



# インターシステムズの用語集

Version 2024.1  
2024-06-03

## インターシステムズの用語集

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 用語集 .....	1
1.1 先頭文字が記号または A の用語 .....	1
1.2 先頭文字が B の用語 .....	3
1.3 先頭文字が C の用語 .....	4
1.4 先頭文字が D の用語 .....	9
1.5 先頭文字が E の用語 .....	11
1.6 先頭文字が F の用語 .....	13
1.7 先頭文字が G の用語 .....	14
1.8 先頭文字が H の用語 .....	14
1.9 先頭文字が I の用語 .....	15
1.10 先頭文字が J の用語 .....	18
1.11 先頭文字が K の用語 .....	18
1.12 先頭文字が L の用語 .....	19
1.13 先頭文字が M の用語 .....	20
1.14 先頭文字が N の用語 .....	22
1.15 先頭文字が O の用語 .....	22
1.16 先頭文字が P の用語 .....	24
1.17 先頭文字が Q の用語 .....	27
1.18 先頭文字が R の用語 .....	28
1.19 先頭文字が S の用語 .....	30
1.20 先頭文字が T の用語 .....	34
1.21 先頭文字が U の用語 .....	35
1.22 先頭文字が V の用語 .....	36
1.23 先頭文字が W の用語 .....	36
1.24 先頭文字が X の用語 .....	37



# 1

## 用語集

以下に、InterSystems IRIS® データ・プラットフォームに関連する用語とその説明を示します。

### 1.1 先頭文字が記号または A の用語

#### **\$X/\$Y action table (\$X/\$Y アクション・テーブル)**

システム。デバイスへの入力、またはデバイスからの出力の際に、文字が \$X か \$Y のいずれか、または両方を変更するか否かを指定する内部テーブルです。\$X 特殊変数と \$Y 特殊変数は、デバイスでの水平位置と垂直位置の情報を追跡します。

#### **\$ZF function (\$ZF 関数)**

ObjectScript。インターシステムズ社の実装固有の関数で、InterSystems IRIS 内から外部プログラムやルーチン呼び出すことができます。UNIX® では、\$ZF 関数を使用して、UNIX® シェル・コマンドの実行、UNIX® システム・サービスの呼び出し、C 言語などの高水準言語で記述されたルーチンの呼び出しを行うことができます。

#### **abstract class (抽象クラス)**

オブジェクト。抽象クラスはインスタンス化することができません。これは、共通の特性を持つ、複数の非抽象サブクラスのテンプレートとして動作します。

#### **abstract persistent class (抽象永続クラス)**

オブジェクト。抽象永続クラスは、インスタンス化することはできませんが、ここから派生するすべてのサブクラスに格納されたデータを含む単一テーブルとして、InterSystems SQL に投影されます。

#### **access mode (アクセス・モード)**

システム。使用可能な接続ツールのカテゴリで、各アクセス・モードには特定の状況で適用される特性があります。アクセス・モードには、次の 3 種類があります。ローカル (ユーザがサーバ・マシン上で InterSystems IRIS と直接対話する)、クライアント・サーバ (通常は別のマシンにある他の実行可能プログラムから、ユーザが InterSystems IRIS に接続する)、および Web (Web ゲートウェイを使用する Web アプリケーションからユーザが InterSystems IRIS と対話する) です。

#### **activate (アクティベート)**

システム。データベース暗号化キーをメモリから取得してその値を内部的に実行中の InterSystems IRIS インスタンスに格納し、InterSystems IRIS で暗号化データベースをマウントおよび操作できるようにすること。

**added role (追加ロール)**

システム。ユーザが InterSystems IRIS に認証を実行した後に、そのユーザに関連付けられる任意のロール。

**ad-hoc query (アドホック・クエリ)**

一般。埋め込み SQL で、または ODBC または JDBC 経由で使用される SELECT 文です。オブジェクト・アプリケーション内で、アドホック・クエリはクラス定義の一部ではなく、クエリのインタフェースを使用して処理することができないという点が、通常のクエリと異なります。

**application mode (アプリケーション・モード)**

システム。InterSystems IRIS がターミナル・セッションを開始するモード。指定されたネームスペースで、指定のプログラムが実行されます。アプリケーション・モードでは、ユーザは InterSystems IRIS プロンプトからプログラムを開始できません。ターミナル・セッションは、ユーザがプログラムをシャットダウンするときに終了します。アプリケーション・モードは、ユーザに一部の限定された操作のみを許可し、[プログラマ・モード](#)で提供されるターミナルのより強力な機能にアクセスできないように設計されています。

**array (配列)**

オブジェクト。データにアクセスするキー値の組み合わせの集合。配列は、SQL に子テーブルとして投影されます。

**ASCII character set (ASCII 文字セット)**

システム。インターシステムズ社の製品では、ASCII は、制約の多い 7 ビット文字セットではなく拡張された 8 ビット文字セットを表します。既定では、InterSystems IRIS インスタンスは、ドキュメント“インターシステムズのサポート対象プラットフォーム”の“サポート対象言語”の章で指定されているように、マシンのロケールの文字セットを使用します。マシンの既定の言語がサポートされていない場合、InterSystems IRIS は、一般に Latin-1 として知られている ISO-8859-1 文字セットを使用します。

**asset (資産)**

システム。InterSystems IRIS のセキュリティによって保護されるデータまたは機能。資産は InterSystems IRIS 内ではリソースとして表されます。各リソースは 1 つまたは複数の資産を示します。

**atomic lock (アトミック・ロック)**

オブジェクト。アトミック・ロックは、データベースのシングル・ノードに保存されているデータをロックしません。1 つ以上のノードに保存されたデータがロードされている間は、共有ロックを適用し、同じデータが保存される間は、排他ロックを適用します。

**attribute (属性)**

オブジェクト。クラスに関連する固有の特性を表すデータの要素。そのクラスの各オブジェクトには、属性のリテラル値が含まれます。また、クラスの属性は、その状態も表します。プロパティとも言われます。厳密に言うと、属性はリレーションシップでないプロパティです。

**audit database (監査データベース)**

システム。InterSystems IRIS の実行中に発生したイベントの一部がリストされたデータベース。監査を有効にすると、InterSystems IRIS 管理者の選択に応じて、InterSystems IRIS でさまざまなイベントが記録されます。アプリケーションでも、アプリケーションのイベントを作成および記録できます。監査データベースは監査ログとも呼ばれます。

## 認証

システム。ユーザが、本人の身分について証明するために使用するプロセス。最も一般的なケースでは、パスワードを提供して実行します。ただし、市販の他の認証スキームもあります。

### authentication mechanism (認証メカニズム)

システム。ユーザが InterSystems IRIS に認証される方法。認証は、既存のオペレーティング・システムの証明書、InterSystems IRIS に最初から備わっているパスワードのプロンプト、LDAP、または代行 (ユーザ定義) コードを使用し、Kerberos を使用して行われます。

### authorization (承認)

システム。認証されたユーザが、InterSystems IRIS システムに対し実行可能な内容を決定するプロセス。承認には、(ロール割り当てメカニズムによって実施される) ユーザへのロールの割り当て、それらのロールとそれらのロールが実施できることの管理、InterSystems IRIS 内のリソースの管理といったさまざまな機能が含まれます。

## 1.2 先頭文字が B の用語

### binary stream (バイナリ・ストリーム)

オブジェクト。バイナリ・ストリームは、イメージなどの大量のバイナリ・データの塊の操作や保存に使用するインタフェースを提供します。バイナリ・ストリームに保存されるデータは、Unicode 変換の際に変換されません。InterSystems IRIS ストリーム・インタフェースは、ObjectScript、SQL、Java でバイナリ・ストリームを操作するのに使用します。

### bind (結合)

InterSystems SQL。アプリケーション・プログラム内の変数を、埋め込み SQL を使用して、ホスト変数と関連付けること。

### BLOB

InterSystems SQL。Binary Large Object の略。大量のデータの塊を意味するリレーショナル用語。InterSystems IRIS は、BLOB の操作にストリーム・インタフェースを使用します。

### breakpoint (ブレイクポイント)

ObjectScript。ZBREAK コマンドで指定した InterSystems IRIS ルーチン内の位置。実行が指定された行に達すると、InterSystems IRIS はルーチンを中断します。オプションとして、定義したデバッグを実行する場合もあります。

### buffer (バッファ)

一般。特定の情報を持つ、コンピュータ・メモリの定義された位置。バッファにより、複数のプロセスが同じデータにアクセスすることが容易になります。例えば、InterSystems IRIS はグローバル・バッファにグローバルを、ルーチン・バッファにルーチンをキャッシュします。

## 1.3 先頭文字が C の用語

### CLASSPATH

Java。Java コンパイラに使用されるシステム変数。コンパイルするクラスで参照している、クラスやパッケージを検索するために使用されます。

### calculated property (計算プロパティ)

オブジェクト。計算プロパティを含むオブジェクトがインスタンス化される際に、メモリ内ストレージが割り当てられないプロパティ。その代わり、その値は要求されるたびに決定されます。

### call method (call メソッド)

オブジェクト。InterSystems IRIS ルーチンを直接呼び出すメソッド。

### callin interface (コールイン・インタフェース)

システム。C プログラム内から、ObjectScript コマンドや式の実行や、評価を行うことができる InterSystems IRIS の機能。\$ZF ルーチンからもコールイン・インタフェースを使用できます。

### callback method (コールバック・メソッド)

オブジェクト。コールバック・メソッドは、システム・メソッドによって呼び出され、これにより特定のイベント中にユーザが追記した処理を行うことができます。これらのメソッドのタイプを区別するため、%OnEvent 形式の名前を付けます。Event は、コールバックをトリガするイベントを示しています。

### canonical form (キャノニック形式)

一般。名前のキャノニック形式とは、不足している要素がなく、標準に従った、曖昧さのない形式です。InterSystems IRIS では、番号、グローバル参照、コマンド・オプション、ファイルおよびパス、デバイスなどのさまざまなエンティティのキャノニック形式があります。

番号のキャノニック形式の詳細は、“ObjectScript リファレンス”の“[\\$NUMBER](#)”のリファレンス・ページを参照してください。

グローバルおよび関連するネイキッド・グローバル参照のキャノニック形式の詳細は、“[ネイキッド・グローバル参照](#)”を参照するか、“ObjectScript リファレンス”の“[\\$NAME](#)”または“[\\$ZREFERENCE](#)”のリファレンス・ページを参照してください。

コマンド・オプションのキャノニック形式の詳細は、各コマンドのリファレンス・ページを参照してください。

ファイル、パス、およびデバイスのキャノニック形式は、オペレーティング・システムによって異なります。

例えば、Windows の場合、..[%Iris.wj](#) というファイル参照はキャノニック形式ではありませんが、[C:%InterSystems%MyIRIS%IRIS.WIJ](#) はキャノニック形式です (MyIRIS という名前の InterSystems IRIS インスタンスがあり、標準の場所にインストールされている場合)。

UNIX® では、キャノニック形式のデバイス名は例えば /dev/tty のようになります。

### cascading dot syntax (カスケード・ドット構文)

オブジェクト。カスケード・ドット構文を使用して、オブジェクトに参照された別のオブジェクトのつながりを追跡し、参照されたオブジェクトのプロパティやメソッドを直接操作することができます。例えば以下ようになります。

#### ObjectScript

```
Write auto.Engine.Cylinder.SparkPlug.IdNumber
```



## 文字セット

システム。インスタンスの InterSystems IRIS データの内部表現を定義する値のセット。

## character stream (文字ストリーム)

オブジェクト。文字ストリームは、大量のテキスト・データの塊の操作や保存に使用するインタフェースを提供します。文字ストリームに保存されているデータは、Unicode 変換中に変換されます。InterSystems IRIS ストリーム・インタフェースは、ObjectScript、SQL、Java で文字ストリームを操作するのに使用します。

