



UNIX®、Linux、および macOS でのインストールの詳細

Version 2024.1
2024-06-03

UNIX®, Linux、および macOS でのインストールの詳細
InterSystems IRIS Data Platform Version 2024.1 2024-06-03
Copyright © 2024 InterSystems Corporation
All rights reserved.

InterSystems®, HealthShare Care Community®, HealthShare Unified Care Record®, IntegratedML®, InterSystems Caché®, InterSystems Ensemble®, InterSystems HealthShare®, InterSystems IRIS®, および TrakCare は、InterSystems Corporation の登録商標です。HealthShare® CMS Solution Pack™ HealthShare® Health Connect Cloud™, InterSystems IRIS for Health™, InterSystems Supply Chain Orchestrator™, および InterSystems TotalView™ For Asset Management は、InterSystems Corporation の商標です。TrakCare は、オーストラリアおよび EU における登録商標です。

ここで使われている他の全てのブランドまたは製品名は、各社および各組織の商標または登録商標です。

このドキュメントは、インターシステムズ社(住所: One Memorial Drive, Cambridge, MA 02142)あるいはその子会社が所有する企業秘密および秘密情報を含んでおり、インターシステムズ社の製品を稼働および維持するためにのみ提供される。この発行物のいかなる部分も他の目的のために使用してはならない。また、インターシステムズ社の書面による事前の同意がない限り、本発行物を、いかなる形式、いかなる手段で、その全てまたは一部を、再発行、複製、開示、送付、検索可能なシステムへの保存、あるいは人またはコンピュータ言語への翻訳はしてはならない。

かかるプログラムと関連ドキュメントについて書かれているインターシステムズ社の標準ライセンス契約に記載されている範囲を除き、ここに記載された本ドキュメントとソフトウェアプログラムの複製、使用、廃棄は禁じられている。インターシステムズ社は、ソフトウェアライセンス契約に記載されている事項以外にかかるソフトウェアプログラムに関する説明と保証をするものではない。さらに、かかるソフトウェアに関する、あるいはかかるソフトウェアの使用から起こるいかなる損失、損害に対するインターシステムズ社の責任は、ソフトウェアライセンス契約にある事項に制限される。

前述は、そのコンピュータソフトウェアの使用およびそれによって起こるインターシステムズ社の責任の範囲、制限に関する一般的な概略である。完全な参照情報は、インターシステムズ社の標準ライセンス契約に記載され、そのコピーは要望によって入手することができる。

インターシステムズ社は、本ドキュメントにある誤りに対する責任を放棄する。また、インターシステムズ社は、独自の裁量にて事前通知なしに、本ドキュメントに記載された製品および実行に対する代替と変更を行う権利を有する。

インターシステムズ社の製品に関するサポートやご質問は、以下にお問い合わせください:

InterSystems Worldwide Response Center (WRC)
Tel: +1-617-621-0700
Tel: +44 (0) 844 854 2917
Email: support@InterSystems.com

目次

1 AIX® プラットフォームの問題	1
1.1 I/O ペーシング・パラメータ	1
1.2 ファイル・システムのマウント・オプション	1
1.3 メモリ管理パラメータ	2
1.4 AIX® の調整可能パラメータ	3
1.4.1 AIX® プロセス間通信の調整可能なパラメータ	3
1.4.2 maxuproc	4
1.4.3 必須の C/C++ ランタイム・ライブラリ	4
1.4.4 InterSystems IRIS 共有ライブラリ・サポートに対する共有ライブラリ環境変数	4
1.4.5 Raw Ethernet の使用法	5
2 Red Hat Linux プラットフォームの問題	7
2.1 共有メモリ制限	7
2.2 ロックインされたメモリ	8
2.3 多数の同時プロセスの調整	8
2.4 ダーティ・ページのクリーンアップ	9
2.5 Kerberos の使用	9
3 SUSE Linux プラットフォームの問題	11
3.1 I/O スケジューラ	11
3.2 共有メモリ制限	11
3.3 ロックインされたメモリ	12
3.4 Kerberos の使用	12
4 Ubuntu プラットフォームの問題	13
4.1 セマフォの削除の設定	13
5 Web サーバのインストール	15
6 所有者およびグループの決定	17
7 非 root ユーザとしてのインストール	19
7.1 InterSystems IRIS インストールで root を使用する理由	19
7.2 非 root インストールの制限事項	20
7.3 非 root インストールの相違点	20
8 自動インストール・パラメータ	23
9 自動インストール・パッケージ	29
10 InterSystems IRIS の開始	31
11 macOS での多数の同時プロセスの調整	33
12 開発環境のインストール	35
13 InterSystems IRIS ディストリビューションへの UNIX® インストール・パッケージの追加	37
13.1 チュートリアル	37
13.2 parameters.isc ファイルの内容	40

テーブル一覧

テーブル 8-1: 自動インストール・パラメータ	23
テーブル 13-1: InterSystems IRIS のインストール・パラメータ・ファイルの変数	41

1

AIX® プラットフォームの問題

一部の AIX® パラメータの既定の設定は、パフォーマンスに悪影響を及ぼす可能性があります。設定および推奨事項については以下を参照してください。

- ・ [I/O ペーシング・パラメータ](#)
- ・ [ファイル・システムのマウント・オプション](#)
- ・ [メモリ管理パラメータ](#)
- ・ [AIX® の調整可能パラメータ](#)
- ・ [必須の C/C++ ランタイム・ライブラリ](#)
- ・ [InterSystems IRIS 共有ライブラリ・サポートに対する共有ライブラリ環境変数](#)
- ・ [Raw Ethernet の使用法](#)

1.1 I/O ペーシング・パラメータ

AIX® は、InterSystems IRIS ライト・デーモンの動作を阻害する可能性のある I/O ペーシング・アルゴリズムを実装しています。AIX® 5.2 および AIX® 5.3 では、HACMP クラスタを使用すると I/O ペーシングが自動的に有効になります。一方、AIX® 6.1 以降では、すべてのシステムで I/O ペーシングが有効となり、既定の最高水準点は以前のバージョンよりも大きくなっています。

ライト・デーモンの速度が遅い場合やライト・デーモンが停止する場合は、この最高水準点の調整が必要なことがあります。詳細は、IBM AIX ドキュメントの [“ディスク入出力ペーシング”](#) を参照してください。

重要 IBM AIX® 6.1 以降では、最高水準点の調整は不要です。

システムへの影響について疑問点がある場合は、設定を変更する前に [インターシステムズのサポート窓口](#) または AIX® サプライヤまでお問い合わせください。これらの推奨事項は、JFS と Enhanced JFS (JFS2) 両方のファイル・システムに適用されます。

1.2 ファイル・システムのマウント・オプション

さまざまなマウント・オプションを使用することで作業負荷のパフォーマンスを向上させられる可能性があります。

注釈 ファイル・システムによるキャッシュが効果を発揮する、InterSystems IRIS ではない負荷 (例えば、オペレーティング・システム・レベルのバックアップやファイル・コピー) は、cio マウント・オプションにより速度が低下します。

ハード・シャットダウンやシステム・クラッシュの後の IRIS.WIJ ファイルを使用したリカバリ速度を向上させるには、IRIS.WIJ ファイルを扱うファイル・システムで、ファイル・システム・バッファリングを伴うマウント・オプション (例えば、rw) を使用することをお勧めします。

mount オプションの詳細は、IBM AIX ドキュメントの “[mount コマンド](#)” を参照してください。

1.3 メモリ管理パラメータ

ファイル・システムの数およびファイル・システムに対するアクティビティの量によって、JFS または JFS2 のメモリ構造が制限され、それらのメモリ構造を使用する I/O 処理に遅延が生じる可能性があります。

こうした状況を監視するには、vmstat -vs コマンドを実行し、2 分経過してから、もう一度 vmstat -vs コマンドを実行します。次のような結果が出力されます。

```
# vmstat -vs
1310720 memory pages
1217707 lruable pages
144217 free pages
1 memory pools
106158 pinned pages
80.0 maxpin percentage
20.0 minperm percentage
80.0 maxperm percentage
62.8 numperm percentage
764830 file pages
0.0 compressed percentage
0 compressed pages
32.1 numclient percentage
80.0 maxclient percentage
392036 client pages
0 remote pageouts scheduled
0 pending disk I/Os blocked with no pbuf
5060 paging space I/Os blocked with no psbuf
5512714 filesystem I/Os blocked with no fsbuf
194775 client filesystem I/Os blocked with no fsbuf
0 external pager filesystem I/Os blocked with no fsbuf
```

以下のパラメータが増加している場合は、InterSystems IRIS のパフォーマンスを向上するため、それらの値を大きくしてください。

- ・ pending disk I/Os blocked with no pbuf
- ・ paging space I/Os blocked with no psbuf
- ・ filesystem I/Os blocked with no fsbuf
- ・ client filesystem I/Os blocked with no fsbuf
- ・ external pager filesystem I/Os blocked with no fsbuf

