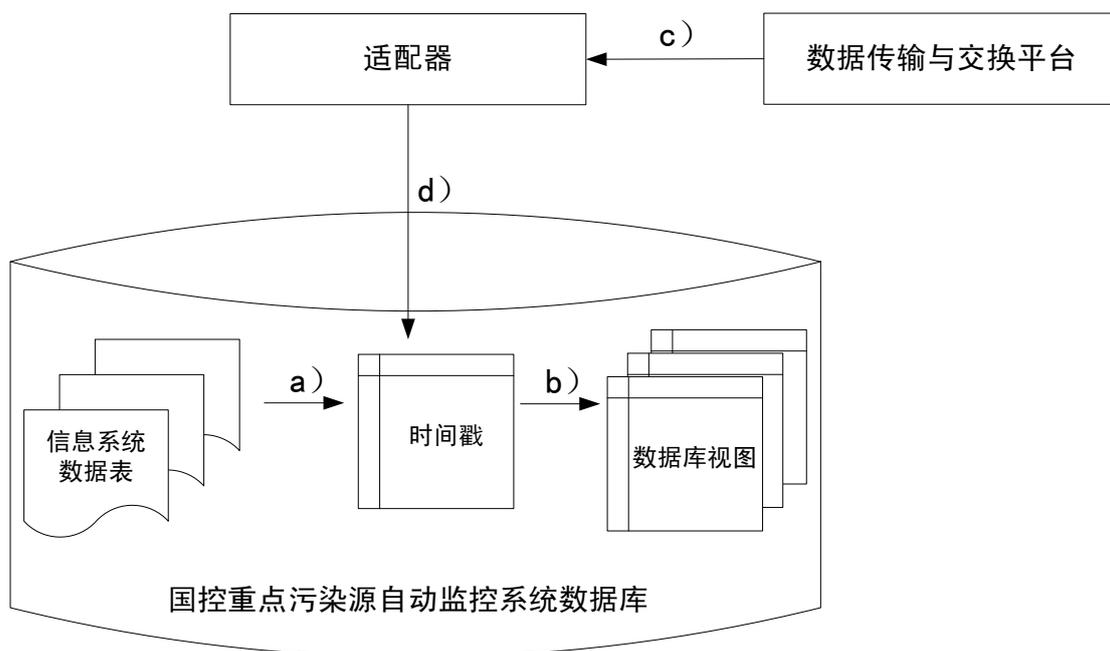


图 E.1 数据传输与交换平台访问污染源自动监控系统数据库方式

- a) 在国控重点污染源自动监控系统数据表中，将数据更新时间作为时间戳字段使用；
- b) 根据数据传输与交换平台的访问要求，将包含时间戳字段的数据表按照数据库视图接口方式创建访问视图（为防止影响信息系统数据表，采取视图方式）；
- c) 数据传输与交换平台通过适配器组件访问国控重点污染源自动监控系统数据库；
- d) 通过适配器配置轮询方式，定期访问数据库视图，根据当前时间与时间戳字段的比较，访问最新数据。