

Actian Zen のキャッシュエンジンおよび Reporting Engine



株式会社エージーテック

2019 年 9 月 13 日

免責事項

株式会社エージテックは本書の使用を、利用者またはその会社に対して「現状のまま」でのみ許諾するものです。株式会社エージテックは、いかなる場合にも本書に記載された内容に関するその他の一切の保証を、明示的にも黙示的にも行いません。本書の内容は予告なく変更される場合があります。

商標

© Copyright 2019 AG-TECH Corp. All rights reserved. 本書の全文、一部に関わりなく複製、複写、配布をすることは、前もって発行者の書面による同意がない限り禁止します。

すべての **Pervasive** ブランド名および製品名は、**Pervasive Software Inc.** の米国およびその他の国における登録商標または商標です。また、すべての **Actian** のブランド名は、**Actian Corporation** の米国およびその他の国における登録商標または商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。

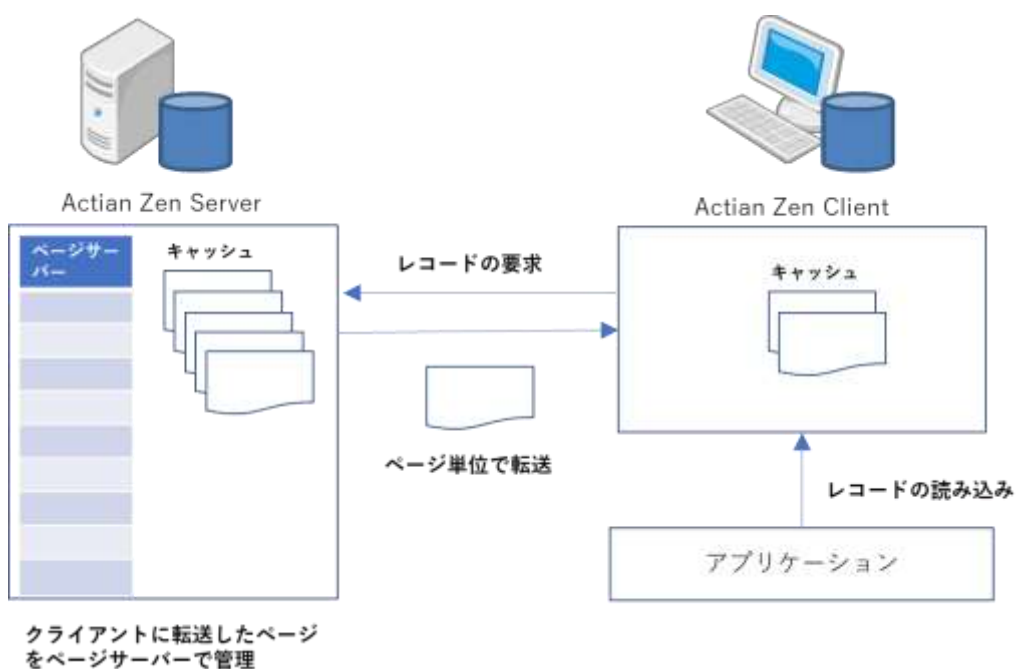
Actian Zen のキャッシュエンジンおよび Reporting Engine

