



脚本化语言

用于 **PHP** 的 **Adaptive Server**[®]
Enterprise 扩展模块 **15.7**

文档 ID: DC01816-01-1570-02

最后修订日期: 2012 年 6 月

版权所有 © 2012 Sybase, Inc. 保留所有权利。

除非新版本或技术声明中另有说明, 否则本出版物适用于 Sybase 软件及所有后续版本。本文档中的信息如有更改, 恕不另行通知。本出版物中描述的软件按许可证协议提供, 其使用或复制必须符合协议条款。

仅在定期安排的软件发布日期提供升级。未经 Sybase, Inc. 事先书面许可, 本书的任何部分不得以任何形式、任何手段(电子的、机械的、手动、光学的或其它手段)进行复制、传播或翻译。

可在 <http://www.sybase.com/detail?id=1011207> 上的 Sybase 商标页中查看 Sybase 商标。Sybase 和列出的标记均是 Sybase, Inc. 的商标。® 表示已在美国注册。

SAP 和此处提及的其它 SAP 产品与服务及其各自的徽标是 SAP AG 在德国和世界各地其它几个国家/地区的商标或注册商标。

Java 和所有基于 Java 的标记都是 Oracle 和/或其在美国和其它国家/地区的附属机构的商标或注册商标。

Unicode 和 Unicode 徽标是 Unicode, Inc. 的注册商标。

本书中提到的所有其它公司和产品名均可能是与之相关的相应公司的商标。

Use, duplication, or disclosure by the government is subject to the restrictions set forth in subparagraph (c)(1)(ii) of DFARS 52.227-7013 for the DOD and as set forth in FAR 52.227-19(a)-(d) for civilian agencies.

Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

目录

用于 PHP 的 Adaptive Server Enterprise 扩展模块	1
安装用于 PHP 的扩展模块	2
配置概述	2
示例 PHP 脚本	3
用于 PHP 的扩展模块的 API 参考	3
sybase_affected_rows	3
sybase_close	4
sybase_connect	4
sybase_data_seek	5
sybase_fetch_array	5
sybase_fetch_assoc	6
sybase_fetch_field	6
sybase_fetch_object	7
sybase_fetch_row	7
sybase_field_seek	8
sybase_free_result	8
sybase_get_last_message	8
sybase_get_last_status	9
sybase_next_result	9
sybase_num_fields	9
sybase_num_rows	10
sybase_pconnect	10
sybase_query	11
sybase_rpc_bind_param_ex	11
sybase_rpc_execute	12
sybase_rpc_init	12
sybase_select_db	13
sybase_set_message_handler	13
sybase_unbuffered_query	14
sybase_use_result	14
安全性和目录服务	15

目录

其它资源	15
词汇表	17
索引	19

用于 PHP 的 Adaptive Server Enterprise 扩展模块

用于 PHP 脚本化语言的 Sybase® Adaptive Server® Enterprise 扩展模块允许 PHP 开发人员针对 Adaptive Server 数据库执行查询。

用于 PHP 的扩展模块是特定于供应商的数据库接口驱动程序，用以针对 Adaptive Server 数据库执行查询。开源编程语言 PHP (Hypertext Preprocessor) 能让您从一般数据库中检索信息。用于 PHP 脚本化语言的 Adaptive Server 扩展模块包括必需的 PHP API，以使 PHP 开发人员能够编写独立脚本来针对 Adaptive Server 数据库执行查询。

用于 PHP 的扩展模块已经针对命令行 PHP 实施进行了测试。扩展的使用（例如，在 Web 服务器中使用 CGI 模块）超出了 Sybase 测试的范围。

PHP 扩展数据流

下图显示从 PHP 脚本到 Adaptive Server 数据库的数据流。



必需组件

使用 PHP 编程语言访问 Adaptive Server 数据库需要以下组件：

- PHP 脚本 - 用以连接到 Adaptive Server 数据库服务器的应用程序脚本。
- 用于 PHP 的 Adaptive Server Enterprise 扩展模块 - 是 Sybase 支持的、特定于供应商的驱动程序。此扩展是通过 PHP 脚本调用的，并连接到 API 层 DBCAPI。
- DBCAPI - 是一个函数库，充当 PHP 扩展和 CT-Library 之间的中间转换层。
- CT-Library - 是一个 Sybase Open Client 库，可用于向 Adaptive Server 发送命令、处理结果并返回数据。

