

# Tmax プログラミングガイド (RPC)

Tmax v6.0



Copyright © 2016 TmaxSoft Co., Ltd. All Rights Reserved.

## Copyright Notice

Copyright © 2016 TmaxSoft Co., Ltd. All Rights Reserved.

45, Jeongjail-ro, Bundang-gu, Seongnam-si, Gyeonggi-do, 13613, South Korea

## Restricted Rights Legend

All TmaxSoft Software (Tmax®) and documents are protected by copyright laws and international convention. TmaxSoft software and documents are made available under the terms of the TmaxSoft License Agreement and this document may only be distributed or copied in accordance with the terms of this agreement. No part of this document may be transmitted, copied, deployed, or reproduced in any form or by any means, electronic, mechanical, or optical, without the prior written consent of TmaxSoft Co., Ltd.

Nothing in this software document and agreement constitutes a transfer of intellectual property rights regardless of whether or not such rights are registered) or any rights to TmaxSoft trademarks, logos, or any other brand features. This document is for information purposes only. The company assumes no direct or indirect responsibilities for the contents of this document, and does not guarantee that the information contained in this document satisfies certain legal or commercial conditions. The information contained in this document is subject to change without prior notice due to product upgrades or updates. The company assumes no liability for any errors in this document.

このソフトウェア(Tmax®)マニュアルの内容とプログラムは、日本国の著作権法および国際条約によって保護されています。マニュアルの内容とプログラムは、TmaxSoft Co., Ltd.との使用許諾契約書の下でのみ使用することができ、マニュアルは使用許諾契約で許可されている範囲を除いては、配布または複製することができません。TmaxSoftの書面による事前の承諾を得ることなく、このマニュアルの全部または一部を電子的または機械的な方法を問わず、転送、複製、配布したり、または二次的著作物を作成する等の行為を一切禁じます。

このソフトウェアのマニュアルとプログラムの使用許諾契約は、いかなる場合においても、マニュアル及びプログラムと関連する知的財産権(登録の有無を問わず)を譲渡するものと解釈されず、TmaxSoftのブランド、ロゴ、商標等の使用権限を与えるものではありません。マニュアルは、情報を提供する目的でのみ提供しており、これに伴う契約上の直接的ないしは間接的な責任を負わず、マニュアルの内容は法律上もしくは商業的な特定の条件が満たされることを保証しません。マニュアルの内容は、製品のアップグレード及び修正により、その内容が予告なく変更されることがあり、内容上の誤りがないことを保証しません。

## Trademarks

Tmax®, Tmax WebtoB® and JEUS® are registered trademark of TmaxSoft Co., Ltd. Other products, titles or services may be registered trademarks of their respective companies.

Tmax®, Tmax WebtoB®, JEUS® は、TmaxSoft Co., Ltd.の登録商標です。その他、記載されている会社名、製品名などは、各社の商標または登録商標です。

## Open Source Software Notice

Some modules or files of this product are subject to the terms of the following licenses. : openssl-0.9.7.m, zlib-1.1.4, expat-2.0.0, net-snmp, DCE1.0, pthread, google-diff-match-patch, libevent, getopt

---

Detailed Information related to the license can be found in the following directory :  
\${INSTALL\_PATH}/license/oss\_licenses

この製品の一部ファイルまたはモジュールは、openssl-0.9.7.m、zlib-1.1.4、expat-2.0.0、net-snmp、DCE1.0、pthread、google-diff-match-patch、libevent、getoptライセンスを遵守します。

詳細情報については、製品ディレクトリーの\${INSTALL\_PATH}/license/oss\_licensesに記載されている事項を参照してください。

## 文書情報

文書名: Tmax プログラミングガイド (RPC)

発行日: 2016年8月5日

ソフトウェアバージョン: Tmax v6.0

ガイドバージョン: v2.1.1

---



# 目次

このガイドについて .....	ix
<b>第1章 紹介 .....</b>	<b>1</b>
1.1. 概念 .....	1
1.2. 種類 .....	1
1.2.1. DAサーバー .....	2
1.2.2. TPサーバー .....	4
1.2.3. ファンクション・サーバー .....	6
1.3. 制約事項 .....	8
<b>第2章 構成要素 .....</b>	<b>9</b>
2.1. 概要 .....	9
2.2. インターフェース定義ファイル .....	9
2.2.1. SQLインターフェース定義ファイル .....	9
2.2.2. API定義ファイル .....	12
2.3. idlc .....	13
2.4. 共通ライブラリー .....	15
<b>索引 .....</b>	<b>17</b>



## 図目次

[図 1.1]	DAサーバーを使用する場合のTmaxプロセス .....	2
[図 1.2]	DAサーバーを使用するためのサーバーとクライアントの構成 .....	2
[図 1.3]	TPサーバーを使用する場合のTmaxプロセス .....	4
[図 1.4]	TPサーバーのためのクライアントとサーバーの構成 .....	5
[図 1.5]	ファンクション・サーバーを使用する場合のTmaxプロセス .....	6
[図 1.6]	ファンクション・サーバーのためのクライアントとサーバーの構成 .....	7





# このガイドについて

## 対象読者

本書は、Tmax<sup>®</sup>(以下、Tmax)プログラム・モデルのうち、RPCについての概要と機能および特性について説明しています。同書は、Tmax RPC(Remote Procedure Call)を用いて開発する開発者とシステム管理者を対象としています。

