
Article

[Lorenzo Scalese](#) · Mai 12, 2022 8m de lecture

[Open Exchange](#)

Lecture des données AWS S3 sur le COVID en tant que table SQL dans IRIS

IRIS External Table est un projet Open Source de la communauté InterSystems, qui vous permet d'utiliser des fichiers stockés dans le système de fichiers local et le stockage d'objets en nuage comme AWS S3, en tant que tables SQL.

Il peut être trouvé sur Github <https://github.com/intersystems-community/IRIS-ExternalTable> Open Exchange <https://openexchange.intersystems.com/package/IRIS-External-Table> et est inclus dans InterSystems Package Manager ZPM.

Pour installer External Table depuis GitHub, utilisez :

```
git clone https://github.com/antonum/IRIS-ExternalTable.git
iris session iris
USER>set sc = ##class(%SYSTEM.OBJ).LoadDir("<path-to>/IRIS-ExternalTable/src", "ck",,1)
```

Pour installer avec ZPM Package Manager, utilisez :

```
USER>zpm "install external-table"
```

Travailler avec des fichiers locaux

Créons un fichier simple qui ressemble à ceci :

```
a1,b1
a2,b2
```

Ouvrez votre éditeur préféré et créez le fichier ou utilisez simplement une ligne de commande sous linux/mac :

```
echo '$a1,b1\na2,b2' > /tmp/test.txt
```

Dans IRIS SQL, créez une table pour représenter ce fichier :

```
create table test (col1 char(10),col2 char(10))
```

Convertissez la table pour utiliser le stockage externe :

```
CALL EXT.ConvertToExternal(  
    'test',  
    '{  
        "adapter":"EXT.LocalFile",  
        "location":"/tmp/test.txt",  
        "delimiter": ",",  
    }')
```

Et enfin - interrogez la table :

```
select * from test
```

Si tout fonctionne comme prévu, vous devriez voir un résultat comme celui-ci :

```
col1    col2  
a1    b1  
a2    b2
```

Retournez maintenant dans l'éditeur, modifiez le contenu du fichier et réexécutez la requête SQL. Ta-Da !!! Vous lisez les nouvelles valeurs de votre fichier local en SQL.

```
col1    col2  
a1    b1  
a2    b99
```

Lecture de données à partir de S3

Sur <<https://covid19-lake.s3.amazonaws.com/index.html>>, vous pouvez avoir accès à des données constamment mises à jour sur le COVID, stockées par AWS dans la réserve publique de données.

Essayons d'accéder à l'une des sources de données de cette réserve de données : s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states

Si vous avez installé l'outil de ligne de commande AWS, vous pouvez répéter les étapes ci-dessous. Sinon, passez directement à la partie SQL. Vous n'avez pas besoin d'installer quoi que ce soit de spécifique à AWS sur votre machine pour suivre la partie SQL.

```
$ aws s3 ls s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states/  
2020-12-04 17:19:10      510572 us-states.csv
```

```
$ aws s3 cp s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states/us-  
states.csv .  
download: s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states/us-  
states.csv to ./us-states.csv
```

```
$ head us-states.csv  
date,state,fips,cases,deaths  
2020-01-21,Washington,53,1,0  
2020-01-22,Washington,53,1,0  
2020-01-23,Washington,53,1,0  
2020-01-24,Illinois,17,1,0
```

```
2020-01-24,Washington,53,1,0
2020-01-25,California,06,1,0
2020-01-25,Illinois,17,1,0
2020-01-25,Washington,53,1,0
2020-01-26,Arizona,04,1,0
```

Nous avons donc un fichier avec une structure assez simple. Cinq champs délimités.

Pour exposer ce dossier S3 en tant que table externe - d'abord, nous devons créer une table "normal" avec la structure désirée :

```
-- create external table
create table covid_by_state (
    "date" DATE,
    "state" VARCHAR(20),
    fips INT,
    cases INT,
    deaths INT
)
```

Notez que certains noms de champs comme "Date" sont des mots réservés dans IRIS SQL et doivent être entourés de guillemets doubles.

Ensuite - nous devons convertir cette table "régulière" en table "externe", basé sur le godet AWS S3 et le type de CSV.

```
-- convertissez le tableau en stockage externe
call EXT.ConvertToExternal(
    'covid_by_state',
    '{
        "adapter": "EXT.AWSS3",
        "location": "s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states/",
        "type": "csv",
        "delimiter": ",",
        "skipHeaders": 1
    }'
)
```

Si vous regardez de près - les arguments des procédures EXT.ExternalTable sont le nom de la table et ensuite une chaîne JSON, contenant plusieurs paramètres, tels que l'emplacement pour rechercher les fichiers, l'adaptateur à utiliser, le délimiteur, etc. En plus de AWS S3 External Table supporte le stockage Azure BLOB, Google Cloud Buckets et le système de fichiers local. GitHub Repo contient des références pour la syntaxe et les options supportées pour tous les formats.

Et enfin - faites une requête sur le tableau :

```
-- faites une requête sur le tableau :
select top 10 * from covid_by_state order by "date" desc
```

```
[SQL]USER>>select top 10 * from covid_by_state order by "date" desc
2. select top 10 * from covid_by_state order by "date" desc
```

date	état	fips	cas	morts
2020-12-06	Alabama	01	269877	3889
2020-12-06	Alaska	02	36847	136

```
2020-12-06  Arizona 04  364276  6950
2020-12-06  Arkansas 05  170924  2660
2020-12-06  California 06  1371940  19937
2020-12-06  Colorado 08  262460  3437
2020-12-06  Connecticut 09  127715  5146
2020-12-06  Delaware 10  39912  793
2020-12-06  District of Columbia 11  23136  697
2020-12-06  Florida 12  1058066  19176
```

L'interrogation des données de la tableau distant prend naturellement plus de temps que celle de la table "IRIS natif" ou global, mais ces données sont entièrement stockées et mises à jour sur le nuage et sont introduites dans IRIS en coulisse.

