



# データ整合性ガイド

Version 2024.1  
2024-06-03

## データ整合性ガイド

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](mailto:support@InterSystems.com)

# 目次

1 復元可能性とデータ整合性 .....	1
1.1 体験 .....	1
1.2 詳細 .....	1
1.3 関連項目 .....	1
2 InterSystems IRIS でのデータ整合性 .....	3
2.1 基本的なデータ整合性の保護 .....	3
2.2 整合性の検証とリカバリのメカニズム .....	4
2.3 構造的な整合性の検証 .....	4
2.3.1 整合性チェックの誤検出 .....	5
2.3.2 整合性チェックの出力 .....	5
2.3.3 管理ポータルを使用した整合性のチェック .....	7
2.3.4 Integrity ユーティリティを使用したインタラクティブな整合性チェック .....	7
2.3.5 整合性チェック API .....	7
2.3.6 整合性チェックのパフォーマンスの調整 .....	9
2.3.7 整合性チェックの分離 .....	10
3 ライト・イメージ・ジャーナリングとリカバリ .....	13
3.1 ライト・イメージ・ジャーナリング .....	13
3.1.1 2 フェーズ・ライト・プロトコル .....	13
3.1.2 ライト・イメージ・ジャーナル (WIJ) 設定 .....	14
3.2 リカバリ .....	14
3.2.1 WIJ のリストア .....	14
3.2.2 WIJ ブロック比較 .....	15
3.3 ライト・イメージ・ジャーナリングの限界 .....	18
4 バックアップとリストア .....	19
4.1 バックアップの整合性と復元可能性 .....	19
4.2 ジャーナル・リストアの重要性 .....	20
4.3 バックアップの方法 .....	20
4.3.1 外部バックアップ .....	21
4.3.2 オンライン・バックアップ .....	23
4.3.3 コールド・バックアップ .....	24
4.3.4 並行外部バックアップ .....	25
4.4 バックアップからのリストア .....	26
4.4.1 バックアップ・リストアのシナリオ .....	26
4.4.2 メンテナンス・モードによる InterSystems IRIS の起動 .....	33
4.4.3 バックアップ・リストア後のジャーナル・リストア .....	34
4.4.4 自動 WIJ およびジャーナル・リカバリを使用しない InterSystems IRIS の起動 .....	34
4.5 オンライン・バックアップ設定の構成 .....	35
4.5.1 データベース・バックアップ・リストの定義 .....	36
4.5.2 バックアップ・タスクの構成 .....	36
4.5.3 バックアップ・タスクのスケジュール .....	37
4.6 オンライン・バックアップの管理 .....	37
4.6.1 バックアップ・タスクの実行 .....	38
4.6.2 バックアップ・ステータスの表示 .....	39
4.6.3 実行中のバックアップの中止 .....	39
4.6.4 バックアップ履歴の表示 .....	39
4.7 オンライン・バックアップ・ユーティリティ .....	40

4.7.1	DBSIZE によるバックアップ・サイズの見積もり	40
4.7.2	BACKUP によるバックアップおよびリストアのタスクの実行	43
4.7.3	DBACK によるデータベースのバックアップ	43
4.7.4	BACKUP によるバックアップに必要なディレクトリ・リストの編集/表示	48
4.7.5	BACKUP による実行中のバックアップの中止	50
4.7.6	BACKUP によるバックアップ・ボリュームに関する情報の表示	50
4.7.7	BACKUP によるバックアップまたはリストアの進行状況の監視	50
4.8	オンライン・バックアップ・リストア・ユーティリティ	51
4.8.1	DBREST によるすべてのデータベースのリストア	52
4.8.2	DBREST による選択したデータベースまたは名前を変更したデータベースのリストア	55
4.8.3	バックアップ履歴によるデータベースのリストア	56
4.8.4	DBREST による自動リストア	57
4.8.5	ミラーリングされたデータベースの考慮事項	59
5	ジャーナリングの概要	63
5.1	ジャーナリングの概要	63
5.2	ジャーナリングとライト・イメージ・ジャーナリングの違い	64
5.3	データベースの整合性の保護	64
5.4	トランザクションの自動ジャーナリング	64
5.5	不完全トランザクションのロールバック	65
5.6	データベースをジャーナリングしない場合の影響	65
5.7	ジャーナル書き込みサイクル	66
5.8	ジャーナル・ファイルとジャーナル履歴ログ	66
5.9	一時グローバルと IRISTEMP の使用	67
5.10	ジャーナル管理のクラスとグローバル	68
5.11	関連項目	68
6	ジャーナリングの構成	69
6.1	ジャーナリングの有効化	69
6.2	ジャーナル・ファイル名とロールオーバー	69
6.3	ジャーナリングの最善の使用方法	70
6.4	ジャーナル・アーカイブ・ターゲットの構成	71
6.5	ジャーナル設定の構成	72
6.6	関連項目	74
7	基本的なジャーナリング操作	75
7.1	ジャーナリングの開始	75
7.2	ジャーナリングの停止	75
7.3	ジャーナル・ファイルの表示	76
7.4	ジャーナル・ファイルの切り替え	77
7.5	ジャーナル・ディレクトリの切り替え	77
7.6	ジャーナル・ファイル・プロファイルの表示	78
7.7	ジャーナル・ファイルの整合性のチェック	78
7.8	ジャーナル・ファイルの要約の表示	78
7.9	ジャーナル・ファイルの削除	79
7.10	ミラー・ジャーナル・ファイルの削除	79
7.11	ジャーナル・ファイルのリストア	80
7.12	関連項目	80
8	ジャーナリング・ユーティリティ	81
8.1	JOURNAL を使用したジャーナリング・タスクの実行	81
8.1.1	JRNSTART を使用したジャーナリングの開始	82

8.1.2	^JRNSTOP を使用したジャーナリングの停止	82
8.1.3	^JRNSWTC を使用したジャーナル・ファイルの切り替え	83
8.1.4	SWDIR^JOURNAL を使用したジャーナル・ディレクトリの切り替え	83
8.1.5	^JRNRESTO を使用したジャーナル・ファイルからのグローバルのリストア	83
8.1.6	^JRNDUMP を使用したジャーナル・レコードの表示	93
8.1.7	PURGE^JOURNAL を使用したジャーナル・ファイルの削除	99
8.1.8	^JRNOPTS を使用したジャーナル設定の更新	100
8.1.9	ENCRYPT^JOURNAL を使用したジャーナルの暗号化	100
8.1.10	Status^JOURNAL によるジャーナルの状態の表示	100
8.1.11	Manage^JRNROLL を使用したトランザクション・ロールバックの管理	101
8.1.12	MirrorCatchup^JRNRESTO を使用したミラーリングされるデータベースへのジャーナルのリストア	102
8.2	^STURECOV を使用した起動エラーの回復	103
8.3	^JCONVERT および ^%JREAD を使用したジャーナル・ファイルの変換	106
8.4	^JRNMARK を使用したジャーナル・マーカの設定	109
8.5	^JRNUTIL を使用したジャーナル・ファイルの操作	110
8.6	DISABLE^%NOJRN を使用したプロセス・レベルでのジャーナリングの管理	111
8.7	^ARCHIVE を使用したジャーナル・アーカイブ・ファイルの管理	111
8.8	関連項目	111
9	ジャーナル入出力エラー	113
9.1	ジャーナルの [エラー時に凍結] 設定が [いいえ] の場合	113
9.2	ジャーナルの [エラー発生時に凍結する] 設定が [はい] の場合	114
9.3	ジャーナルの [エラー発生時に凍結する] 設定の TROLLBACK を使用したトランザクションのロールバックに対する影響	115
9.4	関連項目	115
10	ジャーナリングに関する特別な考慮事項	117
10.1	パフォーマンス	117
10.2	UNIX® ファイル・システムの推奨事項	117
10.3	システム時計の推奨事項	117
10.4	ファイリング操作に対するジャーナリングの無効化	118
10.5	関連項目	118
11	複数のシステムでのデータ整合性	119
11.1	DataCheck の概要	119
11.1.1	DataCheck クエリ	119
11.1.2	DataCheck ジョブ	120
11.1.3	DataCheck 結果	120
11.1.4	DataCheck ワークフロー	121
11.2	ミラー用の DataCheck の構成	122
11.2.1	ミラー内での DataCheck の計画	122
11.2.2	チェックするグローバルの選択	123
11.3	DataCheck のセットアップ手順	123
11.3.1	DataCheck サービスの有効化	124
11.3.2	チェックするグローバルおよび添え字範囲の指定	125
11.4	^DATACHECK ルーチン	128
11.4.1	新規構成の作成	129
11.4.2	構成の編集	129
11.4.3	詳細表示	131
11.4.4	DataCheck のソースであるこのシステムへの着信接続	131
11.5	データ・チェックに関する特別な考慮事項	132

11.5.1 パフォーマンスに関する考慮事項 .....	132
11.5.2 セキュリティに関する考慮事項 .....	133

# テーブル一覧

テーブル 4-1: バックアップ・タスクの説明 .....	36
テーブル 8-1: ^JRNDUMP で表示されるジャーナル・データ・レコード・フィールド .....	96
テーブル 8-2: ジャーナル・ファイルの処理 .....	97
テーブル 8-3: ^JRNUTIL で使用可能な関数 .....	110





# 1

## 復元可能性とデータ整合性

InterSystems IRIS では、さまざまな高度なメカニズムを使用して、エラー、障害、および災害からデータの構造と論理整合性を保護します。システム障害の発生時に、[ライト・イメージ・ジャーナリング](#)・テクノロジーによって構造的な整合性が失われないように保護され、[ジャーナリング](#)とトランザクション処理によって論理的な整合性が維持されます。[バックアップ](#)方法と組み合わせることで、InterSystems IRIS のデータ整合性メカニズムは、システム障害からの迅速なリカバリをデータ損失なしに提供します。

### 1.1 体験

[InterSystems IRIS デモ：データ復元可能性とミラーリング](#)

### 1.2 詳細

[InterSystems IRIS でのデータ整合性](#)

[システム管理ガイド](#)

### 1.3 関連項目

[バックアップとリストア](#)

[ミラーリングと高可用性](#)



# 2

## InterSystems IRIS でのデータ整合性

すべての InterSystems IRIS® データ・プラットフォーム・データベースのデータの整合性は、このドキュメントの説明にある機能により、インスタンスおよびシステムの障害の影響から保護されています。

InterSystems IRIS ライト・イメージ・ジャーナリング・テクノロジーによって構造的な整合性が失われるように保護され、InterSystems IRIS のジャーナリングとトランザクション処理によって論理的な整合性が維持されます。バックアップ方法とジャーナリングの組み合わせにより、物理的な整合性が失われても迅速なリカバリが可能です。

ここでは、構造的な整合性がどのように維持されているか、およびその検証方法について説明します。この後のセクションでは、以下のトピックについて説明します。

- ・ [ライト・イメージ・ジャーナリング・テクノロジー](#)は、システム障害の発生時に構造的な整合性が失われるように保護します。
- ・ [バックアップおよびリストア方法](#)は、ジャーナリングと組み合わせて使用することで、物理的な整合性が失われても迅速なリカバリが可能です。
- ・ [ジャーナリングおよびトランザクション処理](#)では、論理的な整合性が維持され、バックアップおよびリストアと共に使用することで、物理的な整合性が失われても迅速なリカバリが可能です。
- ・ [DataCheck ユーティリティ](#)では、2 つのシステムに存在するデータの状態を比較して、一貫性があるかどうかを判断できます。

### 2.1 基本的なデータ整合性の保護

一般に、整合性は 2 つの異なるレベルで表示できます。

- ・ 構造的なデータベースの整合性または物理的な整合性は、ディスク上のデータベース・ブロックのコンテンツを指します。構造的な整合性を持つには、データベース・ブロックが自己整合しており、グローバルが検索可能でなければなりません。システム・クラッシュ時の構造的な整合性は、"[ライト・イメージ・ジャーナリングとリカバリ](#)" に説明があるインターシステムズのライト・イメージ・ジャーナル (WIJ) テクノロジー、および InterSystems IRIS の内部アルゴリズムによって保持されます。
- ・ 論理的な整合性は、データベース内のグローバルによって表現されるデータを指し、アプリケーションによって作成されるデータの自己一貫性、そのトランザクション整合性、および現実との最新性が含まれます。システム・クラッシュ時の論理的な整合性は、InterSystems IRIS ジャーナリング ("[ジャーナリング](#)" を参照) およびトランザクション処理によって保持されます。(論理的な整合性の他の点は、インタロック、トランザクション、およびアプリケーションが使用するプログラミング・パラダイムに固有の他のメカニズムを適切に使用することにより、アプリケーション・コードで管理されます。)

自動的な WIJ およびジャーナルのリカバリは、InterSystems IRIS データベースをシステム障害から保護する、インターシステムズのエラープルーフ・データベース・アーキテクチャの基本的な構成要素です。

## 2.2 整合性の検証とリカバリのメカニズム

システム・クラッシュだけで整合性の低下を招く可能性はないとしても、ストレージ・デバイスに破壊的な障害が生じたり、物理的な損傷を受けたり、改ざんされたりする可能性は常に存在します。その場合、データベース、WIJ、およびジャーナルの整合性が危険にさらされる可能性があります。そうした災害の影響を軽減するために、InterSystems IRIS は以下の機能を提供します。

- ・ データベースの構造的な整合性を確認するためのツール。詳細は、“[構造的な整合性の検証](#)”を参照してください。
- ・ バックアップ・メカニズム。詳細は、“[バックアップとリストア](#)”を参照してください。
- ・ ミラーリング (“[ミラーリング](#)”を参照)を使用した、自動フェイルオーバーおよび災害復旧のためのジャーナリングベースの論理データ・レプリケーション。
- ・ DataCheck。これは、ミラーリングなどのテクノロジーがデータの複製コピーを保持する際、複数のシステム間でデータの整合性を確認するツールです (“[複数のシステムでのデータ整合性](#)”を参照)。

## 2.3 構造的な整合性の検証

整合性チェックにより、データベースのセット、またはデータベース内のグローバルのサブセットの構造的な整合性を検証します (“[基本的なデータ整合性の保護](#)”を参照)。

整合性チェックの実行の利点は以下のとおりです。

- ・ 整合性チェックとバックアップ方法を統合して、バックアップ時にデータベースのコピーが損なわれず、バックアップ自体にエラーが入り込んでいないことを確認できます。詳細説明は、“[外部バックアップ](#)”にあります。
- ・ 整合性チェックでは、破損が生じる前にそれを検出できるため、ユーザが影響を受ける前に計画を立てる時間があります。
- ・ 定期的な整合性チェックにより、検出された構造的な整合性の問題の原因を早めに正確に突き止められるため、根本原因を特定できる可能性が高くなります。

整合性チェックにより、選択したデータベース内のすべてのグローバルの整合性、または指定された単一のデータベースに格納されている選択したグローバルの整合性を検証できます。整合性チェックは、管理ポータルから実行することも、ターミナル・ウィンドウで Integrity ユーティリティを使用して実行することもできます。このセクションでは、以下のトピックについて説明します。

- ・ [整合性チェックの誤検出](#)
- ・ [整合性チェックの出力](#)
- ・ [管理ポータルを使用した整合性のチェック](#)
- ・ [Integrity ユーティリティを使用した整合性のチェック](#)
- ・ [整合性チェック API](#)
- ・ [整合性チェックのパフォーマンスの調整](#)
- ・ [整合性チェックの分離](#)

## 2.3.1 整合性チェックの誤検出

揮発性データベースに対して整合性チェックを実行すると、進行中のデータベース更新が原因で、データベース整合性エラーのレポートが誤って生成されることがあります。

“[管理ポータルを使用したデータベース整合性のチェック](#)”で説明されているように、整合性チェックが管理ポータルまたはタスク・マネージャから実行される場合、バックグラウンドで実行され、エラーが検出されたすべてのグローバルを自動的に再テストします。この自動の2回目のパスを含む整合性チェックからの出力では、以下の方法でエラーを報告します。

- ・ グローバルのエラーが1回目のパスでは検出されたが、2回目のパスでは検出されない場合、最初のエラーは誤検出と想定され、エラーは報告されません。
- ・ 2回目のパスで検出されたグローバルのエラーが、1回目のパスで検出されたエラーと異なる場合、2回目のパスのエラーのみ報告され、`These errors in global <global_name> differ from the errors prior to the retry` というテキストが表示されます。
- ・ 両方のパスでグローバルに同じエラーが検出される場合、エラーは、`When retried the errors in global <global_name> remained unchanged` というメッセージで報告されます。

“[Integrity ユーティリティを使用したデータベース整合性のチェック](#)”に説明がある Integrity ユーティリティまたはエントリ・ポイントのいずれかを使用して手動で実行される整合性チェックでは、1回目のパスでエラーが報告されたグローバルの再テストは行われません。エラーが返される場合、その特定のデータベースのチェックを繰り返します。

一般に、アクティブ・システムに対して実行される整合性チェックの場合、2回目のパスで繰り返されないエラーは誤検出であり、2回目のパスで持続するエラーは実際の整合性の問題を表します。後者のエラーは調査する必要があります。また、前者のエラーも、アクティビティのレベル、エラーの数、以前に誤検出が発生したエクステンツに応じて、調査するメリットがあります。調査の種類は、専門知識のレベルや過去の誤検出の経験によって決まります。実行できる手順は以下のとおりです。

- ・ 可能であれば、システム・アクティビティの少ない期間に、整合性チェックを再度実行します。
- ・ 最新のバックアップのリストアされたコピーに対して整合性チェックを実行します。
- ・ 根本原因の手掛かりを得るために、問題となっているデータの範囲を確認します。
- ・ [インターシステムズのサポート窓口](#)に問い合わせます。

誤検出の問題は、“[外部バックアップ](#)”で説明されているような標準のバックアップ手順に、整合性チェックを統合することで回避できます。これにより、データベースは、そのデータベースが存在する論理ディスク・ボリュームのスナップショットが取られた直後に、“[整合性チェックの分離](#)”で説明されているように、プロダクションから切り離してチェックされるようになります。

## 2.3.2 整合性チェックの出力

整合性チェックは、検出したエラーを報告するだけでなく、各グローバルに含まれるブロックの数と、使用中のそれらのブロックの割合についての情報をブロック・レベルで分割して報告します。その例として、20,000 件のユーザのデータが入力された **DATA** データベースに対する整合性チェックの出力の一部を次に示します。

```
File Name: c:\intersystems\20182555dec15a\mgr\integ.txt
IRIS Database Integrity Check - Report Created 01/25/2018 10:41:16
System: BBINSTOCK6440 Configuration: 20182555DEC15A

No Errors were found.

Full Listing of Databases Checked

Directory: C:\InterSystems\20182555DEC15A\Mgr\DATA\
```

```

0 globals with errors found

Global: Aviation.AircraftD                                0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (6% full)
Data Level:           # of blocks=64     512kb (87% full)
Total:                 # of blocks=65     520kb (85% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:15

Global: Aviation.AircraftI                                0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (0% full)
Data Level:           # of blocks=4      32kb (83% full)
Total:                 # of blocks=5      40kb (67% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:15

Global: Aviation.Countries                                0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (0% full)
Data Level:           # of blocks=1      8kb (52% full)
Total:                 # of blocks=2      16kb (26% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:15

Global: Aviation.CrewI                                    0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (1% full)
Data Level:           # of blocks=5      40kb (90% full)
Total:                 # of blocks=6      48kb (75% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:15

Global: Aviation.EventD                                    0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (41% full)
Data Level:           # of blocks=377    3,016kb (78% full)
Big Strings:          # of blocks=776    6,208kb (72% full) # = 479
Total:                 # of blocks=1,154 9,232kb (74% full)
Elapsed Time = 0.1 seconds, Completed 01/25/2018 10:41:15

Global: Aviation.EventI                                    0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (0% full)
Data Level:           # of blocks=3      24kb (77% full)
Total:                 # of blocks=4      32kb (58% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:15

...

Global: ROUTINE                                            0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (1% full)
Data Level:           # of blocks=6      48kb (78% full)
Total:                 # of blocks=7      56kb (67% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:16

Global: SYS                                                0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (0% full)
Data Level:           # of blocks=1      8kb (0% full)
Total:                 # of blocks=2      16kb (0% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:16

Global: Data.CompanyD                                      0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (0% full)
Data Level:           # of blocks=1      8kb (35% full)
Total:                 # of blocks=2      16kb (17% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:16

Global: Data.CompanyI                                      0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (0% full)
Data Level:           # of blocks=1      8kb (9% full)
Total:                 # of blocks=2      16kb (4% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:16

Global: Data.PersonD                                       0 errors found
Top/Bottom Pnt Level: # of blocks=1      8kb (0% full)
Data Level:           # of blocks=5      40kb (81% full)
Total:                 # of blocks=6      48kb (67% full)
Elapsed Time = 0.0 seconds, Completed 01/25/2018 10:41:16

...

```

管理ポータルからの実行時には、整合性チェックで生成されたエラーと警告があれば、レポートの先頭にそのリストが示されます。Integrity ユーティリティを使用して実行する場合、エラーの概要は出力の末尾に示されます。

### 2.3.3 管理ポータルを使用した整合性のチェック

選択した複数のデータベースの整合性や、1 つのデータベースに格納された複数の選択したグローバルの整合性をチェックするには、管理ポータルの [データベース] ページ ([ホーム]→[システム処理]→[データベース]) に移動し、次の手順を実行します。

1. **[整合性チェック]** をクリックして、データベースのリストを表示します。
2. 確認対象のデータベースのチェック・ボックスにチェックを付けます。  
単一のデータベースに格納されたグローバルをチェックする場合は、チェック対象のグローバルが格納されたデータベースのみを選択して **[グローバルを選択]** をクリックし、選択したデータベースに格納されたグローバルのリストを表示します。チェックするグローバルを選択して、**[保存]** をクリックします。リストからグローバルを選択していないと、選択したデータベースのグローバルがすべてチェックされます。
3. エラー発生時にデータベース整合性チェックを停止する場合、**[エラー発生後に停止する]** チェック・ボックスにチェックを付けます。
4. **[OK]** をクリックし、整合性チェックを開始します。整合性チェックのプロセスがバックグラウンドで実行されます。

**[整合性ログ]** をクリックすると、ポータルを使用して最後に実行した整合性チェックの出力が表示されます。このファイルのパスと内容が自動的に表示されます。

整合性チェックは、また、タスク・マネージャの、既定のシステム・バックグラウンド・タスクの 1 つです。必要に応じて、複数の整合性チェックをスケジュールすること（異なる時間に異なるデータベースをスケジュールするなど）もできます。システム・タスクのスケジューリングの詳細は、[“タスク・マネージャの使用”](#) を参照してください。

注釈 データベースはマウントされているが、管理ポータルのデータベース構成に一度も追加されたことがない場合、または管理ポータルのデータベース構成から削除された場合（[“データベースの構成”](#) を参照）、そのデータベースは**整合性チェック**機能で表示されるデータベース・リストには含められません。

### 2.3.4 ^Integrity ユーティリティを使用したインタラクティブな整合性チェック

手動の整合性チェックは、^Integrity ユーティリティを使用して実行できます。このユーティリティを実行するには、ターミナル・ウィンドウを開き、%SYS ネームスペースに切り替えて、**do ^Integrity** と入力します。これは管理ポータルの [データベース] ページから整合性チェックを実行することと同様です。ただし、[“整合性チェックの誤検出”](#) で説明したとおり、^Integrity ユーティリティは、完了して結果を報告する前に、1 回目のパスでエラーが検出されたグローバルを再チェックすることはできないため、エラーが報告されたグローバルを再チェックして誤検出を排除することが重要になります（管理ポータルの整合性チェックでは、^Integrity ユーティリティのような単一のジョブを実行するのではなく、整合性チェックも複数のジョブに分散されます）。

以下の ^Integrity エントリ・ポイントをインタラクティブに使用することもできます。

- ・ **Do CheckPointer^Integrity** は、チェックを開始するディレクトリとポインタ・ブロックを尋ねます。
- ・ **Do Exclude^Integrity** でチェックから除外するためのデータベースのリストを要求します。? を入力してマウントされたデータベースのリストを表示します。

### 2.3.5 整合性チェック API

以下の ^Integrity エントリ・ポイントは、プログラムによる使用が可能です。

- ・ **Do**  
**CheckList^Integrity(outputglobal,dirlist,stopafteranyerror,listofglolist,maxproc,partialcheck)**  
は整合性チェックを実行し、情報や警告を含む結果を格納します。パラメータの使用法は以下のとおりです。

- **outputglobal**

結果を格納するグローバルを指定します。このパラメータの呼び出しを省略した場合、結果は `^IRIS.TempIntegrityOutput(+$JOB)` に格納されます。このパラメータを使用する際には、`$datat(@outputglo)=0` であること、およびこれが複数の並行呼び出しで共有されていないことを確認する必要があります。

- **dirlist**

チェックするすべてのディレクトリの `$LIST` を指定します。省略した場合は、すべてのディレクトリをチェックします。

- **stopafteranyerror**

エラー発生時の整合性チェックの動作を指定します。1 を指定した場合、エラーが検出されるとディレクトリのチェックは停止され、0 (既定値) を指定した場合、エラー発生時にチェックは続行されます。

- **listofglolist**

グローバル名を指定した `$LIST` を複数含む `$LIST` を、**dirlist** で指定されたディレクトリごとに 1 つ指定します。これらのパラメータを使用すると、例えば、`$LB($LB("oddDEF"))` を指定してすべてのディレクトリですべての `oddDEF` グローバルをチェックできます。**listofglolist** の要素が **dirlist** よりも少ない場合、前者の最後の要素が後者の残りのディレクトリに使用されます。

- **maxproc**

使用する並列処理の最大数を指定します。既定値は 8 です。1 未満の値を指定した場合、ホスト・システムのコアの数が使用されます。**maxproc** が 1 より大きい場合、効率化のために内部的にその値が使用されます。**maxproc** が 1 の場合、並列処理なしでチェックが実行されます。

- **partialcheck**

既定値 0 を指定すると、グローバル全体がチェックされ、大規模なグローバルの場合は範囲別に分割できます。-1 を指定した場合もグローバル全体がチェックされますが、大規模なグローバルを範囲別に分割することはできません。この値は、予期しない問題が発生した場合のフォールバックとしてのみ提供されており、今後削除される可能性があります。1 を指定すると、大規模なグローバルの場合はポインタ構造のみがチェックされます。この値ではそれほど詳細なチェックは行われず、データ・ブロックや大規模な文字列ブロックのエラーは検出されません。

`Do CheckList^Integrity` は `%Status sc` を返し、これは次のように評価されます。

- `$system.Status.IsOK(sc)` が返される場合、整合性チェックが実行され、エラーは検出されていません。
- 単一のエラー・コード `$$$IntegrityCheckErrors` の戻りステータス、すなわち `$system.Status.GetError-Codes(sc)=$$ERRORCODE($$IntegrityCheckErrors)` の場合、整合性チェックは成功していますが、エラーが検出されています。
- 前述のどちらも返されない場合、問題が発生し、整合性チェックで完全な結果を返せない可能性があります。この場合、`sc` に複数のエラー・コードが含まれ、そのいずれかが `$$$IntegrityCheckErrors` である可能性があります。

**Do Display^Integrity(integritout,flags,dirsum)** は、整合性チェックの結果を表示します。パラメータの使用法は以下のとおりです。

- **integritout**

整合性チェックの結果を格納したグローバルの名前を (`CheckList^Integrity` により) 指定します。指定がない場合は、既定の `^IRIS.TempIntegrityOutput(+$JOB)` になります。

- **flags**

以下の値に基づいて、表示されるメッセージを決定します。



- ・ 0 は、すべてのメッセージを表示します。
- ・ 1 は、エラーと警告のみを表示します。
- ・ 2 は、エラーのみを表示します。

指定されていない場合は、すべてのメッセージが表示されます。

#### - dirsum

指定されていて 0 ではない場合、表示にはスキャンされたそれぞれのディレクトリのブロックの概要が含まれます。

以下に、5 つのプロセスを使用して 3 つのデータベースをチェックする例を示します。ここで `dblist` パラメータを省略すると、すべてのデータベースがチェックされます。(上記 `CheckList Integrity` の戻り値の説明で記載した `sc` の評価で、結果を表示する必要はありません。)

```
set dblist=$listbuild("/data/db1/", "/data/db2/", "/data/db3/")
set sc=$$CheckList^Integrity(,dblist,,5)
do Display^Integrity()
kill ^IRIS.TempIntegrityOutput(+$job)
```

以下のエントリ・ポイントは、従来の使用法でのみサポートされます。

- ・ **Do Silent^Integrity(logfilename,dirlist)** は、選択したデータベース、またはすべてのデータベースで整合性チェックを実行するバックグラウンド・プロセスを開始し、結果を `logfilename` パラメータで指定されたファイルに出力します。オプションの `dirlist` パラメータは、チェック対象データベースの `$LIST` を指定します。指定しない場合は、すべてのデータベースがチェックされます。これは、管理ポータルの [データベース] ページから整合性チェックを実行するのと同じです。
- ・ **Do SilentGlobalCheck^Integrity(logfilename,dir,gblist)** は、選択したデータベース内の選択したグローバルの整合性チェックを実行するバックグラウンド・プロセスを開始し、結果を `logfilename` パラメータで指定されたファイルに出力します。必須の `dir` パラメータは、チェックするグローバルを含むデータベースを特定します。必須の `gblist` パラメータは、チェックする 1 つ以上のグローバルの `$LIST` を指定します。これは、管理ポータルの [データベース] ページから整合性チェックを実行するときに、**[グローバル選択]** を選択するのと同じです。
- ・ **Do Query^Integrity(logfilename,outdevice)** は整合性チェックを実行しませんが、`logfilename` パラメータで指定されたファイルのコンテンツ(前回の実行で保存された結果)を、オプションのパラメータ `outdevice` で指定されたデバイスに出力します。`outdevice` の例として、現在のデバイス(既定)、プリンタ、その他のディスプレイ・デバイス、その他のオペレーティング・システム・ファイル名(`logfilename` のコピー先)があります。

## 2.3.6 整合性チェックのパフォーマンスの調整

整合性チェックでは、チェック対象のグローバルのすべてのブロックを(まだバッファに入っていない場合)、各グローバルの構造で指示された順序で読み取る必要があるため、この処理にはかなりの時間がかかり、ストレージ・サブシステムの帯域幅の多くを使用する可能性があります。整合性チェックの速度とパフォーマンスに与える影響の最適なバランスは、整合性チェックを実行する理由とタイミングによって決まります。例えば、ストレージの破損を伴う災害に対応して整合性チェックを実行する場合は、できるだけ早く結果を入手したいと考えますが、プロダクションと並行して実行する場合は、ストレージ・サブシステムへの影響を最小限に抑えたいと考えるでしょう。

整合性チェックでは、ストレージ・サブシステムで可能な限りの速度で読み取りを行うことができます。使用中の整合性チェック・プロセスの数に、各プロセスに許容される同時有効非同期読み取りの最大数(既定値は 8)を掛けた値が、全体的な同時読み取り数の上限ですが、平均値はその半分です。この予測値をストレージ・サブシステムの容量と比較して、最適なプロセス数を決定します。例えば、ストレージが 20 のドライブにストライピングされ、プロセス当たりの同時読み取り数が既定の 8 の場合、ストレージ・サブシステムの全容量を捕捉するには、5 つ以上のプロセスが必要です( $5 \times 8 / 2 = 20$ )。(プロセスへの割り当てはグローバルごとに行われるため、特定のグローバルは常に 1 つのプロセスのみによってチェックされます。)

整合性チェックのパフォーマンスを調整するための推奨アプローチは次のとおりです。

1. 最初の手順では、整合性チェックを起動するための最適な方法を次のように選択します。
  - ・ 整合性チェックを起動する CheckList^Integrity エントリ・ポイントでは、プロセス数を指定できるため、パフォーマンスを最大限制御できます。このため、整合性チェックのパフォーマンスを調整するための最も簡単なアプローチとなっています。整合性チェックで使用する並列プロセスの数を指定する、CheckList^Integrity の `maxproc` 引数については、“[整合性チェック API](#)”を参照してください。
  - ・ 管理ポータルの[整合性チェック]オプションとタスク・マネージャの[整合性チェック]システム・バックグラウンド・タスクは複数のプロセスを使用しますが、CheckList^Integrity と同じ制御ではありません。管理ポータルのオプション、タスク・マネージャのタスク、および `SYS.Database.IntegrityCheck()` API 呼び出しはすべて、CPU コア数と同じプロセス数（一般に比較的大きな数になります）を選択します。また、これらのインタフェースは、同時更新による誤検出を特定する際にエラーが報告されたグローバルの完全な再チェックも実行します。この再チェックは、整合性チェック・アルゴリズムに組み込まれた誤検出の軽減策に加えて行われるもので、時間がかかるため不要な場合があります。
  - ・ ターミナルの `^Integrity` ルーチンや `Silent^Integrity` エントリ・ポイントなどその他の起動方法は、整合性チェックを単一プロセスで実行するため、すぐに結果が必要なものには役に立ちません。これらの起動方法にも、結果をファイルやターミナルに即座に出力して、整合性チェックの進行中にユーザに表示できるという利点があります。
2. 整合性チェック・プロセスではグローバルのポインタ・ブロックを一度に 1 つずつ、指し示すデータ・ブロックの内容に対して検証します。ストレージ・サブシステムが複数の同時有効読み取り要求を処理できるように、データ・ブロックは非同期入出力で読み取られ、各読み取りが完了すると検証が実行されます。既定では、同時読み取り要求の最大数は 8 です。以下の点に注意してください。
  - ・ 使用中のプロセスの数にプロセス当たりの最大同時読み取り数を掛けた値が、全体的な同時読み取り数の上限であるため、整合性チェックの並列プロセス数を変更することが、通常最初に行う調整になります。ただし、最大同時読み取りパラメータを変更すると、追加制御が提供される場合があります。また、整合性チェックが単一プロセスに制限されている場合は（例えば、1 つの非常に大きいグローバルがある場合や、その他の外部制約がある場合）、このパラメータを調整することが、パフォーマンスを調整する主な手段となります。
  - ・ このパラメータを引き上げるメリットは、ストレージ・サブシステムの同時読み取りを処理する能力によって制限されます。データベースが単一のローカル・ドライブに格納されている場合、値が大きいとメリットはなくなります。他方、多数のドライブにストライピングされたストレージ・アレイは多数の読み取りを同時に処理することができます。このメリットは、さまざまな要素の中で、特に計算時間によっても制限されます。

同時読み取りの最大数を調整するには、`%SYS` ネームスペースでターミナルを開いて、`write $$GetAsyncReadBuffers^Integrity()` を入力して現在の値を表示し、必要に応じて `do SetAsyncReadBuffers^Integrity(value)` を入力して変更します。最大値は 128 です。変更は、次にグローバルをチェックするときに反映されます。（この設定は、インスタンスが再起動されている間、保持されるわけではありません。）

ディスク上でブロックが連続している場合に（または、それに近い場合）、各読み取りの最大サイズを制御する、似たようなパラメータがあります。このパラメータが必要になることはそれほどありませんが、待ち時間の長いストレージや、ブロック・サイズが大きいデータベースは、調整することで効果が得られる可能性があります。このパラメータのコマンドは `write $$GetAsyncReadBufferSize^Integrity()` と `do SetAsyncReadBufferSize^Integrity(value)` です。値は 64 KB の単位で設定されるため、値 1 は最大値として 64 KB を設定し、4 は 256 KB を設定し、最大 512 (32,768 KB) まで設定できます。既定値は 0 で、インスタンスが値を選択できます。現在、64 KB を表す 1 が選択されています。

## 2.3.7 整合性チェックの分離

多くのサイトでは、プロダクション・システムで直接、定期的に整合性チェックを実行しています。これは簡単であるというメリットがありますが、整合性チェックがストレージの帯域幅に与える影響が懸念されるほかに、データベース同時更新ア

クティビティで誤検出エラーが発生することもあります（組み込みの移行手段にもかかわらず）。プロダクション・システムでの整合性チェックでこのエラーが報告された場合、この評価や再チェックは管理者が行う必要があります。

誤検出を避けるために、整合性チェックをプロダクションから切り離すことができます。それには、ストレージ・スナップショットやバックアップ・イメージを別のホストにマウントして、切り離された InterSystems IRIS インスタンスで整合性チェックを実行させます。ストレージもプロダクションから切り離されていれば、整合性チェックのパフォーマンスを最大限に高め、プロダクションのストレージに与える影響を心配することなく、結果を可能な限り早く入手することができます。このアプローチは、整合性チェックを使用してバックアップを検証する場合に適しています。検証されたバックアップは、バックアップを作成した時点でのプロダクション・データベースを効果的に検証します。また、クラウドおよび仮想化プラットフォームでも、スナップショットから使用に適した切り離された環境を容易に構築できます。



# 3

## ライト・イメージ・ジャーナリングとリカバリ

InterSystems IRIS® は、ライト・イメージ・ジャーナリングを使用して、InterSystems IRIS データベースの内部の整合性を維持します。これは、データベース・リカバリ処理の基本です。

### 3.1 ライト・イメージ・ジャーナリング

InterSystems IRIS は、ライト・イメージ・ジャーナリングという 2 フェーズ技術を使用してデータベースの更新を保護します。このプロセスでは、InterSystems IRIS の起動時にライト・デーモンがライト・イメージ・ジャーナル・ファイル (WIJ) を作成します。データベースの更新は、最初はメモリからこの中間ジャーナル **IRIS.WIJ** に書き込まれ、次にデータベースに書き込まれます。システムが第 2 フェーズ中にクラッシュした場合、更新処理はリカバリを再度適用します。

以下の項目の詳細を説明しています。

- ・ [2 フェーズ・ライト・プロトコル](#)
- ・ [ライト・イメージ・ジャーナル \(WIJ\) 設定](#)

#### 3.1.1 2 フェーズ・ライト・プロトコル

InterSystems IRIS は、データベースが高速に効率よく検索、更新できるような構造で、アプリケーション・データを維持します。一般的に、アプリケーションがデータを更新する際、InterSystems IRIS は、データベース構造で複数のブロックを更新し、変更を反映する必要があります。クラッシュが発生した場合、InterSystems IRIS は 2 フェーズ・ライト・プロトコルを使用してデータベースの整合性を保護し、ブロック書き込みのサブセットが失われるのを防ぎます。このプロトコルは次のように動作します。

- ・ 第 1 フェーズでは、InterSystems IRIS は WIJ に、更新されたブロックを書き込みます。すべての更新が確実に書き込まれたら、InterSystems IRIS は WIJ にフラグを設定して、リストアすべきブロックがあることを示します。その後、第 2 フェーズが開始されます。
- ・ 第 2 フェーズでは、InterSystems IRIS は、WIJ 内に記録されたものと同じブロック・セットをディスク上のデータベースに書き込みます。これらのブロックが確実に書き込まれたら、InterSystems IRIS はフラグをクリアして、WIJ からリストアすべきブロックがないことを示します。

InterSystems IRIS を起動すると、WIJ が自動的にチェックされ、異常なシャットダウンが検出された場合はリカバリ・プロセスが実行されます。このプロセスが正常に完了すれば、データベースの内部整合性はリストアされています。シャットダウンに続き、InterSystems IRIS は WIJ のリカバリも実行します。これはデータベースが確実にバックアップされるようにするための安全対策です。

### 3.1.2 ライト・イメージ・ジャーナル (WIJ) 設定

WIJ ファイルは、既定では **IRIS.WIJ** という名前になり、システム管理者ディレクトリ (`install-dir/mgr`) に格納されます。`install-dir` は、インストール・ディレクトリです。管理ポータル の **[ジャーナル設定]** ページから、このファイルの別の場所を指定したり、ターゲット・ファイル・サイズを指定することができます。そのためには、以下の操作を実行します。

1. 管理ポータル の **[ジャーナル設定]** ページ (**[システム管理]**→**[構成]**→**[システム構成]**→**[ジャーナル設定]**) に移動します。
2. WIJ の新しい格納場所を **[ライトイメージジャーナルディレクトリ]** ボックスに入力し、**[保存]** をクリックします。この名前にはシステム上の既存のディレクトリを指定する必要があり、最長で 63 文字です。クラスタ化されたインスタンスでこの設定を編集した場合、変更内容を適用するには、InterSystems IRIS を再起動してください。スタンドアロン・インスタンスの場合、再起動は必要ありません。
3. **[WIJ のターゲット・サイズ (MB) (0 = 未設定)]** というプロンプトが表示されたら、WIJ のターゲット・サイズを入力します。既定値のゼロを使用すると、必要に応じて WIJ を増加させることができますが、そのための領域は予約されません。ゼロ以外の値を入力すると、指定した領域がストレージ・デバイス上で予約されます。

インスタンスの `iris.cpf` ファイルに含まれる上述の 2 つの設定の詳細は、“構成パラメータ・ファイル・リファレンス” の `[config]` セクションの “`targwijsz`” および “`wijdir`” を参照してください。

## 3.2 リカバリ

システム・クラッシュなどの重大なシステム障害が発生した場合には、WIJ のリカバリが必要となります。InterSystems IRIS を起動すると、WIJ が自動的にチェックされます。異常なシャットダウンが検出された場合はリカバリ・プロシージャが実行されます。2 つのフェーズで構成されるライト・プロトコル・プロセスのどちらに WIJ があるかに応じて、リカバリは以下のように実行されます。

- ・ WIJ の最終更新が完了した後、対応するデータベースの更新が完了する前 (つまり、プロセスの第 2 フェーズ中) にクラッシュが発生した場合は、“**WIJ のリストア**” の説明に従って WIJ がリストアされます。
- ・ WIJ の最終更新が確実にデータベースに書き込まれた後 (つまり、両方のフェーズが完了した後) にクラッシュが発生した場合は、“**WIJ ブロック比較**” の説明に従って最新の WIJ 更新と影響を受けたデータベースの間のブロック比較が行われます (Windows および UNIX®/Linux のみ)。

### 3.2.1 WIJ のリストア

WIJ が “アクティブ” とマークされている場合、ライト・デーモンは WIJ への変更されたディスク・ブロックの書き込みを完了していますが、それぞれのデータベースへの書き込みは完了していません。この場合、WIJ のリストアが必要です。リカバリ・プログラム `iriswdimj` は、以下を実行します。

- ・ メッセージ・ログ (`messages.log`) ファイルによってシステム管理者に通知します。“監視ガイド” の “管理ポータルを使用した InterSystems IRIS の監視” の章にある “**ログ・ファイルの監視**” を参照してください。
- ・ **データセット・リカバリ**を実行します。

通常すべてのリカバリは、`iriswdimj` プログラムの 1 回の実行で処理されます。



### 3.2.1.1 データセット・リカバリ

データセットは、特定の InterSystems IRIS システム上の特定のデータベース・ディレクトリです。iriswdimj プログラムは、異常なシャットダウン後に再起動する InterSystems IRIS インスタンスで構成されたすべてのデータセットをリストアします。

iriswdimj プログラムは、対話式または非対話式で実行できます。実行方法は、以下のようにプラットフォームによって異なります。

- ・ Windows — 常に非対話式で実行されます。
- ・ UNIX®/Linux — エラーが発生するまでは非対話式で実行され、その後プロンプトに応答するオペレータがいる場合には対話式で実行されます。

**注釈** UNIX/Linux システムで iris start quietly コマンドを使用すると、このコマンドは常に非対話式で実行されます。

リカバリ・プロシージャが完了すると、iriswdimj は WIJ の内容に“削除済み”とマークし、起動が継続されます。

書き込み中にエラーが発生した場合は、WIJ がアクティブ状態のままとなり、InterSystems IRIS は起動しません。このオプションが（対話モードで）上書きされない限り、次の InterSystems IRIS 起動時にリカバリが繰り返されます。

**注意** オプションを上書きして WIJ をリストアすると、データベースが破損するか、またはデータが消失します。

以下の項目を詳細に説明します。

- ・ [対話式のデータセット・リカバリ](#)
- ・ [非対話式のデータセット・リカバリ](#)

#### 対話式のデータセット・リカバリ

リカバリ手順により、データセットごとにリカバリするかどうかを確認できます。通常は、すべてのデータセットを指定します。それぞれのデータセットのプロンプトの後、以下を入力します。

- ・ Y — データセットをリストアする
- ・ N — データセットをリストアしない

また、データセットへのパスが失われても、そのデータセットにアクセスできる場合は、そのデータセットの新しい場所を指定できます。データセットがリカバリされると、リカバリを要求するデータセットのリストからそのデータセットは削除されます。また、iriswdimj プログラムが引き続き実行されている間は、リカバリする必要が発生してもリカバリされません。

#### 非対話式のデータセット・リカバリ

リカバリ手順が非対話式で実行される場合、InterSystems IRIS はすべてのデータセットをリストアし、WIJ を削除済みとマークしようとします。Unix® プラットフォームおよび Windows プラットフォームでは、InterSystems IRIS は、最初にすべてのデータセットの高速並行リストアを試行します。高速リストア中に 1 つまたは複数のエラーが発生すると、データセットは一度に 1 つずつリストアされ、完全にリカバリされたデータベースを識別できるようにします。少なくとも 1 つのデータセットをリストアできない場合、以下のようになります。

- ・ iriswdimj プログラムが中止され、システムは起動しません。
- ・ 正常にリカバリされなかったデータセットは、WIJ に、リカバリが必要だと記録されたままになります。

## 3.2.2 WIJ ブロック比較

通常、実行中の InterSystems IRIS インスタンスがデータベースへのアクティブな書き込みを行うのはわずかな時間のみです。したがって、ほとんどのクラッシュでは、WIJ に最後に書き込まれたブロックは、クラッシュの前にデータベースに確実に書き込まれたことが確認されます。WIJ はアクティブとマークされず、実行される WIJ のリストアはありません。しか

し、このようなクラッシュの後に InterSystems IRIS が起動すると、最新の WIJ 更新が行われたブロックは、影響を受けたデータベースの対応するブロックと高速整合性チェックの形式で比較され、ストレージ・サブシステムの障害を伴ったクラッシュの後に不明な状態でインスタンスを開始しないようにします。この比較は、可用性への影響を回避するために短時間で実行され、スループットを最大化するために非同期入出力が利用されます。すべてのブロックが一致するまたは 10 秒以内に不一致が検出されない場合、通常どおりに起動が継続されます。この時間内に不一致が見つかった場合、以下ようになります。

- ・ 使用可能なすべての WIJ ブロックの比較が完了するまで比較動作が継続されます。
- ・ 一致しなかった WIJ ブロックは、WIJ ディレクトリにある **MISMATCH.WIJ** というファイルに書き込まれます。
- ・ 通常の起動が中止され、InterSystems IRIS はシングル・ユーザ・モードで起動し、以下のようなメッセージが表示されます。

```
There exists a MISMATCH.WIJ file.
Startup aborted, entering single user mode.
Enter IRIS with
    iris terminal [instancename] -B
and D ^STURECOV for help recovering from this error.
```

**注意** **MISMATCH.WIJ** が見つかった場合は、先に進む前に[インターシステムズのサポート窓口](#)までお問い合わせください。

この状況はデータの整合性に影響を与えるため、早急な対応が必要です。ただし、**MISMATCH.WIJ** に対して間違ったアクションを実行すると、状況が悪化する可能性があります。**MISMATCH.WIJ** の使用経験がある場合を除き、既知の良好なバックアップに戻して、そこから継続するか、WRC に問い合わせる方が安全です。

