

# 虚谷数据库 V12.7

## 备份恢复指南

文档版本 01

发布日期 2025-01-10



版权所有 © 2025 成都虚谷伟业科技有限公司。

## 声明

未经本公司正式书面许可，任何企业和个人不得擅自摘抄、复制、使用本文档中的部分或全部内容，且不得以任何形式进行传播。否则，本公司将保留追究其法律责任的权利。

用户承诺在使用本文档时遵守所有适用的法律法规，并保证不以任何方式从事非法活动。不得利用本文档内容进行任何侵犯他人权益的行为。

## 商标声明



为成都虚谷伟业科技有限公司的注册商标。

本文档提及的其他商标或注册商标均非本公司所有。

## 注意事项

您购买的产品或服务应受本公司商业合同和条款的约束，本文档中描述的部分产品或服务可能不在您的购买或使用范围之内。由于产品版本升级或其他原因，本文档内容将不定期进行更新。

除非合同另有约定，本文档仅作为使用指导，所有内容均不构成任何声明或保证。

## 成都虚谷伟业科技有限公司

地址：四川省成都市锦江区锦盛路 138 号佳霖科创大厦 5 楼 3-14 号

邮编：610023

网址：[www.xugudb.com](http://www.xugudb.com)

# 前言

## 概述

本文档以实际操作为例，说明如何对数据进行备份与恢复操作。可以通过阅读本文了解数据库的备份策略以及操作流程。

## 读者对象

- 数据库管理员
- 数据库用户

## 符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 <b>注意</b>	用于传递设备或环境安全警示信息，若不避免，可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。
 <b>说明</b>	对正文中重点信息的补充说明。“说明”不是安全警示信息，不涉及人身、设备及环境伤害信息。

## 修改记录

文档版本	发布日期	修改说明
01	2025-01-10	第一次正式发布。

# 目录

1	备份恢复介绍	1
1.1	备份恢复概述	1
1.2	物理备份恢复概述	2
1.3	逻辑备份恢复概述	3
2	系统级备份恢复	5
2.1	系统级备份	5
2.2	系统级恢复	7
3	逻辑备份恢复	8
3.1	库级逻辑备份恢复	8
3.1.1	库级逻辑备份	8
3.1.2	库级逻辑恢复	9
3.2	用户级逻辑备份恢复	10
3.2.1	用户级逻辑备份	10
3.2.2	用户级逻辑恢复	12
3.3	模式级逻辑备份恢复	13
3.3.1	模式级逻辑备份	13
3.3.2	模式级逻辑恢复	14
3.4	表级逻辑备份恢复	15
3.4.1	表级逻辑备份	15
3.4.2	表级逻辑恢复	16

# 1 备份恢复介绍

## 1.1 备份恢复概述

在实际生产环境中，很多异常（比如硬件资源故障）都有可能导导致数据丢失，为保证数据的完整性，除多副本存储机制外，数据库还提供了更加灵活的备份方案供用户选择。虚谷数据库提供两种备份恢复方案。

- 物理备份恢复：即系统级备份恢复，直接备份数据库的数据文件，备份文件中包含所有库及库中的所有数据。恢复时通过替换集群各个节点对应备份的数据库文件即可。
- 逻辑备份恢复：对特定的数据库对象进行备份，支持库级、用户级、模式级、表级四种层级的备份方案，用户根据需要制定相关的备份恢复方案。

从数据库的备份策略角度出发，虚谷数据库支持全量备份和增量备份两种备份策略。

- 全量备份：每次对数据进行完整的备份，即对整个数据库、数据库结构和文件结构的备份，保存的是备份完成时刻的数据库，是差异备份与增量备份的基础。全量备份的备份与恢复操作都非常简单方便，但是数据存在大量的重复，并且会占用大量的磁盘空间，备份的时间也很长。
- 增量备份：只有那些在上次全量备份或者增量备份后被修改的文件才会被备份。以上次完整备份或上次增量备份的时间为时间点，仅备份这之间的数据变化，因而备份的数据量小，占用空间小，备份速度快。但恢复时，需要从上一次的完整备份开始到最后一次增量备份之间的所有增量依次恢复，如中间某次的备份数据损坏，将导致数据的丢失。

