



ApacheとPHPを使用した Adaptive Server Anywhere データからのコンテンツ配信

PHP について

PHP は、perl や ASP のようなサーバ側スクリプト言語で、PHP のマニュアル (<http://www.php.net/manual/>) では、以下のように説明されています。

PHP は、HTML に埋め込むスクリプト言語で、その構文のほとんどは C、Java、Perl に類似していますが、PHP 固有の機能も追加されています。そして、動的に生成されるページを Web 開発者が短時間で作成できるように設計されています。

PHP の優れた点として、Adaptive Server Anywhere データベースなどのさまざまなデータベースからデータを取り出せることが挙げられます。これは、SQLAnywhere PHP モジュールなどのモジュールや、iODBC や unixODBC などの ODBC ドライバ・マネージャのサポートにより実現されています。



目次

PHP について.....	表紙
目次.....	3
第 1 項：概要.....	4
PHP を使用してデータベースに接続する方法	4
ODBC と SQLAnywhere PHP モジュールの違い.....	4
要件	5
第 2 項：SQLAnywhere PHP モジュールを使用して Adaptive Server Anywhere に接続するように PHP を設定	6
SQLAnywhere モジュールのファイルを PHP のソース・ツリーに追加	6
Apache と PHP の設定.....	7
第 3 項：ODBC PHP モジュールを使用して Adaptive Server Anywhere に接続するように PHP を設定	10
第 4 項：Adaptive Server Anywhere で PHP を使用	13
Apache バージョンの手順	13
CGI 実行プログラム・バージョンの手順	15
第 5 項：トラブルシューティング	17

第 1 項:概要

PHP を使用してデータベースに接続する方法

PHP を使用してデータベース要求を受け取ることができるシステムを作成するには、以下のいずれかを利用します。

SQLAnywhere PHP モジュール (第 2 項を参照)

ODBC (第 3 項を参照)

ODBC と SQLAnywhere PHP モジュールの違い

PHP から他の種類のデータベースに接続する必要がある場合、または ODBC で使用する PHP スクリプトをすでに多数記述している場合は、ODBC を選択します。

それ以外の場合は、SQLAnywhere PHP モジュールの方が設定が簡単で、システム・リソースのリークの危険性も低いいため、こちらの使用をお勧めします。

要件

このホワイトペーパーで説明する手順を実行するには、システムに以下のソフトウェアが必要です。

Adaptive Server Anywhere 7.0.x : Apache Web サーバと同じマシンでも別のマシンでも実行できません。

PHP ソース・コード : <http://www.php.net> からダウンロードできます。

Apache Web Server のソース・コード : <http://httpd.apache.org> からダウンロードできます。コンパイル済みバージョンを使用する場合は、*apache* と *apache-devel* の両方をインストールしてください。

SQLAnywhere PHP モジュール (使用する場合) : http://www.sybase.com/detail_list?id=5329 で入手できます。

sqlpp および *libdblib7.so* もインストールしておく必要があります (Adaptive Server Anywhere の *lib* ディレクトリを確認してください)。*sqlpp* は、Adaptive Server Anywhere で Embedded SQL および Ultra Light Development のインストール・オプションを指定するとインストールされます。

Unified ODBC PHP モジュールを使用する場合は、*dbodbc7.so* をインストールする必要があります (Adaptive Server Anywhere の *lib* ディレクトリを確認してください)。

make、*automake*、*autoconf*、*libtool*、*makeinfo*、*bison*、*gcc*、*cpp*、*glibc-devel*、*kernel-headers*、*flex* の各バイナリがまだインストールされていない場合は、Linux のインストール・ディスクからインストールしてください。また、RedHat Linux を使用している場合は、RPM で入手することもできます。

一部のインストール手順を実行するには、PHP をインストールしたユーザと同じアクセス権が必要です。

第 2 項: SQLAnywhere PHP モジュールを使用して Adaptive Server Anywhere に接続するように PHP を設定

SQLAnywhere PHP モジュールを使用して PHP から Adaptive Server Anywhere に接続するには、SQLAnywhere モジュール・ファイルを PHP のソース・ツリーに追加して PHP を再コンパイルしてください。