## child table (子テーブル)

InterSystems SQL。別のテーブル (親テーブル) に依存しているテーブル。子テーブルの行は、親テーブルの行に対するポインタを持つ必要があります。

## circular reference (循環参照)

オブジェクト。異なるクラスに 2 つのプロパティがあり、それぞれのプロパティが他方のクラスの参照である状態を表します。InterSystems IRIS は、両方のプロパティが必須プロパティでない限り、循環参照を完全にサポートします。循環参照の両方のプロパティが必須である場合、両方のクラスのオブジェクトを保存することができません。

## class compiler (クラス・コンパイラ)

オブジェクト。クラス・コンパイラは、InterSystems IRIS クラスをコンパイルします。これには、IDE からクラス・コンパイラを使用するか、または InterSystems IRIS オブジェクト・ユーティリティ・ライブラリのコンパイル呼び出しのうちの 1 つを実行します。

## Class Definition Language (クラス定義言語)

オブジェクト。InterSystems IRIS のクラス定義に使用するキーワード・ベースの言語。

## class descriptor (クラス記述子)

オブジェクト。実行時にオブジェクトを使用するために必要な情報 (メソッドやプロパティのリスト) を含む、特別なタイプのルーチン。

## class dictionary (クラス・ディクショナリ)

オブジェクト。クラス・ディクショナリは、コンパイル済みのすべてのクラスの、現在のクラス定義を保持します。InterSystems SQL が有効なテーブル定義を決定する際にも、クラス・ディクショナリを使用します。

## class hierarchy (クラス階層)

オブジェクト。クラス階層は、各ルート・クラスからサブクラスに分かれ、さらにサブクラスに分かれるという階層構造を表します。

## class member (クラス・メンバ)

オブジェクト。クラスに属する要素。クラス・メンバは、プロパティ、メソッド、パラメータ、クエリ、インデックス、トリガ、XData ブロックです。

## class method (クラス・メソッド)

オブジェクト。クラスのインスタンスがメモリ内にあるか否かにかかわらず、呼び出すことのできるメソッド。

**class**

オブジェクト。クラスは、単一エンティティの状態と動作をカプセル化します。クラスは、タイプ情報のほか、プロパティ、メソッド、パラメータ、クエリ、インデックスなどのクラス・メンバから構成されます。

InterSystems IRIS は、2 種類の主なクラス、データ型クラスとオブジェクト・クラスをサポートします。また、抽象クラス、抽象永続クラス、永続クラス、埋め込み可能なクラス、非登録クラス、登録クラス、システム・クラスなど、さまざまなオブジェクト・クラスをサポートします。

**client data type (クライアント・データ型)**

オブジェクト。クライアント・データ型は、InterSystems IRIS オブジェクト・サーバ経由でクライアントにデータを投影するために必要な情報の指定に使用されます。すべてのデータ型クラスは、クライアント・データ型を持つ必要があります。データ型クラスに基づく属性は、そのクラスのクライアント・データ型を属性のデータ型として、SQL や、Java などのクライアントに投影します。

**client lock (クライアント・ロック)**

システム。"outgoing lock (外向きロック)" を参照してください。

**クライアント**

システム。ユーザのアプリケーションが動作するマシン。これが、ご使用のデータベースが属するサーバ・マシンに接続されます。また、他のノードからの情報を要求できるネットワーク上のノードも、クライアントです。

**client application (クライアント・アプリケーション)**

システム。InterSystems IRIS 実行可能ファイルとは別の実行可能ファイル。ただし、データ・ソースとしてなどで InterSystems IRIS に依存します。このようなアプリケーションは InterSystems IRIS に登録でき、その結果として InterSystems のセキュリティ・モデルの一部として制限されます。クライアント・アプリケーションのユーザは、サポートされる認証メカニズムのいずれかを使用して、InterSystems IRIS で認証される必要があります。

**code method (コード・メソッド)**

オブジェクト。ObjectScript を実行するメソッド。

**collation rule (照合規則)**

システム。InterSystems IRIS がグローバル・ノードを検索する順番を決定するアルゴリズム。InterSystems IRIS 組み込みの照合規則には、UNICODE、ANSI、文字列が含まれます。ANSI 照合では、キャノニックの数値添え字が最初に数字の順番に並び、次にそのほか文字列順に並びます。文字列順は、文字セットの文字の順番です。文字列規則は、ObjectScript の "]" 演算子と同様に、すべての添え字を文字セットの文字列のように順番付けします。

**collation sequence (照合順)**

システム。InterSystems IRIS がグローバル・ノードを順番に検索する必要がある場合に、その順番を指定します。照合順は、文字セットと照合規則から決まります。各グローバルは、特定の照合順と対応付けられています。

**collation (照合)**

InterSystems SQL。照合は、データを保存する前に InterSystems SQL がデータを変換する方法を指定します。各データ型は、既定の照合タイプを持ちます。データ型に基づくプロパティは、プロパティの照合タイプが明示的に定義されていない限り、その照合タイプを使用します。インデックスは、インデックスの照合タイプが明示的に指定されていない限り、それらがインデックス付けしているプロパティの照合タイプを使用します。

InterSystems SQL は、優先照合タイプ EXACT、SQLUPPER、および SQLSTRING をサポートしています。SQLUPPER は、%String に基づくデータ型の既定で、EXACT はその他すべてのデータ型の既定です。

**collection (コレクション)**

オブジェクト。複数の要素 (リテラル値またはオブジェクト) を含むプロパティ。InterSystems IRIS は、リストと配列という 2 つのタイプのコレクションをサポートします。

**column**

InterSystems SQL。列は、テーブルに保存された各インスタンスに対する、データの特定の要素を含みます。プロパティは、列として SQL に投影されます。

**communications protocol (コミュニケーション・プロトコル)**

一般。ネットワーク上のコンピュータ間でデータを送信する方法を定義する一連の規約。厳密に言うと、メッセージ・パケットのフォーマット方法を指定するソフトウェアです。コミュニケーション・プロトコルを使用して、ネットワーク・インタフェース・デバイスと通信します。1 つ以上のコミュニケーション・プロトコルが、同じ物理インタフェース・デバイスを共有することができます。

**compilation flag (コンパイル・フラグ)**

オブジェクト。コンパイル・フラグは、クラス・コンパイラにクラスのコンパイル方法を指示します。コンパイル・フラグは、コンパイルするクラス、ソース・コードをシステムから削除するか否か、および InterSystems IRIS がメモリ内オブジェクトを持つクラスをコンパイルする方法を制御します。

**compilation (コンパイル)**

オブジェクト。InterSystems IRIS クラスとルーチンを実行可能なコードに変換するプロセス。

**compiler directive (コンパイラ指示文)**

オブジェクト。コンパイラ指示文は、特定のクラスまたはルーチンのコンパイルを成功させるために必要なファイルの情報を提供します。コンパイラ指示文は、インクルード・ファイルを指定したり、1 つのクラスのコンパイルが終了した後にクラスをコンパイルするように指定します。

**computed field (計算フィールド)**

オブジェクト。ObjectScript のコンパイル済みコードに定義された計算によって取得した値を持つフィールドです。ObjectScript のコンパイル済みコードは、関連するベース・テーブルの別のフィールドや、ObjectScript 関数、あるいは特殊変数を参照できます。

**computer name (コンピュータ名)**

システム。1 つのプロセッサと、その周辺機器、およびそれに関連するデータセットに割り当てられた名前。この名前は、ネットワークの中で一意である必要があります。"System Name (システム名)" を参照してください。

**concurrency mode (並行処理モード)**

オブジェクト。並行処理モードは、オブジェクトをオープンするとき、または保存するときに実行するロックのタイプを決定します。InterSystems IRIS は、5 種類の並行処理モード (0 - 4) をサポートし、それぞれロックなし、アトミック・ロック、共有ロック、共有保持ロック、排他ロックを指定することができます。

**並行処理**

オブジェクト。並行処理は、ロックを利用してデータの整合性を管理する機能を提供します。InterSystems IRIS は、ロックを実行しないモードから、オブジェクトがメモリにロードされてから保存するまで排他的にロックを保持するモードまで、5 種類の並行処理モードをサポートします。既定では、InterSystems IRIS はすべてのオブジェクトに対し、アトミック・ロックを使用します。

**concurrent backup (並行バックアップ)**

システム。バックアップは、他のデータベース活動を停止しないで実行します。InterSystems IRIS は、バックアップの整合性を保ちながら、ユーザに与えるバックアップの影響を最小限に抑えた、マルチパス・メソッドを使用しています。

メモ：ユーザがバックアップ作業の一環として、現在のジャーナル・ファイルを消去、削除、または上書きした場合、バックアップの間に発生したトランザクション処理は、完全にはジャーナルされない場合もあります。

**configuration (構成)**

システム。InterSystems IRIS 構成は、起動時の InterSystems IRIS リソースを示します。構成は管理ポータルで定義します。1 つ以上の構成を作成することはできますが、一度に使用できるのは 1 つだけです。InterSystems IRIS は、起動時に現在の構成を使用します。

**connection (接続)**

システム。InterSystems IRIS とクライアント・アプリケーションまたはツールとの間の接続のこと。それぞれの接続は、InterSystems IRIS サーバ上に関連するプロセスを持ちます。

**connection security level**

システム。クライアント・サーバ接続を保護する Kerberos 機能を指定します。次の機能レベルを使用できます。最初の認証のみ (単に Kerberos と呼ばれる)、最初の認証およびパケットの整合性 (Kerberos パケット整合性と呼ばれる)、あるいは最初の認証パケットの整合性、およびすべてのメッセージの暗号化 (Kerberos 暗号化と呼ばれる) です。

**connection tool (接続ツール)**

システム。ユーザが InterSystems IRIS との接続を確立するための手段 (ターミナル、Java など)。

**conversion code (変換コード)**

オブジェクト。InterSystems IRIS がデータ値を内部保存形式から外部の表示形式に変換する際、および外部の入力形式から内部の保存形式に変換する際に、InterSystems IRIS が使用するコンパイル済みコード。

**CPF**

システム。構成パラメータ・ファイルの省略形。CPF は、拡張子 **.cpf** の付いたファイルで、構成を定義します。“configuration (構成)” を参照してください。

**credentials cache (証明書 cache)**

システム。以前に認証されたユーザの認証情報が記述されたファイル。証明書 cache は、同じ処理を繰り返し実行する必要をなくし、認証プロセスを高速化するために使用されます。Kerberos 認証への参照に最も頻繁に使用されます。

**current device (現在のデバイス)**

システム。このデバイスを通して入出力コマンドが処理されます。ユーザがログオンすれば、ユーザの現在のデバイスが主デバイスになります。通常これは、ユーザがログオンしている端末、またはパーソナル・コンピュータです。

**current directory (現在のディレクトリ)**

一般。現在使用中のディレクトリ。

### cursor-based SQL (カーソル・ベース SQL)

InterSystems SQL。クエリを処理するためにカーソルをオープンする、[埋め込み SQL](#) クエリのタイプ。ご使用のアプリケーションが複数行のデータにアクセスする場合、カーソルを使用する必要があります。カーソルは、ポインタの動作と同様に、一度に 1 行のアクセスと処理に集中し、続いて次の行に移ります。

### cursor

InterSystems SQL。複数行のデータ内で、前方に動く反復子。

### custom storage (カスタム・ストレージ)

オブジェクト。カスタム・ストレージにより、ストレージ・インタフェース・メソッドのユーザ独自の実装を記述して、オブジェクトのストレージ構成を指定できます。通常、カスタム・ストレージを使用するクラスは、SQL に投影されません。