### 注意

- 本手册以实际操作为例，说明如何对数据进行备份与恢复操作。您可以通过阅读本文，了解数据库的备份恢复基本概念、方式、策略以及操作流程，本手册适用于所有的数据库用户。
- 阅读本文档前，请确保已了解数据库基础对象以及控制台的使用，基础对象及语法的详细信息请参见《SQL 语法参考指南》，控制台使用的详细信息请参见《控制台工具使用指南》。
- 本文使用 Linux 版本服务端及控制台工具进行演示，Windows 版本操作步骤（除登录操作外）与本文一致。
- 备份与恢复操作必须在同一平台以及操作系统下进行。如：在 Windows 平台备份的数据不能在 Linux 平台下恢复。

## 1.2 物理备份恢复概述

物理备份是数据库管理中重要的一环，能够确保数据的安全性和可恢复性。

### 定义

物理备份是指直接复制数据库的二进制文件，包括数据文件、日志文件和索引文件等，是磁盘块为基本单位将数据从主机复制到备机。

虚谷数据库物理备份恢复即系统级备份恢复，该备份为全备份，备份文件中包含所有库及库中的所有数据。由于系统级备份涉及整个系统内所有的数据，所需权限较高，因此该操作只能由系统管理员 **SYSDBA** 登录系统库“SYSTEM”执行。

### 应用场景

- 大型数据库快速恢复：物理备份适用于大型且重要的数据库。在数据库出现问题时，物理备份能够快速恢复数据库，减少业务中断时间。
- 细粒度恢复需求：物理细粒度备份恢复支持从集群级全量备份集中恢复单表或多表，以及备份指定的 schema 全量数据，满足细粒度恢复的需求。

### 优势

- 速度快时间短：直接备份恢复数据页，无需加载和解释语句，速度快。
- 占用空间小：只保存有效的数据页，节省存储空间。直接复制数据库的二进制文件，效率高且浪费空间较少。

但需注意，物理备份灵活性差，只能备份整个数据库或表空间，且备份文件依赖于特定的操作系统和数据库版本，不易查看和验证数据，建议根据实际需求选择合适的备份方式。

虚谷数据库支持的系统级备份与恢复功能对照如表 1-1 所示。

表 1-1 系统级备份恢复功能对照

备份恢复能力	系统级备份恢复
全量备份	支持
增量备份	支持
单机恢复	支持

接下页

备份恢复能力	系统级备份恢复
集群恢复	不支持
表级恢复	支持
模式级恢复	支持
用户级恢复	支持
库级恢复	支持

## 1.3 逻辑备份恢复概述

逻辑备份是数据库管理中常用的一种备份策略，特别适用于中小型数据库和需要跨平台迁移的场景。

### 定义

逻辑备份是数据库的一种备份方式，涉及将数据库中的数据和结构以逻辑方式导出，并保存到文件中。

除系统级备份外，虚谷数据库还提供逻辑备份功能。逻辑备份相对系统级备份而言，其备份的数据更加灵活，备份最小粒度支持到表级，提供了更多的备份选择。同时，逻辑备份不受数据库系统的限制，在单机和集群上均可执行逻辑备份操作。

备份内容包括数据库表的数据、结构（如表定义、索引等）、视图、存储过程和触发器等，这些都被导出为 SQL 脚本文件。

### 应用场景

- 小量数据迁移：逻辑备份特别适合于帮助业务用户、DBA 方便、简洁地迁移小量数据（通常认为在 5GB 内）。在跨操作系统、跨数据库版本的背景下，逻辑备份相比物理备份更为灵活方便和易操作。
- 跨平台和跨版本恢复：逻辑备份可以跨平台和跨版本恢复数据，为数据库迁移和升级提供了便利。
- 特定对象备份：可以选择性地备份特定的数据库对象（如库、模式、表），从而节省备份空间和时间。

- 数据查看和验证：逻辑备份文件是文本格式，方便用户查看和验证数据内容。

### 优势

- 灵活性和可移植性：可以选择备份特定对象，备份文件不受操作系统或数据库版本限制，易于迁移和跨平台使用。
- 易于管理和修改：备份文件是文本格式，方便查看、验证、编辑和管理，易于管理和修改。
- 协同工作：能够与正在运行的数据库自动协同工作，确保备份是当时的点。

