



Technical Note 04-29

キャッシュ書き込み

By ?

Technical Note 04-29

(原題: Optimizing Writing to the Cache with 4th Dimension)

概要

ディスクへのキャッシュ書き込み（インデックスページ、レコード、削除マーカ）は 4th Dimension 2003 で最適化され、パフォーマンスの向上に貢献しています。とりわけ遅いディスク、MacOS、膨大なデータという組み合わせではパフォーマンスの向上が顕著です。

内容

キャッシュとは何か

キャッシュの中身

バージョン 2003 における最適化

バージョン 2003 でのメモリの設定

キャッシュとは何か

キャッシュは、ハードディスクに対するアクセスを最適化するためのメカニズムです。キャッシュの中には、ディスクへの書き出しやディスクからの読み込みをされようとしているデータが納められています。

レコードの一部をキャッシュに置いておくことで、ディスクアクセスを幾らか軽減することができます。4D は特定のデータを必要とする場合、まずキャッシュの中を探します。キャッシュの容量はディスクと比べると非常に限られているので、通常はすべてのデータをキャッシュに納めることはできません。キャッシュにデータがない場合、4D はディスクからそのデータをロードし、またキャッシュにもそのデータを書き込みます。この時点では速度面の利点はありません。しかし、次に同じデータが必要になった時には、キャッシュから読み込むことができるので、処理はずっと速くなります。

したがって、無数のレコードを更新して保存したつもりでも、実際にはキャッシュに保存されただけでディスクには書き込まれていない場合があります。4D がキャッシュをディスクに保存するのは一定の時間が経過したとき（デフォルトで 15 分）、あるいはキャッシュが満杯になったときです。仮に停電が起これば、キャッシュが消えてしまい、いまだディスクに保存されていなかったデータは失われてしまうことになります。そのような事態を防ぐ為にも、UPS を備えておくのは賢明です。

キャッシュの中身

4D を起動すると、メモリの連続した領域が確保され、キャッシュメモリとして使用されます。その中には次のものが納められます。

- ディスクから出し入れされるレコード
- ビットテーブル（ビットマップとも呼ばれる）
- レコードアドレステーブル
- インデックスページ（プライマリおよびセカンダリ）
- インデックスアドレステーブル
- カレントセクション
- その他

Windows のキャッシュメモリ

データベースを起動すると、キャッシュメモリは 1 ブロックとして確保され、実際のキャッシュとして使用されます。このブロックはロックされており、したがってディスクとスワップすることはできません。メモリの残りの領域はロックされていません。キャッシュに必要なメモリが確保できないと 4D は強制終了します。

MacOS のキャッシュメモリ

「新しいメモリ割り当て方式」を使用すると、メモリ確保の最小値と最大値を指定することができます。4D は最大のサイズを確保しようとし、できない場合は、確保できるまで 32K ずつサイズを減らしながら再試行をします。最小のサイズすら確保できないと 4D は強制終了します。

クライアント/サーバのキャッシュメモリ

キャッシュはサーバで管理されるので、クライアントにはキャッシュがなく、レコードがクライアントにロードされる時はサーバのキャッシュからクライアントのメモリにコピーされます。レコードはそのままクライアントのメモリに留まり、保存をしたときに始めてサーバに送り返されます。

バージョン 2003 における最適化

バージョン 6.8 では、キャッシュのデータを保存するときの物理的な順番は、データがキャッシュにコピーされた順番で決まっていた。コピーされた順番は、ほとんどの場合、データがディスクに保存される順番とは違うものです。したがってキャッシュをディスクに書き込むときには、ハードディスクの中で頻繁な往復動作が生じていました。メモリアクセスはナノ秒単位の世界であるのに対し、ハードディスクアクセスはミリ秒単位の世界なので、データの量が多く、またハードディスクの速度が遅い場合には、書き込みにかなりの時間を要することがありました。

バージョン 2003 では、ハードディスクへの書き込みの前に 4D がキャッシュのデータを

レコードアドレス順に並び替えて保存するので、ハードディスクの往復動作がなく、一定方向に向かってデータを書き込むことができます。これにより、かなりの程度、保存に要する時間を削減することができました。

この新しい仕組みはデフォルトで有効にされていますが、SET DATABASE PARAMETER コマンドで変更することができます。両方の設定を使用できるので、それぞれのパフォーマンスを実際に確かめてみる事が可能です。