最終更新：2019 年 9 月 13 日

キャッシュエンジン

ここでは Actian Zen のキャッシュエンジンの動作について説明します。

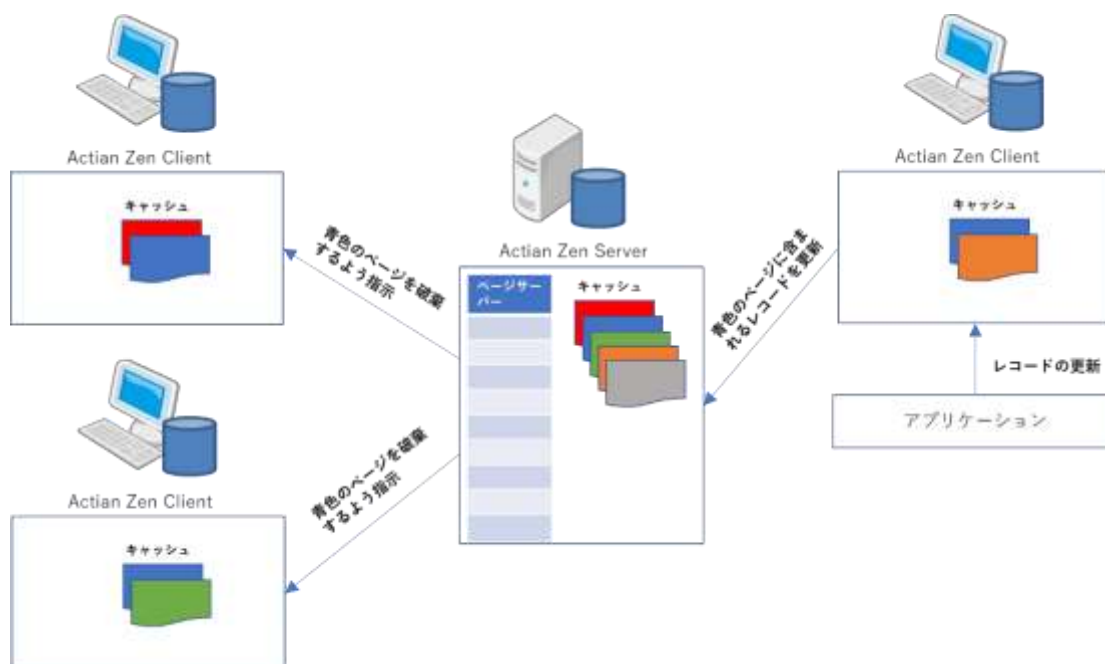
キャッシュエンジンが有効な環境でクライアントアプリケーションからレコードの読み込みを行うと、サーバー側では、該当レコードに関連するページがまとめて転送されます。この時、サーバー側のページサーバーがクライアントに転送したページの情報を記録し管理します。



サーバー上でデータが更新されると、ページサーバーはクライアントに転送したページ情報をチェックし、該当ページが含まれると、追跡テーブルに無効とマークします。デフォルトでは、5 秒ごとに追跡テーブルを検索し、全てのクライアントに該当ページを破棄するよう通知を行います。

キャッシュエンジンにデータがキャッシュされている状態でアプリケーションがデータの読み込みを行うと、キャッシュエンジンは、キャッシュ上で該当データを検索します。該当データがキャッシュ上に存在すれば、サーバーにはアクセスせず、アプリケーションにデータを返します。

アプリケーションでデータを読み込む際、レコードロックを行うと、キャッシュエンジンはキャッシュ上のデータにはアクセスせず、サーバー上のエンジンから読み込みます。この時、サーバーからは、読み込むデータが含まれるページが送られ、キャッシュされます。



クライアントのキャッシュエンジンでは、該当ページが破棄されますが、更新されていないページは引き続きキャッシュされます。

つまり、一度キャッシュしたページは、ページサーバーからの破棄通知を受け取るまで有効で、更新されないデータは、サーバーから 1 度しか読み込まないこととなります。

アプリケーションが読み込みを行うデータがクライアントのキャッシュに存在していた場合に、サーバー上で更新されていると、読み込んだデータとサーバー上のデータが異なることになります。

このような場合に、データの更新 (update オペレーション) を行うと、ステータス 80 (MicroKernel でレコード レベルの矛盾が発生しました。) がアプリケーションに返ります。アプリケーションは、データの再読み込みを行い、最新データを読み込んだ後に、データの更新を行う必要があります。

例えば、商品在庫ファイルの在庫数を変更する場合、ステータス 80 を受け取った時点で、在庫数は変更されていると考えられます。

ステータス 80 を受け取る前の在庫数が 100 で、在庫から 50 個引き当てる場合、残りの在庫は 50 個ですが、ステータス 80 を受け取った時点の在庫数が 80 個であった場合、更新後の在庫数は 30 個でなければいけません。

