



## TRIBON M2 网络管理员指南

1	Tribon 环境管理 Administration of the Tribon Environment .....	3
1.1	程序环境 Program Environment.....	3
1.1.1	系统目录 System Directories.....	3
1.1.2	系统文件 System Files .....	3
1.1.3	程序执行 Program Execution - the Tribon M2 Job Launcher .....	4
1.1.3.1	Tribon 应用 Tribon Applications.....	4
1.1.3.2	启动方法 Start Method.....	6
1.2	工程 Projects.....	6
1.2.1	创建一个新工程 Creating a New Project .....	8
1.2.2	使用样品工程 Using the Sample Project .....	8
1.3	数据库及目标 Data Banks & Objects .....	9
1.3.1	数据库 Data Banks.....	9
1.3.2	数据库的组织 Data Bank Organisation .....	9
1.3.2.1	编入索引数据库 Indexed Data Banks .....	10
1.3.2.2	顺序数据库 Sequential Data Banks .....	10
1.3.3	Windows 文件系统数据库处理 Windows File System Handling of Data banks .....	10
1.3.4	从应用程序的数据库访问 Data Bank Access from Applications ..	11
1.4	Tribon M2 运行时间系统 Tribon M2 Runtime System .....	11
1.4.1	Tribon M2 工程服务器 Tribon M2 Project Server .....	11
1.4.2	数据库服务器 Database Server .....	14
1.4.2.1	本机文件系统数据库访问 Local File System Database Access .....	14
1.4.2.2	Tribon 数据库服务器访问 Tribon Database Server Access .....	14
1.5	Tribon M2 控制面板 Tribon M2 Control Panel .....	17
1.5.1	工程创建 Tribon M2 Project Setup .....	17
1.5.2	Tribon M2 环境设置 Tribon M2 Environment Setup.....	25
1.5.3	Tribon M2 数据库服务器的维护 Tribon M2 Database Server Maintenance .....	26
1.5.4	Tribon 表面服务器 Tribon Surface Server .....	28
1.5.4.1	Tribon 表面服务器初始设计 Tribon Initial Design Surface Server ..	28
1.5.4.2	NAPA 表面服务器 NAPA Surface Server .....	30
1.5.5	Tribon M2 工程服务器设置 Tribon M2 Project Server Setup.....	31
1.5.6	Tribon M2 服务设置 Tribon M2 Service Setup.....	32
1.5.7	Tribon M2 数据库效用 Tribon M2 Data Bank Utility.....	33
1.5.7.1	查找目标 Find Objects .....	33
1.5.7.2	数据库维护 Data Bank Maintenance .....	36
1.5.8	Tribon 许可操作 Tribon License Handling .....	41
1.6	命令行效用 Command-line Utilities.....	41
1.6.1	数据库的效用 Data Bank Utilities.....	41
1.6.1.1	备份 Backup.....	42
1.6.1.2	重组 Reorganisation .....	42



---

1.6.1.3 程序 Programs .....	42
1.6.2 Tribon 环境效用 Tribon Environment Utilities.....	47

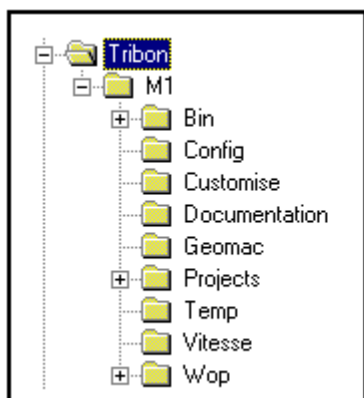


## 1 Tribon 环境管理 Administration of the Tribon Environment

### 1.1 程序环境 Program Environment

#### 1.1.1 系统目录 System Directories

Tribon 系统总是在根目录中被安装。在子目录结构中包含安排的许多不同的文件。



上图 Tribon M2 系统根目录和子目录。

#### 子目录

Bin: Tribon 系统可执行的程序和其他的文件。由 Tribon 环境变量 SB\_SYSTEM 识别。

Config: 储存现在的工程配置 & 设定。

Customise: 工程独立的例子文件。

Documentation: 在线的 HTMLHelp 文件。

Geomac: 程序独立的几何宏文件。

Projects: 包含样品和随系统一起安装的样板工程。

Temp: 临时的文件在程序运行期间产生。

Vitesse: Vitesse 源编码文件。

Wop: Tribon M2 装配计划和 Tribon M2 焊接计划可执行的和其他的文件。

TID: Tribon M2 初始设计可执行的文件和其他的文件

#### 1.1.2 系统文件 System Files

Bin, Config 和 Documentation 目录包含 Tribon M2 系统安装而且不应该用手编辑。



那些目录可能依次包含子目录。Bin包含子目录python（为python手写体，在Vitesse用），ora（在ORACLE桌面上为手写体设定）和其他。（为一般的Tribon资源）

## 1.1.3 程序执行 Program Execution - the Tribon M2 Job Launcher

Tribon章节参见单一Tribon运行文件实行，也就是任何的Tribon可执行的。Tribon例子如Drafting, Curved Hull Modelling 和 Ppanparts。

Tribon环境在Tribon环境桌面中表示所有的环境变量列出。Tribon环境桌面从窗口系统和用户变量被分开。

Tribon运行期间和他们的对应输入一起和输出参数叫执行Tribon工作。一个Tribon工作文件表示特定的工作一个例证。执行Tribon运行期间有一个包含关于申请的资讯，输入/输出参数的对应工作文件，日期实行及其他工作文件被认为是临时的；也就是他们没有把握真实的数据，但是只有关于那种明确工作的资讯的参数。他们被储存在一个子目录中在SB\_TEMP目录下面工作。

在实践中，工作文件对用户和使用的是看不见的，例通过Tribon Log Viewer看输入/输出参数和追踪执行状态。

### 1.1.3.1 Tribon 应用 Tribon Applications

所有的Tribon运用（除了公用程序以外）在一个xml文件中被定义提交如Tribon运用定义文件。运用定义文件位于SB\_SYSTEM目录而且被称为applications.xml。对于每个运用，那里有下列各项参数的一个<运用>xml节点：

- A. 运用名字
- B. 短运用名字（举例来说用名字给log文件）
- C. 可执行的文件名字
- D. 输入参数（描述和类型）
- E. 输出参数（描述，类型和文件命名扩展名）

例子：

```
<application>
  <name>Pipe Splitting</name>
```



```
<short_name>pi pespl </short_name>
<executable>sp600.exe</executable>
<inputs>
  <input>Input file</input>
</inputs>
<outputs>
  <output extension="lst">Verification list file</output>
  <output extension="lst">Result list file</output>
</outputs>
</application>
```

输出文件的名字在运行时被工作发射台选择，但是文件名字扩展名在这里被定义。

正常地，工作发射台寻找可执行的在 SB\_SYSTEM 目录中。这是真实的，如果文件名字在< executable >节没有路径被给，举例来说：

```
<executable>sp600.exe</executable>
```

开始从另外的一个位置来的一个运用超过 SB\_SYSTEM，然后完整的路径必须被给出在<executable>节，举例来说：

```
<executable >c:\winnt\system32\notepad.exe</executable>
```

Tribon 工作发射台也使用称为 userapps.xml 的另外一个文件，有如同 applications.xml 文件一般的语法。意图是 applications.xml 被留下无变化的和那个用户定义的运用和手写体被放在userapps.xml之内。相同的运用不应该被定义两次，也就是全部<name>节在 applications.xml 和 userapps.xml 都中应该是唯一的。

一个样品 userapps.xml 文件被包含在安装和举例说明中该如何执行 Microsoft® Notepad application 来自 Tribon 工作发射台。

```
<?xml version="1.0" standalone="yes"?>
<applications>
  <system name="User">
    <application>
      <name>Notepad</name>
      <short_name>np</short_name>
      <executable>c:\winnt\system32\notepad.exe</executable>
    </application>
  </system>
</applications>
```

请注意记事本运用仅仅被选择为例举例说明 xml 文件语法。



### 1.1.3.2 启动方法 Start Method

每 Tribon 运行期间给外界的私人拷贝和用它所有的输入数据供应运用, 每运行期间被 Tribon 工作发射台开始。这由开始 Tribon 运行期间有一个特别的起动器程序 (tbstartjob.exe)负责达成。

这条指令线将会产生并且执行 Tribon Drafting的一个新的例证。

tbstartjob.exe- application "drafting"

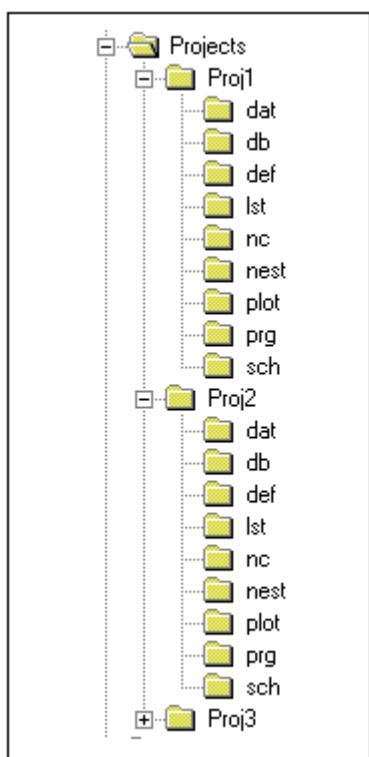
所有short-cuts对Tribon 运用在开始菜单使用这个方法。

工作发射台在后台为运用等候, 结束而且送通知信息给有关实行状态的log viewer。

## 1.2 工程 Projects

数据的收集在使用 Tribon船舶设计的工作运行期间用叫执行一个工程。数据构成一个工程被储存在许多的数据中和文件。 这些文件依照某目录结构被安排。

为每个标准的工程目录结构由根目录和一组子目录所组成。 这些子目录包含用户产生数据文件 (对 Tribon 工程的输入, 来自 Tribon 工程等的输出) 和其他的数据文件 ( 数据库, 默认值文件等) 对于设计工程的明确说明。标准的 Tribon 环境由接着组合子目录所组成的( proj 1- proj 3 可以是任意工程名字):



图Tribon M2 的例子工程目录。

## 子目录

## 例子的内容

Proj \dat

各种的输入数据文件。

Proj \db

数据库。

Proj \def

默认和标准。(举例来说缺省文件)

Proj \lst

输出和log文件。

Proj \nc

对NC控制机器的输出。

Proj \nest

套料零件 - 菜单文件。

Proj \plot

后操作的输出。

Proj \prg

批操作管模输入schemes

Proj \sch

结构生成schemes, curved panel schemes.

## 缺省文件

缺省文件为个别的 Tribon 文件包含细节和详细的设定。一个缺省文件被构成使用关键字和值。缺省文件由正在使用Tribon 环境指出目录路径和文件缺省的文件名字的变量文件是接近的。Tribon 环境变量一个缺省文件的一个例子是: SBD\_DEF1 Drafting运用缺省文件。

## IP 文件

IP 文件是类似的对于缺省文件在 Tribon 船体文件里面用。



## Tribon Settings Objects

在数据库中一些文件特殊设置被储存如目标。

## Windows Registry

工程单独的设定，不在用户之间分享，被储存在窗口记入中。

### 1.2.1 创建一个新工程 Creating a New Project

在安装的时候，可能安装一个从一个样板复制到另外的一个目录作为一个新的工程形成。运行这，执行接着的步骤：

1. 它所有的子目录和他们的内容一起在 Tribon 根目录之下，位于工程/ 样板目录的拷贝。你愿在窗口中与其他 Tribon 客户共享工程，多用户环境，只是产生对所有的 Tribon 客户是可接近的一个被共享的一个工程目录，而且在那里放置产生的样板目录的拷贝。

2. 重新命名新产生了样板文件夹拷贝，给它你愿产生的工程名字。请记工程名字现在被限制在长度八个字节而且他们不应该包括间隔。

3. 在工程目录中拷贝并且在 Tribon 根目录下面重新命名 d065template.sbd 文件，替换'template' 用你首选的工程名字。万一工程要被共享多用户环境，移动新产生的 d065<project>.sbd到先前创建的工程目录在共享的盘上（在第 1 步骤中）。工程选择工具（tbship.exe）能配置成从这工作共享的位置。

4. 编辑新的 d065<proj name> .sbd 文件并且替换所有的'template'，用你的新工程名字和目录路径。。

如果你的现在 Tribon 安装是单机和单一客户安装，第 5 步骤 - 9 不需要被运行：

5. 请注意所有的 Tribon 环境变量识别文件或目录（除数据库，在下面看）必须与有效的所有客户的路径一起定义。这由为所有的客户上的被共享的盘有通常的增长点（驱动器名）达成。

6. 确定一个 Tribon 数据库服务器（看见 1.5.2 数据库服务器）被安装和涉及有 Tribon 数据库的盘上地被装的机器。此外环境变量在 d065<project>.sbd文件中识别数据库的必须与路径一起定义，有效的在服务器机器上。

7. 在 d065<project>.sbd，建立 SB\_DB\_LOn Tribon 环境变量（看1.5.2 数据库服务器）指出新的数据库的位置文件。

### 1.2.2 使用样品工程 Using the Sample Project

可能安装一个局部的样本工程。准备好从最大多数文件的使用。包括lines 形成，但是目前不其他 TID 组件。

在这个安装引导中ds4工程名字被用为例子。

**注意：**在安装中样本工程将建立在工程子目录中在，在 Tribon 根目录之





下。如果样本工程从对另外的一个安装将被保存，那么必须在 Tribon 根目录外面被移动。进一步，所有的用户创建工程在 Tribon 根目录外面，保持新软件安装不受影响的。