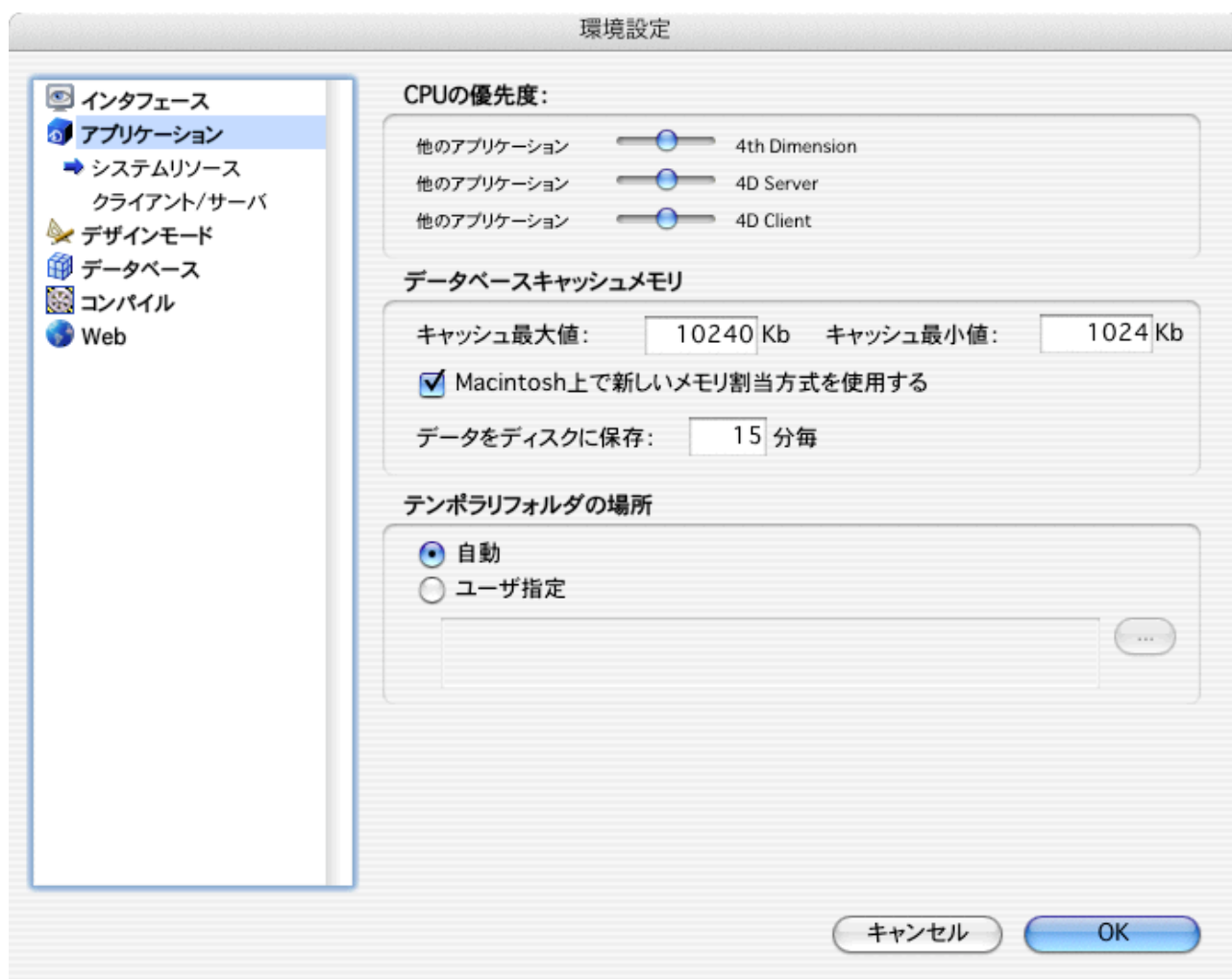
セレクトの番号は 26 で、値には 0 (enabled) あるいは 1 (disabled) を設定します。