版本要求

有关平台支持的信息，请参见所用平台的 Software Developers Kit and Open Server Installation Guide（《软件开发工具包和 Open Server 安装指南》）。

- Adaptive Server Enterprise - 15.7 版或更高版本。此 PHP 驱动程序是针对 15.7 版开发和测试的。但是，此驱动程序可以连接到低于 15.7 版的 Adaptive Server。
- PHP 安装 - 5.3.6 版。
- Open Client SDK 中的 CT-Library - 15.7 版。
- DBCAPI - Sybase 建议您使用同一 SDK 安装中的 DBCAPI 库和 PHP 驱动程序。

安装用于 PHP 的扩展模块

用于 PHP 的扩展模块是一个可通过 Sybase 安装程序安装的组件。

当您选择“自定义”(Custom)作为安装类型时，用于 PHP 的扩展模块是可选的安装组件。如果您选择的安装类型是“典型”(Typical)或“完全”(Full)，则会缺省安装此扩展。

这是扩展模块安装的概述；有关完整安装说明，请参见所用平台的 *Software Developers Kit and Open Server Installation Guide*（《软件开发工具包和 Open Server 安装指南》）。

- 预安装要求：
 - PHP 安装 5.3.6 - 仅限 64 位。
 - Open Client SDK 是必需的。DBC-API 库包括在 Open Client SDK 中，将和核心 Open Client (CT-Library) 一起安装。

不要将 DBC-API 作为驱动程序的一部分安装。

- 用于 PHP 的扩展模块将安装为动态库中的 PHP 扩展。

Sybase 安装程序将此扩展模块作为一个名为 `sybaseasephp.so` 的共享库安装到 Open Client (CT-Library) 目录中。

- 以下文件将随此扩展模块一起安装：

```
$SYBASE/$SYBASE_OCS/php/php536_64/lib/sybaseasephp.so
$SYBASE/$SYBASE_OCS/php/php536_64/devlib/sybaseasephp.so
$SYBASE/$SYBASE_OCS/sample/php/README
$SYBASE/$SYBASE_OCS/sample/php/firstapp.php
$SYBASE/$SYBASE_OCS/config/generate_php_ini.sh
```

配置概述

配置环境以便定位此扩展模块和 OCS 安装。

环境变量

若要成功使用此扩展模块，请在运行 PHP 可执行文件的环境中设置 Sybase 环境变量。至少必须设置 `$SYBASE` 和 `$SYBASE_OCS` 环境变量。将 `LD_LIBRARY_PATH` 变量设置为指向 `$SYBASE/$SYBASE_OCS/lib` 目录下的正确位置。扩展模块会从这个位置装载 DBC-API 库。

示例脚本

会提供一个脚本，用以创建示例 `php.ini` 文件，以便 PHP 从安装目录中定位扩展模块。此脚本，即 `generate_php_ini.sh`，位于 `$SYBASE/$SYBASE_OCS/config` 中。必须正确设置 `generate_php_ini.sh` 脚本的 `$SYBASE` 和 `$SYBASE_OCS`，才能生成正确的 `php.ini` 文件。

可以使用生成的 `php.ini` 文件来运行 `firstapp.php` 示例或任何其它使用此扩展模块的测试。对于典型安装，`php.ini` 文件已经存在，而且示例 `php.ini` 的 `generate_php_ini.sh` 脚本中的信息会复制到现有 `php.ini` 文件中。

系统管理员可以通过修改 `php.ini` 文件，将扩展模块复制到其它特定于扩展的目录并从该目录中装载扩展模块，例如：

```
; Set the default extension directory.  
extension_dir = "/usr/local/lib/php/extensions"  
;Load the Sybase ASE PHP driver: sybaseasephp  
extension="sybaseasephp.so"
```

修改此文件后，便可以将 `$$SYBASE/$$SYBASE_OCS/php/php536_64/lib/sybaseasephp.so` 或 `$$SYBASE/$$SYBASE_OCS/php/php536_64/devlib/sybaseasephp.so` 库复制到 `/usr/local/lib/php/extensions` 中。