SQLAnywhere モジュールのファイルを PHP のソース・ツリーに追加

- 1 `sqlanywhere.tar` を保存したディレクトリから、PHP のソース・ツリーの `ext` サブディレクトリに、ファイルを展開します。

```
$ tar -xf sqlanywhere.tar PHP_source_directory/ext/
```

例:

```
$ tar -xf sqlanywhere.tar ~/php-4.1.x/ext
```

- 2 PHP でモジュールを認識させます。

```
$ cd PHP_source_directory/ext/sqlanywhere
```

```
$ touch *
```

```
$ cd ~/php-4.1.x
```

```
$ ./buildconf
```

例:

```
$ cd ~/php-4.1.x/ext/sqlanywhere
```

```
$ touch *
```

```
$ cd ~/php-4.1.x
```

```
$ ./buildconf
```

- 3 PHP がモジュールを認識していることを確認します。

```
$ ./configure -help | egrep sqlanywhere
```

PHP が SQLAnywhere モジュールを認識すると、以下のメッセージが表示されます。

```
--with-sqlanywhere=[DIR]
```

Apache と PHP の設定

PHP は、Web サーバ (Apache など) の共有モジュールとして、または CGI 実行プログラムとしてコンパイルできます。PHP がサポートしていない Web サーバを使用している場合、または PHP スクリプトを Web ページではなくコマンド・シェルで実行する場合は、PHP を CGI 実行プログラムとしてコンパイルし、PHP を Apache で使用する場合は、Apache モジュールとしてコンパイルします。

また、PHP のコンパイル形式に応じて、「PHP を Apache モジュールとしてコンパイルする」または「PHP を CGI 実行プログラムとしてコンパイルする」のいずれかを参照してください。

PHP を Apache モジュールとしてコンパイル

以下の手順 1 および 2 は、共有モジュールを認識させるために Apache をコンパイルする手順です。Apache のコンパイル済みバージョンがすでにシステムにインストールされている場合は、手順 3 に進んでください。

- 1 共有モジュールを認識するように Apache を設定します。

Apache ファイルの展開先ディレクトリから、以下のコマンドを実行します (各コマンドは、すべて 1 行で入力します)。

```
$ cd Apache_source_directory
$ ./configure --enabled-shared=max --enable-module=most --
prefix=/Apache_installation_directory
```

例:

```
$ cd ~/apache_1.3.14
$ ./configure --enabled-shared=max --enable-module=most --
prefix=/usr/local/web/apache
```

- 2 関連コンポーネントを再コンパイルしてインストールします。

```
$ make
$ make install
```

これで、PHP を Apache の共有モジュールとしてコンパイルできます。

- 3 Adaptive Server Anywhere 用に環境が設定されていることを確認します。

SQL Anywhere Studio のインストール先ディレクトリ (デフォルトでは */opt/sybase/SYBSsa7*) から、使用しているシェルに応じたコマンドを入力します。

使用しているシェル	実行するコマンド
sh, ksh, bash	<code>../bin/asa_config.sh</code>
csh, tcsh	<code>source ../bin/asa_config.csh</code>

- 4 PHP を Apache モジュールとして、SQLAnywhere PHP モジュールとともに設定します。

以下のコマンドを実行します。

```
$ cd PHP_source_directory
```

```
$ ./configure --with-sqlanywhere --with-  
apxs=/Apache_installation_directory/bin/apxs
```

例：

```
$ cd ~/php-4.1.x/  
$ ./configure --with-sqlanywhere --with-  
apxs=/usr/local/web/apache/bin/apxs
```

configure スクリプトは、Adaptive Server Anywhere のインストールのバージョンと場所を特定します。このコマンドの出力内容には、以下のような行が含まれています。

```
checking for SQLAnywhere support... yes  
checking    SQLAnywhere install dir... /opt/sybase/SYBSSa7  
checking    SQLAnywhere version... 7
```

