

複数CPU間のための共有メモリ

小島 隆史 (中央大学大学院理工学研究科 國井研究室)

概要:

複数CPU間の共有メモリを同期し、さらに、要求があったクライアントコンポーネントに対しては、共有メモリを簡単に利用できるような枠組みを実現する。

このコンポーネントを介することによって、単一CPU、複数CPU間を意識することなく、共有メモリが利用できる。

特徴:

- ◆複数CPU間で共有メモリを仮想的に使うことができ、高速な通信が実現できる。
- ◆同期機能は、データの送信と、受信を同時に行える
- ◆クライアントは、サイズと、特定の文字列のみを送るため、負荷が少ない。

※同期通信速度はRTコンポーネントの通信速度に依存

インタフェース:

サービスポート:

用途、大きさによる共有メモリID及び補助用セマフォを提供する

出力ポート:

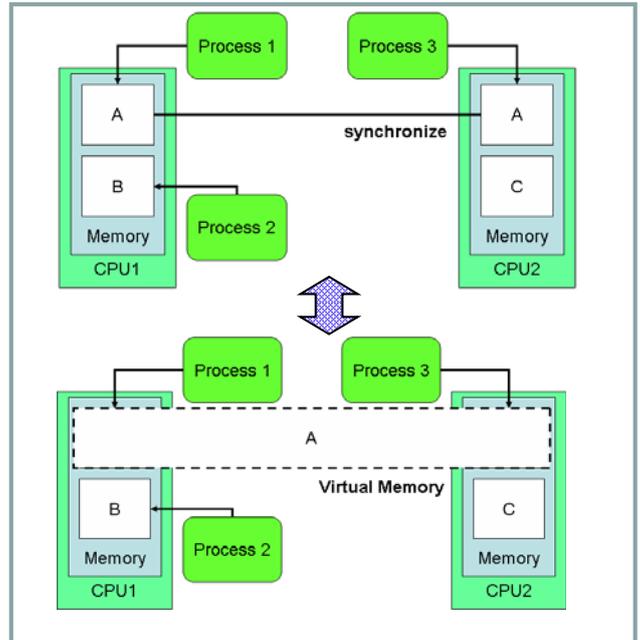
同期ポート (他同コンポーネント接続し、同期をおこなう)

入力ポート:

- ①同期ポート (出力ポートに同じ)
- ②更新通知ポート (クライアント側から更新通知を受け取る)

ライセンス (公開条件):

RTミドルウェアの公開条件に準拠



連絡先:

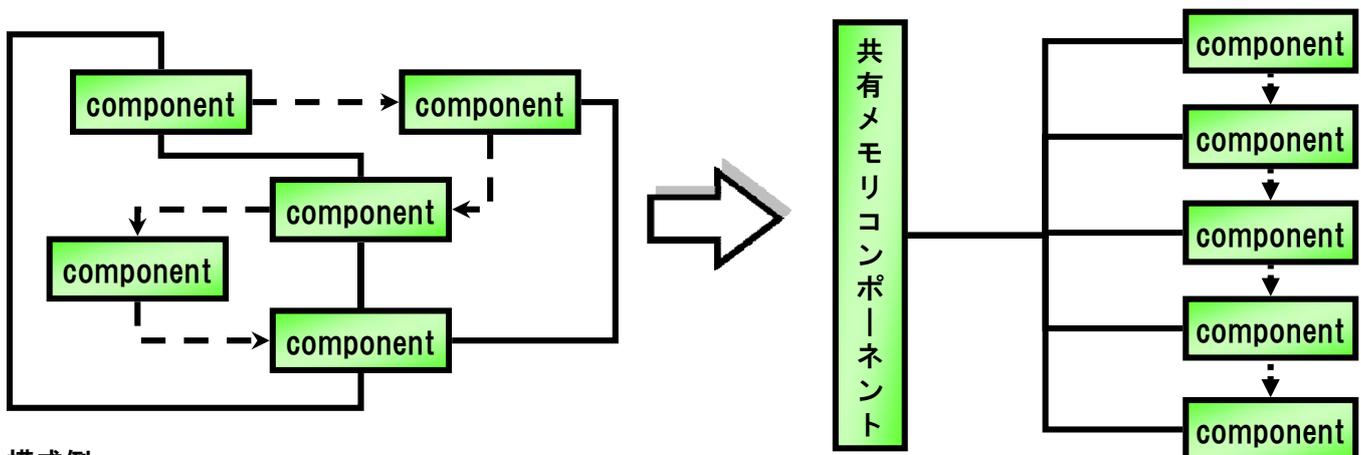
中央大学 國井研究室

URL:

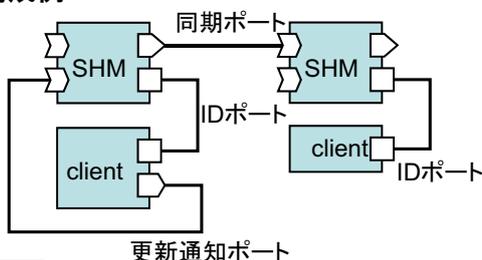
<http://www.elect.chuo-u.ac.jp/kunii/index.html>

複数CPUのための共有メモリコンポーネントは、通常、1つのCPU内でしか解決できない共有メモリをRTミドルウェアの通信機能を使って同期機能を提供しています。これにより、CPU間のデータ共有や、接続をRTC-Linkなどを使うことで、簡単に変更することができます。

独自のデータ形式による受け渡しを行うコンポーネントが多い場合、独自のデータ形式は、共有メモリを介して行うことができるため、コンポーネントの追加・削除のための依存解決コストを抑え、開発速度を高めます。また、データの管理を一元化することで、構成を単純化し運用がおこなえるようになります。



構成例



簡単な同期確認であれば例のように構成することで確認ができます。

開発環境

OS: Fedora core 6 (kernel:2.6.18-1.2798)

Compiler: gcc 4.1.1-30

CORBA: omniORB 4.0.7

ACE: ace 5.5.4

OpenRTM-aist: OpenRTM-aist-0.4.1-RELEASE