



クラウド・ストレージへのアクセス

Version 2024.1
2024-06-06

クラウド・ストレージへのアクセス

InterSystems IRIS Data Platform Version 2024.1 2024-06-06

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動および維持するためにのみ提供される。この発行物のいかなる部分も他の目的のために使用してはならない。また、インターシステムズ社の書面による事前の同意がない限り、本発行物を、いかなる形式、いかなる手段で、その全てまたは一部を、再発行、複製、開示、送付、検索可能なシステムへの保存、あるいは人またはコンピュータ言語への翻訳はしてはならない。

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

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

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

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

InterSystems Worldwide Response Center (WRC)

Tel: +1-617-621-0700

Tel: +44 (0) 844 854 2917

Email: support@InterSystems.com

目次

1 クラウド・ストレージ・アダプタの概要	1
2 クラウド・ストレージの受信アダプタ	3
2.1 BLOB の取得	3
2.2 ビジネス・サービスの作成	4
3 クラウド・ストレージ用の送信アダプタ	5
3.1 BLOB の削除	5
3.2 BLOB のアップロード	5
3.3 ビジネス・オペレーションの作成	6
4 クラウド・ストレージ・アダプタの詳細	7
5 クラウド・ストレージ API	9
5.1 クライアントの作成	9
5.1.1 クライアントの終了	10
5.2 バケットとの連携	10
5.2.1 バケットの詳細	10
5.3 BLOB 情報の取得	11
5.3.1 BLOB の詳細	11
5.4 BLOB のアップロード	11
5.5 BLOB のダウンロード	12
5.6 BLOB をアップロードするための単一のメソッド	12
5.7 BLOB をダウンロードするための単一のメソッド	12
5.8 BLOB の削除	13

1

クラウド・ストレージ・アダプタの概要

InterSystems IRIS® により、Amazon Web Services (AWS)、Azure Blob Storage (Azure)、Google Cloud Platform (GCP) などのクラウド・ストレージ・プロバイダのデータを取得、格納、および削除することが容易になります。インターシステムズ製品を使用してクラウド・ストレージにアクセスする際には、相互運用プロダクションを使用する、または低レベルの API を呼び出すという 2 つのオプションがあります。

相互運用プロダクションは、外部システムに接続することを目的としており、ユーザはこれを使用してクラウド・ストレージにアクセスできます。クラウド・ストレージからプロダクションにデータを読み込む場合、[受信アダプタ](#)を使用するビジネス・サービスを作成します。[送信アダプタ](#)を使用するビジネス・オペレーションを作成すると、クラウド内のデータを削除またはクラウドにアップロードすることができます。

[低レベル API](#) では、プロダクション・フレームワークを使用することなく、クラウド・ストレージにアクセスするコードを作成できます。これらは、コード内のシンプルな呼び出しを使用して、クラウド・ストレージ・プロバイダへのプログラムによるアクセスを提供します。

AWS では、データはオブジェクトとして格納されます。ただし、その他のクラウド・ストレージ・プロバイダでは、BLOB という用語で同じ概念を表しています。InterSystems IRIS 内では、クラウド・ストレージのデータは BLOB と呼ばれています。

2

クラウド・ストレージの受信アダプタ

受信アダプタにより、ビジネス・サービスはクラウドからデータを取得し、[相互運用プロダクション](#)内で処理することができます。このアダプタは、クラウド・ストレージからデータを取得し、InboundInput オブジェクトに配置します。これは、ビジネス・サービスがデータを処理するために使用するオブジェクトです。

受信アダプタのクラス名は `EnsLib.CloudStorage.InboundAdapter` です。

[クラウド・ストレージに接続する](#)ために使用されるプロパティに加え、受信アダプタには、取得する BLOB や、アダプタによって取得された後でクラウド BLOB を削除するかどうかを決定するプロパティも含まれます。

2.1 BLOB の取得

受信アダプタには、クラウド・ストレージから取得する BLOB を決定する 2 つのプロパティ、**Blob Name Prefix** および **Blob Name Pattern** が含まれています。これらのプロパティがどのように連携するのかを理解するには、例を確認するのが最も簡単です。次の BLOB を含む AWS S3 バケットについて考えます。

```
foo/bar/baz
foo/bar/bash
foo/bar/bang
foo/boo
```