後に続く情報を利用して、適切な処置を判断してください。リカバリ手順が完了したら、InterSystems IRIS の起動を継続する前に、STURECOV ルーチンを使用するかまたは外部から、**MISMATCH.WIJ** ファイルの名前を変更する必要があります。このファイルは永続的で、存在するとインスタンスを通常どおり起動できません。

指示されたコマンドを実行し、システム管理者として緊急ログインを行います（“システム管理ガイド”の“ライセンス”の章にある“[管理者ターミナル・セッション](#)”を参照してください）。

これで管理者のネームスペースに入ることができたので、Do ^STURECOV コマンドを使用してスタートアップ・リカバリ・ルーチンを実行できます。UNIX®/Linux システムでは、次の WIJ 不一致リカバリ・メッセージおよびメニューが表示されます。

```
The system crashed and some database blocks do not match what was
expected based on the contents of write image journal (the WIJ).
The WIJ blocks have been placed in the MISMATCH.WIJ file. If any
database files, or the WIJ, were modified or replaced since the crash,
you should rename the MISMATCH.WIJ. Otherwise, MISMATCH.WIJ probably
contains blocks that were lost due to a disk problem. You can view
those blocks and apply them if necessary. When finished, rename the
MISMATCH.WIJ in order to continue startup.
```

- 1) List Affected Databases and View Blocks
- 2) Apply mismatched blocks from WIJ to databases
- 3) Rename MISMATCH.WIJ
- 4) Dismount a database
- 5) Mount a database
- 6) Database Repair Utility
- 7) Check Database Integrity
- 8) Bring up the system in multi-user mode
- 9) Display instructions on how to shut down the system

```
-----
H) Display Help
E) Exit this utility
-----
```

Windows システムでは、オプション 8 および 9 が 8) Bring down the system prior to a normal startup に置き換えられます。



オプション `Rename MISMATCH.WIJ` は日付を追加することでファイル名を変更します。既にその名前で変更された `MISMATCH.WIJ` が存在する場合は、数値 (`_1` など) が追加されます。

WIJ 不一致発生時の適切なアクションは、企業のニーズおよびポリシーに基づいて異なりますが、データ整合性の問題を示唆するイベントに対応する際にユーザのサイトにおいて行う既存の慣行とほぼ同じです。考慮事項には、リスクに対する許容範囲、影響を受けるデータベースの重要度、稼働時間の要件、疑われる原因などがあります。

次に、WIJ ブロック比較プロセスに固有の考慮事項および推奨事項を示します。

- ・ クラッシュの後かつリカバリの前にデータベースまたは WIJ ファイルを置換、リストア、または変更すると、不一致が発生する可能性があります。この不一致は、WIJ 比較時に見つかり、`MISMATCH.WIJ` ファイルに記録されます。これが発生した場合は、`MISMATCH.WIJ` の名前を変更してください。

**注釈** クラッシュの後にデータベースをリストアする場合、このリストアを実行する前に必ず WIJ およびジャーナルのリカバリを行わない状態でインスタンスを開始してください (このドキュメントの“バックアップとリストア”の章の“[自動 WIJ およびジャーナル・リカバリを使用しない InterSystems IRIS の起動](#)”を参照)。こうすることによって、WIJ 比較で検出されることになる不一致の作成と WIJ ブロックまたはジャーナル・データ (このドキュメントの“[ジャーナリング](#)”の章を参照) の意図しないデータベースのバージョンへの誤った適用の両方が回避されます。

- ・ 一部のストレージ・サブシステム (特に、ラップトップおよびワークステーションのローカル・ドライブ) は、バッテリーまたは非揮発性のメモリでバックアップされていない安全でない形式のライト・バック・キャッシュを使用しています。これは、InterSystems IRIS が実行する 2 フェーズ・ライト・プロトコルを無効にし、ハードウェアのクラッシュまたは電源の喪失の後の破損 (WIJ 比較時に検出される) をもたらす可能性があります。このことがユーザのシステムに当てはまる場合、`MISMATCH.WIJ` にはより新しいデータが格納されている可能性が高く、したがって、データベースに安全に適用できます (多くの注意を必要としないシステムであることが前提)。
- ・ WIJ 比較の後にデータベースを変更した場合、`MISMATCH.WIJ` は有効ではなくなり、WIJ ブロックの適用は安全ではなくなります。
- ・ 全社クラスのストレージ、または安全でない形式のライト・バック・キャッシュを使用していないストレージ・サブシステムを備えたサーバの場合、WIJ 比較時に見つかった不一致は常に予期しないものであり、より深刻で広範囲に及ぶ問題の兆候である場合があるため、重視する必要があります。
- ・ 問題の原因によっては、データベースには問題がなく、破損しているのは WIJ である場合があります。影響を受けたデータベースの整合性チェックによってこの可能性を判断できます。
- ・ WIJ 比較は最後に書き込まれたブロックのみを対象とするため、追加のブロックに影響する問題がある可能性があり、これは完全な整合性チェックで検出できます (このガイドの“データ整合性の概要”の章にある“[構造的な整合性の検証](#)”を参照してください)。
- ・ データベースが小さく、時間が許すのであれば、次のような手順に従うことによって最大限の安全性を確保できます。
  1. データベースの完全な整合性チェックを実行します。
  2. 破損がない場合、`MISMATCH.WIJ` の名前を変更して起動します。
  3. 1 つまたは複数のデータベースが破損している場合、すべてのデータベースの `IRIS.DAT` ファイルをコピーし、`MISMATCH.WIJ` からすべてのブロックを適用し、完全な整合性チェックを再実行します。
  4. `MISMATCH.WIJ` を適用した後に破損しているデータベースがある場合、そのデータベースを前のコピーに戻すか、以前のバックアップからリストアできます。
- ・ 暗号化データベースは、WIJ ブロック比較から除外されます。

## 3.3 ライト・イメージ・ジャーナリングの限界

2 フェーズ・ライト・プロトコルは、構造的なデータベースの整合性を保護しますが、データの損失は保護しません。WIJ に完全な更新を書き込む前にシステム障害が発生した場合、InterSystems IRIS は、ディスクを完全に更新するために必要な情報が不足している状態になるため、そのデータは失われます。ただし、ジャーナル・ファイルに書き込まれたデータは、この章の“[リカバリ](#)”の説明に従ってリカバリされます。

また、以下の場合、ライト・イメージ・ジャーナリングは、データベースの劣化を防ぐことはできません。

- ・ メモリまたはストレージの破損によるハードウェア障害。
- ・ メモリ、ファイル・システム、またはストレージの破損によるオペレーティング・システム障害。
- ・ WIJ が削除された状態。
- ・ ライト・バック・キャッシュの内容の消失。停電の場合には、ライト・バック・キャッシュが消失し、データベースの劣化が生じる可能性があります。このような劣化を防ぐには、ストレージレイでライト・バック・キャッシュに非揮発性のメモリを使用するか、または揮発性のライト・バック・キャッシュにバッテリー・バックアップを使用するようにしてください。

上記のいずれかの状況が発生したと思われる場合には、[インターシステムズのサポート窓口](#)までお問い合わせください。

# 4

## バックアップとリストア

ここでは、システムの確かなバックアップ計画を立てる際に考慮すべき要素について説明します。バックアップの整合性と復元可能性を確保する方法および推奨バックアップ方法について説明します。さらに、ネイティブ・ユーティリティまたはサードパーティ・ユーティリティを使用してこれらの作業を実行する手順についても詳しく説明します。

バックアップ方法は、オペレーティング・システム、使用するバックアップ・ユーティリティ、ディスク構成、バックアップ・デバイスなどによって異なる場合があります。ご使用の環境に適したバックアップ方法の開発や現在のバックアップ方法の見直しのために詳細な情報が必要な場合は、[インターシステムズのサポート窓口](#)までお問い合わせください。

### 4.1 バックアップの整合性と復元可能性

バックアップ方法に関係なく、定期的にバックアップをリストアして、ご使用の手順を検証する必要があります。最善の方法は、運用環境のあらゆるバックアップを代替サーバにリストアしてから、リストアしたデータベースの整合性を検証することです（詳細は、「[構造的な整合性の検証](#)」を参照）。これには以下のメリットがあります。

- ・ バックアップ・メディアの復元可能性を検証します。
- ・ バックアップ内のデータベースの物理的な整合性を検証し、実行中の更新の影響を受けている揮発性データベースで整合性チェックを実行したときに整合性エラーが誤って報告されるという問題を回避します。
- ・ バックアップのウォーム・コピーを提供し、災害時のバックアップのリストアに必要な時間を大幅に短縮します。そのような事態が発生した場合でも、必要な処理はジャーナル・ファイルの更新のリストアのみです。
- ・ 最後の良好なバックアップを構築します。

このドキュメントで説明するバックアップ方法では、データベースの物理構造が保持されます。したがって、リストアされたコピーの整合性チェックを完全に行い、それに問題がなければ、バックアップ時点のプロダクション・データベースの整合性にも問題がなかったことを意味します。ただし、その逆は True ではありません。リストアされたデータベースのコピーに整合性のエラーが検出されても、それは必ずしもプロダクション・データベースの整合性に問題があったことを意味しません。例えば、バックアップ・メディアにエラーが発生した可能性もあります。リストアされたデータベースで整合性エラーが検出された場合は、即座に実稼働データベースで整合性チェックを行い、プロダクション・システムの整合性を検証してください。

リストアされたデータベースでアプリケーションが適切に動作することを検証するには、アプリケーション・レベルでチェックを実行することも可能です。これらのチェックを実行するには、ジャーナル・ファイルをリストアしてトランザクションの整合性をリストアすることが必要な場合もあります。詳細は「[ジャーナルの重要性](#)」を参照してください。

バックアップをリストアし、それからリカバリが可能であることを確認したら、次の良好なバックアップが作成されるまでそのコピーを保持しておくことをお勧めします。したがって、バックアップの検証に使用するサーバには、最新の良好なバックアップと現在検証中のバックアップを格納するために、プロダクション領域の 2 倍のストレージ領域があることが理想的

です(ニーズによっては、バックアップのリストアに使用するストレージ・デバイスの性能の要件は厳しくなく、ストレージ・ソリューションがより安価であっても問題がない場合があります)。このようにすると、現在のバックアップが失敗しても、災害時に最新の良好なバックアップがすぐに使用できる状態にあります。データ・センタを破壊する可能性がある災害から企業のデータベースを保護するには、定期的にバックアップ・メディアを安全なオフサイトの場所に送ります。

## 4.2 ジャーナル・リストアの重要性

InterSystems IRIS® データベースのリストア可能なバックアップだけでは、プロダクション・データのリストアには十分ではありません。障害が発生し、バックアップからリストアする必要性が生じた場合は、リストアしたデータベースのコピーにジャーナル・ファイル (“[ジャーナリング](#)” を参照) を適用することも必要です。このジャーナル・リストアでは以下のことが可能です。

- ・ バックアップの時点から障害発生時までの間にジャーナルに記述された更新をすべてリストアします。
- ・ コミットされていないトランザクションをロールバックして、データベースのトランザクションの整合性をリストアします。

このトピックで説明するバックアップ・アプローチでも、個々にリストアできるデータベースの物理的に整合性が保たれたコピーを提供できますが、バックアップ時点で開いていたトランザクションを確実にロールバックするには、バックアップ時点より新しいジャーナルがなくても、ジャーナル・リストアも実行する必要があります。これは、使用するバックアップ・アプローチに各データベースのジャーナル・ファイルが含まれている必要があることを意味します。

システム全体のクラッシュ整合性スナップショット・イメージをリストアすると(仮想環境で一般的)、InterSystems IRIS の通常の起動リカバリによって、ライト・イメージ・ジャーナル (WIJ、"[ライト・イメージ・ジャーナリング](#)" を参照) による物理的な整合性(中断された書き込みの完了を含む)とジャーナリングによる論理的な整合性(トランザクションのロールバックを含む)の両方が自動的に確保されます。ただし、これは、データベース、インスタンスのインストール・ディレクトリ、WIJ、ジャーナル・ファイルに加えて、データベースに関連付けられたその他の外部ファイルなど、ある瞬間のすべてのコンポーネントがデータベースのこのようなスナップショット・バックアップ・イメージに含まれている必要があることを意味します。

リカバリのためにジャーナルの可用性を確保することに関する重要な情報については、“[ジャーナリングの最善の使用方法](#)” を必ず参照してください。特に、そのセクションで説明されるように、パフォーマンスと復元可能性を高めるために、プライマリおよび代替ジャーナル・ディレクトリは、それぞれ異なるストレージ・デバイスに配置し、データベースや WIJ が使用しているストレージ・デバイスとも異なるストレージ・デバイスに配置することをお勧めします。

**重要**                      定期的にディザスタ・リカバリの方法全体を最初から最後までテストする必要があります。これには、バックアップのリストア、ジャーナルのリストア、およびリストアされた環境でのユーザの操作のシミュレーションが含まれます。

## 4.3 バックアップの方法

データベースをバックアップする最適な方法は、外部バックアップとオンライン・バックアップです。このセクションでは、この 2 つの方法と、目的が特殊なその他の方法について説明します。

- ・ [外部バックアップ](#)
- ・ [オンライン・バックアップ](#)
- ・ [コールド・バックアップ](#)
- ・ [並行外部バックアップ](#)

### 4.3.1 外部バックアップ

外部バックアップは、現時点でデータベース全体のバックアップに推奨される最適な方法です。外部バックアップは、既存のシステム・バックアップ手順に簡単に統合でき、通常はダウンタイムなしでバックアップできることを見越しておけます。外部バックアップは主に、論理ディスク・ボリュームの有効な“スナップショット”を迅速に作成するテクノロジーと共に使用します。このようなテクノロジーは、ストレージアレイからオペレーティング・システム、ディスクの単純なミラーリングに至るまで、さまざまなレベルで存在します。このセクションでは、スナップショット・テクノロジーが使用できないシステムに関する特別な考慮事項についても説明します。

スナップショットの整合性を確保するために、InterSystems IRIS には、スナップショットの作成中にデータベースへの書き込みをフリーズする方法があります。スナップショットの作成中は、データベース・ファイルへの物理的な書き込みのみがフリーズされ、ユーザ・プロセスによるメモリ内の更新は引き続き実行できます。このスナップショットは通常、システムで使用中のファイル・システムすべてのスナップショットになります。このスナップショットには、少なくとも InterSystems IRIS データベースが使用するすべてのディレクトリが何らかの形で収録されています。このディレクトリには、インストール・ディレクトリ、データベース・ディレクトリ、ジャーナル・ディレクトリおよび代替ジャーナル・ディレクトリ、WIJ ディレクトリ、システムで使用する外部ファイルを格納したあらゆるディレクトリなどがあります。書き込みができるようになったら、スナップショットをバックアップすることができます。その後、プロダクションに再結合するか、またはウォーム・バックアップとしてオンラインのままにしておくことができます（使用されている具体的なテクノロジーによって異なる）。

**注釈** 使用中のデータベースの整合性チェック実行時にエラーが誤って報告されるという問題を回避するために、ファイル・システムのスナップショットが取られた直後にデータベースがチェックされるように、整合性チェックをこのセクションで説明したような手順に統合することができます。（データベースの整合性のチェックの詳細は、“[構造的な整合性の検証](#)”を参照してください。）

以下の表に、外部バックアップ方法のメリットとデメリットを示します。

メリット	デメリット
ほとんどのシステムで、ユーザの操作が停止されことなく、ダウンタイムなしのバックアップが可能です。	書き込みのフリーズが 10 分を超える場合に、引き続きユーザに実行を許可するには、特別な考慮が必要になります。
既存のバックアップ手順と簡単に統合できます。	

データベースをフリーズするクラス・メソッドは `Backup.General.ExternalFreeze()` で、フリーズを解除するクラス・メソッドは `Backup.General.ExternalThaw()` です。書き込みの停止に加えて、フリーズする方法では、ジャーナル・ファイルの切り替えやジャーナルへのバックアップ・マーカの書き込みも処理されます。データベースの物理的な書き込みがフリーズされている間も、ジャーナル・ファイルには引き続き正常に書き込まれます。データベースの物理的な書き込みがフリーズしている間にシステムがクラッシュした場合、通常どおり起動時にジャーナルからデータがリカバリされます。

外部バックアップを設定するには、以下の手順を実行するスクリプトを記述します。

1. `Backup.General.ExternalFreeze()` メソッドを使用して、データベースへの書き込みをフリーズします。このメソッドをさまざまなプラットフォームで使用する例が、クラス・ドキュメントに含まれています。

**注釈** セキュリティ構成で、バックアップ・スクリプトがデータベース資格情報を提供するように求められる場合、必要な資格情報が含まれているファイルから入力をリダイレクトすることにより、これを実行できます。あるいは、OS レベルの認証を有効にし、スクリプトを実行している OS ユーザの InterSystems IRIS アカウントを作成できます。

2. 外部のスナップショット・ユーティリティを使用して、ファイル・システムのスナップショットを作成します。
3. `Backup.General.ExternalThaw()` メソッドを使用して、データベース書き込みを再開します。



**注釈** Backup.General.ExternalThaw() メソッドでライト・デーモンが再開されず、インスタンスのフリーズが解除されない場合、**-W オプションを指定して irisstat コマンド**を発行することで、これを行うことができます (このオプションでは、バックアップ以外が原因で発生したハングまたは一時停止の場合、ライト・デーモンのフリーズは解除されません)。

4. バックアップ・メディアにスナップショットをコピーします。
5. バックアップが完了したら、Backup.General.ExternalSetHistory() メソッドを使用して、バックアップをバックアップ履歴に記録できます。バックアップを記録することで、[ジャーナル設定] ページの **[ジャーナルファイルを削除するタイミング]→[この回数のバックアップ成功後]** の設定 (既定値は 2) によっては、ジャーナルの削除がトリガされる可能性があることに注意してください。詳細は、“**ジャーナル設定の構成**”を参照してください。

これらのメソッドのプラットフォーム固有の例については、“**Backup.General**”を参照してください。

スナップショット・テクノロジーが使用できないシステムの場合、上記のスナップショットの作成をファイル・システムのコピーに置き換えることにより、外部バックアップ・アプローチで低速のファイル・システム・コピーを使用できます。これは以下のいずれかの方法で実行するか、または必要に応じて**オンライン・バックアップ**を代替手段として使用することができます。

- ・ **ダウンタイムの追放** :Backup.General.ExternalFreeze() を呼び出すときに ExternalFreezeTimeOut パラメータの値を指定し、フリーズ中でもデータベースの更新をメモリのバッファに格納できるように、十分なデータベース・キャッシュを構成しておきます。この場合、ユーザは物理的な書き込みがフリーズしている間に、最大で ExternalFreezeTimeOut に指定した値まで操作を続行できます。ジャーナリングは、書き込みがフリーズされている間にシステム・クラッシュが発生した場合に、データが失われないようにするうえで重要です。クラッシュが発生した場合は、通常よりもシステムの起動に時間がかかる可能性があります。**[エラー発生時に凍結する]** に対するジャーナルの設定は、**[はい]** に設定する必要があります (詳細は、“**ジャーナル入出力エラー**”を参照)。
- ・ **ユーザ操作の停止** :フリーズ時間として想定する時間の既定値を 10 分にします。この時間が経過すると、Backup.General.ExternalThaw() が呼び出されるまで、ユーザの操作は一時停止状態になります。

**重要** バックアップされているインスタンスがミラーのプライマリ・フェイルオーバー・メンバである場合 (“**ミラーリング**”を参照)、外部フリーズが Backup.General.ExternalFreeze() の ExternalFreezeTimeOut パラメータの指定よりも長く更新を中断しないようにする必要があります。指定時間より長い中断が起こると、ミラーがバックアップ・フェイルオーバー・メンバにフェイルオーバーし、その結果、進行中のバックアップ操作が終了されてしまいます。

**注釈** InterSystems IRIS は、Windows のボリューム・シャドウ・コピー・サービス (VSS) をサポートしており、その際にはデータベースに代わってライタとして動作します。VSS シャドウに含まれる InterSystems IRIS データベースのコピーには、トランザクション上の論理的な整合性はありませんが、物理的な整合性があるため、個別にリストアできます。リストアしたデータベースのトランザクション上の整合性を確保するには、ジャーナル・ファイルもリストアする必要があります。VSS シャドウの作成時にマウントされたデータベースのみが VSS シャドウに含まれます。

InterSystems IRIS 用 VSS ライタは、管理者のみが起動できます。

Windows システムでは、iris.cpf ファイルの **EnableVSSBackup** パラメータは既定で 1 (有効) に設定されています。InterSystems IRIS の起動時に、“InterSystems IRIS VSS ライタが起動しました” というメッセージがメッセージ・ログに書き込まれます。VSS のシャドウ・コピーを作成すると、InterSystems IRIS は、メッセージ・ログのメッセージに示されているように、Backup.General.ExternalFreeze() および Backup.General.ExternalThaw() を自動的に呼び出します。

## 重要

VSS を使用する場合は、vSphere のスナップショット・オプション [Quiesce guest file system] を使用するようにします。このオプションは VSS コールバックを呼び出して、スナップショットの前にデータベースをフリーズし、スナップショットの後にデータベースのフリーズを解除します。メッセージ・ログには、VSS Writer: OnFreeze と VSS Writer: OnThaw が表示されます。

一方、このオプションを使用しない場合、vSphere はメモリ・スナップショットのみを実行し、メッセージ・ログにこれらのメッセージは含まれません。

### 4.3.2 オンライン・バックアップ

InterSystems IRIS には、プロダクション・システムのユーザにダウンタイムがほとんど、またはまったく発生しないよう設計された、専用のバックアップ・メカニズムが実装されています。オンライン・バックアップでは、IRIS.DAT ファイルのデータのみがバックアップされ、データベース内でデータに割り当てられているブロックがすべて取得されます。シーケンシャル・ファイルに出力されます。このバックアップ方法は、システム・バックアップと関係させて、オンライン・バックアップの出力ファイルを、他のファイルと共にバックアップ・メディアにコピーする必要があります。このシステム・バックアップには、システムで使用されているすべてのファイル・システム (IRIS.DAT ファイルは除く) を含める必要があります。少なくとも、インストール・ディレクトリ、ジャーナル・ディレクトリと代替ジャーナル・ディレクトリ、Web ゲートウェイ・ファイル、および InterSystems IRIS で使用される外部ファイルを格納したあらゆるディレクトリを含める必要があります。IRIS.DAT ファイルは含めないでください。

オンライン・バックアップ・プロシージャでは、複数のパスを使用してデータがコピーされます。前のパスにある間に変更されたデータ・ブロックのリストがコピーされますが、連続的なパスを経るにつれ、通常、このリストははだいに小さくなります。一般に、バックアップを完了するには 3 つのパスで十分です。最終パスの期間全体と、先行する各パスにある間の短期間、データベースへの書き込みが停止されます。バックアップの際にデータベースへの物理的な書き込みが一時停止している間も、ユーザ・プロセスによるメモリ内の更新は引き続き実行できます。

今日の外部バックアップ・オプションの機能を考えると、通常は、オンライン・バックアップよりもニーズに合った外部バックアップ・アプローチを見つけることができます。以下の表に、オンライン・バックアップ方法のメリットとデメリットを示します。

メリット	デメリット
ほとんどのシステムで、ダウンタイムなしのバックアップが可能です。	データベースのみをバックアップします。
定期的な累積バックアップやインクリメンタル・バックアップをサポートします。	バックアップとリストアは比較的低速です。
エンタープライズクラス・ストレージは不要です。	1 つのデータベースをリストアするには、バックアップ・ファイル全体を処理する必要があります。
	リストアを実行するには、InterSystems IRIS のインスタンスを実行しておく必要があります。
	データ量が大きすぎてリストアしにくい場合があります。
	暗号化データベースのバックアップは暗号化されません。
	最新の外部バックアップ・テクノロジーでよく見られる一部の機能が含まれていない可能性があります。

オンライン・バックアップには以下の種類があります。これらを組み合わせることによって、バックアップ出力のサイズとバックアップからの回復に要する時間のバランスを調整できます。

- ・ フル・バックアップ – 使用中のすべてのブロックのイメージをバックアップ・メディアに書き込みます。
- ・ 累積バックアップ – 最後のフル・バックアップ以降に変更されたすべてのブロックを書き込みます。以前のフル・バックアップと共に使用する必要があります。
- ・ インクリメンタル・バックアップ – タイプに関係なく最後に行われたバックアップ以降に変更されたすべてのブロックを書き込みます。以前のフル・バックアップと共に使用する必要があります。また、必要に応じてそのフル・バックアップ以降に実行した累積バックアップあるいはインクリメンタル・バックアップも使用します。

オンライン・バックアップを使用する場合は、最初にフル・バックアップを実行する必要があります。累積バックアップおよびインクリメンタル・バックアップ（またはこのいずれか）は、その後で実行できます。

オンライン・バックアップでは、すべてのデータベース・ブロックがインターリーブ方式で 1 つのファイルに書き込まれます。極端に大きいデータをオンライン・バックアップでバックアップした場合、リストアは煩雑になる可能性があります。このような場合は、バックアップ方法の計画時に対応を検討します。

リストア検証プロセスでは、リストアされたデータベースのコピーがオンラインで提供されるため、制約の解消に役立ちます。このバックアップ検証方法は、インクリメンタル・バックアップや累積バックアップにも使用されます。インクリメンタル・バックアップまたは累積バックアップは、それぞれ実行後に直ちに代替サーバにリストアできます。例えば、フル・バックアップを毎週行い、インクリメンタル・バックアップを毎日行う計画を立てたとします。毎日のバックアップそれぞれには、その日に変更されたブロックのみが含まれているので、問題は発生しません。この方法を使用すると、毎日そのインクリメンタル・バックアップを代替サーバにリストアし、リストアされたデータベースの整合性を確認する必要があります。

現在検証中のバックアップをリストアする際は、健全性が確認された最新のバックアップのウォーム・コピーを上書きしないようにする必要があります。また、既存のリストアされたデータベースにインクリメンタル・バックアップをリストアする場合にもこれに注意する必要があります。バックアップが最新で健全性にも問題がないと判定された時点から、それに次の日のインクリメンタル・バックアップまたは累積バックアップを適用するまでの間は、それ以降のインクリメンタル・リストアに失敗した場合に備えて、コピーを保存し、最新の健全なバックアップを常にオンラインで使用できるようにしておく必要があります。リストアしたバックアップで整合性の問題が検出された場合は、そのバックアップを破棄しなければいけません。以降のインクリメンタル・リストアのターゲットに使用することはできません。

オンライン・バックアップからシステムをリストアする場合は、最初に最新のフル・バックアップを、次に最新の累積バックアップを、その後で、その累積バックアップ以降に行ったすべてのインクリメンタル・バックアップをリストアします。

オンライン・バックアップの構成の詳細は、“[オンライン・バックアップ設定の構成](#)”を参照してください。オンライン・バックアップの実行の詳細は、“[オンライン・バックアップの管理](#)”を参照してください。

### 4.3.3 コールド・バックアップ

コールド・バックアップとは、データベースが正常にシャットダウンされたときに行われる、システムの外部バックアップのことです。このバックアップは通常、システムで使用されているファイル・システムすべてのバックアップです。このスナップショットには、少なくとも InterSystems IRIS が使用するすべてのディレクトリが何らかの形で収録されています。このディレクトリには、インストール・ディレクトリ、データベース・ディレクトリ、ジャーナル・ディレクトリおよび代替ジャーナル・ディレクトリ、WIJ ディレクトリ、データベースで使用する外部ファイルを格納したあらゆるディレクトリなどがあります。この方法は、InterSystems IRIS が通常夜間に停止されるシステムに適しています。

以下の表に、コールド・バックアップ方法のメリットとデメリットを示します。

メリット	デメリット
手順が簡単です（停止してインスタンス全体をコピー）。	データベースは使用できません。
場合によっては、システムがシャットダウンしているときにバックアップを実行すると役に立ちます。	



### 4.3.4 並行外部バックアップ

並行外部バックアップ方法（“ダーティ・バックアップ”とも呼ばれる）は、ダウンタイムなしのバックアップが必要だが、導入の状況により、標準のダウンタイムなしの外部バックアップも、ダウンタイムなしのオンライン・バックアップも不可能な場合に使用できます。この方法の一般的なシナリオとしては、例えば、スナップショット・テクノロジーが使用できないうえ、データベースがきわめて大規模なため、以下の両方が当てはまる状況が考えられます。

- ・ **外部バックアップ**を使用すると、（データベースのコピー中に）ユーザに影響が及ぶほど長い間書き込みがフリーズする。
- ・ **オンライン・バックアップ**の使用は実用的ではない。

並行外部バックアップ方法は、通常のデータベースの書き込みが行われている間にシステムの外部バックアップを実行できるようにすることによって機能します。この方法で取得したデータベースのコピーは、破損している（“ダーティ”である）と予想されます。このコピーの後にインクリメンタル・オンライン・バックアップを行います。これにより、コピーが行われている間に変更されたすべてのデータベース・ブロックが取得されます。バックアップのリストア時には、まずデータベースのコピーをリストアし、続いてインクリメンタル・バックアップをリストアします。詳細な手順は以下のとおりです。

1. 最新のバックアップ以降に変更されたデータベース・ブロックのリストを削除します。

#### ObjectScript

```
Do CLRINC^DBACK("QUIET")
```

詳細は、“[CLRINC^DBACK](#)”を参照してください。

2. 選択したオペレーティング・システム、またはサードパーティのバックアップ・ユーティリティを使用して、**IRIS.DAT** データベース・ファイルをコピーします。
3. 外部バックアップ・ユーティリティを使用したことを示す E パラメータを使用して、**\$\$BACKUP^DBACK** を呼び出します。以下に例を示します。

#### ObjectScript

```
Set x=$$BACKUP^DBACK("", "E", "Dirty external backup - incrementals must be applied.", "", "", "")
```

詳細は、“[BACKUP^DBACK](#)”を参照してください。

4. インクリメンタル・バックアップを実行します。これによって、**IRIS.DAT** ファイルのコピー中に変更されたブロックがすべてコピーされます。構成によっては、このときにごく短時間ですがユーザ・プロセスが一時停止する場合があります。

#### ObjectScript

```
Set x=$$BACKUP^DBACK("", "I", "Nightly", "test.bck", "N", "bck.log", "QUIET", "N", "Y")
```

詳細は、“[BACKUP^DBACK](#)”を参照してください。

5. “[システム全体のリストア](#)”の“[オンライン・バックアップまたは並行外部バックアップからのシステム全体のリストア](#)”の手順に従ってバックアップをリストアします。

以下の表に、並行外部バックアップ方法のメリットとデメリットを示します。

メリット	デメリット
他の推奨アプローチのいずれも使用できないサイトににおいて、停止時間がほとんど、あるいはまったく発生せずにバックアップ可能な方法です。	複数のファイルをリストアする必要があります（IRIS.DAT データベース・ファイルとインクリメンタル・バックアップ・ファイル）。このため、リストア処理に長時間を要します。

メリット	デメリット
	手順が複雑です。
	リストアを実行するには、InterSystems IRIS のインスタンスを実行しておく必要があります。

## 4.4 バックアップからのリストア

定期的にバックアップをリストアして、ご使用の手順をテストする必要があります。“[バックアップの整合性と復元可能性](#)”の説明に従って、検証時にはあらゆるバックアップを代替サーバにリストアし、リストアのウォーム・コピーとして機能するようにすることをお勧めします。

バックアップ・リストアは、複雑かつ時間がかかる場合がありますが、定期的にリストアを実施しておくことで、関連するリスクの軽減に役立ちます。バックアップ・リストアは、大規模な災害復旧計画の一環として検討する必要があります。災害復旧の代替メカニズムとして、[ミラーリング](#)も使用できます。

バックアップ・リストアのシナリオの詳細は、[インターシステムズのサポート窓口](#)までお問い合わせください。

このセクションでは、以下のトピックについて説明します。

- ・ [バックアップ・リストアのシナリオ](#)
- ・ [メンテナンス・モードによる InterSystems IRIS の起動](#)
- ・ [バックアップ・リストア後のジャーナル・リストア](#)
- ・ [自動 WIJ およびジャーナル・リカバリを使用しない InterSystems IRIS の起動](#)

### 4.4.1 バックアップ・リストアのシナリオ

バックアップ・リストアを実行するには、さまざまなシナリオがあります。多くの場合、一連の固有の要件やリストア手順があります。このセクションでは、各種バックアップ方法のバリエーションを含め、以下の一般的なリストア・シナリオについて説明します。

- ・ [システム全体のリストア](#) – InterSystems IRIS またはシステムで使用中のストレージのすべてまたは大部分が、全体として使用不可能になった場合に使用します。バックアップ・メディアからシステム全体のバックアップがリストア済みで、これから実働環境で使用できる状態まで InterSystems IRIS をリカバリする必要があります。
- ・ [データベースのみのリストア](#) – InterSystems IRIS データベースの一部またはすべてが使用不可能になり、バックアップからリカバリする必要がある場合に使用します。
- ・ [別のシステムへのデータベースのリストア/移行](#) – 1 つ以上のデータベースが別のシステムにリストアされる場合に使用します。ターゲット・システムでは、新しいハードウェアや異なるプラットフォームを使用することも、同一ハードウェア上にデータベースの別のインスタンスを稼働させることも可能です。ターゲット・システムでは、ファイル・システムのレイアウトが異なってもかまいません。これを使用して、データベースをバックアップ・メンバまたは非同期ミラー・メンバに追加できます（“[ミラーリング](#)”を参照）。

**重要** バックアップを作成したデータベース・インスタンスと異なるインスタンスにデータベースをリストアまたは移行する場合、データの互換性がなくならないように、以下の要件を満たしている必要があります。

- ・ ターゲット・インスタンスは、ソース・インスタンスと文字幅を同一にして (8 ビットまたは Unicode。“インストール・ガイド”の“[文字幅設定](#)”を参照)、ロケールも同様に同一 (“[管理ポータル](#)の NLS 設定 ページの使用法”を参照) にする必要があります。

この要件に対する唯一の例外は ISO 8859 Latin-1 文字セットに基づいたロケールを使用する 8 ビット・インスタンスで、これは対応する幅の文字ロケールを使用する Unicode インスタンスと互換性があります。例えば、enu8 ロケールを使用する 8 ビット・インスタンスで作成されたデータベースは、enuw ロケールを使用する Unicode インスタンスで使用できます。

- ・ ソース・インスタンスとターゲット・インスタンスがエンディアン異なるシステム上に存在する場合 (“プラットフォームのエンディアン”を参照)、使用前にデータベースをターゲット・インスタンスのエンディアンに変換する必要があります。その手順については、“[エンディアン異なるターゲットへのリストア](#)”を参照してください。

**注釈** [自動開始](#)が構成されている場合に、ホスト OS の再起動時にインスタンスが自動開始しないようにしたいことがあります。これは、OS レベルでのみ行うことができ、構成に依存します。

#### 4.4.1.1 システム全体のリストア

このシナリオでは、以下の開始点を想定しています。

- ・ リストアが実行されているシステムは、バックアップを生成したシステムと同一のシステムです。
- ・ すべてのファイル・システム (あるいは、少なくともバックアップ済みのすべてのファイル) は、バックアップ・メディアからリストアされており、オペレーティング・システム・レベルでのパス名は変更されていません。
- ・ InterSystems IRIS はダウンしており、リストア以降は起動されていません。

#### 外部バックアップまたはコールド・バックアップからのシステム全体のリストア

“外部バックアップ”または“コールド・バックアップ”の手順を使用した場合、データベースは物理的な整合性が保たれた状態にリストアされますが、トランザクションの整合性をリストアするには、少なくともトランザクションのロールバックが必要になります。また、トランザクションのロールバック操作の前に、バックアップの時点以降のジャーナル・ファイルを適用することもあります。このセクションでは、必要なジャーナル・リストアとトランザクションのロールバックに使用する 2 つの推奨手順について説明します (実行したバックアップおよびリストアの詳細に応じて、どちらかの推奨手順を使用します)。

リストアされたシステムのイメージが、IRIS.WIJ ファイルとジャーナル・ファイルを含むある瞬間の InterSystems IRIS の全要素のスナップショット (クラッシュ整合性イメージ) であり、それよりも新しい適用対象ジャーナル・ファイルが存在しない場合、通常どおりに InterSystems IRIS を起動できます。InterSystems IRIS は、ジャーナル・リカバリとトランザクションのロールバックを自動的に実行します。これは、メンテナンス・モードで InterSystems IRIS を起動する場合にも当てはまります (“[メンテナンス・モードによる InterSystems IRIS の起動](#)”を参照)。

**注釈** 上記の推奨手順は、通常の外部バックアップ手順 (“外部バックアップ”および“コールド・バックアップ”を参照) が使用されていない場合でも、システム全体のクラッシュ整合性イメージがリストアされた後に適用します。その場合、リストア・イメージ内のデータベース・ファイルは、物理的な整合性が保たれていない可能性があります。ただし、そうしたデータベース・ファイルは、ジャーナル・リカバリとトランザクションのロールバックが自動実行される前に、IRIS.WIJ ファイルから自動的にリカバリされます。リカバリの詳細は、“[基本的なデータ整合性の保護](#)”とそれに続くセクションを参照してください。

リストア・イメージに、ある瞬間の InterSystems IRIS の全要素 (IRIS.WIJ ファイル、ジャーナル・ファイル、およびその他の InterSystems IRIS ファイル) が含まれていない場合や、そのときよりも新しい適用対象ジャーナル・ファイルがある場合、手動でジャーナル・ファイル・リストア (トランザクションのロールバックを含む) を実行するために、以下の手順を使用する必要があります。

## 重要

この手順では、IRIS.WIJ ファイルがバックアップ・スナップショットの一部としてバックアップされていて、そのファイルがデータベース・ファイルなどのファイルおよびファイル・システムと共にリストアされていることを前提としています。バックアップとリストアの手順を計画する際には、この方法をお勧めします。バックアップに IRIS.WIJ ファイルが含まれていない場合、“[自動 WIJ およびジャーナル・リカバリを使用しない InterSystems IRIS の起動](#)” で InterSystems IRIS の起動方法の詳細を参照してください。

1. メンテナンス・モードで InterSystems IRIS を起動して (“[メンテナンス・モードによる InterSystems IRIS の起動](#)” を参照)、ジャーナル・リストアを実行してからユーザがシステムにアクセスできるようにします。インスタンスが既に実行中の場合、まずそれを停止してから、メンテナンス・モードで再起動する必要があります。

注釈 システムがダウンしたときに[自動開始](#)が構成されている場合、インスタンスが自動開始しないようにして、メンテナンス・モードで直接インスタンスを起動できるようにしたいことがあります。これは、OS レベルでのみ行うことができ、構成に依存します。

2. ジャーナル・リストアは、“[バックアップ・リストア後のジャーナル・リストア](#)” の説明に従って実行します。
3. ジャーナル・リストアが完了すると、アプリケーションの検証やその他のメンテナンスを必要に応じて実行できます。
4. 通常どおりに InterSystems IRIS を再起動して、ユーザがシステムにアクセスできるようにします。

### オンライン・バックアップまたは並行外部バックアップからのシステム全体のリストア

このセクションの以降の部分では、[オンライン・バックアップ](#)または[並行外部バックアップ](#)のユーザを対象に、システム全体のリストアについて説明します。オンライン・バックアップ・リストアを実行して、バックアップ出力ファイルからデータベースをリストアする必要があります。オンライン・バックアップでは、“[オンライン・バックアップ](#)” で説明されているように、バックアップ出力ファイルはシステム全体のバックアップの一環としてバックアップ・メディアに書き込まれます。

オンライン・バックアップ・リストア操作は実行中のインスタンスで実行する必要がありますが、システム全体のリストアのターゲットを操作中に実行することはできません。このため、システム全体のオンライン・バックアップ・リストア操作を実行する、InterSystems IRIS の別個のセカンダリ・インスタンスをインストールして使用可能にする必要があります。リストア中のデータベースでは、セカンダリ・インスタンスにデータベース定義もネームスペース定義も不要です。

インストーラを実行して、ターゲット・インスタンス (IrisDB-1) とは異なるインスタンス名とインストール・ディレクトリを選択することで、このリストア手順の一環としてセカンダリ・インスタンス (IrisDB-2) をインストールできます。あるいは、リストア時にこの手順を回避するには、オンライン・バックアップ方法の使用を決めたときに、セカンダリ “リストア” インスタンスをインストールします。このセカンダリ・インスタンスは常に停止状態にしておき、システム全体のバックアップの一環としてインストール・ディレクトリをバックアップしておく必要があります (つまり、これはセカンダリ・インスタンスの[コールド・バックアップ](#)になります)。システム全体のリストア中に、ターゲット・インスタンスのオンライン・バックアップ・リストアを実行するために、セカンダリ・インスタンスを簡単に起動できます。

システム全体のリストア後にオンライン・バックアップまたは並行外部バックアップからリストアするには、以下の手順を実行します。

1. データベースのセカンダリ・インスタンス (IrisDB-2) を使用し、“[オンライン・バックアップ・リストア・ユーティリティ](#)” の指示に従って、以下の手順を実行します。
  - a. ターゲット・インスタンスの最新のフル・バックアップをリストアします ([並行外部バックアップ](#)を使用すると、ファイル・システムのリストア中にフル・バックアップが既に外部でリストアされています)。
  - b. フル・バックアップの後に累積バックアップを実行している場合は、最新の累積バックアップをリストアします。累積バックアップを実行していない場合は、次の手順に進んでください。
  - c. 最新の累積バックアップ (存在する場合)、または最新のフル・バックアップ (累積バックアップが存在しない場合) 以降に実行されたすべてのインクリメンタル・バックアップを、バックアップが実行された順序に従ってリストアします。
2. データベースのセカンダリ・インスタンス (IrisDB-2) を使用し、“[バックアップ・リストア後のジャーナル・リストア](#)” の指示に従います。

3. データベースのセカンダリ・インスタンス (IrisDB-2) をシャットダウンします。
4. “自動 WIJ およびジャーナル・リカバリを使用しない InterSystems IRIS の起動” の説明に従って、ターゲット・インスタンス (IrisDB-1) を起動します。この手順では、InterSystems IRIS データベースがメンテナンス・モードで実行状態を維持するので、アプリケーションの検証やその他のメンテナンスを実行できます。
5. 通常どおりにターゲット・インスタンス (IrisDB-1) を再起動して、ユーザがシステムにアクセスできるようにします。

**重要** 非同期ミラー・メンバ上のミラーリングされているデータベースで並行外部バックアップを使用した場合、ミラーの起動後にそのデータベースを有効にする必要があります。

#### 4.4.1.2 データベースのみのリストア

このシナリオでは、以下の開始点を想定しています。

- ・ リストアが実行されているシステムは、バックアップを生成したシステムであり、データベースは元の場所にリストアされます。
- ・ リストアが必要な一連のデータベースを特定しておきます。
- ・ InterSystems IRIS が起動しており、ユーザが他のデータベースに正常にアクセスしていることがあります。リストアが必要なデータベースを起動する必要がない場合があります。
- ・ InterSystems IRIS を完全に起動できない場合は、リストアが必要なデータベースを起動する必要があるか、または起動を妨げる破損が生じている可能性があります。

**注釈** 外部バックアップまたはコールド・バックアップを使用しており、リストア対象のデータベースが複数存在している場合は、InterSystems IRIS をシャットダウンしてすべての **IRIS.DAT** ファイルをリストアし、“システム全体のリストア” の説明にあるリストア手順を実行するとより簡単です。特に **IRISSYS** データベースをリストアする必要がある場合は、このケースに該当します。

以下は、データベースのみのリストアを実行する方法です。

1. データベースが起動していない場合は、“メンテナンス・モードによる InterSystems IRIS の起動” の説明に従って起動します。データベースを起動すると、既存のデータベースに対する保留中の自動リカバリがあれば直ちに行われるため、今後の起動時に、リストア対象のデータベースと競合することはありません。

既にシステムが起動しており、リストア処理中にユーザがアクセスする可能性のあるデータベースをリストアする場合は、“メンテナンス・モードによる InterSystems IRIS の起動” の説明に従って、シャットダウンおよび再起動することをお勧めします。

以下の表に、データベースを正常に起動できない場合の問題を解決する推奨手順を示します。



問題	解決策
1.InterSystems IRIS が起動に失敗する	<p>メンテナンス・モードで InterSystems IRIS を起動すると、起動に必要なデータベースがマウントできない場合でも、システムを起動することができます。ただし、起動に必要とマークされているデータベースに自動ジャーナル・リカバリが保留されている場合は、この手順はスキップされません。</p> <p>したがって、データベースが破損しているために引き続きシステムが起動できない場合には、以下の手順を実行します。</p> <ol style="list-style-type: none"> <li>IRIS.DAT ファイルの名前を変更して、データベースにいったいアクセスできないようにします。</li> <li>構成を変更して、これらのデータベースが起動に必要なとないようにマークします。</li> <li>“メンテナンス・モードによる InterSystems IRIS の起動” の説明に従って、再度システムを起動します。</li> </ol>
2.IRISSYS データベースの損傷により InterSystems IRIS が起動できない	<p>一時的に IRISSYS データベースを外部バックアップからコピーしたもの、または同一バージョンの InterSystems IRIS の新しいインストールからコピーしたものに置き換えます。これは、システムを起動してリストア手順を完了するための一時的な解決策です。</p>
3.InterSystems IRIS が引き続き起動しない	<p>“システム全体のリストア” の手順に従って、すべてのデータベースをリストアすることができます。</p> <p>詳細は、<a href="#">インターシステムズのサポート窓口</a>までお問い合わせください。</p>

2. データベースのバックアップに使用した方法に応じて、以下の手順に従い、それぞれのデータベースをリストアします。

使用したバックアップ方法	データベースのリストアのサブ手順
外部バックアップまたはコールド・バックアップ	<p>IRISSYS 以外のすべてのデータベースに以下の手順を実行します。</p> <ol style="list-style-type: none"> <li>InterSystems IRIS からデータベースをディスマウントします。</li> <li>バックアップ・メディアから元の場所に IRIS.DAT ファイルをコピーします。</li> <li>データベースを再度マウントします。</li> </ol> <p>IRISSYS データベースをリストアする場合は、一時的に代替ディレクトリにリストアします。</p>
オンライン・バックアップ	<p>“オンライン・バックアップ・リストア・ユーティリティ” の指示に従って、以下の手順を実行します。</p> <ol style="list-style-type: none"> <li>最新のフル・バックアップをリストアします。</li> <li>フル・バックアップの後に累積バックアップを実行している場合は、最新の累積バックアップをリストアします。累積バックアップを実行していない場合は、次の手順に進んでください。</li> <li>最新の累積バックアップ（存在する場合）、または最新のフル・バックアップ（累積バックアップが存在しない場合）以降に実行されたすべてのインクリメンタル・バックアップを、バックアップが実行された順序に従ってリストアします。</li> </ol>

使用したバックアップ方法	データベースのリストアのサブ手順
並行外部バックアップ	(この表の) 外部バックアップについて記述されている手順を実行して、IRIS.DAT ファイルの“ダーティ”コピーをリストアしてから、(この表の) オンライン・バックアップについて記述されているインクリメンタル・バックアップを適用します。

3. “バックアップ・リストア後のジャーナル・リストア”の指示に従います。
4. “メンテナンス・モードによる InterSystems IRIS の起動”の説明に従って InterSystems IRIS を起動し、ユーザがシステムを使用できる状態にある場合には、通常どおりに InterSystems IRIS を停止および再起動します。再起動の前に、アプリケーションの検証やその他のメンテナンスを実行できます。