5 関連コンポーネントを再コンパイルします。

```
$ make
```

6 ライブラリが正しくリンクされていることを確認します。

```
$ ldd ../libs/libphp4.so
```

このコマンドは、*libphp4.so* が使用するライブラリのリストを出力します。そのリストに *libdblib7* が含まれていることを確認します。

7 PHP のバイナリを Apaches の *lib* ディレクトリにインストールします。

```
$ make install
```

8 *httpd.conf* 設定ファイルを編集し、Apache が *.php* ファイルを PHP スクリプトとして認識するようにします。

httpd.conf は、Apache のインストール先ディレクトリの *conf* サブディレクトリにあります。

```
$ cd Apache_installation_directory/conf
```

例：

```
$ cd /usr/local/web/apache/conf
```

httpd.conf のバックアップを作成してから、ファイルを編集します (必要に応じて、*pico* を任意のテキスト・エディタに置換してください)。

```
$ cp httpd.conf httpd.conf.backup
```

```
$ pico httpd.conf
```

httpd.conf で以下の行を追加するか、コメントをはずします (ファイル内でこれらの行が同じ箇所に記述されていないことがあります)。

```
LoadModule php4_module    libexec/libphp4.so
```

```
AddModule mod_php4.c
```

```
AddType application/x-httpd-php .php
```

```
AddType application/x-httpd-php-source .phps
```

最初の 2 行は、PHP コードの解釈に使用されるファイルを Apache で指定します。後の 2 行は、拡

張子が `.php` または `.phps` のファイルを Apache が認識して正しく処理できるように、ファイル・タイプを宣言します。

これで設定は終了です。新しい設定の使用方法およびテスト方法については、第 4 項の「Adaptive Server Anywhere で PHP を使用」を参照してください。

PHP を CGI 実行プログラムとしてコンパイル

1 Adaptive Server Anywhere 用に環境が設定されていることを確認します。

SQL Anywhere Studio のインストール先ディレクトリ (デフォルトでは `/opt/sybase/SYBSsa7/`) から、使用しているシェルに応じたコマンドを入力します。

使用しているシェル	実行するコマンド
sh, ksh, bash	<code>../bin/asa_config.sh</code>
csh, tcsh	<code>source ../bin/asa_config.csh</code>

2 PHP を CGI 実行プログラムとして、SQLAnywhere PHP モジュールとともに設定します。

PHP ファイルの展開先ディレクトリから、以下のコマンドを実行します。

```
$ cd PHP_source_directory
```

```
$ ./configure --with-sqlanywhere
```

例：

```
$ cd ~/php-4.1.x/
```

```
$ ./configure --with-sqlanywhere
```

`configure` スクリプトは、Adaptive Server Anywhere のインストールのバージョンと場所を特定します。このコマンドの出力内容には、以下のような行が含まれています。

```
checking for SQLAnywhere support... yes
```

```
checking SQLAnywhere install dir... /opt/sybase/SYBSsa7
```

```
checking SQLAnywhere version... 7
```

3 実行プログラムをコンパイルします。

```
$ make
```

これで設定は終了です。PHP の使用方法およびテスト方法については、第 4 項の「Adaptive Server Anywhere で PHP を使用」を参照してください。

第 3 項: ODBC PHP モジュールを使用して Adaptive Server Anywhere に接続するように PHP を設定

Adaptive Server Anywhere の ODBC ドライバに加えて他の ODBC ドライバを使用する場合は、ODBC ドライバ・マネージャ (iODBC や unixODBC など) をインストールする必要があります。ドライバ・マネージャをインストールし、対応する PHP モジュールを使用して PHP をコンパイルした後は (PHP またはドライバ・マネージャのマニュアルを参照)、ODBC ドライバ・マネージャで 2 つの作業を実行するだけで、データにアクセスできます。