扩展库

当执行 `php -m` 时，应该会看到 `sybaseasephp` 扩展列在 PHP 安装中的活动扩展内。

注意： 无需将特定 PHP 安装的所有扩展库都放在同一目录中。

示例 PHP 脚本

示例脚本安装在 `$$SYBASE/$$SYBASE_OCS/sample/php/firstapp.php` 中。

有关运行示例脚本的信息，请参见位于同一目录中的 `README` 文件。有关平台安装信息，另请参见 `Software Developers Kit and Open Server Installation Guide`（《软件开发工具包和 Open Server 安装指南》）。

用于 PHP 的扩展模块的 API 参考

扩展模块界面 API。

sybase_affected_rows

返回 `$conn` 所引用的连接上的上次 **INSERT**、**UPDATE** 或 **DELETE** 查询所影响的行数。

此函数通常用于 **INSERT**、**UPDATE** 或 **DELETE** 语句。对于 **SELECT** 语句，请改用 `sybase_num_rows()` 函数。

语法

```
int sybase_affected_rows([resource $conn])
```

参数

- **\$conn** - 连接打开函数返回的连接资源。如果未指定连接资源，则会使用最近打开的连接。

返回

上次 **INSERT**、**UPDATE** 或 **DELETE** 查询所影响的行数。

sybase_close

关闭连接。

语法

```
bool sybase_close([resource $conn])
```

参数

- **\$conn** - 连接打开函数返回的连接资源。如果未指定连接资源，则会关闭最近打开的连接。

返回

成功时为 **TRUE**。

失败时为 **FALSE**。

sybase_connect

打开与 Adaptive Server 的连接。

语法

```
resource sybase_connect([string $servername  
[, string $username  
[, string $password  
]]) )
```

参数

- **\$servername** - 在相关 Sybase 目录服务中定义的服务器。
- **\$username** - 用于登录到 Adaptive Server 的用户帐户。
- **\$password** - 用于登录到 Adaptive Server 的用户帐户的口令。

返回

成功时为正连接标识符。

失败时为 **FALSE**。

sybase_data_seek

移动与结果标识符相关联的结果集中的内部行指针，使其指向指定的行号。

语法

```
bool sybase_data_seek(resource $result, int $row)
```

参数

- **\$result** - 调用 **sybase_query()** 所得的结果资源
- **\$row** - 设置内部指针指向的行号（从 0 开始）。

返回

TRUE - 定位内部指针成功。

FALSE - 正确设置内部指针失败。

sybase_fetch_array

以关联数组、数值数组或二者兼具的形式读取结果行。

语法

```
mixed sybase_fetch_array(resource $result  
[, int $result_type ] )
```

参数

- **\$result** - 调用 **sybase_query()** 所带来的结果资源。
- **\$result_type** - 要返回的数组的类型。它接受使用以下值返回数组：
 - SYB_FETCH_ASSOC - 关联数组
 - SYB_FETCH_NUM - 数值数组
 - SYB_FETCH_BOTH - （缺省）关联索引和数值索引

返回

\$result_type 为 SYB_FETCH_ASSOC 或 SYB_FETCH_BOTH 时，返回关联数组（与 **sybase_fetch_assoc()** 相同）。

\$result_type 为 SYB_FETCH_NUM 或 SYB_FETCH_BOTH 时，返回数值数组（与 **sybase_fetch_row()** 相同）。

FALSE - 没有更多可供读取的行。

sybase_fetch_assoc

从与关联数组中的指定结果标识符相关联的结果集中读取一行数据。

将内部指针在结果集中前移一行。后续对 **sybase_fetch_assoc()** 的调用会返回结果集中的下一行，如果没有更多行，则返回 **FALSE**。

语法

```
array sybase_fetch_assoc(resource $result)
```

参数

- **\$result** - 调用 **sybase_query()** 所带来的结果资源。

返回

成功时，返回结果集中的下一行。

FALSE - 没有更多可供读取的行。

sybase_fetch_field

返回包含字段信息的对象。

此函数可用于获取与提供的查询结果中的字段相关的信息。

语法

```
object sybase_fetch_field(resource $result [, int $field_offset ])
```

参数

- **\$result** - 调用 **sybase_query()** 所带来的结果资源。
- **\$field_offset** - 用于检索信息的字段编号（从 0 开始）。如果未指定字段偏移，则使用此函数尚未检索的下一字段。