これらのパラメータを既定値より大きくする場合は、以下の手順に従ってください。

1. 現在の値を 50% 大きくします。
2. vmstat 出力を確認します。
3. 2 分間隔で、vmstat を 2 回実行します。
4. フィールドがまだ増大している場合は、もう一度、パラメータ値を同じ割合だけ大きくして、上記の手順に従います。vmstat を 2 回実行したとき、フィールドが増大しなくなるまで繰り返してください。

重要 I/O パターンは時間の経過と共に変化する可能性があるので、現在値と再起動値の両方を変更し、vmstat 出力を定期的 (時間単位、日単位、または週単位) に確認します。

詳細は、IBM の以下の Web ページを参照してください。

- ・ vmstat によって報告される各フィールドの詳細は、IBM AIX ドキュメントの “[vmstat コマンド](#)” を参照してください。
- ・ これらのパラメータ値を大きくする方法については、IBM AIX ドキュメントの “[VMM ページ置換のチューニング](#)” を参照してください。
- ・ I/O 調整可能パラメータの管理の詳細は、IBM AIX ドキュメントの “[ioo コマンド](#)” を参照してください。

1.4 AIX® の調整可能パラメータ

1.4.1 AIX® プロセス間通信の調整可能なパラメータ

以下のテーブルは、IBM pSeries AIX® 5.2 オペレーティング・システムの調整可能なパラメータの一覧です。以下に一覧表示されているパラメータは調整の必要がありません。これらのパラメータは、必要に応じて、カーネルによって動的に調整されます。詳細は、AIX ドキュメントの “[プロセス間通信のチューナブル・パラメーター](#)” を参照してください。

パラメータ	目的	動的値
msgmax	最大メッセージ・サイズを指定します	最大値 4 MB
msgmnb	キューでの最大バイト数を指定します	最大値 4 MB
msgmni	メッセージ・キュー ID の最大数を指定します	最大値 4096
msgmnmx	キューごとの最大メッセージ数を指定します	最大値 524288
semaem	終了の調整最大値を指定します	最大値 16384
semmni	セマフォ ID の最大数を指定します	最大値 4096
semmsl	ID ごとの最大セマフォ数を指定します	最大値 65535
semopm	semop() 呼び出しごとの最大操作数を指定します	最大値 1024
semume	プロセスごとの最大 undo エントリ数を指定します	最大値 1024
semvmx	セマフォの最大値を指定します	最大値 32767
shmmax	共有メモリ・セグメントの最大サイズを指定します	32 ビット・プロセスでは最大値 256 MB、64 ビットでは最大値 0x80000000u
shmmni	共有メモリ・セグメントの最小サイズを指定します	最小値 1
shmmni	共有メモリ ID の最大数を指定します	最大値 4096

1.4.2 maxuproc

maxuproc は、root 以外の単独ユーザが開始可能なプロセスの最大数を指定する調整可能パラメータで、このサブセクションの説明に従って調整できます。

このパラメータの設定が低すぎると、プロセスの開始を試みるユーザがさらに増えた場合に、CSP ページの消失、バックグラウンド・タスクの失敗など、オペレーティング・システムのさまざまなコンポーネントで障害が起こる可能性があります。したがって、maxuproc パラメータは、root ユーザ以外（インタラクティブ・ユーザ、Web サーバ・プロセスなど、プロセスを開始可能なものすべて）による開始が見込まれるプロセスの最大数よりも大きく設定する必要があります。

注釈 この値を過度に高く設定しないでください。この値には不必要に新規プロセスを作成し続ける暴走アプリケーションからサーバを保護する目的があります。しかし、設定が低すぎると不明な問題の原因になります。

maxuproc を予想される最大プロセス数の倍にして、エラーを起こさないだけの余裕をとりつつ、暴走プロセスから保護することをお勧めします。例えば、システムに 1000 人のインタラクティブ・ユーザがいて、頻繁に 500 個のバックグラウンド・プロセスを実行する場合、この値は最低でも 3000 を選択するとよいでしょう。

maxuproc 値は、コマンド行もしくは smit/smitty 管理者用ユーティリティのいずれかより、以下のように双方共に root ユーザとして検証および変更ができます。

- ・ コマンド行より、現在の設定を表示します。

```
# lsattr -E -l sys0 -a maxuproc
```

次に、以下のように値を変更します。

```
# chdev -l sys0 -a maxuproc=NNNNNN
```

ここで、NNNNNN は新規の値です。

- ・ 管理者用ユーティリティ smit (または smitty) で、[システム環境]→[オペレーティングシステム特性の変更/表示]→[ユーザあたりの許容プロセス最大数]を選択します。

maxuproc の値を増やすと、変更は瞬時に有効となります。maxuproc の値を減らすと、変更は次のシステムの再起動まで有効になりません。どちらの場合も、変更はシステムを再起動しても存続します。

1.4.3 必須の C/C++ ランタイム・ライブラリ

InterSystems IRIS をインストールする前に、必須の C/C++ ランタイム・ライブラリが IBM AIX® システムにインストールされていることを確認する必要があります。

InterSystems IRIS for AIX は、IBM XL C/C++ for AIX 16.1.x および 17.1.x コンパイラを使用してコンパイルされます。InterSystems IRIS のインストール先のシステムに、既にインストールされているランタイムに対応するバージョンがない場合は、インストールする必要があります。最新の 16.1.x および 17.1.x ランタイムは、IBM Support Web サイトの ["Fix list for XL C/C++ Runtime for AIX 16.1"](#) からダウンロードできます。

1.4.4 InterSystems IRIS 共有ライブラリ・サポートに対する共有ライブラリ環境変数

InterSystems IRIS 共有ライブラリ・サポートには、インストールされたすべての C リンカを参照するバッチ・ファイルが含まれています。

標準 UNIX® C ライブラリまたは独自の C ライブラリを LIBPATH 環境変数で定義した環境が必要です。

このような定義がない環境では、標準 UNIX® C ライブラリのパスである /usr/lib と /lib を LIBPATH に追加します。

1.4.5 Raw Ethernet の使用法

Raw Ethernet を使用するには、IBM AIX® マシンに DLPI (データ・リンク・プロバイダ・インタフェース) パッケージがインストールされている必要があります。マシンに DLPI パッケージがない場合、IBM プロバイダから取得し、以下の手順に従って DLPI デバイスを作成してください。

1. root としてログインします。
2. `/etc/pse.conf` ファイルの PSE ドライバ・セクションで、DLPI ドライバを参照している 4 行をアンコメントします。
3. ファイルを保存します。
4. コンピュータを再起動します。

DLPI デバイスがインストールされていない場合、`%SYSTEM.INetInfo` クラスの `EthernetAddress()` メソッドは、イーサネット・デバイスの情報ではなく、NULL 文字列を返します。

2

Red Hat Linux プラットフォームの問題

このトピックには、以下の調整に関する情報が含まれます。

- ・ [共有メモリ制限](#)
- ・ [ロックインされたメモリ](#)
- ・ [多数の同時プロセスの調整](#)
- ・ [ダーティ・ページのクリーンアップ](#)
- ・ [Kerberos の使用](#)

2.1 共有メモリ制限

Linux プラットフォームでの、既定の共有メモリ制限 (shmmax) は、32 MB です。この値は InterSystems IRIS では小さすぎますが、**proc** ファイル・システムで変更することができます。その場合、再起動は必要ありません。新しいメモリ制限は、Red Hat Linux システムを再起動するまで維持されます。

例えば 128 MB で設定したい場合は、以下のコマンドを入力します。

```
$ echo 134217728 >/proc/sys/kernel/shmmax
```

上記のコマンドは、起動スクリプトに入力することができます。

また、`sysctl(8)` が利用できる場合はこれを利用して、パラメータを恒久的に設定することもできます。以下のような行を `/etc/sysctl.conf` に追加します。

```
kernel.shmmax = 134217728
```

このファイルは通常、スタートアップ時に処理されますが、`sysctl` は後で明示的に呼び出すこともできます。

重要 1 台のマシン上で複数の InterSystems IRIS インスタンスを実行する場合、`msgmni` パラメータの設定値が小さすぎる可能性があります。この値には、システムで同時に実行する InterSystems IRIS インスタンス数の 3 倍に相当する値を設定してください。

他のパラメータは、InterSystems IRIS アプリケーションに対して十分にサイズ調整されています。他のパラメータの値を表示するには、`/usr/src/linux/include/asm-xxx/shmparam.h` と `/usr/src/linux/include/linux/sem.h` の 2 つのファイルを調べてください。

詳細は、“Red Hat Enterprise Linux 4: リファレンス・ガイド” の “[proc ファイル・システム](#)” の章を参照してください。

2.2 ロックインされたメモリ

Linux プラットフォームでは、共有メモリがヒュージ・ページに割り当てられている場合、ヒュージ・ページはメモリ内で自動的にロックされます (それ以上の操作は必要ありません)。ヒュージ・ページの割り当ての詳細は、この章の “[Linux でのヒュージ・ページの構成](#)” を参照してください。