いずれの設定を使用するにしても、データファイルの構造つや、キャッシュの構造は同じです。設定が関係するのはデータをディスクに保存するときの動作のみです。

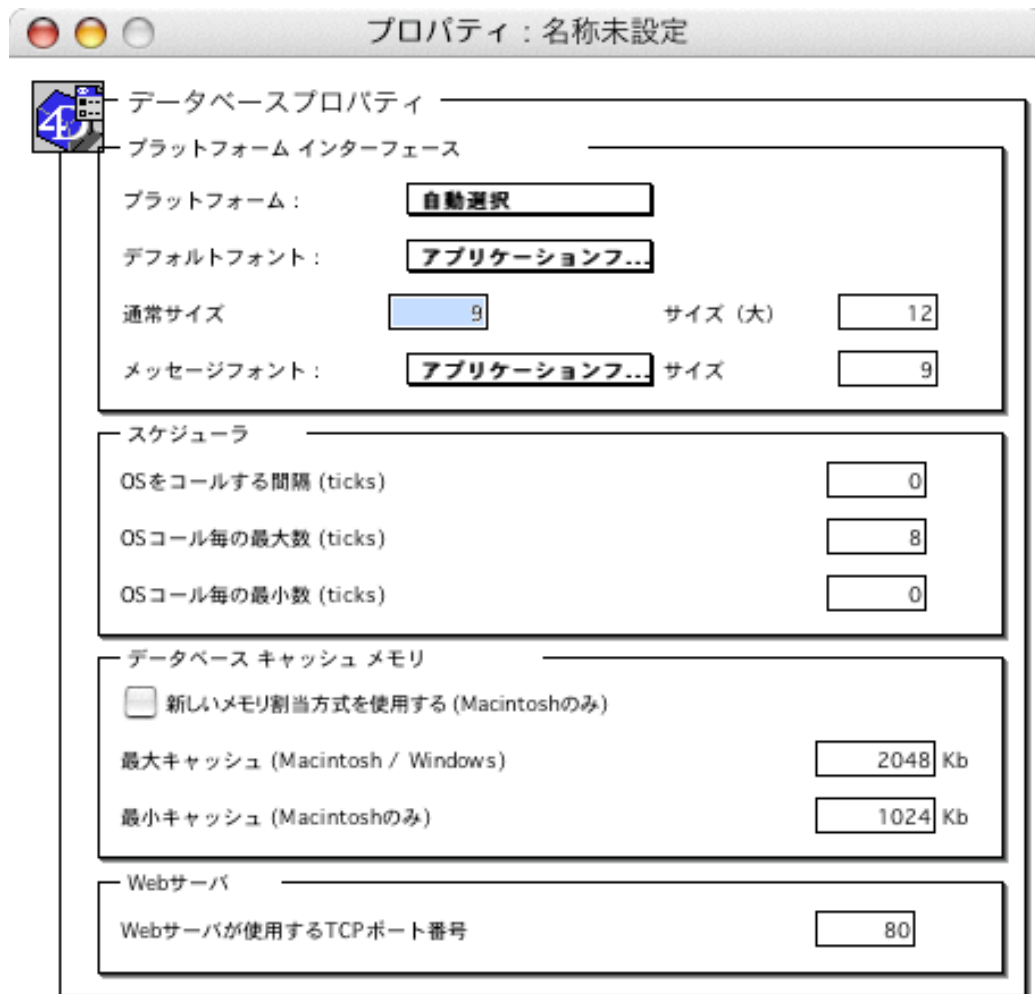
バージョン 2003 でのメモリの設定

キャッシュのサイズを設定する方法は 2 とおり存在します。

環境設定による方法



Customizer Plus による方法



キャッシュの最適な設定値

これには決まった答えというものがなく、幾つもの要素を考慮に入れて、最適の値を決めなくてはなりません。考慮に入れる要素には、次のようなものがあります。

データファイルのサイズ
キャッシュで扱うデータの種類
行なう処理の内容
クライアントの数
インデックスの数
テーブルの数

基本的に、キャッシュが大きいほどパフォーマンスは向上します。しかしながら、余りにキャッシュを大きくすると、他のアプリケーションにしわ寄せがゆき、4D レベルでみても願わしい結果にはならないものです。

それで、最適なキャッシュサイズを特定するためには、ランタイムエクスプローラでキャッシュの統計を監視し、キャッシュヒット率が 100%に近づくまで、少しずつキャッシュを増やしてゆくのが理想的です。