注釈 この表で説明されているこの手順の手順2のように、外部バックアップまたはコールド・バックアップから一時的な代替ディレクトリに **IRISSYS** データベースをリストアしている場合は、InterSystems IRIS をシャットダウンした後で（再起動する前に）既存の **IRISSYS** データベースを一時ディレクトリの **IRIS.DAT** ファイルに置き換える必要があります。

#### 4.4.1.3 別のシステムへのデータベースのリストア/移行

このシナリオでは、以下の開始点を想定しています。

- ・ 1 つ以上のデータベースがターゲット・システムにリストアされています。このターゲット・システムは、バックアップを生成したシステムとは別のシステムです。
- ・ ターゲット・システムでは、新しいハードウェアや異なるプラットフォームを使用することも、同一ハードウェア上で単に別の InterSystems IRIS インスタンスを稼働させることも可能です。
- ・ ターゲット・システムでは、ソース・システムと異なるファイル・システムのレイアウトを使用してもかまいません。
- ・ ターゲット・システムには既に InterSystems IRIS がインストールされています (InterSystems IRIS のインストール、または **IRISSYS** データベースを、ソースとは異なるターゲット・マシンにコピーすることはできません)。

このタイプのリストアの用途には、以下のようなことが挙げられます。

- ・ ハードウェアの移行
- ・ データベースをバックアップ・メンバまたは非同期ミラー・メンバに追加する (“ミラーリング”を参照)
- ・ データベースを別のシステムにコピーして、開発、テスト、または配置用とする

別のシステムにデータベースをリストア/移行するには、以下の手順を実行します。

**重要** システム全体のリストアの一部でない限り、宛先システムの **IRISSYS** データベースをソースの **IRISSYS** データベースのリストアに置き換えないでください。ソースの **IRISSYS** データベースの必要なデータはすべて、エクスポートしてから宛先の **IRISSYS** データベースにインポートすることができます。

ソース・システムとターゲット・システムで使用しているエンディアンが異なる場合は、“[エンディアンが異なるターゲットへのリストア](#)”を参照してください。ソース・システムまたはターゲット・システムでデータベース暗号化を使用している場合は、この手順の後にある“[データベース暗号化に関する注意事項](#)”のサブセクションを参照してください。

1. まだ起動していない場合は、InterSystems IRIS を起動します。InterSystems IRIS を起動すると、既存のデータベースに対する保留中の自動リカバリがあれば直ちに行われるため、リストア対象のデータベースと競合することはありません。

リストア処理中にユーザがアクセスする可能性のあるデータベースをリストアする場合は、“[メンテナンス・モードによる InterSystems IRIS の起動](#)”の説明に従って起動する必要があります。

2. データベースのバックアップに使用した方法に応じて、以下の手順に従い、それぞれのデータベースをリストアします。

使用したバックアップ方法	データベースのリストアのサブ手順
外部バックアップまたはコールド・バックアップ	<p>データベースをリストアするには、以下の手順を実行します。</p> <ol style="list-style-type: none"> <li>リストア対象のデータベースがターゲット・インスタンスにマウントされている場合は、すべてディスマウントします。</li> <li>バックアップ・メディアから目的の場所に IRIS.DAT ファイルをコピーします。</li> <li>新しいデータベースがあればすべて構成に追加します。</li> <li>ディスマウントしたデータベースを再度マウントします。</li> </ol>
オンライン・バックアップ	<p>“オンライン・バックアップ・リストア・ユーティリティ” の指示に従って、以下の手順を実行します。</p> <ol style="list-style-type: none"> <li>最新のフル・バックアップをリストアします。</li> <li>フル・バックアップの後に累積バックアップを実行している場合は、最新の累積バックアップをリストアします。累積バックアップを実行していない場合は、次の手順に進んでください。</li> <li>最新の累積バックアップ（存在する場合）、または最新のフル・バックアップ（累積バックアップが存在しない場合）以降に実行されたすべてのインクリメンタル・バックアップを、バックアップが実行された順序に従ってリストアします。</li> </ol>
並行外部バックアップ	<p>(この表の) 外部バックアップについて記述されている手順を実行して、IRIS.DAT ファイルの“ダーティ”コピーをリストアしてから、(この表の) オンライン・バックアップについて記述されているインクリメンタル・バックアップを適用します。</p> <p><b>重要</b> リストアするデータベースがミラーリングされていて、別のシステムのデータベースである場合、インクリメンタル・バックアップを適用する前に、そのデータベースをミラーで有効にするか、ミラーから削除する（これを意図している場合）必要があります。</p>

3. “バックアップ・リストア後のジャーナル・リストア” の指示に従います。

**注釈** オンライン・バックアップからリストアして、データベースをバックアップ・メンバまたは非同期ミラー・メンバに追加（“ミラーリング”を参照）する場合は、この手順をスキップしてください。このような場合は、ジャーナルが自動的に適用されます。

4. “メンテナンス・モードによる InterSystems IRIS の起動” の説明に従ってシステムを起動し、ユーザがシステムを使用できる状態にある場合には、通常どおりにシステムを停止および再起動します。

### エンディアンが異なるターゲットへのリストア

ソース・システムと異なるエンディアン（一方がビッグ・エンディアンでもう一方がリトル・エンディアン）を使用しているターゲット・システムにバックアップをリストアするには、データベースのバックアップに使用した方法に応じて追加の手順が必要になります。サポート対象プラットフォームのエンディアンの詳細は、“プラットフォームのエンディアン”を参照してください。

- 外部バックアップまたはコールド・バックアップの場合 – “evendian を使用したビッグ・エンディアン・システムとリトル・エンディアン・システム間の変換” の説明に従って、コピーした IRIS.DAT ファイルを変換します。



- ・ [オンライン・バックアップ](#)または[並行外部バックアップ](#)の場合 – ソース・システムと異なるエンディアンを使用しているターゲット・システムに **IRIS.DAT** ファイルをリストアすることはできません。代わりに、以下の手順を実行します。
  1. このセクションに記述されている手順を使用して、ソース・システムと同じエンディアンを使用しているシステムにリストアします。
  2. InterSystems IRIS をシャットダウンするか、またはデータベースをディスマウントします。
  3. [“evendian を使用したビッグ・エンディアン・システムとリトル・エンディアン・システム間の変換”](#) の説明に従ってデータベースを変換します。
  4. ターゲット・システムに **IRIS.DAT** ファイルをコピーします。
  5. このセクションの外部バックアップ用の説明のとおりデータベースを変換し、このセクションに記述されている手順を使用してターゲット・システムにリストアします。

### データベース暗号化に関する注意事項

ターゲット・システムにデータベースをリストアする場合、リストア後のデータベースではデータベース暗号化キーの変更が必要になることがあります。データベースのバックアップに使用した方法に応じて、リストアしたデータベースのキーを以下のように変換しなければならない場合があります。

- ・ [オンライン・バックアップ](#)の場合 – オンライン・バックアップでは、データベースの内容が暗号化せずに格納されます。ターゲット・システムにある既存の暗号化データベースにオンライン・バックアップ・ファイルをリストアする場合は、オンライン・バックアップ・リストア・ユーティリティによってターゲット・システム上でファイルが動的に暗号化されます。それ以外は、リストアしたデータベースは暗号化されません。
- ・ [外部バックアップ](#)、[コールド・バックアップ](#)、または[並行外部バックアップ](#)の場合 – ソース・システムの **IRIS.DAT** ファイルの状態によっては、ターゲット・システムにファイルをリストアした後にデータベース暗号化の管理が必要になる場合があります。例えば、データベース・ファイルで使用している暗号化キーがソース・システムとターゲット・システムとで異なる場合は、ターゲット・システムでソース・システムの暗号化キーを有効にする ([“キー管理タスク”](#) を参照) か、ターゲット・システムの暗号化キーを使用するように、リストアしたデータベースを変換する ([“新しいキーを使用するように暗号化データベースを変換する”](#) を参照) 必要があります。

## 4.4.2 メンテナンス・モードによる InterSystems IRIS の起動

いずれのリストア手順でも、バックアップ・リストアおよびジャーナル・リストアが完全に終了し、対象のアプリケーション・レベルの検証がすべて完了するまで、リストア中のデータベースでアプリケーションの動作が発生しないようにする必要があります。アプリケーション環境または運用環境に、データベースの起動時にユーザの操作が発生しないようなメカニズムがある場合は、そのメカニズムを利用できます。例えば、完全に Web ベースのアプリケーションを使用している場合は、Web サーバをシャットダウンすれば、リストア手順に関与している管理者のみがシステムにアクセス可能となることを保証するには十分です。

InterSystems IRIS には、メンテナンスの実行中にアクセスを制限してシステムを起動できるようにするメカニズムが用意されています。

メンテナンス・モードで InterSystems IRIS を起動するには、以下の手順で行います。

1. すべてのメンテナンス作業時に使用する一意のユーザ名とパスワードを指定し、緊急アクセス・モードで InterSystems IRIS を起動します。このユーザ名とパスワードは、どの既存ユーザも使用していないものであることが必要です。

- ・ Windows の場合は、インストール先の **bin** ディレクトリから管理者として以下のコマンドを呼び出します。

```
iris start <db-instance-name> /EmergencyId=<username>,<password>
```

- ・ UNIX®/Linux および macOS の各プラットフォームでは、以下のようにコマンドを呼び出します。

```
iris start <db-instance-name> EmergencyId=<username>,<password>
```

詳細は、“[緊急アクセス](#)”を参照してください。

2. 使用しているリストアのシナリオの説明に従って、リストア手順の残りの部分を実行します。
3. リストア手順（あらゆるジャーナル・リストアやアプリケーションの検証などのメンテナンス作業を含む）が完了したら、以下の手順を実行して、ユーザがシステムを使用できるようにします。
  - a. データベースをシャットダウンします。
  - b. EmergencyId スイッチを指定せずにデータベースを起動します。

### 4.4.3 バックアップ・リストア後のジャーナル・リストア

バックアップ・リストアに続いて、ジャーナル・ファイルを適用する必要があります。“[ジャーナルの重要性](#)”の説明にあるように、通常、バックアップ以降のジャーナル・ファイルをリストア時に使用できるようにしておきます。これらのジャーナル・ファイルを適用すると、バックアップの時点から災害発生時までの間にジャーナルに記述された更新がすべてリストアされます。

また、コミットされていないトランザクションをロールバックして、リストアした（不完全なトランザクションが存在する可能性がある）データベースのトランザクションの整合性をリストアするには、ジャーナルを適用する必要があります。バックアップ直前までのジャーナル・ファイルが存在し、それ以降に新しいジャーナル・ファイルが作成されていない場合でもこの作業は必要です。

**重要**                    少なくともバックアップ直前までのジャーナル・ファイルが存在しない場合は、いずれのジャーナル・ファイルも適用しないでください。

リストアを開始する際に、ジャーナル・リストアの開始点を選択する必要があります。ジャーナル・リストアでは、ジャーナル・ストリームをさかのぼって、バックアップよりも前に開始されたトランザクションの開始を検索することもできます。[オンライン・バックアップ](#)を使用すると、推奨される開始点がバックアップに格納されます。[`DBACK` または `BACKUP` ユーティリティを使用してバックアップすると](#)、リストア時のトランザクション・ロールバックに必要な最も古いジャーナル・ファイルに関する情報が 3 番目または最後のパスの開始時に表示されます。

バックアップ・リストア後のジャーナルの適用方法については、“[`DBREST` によるすべてのデータベースのリストア](#)” および “[ジャーナル・ファイルのリストア](#)”を参照してください。

### 4.4.4 自動 WIJ およびジャーナル・リカバリを使用しない InterSystems IRIS の起動

このページで説明されているリストア手順の一部では、InterSystems IRIS の起動時にデータベースの自動リカバリが発生しないように起動する必要があります。これは、リストア手順でリカバリをすべて手動で実行することになっている手順で、起動時の自動リカバリとこの手動リカバリの結果に不整合が生じる可能性がある場合に当てはまります。

**注意**                    このページで説明しているリストア手順は、特に以下の手順が必要となる場合を目的としています。他の条件下でこの手順を実行することはきわめて危険であり、データベースの整合性が損なわれる可能性があります。

自動 WIJ およびジャーナル・リカバリを使用せずに InterSystems IRIS を起動するには、以下の手順で行います。

1. **IRIS.WIJ** ファイルを取り除き（ただし完全に削除はしません）、以下のように引き続き分析に使用できるようにします。
  - a. 管理者ディレクトリに格納されている **startup.last** ファイルの **wijlast=** 行を確認します。
  - b. この行で指定されているディレクトリに格納されている **IRIS.WIJ** ファイルを移動するか、名前を変更します。NULL ディレクトリが指定されている場合、**IRIS.WIJ** ファイルは管理者ディレクトリに格納されています。
2. すべてのメンテナンス作業時に使用する一意のユーザ名とパスワードを指定し、緊急アクセス・モードで InterSystems IRIS を起動します。このユーザ名とパスワードは、どの既存ユーザも使用していないものであることが必要です。

- Windows の場合は、InterSystems IRIS インストールの **bin** ディレクトリから以下のようにコマンドを呼び出します。

```
iris start <iris-instance-name> /EmergencyId=<username>,<password>
```

- UNIX®/Linux および macOS の各プラットフォームでは、以下のようにコマンドを呼び出します。

```
iris start <iris-instance-name> EmergencyId=<username>,<password>
```

詳細は、“[緊急アクセス](#)”を参照してください。

- ほとんどの場合、WIJ 内のジャーナル・リカバリ情報が失われているため、InterSystems IRIS は完全には起動しません。InterSystems IRIS が正常に起動する場合は、この手順の残りはスキップしてください。
- messages.log** の指示に従って、**-B** オプションを選択して InterSystems IRIS に入り、do ^STURECOV を実行します。
  - オプション 8 を選択して、ジャーナル・リカバリを使用しないで起動するように InterSystems IRIS をリセットします。
  - オプション 9 を選択して、InterSystems IRIS をシャットダウンします（表示される手順を使用）。
- 手順 2 の説明に従って、再度 InterSystems IRIS を緊急アクセス・モードで起動します。
- 使用しているリストアのシナリオの説明に従って、リストア手順の残りの部分を実行します。
- リストア手順（あらゆるジャーナル・リストアやアプリケーションの検証などのメンテナンス作業を含む）が完了したら、以下の手順を実行して、ユーザがシステムを使用できるようにします。
  - InterSystems IRIS をシャットダウンします。
  - EmergencyId (Windows プラットフォームの場合は /EmergencyId) スイッチを使用しないで InterSystems IRIS を起動します。

## 4.5 オンライン・バックアップ設定の構成

オンライン・バックアップは、実装可能なバックアップ方法の一つです。サポートされている方法およびその最適な使用方法については、“[バックアップの方法](#)”を参照してください。

管理ポータルの **[システム管理]**→**[構成]**→**[データベースバックアップ]** メニューの選択肢を使用して、オンライン・バックアップの設定を構成できます。これらのタスクの一部は、**Backup.General** クラスのメソッドを使用しても実行できます。適切なメソッドがある場合は、ここで紹介しています。

オンライン・バックアップが完了すると、バックアップの説明がバックアップ履歴に書き込まれます（“[バックアップ履歴の表示](#)”を参照）。これにより、**[ジャーナル設定]** ページの **[ジャーナルファイルを削除するタイミング]**→**[この回数のバックアップ成功後]** の設定（既定値は 2）によっては、ジャーナルの削除がトリガされる可能性があることに注意してください。詳細は、“[ジャーナル設定の構成](#)”を参照してください。

システム管理ポータルから、次の構成タスクを実行できます。

- [データベース・バックアップ・リストの定義](#)
- [バックアップ・タスクの構成](#)
- [バックアップ・タスクのスケジュール](#)

注釈 この構成タスクは、管理ポータル・アプリケーションの管理部分にあります。バックアップの設定を構成するには、適切なマネージャレベルのセキュリティ権限が必要です。

## 4.5.1 データベース・バックアップ・リストの定義

InterSystems IRIS では、バックアップの対象となるデータベースを指定するデータベース・リストが管理されます。このリストは、管理ポータルの [システム管理]→[構成]→[データベースバックアップ]→[データベース・バックアップ・リスト] ページに移動すると表示できます。

矢印ボタンを使用して、バックアップしない データベースを [使用可能] リストに移動し、バックアップ する データベースを [選択済み] リストに移動します。[保存] をクリックします。どのデータベースも選択していない場合は、すべてのデータベースがバックアップされます。

システムに新規データベースを追加すると、そのデータベースはデータベース・リストに自動的に追加されます。その新規データベースがバックアップ対象でない場合は、[バックアップ・データベース・リスト] から必ず削除してください。

注釈 **IRISTEMP**、**IRISLIB**、および **IRISLOCALDATA** のデータベースを除くすべてのデータベースのバックアップを実行する FullAllDatabases バックアップ・タスクでは、定義されたデータベース・リストが無視されます。

バックアップ・データベース・リストは、`BACKUP` ルーチンまたは `Backup.General.AddDatabaseToList()` メソッドおよび `Backup.General.RemoveDatabaseFromList()` メソッドを使用して管理することもできます。これらのメソッドの使用法の詳細は、“**Backup.General**” を参照してください。

## 4.5.2 バックアップ・タスクの構成

InterSystems IRIS には、複数のタイプのバックアップ・タスクがあります。各タイプは、[データベースバックアップ設定] メニューの項目として表示されます。このバックアップ・タスクの構成とは、以下のとおりです。

- すべてのデータベースのフル・バックアップの構成
- データベース・リストのフル・バックアップの構成
- データベース・リストのインクリメンタル・バックアップの構成
- データベース・リストの累積バックアップの構成

これらは、あらかじめ定義されたバックアップ・タスクで、ポータルの [システム処理]→[バックアップ] ページから必要に応じて実行できます。タスク・マネージャを使用することで、これらのバックアップ・タスクを組み合わせることもできます。詳細は、“**バックアップ・タスクのスケジュール**” を参照してください。

これらの各タスクを構成する手順は同じです。[名前]、[説明]、および [タイプ] の各フィールドは読み取り専用であり、以下のテーブルで説明するように、選択したメニュー項目のものを反映します。

テーブル 4-1: バックアップ・タスクの説明

名前	説明	タイプ
FullAllDatabases	[バックアップ・データベース・リスト] に記載されているかどうかに関係なく、共通に更新されるすべてのデータベースのフル・バックアップを実行します。	フル
FullDBList	[バックアップ・データベース・リスト] 内の InterSystems IRIS データベースのフル・バックアップを行います。	フル
IncrementalDBList	最後のフル・バックアップまたは累積バックアップ以降のデータ変更のインクリメンタル・バックアップを実行します。[バックアップ・データベース・リスト] に記載されているデータベースがバックアップされます。	インクリメンタル
CumulativeDBList	最後のフル・バックアップ以降にデータに加えられたすべての変更の累積バックアップおよびインクリメンタル・バックアップ。[バックアップ・データベース・リスト] に記載されているデータベースがバックアップされます。	累積

ディスク上のディレクトリにバックアップ出力を送信するには、[デバイス] フィールドでファイルのパス名を指定します。[参照] をクリックして、ディレクトリを選択します。

[バックアップ・データベース・リスト] の管理方法の詳細は、“データベース・バックアップ・リストの定義” を参照してください。

#### 4.5.2.1 バックアップ・ファイル名

既定の場合、バックアップ・ファイルは `install-dir¥Mgr¥Backup` に格納されます。バックアップ・ログ・ファイルは同じディレクトリに格納されます。バックアップ・ファイルの接尾語は `.cbk` です。バックアップ・ログ・ファイルの接尾語は `.log` です。バックアップ・ファイルとバックアップ・ログ・ファイルには、同じ名前付け規約を使用します。

- ・ バックアップ・タスク名の後にはアンダースコア文字 ( `_` ) が続きます。
- ・ `yyyymmdd` 形式のバックアップ日付の後には、アンダースコア文字 ( `_` ) が続きます。
- ・ バックアップ・タスクの開始時にトリガされたジャーナル切り替え後の、新しいジャーナル・ファイル名の数字の接尾語 `nnn` (“ジャーナル・ファイルの切り替え” を参照)
- ・ `.log` や `.cbk` 接尾語を使用します。

例えば、FullDBList バックアップ・タスクによって切り替えがトリガされた後の新しいジャーナル・ファイルに `20211031.003` という名前が付けられる場合、バックアップ・ファイルの名前は `FullDBList_20211031_003.cbk`、バックアップ・ログ・ファイルの名前は `FullDBList_20211031_003.log` になります。

#### 4.5.3 バックアップ・タスクのスケジュール

バックアップの実行スケジュールは、適切に設定する必要があります。バックアップは、システムを利用しているユーザが可能な限り少ない時間に実行します。

既定のタスクとして提供され、必要に応じて実行される 4 種類のバックアップ・タスクのほかに、これら 4 つのバックアップ・タスクに追加してバックアップ定義を作成できます。例えば、2 つのディスク・ドライブ間でバックアップする代わりに、ドライブごとにバックアップ・タスクを生成できます。

これらのバックアップ・タスクをスケジュールするには、インターシステムズのタスク・マネージャを使用します。

1. 管理ポータルで、[タスクスケジューラウィザード] に移動します ([システム処理]→[タスクマネージャ]→[新規タスク])。
2. [新規タスクスケジュール] をクリックします。
3. [タスク名]、[説明]、[タスクタイプ] を指定して、既定値に変更を加えてから [次へ] をクリックします。
4. 必要なスケジュール情報を入力して、[完了] をクリックします。

これで、[タスクスケジュール表示] ページ ([システム処理]→[タスクマネージャ]→[タスクスケジュール表示]) に目的のタスクを表示できるようになります。このタスクは、その行の [詳細] をクリックすることで他のタスクと同様に管理できます。

## 4.6 オンライン・バックアップの管理

オンライン・バックアップは、実装可能なバックアップ方法の一つです。サポートされている方法およびその最適な使用方法については、“バックアップの方法” を参照してください。

オンライン・バックアップは、管理ポータルとインターシステムズの `BACKUP` ユーティリティのいずれからでも管理できます。ここでは、ポータルでの手順を中心に説明し、必要に応じてユーティリティ使用の手順が記載されたセクションを紹介します。



データベースのバックアップ・タスクの実行とバックアップ履歴の表示は、管理ポータルの [システム処理]→[バックアップ] ページで行えます。タスク・マネージャを使用して別のバックアップ・タスクのスケジュールを設定する場合は、管理ポータル [システム処理]→[タスクマネージャ]→[タスクスケジュール] ページで管理できます。

システム管理ポータルからは、次のバックアップ・タスクを実行できます。

- ・ [バックアップ・タスクの実行](#)
- ・ [バックアップ・ステータスの表示](#)
- ・ [実行中のバックアップの中止](#)
- ・ [バックアップ履歴の表示](#)

システムに新規データベースを追加する場合は、フル・バックアップを実行する必要があります。インクリメンタル・バックアップを実行できるのは、フル・バックアップが存在するようになってからです。インクリメンタル・バックアップや累積バックアップの機能の使用を計画している場合は、バックアップ・データベース・リストを管理して、そのリストのフル・バックアップを実行する必要があります。

インストール後、FullDBList バックアップを実行して完全なバックアップを作成しておき、その後の他のバックアップ・タスクで使用できるようにすることをお勧めします。

注釈 これらのバックアップ・タスクは、管理ポータル・アプリケーションの処理部分にあります。バックアップ・タスクを実行するには適切なオペレータレベルのセキュリティ権限が必要です。

## 4.6.1 バックアップ・タスクの実行

管理ポータル [バックアップ] メニュー・オプション ([システム処理]→[バックアップ]) では、4 つのタイプのバックアップ・タスクを実行でき、それぞれのタスクに専用のメニュー項目があります。

- ・ [すべてのデータベースのフル・バックアップの実行](#)
- ・ [バックアップ・データベース・リストのフル・バックアップの実行](#)
- ・ [バックアップ・データベース・リストのインクリメンタル・バックアップの実行](#)
- ・ [バックアップ・データベース・リストの累積バックアップの実行](#)

インクリメンタル・バックアップや 累積 バックアップの前には、データベースの フル バックアップを実行する必要があります。オンライン・バックアップを実行する手順は以下のとおりです。

1. 適切なバックアップ・タイプのメニュー項目をクリックします。
2. [実行バックアップ・タスク] ボックスの設定が正しいかどうかを確認します。
3. バックアップ・オプションが正しければ、[OK] をクリックしてバックアップを開始します。  
バックアップ・オプションが適切でない場合は、"[オンライン・バックアップ設定の構成](#)" の手順に従って更新してください。
4. バックアップの実行中は、[開始済みのバックアップ] の横のテキストをクリックすると、実行中のバックアップのステータスを表示できます。詳細は、"[バックアップ・ステータスの表示](#)" を参照してください。

BACKUP ユーティリティから 1) Backup オプションを選択するか、DBACK ユーティリティを実行することによって、最近実行された直前 3 回までのバックアップ・タスクを実行することもできます。詳細は、"[DBACK によるデータベースのバックアップ](#)" を参照してください。

### マルチボリューム・バックアップ

バックアップには、複数のディスク・ファイルが必要になる場合があります。現時点では、管理ポータルでマルチボリューム・バックアップを実行することはできません。マルチボリューム・バックアップが必要な場合は、BACKUP ユーティリティ

を使用してください。ディスクがいっぱいになった場合、`BACKUP` によって、別のディスク上の別のディスク・ファイル名を入力するように要求されます。

バックアップ中にエラーが発生した場合は、その後に続くボリュームからバックアップを再開することはできません。バックアップを最初から実行する必要があります。

これらのタスクを実行する前に、バックアップのサイズを見積もる方法については、“[`DBSIZE` によるバックアップ・サイズの見積もり](#)”を参照してください。

## 4.6.2 バックアップ・ステータスの表示

バックアップ処理の進行状況を監視するには、[システム処理]→[バックアップ] ページの [バックアップステータス参照] をクリックするか、実行中のバックアップ・プロセスの [表示] をクリックします。それと同じ情報が、バックアップ処理のログ・ファイルにも記録されます。後で、[バックアップ履歴の表示] ページでこのログ・ファイルを表示できます。

バックアップを開始すると、リストの [日時] 列と [状態] 列が更新されます。[日時] 列には、バックアップの完了時ではなく、開始日時が記録されます。[状態] 列は、「☐」と表示されます。

バックアップの終了時、再度 [状態] 列が更新され、バックアップの最終的な状態が表示されます。以下に例を示します。

- ・ ☐ は、バックアップが成功したことを示します。
- ・ 「Failed」は、バックアップが実行されなかったこと、またはユーザによって停止されたことを示します。
- ・ 「☐」は、途中で問題が発生したが、完了したことを示します。

対応するバックアップ・ログには、有益な情報が記載されています。タスクの横にある [ビュー] をクリックすると、バックアップ・ログが表示されます。

“[実行中のバックアップの中止](#)”では、実行中のバックアップに表示されるその他のオプションについて説明します。

## 4.6.3 実行中のバックアップの中止

実行中のバックアップを中止するには、以下の 3 つの方法を使用できます。

- ・ [バックアップ・ステータスの表示] ページに移動し、実行中のタスクの横の [中止] をクリックしてから [OK] をクリックして、バックアップ中止の要求を確定します。
- ・ `Backup.General` クラスのメソッドを呼び出します。詳細は、“`Backup.General.GetAbortStatus()`”、“`Backup.General.ClearAbortStatus()`”、および “`Backup.General.AbortBackup()`” のメソッドの説明を参照してください。
- ・ `BACKUP` ルーチンで中止オプションを選択します。詳細は、“[`BACKUP` による実行中のバックアップの中止](#)”を参照してください。

## 4.6.4 バックアップ履歴の表示

バックアップ処理ごとに、独立したバックアップ・ログ・ファイルが生成されます。ログは、“[バックアップ・ファイル名](#)”で説明した名前付け規約に従います。

このポータルから、完了したバックアップ・タスクのシステム・バックアップ・ログのリストを表示できます。

1. 管理ポータルの [システム処理]→[バックアップ] ページに移動します。
2. 右側の列の [バックアップ履歴の表示] をクリックして、[バックアップ履歴] ページを表示します。このページには、完了したバックアップに関する情報が一覧表示されます。
3. 特定のバックアップの結果を表示するには、該当する行の右側の列の [ビュー] をクリックします。バックアップ・ログ・ファイルの内容を表示し、そのファイル内の文字列を検索できます。



## バックアップ・エラー

バックアップで入出力エラーが発生した場合、バックアップは中止され、バックアップ・ログ・ファイルにシステム・エラーが記録されます。これらのログ・ファイルに記録された情報を使用すると、問題の発生個所をすばやく確認し、修正できます。

バックアップが失敗する原因の 1 つは、ディスマウントされているデータベースでバックアップを実行しようとしたことです。バックアップ中にエラーが発生すると、バックアップ・ユーティリティにより、エラーが発生したデバイスでのバックアップを再試行できます。または、バックアップを中止することもできます。

## 4.7 オンライン・バックアップ・ユーティリティ

InterSystems IRIS には、バックアップ・タスクを実行するためのユーティリティが用意されています。`BACKUP ルーチンには、共通のバックアップ手順を実行するためのメニュー項目があります。これらのバックアップ手順は、単独で実行することもでき、場合によっては、スクリプトを使用して、非インタラクティブに実行することもできます。

ユーティリティ名は、大文字と小文字を区別します。これらのユーティリティはすべて、%SYS ネームスペースから実行します。

- ・ ``DBSIZE` によるバックアップ・サイズの見積もり
- ・ ``BACKUP` によるバックアップおよびリストアのタスクの実行
- ・ ``DBACK` によるデータベースのバックアップ
- ・ ``BACKUP` によるバックアップに必要なディレクトリ・リストの編集/表示
- ・ ``BACKUP` による実行中のバックアップの中止
- ・ ``BACKUP` によるバックアップ・ボリュームに関する情報の表示
- ・ ``BACKUP` によるバックアップまたはリストアの進行状況の監視

InterSystems IRIS では、最終バックアップ以降に変更されたデータベース・ブロックのビットマップ・リストが保管されます。このリストは、バックアップ・ステージにおけるデータベース更新の追跡に利用できます。``DBSIZE` ユーティリティは、これらのビットマップを調べ、バックアップのサイズの計算に利用できます。InterSystems IRIS ``BACKUP` ユーティリティは、これらのリストのマルチパス・スキャンを使用して最終パス以降に変更されたブロックのみをバックアップします。

### 4.7.1 ``DBSIZE` によるバックアップ・サイズの見積もり

InterSystems IRIS バックアップの実行直前に、``DBSIZE` ユーティリティを使用して、出力のサイズを見積もることができます。バックアップが開始されると、変更されるブロック数を認識する手段がないため、このサイズは単なる推定値です。``DBSIZE` 実行中にグローバル更新が行われないようにして、グローバル更新の再開前にバックアップを実行すると、取得される見積もりの精度がさらに向上します。

以下の 2 つの方法で、バックアップ・サイズを指定できます。

- ・ ``DBSIZE` のインタラクティブな実行
- ・ ルーチンからの ``DBSIZE` の呼び出し

注釈 データベースを ``DBSIZE` で評価できるようにするには、データベース・バックアップ・リストに含まれている必要があります。詳細は、“データベース・バックアップ・リストの定義”を参照してください。

### 4.7.1.1 ^DBSIZE のインタラクティブな実行

以下では、^DBSIZE をインタラクティブに実行するために必要な手順を説明します。

1. ターミナルを開いて、%SYS ネームスペースからユーティリティを実行します。

```
%SYS>Do ^DBSIZE
```

2. InterSystems IRIS に、以下のような ^DBSIZE メイン・メニューが表示されます。

```
Incremental Backup Size Estimator

What kind of backup:
  1. Full backup of all in-use blocks
  2. Incremental since last backup
  3. Cumulative incremental since last full backup
  4. Exit the backup program
1 =>
```

3. 適切なメニュー・オプションを入力して、見積もりが必要なバックアップのタイプを full (フル)、incremental (インクリメンタル)、または cumulative (累積) から選択します。このプロンプトは各タイプで同じです。
4. 次のプロンプトで、更新を一時停止して見積もりの精度を上げる場合は **Enter** キーを押し、更新を許可する場合は「**No**」を入力します。

```
Suspend Updates? Yes=>
```

更新の一時停止を選択した場合は、以下のメッセージが表示されます。

```
WARNING: Switch is set and may affect production for up to 30 seconds.
Waiting for disk cleanup to finish... ready.
```

5. 出力の最初に、選択したバックアップ・タイプに応じて実行が必要な InterSystems IRIS ブロック数が表示されます。バックアップ・リストに記載されたディレクトリ別の見積もりと合計が表示されます。以下は、フル・バックアップの場合の表示例です。

Directory	In-Use Blocks	Block Size
c:\iris-install\mgr\	19,963	(8KB)
c:\iris-install\mgr\irisaudit\	189	(8KB)
c:\iris-install\mgr\user\	61	(8KB)
Total number of database blocks:		36,228

インクリメンタル・バックアップと累積バックアップの表示は、以下のようになります。

Directory	Modified Blocks	Block Size
c:\iris-install\mgr\	5	(8KB)
c:\iris-install\mgr\irisaudit\	3	(8KB)
c:\iris-install\mgr\user\	1	(8KB)
Total number of database blocks:		10

6. その後に、バックアップ・ディスク・ファイルの領域に関する情報が表示されます。

```
Total backup size, including overhead of volume and pass labels:
```

```
For a disk file:
  Number of 512-byte blocks:  598,092  (306,223,104 bytes)
```

### 4.7.1.2 ^DBSIZE 関数の使用

ルーチンから ^DBSIZE を呼び出すこともできます。このためには、以下の関数を使用します。

```
$$INT^DBSIZE(backup_type)
```

重要            ^DBSIZE をインタラクティブに実行している場合、backup\_type の値はメニュー・オプションの番号と異なります。

バックアップ・タイプ	説明
1	インクリメンタル・バックアップ
2	フル・バックアップ
3	累積インクリメンタル・バックアップ

例えば、フル・バックアップ実行のサイズ見積もりを確認する場合は以下のようになります。

```
%SYS>w $$INT^DBSIZE(2)
18950^5^160952320^2710^160309248^313104^313104
```

戻り値は、キャレット (^) で区切られた 7 つの数字です。この例で、戻り値 18950 は、ブロック単位のバックアップの合計見積もりサイズで、5 はバックアップ対象のデータベース・ディレクトリ数を示しています。下表は、出力の各部の内容を示しています。

出力位置	説明
部分 1	バックアップ対象のデータベース・ブロック数。 (処理中にエラーが発生した場合、または無効なパラメータがある場合は -1)
部分 2	バックアップ対象のディレクトリ数。
部分 3	メディアのバイト数 (レコード間ギャップは除外)。
部分 4	メディアのブロック数。
部分 5	ディスク・ファイルのバイト数。
部分 6	ディスク・ファイルの 512 バイト・ブロック数。

以下は、各タイプのバックアップで ^DBSIZE 関数を呼び出したときの出力を示しています。

```
%SYS>w $$INT^DBSIZE(1)
466^5^4157440^70^4122624^8052^8052
```

```
%SYS>w $$INT^DBSIZE(2)
18950^5^160952320^2710^160309248^313104^313104
```

```
%SYS>w $$INT^DBSIZE(3)
466^5^4157440^70^4122624^8052^8052
```

## 4.7.2 ^BACKUP によるバックアップおよびリストアの実行

InterSystems IRIS ^BACKUP ユーティリティによって、以下の例に示すように、中央メニューから InterSystems IRIS のバックアップおよびリストアのタスクを実行できます。

```
%SYS>Do ^BACKUP
```

```
1) Backup
2) Restore ALL
3) Restore Selected or Renamed Directories
4) Edit/Display List of Directories for Backups
5) Abort Backup

6) Display Backup volume information
7) Monitor progress of backup or restore
```

Option?

対応するルーチンを開始するには、適切なメニューのオプション番号を入力します。ユーティリティを終了するには、オプションの番号を入力せずに **Enter** キーを押します。

このドキュメントの以降のセクションでは、対応するオプションを選択することによって開始されるユーティリティについて説明します。

1. ^DBACK によるデータベースのバックアップ
2. ^DBREST によるすべてのデータベースのリストア
3. ^DBREST による選択したデータベースのリストア
4. ^BACKUP によるバックアップに必要なディレクトリ・リストの編集/表示
5. ^BACKUP による実行中のバックアップの中止
6. ^BACKUP によるバックアップ・ボリュームに関する情報の表示
7. ^BACKUP によるバックアップまたはリストアの進行状況の監視

## 4.7.3 ^DBACK によるデータベースのバックアップ

インターシステムズの ^DBACK ユーティリティによって、InterSystems IRIS データベースをバックアップできます。^DBACK ユーティリティの実行は、^BACKUP ユーティリティ・メニューからオプション 1 を選択した場合と同じです。以下は、ターミナルからユーティリティを実行する場合の例です。

1. まず、データベース・バックアップ・リストが定義されていることを確認し、バックアップ・ファイルの出力デバイスを選択します。
2. %SYS ネームスペースから、^DBACK ルーチンを実行します。

```
%SYS>Do ^DBACK
```

```

                        Backup Utility
                -----
What kind of backup:
  1. Full backup of all in-use blocks
  2. Incremental since last backup
  3. Cumulative incremental since last full backup
  4. Exit the backup program
```

3. 対応するメニュー・オプション番号を入力して、実行するバックアップ・タイプ (フル、インクリメンタル、または累積) を選択します。バックアップ実行手順は、すべてのタイプで共通です。

4. 出力デバイスとバックアップの内容を入力します。

```
Specify output device (type STOP to exit)
Device: c:\iris-install\mgr\backup\FullDBList_20081201_003.cbk =>

Backing up to device: c:\iris-install\mgr\backup\FullDBList_20081201_003.cbk
Description: Full Backup using ^DBACK
```

前回のバックアップのデバイスが既定として取得されます。この例では、オンライン・バックアップ・デバイスの形式を使用しています。

5. 次に、バックアップ対象のデータベース・ディレクトリのリストが表示されます（データベース・バックアップ・リストに記載されているもの）。以下に例を示します。

```
Backing up the following directories:
c:\iris-install\mgr\
c:\iris-install\mgr\irisaudit
c:\iris-install\mgr\user\
```

6. 「Y」で応答するとバックアップが開始されます。ジャーナル・ファイルが切り替わり、バックアップの進捗状況が表示されます。

```
Start the Backup => [answer Y or N] => y
Journal file switched to:
c:\iris-install\mgr\journal\20081201.004
.
Starting backup pass 1
```

7. 3 番目または最後のパスの開始時に、バックアップをリストアするときのトランザクション・ロールバックに必要な最も古いジャーナル・ファイルに関する情報が表示されます。次に例を示します。

```
Starting backup pass 3

Journal file 'c:\intersystems\test62\mgr\journal\20180919.003' and the subsequent ones
are required for recovery purposes if the backup were to be restored

Journal marker set at
offset 197180 of c:\intersystems\test62\mgr\journal\20180919.003
```

この情報はバックアップ・ログに記録されます。

8. バックアップが正常に完了すると、以下のように表示されます。

```
***FINISHED BACKUP***

Global references are enabled.

Backup complete.

%SYS>
```

**注釈** バックアップが正常に完了すると、^DBACK がバックアップの説明をバックアップ履歴に書き込みます（“[バックアップ履歴の表示](#)”を参照）。これにより、[ジャーナル設定] ページの [ジャーナルファイルを削除するタイミング]→[この回数のバックアップ成功後] の設定（既定値は 2）によっては、ジャーナルの削除がトリガされる可能性があることに注意してください。詳細は、“[ジャーナル設定の構成](#)”を参照してください。

“[^DBACK に対する外部エントリ・ポイント](#)”では、指定されたエントリ・ポイントを使用してこのユーティリティを外部のバックアップ・プロシージャから呼び出す方法を説明します。

#### 4.7.3.1 ^DBACK に対する外部エントリ・ポイント

^DBACK ユーティリティに対して指定されたエントリ・ポイントは、外部バックアップ・プロシージャで使用できます。バックアップの整合性確保のため、このポイントは説明されたとおりに使用する必要があります。エントリ・ポイントはすべて、正

常完了の場合は 1 を返し、失敗の場合は 0 を返します。`DBACK` ルーチンでは、以下の外部エントリ・ポイントを使用できます。

- ・ `BACKUP`DBACK`
- ・ `LISTDIRS`DBACK`
- ・ `CLRINC`DBACK` (並行外部バックアップの場合のみ)

## BACKUP`DBACK

このプロシージャによってバックアップが起動され、`DBACK` のインタラクティブな実行の際に示される質問に対しては、渡された引数を使用して応答します。

```
BACKUP`DBACK(argfile,TYPE,DESC,OUTDEV,kiljrn,LOGFILE,MODE,clrjrn,swjrn,
nwjrnfil,quietimeout,taskname)
```

大文字で示された引数は、今回の InterSystems IRIS リリースのバックアップ・プロシージャで最も重要な引数です。ジャーナリング・メカニズムが変更されているため、バックアップ・ユーティリティでは、現在のジャーナル・ファイルの削除、現在のジャーナル・ファイルのクリア、および新しいジャーナル・ファイル名の指定はできなくなりました。このため、今回のバージョンの InterSystems IRIS では、次の表に示されるように多数の引数が無視されます。

引数	説明
argfile	使用できなくなりました。必ず NULL 値を使用します。
TYPE	バックアップのタイプ (必須) 値 : I – インクリメンタル C – 累積 F – フル E – サードパーティ、またはオペレーティング・システムのバックアップ・ユーティリティによる外部フル・バックアップ。
DESC	バックアップ・ラベルおよびバックアップ履歴グローバルに保存されている説明。空白が許容される自由形式のテキスト文字列。

外部 (タイプ E) のバックアップでは、次の表の引数が無視されます。この表では、他のタイプのバックアップで無視されるか必須であるかについても記載されています。

引数	説明
OUTDEV	バックアップの出力ファイル名 (必須)。
kiljrn	ジャーナル・ファイルを切り替えるかどうかを指定します。 Y – ジャーナル・ファイルを切り替えます (推奨)。 N – 切り替えません。 (現在のジャーナル・ファイルを削除することはできなくなりました。)
LOGFILE	ログに記録するかどうか、および関数の出力を書き込むかどうかを指定します。 ファイル名 – ターミナルに送信されるすべてのメッセージをこのファイルに書き込みます (推奨)。 Null 値 – ログファイルがありません。

引数	説明
MODE	<p>プロシージャの実行時にターミナルに書き込まれる出力を指定します。ログ・ファイルに書き込まれる内容を制御するものではありません。</p> <p>値：</p> <p>“NOISY” – ターミナルのすべてのテキストを書き込みます（既定）。</p> <p>“QUIET” – 異常な状態に関係するテキストのみを書き込みます。</p> <p>“NOINPUT” – このプロセスにアタッチされたターミナルがないため、出力は送信されません。読み取りが必要な場合は、バックアップが中止されます。</p>
cljrjn	<p>ジャーナル・ファイルを切り替えるかどうかを指定します。</p> <p>Y – ジャーナル・ファイルを切り替えます（推奨）。</p> <p>N – 切り替えません。</p> <p>（現在のジャーナル・ファイルを消去することはできなくなりました。）</p>
swjrn	<p>ジャーナル・ファイルを切り替えるかどうかを指定します。</p> <p>Y – ジャーナル・ファイルを切り替えます（推奨）。</p> <p>N – 切り替えません（cljrjn 引数または kiljrjn 引数の値が Y である場合、swjrn は Y に変換し、ジャーナル・ファイルが切り替えられます）</p>
nwjrnfil	（無視）

以下は、2 つのオプションの引数です。

引数	説明
quietimeout	システムが停止するまで待機する秒数。この秒数を経過するとバックアップが中止されます。ゼロ (0) または負の値は、無限に待機することを示します（既定 = 60）。
taskname	バックアップ・タイプ F、I、および C の場合に内部で使用されるタスク名。

これらの外部エントリ・ポイントを使用するには、すべてのモードで `SYS グローバルへの読み取り/書き込みアクセス権が必要です。外部（タイプ E）バックアップでは、`SYS グローバル内のディレクトリの .IND ファイル、.INE ファイル、および .INF ファイルが削除されます。また、ルーチンによって、このバックアップが LASTFULL バックアップであるという説明と共にバックアップ履歴グローバルに記録されます。

このルーチンを呼び出す際に、スイッチ 10 を設定している場合は、バックアップ全体を通じてこの設定が保持されます。スイッチ 13 を設定している場合、バックアップで必要な Set および Kill コマンドが実行される間はプロシージャによってスイッチ 10 に変換され、終了時にスイッチ 13 に復元されます。

このルーチンは、このエントリ・ポイントによって定義された変数に対しては New コマンドを実行しますが、`DBACK プロシージャの本体内で定義された変数では実行しません。このため、`DBACK の呼び出し前に呼び出されるすべてのルーチンで、その他のすべての変数を慎重に保護する必要があります。

返り値：

- 0 – 失敗しました。ログ・ファイルを指定している場合はこのファイルを確認してください。
- 1 – 成功
- 1, warning message string – 成功しましたが、警告があります。この警告は、文字列内でチルダ文字 (~) によって区切られています。



## LISTDIRS^DBACK

このプロシージャは、出力ファイルを開き、バックアップに含まれるデータベース・ディレクトリに関するリストを書き込みます。リストが空白の場合は、すべてのデータベース・ディレクトリ (**IRISTEMP**、**IRISLIB**、および **IRISLOCALDATA** を除く) がプロシージャによってバックアップされます。この関数の形式は以下のとおりです。

LISTDIRS^DBACK (file, mode)

引数	説明
file	関数がバックアップ・リスト内のデータベース・ディレクトリのリストを書き込む出力ファイルの名前。
mode	このプロシージャの実行時にターミナルに書き込まれる出力内容を指定します。 “QUIET” – 返り値、および異常な状態に関するテキストのみを表示します。  その他のすべての値 – ターミナルにディレクトリ・リストを表示します。

以下は、ターミナルに表示される出力の表示例です。

```
%SYS>w $$LISTDIRS^DBACK("c:\temp\listdirs.txt", "NOISY")

List of directories to be backed up
c:\iris-install\mgr\irisaudit\
c:\iris-install\mgr\irislib\
c:\iris-install\mgr\
c:\iris-install\mgr\user\
1
%SYS>
```

このプロシージャに関する注意事項は以下のとおりです。

- ・ ^SYS グローバルに対する読み取りアクセス権が必要です。
- ・ スイッチの設定やクリアは行われません。
- ・ すべてのローカル変数に対して New コマンドを発行します。

## CLRINC^DBACK (並行外部バックアップの場合のみ)

このプロシージャを呼び出すと、バックアップ・リスト内のデータベースで変更されているブロックを示すインクリメンタル・バックアップ・ビットマップがクリアされます。同時に、先にフル・バックアップを実行しなければあらゆる種類のバックアップが実行できなくなるため、バックアップ履歴も削除されます。このプロシージャは、InterSystems IRIS 実行状態での外部フル・バックアップの実行前に使用することを目的としています。InterSystems IRIS がダウンした状態で外部バックアップを実行する場合は、システムのシャットダウン前にこのエントリ・ポイントを呼び出します。この関数の形式は以下のとおりです。

CLRINC^DBACK (mode)

引数	説明
mode	このプロシージャの実行時にターミナルに書き込まれる出力内容を指定します。 “QUIET” – 返り値、および異常な状態に関するテキストのみを表示します。  その他のすべての値 – ディレクトリの処理時にその名前を表示します。