ヒュージ・ページを使用しない場合、共有メモリ・セグメントをメモリにロックしてページングを防ぐように InterSystems IRIS を構成できます。これについては、“構成パラメータ・ファイル・リファレンス” の “memlock” エントリの [LockSharedMemory](#) セクションで説明されています。

それ以外の場合は、メモリ内にロックされる可能性のある最大サイズを増やす必要があります。既定値は 32 KB です。現在の値を表示するには、ulimit コマンドを使用します。

例えば、現在の制限値をすべて表示するには、以下のようになります。

```
bash$ ulimit -a
core file size (blocks, -c) unlimited
data seg size (KBytes, -d) unlimited
file size (blocks, -f) unlimited
pending signals (-i) 1024
max locked memory (KBytes, -l) 32 <----- THIS ONE
max memory size (KBytes, -m) unlimited
open files (-n) 1024
pipe size (512 bytes, -p) 8
POSIX message queues (bytes, -q) 819200
stack size (KBytes, -s) 10240
cpu time (seconds, -t) unlimited
max user processes (-u) 49000
virtual memory (KBytes, -v) unlimited
file locks (-x) unlimited
```

max-locked memory のみ表示するには、-l オプションを使用します。

```
bash$ ulimit -l
32
```

権限を持っていれば、ulimit コマンドを使用して直接値を変更できますが、`/etc/security/limits.conf` ファイルで memlock パラメータを更新することをお勧めします。memlock の制限値が小さすぎる場合、Linux から ENOMEM () エラーが表示されますが、これでは原因が明らかになりません。実際にメモリは割り当てられています。エラーを起こしているのはロックなのです。

詳細は、“構成パラメータ・ファイル・リファレンス” の “[memlock](#)” を参照してください。

2.3 多数の同時プロセスの調整

多数のプロセスまたは Telnet ログインを必要とするシステムを実行する場合は、以下の調整を行う必要があります。

1. `/etc/xinetd.d/telnet` ファイルに、以下の行を追加します。

```
instances = unlimited
```

2. `/etc/xinetd.conf` ファイルで、インスタンスの設定を以下のように追加または変更します。

```
instances = unlimited
```

3. これらの変更が完了したら、以下のコマンドを使用して xinetd を再起動します。

```
# service xinetd restart
```

4. 既定の `pty` (擬似ターミナル接続) 制限は 4096 です。これでも足りない場合は、`/etc/sysctl.conf` ファイルに最大 `pty` 行を追加するか変更します。以下に例を示します。

```
kernel.pty.max=10000
```

2.4 ダーティ・ページのクリーンアップ

(例えば 8 GB 以上の) 大規模なメモリ・システムでは、フラット・ファイルを何度も書き込む場合 (例えば InterSystems IRIS のバックアップやファイル・コピーなど)、`proc/sys/vm/` に格納されている以下のパラメータを調整することにより、パフォーマンスを向上させることができます。

- ・ **dirty_background_ratio** – `pdflush` が書き込みを開始するまでに、ダーティ・ページで満たすことのできるアクティブの最大割合。このパラメータは 5 に設定することをお勧めします。
- ・ **dirty_ratio** – より多くの書き込みが許可される代わりに、タイム・スライス間のダーティ・バッファを書き込むプロセスが強制されるまでに、ダーティ・ページで満たすことのできる合計メモリの最大割合。このパラメータは 10 に設定することをお勧めします。

これらの変数は、`/etc/sysctl.conf` ファイルに以下を加えることで設定できます。

```
vm.dirty_background_ratio=5  
vm.dirty_ratio=10
```

これらの変更は、Linux `pdflush` デーモンに対して、大量の更新バーストでストレージが溢れる可能性のある大量の更新をキューに入れるより頻繁に、ダーティ・ページへの書き出しを強制するものです。

2.5 Kerberos の使用

Red Hat Linux プラットフォームで Kerberos を使用するには、`krb5-libs` パッケージに加えて `krb5-devel` パッケージもインストールする必要があります。`krb5-devel` をインストールすると、Kerberos を使用するために必要なシンボリック・リンクが設定されます。このパッケージは、開発環境だけでなく、実稼働環境にも必要です。これらのコンポーネントの詳細は、[Red Hat Network Web サイト](#)を参照してください。

3

SUSE Linux プラットフォームの問題

このトピックには、以下の調整に関する情報が含まれます。

- ・ [I/O スケジューラ](#)
- ・ [共有メモリ制限](#)
- ・ [ロックインされたメモリ](#)
- ・ [Kerberos の使用](#)

3.1 I/O スケジューラ

SUSE Linux の I/O スケジューラは、ストレージ・デバイスに送信された入出力要求の順序付けを行う役割を持ちます。SUSE Linux 15 では、スケジューラは既定で BFQ (Budget Fair Queueing) に設定される可能性があり、これは InterSystems IRIS でパフォーマンスの問題を引き起こすことが知られています。この設定を `NONE` に変更することをお勧めします。I/O スケジューラの変更の詳細は、SUSE のドキュメント ["Tuning I/O Performance"](#) を参照してください。

注釈 I/O スケジューラを `NONE` に設定することは、すべてのユース ケースで最適とは限りません。ユーザは変更を行った後、システムのアプリケーション・ワークロードをテストする必要があります。

3.2 共有メモリ制限

SUSE Linux プラットフォームでの、既定の共有メモリ制限 (`shhmax` と `shmall`) は InterSystems IRIS にとって小さすぎます。この制限は、再起動しなくても `proc` ファイル システム内で変更できます。

InterSystems IRIS はデータベース・バッファ、グローバル・バッファ、ルーチン・バッファ、およびライセンス使用に共有メモリを使用します。InterSystems IRIS 専用にマシンを使用している場合は、共有メモリをメモリの総容量の約半分に設定することをお勧めします。詳細は、"システム・リソースの計画と管理" の ["メモリの計画"](#)、および "システム管理ガイド" の ["メモリと開始設定"](#) と ["ライセンスの機能範囲と使用の決定"](#) を参照してください。

注釈 共有メモリ制限を変更する推奨事項は、SUSE Linux 64 ビット・システムには適用されません。

例えば 512 MB で設定したい場合は、以下のコマンドを入力します。

```
#sets shmall and shmmx shared memory
echo 536870912 >/proc/sys/kernel/shmall      #Sets shmall to 512 MB
echo 536870912 >/proc/sys/kernel/shmmx      #Sets shmmx to 512 MB
```

上記のコマンドは、スタートアップ時のスクリプトで入力できます。SUSE Linux 製品のドキュメントでは、`/etc/init.d/boot.local` スクリプト・ファイル内にコマンドを入れることが推奨されています。

また、`/etc/profile.local` というファイルを変更することで、システム・メモリ・ユーザ制限に対する設定を変更できます。以下のような行を追加します。

```
#sets user limits (ulimit) for system memory resources
ulimit -v 512000      #set virtual (swap) memory to 512 MB
ulimit -m 512000      #set physical memory to 512 MB
```

この同じファイルで、以下のような行を追加することで、PATH パラメータと CLASSPATH パラメータの値を永続的に変更できます。

```
#sets env values PATH and CLASSPATH
export PATH=$PATH:/usr/iris/bin:/path/to/j2sdk/bin:.
export CLASSPATH=
      $CLASSPATH:/iris/dev/java/lib/JDK18/intersystems-jdbc-3.0.0.jar.
```

重要 システム・アップグレード時に変更が失われるリスクを回避するために、`/etc/profile` ファイルは変更しないでください。

3.3 ロックインされたメモリ

Linux プラットフォームでは、共有メモリがヒュージ・ページに割り当てられている場合、それらはメモリ内で自動的にロックされます（それ以上の操作は必要ありません）。ヒュージ・ページの割り当ての詳細は、“[Linux でのヒュージ・ページの構成](#)”を参照してください。

ヒュージ・ページを使用しない場合、共有メモリ・セグメントをメモリにロックしてページングを防ぐように InterSystems IRIS を構成できます。これについては、“構成パラメータ・ファイル・リファレンス”の“memlock”エントリの [LockSharedMemory](#) セクションで説明されています。

それ以外の場合は、メモリ内にロックされる可能性のある最大サイズを増やす必要があります。手順については、この章の“Red Hat Linux プラットフォームの問題”にある“[ロックインされたメモリ](#)”のセクションを参照してください。

3.4 Kerberos の使用

SUSE Linux プラットフォームで Kerberos を使用するには、`krb5-libs` パッケージに加えて `krb5-devel` パッケージもインストールする必要があります。`krb5-devel` をインストールすると、Kerberos を使用するために必要なシンボリック・リンクが設定されます。このパッケージは、開発環境だけでなく、実稼働環境にも必要です。これらのコンポーネントの詳細は、[SUSE documentation](#) の Web サイトを参照してください。