## 前提知識

本書は、Tmaxシステムの概要とTmaxシステムが提供する各種機能や特性などを習得するための手引書です。

本書を理解するためには、以下の事項についての知識が必要です。

- ミドルウェアおよびUNIXシステムについての基本知識
- Tmaxの基本概念
- JavaとCプログラミングについての知識

## 制限事項

本書では、RPCについてのみ説明しています。実務上の具体的な使用方法や管理・運用についての内容は、各製品ガイドを参照してください。

---

### 参考

Tmaxシステムの開発についての基本的な内容は、『Tmax 運用ガイド』および『Tmax アプリケーション開発ガイド』を、Tmaxが提供するコマンドとC APIについては、『Tmax リファレンスガイド』を参照してください。

---

# 本書の構成

本書は、計2章で構成されています。

各章の主な内容は以下のとおりです。

- 第1章: 紹介

RPCの概念と種類および制約事項について説明します。

- 第2章: 構成要素

RPCの構成要素について説明します。

## 表記上の規約

表記	意味
<AaBbCc123>	プログラム・ソースコードのファイル名、ディレクトリー
<Ctrl> + C	CtrlキーとCキーを同時に押す
[Button]	GUIのボタン、メニュー名
太字	強調
「」、『』（鍵カッコ）	関連文書、あるいはガイド内の他の章および節の表示
「入力項目」	画面UI上の入力項目
ハイパーリンク	メール・アカウント、Webサイト
>	メニューの実行順
+ ----	下位ディレクトリー/ファイル有り
----	下位ディレクトリー/ファイル無し
<div>参考</div>	参照/注意事項
[図1.1]	図の名前
[表1.1]	表の名前
AaBbCc123	コマンド、コマンド実行結果の画面出力、サンプル・コード
[ ]	オプション・パラメータ値
	選択パラメータ値

## システム要件

	要求事項
プラットフォーム	IBM AIX 5.x / 6.1 / 7.1
	HP-UX 11.xx
	SunOS 5.7~5.9 / SunOS 5.10 / SunOS 5.11
ハードウェア	1GB以上のハードディスク空き容量
	512MB以上のメモリー空き容量
データベース	Oracle 9~12
	Tibero 4~5
	DB2
	Informix

## 関連文書

ガイド	説明
Tmax 運用ガイド	Tmaxを利用するための環境設定ファイルとシステム運用方法について説明しています
Tmax アプリケーション開発ガイド	Tmaxアプリケーション・プログラムの開発で使用するAPIの概念と使用方法および例について説明しています
Tmax リファレンスガイド	Tmaxアプリケーションの開発に使用するコマンドおよびクライアントとサーバーの接続、通信に使用する関数の使用方法と例について説明しています

## お問合せ先

### Korea

TmaxSoft Co., Ltd.  
45, Jeongjail-ro, Bundang-gu,  
Seongnam-si, Gyeonggi-do, 13613  
South Korea  
Tel: +82-31-8018-1000  
Fax: +82-31-8018-1115  
Email: [info@tmax.co.kr](mailto:info@tmax.co.kr)  
Web (Korean): <http://www.tmaxsoft.com>  
TechNet: <http://technet.tmaxsoft.com>

### USA

TmaxSoft Inc.  
101 North Wacker Drive, Suite 2014,  
Chicago, IL 60606  
U.S.A  
Tel: +1-312-525-8330  
Email: [info@tmaxsoft.com](mailto:info@tmaxsoft.com)  
Web (English): [http://www.tmaxsoft.com/us\\_en/home](http://www.tmaxsoft.com/us_en/home)

### Japan

TmaxSoft Japan Co., Ltd.  
5F Sanko Bldg, 3-12-16 Mita, Minato-Ku, Tokyo, 108-0073  
Japan  
Tel: +81-3-5765-2550  
Fax: +81-3-5765-2567  
Email: [info@tmaxsoft.co.jp](mailto:info@tmaxsoft.co.jp)  
Web (Japanese): <http://www.tmaxsoft.co.jp>

## China

Beijing TmaxSoft System Software Co., Ltd.  
Room103, No.2 Huizhong Building, Seven Street Shangdi,  
Haidian District, Beijing, 100085  
P.R.China  
Tel: +86-10-6298-8827  
Email: [info@tmaxsoft.com.cn](mailto:info@tmaxsoft.com.cn)  
Web (Chinese): [http://www.tmaxsoft.com/cn\\_en/home\\_cn\\_en](http://www.tmaxsoft.com/cn_en/home_cn_en)

## Brazil

Tmax Brasil Sistemas e Serviços Ltda.  
Av. Copacabana, 177, sala 32~35 Empresarial 18 do Fortel  
Alphaville Barueri, Sao Paulo, 06472-001  
Brazil  
Tel: +55-11-4191-3100  
Fax: +55(11) 4191-3705 (extension#112)  
Email: [info.bra@tmaxsoft.com](mailto:info.bra@tmaxsoft.com)  
Web (Portuguese): [http://www.tmaxsoft.com/br\\_en/home\\_br\\_en](http://www.tmaxsoft.com/br_en/home_br_en)

## Russia

Tmax Rus L.L.C.  
Leninsky prospekt, 113/1 (Park Place Moscow),  
Office 318e, Moscow, 117198  
Russia  
Tel: +7(495)970-01-35  
Email: [info.rus@tmaxsoft.com](mailto:info.rus@tmaxsoft.com)  
Web (Russian): [http://www.tmaxsoft.com/ru\\_ru/home\\_ru\\_ru](http://www.tmaxsoft.com/ru_ru/home_ru_ru)