逻辑备份虽然速度较慢且占用空间较大，但其灵活性和可移植性使其在特定场景下成为优选。建议根据实际需求选择合适的备份方式。

虚谷数据库支持的逻辑备份与恢复功能对照如表1-2所示。

**表 1-2 逻辑备份恢复功能对照**

备份恢复能力	库级逻辑备份	用户级逻辑备份	模式级逻辑备份	表级逻辑备份
全量备份	支持	支持	支持	支持
增量备份	不支持	不支持	不支持	不支持
单机恢复	支持	支持	支持	支持
集群恢复	支持	支持	支持	支持
表级恢复	支持	支持	支持	支持
模式级恢复	支持	支持	支持	不支持
用户级恢复	支持	支持	不支持	不支持
库级恢复	支持	不支持	不支持	不支持

# 2 系统级备份恢复

## 2.1 系统级备份

### 功能说明

系统级备份：全备份，备份文件中包含所有库及库中的所有数据。

由于系统级备份涉及整个系统内所有的数据，所需权限较高，因此该操作只能由系统管理员 **SYSDBA** 登录系统库 “SYSTEM” 执行。

### 语法格式

```
BACKUP SYSTEM [opt_increment] [APPEND] TO file_path [opt_online] [opt_encryptor] [opt_compress];
```

### 参数说明

- **opt\_increment**: [ ALL | INCREMENT ] 是可选项，ALL 是新建备份，INCREMENT 是增量备份，缺省是 ALL。一般用于系统级增量备份。系统级备份推荐使用 ALL。增量备份时，建议使用追加方式，将增量数据合并到上一次备份文件中，如果增量使用新建文件保存，在恢复时先清除 XHOME，再按照全量文件，增量文件的顺序一次恢复。
- **[APPEND] TO**: APPEND 是可选项，用于追加增量备份，只能在 INCREMENT 存在时使用，全量备份不允许追加到现有文件；增量备份可以新建文件也可以追加到现有文件，其中 INCREMENT APPEND 是追加到全量备份文件，而只加 INCREMENT 是新建文件备份。
- **file\_path**: 备份文件名，包含存储目录（指数据库的系统目录），需要用单引号包裹，如：/BACKUP/DEMO.DMP。
- **opt\_online**: [ ONLINE | OFFLINE ] 是可选项，备份时可选在线或离线备份，系统级的离线会强制断开当前所有的连接，缺省是 OFFLINE。
- **opt\_encryptor**: 可选项，加密的密钥。一般用于需要在备份时对导出的内容加密使用。
- **opt\_compress**: [ COMPRESS | NOCOMPRESS ] 是可选项，指示是否启用压缩，缺省启用压缩，压缩可以对重复度高的数据减少物理存储的空间占用。

**注意**

- 本文 `file_path` 中的目录均为数据库内部的逻辑目录，并非操作系统目录。其对应的操作系统目录默认映射至安装目录下的 `XHOME/BACUP`，具体的映射目录由 `SETUP/mount.ini` 中的配置确定，此配置文件详情可参考《安装指南》的安装前配置章节。
- 对于集群环境下的全量备份恢复和增量备份恢复，由于暂时不支持三节点以上的集群进行备份，在备份时节点个数不能超过三个，并且备份操作需在主节点上进行。

**示例**

由系统管理员 **SYSDBA** 登录系统库 “SYSTEM” 执行全量和增量备份命令。

```
SQL> BACKUP SYSTEM TO '/BACKUP/SYS.DMP';
prepare system backup,close other db connection
create new backup file,start backing up entire system
backup file /CATA/GSYS1.SYS success
backup file /CATA/UNDO.SYS success
backup file /CATA/LSYS1.SYS success
backup file /DATA/DATA1.DBF success
backup file /DATA/DATA2.DBF success
backup file /DATA/DATA3.DBF success
backup file /DATA/DATA4.DBF success
backup system finish

Execute successful.
Use time:4517 ms.

-- 增量备份
-- 新增增量备份文件方式
SQL> BACKUP SYSTEM INCREMENT TO '/BACKUP/SYS1.DMP'; -- 会新增一个
SYS1.DMP 文件

-- 追加到全量备份文件的方式
SQL> BACKUP SYSTEM INCREMENT APPEND TO '/BACKUP/SYS.DMP'; -- 会将后
续的修改保存到已经存在的全量备份文件中
```

