

3L-2

XDM分散機能

XDM/DFの 分散AP機能について

片山 一浩† 石川 博道† 木村 淳美‡

(†(株)日立製作所, ‡日立西部ソフトウェア(株))

1. はじめに

XDM分散機能では、大規模システムの構築に伴う負荷分散、危険分散を実現するために、分散DB機能および、高頻度アクセス時の通信オーバーヘッド軽減を目的とした分散AP機能として同期付きトランザクション機能を提供している。

本稿では、同期付きトランザクション機能におけるトランザクション処理方式と効率的資源管理方式について示す。

2. 同期付きトランザクション機能

データを分散配置したシステム間において分散配置されたデータをアクセスする場合、同一ノードに対して高頻度のデータアクセス要求が発生すると、各処理要求(データアクセス)毎に処理要求元(クライアント)と処理実現元(サーバ)間の通信オーバーヘッドがかかるという問題発生する。そこで、この問題を解決する一手段として同期付きトランザクション機能が考案された。

同期付きトランザクション機能は、分散AP機能の一種で処理実現元(サーバ)に予め実行する処理(トランザクション)を定義したUAPを起動しておき、該当処理要求発生時に処理要求元(クライアント)UAPからRPC(Remote Procedure Call)することにより処理を開始する。これにより、高頻度データアクセスを実行する処理の場合など、この処理(トランザクション)を定義しておくことで、処理要求元(クライアント)との通信オーバーヘッドを低く押さえることが可能である。また、各ノードに分散配置されているデータを処理要求元(クライアント)システムにおいて分散管理する必要がなく、各トランザクション処理は各処理実現元(サーバ)に閉じた処理として扱うことが出来る。

XDM/DFでは、RPC(Remote Procedure Call)を処理要求元(クライアント)UAPからのSEND/RECEIVE命令を発行することにより行い、必要情報をパラメタとしてサーバにメッセージ送信することで実現している。また、各トランザクション処理の整合性は、トランザクション処理に2相コミット方式を採用することにより実現している。

図1に当該機能の概略図を示す。

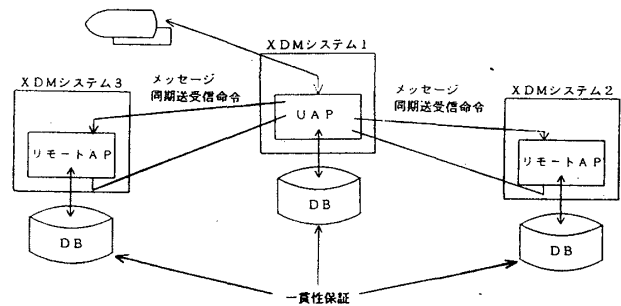


図1. 分散AP機能の概要

3. トランザクション処理

3.1 資源管理方式

XDMでは、各処理単位の資源をアクティビティという単位により管理している。アクティビティとは、各サブシステム間にまたがるトランザクション処理の一貫性を保証する単位であるとともに、処理要求元(クライアント)トランザクション処理と処理実現元(サーバ)トランザクション処理を関連付けるものである。このアクティビティは、各サーバに対するトランザクション処理要求ごとに確保する必要があるが、トランザクション処理実行時、一つのサーバでは一つの資源しか管理出来ない。このため、資源の保有はトランザクションの生成と同期する必要がある。この流れを示すと、サーバトランザクション開始と同時にアクティビティが割当てられ、資源を確保してトランザクション処理を実行する。そして、サーバトランザクション

Function of Remote AP for

Distributed Access Service XDM/DF(Extensible Data Manager/Distributing Facility)

Kazuhiro KATAYAMA†, Hiromichi ISHIKAWA†, Atsumi KIMURA‡

†:Hitachi, Ltd ‡:Hitachi Seibu Software Co.,Ltd.