## Singapore

Tmax Singapore Pte. Ltd.  
430 Lorong 6, Toa Payoh #10-02,  
OrangeTee Building, 319402  
Singapore  
Tel: +65-6259-7223  
Fax: +65-6258-7112  
Email: [info.sg@tmaxsoft.com](mailto:info.sg@tmaxsoft.com)  
Web (English): [http://www.tmaxsoft.com/sg\\_en/home\\_sg\\_en](http://www.tmaxsoft.com/sg_en/home_sg_en)

## United Kingdom

TmaxSoft UK Ltd.  
215 Knyvett House, Watermans Business Park,  
The Causeway, Staines TW18 3BAB  
United Kingdom  
Tel: +44-1784-895005  
Email: [info.uk@tmaxsoft.com](mailto:info.uk@tmaxsoft.com)  
Web (English): [http://www.tmaxsoft.com/gb\\_en/home\\_gb\\_en](http://www.tmaxsoft.com/gb_en/home_gb_en)

## Canada

TmaxSoft Canada, Inc.  
2425 Matheson Blvd East, 8th floor,  
Unit 824 Mississauga, ON, L4W 5K4  
Canada  
Tel: +1-905-361-2888  
Email: [info.canada@tmaxsoft.com](mailto:info.canada@tmaxsoft.com)  
Web (English): [http://www.tmaxsoft.com/ca\\_en/home\\_ca\\_en](http://www.tmaxsoft.com/ca_en/home_ca_en)



## Australia

TmaxSoft Proprietary Limited  
L32, 101 Miller Street, North Sydney 2060  
Australia  
Tel: +91-9845-330-704  
Email: [info.aus@tmaxsoft.com](mailto:info.aus@tmaxsoft.com)  
Web (English): [http://www.tmaxsoft.com/au\\_en/home\\_au\\_en](http://www.tmaxsoft.com/au_en/home_au_en)

## India

TmaxSoft Technologies Private Limited  
Sobha Alexander Plaza, 3rd Floor,  
16/2 Commissariat Road, Bangalore-560025  
India  
Tel: +91-9845-330-704  
Email: [info.india@tmaxsoft.com](mailto:info.india@tmaxsoft.com)  
Web (English): [http://www.tmaxsoft.com/in\\_en/home\\_in\\_en](http://www.tmaxsoft.com/in_en/home_in_en)

## Turkey

TmaxSoft Co., Ltd. Turkey Liaison Office  
Windowist Tower. Eski Buyukdere Cad. No:26,  
Maslak 34467 Istanbul  
Turkey  
Tel: +90-544-553-6045  
Email: [cslee@tmaxsoft.com](mailto:cslee@tmaxsoft.com)  
Web (English): [http://www.tmaxsoft.com/tr\\_en/home\\_tr\\_en](http://www.tmaxsoft.com/tr_en/home_tr_en)



# 第1章 紹介

本章では、Tmax RPC(Remote Procedure Call)の概念と種類および制約事項について説明します。

## 1.1. 概念

Tmaxは、tpcall、tpacallのようなAPIを利用して、サーバー上のサービスを呼び出す仕組みです。

アプリケーションの開発者は、事前に該当するサービス名に応じた入力引数や出力引数を文書にして渡すか、打ち合わせを行ってから開発に取り掛かる必要があります。そのほか、サーバーから呼び出す関数のインターフェースを定義した後、そのインターフェースを介して特定のユーティリティを利用する方法があります。この方法を利用すると、サーバーはスケルトン・ソース・ファイルを、クライアントはスタブ・ソース・ファイルを自動的に作成してデプロイします。これにより、サーバーとクライアント・アプリケーションを実装することができます。

Tmax RPCは、このような場合にインターフェースを定義してTmaxをすぐに使用できるようにする構成要素です。Tmax RPCは、SQL文を使用して関数のインターフェースを抽出する方法でインターフェースを定義します。ユーザーがSQL文を作成すると、それを分析して関数の引数として定義することができる構造です。

以下は、SQL文の作成例です。

```
ins_test:
INSERT INTO sample(no, name) VALUES ($empno, '$name');
```

上記のSQL文によって以下のような関数が定義されるので、クライアント・アプリケーションを実装する際、すぐに使用することができます。

```
int ins_test(char *empno, char *name)
```

## 1.2. 種類

Tmax RPCは、サービスの方法に応じて次のように3つに区分されます。

- データ・アクセス・サーバー(以下、DAサーバー)
- TPサーバー
- ファンクション・サーバー

## 1.2.1. DAサーバー

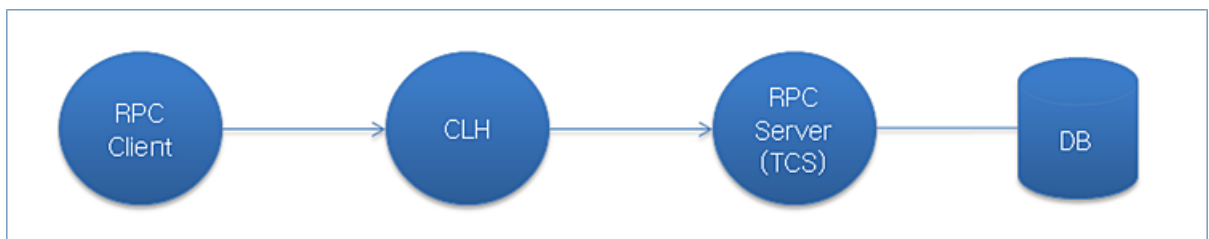
DAサーバーは、1つのSQLクエリーで構成された業務に使用します。インターフェースはSQLを使用して定義します。

### 参考

DAサーバーは1件のクエリーを実行するので、業務サーバーは非XAモードに設定する必要があります。設定についての詳細は、『Tmax アプリケーション開発ガイド』の「第6章 トランザクション」を参照してください。