## 1.3 数据库及目标 Data Banks & Objects

### 1.3.1 数据库 Data Banks

Tribon 数据库对于模型目标永久的储存。系统是结构化以便有不同的目的许多被预先定义的数据库。举例来说有一个数据库对于船体板架, 板子、扶强材部分、管子和设备等等是分开的。当 Tribon 文件存取的时候，数据库正常地提交到使用 Tribon 环境变量。Tribon 环境变量为数据库方面包括路径, 和没有文件类型扩展名的文件名字。如此的变量例子是:

变量	例子值
SB_OGDB	C:\tribon\m2\projects\myproj\db\ogdb
SB_CGDB	C:\tribon\m2\projects\myproj\db\cgdb

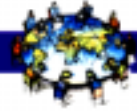
每个数据库可能包含一些不同的类型 Tribon 模型目标。目标为内在确认目的被一个名字识别而且有二个目标编码。一连串的目标在一个数据库中看起来像下列各项一样:

OBJECT	NAME	CREATION DATE	OC10C2
PSG=FUN		1999-08-10 15:40:56.94	4102
FD=114=		1999-10-12 18:21:14.07	4200
FD=114=510P101		1999-10-22 11:21:18.43	4503
FD=114=510P102		1999-10-22 11:21:23.37	4503
FD=114=510P103		1999-10-22 11:21:33.77	4503
FD=114=510P104		1999-10-22 11:21:58.21	4503
FD=114=720P801		1999-10-22 11:23:11.36	4501
FD=114=720P802		1999-10-22 11:23:13.16	4501
FD=114=720P807		1999-10-22 11:23:14.55	4501

目标名字使用惯例被建立和对每个 Tribon 文件是特定的限器。在目标之间有内部储存了参考。

### 1.3.2 数据库的组织 Data Bank Organisation

二种类型数据库存在，编入索引和次序的数据库。在一个被编入索引的数据库中，目标在任意的次序中被他们的名字存取。一个次序的数据库是被次序



的储存而且取回的目标序列。

### 1.3.2.1 编入索引数据库 Indexed Data Banks

大多数的数据库使用被编入索引的格式。 一个被编入索引的数据库由三个文件所组成，二个文件为索引和一数据文件。 作为一个数据库的文件例子：

c:\tribon\m2\projects\myproj\db\ogdb.dat Index file

c:\tribon\m2\projects\myproj\db\ogdb.idx Index file

c:\tribon\m2\projects\myproj\db\ogdb.eob Data file

数据库文件名字被限制在 8个字节，A到 Z,0 到 9. 扩展名是.dat为第一个索引文件,.idx为第二个索引文件和.eob为数据文件。

**重要的：** 重要的是索引和数据文件属于一个数据库必须在相同的磁盘片目录上。

### 1.3.2.2 顺序数据库 Sequential Data Banks

一个次序的数据库由次序的组织文件所组成。 数据库是顺序的目标，一些目标有相同的名字可能存在。

次序的数据库文件的名字是：

<file>.SDB

在那里：

<file> Data bank name

A到 Z,0 到 9, 最大值 63个字节。

### 1.3.3 Windows 文件系统数据库处理 Windows File System Handling of Data banks

数据库能被拷贝，删除和重新命名由使用正常的文件操作工具例如 Windows Explorer。

**注意：** 所有的三个文件为编入索引数据库必须在这些操作运行期间被一起保持。



**重要的：** 请注意数据库在如此的文件操作运行期间不能通过Tribon 应用访问。意外的目标访问可能引起目标/数据库破坏。

### **1.3.4 从应用程序的数据库访问 Data Bank Access from Applications**

数据库以二不同的方式从应用能被存取：

A . 局部的文件系统访问

B . Tribon 数据库服务器访问

较多的数据在这份文件中请看 1.5.2 数据库服务器。

## **1.4 Tribon M2 运行时间系统 Tribon M2 Runtime System**

Tribon M2 Runtime 系统是一个集体的名字 ,在网络的环境中管理Tribon 工程责任的许多功能。 它也包括多平台支持。 Tribon M2 Runtime 系统的主要特征要提供网络平台- 单独访问：

A . 数据库

B . 工程定义 (d065<project>.sbd 文件)

C . 缺省文件和 IP 文件

D . 应用输入和输出文件

Tribon M2 Runtime 系统功能性在二个分开服务器团体中被实现：

A . 工程服务器

B . 数据库和表面服务器

对 Tribon 数据的数据库服务器操作网络访问存入库, 而工程服务器操作访问工程定义和文件。(举例来说缺省文件) 数据库服务器和工程服务器能被独立地用依赖需要的环境。

### **1.4.1 Tribon M2 工程服务器 Tribon M2 Project Server**

概念和定义：

工程服务器落实以一个工程服务器服务外面的工程定义和促成客户访问他们为基础原则。 这些工程定义对配置成的所有客户将会是可用的，使用那个特别的主机上的工程服务器。 工程定义文件(a. k. a. d065-文件)一个工程服务器服务从工程服务器正在运行的主机必须是可接近的，也就是他们也必须被储存在局部的或在一个被共享的网络磁盘上。 如果一个工程服务器正在服务被储存在一个被共享的磁盘片上的外面工程定义，那么它必须配置成运行使用被叙述的用户帐户的安全上下文。 安全和证明稍后在这个区段中将会被描述。

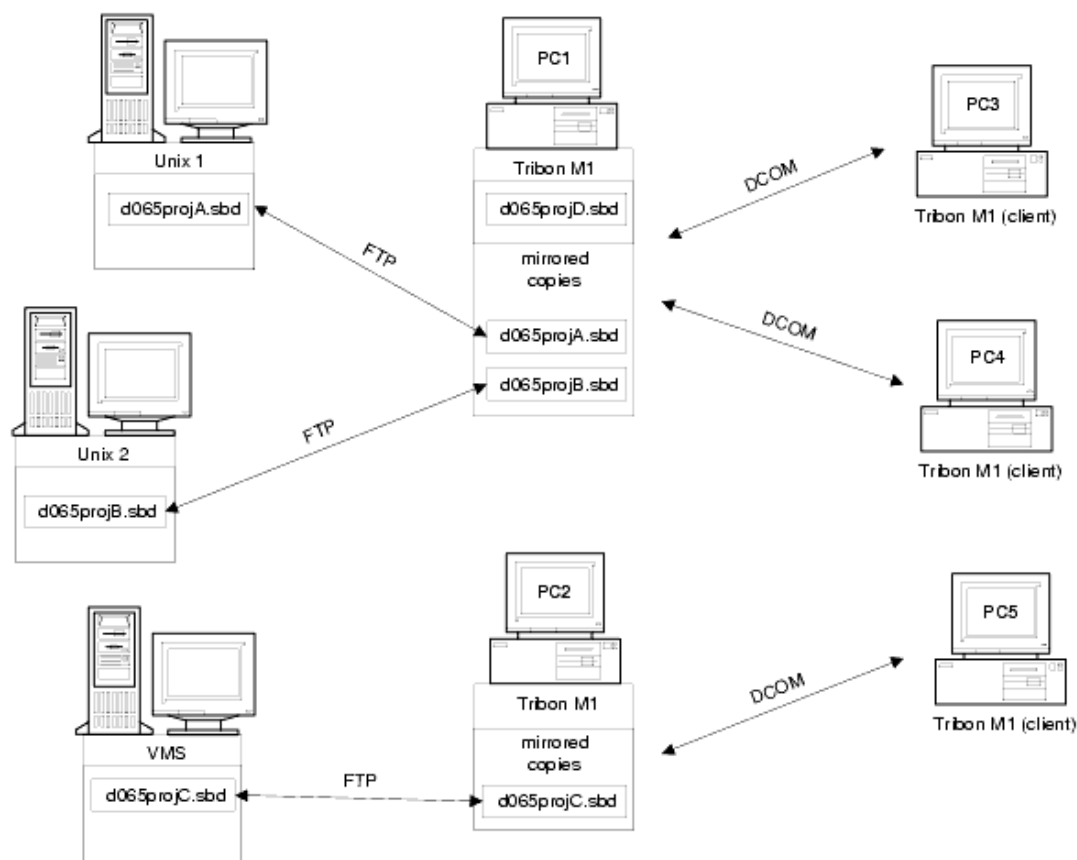
在这份文件，实际上服务外面的一个工程意谓服务外面工程定义文件。(也就是 d065-文件)



一个工程服务器只能在窗口平台上被运行。位于 Unix/ VMS 机器上的工程被称为遥远的工程。当一个工程服务器正在服务遥远的工程时候, 它将会保存局部地反映遥远的工程定义的拷贝。这相当重要, 既然它需要一些同步使两边上的工程定义最新。这不是一个问题如果工程管理人总是使用 *Project Setup applet* (在稍后被描述, 在 1.6.1 Tribon M2 中工程装备) 编辑工程定义和从不手工编辑的 d065- 文件。这个工具总是确定遥远的工程定义和反映的拷贝都将会与更新一起执行任何变化。

**注意:** 它不在 Tribon M2 和早先的 Tribon 之间支援到部份工程释放。功能的目的连接到除了 Windows 平台之外, 位于平台上的工程要帮助对 Tribon M2 工程的数据移植。

Tribon M2 客户机器必须选择一 (和唯一的) 机器制造运行工程服务器, 或在网络中的局部的机器或另外的一个个人计算机。然而, 所有的客户机器不必连接到相同的工程服务器机器。举例来说, PC3 和 PC4 同时可能使用 PC1 上的一个工程服务器同时 PC5 使用 PC2 上的一个工程服务器。在这份文件处, 这将会指示如 PC2 为 PC5 是被指定的工程服务器而 PC3's 和 PC4's 指定工程服务器是 PC1。(在下面看照片) 在这张照片, PC3 和 PC4 中将会可以存取 PC1 上的工程和 Unix 机器(也就是 proj A, proj B 和 proj D), 而 PC5 只将会有机会接近 VMS 机器上的工程。(也就是 proj C)



图Tribon M2 工程服务器 - 概观



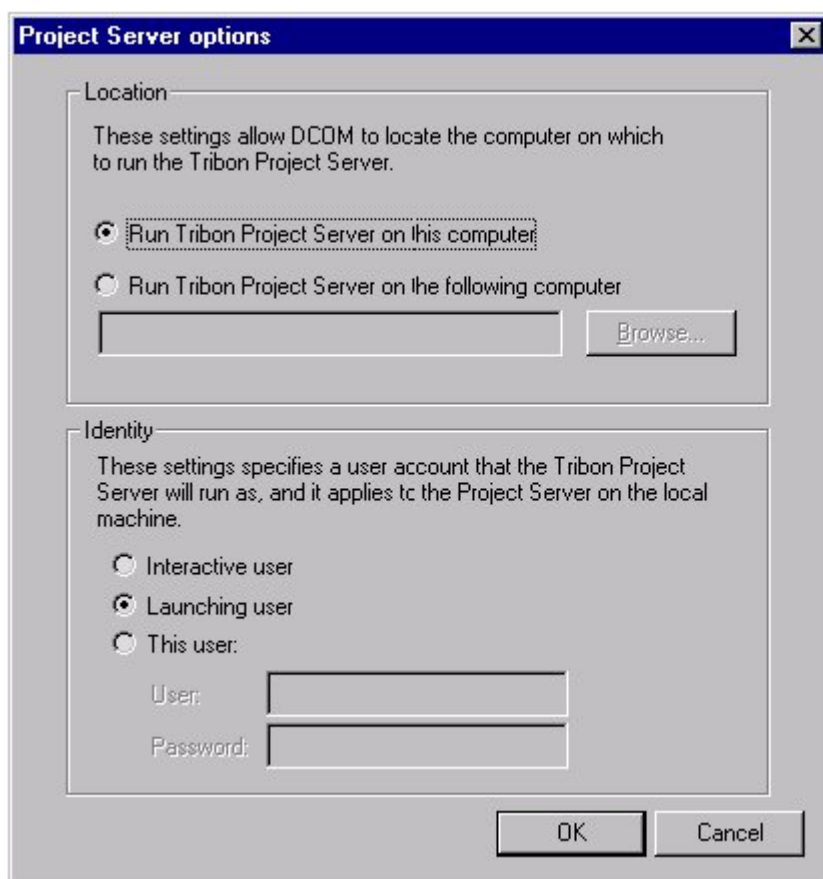
## 安全和认证

被提到有关工程服务器结构的事二个重要的争议。

A . 位置信息

B . 识别信息

位置信息告诉系统该如何找出正确的计算机。 依照上方提到了，每部机器流动的 Tribon M2 必须选择一个被指定的工程服务器。（或局部的机器或另外的一部机器在网络中）它意谓是用户应该叙述指定工程服务器，正在服务外面的工程，他/她想要得到。 如果用户想要操作局部的机器上的工程服务器的工程，*Run Tribon Project Server on this compute*按钮应该被选择。 请注意这个设定是每一机器局部的设定。



图为工程服务器选项

为了要改变局部的机器上的工程服务器设定，工程服务器选项会话盒子从 *Options*能显示在 *Project Selection tool*和 *Project Setup applet*。Tribon M2 控制板架程序叫执行 *Tribon M2 Project Server Setup*能被用很远地在网络中配置所有机器上的工程服务器设定，并且稍后在这个区段中将会被描述。

识别信息叙述一个用户帐户工程服务器将会运行，它只是工程服务器运行的机器上的一个有关的设定。 服务器运行，当这个用户要在领域中被认证。 它可能是这三替代方案之中的任何一个：



## A. 交互式用户

选项叙述工程服务器将会运行目前登陆的机器和使用安全文本(交互式用户)。交互式用户可能是相同于下载用户。 注意如果选项被选择, 如用户不登陆, 工程服务器将不启动。

## B. 下载用户

选项叙述工程服务器将会运行使用启动了工程服务器的用户的安全文本。  
(下载用户)