Adaptive Server Anywhere ODBCドライバの `dbodbc7.so` を設定する

ODBC データ・ソースを作成する

Adaptive Server Anywhere の ODBC ドライバにドライバ・マネージャの機能が含まれているものの、他の ODBC ドライバを使用してデータベースに接続する必要がない場合は、ドライバ・マネージャのインストールおよび設定を行わずに ODBC を使用できます。

ODBC ドライバ・マネージャなしで ODBC を使用して Adaptive Server Anywhere にアクセスするように PHP を設定するには、以下の手順に従います。

- 1 Adaptive Server Anywhere のライブラリ・ディレクトリに移動します。

```
$ cd SQL_Anywhere_installation_directory/lib
```

例:

```
$ cd /opt/sybase/SYBSsa7/lib
```

- 2 Adaptive Server Anywhere の ODBC ドライバをドライバ・マネージャとして使用するために、シンボリック・リンクを設定します。

```
$ ln -s dbodbc7.so libodbc.so
```

```
$ ln -s dbodbc7.so libodbc.so.1
```

```
$ ln -s dbodbc7.so libodbcinst.so
```

```
$ ln -s dbodbc7.so libodbcinst.so.1
```

- 3 環境変数を設定します。

```
$ export CFLAGS=?-DODBC_UNIX?
```

```
$ export CUSTOM_ODBC_LIBS=?-lodbc?
```

- 4 以下のコマンドを実行して、Unified ODBC PHP モジュールを使用して PHP を Apache モジュールとして設定します。

```
$ cd PHP_source_directory
```

```
$ ./configure --with-apxs=/Apache_installation_directory/bin/apxs
```

```
--with-custom-odbc=/SQL_Anywhere_installation_directory/
```

例:

```
$ cd ~/php-4.1.x
$ ./configure --with-apxs=/usr/local/web/apache/bin/apxs --with-
custom-odbc=/opt/sybase/SYBSsa7/
```

- 5 関連コンポーネントを再コンパイルします。

```
$ make
```

- 6 ライブラリが正しくリンクされていることを確認します。

```
$ ldd ../libs/libphp4.so
```

このコマンドは、*libphp4.so* が使用するライブラリのリストを出力します。そのリストに *libodbc.so* が含まれていることを確認します。

- 7 PHP のバイナリを Apaches の *lib* ディレクトリにインストールします。

```
$ make install
```

- 8 Apache が Adaptive Server Anywhere の ODBC ドライバをロードできるように、環境を設定します。

```
$ export ODBC_INI=path_to_.odbc.ini
```

例：

```
$ export ODBC_INI=~/.odbc.ini
```

SQL Anywhere Studio のインストール先ディレクトリ (デフォルトでは */opt/sybase/SYBSsa7/*) から、使用しているシェルに応じたコマンドを入力します。

使用しているシェル	実行するコマンド
sh, ksh, bash	../bin/asa_config.sh
csh, tcsh	source ../bin/asa_config.csh

- 9 *httpd.conf* 設定ファイルを編集し、Apache が *.php* ファイルを PHP スクリプトとして認識するようにします。*httpd.conf* は、Apache のインストール先ディレクトリの *conf* サブディレクトリにあります。まず、*conf* ディレクトリに移動します。

```
$ cd Apache_installation_directory/conf
```

例：

```
$ cd Apache_installation_directory/conf
```

httpd.conf のバックアップを作成してから、ファイルを編集することをお勧めします。

```
$ cp httpd.conf httpd.conf.backup
```

```
$ pico httpd.conf
```

httpd.conf を開いたら、以下の行を追加するか、コメントをはずします (ファイル内でこれらの行が同じ箇所に記述されていないことがあります)。

```
LoadModule php4_module libexec/libphp4.so
AddModule mod_php4.c
AddType application/x-httpd-php .php
AddType application/x-httpd-php-source .phps
```

これで設定は終了です。新しい設定の使用方法およびテスト方法については、第 4 項の「Adaptive Server Anywhere で PHP を使用」を参照してください。