4

Ubuntu プラットフォームの問題

このトピックには、以下の調整に関する情報が含まれます。

- ・ [セマフォの削除の設定](#)

4.1 セマフォの削除の設定

状況によっては、インスタンス所有者が SSH などを使用して Ubuntu ホストに接続すると、OS によってインスタンスのセマフォが削除される場合があります。これを防止するには、`/etc/systemd/logind.conf` ファイルを編集して、`RemoveIPC=yes` (既定値) を `RemoveIPC=no` に変更します。

Ubuntu の新規バージョンに更新すると、`RemoveIPC` が既定値の `yes` に戻ることがあります。Ubuntu を更新したら、不要なセマフォの削除を防止するために、必ず `RemoveIPC` を変更してください。

5

Web サーバのインストール

InterSystems IRIS をインストールする前に、サポート対象の Web サーバをインストールして、管理ポータルを含む Web アプリケーションにアクセスできるようにする必要があります。インストール手順は、選択する Web サーバに応じて異なります。詳細は、使用する Web サーバのドキュメントを参照してください。

多くの場合、InterSystems IRIS インストーラは、選択した Web サーバを使用する組み込みの Web アプリケーションを処理するよう、新しいまたはアップグレードされたインスタンスを自動的に構成できます。詳細は、“[Web サーバに自動的に接続する](#)”を参照してください。

Web サーバを手動でセットアップする場合は、IRIS をインストールした後にこれを実行できます。詳細は、“[Web サーバに手動で接続する](#)”を参照してください。

Web サーバを手動で構成する予定で（または Web サーバを構成するつもりはない）、自動インストールを実行している場合は、パラメータ `ISC_PACKAGE_WEB_CONFIGURE="N"` を設定するようにしてください。そうしないと、インストールで Web サーバの自動構成が試行され、Web サーバが検出されないと失敗します。

重要 Apache httpd Web サーバは、インストール・プロセスで自動的に構成できるため、この Web サーバの使用をお勧めします。インストール・プロセスの開始前に、これがインストールされ、実行されていることを確認してください。ほとんどの場合、Apache Web サーバを手動で構成する必要はありません。

6

所有者およびグループの決定

インストール・プロセスでは、以下のユーザとグループの情報の入力を求められます。

- ・ インスタンスの所有者
- ・ InterSystems IRIS スーパーサーバの実効ユーザとそのジョブ
- ・ InterSystems IRIS プロセスの実効グループ
- ・ インスタンスの開始および停止を許可されたグループ

これらのカテゴリの詳細は、“システム管理ガイド”の“UNIX®、Linux、および macOS での InterSystems IRIS 使用法”の章にある“[UNIX® ユーザとグループの識別](#)”のセクションを参照してください。

重要 InterSystems IRIS はインストール対象のファイルに対するユーザやグループなどの許可を設定する必要があります。これを行うために、InterSystems IRIS はインストール・プロセスの `umask` を `022` に設定します。インストールが完了するまで、`umask` を変更しないでください。

インスタンスの所有者として指定するユーザ・アカウントと、インスタンスの開始および停止を許可されたグループとして指定するグループの両方が、インストールを開始する前に存在している必要があります。これらのプロンプトの 1 つで指定したエントリが存在しない場合、そのプロンプトが再表示されるため、指定しようとしているユーザおよびグループがインストール開始前に存在することを確認してください。

使用しているオペレーティング・システムに `useradd` ユーティリティおよび `groupadd` ユーティリティ (または AIX® では `mkgroup` および `mkuser`) が含まれている場合は、指定したエントリが存在しなければ、InterSystems IRIS スーパーサーバ用の実効ユーザのアカウントおよび InterSystems IRIS プロセスの実効グループが作成されます。ただし、これらのユーティリティがなく、指定したエントリが存在しない場合、該当のプロンプトが再表示されます。システムにこれらのユーティリティがあるかどうか不明な場合は、インストールを開始する前に、指定しようとしているユーザおよびグループが存在するかどうかを確認します。

注釈 オペレーティング・システムで Network Information Services (NIS) やその他のネットワークベースのユーザまたはグループのデータベースを使用している場合は、`groupadd` ユーティリティと `useradd` (または AIX® では `mkgroup` および `mkuser`) ユーティリティによりローカル・ユーザおよびグループが作成され、ネットワーク・データベースに既に存在するエントリと競合する可能性があります。この問題を回避するには、ユーティリティに InterSystems IRIS 実効グループおよび実効ユーザを作成させるのではなく、インストールを開始する前に適切な管理ツールを使用して、これらをネットワーク・データベースに作成することをお勧めします。

プロセスの所有権を表示するために UNIX® のオペレーティング・システムで使用するツールでは、実際の所有権と実効所有権との対応が必ずしも表示できるとは限りません。InterSystems IRIS による許可の割り当ての詳細は、“システム管理ガイド”の“[UNIX® のユーザ、グループおよび権限](#)”の章を参照してください。

7

非 root ユーザとしてのインストール

プロダクション環境に InterSystems IRIS をインストールする場合、root 特権を使用することが推奨されます。root 特権なしで InterSystems IRIS インストールを実行することは可能ですが、このインストールにはいくつかの制限があります。以下のセクションで、これらの制限について、および標準の InterSystems IRIS インストールとの違いについて説明します。

- ・ [InterSystems IRIS インストールで root を使用する理由](#)
- ・ [非 root インストールの制限事項](#)
- ・ [非 root インストールの相違点](#)

重要 root 特権は、InterSystems IRIS をインストールするときのみ使用する必要があります。インストールが完了したら、すべてのユーザは非 root 特権を使用して InterSystems IRIS とやり取りする必要があります。

7.1 InterSystems IRIS インストールで root を使用する理由

通常、InterSystems IRIS は root を使用してインストールされ、非 root 特権を使用して操作されます。いくつかの機能には root アクセス権が必要ですが、ほとんどのプロセスはインストール時に指定したユーザまたはグループとして実行されます。これらのユーザとグループの目的、および root をどのように使用するかについては、“[UNIX のユーザ、グループおよび権限](#)”で説明します。

root 特権を使用する InterSystems IRIS プロセスには次のようなものがあります。

- ・ オペレーティング・システムでネットワーク設定を変更する有効なユーザ ID (UID) として root を使用する、[仮想 IP](#) プロセス。
- ・ 有効な UID としてインスタンス所有者を使用し、実際の UID として root を使用する制御プロセス。実際の UID は、開始時に[ラージ・ページ](#)を取得し、他の InterSystems IRIS プロセスと通信するために使用します。
- ・ 有効な UID として root を使用する開始実行可能ファイル。

InterSystems IRIS を root としてインストールすると、root 特権を持つユーザのみがファイル構造を変更または置換できるため、セキュリティも強化されます。

7.2 非 root インストールの制限事項

InterSystems IRIS の非 root インストールはサポートされてはいますが、この方法でインストールされたインスタンスでは、使用できない機能がいくつかあります。

- ・ ヒュージ・ページは使用できません。
- ・ インストールのマウント・ポイントは、nosuid を設定するとマウントできません。
- ・ 外部 Web サーバを使用するように Web ゲートウェイを構成することはできません。
- ・ [ミラー仮想 IP](#) は使用できません。

注釈 ネットワーク・ロード・バランサや Web ゲートウェイを使用するなど、ネットワーク・トラフィックをルーティングする別の方法については、“[フェイルオーバーまたは災害復旧後のアプリケーション接続のリダイレクト](#)”を参照してください。

- ・ インストール時に InterSystems IRIS を開始および停止できるインスタンス所有者およびグループを指定するオプションはありません (“[所有者およびグループの決定](#)”を参照)。
- ・ グループ・アクセスは存在しません。レジストリを含むすべてのインスタンス・ファイルはインストールするユーザの所有となり、このユーザによってのみ読み込み、書き込み、および実行が可能となります。

例えば、標準インスタンスが以下であるとしてします。

```
-rws--x--- 5 root develop 43282 Aug 28 07:52 irismgr
-r-x--s---x 1 <nonroot-user> irisusr 23058 Aug 28 07:52 irisuxsession
```

非 root インスタンスは以下のようになります。

```
-rwx----- 5 <installing-user> develop 43282 Aug 28 07:52 irismgr
-r-x----- 1 <installing-user> develop 23058 Aug 28 07:52 irisuxsession
```

レジストリは IRISYS で指定するディレクトリに配置され、そのレジストリに非 root インスタンスがあります(iris の実行可能ファイルも、このディレクトリにあります)。この非 root レジストリには、非 root のインスタンスのみが存在します。非 root レジストリから root によりインストールされたインスタンスへのアクセスはいずれも失敗します。反対に、非 root のインスタンスを root レジストリに定義することはできますが、所有者以外のユーザによるこのインスタンスへのアクセスはいずれも失敗します。

注釈 レジストリは、NFS ディレクトリではなく、インスタンスがインストールされたマシンのローカル・ディレクトリに配置することをお勧めします。標準の場所 `/usr/local/etc` がそのようなディレクトリとなります。