**集群备份示例：**

1. 当前示例环境为 3 节点，1、2 号节点为 Master，在 1 号与 2 号节点处均可进行备份操作。

```
-- 集群备份，先判断当前节点是否是主节点，并且确定当前集群节点数
小于等于3
SQL> BACKUP SYSTEM TO '/BACKUP/SYS_CLUSTER.DMP';

-- 集群增量备份和单机语法一样
SQL> BACKUP SYSTEM INCREMENT APPEND TO '/BACKUP/SYS_CLUSTER.DMP
'; -- 会将后续的修改保存到已经存在的全量备份文件中
```

2. 执行命令后，等待出现 ‘Execute successful.Use time:xxx ms.’回显，则表示备份成功。

## 2.2 系统级恢复

### 功能说明

系统级恢复操作完成后，服务会立即停止，需要手动重新启动服务。执行恢复操作的权限和执行备份操作的权限一致，只能由系统管理员 **SYSDBA** 登录系统库“SYSTEM”进行操作。

### 语法格式

```
RESTORE SYSTEM FROM file_path [opt_encryptor];
```

### 参数说明

- file\_path: 备份文件的路径字符串，用单引号包裹。
- opt\_encryptor: 加密的密钥。如果在备份时使用了加密，那么在恢复的时候需要指定密钥用于解密。

#### 注意

集群系统级全量恢复和增量恢复集群同样不能超过三个节点，并且恢复时需非活跃主节点全部离线，只留下唯一的活跃主节点来执行恢复语句，恢复完成后服务会立即停止。此时重启集群，会自动将所有备份数据恢复到所有节点。

### 示例

恢复前，需要将备份文件拷贝至目标库的 /XHOME/BACKUP/ 下，然后由系统管理员 **SYSDBA** 执行恢复命令。

```
-- 系统级恢复
RESTORE SYSTEM FROM '/BACKUP/SYS.DMP';

-- 对于增量备份的恢复方式
-- 非追加方式备份
RESTORE SYSTEM FROM '/BACKUP/SYS.DMP'; --先恢复全量备份数据
RESTORE SYSTEM FROM '/BACKUP/SYS1.DMP'; ---恢复增量备份的数据

-- 追加方式
RESTORE SYSTEM FROM '/BACKUP/SYS.DMP'; --由于所有数据都在一个文件中，直接恢复
```

集群恢复示例：集群节点数小于 3，当前节点为活跃主 master 节点，且其余节点均不在线。

```
-- 全量恢复
RESTORE SYSTEM FROM '/BACKUP/SYS.DMP'; --恢复全量备份数据

-- 追加模式，全量恢复文件包含所有要恢复的数据，无需操作

-- 非追加模式
RESTORE SYSTEM FROM '/BACKUP/SYS1.DMP'; ---恢复增量备份的数据
```

# 3 逻辑备份恢复

## 3.1 库级逻辑备份恢复

### 3.1.1 库级逻辑备份

#### 功能说明

库级逻辑备份：备份目标库下所有的对象。因此，库级逻辑备份需要由备份库的管理员

**SYSDBA** 登录至备份目标库进行操作。

#### 语法格式

```
BACKUP DATABASE [opt_increment] [APPEND] TO file_path [
  opt_encryptor] [opt_compress];
```

#### 参数说明

- **opt\_increment**: [ ALL | INCREMENT ] 是可选项，ALL 是新建备份，INCREMENT 是在已存在的备份文件中追加，缺省是 ALL。一般用于系统级增量备份。系统级备份推荐使用 ALL。
- **[APPEND] TO**: 新建备份文件或追加备份内容至已存在的备份文件尾部。
- **file\_path**: 备份文件名，包含存储目录（指数据库的系统目录），需要用单引号包裹，如：  
/BACKUP/DEMO.DMP。
- **opt\_encryptor**: 可选项，加密的密钥。一般用于在备份时对导出的内容进行加密。
- **opt\_compress**: [ COMPRESS | NOCOMPRESS ] 是可选项，指示是否启用压缩，缺省启用压缩，压缩可以对重复度高的数据减少物理存储的空间占用。