## 1.4 先頭文字が D の用語

### database (データベース)

一般。IRIS.DAT ファイル。コードおよびデータを含めることができます。どのデータベースも、グローバル、パッケージ、およびルーチン・マッピングを介して、複数のネームスペースから使用できます。

### データベース・キャッシュ

システム。データベースから取得されるデータをキャッシュするために使用されるシステム・メモリ (RAM)。これにより、同じクエリの繰り返しのインスタンスは、ストレージではなくメモリから結果を取得できるので、パフォーマンス上のメリットが非常に大きくなります。パフォーマンスを最大化するために、データベース・キャッシュをアプリケーション・ワークロードの[作業セット](#)以上の大きさにする必要があります。

### database encryption (データベース暗号化)

システム。InterSystems IRIS データベースを暗号化された状態でディスクに保存するためのプロセス。InterSystems IRIS がデータをディスクから読み取ると、データは実行時に自動的に解読されて、許可されたユーザが利用できるようになります。ディスクに書き込まれるときに、データは暗号化されます。ディスク上のデータは、保存データとも呼ばれます。

### database-encryption key (データベース暗号化キー)

システム。InterSystems IRIS データベースの暗号化に使用される AES キー。

### database integrity (データベースの整合性)

一般。データベースのコンテンツや構成が破損していない状態。データベース内のデータは、その物理的な整合性の低下により、読み取りができなくなることがあります。内部整合性の低下により、内部のポインタが破損することもあります。

### Data Definition Language (データ定義言語)

InterSystems SQL。Data Definition Language (データ定義言語) または DDL は、テーブルの生成、構成の定義、削除に使用するコマンド・ベースの言語です。

### data location (データ位置)

システム。グローバルが属するデータセット。複製グローバルに対するグローバルな SET や KILL のソースは、そのグローバルのデータの場所にマップされたディレクトリ (または、ディレクトリとシステム) です。

**dataset (データセット)**

一般。InterSystems IRIS 内のディレクトリ名、あるいはディレクトリとシステム名を表す論理エンティティ。

**Data Source Name (DSN) (データ・ソース名)**

ODBC。データ・ソース名は、特定のサーバ・システム上の特定のデータベースを識別します。

**data type (データ型)**

オブジェクト。属性のデータ型は、その動作や検証の必要条件、SQL や、Java などのクライアントに投影される方法などを指定します。

**data type class (データ型クラス)**

オブジェクト。データ型は、DATATYPE クラス・キーワード・セットを持つクラスで、データ型インタフェースをサポートします。このインタフェースには、データの妥当性の検証と SQL の相互運用のために設計された、多数の処理が含まれます。

**DDL import utility (DDL インポート・ユーティリティ)**

オブジェクト。InterSystems IRIS リレーショナル・ユーティリティ・ライブラリに含まれる、InterSystems IRIS に DDL ファイルをインポートするためのユーティリティ。このユーティリティは、対応する定義を自動的にクラス・ディクショナリに追加します。

**deep save (ディープ・セーブ)**

オブジェクト。ディープ・セーブは、“%Persistent.%Save” メソッドのクラス・リファレンスのコンテンツで説明されているように、オブジェクト、およびそれらのオブジェクトが参照するすべてのオブジェクトを保存します。

**default dataset (既定のデータセット)**

システム。1 つのネームスペースで、コマンドを実行したり独自にマップされていないグローバルを受け取るディレクトリ。

**default start routine (既定開始ルーチン)**

システム。既定開始ルーチンは、ユーザがログインしたときに自動的に実行されます。

**dependent relationship (依存リレーションシップ)**

オブジェクト。親テーブルの各行に子テーブルを定義する親子リレーションシップ。

**device accounts (デバイス・アカウント)**

システム。デバイス・アカウントは、TELNET ユーザがユーザのシステムの特定のルーチンあるいはネームスペースに、リモートで、またはローカルでログインすることを許可します。管理ポータルを使用して、ユーザ・アカウントやデバイス・アカウントを設定し、特定のルーチン、データベース、ネームスペースに対するユーザのアクセス管理を容易にできます。

**directory (ディレクトリ)**

一般。ファイルを保存できる、ディスク上の位置の名前。InterSystems IRIS データベース・ファイルである IRIS.DAT は、ディレクトリに属します。InterSystems IRIS データベース・ファイルは、1 つのディレクトリに 1 つしか置くことができません。

**dismount、dismounted (ディスマウントする、ディスマウントされている)**

システム。データベースをディスマウントするとは、データベースを InterSystems IRIS のインスタンスから接続解除することです。ディスマウントされているデータベースを使用するには、再度明示的にインスタンスにマウントする必要があります。

**display format (表示形式)**

オブジェクト。プロパティの表示形式は、データの表示や入力に使用する形式です。

**distributed database (分散データベース)**

システム。ネットワーク内で、複数のコンピュータに保存されているデータベース。InterSystems IRIS を分散データベース環境で使用する場合、あるコンピュータに存在する ObjectScript ルーチンは、ネットワーク内の他のコンピュータにあるグローバルに、それらのシステムによって定められたファイル保護制限の範囲内で、アクセスすることができます。

**DMNNET**

システム。ネットワークから入力されるグローバル要求を処理する InterSystems IRIS プロセス。その名前は、システム・ステータス (%SS) 表示、あるいはネットワーク・デーモンを表すシステム・オペレーション・ユーティリティのプロセス・パネルに表示されます。

**dot syntax (ドット構文)**

オブジェクト。ドット構文を使用して、プロパティ値を取得して設定したり、メソッドを実行することができます。さらに、参照オブジェクトや埋め込みオブジェクトのプロパティやメソッドに、参照しているオブジェクトからアクセスできます。

## 1.5 先頭文字が E の用語

**embeddable class (埋め込み可能クラス)**

オブジェクト。埋め込み可能な(シリアル)クラスから得たオブジェクトは、メモリ内では独立して存在できますが、データベースに保存されると、永続オブジェクトに保存されたデータとしてしか存在できません。`%SerialObject` を参照してください。

**embedded HTML (埋め込み HTML)**

ObjectScript。ObjectScript メソッド、あるいはマクロ・ルーチンに直接埋め込まれた HTML。埋め込み HTML は、`&html<>` 文に入れる必要があります。

**embedded object (埋め込みオブジェクト)**

オブジェクト。埋め込み可能なクラスのインスタンス。埋め込みオブジェクトは、メモリ内では個別のオブジェクトですが、永続オブジェクト内では埋め込みデータとしてのみ保存することができます。

**embedded SQL (埋め込み SQL)**

InterSystems SQL。[埋め込み SQL](#) は、ObjectScript メソッド、あるいはマクロ・ルーチンに直接埋め込まれた SQL です。埋め込み SQL は、`&sql()` 文に入れる必要があります。

**empty string (空の文字列)**

システム。論理的に文字が含まれない文字列で、テキストの中では通常 `"` で表現されます。NULL 文字列と呼ばれることもあります。

言語によって空の文字列の表現方法はそれぞれ異なり、それに応じて、コンピュータのメモリ領域を実際使用するかどうか、および使用する場合の方法が決まります。例えば、ObjectScript では " " はメモリ領域を使用しませんが、SQL ではメモリ領域を使用します。また、文字を含まない文字列であっても文字列は文字列であり、コンテキストによっては、NULL (割り当てのない) 値とは異なる方法で処理される可能性があります。

例えば、以下の ObjectScript コードを考えます。

#### ObjectScript

```
New A
New B
Set A = ""
Write A
Write B
```

最初の Write 文は成功し (特に何も表示されません)、2 番目は <UNDEFINED> エラーを返します。

#### encapsulation (カプセル化)

オブジェクト。カプセル化は、パブリック・インタフェースを公開することにより、クラスの内部の詳細を隠します。パブリック・インタフェースは、処理の実行に使用した詳細を一切表示しないで、クラスの許可されたすべての相互作用の概要を説明します。

#### endian

システム。システムがビッグ・エンディアンである、またはリトル・エンディアンであるということは、システムがマルチバイト要素のバイトを格納する順序を指しています。InterSystems IRIS は、ビッグ・エンディアンとリトル・エンディアンの両方をサポートします。また、データベースを一方のエンディアン形式からもう一方のエンディアン形式に変換する cvendian ユーティリティも用意しています。cvendian の詳細は、["cvendian 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"](#) の該当セクションを参照してください。

#### エンタープライズ・キャッシュ・プロトコル

システム。InterSystems IRIS の内部ネットワーキング・システム。ECP ネットワーキング機能により、設定可能な数のノード、およびさまざまなハードウェア構成やソフトウェア構成を持つ分散型データベース環境で、InterSystems IRIS を使用することができます。

#### enumerated attribute (列挙属性)

オブジェクト。列挙属性により、プロパティの値に対し複数の選択肢を与えることができます。これらの属性の値は、前もって指定された選択肢の 1 つである必要があります。

#### exclusive lock (排他ロック)

一般。排他ロックは、他のプロセスが特定のデータを見たり編集するのを防ぎます。

オブジェクトをオープンする際に排他ロックを使用する場合、オブジェクトがオープンしたときに排他ロックを取得し、オブジェクトがクローズしたときに開放します。

#### explicit reference (明示参照)

システム。拡張参照とも呼ばれます。グローバルが属するシステムとネームスペースのすべての定義を持つグローバル参照です。

#### expression method (式メソッド)

オブジェクト。クラス・コンパイラにより生成されるコードの、インラインに置くオブジェクト・メソッド。



**extended global reference (拡張グローバル参照)**

システム。明示参照とも呼ばれます。グローバルの位置を識別するグローバル参照。その位置は、定義されたネームスペース、あるいは暗黙ネームスペースとして指定できます。InterSystems IRIS アプリケーションが、グローバルをマップしている現在のネームスペースをオーバーライドする必要がある場合、この拡張グローバル参照を使用して希望と異なるディレクトリ、あるいは異なるディレクトリとシステムを指定します。例えば、ADMIN というネームスペースで実行中のアプリケーションから、INVEN ネームスペースのグローバル ^PARTS を参照する必要がある場合、拡張参照 ["INVEN"]PARTS を使用します。一方、現在有効なネームスペースにマップする代わりに、特定の物理位置を指定する必要がある場合には、暗黙ネームスペースを使用します。UNIX® コンピュータ上で、“production” というディレクトリ・セット名を持つコンピュータの “inven” ディレクトリにあるグローバル ^PARTS を参照するには、拡張参照 ["production"/usr/inven"]PARTS を作成します。

**extent index (エクステント・インデックス)**

オブジェクト。エクステント・インデックスは、エクステント内のすべてのオブジェクトのインデックスを保持します。

**extent query (エクステント・クエリ)**

オブジェクト。エクステント内の各オブジェクトの ID を返すクエリ。

**extent**

オブジェクト。エクステントは、ルート・クラスと呼ばれる特定のクラスのツリー構造で、全階層の橋渡しをします。InterSystems SQL テーブルは、それに対応するクラスのすべてのエクステントを含みます。

## 1.6 先頭文字が F の用語

**factory class (ファクトリ・クラス)**

オブジェクト。Java のファクトリ・クラスからインスタンス化されたオブジェクトは、InterSystems IRIS との接続を管理します。また、別の InterSystems IRIS クラスのインスタンスを生成してオープンし、その他の管理機能も提供します。