例：

```
%SYS>w $$CLRINC^DBACK("QUIET")
1
%SYS>

%SYS>Do CLRINC^DBACK("NOISY")

      Cleared incremental backup bitmap in c:\iris-install\mgr\
      Cleared incremental backup bitmap in c:\iris-install\mgr\irisaudit\
      Cleared incremental backup bitmap in c:\iris-install\mgr\user\
%SYS>
```

このプロシージャに関する注意事項は以下のとおりです。

- ・ ^SYS グローバルに対する読み取り/書き込みアクセス権が必要です。
- ・ スイッチ 13 の状態を記録し、設定されていない場合は設定して、終了時に元の状態に復元します。
- ・ メイン・バックアップ・プロシージャで定義されたものを除き、すべてのローカル変数に対して New コマンドを発行します。
- ・ このルーチンの呼び出し時にディスマウントされていたディレクトリは、ディスマウントされた状態で保持されます。

#### 4.7.4 ^BACKUP によるバックアップに必要なディレクトリ・リストの編集/表示

**重要** データベースのリストを編集する場合は、ディレクトリ名ではなくデータベース名を使用します。これは管理ポータルでバックアップを構成する場合と同様です。

InterSystems IRIS の ^BACKUP ユーティリティのオプションを使用すると、InterSystems IRIS データベースのバックアップや、既に作成されたバックアップのリストアができます。データベースのリストが作成されていない場合は、すべてのデータベースがバックアップ対象となります。リストを作成すると、スクリプト化されたバックアップの ^BACKUP ユーティリティのエントリ・ポイントの呼び出しも含めて、InterSystems IRIS バックアップ・システムのすべてに適用されます。

以下の例は、メニューの各オプションからの出力サンプルです。

1. ^BACKUP メニューから、オプション 4 を選択して、データベース・バックアップ・リストの管理メニューを表示します。

```
%SYS>Do ^BACKUP

1) Backup
2) Restore ALL
3) Restore Selected or Renamed Directories
4) Edit/Display List of Directories for Backups
5) Abort Backup
6) Display Backup volume information
7) Monitor progress of backup or restore

Option? 4

1) Add a database to the backup
2) Remove a database from the backup
3) Show current list of databases included in backups
4) Show list of available databases
5) Display last full backup information
```

2. リストにデータベースを追加するには、オプション1 を選択します。この例では、データベースが現在のリストに追加されたことを示すため、オプション 3 も使用しています。

```
1) Add a database to the backup
2) Remove a database from the backup
3) Show current list of databases included in backups
4) Show list of available databases
5) Display last full backup information

Option? 1
Enter database to add? DATA
Enter database to add?
```

```
You've selected DATA to be added to the backups
Are you sure you want to do this (yes/no)? y
Completed.
```

- 1) Add a database to the backup
- 2) Remove a database from the backup
- 3) Show current list of databases included in backups
- 4) Show list of available databases
- 5) Display last full backup information

Option? 3

```
The following 3 databases are included in backups
  IRISAUDIT      C:\iris-install\Mgr\irisaudit\
  IRISYS         C:\iris-install\Mgr\
  USER          C:\iris-install\Mgr\User\
```

3. リストのデータベースを削除するには、オプション 2 を選択します。この例では、データベースが現在のリストから削除されたことを示すため、オプション 3 も使用しています。

- 1) Add a database to the backup
- 2) Remove a database from the backup
- 3) Show current list of databases included in backups
- 4) Show list of available databases
- 5) Display last full backup information

Option? 2

```
Enter database to remove (^ when done)? DATA
Enter database to remove (^ when done)?
You've removed DATA from the backups
Are you sure you want to do this? y
Completed.
```

- 1) Add a database to the backup
- 2) Remove a database from the backup
- 3) Show current list of databases included in backups
- 4) Show list of available databases
- 5) Display last full backup information

Option? 3

```
The following 3 databases are included in backups
  IRISAUDIT      C:\iris-install\Mgr\irisaudit\
  IRISYS         C:\iris-install\Mgr\
  USER          C:\iris-install\Mgr\User\
```

4. このインスタンスのすべてのデータベースのリストを表示するには、オプション 4 を選択します。

- 1) Add a database to the backup
- 2) Remove a database from the backup
- 3) Show current list of databases included in backups
- 4) Show list of available databases
- 5) Display last full backup information

Option? 4

```
The following is a list of all databases in the configuration.
Databases which are part of the backup are marked with (*)
  (*) IRISAUDIT      c:\iris-install\mgr\irisaudit\
    IRISLIB (Read Only) c:\iris-install\mgr\irislib\
    IRISLOCALDATA    c:\iris-install\mgr\irislocaldata\
  (*) IRISYS         c:\iris-install\mgr\
  (*) USER          c:\iris-install\mgr\user\
```

5. 最新のフル・バックアップに関する情報を表示するには、オプション 5 を選択します。

- 1) Add a database to the backup
- 2) Remove a database from the backup
- 3) Show current list of databases included in backups
- 4) Show list of available databases
- 5) Display last full backup information

Option? 5

```
=====Last Full Backup Information=====
```

```
Date: 19 Jan 2018
Description: Full backup of all databases that are in the backup database list.
Device: c:\iris-install\mgr\backup\FullDBList_20180119_001.cbk
```

## 4.7.5 ^BACKUP による実行中のバックアップの中止

^BACKUP ユーティリティのメニュー・オプション 5 を使用すると、実行中のバックアップを中止できます。

```
%SYS>Do ^BACKUP

1) Backup
2) Restore ALL
3) Restore Selected or Renamed Directories
4) Edit/Display List of Directories for Backups
5) Abort Backup
6) Display Backup volume information
7) Monitor progress of backup or restore

Option? 5
Are you sure you want to abort the backup operation ? <No> Yes
Backup abortion succeed.
```

このオプションを選択したときにバックアップが実行中でない場合は、以下のメッセージが表示されます。

```
No Backup operation is currently running.
```

## 4.7.6 ^BACKUP によるバックアップ・ボリュームに関する情報の表示

以下に示すように、^BACKUP メニューからオプション 6 を選択すると、バックアップ・ファイルのパス名の入力を求めるプロンプトが表示されます。

```
Enter the backup volume you want to display information from
(Type STOP to exit)
Device: c:\131ul\mgr\backup\FullAllDatabases_20180802_001.cbk =>

This backup volume was created by:
InterSystems IRIS for Windows (x86-64) 2018.1.1 (Build 257U) Sun Jul 1 2018 23:35:31 EDT

The volume label contains:
Volume number          1
Volume backup          AUG 2 2018 12:27PM Full
Previous backup        AUG 2 2018 11:35AM Cumulative Incremental
Last FULL backup      AUG 2 2018 11:24AM
Description            Full backup of ALL databases
Buffer Count           0
Mirror name            MIRROR122
Failover member        NONNIE20/TEST2A
Journal File           c:\131ul\mgr\journal\20180802.005
Log file               c:\131ul\mgr\Backup\FullAllDatabases_20180802_001.log
Backup Status          Completed

Database                Size(mb)  Mirror DB Name
-----
c:\131ul\mgr\          191
c:\131ul\mgr\irisaudit\ 11
c:\131ul\mgr\user\     2113  MUSER
Total of 3 databases, totaling 2315 mb
Backup volume size is 2365 mb
```

## 4.7.7 ^BACKUP によるバックアップまたはリストアの進行状況の監視

^BACKUP を使用して開始された、実行中のバックアップ・ジョブまたはリストア・ジョブの進行状況を監視するには、^BACKUP メニューからオプション 7 を選択します。バックアップ・ジョブまたはリストア・ジョブを開始する前に、別のターミナル・ウィンドウで監視を開始できます。

実行中のバックアップ・ジョブの場合、以下の情報が表示されます。

```

Backup Status
-----
Status: Running
Backup process pid: 13052
Backup file: c:\132ul\mgr\stcfull.bck
Description:
Log file: c:\132ul\mgr\irisbackup.log
Type: Full
Start time: 2018-09-05 14:09:02
Estimated finish time: 2018-09-05 14:09:45
Total time (hh:mm:ss): 00:00:18
Time remaining (hh:mm:ss): 00:00:25
Estimated backup size: 7.68GB
Current backup size: 3.12GB
Current Backup rate: 184MB/sec
% Completed 41

```

実行中のリストア・ジョブの場合、以下の情報が表示されます。

```

Restor Status
-----
Status: Running
Restore process pid: 13052
Restore file: c:\132ul\mgr\stcfull.bck
Description:
Type: Full
Start time: 2018-09-05 14:10:49
Estimated finish time: 2018-09-05 14:12:07
Total time (hh:mm:ss): 00:00:34
Time remaining (hh:mm:ss): 00:00:44
Restore file size: 7.68GB
Restore size completed: 3.28GB
Current restore rate: 103MB/sec
% Completed 43

```

注釈 インクリメンタル・バックアップを実行している場合、フィールドの一部を使用できず、N/A と表示されます。これには、 、 、および % が含まれます。

リストア・ジョブは、リストア開始時にスイッチ 10 が設定された場合、監視できません。

## 4.8 オンライン・バックアップ・リストア・ユーティリティ

オンライン・バックアップ・リストア・ユーティリティ (DBREST) は、以下のアクションを実行します。

- すべてのディレクトリまたは選択したディレクトリのどちらかをリストアするかを指定する必要があります。
- リストア中は、実行されている他の InterSystems IRIS 処理をすべて停止するかどうかを確認します。ほとんどの場合、[はい] を選択します。
- リストアするバックアップを保持しているファイル名を確認します。バックアップが複数のボリュームに存在する場合、バックアップ情報はボリューム・ヘッダに書き込まれます。最初のボリュームのリストア後、次にリストアするボリュームがユーティリティから指示されます。
- ボリューム・ヘッダを表示します。このヘッダには、バックアップ中に確定した情報が記述されています。この情報には、このボリュームのバックアップ日付、前回のバックアップ日付、および最後のフル・バックアップ日付があります。
- リストアするバックアップが現在のバックアップでよいかを確認します。
- リストアするデバイス上のディレクトリをリストします。
- 次にリストアするバックアップを含む他の入力デバイスを指定できます。
- バックアップ・ボリュームに関する情報をいつでも表示できます。

^DBREST メニューのオプション 1、2、および 3 は、^BACKUP メニューのオプション 2、3、および 6 と同じです。

これらのリストア・オプションのいずれかを選択すると、リストアする最初のバックアップが保存されたデバイス名を入力するよう求められます。リストア・オプションの初回入力時の既定値は、最新のフル・バックアップが送信されたデバイスです (存在する場合)。

**注意** インクリメンタル・バックアップ、または累積バックアップをリストアする場合、対象となるデータベースの状態が、最新のフル・バックアップでリストアした時点と完全に同じ状態である必要があります。

後続のすべてのインクリメンタル・バックアップのリストアが完了するまで、リストアしたデータベースは絶対に更新しないでください。この警告を守らないと、インクリメンタル・バックアップや累積バックアップからのリストアで、データベースが破損することがあります。

以下は、ターミナルからユーティリティを実行する場合の例です。

```
%SYS>Do ^DBREST

                                DBREST Utility
                                Restore database directories from a backup archive

Restore: 1. All directories
          2. Selected and/or renamed directories
          3. Display backup volume information
          4. Exit the restore program
1 =>
```

以下のセクションでは、各オプションを選択する手順について説明します。

1. [^DBREST によるすべてのデータベースのリストア](#)
2. [^DBREST による選択したデータベースまたは名前を変更したデータベースのリストア](#)
3. [^BACKUP によるバックアップ・ボリュームに関する情報の表示](#)
4. [バックアップ履歴によるデータベースのリストア](#)
5. [^DBREST による自動リストア](#)
6. [ミラーリングされたデータベースの考慮事項](#)

また、スクリプトで非インタラクティブにこれらの関数を実行できます。詳細は、“[^DBREST の外部エントリ・ポイント](#)”を参照してください。

## 4.8.1 ^DBREST によるすべてのデータベースのリストア

^DBREST メニューから 1 を選択すると、^BACKUP メニューから 2 を選択した場合と同じになります。

**注釈** ミラーリングされたデータベース (ミラーリング・データベース) をリストアする場合は、手順の実行の前に“[ミラーリングされたデータベースの考慮事項](#)”を確認してください。

以下の手順は、すべてのディレクトリをリストアする場合の概要を示す例です。リストア処理の開始プロンプトが示されます。処理を進めるにつれ、リストア・プロセスのステップングの際に非常に個別的な質問が行われます。

1. ユーティリティ・メニューからすべてのディレクトリのリストアを選択します。このオプションでは、バックアップ・メディアにあるすべてのディレクトリがリストアされます。

```
%SYS>Do ^DBREST
```

```

                        DBREST Utility
Restore database directories from a backup archive

Restore: 1. All directories
          2. Selected and/or renamed directories
          3. Display backup volume information
          4. Exit the restore program
1 => 1
```

2. すべてのディレクトリをリストアすることを確認します。

```
Proceed with restoring ALL directories Yes=>
```

3. 次に、すべてのデータベースのリストア先にする最上位パスを入力するよう求められます。これは、リストア対象のデータベースの元のパスに接頭語として自動的に追加されます。これによって、元のディレクトリをこのディレクトリのサブディレクトリとして、すべてのデータベースがこのディレクトリの下位でリストアされます。これを無視して、データベースを元のパスでリストアするには、**Enter** を押します。

```
Top directory for all Databases to be restored to (? for Help)?
```

4. リストア実行中、InterSystems IRIS 処理を一時停止するかどうかを示します。これらの処理は一時停止することをお勧めします。

```
Do you want to set switch 10 so that other processes will be
prevented from running during the restore? Yes =>
```

5. リストア場所から最初のファイルを指定します。**Enter** キーを押すと既定のファイルを指定できます。既定のファイルは、最後のフル・バックアップです。

```
Specify input file for volume 1 of backup 1
(Type STOP to exit)
Device: c:\iris-install\mgr\backup\FullAllDatabases_20180323_001.cbk =>
```

6. バックアップの詳細が正確かどうかをチェックし、このファイルが、リストアするファイルであることを確認します。

```

This backup volume was created by:
InterSystems IRIS for Windows 2018.1.1

The volume label contains:
Volume number      1
Volume backup      MAR 23 2018 09:52AM Full
Previous backup    MAR 22 2018 11:00AM Incremental
Last FULL backup   MAR 16 2018 11:00AM
Description        Full backup of ALL databases, whether or not they are in
                   the backup database list.
Buffer Count       0
Is this the backup you want to start restoring? Yes =>
```

7. リストアするディレクトリがユーティリティから示され、リストア処理が進められます。

```

The following directories will be restored:
c:\iris-install\mgr\
c:\iris-install\mgr\irisaudit\
c:\iris-install\mgr\test\
c:\iris-install\mgr\user\

***Restoring c:\iris-install\mgr\ at 10:46:01
146045 blocks restored in 241.3 seconds for this pass, 146045 total restored.
```



8. リストアを要する次のインクリメンタル・バックアップの入力ファイルを指定します。これ以上リストアする入力ファイルがない場合は「stop」と入力します。

```
Specify input file for volume 1 of backup following MAR 23 2018 09:52AM
(Type STOP to exit)
Device: stop
```

9. 他のバックアップをリストアするかどうかを示します。「Yes」で応答すると、このプロシージャが再び実行されます。「No」で応答すると、リストアされたデータベースがシステムによってマウントされます。

```
Do you have any more backups to restore? Yes => No
Mounting c:\iris-install\mgr\
c:\iris-install\mgr\ ... (Mounted)

Mounting c:\iris-install\mgr\irisaudit\
c:\iris-install\mgr\irisaudit\ ... (Mounted)

Mounting c:\iris-install\mgr\test\
c:\iris-install\mgr\test\ ... (Mounted)

Mounting c:\iris-install\mgr\user\
c:\iris-install\mgr\user\ ... (Mounted)
```

10. リストアしたデータベースに割り当てるジャーナル・エントリと、リストアするジャーナル・ファイル名を指定します。通常は、オプション 1 を選択して、リストアしたディレクトリに影響する変更のみを適用します。

注釈 すべてのディレクトリまたは選択されたディレクトリのすべてのエントリとグローバルをリストアするときには何が起きるかについては、“[JRNRESTOを使用したジャーナル・ファイルからのグローバルのリストア](#)”を参照してください。

```
Restoring a directory restores the globals in it only up to the
date of the backup. If you have been journaling, you can apply
journal entries to restore any changes that have been made in the
globals since the backup was made.
```

What journal entries do you wish to apply?

1. All entries for the directories that you restored
2. All entries for all directories
3. Selected directories and globals
4. No entries

Apply: 1 =>

11. いくつかの情報の確認後に、ジャーナル・ファイルからのリストアが開始されます。

```
We know something about where journaling was at the time of the backup:
0: offset 172940 in c:\iris-install\mgr\journal\20180323.002
```

```
Use current journal filter (ZJRNFLT)? No
Use journal marker filter (MARKER^ZJRNFLT)? No
Updates will not be replicated
```

```
The earliest journal entry since the backup was made is at
offset 172940 in c:\iris-install\mgr\journal\20180323.002
```

```
Do you want to start from that location? Yes => Yes
Final file to process (name in YYYYMMDD.NNN format): <20180323.003> [?]
=>
```

Prompt for name of the next file to process? No => No

Provide or confirm the following configuration settings:

Journal File Prefix: =>

Files to dejournal will be looked for in:

```
c:\iris-install\mgr\journal\
c:\journal\altdir\
```

in addition to any directories you are going to specify below, UNLESS you enter a minus sign ('-' without quotes) at the prompt below, in which case ONLY directories given subsequently will be searched

```

Directory to search: <return when done>
Here is a list of directories in the order they will be searched for files:
  c:\iris-install\mgr\journal\
  c:\journal\altdir\
The journal restore includes the current journal file.
You cannot do that unless you stop journaling or switch
  journaling to another file.
Do you want to switch journaling? Yes => Yes
Journaling switched to c:\iris-install\mgr\journal\20180323.004

You may disable journaling the updates for faster restore.
Do you want to disable journaling the updates? Yes => yes
Updates will NOT be journaled

```

12. ユーティリティによって進捗状況が表示され、完了時点が示されます。

```

c:\iris-install\mgr\journal\20180323.002
 61.32%  65.03%  68.44%  72.21%  75.86%  79.26%  82.73%  86.08%  89.56%
 92.99%  96.07%  98.87%100.00%
***Journal file finished at 11:03:31

c:\iris-install\mgr\journal\20180323.003
 16.17%  17.10%  17.90%  18.90%  20.05%  21.33%  22.58%  23.81%  25.15%
 26.32%  27.65%  28.85%  30.08%  31.37%  32.59%  33.98%  35.16%  36.25%
 37.32%  38.41%  39.55%  40.72%  41.81%  42.83%  43.85%  44.89%  46.00%
 47.15%  48.24%  49.28%  50.32%  51.41%  52.54%  53.71%  54.76%  55.80%
 56.85%  57.97%  59.10%  60.16%  61.17%  62.19%  63.24%  64.32%  65.18%
 66.02%  66.87%  67.71%  68.52%  69.34%  70.14%  70.96%  71.76%  72.60%
 73.58%  74.51%  75.43%  76.35%  77.26%  78.17%  79.07%  79.69%  80.31%
 80.93%  81.56%  82.20%  82.83%  83.47%  84.27%  87.00%  88.57%  91.65%
 93.03%  96.09%  97.44%  99.04%100.00%
***Journal file finished at 11:03:32

Journal reads completed. Applying changes to databases...
 14.29%  28.57%  42.86%  57.14%  71.43%  85.71% 100.00%

[journal operation completed]

```

## 4.8.2 ^DBREST による選択したデータベースまたは名前を変更したデータベースのリストア

^DBREST メニューから 2 を選択すると、^BACKUP メニューから 3 を選択した場合と同じになります。このオプションでは、バックアップ・メディアからリストアするディレクトリを選択できます。また、別のディレクトリ名や別のデバイスでデータベースをリストアすることもできます。

注釈 ミラーリングされたデータベース(ミラーリング・データベース)をリストアする場合は、手順の実行の前に“[ミラーリングされたデータベースの考慮事項](#)”を確認してください。

以下の例は、選択したディレクトリまたは名前を変更したディレクトリのリストア方法を示しています。

1. ^DBREST ユーティリティからオプション 2 を選択します。

```

%SYS>Do ^DBREST

                                DBREST Utility
                                Restore database directories from a backup archive

Restore: 1. All directories
          2. Selected and/or renamed directories
          3. Display backup volume information
          4. Exit the restore program
          1 => 2

```

- リストア実行中、InterSystems IRIS 処理を一時停止するかどうかを示します。これらの処理は一時停止することをお勧めします。

```
Do you want to set switch 10 so that other InterSystems IRIS processes
will be prevented from running during the restore? Yes =>
```

- リストア場所から最初のファイルを指定します。**Enter** キーを押すと既定のファイルを指定できます。既定のファイルは、最後のフル・バックアップです。

```
Specify input file for volume 1 of backup 1
(Type STOP to exit)
Device: c:\iris-install\mgr\backup\IncrementalDBList_20180323_001.cbk =>
```

- バックアップの詳細が正確かどうかをチェックし、このファイルが、リストアするファイルであることを確認します。

```
This backup volume was created by:
  InterSystems IRIS for Windows (Intel) 5.1

The volume label contains:
  Volume number      1
  Volume backup      MAR 23 2018 11:03AM Full
  Previous backup    MAR 23 2018 09:52AM Full
  Last FULL backup   MAR 23 2018 09:52AM
  Description        Incremental backup of all databases that are in the backup
                     database list.
  Buffer Count       0
Is this the backup you want to start restoring? Yes =>
```

- ユーティリティからディレクトリ名を要求されるので、リストアするデータベースとリストア先のディレクトリを指定します。

```
For each database included in the backup file, you can:

-- press RETURN to restore it to its original directory;
-- type X, then press RETURN to skip it and not restore it at all.
-- type a different directory name. It will be restored to the directory
   you specify. (If you specify a directory that already contains a
   database, the data it contains will be lost).

c:\iris-install\mgr\ =>
c:\iris-install\mgr\irisaudit\ =>
c:\iris-install\mgr\test\ =>
c:\iris-install\mgr\user\ =>
```

- ディレクトリに関するプロンプトのそれぞれに応答すると、次のプロンプトが表示されます。

```
Do you want to change this list of directories? No =>
```

選択を編集する場合は「Yes」と入力し、確定する場合は **Enter** キーを押します。

- 前セクションのすべてのディレクトリのリストア手順と同様に処理が続行されます。

この手順は、リストアするシステムと異なるシステム上で実行されたバックアップのリストアに使用できます。

## 4.8.3 バックアップ履歴によるデータベースのリストア

InterSystems IRIS のバックアップ・ユーティリティでは、バックアップ履歴 が管理されるため、バックアップを論理的な順序でリストアできます。リストア・ユーティリティにより、タイプとバックアップ履歴の順序に従ってバックアップがリストアされるように要求されます。

最新のフル・バックアップをリストアした後、バックアップ履歴 の情報を使用して、リストアする次の論理バックアップが示され、履歴の最後までこれが続きます。後続のバックアップは以下の順序でリストアするように要求されます。

- 最新のフル・バックアップの実行後に、最新の累積バックアップ (存在する場合) が要求されます。
- 最新の累積バックアップ以降の (累積バックアップが存在しない場合は、最新のフル・バックアップ以降の) すべてのインクリメンタル・バックアップが要求されます。一番古いものから最新までの順番で、リストアを実行します。

リストア処理で、指示されたバックアップを変更できます。しかし、インクリメンタル・バックアップや累積バックアップは、データベースの完全なコピーではありません。インクリメンタル・バックアップをリストアするには、事前にフル・バックアップをリストアしておく必要があります。

**重要** クラスタから同一データベースへのアクセスがサポートされる InterSystems IRIS プラットフォームでは、指定されたデータベース・ディレクトリを常に同じノードからバックアップする必要があります。これにより、ディレクトリのリストアが必要な場合に、完全なバックアップ履歴の使用が可能になります。

## 4.8.4 ^DBREST による自動リストア

InterSystems IRIS リストア・ユーティリティ ^DBREST には、外部エン트리・ポイントによる非インタラクティブな実行オプションが用意されています。

**注釈** ミラーリングされたデータベース (ミラーリング・データベース) をリストアする場合は、自動リストアを実行する前に [“ミラーリングされたデータベースの考慮事項”](#) を確認してください。

これらの 2 つのエン트리・ポイントのいずれかを呼び出すことによって、無人で実行されるリストアを実装するスクリプトを作成できます。

- EXTALL^DBREST – バックアップ・デバイスからすべてのデータベースをリストアします ([“^DBREST によるすべてのデータベースのリストア”](#) で説明されている手順と同等の自動リストア)。^DBREST ユーティリティの EXTALL エン트리・ポイントを使用する構文は、以下のようになります。

```
EXTALL^DBREST(quietmode,allowupd,inpdev,dirlist,jrnopt,jrnfile,jdirglo)
```

**注釈** ターゲット・ディレクトリが存在するが、IRIS.DAT ファイルが存在しない場合、EXTALL^DBREST は IRIS.DAT ファイルをリストアします。

- EXTSELCT^DBREST – 選択されたファイルをバックアップ・デバイスからリストアするか、またはソース・ディレクトリと異なるターゲット・ディレクトリにリストアします ([“^DBREST による選択したデータベースまたは名前を変更したデータベースのリストア”](#) で説明されている手順と同等の自動リストア)。^DBREST ユーティリティの EXTSELCT エン트리・ポイントを使用する構文は、以下のようになります。

```
EXTSELCT^DBREST(quietmode,allowupd,inpdev,dirlist,jrnopt,jrnfile,jdirglo)
```

両方のエン트리・ポイントは、呼び出しの状態を返す関数です。引数はすべて入力のみです。特に指定がない限り、下表に記載された入力引数は両方の関数で使用されます。

引数	説明
allowupd	リストア処理中に更新を許可するか否かのインジケータ 1 – リストア処理中に更新を許可します。 0 – 更新を許可しません。
dirlist	リストアするディレクトリのリストを含むファイルの名前。リストアする各ディレクトリに 1 つのレコードが存在する必要があります。EXTALL エン트리・ポイントでは無視されます。 <a href="#">“ディレクトリ・リスト・ファイルの要件”</a> を参照してください。 ^DBREST が完了すると、dirlist で指定されたファイルが削除されます。
inpdev	バックアップを含む入力デバイス。
jdirglo	jrnopt が 3 の場合のみ使用します。これは、ジャーナルのリストアのためのディレクトリおよびグローバルの選択条件を含むファイルの名前です。 <a href="#">“ディレクトリとグローバルの選択条件”</a> を参照してください。

引数	説明
jrnfile	ジャーナル・ファイル。NULL の場合、ユーティリティは、`%SYS("JOURNAL","CURRENT")` グローバルに保存されている現在のファイルを使用します。
jrnopt	ジャーナルをリストアするためのオプション 1 – バックアップをリストアしたすべてのディレクトリ 2 – ジャーナル内のすべてのディレクトリ 3 – jdirglo 引数によって指定された、選択されたディレクトリおよびグローバル 4 – ディレクトリなし
quietmode	値は、処理が quiet (非インタラクティブ) モードであることを示し、この外部呼出しに対しては、NULL 以外の値 (通常 1) とする必要があります。

これらの関数の呼び出しから、以下のコードが返される可能性があります。

リターン・コード	説明
3	エラーは発生していません。正常に完了しました。
-1	入力デバイス (リストア元のデバイス) を開くことができません。
-2	ボリュームのラベルが、バックアップ履歴のラベルと一致しません。
-3	バックアップおよびリストアは、既に進行中です。
-4	読み取り選択条件に対して (ディレクトリの選択的リストアに対して) 無効なデバイスです。
-5	無効なジャーナル・リストア・オプションです。
-6	無効なジャーナル・ファイルであるか、ジャーナル・リストアの選択条件のファイルを開けません。

#### 4.8.4.1 ディレクトリ・リスト・ファイルの要件

dirlist によって示されたファイルの要件

- ・ リストアするディレクトリごとに 1 つのレコードを含みます。
- ・ 各フィールドをコンマで区切ります (,)。
- ・ 各レコードの形式は、SourceDir、TargetDir、CreateDir で、これらの意味は以下のとおりです。

引数	説明
SourceDir	リストアするディレクトリの名前。
TargetDir	リストア先のディレクトリの名前。ソース・ディレクトリと同じ名前のディレクトリにリストアする場合は省略します。
CreateDir	ターゲット・ディレクトリが存在しない場合、ターゲット・ディレクトリを作成するか否か。 Y – ターゲット・ディレクトリを作成します。 N – ターゲット・ディレクトリを作成しません。

例えば、テキスト・ファイル (RestoreList.txt) を以下の文字列を含む **C:¥Backup¥** ディレクトリに作成したと仮定します。

```
C:\intersystems\iris\mgr\user\C:\intersystems\iris\mgr\Y
```

以下のルーチンを %SYS ネームスペースで実行すると、C:\intersystems\iris\mgr\user\ から C:\intersystems\iris\mgr\ にデータベースをリストアできます。

```
set SourceDir = "C:\intersystems\iris\mgr\user\"
set BackupDir = "C:\Backup\"
set BackupListFile="C:\Backup\RestoreList.txt"
do EXTSELECT^DBREST(1,0,BackupDir_"back1.cbk",BackupListFile,4,"","")
```

#### 4.8.4.2 ディレクトリとグローバルの選択条件

jdirdglo によって示されたファイルの要件

- ・ リストアするディレクトリのそれぞれのレコードが 1 つずつ格納されます。
- ・ 各フィールドをコンマで区切ります (,)。
- ・ 各レコードの形式は、DirName、RestAll、Globals で、これらの意味は以下のとおりです。

引数	説明
DirName	ジャーナルをリストアするディレクトリの名前。
Globals	ジャーナル・リストアのためにコンマで区切られたグローバルのリスト。RestAll が N の場合のみ使用します。グローバルのリストが大きい場合、次の行に残りのグローバル名を記述できますが、再び、ディレクトリ名に続いて、N およびグローバルのリストを指定する必要があります。
RestAll	ディレクトリ内のすべてのグローバルについてジャーナル・エントリをリストアするか否か。大文字小文字を区別しない、必須の引数です。 Y – このディレクトリ内のすべてのグローバルでジャーナルをリストアします。 N – グローバル・リストにジャーナル・リストアのためのグローバルを指定します。

異なるレコードの例を以下に示します。

```
DUA0:[TEST1],Y
DUA1:[TEST2],N,GLO1,GLO2,GLO3
DUA1:[TEST2],N,GLO4,GLO5,GLO6
DUA1:[TEST3],n
```

そのディレクトリのジャーナルをリストアしたくない場合、最後のレコードも完全に省略できます。

#### 4.8.5 ミラーリングされたデータベースの考慮事項

ミラーリングされたデータベース (ミラーリング・データベース) が InterSystems IRIS バックアップ・リストアのターゲット (対象) である場合、ソース・データベースはターゲット・データベースと一致する必要があります (つまり、同じミラーリング・データベースであることが必要です)。

**注釈** フェイルオーバー・ミラー・メンバで作成されたミラーリング・データベースのバックアップは、そのミラーの任意のメンバ (つまり、他のフェイルオーバー・メンバもしくは非同期メンバ) にリストアできます。非同期メンバで作成されたバックアップからリストアする場合、“[非同期ミラー・メンバからのリストア](#)” を参照してください。

このセクションで、「アクティブなミラーリング・データベース」という語は、現在、ホスト・システムが属するミラーに含まれているデータベースを指します (そのミラーが現在稼働していない場合も含む)。以前にミラーリングされていたデータベースや、ホスト・システムが既に属していないミラーに含まれていたデータベースは該当しません。

ミラーでの InterSystems IRIS バックアップ・リストアの動作は、バックアップを作成したミラー・メンバでリストアしているのか、あるいは別のシステムであるかにより、以下のように異なります。



- ・ プライマリ・フェイルオーバー・メンバー `DBREST` ではプライマリ・フェイルオーバー・メンバのアクティブなミラーリング・データベースをリストアできません。ただし、データベースにリストアの必要がある場合は、ミラーから一時的に削除できます（“[ミラーリングされたデータベースをミラーから削除する](#)”を参照）。ただしデータベースはミラーリングされたデータベースとしてリストアされ（ミラーリングされたデータベースとしてバックアップされたため）、以下の各セクションで説明する機能（ミラーにあるデータベースをアクティブ化するために必要なジャーナル・ファイルの自動リストアなど）が適用されます。
- ・ バックアップ・フェイルオーバー・メンバー・バックアップ・フェイルオーバー・メンバで、アクティブ状態にある、もしくはバックアップのコピーよりも新しいミラーリング・データベースのリストアを行うと、以下のように警告メッセージが表示されます。
  - － フルバックアップ・リストアの場合、それらのデータベースはスキップされます。
  - － 選択的リストアの場合、ターゲットを上書きしていいかどうかを確認されます。

#### 4.8.5.1 ミラーリングされたデータベースのフルバックアップ・リストア

`DBREST` ユーティリティによる対話形式（“[オンライン・バックアップ・リストア・ユーティリティ](#)”を参照）と `EXTALL` エントリ・ポイントによる非対話形式（“[DBREST の外部エントリ・ポイント](#)”を参照）のどちらで実行したかによって、ミラーリングされたデータベース（ミラーリング・データベース）のフル・バックアップ・リストアは以下のように異なります。

- ・ バックアップを作成したシステムでのフルバックアップ・リストアは、ミラーリングされていないシステムに対する場合と同様に行われます。ただし、バックアップのコピーよりも新しい、ローカルのミラーリング・データベース（定義では、現在アクティブなデータベースも含む）はスキップされる点異なります。すべてのデータベースは元の場所へリストアされます。
- ・ バックアップを作成したシステム以外のミラー・メンバでのフルバックアップ・リストアは、既にシステムに存在するミラーリング・データベースのみをリストアします。ただし、バックアップ内のミラーリング・データベースよりも新しい、ターゲット・システムのミラーリング・データベースはスキップされます。
- ・ 新規の最上位のディレクトリを指定するフルバックアップ・リストアでは、データベースの新規パスを生成した後にバックアップ内のデータベースすべてをリストアします。リストアの後、ミラーリング・データベースのコピーが最終的に 2 つになった場合、リストアされたコピーは既にシステム上でアクティブであるためにアクティブ化できないと警告されます。

フルバックアップ・リストア後に、システムは以下を試みます。

1. 必要なジャーナル・ファイルのリストアにより、ミラーリングされたデータベースをアクティブにする。
2. ミラーリング・データベースをアクティブなミラーへリンクする（ミラーが存在し、かつデータベースのコピーは存在しない場合）。

#### 4.8.5.2 ミラーリングされたデータベースの選択的バックアップ・リストア

ミラーリングされたデータベース（ミラーリング・データベース）の選択的バックアップ・リストアは、以下のように異なります。

- ・ バックアップをリストアする際、リストアをミラーリング・データベースのみに制限するかどうかを尋ねられます。ユーザの応答によって以下ようになります。
  - － `[ ]` – ミラーリングされたデータベースのみが表示されます。
  - － `[ ]` – バックアップのデータベースすべての宛先を選択するプロンプトが表示されます。
- ・ 選択的リストアのデータベース選択フェーズの間、バックアップのデータベースのリストが表示され、それらをリストアする場所のパスを指定するよう求められます。ソース・システムにリストアする場合は、パスを入力するか、“X” または “x”（ディレクトリをスキップ）を入力するか、あるいは **Enter** キー（バックアップ内に保存されたパスにリストア）を押します。



バックアップを作成したシステムではないミラー・メンバ上にバックアップをリストアする場合、ミラーリングされたデータベースのディレクトリについて **Enter** キーを押すと、同じミラー・データベース名を持つアクティブなミラーリング・データベースがそのシステム上にあるかどうかによって結果が異なります。

- データベースが存在する場合、**Enter** キーを押すと、既存のデータベースのディレクトリ・パス（そのパスがソース・システムのパスと同じかどうかなど）にかかわらず、リストアされたデータベースで既存のデータベースが上書きされます。
- データベースが存在しない場合、データベースはソース・マシン上にあるディレクトリ・パスにリストアされます。そのパスが存在しない場合は、必要なディレクトリが作成されているかどうか確認する必要があります。データベースがその場所に既にある場合、アクションは不要です。

- ・ 選択的バックアップ・リストアにおいて、以下の状況でミラーリング・データベースがリストアによって上書きされることが検出されると、その続行によってターゲット・データベースが破壊される旨の警告が表示され、続行するかどうかを尋ねられます。

- バックアップのデータベースが現在のデータベースより古い。
- バックアップのデータベースが現在のデータベースのコピーではない（例えば、ミラーリングされていないデータベースや、別のミラーの同じミラー・データベース名のデータベースである場合）。

続行する場合、ターゲット・データベースはミラーから削除され、要求に従い上書きされます。

- ・ 選択的バックアップ・リストアが **EXTSELECT** エントリ・ポイント（“[DBREST の外部エントリ・ポイント](#)”を参照）を介して非対話形式で行われている場合は、ソース・パスおよびターゲット・パスのデータベース選択ファイルは、パス名またはミラー・データベース名のいずれかで指定できます。

ミラーリングされたデータベースの場合、ターゲットを空白にすることでローカル・マシンの該当ミラー・データベースに対するリストアとなります。ターゲットを空白としない場合、データベースは以下のいずれかになります。

- ミラーリングされたデータベースの正規のローカル・コピー。
- ミラーリングされたデータベースのローカル・コピーがシステムにない場合は、ミラーリングされていないデータベース（または、存在しないディレクトリ/データベース）。
- ミラー・データベース名。

**注釈** ターゲットがミラー・データベース名（もしくは空白）で、ミラーリングされたデータベースがローカル・システムに存在しない（もしくはディスクマウントである）場合、バックアップを作成したシステム上でバックアップをリストアする場合に限り、ターゲットは作成されます。バックアップが他のミラー・メンバでリストアされる場合、データベースはスキップされます。

フルバックアップ・リストアと同様に、選択的バックアップ・リストア後に、システムは以下を試みます。

1. 必要なジャーナル・ファイルのリストアにより、ミラーリングされたデータベースをアクティブにする。
2. ミラーリングされたデータベースをアクティブなミラーへリンクする（ミラーが存在する場合）。

### 4.8.5.3 非同期ミラー・メンバからのリストア

非同期ミラー・メンバで作成されたミラーリング・データベースからリストアする際には、以下の点を考慮します。

- ・ 非同期ミラー・メンバで作成されたミラーリング・データベースのバックアップは、同じミラーの任意の非同期メンバに対してリストアできます。
- ・ 非同期ミラー・メンバで作成されたミラーリング・データベースのバックアップは、ミラーリング・データベースの新規コピーを作成する場合に限って使用できます（つまり、フェイルオーバー・メンバの既存のミラーリング・データベース上にリストアすることはできません）。



# 5

## ジャーナリングの概要

グローバル・ジャーナリングは、データベースで実行されるすべてのグローバルな更新処理を記録します。また、バックアップと組み合わせて使用すると、障害またはクラッシュ発生直前の状態にデータベースをリストアできます。

バックアップは物理的なリカバリの基礎ですが、これが常に正しい方法であるとは限りません。バックアップからデータベースをリストアしても、そのバックアップ以後に行われたグローバルな更新は回復されません。バックアップは、物理的な整合性が失われた時点の数時間前に作成されていた場合があります。これらのバックアップ後の更新は、データベースをバックアップからリストアした後に、ジャーナル・ファイルからデータベースにリストアできます。こうして、データベースを最新状態にできます。障害発生時に開いていたトランザクションはロールバックされ、トランスポートの整合性が確保されます。

### 5.1 ジャーナリングの概要

InterSystems IRIS® データ・プラットフォームの各インスタンスはジャーナルを保持します。ジャーナルとは、最後のバックアップ以後にデータベースに対して行われた更新を時間順で保持している一連のファイルです。その処理は冗長性があり、論理的で、InterSystems IRIS ライト・デーモンを使用しません。InterSystems IRIS の[トランザクション処理](#)は、障害の後にデータの論理的整合性を維持するために、ジャーナリングと共に機能します。

また、バックアップとジャーナリングにより、データベースを再作成することが可能になります。データベースが破損する、アクセスできない、または使用不可能になるような障害が発生した場合、最新のバックアップをリストアし、ジャーナル内の変更を適用して、障害発生時点のデータベースを再作成できます。物理的整合性の損失を回復するこの技術は、“ロール・フォワード”リカバリと呼びます。また、ジャーナルは不完全なトランザクションをロールバックするためにも使用されます。

ジャーナル状態は、個々のグローバルのプロパティではなく、データベースのプロパティです。データベースのグローバル・ジャーナル状態は、Yes または No のいずれかになります。既定では、作成するすべてのデータベースがジャーナリングされます ([[グローバルジャーナル状態](#)] が [Yes])。新規にインストールされた InterSystems IRIS インスタンスでは、**IRISAUDIT**、**IRISSYS**、および **USER** の各データベースはジャーナリングされます。**IRISLIB**、**IRISTEMP**、および **IRISLOCALDATA** の各データベースはジャーナリングされません。**IRISTEMP** 内のグローバルに対する操作は、ジャーナリングされません。InterSystems IRIS 一時データベースである **IRISTEMP** に一時[グローバル](#)をマッピングしてください。

**重要**      ジャーナリングされないデータベースのリカバリの制限に関する重要な情報については、“[データベースをジャーナリングしない場合の影響](#)”を参照してください。

InterSystems IRIS を起動すると、最後のライト・デーモン・パス以降のすべてのジャーナル・エントリが再適用されます。ユーザは、ライト・デーモンを使用せずにジャーナルの更新を同時に処理するので、この方法によりクラッシュ前の更新の保存を保証します。

ジャーナリングされるデータベースに対するすべての更新を記録するだけでなく、ジャーナルにはトランザクションの一部であるジャーナリングされないデータベースに対するすべての更新も含まれます (主 Set 操作と Kill 操作)。これによって、システムの信頼性が大きく向上します。つまり、回復後、グローバルの更新の中にジャーナルに記録されていないものがあつたり、トランザクションの対象にならないものがあつたりするために発生する矛盾を防止できます (ローカルでの Set および Kill の操作、およびプロセス・プライベート変数はジャーナルに記録されません)。

クラスタにマウントされたデータベースのグローバル処理をジャーナリングするかどうかは、データベース設定によって決まります。ローカル InterSystems IRIS インスタンスでは、リモート・ノードのグローバルのトランザクション処理はジャーナルしません。ネットワーク構成の場合、ジャーナリングは、グローバルが実際に格納されているノードをサポートしますが、Set または Kill コマンドを要求したノードはサポートしません。したがって、ノード B がノード A の要求で Set コマンドを実行する場合、ジャーナル・エントリは、ノード A ではなく、ノード B のジャーナルに表示されます。

## 5.2 ジャーナリングとライト・イメージ・ジャーナリングの違い

InterSystems IRIS ジャーナルとライト・イメージ・ジャーナル (WIJ) とを混同しないでください。WIJ の詳細は、“[ライト・イメージ・ジャーナリングとリカバリ](#)” を参照してください。ジャーナリングとライト・イメージ・ジャーナリングでは、以下のよう  
に機能が異なります。

- ・ ジャーナリングでは、データベースに既書き込まれているデータベースの変更がすべて記録されます。一部の変更が (例えば、リカバリされたデータベースの最新のバックアップ後に行われたために) 消失した場合は、ジャーナル・ファイルの内容をリストアすることで、データベースに変更をリストアします。
- ・ ライト・イメージ・ジャーナリングでは、データベースにまだ書き込まれていないデータベースの変更がすべて記録されます。システム・クラッシュが発生すると、ライト・イメージ・ジャーナルの内容は、再起動時に自動的にデータベースに書き込まれます。

## 5.3 データベースの整合性の保護

インターシステムズのリカバリ・プロセスは、最大限の保護機能を提供するよう設計されています。

- ・ これには、“ロール・フォワード” 技術が使用されています。システム障害が発生した場合、実行途中の更新はリカバリ機能によって最後まで実行されます。対照的に、他のシステムの中には、回復のために更新を元に戻す、“ロール・バック” 方法を採用するシステムもあります。どちらの方法も内部整合性を保護しますが、InterSystems IRIS は、ロール・フォワード技術を使用することで、データの損失を減少させます。
- ・ 更新順序を保護します。リカバリ後のデータベースに更新が存在する場合、先行するすべての更新も実行されます。更新順序を適切に保護しない他のシステムは、内部的な一貫性を保っていますが、論理的には無効なデータベースになります。
- ・ データベースだけでなく、インクリメンタル・バックアップ・ファイル構造を保護します。クラッシュ後のリカバリに続き、有効なインクリメンタル・バックアップを実行できます。

## 5.4 トランザクションの自動ジャーナリング

InterSystems IRIS アプリケーションでは、トランザクションと呼ばれる作業ユニットを定義できます。InterSystems IRIS のトランザクション処理では、ジャーナルを使用してトランザクションを格納します。InterSystems IRIS では、影響を受ける

グローバルが含まれるデータベースのグローバル・ジャーナル状態設定に関係なく、トランザクションに含まれるグローバルの更新がすべてジャーナルされます。

以下の目的で、コマンドを使用します。

- ・ トランザクションの開始を指示
- ・ トランザクションが通常に終了した場合、トランザクションをコミット
- ・ トランザクション中にエラーが発生した場合、トランザクションをロールバック

InterSystems IRIS は、多くの SQL トランザクション処理コマンドをサポートします。これらのコマンドの詳細は、“[トランザクション処理](#)”を参照してください。

## 5.5 不完全トランザクションのロールバック

トランザクションが完了していない場合、ジャーナル・エントリを使用してトランザクションはロールバックされ、関連するグローバルはトランザクション実行前の値に戻ります。InterSystems IRIS は、データベース更新の一環として、ジャーナルに変更を適用してジャーナル・リストアを実行し、不完全なトランザクションをロールバックします。これは以下の条件で発生します。

- ・ システム・クラッシュ後の InterSystems IRIS 起動で発生するリカバリ中
- ・ トランザクション進行中にプロセスを一時停止した場合
- ・ 管理ポータル の [プロセス] ページ ([システム処理]→[プロセス]) から [終了] オプションを使用してプロセスを終了した場合。Job コマンドによって開始されたプロセスを終了した場合、その中の不完全なトランザクションは自動的にロールバックされます。ユーザ・プロセスを終了すると、システムはユーザにメッセージを送信し、不完全なトランザクションをコミットするかロールバックするかを尋ねます。

アプリケーションにロールバック・コードを記述できます。アプリケーションそのものが問題を検出し、ロールバックを要求します。普通、これはアプリケーション・エラーに続くエラー処理ルーチンで実行します。

詳細は、“[アプリケーションでのトランザクション管理](#)”を参照してください。

## 5.6 データベースをジャーナリングしない場合の影響

ジャーナリングされないデータベースは、InterSystems IRIS の起動時にジャーナル・リカバリおよびトランザクション・ロールバックに参加しません。結果として、障害、バックアップ・リストア、および再起動後に以下の条件が適用されます。