#### 📖 说明

- 系统级备份与库级备份语法上唯一的区别在于关键字（系统级：BACKUP SYSTEM，库级：BACKUP DATABASE），注意区分。
- 库级备份无法指定某个库进行备份。

#### 示例

由备份库的管理员（SYSDBA）登录至需要做备份的库，执行备份命令。

```
SQL> CREATE DATABASE db_bak;
```

```
Execute successful.
Use time:1 ms.

SQL> USE db_bak;
db_name db_bak

Execute successful.
Use time:0 ms.

SQL> CREATE TABLE t_bak(c1 INT,c2 VARCHAR);

Execute successful.
Use time:21 ms.

SQL> BEGIN
2   FOR i IN 1..100 LOOP
3     INSERT INTO t_bak VALUES(i,'test'||i);
4     END LOOP;
5     COMMIT;
6     END;
7   /

Execute successful.
Use time:0 ms.

SQL> BACKUP DATABASE TO '/BACKUP/DB_BAK.EXP';
备份表 SYSDBA.T_BAK --
----- 数据行数 100

Execute successful.
Use time:60 ms.
```

### 3.1.2 库级逻辑恢复

#### 功能说明

为保证历史数据不受恢复操作影响，库级恢复不支持在非空库下进行操作，因此在执行恢复操作前，需要创建一个新库，然后由新库管理员执行。

#### 语法格式

```
RESTORE DATABASE db_name FROM file_path [opt_encryptor] [WITH
REINDEX];
```

#### 参数说明

- db\_name: 库名。
- file\_path: 备份文件的路径字符串，用单引号包裹。
- opt\_encryptor: 加密的密钥。如果在备份时使用了加密，那么在恢复的时候需要指定密钥用于解密。

- WITH REINDEX: 可选项, 表示在恢复的时候重建索引, 缺省是不重建索引。因为重建恢复库中所有的索引有可能耗时很长, 所以在数据量很大的时候, 可以在恢复完成之后手动重建索引。

### 📖 说明

恢复目标库和备份库字符集必须保持一致。

## 示例

库级恢复流程:

1. 恢复系统表。
2. 创建表对象。
3. 导入数据。
4. 重复步骤 2、3。

库级恢复前, 需要将库级备份文件拷贝至恢复目录下, 然后由库管理员登录到目标库执行恢复命令。

```
SQL> CREATE DATABASE db_restore;

Execute successful.
Use time:1 ms.

SQL> USE db_restore;
db_name db_restore

Execute successful.
Use time:0 ms.

SQL> RESTORE DATABASE db_bak FROM '/BACKUP/DB_BAK.EXP';
恢复表 SYSDBA.T_BAK --
----- 数据行数 100

Execute successful.
Use time:97 ms.
```

## 3.2 用户级逻辑备份恢复

### 3.2.1 用户级逻辑备份

#### 功能说明

用户级逻辑备份: 备份指定用户下所有对象。用户级备份需要由库管理员 **SYSDBA** 登录至用

户所在库执行。

## 语法格式

```
BACKUP USER user_name [APPEND] TO file_path [opt_encryptor] [
  opt_compress];
```

## 参数说明

- user\_name: 需要备份的用户名。
- [APPEND] TO: 新建备份文件或追加备份内容至已存在的备份文件尾部。
- file\_path: 备份文件名, 包含存储目录(指数据库的系统目录), 需要用单引号包裹, 如: /BACKUP/DEMO.DMP。
- opt\_encryptor: 可选项, 加密的密钥。一般用于在备份时对导出的内容进行加密。
- opt\_compress: [ COMPRESS | NOCOMPRESS ] 是可选项, 指示是否启用压缩, 缺省启用压缩, 压缩可以对重复度高的数据减少物理存储的空间占用。