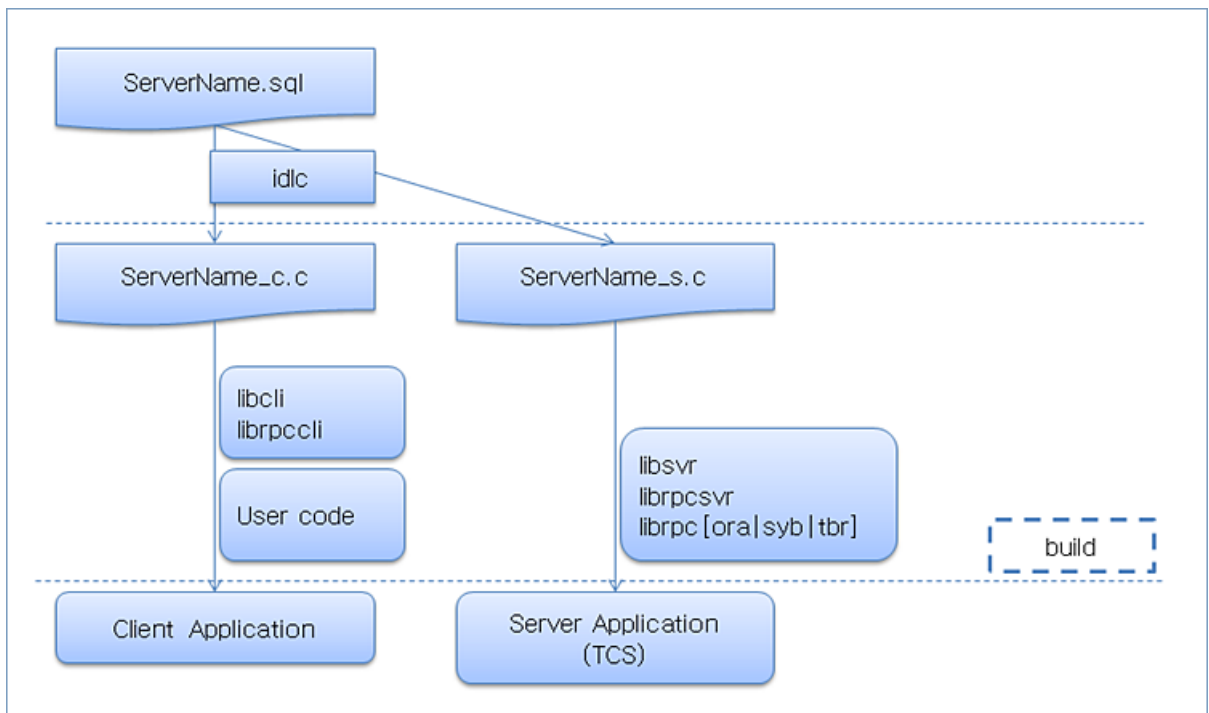
DAサーバーを使用する場合、Tmaxは以下のようなプロセスで動作します。

[図 1.1] DAサーバーを使用する場合のTmaxプロセス



DAサーバーを使用するためのサーバーとクライアントの構成は以下のとおりです。

[図 1.2] DAサーバーを使用するためのサーバーとクライアントの構成



DAサーバーを使用する場合、開発者は、業務サーバーの設定とビルドさえ行えば、追加コーディングなしですぐに業務を実行することができます。

たとえば、ins\_testという名前の挿入業務を作成する場合、dstest.sqlという名前のファイルにインターフェースを定義します。Tmax RPCは、サーバー名をインターフェース・ファイル名と同じく作成するので、サーバー名とサービス名をdstestという名前で定義します(Tmaxでは、サーバーとサービスという概念が存在するので、設定にもこのような概念が反映される必要があります)。開発者は、dstestというサービスを直接呼び出したり、定義したりしませんが、クライアントからins\_testという関数を呼び出すと、内部的にフィールド・バッファを作成してdstestをtpcallし、dstestというサーバーは該当するクエリーを実行します。

以下は、DAサーバーの環境設定とSQLインターフェース・ファイルの例です。

- DAサーバーの環境設定の例

```
*SVRGROUP
svg_rpc      NODENAME =tmaxi4

*SERVER
dstest       SVGNAME = svg_rpc,
             CLOPT="-e $(SVR).out -o $(SVR).out -- -uscott -ptiger"

*SERVICE
dstest       SVRNAME = dstest
```