7.3 非 root インストールの相違点

上述した機能の制限のほかに、InterSystems IRIS の root インストールと非 root インストールにはいくつかの明確な違いがあります。

- ・ IRISYS 環境変数は、インストールするユーザによって書き込み可能な既存ディレクトリとして定義する必要があり、インストール中およびすべてのインスタンス操作中に存在する必要があります。
- ・ [ISCAgent](#) は、IRISYS で指定されたディレクトリにインストールされます。

注釈 非 root インスタンスの ISCAgent の開始の詳細は、“高可用性ガイド”の“ミラーリング”の章にある“UNIX®/Linux および macOS システムでの非 root インスタンス用 ISCAgent の開始”を参照してください。

- ・ インストールしているユーザのアカウントのみが、InterSystems IRIS インスタンスにアクセスして操作できます。
- ・ InterSystems IRIS の実行可能ファイルおよびプロセスはすべて、インストールするユーザとして実行されます。

8

自動インストール・パラメータ

以下のテーブルでは、自動インストールの `irisinstall_silent` スクリプトで使用するパラメータについて説明しています。

テーブル 8-1: 自動インストール・パラメータ

パラメータ	説明
<code>ISC_CPF_MERGE_FILE="<location>"</code>	構成のマージを実行する際に、構成マージ・ファイルの場所を指定します。 詳細は、“ 構成マージを使用した InterSystems IRIS の自動構成 ”を参照してください。
<code>ISC_PACKAGE_INSTANCENAME="<instancename>"</code> (必須)	インストールまたはアップグレードするインスタンスの名前を指定します。インスタンスが存在しない場合は、新規インストールとなります。インスタンスが存在する場合は、アップグレードとなります。以下に例を示します。 <code>ISC_PACKAGE_INSTANCENAME="MyIris"</code> 注釈 新規インストールの場合は <code>ISC_PACKAGE_INSTALLDIR</code> パラメータが必要です。
<code>ISC_PACKAGE_INSTALLDIR="<installdir>"</code> (新規インスタンスで必要)	新しいインスタンスをインストールするインストール・ディレクトリを指定します。以下に例を示します。 <code>ISC_PACKAGE_INSTALLDIR="/opt/MyIris"</code> 指定されたディレクトリが存在しない場合、インストールによりそのディレクトリの作成が試行されます。インストールをアップグレードする場合、このパラメータは無視されます。 注釈 インストール・ディレクトリの選択の詳細は、このドキュメントの“ インストール・ディレクトリ ”のセクションを参照してください。

パラメータ	説明
ISC_PACKAGE_INSTALL_INTEGRATEDML="Y" "N"	<p>InterSystems IntegratedML をインストールするかどうかを指定します。</p> <p>IntegratedML の詳細は、“IntegratedML の使用”を参照してください。</p> <p>注釈 既定では、Y に設定されます。</p>
ISC_PACKAGE_UNICODE="Y" "N" (オプション)	<p>UNICODE インストールかどうかを指定します。有効値は、Y または N です。詳細は、“文字幅設定”を参照してください。</p>
ISC_PACKAGE_INITIAL_SECURITY="Minimal" "Normal" "LockedDown" (オプション)	<p>インストールの初期セキュリティ設定を指定します。有効値は、“Minimal”、“Normal”、または“LockedDown”です。</p> <p>注釈 “Minimal” は、InterSystems IRIS インストールにのみ使用できます。</p> <p>“Normal”、または“LockedDown”に設定する場合、ISC_PACKAGE_USER_PASSWORD が必要です。</p>
ISC_PACKAGE_MGRUSER=" <user> " (オプション)	<p>インストールの所有者のログイン名を指定します。以下に例を示します。</p> <p>ISC_PACKAGE_MGRUSER="jcsmith"</p> <p>注釈 既定では、インスタンスをインストールするユーザのユーザ名に設定されます。</p> <p>セキュリティ・レベルが“Minimal”の場合、このパラメータは無視され、ISC_PACKAGE_MGRUSER は“root”に設定されます。</p>
ISC_PACKAGE_MGRGROUP=" <group> " (オプション)	<p>インスタンスの開始および停止を許可されるグループを指定します。以下に例を示します。</p> <p>ISC_PACKAGE_INITIAL_MGRGROUP="irisusr"</p> <p>注釈 既定では、インスタンスをインストールするユーザのグループが設定されます。</p>

パラメータ	説明
ISC_PACKAGE_USER_PASSWORD=" <i><password></i> " (安全なインスタンスのインストールの場合、必須)	<p>通常のセキュリティまたはロック・ダウン・セキュリティでインスタンスに必要なパスワードを指定します。</p> <p>注釈 セキュリティ・レベルが "Minimal" の場合、このパラメータは無視されます。</p> <p>セキュリティ・レベルが "Normal"、または "LockedDown" で、このパラメータを指定しない場合、インストールは失敗し、エラーがスローされます。</p>
ISC_PACKAGE_CSPSYSTEM_PASSWORD=" <i><password></i> "	<p>CSPSystem ユーザのパスワードを指定します。</p> <p>注釈 セキュリティ・レベルが "Minimal" の場合、このパラメータは無視されます。</p> <p>セキュリティ・レベルが "Normal" または "LockedDown" で、このパラメータを指定しない場合、ISC_PACKAGE_USER_PASSWORD の値が使用されます。</p>
ISC_PACKAGE_IRISUSER=" <i><user></i> " (オプション)	<p>InterSystems IRIS スーパーサーバの実効ユーザを指定します。</p> <p>注釈 既定では、irisusr に設定されます。</p> <p>セキュリティ・レベルが "Minimal" の場合、このパラメータは無視され、既定の設定となります。</p>
ISC_PACKAGE_IRISGROUP=" <i><group></i> " (オプション)	<p>InterSystems IRIS プロセスの実効ユーザを指定します。</p> <p>注釈 既定では、irisusr に設定されます。</p> <p>セキュリティ・レベルが "Minimal" の場合、このパラメータは無視され、既定の設定となります。</p>

パラメータ	説明
ISC_PACKAGE_CLIENT_COMPONENTS=" <i><component1></i> <i><component2></i> ..." (オプション)	<p>client_components パッケージからインストールされるクライアント・バインディングを指定します (“自動インストール・パッケージ” を参照)。</p> <p>注釈 既定では、すべてのクライアント・バインディングがインストールされます。指定するコンポーネント (バインディング) はスペースで区切る必要があります。使用可能なコンポーネントの一覧は <code>package/client_components/manifest.isc</code> にあります。インストールでは指定したコンポーネントを検証して、個々のシステムに存在しない、またはサポートされていないコンポーネントを削除します。</p>
ISC_PACKAGE_WEB_CONFIGURE="Y" "N" (オプション)	<p>外部 Web サーバ用に Web ゲートウェイを構成するかどうかを指定します。</p> <p>注釈 既定では、これは Y に設定されます (外部 Web サーバ用に自動的にゲートウェイを構成します)。Web サーバを自動的に構成しない場合は、このオプションを N に設定してください。</p>
ISC_PACKAGE_WEB_SERVERTYPE="Apache" "None" (オプション)	<p>Web ゲートウェイで使用する既存の Web サーバのタイプ。例： ISC_PACKAGE_WEB_SERVERTYPE="Apache"</p> <p>注釈 既定では、これは Apache に設定されます。</p>
ISC_PACKAGE_WEB_APACHE_VERSION=2.4 (オプション、ISC_PACKAGE_WEB_SERVERTYPE="Apache" と共に使用)	<p>Apache Web サーバのバージョン。</p> <p>注釈 既定では、バージョンは自動検出されます。</p>
ISC_PACKAGE_WEB_APACHE_USER=" <i><username></i> " (オプション、ISC_PACKAGE_WEB_SERVERTYPE="Apache" と共に使用)	<p>Apache Web サーバのユーザ名。</p> <p>注釈 既定では、ユーザ名は自動検出されます。</p>