## 示例

由库管理员登录用户所在库执行备份命令。

```
SQL> USE db_bak;
db_name db_bak

Execute successful.
Use time:0 ms.

SQL> CREATE USER u_bak IDENTIFIED BY 'abc@1234';

Execute successful.
Use time:0 ms.

SQL> CREATE TABLE u_bak.t_bak(c1 INT,c2 VARCHAR);

Execute successful.
Use time:21 ms.

SQL> BEGIN
2   FOR i IN 1..100 LOOP
3     INSERT INTO u_bak.t_bak VALUES(i,'test'||i);
4   END LOOP;
5   COMMIT;
6   END;
7   /

Execute successful.
Use time:0 ms.

SQL> BACKUP USER u_bak TO '/BACKUP/U_BAK.EXP';
```

```
备份表 U_BAK.T_BAK --  
----- 数据行数 100  
  
Execute successful.  
Use time:0 ms.
```

## 3.2.2 用户级逻辑恢复

### 功能说明

用户级恢复：恢复指定用户下所有对象的数据，恢复时，会执行对象创建操作。因此，在恢复前如果已存在此用户，则需要保证该用户下不存在任何数据，否则会出现恢复后的数据与备份数据不一致的情况。

### 语法格式

```
RESTORE USER user_name FROM file_path [opt_encryptor];
```

### 参数说明

- user\_name：恢复用户的用户名。
- file\_path：备份文件的路径字符串，用单引号包裹。
- opt\_encryptor：可选项，加密的密钥。如果在备份时使用了加密，那么在恢复的时候需要指定密钥用于解密。

#### 说明

用户级逻辑恢复时默认重建索引。

### 示例

恢复前，需要将备份文件拷贝至恢复目录，然后由库管理员登录至恢复目标库执行恢复命令。

```
SQL> USE db_restore;  
db_name db_restore  
  
Execute successful.  
Use time:0 ms.  
  
SQL> RESTORE USER u_bak FROM '/BACKUP/U_BAK.EXP';  
恢复表 U_BAK.T_BAK --  
----- 数据行数 100  
  
Execute successful.  
Use time:107 ms.
```

## 3.3 模式级逻辑备份恢复

### 3.3.1 模式级逻辑备份

#### 功能说明

模式级逻辑备份：备份指定模式下的所有对象。模式级备份需要由库管理员 **SYSDBA** 登录至模式所在库执行。

#### 语法格式

```
BACKUP SCHEMA schema_name [APPEND] TO file_path [opt_encryptor] [
    opt_compress];
```

#### 参数说明

- schema\_name: 需要备份的模式名。
- [APPEND] TO: 新建备份文件或追加备份内容至已存在的备份文件尾部。
- file\_path: 备份文件名，包含存储目录（指数据库的系统目录），需要用单引号包裹，如：  
/BACKUP/DEMO.DMP。
- opt\_encryptor: 加密的密钥。一般用于在备份时对导出的内容进行加密。
- opt\_compress: [ COMPRESS | NOCOMPRESS ] 是可选项，指示是否启用压缩，缺省启用压缩，压缩可以对重复度高的数据减少物理存储的空间占用。

#### 示例

由库管理员登录至模式所在库，执行备份命令。

```
SQL> USE db_bak;
db_name db_bak

Execute successful.
Use time:0 ms.

SQL> CREATE SCHEMA s_bak AUTHORIZATION u_bak;

Execute successful.
Use time:0 ms.

SQL> CREATE TABLE s_bak.t_bak(c1 INT,c2 VARCHAR);

Execute successful.
Use time:21 ms.

SQL> BEGIN
2   FOR i IN 1..100 LOOP
3   INSERT INTO s_bak.t_bak VALUES(i,'test'||i);
4   END LOOP;
```

```
5 COMMIT;
6 END;
7 /

Execute successful.
Use time:0 ms.

SQL> BACKUP SCHEMA s_bak TO '/BACKUP/S_BAK.EXP';
备份表 S_BAK.T_BAK --
----- 数据行数 100

Execute successful.
Use time:0 ms.
```

### 3.3.2 模式级逻辑恢复

#### 功能说明

模式级恢复：恢复指定模式下所有的对象，默认恢复至备份模式下。

#### 语法格式

```
RESTORE SCHEMA schema_name [RENAME TO new_schema_name] FROM
file_path [opt_encryptor];
```

#### 参数说明

- schema\_name：需要恢复的模式名。
- new\_schema\_name：可选项，恢复至新的目标模式名。
- file\_path：备份文件的路径字符串，用单引号包裹。
- opt\_encryptor：可选项，加密的密钥。如果在备份时使用了加密，那么在恢复的时候需要指定密钥用于解密。