Explorons quelques autres caractéristiques de la table externe.

%PATH et tables, sur la base de plusieurs fichiers

Dans notre exemple, le dossier du godet ne contient qu'un seul fichier. Le plus souvent, il contient plusieurs fichiers de la même structure, où le nom de fichier identifie soit l'horodatage, soit le numéro de périphérique, soit un autre attribut que nous voulons utiliser dans nos requêtes.

Le champ %PATH est automatiquement ajouté à chaque table externe et contient le chemin d'accès complet au fichier à partir duquel la ligne a été récupérée.

```
select top 5 %PATH, * from covid_by_state
```

```
%PATH  date      state  fips  cases  deaths
s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states/us-
states.csv  2020-01-21  Washington  53  1  0
s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states/us-
states.csv  2020-01-22  Washington  53  1  0
s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states/us-
states.csv  2020-01-23  Washington  53  1  0
s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states/us-
states.csv  2020-01-24  Illinois    17  1  0
s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states/us-
states.csv  2020-01-24  Washington  53  1  0
```

Vous pouvez utiliser ce champ %PATH dans vos requêtes SQL comme n'importe quel autre champ.

Données ETL vers des " tables régulières "

Si votre tâche consiste à charger des données de S3 dans une table IRIS - vous pouvez utiliser l'External Table comme un outil ETL. Il suffit de le faire :

```
INSERT INTO internal_table SELECT * FROM external_table
```

Dans notre cas, si nous voulons copier les données COVID de S3 dans la table local :

```
--create local table
create table covid_by_state_local (
```

```
"date" DATE,  
"state" VARCHAR(100),  
fips INT,  
cases INT,  
deaths INT  
)  
--ETL from External to Local table  
INSERT INTO covid_by_state_local SELECT TO_DATE("date",'YYYY-MM-DD'),state,fips,cases,deaths FROM covid_by_state
```

JOIN entre IRIS tables natif et External Table. Requêtes fédérées

External Table est une table SQL. Il peut être joint à d'autres tables, utilisé dans des sous-sélections et des UNIONS. Vous pouvez même combiner la table IRIS "Régulier" et deux ou plusieurs tables externes provenant de sources différentes dans la même requête SQL.

Essayez de créer une table régulier, par exemple en faisant correspondre les noms d'État aux codes d'État, comme Washington - WA. Et joignez-la à notre table basé sur S3.

```
create table state_codes (name varchar(100), code char(2))  
insert into state_codes values ('Washington','WA')  
insert into state_codes values ('Illinois','IL')  
  
select top 10 "date", state, code, cases from covid_by_state join state_codes on state=name
```

Remplacez 'join' par 'left join' pour inclure les lignes pour lesquelles le code d'état n'est pas défini. Comme vous pouvez le constater, le résultat est une combinaison de données provenant de S3 et de votre table IRIS natif.

Accès sécurisé aux données

La réserve de données AWS Covid est publique. N'importe qui peut y lire des données sans aucune authentification ou autorisation. Dans la vie réelle, vous souhaitez accéder à vos données de manière sécurisée, afin d'empêcher les étrangers de jeter un coup d'œil à vos fichiers. Les détails complets de la gestion des identités et des accès (IAM) d'AWS n'entrent pas dans le cadre de cet article. Mais le minimum, vous devez savoir que vous avez besoin au moins de la clé d'accès au compte AWS et du secret afin d'accéder aux données privées de votre compte. <<https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html#ac...> >

AWS utilise l'authentification par clé de compte/secrète authentification pour signer les demandes.
<https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html#ac...>

Si vous exécutez IRIS External Table sur une instance EC2, la méthode recommandée pour gérer l'authentification est d'utiliser les rôles d'instance EC2 <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/iam-roles-for-amazon...> IRIS External Table sera en mesure d'utiliser les permissions de ce rôle. Aucune configuration supplémentaire n'est requise.

Sur une instance locale/non EC2, vous devez spécifier `AWSACCESSKEYID` et `AWSSECRETACCESSKEY` en spécifiant des variables d'environnement ou en installant et en configurant le client AWS CLI.

```
export AWS_ACCESS_KEY_ID=AKIAEXAMPLEKEY  
export AWS_SECRET_ACCESS_KEY=111222333abcdefghijklmnopqrstuvwxyz
```

Assurez-vous que la variable d'environnement est visible dans votre processus IRIS. Vous pouvez le vérifier en

exécutant :

```
USER>write $system.Util.GetEnviron("AWS_ACCESS_KEY_ID")
```

Il doit renvoyer la valeur de clé.

ou installer AWS CLI, en suivant les instructions ici <https://docs.aws.amazon.com/cli/latest/userguide/install-cliv2-linux.html> et exécuter :

```
aws configure
```

External Table sera alors en mesure de lire les informations d'identification à partir des fichiers de configuration aws cli. Votre shell interactif et votre processus IRIS peuvent être exécutés sous différents comptes - assurez-vous d'exécuter aws configure sous le même compte que votre processus IRIS.

[#Analytique](#) [#Cloud](#) [#CSV](#) [#Interopérabilité](#) [#SQL](#) [#InterSystems IRIS](#)
[Voir l'application sur InterSystems Open Exchange](#)

URL de la
source: <https://fr.community.intersystems.com/post/lecture-des-donn%C3%A9es-aws-s3-sur-le-covid-en-tant-que-table-sql-dans-iris>