



IBM i World 2021

IBM i

IBM i お役立ち機能紹介 - 第3回 -

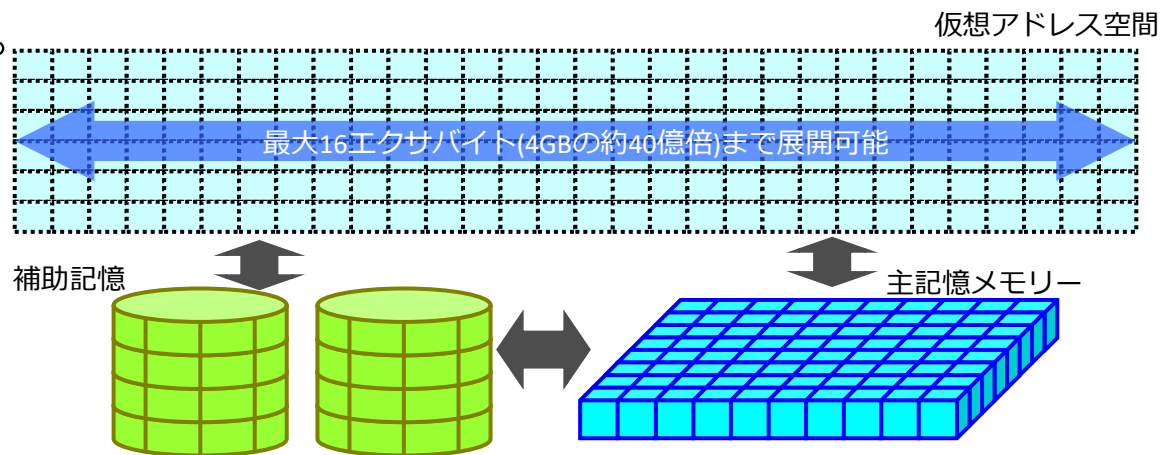
一番ホットなDBをインメモリーDBにしてみよう

日本アイ・ビー・エム株式会社
テクノロジー事業本部
IBM Powerテクニカルセールス
澤田英寿

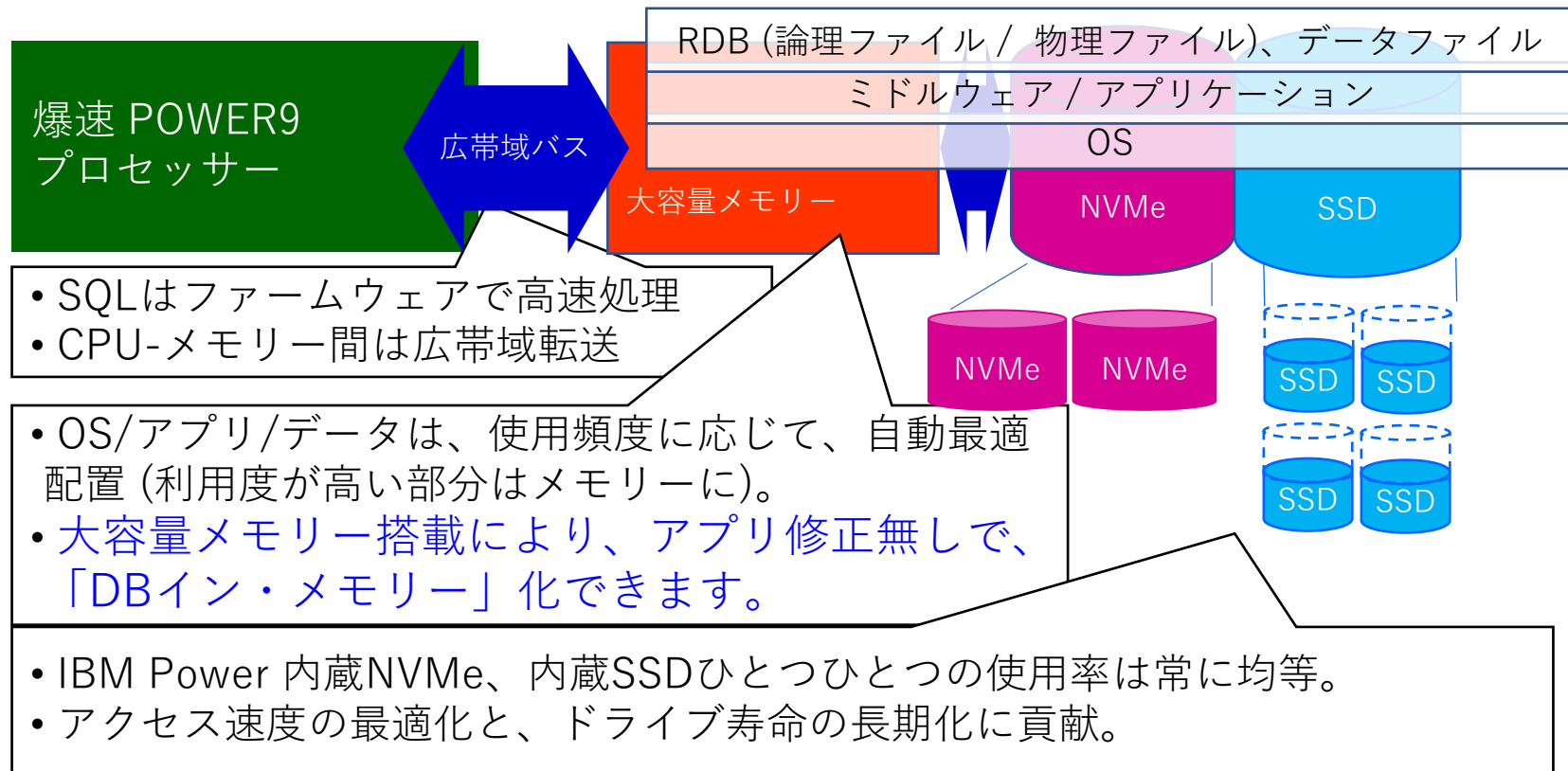
IBM i における「インメモリー」データベースのサポート

1) IBM i のアーキテクチャー (前提知識)

IBM iでは単一レベル記憶 (SLS: Single Level Store) を採用しています。単一レベル記憶とは、仮想メモリー概念を発展させたもの。メモリーと補助記憶 (ディスク) 全体を合わせて1つの仮想アドレス空間と見なし、ユーザーから見ると、オブジェクトが、主記憶メモリー上にあるのか、NVMe/SSD/HDD上にあるのかを意識せずにアクセスすることができます。



