

**STARS クライアント ソフトウェア**

InsertionDevice 制御プログラム

**idgateway**  
**Stars I/O Client Program**

**idgateway 取扱説明書**

第一版

平成 22 年 5 月 7 日発行

物質構造科学研究所  
ビームライン制御チーム

■承認欄■

	物構研	放射光	コラボP	コラボP	作成
					永谷

はじめに

本書は、STARSのネットワーク上で利用される Insertion Device制御用STARS I/Oクライアントプログラム「idgateway」の取り扱いについて記述したものです。

■STARSについて （略称：“Simple Transmission and Retrieval System”）

ネットワーク上において、測定機器や制御機器を制御するI/O Client、GUI(Graphical User Interface) 操作や解析表示を行うApplication Client、これらClientを管理し実行を制御するServer と、処理の分散化を目的として開発されたシステムで、アプリケーション間のメッセージ配信を行う小規模システム向けの通信機構です。

STARS におけるアプリケーション間通信は、TCP/IP Socket を利用したテキストベースのコマンドの送受により行われるため、システムでの扱いが非常に容易となり、それと同時に、開発言語やOS 選択の幅が広がっています。さらに、コアの部分となるプログラムはPerl を使って開発されているので、STARS は様々なプラットフォーム上で動作可能です。

目次

1	はじめに .....	1
1.1	本プログラムについて .....	1
1.2	本プログラム稼働のための事前準備 .....	1
1.2.1	光源 G から IDServer の制御コマンド仕様を受け取る .....	1
1.2.2	光源 G に IDServer に関する情報を伝える .....	1
1.3	当プログラムの ReadMe ファイル確認のお願い .....	1
2	プログラムファイル一覧 .....	2
3	プログラムの設定 .....	3
3.1	idgateway の STARS ノード名の決定および STARS 認証用 Key ファイルの設定 .....	3
3.2	設定ファイルの編集 .....	3
3.2.1	STARS サーバ IP アドレスの設定 .....	3
3.2.2	IDServer IP アドレスとポート番号の設定 .....	3
3.2.3	IDServer の ID 設定 .....	4
3.2.4	IDServer の各種制御項目のプロパティ設定 .....	4
3.2.5	idgateway プログラムのログ出力の設定 .....	5
4	プログラムの起動 .....	6
4.1	本プログラム起動方法について .....	6

## 1 はじめに

### 1.1 本プログラムについて

本プログラム「idgateway」は、光源のIDServerとの相互通信でInsertion Deviceの制御をおこなうことを想定して作られたStars I/O Clientプログラムです。1つのSTARSノードに対し1つのIDServerが対応します。Perl言語で作成されています。

### 1.2 本プログラム稼働のための事前準備

以下、IDServerと当プログラムが連携して動作するために必要な事前手続きを説明します。

#### 1.2.1 光源 G から IDServer の制御コマンド仕様を受け取る

光源GからID Serverの制御コマンド仕様の情報をもらって、その情報を元にプログラムの設定をおこなっていきます。

#### 1.2.2 光源 G に IDServer に関する情報を伝える

- IDServerのビームライン側ローカルアドレス
- IDServerに接続可能なIDアドレス（「idgateway」が動作するマシンのIPアドレス）

上記の2項目を光源Gに伝えてID Serverが設定された後、本プログラムとIDServerの通信が可能となります。

### 1.3 当プログラムの ReadMe ファイル確認のお願い

プログラムフォルダにReadMeファイルを同梱しています。

プログラムの改変履歴や追加情報を更新していきますので、そちらもあわせてご確認ください。

## 2 プログラムファイル一覧

本プログラム「idgateway」のプログラムファイルは同一のフォルダで管理する必要があります。  
デフォルトの格納フォルダ名は「idgateway」です。

### <プログラムファイル一覧>

ファイル名	内容	ファイル種別
idgateway	idgateway Stars I/O Client プログラム本体	perl
idgateway.key	STARS ノード名「idgateway」用の STARS 認証 Key	text
config.pl	idgateway STARS I/O Client 設定ファイル	perl
nportsvr.pl	STARS の通信用ライブラリ	perl
stars.pm	STARS のライブラリ本体	perl
readme.txt	idgateway に関する Readme ファイル	text

### <マニュアル類>

ファイル名	内容	ファイル種別
idgatewayGeneral.pdf	本書 idgateway 取扱説明書	pdf
idgatewayCommandReference.pdf	idgateway STARS コマンド集	pdf

### 3 プログラムの設定

IDServer と STARS ノードは 1 : 1 で対応します。

ビームラインから接続する IDServer の数に応じて STARS ノードを起動する必要があります。

#### 3.1 idgateway の STARS ノード名の決定および STARS 認証用 Key ファイルの設定

本プログラムのSTARSのデフォルトノード名は「idgateway」です。

プログラムフォルダのSTARSノード名「idgateway」用のSTARS認証Keyファイル「idgateway.key」をSTARSサーバのライブラリフォルダ配下にコピーしてください。

他のノード名をご使用になられる場合で設定方法がわからない場合は、プログラム担当者までご連絡ください。

#### 3.2 設定ファイルの編集

本プログラムは設定ファイル「config.pl」を参照して動作します。

設定ファイルconfig.plの中身はperl言語の文法に則って記述する必要があります。

##### 3.2.1 STARS サーバ IP アドレスの設定

STARSサーバのIPアドレスを、設定ファイルconfig.plに記述します。

[例]STARSサーバのIPアドレスを192.168.11.100にする場合

<config.plの記述例>

```
$::Server = 192.168.11.100;
```

(サーバ名の記述)