## C. 此用户

选项将会使工程服务器运行使用被叙述的用户帐户的安全文本。  
在一个新的 Tribon M2 安装上, 缺省的工程服务器设定是局部机器上的 '下载用户'。

**重要的:** 如果一个工程服务器正在服务, 文件在一个被共享的网络磁盘片上储存的工程定义, 它必须被配置运行制造这些被叙述用到其他机器上的客户是看得见的工程。否则, 如果它配置成运行下载用户或交互式用户的时候, Window's的安全系统只会把工程服务器访问给本机器上的磁盘。

## 1.4.2 数据库服务器 Database Server

Tribon 数据库以二不同的方式应用访问:

### A. 本机文件系统访问

### B. Tribon 数据库服务器访问

访问方法被许多的 Tribon 环境变量内容规定。 如果SB\_DB\_LOC1 Tribon 环境变量没有任何的值, 本机文件系统数据库访问被选择。 否则 Tribon 数据库服务器访问被用。

### 1.4.2.1 本机文件系统数据库访问 Local File System Database Access

本机的文件系统访问只能在单一- 用户环境中被用, 举例来说当 Tribon 装在一个单机个人计算机或一部可提式计算机的时候。 然后数据库必须在一个本机的磁盘片上。 网络或共享盘在使用本机文件系统访问方法, 作为数据库不支援。 本机的文件系统访问对SB\_DB\_LOC1 Tribon 环境变量不分配任何的值。

尝试连接到位于另外的一个主人共享的盘上数据库将会造成打开失败。 对数据库打开失败的进一步的解释将会在文件 C:\Temp\dbserver.log 中。

图数据库的例子访问, 通过本机文件访问能像这样在工程装备文件中建立: (d065<project>.sbd on SB\_BIN)

SB\_PDB c:\tribon\m2\projects\ds4\db\ppi db

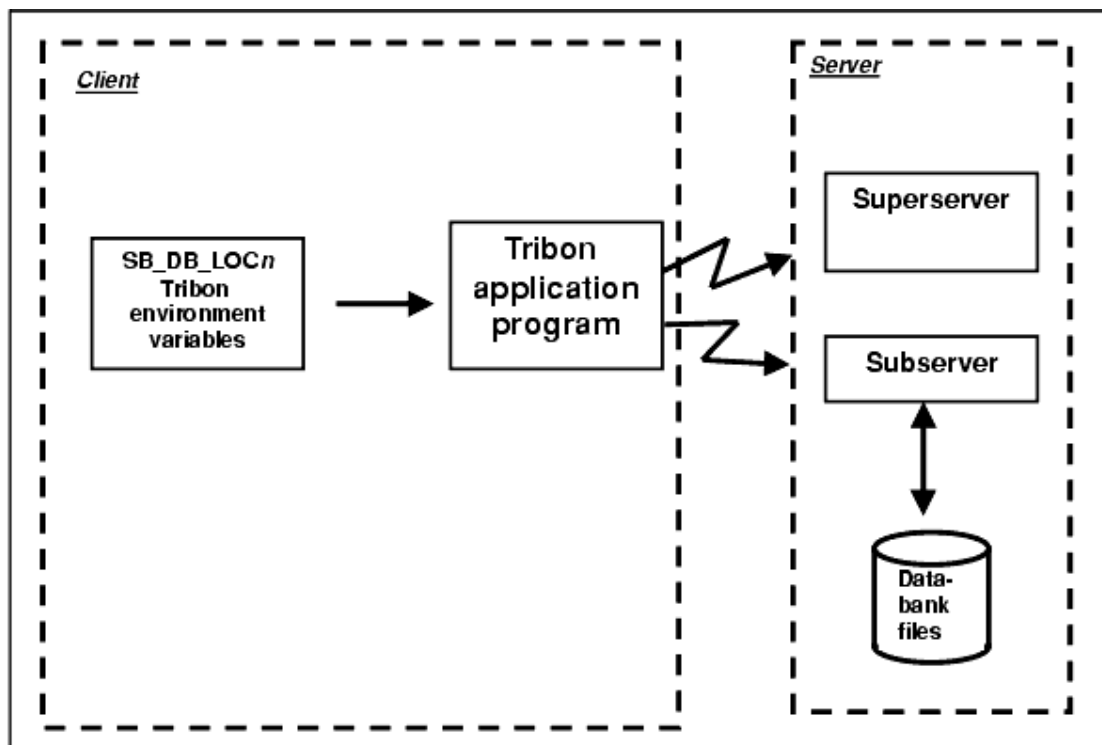
### 1.4.2.2 Tribon 数据库服务器访问 Tribon Database Server Access

Tribon 数据库服务器提供多用户网络访问。 数据库位于盘上, 在网络中的





其他机器上通过 Tribon 数据库服务器访问。在网络中用 Tribon 数据库服务器有一些主机，但仅仅一个数据库服务器在每个主机上。一个工程数据库在这些主机之间能被分配。客户-服务沟通以ONC RPC为基础 (Open Network Computing Remote Procedure Calls)，因此需要一个 portmapper 程序涉及服务器机器。portmapper 被包含在Tribon 传输中和自动地被Tribon 数据库服务器一起安装和配置。



图为数据库服务器概观。

在服务器机器中，有一个 superserver 对来自客户应用的呼叫工程 (ea312.exe)。当一个应用开始，第一数据库呼叫发行的时候，superserver程序产出 subserver(ea310.exe) 为每个程序操作。在应用程序实行的时候，工程直接的与subserver 程序沟通分配。当一个工程结束 subserver 程序自动地停止。

客户经过 Tribon 环境变量 SB\_DB\_LOc1..n 的值找出数据库服务器。

SB\_DB\_LOc n 的内容是文件路径的任意数字和他们对应服务器主机名字。文件/ 变量从对结束到开始扫描，很快执行一相配名字，被定义的路径主机名字将会取得，和那个服务器连络。通配符能用叙述路径和文件命名。在下面 SB\_DB\_LOc n Tribon 环境变量的一个例子：(来自工程设置 d065<project>.sbd 文件)

SB_DB_LOc1	c:\ tribon\ m2\ projects\ ds4* server1
SB_DB_LOc2	c:\ tribon\ m2\ projects\ fd* server2
SB_DB_LOc3	*123.456.789.012



注意: SB\_DB\_LOCn 变量的定义需要一些特别的请注意,因为在相配程序中内在的行相配使用专门的字符预先定义意义。如果它由一个反斜线(\)进行,如此字符的特别意义被转换。

任何的特别字符跟随反斜线(\)被认为是一个相配特别字符本身。特别字符是:

., \*, [, ], 和 \ (分别是句点, 星号, 左方框, 右方框 和反斜线), 总是特别, 除非他们出现在方括号里面。

^(脱字符号或抑扬符号), 在整个的表达开始是特别的, 或直接地跟随一对方括号的左边. ([ ])

\$(美元标记), 在整个的表达结束是特别。

在上述的例子中, 工程 ds4所有的数据库在主机 server1 上连续地经过数据库服务器访问, 工程 fd所有的数据在主机 server2 上和所有的其他数据用主机IP地址 "123.456.789.012"存入库。在指示的服务器主机上一个目录的路径 'c' 必须是可用的, 但是它从不访问客户。这意味当客户指示自己驱动器 'c' 时不需要任何的目录。

如果一个数据库的路径在 SB\_DB\_LOCn 没发现, 那么系统试着去本机访问数据库。在上面例子中, 当最后一行包含单一星号, 这个可能从不发生, 将会捕捉所有先前无法解决的路径。

如果数据库服务器在一个 OpenVMS 平台上运行, OpenVMS 文件名字必须用如在下方的例子中。

注意: 特殊字符必须被一个后斜线前缀。

```
SB_DB_LOC1      vms1\$dka0: \[ tribon\.m2\.projects\.ds4\]*vms1
SB_DB_LOC2      vms2\$dka0: \[ tribon\.m2\.projects\.fd\]*vms2
SB_DB_LOC3      *123.456.789.012
```

如果数据库服务器在一个 UNIX 平台上运行, UNIX 文件名字必须用如在下方的例子中。 SB\_DB\_LOC1 /usr/tribon/m2/projects/ds4/\*unix1

```
SB_DB_LOC2      /usr/tribon/m2/projects/fd/*unix2
SB_DB_LOC3      *123.456.789.012
```

## 应用访问数据库服务器的例子

这个例子指示一个Windows主机上的数据库服务器, 如果数据库服务器正在涉及 UNIX 或 OpenVMS 平台, 相同的原则被用。

1. 应用涉及内部Tribon 环境变量例如 SB\_PDB。同等值为这个变量包含路径和应用数据库的名字。可以在工程文件中建立如下面的: (d065<project>.sbd)





SB\_PDB c:\tribon\m2\projects\ds4\db\ppi db

2. 应用程序将会企图找 Tribon 数据库服务器运行的服务器主机名字或 IP 地址。SB\_DB\_LOcn Tribon 环境变量被读，和主机名字 server1 被发现。
3. 应用程序连络主机 server1 上的 superserver 程序运行，将会为现在的应用例证产生新的subserver 程序。
4. 应用程序为数据库访问与subserver 程序沟通。subserver 访问 ppi db 数据库应用在服务器主机上"c:\tribon\m2\ projects \ds4\db"的目录。

## 1.5 Tribon M2 控制面板 Tribon M2 Control Panel

Tribon M2 控制面板在Tribon M2 中是一个应用程序，以相似如下窗口方式控制面板：

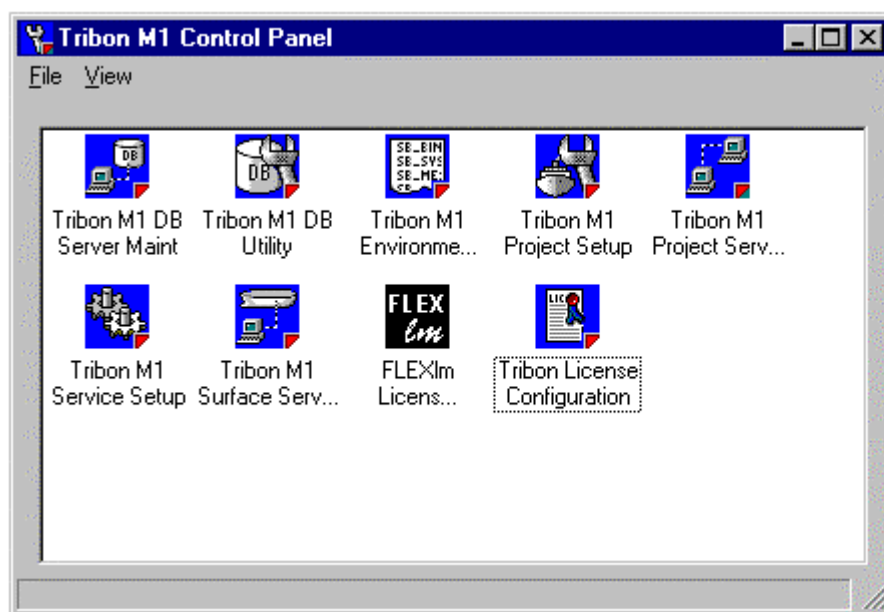


图 为 Tribon M2 控制面板

它包含例如服务管理，工程/ 环境设置和数据库管理的许多程序。

### 1.5.1 工程创建 Tribon M2 Project Setup

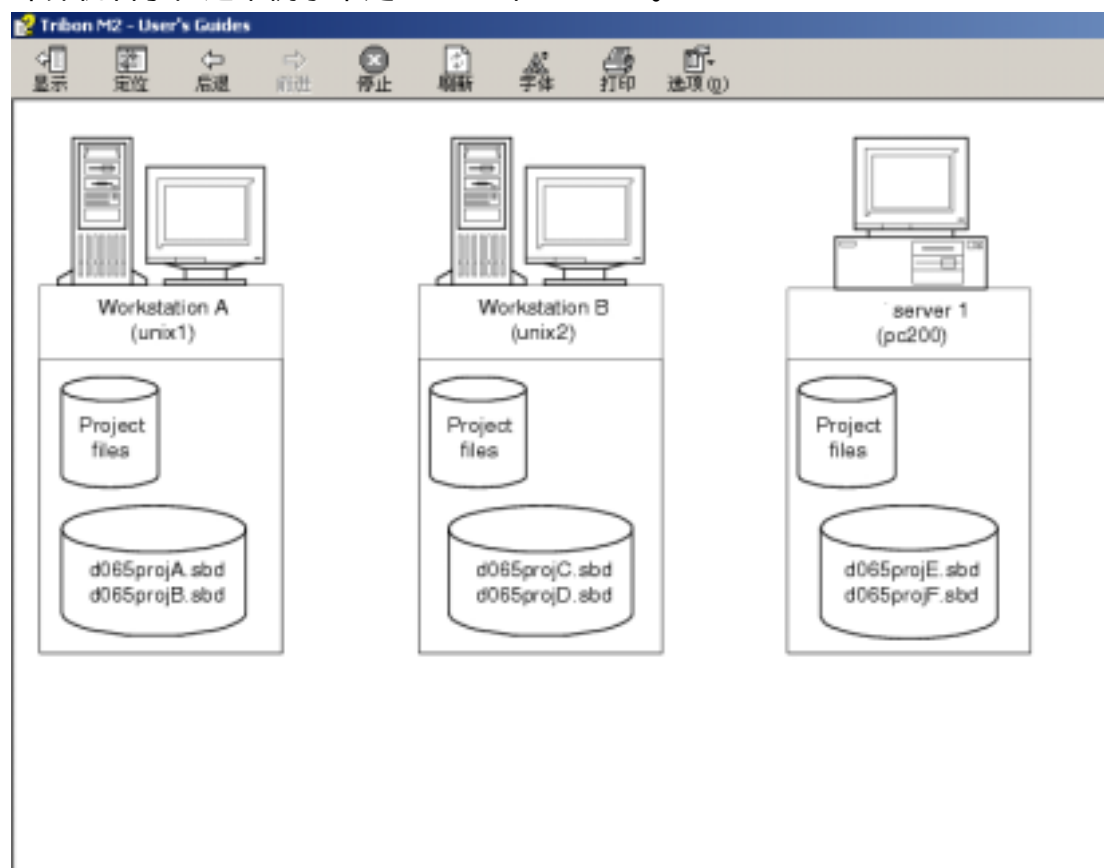
Tribon M2 工程创建程序能用来编辑工程定义，也就是编辑工程参数和创建或删除子工程。

**注意：** 它不支持在 Tribon M2 和早先的 Tribon版本之间的共享工程。功能的目的是连接工程位于除了Windows平台之外的平台，促进 Tribon M2 工程的数据移植。



典型的例子:

这个例子举例说明和配置一个工程位于二个工作站 (Unix/VMS) 上的系统, 怎么设置在一个服务器上。 这三部机器上的工程定义应该对所有客户运行的 Tribon M2是可用的。 当 Windows 服务器的名字是 pc200 的时候, 工作站的计算机名字在这个例子中是 unix1 和 unix2。



图为典型的多平台设定。

每个工程定义文件的收集表示一个工程组。 换句话说, 一个工程组符合一个包含工程定义文件的目录。 这意味如果有工程定义文件在多目录中, 那么对于每个目录工程组是需要的。 工程组不能被嵌套。 一个工程服务器可能有一些工程组, 工程组能自由的命名。这个例子中是三个工程组 - 一个组的每个目录包含 d065-文件。 工作需要被分为二个步骤:

1. 在 pc200 上创建工程组。
2. 配置Windows客户使用 pc200 上的工程服务器。

双击pc200 上的工程设置程序, Tribon M2 Project Setup在 Tribon M2 控制面板中开始。 在这个例子中, pc200共享工程定义。 因此, 确定在 pc200 上工程服务器被配置, 运行本机在工程服务器选项对话框中通过选择Run Tribon Project Server on this computer按钮。

为了要创建一个新的工程组, 在Registered projects树控制右击和在菜单项目上点击Create New Group. . . .