返回

返回属性为以下字段信息的对象。

字段名	字段类型	字段说明
名称	字符串	列名。
table	字符串	从中获取列的表。
max_length	int	列的最大长度。

字段名	字段类型	字段说明
type	字符串	cspublic.h 中定义的列的数据类型。

sybase_fetch_object

以对象形式读取行。

此函数与 **sybase_fetch_assoc()** 类似，但它返回的是对象而非数组。它将内部指针在结果集中前移一行。

语法

```
object sybase_fetch_object(resource $result [, mixed $object ])
```

参数

- **\$result** - 调用 **sybase_query()** 所带来的结果资源。
- **\$object** - (可选) 指定要返回的对象的类型。缺省对象类型为 stdClass。

返回

返回其属性与读取行的字段名称相对应的对象。

FALSE - 没有更多可供读取的行。

sybase_fetch_row

从与数值数组中的指定结果标识符相关联的结果集中读取一行数据。

它将内部指针在结果集中前移一行。

后续对 **sybase_fetch_row()** 的调用会返回结果集中的下一行，如果没有更多行，则返回 FALSE。

语法

```
array sybase_fetch_row(resource $result)
```

参数

- **\$result** - 调用 **sybase_query()** 所带来的结果资源。

返回

数值数组 (从 0 开始) - 与 **sybase_fetch_array(\$result, SYB_FETCH_NUM)** 相同

FALSE - 没有更多可供读取的行。

sybase_field_seek

将内部指针设置为请求的字段偏移。

如果下次调用 **sybase_fetch_field()** 未指定字段偏移，则可以使用内部指针设置的字段。

语法

```
bool sybase_field_seek(resource $result, int $field_offset)
```

参数

- **\$result** - 调用 **sybase_query()** 所带来的结果资源。
- **\$field_offset** - 字段偏移（从 0 开始）。

返回

TRUE - 内部指针设置正确。

FALSE - 正确设置内部指针失败。

sybase_free_result

释放所有与此结果集相关联的内存。

虽然在 PHP 脚本结束时会自动释放结果内存，但 Sybase 建议作为良好的编程习惯，应释放不再需要的内存。

语法

```
bool sybase_free_result(resource $result)
```

参数

- **\$result** - 调用 **sybase_query()** 所带来的结果资源。

返回

TRUE - 内存释放成功。

FALSE - 释放内存失败。

sybase_get_last_message

服务器返回的最后一条消息。

语法

```
string sybase_get_last_message(void)
```

参数

否

返回

服务器返回的最后一条消息。

FALSE - 最后一条服务器消息检索失败。

sybase_get_last_status

返回在连接 \$conn 上发送的最后一个状态结果。

语法

```
int sybase_get_last_status(resource $conn)
```

参数

- **\$conn** - 连接打开函数返回的连接资源。

返回

在连接 \$conn 上发送的最后一个状态结果。

sybase_next_result

返回指向连接 \$conn 中下一结果集的结果集标识符。

语法

```
mixed sybase_next_result(resource $conn)
```

参数

- **\$conn** - 连接打开函数返回的连接资源。

返回

成功时为正 Sybase 结果集标识符。

FALSE - 连接中没有进一步结果集。

sybase_num_fields

返回结果集中的字段数。

语法

```
int sybase_num_fields(resource $result)
```

参数

- **\$result** - 调用 **sybase_query()** 所带来的结果资源。

返回

结果集中的字段数。

FALSE - 字段数检索失败。

sybase_num_rows

返回 SELECT 语句结果集中的行数。

sybase_num_rows() 可在完整结果集读取完成后返回正确的行数。

语法

```
int sybase_num_rows(resource $result)
```

参数

- **\$result** - 调用 **sybase_query()** 所带来的结果资源。

返回

结果集中的行数。

FALSE - 行数检索失败。

sybase_pconnect

打开与 Adaptive Server 的持久连接。

如果之前已使用与本次调用所用参数相同的参数打开过持久连接，则将返回现有连接的标识符，而不打开新的连接。

语法

```
resource sybase_pconnect([string $servername  
[, string $username  
[, string $password  
]]) )
```

参数

- **\$servername** - 在相关 Sybase 目录服务中定义的服务器的名称。例如，**interfaces** 文件。
- **\$username** - 用于登录 Adaptive Server 的用户帐户的名称。
- **\$password** - 用于登录 Adaptive Server 的用户帐户的口令。