第 4 項: Adaptive Server Anywhere で PHP を使用

この項では、PHP の Apache バージョンおよび CGI 実行プログラム・バージョンの使用方法について説明します。

Apache バージョンの手順

この項の構成は、以下のとおりです。

「テスト・ファイルの作成」 PHP 設定のテストに使用できる PHP コードを紹介しています。

「Web サーバの URL」 ドキュメントを使用するためにブラウザで入力する URL について説明しています。

「データベースおよび Web サーバの起動」 Adaptive Server Anywhere および Apache サーバを起動する方法について説明しています。

テスト・ファイルの作成

すべての設定に共通のテスト・ファイル

`phpinfo()` を呼び出して、PHP が正しく設定されているかどうかをテストするファイルを作成します。

以下の PHP コードを記述したファイルを、*info.php* という名前で *htdocs* ディレクトリに保存します。このディレクトリは Apache ディレクトリにあります。

```
<? phpinfo() ?>
```

`phpinfo()` は、システム設定情報のページを生成する PHP 関数です。

SQLAnywhere PHP モジュールを使用する PHP のテスト・ファイル

データベースへの接続を開始し、クエリを実行して、結果を戻し、接続を切断するファイルを作成します。

以下の PHP コードを記述したファイルを、*asa_test.php* という名前で Apache の *htdocs* ディレクトリに保存します。

```
<?
    $conn = sqlanywhere_connect( "uid=DBA;pwd=SQL" );
    $result = sqlanywhere_query( $conn, "select * from employee" );
    sqlanywhere_result_all( $result );
    sqlanywhere_free_result( $result );
    sqlanywhere_disconnect( $conn );
?>
```

Unified ODBC PHP モジュールを使用する PHP のテスト・ファイル

データベースへの接続を開始し、クエリを実行して、結果を戻し、接続を切断するファイルを作成します。

以下の PHP コードを記述したファイルを、[asa_test_odbc.php](#) という名前で Apache の [htdocs](#) ディレクトリに保存します。

```
<?
    $conn = odbc_connect( "my_test_dsn", "DBA", "SQL" );
    $result = odbc_exec( $conn, "select * from employee" );
    odbc_result_all( $result );
    odbc_close( $conn );
?>
```

データベースにアクセスするための ODBC データ・ソースも作成する必要があります。

以下のコードを記述したファイルを、[.odbc.ini](#) という名前でホーム・ディレクトリに保存します。同名のファイルがすでに存在する場合は、そのファイルの最後にコードを追加します。

```
[my_test_dsn]
uid=DBA
pwd=SQL
DatabaseName=asademmo
```

最後に、[asademo.db](#) というデモ・データベースのコピーがホーム・ディレクトリにあることを確認します。このデータベースは、Adaptive Server Anywhere とともにインストールされます。

Web サーバの URL

Web サーバの URL は、`http://hostname[:server_port]/` という形式になります。`hostname` は、`hostname` コマンドをコマンド・プロンプトで実行することで確認できます。また、コンピュータの IP アドレスがわかっている場合はそれを指定できます。`server_port` は、通常は 80 または 8080 です。

たとえば、Web サーバ上の [test.html](#) という名前のファイルにアクセスしようとしているときに、`hostname` の実行結果が `helios` の場合は、`http://helios:8080/test.html` という URL を使用してファイルを表示できます。

データベースおよび Web サーバの起動

- 最初に、Adaptive Server Anywhere 用に環境が設定されていることを確認します。SQL Anywhere Studio のインストール先ディレクトリ (デフォルトでは `/opt/sybase/SYBSsa7/`) から、使用しているシェルに応じたコマンドを入力します。

使用しているシェル	実行するコマンド
sh, ksh, bash	<code>../bin/asa_config.sh</code>
csh, tcsh	<code>source ../bin/asa_config.csh</code>

2 Apache Web サーバを起動します。

```
$ /Apache_installation_directory/bin/apachectl start
```

例：

```
$ /Apache_installation_directory/bin/apachectl start
```

3 設定をテストしていて、「テスト・ファイルの作成」の説明に従ってファイルを作成している場合は、以下の URL を使用して PHP をテストします。

```
http://hostname[:server-port]/info.php
```