処理終了と同時にアクティビティも終了して資源を解放する。この様に、各トランザクションの資源をアクティビティ単位で管理することにより資源確保/解放とトランザクション処理開始/終了の間には密接な関係が必要となった。図2にトランザクション処理と資源保有状態の関係を示す。

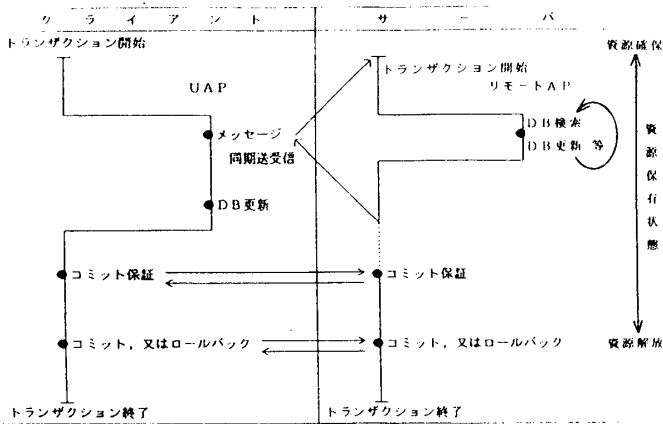


図2. トランザクション処理と資源保有状態

3 2 トランザクション処理と資源管理方式

同一処理実現元（サーバ）に対し連続して同期付きトランザクション処理を実行する場合を考えると、当該機能では、各トランザクション処理要求単位に資源を確保することとなるため、処理要求元からの一つのトランザクション処理に対して複数資源を保有する状態に陥る可能性が極めて高くなる。このため、資源確保/解放のタイミングがトランザクション処理に大きな影響を与えることとなる。これは、処理実現元（サーバ）で処理を実行するUAPを予め起動しておくという当該機能の性格上、回避し難いものである。このため、この問題点を解決する方法として、処理要求元（クライアント）トランザクション処理において、同一資源を引き続き使用する可能性は極めて低いという当該機能使用時のトランザクション処理の実用性を考慮し、一つの処理実現元（サーバ）では一つの資源しか保有しないという前提を設けるのが有効である。また、処理要求元（クライアント）と同期をとって処理実現元（サーバ）トランザクション処理を実行するため、通常の資源管理方式のように確保した資源を処理要求元（クライアント）トランザクション処理が終了するまで解放しないという方式ではなく、確保した資源は早急に解放する必要があるため、データの整合性を保証出来る最も早い時点（同期点）で資源の解放をする独自のプロトコルを採用したトランザクション処理を実行する。これにより、資源管理を効率良く使用するトランザクション処理の実行が可能となった。図3にトランザクション処理と資源管理方式を示す。

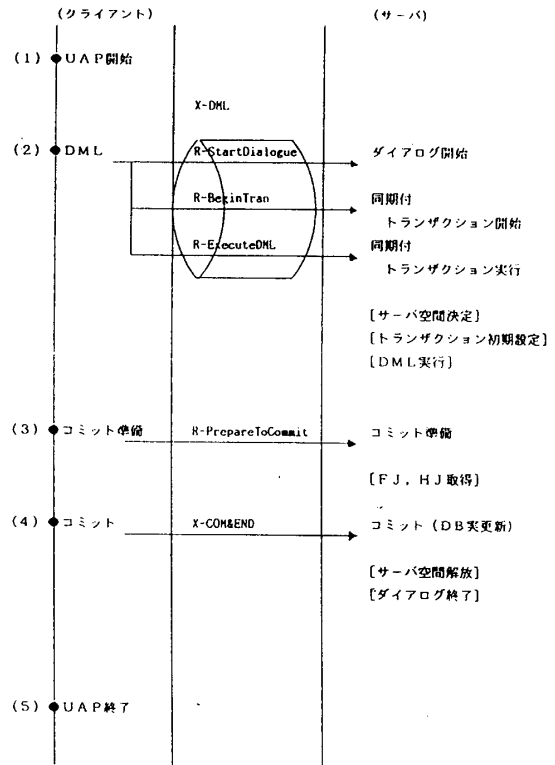


図3. トランザクション処理と資源管理方式

4. おわりに

本稿では、XDM分散機能における分散A P機能（同期付きトランザクション機能）におけるトランザクション処理と効率的資源管理方式について報告した。この機能と分散DB機能（分散データベースアクセス機能）を混在させることにより、柔軟なシステム構築、性能を考慮したトランザクション処理の実行を可能とした。

参考文献

山中 治 他：分散アクセスサービス機能
XDM/DFにおける
サービスプロバイダのメカニズム
第39回情報全大
千々和 啓江 他：分散アクセスサービス機能
XDM/DFにおける
コミットメント制御最適化の一考察
第39回情報全大
村垣 委久夫 他：分散アクセスサービス機能
XDM/DFにおける
トランザクション回復機能
第39回情報全大