AWS では BLOB 名に / を使用して仮想階層を作成します。これはファイル・システムがファイルをディレクトリに編成する方法に似ています。このスキーム内で、**Blob Name Prefix** はディレクトリ名のように機能します。例えば、`foo/` ではすべての BLOB が選択され、`foo/bar/` では最初の 3 つの BLOB が選択されます。これらの選択は、AWS サーバ側で行われます。

クライアントがサーバから BLOB のリストを取得すると、そのリストをさらにフィルタ処理するために **Blob Name Pattern** が使用されます。例えば、`BlobNamePrefix="/foo/bar/"` かつ `BlobNamePattern="*ba?"` の場合、アダプタは最初の BLOB のみを取得します。このフィルタ処理は、クライアント側で行われます。**Blob Name Pattern** プロパティは * と ? のワイルドカードをサポートしています。

Blob Name Prefix と **Blob Name Pattern** によって設定された条件に合う BLOB が複数存在する場合、アダプタは各 BLOB をそれぞれビジネス・サービスの別の InboundInput オブジェクトに転送します。

2.2 ビジネス・サービスの作成

受信アダプタを使用するためのカスタム・ビジネス・サービスを開発する際、ユーザは、最初の引数として `EnsLib.CloudStorage.InboundInput` オブジェクトを受け入れるように `onProcessInput` オブジェクトを定義する必要があります。受信アダプタは、クラウド・ストレージのデータをこのオブジェクト・タイプに配置します。ビジネス・サービスで `InboundInput` オブジェクトを操作する際には、その 3 つのプロパティ、`Name` (クラウド・ストレージ BLOB の名前)、`Meta` (クラウド・ストレージ BLOB に関連付けられたメタデータ)、および `Content` (クラウド・ストレージからのデータを含むストリーム) にアクセスできます。

カスタムのビジネス・サービスの開発に不慣れな場合、詳細は [“ビジネス・サービスの定義”](#) を参照してください。

3

クラウド・ストレージ用の送信アダプタ

相互運用プロダクション内のビジネス・オペレーションは、送信アダプタを使用して、クラウド・ストレージから BLOB を削除したり、新しい BLOB をアップロードすることができます。すべてのビジネス・オペレーションと同様、このビジネス・オペレーションは、いずれかのアダプタ・メソッドを呼び出すことにより、これらのアクションを実行します。クラウド・ストレージ・プロバイダへの接続を可能にする、アダプタのプロパティの詳細は、“[クラウド・ストレージ・アダプタの詳細](#)”を参照してください。

送信アダプタのクラス名は `EnsLib.CloudStorage.OutboundAdapter` です。

3.1 BLOB の削除

ビジネス・オペレーションは、送信アダプタの `DeleteBlob` メソッドを呼び出すことにより、クラウド・ストレージから BLOB を削除します。例えば、ビジネス・オペレーションは以下を呼び出すことができます。

ObjectScript

```
Set tSC = ..Adapter.DeleteBlob(..BucketName, request.BlobName)
```

この例で、ビジネス・オペレーションは、クラウド・ストレージのバケットを特定するプロパティを使用し、プロダクションは、削除される BLOB の名前を指定する要求メッセージを使用します。

3.2 BLOB のアップロード

送信アダプタは、データのソースまたはデータ型に応じて、BLOB をアップロードするための 3 つの異なるメソッドを提供しています。

- ・ `UploadBlobFromStream` – `%Stream.Object` としてデータを受け入れます。
- ・ `UploadBlobFromString` – 文字列としてデータを受け入れます。
- ・ `UploadBlobFromFile` – ファイル・システムからデータを取得します。

各メソッドの最初の 2 つのパラメータは、バケット名と BLOB 名を指定します。各メソッドの 3 番目のパラメータは、ビジネス・オペレーションからデータまたはデータの場所を受け取るために使用されます。例えば、ストリームとしてのデータを含むメッセージを受信するビジネス・オペレーションは、以下を呼び出すことができます。