IBM i 単一レベル記憶を支えるテクノロジー



IBM i における「インメモリー」データベースのサポート

2) SETOBJACC コマンド

IBM i には、SETOBJACC というコマンドがあります。このコマンドで、目的のオブジェクト（ファイルまたはプログラム）を明示的に主記憶メモリー上にロードすることが可能です。

例えばバッチ処理で使用する大容量ファイルをバッチ実行前に、SETOBJACC コマンドでメモリー上にロードすることで、処理時間を大幅に短縮できます。

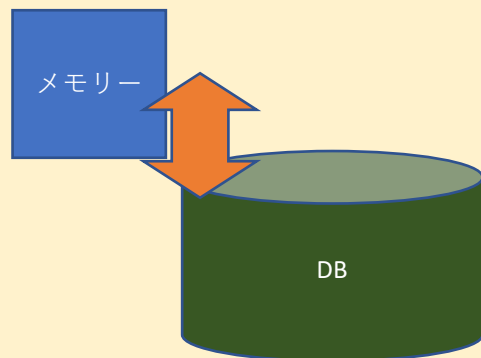
当機能でインメモリーデータベースを実現するためには、対象となるファイル・オブジェクトをロードするのに十分な主記憶メモリー容量があることが前提となります。

(例) 物理ファイル (TOKMSP) を *BASE プールに入れる場合は、下記コマンド
SETOBJACC OBJ(QEOL/TOKMSP) OBJTYPE(*FILE) POOL(*BASE 2)

