TOPPERS TECS WG

大山博司
TOPPERS プロジェクト TECS WG 主査
名古屋大学大学院　情報科学研究科　客員教授
オークマ株式会社 主席技師

目次

- TECS の特徴
- TECS 開発の流れ
- 昨年のトピックス
  - TECSFlow の紹介
  - MrubyInfoBridgePlugin の紹介
- 今年のトピックス
  - TECSUnit (GitHub で公開)
  - TOPPERS/FMP3 対応 (近日早期リリース)
- 開発中
  - ローダブルモジュール
  - デバイスドライバ対応

TOPPERS/FMP3 対応

対応する事項は、3点

1. クラスへの対応
2. システムログ機能の TECS 化
3. TECS 版サンプルプログラムの作成

システムログ機能、TECS版サンプルプログラムは、既にTECS 対応済みの TOPPERS/ASP3, TOPPERS/HRP3 と同様の対応

以下、クラスへの対応について説明

クラスとは

- TOPPERS/FMP3において、クラスは、マルチプロセッサシステムで、カーネルオブジェクトのプロセッサへの割付を制御するものである
  - 「TOPPERS第3世代カーネル（ITRON系）統合仕様書」の定義では「マルチプロセッサに対応するために用いるカーネルオブジェクトの集合を，クラス（class）と呼ぶ．」

(補足)

次シート以降の記述で、TOPPERS/FMP3 を FMP3 と略することがある

本書では、カーネルオブジェクトのセル以外を通常セルと呼ぶ

TECS のクラス

- TECS では、カーネルオブジェクトのクラスと、通常のセルのクラスを分けて指定する
  - TOPPERS/FMP3 では、カーネルオブジェクトに対するクラスを指定するが、非カーネルオブジェクトに対するクラスの指定はない
  - 非カーネルオブジェクトの動作可能なプロセッサを限定する目的
- TECS のクラス名は、TOPPERS/FMP3 のクラス名に対応づける形で、独自に規定する
  - 将来的にクラスの識別名のバリエーションが生まれる可能性があるが、TECS では都度対応とする。プラグイン (FMPClassManager) の修正が必要になる

TOPPERS/FMP3のクラスと TECS のクラスの対応

下表は、プロセッサ1 の場合を例示している。プロセッサ2 の場合は、数字を2 に変更する

FMP3のクラス	TECS のクラス	処理単位の実行プロセッサ	カーネルオブジェクトのセル	通常セル
CLS_PRC1	CLS_P1_P1	プロセッサ Processor1で実行	どのプロセッサからもアクセス可能	プロセッサ Processor1からのみアクセス可能(*2)
	CLS_P1_P1ALL CLS_P1_FLAT (*)			どのプロセッサからもアクセス可能
CLS_ALL_PRC1	CLS_P1ALL_P1ALL CLS_P1ALL_FLAT (*)	実行開始時プロセッサ Processor1。他のプロセッサへ移動可能		

(*) P1ALL と FLAT の違いは、前者はProcessor1からのアクセスが早いのに対し、後者はどのプロセッサからも同じ速度でアクセス可能であること
(*2) Processor2 からのアクセスには Processor1 へ RPC を介する

クラス間の結合

呼び元クラス ＼ 呼び先クラス	P1_P1	P1ALL_P1ALL	P1_FLAT	P1ALL_FLAT	P2_P2	P2ALL_P2ALL	P2_FLAT	P2ALL_FLAT
P1_P1	直結	直結	直結	直結	T-RPC	直結	直結	直結
P1ALL_P1ALL	T-RPC	直結	直結	直結	T-RPC	直結	直結	直結
P1_FLAT	直結	直結	直結	直結	T-RPC	直結	直結	直結
P1ALL_FLAT	T-RPC	直結	直結	直結	T-RPC	直結	直結	直結
P2_P2	T-RPC	直結	直結	直結	直結	直結	直結	直結
P2ALL_P2ALL	T-RPC	直結	直結	直結	T-RPC	直結	直結	直結
P2_FLAT	T-RPC	直結	直結	直結	直結	直結	直結	直結
P2ALL_FLAT	T-RPC	直結	直結	直結	T-RPC	直結	直結	直結