レコードを読み直し、在庫の引き当てをやり直すことで、誤った在庫数になることを防ぎます。

クライアント 1		クライアント 2	
商品在庫ファイルから商品番号 200 のレコードを読み込み	商品番号 200 の在庫数は 100		
		商品在庫ファイルから商品番号 200 のレコードを読み込み	商品番号 200 の在庫数は 100
在庫数から 20 個在庫を引き当て	100 - 20		
商品番号 200 のレコードを更新	商品番号 200 の在庫数は 80		
		在庫数から 50 個在庫を引き当て	100 - 50
		商品番号 200 のレコードを更新	ステータス 80 が返ります
		商品番号 200 のレコードを再読み込み	商品番号 200 の在庫数は 80
		在庫数から 50 個在庫を引き当て	80 - 50
		商品番号 200 のレコードを更新	商品番号 200 の在庫数は 30

Reporting Engine

Reporting Engine は、クライアント側の SQL エンジン (SRDE) が SQL の解釈を行い、キャッシュエンジンを介して、サーバー上の Btrieve ファイルにアクセスします。該当データがキャッシュエンジンにキャッシュされている場合、サーバーへのアクセスは行われません。

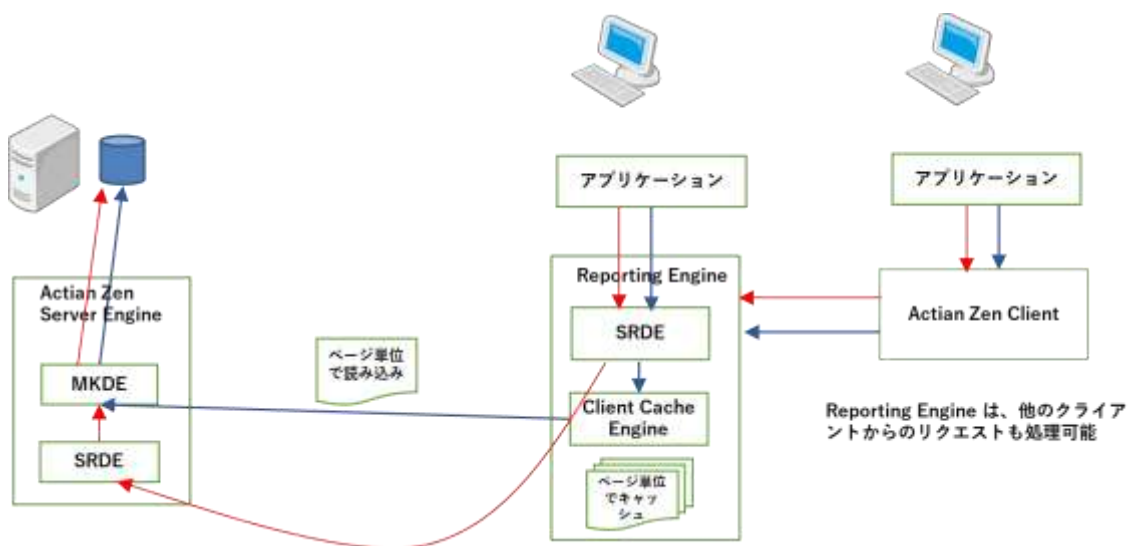
このことから、大量にデータの読み込みを行う場合、サーバーへのアクセスが減少し、パフォーマンスが向上します。

キャッシュエンジンにキャッシュされていないデータは、サーバーからページ単位で読み込みます。

データの更新 (追加、変更、削除) を行う場合は、サーバー上の SQL エンジンに処理をリクエストします。

Reporting Engine を使用することで、負荷分散ができ、データの読み込みも高速化されます。

また、Actian Zen v13 R2 からは Reporting Engine で他のクライアントからのリクエストを処理することが可能となり、これにより更にサーバーの負荷を小さくできます。



<メモ>

Reporting Engine にアクセスするクライアントは、サーバーエンジンでカウントされ、ライセンスを消費します。

上図の例では、2-User 分のライセンスが必要となります。