ObjectScript

```
Set tSC = ..Adapter.UploadBlobFromStream(..BucketName,  
    request.BlobName, request.Content)
```

3.3 ビジネス・オペレーションの作成

[クラウド・ストレージへの接続](#)に使用されるほとんどのプロパティは、送信アダプタで定義されます。ただし、プロダクションが操作する実際のバケット名は、ビジネス・オペレーションで定義し、アダプタ・メソッドに渡される必要があります。一般に、ビジネス・オペレーションは、管理ポータルに表示されるプロパティを使用して、バケット名を定義します。

カスタムのビジネス・オペレーションの開発に関する一般的なガイダンスが必要な場合は、“[ビジネス・オペレーションの定義](#)”を参照してください。

利便性のため、インターシステムズでは、簡単なビジネス・オペレーション、**EnsLib.CloudStorage.BusinessOperation**を提供しています。これは、BLOB を削除し、ストリームから BLOB をアップロードする方法を示しています。サンプルのビジネス・オペレーションでは、2 つの簡単なメッセージ・クラス、**EnsLib.CloudStorage.DeleteRequest** と **EnsLib.CloudStorage.UploadRequest** を使用して、BLOB の削除とアップロードを行います。

EnsLib.CloudStorage.BusinessOperation には、前述のとおり、プロパティ **BucketName** が含まれます。送信アダプタにはこのプロパティは含まれないため、カスタム・ビジネス・オペレーションにこのプロパティを含める必要があります。

4

クラウド・ストレージ・アダプタの詳細

受信アダプタと送信アダプタのどちらにも、クラウド・ストレージ・プロバイダを特定し、そのプロバイダにアクセスするために必要な認証情報を提供するプロパティが含まれています。これらのプロパティの値は、アダプタを使用するビジネス・サービスまたはビジネス・オペレーションがプロダクションに追加されると、管理ポータルで定義できます。クラウド・ストレージ・プロバイダへの接続に使用されるプロパティは次のとおりです。

注釈 設定名では AWS の用語が使用されますが、それらは、同等のプロパティに置換することによって、任意のクラウド・ストレージ・プロバイダで使用できます。例えば、Microsoft Azure を使用する場合、**BucketName** プロパティに“コンテナ名”を使用します。

BucketName (受信のみ) – 操作する BLOB を含むクラウド・ストレージ・バケットを指定します。送信アダプタにこの設定は含まれません。代わりに、関連するビジネス・オペレーションで **BucketName** プロパティを定義する必要があります。用意されているビジネス・オペレーション **EnsLib.CloudStorage.BusinessOperation** には、この設定が含まれています。詳細は、“[ビジネス・オペレーションの作成](#)”を参照してください。

BlobNamePrefix (受信のみ) – “[BLOB の取得](#)”を参照してください。

BlobNamePatter (受信のみ) – “[BLOB の取得](#)”を参照してください。

DeleteAfterDownload (受信のみ) – ダウンロード後に BLOB を削除する必要があることを示すには、このボックスをチェックします。

StorageProvider – クラウド・ストレージ・プロバイダを指定します。

EndPoint – PrivateLink エンドポイントを指定します。

ProviderCredentialsFile – プロバイダへのアクセスに必要な認証情報。これらは安全に格納する必要があります。認証情報ファイルのファイル・パスを指定します。

- ・ AWS – AWS では、これを空白のままにして、[既定の認証情報プロバイダ・チェーン](#)を使用して、S3 バケットへのアクセスに必要な認証情報を取得できます。認証情報ファイルを使用する場合は、AWS から認証情報ファイルをダウンロードしてからそのファイル・パスを指定できます。詳細は、“[Sign Up for AWS and Create an IAM User](#)”を参照してください。
- ・ GCP – “[サービス アカウント キーの作成と管理](#)”に従って、アクセスの認証情報を作成します。
- ・ Azure – Azure は、認証情報ファイルをサポートしていません。代わりに接続文字列を使用します。詳細は、“[Azure Storage の接続文字列を構成する](#)”を参照してください。接続文字列には、セミコロンで区切られたキーと値のペアが含まれます。この文字列は編集してセミコロンを削除し、キーと値のペアのそれぞれを独自の行に配置する必要があります。