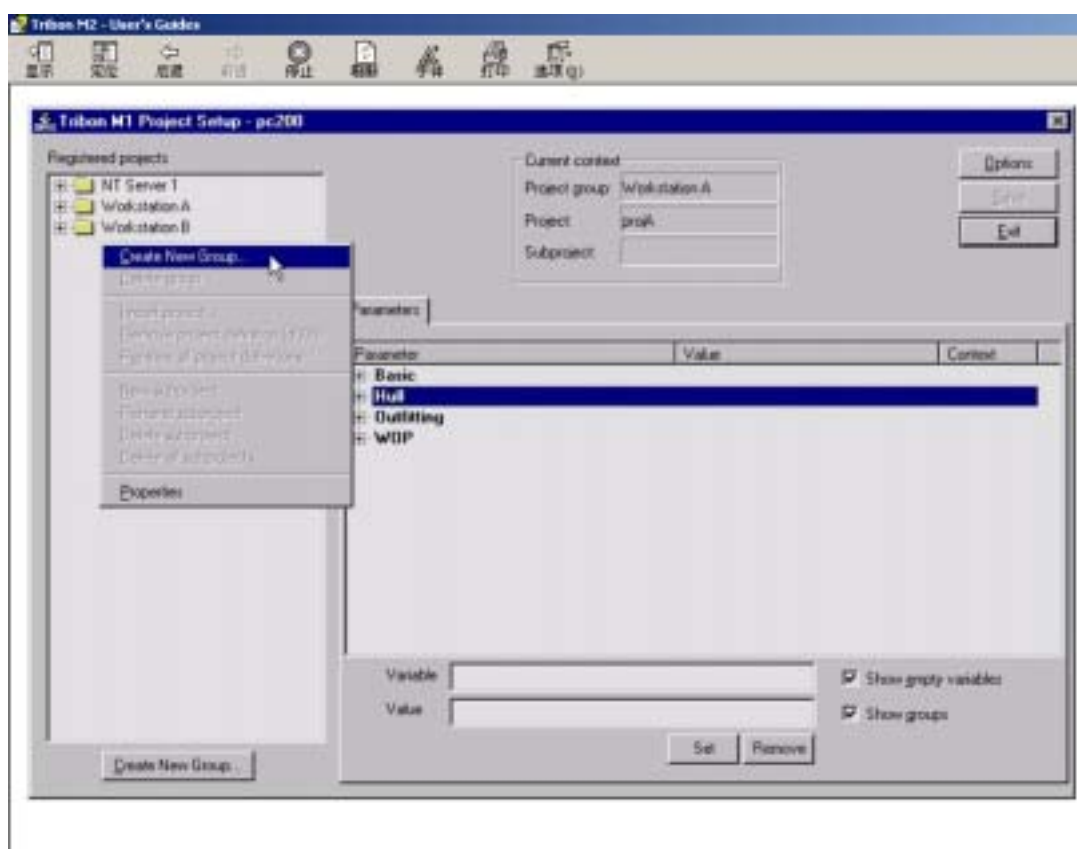


图 为Tribon M2 工程设置 - 主要窗口  
这将会显示Project Group properties对话框。

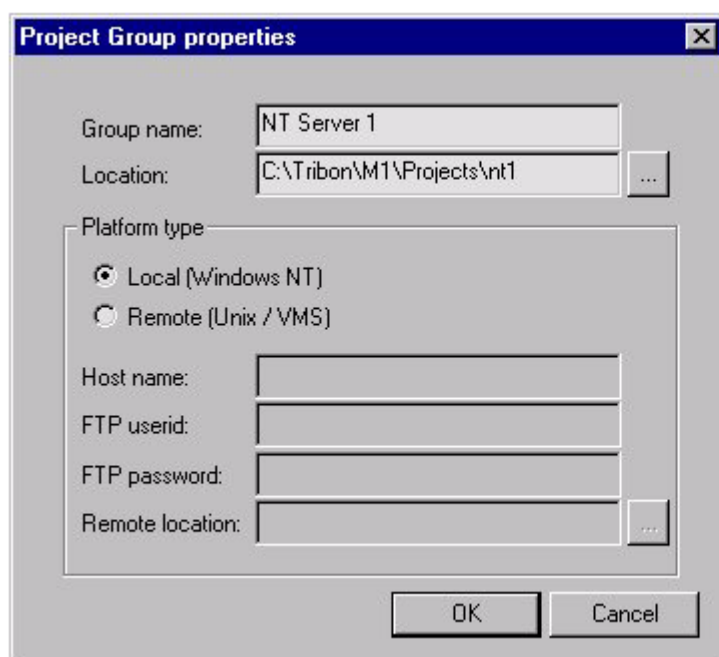
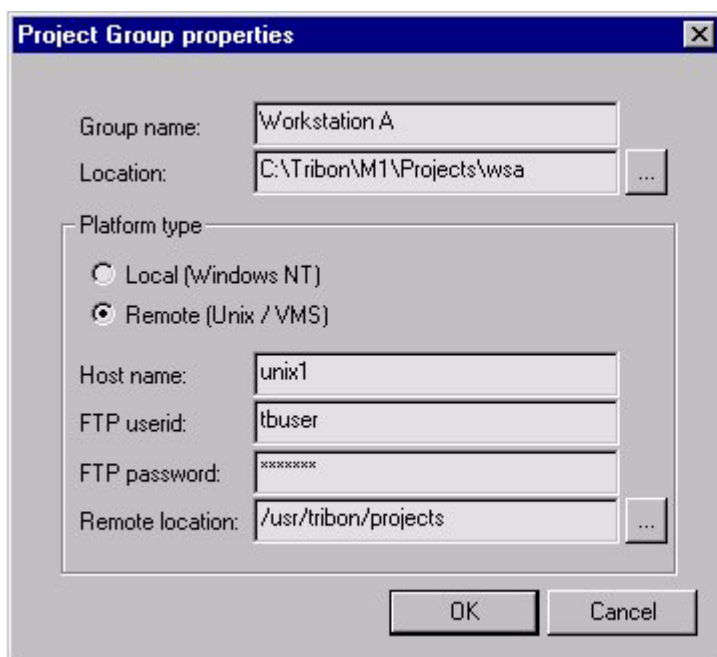




图 为 Tribon M2 工程设置 - Project Group properties

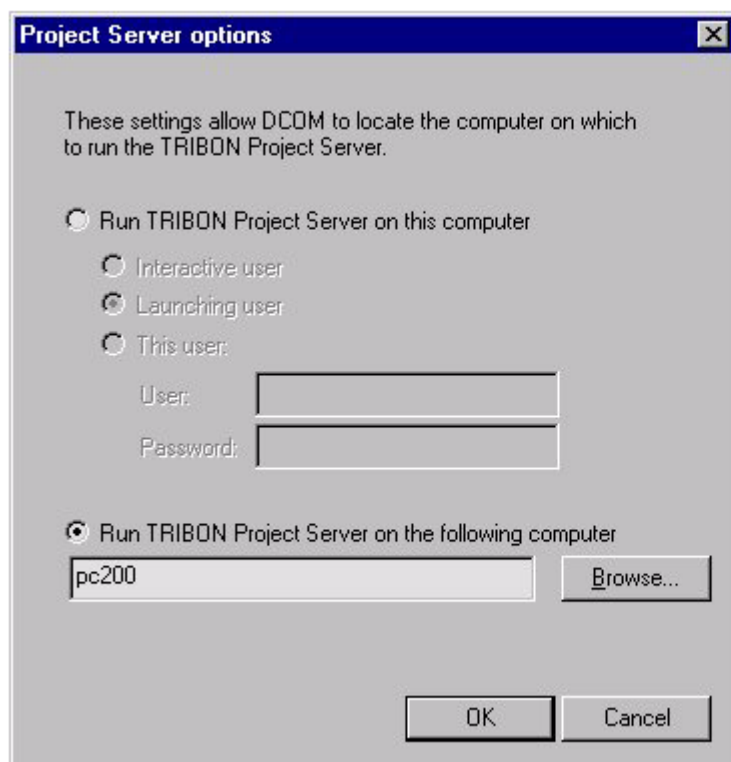
首先, 创建一个相应的工程组, 位于服务器(pc200)上的工程. 给它一个名字和叙述 d065- 文件的位置( 或使用浏览按钮选一个目录)。值得说的是在同样机器上不必使用工程安装程序, 工程是定位的。换句话说, 配置一部遥远的机器上的一个工程服务器是可能的。但是记住 d065- 文件的位置是非常重要的, 指定工程服务器机器是本机的路径。因此当打击 '...' 按钮到位置领域, 'Select folder'对话框将会让你浏览在工程服务器选项对话框中被叙述的机器上的文件夹。工程在 *NT Server 1*工程组里面是本机的 (也就是Windows NT), 如此平台类型应该是*Local (Windows NT)*, 在例子中 „ Windows NT被显示。Windows XP 和 Windows 2000 类似的工作)。



图为Tribon M2 工程设置 - Project Group properties

其次, 创建工程组将会包含反映的工程定义在工作站A和工作站 B 上. (分别地命名unix1 和 unix2) 再一次, 给一个名字和叙述指定的工程服务器机器 (也就是 pc200 上的本机路径) 上的一个位置储存反映的工程定义文件。然后设定平台类型到遥远的 (Unix/VMS) 和叙述工程定义文件的遥远主机名字, 有效的文件传送协议密码和遥远的位置。'...' 紧邻遥远位置的按钮使它可能浏览在主机名字领域中被叙述的遥远机器上的文件夹。(经由FTP, 使用FTP userid 和FTP密码) 当创建一个遥远的工程组和点击OK按钮的时候, 所有的工程定义文件在遥远的主机上被叙述的目录中将会被转移到目录在那上指定工程服务器机器, 给出Location领域。一般推荐'company-wide' FTP userid 被创建, 至少读 /写的目的有权访问到工程数据目录的地方。

最后的步骤要配置Windows客户使用 pc200 上的工程服务器。工程选择应用程序能被处置(tbship.exe)。点击选项按钮提出工程服务器选项对话框, 检查 'Run Tribon Project Server on the following computer' 按钮和进入 pc200 。