- ・ ジャーナリングされないデータベースに対する最新の更新が失われる可能性があります。これらのデータベース内のデータは、ジャーナリングされたデータベース内のデータより前の時点を表します。
- ・ ジャーナリングされないデータベース内のトランザクションまたはトランザクションの一部が、部分的にコミットされた状態になる可能性があります。[同期コミット・トランザクション](#)により提供される持続性は、ジャーナリングされないデータベースには適用されません。
- ・ トランザクションの一部であるジャーナリングされないデータベースに対する更新がジャーナリングされると、これらのジャーナル・レコードは通常の操作中にトランザクションをロールバックするためにのみ使用されます。これらのジャーナル・レコードは、起動時にトランザクションをロールバックする手段にはならず、起動時またはバックアップ・リストア後にデータを回復するために使用することもできません。

## 5.7 ジャーナル書き込みサイクル

ジャーナル・バッファの内容をジャーナル・ファイルに書き込む操作は、ジャーナル同期と呼ばれます。ジャーナル同期により、現在ジャーナル・バッファ内にあるすべての操作が現在のジャーナル・ファイルに確実に書き込まれます。

ジャーナルが同期される頻度は、関連する InterSystems IRIS インスタンスの動作状況によって異なります。ジャーナル同期は、次の場合にトリガされる可能性があります。

- ・ システムがアイドルの場合 2 秒ごとに 1 回トリガされる。
- ・ ECP ベースの分散キャッシュ・クラスタでは、アプリケーション・サーバからの特定の要求 (例えば、\$Increment) に応答して ECP のセマンティックスを保証する場合にデータ・サーバによってトリガされる。
- ・ InterSystems IRIS トランザクションを使用している場合、TCOMMIT によってトリガされる (同期コミット・モードであるため、このトランザクションに関連するデータがディスクにフラッシュされる)。
- ・ 各データベース書き込みサイクルの一部として書き込みデーモンによってトリガされる。
- ・ ジャーナル・バッファがフルの場合にトリガされる。

## 5.8 ジャーナル・ファイルとジャーナル履歴ログ

ジャーナル・ファイルは プライマリ・ジャーナル・ディレクトリ (既定では、install-dir¥Mgr¥journal) に格納され、ジャーナル履歴ログ・ファイルである install-dir¥Mgr¥journal.log に記録されます。このログ・ファイルには、インスタンスで管理されているすべてのジャーナル・ファイルのリストが含まれています。ジャーナルに関連するすべての機能、ユーティリティ、および API では、このログを使用してジャーナル・ファイルを探します。

ジャーナル履歴ログ・ファイルは以下のように更新されます。

- ・ 新規のジャーナル・ファイルが作成されると、エントリがログに追加されます。
- ・ エントリで識別されるジャーナル・ファイルが存在しなくなっていて、エントリの経過日数が 30 日以上の場合、エントリは定期的にファイルの先頭から順に削除されます。エントリがこの 2 つの条件に該当しない場合、削除は行われません。

**注意** journal.log ファイルは変更しないでください。ジャーナル・ユーティリティを使用せずにこのファイルを変更すると、ファイルは破損していると思われるため、ジャーナリングが無効になる可能性があります。このファイルが破損している場合は、[インターシステムズのサポート窓口](#)までお問い合わせください。ジャーナリングが無効になっている場合 (InterSystems IRIS が journal.log ファイルを更新できない場合) は、破損したログ・ファイルの名前を変更してジャーナリングを再開します。

バックアップからのリストアに続いて実行するジャーナルのリストアで必要になったときに必ず利用できるように、journal.log ファイルをバックアップ方針に取り入れることをお勧めします。バックアップとリストアの方針および手順に関する詳細は、["バックアップとリストア"](#) を参照してください。

journal.log ファイルが存在しない場合 (例えば、破損によりファイル名を変更した場合)、システムでは新規ジャーナル・ファイル作成の際に新しいログ・ファイルが作成されますが、このログ・ファイルにはその作成時点以後に作成されたジャーナル・ファイルのみが記録されるので、以前のジャーナル・ファイルについての情報は失われます。journal.log ファイルを使用するジャーナル関連の機能、ユーティリティ、および API では、このログ・ファイルに記録されていないジャーナル・ファイルは利用できません。一方、ジャーナルをリストアするとき、journal.log ファイルが存在しない場合または既存のログ・ファイルを使用しない場合は、手動でジャーナル・ファイルを指定できます ("[JRNRESTO を使用したジャーナル・ファイルからのグローバルのリストア](#)" を参照してください)。



また、`journal.log` ファイルを使用して、以下のように別の場所へのジャーナル・ファイルの移行やリストアも可能です。

1. ターゲットとしている InterSystems IRIS インスタンスで、`install-dir¥Mmgr` ディレクトリ以外の場所に、ジャーナル・ファイルおよび `journal.log` ファイルをコピーします。
2. ターゲット・システム上で `^JRNRESTO` ルーチンを実行し、ジャーナル・ファイルの元のパスに関するプロンプトへの応答で「No」を入力します。
3. 表示されるプロンプトに従って、コピーしたジャーナル・ファイルおよび `journal.log` ファイルのターゲット・システム上での格納場所を指定します。`^JRNRESTO` では、このログ・ファイルを使用して、ターゲット・システムへ移行またはリストアするジャーナル・ファイルの範囲を検証します。
4. “[^JRNRESTO を使用したジャーナル・ファイルからのグローバルのリストア](#)” の説明に従って、プロセスを完了します。

注釈 InterSystems IRIS インスタンスがミラーのメンバとなった場合、以下のミラーリング・サポート用のジャーナル変更が発生します。

- ・ プライマリになる際、`MIRROR-mirror_name` が先頭に付く新規ジャーナル・ファイル (`MIRROR-MIR21-20230921.001` など) に対してのジャーナル切り替えが発動します。その時点より、すべてのジャーナル・ファイルはミラー・ジャーナル・ファイルとして書き込まれ、`mirrorjrn-mirror_name.log` (`mirrorjrn-MIR21-20230921.log` や `journal.log` など) にログが記録されます。
- ・ バックアップまたは非同期になる際には、プライマリより受信したミラー・ジャーナル・ファイルがローカル・インスタンスの標準ジャーナル・ファイルと共に、構成されたジャーナル・ディレクトリに書き込まれて、プライマリのミラー・ジャーナル・ログ (`mirrorjrn-mirror_name.log`) のコピーが `install-dir¥Mmgr` に作成された後に、継続的に更新されます。

ミラーリングにおけるジャーナル・ファイルの役割の詳細は、“[ミラー同期](#)” を参照してください。

## 5.9 一時グローバルと IRISTEMP の使用

IRISTEMP データベースにマップされているものはジャーナルされません。

ネームスペース内のグローバルはそれぞれ別々のデータベースにマッピングされている可能性もあるため、一部はジャーナルに記録され、他は記録されないということもあります。グローバルがマップされているデータベースのジャーナル・プロパティによって、InterSystems IRIS がそのグローバル処理をジャーナルするかどうかが決まります。**IRISTEMP** とジャーナル・プロパティが **[No]** に設定されているデータベースとの違いは、**IRISTEMP** では、トランザクションの更新も含めて何もジャーナルされないことです。

注釈 データベースの構成で **[ジャーナルグローバル]** プロパティを **[いいえ]** に設定すると (“[ローカル・データベースの作成](#)” を参照)、ジャーナル・トランザクションのグローバル Set/Kill 処理のジャーナルが継続されます。これにより、ジャーナル・ファイルが非常に大きくなる可能性があります。しかし、**IRISTEMP** はジャーナル・トランザクション中であっても、Set/Kill 処理のジャーナルを行いません。

ジャーナリングから新規 `z/Z*` グローバルを除外する必要がある場合は、それらのグローバルを、ジャーナル・プロパティを **[No]** に設定してデータベースにマップします。ジャーナリングから `z/Z*` グローバルを常に除外するには、すべてのネームスペース内の `z/Z*` グローバルを **IRISTEMP** データベースにマップする必要があります。

“[一時グローバルと IRISTEMP データベース](#)” も参照してください。



## 5.10 ジャーナル管理のクラスとグローバル

ジャーナリングで使えるメソッドとクエリの詳細は、“%SYS.Journal.System” クラスに関するドキュメントを参照してください。これは %SYS.Journal パッケージに収められています。

また、InterSystems IRIS では、%SYS(“JOURNAL”) グローバル・ノードを使用してジャーナル・ファイルに関する情報を格納します。以下に例を示します。

- ・ %SYS(“JOURNAL”, “ALTDIR”) は、代替ジャーナル・ディレクトリ名を格納します。
- ・ %SYS(“JOURNAL”, “CURDIR”) は、現在のジャーナル・ディレクトリ名を格納します。
- ・ %SYS(“JOURNAL”, “CURRENT”) は、ジャーナルの状態とジャーナル・ファイル名を格納します。

この情報は、管理ポータルの [グローバル] ページ ([システム・エクスプローラ]→[グローバル]) で表示できます。

## 5.11 関連項目

- ・ [ジャーナリングの構成](#)
- ・ [基本的なジャーナリング操作](#)
- ・ [ジャーナリング・ユーティリティ](#)
- ・ [ジャーナル入出力エラー](#)
- ・ [ジャーナリングに関する特別な考慮事項](#)

# 6

## ジャーナリングの構成

ここでは、ジャーナリングを有効化および構成する方法を説明します。

### 6.1 ジャーナリングの有効化

既定では、IRISSYS、IRISAUDIT、および USER の各 InterSystems IRIS® データ・プラットフォーム・データベースのジャーナリングが有効になります。管理ポータルの [ローカルデータベース] ページ ([システム管理]→[構成]→[システム構成]→[ローカルデータベース]) からデータベースごとにジャーナリングを有効または無効にすることができます。目的のデータベースに該当する行で [編集] をクリックし、[グローバル・ジャーナル状態] ボックスで [Yes] または [No] をクリックします。

新しいデータベースのジャーナル状態の既定の設定は Yes です。旧リリースの InterSystems IRIS のデータベースを初めてマウントするときに、この値は Yes に設定されます。新規グローバルの以前の設定値およびそのデータベースにある個々のグローバルの以前の設定は無視されます。

実行中のシステムで、データベースのグローバル・ジャーナル設定を変更できます。これを実行すると、何らかの影響が発生する可能性があることが警告されます。また、監査が有効になっていると変更に対する監査が実行されます。

### 6.2 ジャーナル・ファイル名とロールオーバー

ジャーナル・ファイルの名前は、オプションのユーザ定義の接頭語、その後続く yyyyymmdd の形式で作成された日付と時刻で構成される基本名、ピリオド (.), そして最後に 1 暦日以内に作成されたジャーナル・ファイルをインクリメンタルに番号付けするために使用される 3 桁の接尾語で構成されます。ジャーナル・ファイルが一杯になると、システムは、同じ接頭語と基本名を使用し、接尾語を 1 だけ増加させた新しいファイルに自動的に切り替えます。基本名は、ジャーナル・ファイルの使用中に新しい暦日が始まった場合にのみ変更されます。

例えば、2023 年 4 月 27 日にアクティブである最初のジャーナル・ファイルには、**20230427.001** という名前が付けられます。ファイルが一杯になると、システムは **20230427.002** という新しいファイルを開始します。このプロセスは、ジャーナル・ファイルのロールオーバーと呼ばれます。ただし、ジャーナル・ファイルの使用中に真夜中を過ぎて日付が変わると、そのファイルの名前は **20230428.001** に変更されます。

## 6.3 ジャーナリングの最善の使用法

ジャーナリングを計画および構成する際の重要な考慮事項を以下に示します。

- ・ ジャーナル・ファイルは、プライマリ・ジャーナル・ディレクトリと代替ジャーナル・ディレクトリ(プライマリ・ディレクトリが何らかの理由で書き込み不能になった場合に使用) の両方に保存されます。

パフォーマンスと復元可能性の両方を高めるために、プライマリおよび代替ジャーナル・ディレクトリは、それぞれ異なるストレージ・デバイスに配置し、データベースやライト・イメージ・ジャーナル (WIJ) が使用しているストレージ・デバイスとも異なるストレージ・デバイスに配置することをお勧めします。運用上の理由から、同じ SAN (Storage Area Network) 上の論理ユニット番号 (LUN) が異なるデバイスに配置することもできますが、一般的にはなるべく独立したデバイスに配置するようにします。物理ドライブの個別のセットを使用することを強くお勧めします。このようにデータベース/WIJ ストレージとプライマリ/代替ジャーナル・ストレージを分離することには、以下の主要な利点があります。

- データベースまたは WIJ に発生する可能性がある障害からジャーナル・ディレクトリを分離することにより、そのような障害後にデータベースをリストアするためにジャーナル・ファイルを使用できるようになります。
- プライマリおよび代替ジャーナル・ディレクトリを分離することで、プライマリ・ディレクトリが配置されたデバイスで停止が発生した場合も、ジャーナリングを続行できます。
- ジャーナル入出力パスを分離することは、ほとんどのアプリケーションで必要となる入出力並行処理を実現するための重要な要素です。

簡便性のために、InterSystems IRIS インストールではディレクトリ `install-dir¥Mgr¥journal` を作成し、それをプライマリおよび代替ジャーナル・ディレクトリとして構成して、そのディレクトリ内に既定のジャーナリングされるデータベースの最初のジャーナル・ファイルを作成します。ただし、プライマリおよび代替ジャーナル・ディレクトリ用に別個のストレージ・デバイスを特定して準備し、インストール後できるだけ早くにこれらの設定を再構成する ("[ジャーナル設定の構成](#)") の説明に従って) ことをお勧めします。

**注釈** ジャーナル・ファイルは、"[バックアップとリストア](#)" の説明に従って、常にデータベース・ファイルと共にバックアップしておく必要があります。プライマリ・データ・センタにおける複数のストレージ・デバイスを含む障害からの復旧が可能になるよう、災害復旧目的でジャーナル・ファイルをオフサイトで複製することを検討してください。この目的には、[InterSystems IRIS ミラーリング](#)、ディスクレベル複製、またはその他のメカニズムを使用できます。

- ・ ジャーナリングがすべてのデータベースに対して有効化されていることを確認します (一時的なデータのみを含むデータベースを除く)。

**重要** ジャーナリングされないデータベースのリカバリの制限に関する重要な情報については、"[データベースをジャーナリングしない場合の影響](#)" を参照してください。

- ・ ジャーナルの **[エラー発生時に凍結する]** オプションを **[はい]** に設定することを検討してください。障害によりプライマリおよび代替ジャーナル・デバイスの両方に書き込みできなくなった場合、この設定によりシステムが凍結し、ユーザがシステムを使用できなくなるため、データが失われることがなくなります。もう 1 つの選択肢として、**[エラー発生時に凍結する]** オプションを **[いいえ]** に設定し、システムを続行させてジャーナリングを無効にします。これにより、システムは引き続き利用可能になりますが、データの整合性と回復性が損なわれます。**[エラー発生時に凍結する]** の詳細については、"[ジャーナル入出力エラー](#)" を参照してください。
- ・ バックアップ検証手順で、ジャーナル・ファイルが最後の良好なバックアップの前に閉じたと判断される場合以外は、削除しないでください。ジャーナル・ファイルを保持する日数および、ジャーナル・ファイルに保持する成功したバックアップの回数を適切に設定します。
- ・ ジャーナル・リストア時に最適なパフォーマンスを実現するには、共有メモリ・ヒープ (gmheap) のサイズを増加させることを検討してください。詳細は、"[ジャーナル・ファイルのリストア](#)" を参照してください。

- ・ ジャーナル・ファイルが占有する領域を最小限に抑えるために、ジャーナル・ファイルの圧縮が有効になっていることを確認します。このオプションは、[CompressFiles](#) CPF 設定で制御され、新規の InterSystems IRIS インストールでは既定で有効になっています。
- ・ ジャーナル・ファイルのアーカイブも検討してください。ジャーナリング用のアーカイブ・ターゲットを指定した場合、ファイルが最大サイズに達すると、ジャーナル・ファイルはその場所にコピーされ、新しいジャーナル・ファイルが開始されます。元のジャーナル・ファイルをアーカイブにコピーした後に削除するよう、ジャーナリングを構成することもできます。

## 6.4 ジャーナル・アーカイブ・ターゲットの構成

InterSystems IRIS ジャーナル・ファイルのアーカイブ・ターゲット([ジャーナル設定の構成](#)時に使用できる)を構成するには、管理ポータルの [Archive Targets] ページに移動します ([システム管理]→[構成]→[システム構成]→[Archive Targets])。

このページで、アーカイブ・ターゲットを表示、編集、または削除できます。

このページには、既存のすべてのアーカイブ・ターゲットがリストされます。フィールドは以下のとおりです。

- ・ **[名前]** – ファイルをアーカイブするときに使用する、アーカイブ・ターゲットの一意の名前。
- ・ **[タイプ]** – ファイルをこのターゲットの場所にコピーする方法を指定します。次のいずれかを指定できます。
  - 場所が S3 の場合は **s3**。
  - 場所がオンプレミスの場合は **rsync**。場所が Windows ディレクトリの場合、実際には robocopy が使用されますが、わかりやすくするために、このページではタイプを **rsync** と表示しています。
- ・ **[場所]** – 実際のディレクトリを指定します。形式は、場所が S3 かオンプレミスかによって異なります。
  - S3 の場合、**[場所]** は以下の形式になります。

```
s3://directoryname/
```

以下に例を示します。

```
s3://test-dir/journal-archives/
```

- UNIX® の場合、**[場所]** は以下のいずれかの形式になります。

```
/directoryname/
```

または

```
server:/directoryname/
```

- Windows の場合、**[場所]** は以下のいずれかの形式になります。

```
\\server\share\directoryname\
```

または

```
drive:\directoryname\
```

アーカイブ・ターゲットを作成するには、[Create New Archive Target] をクリックし、上記のように値を指定して [保存] をクリックします。アーカイブ・ターゲットを作成または編集する際、[タイプ] の選択肢は [AWS S3] と [rsync or robocopy] です。後者の場合、指定された場所の形式に基づいて、適用可能なオプションが使用されます。

アーカイブ・ターゲットを編集するには、ターゲットの [編集] リンクをクリックし、必要に応じて値を変更して [保存] をクリックします。

アーカイブ・ターゲットを削除するには、ターゲットの [削除] リンクをクリックし、[OK] をクリックして確定します。

## 6.5 ジャーナル設定の構成

InterSystems IRIS のジャーナリングを構成するには、管理ポータルの [ジャーナル設定] ページ ([システム管理]→[構成]→[システム構成]→[ジャーナル設定]) に移動します。

以下の設定が編集可能です。

- ・ **[主ジャーナルディレクトリ]**—ジャーナル・ファイルを格納するディレクトリの名前を入力します。ディレクトリ名の最大長は 214 文字です。
- ・ **[二次ジャーナルディレクトリ]**—ジャーナリングの際に、現在のディレクトリのディスクが何らかの理由で書き込みできなくなったときに使用する代替ディレクトリの名前を入力します。([“ジャーナル・ディレクトリの切り替え”](#)の説明に従って、ジャーナル・ディレクトリを手動で切り替えることもできます。)ディレクトリ名の最大長は 214 文字です。

**重要** パフォーマンスと復元可能性の両方を高めるために、プライマリおよび代替ジャーナル・ディレクトリは、それぞれ異なるストレージ・デバイスに配置し、データベースやライト・イメージ・ジャーナル (WIJ) が使用しているストレージ・デバイスとも異なるストレージ・デバイスに配置することをお勧めします。詳細は、[“ジャーナリングの最善の使用法”](#) を参照してください。

- ・ **[次のサイズに達すると新規ジャーナルに切り替える]**—ジャーナル・ファイルの最大サイズをメガバイト単位で入力します。このサイズに到達すると次のジャーナル・ファイルに切り替わります。既定のサイズは 1024 MB です。最大サイズは、4096 - 1 - (ジャーナル・バッファのサイズ) MB に制限されます (以下の [“jrnbufs”](#) を参照)。既定のジャーナル・バッファ・サイズ 64 MB を使用した場合、ジャーナル・ファイルの最大サイズは 4031 MB になります。ジャーナル・バッファの最小サイズは 16 MB のため、ジャーナル・ファイルの絶対最大サイズは 4079 MB になります。
- ・ **[ジャーナルファイル接頭語]** (オプション)—ジャーナル・ファイル名の英数字接頭語を入力します。
- ・ **[Archive journal files]** (オプション)—[To Archive Target] のドロップダウン・リストから値を選択するか、[Create New Archive Target...] をクリックして新しい [アーカイブ・ターゲット](#) を定義します。
- ・ **[ジャーナルファイルを削除するタイミング]**—システムで不要になったジャーナル・ファイルを削除するタイミングを指定します。
- ・ この一連のオプションに対して、アーカイブ・ターゲットを指定した場合、ページにはオプションの **[As soon as they are copied to archive]** チェック・ボックスが表示されます。このチェック・ボックスにチェックを付けた場合、次の 2 つのオプションは無視されます (グレー表示されます)。

アーカイブ・ターゲットを指定しない場合、または **[As soon as they are copied to archive]** にチェックを付けない場合は、次の 2 つのオプションのいずれかまたは両方を設定できます。両方の設定にゼロ以外の数値を入力すると、ジャーナル・ファイルは、どちらかの条件を先に満たしたときに削除されます。一方を 0 (ゼロ) に設定し、もう一方をゼロ以外の値に設定すると、ゼロ以外の設定によって削除が判定されます。両方が 0 の場合、ジャーナル・ファイル (およびジャーナル履歴) の自動削除は行われません。

- ・ **[この日数後]**—何日後に削除するかを入力します (有効な値は 0 ~ 100)。

注釈 ジャーナル・ファイルを削除するまでの日数を設定する場合、削除期限の前の日の最後のジャーナル・ファイルも保持されます。例えば、[この日数後] が 1 に設定され、削除が 4 月 1 日の午前 0 時に実行されると、3 月 30 日に作成された最後のジャーナル・ファイルが、3 月 31 日に作成されたファイルと共に保持されます。

- [この回数のバックアップ成功後] – バックアップが何回連続して成功した後に削除するかを入力します (有効な値は 0 ~ 10)。

これには、オンライン・バックアップ、`$$BACKUP`DBACK("","E")` を使用した外部バックアップ、または `Backup.General.ExternalSetHistory()` メソッドを使用したバックアップ履歴への追加が含まれます。

注釈 [この日数後] を 0 (時間を基準にして削除しない) に設定し、[この回数のバックアップ成功後] を 1 に設定している場合は、正常なバックアップを 2 回実行するまでジャーナル・ファイルは削除されません。つまり、“連続” の条件を満たすには、2 回の正常なバックアップが必要です。

`^JRNOPTS` ルーチンを使用するか、または `^JOURNAL` ルーチン・メニューからオプション 7 の [ジャーナル・プロパティの編集] を選択することで、これらの設定を更新することもできます。詳細は、[“^JRNOPTS を使用したジャーナル設定の更新”](#) を参照してください。

注釈 ジャーナル・ファイルは、削除の設定の条件を満たしていても保持される場合があります。この場合、イベントはメッセージ・ログに記録され、理由 (例えば、ジャーナル・ファイルに開いているトランザクションが含まれているなど) が示されます。

- [エラー発生時に凍結する] – この設定は、ジャーナルへの書き込み中にエラーが発生した場合の動作を制御します。既定値は、[いいえ] です (選択されていません)。この設定の詳細は、[“ジャーナル入出力エラー”](#) を参照してください。

注釈 InterSystems IRIS インスタンスがミラーのプライマリ・フェイルオーバー・メンバの場合 ([“ミラーリング”](#) を参照)、インスタンスの [エラー発生時に凍結] の構成が自動的にオーバーライドされ、現在の設定に関係なくジャーナル入出力エラー発生時にすべてのジャーナリングされたグローバル更新がフリーズされます。現在の設定が [いいえ] の場合、インスタンスがプライマリ・フェイルオーバー・メンバでなくなったときに動作がこの設定に戻ります。

- [ウェブセッションをジャーナルする] – この設定によって、Web サーバ・ページ・セッションのジャーナリングを有効にするかどうかを制御されます。既定値は、[いいえ] です (選択されていません)。
- [Compress journal files] – この設定によって、ジャーナル・ファイルを圧縮するかどうかを制御されます。既定値は、[はい] です (選択されています)。詳細は、[“CompressFiles”](#) を参照してください。

注釈 ジャーナルの圧縮とジャーナルの暗号化 ([“ENCRYPT`JOURNAL を使用したジャーナルの暗号化”](#) を参照) の両方が有効な場合、ジャーナル・データは常に、圧縮されてから暗号化されます。暗号化されたデータは圧縮されることはありません。

- [ライト・イメージ・ジャーナル・エントリ] – ライト・イメージ・ジャーナル (WIJ) ファイルの場所を入力します。この設定の詳細は、[“ライト・イメージ・ジャーナリング”](#) を参照してください。
- [WIJ のターゲットサイズ (MB) (0=指定なし)] – WIJ ファイルのターゲット・サイズを入力します。

注釈 このページの設定はすべて、インスタンスの `iris.cpf` ファイルに含まれています。ジャーナル設定の詳細は [“\[Journal\]”](#)、WIJ 設定の詳細は [“ライト・イメージ・ジャーナル \(WIJ\) 設定”](#) および [“targwijsz”](#) と [“wijdir”](#) を参照してください。

特に指示がない限り、これらの設定のほとんどでは変更後に InterSystems IRIS を再起動する必要はありませんが、どの設定を変更した場合でも新しいジャーナル・ファイルが開始されます。



以下に示す 2 つの追加構成設定は、ジャーナリングに影響します。

- ・ **[jrnbufs]** – ジャーナル・バッファに割り当てるメモリの量を指定します。既定値は 64 MB で、最大値は 1024 MB です。最小値は Unicode インスタンスの場合は 16 MB、8 ビット・インスタンスの場合は 8 MB です。この設定値を増やすと ("[jrnbufs](#)" を参照)、メモリに保管できるジャーナル・データの容量が増大するため、ジャーナリングのパフォーマンスが向上しますが、システム障害時に失われる可能性があるジャーナル・データの最大容量も増大します。データは最後のジャーナル同期の後にバッファに書き込まれているためです ("[ジャーナル書き込みサイクル](#)" を参照)。この設定および前述の **[次のサイズに達すると新規ジャーナルに切り替える]** 設定の合計サイズは 4 GB に制限されているため、jrnbufs を増やすとジャーナル・ファイルのサイズが縮小する可能性があります。

[jrnbufs] の設定を変更するには、管理ポータルの **[メモリ詳細設定]** ページに移動します ([**システム管理**]→[**構成**]→[**追加の設定**]→[**詳細メモリ**])。[jrnbufs] の設定は、iris.cpf ファイルを編集することで変更することもできます。詳細は、"[構成パラメータ・ファイル・リファレンス](#)" の "[jrnbufs](#)" を参照してください。

- ・ **[SynchCommit]** – TCOMMIT コマンドがトランザクションに関連するジャーナル・データのディスクへのフラッシュを要求するタイミングを指定します。この設定を **true** に設定すると、ジャーナル・データの書き込み処理が完了するまで TCOMMIT は待機します。**false** (既定値) に設定すると、TCOMMIT は書き込み処理の完了を待機しなくなります。

[SynchCommit] の設定を変更するには、管理ポータル **[互換性設定]** ページに移動します ([**システム管理**]→[**構成**]→[**追加設定**]→[**互換性**])。SynchCommit の詳細は、"[SynchCommit](#)" と [TCOMMIT](#)" を参照してください。

## 6.6 関連項目

- ・ [ジャーナリングの概要](#)
- ・ [基本的なジャーナリング操作](#)
- ・ [ジャーナリング・ユーティリティ](#)
- ・ [ジャーナル入出力エラー](#)
- ・ [ジャーナリングに関する特別な考慮事項](#)



# 7

## 基本的なジャーナリング操作

ここでは、ジャーナリングに関連する基本的なタスクについて説明します。

### 7.1 ジャーナリングの開始

ジャーナリングが停止しているときは、`JRNSTART` ルーチンを使用するか、または `JOURNAL` ルーチン・メニューからオプション 1 の [ジャーナリングの開始] を選択して、ジャーナリングを開始できます。詳細は、“[JRNSTART を使用したジャーナリングの開始](#)” を参照してください。

注釈 管理ポータルからジャーナリングは開始できません。

ジャーナリングを開始すると、監査が有効になっている場合は InterSystems IRIS® データ・プラットフォームによって変更に対する監査が実行されます。

### 7.2 ジャーナリングの停止

システム全体のジャーナリングを停止すると、“[ジャーナルの \[エラー時に凍結\] 設定が \[いいえ\] の場合](#)” で説明するように、多くの望ましくない結果が発生します。

ジャーナルを停止すると、トランザクション処理も中断します。ジャーナリングを停止したときにトランザクションが進行中の場合、ジャーナルには不完全なトランザクションが格納されます。この問題を避けるため、ジャーナルを停止する前に、すべてのユーザがシステムから切断されていることを確認してください。

ジャーナリングを停止した状態で InterSystems IRIS がクラッシュすると、ジャーナリングを停止する前に始まっていたトランザクションが不完全な状態になっていても、起動時のリカバリ・プロセスではそのトランザクションはロールバックされません。これは、トランザクションはコミットされていてジャーナリングされていないからです。

一方、ジャーナル・ファイルを切り替えても、トランザクションには影響がありません。ジャーナルの切り替えで複数のジャーナル・ファイルが生成され、それらにトランザクションが分散しても、ロールバックは正確に実行されます。したがって、可能であれば、ジャーナリングを停止するよりもジャーナル・ファイルを切り替えることをお勧めします。

ジャーナリングを停止するには、`JRNSTOP` ルーチンを使用するか、または `JOURNAL` ルーチン・メニューからオプション 2 の [ジャーナリングの停止] を選択します。詳細は、“[JRNSTART を使用したジャーナリングの停止](#)” を参照してください。

注釈 管理ポータルからジャーナリングは停止できません。

ジャーナリングを停止すると、監査が有効になっている場合は InterSystems IRIS によって変更に対する監査が実行されます。

## 7.3 ジャーナル・ファイルの表示

ジャーナル・ファイルは、管理ポータルの [ジャーナル] ページ ([システム処理]→[ジャーナル]) で表示できます。

1. [ジャーナル] ページでは、必要に応じて [フィルタ] ボックスを使用して、ジャーナル・ファイルのリストの件数を絞り込むことができます。

インスタンスがミラー・メンバとして構成されている場合、既定では、ミラーリングされるものとミラーリングされないものを含めて、すべてのジャーナル・ファイルが表示されます。必要に応じて、ミラー名を含むリンク (例: **Mirror Journal Files Of 'MUNDANE'**) をクリックして、ミラー・ジャーナル・ファイルのみのリストを表示します。インスタンスが複数のミラーのレポート非同期メンバとして構成されている場合、ミラーごとにジャーナル・ファイル用の個別のリンクがあります。すべてのジャーナル・ファイルの表示に戻るには、[すべてのジャーナル・ファイル] リンクをクリックします。

注釈 ミラー・ジャーナル・ファイルの詳細は、“[ジャーナル・ファイルとジャーナル履歴ログ](#)” および “[ミラー同期](#)” を参照してください。

2. ジャーナル・ファイルを表示するには、表示するジャーナル・ファイルの行で [表示] をクリックします。ジャーナル・ファイルは、[ジャーナル表示] ページにレコードごとに表示されます。以下のフィールドがあります。

- ・ **オフセット** – ジャーナル・ファイル内のレコードのオフセット番号。
- ・ **時間** – ジャーナル・レコードを含むバッファが作成された時間 (レコードに記述された操作が発生した時間ではありません)。
- ・ **プロセス** – レコードを作成したプロセスの ID。
- ・ **タイプ** – レコードに記述された操作のタイプ。([タイプ] 列に示される値についての情報は、“[JRNDUMP を使用したジャーナル・レコードの表示](#)” の “[ジャーナル・ファイルの処理](#)” テーブルで提供されています。)
- ・ **トランザクション中** – 操作がトランザクションの一部として発生したかどうか。
- ・ **グローバルノード** – 操作によって変更されたグローバル・ノード。
- ・ **データベース** – 変更が発生したデータベース。

3. 以下を実行できます。

- a. レコードの [オフセット] 列をクリックし、レコードの詳細を含むダイアログ・ボックスを表示します。
- b. 以下の項目ごとにレコードを色分けするかどうかを選択します。バッファ作成時刻、ジャーナルに記録された処理を実行したプロセス、処理のタイプ、処理がトランザクションの一部かどうか、処理に関与したグローバル、処理に関与したデータベース。
- c. [一致] ボックスおよび [検索] ボタンを使用して、レコードの特定のレコード・セットを検索します。
  1. 手動検索の場合、最初のドロップダウンを検索基準にする列に設定し、“等しい” や “等しくない” などの演算子を選択し、比較する値を最も右のボックスに入力し、[検索] をクリックします。
  2. 列の 1 つの特定のセルを比較するには、単にそのセルをダブルクリックします。例えば、KILL 処理を含むすべてのジャーナル・レコードを検索するには、[KILL] を含む [タイプ] 列の任意のセルをダブルクリックします。演算子ドロップダウンは自動的に “等しい” に設定されますが、[検索] を押す前に変更できます。

^JRNDUMP ユーティリティを使用してジャーナル全体を表示したり、SELECT^JRNDUMP のエントリ・ポイントを使用して、選択したエントリを表示したりすることもできます。詳細は、“[^JRNDUMP を使用したジャーナル・レコードの表示](#)”を参照してください。

## 7.4 ジャーナル・ファイルの切り替え

システムは、以下の状況でジャーナル・ファイルを自動的に切り替えます。

- ・ InterSystems IRIS データベースのバックアップ実行後。
- ・ 現在のジャーナル・ファイルのサイズが最大許容量 ([ジャーナル設定] ページで設定可能) に達したとき。
- ・ 代替ディレクトリが指定されている場合に、ジャーナル・ディレクトリが使用不可能になったとき。
- ・ 管理ポータル の [ジャーナル設定] ページ ([システム管理]→[構成]→[システム構成]→[ジャーナル設定]) での設定の更新後。

ジャーナリングを停止して開始するより、ジャーナル・ファイルを切り替える方が望ましい選択です。これは、前者のプロセスでは、停止後、再開するまでに発生したグローバル操作はジャーナリングされないためです。

ジャーナル・ファイルを手動で切り替えるには、以下の手順に従います。

1. 管理ポータル の [ジャーナル] ページ ([システム処理]→[ジャーナル]) に移動します。
2. データベース・ジャーナル・ファイルのリストの上にある [ジャーナルの切り替え] をクリックします。
3. [OK] をクリックして、ジャーナルの切り替えを確定します。

^JRNSWITCH ルーチンを使用するか、または ^JOURNAL ルーチン・メニューからオプション 3 の [ジャーナル・ファイルの切り換え] を選択することで、ジャーナル・ファイルを切り替えることもできます。詳細は、“[^JRNSWITCH を使用したジャーナル・ファイルの切り替え](#)”を参照してください。

## 7.5 ジャーナル・ディレクトリの切り替え

“[ジャーナル設定の構成](#)”の説明のように、ジャーナリングは、プライマリ・ディレクトリが何らかの理由で書き込みできなくなった場合、自動的にセカンダリ・ジャーナル・ディレクトリに切り替わります (構成されていることが前提)。ジャーナル・ディレクトリを手動で切り替えるには、以下を実行します。

1. 管理ポータル の [ジャーナル] ページ ([システム処理]→[ジャーナル]) に移動します。
2. データベース・ジャーナル・ファイルのリストの上にある [ディレクトリの切り替え] をクリックします。
3. [OK] をクリックして、ジャーナルの切り替えを確定します。

^JOURNAL ルーチン・メニューからオプション 13 の [ジャーナルのセカンダリ/プライマリ・ディレクトリへの切り替え] を選択することで、ジャーナル・ディレクトリを切り替えることも可能です。詳細は、“[^SWDIR^JOURNAL を使用したジャーナル・ディレクトリの切り替え](#)”を参照してください。

## 7.6 ジャーナル・ファイル・プロファイルの表示

[ジャーナル] ページでは、ジャーナル・ファイルのグローバル・プロファイルを表示できます。このプロファイルには、ファイルのレコードに表示されるグローバルおよびそれぞれのレコード数が示されます。

1. [ジャーナル] ページでは、必要に応じて **[フィルタ]** ボックスを使用して、ジャーナル・ファイルのリストの件数を絞り込むことができます。
2. ジャーナル・ファイルのプロファイルを表示するには、該当のジャーナル・ファイル行の **[プロファイル]** をクリックします。[ジャーナルプロファイル] ページにプロファイルが表示されます。ジャーナル・ファイルに多数のレコードがある場合、プロファイルの構築に時間がかかる場合があります。
3. ジャーナル・プロファイルは、グローバルごと、または各グローバルが表示されるすべてのレコードの累積サイズごと（バイト単位）にソートすることができます。
4. ジャーナル・ファイルが現在使用中のものの場合、一定の時間経過後に **[再計算]** ボタンを使用してプロファイルを再構築できます。

## 7.7 ジャーナル・ファイルの整合性のチェック

[ジャーナル] ページでは、ジャーナル・ファイルの整合性をチェックできます。この処理では、ジャーナル・ファイルが想定された場所で終了していることを確認し、ファイルの終わりからレコードの欠落がないことを確認します。

1. [ジャーナル] ページでは、必要に応じて **[フィルタ]** ボックスを使用して、ジャーナル・ファイルのリストの件数を絞り込むことができます。
2. ジャーナル・ファイルの整合性チェックを実行するには、該当のジャーナル・ファイル行の **[整合性チェック]** をクリックします。[ジャーナル整合性チェック] ページが表示されます。
3. **[詳細確認]** を選択し、ジャーナル・ファイルのレコードをレコードごとに最初からスキャンし、潜在的な欠落レコードを検出します。
4. **[OK]** をクリックすると、[バックグラウンドタスク] ページ ([システム処理]→[バックグラウンドタスク]) へのリンクが表示され、整合性チェックのステータスおよび結果を表示できます。

## 7.8 ジャーナル・ファイルの要約の表示

[ジャーナル・ファイルの要約] ページでは、ジャーナル・ファイルに関する要約を表示できます。例えば、ジャーナル・ファイルが暗号化されているかどうか、ジャーナル・ファイルに記録されている操作の影響を受けるのはどのデータベースか、などを確認できます。

1. ホーム ページの **[システム処理]** メニューで **[ジャーナル]** をクリックし、インスタンスのジャーナル・ファイルを一覧表示します。必要に応じて、**[フィルタ]** ボックスを使用してリストの件数を絞り込みます。
2. ジャーナル・ファイルに関する情報を表示するには、該当のジャーナル・ファイル行の **[要約]** をクリックします。[ジャーナル・ファイルの要約] ページが表示されます。

## 7.9 ジャーナル・ファイルの削除

使用しなくなったジャーナル・ファイルを削除するタスクを定期的に行うようにスケジュールできます。新しい Inter-Systems IRIS インスタンスには、事前にスケジュールされたジャーナルの削除タスクがあります。このタスクは、毎日午前 0 時に実行するジャーナルの切り替えタスクの後に実行するようにスケジュールされています。ミラー・ジャーナル・ファイルの削除に関する詳細は、“[ミラー・ジャーナル・ファイルの削除](#)”を参照してください。

この削除プロセスでは、[ジャーナル設定] ページの [ジャーナルファイルを削除するタイミング] の設定に基づいてジャーナル・ファイルを削除します。詳細は、“[ジャーナルの設定の構成](#)”を参照してください。

**注釈** ジャーナル・ファイルは、削除の設定の条件を満たしていても保持される場合があります。この場合、イベントはメッセージ・ログに記録され、理由（例えば、ジャーナル・ファイルに開いているトランザクションが含まれているなど）が示されます。

PURGE JOURNAL ルーチンを使用するか、または JOURNAL ルーチン・メニューからオプション 6 の [ジャーナル・ファイルの削除] を選択することで、ジャーナル・ファイルを削除することもできます。詳細は、“[PURGE JOURNAL を使用したジャーナル・ファイルの削除](#)”を参照してください。

**注釈** 構成されているジャーナル削除設定は %ZJRNPURGE ルーチンでオーバーライドできます。詳細は、[インターシステムズのサポート窓口](#)までお問い合わせください。

## 7.10 ミラー・ジャーナル・ファイルの削除

ミラー・ジャーナル・ファイルは、プライマリ・フェイルオーバー・メンバによって他のミラー・メンバに正しく配布され、それぞれでデジャーナルされて、ミラーリングされるデータベースと同期する必要があるため、追加の削除条件に従います（このプロセスの詳細は、“[ミラー同期](#)”を参照してください）。ミラーが通常動作しているときには、ファイルのバックアップへの転送は同期的に実行され、常に高速ですが、非同期メンバへの転送には時間がかかり、非同期メンバがミラーから切断されているときには遅延する場合があります。バックアップ・メンバおよび DR 非同期メンバは、フェイルオーバーまたは災害復旧の状況ではプライマリになる資格があるため、プライマリと同じポリシーにも従う必要があります。したがって、ミラー・ジャーナル・ファイルは以下のように削除されます。

- ・ プライマリ・フェイルオーバー・メンバでは、ローカル・ジャーナル・ファイルの削除条件が満たされたとき（“[ジャーナル設定の構成](#)”を参照）およびバックアップ・メンバ（存在する場合）とすべての非同期メンバに受信されたとき、ファイルは削除されます（いずれか長時間要する方）。ただし、非同期メンバがミラーから 14 日間以上切断されたままの場合、その非同期メンバがファイルを受信していなくてもそのファイルは削除されます。
- ・ バックアップ・フェイルオーバー・メンバ（存在する場合）およびすべての災害復旧 (DR) 非同期メンバでは、そのメンバで完全にデジャーナルされたとき、ローカル・ジャーナル・ファイルの削除条件が満たされたとき、およびすべての非同期メンバに受信されたとき、ファイルは削除されます。ただし、非同期メンバが 14 日間よりも長く切断されたままの場合には同じ例外が適用されます。
- ・ レポート非同期メンバでは、ミラー・ジャーナル・ファイルは、既定ではデジャーナルされた直後に削除され、非同期ミラー・メンバが領域を使い果たさないようにします（特に、非同期メンバが複数のミラーからジャーナル・ファイルを受け取っている場合）。必要に応じて、レポート非同期メンバを構成して、代わりにファイルを保持し、ローカル・ジャーナル・ファイルの削除条件に従って削除するようにできます。“[非同期メンバの編集または削除](#)”を参照してください。

現在開いているトランザクションを含むミラー・ジャーナル・ファイルはどのミラー・メンバでも削除されません。

注釈 ローカル・ジャーナル・ファイルの削除条件の規定時間よりも長くミラー・ジャーナル・ファイルが保持されると、メンバのメッセージ・ログにこれが記録され、その理由が示されます。

SYS.Mirror.JrnPurgeDefaultWait() メソッドを使用すると、ミラー・ジャーナル・ファイルを削除するための既定の設定を変更できます。

## 7.11 ジャーナル・ファイルのリストア

システム・クラッシュあるいはディスク・ハードウェア障害の後、バックアップ・コピーをリストアして、データベースを再生成します。ジャーナリングが有効になっていてジャーナル・ファイルにアクセスできる場合は、最後のバックアップ以降にジャーナルに記録された変更をデータベースに適用することで、さらにデータベースをリストアできます。

ジャーナル・ファイルをリストアする手順は以下のとおりです。

1. まず、すべてのユーザが InterSystems IRIS を終了していることを確認します。
2. ジャーナリングがインスタンスについて有効になっている場合、`JRNSTOP` を使用してジャーナリングを停止します (“[JRNSTOP を使用したジャーナリングの停止](#)” を参照)。
3. データベースの最新バックアップをリストアします。
4. ジャーナル・リストア・ユーティリティを実行します。詳細は、“[JRNRESTO を使用したジャーナル・ファイルからのグローバルのリストア](#)” を参照してください。
5. ジャーナリングが無効の場合、再開します。

注釈 管理ポータルからジャーナル・リストア・プロセスを実行することはできません。

## 7.12 関連項目

- ・ [ジャーナリングの概要](#)
- ・ [ジャーナリングの構成](#)
- ・ [ジャーナリング・ユーティリティ](#)
- ・ [ジャーナル入出力エラー](#)
- ・ [ジャーナリングに関する特別な考慮事項](#)



# 8

## ジャーナリング・ユーティリティ

InterSystems IRIS® データ・プラットフォームには、ジャーナリング・タスクを実行するためのユーティリティが用意されています。^JOURNAL ユーティリティには、いくつかの一般的なジャーナリング・ユーティリティを実行するためのメニュー項目があります。これらのユーティリティは、単独で実行することも可能です。また、%SYS ネームスペースから実行するジャーナリング・ユーティリティもいくつかあります。

以下のセクションでは、サンプル・プロシージャは、**C:\MyIRIS** を InterSystems IRIS インストール・ディレクトリとして示します。

### 8.1 ^JOURNAL を使用したジャーナリング・タスクの実行

以下の例は、^JOURNAL ルーチン呼び出すことによって使用できるメニューを示しています。以下の例ではフル・メニューは繰り返し説明しません。

```
%SYS>Do ^JOURNAL

1) Begin Journaling (^JRNSTART)
2) Stop Journaling (^JRNSTOP)
3) Switch Journal File (^JRNSWTCH)
4) Restore Globals From Journal (^JRNRESTO)
5) Display Journal File (^JRNDUMP)
6) Purge Journal Files (PURGE^JOURNAL)
7) Edit Journal Properties (^JRNOPTS)
8) Activate or Deactivate Journal Encryption (ENCRYPT^JOURNAL())
9) Display Journal status (Status^JOURNAL)
10) -not available-
11) -not available-
12) Journal catch-up for mirrored databases (MirrorCatchup^JRNRESTO)
13) Switch Journaling to Secondary Directory (SWDIR^JOURNAL)

Option?
```

注釈 オプション 11) (Manage^JRNROLL) は、^STURECOV ルーチン (システムの起動時) または ^MIRROR ルーチン (プライマリ・ミラー・メンバの起動時) を実行したときに、保留中または処理中のトランザクション・ロールバックが検出された場合に表示されます。詳細は、["Manage^JRNROLL を使用したトランザクション・ロールバックの管理"](#) を参照してください。

特定のルーチンを開始するには、該当するメニューのオプション番号を入力します。ユーティリティを終了するには、オプションの番号を入力せずに **Enter** キーを押します。以下のサブセクションでは、^JOURNAL ユーティリティで使用可能なオプションについて説明します。

- ・ [^JRNSTART を使用したジャーナリングの開始](#)
- ・ [^JRNSTOP を使用したジャーナリングの停止](#)



- ・ ^JNSWITCH を使用したジャーナル・ファイルの切り替え
- ・ SWDIR^JOURNAL を使用したジャーナル・ディレクトリの切り替え
- ・ ^JNRESTO を使用したジャーナル・ファイルからのグローバルのリストア
- ・ ^JRDUMP を使用したジャーナル・レコードの表示
- ・ PURGE^JOURNAL を使用したジャーナル・ファイルの削除
- ・ ^JRNOPTS を使用したジャーナル設定の更新
- ・ ENCRYPT^JOURNAL を使用したジャーナルの暗号化
- ・ Status^JOURNAL によるジャーナルの状態の表示
- ・ Manage^JRNROLL を使用したトランザクション・ロールバックの管理
- ・ MirrorCatchup^JNRESTO を使用したミラーリングされるデータベースへのジャーナルのリストア

### 8.1.1 ^JRNSTART を使用したジャーナリングの開始

ジャーナリングを開始するには、以下の例に示すように、^JRNSTART を実行するか、または ^JOURNAL メニューの  
・プロンプトで「1」を入力します。