- DAサーバーのSQLインターフェース・ファイルの例

DAサーバーのSQLインターフェース・ファイルは、idlcオプションに応じて作成されるファイルが変わります。SQLインターフェース定義ファイルの作成方法の詳細については、[「2.2.1. SQLインターフェース定義ファイル」](#)を参照してください。

```
### dstest.sql ###
#insert
ins_test:
INSERT INTO sample(no, name) VALUES ($empno, '$name');

#delete
del_test:
DELETE FROM sample WHERE no = $empno;

#select all
sel_all_test:
SELECT [int] no,
       name
FROM sample;

#select name
```

```

sel_test:
SELECT name
FROM sample
WHERE no = $empno;

#delete all
del_all_test:
DELETE FROM sample;

#update
update_test:
UPDATE sample SET name='$name' WHERE no=$empno;

#select binary
b_sel_test:
SELECT [binary] b_data FROM b_test_table;

#insert binary
b_ins_test:
INSERT INTO b_test_table VALUES $b_data[binary] ;

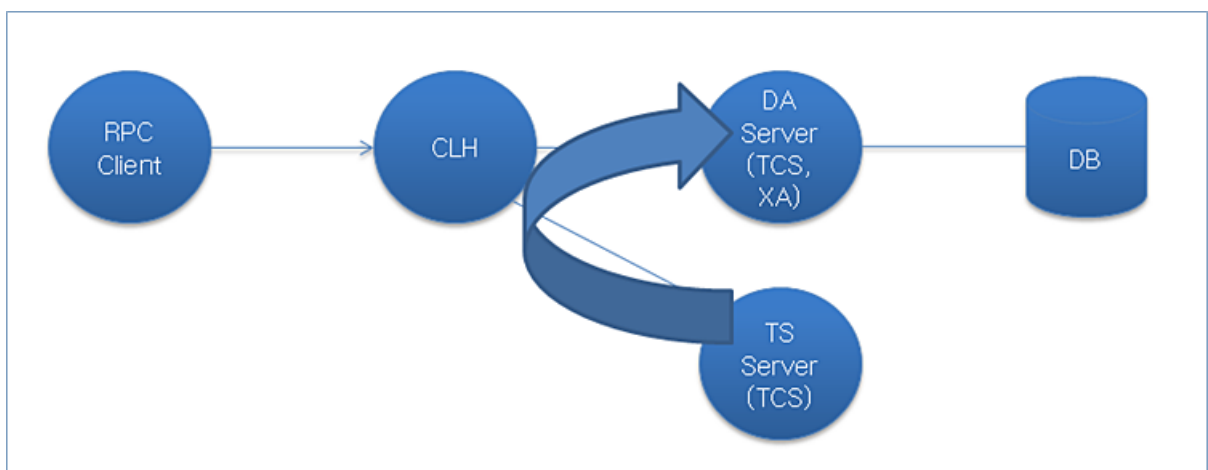
```

## 1.2.2. TPサーバー

TPサーバーは、1つのSQLクエリーではなく、トランザクションを使用して複数のRPCが同時に処理される業務に使用します。

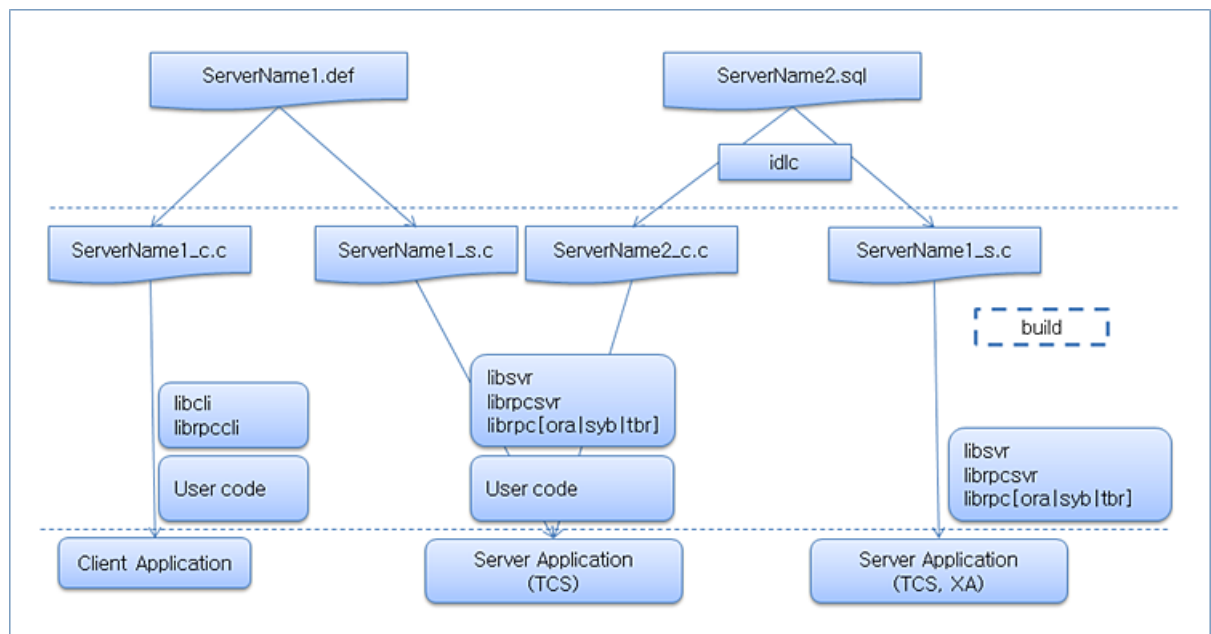
TPサーバーを使用する場合、Tmaxは以下のようなプロセスで動作します。

**[図 1.3] TPサーバーを使用する場合のTmaxプロセス**



TPサーバーを使用するためのサーバーとクライアントの構成は以下のとおりです。

**[図 1.4] TPサーバーのためのクライアントとサーバーの構成**



DAサーバーは非XAで処理を行うため、複数のクエリーを1つの業務に使用することは不適切です。間にTPサーバーが置かれており、そこからトランザクションを開始してDAサーバーの関数を呼び出すと、複数のクエリーを1つのトランザクションとして構成できるモデルが作られます。

TPサーバーを使用するには、クライアントからサーバーを呼び出せるインターフェース・ファイルと、DAサーバーで使用するSQLインターフェース・ファイルが必要です。クライアント・アプリケーションは、TPサーバーのAPIを呼び出し、TPサーバーのAPIは、DAサーバーのAPIを呼び出す方式です。DAサーバーは、1つのトランザクションに含まれる必要があるため、`xa svg`に設定してください。