返回

成功时为正连接标识符。

FALSE - 失败时。

sybase_query

向连接发送查询。

语法

```
mixed sybase_query(string $query [, resource $conn])
```

参数

- **\$query** - 一个包含要发送到 Adaptive Server 的查询的字符串。
- **\$conn** - 连接打开函数返回的连接资源。如果未指定连接资源，则会使用最近打开的连接。

返回

成功时为正 Sybase 结果集标识符。

TRUE - 查询成功，但不返回任何结果集。

FALSE - 查询失败。

sybase_rpc_bind_param_ex

将 PHP 变量绑定到远程过程参数。

语法

```
bool sybase_rpc_bind_param_ex(resource $stmt,
int $param_id,
mixed $var,
string $type
[, bool $is_null
[, int $direction]] )
```

参数

- **\$stmt** - 调用 **sybase_rpc_init()** 返回的语句标识符资源。
- **\$param_id** - 要绑定的存储过程参数的位置索引。第一个参数从 0 开始。
- **\$var** - 对要绑定的 PHP 变量的引用（地址）。
- **\$type** - 要绑定的 PHP 变量的数据类型。可以是以下模式之一：
 - “d” - double
 - “i” - integer

“b” - binary

“s” - string

- **\$is_null** - 指示变量是否包含空值的布尔值（可选）。
- **\$direction** - 可选，为下列各项之一：
 - SASE_D_INPUT - 用于输入参数（缺省）。
 - SASE_D_OUTPUT - 用于输出参数。

返回

TRUE - PHP 变量绑定成功。

FALSE - PHP 变量绑定失败。

sybase_rpc_execute

执行在 \$stmt 中使用 **sybase_rpc_init()** 初始化的远程过程调用。

语法

```
mixed sybase_rpc_execute(resource $stmt)
```

参数

- **\$stmt** - **sybase_rpc_init()** 调用返回的语句标识符资源。

返回

成功时为正 Sybase 结果集标识符。

FALSE - RPC 执行失败。

sybase_rpc_init

返回指向连接 \$conn 中针对 \$procedure 初始化的语句的语句标识符。

语法

```
mixed sybase_rpc_init(resource $conn, string $procedure)
```

参数

- **\$conn** - 连接打开函数返回的连接资源。
- **\$procedure** - 要使用 **sybase_rpc_execute()** 执行的远程（存储）过程的名称。

返回

成功时为正 Sybase 语句标识符。

FALSE - RPC 语句初始化失败。

sybase_select_db

设置连接资源所引用的服务器上的当前活动数据库。

后续的每次 `sybase_query()` 调用都在当前连接资源的活动数据库中进行。

语法

```
bool sybase_select_db(string $database_name [, resource $conn])
```

参数

- **\$database_name** - 要使用的数据库的名称。
- **\$conn** - 连接打开函数返回的连接资源。如果未指定连接资源，则会使用最近打开的连接。

返回

TRUE - 当前数据库设置成功。

FALSE - 当前数据库设置失败。

sybase_set_message_handler

设置在接收到客户端或服务器消息时调用的用户定义回调函数。

语法

```
bool sybase_set_message_handler(callback $handler, int $msg_type  
[, resource $conn])
```

参数

- **\$handler** - 回调处理程序采用以下参数：
 - int - message_number
 - int - severity
 - int - state
 - int - line_number
 - string - description
- **\$msg_type** - 以下各项之一：
 - SYB_CLIENTMSG_CB - 客户端消息回调
 - SYB_SERVERMSG_CB - 服务器消息回调