テストが成功した場合は、Web サーバの設定についての情報を示すページが表示されます。SQLAnywhere PHP モジュールを使用している場合は、*sqlanywhere* モジュールについての情報も表示されます。

4 PHP を使用してアクセスしようとするデータベースを起動します。

たとえば、設定をテストしている場合は、以下のコマンドを使用します。

```
$ dbeng7 asademo.db
```

5 Adaptive Server Anywhere のデモ・データベースに対してクエリを実行して、PHP を使用してデータベースにアクセスできるかどうかを確認します。「テスト・ファイルの作成」の説明に従ってファイルをすでに作成していて、SQLAnywhere PHP モジュールを使用している場合は、以下の URL を使用します。

```
http://hostname[:server-port]/asa_test.php
```

Unified ODBC PHP モジュールを使用している場合は、以下の URL を使用します。

```
http://hostname[:server-port]/asa_test_odbc.php
```

設定に問題がなければ、上記の URL を開くと、ブラウザに *employee* テーブルの内容が表示されません。

テストが終了したら、コンソール・ウィンドウで "q" を入力してデータベース・サーバを停止し、以下のコマンドを使用して Web サーバを停止します。

```
$ /Apache_installation_directory/bin/apachectl stop
```

例：

```
$ /usr/local/web/apache/bin/apachectl stop
```

CGI 実行プログラム・バージョンの手順

1 最初に、Adaptive Server Anywhere 用に環境が設定されていることを確認します。SQL Anywhere Studio のインストール先ディレクトリ (デフォルトでは */opt/sybase/SYBSsa7/*) から、使用しているシェルに応じたコマンドを入力します。

使用しているシェル	実行するコマンド
sh, ksh, bash	<code>../bin/asa_config.sh</code>
csh, tcsh	<code>source ../bin/asa_config.csh</code>

2 PHP を使用してアクセスしようとするデータベースを起動します。

たとえば、設定をテストしている場合は、以下のコマンドを使用します。

```
$ dbeng7 asademo.db
```

3 別のコンソール・ウィンドウを開き、コンパイルした *php* 実行プログラムを使用してスクリプトを実行します。

```
$ cd PHP_source_directory
```

```
$ ./php name_and_location_of_php_script
```

たとえば、設定をテストしている場合は以下のように入力します。

```
$ cd ~/php-4.1.x
```

```
$ ./php ext/sqlanywhere/examples/connect.php
```

設定に問題がなければ、"Connected successfully" と表示されます。

テストが終了したら、データベース・サーバを実行しているコンソール・ウィンドウで "q" を入力して、データベース・サーバを停止します。

第 5 項:トラブルシューティング

make または make install を実行するとエラーが発生する

make clean コマンドを実行してから、もう一度 make または make clean を実行してください。

「要件」で示したコンポーネントがすべてインストールされていることを確認してください。

dbdsn を使用して新しいデータ・ソースを作成すると、"Could not open config file /.odbc.ini" というエラーが発生する

.odbc.ini というファイル (空または他のデータ・ソースを含む) がホーム・ディレクトリにあり、そのファイルへの書き込み権限を持っていることを確認してください。

make install を実行しようとする、Permission Denied エラーが多数発生する

make install が本来の場所以外の場所にファイルをインストールしようとしているか、アクセス権が正しく設定されていません。

make install がファイルをインストールする本来の場所とは異なる場所に PHP をインストールしている場合は、./configure スクリプトで --prefix=location_where_PHP_was_installed オプションを使用して、PHP をインストールした場所を指定してください。

PHP をインストールしたユーザとは異なるユーザの場合は、そのユーザのホーム・ディレクトリに PHP をインストールして最初からやり直すか、su を使用して一時的に別のユーザ (場合によっては root) としてログインしてください。