パラメータ	説明
ISC_PACKAGE_WEB_APACHE_CONF=" <i><path_to_httpd.conf></i> " (オプション、ISC_PACKAGE_WEB_SERVERTYPE="Apache"と共に使用)	<p>Apache Web サーバの構成ファイルの場所 (例 : /etc/httpd/conf/httpd.conf)</p> <p>注釈 既定では、インストールにより、複数ある標準の場所のいずれかのファイルの自動検出が試行されます。 ISC_PACKAGE_WEB_SERVERTYPE="Apache" および httpd.conf の場所が特定されない場合、インストールはエラーで終了します。</p>
ISC_PACKAGE_WEB_GATEWAY_DIR=" <i><web_gateway_directory></i> " (オプション、Web ゲートウェイの新規インストールでのみ)	<p>Web ゲートウェイ・ファイルが保存されるディレクトリ。</p> <p>注釈 既定のディレクトリは /opt/webgateway_ です。</p>
ISC_PACKAGE_STARTIRIS="Y" "N" (オプション)	<p>インストール後に、インストールした InterSystems IRIS インスタンスを開始するかどうかを指定します。</p> <p>注釈 既定では、Y に設定されていて、インスタンスを開始します。</p>
ISC_INSTALLER_MANIFEST=" <i><location></i> "	<p>インストール・マニフェストでインストールする際、エクスポートされたマニフェスト・クラスを指定します。</p> <p>注釈 詳細は、“マニフェストの使用”を参照してください。</p>
ISC_INSTALLER_PARAMETERS=" <i><var>=<value>, <var>=<value> ...</i> "	<p>インストール・マニフェストでインストールする際、変数の名前と値の組み合わせを指定します。</p>
ISC_INSTALLER_LOGFILE=" <i><filename></i> "	<p>インストール・マニフェストでインストールする際、ログ・ファイル名を指定します。</p>
ISC_INSTALLER_LOGLEVEL=" <i><level></i> "	<p>インストール・マニフェストでインストールする際、-1 (“なし”) から 3 (“詳細”) までの数値でログ・レベルを指定します。</p>
ISC_PACKAGE_SUPERSERVER_PORT=" <i><port_number></i> " (オプション)	<p>インストールされるインスタンスで使用するスーパーサーバ・ポートを指定します。</p> <p>注釈 既定では、このポートは 1972 に設定されます (可能な場合)。それ以外の場合、ポートは 51773、またはこれ以降の使用可能な最初のポート番号に設定されます。</p>

パラメータ	説明
ISC_PACKAGE_USER_DATABASE=Y N	オプションのユーザ・データベースをインストールするかどうかを指定します。既定では、Y に設定されます。

9

自動インストール・パッケージ

各コンポーネントのインストール・スクリプトは、`irisinstall_silent` スクリプトを含んでいるディレクトリの配下の `packages` ディレクトリに含まれています。各パッケージは、それぞれのディレクトリ内に配置され、各パッケージのディレクトリには、そのディレクトリ内パッケージの前提条件となるパッケージが定義された `manifest.isc` ファイルが含まれています。

`standard_install` パッケージは、すべてのパッケージがインストールされるサーバ・インストールの開始ポイントです。カスタム・パッケージを定義するには、以下のように `standard_install` パッケージの `manifest.isc` ファイルをテンプレートとして使用します。

1. `standard_install` ディレクトリを新しいディレクトリにコピーします。

例えば、`custom_install` という名前のディレクトリにコピーします。新しいディレクトリにコピーされた `manifest.isc` ファイルの初期の状態は次のようになります。

```
#This is the target for a standard (non-client-only) install
package: standard_install
prerequisite: install_mode
prerequisite: database_server
prerequisite: databases
prerequisite: gadget
prerequisite: fop
prerequisite: renderserver
prerequisite: printserver
prerequisite: exceleporter
prerequisite: callin_components
prerequisite: client_components
prerequisite: addenda
prerequisite: install_confirmation
prerequisite: copyright
```

2. 新しいディレクトリで、以下のように `manifest.isc` ファイルを変更します。

- ・ パッケージ・キーをディレクトリ名の値に設定します (必須)。
- ・ カスタム・インストールの前提条件ファイルを追加または削除します。

例えば、以下の `manifest.isc` ファイルでは、パッケージ・キーの値がディレクトリ名 (`custom_install`) に一致するように変更されています。

```
#This is the target for a custom (non-client-only) install
package: custom_install
prerequisite: install_mode
prerequisite: database_server
prerequisite: gadget
prerequisite: fop
prerequisite: renderserver
prerequisite: printserver
prerequisite: exceleporter
prerequisite: callin_components
prerequisite: client_components
prerequisite: addenda
prerequisite: install_confirmation
prerequisite: copyright
```

次に、自動インストールを実行するときに新しいカスタム・パッケージを指定できます。例えば、`sudo ISC_PACKAGE_INSTANCENAME="MyIris" ./irisinstall_silent custom_install` のように指定します。

注釈 独自の UNIX® インストール・パッケージを作成して、それらを InterSystems IRIS ディストリビューションに追加する方法の詳細は、付録 ["InterSystems IRIS ディストリビューションへの UNIX® インストール・パッケージの追加"](#) を参照してください。

10

InterSystems IRIS の開始

ほとんどのインストールでは、インストールが完了すると InterSystems IRIS は自動的に起動します。InterSystems IRIS を起動する必要がある場合は、最初にオペレーティング・システムにログインし、iris コマンドを使用して InterSystems IRIS を起動します。

```
iris start <instname>
```

instname は、インストール時に指定したインスタンス名です。Red Hat Linux では、起動コマンドは以下のとおりです。

```
service <instname> start
```

InterSystems IRIS の起動および停止には、iris コマンドを使用します。この詳細は、“システム管理ガイド”の“[InterSystems IRIS インスタンスの制御](#)”のセクションを参照してください。

注釈 mgr サブディレクトリへのパスを構成するすべての要素に対する許可を、(少なくとも) irisusr グループが読み取りアクセスできるように設定する必要があります。このような許可を設定しないと、インスタンスは完全には起動せず、「manager_subdirectory (errno 2)」というメッセージが messages.log に記録されます。

InterSystems IRIS が起動したら、iris terminal コマンドを使用して InterSystems IRIS セッションを開始します。“システム管理ガイド”の“InterSystems IRIS の複数インスタンスの使用法”の章にある“[InterSystems IRIS インスタンスの接続](#)”を参照してください。

```
iris terminal <instname> [parameters]
```

instname は、インストール時に指定したインスタンス名です。

詳細は、“システム管理ガイド”の“[UNIX®、Linux、および macOS での InterSystems IRIS 使用法](#)”の章を参照してください。

11

macOS での多数の同時プロセスの調整

多数のプロセスまたは Telnet ログインを必要とするシステムを実行する場合は、以下の調整を行う必要があります。

1. リモート接続 – pty (擬似ターミナル) 接続の数はシステム全体で 128 までに制限されています。アプリケーションが ユーザ・アクセスを Telnet などの pty 使用接続に依存している場合は、この点に注意してください。
2. プロセス数 – 多数のプロセスの実行が必要な場合は、pty 制限に該当しなくても、プロセス数自体に制限があります。
 - ・ システム全体でのプロセス制限 – kern.maxproc および kern.maxprocperuid パラメータの既定値はそれぞれ 532 と 100 に設定されています。これらのパラメータは、次のコマンドを使用して変更できます。

```
administrator$ sudo sysctl -w kern.maxproc=2500
kern.maxproc: 2065 -> 2500
administrator$ sudo sysctl -w kern.maxprocperuid=2500
kern.maxprocperuid: 2000 -> 2500

administrator$ sysctl -a | grep maxproc
kern.maxproc = 2500
kern.maxprocperuid = 2500
```

ただし、2500 が変更不能な絶対上限であることに注意してください。

12

開発環境のインストール

InterSystems IRIS には UNIX® 上の開発環境は含まれません。したがって、Windows クライアント上で実行する必要がある [スタジオ](#) か、[Visual Studio Code](#) の [InterSystems ObjectScript 拡張機能](#) を選択する必要があります。

Web サーバで 80 以外のポートを使用している場合は、IDE と接続できるようにするため、インストール後に、CPF の [WebServerPort](#) および [WebServerURLPrefix](#) パラメータを Web サーバのポート番号に変更する必要があります。

Windows クライアントをインストールするには、“[Windows でのクライアント・インストール](#)” の指示に従います。Windows クライアントがインストールされたら、以下のタスクを実行して、スタジオをインスタンスに接続します。

1. Windows クライアントで、InterSystems IRIS [ランチャー](#) を開きます。
2. [\[優先接続サーバ\]](#) を指して [\[追加/編集\]](#) をクリックし、インストールした InterSystems IRIS インスタンスへの [リモート・サーバ接続を追加](#) します。この接続に適切なポート番号を指定するように注意してください。
3. [\[リモート システム アクセス\]](#)、[\[スタジオ\]](#) の順にポイントし、前の手順で入力した該当する接続名をクリックします。

13

InterSystems IRIS ディストリビューションへの UNIX® インストール・パッケージの追加

この付録では、既存の InterSystems IRIS® Data Platform ディストリビューションに新しい UNIX® インストール・パッケージを追加する方法を説明します。この追加方法はチュートリアル形式で示し、InterSystems IRIS インスタンス・ディレクトリに追加ファイルをコピーする単純なパッケージを作成する場合について解説します。

注釈 インストール・パッケージは UNIX® シェル・スクリプトを使用して実装するため、これよりはるかに複雑な操作を実行するパッケージも作成できます。

13.1 チュートリアル

Foo9000 という名前のイメージング・デバイスに接続するインターシステムズのコールアウト共有ライブラリ (“コールアウト・ゲートウェイの使用法” の “InterSystems コールアウト・ライブラリの作成” の章を参照) を記述したとします。このライブラリを **libfoo9000.so** としてコンパイルします。このライブラリが InterSystems IRIS と共にインストールされるようにしたいとします。また、インストール時に、このライブラリの接続先デバイス (Foo9000) のネットワーク・サーバ名を入力するようにユーザに求めるプロンプトが表示されるようにする必要もあります。この情報は InterSystems IRIS インスタンスの `install-dir¥mgr` ディレクトリ内の構成ファイルに保存されます。