图为工程服务器选项

当用OK关闭对话框的时候，工程然后指定工程服务器服务将会在工程树中被显示：





图为 Tribon M2 工程选择

注意： 工程服务器设置程序能用来很远地配置所有的NT客户。稍后在这个区段中将会被讨论。

该如何配置一个工程：

这个区段描述使用 Tribon M2 工程设置程序该如何配置工程定义，和主要地有三个问题：

- A . 编辑工程参数
- B . 创建/ 删除 子工程
- C . 用手重新输入遥远的工程

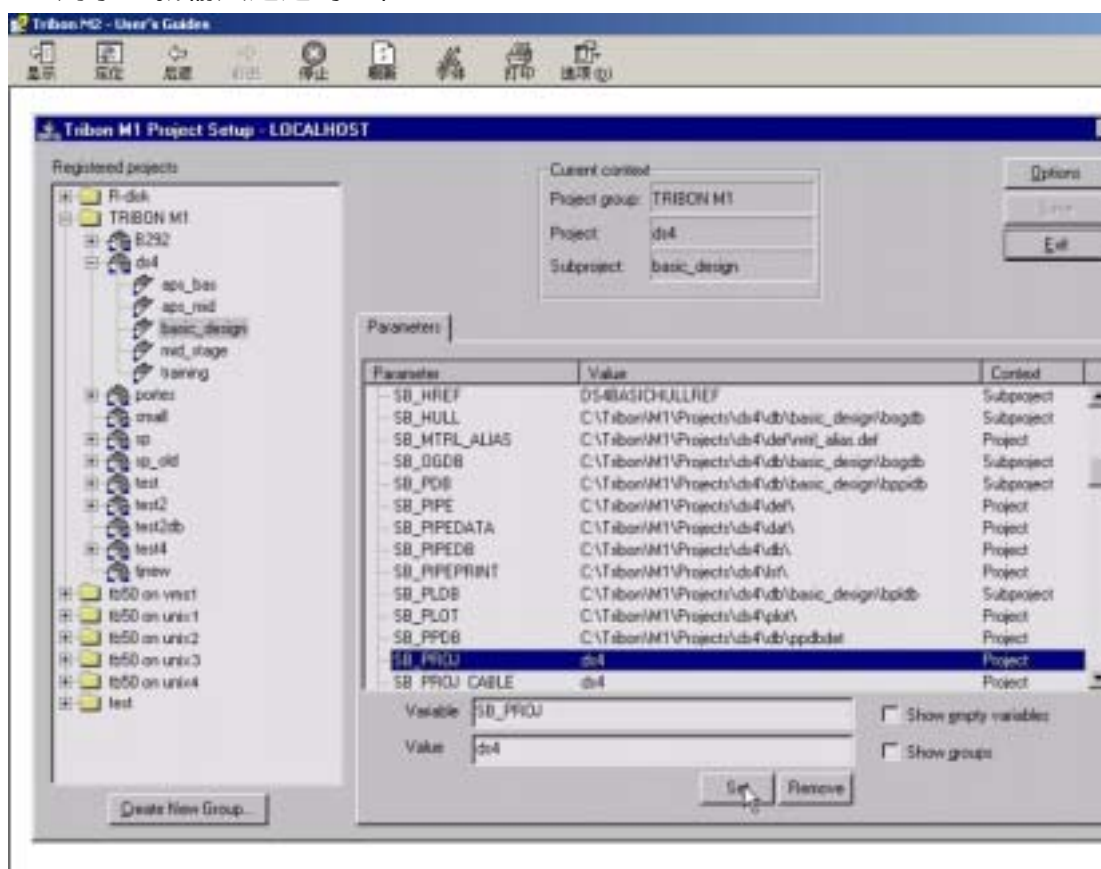


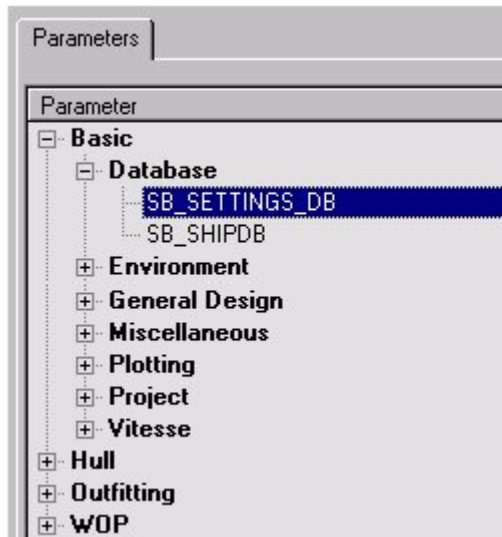
图 为Tribon M2 工程设置 - 主要窗口

第一打开对应的工程组然后在一个工程上按注册工程树中编辑工程参数 ,选择工程(或 子工程)。 当工程和子工程分别地有一艘船和半条船图标的时候，工程组与一个文件夹图标有关。 当一个工程（ 或子工程）被选择的时候，参数将会在Parameters中被显示列出和任何的值能被改变，插入或移动使用Set和Remove按钮。 Show empty variables 控制盒提供选项到显示或只有被分配的参数或所有的可能工程参数。 如果Show groups 控制盒未加抑制, 参数窗口将会在平台的目录中显示工程参数如上面的 screenshot 所显示的。 如果它被选中,



那么参数在树结构中更换将会被分类。（在下面看screenshot）

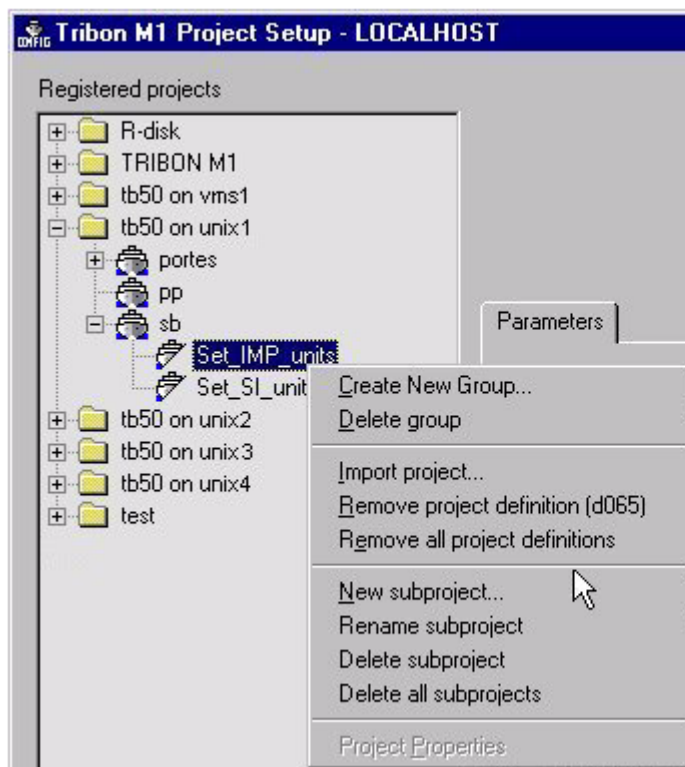
**注意：** 在 TDM 工程中, Tribon 工程用Oracle储存, 数据库变量改变值是不可能的。 因为有关 TDM 的细节, 请查阅 TDM 文件。



图为 分类参数视图

功能例如创造工程组集, 增加或删除子工程通过右鼠标按钮菜单。 当右击在 *Registered projects* 树视图中, 菜单将会显示。 请注意在菜单内的内容改变依赖于正面选的项目类型。 子工程功能 *Rename* 子工程和 *Delete subproject*, 举例来说, 仅仅有关子工程的上下文。 如果正面选一个工程组或一个工程, 这些菜单项目是残废的。





图为 正确的鼠标-菜单

Create new group...

这将会产生工程组特性对话，创建一个新的工程组，如上面例子所示。

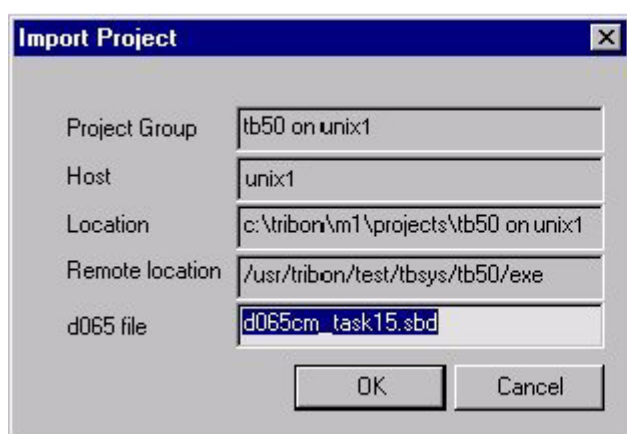
Delete group

将会删除选择的工程组。 请注意这将不删除任何的文件；它仅仅告诉被指定的工程服务器不服务这个工程组。

Import project...

这个菜单项目只是为遥远的工程组能够。 它使它可能用手从遥远的服务器到被指定的工程服务器机器转移一种特定的工程定义文件。 如果一个工程定义文件在遥远的服务器上用手修改，这可能是有用的。（也就是不经过 Tribon M2 工程设置程序）它也能用来对被指定的工程服务器执行新的工程定义文件，在遥远的服务器上创建，看得见。工程定义将会输入通过键入d065- 文件的名字在 *Import Project*对话框中：





图为 Import project

Remove project definition (d065)

这个菜单项目仅仅为工程和子工程。 它将会从被指定的工程服务器机器上的工程组目录中删除 d065- 文件。

Remove all project definitions

这个菜单项目是为工程,子工程 和工程组。 它将会从被指定的工程服务器机器上的工程组目录中删除所有的 d065- 文件。

New subproject...

这个菜单项目仅仅为工程和子工程。 它将会提示用户子工程名字和用那个名字创建新的子工程。

Rename subproject

这个菜单项目仅仅为子工程。 它将会提示用户一个新的子工程名字。

Delete subproject

这个菜单项目仅仅为子工程。 它将会删除现在选的子工程。

Delete all subprojects

这个菜单项目仅仅为工程和子工程。 它将会从现在选择的工程中删除所有的子工程。

## 1.5.2 Tribon M2 环境设置 Tribon M2 Environment Setup

Tribon M2 环境设置程序列出 Tribon 环境变量, 由系统变量和用户变量所组成。 系统变量是非工程变量, 无论选择的工程都是相同的。 当一个工程被选择的时候, 然后它的全部变量被定义为用户变量。 换句话说, 变量在用户环境桌面对用户是看得见的变量。(举例来说从 tbprintenv.exe)。 荧屏底部的控制盒决定如果当一个工程被选择的时候, 系统或工程变量应该首先分配。 如果它是未加抑制的那么在系统桌面中的值将会是对用户是看得见的值, 如果相同的变量在工程中被定义, 没有关系。 相反, 如果选中那么工程变量复盖系统变量。 如果某工程使用符号的另外一个组合应用程序胜于所有其他工程, 那么