^JRNSTART を直接実行する例

```
%SYS>Do ^JRNSTART
```

^JOURNAL メニューからジャーナリングを開始する例

```
%SYS>Do ^JOURNAL
```

```
1) Begin Journaling (^JRNSTART)
...
Option? 1
```

このオプションを選択したときにジャーナリングを実行している場合、以下のようなメッセージが表示されます。

```
Already journaling to C:\MyIRIS\mgr\journal\20231113.001
```

### 8.1.2 ^JRNSTOP を使用したジャーナリングの停止

ジャーナリングを停止するには、以下の例に示すように、^JRNSTOP を実行するか、または ^JOURNAL メニューの  
・プロンプトで「2」を入力します。

注釈 **【エラー時に凍結】** フラグ (“[ジャーナル設定の構成](#)” を参照) が [はい] に設定されている場合、ジャーナリングの停止が許可されます (ただし、データ損失のリスクがあります)。インスタンスは凍結されません。

^JRNSTOP を直接実行する例

```
%SYS>Do ^JRNSTOP
```

```
Stop journaling now? No => Yes
```

^JOURNAL メニューからジャーナリングを停止する例：

```
%SYS>Do ^JOURNAL

...
2) Stop Journaling (^JRNSTOP)
...
Option? 2
Stop journaling now? No => Yes
```

このオプションを選択したときにジャーナリングを実行していない場合、以下のようなメッセージが表示されます。

```
Not journaling now.
```

### 8.1.3 ^JRNSWTC を使用したジャーナル・ファイルの切り替え

ジャーナル・ファイルを切り替えるには、^JRNSWTC を実行するか、または以下の例に示すように、^JOURNAL メニューの [ ] プロンプトで「3」を入力します。

```
%SYS>Do ^JOURNAL

...
3) Switch Journal File (^JRNSWTC)
...
Option? 3
Switching from: C:\MyIRIS\mgr\journal\20231113.002
To: C:\MyIRIS\mgr\journal\20231113.003
```

このユーティリティでは、以前のジャーナル・ファイル名および現在のジャーナル・ファイル名が表示されます。

### 8.1.4 SWDIR^JOURNAL を使用したジャーナル・ディレクトリの切り替え

“[ジャーナル設定の構成](#)”の説明に従ってセカンダリ・ディレクトリが構成されているものとして、ジャーナル・ディレクトリを切り替えるには、SWDIR^JOURNAL を実行するか、または以下の例に示すように ^JOURNAL メニューの [ ] プロンプトで 13 と入力します。

```
%SYS>Do ^JOURNAL

...
13) Switch Journaling to Secondary Directory (SWDIR^JOURNAL)
...
Option? 3
Option? 13
Journaling to \\remote\MyIRIS\journal_secondary\MIRROR-MIRRORONE-20230720.007
```

切り替えの後、現在のジャーナル・ディレクトリとジャーナル・ファイルの名前が表示されます。

### 8.1.5 ^JRNRESTO を使用したジャーナル・ファイルからのグローバルのリストア

InterSystems IRIS ^JRNRESTO ルーチンは、バックアップからデータベースをリストアした後で使用して、ジャーナル・ファイルのデータベース更新内容を適用することで、データベースを障害発生直前の状態に戻します。これは、ジャーナル・リストアと呼ばれています。また、更新の適用プロセスは、デジャーナリングと呼ばれています。ジャーナル・リストアでは、バックアップの作成と障害発生間に作成されたすべてのジャーナル・レコードをデジャーナルします。例えば、データベースのバックアップを火曜日の早朝に作成し、データベースが水曜日の午後にクラッシュした場合は、火曜日のバックアップをリストアした後で、火曜日と水曜日に作成されたジャーナル・ファイルから更新をリストアできます。

使用可能なシステム・リソースが十分にある場合は、1 つのジャーナル・リストア操作の中で最大 16 のデジャーナリング・ジョブで並列して更新を実行できます (“[並列デジャーナリングのシステム要件](#)”を参照)。これにより、操作のパフォーマンスが向上します。これは、並列デジャーナリングと呼ばれます。複数のアップデータで同時に同じデータベースを更新することはできませんが、同じグローバルを更新することはできません。これにより、少数のデータベースに更新を適用する場合は、各グローバル内のデータの論理的な整合性を確保しつつ、スループットを向上させられます。

並列デジャーナリングは、ホスト・システムに並列デジャーナリングをサポートするために十分な CPU が搭載されていて、この目的に割り当て可能な共有メモリ・ヒープ (gmheap) を InterSystems IRIS インスタンスが十分に確保している場合にのみ有効になります。実際には、gmheap が増加されている場合を除き、ほとんどの InterSystems IRIS インスタンスでジャーナル・リストアに並列デジャーナリングが使用されることはありません。並列デジャーナリングのジョブ数は、gmheap のサイズを 200 MB で除算した数を超過することはできません。例えば、同時に 4 つのデジャーナリング・ジョブが実行されているとすると、800 MB 以上の gmheap が必要になります(並列デジャーナリングをサポートするにはメモリが不足している場合でも、共有メモリ・ヒープのサイズを既定値よりも増やすと、デジャーナリングのスループットが向上することがあります)。

**注釈** 共有メモリ・ヒープまたは gmheap (共有メモリヒープや SMH と呼びます) のサイズを変更するには、[\[メモリ詳細設定\]](#) ページ ([\[システム管理\]](#)→[\[構成\]](#)→[\[追加設定\]](#)→[\[メモリ詳細\]](#)) に移動します。詳細は、“システム管理ガイド”の“InterSystems IRIS の構成”の章にある“[メモリと開始設定](#)”を参照してください。

並列デジャーナリングは、InterSystems IRIS のミラーリングでも使用されることがあります。“[並列デジャーナリングの構成](#)”を参照してください。

^JRNRESTO でリストアされるデータベースは、ジャーナルをリストアする時点でジャーナル状態が Yes になっているもののみです。各データベースが初めて検出されたときに、このルーチンによってデータベースのジャーナル状態の検査が行われ、記録されます。リストアのプロセスでは、ジャーナル状態が No であるデータベースのジャーナル・レコードは無視されます。ジャーナル記録対象として指定されているデータベースがない場合、リストアを終了するかどうかを尋ねるメッセージが表示されます。ここで、特定のデータベースについてデータベースのジャーナル状態を Yes に変更し、^JRNRESTO を再起動できます。

**注釈** リストアの時点でのデータベースのジャーナル状態によって、実行するアクションが決まります。InterSystems IRIS では、指定されたジャーナル・レコードを書き込むときに、データベースの現在のジャーナル状態に関しては何もジャーナルに保存しません。つまり、ジャーナル状態が Yes のデータベースに対する変更は永続的ですが、それ以外のデータベースの場合は変更は永続的ではありません。InterSystems IRIS では、物理的な整合性は保証されますが、トランザクションに関与するデータベースのジャーナル状態が No の場合は、アプリケーションの整合性が必ずしも保証されるとは限りません。

^JRNRESTO では、以下のように、ジャーナルのリストアに関するいくつかの決定を行うことができます。

- ・ ルーチンを実行している InterSystems IRIS インスタンスのデータベース、または別の InterSystems IRIS インスタンスのデータベースへのグローバル更新のリストア。現在のインスタンスですべてのデータベースにすべてのグローバルの更新をリストアするように選択したり、現在のインスタンスまたは別のインスタンスで個別データベースを選択できます。また、必要に応じて、それぞれにリストアするグローバルを指定できます。
- ・ ミラーリングされるデータベースへのミラー・ジャーナル・ファイルのリストア (ミラーリングされるデータベースのキャッチアップ) またはミラーリングされないデータベースへのミラー・ジャーナル・ファイルのリストア。以下の手順に示すように、ミラー・メンバでは、ミラーリングされるデータベースをキャッチアップするかどうかを示すように求めるプロンプトが表示されます。これを行う場合、手順は ^JRNRESTO への MirrorCatchup エントリ・ポイントにリダイレクトされます (“[MirrorCatchup^JRNRESTO を使用したミラーリングされるデータベースへのジャーナルのリストア](#)”を参照)。
- ・ 既存のジャーナル・フィルタ (“[ZJRNFLT を使用したジャーナル・レコードのフィルタ処理](#)”を参照) をリストアに適用します。
- ・ リストア元のジャーナル・ファイルの範囲を選択します。
- ・ リストア中に更新のジャーナリングを無効にし、処理を高速化します。
- ・ 以下に示すエラー発生時の処理を選択します。
  - － データベース関連の問題があってもジャーナル・リストアを続行する。例えば、ターゲット・データベースをマウントできない場合や、更新の適用時にエラーが発生した場合です。この設定では、影響を受けたデータベースの更新はスキップされます。これらのデータベースは、操作後に他のデータベースとの整合性がない状態になっていたり、整合性のないグローバルが含まれていたりする場合があります (ただし、各グローバル内のデータの論理的な整合性は保証されています)、個別にリカバリする必要があります。

- データベース関連の問題のために更新をスキップしなければならない場合は、ジャーナルを中止する。この設定では、データベースと、データベースに含まれるグローバルは、リストアが中止される原因となったレコードまでは、お互いに整合性があります。この設定では、並列デジャーナリングは無効になります。

**注意**            プロンプトに応じてジャーナル・リストア・スクリプトを使用する場合には、最新のリリース以降に変更されているプロンプトが存在する可能性があるため、スクリプトを更新する必要があります。

ジャーナル・ファイルからグローバル更新をリストアするには、以下の手順を実行します。

1. %SYS ネームスペースで ^JRNRESTO ルーチンを実行し、Restore the Journal? プロンプトで <Enter> キーを押して続行します。
2. ミラー・メンバでこのルーチンを実行している場合は、以下のプロンプトが表示されます。

Catch-up mirrored databases? No =>

- ・ ミラー・ジャーナル・ファイルが作成されたミラー内のミラーリングされるデータベースにこのミラー・ジャーナル・ファイルをリストアする場合、yes と入力します。このプロシージャは ^JRNRESTO への MirrorCatchup^ エントリ・ポイントにリダイレクトされます ("[MirrorCatchup^JRNRESTO を使用したミラーリングされるデータベースへのジャーナルのリストア](#)" を参照)。
  - ・ ミラー・ジャーナル・ファイルをミラーリングされないデータベースにリストアするか、またはミラー・ジャーナル・ファイルをリストアしない場合は、no と入力するかまたは <Enter> キーを押して、ここで説明する手順を引き続き使用します。
3. 既にジャーナル・フィルタが存在する場合 ("[ZJRNFLT を使用したジャーナル・レコードのフィルタ処理](#)" を参照)、これらのフィルタを使用するかどうかを指定します。

Use current journal filter (ZJRNFLT)?  
Use journal marker filter (MARKER^ZJRNFLT)?

4. ジャーナリングされているグローバルのすべてを現在の InterSystems IRIS インスタンスのすべてのデータベースにリストアするか、または1つ以上のデータベースを指定し、必要に応じて、それぞれのデータベースにリストアするグローバルを指定するかを選択します。

Process all journaled globals in all directories?

- ・ 現在のインスタンスのすべてのデータベースにすべてのグローバルをリストアする場合、「Yes」と入力します。
- ・ 現在のインスタンスまたは別のインスタンスの選択したデータベースのみにリストアする場合、「No」と入力するか、<Enter> キーを押します。以下の手順を実行します。
- 現在のシステムのオペレーティング・システムと異なるオペレーティング・システムでジャーナル・ファイルを作成したかどうかを示します。リストア先のデータベースに指定するディレクトリ・パスはジャーナル・ファイルのパス ([キャノニック形式](#)) と正確に一致している必要があるため、これは重要なことです。「No」と応答した場合、^JRNRESTO は、入力したディレクトリ・パスがジャーナル・ファイルのパスと一致するように、ディレクトリ・パスを現在のオペレーティング・システムのキャノニック形式にします。「Yes」と応答した場合、^JRNRESTO は、ジャーナル・ファイルのキャノニック形式と現在のシステムのキャノニック形式が異なるため、入力したパスを正規化しません。後者の場合、注意深くジャーナル・ファイルのオペレーティング・システムに対して適切なキャノニック形式でディレクトリ・パスを入力し、両方が一致するようにする必要があります。

以下に例を示します。

- ・ Windows システムで作業をしていて、このプロンプトで「No」を入力し、パス c:\intersystems\iris\mgr\user を入力すると、^JRNRESTO は、これを自動的に c:\intersystems\iris\user に正規化し、Windows システムで作成されたジャーナル・ファイルと一致させます。

- ・ Unix® システムで作業していて、ジャーナル・ファイルを Windows システムで作成したためにこのプロンプトで「Yes」と入力した場合、キャノニック形式のパス `c:\intersystems\iris\mgr\user\` を入力し、ジャーナル・ファイルと一致させる必要があります。これは、`^JRNRESTO` がパスを正規化できないためです。
- ディレクトリ・パスを入力してリストアする各データベースを指定します。これは、ジャーナル・レコードの取得元のソース・データベースを示します。Redirect to directory プロンプトで `<Enter>` キーを押し、ソースとターゲットが同じであることを示し、グローバル更新をソース・データベースにリストアします。別のデータベースにリストアしている場合（例えば、ソース・データベースをバックアップから別のシステムにリストアしたため）、ターゲット・データベースのディレクトリ・パスを入力します。

ミラー・ジャーナル・ファイルをミラーリングされないデータベースにリストアする場合は、Directory to restore プロンプトで、以下のいずれかを実行します。

- ・ ソース・データベースのディレクトリ・パスを入力してから、`<Enter>` キーを押すかまたはターゲットのミラーリングされないデータベースのディレクトリ・パスを入力します（前述の説明を参照）。
- ・ ミラーリングされるソース・データベースの大文字と小文字を区別した完全なミラー・データベース名を入力します（例えば、`:mirror:JLAP:MIRRORDB`）。これを見つけるには、`^MIRROR` ユーティリティの **[ミラー・ステータス]** メニューにある **[ミラーリングされるデータベースを一覧表示]** オプションを使用します。その後、ターゲットのミラーリングされないデータベースのディレクトリ・パスを指定します。

注釈 ミラー・ジャーナル・ファイルをミラーリングされるデータベースにリストアする場合、この手順ではここまでは説明しません。[“MirrorCatchup^JRNRESTO を使用したミラーリングされるデータベースへのジャーナルのリストア”](#) を参照してください。

- 指定した各データベースについて、すべてのグローバルの更新をリストアするか、またはリストアする 1 つ以上のグローバルを入力するかを確認します。
- すべてのデータベースを入力した場合、Directory to restore プロンプトで `<Enter>` キーを押し、指定したデータベースおよびグローバルのリストを確認します。

以下に例を示します。

```
Process all journaled globals in all directories? no
Are journal files imported from a different operating system? No => No

Directory to restore [? for help]: c:\intersystems\test23\mgr\user\
Redirect to Directory: c:\intersystems\test23\mgr\user\
=> --> c:\intersystems\test23\mgr\user\
Process all globals in c:\intersystems\test23\mgr\user\? No => yes

Directory to restore [? for help]: c:\intersystems\test23\mgr\data\
Redirect to Directory: c:\intersystems\test23\mgr\data\
=> --> c:\intersystems\test23\mgr\data\
Process all globals in c:\intersystems\test23\mgr\data\? No => no

Global ^Survey.LastName
Global ^Survey.ID
Global ^

Directory to restore [? for help]:

Processing globals from the following datasets:
  1. c:\intersystems\test23\mgr\user\    All Globals
  2. c:\intersystems\test23\mgr\data\    Selected Globals:
      ^Survey.LastName
      ^Survey.ID

Specifications correct? Yes => Yes
```



注釈 複数のデータベースを同じディレクトリにリダイレクトする場合は、同じグローバル選択を行う必要があります。つまり、対象のデータベースすべてについて、「yes」と入力し、すべてのグローバルを処理するか、「no」と入力し、処理するグローバルの同一リストを指定します。複数のデータベースを 1 つのディレクトリにリストアする場合にグローバル選択がすべて同じでないと、データベースのリダイレクトとグローバル選択を変更するか、処理をキャンセルするよう求められます。

5. 正しいジャーナル履歴ログ (“[ジャーナル履歴ログ](#)” を参照) を指定して、リストア元 (リストアしているソース・データベースと同じ InterSystems IRIS インスタンス) のジャーナル・ファイルを指定します。
  - ・ プロンプトで「yes」と入力するかまたは <Enter> キーを押し、現在の InterSystems IRIS インスタンスのジャーナル履歴ログを使用して、処理するジャーナル・ファイルを特定します。例えば、プロセスの最初に Process all journaled globals in all directories? プロンプトで「yes」と入力した場合、ここで「yes」と入力し、現在のインスタンスのジャーナル・ファイルから現在のインスタンスのすべてのデータベースをリストアします。
  - ・ Process all journaled globals in all directories? プロンプトで「no」と入力し、別の InterSystems IRIS インスタンスのデータベースを指定した場合、ここで「no」と入力し、そのインスタンス (またはそのインスタンスからコピーしたファイル) のジャーナル履歴ログおよびジャーナル・ファイルのディレクトリ・パスを指定し、そのインスタンスのジャーナル・ファイルからデータベースをリストアできるようにします。

重要 別の InterSystems IRIS インスタンスからジャーナル履歴ログを使用している場合、実際のログではなくファイルのコピーを使用する必要があります。

以下はその例です。

```
If you have a copy of the journal history log file from the
instance where the journal files were created, enter its full path below;
otherwise, press ENTER and continue.
Journal history log: c:\InterSystems\IRIS23_journals\journal.log

Specify the location of the journal files to be processed
Directory of the journal files: c:\InterSystems\IRIS23_journals\journal\
Directory of the journal files:
```

6. 処理対象とするジャーナル・ファイルの範囲を指定します。以下の点に注意してください。
  - ・ リストアされたバックアップ以後に InterSystems IRIS が複数のジャーナル・ファイルに切り替えた場合、最も古いジャーナル・ファイルから順番に、最新ののものまでをリストアする必要があります。例えば、リストアする必要がある 3 つのジャーナル・ファイル **20230214.001**、**20230214.002**、**20230215.001** がある場合、以下の順番でリストアする必要があります。

**20230214.001**

**20230214.002**

**20230215.001**

- ・ オンライン・バックアップでバックアップすると、リストア時のトランザクション・ロールバックに必要な最も古いジャーナル・ファイルに関する情報が 3 番目または最後のパスの開始時に表示され、バックアップ・ログにも格納されます。詳細は、“[バックアップとリストア](#)” を参照してください。

以下のようにプロンプトに応答してください。

- ・ このインスタンスがジャーナル・ファイルを作成するかどうかに関する前述のプロンプトで「yes」と入力した場合、または「no」と応答してから別のインスタンスのジャーナル履歴ログおよびジャーナル・ファイルの場所を指定した場合、処理する最初のジャーナル・ファイルおよび最後のジャーナル・ファイルのパス名を入力できます。い

ずれのプロンプトにも「?」を入力して、指定した場所にあるファイルの番号付けされたリストを確認し、ファイルの番号を入力できます。以下はその例です。

```
Specify range of files to process
Enter ? for a list of journal files to select the first and last files from
First file to process: ?

1) c:\intersystems\iris2\mgr\journal\20230212.001
2) c:\intersystems\iris2\mgr\journal\20230213.001
3) c:\intersystems\iris2\mgr\journal\20230214.001
4) c:\intersystems\iris2\mgr\journal\20230214.002
5) c:\intersystems\iris2\mgr\journal\20230215.001
6) c:\intersystems\iris2\mgr\journal\20230216.001
7) c:\intersystems\iris2\mgr\journal\20230217.001
8) c:\intersystems\iris2\mgr\journal\20230217.002

First file to process: 5 c:\intersystems\iris2\mgr\journal\20230215.001
Final file to process:
  c:\intersystems\20231316mar14\mgr\journal\20230217.002 =>

Prompt for name of the next file to process? No => no
```

このインスタンスがジャーナル・ファイルを作成するかどうかに関する前述のプロンプトで「no」を入力し、ジャーナル履歴ログを指定しなかった場合、処理する特定のジャーナル・ファイルを特定しようとするプロンプトで処理が続行されます。以下に例を示します。

```
Journal history log:
Specify range of files to process (names in YYYYMMDD.NNN format)

from:      <20230212.001> [?] => 20230215.001

through:    <20230217.001> [?] => 20230217.002

Prompt for name of the next file to process? No => no

Provide or confirm the following configuration settings:

Journal File Prefix: [?] =>

Files to dejournal will be looked for in:
  c:\intersystems\iris\mgr\journal\
in addition to any directories you are going to specify below, UNLESS
you enter a minus sign ('-' without quotes) at the prompt below,
in which case ONLY directories given subsequently will be searched

Directory to search: {return when done} -
  [Directory search list is emptied]
Directory to search: {return when done} c:\intersystems\iris2\mgr\journal
Directory to search: {return when done}
Here is a list of directories in the order they will be searched for files:
  c:\intersystems\iris2\mgr\journal\
```

## 7. 以下のようにジャーナル・ファイルを処理します。

```
Prompt for name of the next file to process? No => No
The following actions will be performed if you answer YES below:

* Listing journal files in the order they will be processed
* Checking for any missing journal file on the list ("a broken chain")

The basic assumption is that the files to be processed are all
currently accessible. If that is not the case, e.g., if you plan to
load journal files from tapes on demand, you should answer NO below.
Check for missing journal files? Yes => Yes
```



8. 指定した範囲内で1つまたは複数のジャーナル・ファイルが見つからない場合は、操作の中止を選択できます。操作を中止しなかった場合、またはファイルがすべて見つかった場合は、処理が続行され、リストアを開始する前にジャーナルの整合性をチェックする機会が得られます。

Journal files in the order they will be processed:

```
1. c:\intersystems\iris2\mgr\journal\20230215.001
2. c:\intersystems\iris2\mgr\journal\20230216.001
3. c:\intersystems\iris2\mgr\journal\20230217.001
4. c:\intersystems\iris2\mgr\journal\20230217.002
```

While the actual journal restore will detect a journal integrity problem when running into it, you have the option to check the integrity now before performing the journal restore. The integrity checker works by scanning journal files, which may take a while depending on file sizes. Check journal integrity? No => No

9. 現在のジャーナル・ファイルがリストアに含まれている場合は、ジャーナリングを別のファイルに切り替える必要があり、そのためのプロンプトが表示されます。

The journal restore includes the current journal file.  
You cannot do that unless you stop journaling or switch journaling to another file.  
Do you want to switch journaling? Yes => yes  
Journaling switched to c:\intersystems\iris2\mgr\journal\20230217.003

10. 次に、リストア中に更新のジャーナリングを無効にし、処理を高速化するかどうかを選択します。

You may disable journaling of updates for faster restore for all databases other than mirrored databases.  
Do you want to disable journaling the updates? Yes => yes  
Updates will NOT be journaled

**重要** リストア中の更新のジャーナリングを無効化していないと、このセクションの冒頭で説明したように、パフォーマンスを向上するための並列デジャーナリングが使用されなくなります。

ジャーナリングが無効になっている状態でデータベースの更新を継続すると、下記のいずれかを確実にしない限り、最新の良好なジャーナルを使用して手動でリストアすることはできなくなります。

- ・ 更新される内容を正確に把握し、アプリケーションの要件を満足するためにリストアされる内容を管理できます。
- ・ 関連するデータベースを最後のバックアップからリストアしたうえで、ジャーナルを適用すると、ジャーナルが無効な状態で書き込まれたデータが喪失することを容認します。

このような状況では、ジャーナルのリストア完了後に下記のコマンドを実行し、オブジェクト ID が同期していることを確認することをお勧めします。同期していないことが判明した ID のみが配列 errors にレポートされます。

```
Do CheckIDCounters^%apiOBJ(.errors)
zwrite errors
```

11. リストア・ジョブのエラー発生時の既定の動作を確認または変更した後、リストアを開始することを確認します。

Before we job off restore daemons, you may tailor the behavior of a restore daemon in certain events by choosing from the options below:

DEFAULT: Continue despite database-related problems (e.g., a target database cannot be mounted, error applying an update, etc.), skipping updates to that database. Affected database(s) may not be self-consistent and will need to be recovered separately

ALTERNATE: Abort if an update would have to be skipped due to a database-related problem (e.g., a target database cannot be mounted, error applying an update, etc.). Databases will be left in a self-consistent state as of the record that caused the restore to be aborted. Parallel dejournaling will be disabled with this setting

DEFAULT: Abort if an update would have to be skipped due to a journal-related problem (e.g., journal corruption, some cases of missing

```
journal files, etc.)
```

```
ALTERNATE: Continue despite journal-related problems (e.g., journal
corruption, some missing journal files, etc.), skipping affected updates
```

```
Would you like to change the default actions? No => No
```

```
Start the restore? Yes =>
```

重要 データベース関連の問題が原因で中断しないようにする場合は、このセクションの冒頭で説明したように、パフォーマンスを向上させるための並列デジャーナリングが使用されなくなります。

12. ジャーナル・リストアの進行状況は間隔をおいて表示され、ジョブが完了すると、このリストアによって更新されたデータベースのリストが表示されます。

```
c:\MyIRIS1\mgr\journal\20230406.001
35.73% 70.61% 100.00%
c:\MyIRIS1\mgr\journal\20230406.002
35.73% 70.61% 100.00%
c:\MyIRIS1\mgr\journal\20230406.003
100.00%
[Journal restore completed at 20230407 02:25:31]
```

```
The following databases have been updated:
```

```
1. c:\MyIRIS1\mgr\source22\
2. c:\MyIRIS1\mgr\source23\
3. c:\MyIRIS1\mgr\irislocaldata\
4. c:\MyIRIS1\mgr\irislib\
5. c:\MyIRIS1\mgr\iristemp\
```

```
The following databases have been skipped:
```

```
1. /bench/user/InterSystems/IRIS/162/
2. /scratch1/user/InterSystems/IRIS/p750.162/mgr/
3. /scratch1/user/InterSystems/IRIS/p750.162/mgr/irislocaldata/
4. /scratch1/user/InterSystems/IRIS/p750.162/mgr/irislib/
5. /scratch1/user/InterSystems/IRIS/p750.162/mgr/iristemp/
6. /scratch1/user/InterSystems/IRIS/p750.162/mgr/user/
```

13. オープン・トランザクションがある場合、リストアの最後に、不完全トランザクションのロールバックを求める以下のようなプロンプトが表示されます。

```
Rollback incomplete transactions? No =>
```

このプロンプトは、リストアされたデータベースに不完全トランザクションがある場合にのみ表示されます。詳細は、“[不完全トランザクションのロールバック](#)”を参照してください。

### 8.1.5.1 不完全トランザクションのロールバック

ジャーナルのリストアでも、不完全トランザクションがある場合はそのロールバックを求めるプロンプトが表示されます。ユーザがすべてのトランザクションを完了したことを確認し、リストアにより、実行中の処理がロールバックされないようにする必要があります。

バックアップをリストアし、ジャーナル・ファイルを消去する前に、すべてのトランザクションが完了しているようにする方法として、インターシステムズは以下の方法を強く推奨します。

- ・ 自分自身が使用しているプロセスのトランザクションをロールバックする必要がある場合は、そのプロセスを中断するか、TROLLBACK コマンドを使用します。
- ・ システム全体でトランザクションをロールバックする必要がある場合、InterSystems IRIS をシャットダウンした後に再起動して、システムに接続しているユーザが存在しないようにします。

### 8.1.5.2 ミラー・ジャーナル・ファイルのリストア

ミラーリングされるデータベースまたはミラーリングされないデータベースのどちらにも、ミラー・ジャーナル・ファイルをリストアできます。ミラーリングされるデータベースにリストアする場合は、“[JRNRESTO を使用したジャーナル・ファイルから](#)

のグローバルのリストア”の手順 2、および“MirrorCatchup^JRNRESTO を使用したミラーリングされるデータベースへのジャーナルのリストア”を参照してください。ミラーリングされないデータベースにリストアする場合は、“^JRNRESTO を使用したジャーナル・ファイルからのグローバルのリストア”の手順 4 を参照してください。

### 8.1.5.3 ^ZJRNFLT を使用したジャーナル・レコードのフィルタ処理

ジャーナル・ファイルの操作のためにジャーナル・フィルタ・メカニズムが用意されています。ジャーナル・フィルタ・プログラムは、^ZJRNFLT と呼ばれ、ユーザが記述する下記の形式のルーチンです。これは、InterSystems IRIS のジャーナル・リストア・プログラム ^JRNRESTO から呼び出され、選択したレコードのみをリストアします。^ZJRNFLT ルーチンは、次の形式を使用して生成します。

```
ZJRNFLT(jidsys,dir,glo,type,restmode,addr,time)
```

引数	タイプ	説明
jidsys	入力	コンマで区切られた 2 つのコンポーネント：jid (ジョブ ID) と remsysid (ECP の場合のみ)。  jidsys を %SYS.Journal.Record.GetRealPIDSYSinFilter メソッドに渡して (以下の“^ZJRNFLT Examples” に示すように)、ジャーナルを生成した PID (プロセス ID) を特定します。
dir	入力	リストアする IRIS.DAT ファイルが格納されたディレクトリのフル・パス名。ジャーナル・レコードでの指定と同じにする必要があります。
glo	入力	ジャーナル・レコードのグローバル。
type	入力	ジャーナル・レコードのコマンド・タイプ (Set の場合は S、Kill の場合は K)。
addr	入力	ジャーナル・レコードのアドレス
time	入力	レコードのタイムスタンプ (\$horolog 形式)。これは、ジャーナル・バッファの作成時刻であり、Set 処理や Kill 処理が実行された時刻ではありません。つまり、この特定の処理が最も早く実行された時刻を示しています。
restmode	出力	0 - レコードをリストアしません。  1 - レコードをリストアします。

#### ^ZJRNFLT の考慮事項

^ZJRNFLT を使用するときは、以下の点を考慮してください。

- ・ スタートアップ・ルーチン (^STU) で ^JRNRESTO を呼び出す場合は、どのような状況でもフィルタ・ルーチンを呼び出すことはありません。
- ・ ジャーナルのリストアでジャーナル・フィルタ (^ZJRNFLT) が呼び出されるのは、それが存在する場合のみです。ジャーナル・フィルタが存在する場合は、リストア処理でそのフィルタを使用するかどうかを指定するようにリストア手順で求められます。
- ・ リストアするジャーナル・ファイルのすべてのレコードに対してジャーナル・フィルタを使用することを指定すると、ここに示した入力引数を使用してジャーナル・フィルタ ^ZJRNFLT が呼び出されるので、現在のレコードをリストアするかどうかを指定します。
- ・ ^ZJRNFLT ルーチンでは、あらゆるロジックを使用して、レコードをリストアするかどうかを指定できます。出力引数 restmode を使用して操作の確認が返されます。
- ・ ^ZJRNFLT ルーチンのロジックでディレクトリ名 dir を使用する場合は、完全ディレクトリ・パス名で指定します。
- ・ グローバル参照全体が、プログラムのロジックで使用するために ^ZJRNFLT に渡されます。

- ・ ジャーナルのリストア処理が完了すると、`ZJRNFLT ルーチンの名前を変更するか、そのルーチンを削除するかの確認を求められます。フィルタの名前変更を選択すると、フィルタの名前は `XJRNFLT に変更され、元の `ZJRNFLT は削除されます。
- ・ `ZJRNFLT ルーチンにエラーが発生すると、リストア処理が中止され、対応するメッセージが表示されます。

### ^ZJRNFLT の例

2 つのグローバル、`ABC および `XYZ がジャーナリングの対象となっています。ジャーナリングが有効になっている間、以下のコードが実行され、ジャーナル・ファイルには、この 2 つのグローバルに対する Set 処理および Kill 処理が記録されます。

#### ObjectScript

```
For I=1:1:500 Set ^ABC(I)=" "
For I=1:1:500 Set ^XYZ(I)=" "
For I=1:1:100 Kill ^ABC(I)
```

1. `ABC のみのレコードをすべてリストアする場合、`ZJRNFLT ルーチンは以下のようになります。

#### ObjectScript

```
ZJRNFLT(jidsys,dir,glo,type,restmode,addr,time) /*Filter*/
Set restmode=1 /*Return 1 for restore*/
If glo["XYZ" Set restmode=0 /*except when it is ^XYZ*/
Quit
;
```

2. `ABC に対する kill 処理以外のレコードをすべてリストアする場合、`ZJRNFLT ルーチンは以下のようになります。

#### ObjectScript

```
ZJRNFLT(jidsys,dir,glo,type,restmode,addr,time) /*Filter*/
Set restmode=1 /*Return 1 for restore*/
If glo["^ABC",type="K" Set restmode=0 /*except if a kill on ^ABC*/
Quit
;
```

3. jid が PID やミラー・メンバに対するものである場合のように、remsysid が実際の ECP システム ID ではないこともあります。そのような場合は、%SYS.Journal.Record.GetRealPIDSYSinFilter メソッドを使用して、実際の PID と同時に実際の ECP システム ID も返されるようにします。

実際の PID (および ECP システム PID) をフィルタで得るには、次のような `ZJRNFLT ルーチンを使用します。

#### ObjectScript

```
ZJRNFLT(jidsys,dir,glo,type,restmode,addr,time) ;
SET restmode=0 ;test only
SET pid=##class(%SYS.Journal.Record).GetRealPIDSYSinFilter(jidsys,.ecpsysid)
DO
##class(%SYS.System).WriteToConsoleLog($SELECT(pid="": "jid=" + jidsys, 1: "pid=" _ pid) _ ", ecpsysid=" _ ecpsysid)
QUIT
```

4. 特定の時刻以降のレコードをすべてリストアする場合、`ZJRNFLT ルーチンは以下のようになります。

## ObjectScript

```

ZJRNFLT(jidsys,dir,glo,type,restmode,addr,time) ;
New zdh
Set zdh=$zdate("08/14/2015 14:18:31") ;in $H format as variable 'time'
If time>zdh Set restmode=1 Quit
If time<zdh Set restmode=0 Quit
If $p(time,"",2)<$p(zdh,"",2) {
    Set restmode=0
} Else {
    Set restmode=1
}
Quit

```

## 8.1.6 ^JRNDUMP を使用したジャーナル・レコードの表示

ジャーナル・ファイルのレコードを表示するには、^JOURNAL メニューの [ジャーナル・ファイルのリスト](#) のプロンプトで「5」を入力するか、または以下の例に示すように、^JRNDUMP を実行します。

1. システム管理者のネームスペースから ^JRNDUMP ユーティリティを実行するには、次のように入力します。

```
%SYS>DO ^JRNDUMP
```

Journal	Directory & prefix
20231113.001	C:\MyIRIS\Mgr\Journal\
20231113.002 [JRNSTART]	C:\MyIRIS\mgr\journal\
20231113.003	C:\MyIRIS\mgr\journal\
20231113.004	C:\MyIRIS\mgr\journal\
20231114.001	C:\MyIRIS\mgr\journal\
20231115.001	C:\MyIRIS\mgr\journal\
20231115.002	C:\MyIRIS\mgr\journal\
20231115.003	C:\MyIRIS\mgr\journal\
> 20231115.004	C:\MyIRIS\mgr\journal\

2. このルーチンは、ジャーナル・ファイルのリストを表示します。現在選択しているファイルの左側に大なり記号 (>) が表示され、その後には以下のプロンプトが表示されます。

```
Pg(D)n,Pg(U)p,(N)ext,(P)rev,(G)oto,(E)xamine,(Q)uit =>
```

これらのオプションを使用して、目的のジャーナル・ファイルに移動します。

- ・ インスタンスがミラー・メンバの場合、M と入力してミラー・ジャーナル・ファイルのみにリストを制限します。(ミラー・ジャーナル・ファイルの詳細は、["ジャーナル・ファイルとジャーナル履歴ログ"](#) および ["ミラー同期"](#) を参照してください。)
- ・ ジャーナル・ファイル・リストの表示ページを変更するには、「D」または「U」を入力します。
- ・ 必要なジャーナル・ファイルに > を移動するには、「N」または「P」を入力します。
- ・ 表示するジャーナル・ファイルのフル・パス名を直接入力するには、「G」を入力します。
- ・ 選択したジャーナル・ファイルの内容を表示するには、「E」を入力します。
- ・ 選択されたジャーナル・ファイルに関する情報、および必要に応じてジャーナルのデータベースのリストを表示するには、「I」を入力します。
- ・ ルーチンを終了するには、「Q」を入力するか、Enter キーを押します。

3. 「I」の入力時、現在選択されているジャーナル・ファイルを受け入れた場合、または別のジャーナル・ファイルを指定した場合、以下のような情報が表示されます。

```
Journal: C:\MyIRIS\mgr\journal\20231113.003
File GUID: 97734819-CA75-4CB1-9C3E-74D294784D23
Max Size: 1073741824
Time Created: 2023-11-13 10:44:52
File Count: 22
Min Trans: 22,3497948
Prev File: C:\MyIRIS\mgr\journal\20231113.002
Prev File GUID: 8C5D3476-F12C-4258-BF6C-7423876653A4
Prev File End: 0
Next File: C:\MyIRIS\mgr\journal\20231113.004
Next File GUID: 4F4D20B1-D38C-473E-8CF0-4D04C6AF90B0
```

(D)atabases, (Q)uit =>

[最小トランザクション] は、ファイル数および最小トランザクションの位置のオフセットです (開いているトランザクションは、すべてこの時点以降から開始しています)。

選択されているファイルがミラー・ジャーナル・ファイルである場合、追加情報が表示されます。

下部のプロンプトで「Q」を入力すると、ジャーナル・ファイル・リストが返されます。「D」を入力すると、以下のようなデータベース情報が表示されます。

```
Journal: C:\MyIRIS\mgr\journal\20231113.003
sfn Directory or Mirror DB Name
=====
0 C:\MyIRIS\mgr\
1 C:\MyIRIS\mgr\irislib\
2 C:\MyIRIS\mgr\iristemp\
3 :mirror:MIR:MIRTEST
5 C:\MyIRIS\mgr\irislocaldata\
6 C:\MyIRIS\mgr\user\
```

(P)rev, (N)ext, (Q)uit =>

「Q」を入力すると、ジャーナル・ファイル情報の画面に戻ります。

4. 「G」または「E」を入力すると、ユーティリティにはそのジャーナル・ファイル名が表示され、オフセット・アドレス別にファイルの内容のリスト生成が開始されます。以下に例を示します。

```
Journal: C:\MyIRIS\mgr\journal\20230330.002
Address Proc ID Op Directory Global & Value
=====
131088 2980 S C:\MyIRIS\mgr\ SYS("shdwcli","doctest","remend") = 1+
131156 2980 S C:\MyIRIS\mgr\ SYS("shdwcli","doctest","end") = 1013+
131220 2980 S C:\MyIRIS\mgr\ SYS("shdwcli","doctest","jrnend") = 1+
...
```

5. 現在のリスト表示ページの下部には、そのジャーナル・ファイルに関する情報と以下のプロンプトが表示されます。

```
Last record: 573004; Max size: 1073741824
(N)ext, (P)rev, (G)oto, (F)ind, (E)xamine, (Q)uit =>
```

これらのオプションを使用して、表示するジャーナル・レコードに移動します。

- ・ アドレスの次のページまたは前のページを表示するには、「N」または「P」を入力します。
- ・ 特定のアドレスに表示を移動するには、「G」を入力します。
- ・ ジャーナル・ファイル内の特定の文字列を検索するには、「F」を入力します。
- ・ ジャーナル・レコードのアドレスを入力してその内容を表示するには、「E」を入力します。
- ・ ジャーナル・ファイルのリストに戻るには、「Q」を入力します。



- 「E」または「G」を入力した後、プロンプトでアドレスを入力します。E オプションでは、入力したアドレスのジャーナル・レコードの内容、またはそのアドレス周辺のジャーナル・レコードの内容が表示されます。G オプションでは、そのアドレスを先頭とするジャーナル・レコードのページが表示されます。

いずれのオプションでも、指定したオフセット・アドレスに最も近いレコードが探し出されます。ジャーナル・レコードの有効なアドレスを指定する必要はありません。また、ジャーナル・ファイルの先頭に移動するには、「0」(ゼロ)を入力します。ジャーナル・ファイルの末尾に移動するには、「-1」を入力します。

- 「N」を入力すると、次のページにあるジャーナル・レコードの内容が表示され、「P」を入力すると、前のページにあるジャーナル・レコードの内容が表示されます。レコードの表示を終了するには、プロンプトで「Q」を入力し、ジャーナル・レコードのリストに戻ります。

ジャーナル・レコードには、さまざまな種類があります。

- ジャーナル・ヘッダの長さは 8192 バイトですヘッダは、各ジャーナル・ファイルの先頭に一度出現します。^JRNDUMP ユーティリティは、ジャーナル・ヘッダ・レコードを表示しません。
- ジャーナル・データ・レコード。
- ジャーナル・マーカ

以下は、^JRNDUMP で表示されるジャーナル・ファイル・データ・レコードのサンプルです。例では、Set コマンドの記録方法を示しています。Set がトランザクションの範囲外に発生したため、新しい値は記録されますが、古い値は記録されません。

Journal: C:\MyIRIS\mgr\journal\20230119.004

```
Address:      233028
Type:         Set
In transaction: No
Process ID:   4836
ECP system ID: 0
Time stamp:   60284,53240
Collation sequence: 5
Prev address: 232984
Next address: 0
```

```
Global:      ^["^C:\MyIRIS\mgr\" ]ABC
New Value: 2
```

(N)ext,(P)rev,(Q)uit =>

トランザクションで古い値も記録し、以下の 2 番目の例に示すように、トランザクション・ロールバックを許可します。

Journal: C:\MyIRIS\mgr\journal\20231115.004

```
Address:      204292
Type:         Set
In transaction: Yes
Process ID:   458772
ECP system ID: 0
Time stamp:   60584,52579 - 11/15/2023 14:36:19
Collation sequence: 5
Prev address: 204224
Next address: 204372
```

```
Global:      ^["^C:\MyIRIS\mgr\" ]ABC
New Value: 5
Old Value: 2
```

(N)ext,(P)rev,(Q)uit =>

以下は、インクリメンタル・バックアップにより作成されたジャーナル・マーカ・レコードの例です。

Journal: C:\MyIRIS\mgr\journal\20231115.004

```
Address:          210848
Type:             JrnMark
Marker ID:        -1
Marker text:      NOV 15 2023;03:14PM;Incremental
Marker seq number: 1
Prev marker address: 0
Time stamp:       60584,52579 - 11/15/2023 14:36:19
Prev address:     210744
Next address:     210940
```

(N)ext, (P)rev, (Q)uit =>

以下のテーブルは、ジャーナル・データ・レコードのそれぞれのフィールドです。

テーブル 8-1: ^JRNDUMP で表示されるジャーナル・データ・レコード・フィールド

フィールド	説明
Address	ファイルの最初から数えたバイト数によるレコード位置。レコードを選択するために値を入力する唯一のフィールドです。
Type	ジャーナル・レコード・エントリに記録される処理のタイプ。利用可能なタイプの詳細は、“ <a href="#">ジャーナル・ファイルの処理</a> ”のテーブルを参照してください。
In transaction	更新がトランザクション中に発生するかどうかを示します。
Process ID	コマンドを発行するプロセスのプロセス ID 番号。
ECP system ID	ECP システムの ID 番号 (ローカル・プロセスの場合は 0)。
Time stamp	人間が読める \$HOROLOG 形式での、ジャーナル・バッファの作成時刻。これは、Set 処理や Kill 処理が実行された時刻ではありません。つまり、この特定の処理が最も早く実行された時刻を示しています。
Collation sequence	更新されるグローバルの照合順序。
Prev address	以前のレコードの位置 (0 は、現在のレコードが最初のレコードであることを示します)。
Next address	次のレコードの位置 (0 は、現在のレコードが最後のレコードであることを示します)。
Cluster sequence #	クラスタ・マウント・データベースのグローバルの順序。クラスタ・フェイルオーバー中、別のノードからのジャーナル・エントリは、このクラスタ順どおりに更新されます。
Mirror Database Name	ミラー・ジャーナル・ファイルの場合、操作が行われたデータベースのミラー名。
Global	更新されるグローバルの拡張参照。
New Value	Set 処理についてグローバルに割り当てられた値。
Old Value	トランザクションの Set 処理または Kill 処理について、その処理が実行される前にグローバルに存在した値。

以下のテーブルでは、^JRNDUMP ジャーナル・ファイルの表示の [Op] 列および ^JRNDUMP ジャーナル・レコード・リストの [タイプ] フィールドに表示されるジャーナル処理のリストを示し、説明します。例えば、ジャーナル・ファイルの表示の前述の例では、[Op] 列の [S] が JRNSET 処理を表すのに対し、ジャーナル・レコードの表示の例では、[Type] フィールドに表示される [Set] が JRNSET 処理を表します。管理ポータルのジャーナル・レコード表示の [タイプ] 列 (“[ジャーナル・ファイルの表示](#)”を参照) は、いくつかの処理の点で ^JRNDUMP リストの [タイプ] フィールドと異なります。例えば、JRNSET 処理は、ポータルでは RemoteSET で示され、^JRNDUMP 出力では NSet で示されます。これらの違いをテーブルに示します。

このテーブルでは、[SELECT^JRNDUMP](#) 関数の使用時に、処理に基づいてジャーナル・レコードをフィルタ処理するために指定できるコードも示します。

テーブル 8-2: ジャーナル・ファイルの処理

操作	説明	ファイル・リスト内の [Op]	^JRNDUMP レコード・リスト内の [タイプ]	管理ポータル のレコード・リスト内の [タイプ]	数値 の SELECT コード	アル ファ ベットの SELECT コード
JRNSET	ノードを設定、ローカル	S <sup>1</sup>	Set	SET	6	s
JRNNSET	ノードを設定、リモート	S <sup>1</sup>	NSet	RemoteSET	10	s
JRNMIRSET	内部ミラー処理 <sup>2</sup>	S <sup>1</sup>	Mirror Set	MirrorSET	19	s
JRNBITSET	ノード内の指定されたビット位置を設定	b <sup>1</sup>	BitSet	BitSET	14	bs
JRNKILL	ノードを削除、ローカル	K <sup>1</sup>	KillNode	KILL	7	k
JRNNKILL	ノードを削除、リモート	K <sup>1</sup>	NKill	RemoteKILL	11	k
JRNKILLDES	下位ノードを削除	k <sup>1</sup>	KillDesc	KILLdes	8	k
JRNMIRKILL	内部ミラー処理 <sup>2</sup>	k <sup>1</sup>	Mirror Kill	MirrorKILL	20	k
JRNZKILL	従属ノードを削除せずにノードを削除、ローカル	k <sup>1</sup>	ZKill	ZKILL	9	zk
JRNNZKILL	従属ノードを削除せずにノードを削除、リモート	k <sup>1</sup>	NZKill	RemoteZKILL	12	zk
JRNBEGTRANS	トランザクションを開始	BT	BeginTrans	BeginTrans	4	--
JRNTBEGINLEVEL	トランザクション・レベルを開始	BTL	BeginTrans with Level	BeginTrans with level	16	--
JRNCOMMIT	トランザクションをコミット	CT	CommitTrans	CommitTrans	5	--
JRNTCOMMITLEVEL	分離されたトランザクション・レベルをコミット	CTL	CommitTrans with Level	CommitTrans with level	18	--
JRNTCOMMIT-PENDLEVEL	保留中のトランザクション・レベルをコミット	PTL	CommitTrans Pending with Level	CommitTrans Pending with level	17	--
JRNMARK	ジャーナル・マーカ	M	JrnMark	Marker	13	--