以下は、TPサーバーの環境設定とインターフェース・ファイルの例です。

- TPサーバーの環境設定の例

```
*SVRGROUP
svg_xarpc      NODENAME = tmaxi4,
                OPENINFO = "Oracle_XA+Acc=P/scott/tiger+SesTm=60",
                DBNAME = ORACLE,
                TMSNAME = "tms_ora"

*SERVER
tstest         SVGNAME = "svg_rpc"
db1            SVGNAME = "svg_xarpc"

*SERVICE
```

```
TSTEST          SVRNAME = ttest
db1             SVRNAME = db1
```

- TPサーバーのAPIインターフェース・ファイルの例

APIインターフェース・ファイルの作成方法の詳細については、「[2.2.2. API定義ファイル](#)」を参照してください。

<\*.def>

```
### ttest.def ###
interface TSTEST {
    long trans_select (
        [out] long    sel_rows,
        [out] int     empno[sel_rows],
        [out] char    ename[][]);

    long trans_insert (
        [in]  int     dummy);

    long trans_clear (
        [in]  int     dummy);
}
```

- DAサーバーのSQLインターフェース・ファイルの例

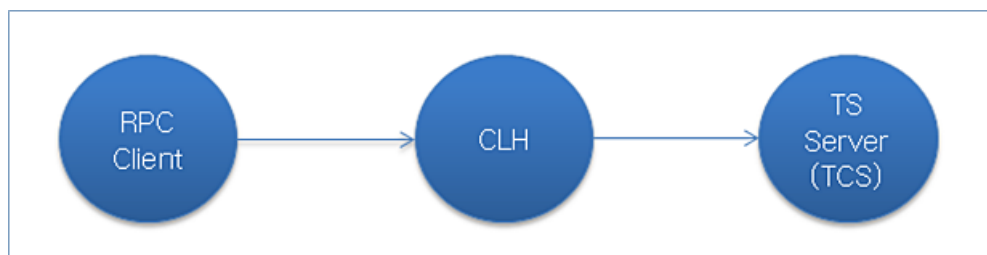
DAサーバーのSQLインターフェース・ファイルの例は、「[1.2.1. DAサーバー](#)」を、作成方法の詳細については、「[2.2.1. SQLインターフェース定義ファイル](#)」を参照してください。

## 1.2.3. ファンクション・サーバー

ファンクション・サーバーは、SQLとは関係のないユーザー定義関数を呼び出すために使用します。たとえば、クライアントが要求したファイルを読み込むなどの業務を実行します。

動作方式は、DAサーバーが不要であることを除けば、TPサーバーとほぼ同じです。

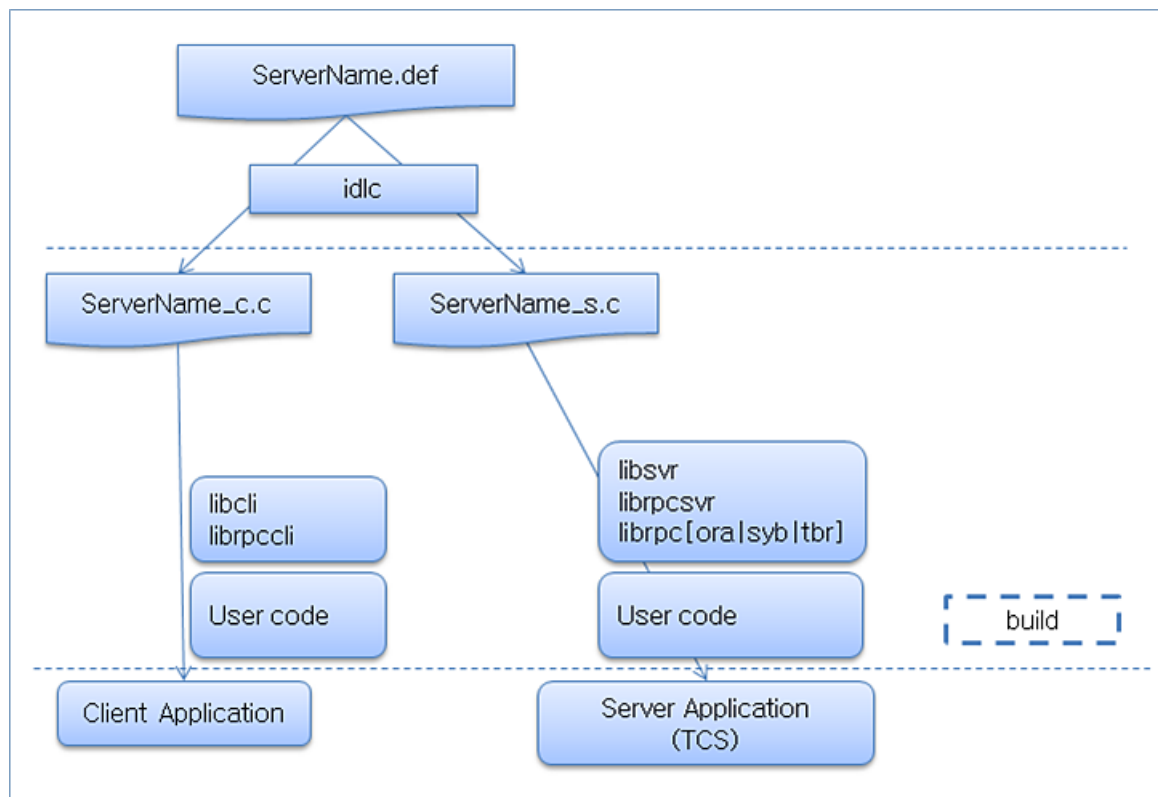
**[図 1.5] ファンクション・サーバーを使用する場合のTmaxプロセス**