十分な主記憶メモリーがあれば、
アプリ修正なしにインメモリーDBに移行できます。

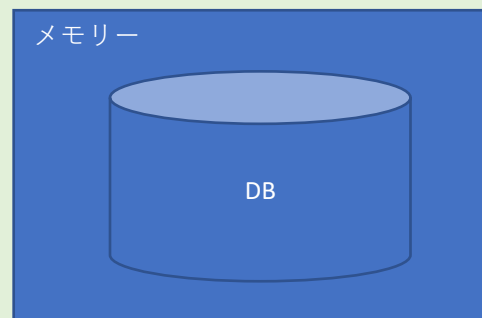
- DBイン・メモリーの利点

メモリーサイズ < DBサイズ の場合



ストレージI/Oが頻繁に発生
→ パフォーマンスに課題

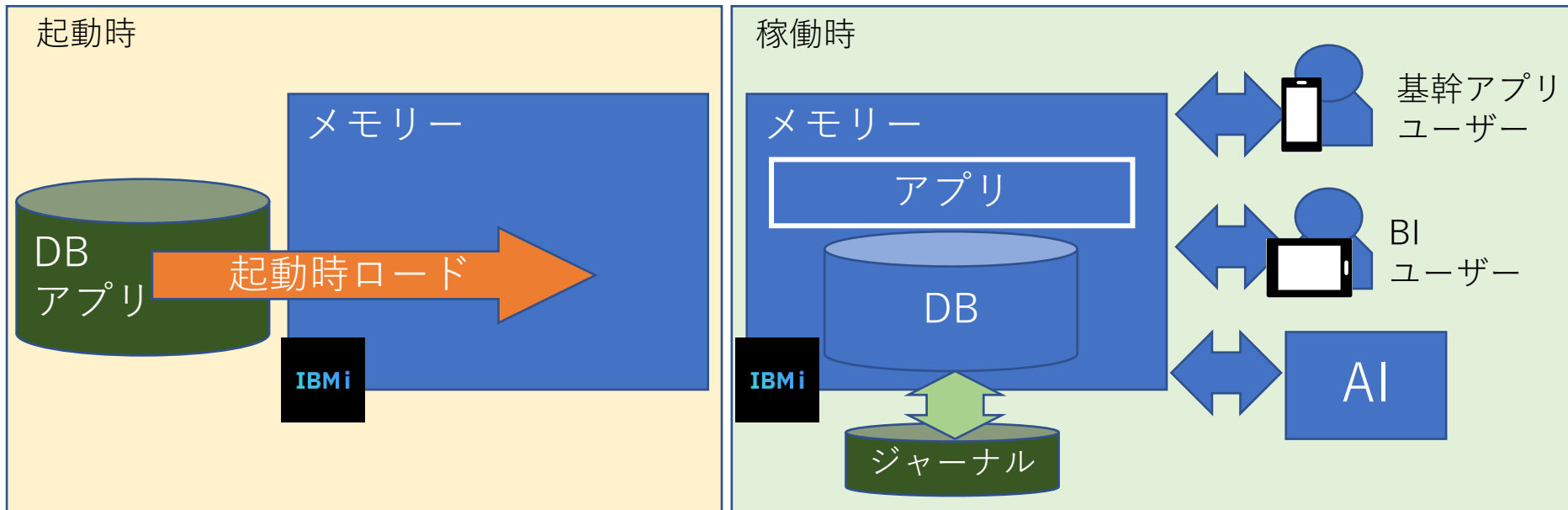
メモリーサイズ > DBサイズ の場合



ストレージI/Oがデータのロード時のみ
それ以降は高速メモリーにアクセス
→ パフォーマンスを最大化

IBM i ならアプリの変更は一切不要

DXで増えるユーザー数や、DXのヘビーなアプリも、サクサク対応できます



- SETOBJACC命令を、起動時に自動実行するようにセットすれば、IPL直後よりデータベースとアプリケーションをメモリー上にロード。アプリ修正不要。
- ジャーナルをNVMeやSSDで構成したASPに書き出しすることで障害発生時のDB保護も可能。

IBM i における「インメモリー」データベースのサポート

2) SETOBJACCコマンドのパラメータ

SETOBJACCコマンド使用法 (主なパラメータ)

1. オブジェクト名 (OBJ):
対象となるオブジェクト名を指定
2. オブジェクトタイプ (OBJTYPE):
*FILEまたは*PGMを指定
3. 記憶域プール (POOL):
ロード先のPOOLを指定。*PURGEを指定するとメモリー上からオブジェクトを除去できます
4. メンバー (MBR):
対象がファイルの場合、データベースファイルメンバーを指定
5. メンバーデータ (MBRDATA):
対象がファイルの場合、データかアクセスパスか、その両方かを指定できます。

