TOPPERS/HRP3 における TECS の使い方

TOPPERS プロジェクト TECS WG

2018.5.6

目次

1. はじめに
2. TECS CDL の書き方
3. 呼び先がカーネルオブジェクト以外の場合の結合方法
4. 呼び先がカーネルオブジェクトの場合の結合方法

章目次

概要

- TOPPPERS/HRP3 での TECS の役割を説明する
  - TOPPERS/HRP3 の仕様であることを明示しない場合、TECS の仕様を説明
  - TECS 化以外については、TOPPERS 統合仕様書 V3.1 を参照
  - TECSジェネレータ V1.6.0 から TOPPERS/HRP3 を正式サポート
    - 本書は V1.6.1 での仕様を説明 (一部 V1.6.0 と相違)

特徴

- TOPPERS/HRP3の保護ドメインを活用するにおいて、TECS は必需品といっても過言ではない
- コンポーネント図上で、どのように保護するか検討できる
  - 図で検討できるため、保護を可視化できる
- その結果を TECS CDL に直接的に反映できる
- TECS CDL から、面倒な ATT_MOD の記述を自動生成
- ドメインをまたぐ際に必要なコードを自動生成

用語

- 保護ドメイン… TOPPERS/HRP3 カーネルで用いられる、メモリ保護の単位
- リージョン… TECS で用いられる、セルの配置を制御する単位
  - 指定によりノード、リンク単位、ドメイン、クラスとなる
  - HRP3 の保護ドメインは、TECSではドメイン指定されたリージョンが対応する
  - リージョン間のセルの結合を制限できる
  - ネームスペースの機能も併せ持つ
- ASP3
  - TOPPERS/ASP3 のことを、略して ASP3 と記載することがある
- HRP3
  - TOPPERS/HRP3 のことを、略して HRP3 と記載することがある

TOPPERS/HRP3 での TECS の役割

- ASP3 と同様に以下の目的で使用される

- システムログ部およびシリアルI/F部の TECS 化
  - ターゲットごとの改変を、コンポーネントの変更、置き換えにより対応（プロダクトライン的な使い方）
- TECSを使用したアプリ開発用に、カーネルオブジェクトを TECS コンポーネント化
  - TECS 版のサンプルアプリケーションも同梱

TECS 化のメリット

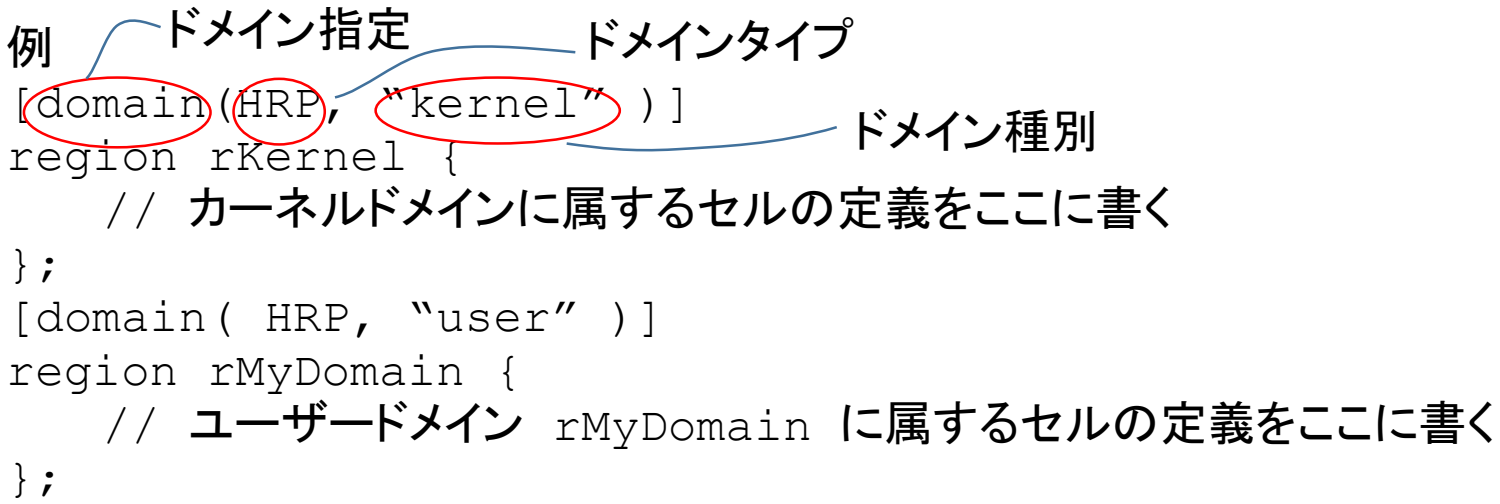
- ポーティング
  - シリアルI/F部を、ASP3 用と共通化できる
    - HRP3 では、カーネルドメイン用にコードを書くだけ
- アプリケーション開発
  - 保護ドメインへの分割を TECS のリージョンで対応
    - 保護ドメインごとに、分けてコードを生成
    - ATT_MOD の記述を自動生成するので、セルをリージョンに置くだけで保護を実現
    - アクセス許可ベクタ(アクセスパターン）を適切に設定
  - 保護ドメインをまたぐ呼出しのコードを自動生成
    - ユーザードメイン⇒カーネルドメイン
      - 拡張サービスコール
    - ユーザードメイン⇒他のユーザードメイン
      - RPC (リモート呼び出し）