呼び元クラス、呼び先クラス： セルの属するリージョンに指定されたクラス

P1_P1, P1ALL_P1ALL,..: 接頭辞 “CLS_” を省略

直結： 直接結合の略、T-RPC：Transparent RPC の略

呼び先セルが、カーネルオブジェクトの場合は、いずれの場合も直接結合

TOPPERS

Transparent RPC について

- クラス間の直接結合できない場合、Transparent RPC† を使用して接続する

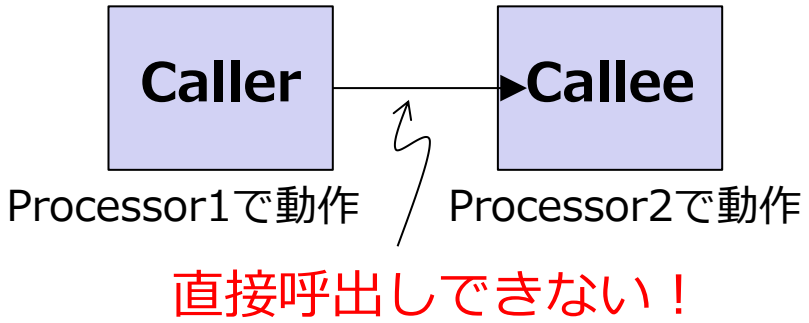
- 呼び先セル (動作可能プロセッサが制限されている) と処理単位の実行が同じプロセッサに限定されているものの FLAT なリージョンを生成する
  - 通常の Transparent RPC は呼び元セルの属するリージョンに RPC チャンネルセルを生成する

† Transparent RPC は、ポインタ透過性がある場合、すなわち共有メモリ型のマルチプロセッサ等で用いる RPC、一方、Opaque RPCは、ポインタ透過性がない場合、すなわちメモリ共有していない場合に用いる

元の CDL (RPCチャンネル挿入前)

```
[class(FMP, “CLS_P1_FLAT)]
region rP1_FLAT {
    cell tCaller Caller { cCall = rP2_P2::Callee.eEntry; };
};
[class(FMP, “CLS_P2_P2)]
region rP2_P2 {
    cell tCallee Callee {};
};
```

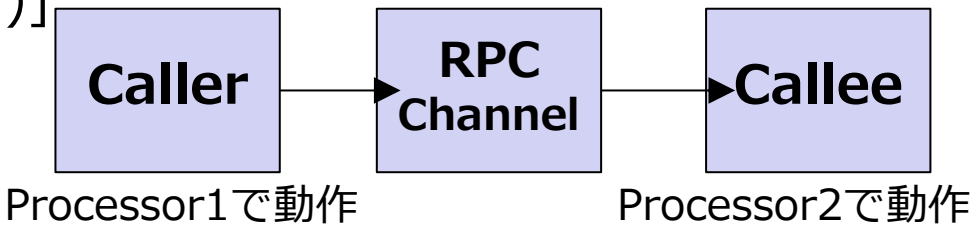
通常セルの動作が Processor2 に限定されているため、直接結合できない



RPC チャンネル挿入後の CDL

(TECS ジェネレータプラグインにより内部に生成されたCDL)

```
[class(FMP, "CLS_P1_FLAT")]
region rP1_FLAT {
    cell tCaller Caller { cCall = ~~rP2_P2::Callee.eEntry;~~ };
                                  ↓ ↓置き換えられる↓ ↓
                                  rCLS_P2_FLAT::RPCChannel. eThrough
};
[class(FMP, "CLS_P2_P2")]
region rP2_P2 {
    cell tCallee Callee {};
};
```




```
[class(FMP,"CLS_P2_FLAT")]
region _rCLS_P2_FLAT {
    cell tRPCChannel RPCChannel {
            cCall = rP2_P2::Callee.eEntry;
    };
};
```

プラグインにより自動生成されるチャンネル

Processor2 で動作するが、他のプロセッサからもアクセス可能なリージョンが自動生成される

TOPPERS

わからないときは

- TOPPERS 会員の皆さま
  - com-wg@toppers.jp … TECS WG の ML
  - dev@toppers.jp … 開発者 ML
- 非会員の皆さま
  - users@toppers.jp … ユーザー ML