既存の InterSystems IRIS キットから開始します。

```
~/kit:>ls
irisinstall  cplatname docs  lgpl.txt NOTICE
copyright.pdf dist  kitlist LICENSE package
```

また、コンパイルしたライブラリ (**libfoo9000.so**) を使用します。

```
~/lib:>ls
libfoo9000.so
```

まず、ライブラリを保存するキット内の場所を選択し、その場所にライブラリをコピーする必要があります。規約により、プラットフォーム別のライブラリは `dist/package/platform` ディレクトリ (`~/kit/dist/foo9000/lnxsusex64` など) に保存します。

```
~/kit:>cd dist
~/kit/dist:>mkdir foo9000
~/kit/dist:>cd foo9000
~/kit/dist/foo9000:>mkdir lnxsusex64
~/kit/dist/foo9000:>cd lnxsusex64
~/kit/dist/foo9000/lnxsusex64:>cp ~/lib/libfoo9000.so .
```

次に、インストール・パッケージのディレクトリを作成し、パッケージについて記述する `manifest.isc` ファイルをこのディレクトリに追加します。最も単純な形式の `manifest.isc` ファイルでは、パッケージ名のみを記述します。この名前はパッケージ・ディレクトリの名前 (`foo9000`) と一致させる必要があります。

```
~/kit/package:>mkdir foo9000
~/kit/package:>cd foo9000
~/kit/package/foo9000:>emacs manifest.isc
package: foo9000
```

パッケージ内が空だと処理は何も実行されませんが、このチュートリアルでは、次の処理を実行する必要があります。

1. Foo9000 をホストするサーバ名の入力をユーザに求めるプロンプトを表示する。
2. `install-dir¥mgr` ディレクトリ内の構成ファイルにこの情報を保存する。
3. インスタンス・バイナリ・ディレクトリにライブラリ (`libfoo9000.so`) をコピーする。

パッケージ・インストーラは複数のフェーズで処理を実行します。そのうち重要なフェーズは次のフェーズです。

- ・ “parameters” フェーズ
- ・ “install” フェーズ

注釈 パッケージには、フェーズごとに、それぞれのフェーズの名前を持つ Bourne シェル・スクリプトを格納することができます。パッケージ・インストーラは、フェーズ内の適切な時点で、各パッケージのスクリプトを実行します。パッケージのスクリプトで所定のフェーズの実行が正常に終了すると、最終コマンドによって明示的または暗黙的にエラー・コード 0 が返されます。正常に終了しなかった場合は、0 以外のエラー・コードが返されます。

“parameters” フェーズでは、パッケージのインストールに必要な情報を (通常ユーザに入力を求めることによって) 収集します。このフェーズでは、システムへの永続的な変更は行いません。通常、“parameters” フェーズ後に、ユーザがインストールの取り消しを必要に応じて選択できるようにします。取り消しが選択された場合、システムがインストール・プロセスによる影響を何も受けていない状態になっていなければなりません。

“install” フェーズでは、システムを変更します。無人インストールや自動インストールが実行される場合があるため、install フェーズ中はユーザに情報の入力を求めないようにします。

パッケージによっては、ユーザからの情報の入力が不要、つまり “parameters” スクリプトが不要な場合があります。特定のフェーズのスクリプトがパッケージに含まれていない場合、そのフェーズ中、そのパッケージの処理は何も実行されません。

まず、`foo9000` パッケージ用の “parameters” スクリプトとして、以下のようなスクリプトを作成したとします。

```
~/kit/package/foo9000:>emacs parameters
#!/bin/sh
echo "Please enter host name of the Foo9000 imaging server: "
read host
echo "Host $host entered."
```

このスクリプトを実行すると、以下のように、ホスト名の入力を求めるプロンプトが実際に表示されます。この情報は `host` 変数に記録されます。

```
~/kit/package/foo9000:>sh parameters
Please enter host name of the Foo9000 imaging server:
host1
Host host1 entered.
```

しかし、`host` 値を取得した後、この値をどう処理したらいいでしょうか。スクリプトの実行が終了すると、この情報は失われるため、“install” フェーズでこの情報を構成ファイルに書き込む必要が生じたときに利用できません。

注釈 “parameters” フェーズではユーザのシステムに何の影響も及ぼさないようにするため、この時点ではまだ構成ファイルを作成しません。

パッケージ・インストーラは、複数のフェーズや複数のパッケージで情報を共有できるようにする便利な関数のペア Import と Export を備えています。通常のシェル・スクリプトのメカニズムによって、これらの関数を `parameters.include` ファイルに組み込んで使用できます。

```
#!/bin/sh
. parameters.include
echo "Please enter host name of the Foo9000 imaging server: "
read host
echo "Host $host entered."
Export foo9000.host $host
```

Export 関数は、エクスポートする (通常、スクリプトのローカル変数から) パラメータ変数の名前とその値を取ります。Import 関数ではこの逆になります。最初の引数には、あらかじめエクスポートされている値のインポート先のローカル変数を指定し、2 番目の引数には、エクスポート先のパラメータ変数の名前を指定します。

注釈 規約により、パラメータ関数の名前は、`package name.local variable name` 形式の名前 (`foo9000.host` など) にします。

これで、インストール・プロセスに必要なすべての Foo9000 情報が “parameters” スクリプトで収集されるようになったため、次は “install” スクリプトを作成します。

```
~/kit/package/foo9000:>emacs install
#!/bin/sh
. parameters.include
Import host foo9000.host
echo host=$host > ???/mgr/foo9000.cfg
cp ???/dist/foo9000/???/libfoo9000.so ???/bin
```

以下の詳細を指定する必要があります (上記スクリプトの ??? の部分)。

- ・ インストールを作成するインスタンス・ディレクトリの場所。
- ・ インストール元のキットの場所。
- ・ インストールするプラットフォームの種類。

これらの質問は “parameters” スクリプトに組み込むこともできますが、ユーザはこれらの情報をインストールの初期段階で既に入力しているため、混乱を招く可能性があります。代わりに、必要な情報を提供する他のパッケージからパラメータ変数をインポートします。これが可能なのは、`irisinstall`、`irisinstall_client` または `irisinstall_silent` スクリプトを使用したインストールが成功すると (“[InterSystems IRIS のインストール](#)” を参照)、これらの変数と値を含んだ `parameters.isc` ファイルがインストール・ディレクトリに作成されるためです。`parameters.isc` ファイルの変数の一覧は、この付録の最後にある “[InterSystems IRIS インストール・パラメータ・ファイルの変数](#)” のテーブルに掲載しています。

注釈 セキュリティ上の理由により、`parameters.isc` ファイルには root ユーザのみがアクセスできます。

特定のパッケージのパラメータ変数を使用するためには、現在のパッケージ (`foo9000`) が別のパッケージに依存すること、したがって各フェーズで別のパッケージの後に処理する必要があることを、パッケージ・インストーラに認識させる必要があります。それには、以下に示すように、現在のパッケージの `manifest.isc` ファイルに “prerequisite” の各値を追加します。

```
~/kit/package/foo9000:>emacs manifest.isc
package: foo9000
prerequisite: server_location
prerequisite: legacy_dist
prerequisite: platform_selection
```

これで、指定したパッケージからパラメータ変数をインポートし、現在のパッケージのインストール・スクリプトの実行に使用できるようになります。

```
~/kit/package/foo9000:>emacs install
#!/bin/sh
. parameters.include
Import host foo9000.host
Import tgt_dir "server_location.target_dir"
Import src_dir "legacy_dist.source_dir"
Import platform_family "platform_selection.platform_family"
echo host=$host > $tgt_dir/mgr/foo9000.cfg
cp $src_dir/dist/foo9000/$platform_family/libfoo9000.so $tgt_dir/bin
```

パッケージ **foo9000** はほぼ完成しました。最後の作業として、関連する既存のパッケージの prerequisite リストにこのパッケージを追加します。それにより、パッケージ・インストーラは、その関連パッケージのインストールを完了するために、今回作成したパッケージを処理するようになります。この場合、InterSystems IRIS サーバのインストール時に必ずこのライブラリのインストールと構成が行われるようにする必要があるため、「database_server」パッケージの **manifest.isc** ファイルで、このパッケージの prerequisite リストに新しいパッケージを追加します。

```
~/kit/package/database_server:>emacs manifest.isc
package: database_server
prerequisite: legacy_dist
prerequisite: platform_selection
prerequisite: server
prerequisite: server_location
prerequisite: upgrade
prerequisite: available_disk_space
prerequisite: posix_tools
...
prerequisite: isql
prerequisite: zlib
prerequisite: udp
prerequisite: bi
prerequisite: foo9000
```