ファンクション・サーバーを使用するためのサーバーとクライアントの構成は以下のとおりです。

【図 1.6】 ファンクション・サーバーのためのクライアントとサーバーの構成



ファンクション・サーバーを使用するには、APIインターフェース・ファイルを作成して、サーバー・コードとクライアント・コードを作成して実行する必要があります。

以下は、ファンクション・サーバーの環境設定とインターフェース・ファイルの例です。

- ファンクション・サーバーの環境設定の例

```
*SVRGROUP
svg_rpc      NODENAME =tmaxi4

*SERVER
fstest       SVGNAME = svg_rpc,
             CLOPT="-e $(SVR).out -o $(SVR).out -- -uscott -ptiger"

*SERVICE
fstest       SVRNAME = fstest
```

- ファンクション・サーバーのAPIインターフェース・ファイルの例

APIインターフェース・ファイルの作成方法の詳細については、「[2.2.2. API定義ファイル](#)」を参照してください。

<\*.def>

```

interface FSTEST {
    long plus(
        [in]    int a,
        [in]    int b,
        [out]   int c
    );

    long savefile (
        [in]    char filename[],
        [in]    char data[]
    );

    long readfile (
        [out]   char filename[],
        [out]   char data[]
    );
}

```

## 1.3. 制約事項

Tmax RPCは、一般的なサービスを実行するサーバーがRPCサーバーを呼び出すことはできません。

また、以下のOSとDBにのみ対応しています。(64ビットのみ対応)

OS	Oracle 11	Sybase 15.5	Tibero 4 Sp1
HP-IA 11.23	O	O	O
HP-IA 11.31	O	O	O
AIX 5.3	O	O	O
AIX 6.1	X	O	O
SUN5.10 sparc	O	O	O
Linux x86_64	O	O	O

## 第2章 構成要素

本章では、RPCプログラムを構成するための構成要素について説明します。また、種類別に実行例を挙げ、使用方法について説明します。

### 2.1. 概要

RPCプログラムのために基本的に必要な構成要素は以下のとおりです。

- インターフェース定義ファイル

インターフェース定義ファイルは、SQLインターフェース定義ファイルとAPI定義ファイルに区分されます。

- idlc

ソースを生成するユーティリティです。

- 共通ライブラリー

アプリケーションを構成するときに使用します。

### 2.2. インターフェース定義ファイル

インターフェース定義ファイルは、SQLインターフェース定義ファイルとAPI定義ファイルの2種類に区分されます。

#### 2.2.1. SQLインターフェース定義ファイル

SQLインターフェース定義ファイルは、SQLを介して業務を行う場合に使用するファイルとして、DAサーバーとTPサーバーを構成するときに使用します。1つのファイルの中に複数のSQL文を定義することができ、定義されたSQLは、それぞれ1つの関数を表現します。最終的には1つのファイルが1つのTCSサーバーとして表示されます。

以下は、SQLインターフェース定義ファイルの基本的な形式と各項目についての説明です。「#」で始まる行はコメントなので解析しません。

```
#comment
function_name:
SQL文;
```

```
function_name:
SQL文;
```

- function\_name

SQL文を実行するクライアントから呼び出す関数名です。コロン(:)までを関数名として認識します。

- SQL文

- 実行するSQL文です。複数行で表現することが可能であり、SELECT、INSERT、UPDATE、DELETE文を定義することができます。
- SQL文のうち「\$」で始まる単語は、関数に渡す引数として認識します。各引数はCスタイルのCHAR文字列で構成される必要があります。
- SELECTからFROMの間のフィールド名は、それぞれ関数の結果を保存できる引数として認識します。
- フィールド名の前に各フィールドのタイプを「[フィールドタイプ]」のような形式で定義すると、関数の作成時にそのタイプをそのまま使用することができます。char、int、long、float、double、string、binaryなどのフィールド・タイプを定義することが可能であり、binaryフィールド・タイプは、BLOB型のデータを処理する場合に使用します。

## SELECT文

以下は、「sel\_test」という名前のSELECT文を定義して変換した例です。

- SQL定義

```
sel_test:
SELECT
    [int] no,
    [double] comm,
    name
FROM sample
WHERE no = $no;
```

- 関数の変換

以下のように、\$noがchar \*noに変換されます。SELECT文は複数行を戻すことができるので、「tmax\_sel\_rows」という変数に行数を保存することが可能です。また、SELECTからFROMの間のフィールドを、それぞれのタイプに合わせて保存できるように順番にソートします。

```
long int sel_test(
    /* [in] */ char *no,
    /* [out] */ long int *tmax_sel_rows,
```

```

/* [out] */ int **outvars_0,
/* [out] */ double **outvars_1,
/* [out] */ char ***outvars_2
);

```

## INSERT文

以下は、「int\_test」という名前のINSERT文を定義して変換した例です。

- SQL定義

```

ins_test:
INSERT INTO sample(no, name) VALUES ($no, '$name');

```

- 関数の変換

以下のように、\$noと\$nameを引数として受け取る関数に変換されます。

```

long int ins_test(
    /* [in] */ char *no,
    /* [in] */ char *name
);

```

## UPDATE文

以下は、「update\_test」という名前のUPDATE文を定義して変換した例です。

- SQL定義

```

update_test:
UPDATE sample SET name='$name' WHERE no=$no;

```

- 関数の変換

以下のように、\$noと\$nameを引数として受け取る関数に変換されます。

```

long int update_test(
    /* [in] */ char *name,
    /* [in] */ char *no
);