章目次

保護ドメインとリージョン

- ドメイン指定されたリージョンが保護ドメインとなる
  - ドメイン指定の引数により、ドメインタイプ、ドメイン種別を指定する
  - ドメインタイプは TOPPERS/HRP3の場合 HRP、TOPPERS/HRP2の場合 HRP2
  - ドメイン種別は、ドメインタイプに依存する
    - HRP (HRP3)の場合 kernel, user, OutOfDomain(無所属) のいずれか
    - HRP2の場合 trusted, nontrusted, OutOfDomain のいずれか

例



保護ドメインとリージョン (2)

- ドメイン指定されたリージョンがドメインルートとなる
  - ドメインルートが、HRP3の保護ドメインとなる
- ドメインルート以外のリージョンは HRP3 カーネルの保護ドメインではない (TECS の結合制限のみ)

例

```
[domain(HRP, “kernel” )]
region rKernel {
    // カーネルドメインに属するセルの定義をここに書く
    region rSubKernelDomain {// ドメインルートは rKernel
            // セルまたは子リージョンをここに書く
    };
};
[domain( HRP, “user” )]
region rMyDomain {          //rMyDomain は HRP のドメイン名
    // ドメインに属するセルの定義をここに書く
    region rSubRegion{    //ドメインルートは rMyDomain
            // セルまたは子リージョンをここに書く
    };
};
```

保護ドメインとリージョン (3)

- 一つのノードには、一つのドメインタイプを指定できる
  - ドメインタイプ (HRP, HRP2) はノード内で一致しなくてはならない

不可の例

同一ノードで、異なるドメインタイプの指定は不可

```
[domain(HRP, "kernel")]
region rKernel {
    // カーネルドメインに属するセルの定義をここに書く
};
[domain(HRP2, "trusted")]// ドメインタイプHRP, HRP2不一致
region rKernel2 {
    // カーネルドメインに属するセルの定義をここに書く
};
```

リージョンの書き方のポイント

リージョンは、分けてかける

例

```
[domain(HRP,"kernel")]  // 初出でドメイン指定子を書く
region rKernelDomain {   // kernel.cdl で定義済み
};

[domain(HRP,"kernel")] // 2回目以降、指定できない
region rKernelDomain {  // 再び rKernelDomain
    // カーネルリージョンのセル
};
```

- rKernelDomain は、HRP3 の kernel.cdl で定義済みのため、ユーザーの CDL では 2回目以降の定義となり、rKernelDomain に対するドメイン指定子を記述することはない。