IBM i における「インメモリー」データベースのサポート

3) IBM i 7.1以降での新しいインメモリー用コマンド

前述のSETOBJACCコマンドとは別に、IBM i 7.1以降ではCHGPFやCHGLFコマンドのメモリー保管パラメータ (KEEPINMEM) で、物理/論理ファイルをメモリーに保管することができます。KEEPINMEMで*YESを指定すると、初めてSQL言語でオブジェクトにアクセスした時に、主記憶メモリーにロードされます。

ただし、この操作が有効なのはSQL言語環境を使用する場合のみです。

KEEPINMEMの*YESは、RPG/COBOLでのDBアクセスには無効です。

READ/WRITE方式ご利用の場合は、SETOBJACCコマンドをご利用ください。

(例) 物理ファイル (TOKMSP) をメモリー上に入れる場合は、下記コマンド
CHGPF FILE(QEOL/TOKMSP) KEEPINMEM(*YES)

(参考資料) <https://ibm.ent.box.com/s/67j3bh05mu011ggtnnot30yv9v2htp7g>

SETOBJACCコマンドとCHGPF/CHGLFのKEEPINMEMパラメータ設定との違い

	メリット	デメリット
SETOBJACCコマンド	<ul style="list-style-type: none">• RPG/COBOLREAD/WRITE命令とSQLコマンドの両方で利用できる• データベースとアプリケーションプログラムの両方で利用できる	<ul style="list-style-type: none">• メモリーロードがシングルスレッド実行のため、ロードに多少時間を要する
CHGPF/CHGLFのメモリー保管パラメータ (KEEPINMEM) の利用	<ul style="list-style-type: none">• メモリーロードが並列スレッドのため、ロードが高速である	<ul style="list-style-type: none">• SQLコマンドを使用するプログラム、スクリプト、コマンドラインでしか利用できない (RPGやCOBOLのREAD/WRITEでは利用できない)• データベースのみ利用可能 (アプリケーションプログラムのインメモリー化はできない)

IBM iにおける「インメモリー」データベースのサポート

4) IBM iのエクスパートキャッシュ機能について

SETOBJACCや、CHGPF/CHGLFでのメモリー保管パラメータ (KEEPINMEM) を利用するには、対象となるDBやアプリケーションプログラムの所在 (ライブラリー名やファイル名) を理解している必要があります。

IBM iには、これらのコマンドとは別に、エクスパートキャッシュという、システムが自動的に、余裕のある主記憶メモリー資源を活用する機能があります。

インメモリーDB化のための機能とは少し異なりますが、主記憶メモリーの有効活用機能として併せてご紹介いたします。エクスパートキャッシュは、メモリープールに、キャッシュ機能を機能させる設定をするだけで利用できます。対話型ジョブのプールでエクスパートキャッシュを有効にすると、応答時間の短縮、バッチプールを有効にすると、処理時間の短縮になります。ただし、効果のある場合と、ない場合があります。

<効果が期待できない>

- ・メモリー資源が不足している状態で、ページフォルトが多い
- ・該当ジョブが、プロセッサーによる制限がある。

<効果が期待できる>

- ・メモリー資源に余裕がある
- ・HDDご使用の場合、ディスクアームの使用率が40%を超えている。
- ・プロセッサーの使用率が低いパフォーマンスが悪い
- ・データの参照方法が順次処理

具体的は設定方法は次ページです。

エキスパートキャッシュを有効にする方法

WRKSYSSTSコマンドで、PAGINGオプション欄を*FIXEDから、*CALCに変えるだけで、エキスパートキャッシュは有効になります。

WRKSYSSTSコマンドで、F11を複数回押すと下記画面になります。

例えば、システムプール2の「PAGINGオプション」を「*CALC」に変更すると、エキスパートキャッシュが有効になります。

システム状況

```

CPU 使用 % . . . . . : .7
経過時間 . . . . . : 00:01:36
システム内のジョブ . . . . . : 688
永久アドレス % . . . . . : .009
一時アドレス % . . . . . : .012
    
```