**field**

オブジェクト。SQL 列の別名。

**file stream (ファイル・ストリーム)**

オブジェクト。ファイル・ストリームは、外部ファイル内の大きなテキスト・データやバイナリ・データの塊の操作や保存に使用するインタフェースを提供します。InterSystems IRIS ストリーム・インタフェースは、ObjectScript、SQL、Java でファイル・ストリームを操作するのに使用します。

**final class (最終クラス)**

オブジェクト。拡張したりサブクラスを定義することができないクラス。

**final method (最終メソッド)**

オブジェクト。オーバーライドできないメソッド。

**final property (最終プロパティ)**

オブジェクト。オーバーライドできないプロパティ。

**foreign key (外部キー)**

InterSystems SQL。外部キーは、テーブルの列が他のテーブルの列を指し示すように制約を加えます。最初の列に与えられた値は、2 番目の列にも存在する必要があります。

## 1.7 先頭文字が G の用語

**global (グローバル)**

システム。多次元ストレージ構造。[グローバル](#)は、バランスト・ツリー・テクノロジーを使用して InterSystems IRIS データベースに実装します。

**global database (グローバル・データベース)**

システム。基礎となる、InterSystems IRIS の論理および物理データ・ストレージ構造。すべてのデータは、この中の "グローバル" と呼ばれる複数の添え字を持つ配列のシステムに保存されます。

**global directory (グローバル・ディレクトリ)**

システム。グローバル・データベースを含むディレクトリ。グローバル・ディレクトリには、ディレクトリ内のデータベース・ファイルやすべてのグローバルのリスト、および関連情報が含まれます。

**globally unique identifier (GUID) (グローバル一意識別子)**

システム。GUID は、複数のシステムにわたる場合も含め、InterSystems IRIS のすべてのインスタンスでクラスのインスタンスなどのあらゆるエンティティを一意に特定できる識別子です。例えば、2 つの独立した InterSystems IRIS インスタンスで、それぞれのインスタンスの GUID を指定した共通のクラス定義を使用すると、それら 2 つのインスタンスからデータを同時に取得しても、GUID 値が重複することはありません。InterSystems IRIS は、GUID を[オブジェクトの同期化](#)の一部として使用しています。GUID の使用に関する一般的な情報については、`%ExtentMgr.GUID` および `%Library.GlobalIdentifier` のクラス・ドキュメントを参照してください。

**GSA file (GSA ファイル)**

システム。グローバル保存ファイル。GSA ファイル拡張子は必須ではありませんが、この拡張子により InterSystems IRIS やプログラマが、保存されたグローバルを容易に識別できます。ファイル拡張子以外には、GSA ファイルに重要な意味はありません。

## 1.8 先頭文字が H の用語

**host name (ホスト名)**

一般。サーバ・システム名。

**host variable (ホスト変数)**

InterSystems SQL。[埋め込み SQL](#) 文内で、アプリケーション・プログラムの変数にリンクされている変数。

## 1.9 先頭文字が I の用語

### identified by

オブジェクト。クラスは、その存在を別のクラスに論理的に依存しているとき、別のクラスにより識別されます。

### identifying relationship (識別リレーションシップ)

オブジェクト。識別リレーションシップは、1 つのクラスがもう一方のクラスにその存在を依存している、2 つのクラス間のリレーションシップを定義します。

### ID

オブジェクト。オブジェクトの ID は、そのエクステンション内で一意です。

### idkey

オブジェクト。オブジェクトの ID のコンテンツを示すために使用するインデックス。idkey で使用されたプロパティは、オブジェクトが継続している間は、静的な状態を維持する必要があります。

### implicit global reference (暗黙グローバル参照)

システム。"Mapped Global Reference (マップされたグローバル参照)" を参照してください。

### implied namespace (暗黙ネームスペース)

システム。拡張グローバル参照で、ユーザがディレクトリ名、あるいはディレクトリ名とシステム名を使用したときに、InterSystems IRIS が内部的に生成するネームスペース。

### include file (インクルード・ファイル)

ObjectScript。ObjectScript のソース・コードのコンパイル時に、プリプロセッサ段階で使用される定義を含むファイル。マクロ・ソース・ルーチンを拡張し、任意のコード行をインクルードするか否かを指定します。インクルード・ファイルは、複数のルーチンで共通のコードをインクルードする場合にも使用できます。これにより、共通のサブルーチンへの呼び出しのオーバーヘッドを削減します。

### incoming lock (内向きロック)

システム。ローカル・コンピュータ上のアイテムに対して、リモート・クライアント・コンピュータのプロセスにより発行されたロック。ロックされたアイテムはローカル・コンピュータに属し、ローカル・コンピュータはサーバのように動作するため、サーバ・ロックとも呼ばれます。このロックはロック・テーブル・ディスプレイに表示されます。その際、LOCK 要求を発行したリモート・コンピュータのシステム名が、所有者列に表示されます。ローカル・サーバ・コンピュータは、リモート・クライアント・コンピュータのどのプロセスが LOCK を発行したかに関知せず、アイテムに発行されたロックの数も追跡しません。

### index

オブジェクト。インデックスは、クラスに属する各オブジェクトのデータの、ソートされたサブセットを保存することにより、データ検索を最適化します。

### index collation (インデックス照合)

オブジェクト。インデックス照合は、インデックスにデータを保存するときに使用するデータ変換を指定します。

**inheritance (継承)**

オブジェクト。クラスの特性やメンバを、そのクラスのすべてのサブクラスに受け継ぐこと。複数のクラスに共通する特性を、1 つのスーパークラスにまとめることができます。

**in-memory value (メモリ内の値)**

オブジェクト。プロパティがメモリ内にある間のプロパティの値。プロパティのタイプによっては、ディスク上の値や保存される値と異なる場合があります。

**install-dir**

システム。一般に InterSystems IRIS インストール・ディレクトリを参照する場合、ドキュメントでは install-dir という用語を使用します。ドキュメントの例では、**C:\MyIRIS** が使用されています。[“既定のインストール・ディレクトリ”](#)のセクションでは、サポートされているすべてのオペレーティング・システムについて InterSystems IRIS のインストール先を説明しています。

**instance**

オブジェクト。特定のエンティティを表すクラスの実装。インスタンスとオブジェクトという用語は、同じ意味で使用される場合があります。

**インスタンス認証**

システム。InterSystems に最初から備わっている認証システム。ユーザはパスワードの入力を要求され、入力されたパスワードのハッシュが InterSystems IRIS サーバに渡され、そのハッシュがサーバに保存されている既存のパスワードのハッシュと比較されます。この 2 つの値が同じであれば、ユーザは特権を持つリソースに対して承認されます。

このメカニズムは、管理ポータルに **[パスワード]** 認証として表示されます。

**instance method (インスタンス・メソッド)**

オブジェクト。クラスの特定のインスタンスから呼び出されるメソッド。そのインスタンスに連動したアクションを実行します。

**instantiate (インスタンス化)**

オブジェクト。オブジェクト・インスタンスを、プログラムが動作できるメモリ内に置くこと。

**intermediate source code (中間ソース・コード)**

ObjectScript。標準 3GL (第 3 世代言語) ObjectScript ソース・コードは、InterSystems IRIS で使用できます。中間コードは、InterSystems IRIS コンパイラがマクロ・ソースから作成します。このレベルでは、埋め込み SQL を含むすべてのプリプロセッサ構文が解析され、ルーチンは純正のソース・コードのみを含みます。この段階で ObjectScript ルーチンを記述することは可能ですが、埋め込み SQL やマクロなど他のプリプロセッサ構文を利用することはできません。

**InterSystems IRIS ランチャー**

システム。Windows システム上で、InterSystems IRIS を起動したときにタスクバーに表示されるアイコン。このアイコンから、[InterSystems IRIS システムの構成や管理](#)、クラスやルーチンの生成や管理を実行します。

**InterSystems IRIS データベース**

システム。1 つのディレクトリ、ネームスペース、あるいは UIC 内のグローバルとルーチンに保存されている、関連のあるデータの集合。

## InterSystems IRIS Object Server for Java (Java 用 InterSystems IRIS オブジェクト・サーバ)

オブジェクト。InterSystems IRIS Java バインディングにより、Java で記述されたクライアント・アプリケーションは、サーバ・ベースの InterSystems IRIS オブジェクトにアクセスすることができます。

## InterSystems IRIS object utility library (InterSystems IRIS オブジェクト・ユーティリティ・ライブラリ)

オブジェクト。InterSystems IRIS オブジェクト・ユーティリティ・ライブラリが提供するインタフェースにより、InterSystems IRIS のオブジェクト・コンポーネントを構成したり、クラスを操作、コンパイルしたり、インタラクティブにオブジェクトを使用することができます。これらのユーティリティに対するプライマリ・インタフェースは、`%SYSTEM.OBJ` クラスを経由します。

## InterSystems IRIS relational utility library (InterSystems IRIS リレーショナル・ユーティリティ・ライブラリ)

InterSystems SQL。InterSystems IRIS リレーショナル・ユーティリティ・ライブラリは、InterSystems SQL を構成し、InterSystems SQL サーバを管理し、他のリレーショナル・データベースから DDL をインポートするインタフェースを提供します。これらのユーティリティに対するプライマリ・インタフェースは、`%SYSTEM.SQL` クラスを経由します。

## InterSystems IRIS サーバ

システム。ネットワーク・システム内の分散した InterSystems IRIS データベースを使用するための機能。

## InterSystems IRIS ストレージ

オブジェクト。永続オブジェクトの既定のストレージ・タイプ。InterSystems IRIS ストレージを使用すると、ユーザがデータの保存方法の詳細を指定する必要がなく、クラスは自動的に SQL に投影されます。

## InterSystems SQL

InterSystems SQL。InterSystems IRIS への高性能リレーショナル・インタフェース。InterSystems SQL は、InterSystems IRIS のオブジェクト機能と完全に統合されており、テーブル定義のクラス・ディクショナリを共有し、クラス定義としてアドバンスド・データ型を使用します。

## InterSystems SQL サーバ

InterSystems SQL。InterSystems SQL サーバは、クライアント・アプリケーションからの SQL 要求を処理して、クエリへの適用、InterSystems IRIS データベースの更新、エラーのログ作成、その他の SQL 関連のタスクを実行します。これは、ODBC や JDBC ドライバが接続するサーバ・プロセスです。

## InterSystems SQL ストレージ

オブジェクト。InterSystems SQL ストレージは、SQL を使用してオブジェクト永続インタフェース経由でデータを挿入、更新、検索するメカニズムを提供します。クラスのストレージ定義で、`%Storage.SQL` ストレージ・クラスを指定し、SQL マッピング・セクションを含む場合に、クラスは InterSystems SQL ストレージを使用します。

## I/O translation (入出力変換)

システム。[各国言語サポート](#)の機能。コンピュータの文字セットと特定のデバイスの文字セット間で、変換を実行する一連のテーブルで構成されます。“言語構成”も参照してください。

## IP アドレス

InterSystems IRIS システム。インターネット・プロトコル (IP) に基づくコンピュータの数値識別子。InterSystems IRIS は、IPv6 および IPv4 の両方の形式をサポートします。InterSystems IRIS での IPv6 のサポートの詳細は、“[IPv6 アドレスの使用](#)”を参照してください。

## IRIS.DAT

システム。InterSystems IRIS データベースのプライマリ・ボリューム。InterSystems IRIS のグローバルとルーチンが格納されています。

## ITG file (ITG ファイル)

システム。データベース整合レポートを持つ ASCII ファイル。容易に識別できるように、拡張子 .ITG を付けます。このファイルは、データベース整合チェックにより作成されます。