ドメイン種別

- HRP3 のドメイン種別は、3種類ある
  - カーネルドメイン kernel
  - ユーザードメイン user
  - 無所属 OutOfDomain
- それぞれの書き方を以下のページで説明

カーネルドメイン

- 標準のカーネルドメイン
  - kernel.cdl で定義されているドメイン

```
[domain( HRP, “kernel” )]
region rKernelDomain {
    // セルや子リージョンを書く
};
```

リージョン名は、ドメイン名に反映しない

- 他にもカーネルドメインを設けることができる (上級向け)
  - 例

```
[domain(HRP,”kernel”)]     // kernel を指定
region rKernelMyDriver {
        …
};
```

  - TECS CDL で結合を制限する（保護ドメインによる保護はない）

ユーザードメイン

- ユーザードメインの場合、リージョン名が保護ドメイン名となる
  - 以下の場合 `rMyDomain`
  - リージョン名が異なる場合、別のドメイン
- 同一ユーザードメイン間の結合に制限は、ない

```
[domain(HRP, “user” )]
region rMyDomain {
    cell tClient Client{
            cClient = Server::eServer;
    };
    cell tServer Server{
    };
};
```

リージョン名がHRP3 のドメイン名になる

- 異なるユーザードメインへの結合は、RPC となる
  - 呼び先のドメインに属するタスクにより実行させる

無所属

- ドメイン指定は、親リージョンにも波及し、HRP3 の無所属となる
  - ノード指定された親リージョンまで波及する
    - ルートリージョンは、暗黙的にノード指定されている

```
[domain(HRP, “kernel” )]
region rKernelDomain {//リージョン名は保護ドメインに反映されない
    // セルや子リージョンを書く
};

// 以下は、指定されてないくても HRP3 の無所属となる
cell tCelltype OutOfDomainCell {}; // 無所属のセル


region rImpliedOutOfRegion {//暗黙的に HRP3 の無所属(上級向)
    // 無所属のセルや子リージョンを書く
};

// 明示的に無所属を指定することもできる (上級向け)
[domain(HRP,”OutOfDomain”)]
region rOutOfRegion {
    // 無所属のセルや子リージョンを書く
};
```

章目次

3. 呼び先がカーネルオブジェクト以外の場合の結合方法

- 結合方法 (呼び先がカーネルオブジェクト以外の場合)
- 結合方法まとめ (呼び先がカーネルオブジェクト以外の場合)
- RPCによる接続

結合方法 (呼び先がカーネルオブジェクト以外の場合)

- 結合方法は、呼び元セルの所属ドメイン、呼び先セルの所属ドメインによって異なる
  - 同じドメインに属する場合は、直接結合となる
  - 異なるドメインの場合、結合方法がことなる

結合方法まとめ (呼び先がカーネルオブジェクト以外の場合)

呼び先セルのドメイン / 呼び元セルのドメイン	カーネル	ユーザー	無所属
カーネル	直接結合[*1,*4] または 不可[*2, *3]	RPC[*6] または 不可[*7]	直接結合
ユーザー	SVC[*5]	直接結合[*1] または RPC[*2, *6]	直接結合
無所属	SVC[*5]	RPC[*6] または 不可[*7]	直接結合

*1 同一リージョン またはドメイン
*2 異なるリージョン またはドメイン
*3 TECS CDL 上の結合制限, カーネルによる制限はない
*4 直接結合は、スループラグイン(RPC, SVC)によるセル挿入なし
*5 HRPSVCPlugin (拡張サービスコール)適用
*6 HRPRPCPlugin (リモート呼び出し)適用
*7 非タスクコンテキストは不可

RPCによる接続

- ユーザードメイン間は RPC により接続する
  - 呼び先セルの所属ドメインに属するタスクにより処理
  - 以下の例で Client ⇒ Server の結合は RPC により接続