変更データを入力して（可能な場合），実行キ

システム プール	プール サイズ (M)	予約済 サイズ (M)	最大 活動	PAGING オプション
1	1667.69	861.41	+++++	*FIXED
2	19489.37	12.61	344	*FIXED
3	6721.33	.00	704	*FIXED
4	281.59	.00	5	*FIXED



システム状況

```

CPU 使用 % . . . . . : 1.0
経過時間 . . . . . : 00:06:11
システム内のジョブ . . . . . : 688
永久アドレス % . . . . . : .009
一時アドレス % . . . . . : .012
    
```

変更データを入力して（可能な場合），実行キ

システム プール	プール サイズ (M)	予約済 サイズ (M)	最大 活動	PAGING オプション
1	1667.69	861.41	+++++	*FIXED
2	19489.37	12.61	344	*CALC
3	6721.33	.00	704	*FIXED
4	281.59	.00	5	*FIXED



ワークショップ、セッション、および資料は、IBMによって準備され、IBM独自の見解を反映したものです。それらは情報提供の目的のみで提供されており、いかなる読者に対しても法律的またはその他の指導や助言を意図したのではなく、またそのような結果を生むものでもありません。本資料に含まれている情報については、完全性と正確性を期するよう努力しましたが、「現状のまま」提供され、明示または暗示にかかわらずいかなる保証も伴わないものとします。本資料またはその他の資料の使用によって、あるいはその他の関連によって、いかなる損害が生じた場合も、IBMは責任を負わないものとします。本資料に含まれている内容は、IBMまたはそのサプライヤーやライセンス交付者からいかなる保証または表明を引きだすことを意図したもので、IBMソフトウェアの使用を規定する適用ライセンス契約の条項を変更することを意図したものでなく、またそのような結果を生むものでもありません。

本資料でIBM製品、プログラム、またはサービスに言及していても、IBMが営業活動を行っているすべての国でそれらが使用可能であることを暗示するものではありません。本資料で言及している製品リリース日付や製品機能は、市場機会またはその他の要因に基づいてIBM独自の決定権をもっていつでも変更できるものとし、いかなる方法においても将来の製品または機能が使用可能になると確約することを意図したものではありません。本資料に含まれている内容は、読者が開始する活動によって特定の販売、売上高の向上、またはその他の結果が生じると述べる、または暗示することを意図したもので、またそのような結果を生むものでもありません。パフォーマンスは、管理された環境において標準的なIBMベンチマークを使用した測定と予測に基づいています。ユーザーが経験する実際のスループットやパフォーマンスは、ユーザーのジョブ・ストリームにおけるマルチプログラミングの量、入出力構成、ストレージ構成、および処理されるワークロードなどの考慮事項を含む、数多くの要因に応じて変化します。したがって、個々のユーザーがここで述べられているものと同様の結果を得られると確約するものではありません。

記述されているすべてのお客様事例は、それらのお客様がどのようにIBM製品を使用したか、またそれらのお客様が達成した結果の実例として示されたものです。実際の環境コストおよびパフォーマンス特性は、お客様ごとに異なる場合があります。

IBM、IBM ロゴ、ibm.com、Db2、Power Systems、POWER6、POWER6+、POWER7、POWER7+、POWER8、POWER9は、世界の多くの国で登録されたInternational Business Machines Corporationの商標です。

他の製品名およびサービス名等は、それぞれIBMまたは各社の商標である場合があります。

現時点でのIBMの商標リストについては、www.ibm.com/legal/copytrade.shtmlをご覧ください。

インテル、Intel、Intelロゴ、Intel Inside、Intel Insideロゴ、Centrino、Intel Centrinoロゴ、Celeron、Xeon、Intel SpeedStep、Itanium、およびPentium はIntel Corporationまたは子会社の米国およびその他の国における商標または登録商標です。

Linuxは、Linus Torvaldsの米国およびその他の国における登録商標です。

Microsoft、Windows、Windows NT および WindowsロゴはMicrosoft Corporationの米国およびその他の国における商標です。

ITILはAXELOS Limitedの登録商標です。

UNIXはThe Open Groupの米国およびその他の国における登録商標です。

JavaおよびすべてのJava関連の商標およびロゴはOracleやその関連会社の米国およびその他の国における商標または登録商標です。