変数\$::Server に STARS サーバの IP アドレスを記述します。ホスト名による指定も可能です。

##### 3.2.2 IDServer IP アドレスとポート番号の設定

IDServerのIPアドレスとポート番号を、設定ファイルconfig.plに記述します。

[例] ID Serverのローカル側IPアドレスが192.168.11.122、ポート番号8881の場合

<config.plの記述例>

```
$::NPORT_HOST = '192.168.11.122';  
$::NPORT_PORT = 8881;
```

(IDServer のローカル IP アドレスの記述)

変数\$::NPORT\_HOST に IDServer の ‘ビームライン側 IP アドレス’ を記述します。

(IDServer のポート番号の記述)

変数\$::NPORT\_PORT に IDServer の ‘Port 番号’ (通常 8881)を記述します。

## 3.2.3 IDServer の ID 設定

IDServerのIDを、設定ファイルconfig.plに記述します。

[例] ID ServerのIDが” NW2” の場合

<config.plの記述例>

```
@::IDCANWRITE = qw(NW2);
```

(IDServer のローカル IP アドレスの記述)

変数@::IDCANWRITE=qw()内に ‘IDServer の ID’ を記述します。

() に記述する際スペースは含めないよう気をつけてください。

## 3.2.4 IDServer の各種制御項目のプロパティ設定

IDServerのIDの各種制御項目のプロパティを指定します。

<制御項目名>: ”Gap” , ” Tpr” …

プロパティ名	意味
\$::ConfigLimitMaximum{<制御項目名>}	移動先として指定可能な最大値(以下)
\$::ConfigLimitMinimum{<制御項目名>}	移動先として指定可能な最小値(以上)
\$::ConfigMaximumDifference{<制御項目名>}	許容誤差 (以下) 移動時に移動先として指定した値と、実際の移動値が当プロパティの値以内であれば移動完了とみなす。
\$::ConfigAutoResetBusyTime{<制御項目名>}	移動命令を実行後IDServerの状態に変化が見られない場合の、Busy状態自動解除までの待ち時間(mS) 移動命令を実行したのにもかかわらず、IDServerの状態に変化がみられず移動先として指定した値に到達しないと、当プログラムはBusy状態のままになります。 当パラメータを指定すると、一定時間状IDServerの状態に変化がみられない場合Busy状態を自動解除し、STARSのオペレーションを続行できるようにします。
\$::ConfigFunctionEnable{<制御項目名>}	値0: 当プログラム側で移動不可とする 値1: ID Server側の設定に従う (デフォルト) ID Server側に移動命令を送信する前に移動できないようにする場合は0を設定します。 それ以外は、ID Server側の設定に従います。

上記のプロパティを、設定ファイルconfig.plに記述します。

(注意)

ID161については上記プロパティと関係なく、可変範囲のチェック等をプログラム独自でおこなっています。条件の変更等がある場合は、ビームライン制御チームまでご連絡ください。

[例] 制御項目「Gap」と「Tpr」についての設定例

<config.plの記述例>

```
$::ConfigLimitMaximum{"Gap"}=47;
$::ConfigLimitMinimum{"Gap"}=18;
$::ConfigMaximumDifference{"Gap"}=0.002;
$::ConfigAutoResetBusyTime{"Gap"}=30000; #MiliSeconds
$::ConfigLimitMaximum{"Tpr"}=1;
$::ConfigLimitMinimum{"Tpr"}=-1;
$::ConfigMaximumDifference{"Tpr"}=0.002;
$::ConfigAutoResetBusyTime{"Tpr"}=45000; #MiliSeconds
```

### 3.2.5 idgateway プログラムのログ出力の設定

idgatewayプログラムのログを出力することができます。

障害時の原因特定に役立ちますので、なるべくログ出力するように設定してください。

(ログファイルのサイズが大きくなりすぎないように、IDServerの状態が変更された場合にのみレコードを出力するようにしています)

[例] idgatewayプログラムログ出力の指定例

<config.plの記述例>

```
$::Logging      = 1;                      #Log Enabled
$::LogDir       = '/home/xafs-stars/stars/logdata';
```

(Log 出力有効無効の記述)

変数\$::Logging に 0 か ' ' (ブランク)を指定するとログ出力が無効となります。

(Log 出力先ディレクトリ名の記述)

変数\$::LogDir にログ出力先ディレクトリ名を記述します。

Windows OS の場合は ' c:/stars/logdata ' という風に記述します。



## 4 プログラムの起動

### 4.1 本プログラム起動方法について

[例] idgateway という STARS のノード名で起動する場合

コマンドプロンプトから起動します。

```
C:\¥stars>cd idgateway  
C:\¥stars¥idgateway >perl idgateway idgateway
```

(1 行目)

idgateway プログラムファイル格納フォルダをカレントディレクトリとします。

(2 行目)

Perl コマンドの 1 番目の引数は STARS I/O Client idgateway の Perl プログラムのファイル名です。

Perl コマンドの 2 番目の引数は STARS ノード名です。ノード名はユーザが自由に決めることができますが、その場合は STARS 認証 Key (ノード名.key) の準備を忘れずおこなってください。

ちなみに、-d オプションを付けることによってデバッグメッセージをスクリーン出力することができます。

```
C:\¥stars¥idgateway >perl idgateway idgateway -d
```

以上のステップで idgateway プログラムは起動されます。