操作	説明	ファイル・リスト内の [Op]	^JRNDUMP レコード・リスト内の [タイプ]	管理ポータルレコード・リスト内の [タイプ]	数値の SELECT コード	アルファベットの SELECT コード
JRNBIGNET	ECP ネットワーキング	NN	NetReq	netsyn	15	--
JRNTROLEVEL	トランザクションをロールバック	RB	Rollback	Rollback	21	--

<sup>1</sup> 「T」は、トランザクション内で処理が実行されるときに付加されます。例えば、「ST」はトランザクション内の Set 処理を表し、「kT」はトランザクション内の ZKill 処理を表します。

<sup>2</sup> ジャーナル・リストア中は処理が無視されます。

### 8.1.6.1 ダンプするジャーナル・レコードの選択

SELECT^JRNDUMP 関数は、ジャーナル・ファイルの任意のレコードまたはすべてのレコードを表示します。この関数に渡した引数に基づき、ジャーナル・ファイルの選択したレコードがファイルの先頭から順にダンプされます。

^JRNDUMP ユーティリティの SELECT エントリ・ポイントを使用する構文は、以下のようになります。

```
SELECT^JRNDUMP(%jfile,%pid,%dir,%glo,%gloall,%operation,%remsysid)
```

引数	説明
%jfile	ジャーナル・ファイル名。既定では現在のジャーナル・ファイルです。完全修飾パスでジャーナル・ファイルを指定する必要があります。
%pid	ジャーナル・レコードのプロセス ID。既定では任意のプロセスです。
%dir	ジャーナル・レコードのディレクトリ。既定では任意のディレクトリです。
%glo	ジャーナル・レコードのグローバル参照。既定では任意のグローバルです。
%gloall	glo で指定した名前を含むすべてのグローバル・ノードに関連するエントリを列挙するかどうかを指定するグローバル・インジケータ。0 - glo で指定した名前に完全一致するグローバル参照。1 - 部分一致。glo で指定した名前を使用しているグローバル参照を持つすべてのレコード。既定値は 0 です。
%operation	ジャーナル・レコードの処理タイプ。既定では任意の処理です。“ <a href="#">ジャーナル・ファイルの処理</a> ”のテーブルにリストされている数値コードまたはアルファベット・コードを使用します。
%remsysid	ジャーナル・レコードの ECP システム ID。既定では任意のシステムです。 <sup>1</sup>

<sup>1</sup> %pid が指定されている場合、既定では %remsysid はローカル・システム (0) に設定されます。それ以外の場合は、0 に指定されている場合と同様に、既定では任意のシステムに設定されます。つまり、ローカル・システムのみからジャーナル・エントリを選択することはできません。

引数として NULL 文字列を渡すことができます。この場合、ルーチンはそれぞれの既定値を使用します。

ターミナルのその他の関数と同様に、Device: プロンプトを使用して、SELECT^JRNDUMP の出力先をターミナル以外のデバイス、またはファイルに指定できます。(ユーザによるデバイスの選択の詳細は、“[入出力の概要](#)”を参照してください。)出力先をファイルに指定した場合、パラメータの入力を求められます。ファイルに書き込む場合は、R と W を含める必要があります。既存のファイルの場合は、A を含めることで、既存の内容に出力が上書きされるのではなく、追加さ

れます。新しいファイルの場合は、Nを指定する必要があります。Parameters? プロンプトで「?」を入力すると、すべての利用可能な選択肢を表示できます。

注釈 上書き先のファイルが現在の出力よりも長い場合、元のファイルの余分な行が更新されたファイルから削除されない場合があります。

## SELECT^JRNDUMP の例

以下の例では、特定のジャーナル・レコードを選択するためのさまざまな方法を示します。

プロセス ID が 1203 のジャーナル・ファイル内のすべてのレコードを選択して、JRNDUMP.OUT という名前の新しいファイルに出力を送信する場合は、以下のようになります。

```
%SYS>Do SELECT^JRNDUMP("C:\MyIRIS\mgr\journal\120020507.009","1203")
Device: SYS$LOGIN:JRNDUMP.OUT
Parameters: "RWN"=>
```

グローバル参照 ^ABC を含むジャーナル・ファイルのすべてのレコードを選択する場合は、以下のようになります。

```
DO SELECT^JRNDUMP("C:\MyIRIS\mgr\journal\20050327.001","", "", "", "^ABC", 1)
```

グローバル参照 ^ABC に完全に一致するレコードのみを選択する場合は、以下のようになります。

```
DO SELECT^JRNDUMP("C:\MyIRIS\mgr\journal\20050327.001","", "", "", "^ABC", 0)
```

注釈 ^ABC(1) または ^ABC(100) など、完全には一致しないレコードは選択されません。

グローバル ^ABC に対するローカルの Set 処理のレコードのみを選択する場合は、以下のようになります。

```
DO SELECT^JRNDUMP("C:\MyIRIS\mgr\journal\20050327.001","", "", "", "^ABC", "", "6")
```

グローバル ^ABC に対するローカルおよびリモートの Set 処理のレコードのみを選択する場合は、以下のようになります。

```
DO SELECT^JRNDUMP("C:\MyIRIS\mgr\journal\20050327.001","", "", "", "^ABC", "", "s")
```

## 8.1.7 PURGE^JOURNAL を使用したジャーナル・ファイルの削除

ファイルを削除するには、以下の例に示すように、PURGE^JOURNAL ルーチンを使用するか、または ^JOURNAL メニューの **6** プロンプトで「6」を入力します。

PURGE^JOURNAL を直接実行する例

```
set $namespace="%SYS"
%SYS>Do PURGE^JOURNAL
```

^JOURNAL メニューからジャーナリングを開始する例

```
%SYS>Do ^JOURNAL

...
6) Purge Journal Files (PURGE^JOURNAL)
...
Option? 6

1) Purge any journal NOT required for transaction rollback or crash recovery
2) Purge journals based on existing criteria (2 days or 2 backups)
Option?
```

このルーチンは、指定したオプションに対応して実行されるアクションについて報告します。以下に例を示します。

Option? 1

The following files have been purged (listed from latest to oldest):

```
3. c:\intersystems\iris\mgr\journal\20230714.001
2. c:\intersystems\iris\mgr\journal\20230713.001
1. c:\intersystems\iris\mgr\journal\20230710.003
```

削除するファイルがない場合、以下のメッセージが表示されます。

None purged

## 8.1.8 ^JRNOPTS を使用したジャーナル設定の更新

管理ポータルの[ジャーナル設定]ページを使用する代わりに、^JRNOPTS ルーチンを使用するか、JOURNAL メニューの [プロンプト](#) で「7」を入力して基本的なジャーナル構成設定を更新できます。設定を変更するには、プロンプトで新しい値を入力して、Enter キーを押します。以下に例を示します。

SYS>Do ^JRNOPTS

```
1) Primary Journal Directory: C:\MyIRIS\Mgr\Journal\
2) Alternate Journal Directory: D:\irissys\altjournal\
3) Journal File Size Limit (MB) 1024
4) Journal File Prefix:
5) Journal Purge Options: 2 days OR 2 backups, whichever comes first
6) Compress Journal files: Yes
```

疑問符 (?) を入力すると、ヘルプが表示されます。以下に例を示します。

```
Journal File Prefix: ?
Enter an alphanumeric string ('_' allowed) or . to reset prefix to null
```

設定を変更し、Change Property? プロンプトで Enter キーを押すと、変更内容が有効になります。

```
Change Property?
Save and activate changes? Yes =>
*** Journal options updated.
```

設定を変更しない場合、以下のメッセージが表示されます。

```
*** Nothing changed
```

## 8.1.9 ENCRYPT^JOURNAL を使用したジャーナルの暗号化

オプション 8) [\(ENCRYPT^JOURNAL\)](#) の詳細は、ジャーナル・ファイルの暗号化について詳しく説明している [“暗号化の起動設定の構成”](#) を参照してください。

注釈 ジャーナルの暗号化とジャーナルの圧縮 ([“ジャーナル設定の構成”](#) を参照) の両方が有効な場合、ジャーナル・データは常に、圧縮されてから暗号化されます。暗号化されたデータは圧縮されることはありません。

## 8.1.10 Status^JOURNAL によるジャーナルの状態の表示

オプション 9) [\(Status^JOURNAL\)](#) を選択すると、ジャーナルの状態に関する次の簡単な説明が表示されます。

- ・ 現在のジャーナルのディレクトリとその空き領域
- ・ 代替ジャーナルのディレクトリ (現在のジャーナルと異なる場合) とその空き領域



- ・ 現在のジャーナル・ファイル、その最大サイズ、および使用している領域
- ・ ジャーナルの状態。以下のいずれかの状態となります。
  - 有効
  - 無効 (停止)
  - I/O エラーのために無効 (中断)
  - I/O エラーのためにフリーズ
  - ジャーナルの切り替え中 (一時停止)

I/O エラーのために中断した状態とフリーズした状態はジャーナル状態としては同じですが、実行されるアクションは異なり、フリーズした場合はジャーナル・データが破棄されます。

- ・ 該当する場合、^JRNSTART、^JRNSTOP、または ^JRNSWTCH を実行しているすべてのプロセスのプロセス ID

以下に例を示します。

```
%SYS>Do ^JOURNAL

...
9) Display Journal status (Status^JOURNAL)
...
Option? 9

Current journal directory: C:\MyIRIS\Mgr\Journal\
Current journal directory free space (KB): 53503904
Alternate journal directory: C:\MyIRIS\Mgr\
Alternate journal directory free space (KB): 53503904
Current journal file: C:\MyIRIS\mgr\journal\20231129.001
Current journal file maximum size: 1073741824
Current journal file space used: 1979276
Journaling is enabled.
```

### 8.1.11 Manage^JRNROLL を使用したトランザクション・ロールバックの管理

InterSystems IRIS には、ジャーナルのレコードに対して部分的に完了したトランザクションをロールバックするための ^JRNROLL ユーティリティが用意されています。システムの起動時またはプライマリ・ミラー・メンバの起動時に保留中または処理中のトランザクション・ロールバックがある場合、Manage エントリ・ポイント (Manage^JRNROLL) を使用します。

トランザクション・ロールバックの管理を開始するには、以下の例に示すように、Manage^JRNROLL を実行するか、または ^JOURNAL メニューの `11` プロンプトで「11」を入力します。

```
%SYS>Do ^JOURNAL

...
11) Manage pending or in progress transaction rollback (Manage^JRNROLL)
...
Option? 11
```

オプション 11) (Manage^JRNROLL) を選択すると、以下のようなメッセージが表示されます。

```
Transaction rollback is pending or in progress
Do you wish to run Manage^JRNROLL? Yes => Yes
Rollback operations currently in progress
  ID   Phase      MB Remaining   Current Open Transaction Count
  1    scan       307           2
Rollback at system startup at 11/29/2023 15:54:35 (578MB)
20231129.004 has 2 open transaction(s) starting at offset 11303
2 file(s) remaining to process

1) Restart pending rollback
2) Interrupt transaction rollback
3) Redisplay rollback information
4) Discard pending rollback

Option?
```

このオプションによって、トランザクション・ロールバックの現在のフェーズ (スキャンまたはロールバック)、処理される残りのデータの量 (MB)、検出された開いているトランザクションの数などの状態が表示されます。

さらに、リストされたトランザクション・ロールバックを管理するためのサブオプションがリストされます。例えば、処理を中断することができます。この場合、その処理は“保留中”の処理としてキューに登録されるため、保留中のロールバックを再開することが可能です。

**注意** オプション 4) Discard pending rollback は元に戻せません。保留中のロールバックを永久に破棄しても問題がないことが確実である場合以外は、このオプションを使用しないでください。

**注釈** ミラーでは、トランザクション・ロールバックが 2 回実行されます。ミラーリングされないデータベースに対して 1 回 (システムの起動時) と、ミラーリングされるデータベースに対して 1 回 (システムがプライマリ・ミラー・メンバになったとき) です。その結果、プライマリ・ミラー・メンバの開始時に、ロールバックを 2 回中断することが必要になる場合があります。そうすると 2 つの処理が保留中になります。保留中の処理を再開すると、ミラー以外のロールバックとミラー・ロールバックが別々に実行されます。

ロールバックの間、一連の処理の (およそ) 10% ごとにメッセージがメッセージ・ログ (messages.log) に書き込まれ、処理のためにどのくらいの容量が残されているかと、開いているトランザクションがいくつリストされているかが示されます。

ジャーナル・ファイルが削除される場合、保留中のトランザクション・ロールバックに必要なファイルは保持されます (例えば、そうしなければそれらのファイルが削除されていた場合)。

## 8.1.12 MirrorCatchup^JRNRESTO を使用したミラーリングされるデータベースへのジャーナルのリストア

ミラー・ジャーナル・ファイルをミラーリングされるデータベースにリストアするには、JOURNAL メニューの `^JRNRESTO` プロンプトで「12」を入力するか、またはオプション 4 の `Restore Globals From Journal (^JRNRESTO)` を使用したときに `Catch-up mirrored databases?` プロンプトで「yes」と入力します。以下に例を示します。

```
%SYS>Do ^JOURNAL

...
12) Journal catch-up for mirrored databases (MirrorCatchup^JRNRESTO)
...
Option? Option? 12

Specify the list of mirrored databases you want to catch-up.
Enter database, * for all, ? for a list or to end list? *
Enter database or to end list?
Starting catch-up for the following mirrored database(s):
  sfn #6: c:\intersystems\iris\mgr\mirrordb3\
Catch-up succeeded.
```

ミラーリングされるデータベースをキャッチアップするには、ジャーナリングを実行している必要はありませんが、現在のジャーナル・ディレクトリがメモリから確実に利用できるように、少なくとも 1 回はジャーナリングを起動しておく必要があります。

## 8.2 ^STURECOV を使用した起動エラーの回復

InterSystems IRIS のスタートアップ・プロシージャでは、ジャーナルまたはトランザクションのリストア処理の際に <FILEFULL> や <DATABASE> などのエラーが発生した場合、そのエラーをメッセージ・ログ (messages.log) に記録して、システムをシングル・ユーザ・モードで起動します。

エラーを修復して InterSystems IRIS をマルチユーザ・モードで起動できるように、InterSystems IRIS にはユーティリティ ^STURECOV が用意されています。このルーチンには、いくつかのオプションがあります。これらのオプションを使用し、失敗した処理を再試行してシステムを起動する操作や、エラーを無視してシステムを起動する操作が可能です。ジャーナルのリストア段階では、起動が停止する前にできる限り多くの作業を実行しようとします。データベースから 4 件以上のエラーがトリガされると、そのデータベースのリカバリは中止され、マウントされないままとなります。

注釈 ^STURECOV ユーティリティは、保留中または処理中のトランザクション・ロールバックがあるミラー・メンバでは機能しません。これは、トランザクション・ロールバックが完了するまで、ミラーリングされるデータベースの読み取り/書き込みがシステムによって有効化されないためです。この場合、InterSystems IRIS では、Manage^JRNROLL ルーチンを実行できます。このルーチンによって、システムを強制的に起動して、トランザクション・ロールバック情報を格納することができ、システムの起動後にその情報を使用してトランザクションをロールバックすることが可能です。詳細は、["Manage^JRNROLL を使用したトランザクション・ロールバックの管理"](#) を参照してください。

トランザクションのロールバックでデータベースに最初に発生したエラーによって、今後のロールバック・プロセスではそのデータベースは処理されなくなります。このプロセスでは、そのデータベースを参照するトランザクションが完全な形では再生されません。リカバリ・プロセスの間、トランザクションはロールバックに備えて保存されます。

InterSystems IRIS の起動のデジャーナリング・フェーズで問題が発生した場合、以下のような一連のメッセージ・ログ・メッセージが生成されます。

```
08/10-11:19:47:024 ( 2240) System Initialized.
08/10-11:19:47:054 ( 2256) Write daemon started.
08/10-11:19:48:316 ( 1836) Performing Journal Recovery
08/10-11:19:49:417 ( 1836) Error in JRNRESTB: <DATABASE>restore+49^JRNRESTB
C:\MyIRIS\mgr\journal\20230810.004 addr=977220
^["^C:\MyIRIS\mgr\jnl666\" ]test(4,3,28)
08/10-11:19:49:427 ( 1836) Error in JRNRESTB: <DATABASE>restore+49^JRNRESTB
C:\MyIRIS\mgr\journal\20230810.004 addr=977268
^["^C:\MyIRIS\mgr\test\" ]test(4,3,27)
08/10-11:19:49:437 ( 1836) Error in JRNRESTB: <DATABASE>restore+49^JRNRESTB
C:\MyIRIS\mgr\journal\20230810.004 addr=977316
^["^C:\MyIRIS\mgr\test\" ]test(4,3,26)
08/10-11:19:49:447 ( 1836) Error in JRNRESTB: <DATABASE>restore+42^JRNRESTB
C:\MyIRIS\mgr\journal\20230810.004 addr=977748
^["^C:\MyIRIS\mgr\test\" ]test(4,2,70)
08/10-11:19:50:459 ( 1836) Too many errors restoring to C:\MyIRIS\mgr\test\.
Dismounting and skipping subsequent records
08/10-11:19:50:539 ( 1836) 4 errors during journal restore,
see console.log file for details.
Startup aborted, entering single user mode.
```

このエラーがトランザクションのロールバックで発生している場合、出力は以下のようになります。

```
08/11-08:55:08:732 ( 428) System Initialized.
08/11-08:55:08:752 ( 1512) Write daemon started.
08/11-08:55:10:444 ( 2224) Performing Journal Recovery
08/11-08:55:11:165 ( 2224) Performing Transaction Rollback
08/11-08:55:11:736 ( 2224) Max Journal Size: 1073741824
08/11-08:55:11:746 ( 2224) START: C:\MyIRIS\mgr\journal\20230811.011
08/11-08:55:12:487 ( 2224) Journaling selected globals to
```

```

C:\MyIRIS\mgr\journal\20230811.011 started.
08/11-08:55:12:487 ( 2224) Rolling back transactions ...
08/11-08:55:12:798 ( 2224) Error in %ROLLBACK: <DATABASE>set+2^%ROLLBACK
C:\MyIRIS\mgr\journal\20230811.010 addr=984744
^["^C:\MyIRIS\mgr\test\"]test(4,1,80)
08/11-08:55:12:798 ( 2224) Rollback of transaction for process id #2148
aborted at offset 984744 in C:\MyIRIS\mgr\journal\20230811.010.
08/11-08:55:13:809 ( 2224) C:\MyIRIS\mgr\test\ dismounted -
Subsequent records will not be restored
08/11-08:55:13:809 ( 2224) Rollback of transaction for process id #924
aborted at offset 983464 in C:\MyIRIS\mgr\journal\20230811.010.
08/11-08:55:14:089 ( 2224) STOP: C:\MyIRIS\mgr\journal\20230811.011
08/11-08:55:14:180 ( 2224) 1 errors during journal rollback,
see console.log file for details.
Startup aborted, entering single user mode.

```

どちらの出力表示も、次のような指示で終了します。

```

Enter IRIS with
C:\MyIRIS\bin\irisdb -sC:\MyIRIS\mgr -B
and D ^STURECOV for help recovering from the errors.

```

InterSystems IRIS が正常に起動できない場合、シングル・ユーザ・モードで起動します。このモードでは、これらの指示に示されているコマンドを実行して InterSystems IRIS を開始します（“[管理者ターミナル・セッション](#)”を参照）。

これで管理者のネームスペースに入ることができたので、スタートアップ・リカバリ・ルーチン ^STURECOV を実行できます。

```
Do ^STURECOV
```

以下のように、^STURECOV のジャーナル・リカバリ・メニューが表示されます。

```

Journal recovery options
-----
1) Display the list of errors from startup
2) Run the journal restore again
3) Bring down the system prior to a normal startup
4) Dismount a database
5) Mount a database
6) Database Repair Utility
7) Check Database Integrity
8) Reset system so journal is not restored at startup
9) Display instructions on how to shut down the system
10) Display Journaling Menu (^JOURNAL)
-----
H) Display Help
E) Exit this utility
-----

Enter choice (1-10) or [Q]uit/[H]elp?

```

このメニューのオプション 9 は、UNIX®/Linux システムにのみ表示されます。

ジャーナル・リストアやトランザクション・ロールバックで発生する障害の原因となっているエラーを修正した後、マルチユーザ・モードでシステムを起動します。実行できる処理として、以下のものがあります。

- ・ オプション 1 – ジャーナル・リストアとトランザクション・ロールバックのプロシージャで、エラーのリストを ^%SYS() グローバルに保存します。このオプションは必ず利用できるとは限りません。システムの不具合内容によっては利用できないこともあります。このエラー情報を入手できる場合は、そのエラーが表示されます。
- ・ オプション 2 – システム起動時に実行したものと同じジャーナル・リストアおよびトランザクション・ロールバックを実行します。データの量が少ないため、エラーが発生した時点から試して再起動する必要はありません。
- ・ オプション 3 – システムが使用できる状態になったらこのオプションを使用して、インスタンスをシャットダウンしてから、通常どおり再起動します。
- ・ オプション 4 – このオプションを使用するとデータベースをディスマウントできます。一般的には、ユーザがシステムを使用できるようにはするが、障害が残っているデータベースへのアクセスは禁止する場合に、このオプションを使用します（DISMOUNT ユーティリティ）。

- ・ オプション 5 – このオプションを使用するとデータベースをマウントできます (^MOUNT ユーティリティ)。
- ・ オプション 6 – このオプションを使用するとデータベース構造を編集できます (^REPAIR ユーティリティ)。
- ・ オプション 7 – このオプションを使用するとデータベース構造を検証できます (^INTEGRIT ユーティリティ)。
- ・ オプション 8 – 起動時にジャーナル・リストアやトランザクション・ロールバックを試行しないようにシステムを更新します。次の起動プロセス実行のときにのみ適用されます。ジャーナル・リカバリは完了できないが、ユーザにはシステムを使用できるようにする必要がある場合に、このオプションを使用します。回復されなかったデータベースをディスマウントすることを検討してください。この処理は元に戻せません。^JRNRESTO ユーティリティを使用すると、ジャーナル・リストアを手動で実行できます。
- ・ オプション 9 – このユーティリティからはシステムをシャットダウンできませんが、このオプションでは、UNIX® のコマンド行からシステムをシャットダウンする方法が表示されます。
- ・ オプション 10 – ジャーナリング・メニューを表示し、ジャーナル・ファイルの閲覧とリストアができるようにします。ジャーナリングを開始および停止するオプションはありますが、これらは通常、起動時のジャーナリングに関する問題の解決には関係ありません。

問題を解決するうえで必要な修正アクションであれば、どのようなものでも実行する必要があります。このアクションとしては、例えば ^DATABASE ルーチンを使用してデータベースの最大サイズを拡張する処理があります。また、ファイル・システムで領域を解放する処理や、^INTEGRIT ユーティリティおよび ^REPAIR ユーティリティを使用してデータベースの破損を検出、修復する処理が必要になることもあります。これらの作業と並んで、^STURECOV ユーティリティのオプション 2 を使用し、必要に応じてジャーナル再生やトランザクション・ロールバックをやり直すことができます。オプション 1 を使用して、システムの起動時からのエラーも含め、発生したすべてのエラーを表示できます。すべての不具合を修正し、オプション 2 を実行してエラーが発生しなければ、オプション 3 を使用してシステムをマルチユーザ・モードで起動します。

不具合は解決できていないが、システムを起動することが必要な場合は、オプション 8 を使用し、起動時にジャーナル・リストアとトランザクション・ロールバックをトリガする InterSystems IRIS イメージ・ジャーナル (.wij ファイル) 内の情報をクリアします。また、このオプションでは、現時点のこの情報がメッセージ・ログに記録されます。以上の作業が完了した後、オプション 3 を使用してシステムを起動します。この方法では結果を元に戻せないため、実行には注意が必要です。

InterSystems IRIS の起動時に、^STURECOV で表示するためのエラー情報を ^%SYS() グローバルに保存できない場合、メニューが表示される前に以下のようなメッセージが表示される場合があります。

```
There is no record of any errors during the prior startup
This could be because there was a problem writing the data
Do you want to continue ? No => yes
Enter error type (? for list) [^] => ?
```

```
Supported error types are:
JRN - Journal and transaction rollback
```

```
Enter error type (? for list) [^] => JRN
```

ジャーナリング・エラーは、このトピックで扱う、このユーティリティが対応しているエラーの 1 種類です。他のエラーの種類については、ドキュメントの該当するセクションで説明します。

**注意** 起動時にエラーが発生して、システムがシングル・ユーザ・モードで動作している場合にのみ、^STURECOV ユーティリティを使用します。システムがシングル・ユーザ・モード以外の状態にあるとき (正常に動作しているときなど) にこのユーティリティを使用すると、データに重大な破損が発生することがあります。それは、このユーティリティでは必要に応じてジャーナル情報がリストアされますが、そのリストアされた情報が最新のものではないことがあるからです。^STURECOV ユーティリティでは警告が表示されますが、そのまま実行に移ることができます。

## 8.3 ^JCONVERT および ^%JREAD を使用したジャーナル・ファイルの変換

^JCONVERT ルーチンは、ジャーナル・ファイルを読み取り、それを可変レコード形式で共通ファイルに変換するユーティリティです。^%JREAD ユーティリティは、このファイルを読み取り、ジャーナル・トランザクションを別のシステム上のデータベースに適用できます。^JCONVERT ユーティリティは、InterSystems IRIS のすべてのバージョンのほか、インターシステムズの以前のデータベース製品にも用意されています。これらのユーティリティを使用して、互換性のあるジャーナル・ファイルを持たない、異なるシステムのバージョン間でジャーナル・データを移動します。

例えば、新しいバージョンの InterSystems IRIS に変換する際に停止時間を最小限とするには、以下の手順を実行します。

1. 旧システムのジャーナリングを有効にします。
2. 旧システムでバックアップを実行します。これによって、旧システムのジャーナル・ファイルが新しいファイルに切り替わります。
3. 旧システムのジャーナリングを継続します。
4. 旧システムのバックアップを新システム上にリストアし、必要な変換処理を実行します。
5. 旧システムを停止し、バックアップ以降に旧システム上に作成されたジャーナル・ファイルに対して ^JCONVERT を実行します。
6. ^JCONVERT の実行で作成されたファイルを新システム上で ^%JREAD への入力として使用し、旧システムのトランザクションを新システムに適用します。

^JCONVERT ユーティリティでは、ジャーナル・リストア・ユーティリティと同じプロセスを使用して、処理対象のジャーナル・ファイルを選択し、フィルタ処理します。複数のジャーナル・ファイルを入力として指定し、1つの出力ファイルを作成できます。ジャーナル・ファイルの選択とフィルタ処理の詳細は、“[^JRNRESTO を使用したジャーナル・ファイルからのグローバルのリストア](#)”を参照してください。

変換されたファイルは可変レコード形式です。既定の文字エンコードは UTF8 で、これはすべてのプラットフォームでの現行の ^%JREAD ユーティリティと互換性があります。また、バイナリ FTP を使用して、複数のプラットフォーム間で移動できます。[UTF8 文字変換を使用しますか?] プロンプトに [いいえ] と答える場合、文字エンコードは適用されません。

ジャーナル・ファイルのグローバルは、そのグローバル参照に特定のディレクトリ参照が付加された状態で格納されています。変換先のファイルにこのディレクトリ参照を追加するかどうかを選択できます。ディレクトリ参照を追加しておくと、後で ^%JREAD プロシージャを実行して、いつでもディレクトリ参照をフィルタで除外したり、変更したりできます。

ディレクトリ参照は、ターゲット・システム上で ^%JREAD がグローバルを設定する場所を指定します。ディレクトリ参照を追加していない場合、^%JREAD では現在のディレクトリにすべてのセットが作成されます。ディレクトリ参照を追加していない場合は、ユーザ指定の ^%ZJREAD プログラムでディレクトリが変換されない限り、このユーティリティではソース・システムと同じディレクトリにセットが作成されます。ターゲット・システムが別のオペレーティング・システム上にある場合、またはデータベースがターゲット・システム上の別のディレクトリに存在する場合は、ディレクトリ参照を変換する ^%ZJREAD ルーチンを指定する必要があります。

^%JREAD ルーチンは、共通ジャーナル・ファイル形式を読み取り、ターゲット・システム上のデータベースにジャーナル・トランザクションを適用します。レコードをインポートするとき、^%ZJREAD ルーチンが存在すれば、ジャーナル・トランザクションごとにこのユーティリティから呼び出されます。これにより、そのジャーナル・レコードを操作できます。^%ZJREAD では、以下の変数を参照できます。

```

type      - Transaction type
gref      - Global reference
value     - Global value
%^ZJREAD - 1:Apply transaction, 0:Do not apply transaction

```



トランザクションを適用しない場合は、変数 %ZJREAD を 0 (ゼロ) に設定してそのレコードをスキップします。他の変数を変更することもできます。例えば、gref を修正することでディレクトリ指定を変更できます。

^%JREAD ルーチンの例を以下に示します。このルーチンは %SYS("JOURNAL" への更新を含むトランザクションを検索し、それが適用されないようにします。このルーチンをコピーし、目的に応じて修正して使用できます。

## ObjectScript

```
%ZJREAD;
/*The following variables are defined; you can modify them
   before the transaction gets applied

       type - Transaction type
       gref - Global reference
       value - Global value
   %ZJREAD - 1:Apply transaction, 0:Do not apply transaction
*/
If gref["SYS("JOURNAL" Set %ZJREAD=0
Quit
```

## ^JCONVERT の実行例

以下は、^JCONVERT ユーティリティの実行例です。

```
%SYS>Do ^JCONVERT
```

```
Journal Conversion Utility  [ IRIS Format --> Common Format ]
```

```
The converted file will be in variable record format.
The default character translation UTF8 is compatible with current ^%JREAD
on all platforms and can be moved among platforms with binary FTP.
If you answer NO, no character translation will be applied.
```

```
Use UTF8 character translation? <Yes>
```

```
Globals in the journal file are stored with a specific directory reference
appended to the global reference. You can choose either to include
the directory reference in the converted file, or exclude it. Note that
if you include it, you can always filter it out or change it later during
the %JREAD procedure. The directory reference determines where ^%JREAD sets
the global on the target system. If the directory reference is not included,
all sets are made to the current directory. If the directory reference is
included, sets will be made to the same directory as on the source system
unless translated by a ^%ZJREAD program you supply. If the target system
is on a different operating system or the databases reside in different
directories on the target system, the ^%ZJREAD program must be used to
```



translate the directory reference.

Include the directory reference? <Yes>

Enter common journal file name: common.jrn

Common journal file: common.jrn

Record separator: Variable

Directory reference: Yes

Use current journal filter (ZJRNFLT)? no

Use journal marker filter (MARKER^ZJRNFLT)? no

Process all journaled globals in all directories? enter Yes or No, please

Process all journaled globals in all directories? yes

Specify range of files to process (names in YYYYMMDD.NNN format)

from: <20231201.001> [?] => 20231202.001

through: <20231204.001> [?] =>

Prompt for name of the next file to process? No => No

Provide or confirm the following configuration settings:

Journal File Prefix: =>

Files to dejournal will be looked for in:

C:\MyIRIS\mgr\journal\

C:\MyIRIS\mgr\

in addition to any directories you are going to specify below, UNLESS

you enter a minus sign ('-' without quotes) at the prompt below,

in which case ONLY directories given subsequently will be searched

Directory to search: <return when done>

Here is a list of directories in the order they will be searched for files:

C:\MyIRIS\mgr\journal\

C:\MyIRIS\mgr\

You may tailor the response to errors by choosing between the alternative actions described below. Otherwise you will be asked to select an action at the time an error actually occurs.

Either Continue despite database-related problems (e.g., a target database is not journaled, cannot be mounted, etc.), skipping affected updates

or Abort if an update would have to be skipped due to a database-related problem (e.g., a target database is not journaled, cannot be mounted, etc.)

Either Abort if an update would have to be skipped due to a journal-related problem (e.g., journal corruption, some cases of missing journal files, etc.)

or Continue despite journal-related problems (e.g., journal corruption, some missing journal files, etc.), skipping affected updates

Either Apply sorted updates to databases before aborting

or Discard sorted, not-yet-applied updates before aborting (faster)

Would you like to specify error actions now? No => yes

1. Continue despite database-related problems (e.g., a target database is not journaled, cannot be mounted, etc.), skipping affected updates

2. Abort if an update would have to be skipped due to a database-related problem (e.g., a target database is not journaled, cannot be mounted, etc.)

Select option [1 or 2]: 1

1. Abort if an update would have to be skipped due to a journal-related problem (e.g., journal corruption, some cases of missing journal files, etc.)

2. Continue despite journal-related problems (e.g., journal corruption, some missing journal files, etc.), skipping affected updates

Select option [1 or 2]: 2

1. Apply sorted updates to databases before aborting
2. Discard sorted, not-yet-applied updates before aborting (faster)

Select option [1 or 2]: 2

Based on your selection, this restore will

\*\* Continue despite database-related problems (e.g., a target database is not journaled, cannot be mounted, etc.), skipping affected updates

\*\* Continue despite journal-related problems (e.g., journal corruption, some missing journal files, etc.), skipping affected updates

\*\* Discard sorted, not-yet-applied updates before aborting (faster)

C:\MyIRIS\mgr\journal\20231202.001

```
13.98% 14.93% 15.95% 17.14% 18.25% 19.27% 20.49% 21.63% 22.65% 23.84%
24.99% 25.97% 27.10% 28.25% 29.31% 30.50% 31.72% 32.84% 33.84% 34.84%
35.84% 36.85% 37.91% 38.99% 40.10% 41.08% 42.03% 42.97% 43.93% 44.94%
45.95% 47.05% 48.11% 49.07% 50.04% 51.02% 52.03% 53.07% 54.14% 55.25%
56.21% 57.17% 58.15% 59.14% 60.18% 61.24% 62.33% 63.28% 64.20% 65.15%
66.10% 67.11% 68.13% 69.05% 69.94% 70.83% 71.61% 72.41% 73.09% 73.85%
74.59% 75.32% 76.06% 76.75% 77.73% 78.70% 79.65% 80.59% 81.53% 82.46%
83.40% 84.33% 85.27% 86.05% 86.59% 87.13% 87.67% 88.23% 88.78% 89.34%
89.89% 90.61% 93.28% 94.38% 97.12% 98.21% 99.93%100.00%
```

\*\*\*Journal file finished at 11:31:36

C:\MyIRIS\mgr\journal\20231203.001

```
14.01% 14.96% 15.98% 17.18% 18.29% 19.31% 20.53% 21.67% 22.69% 23.88%
25.03% 26.01% 27.15% 28.30% 29.36% 30.55% 31.78% 32.90% 33.90% 34.90%
35.91% 36.92% 37.99% 39.06% 40.17% 41.16% 42.11% 43.05% 44.01% 45.03%
46.04% 47.14% 48.20% 49.17% 50.14% 51.11% 52.13% 53.17% 54.25% 55.36%
56.33% 57.29% 58.27% 59.26% 60.30% 61.36% 62.46% 63.40% 64.33% 65.28%
66.23% 67.24% 68.26% 69.19% 70.08% 70.97% 71.76% 72.56% 73.25% 74.01%
74.75% 75.47% 76.22% 76.91% 77.89% 78.87% 79.83% 80.77% 81.70% 82.64%
83.58% 84.52% 85.46% 86.24% 86.78% 87.32% 87.87% 88.42% 88.98% 89.53%
90.09% 90.81% 93.49% 94.59% 97.33% 98.42%100.00%
```

\*\*\*Journal file finished at 11:31:37

C:\MyIRIS\mgr\journal\20231204.001

```
13.97% 14.92% 15.93% 17.12% 18.24% 19.25% 20.47% 21.61% 22.62% 23.82%
24.96% 25.94% 27.07% 28.22% 29.28% 30.46% 31.69% 32.80% 33.80% 34.80%
35.80% 36.81% 37.87% 38.94% 40.05% 41.04% 41.98% 42.92% 43.88% 44.89%
45.90% 47.00% 48.06% 49.02% 49.98% 50.96% 51.97% 53.01% 54.08% 55.19%
56.15% 57.11% 58.08% 59.07% 60.12% 61.17% 62.26% 63.20% 64.13% 65.07%
66.02% 67.03% 68.05% 68.97% 69.86% 70.75% 71.53% 72.33% 73.01% 73.77%
74.51% 75.23% 75.98% 76.67% 77.64% 78.61% 79.56% 80.50% 81.43% 82.37%
83.30% 84.24% 85.17% 85.95% 86.49% 87.03% 87.57% 88.13% 88.68% 89.23%
89.79% 90.51% 93.18% 94.27% 97.01% 98.10% 99.81%100.00%
```

\*\*\*Journal file finished at 11:31:38

[journal operation completed]  
Converted 26364 journal records

## 8.4 ^JRNMARK を使用したジャーナル・マーカの設定

ジャーナル・ファイルにジャーナル・マーカを設定するには、以下のルーチンを使用します。

```
SET rc=$$ADD^JRNMARK(id,text)
```

引数	説明
id	マーカ ID (例 : バックアップは -1)
text	256 文字以内の文字列のマーカ・テキスト (例えば、バックアップについては “timestamp”)

引数	説明
rc	マーカのジャーナル位置(コンマで区切られたジャーナル・オフセットとジャーナル・ファイル名)、または、処理が失敗した場合、コンマの後にエラーを説明するメッセージが付いた負のエラーコード。ジャーナル・オフセットは正の数である必要があります。

## 8.5 ^JRNUTIL を使用したジャーナル・ファイルの操作

インターシステムズは、^JRNUTIL ルーチンでいくつかの機能を提供します。サイト特有のルーチンを記述するこれらの機能を使用して、ジャーナル・レコードとファイルを操作できます。

以下のテーブルは、このルーチンで使用可能な関数の一覧です。

テーブル 8-3: ^JRNUTIL で使用可能な関数

ジャーナリング・タスク	関数構文
ジャーナル・ファイルを閉じる	\$\$CLOSEJRN^JRNUTIL(jrnfile)
ジャーナル・ファイルの削除	\$\$DELFILE^JRNUTIL(jrnfile)
ジャーナル・ファイルからローカル配列にレコードの読み取り	\$\$GETREC^JRNUTIL(addr,jrnode)
別のジャーナル・ファイル・ディレクトリへの切り替え	\$\$JRNSWCH^JRNUTIL(newdir)
ジャーナル・ファイルを開く	\$\$OPENJRN^JRNUTIL(jrnfile)
開かれているジャーナル・ファイルの使用	\$\$USEJRN^JRNUTIL(jrnfile)

**重要** DELFILE^JRNUTIL 関数は、ジャーナル・ファイルを削除する前に、開いているトランザクションがあるかどうかをチェックしません。

以下のテーブルは、このユーティリティで使用する引数です。

引数	説明
addr	ジャーナル・レコードのアドレス
jrnfile	ジャーナル・ファイルの名前
newdir	新しいジャーナル・ファイル・ディレクトリ
jrnode	ジャーナル・レコード情報を返すために、参照から渡されるローカル変数

## 8.6 DISABLE^%NOJRN を使用したプロセス・レベルでのジャーナリングの管理

ジャーナリングがシステム全体で有効になっている場合、特定のプロセス内のグローバルに対して Set 処理および Kill 処理を実行するために、以下のようにアプリケーション内またはプログラマ・モードからユーティリティ^%NOJRN を呼び出して、ジャーナリングを停止できます。

```
%SYS>DO DISABLE^%NOJRN
```

ジャーナリングは、以下のイベントのいずれかが発生するまで無効になっています。

- ・ プロセスが中断
- ・ ジャーナリングを再度アクティブにするために、プロセスが以下の呼び出しを発行

```
%SYS>DO ENABLE^%NOJRN
```

注釈 DISABLE^%NOJRN を使用してジャーナリングを無効にしても、ミラーリングされたデータベースに影響はありません。

DISABLE^%NOJRN を使用するには、少なくとも **%Admin\_Manage** リソースへの読み取りアクセスが必要です。

## 8.7 ^ARCHIVE を使用したジャーナル・アーカイブ・ファイルの管理

**%SYS** ネームスペース内で使用できる ^ARCHIVE ルーチンを使用して、ジャーナル・アーカイブ・ファイルを管理できます。このルーチンは、以下のメニューを提供します。

```
%SYS>do ^ARCHIVE
```

- 1) Copy a file to an archive target
- 2) Retrieve an archived file
- 3) Delete an archived file
- 4) Manage archive targets

Option?

API については、クラス “**SYS.ArchiveManager**” を参照してください。

## 8.8 関連項目

- ・ [ジャーナリングの概要](#)
- ・ [ジャーナリングの構成](#)
- ・ [基本的なジャーナリング操作](#)
- ・ [ジャーナル入出力エラー](#)

- ・ ジャーナリングに関する特別な考慮事項

# 9

## ジャーナル入出力エラー

ここでは、ジャーナル・ファイルの入出力エラーが発生したときに、InterSystems IRIS® データ・プラットフォームがどのように応答するかを説明します。

応答は [エラー時に凍結] のジャーナル設定によって異なります。この設定は、管理ポータルの [ジャーナル設定] ページにあります。[エラー時に凍結] 設定は次のように機能します。

- ・ [エラー発生時に凍結する] 設定が [いいえ] (既定) の場合、ジャーナル・デーモンは成功するまで、またはいくつかの条件が満たされるまで、失敗した操作を再試行します。成功または条件が満たされた時点で、すべてのジャーナリングは無効化されます。このアプローチにより、システムは引き続き利用可能になりますが、ジャーナリングを無効にすることでデータの整合性と回復性が損なわれます。
- ・ [エラー発生時に凍結する] が [はい] に設定されている場合、すべてのジャーナリングされたグローバル更新が凍結されます。これにより、システムは使用できなくなりますが、データの整合性は保護されます。

ローカル・トランザクションのロールバックに失敗すると、[エラー発生時に凍結する] 設定もアプリケーションの動作に影響します。

このページの情報を使用して、業務上の必要性を確認し、環境に最適な手法を取ることをお勧めします。

### 9.1 ジャーナルの [エラー時に凍結] 設定が [いいえ] の場合

ジャーナル・ファイル入出力エラー時にフリーズしないよう InterSystems IRIS を構成している場合、ジャーナル・デーモンは、成功するまで、または以下の条件のいずれかが満たされるまで、定期的に (通常 1 秒ごと) 失敗した操作を再試行します。

- ・ デーモンが事前設定された期間 (通常 150 秒) 操作を再試行した
- ・ システムがそれ以上ジャーナリングされた更新をバッファできない

上記の条件のいずれかが満たされると、ジャーナリングは無効化され、データベースの更新がジャーナリングされなくなります。結果として、ジャーナルは、システムがクラッシュしたときにデータベースを回復するためソースとしては、信頼性に欠けるものになります。ジャーナリングが無効になっている場合は、以下の状況が発生します。

- ・ トランザクションのロールバックは失敗し、<ROLLFAIL> エラーが生成され、トランザクションが部分的にコミットされた状態になります。
- ・ コミットされていないデータのクラッシュ回復機能が無効になります。
- ・ 完全に回復するためのデータが存在しなくなります。前回のバックアップの状態にのみ回復できます。
- ・ ECP のロック機能とトランザクションの回復可能性から信頼性が失われます。



- ・ システムがクラッシュすると、ジャーナリングが無効になる前に始まっていたトランザクションが不完全な状態になっていても、InterSystems IRIS が起動するときのリカバリ・プロセスではそのトランザクションはロールバックされません。これは、トランザクションはコミットされていてもジャーナリングされていないからです。

### ジャーナリングが無効になった場合の措置

ジャーナリングが無効になった場合に実行する手順を以下にまとめます。

1. 問題の解決 – 可能な限り早急に、ジャーナリングの無効化を引き起こした問題を解決します。
2. ジャーナル・ファイルの切り替え – ジャーナル・デーモンは、失敗した入出力処理を定期的に再試行し、ジャーナリングが無効になる前に蓄積されたジャーナル・データを保持しようとします。必要に応じ、ジャーナル・ファイルを新しいディレクトリに切り替えてエラーを解決できます。ただし、InterSystems IRIS は、失敗した入出力の処理に成功してジャーナリングを新しいファイルに切り替えても、ジャーナリングを自動的に再び有効にすることはありません。また、ジャーナル・ファイルを手動で切り替えた場合も、ジャーナリングが自動的に再び有効になることはありません。
3. データベースのバックアップ – メイン・サーバのデータベースをバックアップします (バックアップを実行すると、ジャーナリングは自動的に再び有効になります)。

エラーの発生後は、データの損失を防止するために、できる限り早い時点でデータベースをバックアップすることを強くお勧めします。実際に、ジャーナリングの無効化の原因となった入出力エラーが解決済みで、ユーザに適切な特権があれば、そのエラーでジャーナリングが無効になっている状態でオンライン・バックアップを実行すると、ジャーナリングが自動的に再開されます。`JRNSTART` を実行してジャーナリングを有効にすることもできます。

バックアップが正常に完了してジャーナリングが再開されると、InterSystems IRIS では、保留中のすべてのジャーナル入出力が破棄されます。これは、保留中のジャーナル入出力で扱われているデータベース更新はすべて、正常に完了したバックアップに記録されているからです。

重要          ジャーナリングを開始するには、バックアップを実行する場合よりも高い特権が必要です。

## 9.2 ジャーナルの [エラー発生時に凍結する] 設定が [はい] の場合

ジャーナル・ファイルに入出力エラーが発生したときにフリーズするように InterSystems IRIS を構成しておくと、すべてのジャーナリングされたグローバル更新は、そのようなエラーの発生時に直ちにフリーズされます。これにより、システムの可用性は低下しますが、ジャーナル・データの喪失を防止できます。ジャーナル・デーモンがジャーナル書き込みを最低 30 秒間完了できない場合にも、グローバル更新はフリーズされます。

ジャーナル・デーモンは失敗した入出力操作を再試行し、成功するとグローバル更新のフリーズを解除します。グローバル更新がフリーズしている間、その他のジョブも停止されます。一般的な結果として、ジャーナリングの問題が解決するまで InterSystems IRIS は停止したままとなり、エンドユーザにはシステムの運用が停止したように見えます。InterSystems IRIS が停止している間に問題の対策をとることができます。例えば、ディスクの領域の解放、別のディスクへのジャーナル切り替え、ハードウェア障害の修正などの措置をとります。

このオプションの利点は、問題が解決して InterSystems IRIS が通常の動作を再開した時点で、失われるジャーナル・データは存在しないことです。欠点は、問題を解決している間、システムの可用性が低下するか、まったく使用できなくなることです。

失敗した入出力処理をジャーナル・デーモンが再試行している間、InterSystems IRIS はアラート (深刻度 3) を `messages.log` ファイルに定期的に記録します。

## 9.3 ジャーナルの [エラー発生時に凍結する] 設定の TROLLBACK を使用したトランザクションのロールバックに対する影響

選択した [エラー発生時に凍結する] 設定がジャーナリングと関係のないアプリケーションの動作に大きな意味を持つ可能性があるということを認識しておくことが重要です。アプリケーションが TROLLBACK コマンドを使用して開いているトランザクションをロールバックしようとして ("[TROLLBACK](#)" を参照)、これに失敗すると、ジャーナル入出力エラー発生時に直面するトレードオフ（データの整合性と可用性）と同じトレードオフが発生します。ジャーナリングと同様に、TROLLBACK は [エラー時に凍結] 設定を使用して、以下のように適切な動作を決定します。

- ・ [エラー発生時に凍結する] 設定が [いいえ] (既定) の場合、トランザクションおよび TROLLBACK を開始したプロセスがエラーを受け取り、そのトランザクションが閉じられ、そのトランザクションに対して保持されているロックが解放されます。このアプローチにより、アプリケーションは引き続き利用可能になりますが、データの整合性と回復性が損なわれます。
- ・ [エラー発生時に凍結する] が [はい] に設定されている場合、開始したプロセスは停止し、CLNDMN は開いているトランザクションのロールバックを繰り返し試行します。CLNDMN が再試行している間、トランザクションに対して保持されているロックは変更されませんが、アプリケーションは停止する場合があります。これにより、アプリケーションは使用できなくなりますが、データの整合性は保護されます。

(メッセージ・ログで報告されるように) 停止したジョブの開いているトランザクションのロールバックを CLNDMN が何度も試行して失敗している場合、Manage<sup>®</sup>CLNDMN ユーティリティを使用して、トランザクションを手動で終了できます。

注釈 [エラー発生時に凍結する] 設定はローカルの (ECP でない) トランザクションのロールバックにのみ影響します。

## 9.4 関連項目

- ・ [ジャーナリングの概要](#)
- ・ [ジャーナリングの構成](#)
- ・ [基本的なジャーナリング操作](#)
- ・ [ジャーナリング・ユーティリティ](#)
- ・ [ジャーナリングに関する特別な考慮事項](#)



# 10

## ジャーナリングに関する特別な考慮事項

ここでは、InterSystems IRIS® データ・プラットフォームでジャーナリングを使用する方法に影響する可能性がある、特別な考慮事項について説明します。

### 10.1 パフォーマンス

データベースの整合性を確保するうえでジャーナリングは重要ですが、ジャーナルの対象となるグローバルの更新数によっては、ディスク領域が消費され、性能が低下します。

変更がデータベースとジャーナル・ファイルの両方に記録されるため、更新処理は二重に行われます。この結果、ジャーナリングは性能に影響を与えます。InterSystems IRIS はフラット・ファイルでのジャーナリングを使用しているため、性能の低下が最小限になります。

### 10.2 UNIX® ファイル・システムの推奨事項

“サポートされているファイル・システム” では、UNIX®/Linux プラットフォームでインターシステムズが推奨しサポートしているファイル・システムの概要を示していますが、ここには、最適なジャーナル・パフォーマンスを実現するためのマウント・オプションに関する注意事項も記載されています。

**注釈** 推奨されるマウント・オプションが存在しないファイル・システムでプライマリまたは代替のジャーナル・ディレクトリを構成すると、以下のようなメッセージがメッセージ・ログに入力されます。

The device for the new journal file was not mounted with a recommended option (cio).