接続文字列のサンプルを以下に示します。

```
DefaultEndpointsProtocol=https;AccountName=sampleuser;AccountKey=5X774mwEs41WxQsOw19PB2Y;EndpointSuffix=core.windows.net
```

これを分解して、以下のようなファイルを作成する必要があります。

```
DefaultEndpointsProtocol=https
AccountName=sampleuser
AccountKey=5X774mvEs4lWxQsOw19PB2Y
EndpointSuffix=core.windows.net
```

StorageRegion — クラウド・ストレージのリージョンを指定します。

- ・ AWS — AWS リージョンのリストは、“[Amazon Regions, Availability Zones, and Local Zones](#)”を参照してください。
- ・ GCP — GCP リージョンのリストは、“[バケットの保存場所](#)”を参照してください。
- ・ Azure — リージョンは接続文字列に暗黙的に示されます。明示的な設定はありません。

注釈 クラウド・ストレージ・アダプタは、InterSystems PEX フレームワークを使用して開発されているため、このアダプタの ObjectScript ソース・コードは、他のアダプタとは異なるように見えます。例えば、送信アダプタのメソッドは、実際には Java PEX コンポーネント内に記述されたメソッドのラップです。

5

クラウド・ストレージ API

ObjectScript コードは、一連の低レベル API を呼び出すことで、クラウド・ストレージ・プロバイダにデータをアップロードしたり、ここからデータをダウンロード、あるいは削除することができ、ユーザは相互運用プロダクションを使用することなく、クラウド・ストレージにアクセスできます。コードは、クライアントを作成し、クライアントのメソッドを呼び出して、BLOB のアップロードや削除などのアクションを実行することで、クラウド・ストレージ・プロバイダとやり取りします。このクラウド・ストレージ・クライアントのクラスは、`%Net.Cloud.Storage.Client` です。これは、各クラウド・ストレージ・プロバイダで同じクラスです。

クラウド・ストレージ API は簡単に使用できます。例えば、Amazon Web Services S3 バケットにファイルをアップロードするために必要なのは、次のコードのみです。

ObjectScript

```
Set bucketName = "s3-bucket"
Set blobName = "s3-object-blob"
// Create Cloud Storage Client for S3
Set myClient = ##class(%Net.Cloud.Storage.Client).CreateClient(,0,
    "/home/AWSCredentials", "us-east-1", .tSC)

// Upload file to S3
If myClient.BucketExists(bucketName){
    Do myClient.UploadBlobFromFile(bucketName, blobName, "/usr/file.jpg")
}
// Close client
Do myClient.Close()
```

5.1 クライアントの作成

クラウド・ストレージ・プロバイダのバケットと BLOB を処理する前に、コードで次の構文を使用してクラウド・ストレージ・クライアントを作成する必要があります。

ObjectScript

```
Set myClient = ##class(%Net.Cloud.Storage.Client).CreateClient(javaServer,
    provider,credentialsFile,region,.tSC,endpoint)
```

説明：

- ・ `javaServer` は、インターシステムズの Java 用外部サーバ (Java ゲートウェイとも呼ばれます) の名前です。カスタムのサーバを作成するのではなく、既定の Java 外部サーバを使用するには、この引数を空のままにします。
- ・ `provider` は、このクライアントでアクセスするクラウド・ストレージ・プロバイダを示す整数です。S3 バケットの場合、0 を使用します。

- ・ `credentialsFile` は、クラウド・ストレージ・プロバイダにアクセスするために使用される認証情報を含むファイルです。このファイルは、プロバイダの仕様に従った形式にする必要があります。S3 バケットにアクセスする場合は、この引数を空のままにして、[既定の認証情報プロバイダ・チェーン](#)を使用できます。
- ・ `region` は、連携させるバケットを含むリージョンです。AWS リージョンのリストは、["Amazon Regions, Availability Zones, and Local Zones"](#) を参照してください。
- ・ 参照によって返される `tSC` は、このメソッド呼び出しで返されるステータス・コードです。
- ・ `endPoint` は、[AWS PrivateLink](#) のオプションのエンドポイントです。