```

## DELETE文

以下は、「del\_test」という名前のDELETE文を定義して変換した例です。

- SQL定義

```
del_test:
DELETE FROM sample WHERE no=$no;
```

- 関数の変換

以下のように、\$noを引数として受け取る関数に変換されます。

```
long int del_test(
    /* [in] */ char *no
);
```

## 2.2.2. API定義ファイル

TPサーバーとファンクション・サーバーで使用するAPIを定義する際に必要なファイルです。

サーバー上で実行する関数のプロトタイプを、以下のような形式で定義します。リターンと引数のタイプは、Cで定義するタイプで定義する必要があります。inまたはoutを使って引数のタイプを指定することができます。

```
interfaceサービス名 {
    リターンタイプ 関数名 (
        [in|out]    引数タイプ 引数名
    );
}
```

以下は、TSTESTというサービス名の関数を定義して変換した例です。

- 関数の定義

```
interface TSTEST {
    long trans_select (
        [out]    long    sel_rows,
        [out]    int     empno[sel_rows],
        [out]    char    ename[][]);

    long trans_insert (
        [in]     int     dummy
    );

    long trans_clear (
        [in]     int     dummy
    );
}
```

- 関数の変換

```

long int trans_select(
    /* [out] */ long int *sel_rows,
    /* [out] */ int **empno,
    /* [out] */ char ***ename
);
long int trans_insert(
    /* [in] */ int dummy
);
long int trans_clear(
    /* [in] */ int dummy
);

```

## 2.3. idlc

Enteraの変換時に、定義ファイルを実際にプログラミングが可能なソース(stub)に変換するユーティリティです。

以下は、idlcで使用可能なオプションとその説明です。

- 使用方法

```

idlc -inf インターフェース名
    -sql SQL インターフェース名
    -dbtype ora | tbr | syb
    -clang c | delphi | vbasic | pbuilder | java | delphi6
    -xa
    -header ファイル名
    -sstub ファイル名
    -cstub ファイル名

```

項目	説明
-inf インターフェース名	DAサーバーを使用する場合のインターフェース名を指定します
-sql SQL インターフェース名	DAサーバーとTPサーバーを使用する場合のインターフェース・ファイル名を指定します
-dbtype	DAサーバーを使用する場合のDBの種類を選択します。RPCがサポートするOSとDBの種類は「 <a href="#">1.3. 制約事項</a> 」を参照してください。 <ul style="list-style-type: none"> <li>– ora : Oracle</li> <li>– syb : Sybase</li> <li>– tbr : Tiberio</li> </ul>
-clang	作成するクライアントのソースStubファイルの言語を選択します

項目	説明
	<ul style="list-style-type: none"> <li>– c</li> <li>– delphi</li> <li>– vbasic</li> <li>– pbuilder</li> <li>– java</li> <li>– delphi6</li> </ul>
-xa	XAサーバーで作成する場合に指定します
-headerファイル名	作成するヘッダー・ファイル名を指定します
-sstubファイル名	作成するサーバーStubファイル名を指定します
-cstubファイル名	作成するクライアントStubファイル名を指定します

- 例

以下は、サーバーの種類別の使用例です。

- DAサーバー

Oracleを使用し、dstest.sqlを業務に使用し、クライアントはC言語を使用している場合、以下のようにオプションを設定します。

```
idlc -inf dstest
      -sql dstest.sql
      -dbtype ora
      -clang c
```

- TSサーバー

- 「.def」ファイルを使用する場合

「.def」ファイルを使用する場合、以下のようにオプションを設定します。

```
idlc -xa
      -clang c tstest.def
```

- DAサーバーを使用する場合

TPサーバーで呼び出すDAサーバーを使用する場合、以下のようにオプションを設定します。

```
idlc -inf db1
      -sql db1.sql
```



```
-dbtype ora
-clang c
-xa
-tps
```

#### – ファンクション・サーバー

ファンクション・サーバーは、TPサーバーと同様にオプションを設定して実行します。

```
idlc -xa
      -clang c fstest.def
```

## 2.4. 共通ライブラリー

TCSタイプのサーバーと一般的なクライアントのためには、デフォルトで提供するサーバー・ライブラリーとクライアント・ライブラリーのほか、RPCのためのライブラリーが必要です。

以下は、必要な共通ライブラリーについての説明です。

#### ● サーバーのためのライブラリー

- TCSタイプのサーバーのためにデフォルトで提供するサーバー・ライブラリです。
- libsvrを使用して必要な関数を実行できるようにします。
- UNIXでは、「librpcsvr.\*」という名前で提供します。

#### ● クライアントのためのライブラリー

- 一般的なクライアントのために提供するクライアント・ライブラリーです。
- libcliを使用して通信できるようにします。
- UNIXでは「librpccli.\*」、Windowsでは「tmaxrpccli.dll」という名前で提供します。

#### ● DBアクセスのためのライブラリー

- DBごとに特性があるので、DBに応じて異なるライブラリーが必要です。
- Oracleは「librpcsql.\*」、Tiberoは「librpctbr.\*」、Sybaseは「librpcsyb.\*」を使用します。



# 索引

## D

DAサーバー, 2

## I

idlcオプション

- clang, 13
- cstub, 14
- dbtype, 13
- header, 14
- inf, 13
- sql, 13
- sstub, 14
- xa, 14

## S

SQLインターフェース定義ファイル

- DELETE文, 11
- function\_name, 10
- INSERT文, 11
- SELECT文, 10
- SQL文, 10
- UPDATE文, 11

## T

- Tmax RPC, 1
- TPサーバー, 4

## は

ファンクション・サーバー, 6