### 10.3 システム時計の推奨事項

InterSystems IRIS でサポートされるオペレーティング・システムのすべてに Network Time Protocol (NTP) クライアントがあり、システム時計と参照システムの同期が保持され、サマータイムと標準時間との間でシステム時計を自動調整する機能があります。

システム時計の同期の保持および管理には、システム時計を手動で調整するのではなく、オペレーティング・システムの自動クロック管理機能を利用することをお勧めします。

テスト作業などで手動で時間を調整する必要がある場合は、実稼働環境ではなく、必ずテスト環境を使用してその作業を実行してください。また、時計を進めたり、戻したりする調整などの非時系列的イベントは一部のユーティリティで障害発生の原因になる可能性があるため、手動の調整は慎重に行う必要があります。

## 10.4 ファイリング操作に対するジャーナリングの無効化

オブジェクトの保存や削除などのファイリング操作に対するジャーナリングを無効にすることが効果的な場合や必要な場合があります。共有するには、以下の 2 つの方法があります。

- ・ オブジェクトを開く (通常は %OpenId または %Open を使用します) ときに、並行処理の値として 0 を指定します。0 より大きい並行処理の値を指定してそのオブジェクトを既に開いている場合は、並行処理の値に 0 を指定しても効果はありません。
- ・ 現在のプロセスに対するオブジェクト・ファイラ・トランザクション処理を中断します。これを行うには、`$system.OBJ.SetTransactionMode(0)` を呼び出します (これは **%SYSTEM.OBJ** クラスの `SetTransactionMode` メソッドなので、専用の `$system` オブジェクトを介して呼び出すことができます)。`SetTransactionMode` メソッドは 0 または 1 の値をとり、オブジェクト・ファイラ・トランザクションは 0 で無効、1 で有効になります。この設定は現在のファイリング操作だけでなく、プロセス全体に影響します。

**重要**      ジャーナリングの無効化が求められることもありますが、実際にその必然性があるかどうかを確認してください。この確認を怠ると、必要なデータが部分的に欠損したジャーナルが生成され、そのデータは永久に失われる可能性があります。

## 10.5 関連項目

- ・ [ジャーナリングの概要](#)
- ・ [ジャーナリングの構成](#)
- ・ [基本的なジャーナリング操作](#)
- ・ [ジャーナリング・ユーティリティ](#)
- ・ [ジャーナル入出力エラー](#)

# 11

## 複数のシステムでのデータ整合性

ミラーリング、またはその他のメカニズムを使用して、別のシステムでデータのコピーを保持する場合、2 つのシステム間のデータの整合性を確認することができます。DataCheck では、この確認、および一時的な不一致の再確認も行います。

### 11.1 DataCheck の概要

DataCheck は、2 つのシステム (DataCheck のソースと DataCheck の宛先) にあるデータの状態を比較し、それらが一致しているかどうかを判断するメカニズムを提供します。チェックのすべての構成、動作制御および結果は、宛先システム上で提供され、ソース・システムは基本的に受動的です。

DataCheck の宛先として機能する InterSystems IRIS® のインスタンスで、DataCheck の宛先構成を作成する必要があります。同じインスタンスに複数の宛先構成を作成して、複数のソース・システムに対してデータをチェックするように構成する (または、1 つのソースに対してさまざまなデータをチェックするように構成する) ことができます。DataCheck を使用してミラーの整合性をチェックする場合の詳細は、“[ミラー用の DataCheck の構成](#)” を参照してください。

DataCheck のトピックについては、以下のサブセクションで詳しく説明します。

- ・ [DataCheck クエリ](#)
- ・ [DataCheck ジョブ](#)
- ・ [DataCheck 結果](#)
- ・ [DataCheck ワークフロー](#)

#### 11.1.1 DataCheck クエリ

宛先システムは、DataCheck “クエリ” と呼ばれる作業ユニットをソース・システムに送信します。各クエリは、データベース、最初のグローバル参照、ノード数、およびターゲット・グローバル参照を指定します。両方のシステムで、最初のグローバル参照から始めて指定数のグローバル・ノードを検索し、グローバル・キーと値をハッシュすることにより、回答を計算します。回答が一致した場合、宛先システムはその結果を記録し、ノード数を増やし、最初のグローバル参照を高度化してそのクエリを再送信します。回答が一致しなかった場合、クエリは、その不一致が最小の構成クエリ・サイズに隔離されるまでノード数を減らして再送信されます。

まだ処理されていないグローバル (添え字の包含/除外範囲が使用されている場合はグローバル範囲)、DataCheck により現在処理されているアクティブ・クエリなど、宛先システムによって送信されたクエリに関する情報を表示するには、`DATACHECK ルーチンの [\[詳細表示\]](#) サブメニューの [\[クエリ表示\]](#) オプションを使用できます。

## 11.1.2 DataCheck ジョブ

各クエリへの回答は、ソース・システムと宛先システムの両方で実行される DataCheck ワーカ・ジョブにより計算されます。ワーカ・ジョブの数は、宛先システムの動的に調整可能なパフォーマンス設定により決定されます。詳細は、この章の“[パフォーマンスに関する考慮事項](#)”を参照してください。

ワーカ・ジョブに加え、各システムにはその他のジョブがあります。宛先システムでは、さらに以下のジョブが実行されます。

- ・ マネージャ・ジョブ – クエリのロードと配信、クエリの回答の比較、およびワークフロー・フェーズでの進捗管理を行います。このジョブは、ソース・システムのマネージャ・ジョブに接続されます。
- ・ レシーバ・ジョブ – ソース・システムからの回答を受信します。

ソース・システムでは、さらに以下のジョブが実行されます。

- ・ マネージャ・ジョブ – 宛先システムのマネージャ・ジョブから要求を受信し、これらをワーカ・ジョブに送信します。
- ・ センダ・ジョブ – ワーカ・ジョブからのクエリの回答を受信し、これらを宛先システムのレシーバ・ジョブに送信します。このジョブは、宛先システムのレシーバ・ジョブに接続されます。

## 11.1.3 DataCheck 結果

チェックの結果では、以下のいずれかの状態のグローバル添え字範囲をリストします。

- ・ [不明] – DataCheck は、まだこの範囲にチェックを付けていません。
- ・ [一致] – DataCheck は、この範囲が一致していることを確認しました。
- ・ [不一致] – DataCheck は、この範囲内で不一致を検出しました。
- ・ [照合不一致] – ソース・システムと宛先システムの間の照合が異なることがグローバルで検出されました。
- ・ [除外] – この範囲はチェックから除外されています。

現在のチェック結果および前回のチェックの最終結果は、宛先システムで表示することができます。詳細は、**SYS.DataCheck.RangeList** クラスを参照してください。DataCheck 内のすべての添え字範囲は、範囲の最初を含みますが、範囲の最後を含みません。添え字範囲の詳細は、この章の“[チェックするグローバルおよび添え字範囲の指定](#)”を参照してください。

チェック結果の例を次に示します。

```
c:\InterSystems\iris\mgr\mirror2 ^XYZ          Unmatched
^XYZ --Matched--> ^XYZ(3001,4)
^XYZ(3001,4) --Unmatched--> ^XYZ(5000)
^XYZ(5000) --Matched--> [end]
```

この結果は、^XYZ から ^XYZ(3001,4) までの範囲 (^XYZ(3001,4) は含まない) のノードが一致し、^XYZ(3001,4) から ^XYZ(5000) までの範囲 (^XYZ(5000) は含まない) に少なくとも 1 つの不一致があることを示しています。^XYZ(5000) から最後までの範囲のノードは一致しています。

不一致範囲に含まれる不一致の最小数と頻度は、最小クエリ・サイズによって異なります (“[パフォーマンスに関する考慮事項](#)”を参照)。例えば、この例で最小クエリ・サイズが既定値の 32 に設定されている場合、^XYZ(3001,4) から ^XYZ(5000) までの範囲には、32 個のノードごとに少なくとも 1 つの不一致があります。この範囲内で連続して 33 個以上のノードに不一致がなかった場合、別個の一致範囲として結果に表示されます。



## 11.1.4 DataCheck ワークフロー

チェック中にデータが変更され、一時的な不一致が記録される場合があります。このような一時的な不一致を解消するため、再チェックが必要となることがあります。宛先システムには、グローバルのチェック方法を定義するワークフローがあります。

一般的なワークフローは、フェーズ #1 の“チェック”フェーズで始まります。(フェーズ #1 は、常にチェック・サイクルの論理的な開始ポイントとして定義する必要があります。なぜなら、次のセクションで説明されるように、フェーズ #1 は、ワークフロー・タイムアウトや `DATACHECK` ルーチンの [開始] ダイアログによって、最初からの“リセット”を示すために使用されるからです。)このフェーズの最初に、最後に完了した結果として現在の結果セットが保存され、新しいアクティブな結果セットが構築されます。DataCheck は、チェックに含めるよう指定されたすべてのグローバルの最初の受け渡しを行います。

チェック・フェーズに続き、通常は必要に応じた繰り返し回数で“不一致の再チェック”フェーズが指定されます。一時的な不一致を解消するため、繰り返しのたびに、一致しないすべての範囲を再チェックします。

ワークフローの各フェーズが完了すると、DataCheck は次のフェーズに移ります。最後のフェーズが完了すると、ワークフローは暗黙的にフェーズ #1 から再度開始されます。“停止”フェーズはすべての DataCheck ジョブをシャットダウンし、手動で次のフェーズが指定されるまで“アイドル”フェーズは DataCheck を待機させます。

### 11.1.4.1 DataCheck の開始、停止、および再接続

DataCheck はいつでも停止および開始できます。DataCheck を開始すると、その停止位置からワークフローを再開します。さらに、現在のフェーズに続いたり現在のフェーズを中止する別のワークフロー・フェーズを、いつでも指定することができます。

チェック中に DataCheck の停止、切断、またはミラーリングによる一時停止が発生する場合、ルーチンにより、システムが停止した理由、停止したフェーズ、および開始時に行われる操作 (例えば、再開処理、次のフェーズへの移動、ユーザ要求によるフェーズの変更、ワークフロー・タイムアウトによるフェーズ #1 の再開など) が報告されます。開始時に、DataCheck が現行フェーズの処理を再開しようとしていたり、フェーズ #1 以外のフェーズに移行しようとする場合、以下の例に示すように、フェーズ #1 の再開というオプションが表示されます。

```
Option? 4
Configuration Name: test
State: Stopped due to Stop Requested
Current Phase: 1 - Check
Workflow Phases:
  1 - Check
  2 - RecheckDiscrepancies, Iterations=10
  3 - Stop
  (restart)
Workflow Timeout: 432000
New Phase Requested: 2
Abort Current Phase Requested

DataCheck is set to abort the current phase and transition to phase #2.

You may enter RESTART to restart at phase #1

Start Datacheck configuration 'test'? (yes/no/restart)
```

DataCheck が切断され、長時間経過した後でしか再接続しない場合は、代わりにワークフローのフェーズ #1 から再開する方が望ましい場合があります。例えば、システムがチェックの途中で数週間切断され、その後チェックが再開された場合、その結果の値は疑わしいものになります。なぜなら、一部は 2 週間前から、一部は現時点から収集されたものであるためです。このワークフローには、時間を秒単位で指定する **Timeout** プロパティが含まれ、その時間内であれば、DataCheck は部分的に完了したワークフロー・フェーズを再開できます。**timeout** を超えると、DataCheck は、次回実行状態になったときにフェーズ #1 から再開します。既定値は 5 日間 (432,000 秒) です。これは、この DataCheck 構成により大量のデータがチェックされ、正常に完了するのに数時間から数日かかる可能性があるという想定に基づいています。そのため、より短時間でチェックが完了する構成の場合、より小さな値にすることをお勧めします。ゼロを指定すると、タイムアウトは行われません。

注釈 前述のとおり、フェーズ #1 は、常にチェック・サイクルの論理的な開始ポイントとして定義する必要があります。なぜなら、前述の例で示されたように、フェーズ #1 は、ワークフロー・タイムアウトや `DATACHECK` ルーチンの [開始] ダイアログによって、最初からの “リセット” を示すために使用されるからです。

## 11.2 ミラー用の DataCheck の構成

DataCheck の宛先構成の作成時にシステムがミラーのメンバである場合 (“高可用性ガイド” の “[ミラーリング](#)” の章を参照)、ミラーリングされるデータをチェックするように DataCheck を構成できるオプションが提供されます。このオプションを選択する場合、DataCheck のソースとして機能するミラー・メンバを選択するだけで、構成の残りは自動的に設定されます。

チェックの開始時、ミラーリングされるデータベースはすべてチェックに含まれるので、データベースを個々にマップする必要はありません。また、“[チェックするグローバルおよび添え字範囲の指定](#)” の説明に従って、チェックするグローバルを指定したり、データベース全体を除外することができます。ミラーベースの DataCheck 構成を、ミラーリングされないデータベースのチェックに使用できませんが、この目的で別のミラーベースでない DataCheck 構成を作成することはできます。

このセクションでは、以下の項目について説明します。

- ・ [ミラー内での DataCheck の計画](#)
- ・ [チェックするグローバルの選択](#)

### 11.2.1 ミラー内での DataCheck の計画

DataCheck の各宛先構成は、それぞれ 1 つのソース・ミラー・メンバに接続します。ソース・メンバを変更することはできませんが、追加の DataCheck 構成を作成して、複数のソース・ミラー・メンバ (または、同じソースのさまざまなデータ・セット) に対してチェックを行うことができます。

このセクションでは、以下のメンバ固有のサブセクションについて説明します。

- ・ [フェイルオーバー・メンバ間のデータのチェック](#)
- ・ [非同期メンバのデータのチェック](#)

#### 11.2.1.1 フェイルオーバー・メンバ間のデータのチェック

フェイルオーバー・ミラー・メンバ間でチェックを行う場合、チェックは通常、以下の理由で DataCheck の宛先として構成されたバックアップ・フェイルオーバー・メンバを使用して実行されます。

- ・ DataCheck の宛先は、チェックの結果やその他の状態情報 (それ自体がジャーナリングしたもの) を保持するために、ソースよりも多くのリソースを使用します。
- ・ バックアップ・フェイルオーバー・メンバが DataCheck の宛先である場合、プライマリ・フェイルオーバー・メンバが停止しても、バックアップ上で結果を確認できます。

注釈 ほとんどの構成では、フェイルオーバーが既に発生し、結果の確認はたいていフェイルオーバーの決定後に行われるということが前提となっています。

DataCheck がソースへの接続を失ったときは必ず、ソース・マシンが再度使用可能になるのを無限に待機して、接続を再試行します。ミラーベースの DataCheck がプライマリ・フェイルオーバー・メンバでなかったときに宛先で開始され、その後そのメンバがプライマリになった場合、DataCheck は自動的に再接続を試みるのではなく、停止します。これにより、

DataCheck がプライマリで誤って実行されることがなくなります。再接続の詳細は、この章の“[DataCheck の開始、停止、および再接続](#)”を参照してください。

### 11.2.1.2 非同期メンバのデータのチェック

ミラーベースの DataCheck がフェイルオーバー・メンバと非同期メンバ間のチェックを行う場合は、非同期メンバが宛先となることが一般的です。これは、フェイルオーバー・メンバ間のチェックに関する前述の理由と同じですが（“[フェイルオーバー・メンバ間のデータのチェック](#)”を参照）、主として障害復旧時にチェックの結果を非同期メンバに格納する必要があるためです。

2つのフェイルオーバー・メンバがある場合、ソースである2つのフェイルオーバー・メンバのそれぞれに対して、非同期メンバで DataCheck の宛先構成を1つ作成すると一般的に便利です。`DATACHECK` ルーチンでは、両方を自動的に作成できるだけでなく、2つのうちどちらをプライマリ・フェイルオーバー・メンバとするかについてそれぞれの動作を設定できます。

DataCheck の各構成には、プライマリ・メンバとしてソース・フェイルオーバー・メンバの状態に基づいてその動作を制御するための設定が含まれます。設定は以下のとおりです。

- ・ **【制限なし】**  
フェイルオーバー・メンバ間で DataCheck を実行する必要がなく、両方のフェイルオーバー・メンバをチェックするエージェントとして非同期メンバを使用するため、両方を制限なしでチェックすることが適している場合です（既定）。
- ・ **【プライマリのみチェック】**（DataCheck のソースがプライマリになるまで一時停止）  
プライマリがこの非同期メンバのデータの正当なソースであるため、プライマリに対してのみチェックを行うことが適している場合です。
- ・ **【プライマリをチェックしない】**（DataCheck のソースがプライマリの場合は一時停止）  
プロダクション・プライマリ・システム上のリソースを消費しないため、バックアップに対してチェックを行うことが適している場合です。

注釈 一時停止後の再接続の詳細は、この章の“[DataCheck の開始、停止、および再接続](#)”を参照してください。

システム管理者によって手動で（必要に応じて）実行される DataCheck 構成の場合、これらの設定は特に重要でない場合があります。これらは、継続的に（または、ほぼ継続的に）実行される DataCheck 構成の場合により重要になります。

特別な関係がなくても、いずれのメンバももう1つのメンバをチェックできます。例えば、両方のフェイルオーバー・メンバをチェックするために非同期メンバを使用している場合でも、その他の非同期メンバに対するチェックのソースとして使用できます。これにより、他の非同期メンバを使用してフェイルオーバー・メンバに対するチェックを行う必要がなくなります。

## 11.2.2 チェックするグローバルの選択

DataCheck の実行時に存在するミラーリングされるデータベースは、すべて自動的にチェックされます。チェックするグローバルおよびデータベースの制御方法の詳細は、この章の“[チェックするグローバルおよび添え字範囲の指定](#)”を参照してください。

## 11.3 DataCheck のセットアップ手順

`DATACHECK` ルーチンにより DataCheck の宛先システムを設定し、管理ポータルにより DataCheck のソース・システムを有効にすることができます。新しい DataCheck システムを設定するには、以下を実行します。

1. 新しい宛先システムを作成します。

2. 宛先システムの構成を、以下のように設定または編集します。
  - a. ミラーベースでない構成の場合、ソース・システムへの TCP 接続用に、ホスト名/IP アドレス、スーパーサーバ・ポート、およびオプションの TLS 構成を指定します。  
ミラーベースの構成の場合、チェックするミラー・メンバを指定します。
  - b. ミラーベースでない構成の場合、チェックするデータベース一式とソース・システム上の対応するパスを指定します。  
ミラーベースの構成の場合、ミラーリングされるデータベースがすべて含まれます。
  - c. 必要に応じて、どのデータベース、グローバル、およびグローバル範囲を含めるか、あるいは除外するかをきめ細かく制御するため、グローバル選択マスクおよび添え字範囲を指定します。詳細は、この章の“[チェックするグローバルおよび添え字範囲の指定](#)”を参照してください。
  - d. 必要に応じて、動的に調整可能な設定を調整し、チェックのパフォーマンスおよびシステム・リソースの使用量を制御します。詳細は、この章の“[パフォーマンスに関する考慮事項](#)”を参照してください。
  - e. 必要に応じて、チェック方法を指定するワークフローを変更します。詳細は、この章の“[DataCheck ワークフロー](#)”を参照してください。
3. ソース・システムで **%Service\_DataCheck** サービスを有効にします。詳細は、この章の“[DataCheck サービスの有効化](#)”を参照してください。
4. チェックを制御する宛先システムを起動します。
5. 以下のように、チェックの状態を監視します。
  - ・ ソース・システムで、状態とログ・ファイルを表示します。
  - ・ 宛先システムで、状態とログ・ファイルのほか、クエリと結果のリストも表示します。

### 11.3.1 DataCheck サービスの有効化

ソース・システムで実行している InterSystems IRIS インスタンスの管理ポータルから、データ・チェック・サービスを有効にし、必要に応じて接続を制限します。

1. 管理ポータルの [サービス] ページ ([システム管理] > [セキュリティ] > [サービス]) に移動します。
2. サービス名のリストで [**%Service\_DataCheck**] をクリックし、データ・チェック・サービス・プロパティを編集します。
3. [サービスの有効化] チェック・ボックスにチェックを付けます。[保存] をクリックする前に、このデータベース・ソースに接続できる IP アドレスを制限することもできます。その場合は、以下の手順を実行してから [保存] をクリックします。

**注釈** ミラーをチェックするように構成した場合、DataCheck は、TLS を使用するようにミラーが設定されていれば TLS を使用します (詳細は、この章の“[ミラー用の DataCheck の構成](#)”を参照してください)。ただし、DataCheck サービスは、ミラー・メンバに対してのみアクセスを自動的に制限することはありません。他のシステムからの DataCheck 接続を制限する場合、**%Service\_DataCheck** サービスに対して [許可済みの接続元] を構成する必要があります。

4. 必要に応じて、サービスへのアクセスを制限するため、[許可済みの接続元] ボックス (以前入力したサーバ・アドレスが表示されています) で、[追加] をクリックし、[IP アドレス] を追加します。使用するすべてのアドレスを入力し終えるまで、この手順を繰り返します。

これらのアドレスは、該当する行の [削除] をクリックして個別に削除できます。また、[すべて削除] をクリックしてすべてのアドレスを削除し、どのアドレスからも接続を許可することもできます。

## 11.3.2 チェックするグローバルおよび添え字範囲の指定

DataCheck では、以下に詳細を示すオプションを使用して、チェックに含めたりチェックから除外するグローバル名および添え字範囲を指定できます。

注釈 グローバルおよび添え字範囲を指定する際、リテラル値のみがグローバル名および添え字として受け入れられます。

- ・ **[[マップされた/ミラーリングされる] 全データベースの全グローバルをチェックする]** – ミラーベースでない構成では、すべてのマップされたデータベースの全グローバルをチェックし、ミラーベースの構成では、すべてのミラーリングされるデータベースの全グローバルをチェックします。
- ・ **[一部のグローバル/データベースを含む/除外する]** – 指定したマスクに基づいて選択されたデータベースのグローバルをチェックします。添え字は許可されません。

マスク、またはコンマ区切りのマスクのリストを以下のように追加または編集できます。

- \* – すべてのグローバルをチェックします (既定)。
- 最後の文字に \* – 前にある文字 (列) で始まるすべてのグローバルをチェックします。
- マスクの前の ' – グローバルをチェックから除外します。

以下はその例です。

- ABC\* – ABC で始まるすべてのグローバル名
- A:D – A から D のすべてのグローバル名
- A:D,Y\* – A から D、および Y で始まるすべてのグローバル名
- \*, 'C\*', 'D\*' – C または D で始まるグローバルを除くすべてのグローバル
- ' \* – すべてのグローバルを除外します。

特定のデータベースのグローバル選択マスクを定義するだけでなく、グローバル選択マスクが定義されていないデータベースに対して使用される“既定のグローバル選択マスク”を明示的に設定することもできます。当初、既定のマスクは \* に設定されます。

注釈 ミラーベースの DataCheck の場合、新しく追加されたミラーリングされるデータベースが次のチェックに含まれます。そのため、新しく追加されたミラーリングされるデータベースが自動的にチェックされないようにする場合は、既定のマスクを ' \* に設定します。

例えば、マスクが定義されていないすべてのデータベースに対する既定のマスク (\*, '^DontCheckMe) を指定すると同時に、特に **USER** および **USER2** データベースに対してグローバル選択マスク (A:D) を指定するには、**^DATACHECK** ルーチンの **[構成の編集]** サブメニューから以下を実行します (この章の “[^DATACHECK ルーチン](#)” を参照してください)。

```
1) Import Settings from a Shadow
2) Connection Settings
3) Database Mappings
4) Globals to Check
5) Performance Settings
6) Manage Workflow
```

Option? 4

```
1) Check All Globals in All Mapped Databases
2) Include/Exclude Some Globals/Databases
3) Include/Exclude Some Globals/Databases and Subscript Ranges
```

Option? 1 => 2

Save changes? Yes =>

```
1) Options for selecting globals to check
2) Set default include/exclude mask for databases with no mask defined
3) Add or remove include/exclude mask for databases
4) View include/exclude masks
```

Option? 2

Enter a mask string, \* to include all, ^\* to exclude all, ? for help

Mask: \* => \*, ^DontCheckMe

Save changes? Yes =>

```
1) Options for selecting globals to check
2) Set default include/exclude mask for databases with no mask defined
3) Add or remove include/exclude mask for databases
4) View include/exclude masks
```

Option? 3

```
1) C:\InterSystems\IRIS\mgr\docbook\ [no mask defined, use default]
2) C:\InterSystems\IRIS\mgr\user\ [no mask defined, use default]
3) C:\InterSystems\IRIS\mgr\user2\ [no mask defined, use default]
```

Database (multiple selections allowed): 2,3

Enter a mask string, \* to include all, ^\* to exclude all, ? for help

! to delete this mask and revert to default

Mask: A:D

```
1) C:\InterSystems\IRIS\mgr\docbook\ [no mask defined, use default]
2) C:\InterSystems\IRIS\mgr\user\ [A:D]
3) C:\InterSystems\IRIS\mgr\user2\ [A:D]
```

Database (multiple selections allowed):

Save changes? Yes =>

```
1) Options for selecting globals to check
2) Set default include/exclude mask for databases with no mask defined
3) Add or remove include/exclude mask for databases
4) View include/exclude masks
```

Option?

・ [一部のグローバル/データベースおよび添え字範囲を含む/除外する]—[一部のグローバル/データベースを含む/除外する] オプションと同じタスクを実行できるだけでなく、このオプションでは、特定のグローバルの添え字範囲を識別することができます。含めるようにマークされたグローバル添え字範囲は、そのグローバルがグローバル選択マスクに含まれるかどうかに関係なく含められます。DataCheck 内のすべての添え字範囲は、範囲の最初を含みますが、範囲の最後を含みません。

注釈 DataCheck では、処理の過程で除外範囲のデータに一致が検出された場合、そのデータが一致としてマークされる可能性があります。一方、除外された範囲の不一致がマークされることはありません。

例えば、上記の例の続きで、**USER2** データベースに対し、グローバル選択マスク (**A:D**) を指定したとします。以下のようにプロンプトに応答して、**NAME** グローバルに添え字範囲を含めることができます。

```
1) Options for selecting globals to check
2) Set default include/exclude mask for databases with no mask defined
3) Add or remove include/exclude mask for databases
4) View include/exclude masks
```

Option? 1

```
1) Check All Globals in All Mapped Databases
2) Include/Exclude Some Globals/Databases
3) Include/Exclude Some Globals/Databases and Subscript Ranges
```

Option? 2 => 3

Save changes? Yes =>

```
1) Options for selecting globals to check
2) Set default include/exclude mask for databases with no mask defined
3) Add or remove include/exclude mask for databases
4) View include/exclude masks
5) Add/Edit Subscript Ranges for a Global
6) Delete All Subscript Ranges for a Global
7) Delete All Subscript Ranges
8) View Defined Subscript Ranges
```

Option? 5



```

1) C:\InterSystems\IRIS\mgr\docbook\
2) C:\InterSystems\IRIS\mgr\user\
3) C:\InterSystems\IRIS\mgr\user2\

Database: 3 C:\InterSystems\IRIS\mgr\user2\
Global Name: ^NAME
There are no subscript ranges defined for this global.
You may start by including all or excluding all subscripts.
Answer YES to include, NO to exclude: no

C:\InterSystems\IRIS\mgr\user2\      ^NAME
    ^NAME --Excluded--> [end]

From (inclusive):  ?

    Enter a global reference with or without subscripts or null for end.
    The leading ^ may be omitted.  For subscripted references the entire
    global name may be omitted and simply begin with open parentheses

From (inclusive):  (10)
To (exclusive):    (20)
Answer YES to include, NO to exclude: yes

C:\InterSystems\IRIS\mgr\user2\      ^NAME
    ^NAME --Excluded--> ^NAME(10)
    ^NAME(10) --Included--> ^NAME(20)
    ^NAME(20) --Excluded--> [end]

From (inclusive):
Continue editing subscript ranges for this global? Yes => no

C:\InterSystems\IRIS\mgr\user2\      ^NAME
    ^NAME --Excluded--> ^NAME(10)
    ^NAME(10) --Included--> ^NAME(20)
    ^NAME(20) --Excluded--> [end]

Save changes? Yes =>

1) Options for selecting globals to check
2) Set default include/exclude mask for databases with no mask defined
3) Add or remove include/exclude mask for databases
4) View include/exclude masks
5) Add/Edit Subscript Ranges for a Global
6) Delete All Subscript Ranges for a Global
7) Delete All Subscript Ranges
8) View Defined Subscript Ranges

Option?

```

マスク情報は、以下のように表示できます。

```

Option? 4
The default include/exclude mask is:
    *,'^DontCheckMe

The following databases are using non-default global selection criteria

C:\InterSystems\IRIS\mgr\user\
A:D
C:\InterSystems\IRIS\mgr\user2\
    * Has additional global subscript ranges to include/exclude that apply
    regardless of whether those globals are included in this mask.
A:D

1) Options for selecting globals to check
2) Set default include/exclude mask for databases with no mask defined
3) Add or remove include/exclude mask for databases
4) View include/exclude masks
5) Add/Edit Subscript Ranges for a Global
6) Delete All Subscript Ranges for a Global
7) Delete All Subscript Ranges
8) View Defined Subscript Ranges

Option?

```

マスク情報には添え字範囲の記述が含まれているため、その情報は以下のように表示できます。

```

Option? 8
Device:
Right margin: 80 =>

DataCheck Destination System: GLOBTEST

```



```
Global Selection Subscript Ranges

C:\InterSystems\IRIS\mgr\user2\      ^NAME
    ^NAME --Excluded--> ^NAME(10)
    ^NAME(10) --Included--> ^NAME(20)
    ^NAME(20) --Excluded--> [end]

1) Options for selecting globals to check
2) Set default include/exclude mask for databases with no mask defined
3) Add or remove include/exclude mask for databases
4) View include/exclude masks
5) Add/Edit Subscript Ranges for a Global
6) Delete All Subscript Ranges for a Global
7) Delete All Subscript Ranges
8) View Defined Subscript Ranges

Option?
```

## 11.4 ^DATACHECK ルーチン

^DATACHECK ルーチン (%SYS ネームスペース内) を使用して、データ・チェックを構成および管理できます。任意のプロンプトでヘルプを表示するには、? を入力します。

^DATACHECK ルーチンを開始するには、以下を実行します。

1. ターミナルで、以下のコマンドを入力します。

```
set $namespace = "%SYS"
%SYS>do ^DATACHECK
```

2. メイン・メニューが表示されます。選択番号を入力します。ルーチンを終了するには **Enter** キーを押します。

```
1) Create New Configuration
2) Edit Configuration
3) View Details
4) Start
5) Stop
6) Delete Configuration
7) Incoming Connections to this System as a DataCheck Source

Option?
```

注釈 オプション 2 ～ 6 では、複数の宛先システムを作成した場合、アクションを実行する宛先システムを選択できるようにリストが表示されます。

メイン・メニューでは、以下のテーブルのように実行する DataCheck タスクを選択できます。

オプション	説明
1)	<a href="#">[新規構成の作成]</a> プロンプトによって、新しい DataCheck 宛先システム構成の名前が要求されます。
2)	<a href="#">[構成の編集]</a> サブメニューが表示されます。
3)	<a href="#">[詳細表示]</a> サブメニューが表示されます。
4)	宛先システムを起動または再開します。再開する場合は、停止位置から再開されます。
5)	宛先システムを停止します。停止した後に宛先システムを再開する場合、宛先システムは停止した位置から再開されます。

オプション	説明
6)	指定した宛先システム構成を削除します。
7) DataCheck	<p>[DataCheck のソースであるこのシステムへの着信接続] サブメニューが表示されます。</p> <p>注釈 ソース・システムでは、このオプションを選択する必要があります。</p>

## 11.4.1 新規構成の作成

このサブメニューでは、宛先システムを構成できます。このオプションを選択すると、以下のプロンプトが表示されます。

Configuration Name:

ミラー・メンバでないシステムで DataCheck の構成を作成する場合、[設定の編集] サブメニューが表示されます。“[ミラーベースでないシステムでの DataCheck の構成の編集](#)”の説明に従って、手動で構成を完了してください。

ミラー・メンバであるシステムで DataCheck の構成を作成する場合、データ・チェックをミラーリングに基づくかどうかによって追加情報を求められます。ミラーリングに基づかない DataCheck を構成することを選択した場合は、[設定の編集] サブメニューが表示されます。このサブメニューを使用して、“[ミラーベースでないシステムでの DataCheck の構成の編集](#)”の説明に従って、手動で構成を完了してください。一方、ミラーリングに基づく DataCheck を構成することを選択した場合、ミラーリングされるデータベースにデータ・チェックが制限され、宛先システムがフェイルオーバー・メンバか非同期ミラー・メンバかによって後続のプロンプトが変わります。詳細は、この章の“[ミラー用の DataCheck の構成](#)”を参照してください。

## 11.4.2 構成の編集

このサブメニューでは、宛先システムの構成を変更できます。サブメニューのオプションは、ミラーベースの構成を編集するか、ミラーベースでない構成を編集するかによって異なります。詳細は、以下のサブセクションを参照してください。

- ・ [ミラーベースでないシステムでの DataCheck の構成の編集](#)
- ・ [ミラーベースの DataCheck の構成の編集](#)

### 11.4.2.1 ミラーベースでないシステムでの DataCheck の構成の編集

ミラーベースでないシステムでは、このオプションを選択すると、以下のプロンプトが表示されます。

Configuration Name: dc\_test

```

1) Import Settings from a Shadow (static)
2) Connection Settings          (static)
3) Database Mappings            (static)
4) Globals to Check             (dynamic)
5) Performance Settings         (dynamic)
6) Manage Workflow              (dynamic)

```

Option?

注釈 編集モードでは、複数の宛先システムを作成した場合、編集する宛先システムを選択できるようにリストが表示されます。また、オプション 1 ～ 3 の設定を編集するには、まずシステムを停止する必要があります。

選択番号を入力します。前のメニューに戻るには ^ キーを押します。このサブメニューの各オプションでは、以下のテーブルのように宛先システムを構成できます。

オプション	説明
1)	非推奨です。使用しないでください。
2)	ソース・システムに接続するための情報。
3)	ソース・システムおよび宛先システムに対して、データベース・マッピングを追加、削除、またはリストできます。
4)	チェックするグローバルまたはチェックから除外するグローバル。詳細は、この章の“ <a href="#">チェックするグローバルおよび添え字範囲の指定</a> ”を参照してください。
5)	使用するシステム・リソース（スロットル）および DataCheck が不一致を隔離する粒度（最小クエリ・サイズ）を調整します。詳細は、この章の“ <a href="#">パフォーマンスに関する考慮事項</a> ”を参照してください。
6)	ワークフロー・フェーズの順序を管理します。詳細は、この章の“ <a href="#">DataCheck ワークフロー</a> ”を参照してください。

### 11.4.2.2 ミラーベースの DataCheck の構成の編集

ミラーベースのシステムでは、以下のサブメニューが表示されます。

Configuration Name: MIRRORSYS2\_MIRRORX201112A\_1

- 1) Globals to Check
- 2) Performance Settings
- 3) Manage Workflow
- 4) Change Mirror Settings (Advanced)

Option?

選択番号を入力します。前のメニューに戻るには ^ キーを押します。このサブメニューの各オプションでは、以下のテーブルのように宛先システムを構成できます。

オプション	説明
1)	チェックするグローバルまたはチェックから除外するグローバル。詳細は、この章の“ <a href="#">チェックするグローバルおよび添え字範囲の指定</a> ”を参照してください。
2)	使用するシステム・リソース（スロットル）および DataCheck が不一致を隔離する粒度（最小クエリ・サイズ）を調整します。詳細は、この章の“ <a href="#">パフォーマンスに関する考慮事項</a> ”を参照してください。
3)	ワークフロー・フェーズの順序を管理します。詳細は、この章の“ <a href="#">DataCheck ワークフロー</a> ”を参照してください。
4) ( )	この章の、“ミラーリングに関する考慮事項”セクションの“ <a href="#">ミラー内での DataCheck の計画</a> ”を参照してください。

## 11.4.3 詳細表示

このサブメニューでは、宛先システムの状態の監視および、実行しているクエリやデータ・チェック結果の詳細情報の表示を行うことができます。

System Name: dc\_test

- 1) View Status
- 2) View Results
- 3) View Queries
- 3) View Log

Option?

選択番号を入力します。前のメニューに戻るには ^ キーを押します。このサブメニューの各オプションでは、以下のテーブルのように宛先システムの情報を表示できます。

オプション	説明
1)	選択した宛先システムに関する情報を表示します。この情報には、DataCheck ワーク・ジョブのパフォーマンス・メトリック、ソースと状態、現在のフェーズ、ワークフロー・タイムアウト、要求された新規フェーズ、現行フェーズで完了したクエリの割合、このフェーズで記録された不一致の数などが含まれます。
2)	選択した宛先システムの結果を表示します。詳細は、この章の “ <a href="#">DataCheck 結果</a> ” を参照してください。
3)	選択されている宛先システムによって送信されたクエリに関する情報を表示します (“ <a href="#">DataCheck クエリ</a> ” を参照してください)。これには、まだ処理されていないグローバル (添え字の包含/除外範囲が使用されている場合はグローバル範囲) が含まれ、DataCheck により現在処理されているアクティブ・クエリが示されます。このリストの末尾には合計数が表示されます。
4)	選択した宛先システムのログ・ファイルが表示されます。

注釈 ^DATACHECK が 2 つのミラー・メンバ・インスタンスでミラーリング・データベースの 2 つのコピーに対して実行され、そのデータベースでグローバル全体の設定と削除がすぐに行われる場合、[ ] オプションを [ ] オプションと比較すると、表示される結果がわかりにくくなる可能性があります。例えば、一致しない回答が状態に含まれていることが報告されるが、その回答の原因となったグローバルが結果では実際に報告されないことがあります (それ以降の試行で不一致が解決したため)。さらに、表示される回答の数が、インスタンス内の実際のグローバルの数 (管理ポータルで表示されるものと結果で実際に報告されるもの) より多くなる可能性があります。

[受け取った回答] にはゼロ以外の一致しない値があるが不一致にはゼロの値があることを [View Status] で示す場合、これは一時グローバルであり、データの問題ではないことを示します。

## 11.4.4 DataCheck のソースであるこのシステムへの着信接続

このサブメニューでは、ソース・システムに関する情報を表示できます。

- 1) List Source Systems
- 2) View Log

Option?

選択番号を入力します。前のメニューに戻るには ^ キーを押します。このサブメニューの各オプションでは、以下のテーブルのようにソース・システムの情報を表示できます。

オプション	説明
1)	DataCheck のソース・システムに関する情報を表示します。
4)	ソース・システムのログ・ファイルが表示されます。

## 11.5 データ・チェックに関する特別な考慮事項

DataCheck を使用するときは、以下の特別な考慮事項を確認してください。

- ・ パフォーマンスに関する考慮事項
- ・ セキュリティに関する考慮事項

### 11.5.1 パフォーマンスに関する考慮事項

複数のシステムでのデータベースの整合性を確保するうえでデータ・チェックは有用ですが、ソース・システムと宛先システムの両方でリソースが消費されます。これにより、負荷および DataCheck の構成によっては、いずれかのシステムで他のプロセスのパフォーマンスに悪影響を及ぼす可能性があります。DataCheck には、パフォーマンスの管理を支援するコントロールが含まれます。

スロットルは、DataCheck が使用できるシステム・リソース (CPU、ディスク I/O、データベース・キャッシュ) の量を制御する 1 から 10 までの整数です。スロットル値は、いつでも変更してすぐに反映させることができます。例えば、システムの負荷が軽いと思われる時間には値を大きくし、システムの負荷が重い時間には小さくすることができます。これは、実行時間が長くなると予想されるチェックに有用です。DataCheck ルーチンは、負荷の重い時間に停止する可能性もありますが、再起動すると、チェックが停止された時点から自動的に再開されます。

各システムの特徴は異なりますが、スロットル値の以下の一般的な記述が当てはまります。

- ・ スロットル値を 1 に設定した場合、DataCheck クエリの実行について 1 つのプロセスを超えるリソースは使用されません。つまり、CPU を最大 1 つのみを使用して、一度に 1 つのディスク I/O のみを実行します。使用されるリソースが主に CPU なのか主にディスク I/O なのかは、データが既にバッファにあるかどうかによります。これは、チェックの進捗に伴って変わる場合があります。
- ・ スロットル値を 8 に上げると、より多くのシステム・リソースが各ステップで消費されます。大量のリソース (多くの CPU など) が存在するシステムの場合、8 のスロットル設定時、DataCheck が CPU の数や他の要素を考慮しながら、システム・リソースの大部分を使用するように、リソースの消費量がほぼ同じ倍数増加するよう各間隔が調整されます。ただし、8 のスロットル設定時でも、システムは負荷の軽いアプリケーション・アクティビティに応答することが想定されるため、高負荷時以外の時間帯において、通常のシステムでは 6、7、または 8 の設定が適切な場合があります。
- ・ 9 のスロットル設定は 8 とあまり変わりませんが、DataCheck ジョブによるバッファ・プール全体の使用を可能にします (バッチ・フラグの設定を解除します)。
- ・ 10 のスロットル設定では、ほぼすべてのシステム・リソースを使用してチェックを完了するよう試行されます。

^DATACHECK の [ ] サブメニューの [ ] オプションにより、DataCheck ワーク・ジョブのパフォーマンス・メトリックが表示されます。これにより、パフォーマンスの特性、およびそれらの特性がスロットル設定とどのように関連するのかを理解できます。

スロットルの実装は、ソフトウェア特性およびハードウェア特性の進化とともに変わる場合があります。

最小クエリ・サイズは、クエリの検索で許可されるグローバル・ノードの最小数を表します。つまり、DataCheck が不一致を隔離する際に使用するグローバル・ノードの範囲の最小サイズを決定します。値を小さくすると不一致を見つけやすくなり、値を大きくすると不一致部分のチェック速度が大幅に向上します。例えば、最小クエリ・サイズを 1 (非推奨) に設定すると、個々の不一致ノードは、別個の不一致範囲 (または少なくともすべての一致しないグローバルを含む範囲) として報告され、不一致が正確に識別できますが、パフォーマンスに大きな影響が出ます。最小クエリ・サイズを 1000 (やはり非推奨) に設定すると、1 つまたは複数の不一致は、少なくとも 1000 個の一致しないノードを含む範囲として報告され、一致しないノードを見つけにくくなりますが、チェックは大幅に高速化します。既定値は 32 です。これは十分に小さいため、パフォーマンスに大きな影響を及ぼすことなく、管理ポータルを使って範囲内のグローバル・ノードを比較的簡単に目視検査できます (“グローバルの使用法” の “[グローバルの管理](#)” の章を参照)。

## 11.5.2 セキュリティに関する考慮事項

宛先システムでは、チェックが完了したグローバルおよびチェック中のグローバルの添え字範囲を保存します (結果およびクエリ)。(この章の “[チェックするグローバルおよび添え字範囲の指定](#)” を参照してください。)この添え字データは、%SYS ネームスペース (既定では IRISSYS データベース内) の ^SYS.DataCheck\* グローバルに保存されます。グローバルの値は保存されず、添え字のみが保存されます。%SYS ネームスペースに保存されるその他のデータベースのグローバル添え字には、セキュリティ構成によっては、それ以外の場合に一部のユーザには表示されない機密情報が含まれることがあります。したがって、保護された配置では、特に注意が必要です。

構成、起動、および停止の機能を含め、^DATACHECK ルーチンを使用するには、^SYS.DataCheck\* グローバルを含むデータベース (既定では IRISSYS) に、%Admin\_Operate:Use 特権と Read/Write 特権 (チェックの構成には Write、その他のすべてのタスクには Read) の両方が必要です。^SYS.DataCheck\* グローバルに保存される構成および結果データは、十分なデータベース特権があればだれでも、ルーチンの外部から表示および操作することができます。

DataCheck データへのアクセス権を持つべきではないユーザに %DB\_IRISSYS:Read 特権が与えられるセキュアな配置では、%SYS ネームスペースにグローバル・マッピングを追加して、^SYS.DataCheck\* グローバルを IRISSYS 以外の別のデータベースにマップできます。このデータベースには、新しいリソース名を割り当てることができます。これで、DataCheck の使用を許可されたロールにこのリソースへの読み取り許可を制限できます。

別の宛先システムがソースとしてこのシステムに接続できるかどうかは、このシステムの %Service\_DataCheck サービスにより制御されます。このサービスは、新しいインストールでは既定で無効になり、許可された IP アドレスのリストで構成することができます。詳細は、この章の “[DataCheck サービスの有効化](#)” を参照してください。

2 つのシステム間の通信の暗号化に関しては、TLS を使用してソースに接続するよう宛先システムを構成することができます。詳細は、“[TLS を使用するための InterSystems IRIS スーパーサーバの構成](#)” を参照してください。