5.1.1 クライアントの終了

プロバイダのバケットと BLOB の処理が終了したら、必ず `close()` メソッドを使用して作成したクライアントを閉じます。次に、例を示します。

ObjectScript

```
Do myClient.Close()
```

5.2 バケットとの連携

クラウド・ストレージ・クライアントには、プロバイダのバケット (BLOB 用のストレージ・コンテナ) と連携するように設計されている一連のメソッドが含まれています。これらのメソッドのシグニチャは以下のとおりです。

```
Method BucketExists(bucketName As %String) As %Boolean
Method GetBucketInfo(bucketName As %String) As BucketInfo
Method ListBuckets() As %ListOfObjects
Method CreateBucket(bucketName As %String)
Method DeleteBucket(bucketName As %String)
```

例えば、バケットに関する詳細を取得するため、ストレージ・クライアントを作成するには、以下を入力します。

ObjectScript

```
Set bucketName = "s3-bucket"
Set myClient = ##class(%Net.Cloud.Storage.Client).CreateClient(,0,
    "/home/AWSCredentials", "us-east-1", .tSC)
Set bucketDetails = myClient.GetBucketInfo(bucketName)
Do myClient.Close()
```

5.2.1 バケットの詳細

クラウド・ストレージ・クライアントは、`%Net.Cloud.Storage.BucketInfo` オブジェクトを使用して、バケットに関する詳細を表します。`GetBucketInfo()` を呼び出すと、指定したバケットの詳細が、`%Net.Cloud.Storage.BucketInfo` オブジェクトのインスタンスに返されます。同様に、`ListBuckets()` を呼び出すと、これらのオブジェクトのコレクション内の使用可能なすべてのバケットが返され、ユーザは各バケットの詳細にアクセスできます。使用可能なバケットの詳細については、`%Net.Cloud.Storage.BucketInfo` のプロパティを参照してください。

便宜上、`%Net.Cloud.Storage.BucketInfo` クラスには、バケット詳細を JSON 形式にできるメソッド `toJSON()` が含まれています。

5.3 BLOB 情報の取得

クラウド・ストレージ・クライアントは、次のメソッドを使用して、特定のバケット内の BLOB に関する情報を取得します。

```
Method BlobExists(bucketName As %String, blobName As %String) As %Boolean
Method GetBlobInfo(bucketName As %String, blobName As %String) As BlobInfo
Method ListBlobs(bucketName As %String) As %ListOfObjects
```

クライアントは、[BLOB のコンテンツをダウンロードする](#)には、別のメソッドを指定します。

例えば、サイズなどの特定の BLOB に関する詳細を取得する場合、以下を入力します。

```
Set bucketName = "s3-bucket"
Set blobName = "s3-object-blob"
Set myClient = ##class(%Net.Cloud.Storage.Client).CreateClient(,0,"/home/AWSCredentials", "us-east-1",
    .tSC)
Set blobDetails = myClient.GetBlobInfo(bucketName, blobName)
Do myClient.Close()
```

5.3.1 BLOB の詳細

クラウド・ストレージ・クライアントは、`%Net.Cloud.Storage.BlobInfo` オブジェクトを使用して、BLOB に関する詳細を表します。GetBlobInfo() を呼び出すと、指定した BLOB の詳細が、`%Net.Cloud.Storage.BlobInfo` オブジェクトのインスタンスに返されます。同様に、ListBlobs() を呼び出すと、これらのオブジェクトのコレクション内の使用可能なすべての BLOB が返され、ユーザは各 BLOB の詳細にアクセスできます。使用可能な BLOB の詳細については、`%Net.Cloud.Storage.BlobInfo` のプロパティを参照してください。

便宜上、`%Net.Cloud.Storage.BlobInfo` クラスには、BLOB の詳細を JSON 形式にできるメソッド toJSON() が含まれています。

5.4 BLOB のアップロード

クラウド・ストレージ API により、InterSystems IRIS® からクラウド・ストレージにデータやファイルをアップロードできます。BLOB データのソースに応じて、以下の任意のメソッドを使用し、BLOB をクラウド・ストレージ・プロバイダにアップロードできます。

```
Method UploadBlobFromString(bucketName As %String, blobName As %String, content As %String)
Method UploadBlobFromFile(bucketName As %String, blobName As %String, filePath As %String)
Method UploadBlobFromStream(bucketName As %String, blobName As %String, stream As %GlobalBinaryStream)
```