# 1.10 先頭文字が J の用語

## Java

一般。元々 Sun Microsystems 社によって開発されたオブジェクト指向言語。

## Java Database Connectivity (JDBC)

一般。JDBC は、リレーショナル・データ・アクセスに対する標準 Java インタフェースを提供します。

## JOBbed process (ジョブ化されたプロセス)

システム。InterSystems IRIS プロンプトで、あるいはアプリケーションから ObjectScript の JOB コマンドを実行することにより、生成されたバックグラウンド・プロセス。リレーショナル・サーバ・システム、サーバ・マスタ、およびそれらが生成するサーバ・プロセスは、ジョブ化されたプロセスの一例です。

## join (結合)

InterSystems SQL。InterSystems IRIS リレーショナル・データベースのベース・テーブルから (クエリ、フォーム、レポートにより) 1 つ以上のテーブルから成るデータを取得するリクエスト。これにはテーブル間にリンクが必要です。

## journaling (ジャーナリング)

システム。システム管理者による選択に基づき、すべてのグローバルまたは指定したグローバルに対する変更のログを InterSystems IRIS でジャーナル・ファイルに記録する機能。システム障害が発生した場合、これらの変更をロール・フォワードすることができます。つまり、修復中に、すべてのトランザクションをデータベースに再度適用します。“Write Image Journaling (ライト・イメージ・ジャーナリング)” も参照してください。

# 1.11 先頭文字が K の用語

## KDC

システム。鍵配布センター (Key Distribution Center)。Kerberos インストールの一部として、すべてのパーティの正確な認証を保証する中央の Kerberos サーバです。KDC は、信頼されたサードパーティ Kerberos サーバの一部であり、TGT (チケット保証チケット) およびサービス・チケットの基盤となるキーを生成します。Windows では、KDC は Windows のドメイン・コントローラ (DC) の一部で、ドメイン・コントローラと呼ばれることもあります。これら 2 つの名称の頭文字が似ているのは偶然です。



**Kerberos**

システム。MIT の Athena プロジェクトで開発された、信頼されるサードパーティの認証システム。このシステムでは、認証情報が含まれるデータベースを作成して、ユーザまたはアプリケーション（これらをまとめてプリンシパルと呼びます）を認証します。このデータベースはセキュリティで保護されており（そのため信頼されます）、認証を実行する 2 つの任意のパーティから分離されています（そのためサードパーティシステムと呼ばれます）。Kerberos は、インターネットのような必ずしもセキュリティが保証されないネットワーク上で使用するために設計されています。1980 年代後半から、大規模な民間機関および教育機関で使用されてきました。

**key (unique index) (キー (一意のインデックス))**

オブジェクト。一意のインデックスの別名。

**key (encryption) (キー (暗号化))**

システム。通常は、データを暗号化または解読する暗号化アルゴリズムと共に使用されます。

**key-encryption key (キー暗号化キー)**

システム。InterSystems IRIS データベース暗号化では、プロセスの 2 番目のキーです。最初のキーはデータベースの暗号化に使用され、2 番目のキーであるキー暗号化キーは、最初のキーを暗号化して保護するために使用されます。データベース暗号化キーがアクティブ化されるとき、キー暗号化キーによって解読され、メモリにロードされ使用可能となります。

**keyword (class definition) (キーワード (クラス定義))**

オブジェクト。キーワードは、クラス定義内で個別の特性を定義します。クラス・キーワードとも言われます。

**keyword (system element) (キーワード (システム要素))**

システム。InterSystems IRIS システムでは、関数名や演算子などをキーワードと呼ぶこともあります。

## 1.12 先頭文字が L の用語

**language configuration (言語構成)**

システム。文字セット、照合順、\$X/\$Y アクション、パターン照合という 4 つのテーブルのセット。各国言語サポートのデバイスの独立性を定義します。デバイスに依存した各国言語サポートの特性である、入出力変換と対を成すものです。

**license (ライセンス)**

システム。インターシステムズ社とユーザの間の契約。これにより、ユーザが使用することのできる InterSystems IRIS ソフトウェアのコンポーネントや、各コンポーネントを使用できるユーザの数が指定されます。InterSystems IRIS を使用するには、ライセンスを取得する必要があります。ライセンス情報は、プロダクト・アクティベーション・キーの中に含まれており、ご使用のシステムの **iris.key** という名前のファイルに保存されています。

**list**

オブジェクト。データへのアクセスにスロット番号を使用する、順序付けされた集合。各リストは、1 つのリスト・フィールドとして SQL に投影されます。

## ロケール

システム。ユーザの言語、国、およびその他の特殊な可変設定を指定するパラメータ。ロケールは、数や日付の表記、曜日や月の名前など、入力、出力、およびデータ処理におけるユーザへの表示に関する規約を指定します。

## lock table (ロック・テーブル)

システム。プロセスによって発行されたすべての LOCK コマンドが保存される InterSystems IRIS の内部テーブル。このテーブルを見るには、システム・ビューワを使用します。

## ログ・ファイル

システム。[システム管理者用ディレクトリ](#)に存在し、システム操作、エラー、およびメトリックに関するメッセージを含むファイル。これらのファイルには、メッセージ・ログ ([messages.log](#))、システム・モニタ・ログ ([SystemMonitor.log](#))、アラート・ログ ([alerts.log](#))、初期化ログ ([iboot.log](#))、およびジャーナル履歴ログ ([journal.log](#)) があります。これらのログ・ファイルの詳細は、“監視ガイド”の“[ログ・ファイルの監視](#)”を参照してください。

## logical format (論理形式)

オブジェクト。オブジェクト・プロパティの論理形式は、メモリ内で使用される形式です。すべての比較や計算の実行に、この形式を使用します。

## login role (ログイン・ロール)

システム。InterSystems IRIS への認証の操作によって（その後ではなく）、ユーザに関連付けられる任意のロール。

# 1.13 先頭文字が M の用語

## macro preprocessor (マクロ・プリプロセッサ)

ObjectScript。マクロ・コードを、使用可能な ObjectScript コードに変換する ObjectScript コンパイラの一部。

## macro source code (マクロ・ソース・コード)

ObjectScript。ルーチンやメソッドを記述できる、高レベルで柔軟性、許容性の高いコード。マクロ・ソース・コードは、ObjectScript 構文、特殊マクロ・プリプロセッサ・コマンド、および ANSI 標準 SQL の組み合わせを使用して、マクロと埋め込み SQL 文の定義を許可します。

## map (マップ)

システム。InterSystems IRIS で InterSystems SQL ストレージを使用する際に、データの保存方法を指定する定義。

2048 バイトのブロック 400 個で構成される InterSystems IRIS データベース内のユニットです。Windows では、フラット・ファイルに属します。

UNIX® では、1 つの UNIX® ファイルカロー・パーティションに属します。



**mapped global reference (マップされたグローバル参照)**

システム。拡張参照構文を使用しない、異なるディレクトリに属するグローバル、またはリモート・グローバルへの論理参照。ユーザからは、データベースのデータの場所にグローバルが存在しているように見えます。そのホーム・ディレクトリは、同じコンピュータでも、ご使用の InterSystems IRIS サーバのネットワーク上の他のコンピュータでもかまいません。システム管理者は、ネームスペース/ネットワーク構成エディタを使用して、マップされたグローバルの実際の位置を定義します。

**matching role (マッチング・ロール)**

システム。セキュリティで保護された InterSystems IRIS アプリケーションで、追加の特権が付与されるようにするロール。ユーザがマッチングロールを保持していると、アプリケーションの使用で、そのユーザにはすべての指定されたターゲット・ロールも付与されます。マッチ・ロールと呼ばれることもあります。

**metadata class (メタデータ・クラス)**

オブジェクト。メタデータ・クラスは、オブジェクト・アプリケーションに保存されたデータを検証するインタフェースを提供します。“%Dictionary.ClassDefinition” クラスを参照してください。

**metadata**

オブジェクト。メタデータは、データおよびデータの構成を示します。

**method generator (メソッド・ジェネレータ)**

オブジェクト。[メソッド・ジェネレータ](#)は、クラス・パラメータの値に基づいて実行時コードを生成するメソッドです。

**method**

オブジェクト。オブジェクトで呼び出すことのできる処理。

**mount、mounted (マウントする、マウントされている)**

システム。データベースをマウントするとは、データベースを InterSystems IRIS のインスタンスに明示的に接続することです。これによって、データベースのコンテンツを InterSystems IRIS のインスタンスで直接使用できるようになります。この状態のデータベースを“マウントされている”と表現します。

**multidimensional property (多次元プロパティ)**

オブジェクト。多次元プロパティは、配列ノードと同様の動作をします。多次元属性は、連動するプロパティ・メソッドを持たず、ドット構文を使用してアクセスすることができません。また SQL や Java に投影することもできません。

**multidrop link (マルチドロップ・リンク)**

一般。2 つ以上のコンピュータに接続しているネットワーク・ハードウェア。

**multiple inheritance (多重継承)**

オブジェクト。多重継承により、1 つのクラスが 1 つ以上のスーパークラスを持つことができます。クラスは、それぞれのスーパークラスから、その特性とクラス・メンバを継承します。

## 1.14 先頭文字が N の用語

### namespace (ネームスペース)

システム。データベースに物理的に格納されているデータとコードへのアクセスを提供する論理エンティティ。データベース・オブジェクトへの参照は、物理的なデータベースの場所ではなく、ネームスペースを介して論理的に行います。グローバルのデータベースの場所はネームスペース・マッピングによって指定されるため、ユーザとコードは、単に特定のネームスペース内でグローバルを参照すればよく、グローバルに格納されているデータの物理的な場所を気にする必要はありません。例えば、ネームスペース・マッピングにより、複数のデータベースから単一のネームスペースにオブジェクトをまとめ、それらすべてが一箇所に保存されているかのように、単一の参照フレームとして機能させることができます。

ネームスペース名の詳細は、“[ネームスペースの構成](#)”を参照してください。

### National Language Support (NLS) (各国言語サポート (NLS))

システム。入力、表示、処理、保存、データ変換の方法に影響を及ぼす各国語間の差異を補うための InterSystems IRIS の機能。NLS は、内部文字セット、照合順、パターン照合、\$X/\$Y アクション、および入出力変換を指定する一連のテーブルから構成されます。システム管理者は、これらのテーブルを NLS ユーティリティで定義しロードします。ユーザは、プロセスに対するテーブル、%NLS ユーティリティを持つデバイス、OPEN コマンドと USE コマンドの K と Y パラメータを選択します。

### network interface device (ネットワーク・インタフェース・デバイス)

一般。コンピュータをネットワーク・リンクに接続するハードウェア。

### network (ネットワーク)

一般。コンピュータと接続の集合。これにより、1 つのコンピュータ上のユーザやプログラムが、ネットワーク内の別のユーザやプログラムと通信することができます。

### node

システム。多次元配列 (グローバル) 内の 1 つのエントリ。

### node

一般。ネットワーク内の 1 つのコンピュータ、またはクラスタ化されたシステム。

## 1.15 先頭文字が O の用語

### object

オブジェクト。特定の項目を表すすべてのデータと、そのデータを操作するインタフェースをカプセル化する論理エンティティ。

### object class (オブジェクト・クラス)

オブジェクト。オブジェクト・クラスは、特定のエンティティを表します。プロパティを持ち、直接インスタンス化することができます。

**object code (オブジェクト・コード)**

システム。InterSystems IRIS コンパイラにより作成される、最もレベルの低いコード。これが、実際に翻訳され、実行されるコードです。

**object identifier (オブジェクト識別子)**

オブジェクト。オブジェクト識別子 (OID) は、データベース全体で、ディスク上のオブジェクトを一意に識別します。OID は、オブジェクトが継続している間は有効ですが、オブジェクトが削除されると再利用することができません。