詳細については、以下を参照してください。

SQL Anywhere Studio の CD-ROM には、オンライン・ヘルプが収録されています。CD-ROM を /mnt/cdrom/ にマウントするように設定している場合、ヘルプは /mnt/cdrom/help/contents.htm にありますので、Netscape Navigator などの、テーブルをサポートしている Web ブラウザで、このヘルプ・ファイルを開いてください。スタイル・シートをサポートしているブラウザを使用することをお勧めしますが、必須ではありません。

Adaptive Server Anywhere の Linux バージョンの FAQ は <http://www.sybase.com/detail?id=1011965> (英語) にあります。(詳細は<http://www.ianywhere.jp>よりお問い合わせください。)

SQL Anywhere Studio for Linux 7.0.2 のインストールおよび基本的な使用方法についての HOWTO は、『Linux Documentation Project』(<http://www.linuxdoc.org/>) (英語) から入手できます。詳細については、詳細は<http://www.ianywhere.jp>よりお問い合わせください。

Bug Fix などの追加は、サポート契約者向けWebページにてご案内しています (ご利用にはサポート契約が必要です)。マイナーバージョンアップ版へのパッチは<http://www.ianywhere.jp/dl/index.html>よりダウンロードください。

関連するニュースグループとして、sybase.public.sqlanywhere.general や sybase.public.sqlanywhere.linux があります。これらは、Web またはニュースグループ・リーダで購読できますが、問題によってはすでに解決されている場合があるため、以前のスレッドの内容も確認してください。

ニュースグループ・リーダを使用する場合は、forums.sybase.com で購読できます。Web を使用する場合は、<http://www.sybase.com/support/newsgroups> で確認できます。（英語です）

法的注意

Copyright(C) 2000-2003 iAnywhere Solutions, Inc. All rights reserved.

Adaptive Server、iAnywhere、iAnywhere Solutions、SQL Anywhere、Anywhere、iAnywhere Solutions、のロゴは、米国法人iAnywhere Solutions, Incまたは米国法人Sybase,Inc.とその系列会社の米国または日本における登録商標または商標です。その他の商標はすべて各社に帰属します。

Mobile Linkの技術には、Certicom,Inc.より供給を受けたコンポーネントが含まれています。これらのコンポーネントは特許によって保護されています。

本書に記載された情報、助言、推奨、ソフトウェア、文書、データ、サービス、ロゴ、商標、図版、テキスト、写真、およびその他の資料(これらすべてを"資料"と総称する)は、iAnywhere Solutions,Inc.とその供給元に帰属し、著作権や商標の法律および国際条約によって保護されています。また、これらの資料はいずれも、iAnywhere Solutions,Inc.とその供給元の知的所有権の対象となるものであり、iAnywhere Solutions,Inc.とその供給元がこれらの権利のすべてを保有するものとします。

資料のいかなる部分も、Sybaseの知的所有権のライセンスを付与したり、既存のライセンス契約に修正を加えることを認めるものではないものとします。

資料は無保証で提供されるものであり、いかなる保証も行われません。iAnywhere Solutionsは、資料に関するすべての陳述と保証を明示的に拒否します。これには、商業性、特定の目的への整合性、非侵害性の黙示的な保証を無制限に含みます。

iAnywhere Solutionsは、資料自体の、または資料が依拠していると思われる内容、結果、正確性、適時性、完全性に関して、いかなる理由であろうと保証や陳述を行いません。iAnywhere Solutionsは、資料が途切れていないこと、誤りがなく、いかなる欠陥も修正されていることに関して保証や陳述を行いません。ここでは、「iAnywhere Solutions」とは、iAnywhere Solutions, Inc.またはSybase,Inc.とその部門、子会社、継承者、および親会社と、その従業員、パートナー、社長、代理人、および代表者と、さらに資料を提供した第三者の情報元や提供者を表します。

* 本書は、米国iAnywhere Solutions社が作成・テストしたものを日本語に翻訳したものです。



アイエニウェア・ソリューションズ株式会社
<http://www.ianywhere.jp>