注意：虽然 `$msg_type` 是强制性的，但由于客户端和服务器消息均要调用安装的消息处理程序，因此当前将其忽略。

- `$conn` - 连接打开函数返回的连接资源。如果未指定连接资源，则会使用最近打开的连接。

返回

TRUE - 回调函数安装成功。

FALSE - 回调函数安装失败。

sybase_unbuffered_query

向 `$conn` 所引用的连接发送查询。

与 `sybase_query()` 不同，不会自动读取和缓冲完整结果集。这样可实现更高的性能，尤其是对于较大的结果集。

使用 `sybase_fetch_array()` 和类似的函数根据需要读取更多的行，使用 `sybase_data_seek()` 跳转到目标行。

完整结果集读取完成后，使用 `sybase_num_rows()` 返回正确的行数。

语法

```
mixed sybase_unbuffered_query(string $query [, resource $conn])
```

参数

- `$query` - 包含要发送到 Adaptive Server 的查询的字符串。
- `$conn` - 连接打开函数返回的连接资源。

如果未指定连接资源，则会使用最近打开的连接。

返回

成功时为正 Sybase 结果集标识符。

TRUE - 查询成功，但未返回任何结果集。

FALSE - 查询失败。

sybase_use_result

存储连接 `$conn` 中上一次未缓冲查询的结果集，并返回指向该存储结果集的结果集标识符。

语法

```
mixed sybase_use_result(resource $conn)
```

参数

- `$conn` - 连接打开函数返回的连接资源。

返回

成功时为正 Sybase 结果集标识符。

FALSE - 连接中没有可供存储的进一步结果集。

安全性和目录服务

使用 `ocs.cfg` 和 `libtcl.cfg` 文件配置安全选项。

当前不支持启用安全性的扩展模块选项。

对于连接，使用 `ocs.cfg` 设置目录和安全属性。编辑 `libtcl.cfg` 以装载安全性和目录服务器驱动程序。

有关详细信息，请参见《适用于 UNIX 的 Open Client 和 Open Server 配置指南》中的“配置文件”。

其它资源

有关安装和配置此扩展模块的其它信息。

- 有关配置信息的 Open Client 和 Open Server 文档：
《适用于 UNIX 的 Open Client 和 Open Server 配置指南》> “配置文件”
- 适用于所有 Open Client 和 Open Server 产品的平台相关问题：
《适用于 UNIX 的 Open Client 和 Open Server 的程序员补充说明》
- 使用 Open Client 和 Open Server 运行时配置文件：
《Open Client Client-Library/C 参考手册》> “Using the runtime configuration file”
(使用运行时配置文件)> “Open Client and Open Server runtime configuration file syntax” (Open Client 和 Open Server 运行时配置文件语法)
- 平台支持：
针对所用平台的 Software Developer Kit and Open Server Installation Guide (《软件开发工具包和 Open Server 安装指南》)。

词汇表

特定于脚本化语言的术语词汇表。

- **Client-Library** - Open Client 的一部分，一组用于编写客户端应用程序的例程。Client-Library 专用于容纳游标和 Sybase 产品系列中的其它高级功能。
- **CS-Library** - 在 Open Client 和 Open Server 产品中附带提供，一组对 Client-Library 和 Server-Library 应用程序都可用的实用程序例程。
- **CT-Library** - (CT-Lib API) 是 Open Client 套件的一部分，是让脚本化应用程序连接到 Adaptive Server 所必需的。
- **DBD** - 特定于数据库供应商的驱动程序，将 DBI 数据库 API 调用转换为可被目标数据库 SDK 理解的形式。
- **PHP** - Hypertext 预处理程序的自参照首字母缩略词
- **线程** - 是通过 Open Server 应用程序和库代码完成的执行的路径，以及路径的关联堆栈空间、状态信息和事件处理程序。
- **Transact-SQL** - 数据库语言 SQL 的增强版。应用程序可以使用 Transact-SQL 与 Adaptive Server Enterprise 通信。

索引

A

安装

- 安装文件 2
- 概述 2
- 要求 2

B

- 版本要求 1

C

- 词汇表 17

D

- DBC-API 1

H

- 环境变量 2

K

扩展模块

- PHP API 参考 3
- 扩展模块 API 参考
 - sybase_affected_rows 3
 - sybase_close 4
 - sybase_connect 4
 - sybase_data_seek 5
 - sybase_fetch_array 5
 - sybase_fetch_assoc 6
 - sybase_fetch_field 6
 - sybase_fetch_object 7
 - sybase_fetch_row 7
 - sybase_field_seek 8

- sybase_free_result 8
- sybase_get_last_message 8
- sybase_get_last_status 9
- sybase_next_result 9
- sybase_num_fields 9
- sybase_num_rows 10
- sybase_pconnect 10
- sybase_query 11
- sybase_rpc_bind_param_ex 11
- sybase_rpc_execute 12
- sybase_rpc_init 12
- sybase_select_db 13
- sybase_set_message_handler 13
- sybase_unbuffered_query 14
- sybase_use_result 14

P

配置

- 安全性和目录服务 15
- 扩展库 2
- 示例脚本 2
- 要求 2

Q

- 其它资源 15

S

- 示例脚本 3
- 数据流框图 1

Z

组件

- 必需 1
- 说明 1