**object model (オブジェクト・モデル)**

オブジェクト。開発時に使用されるアプリケーションとクラス階層の要件の概要。

**object reference (オブジェクト参照)**

オブジェクト。オブジェクト参照は、現在メモリに存在する特定のオブジェクトを指します。オブジェクト参照は、オブジェクトがインスタンス化、あるいはオープンされてから、クローズされるまでの間のみに有効です。

**ObjectScript**

システム。InterSystems IRIS がサポートするプログラミング言語の 1 つ。

**ObjectScript query (ObjectScript クエリ)**

オブジェクト。ObjectScript クエリは、ObjectScript コードを使用してデータベースに問い合わせます。

**object-SQL projection (オブジェクト SQL プロジェクション)**

オブジェクト。オブジェクト SQL プロジェクションは、InterSystems IRIS オブジェクト機能が InterSystems SQL の要素に投影される方法を指定します。

**ODBC**

一般。“Open Database Connectivity (ODBC)” を参照してください。

**ODBC format (ODBC 形式)**

オブジェクト。ODBC 形式は、データを ODBC に要求される出力形式で表します。

**ODBC Type (ODBC タイプ)**

オブジェクト。データ型の ODBC タイプは、そのデータ型に基づいたプロパティが使用する ODBC データ型を指定します。

**OID**

オブジェクト。“object identifier (オブジェクト識別子)” を参照してください。

**one-way outer join (単方向外部結合)**

InterSystems SQL。プログラマが定義する結合。結合条件で示された最初のテーブルをソース・テーブルと指定し、2 番目のテーブルに一致する行がない場合でも、そのソース・テーブルのすべての行を出力テーブルに含めます。ソース・テーブルは 2 番目のテーブルから関連する情報を引き出しますが、2 番目のテーブルに一致する行がなくても、ソース・テーブルの行を省略することではなく、すべての行を出力します。

## ODBC (Open Database Connectivity)

一般。Microsoft 社の Open Database Connectivity (ODBC) は、Windows でデータにアクセスするための標準インタフェースを提供します。

## Operating-System-based authentication (オペレーティング・システム・ベースの認証)

システム。オペレーティング・システムで認証されたユーザの ID を InterSystems IRIS が受け入れる認証メカニズム。InterSystems IRIS ユーザのアカウントは、オペレーティング・システムのアカウント名と一致する必要があります。また、認証された後は、そのアカウントのロールによってユーザのアクティビティが制限されます。

## OREF

オブジェクト。"object reference (オブジェクト参照)" を参照してください。

## outgoing lock (外向きロック)

システム。リモート・サーバ・コンピュータ上のアイテムに対して、ローカル・クライアント・コンピュータのプロセスにより発行されたロック。ローカル・コンピュータはクライアントなので、クライアント・ロックとも呼ばれます。ターミナルから LOCKTAB ユーティリティを実行したとき、あるいはシステム・オペレーション・ユーティリティの Locks ユーティリティを実行したとき、このロックはロック・テーブル・ディスプレイに表示されます。このとき、アスタリスクの後にロックを発行したプロセスの pid が表示されます。増分ロックは、ロック・テーブルに既存のアイテムに対して発行されます。これらの要求は、ネットワークには送られません。正確には、ローカル・コンピュータが外向きロックの増分ロックの数を追跡します。

## override (オーバーライド)

オブジェクト。既定では、すべてのクラスは、そのスーパークラスのクラス・メンバを継承します。しかし、メンバが最終でなければメンバを選択して変更することができ、これによってサブクラス内で異なった動作をします。これを、メンバのオーバーライドと言います。

# 1.16 先頭文字が P の用語

## package

オブジェクト。[パッケージ](#)は、一連の関連するクラスを含みます。

## パラメータ

オブジェクト。クラス・パラメータは、アプリケーション作成時に設定される特別な定数です。クラスのすべてのインスタンスがクラス・パラメータを利用することができ、あらゆる目的に使用できます。すべてのデータ型クラスは一連のクラス・パラメータを持ち、属性がそのデータ型に基づいて振る舞うように指定します。

## parent ID (親 ID)

オブジェクト。InterSystems IRIS リレーショナル・データベースでは、キャラクタリスティック・リンクが定義されていると、依存する子テーブルの行 ID を定義するために、自動的にフィールドが生成されます。親 ID は、親テーブルと同じ名前を持ち、子テーブルから親テーブルへの指定参照のように動作します。

## parent-child relationship (親子リレーションシップ)

オブジェクト。親子リレーションシップでは、子オブジェクトの存在は、親オブジェクトの存在に依存します。つまり、子オブジェクトは親オブジェクトと連携する必要があり、親オブジェクトが削除されると子オブジェクトも削除されます。

**partition (パーティション)**

システム。メモリのプロセス・プライベート・セクションの別名。各プロセスは、そのプロセス固有のデータのための独自のパーティションを持ちます。InterSystems IRIS パーティション、ユーザ・パーティションとも呼ばれます。UNIX® については、“ロー・ディスク・パーティション” も参照してください。

**passing by reference (参照渡し)**

システム。値ではなく、アドレスを指定して引数を渡す方法。これにより、実際の変数へのアクセスが可能になり、アドレスを受け取ったメソッド、関数、またはルーチンから変数の実際の値を変更できます。

**passing by value (値渡し)**

値を指定して引数を渡す方法。これにより、変数のコピーが実現します。したがって、値を受け取ったメソッド、関数、またはルーチンから変数の実際の値を変更することはできません。

**pattern match table (パターン照合テーブル)**

システム。文字をアルファベット、句読点、数値、あるいは制御文字として処理するか否かを InterSystems IRIS に伝える内部テーブル。

**permission (権限)**

システム。リソースに対して動作を実行するために必要なもの。データベース・リソースの場合、Read および Write の権限があります。サービス、アプリケーション、または管理動作の場合には、Use 権限があります。

**persistent class (永続クラス)**

オブジェクト。永続クラスのオブジェクトは、データベース内に保存されます。永続クラスは、データ保存および検索を管理するために、%Persistent クラスから永続インタフェースを継承します。

**persistent interface (永続インタフェース)**

オブジェクト。オブジェクトの保存と検索に使用される一連のメソッド。

**point-to-point link (ポイント・ツー・ポイント接続)**

一般。2 つのコンピュータを、両端に 1 つずつ接続するネットワーク・リンク。

**polymorphism (ポリモフィズム)**

オブジェクト。オーバーライドされたメソッドへの実行時参照は、オブジェクトのタイプに基づいて処理されます。例えば、Person クラスが オーバーライドされた Print() メソッドを含む Student サブクラスを持つとします。この場合、それらが Person オブジェクトとしてオープンされたときでも、Student オブジェクトは常に Student の Print() メソッドを使用します。

**populate utility (populate ユーティリティ)**

オブジェクト。%Populate クラスにより実装された Populate ユーティリティによって、ご使用のデータベースにダミー・オブジェクトを追加して、テストすることができます。

**port number (ポート番号)**

一般。特定のネットワーク・ポートを識別するデバイス番号。クライアントからサーバに接続する際に、この番号を指定する必要があります。

**port**

一般。ネットワーク・ソフトウェア・プロトコルとインタフェース・デバイスの組み合わせ。1 つ以上のポートが、同じインタフェース・デバイスを使用できます。

**primary key (主キー)**

InterSystems SQL。システムによっては、データを識別するために主キーを使用します。

**primary persistent superclass (プライマリ永続スーパークラス)**

オブジェクト。クラスのプライマリ永続スーパークラスは、そのクラスの永続的な動作を指定します。既定では、プライマリ永続スーパークラスは、スーパークラスのリストの中で、最も左にある永続スーパークラスです。通常、同じプライマリ永続スーパークラスを持つクラスに関するすべてのデータは、まとめて保存されます。

**primary volume (プライマリ・ボリューム)**

システム。ボリューム・セットの中で、最初のボリューム、あるいは唯一のボリュームのこと。

**principal device (主デバイス)**

システム。プロセスに関連する入出力デバイス。通常は、ユーザの端末、あるいはコンピュータのキーボードとモニタを示します。ジョブ化されたプロセスに対し、JOB コマンドで主デバイスを割り当てたり、システム構成エディタで、親プロセスの主デバイスとして設定したりできます。上記の方法でデバイスを設定しない場合、ジョブ化されたプロセスに対する既定のデバイスは NULL デバイスです。

(UNIX®) インタラクティブ・ユーザにとっては、ユーザが InterSystems IRIS にサインオンする端末が主デバイスです。これは、UNIX® がそのユーザに割り当てた主入力デバイスと同じです。ジョブ化されたプロセスに対し、JOB コマンドにより主デバイスを割り当てることができます。既定は、/dev/null です。

**priority setting (優先度設定)**

システム。UNIX® では、優先度はシステム・レベルの順序定義であり、この順序に従ってジョブにシステムのリソースと応答が割り当てられます。

**priority**

システム。バッチ・ジョブ、あるいはシステム・デーモンの重要度。システム・リソースのバランスを取るために、ジョブまたはデーモンの優先度を調節することにより、ご使用のシステムに対する最高の総合的なパフォーマンスを得ることができます。

**private**

オブジェクト。プライベート・クラス・メンバは、そのクラスに属するメソッドからのみアクセスできます。メソッドとプロパティの両方とも、プライベートに設定することができます。

**privilege (特権)**

システム。特定のリソースに特定の動作を実行するための能力の各明細 (例えば DocBook データベースの読み取ることができるなど)。特権を保持できるのはロールのみです。

**privileged routine application (特権ルーチン・アプリケーション)**

システム。セキュリティ上の目的のためにグループ化される 1 つまたは複数の InterSystems IRIS ルーチンのセット。そのようなグループは単一のアプリケーションとして扱われ、単一のリソースとして示されます。

**procedure (プロシージャ)**

システム。文の指定されたシーケンス。ユニットとして実行されます。

**process-private section (プロセス・プライベート・セクション)**

システム。特定のプロセスのみが使用するメモリの領域。

**process**

システム。システム・ソフトウェアによりスケジュールされるエンティティで、イメージを実行するコンテキストを提供します。InterSystems IRIS 内で、プロセスはサーバ・ベースのコードを実行するコンテキストです。

**Product Activation Key (プロダクト・アクティベーション・キー)**

システム。ソフトウェアに同梱されているペーパー・キーには、暗号化された InterSystems IRIS ライセンスが記載されています。ライセンスを起動するには、IRIS.KEY と呼ばれるファイルにこの情報を入力してください。

**programmer mode (プログラマ・モード)**

システム。プログラム開発は、すべてこのモードで実行します。プログラマ・モードでは、InterSystems IRIS プロンプトからプログラムを開始し、実行したすべてのプログラムの終了時にも再度 InterSystems IRIS プロンプトを使用します。プログラマ・モードは、InterSystems SQL 開発環境および実行時環境も含め、InterSystems IRIS 環境とそこから呼び出すことのできるすべてのプログラムに対応します。プログラマ・モードでは、その後にユーザが[アプリケーション・モード](#)で実行できるアプリケーションを作成することができます。

**project (プロジェクト)**

システム。関連するソース文書 (ルーチンやクラス定義など) のユーザ定義リスト。プロジェクトは、アプリケーション開発を体系化するためにスタジオで使用されます。

**property (プロパティ)**

オブジェクト。クラスに関連する固有の特性を表すデータの要素。そのクラスの各オブジェクトは、そのプロパティのリテラル値、あるいはデータを表す別のオブジェクトへの参照を含みます。また、クラスのプロパティはそのクラスの状況も表します。