    ** の箇所のようにスループラグインが指定されたイメージ

```
[domain(HRP, “user” )]
region rMyDomain {
    cell tClient Client{
            /* [through(HRPRPCPlugin),””]  **  */
            cClient = rMydomain2::Server::eServer;
    };
};
[domain(HRP, “user” )]
region rMyDomain2 {//rMyDomain2がHRP3のドメイン名になる
    cell tServer Server{
    };
};
```

- メモリ透過性を前提としない Opaque RPC をベースとした RPC

章目次

4. 呼び先がカーネルオブジェクトの場合の結合方法

- 結合方法 (呼び先がカーネルオブジェクトの場合)
- カーネルオブジェクト
- タイムイベント通知
- アクセス許可ベクタ自動設定のまとめ

結合方法 (呼び先がカーネルオブジェクトの場合)

- システムコール
  - 所属ドメインからの結合に制限は、ない
    - ユーザードメインからカーネルドメイン、異なるユーザードメインには結合できない
    - カーネルドメインから
  - しかし、tKernel の lockCpu などのように、カーネル側で呼出しを制限するものがある
    - もし lockCpu を信頼できないコードから呼び出せると、システムをフリーズさせてしまう可能性がある。delay なら問題は、ない

カーネルオブジェクト

- HRP3 のアクセス許可ベクタ(SAC)は、自動設定される
  - アクセス許可ベクタが未設定の場合、自動設定される
    - アクセス許可ベクタ　＝ カーネルによるアクセス制限
    - セルの属性 accessPattern1 ～ accessPattern4
- ユーザードメインのカーネルオブジェクトは、所属ドメインおよびカーネルドメインからのみアクセスできる
  - 従って、共用するカーネルオブジェクトは、無所属とする
  - **HRP3 では SAC により所属外からもアクセスできるが、TECS では、セルの CB へアクセスできないため、制限される**
  - **タスクは無所属にはできないため、属するドメインまたはカーネルドメインからのみ結合できる**
- 配置制限
  - タスクのセルは、カーネルまたはユーザードメインに配置できる
  - タイムイベント通知のセルは、「タイムイベント通知」の項を参照
  - 割込み関連3つ、CPU例外、初期化、終了ルーチンはカーネルドメインに配置できる
  - 複数のユーザードメインから結合されるカーネルオブジェクトは、無所属とする(アクセス許可ベクタは呼び元のドメインに設定される。ただし無所属のセルから結合されていると、すべてに許可を与える)

タイムイベント通知

- 配置制限

  イベント通知先、エラー通知先により配置可能なドメインが制限される

  - イベント通知先が、ハンドラの場合
    - カーネルドメインにのみ配置できる
  - イベント通知先がハンドラ以外の場合
    - カーネルドメインまたは、ユーザードメインに配置できる

アクセス許可ベクタ自動設定のまとめ

- 呼び先のセルの属性 accessPattern1 ~accessPattern4 に設定される値
  - 自動設定の場合、accessPattern1 ～ accessPattern4 は同じ値

		呼び先ドメイン種別		
		kernel	**user**	**OutOfDomain**
呼び元ドメイン種別	**kernel**	**TACP_KERNEL**	**TACP_KERNEL**	**TACP_KERNEL**
	user	**不可**	**TACP(callerDomain) †**	**TACP(callerDomain)**
	OutOfDomain	**TACP_KERNEL ‡**	**TACP(calleeDomain)‡**	**TACP_SHARED**

callerDomain は呼び元ドメインを、calleeDomain は呼び先ドメインを表す。

複数のドメインから結合される場合、ビット OR '|' を取った値が設定される。

† 異なるユーザードメインの場合、結合不可

‡ 呼び先のメモリのアクセス権がない場合、実行時エラーとなる

この表は TECS ジェネレータ V1.6.1 での実装を示す。V1.6.0 では、呼び先ドメイン種別が OutOfDomain の場合、すべて TACP_SHARED となる。また、不可の場合、ビルド時エラーとならない。実行時にCB にアクセスできず、メモリ保護違反となる。