例えば、ファイルを S3 バケットにアップロードするには、以下を含めます。

ObjectScript

```
Set bucketName = "s3-bucket"
Set blobName = "s3-object-blob"
Set myClient = ##class(%Net.Cloud.Storage.Client).CreateClient(,0,
    "/home/AWSCredentials", "us-east-1", .tSC)
Do myClient.UploadBlobFromFile(bucketName, blobName, "/usr/file.jpg")
Do myClient.Close()
```

5.5 BLOB のダウンロード

InterSystems IRIS で使用するため、クラウド・ストレージ API を使用して、クラウド・ストレージ・プロバイダからデータをダウンロードできます。さまざまなメソッドが利用可能で、データのターゲット・フォーマットを選択できます。

```
Method DownloadBlobToString(bucketName As %String, blobName As %String) As %String
Method DownloadBlobToFile(bucketName As %String, blobName As %String, filePath As %String)
Method DownloadBlobToStream(bucketName As %String, blobName As %String) As %GlobalBinaryStream
```

例えば、S3 バケットから BLOB をダウンロードし、ストリームに格納するには、以下を入力します。

ObjectScript

```
Set bucketName = "s3-bucket"
Set blobName = "s3-object-blob"
Set myClient = ##class(%Net.Cloud.Storage.Client).CreateClient(,0,
    "/home/AWSCredentials", "us-east-1", .tSC)
Set IRISStream = myClient.DownloadBlobToStream(bucketName, blobName)
Do myClient.Close()
```

5.6 BLOB をアップロードするための単一のメソッド

クラウド・ストレージ API により、相互運用プロダクションを使用せずに、クラウド・ストレージにデータやファイルをアップロードできます。これらのクラス・メソッドにより、クライアントを作成し、BLOB をアップロードして、クライアントを閉じる単一の呼び出しを行うことができます。

- SingleUploadBlobFromFile
- SingleUploadBlobFromStream
- SingleUploadBlobFromString

例えば、ファイルを Amazon S3 にアップロードするには、以下を含めます。

ObjectScript

```
Set bucketName = "s3-bucket"
Set blobName = "s3-object-blob"
Set credentials = "/home/AWSCredentials"
Set region = "us-east-1"
Set filePath = "/usr/file.jpg"
Set status = ##class(%Net.Cloud.Storage.Client).SingleUploadBlobFromFile(, 0,
    credentials, region, bucketName, blobName, filePath)
```

5.7 BLOB をダウンロードするための単一のメソッド

InterSystems IRIS で使用するため、クラウド・ストレージ API を使用して、クラウド・ストレージ・プロバイダからデータをダウンロードできます。これらのクラス・メソッドにより、クライアントを作成し、BLOB をダウンロードして、クライアントを閉じる単一の呼び出しを行うことができます。

- SingleDownloadBlobToFile
- SingleDownloadBlobToStream
- SingleDownloadBlobToString

ObjectScript

```
Set bucketName = "s3-bucket"
Set blobName = "s3-object-blob"
Set credentials = "/home/AWSCredentials"
Set region = "us-east-1"
Set status = ##class(%Net.Cloud.Storage.Client).SingleDownloadBlobToStream(, 0,
    credentials, region, bucketName, blobName)
```

5.8 BLOB の削除

他のクラウド・ストレージ API と同様、クラウド・ストレージから BLOB を削除する方法は単純です。必要なのは、削除する BLOB の名前のみ（格納されているバケットを含む）です。

Method DeleteBlob(bucketName As %String, blobName As %String)

例えば、S3 バケットから BLOB を削除するには、以下を入力します。

ObjectScript

```
Set bucketName = "s3-bucket"
Set blobName = "s3-object-blob"
Set myClient = ##class(%Net.Cloud.Storage.Client).CreateClient(,0,
    "/home/AWSCredentials", "us-east-1", .tSC)
Do myClient.DeleteBlob(bucketName, blobName)
Do myClient.Close()
```