**property collation (プロパティ照合)**

オブジェクト。プロパティ照合は、属性のデータ型によって提供された既定の変換を、属性の定義で指定された照合でオーバーライドします。

**property method (プロパティ・メソッド)**

オブジェクト。プロパティの動作を制御するために生成されたメソッド。プロパティ名は、そのデータ型からの動作メソッドやプロパティの動作クラスと、コンパイル時に連結され、そのプロパティのプロパティ・メソッドの完全なセットを生成します。

**パブリック**

オブジェクト。パブリック・クラス・メンバは、どのクラスからでも見たり使用することができます。

## 1.17 先頭文字が Q の用語

**query (クエリ)**

一般。指定された基準に合うデータに対する要求。InterSystems IRIS は、SQL や (クラスで定義された) ObjectScript で事前に定義されたクエリ、あるいは (埋め込み SQL や ODBC から) アドホック・クエリをサポートします。

**query interface (クエリ・インタフェース)**

オブジェクト。InterSystems IRIS クエリ・インタフェースは、クエリのタイプや言語にかかわらず、クエリの準備、実行、処理に共通のメカニズムを提供します。

## 1.18 先頭文字が R の用語

**range indicator (レンジ・インジケータ)**

システム。レンジ・インジケータ (:) により、1 つのエントリのグローバルの範囲を示します。例えば、A:Cost は ^Aardvark から ^Cost までのすべてのグローバルを含みますが、^CrossReference は含みません。

**raw disk partition (ロー・ディスク・パーティション)**

システム。UNIX® オペレーティング・システムのディスク区分であり、UNIX® ファイル・システムには関連付けられていません。InterSystems IRIS データベースがロー・ディスク・パーティションに配置されている場合は、このデータベースは UNIX® ファイル名を持ちません。

**RECEIVE**

システム。InterSystems IRIS ネットワーク構成情報を、すべてのリモート・コンピュータに伝える InterSystems IRIS システム・プロセス。

**reconfiguration (再構成)**

システム。InterSystems IRIS ネットワーキングで、ネットワーク・ハードウェア、リモート・システム、データセット、ネームスペース定義のいずれかを変更すること。

**recovery (リカバリ)**

システム。システム障害が発生した後に、喪失したデータの整合性を回復するメソッド。

**redirection (リダイレクト)**

システム。グローバルの位置をアプリケーションから見るができないようにする方法。InterSystems IRIS は、ネームスペース定義と複製という 2 つのタイプのリダイレクトをサポートします。

**reference**

オブジェクト。参照は別のオブジェクトやテーブルを指し示し、プロパティやフィールドと、参照されたオブジェクトやテーブルとの間に一方向のリレーションシップを作成します。

**registered class (登録クラス)**

オブジェクト。登録クラスは、%RegisteredObject クラスから派生します。InterSystems IRIS は、自動的に登録クラスのオブジェクト参照を管理し、ポリモフィズム (多態) をサポートします。

**relational database (リレーショナル・データベース)**

一般。リレーショナル・モデルに従って体系化された、関連のあるデータの集合。

**relationship (リレーションシップ)**

オブジェクト。2 つの異なるクラスのプロパティ間の、双方向の参照。



**relative dot syntax (相対ドット構文)**

ObjectScript. 相対ドット構文 (..) は、プロパティの参照、あるいは現在のクラスのメソッド呼び出しに使用します。

**remote computer (リモート・コンピュータ)**

システム。ネットワーク・リンクを経由してローカル・コンピュータに接続されるすべてのコンピュータ。

**replicated global (複製グローバル)**

システム。ネームスペース・マッピングが、複製のコピー用に、プライマリ・データ位置以外にも 1 つ以上の位置を定義しているグローバル。複製されたグローバルのオリジナルに対して実行された SET あるいは KILL 処理は、そのグローバルのすべてのコピーに対して実行されます。複製されたグローバルのコピーに対して実行された SET あるいは KILL 処理は、コピーに対してのみ影響を与えます。

**required field (必須フィールド)**

オブジェクト。必須フィールドは、行が保存される前に、有効な NULL でない値を持つ必要があります。"Required Property (必須プロパティ)" を参照してください。

**required property (必須プロパティ)**

オブジェクト。必須プロパティは、オブジェクトを保存する前に、値を割り当てる必要があります。

**resource**

システム。InterSystems のセキュリティで保護可能な最小単位。1 つのリソースは 1 つまたは複数のアセットを表します。アセットとは、データベース、アプリケーションへのアクセス、または他の要素です。一般的に、リソースには同種のアセットのみを含めることができます。

**result set (結果セット)**

オブジェクト。結果セットは、クエリによって返されたデータを含みます。これは、クエリ・インタフェースを使用して処理されます。

**role (ロール)**

システム。アクションの実行とデータの表示/変更の能力を付与できる、InterSystems のセキュリティのエンティティ。これらの能力は特権という形式でロールに付与されます。ユーザがロールのメンバになると、ロールの特権を付与されます。

**ロール割り当てメカニズム**

システム。InterSystems IRIS を使用するためのさまざまな[ロール](#)をユーザに割り当てる手段です。ロールの割り当ては、InterSystems IRIS に最初から備わっているメカニズム、LDAP、または代行 (ユーザ定義) コードを使用して行うことができます。これは[承認](#)プロセスの一部です。

**roll back (ロール・バック)**

システム。ご使用の InterSystems IRIS データベースから不完全なトランザクションを削除するプロセス。InterSystems IRIS は、これらの変更をリカバリ・プロシージャの過程で削除します。"Journaling (ジャーナリング)"、"Transaction (トランザクション)" を参照してください。

**roll forward (ロール・フォワード)**

システム。システム障害が発生した後、バックアップからデータベースを再度作成するイベントの中で、ジャーナルされたデータベースの変更を再び適用するプロセス。"Journaling (ジャーナリング)" を参照してください。

**root class (ルート・クラス)**

オブジェクト。エクステントのルート・クラスは、その階層の中で最上位のクラスです。すべてのクラスは、そのクラスのエクステントのルート・クラスです。しかし、特にデータ・ストレージのコンテキストでは、通常ルート・クラスはクラス階層のプライマリ永続スーパークラスを意味します。

**routine**

システム。InterSystems IRIS では、ObjectScript プログラム。ルーチンは InterSystems IRIS データベースに保存されます。

**routine buffers (ルーチン・バッファ)**

システム。仮想メモリ内で、ルーチンが使用できる領域の量。

**row (行)**

InterSystems SQL。リレーショナル・テーブルのドメインのエンティティを表す、関連するフィールド値のグループ。例えば、**Customers** テーブルでは、1 行に 1 人の顧客について記述します。従来のデータ・プロセス用語ではレコード、リレーショナル・データベース用語ではタプルとも呼ばれます。

**Row ID (行 ID)**

オブジェクト。行 ID は、テーブルの行を一意に識別します。テーブルが InterSystems IRIS クラスから投影される場合、行 ID は相当するオブジェクトの ID (自動的に生成されるか、ID キーにより指定されたもの) です。

**row specification (行指定形式)**

オブジェクト。クラス・クエリの行指定形式は、そのクエリに返されるフィールドの情報を提供します。

## 1.19 先頭文字が S の用語

**schema evolution (スキーマ進化)**

オブジェクト。スキーマ進化により、従来のクラス定義で保存されたデータにアクセスする機能を失わずに、新規のクラス・メンバを追加することができます。

**search user (検索ユーザ)**

システム。InterSystems IRIS で LDAP サーバに接続し、LDAP データベースを検索できる特権を持っているユーザ。この値は、[LDAP 構成] ページ ([システム管理] > [セキュリティ] > [システム・セキュリティ] > [LDAP 構成]) の [LDAP 検索ユーザ DN] フィールドまたは [検索に使用するLDAPユーザ名] で指定します (Kerberos が有効になっている場合は、ページ名とメニュー項目に Kerberos が含まれることに注意してください)。

**secondary volume (セカンダリ・ボリューム)**

システム。ボリューム・セットに含まれる、プライマリ・ボリューム以外のボリューム。InterSystems IRIS データベースでは、ゼロから 7 までの、合計 8 個までセカンダリ・ボリュームを持つことができます。

**Security Advisor (セキュリティ・アドバイザー)**

システム。一般的に推奨されるセキュリティ基準と異なる設定が InterSystems IRIS インスタンスに存在する場合に、それに関するフィードバックを提供する、管理ポータル診断ツールです。

**security domain (セキュリティ・ドメイン)**

システム。Kerberos レalmや Windows ドメインに相当する、マシンの論理的なグループ。

**selectivity (選択性)**

InterSystems SQL。プロパティの選択性により、値の全体の分布の中で、特定の値が出る頻度を指定します。

**sequential file (シーケンシャル・ファイル)**

一般。このファイルでは、データが入力された順番に保存されます。

**server lock (サーバ・ロック)**

システム。“Incoming Lock (内向きロック)” を参照してください。

**server master (サーバ・マスタ)**

システム。サーバ・ソフトウェアのコンポーネント。TCP 経由で接続されたクライアントからの接続要求を検知し、それらの接続に対応するためのサーバ・プロセスを生成します。各サーバ・マスタは InterSystems IRIS プロセスです。

**server process (サーバ・プロセス)**

システム。1 つのクライアント接続に対応する、サーバ上のジョブ化されたプロセス。サーバ・マスタは、クライアントに接続するためにサーバ・プロセスを生成します。

**service**

システム。既存の経路 (Telnet、JDBC など) を介した InterSystems IRIS へのアクセスを規制する、InterSystems のセキュリティのエンティティ。InterSystems IRIS への接続時には、アプリケーションはサービスの上部に位置します。

**shallow save (シャロウ・セーブ)**

オブジェクト。シャロウ・セーブは、オブジェクトを保存しますが、それが参照するオブジェクトは保存しません。詳細は、“%Persistent.%Save” メソッドのクラス・リファレンス・コンテンツを参照してください。

**shared lock (共有ロック)**

オブジェクト。共有ロックは、オブジェクトがデータベースからロードされる間は共有ロックを保持し、オブジェクトが 1 つ以上のノードにわたる場合や更新されている場合、保存される間は排他ロックを保持します。1 つのノードにあるオブジェクトが初めて保存される場合は、ロックは適用されません。

**shared retained lock (共有保持ロック)**

オブジェクト。共有保持ロックは、オブジェクトがデータベースからロードされてからクローズするまでの間は共有ロックを保持し、オブジェクトが 1 つ以上のノードにわたる場合や更新されている場合に、保存している間は排他ロックを保持します。1 つのノードにあるオブジェクトが初めて保存される場合は、ロックは適用されません。

**SOAP**

一般。Web サービスを実装するためのプロトコル仕様。そのメッセージ形式として XML を使用し、通常はメッセージのネゴシエーションおよび転送に対してリモート・プロシージャ呼び出し (RPC) や HTTP などの他のアプリケーション層プロトコルに依存します。**SOAP** は、以前は Simple Object Access Protocol を表す頭文字でしたが、現在は単にプロトコルの名前になりました。

**special variable (特殊変数)**

システム。InterSystems IRIS が持つ変数のうちの 1 つ。アプリケーションの開発において、特殊変数を調べることで、システムについての情報を得ることができます。

**SQL**

一般。Structured Query Language (構造化照会言語)。SQL は ANSI 標準の第 4 世代プログラミング言語で、特にリレーショナル・データベースにアクセス、維持するために設計されたものです。