SBB\_SYMBDIR 可以在工程中被重新定义使它不考虑系统变量 SBB\_SYMBDIR 。

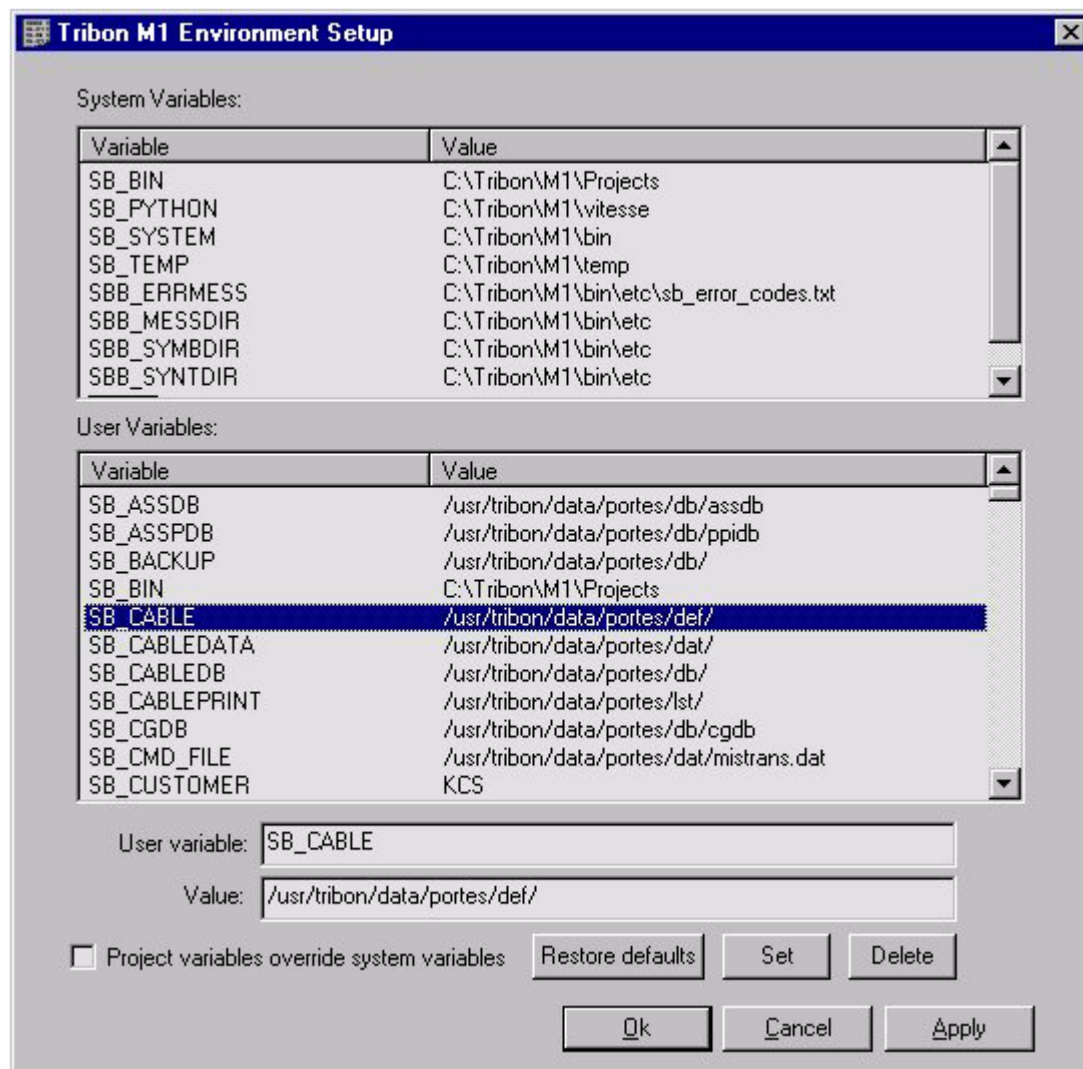


图 为 Tribon M2 环境设置

增加或改变环境变量, 在目录中按它的名字或在Variable盒子中键入一个名字, 类型值在Value中然后点Set。

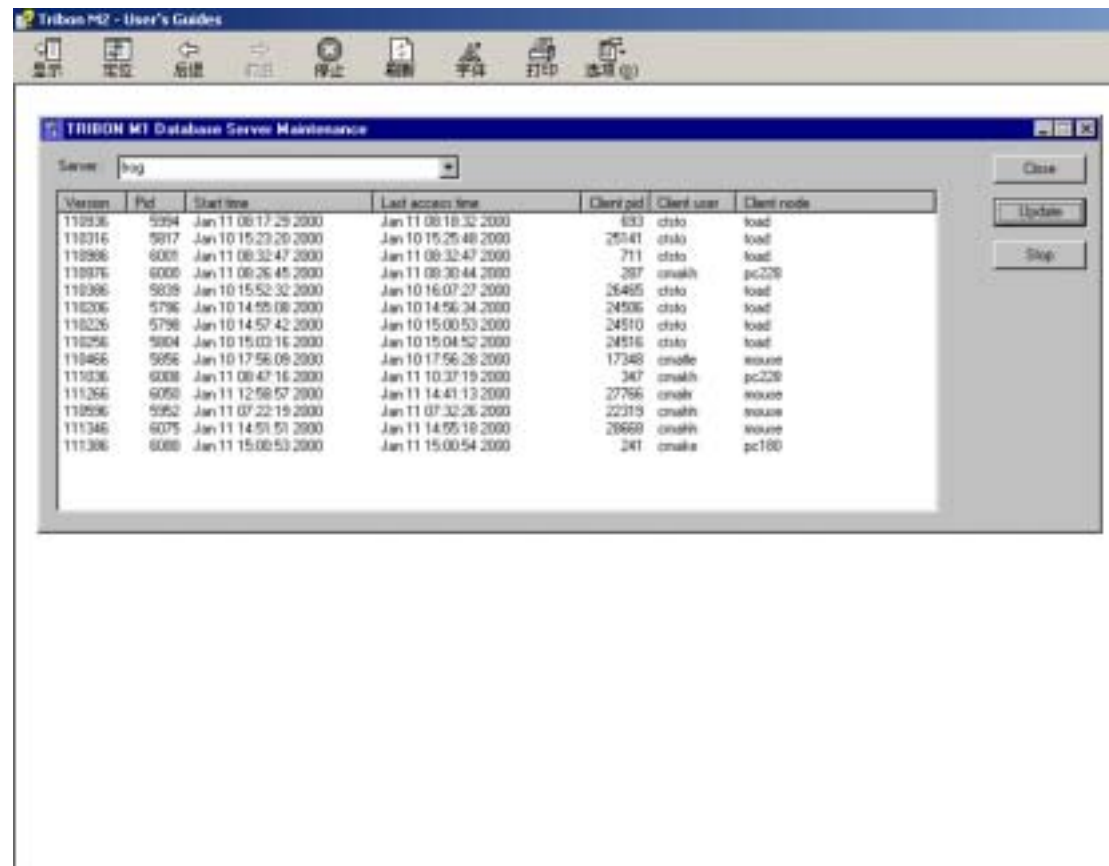
**注意:** 在 TDM 工程中, Tribon 用Oracle储存, 它是不可能换数据库变量的值。有关 TDM 的细节, 请查阅 TDM 文件。

## 1.5.3 Tribon M2 数据库服务器的维护 Tribon M2 Database Server Maintenance

依照比较早地在这份文件中讨论, 每一次一个 Tribon 应用程序访问服务器上的一个数据库, 一个数据库 subserver 程序被开始, 那将会服务应用程序这个



特别的要求。当应用程序是关闭的时候, subserver 将会自动地被停止。在稀罕的环境下, 举例来说当一个应用程序由于一个错误停止的时候, 它可以发生数据库 subserver 程序不被结束。Tribon 应用Tribon M2 数据库服务器维护, 然后能用来检查数据库服务器机器的状态, 和停止旧的废数据库 subservers。



图为 Tribon M2 数据库服务器维护。

注意: 在网络中运行这个工程任何的节点是可能的, 不仅仅在数据库服务器机器上。

Server 给数据库服务器机器的名字

Update

使视图更新。subservers 当前地运行在服务器机器上列出。目录包含关于subserver的资讯, 何时被开始, 最后的访问时候, 谁开始了它和从哪一机器。通过指示对应的专栏标题分类数据是可能的。

Stop

选择数据库服务器在目录窗口中停止, 和按stop按钮。这将会停止服务器



机器上的数据库 subservers 。

#### 1.5.4 Tribon 表面服务器 Tribon Surface Server

Tribon 现在能够朝二种类型船体表面系统之一工作, Tribon 初始设计和 NAPA。对分别的船体表面系统的访问经由一个表面的服务器执行, 是应用程序涉及一部服务器机器, 和到哪一应用程序连接取回表面的数据。服务器必须在表面的数据库居在地的网络节点上运行, Tribon 应用程序经由 ONC RPC 以一相似的方式与服务器沟通, 如 Tribon 数据库服务器。对一个表面数据库的访问总是经过 Tribon 表面服务器, 也意味着在一个单机中操作单一客户 Windows 安装, 表面服务器必须装在机器运行 Tribon。这情况相同的机器都会运行如服务器和客户。

哪个对使用表面系统 (s) 在安装被决定。选择的表面服务器 (s) 将会安装如 Windows 服务。在安装时候, 服务配置用手工开始, 它将会在使用之前需要适当设置。这个配置用一个工具处置叫执行 Tribon M2 表面服务器维护 (tbsurfservmaint.cpl), 包含在 Tribon M2 控制面板中。

**注意:** 请注意这个维护工程必须在 Tribon 表面服务器服务, 安装的目的机器上运行。

##### 1.5.4.1 Tribon 表面服务器初始设计 Tribon Initial Design Surface Server

Tribon 初始设计服务器的配置在安装程序的首页中被执行。

#### Surface directory

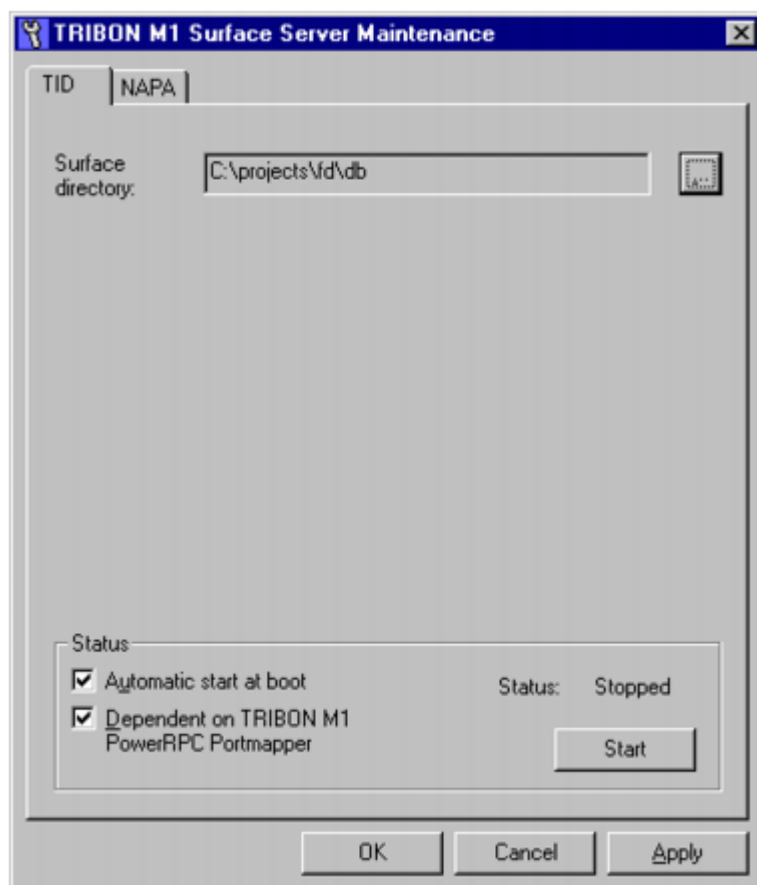
必需的参数。这是表面数据库位于的目录。**注意**在那时它从一个个目录对服务器操作表面是可能的。这意味着如果表面要在不同的位置中被保持, 和同时被用, 超过一部表面服务器机器必须被用。这是因为一个服务器机器仅能一个表面服务器主机。

#### Automatic start at boot

如较早提到的, 服务在按装时为手工配置开始。当服务有适当地配置的时候, 它更方便在导入时自动地开始它。



(如果示范工程 ds4 在安装 Tribon 表面服务器被选择将会开始和配置。然而，在导放时自动开始。 客户必须手工改变配置)



图为Tribon初始设计表面服务器的设置页

## Dependent on Tribon M2 PowerRPC Portmapper

正常地这个盒子应该总是被选中。 对不选中盒子的唯一理由是，如果 portmapper 已经在机器上是运行的,在哪一情况Tribon 递送不完全的安装。 如果盒子不被选中， 它强烈地推荐配置表面的服务用手工开始而不是自动。 这是因为 Tribon 在这情况不能够确定 portmapper在企图开始表面服务之前运行。

### Start/Stop

如果设定是满意的，服务是仅仅为一些理由停止,开始按钮能用。 如果一些设置已经改变，apply或OK应该被用。在这情况，现在的设定将会在窗口记入中被储存,服务将会停止和然后重新反映新的设定。 如果服务正在运行,用这个按钮停止它是可能的。



## 1.5.4.2 NAPA 表面服务器 NAPA Surface Server

NAPA表面服务器在安装运行期间被选择, 第二页将会在工程中出现, 允许那个服务器的配置。

图为NAPA表面服务器设置页

NAPA表面服务器服务能用个别的设定操作达到 10个NAPA表面服务器。 这个功能提供允许对一个服务器主机上的一些NAPA表面工程访问。 如NAPA表面服务器服务开始, 那些定义的NAPA表面服务器将会被开始。

NAPA , NAPAPROJDB , NAPA\_GMTOL , NAPA\_CGRID , NAPALICENSE , NAPA\_LOG, NAPA\_PRINT

这些是NAPA用的环境变量。 请为这些变量的解释查阅NAPA文件。

**注意:** 为了可以运行你必须获得从NAPA来的执照NAPA表面服务器。 这不同 Tribon 一起递送。 不超过一个激活的服务器应该连接到相同的NAPA工程和NAPA变量定义的log-file , NAPA\_PRINT 不应该在不同的服务器之间被共享。

**Automatic start at boot**





如较早地提及, 服务在装时为手工配置开始。当服务适当地配置的时候, 在导入时更方便自动地开始。

#### **Dependent on Tribon M2 PowerRPC Portmapper**

正常地这个盒子应该总是被选中。对不选中盒子的理由是, 如果 portmapper 已经是运行在机器上, 在哪一情况 Tribon 递送将不完全安装。如果盒子不被选中, 强烈地推荐配置表面的服务用手工开始而不是自动。这是因为 Tribon 在这情况不能够确定 portmapper 在企图开始表面服务之前运行。

#### **Start/Stop**

如果设定是满意的, 服务因为一些理由停止, 开始按钮能被用。如果一些设定虽然改变, apply 或 OK 应该被用来替代。在这情况, 当前的设定将会在窗口记入中被储存, 服务将会被停止和然后重新开始反映新的设定。

如果服务正在运行, 用这个按钮停止它是可能的。

一些 Tribon 环境变量规则怎么应用连接选择的表面服务器。 这些在 d065<project>.sbd 文件中放置。

SBB\_SURFACE\_SYSTEM 是在用的 TID 表面系统中设置 bmt\_1 和设置 napa\_1 如果 NAPA 表面系统被用

SBB\_SURFACE\_SERVER\_HOST 是设置表面服务器正在运行的服务器的机器名字。

SBB\_SURFACE\_SERVER\_NUMBER 是设置适当的 NAPA 表面服务器数字 (1- 10)。如果变量不设置, 1 的值是假定的。

当使用 TID 表面服务器的时候, 这个变量无关。

在 d065<project>.sbd 文件中的条目例子:

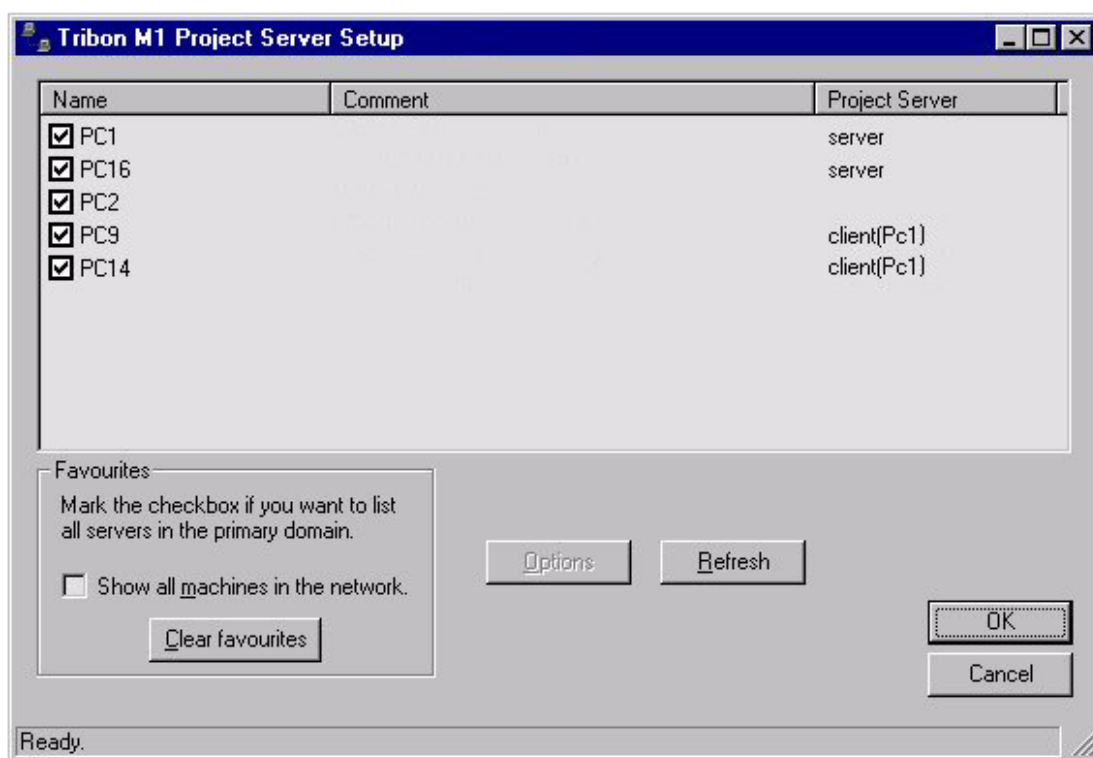
例子:

SBB\_SURFACE\_SYSTEM bmt\_1

SBB\_SURFACE\_SERVER\_HOST the\_surface\_server

### **1.5.5 Tribon M2 工程服务器设置 Tribon M2 Project Server Setup**

叫执行 *Tribon M2 Project Server Setup* 的 Tribon M2 控制面板程序能用来很远地配置工程服务器设定在全部网络 Windows 机器上。然而, 这需要用户有所有机器上的管理特权。用户可选择一或较多的机器在列表中, 通过点击 *Options* 按钮调出 *Project Server Options* 对话框。如果用 OK 选项对话关闭, 设定将会适用对于在目录中所有选择的机器。

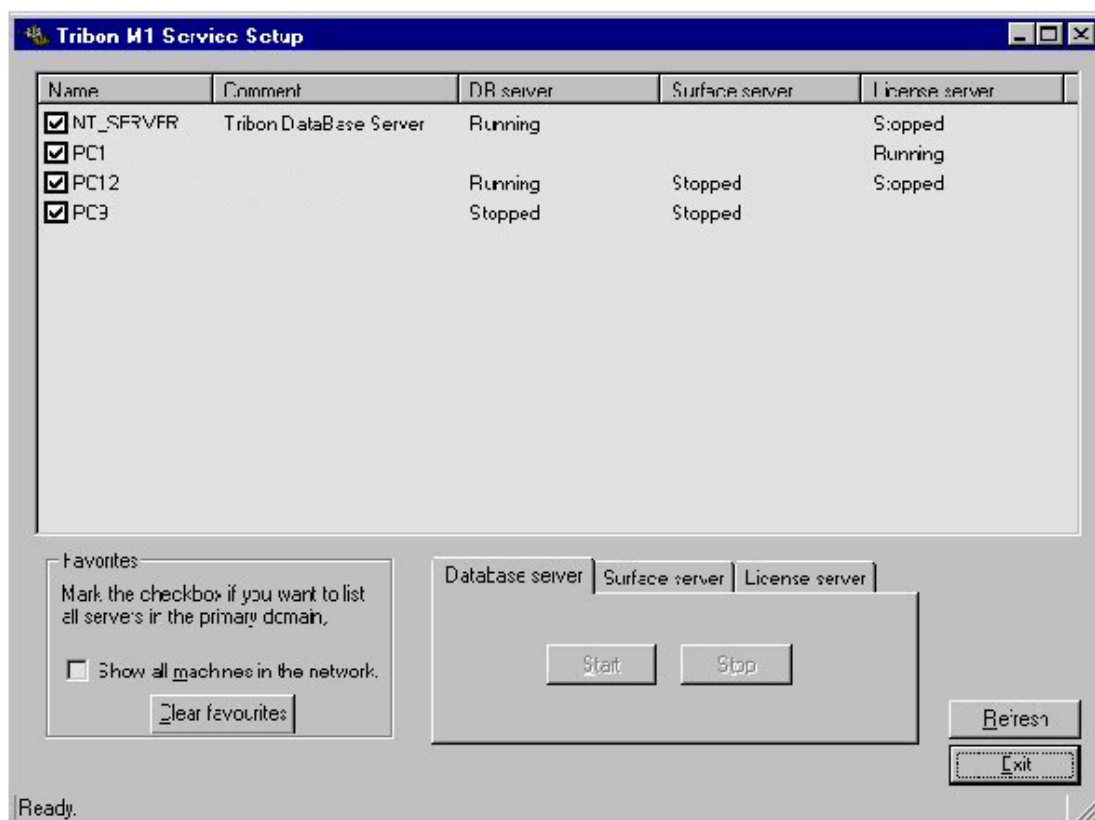


图为 Tribon M2 工程服务器设置

## 1.5.6 Tribon M2 服务设置 Tribon M2 Service Setup

Tribon M2 服务设置程序能用来得到 Tribon 服务的网络概观。在另外一部机器上很远地开始和停止服务是可能的,但是这需要用户有那部机器上的管理特权。





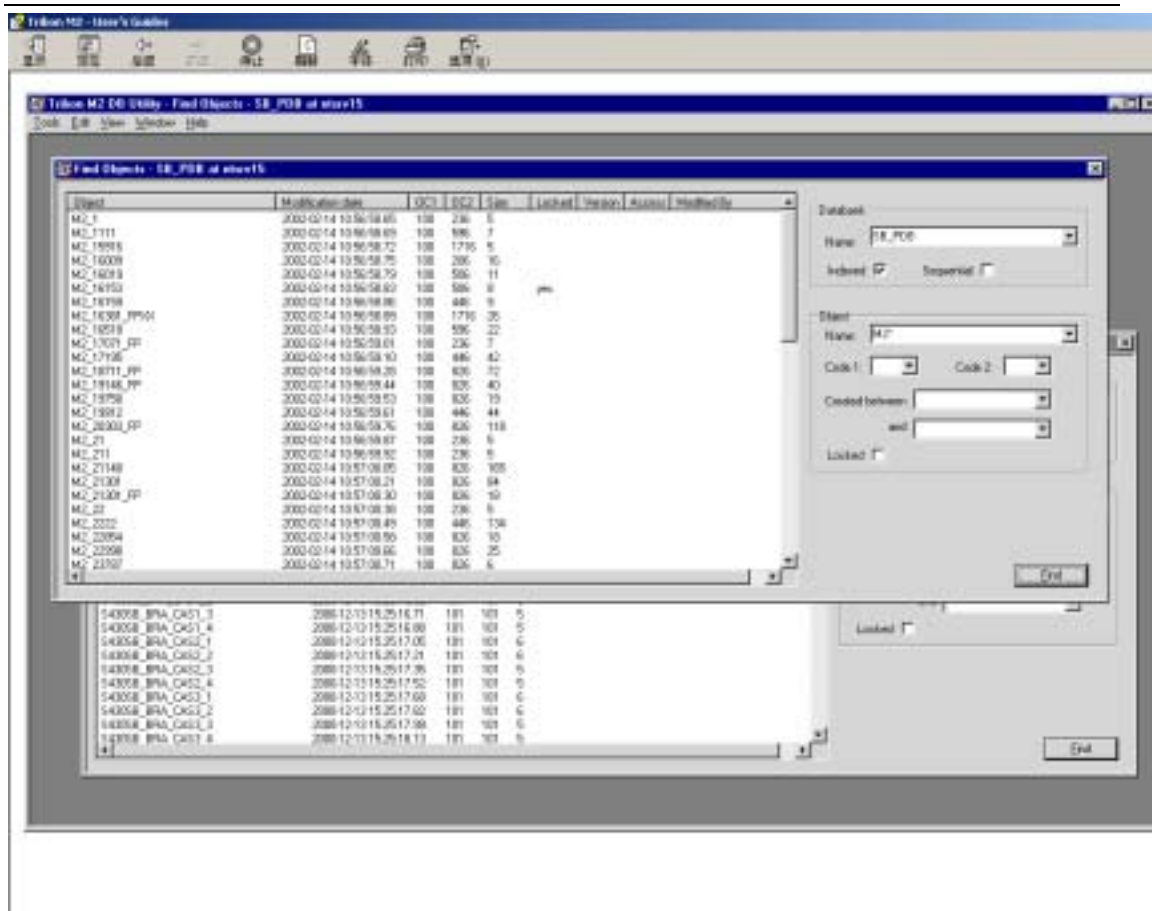
图为Tribon M2 服务设置

## 1.5.7 Tribon M2 数据库效用 Tribon M2 Data Bank Utility

程序是Tribon 目标数据库的一个有用程序。(indexed, sequential或Oracle) 应用程序有二主要功能访问通过工具菜单。 功能是找目标和数据库维护.(仅仅编入索引数据库)

### 1.5.7.1 查找目标 Find Objects

程序部份有多样- 文件界面，在一父窗口里维持多个子窗口。 到子窗口，连接任何的数据库和通过cut – copy在数据库中对目标进行操作是可能的，在不同的子窗口之间粘贴操作是可能运行的，也就是在数据库之间。



图为 Tribon M2 效用

专栏Object , Modification date , OC1 , OC2, Size和Locked对所有类型目标和数据库是有用的, 当 Tribon 数据管理解决被用的时候, 专栏Version, Access和Modified是有效的。

按查找按钮, 一个搜寻操作开始, 如依照搜寻标准被叙述:

Data bank name

被操作的数据库名字。

在传统的 Tribon 工程中名字可能是一条完整的文件- 路径到编入索引或连续的数据库和一个 Tribon 环境变量适当的设定。 如此的一个数据库被访问或从本机文件系统或经过被描述在 1.3.4 数据库中来自应用程序的访问一个 Tribon 数据库服务器。 子窗口标题文本包含主机的名字, 为给定的数据库当作数据库服务器用。

在 TDM 中激活工程, 使用Oracle如储藏Tribon 目标, Tribon 环境变量识别数据库必须被用。 这些数据库变量 (举例来说 SB\_OGDB , SB\_CGDB , SB\_PDB, SB\_PSDB 及其他) 不应该是用户明确地设置, 但是会仍然在表示关系数据库里面被视为使用适当选择适当的子 Tribon 目标的组合。

Object name

给明确的目标名字或有广泛性的一个名字。一般承认广泛性的是星号 '\*' 字符和百分比 % 告示的一个字符，不给是 ' % '。

**Object code 1**

用叙述的目标编码1对待目标。

**Object code 2**

用叙述的目标编码2对待目标。

Created between date

在这日期之后创建的目标被操作。当前的日期被保持在名册列表中当作语法的一个例子

and date

在这日期之前创建的目标被操作。当前的日期被保持在名册列表中当作语法的一个例子。

Locked

仅仅锁的目标被操作。

下列各项操作从编辑菜单或在菜单上面取出能被运行：



Copy - Paste - Paste special



目标在数据库之间拷贝。粘贴专门保持目标(s)最初的储存日期。

**注意:** 请注意在 TDM 中激活工程,使用Oracle储存Tribon 目标, 在同一工程中数据库之间拷贝目标是只可能的, 或来自连续数据库。

Cut - Paste - Paste special

目标在数据库之间被移动。粘贴专门保持目标(s)最初的储存日期。

**注意:** 请注意在 TDM 中激活工程,使用Oracle储存Tribon 目标, 在同一工程中数据库之间移动目标是只可能的, 或来自连续数据库。

在工具菜单中通过选项功能,决定是否创建日期应该被保持或对于被操作的目标当前的日期应该设置是可能的

Delete

目标从一个数据库删除。

**Rename**

目标在一个数据库中被重新命名。通过组名字详述重新命名一定范围的目标是可能的。 一个组名字给零或被一个星号 ‘ \* ’ 跟随的一些字符。

Unlock

开启在一个数据库中锁的目标。 功能必须被最大的注意,仅仅当一种错误情形引起锁知道的时候。

Split

在用户叙述的目录中创建来自一个被编入索引数据库的每个目标的一个连续的数据库和设置他们

Merge

在一个由用户叙述的目录中取回来自连续数据库的目标, 储存目标在一个索引数据库中

Clear

将会清除列出的目标。

## 1.5.7.2 数据库维护 Data Bank Maintenance

这个部份程序操作 Tribon 编入索引数据库的创建和备份/ 恢复。 数据库的最大大小限制在 2 Gbyte 。

**注意:** 备份/ 恢复一个不是本机的数据库是可能的,这个工程运行仅仅是服务器操作的机器是Windows机器。 UNIX 或 VMS 机器的遥远备份是不可能执行的。



## Create

The screenshot shows a Windows-style dialog box titled "Tribon M1 Database Maintenance". It has three tabs: "Create", "Backup", and "Restore". The "Create" tab is active. Inside the dialog, there are two input fields: "Name:" and "Directory:". The "Directory:" field has a small browse button (represented by a folder icon) to its right. At the bottom of the dialog, there are three buttons: "OK", "Cancel", and "Apply".

该页操作新的数据库的创建。新的数据库的名字必须给, 目录该在哪里创建, 能被键入或经由浏览发现。

## Backup



**Tribon M1 Database Maintenance**

Create Backup Restore

Databank:  ...

Backup directory:  ...

Backup:  ▼

List file ☐

OK Cancel Apply

该页操作数据库的备份。

备份数据库的拷贝是二个类型：总拷贝和增加的拷贝。总拷贝包含所有的目标在父数据库中储存。它也包含是必需重建的结构信息数据。总拷贝正常地用一相对地低的频率创建，举例来说一星期一次或一个月一次。增加的拷贝包含自从最后增加的拷贝以后被增加到目标父数据库的所有目标，如果所有的，或总拷贝被创建。它也包含在相同的时段运行期间删除的所有目标的列表。增加的拷贝用高频率创建，举例来说每天一次或每二天一次。

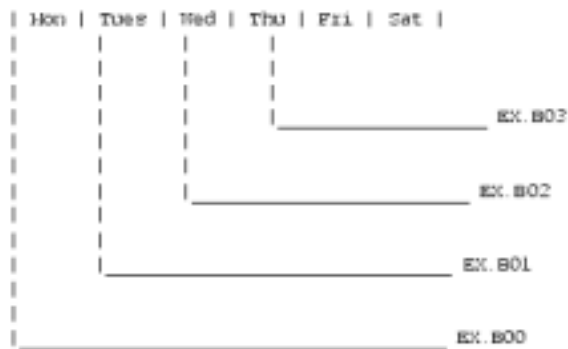
拷贝的文件名字是：<file>.B<I>，在那里

<file>是父数据库的名字，

和 <I>是备份的拷贝顺序号。

总拷贝是 00, 后来增加拷贝 01, 02, 及其他，因此，如果一个数据库名字 AGFR, 它的备份拷贝 1 号将会文件命名 AGFR. B01 。

当前, 让我们举例说明一个例子二类型备份拷贝的使用。数据库有名字 EX 。



如果父数据库星期五被破坏, 操作的下列各项序列:

1. 初始化新的数据库 EX。
  2. 取回 EX.B00 的内容。
  3. 取回 EX.B01 , EX.B02 和 EX.B03 的内容。
- 当 EX.B03执行的时候, 父数据库用它有的内容重建。 从增加的拷贝数据的取回包括删除的目标的列表, 这也被运行。

**backup**领域有三选项

## Total

这创建新的总备份数据库的拷贝

## Incremental

创建新的增加备份拷贝

## Ini t

重新初始化增量备份的计算, 不创建新的总备份。

如果List file被选中, 显示程序运行目录文件将会在备份目录中被创建。

## Restore

依照上方提到, 重建的数据库将会有如同最初同样的内容。这意味它包含相同的目标, 但是结构的数据可能有改变, 因为创建来自总拷贝的一个数据库的过程包括重组过程, 在数据库中一个释放空间将会被更有效的用。



**Tribon M1 Database Maintenance**

Create | Backup | **Restore**

Databank:  ...