#### 📖 说明

- 模式级逻辑恢复时默认重建索引。
- 执行恢复操作时，schema\_name 必须是一个已经备份的模式。如果提供的 schema\_name 与备份中的模式名称不一致，会报需恢复的模式不存在的错误。

#### 示例

由库管理员 **SYSDBA** 登录至恢复目标库执行恢复命令。

```
SQL> USE db_restore;
db_name db_restore

Execute successful.
```

```
Use time:0 ms.

SQL> RESTORE SCHEMA s_bak FROM '/BACKUP/S_BAK.EXP';
恢复表 S_BAK.T_BAK --
----- 数据行数 100

Execute successful.
Use time:107 ms.
```

## 3.4 表级逻辑备份恢复

### 3.4.1 表级逻辑备份

#### 功能说明

表级逻辑备份：相比于库级逻辑备份，备份粒度更细，用户可通过此种方式精确备份指定的表。执行表级逻辑备份时，可以将一个表备份至一个文件，也可以将多个表备份至同一个文件。

#### 语法格式

```
BACKUP TABLE [schema_name.]table_name [APPEND] TO file_path [
    opt_encryptor] [opt_compress];
```

#### 参数说明

- table\_name: 目标备份的表名。
- schema\_name: 表所属的模式名，默认为当前登录用户的同名模式。
- [APPEND] TO: 新建备份文件或追加备份内容至已存在的备份文件尾部。
- file\_path: 备份文件名，包含存储目录（指数据库的系统目录），需要用单引号包裹，如：  
/BACKUP/DEMO.DMP。
- opt\_encryptor: 加密的密钥。一般用于在备份时对导出的内容进行加密。
- opt\_compress: [ COMPRESS | NOCOMPRESS ] 是可选项，指示是否启用压缩，缺省启用压缩，压缩可以对重复度高的数据减少物理存储的空间占用。

#### 示例

由对象所属的用户登录数据库，执行备份命令。

```
SQL> CREATE TABLE t_backup(c1 INT,c2 VARCHAR);

Execute successful.
Use time:21 ms.
```

```
SQL> BEGIN
2   FOR i IN 1..100 LOOP
3   INSERT INTO t_backup VALUES (i, 'test' || i);
4   END LOOP;
5   COMMIT;
6   END;
7   /

Execute successful.
Use time:0 ms.

SQL> BACKUP TABLE t_backup TO '/BACKUP/T_BACKUP.EXP';
备份表 SYSDBA.T_BACKUP --
----- 数据行数 100

Execute successful.
Use time:0 ms.
```

## 3.4.2 表级逻辑恢复

### 功能说明

执行表级恢复时，允许要恢复的表在恢复库中已经存在，前提是两个表的表结构完全一致，否则无法恢复。表备份恢复支持两种恢复模式，按原表名恢复和按新表名恢复。

### 语法格式

```
RESTORE TABLE [schema_name1.]table_name [RENAME TO [schema_name2.]
table_rename] FROM file_path [opt_encryptor];
```

### 参数说明

- table\_name: 目标备份的表名。
- schema\_name: 表所属的模式名，默认为当前登录用户的同名模式。
- RENAME: 可以将原备份的表恢复到指定模式下的表。
- file\_path: 备份文件的路径字符串，用单引号包裹。
- opt\_encryptor: 可选项，加密的密钥。如果在备份时使用了加密，那么在恢复的时候需要指定密钥用于解密。

### 📖 说明

表级逻辑恢复时默认重建索引。

### 示例

由库管理员 **SYSDBA** 登录至恢复目标库，执行恢复命令。

```
SQL> DROP TABLE t_backup;

Execute successful.
Use time:0 ms.

SQL> RESTORE TABLE t_backup FROM '/BACKUP/T_BACKUP.EXP';
恢复表 SYSDBA.T_BACKUP --
----- 数据行数 100

Execute successful.
Use time:107 ms.
```



成都虚谷伟业科技有限公司

联系电话：400-8886236

官方网站：[www.xugudb.com](http://www.xugudb.com)