**SQLCODE**

InterSystems SQL。実行された SQL 文の状態を保持するローカル変数。実行時にエラーが発生した場合、SQLCODE はエラーの番号を示します。

**SQL computed field (SQL 計算フィールド)**

オブジェクト。SQL 計算フィールドは、要求に応じて、他のフィールドに保存された値に基づいて計算されます。

**SQL mapping (SQL マッピング)**

オブジェクト。SQL マッピングは、既存のデータ構造をクラスやテーブルにマップするために使用します。

**SQL query (SQL クエリ)**

InterSystems SQL。SQL クエリは、SQL を使用して、データベースに保存されている情報の場所を確認し、検索します。

**SQL reserved words (SQL 予約語)**

InterSystems SQL。[SQL 予約語](#)は、InterSystems SQL において特別な意味を持ち、テーブル、フィールド、クエリ、ビュー、ストアド・プロシージャ、インデックス名に使用することができません。SQL 予約語は、代わりのテーブル名およびフィールド名が定義されている場合、クラス名およびプロパティ名として使用することができます。

**status code (状態コード)**

オブジェクト。状態コードは、いくつかのメソッドによって返されるコードで、そのメソッドの実行が成功したか否かを示します。[%Status](#) を参照してください。

**storage definition (ストレージ定義)**

オブジェクト。ストレージ定義は、ストレージ・クラスと定義された SQL マッピングから構成されます。これらの要素が、データを保存する方法を指定します。

**storage interface (ストレージ・インタフェース)**

オブジェクト。カスタム・ストレージを使用するとき、あるいはユーザ独自のストレージ・クラスを記述するときに、実装する必要のある一連のメソッド。

**storage strategy (ストレージ・ストラテジ)**

オブジェクト。クラスは、コンパイル時にストレージ・ストラテジを使用してストレージ定義を評価し、データを保存する方法を指定します。

**stored procedure (ストアド・プロシージャ)**

InterSystems SQL。ストアド・プロシージャにより、ODBC または JDBC からクエリやクラス・メソッドを実行することができます。

**stream interface (ストリーム・インタフェース)**

オブジェクト。InterSystems IRIS ストリーム・インタフェースは、ObjectScript、SQL、Java でストリームを操作するのに使用します。

**stream**

オブジェクト。ストリームは、大量のデータの塊の操作や保存に使用するインタフェースを提供します。InterSystems IRIS ストリーム・インタフェースは、ObjectScript、SQL、Java でストリームを操作するのに使用します。

**superserver (スーパーサーバ)**

システム。指定したポートで InterSystems IRIS への着信接続をリッスンし、それを適切なサブシステムに配信するサーバ。既定のスーパーサーバ・ポートは 1972 です。1972 が使用できない場合、スーパーサーバは 51773 以降で最初に使用可能なポートをリッスンします。スーパーサーバのポート番号を設定するには、管理ポータル の [メモリと開始設定] ページの [スーパーサーバポート番号] フィールドを使用します ([システム管理]→[構成]→[システム構成]→[メモリと開始設定])。

**swizzling (スウィズリング)**

オブジェクト。オブジェクトが参照されるとすぐに、埋め込みオブジェクトおよび永続オブジェクトを自動的にメモリへ読み取るプロセス。遅延ロードとも呼ばれます。

**system class (システム・クラス)**

オブジェクト。InterSystems IRIS に組み込まれた機能を提供するクラス。

**system configuration (システム構成)**

システム。InterSystems IRIS が起動時に使用するシステム・リソースの定義。システム構成は管理ポータルで定義します。

**system manager's directory (システム管理者用ディレクトリ)**

システム。InterSystems IRIS データベースが属するディレクトリ。InterSystems IRIS システム・グローバル、システム・ルーチン、InterSystems IRIS マネージャ用ルーチン、%ユーティリティなどを含みます。ご使用の InterSystems IRIS インストール・ディレクトリでは、サブ・ディレクトリ MGR がシステム管理者用ディレクトリです。

**system name (システム名)**

システム。ネットワークのノードに割り当てられた名前。ネットワーク内で一意である必要があります。ホスト名、あるいはコンピュータ名とも呼ばれます。ネームスペース/ネットワーク構成エディタで使用され、ネットワーク構成のコンピュータを識別します。MNET ユーティリティのディレクトリ・セットと呼ばれます。

**system processes (システム・プロセス)**

システム。(Windows) Windows では、プロセスの優先度を調節することはできません。

(UNIX®) UNIX® では、優先度は nice 値により制御されます。プロセスの nice 値を上げると優先度が低くなり、nice 値を下げると優先度が高くなります。

## 1.20 先頭文字が T の用語

### table (テーブル)

InterSystems SQL。特定のエンティティを表す行と、各エンティティについての特定のデータ・ポイントを表す列から成るデータ構造。

### target role (ターゲット・ロール)

システム。セキュリティで保護された InterSystems IRIS アプリケーション内で、既に別のロール (マッチングロールと呼ばれる) のメンバであるユーザにアプリケーションが付与するロール。ユーザがマッチングロールを保持していると、アプリケーションの使用で、そのユーザに 1 つ以上の追加のターゲット・ロールを付与することができます。

### target user (ターゲット・ユーザ)

システム。LDAP サーバへの認証を試行しているユーザ。InterSystems IRIS は、特定の LDAP 構成の [LDAP 構成の編集] ページの [LDAP に一意の検索属性] フィールドに指定されている値を使用して、LDAP データベース内でこのユーザの検索を試みます。[LDAP 構成の編集] ページには、[LDAP 構成] ページ ([システム管理] → [セキュリティ] → [システム・セキュリティ] → [LDAP 構成]) からアクセスできます。

### TCP/IP

一般。Transmission Control Protocol/Internet Protocol の略。リレーショナル・クライアントとリレーショナル・サーバとの間の接続を管理することができる通信プロトコルの 1 つ。TCP とも言われます。

### 一時グローバル

システム。一時データベース IRISTEMP に格納されるグローバル。“[一時グローバルと IRISTEMP データベース](#)” を参照してください。

### ターミナル

システム。正式には、この用語は Windows ターミナル・アプリケーションを指します。非公式には、InterSystems IRIS に対するインタラクティブなコマンド行インタフェースである ObjectScript シェルを指す場合もあります。ターミナル・アプリケーションは ObjectScript シェルだけでなく、メニューと追加オプションも提供します。“[ターミナルの使用法](#)” および “[ObjectScript シェルの使用法](#)” を参照してください。

### transaction (トランザクション)

一般。動作の論理ユニット。アプリケーション開発者は、SQL または ObjectScript コマンドを使用して、トランザクションを定義できます。InterSystems IRIS は、ジャーナル・ファイルのトランザクション内のグローバルに最新情報を記録します。トランザクションが完了しなければ、ロール・バックできます。

### transient property (一時プロパティ)

オブジェクト。一時プロパティは、ディスクではなくメモリ内に保存されます。

### translation methods (変換メソッド)

オブジェクト。変換メソッドは、ODBC 形式、表示形式、論理形式、ストレージ形式間で値を伝達するために使用します。

## トリガ

InterSystems SQL。InterSystems SQL アプリケーション、あるいは InterSystems SQL ストレージを使用したオブジェクト・アプリケーションのさまざまな段階で実行されるように、開発者によって定義されたアクションの順序。トリガはデータベースのアクションで、テーブルで実行される INSERT、UPDATE、DELETE アクションによって開始します。トリガは、整合性制約やその他のデータ依存関係の維持に役立ちます。

## 1.21 先頭文字が U の用語

### UIC

システム。ユーザ識別コード。データベースの UIC は、そのデータベースでグローバルへのアクセス権を持つユーザを指定します。UNIX® では、UIC は groupid、それに続くコンマ(,)、および userid で構成されています。システム管理者は、UNIX® レベルでユーザに UIC を割り当て、InterSystems IRIS MSU ユーティリティを使用して各データベースに UIC を割り当てます。

### unauthenticated access (認証されないアクセス)

システム。ユーザ ID に基づいた検証がされない InterSystems IRIS へのアクセス。

### Unicode

システム。16 ビットの文字セットを処理するために設計された InterSystems IRIS 照合。ANSII では 256 文字しか使用できないのに対し、Unicode では 64,000 文字を使用できます。この照合により、256 文字以上を持つ他言語の文字セットに、スムーズに移行することができます。

### unique index (一意のインデックス)

オブジェクト。一意のインデックスは、インデックス内の属性や属性の組み合わせに対して、各オブジェクトが一意の値を持つことを指定します。

### unmounted (マウントされていない)

システム。マウントされていないデータベースとは、InterSystems IRIS のインスタンスで使用可能な、現在使用されていないデータベースです。マウントされていないデータベースのコンテンツを参照するには、そのデータベースを明示的にマウントします。

### user account (ユーザ・アカウント)

システム。ご使用のシステムの特定のルーチンまたはネームスペースに、個々のユーザがリモートで、またはローカルでログインすることを許可するエンティティ。システム・オペレーション・ユーティリティの User Acct を使用してユーザ・アカウントやデバイス・アカウントを設定し、特定のルーチン、データベース、ネームスペースへの各ユーザのアクセス権を管理することができます。

### user process (ユーザ・プロセス)

システム。ユーザが InterSystems IRIS にサインオンした特定の端末に関連する InterSystems IRIS インタラクティブ・プロセス。



## 1.22 先頭文字が V の用語

### valid subscript ranges (有効な添え字範囲)

システム。指定された範囲内にあるすべての添え字が有効であるときに、添え字範囲は有効です。例えば、有効な添え字範囲は、(1):(10) あるいは ("a"):(“P”) のように表されます。最初の添え字は範囲に含まれ、2 番目の添え字は含まれません。つまり、(1):(10) には添え字の値 1 は含まれますが値 10 は含まれません。これにより、InterSystems IRIS で添え字範囲の上限を定義できます。

### variable (変数)

システム。データ値を参照するために使用される記号名。変数は、ローカルにもグローバルにもなり得えます。ローカル変数は、任意のパーティションに関連するローカル・シンボル・テーブルに属し、グローバル変数は、ディスクに属します。

### view (ビュー)

InterSystems SQL。ビューは、1 つ以上のテーブルから、特定の基準に合うデータを表示します。

### virtual field (仮想フィールド)

InterSystems SQL。1 つの保存値に直接的に一致するのではなく、数個の保存値からなるフィールド。例えば、子テーブルの行 ID は 2 つの保存値、つまり親テーブルの行 ID と子テーブルの特定の行に対応する添え字からなることがあります。

### virtual table (仮想テーブル)

InterSystems SQL。1 つ以上のベース・テーブルから取得した指定されたテーブル。物理ストレージで直接表示されません。ビューやクエリの出力は、この仮想テーブルの一例です。

## 1.23 先頭文字が W の用語

### watchpoint (ウォッチポイント)

システム。ZBREAK コマンド内で使用する変数。SET コマンドあるいは KILL コマンドで値を変更すると、ルーチンの実行を中断できます。その後、ZBREAK コマンド内で定義したデバッグを中断する場合があります。

### Web アプリケーション

システム。InterSystems IRIS に登録でき、その結果として InterSystems のセキュリティ・モデルの一部として制限される、Web ベースのアプリケーション。Web アプリケーションのユーザは、サポートされる認証メカニズムのいずれかを使用して、InterSystems IRIS で認証される必要があります。

### 作業セット

システム。通常は指定のアプリケーション・ワークロードによってメモリにキャッシュされるデータ。“データベース・キャッシュ”を参照してください。

### write image journaling (ライト・イメージ・ジャーナリング)

システム。InterSystems IRIS データベースを更新するメソッド。ライト・デーモンが実際にはまだデータベースに記入していない、すべての変更のコピーを維持することにより、データベースの内部的な整合性の低下を最小限に抑えます。



## 1.24 先頭文字が X の用語

### XData

一般。コンパイル後に使用するために、適格な XML の指定ブロックを保持しているクラス・メンバ。XData ブロックには任意の構造化した情報を収めることができますが、一般的にはプロダクション関連クラスで使います。