Backup directory:  ...

First copy:  Last copy:

Object:

List file ☐

OK Cancel Apply

该页操作对一个数据库的备份拷贝的恢复

数据库名字和备份拷贝居住的目录必须给。 同样首先和最后的备份拷贝号码必须被定义。 如果仅仅某备份拷贝被恢复，首先和最后的备份拷贝号码可能是相同的。

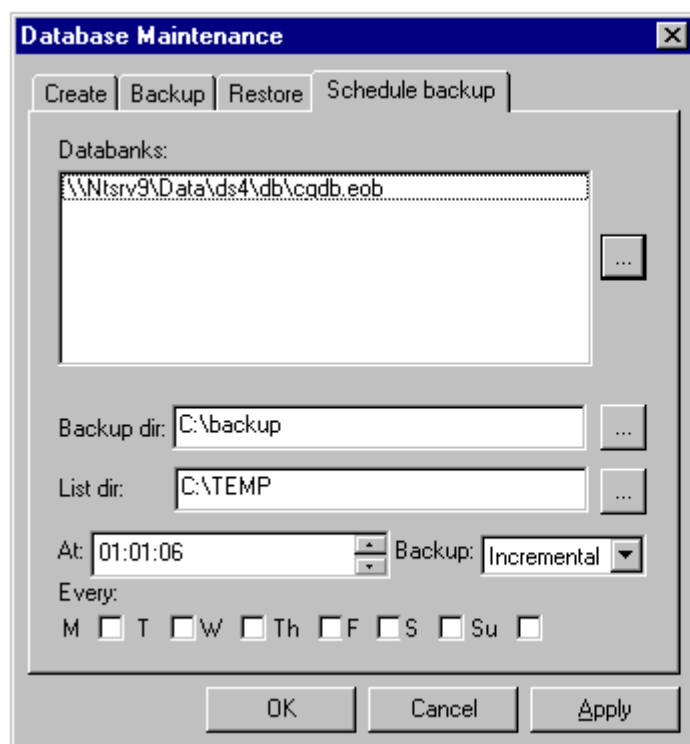
如果一个目标名字在**Object**领域中被定义，这工程将从备份的拷贝恢复。

如果**List file**盒子被选中，显示程序运行目录文件将会在备份目录中被创建。

### Schedule Backup

上方描述的备份功能，直接运行备份叙述的数据库的拷贝，有一扩展名使进度表备份有规律性是可能的。 这个特征需要时间表服务在工作交付给计算机开始。一旦备份被预定，它通过OS指令AT被控制。





## 1.5.8 Tribon 许可操作 Tribon License Handling

关于FLEXIm 管理器的资讯和 Tribon 执照配置能在 Tribon M2 安装指南中找到。

## 1.6 命令行效用 Command-line Utilities

虽然 Tribon M2 控制面板程序是使用首选工具, 选择功能通过命令行应用程序也是可得的。

### 1.6.1 数据库的效用 Data Bank Utilities

一个家属的工具为编入索引数据库的备份, 恢复和其他的类型维护被发展。这些工具在下面被描述。一些工具也能用于顺序数据库。(举例来说拷贝和目标的列表)

这些工具被认为是管理员工具, 不应该被正常的 Tribon 用户用。

**注意:** 推荐避免在数据库备份/重组运行期间被其他的 Tribon 应用程序访问数据库。



### 1.6.1.1 备份 Backup

备份数据库的拷贝是二个类型：总拷贝和增加的拷贝。总拷贝包含在父数据库中储存的所有的目标。它也包含来自总拷贝必要的结构数据信息建立数据文件。增加的拷贝包含所有被增加到父数据库的目标，自从最后增加的拷贝。如果所有的，或因此总拷贝被创建。它也包含在相同的时段运行期间删除的所有目标列表。

拷贝的文件名字是：<file>.B<l>，那里

<file>

父数据库的名字。

<l>

备份拷贝的运行号码。总拷贝是 00, 后来的增加拷贝 01, 02, 及其他

因此, 如果一个数据库名字 ogdb, 它的备份拷贝 1 号文件命名 ogdb.B01。现在, 让我们举例说明一个例子的二种类型备份的拷贝使用。数据库名字 ogdb。如果父数据库星期五被破坏, 它被操作下列各项序列:

1. 初始化新的ogdb数据库。
2. 取回 ogdb.B00 的内容。
3. 取回 ogdb.B01, ogdb.B02 和 ogdb.B03 的内容。

当 ogdb.B03 被执行的时候, 父数据库用它有的内容重建。来自增加拷贝数据的取回, 包括删除的目标目录, 这也被运行。

### 1.6.1.2 重组 Reorganisation

依照上方提到, 重建数据库将会有如同最初的一般内容。这意味它包含相同的目标, 但是结构数据可能被改变, 因来自总拷贝的一个数据库的创建过程包括一个重组过程, 在数据库中释放的空间将会更有效的用改组程序。一个结论是一个数据库被接着的步骤重新组织:

1. 创建总拷贝。
2. 初始化一个新的数据库。
3. 写总拷贝的内容给新的数据库。

### 1.6.1.3 程序 Programs

数据库有用程序为编入索引的数据库在下面被描述。

#### SA001- 数据库的备份



工程创建数据库的总拷贝和增加的拷贝。

**注意：** 请注意程序必须在数据库磁盘实体被装好的计算机上运行。

对工程的输入经由合格的指令

/ DBNAME=name            数据库的名字。  
 [/DBDIR=directory]      数据库的目录。  
 [/BUDIR=directory]      备份文件目录。

如果数据库或备份被储存在工程进行的目录中，目录名字能被省略。

	TOTAL	总备份
/BACKUP=INCR		增加的备份
	INIT	开始新的增加备份不取总备份
[/ LIST]	拷贝目标的列表在文件 sa001.lst 生产。	
[/ LOG]	拷贝目标在终端机上被列出。	

仅仅三个限定的主要特征是有关的。

错误讯息在标准输出装置上被给，正常地终端从程序开始的地方。讯息是不需加以说明的。

## SA002- 重新组织数据库

工程重建来自许多的备份拷贝的数据库或创建一个数据库通过删除。

**注意：** 请注意程序必须在数据库磁盘实体完好的计算机上运行。

对程序的输入经由限定指令给出。

/ DBNAME= name            数据库的名字。  
 [/DBDIR=directory]      数据库的目录。  
 [/BUDIR=directory]      备份文件目录。

如果数据库或备份被储存在工程进行的目录中，目录名字能被省略。

/ STATUS = NEW  
 一个新数据库将会被初始化和旧的拷贝，如果所有的，将会拷贝数据库。  
 / STATUS = OLD  
 备份的拷贝将会拷贝到一个已存在的数据库。

编入索引数据库的最大大小是 4194304 512 字节块或 2GB。

[/COPNO=no1, no2]

备份拷贝数。 no1 是首先和 no2 后来者，分别地，在拷贝的序列中。 0 是总拷贝和 1, 2, 3... 及其他是增加的拷贝。

no1 可能和 no2 相等。

如果 no1=-1 或如果 COPNO 缺少的，没有备份拷贝被读。



[/ OBJECT = name]

仅仅从给的备份文件中给定的目标被转移

[/LIST]            转移目标的列表在文件 sa002.lst 被产生。

[/LOG]            拷贝目标在终端被列出。

**注意：**    限定句在逻辑组合中必须被给出。

如果，举例来说 no1=0 在 /COPNO之后, 只有当一个新的数据库被初始化的时候，既然总拷贝的取回，/SIZE同样被给。

If /STATUS=NEW            no1 必须是 0 或 -1. ( 或缺少的)

If /STATUS=OLD            no1 必须是 >0.

错误讯息在标准的输出装置上被给，正常地终端从程序被开始的地方。 讯息是不需加以说明的。

## SA004- 数据库效用Data Bank Utility

程序是对于( 编入索引或顺序的)数据库一个效用程序。 对工程的输入程序要求是交互式给出。

下列各项操作能被运行:

**Copy**            目标在数据库之间被拷贝。

拷贝的目标可以被给不同于最初目标名字的新名字。

**Move**            目标在数据库之间移动。

移动的目标可以被给不同于最初的目标名字的新名字。

**注意：**    请注意从一个顺序数据库中移动目标不可能的。

**Delete**            目标从一个被编入索引的数据库删除。

**List**            目标在数据库中列出。

**Archive**        目标在不同的数据库之间移动，同样通过操作移动，但是目标的编制日期被保留。

如果数据库被储存在另外一个目录 ,从那一工程被运行, 目录必须被分开叙述。

目标能依照下列的情况被叙述，能被任意地结合。

## Object name

你能给一个明确的目标名字，一组名字或一个有通配符名字。

在一个名字中用取配符，一个星号能代替许多字符和一个百分号能代替一个字符。 例子:

TB-272-\* -1



\*272\*

TB%72\*

一个组名字被给通过零或一个星号跟随的一些字符。 例子:

TB272\*

TB\*

当一个明确的目标名字 (没有通配符) 或一个组名字被给的时候, 由如最初的名字同样地给一个新的名字重新命名目标 (s)是可能的。 例子:

Object name TB272\*

New object name

TB114\*

Object code 1            拥有指定目标代码 1 的目标被操作。

Object code 2            拥有指定目标代码 2 的目标被操作。

After date                在这日期之后创建的目标被操作。

Before date               在这日期之前创建的目标被操作。

错误的信息在标准的输出装置上给出, 正常地终端从程序开始的地方。 信息是不需加以说明的。

执行的操作清单在一个被命名 sa004.lst文件中产生。

## SP304 – 目标设施

程序 sp304执行Tribon 目标的操作, 像列表和开锁他们。 它是一个控制台应用程序, 通过问问题和提示用户输入交互影响。 工程被设计用于除去错误, 并从错误情形恢复。

Data Bank Name

当工程开始, 用户被要求键入数据库的名字取得目标和信息。 逻辑名字被用。

Cmd

当提示 'Cmd: ' 被显示, 通过键入HELP可用的指令能被列出。 不同的指令在下面被描述。 所有的指令能被缩写到一个级别, 而它们仍然是唯一的。

HELP

HELP指令在荧屏上列出可用指令, 有指令的短描述。

EXIT

EXIT指令将使程序退出。

OUTPUT

目标清单被执行缺省在荧屏上, 但是用户能通过使用OUTPUT指令列出清单文件。 文件的名字提示为. Not, 不规定输出文件名。

LIST

在当前的列表文件中LIST指令抑制目标的数据(看上面OUTPUT)。 项目表格式符合在系统描述方面元素定义, 在这里没有较进一步解释。列表主要地想要由系



统管理人学习当由于错误调试，测试等等。程序提示目标名，直到没有名字进入。

BRIEF

BRIEF指令抑制场目标数据如同LIST指令，没有数据分配。

STATUS

如果目标在数据库中被锁，当选中的时候，STATUS指令能用为开锁目标。指令使用时必须十分当心，仅当目标由错误而锁时。程序保持提示目标名字直到没有名字进入。通配符\*，%(星号，百分号\*表示 0到 n 个的任何字符和%意谓任何字符中的 1个。不被锁的目标在荧屏上被列出，锁的目标将使程序问用户是否解锁。

接着的段落展现 sp304 的所有可用的指令在用的工程运行期间。

数据库名字: SB\_PSDb

Cmd: HELP

-- Commands -----

-- EXIT exit program --

-- HELP display this text --

-- BRIEF list object, without data part --

-- LIST list object, including data part --

-- OUTPUT redefine list file of object --

-- STATUS display object status and optionally unlock --

-----  
Cmd: OUTPUT

Filename: dump.log

%SP304-I-LISTFILE, the new list file is dump.log.

Cmd: LIST

Objectname: SB=230=M101

%SP304-S-LIST, SB=230=M101 listed.

Objectname:

Cmd: STATUS

Objectname: SB=230=M101

%SP304-I-NLCK, SB=230=M101 is not locked.

Objectname: SB=\*CS\*

%SP304-I-NLCK, SB=330=CS1 is not locked.

%SP304-I-NLCK, SB=330=CS17 is not locked.

%SP304-I-NLCK, SB==CS19 is not locked.

%SP304-I-EXTLCK, SB==CS9 is locked by an other user.



---

Forced unlock (Y/N): Y  
%SP304-I-NLCK, SB==CS9 is not locked.  
Objectname:  
Cmd: EXIT

## 1.6.2 Tribon 环境效用 Tribon Environment Utilities

tbprintenv [var]	打印 Tribon 环境变量的值
tbsetenv var new_val	对 Tribon 环境变量赋值。
tbunsetenv var	删除 Tribon 环境变量的值。