上のリストから明らかなように、サーバ・インストールの作成には多くのパッケージが必要ですが、これで、**irisinstall** を実行すると、パッケージ **foo9000** の構成とインストールが実行されるようになりました。

```
~/kit:>sudo ./irisinstall
Your system type is 'SuSE Linux Enterprise Server 10 (x64)'.
Currently defined instances:
IRIS instance 'INSTANCE1'
directory: /home/testUser/INSTANCE1
versionid: 2018.1.0.508.0
conf file: iris.cpf (SuperServer port = 1972)
status: crashed, last used Sat Sep 22 08:37:32 2018
Enter instance name: INSTANCEPACK1
Do you want to create IRIS instance 'INSTANCEPACK1' ? Y
...
Please enter host name of the Foo9000 imaging server:
host1
Host host1 entered.
...
Do you want to proceed with the installation ? Y
...
Installation completed successfully
~/INSTANCEPACK1/bin:>ls libfoo*
libfoo9000.so
~/INSTANCEPACK1/mgr:>cat foo9000.cfg
host=host1
```

13.2 parameters.isc ファイルの内容

以下の表は、**parameters.isc** ファイルに記載されている変数とその説明、およびサンプル値、または有効な値の一覧です。

テーブル 13-1: InterSystems IRIS のインストール・パラメータ・ファイルの変数

変数名	説明 ()
dist.source_dir	インストール媒体のソース・ディレクトリ。 /iriskit
legacy_dist.source_dir	従来の目的では、インストール媒体のソース・ディレクトリ。 /iriskit
product_info.version	インターシステムズ社製品のバージョン番号。 2018.1.0.100.0
product_info.name	インターシステムズ社製品の名前。 InterSystems IRIS
platform_selection.platform	インストール先プラットフォームに対するインターシステムズ社の略称。 lnxrhx64
platform_selection.platform_family	インストール先プラットフォーム・ファミリーに対するインターシステムズ社の略称。 lnxrhx64
platform_selection.endianness	プラットフォームのエンディアン・バイト・オーダー。 (big/little)
platform_selection.os	プラットフォームのオペレーティング・システム。uname コマンドの値。 Linux
posix_tools.user_add	Portable Operating System Interface (POSIX) 準拠のユーザ追加ツール。 /usr/sbin/useradd
posix_tools.group_add	POSIX 準拠のグループ追加ツール。 /usr/sbin/groupadd
posix_tools.grep	POSIX 準拠の grep ユーティリティ。 grep
posix_tools.id	POSIX 準拠の id ユーティリティ。 id
posix_tools.ps_opt	プロセスのリスト表示ですべて展開するオプション。 -ef
posix_tools.gzip	Gnu 準拠の zip ユーティリティ。 gzip

変数名	説明 ()
posix_tools.shared_ext	共有ライブラリ・ファイルの拡張子。 so
posix_tools.symbolic_copy	POSIX 準拠のシンボル・コピー・コマンド。 cp -Rfp
posix_tools.tr	POSIX 準拠の変換ユーティリティ。 tr
posix_tools.shared_ext1	共有ライブラリ・ファイルの代替の拡張子。 so
posix_tools.permission	選択されたファイルに適用される POSIX 準拠の特権。 755
posix_tools.dir_permission	選択されたディレクトリに適用される POSIX 準拠の特権。 775
server_location.target_dir	サーバ・インストールのターゲット・ディレクトリ。 /test/IRIS
server_location.is_server_install	これがサーバ・インストールであるかどうかを示す。 (N/Y)
server_location.is_nonroot_install	これが非 root インストールであるかどうかを示す。 (N/Y)
server_location.instance_name	インスタンス名。 IRIS
server_location.is_new_install	これが新規インストールであるかどうかを示す。 (N=アップグレード/Y=新規)
server_location.is_new_directory	新規ディレクトリを作成するかどうかを示す。 (N/Y)
server_location.registry_dir	InterSystems IRIS レジストリ・ディレクトリの場所 (ローカルの ファイル・システム上に配置する必要がある)。 /usr/local/etc/irissys
server_location.iris	インストール中に、iris が存在するディレクトリ。 /iriskit/dist/lnxrhx64/bin/shared/iris
server_location.is_dr_mirror	このインスタンスが災害復旧 (DR) ミラー・メンバかどうかを示 す。 (N/Y)

変数名	説明 ()
postinstall*	パラメータ・ファイル・フェーズの後で実行するパッケージを指定。 upgrade
install_mode.setup_type	インストールのタイプ。 (Development/Server/Custom)
unicode_selection.binary_type	インストールするバイナリのタイプ。 (unicode/eightbit)
unicode_selection.install_unicode	この製品の Unicode 版をインストールするかどうかを示す。 (N/Y)
security_settings.iris_user	InterSystems IRIS スーパーサーバの実効ユーザ irisusr
security_settings.iris_group	InterSystems IRIS の実効グループ irisusr
security_settings.manager_user	インスタンスの所有者。 root
security_settings.manager_group	インスタンスの開始および停止を許可されたグループ。 develop
security_settings.dbencrypted	起動時に暗号化キーを有効にするかどうかを示す。 (0/1)
security_settings.dbenckeyfile	暗号化キーのパス。 このパラメータは空白でもかまいません。
security_settings.dbenckeyuser	キーを有効にできる管理者の名前。 このパラメータは空白でもかまいません。
security_settings.dbenckeypassword	キー管理者のパスワード。パラメータ・ファイルが保存される前にクリアされます。 このパラメータは空白でもかまいません。
security_settings.personal_database	パーソナル・データベース機能を使用するかどうかを示す。 (N/Y)
security_settings.initial_level	初期のセキュリティ設定。 (Minimal/Normal/LockedDown)
security_settings.already_secured	5.1 以前のインスタンスからアップグレードしている場合に、セキュリティ設定が必要かどうかを示す。 (N/Y)

変数名	説明 ()
security_settings.password	irisinstall から実行している場合は、パラメータ・ファイルの格納前にパスワード・フィールドがクリアされる。
installer.manifest	DefaultInstallerClass.xml (エクスポートされた %Installer クラス) の場所。以下はその例です。 /home/user/Downloads/DefaultInstallerClass.xml
installer.manifest_parameters	インストーラ・マニフェストのパラメータの場所。 SourceDir=/home/user/Downloads
installer.manifest_loglevel	マニフェストのログ・レベルを指定する。 (-1/0/1/2/3)
installer.manifest_logfile	ログ・ファイル名を指定する。 /manifests/IRIS-installManifestLog.txt
port_selection.superserver_port	スーパーサーバ・ポート番号。 1972
port_selection.jdbcgateway_port	Java Database Connectivity (JDBC) ゲートウェイ・ポート番号。 62972
csp_gateway.configure	外部 Web サーバ用に Web ゲートウェイを構成するかどうかを示す。 (N/Y)
csp_gateway.web_server_type	Web ゲートウェイで使用する既存の Web サーバのタイプ。 (Apache/None)
csp_gateway.apache_version	Apache Web サーバのバージョン
csp_gateway.apache_user	Apache Web サーバのユーザ名
csp_gateway.apache_conf_file	Apache Web サーバ構成ファイルの場所。 /etc/httpd/conf/httpd.conf
csp_gateway.apache_pid_file	Apache Web サーバ・デーモンのプロセス ID を記録するファイル。 /usr/local/apache/logs/httpd.pid
csp_gateway.apache_use32bit	Apache Web サーバで 32 ビット・アーキテクチャを使用するかどうかを示す。 Y/N
csp_gateway.directory	Web ゲートウェイ・ファイルが保存されるディレクトリ。
license_key.enter_key	インストール中にキーをインストールするかどうかを示す。 N/Y

変数名	説明 ()
license_key.license_file	enter_key の値が Y の場合、キー・ファイル情報の場所。
agent.user_account	ISCAgent のユーザ名。 iscagent
agent.user_group	ISCAgent のグループ名。 iscagent
agent.install	ISCAgent がインストールされているかどうかを示す。 (N/Y)
client_location.target_dir	クライアントのみのインストールのターゲット・ディレクトリ。 test/IRIS
client_location.is_client_install	これがクライアント・インストールであるかどうかを示す。 (N/Y)
install*	database_server
postinstall*	database_server
japanese_docs.install	日本語ドキュメント・ソースをインストールするかどうかを示す。 (N/Y)
install*	インストールするコンポーネント名。

* このインストール変数は、インストールするコンポーネント 1 つにつき 1 回ずつ、パラメータ・ファイルに登場します。カスタム・インストール、またはクライアントのみのインストールでは、条件に応じて、以下のいずれか、またはすべてが生成されます。

- ・ dev_kit
- ・ odbc
- ・ cpp_binding
- ・ cpp_sdk
- ・ engine_link_libraries
- ・ light_cpp_binding
- ・ addenda
- ・ install_confirmation
- ・ copyright

