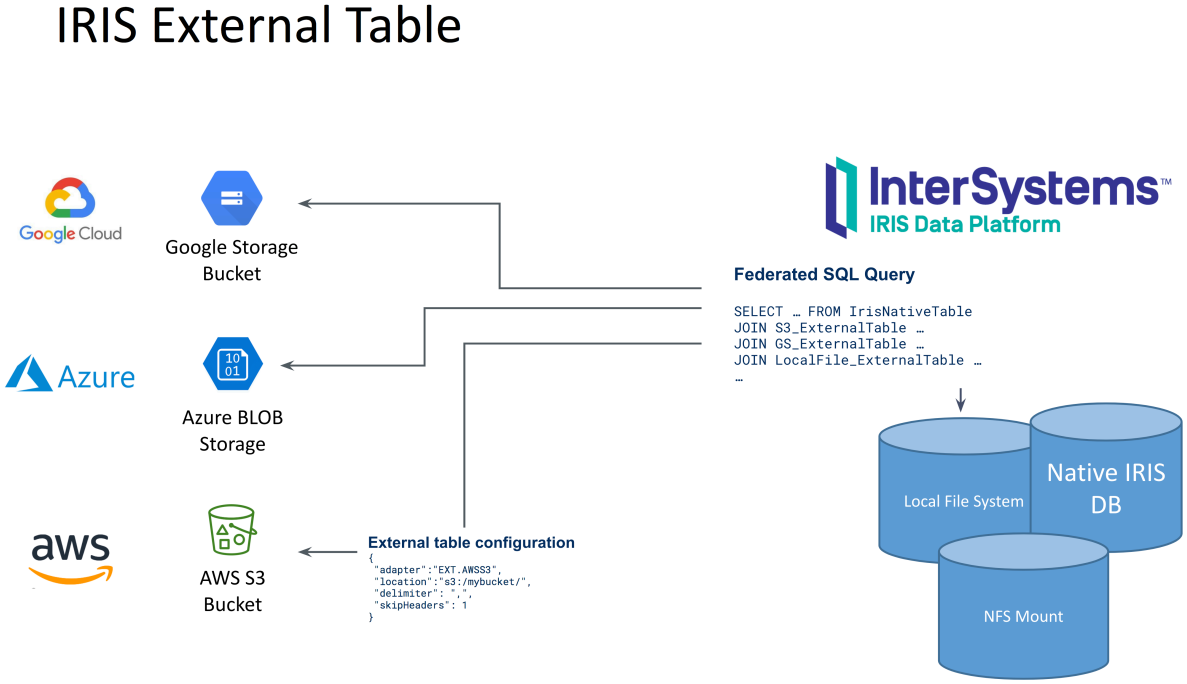
Artigo
Anton Umnikov · Jan. 11, 2021 7min de leitura

Open Exchange

Lendo dados do COVID com AWS S3 como tabela SQL no IRIS

IRIS External Table é um projeto de código aberto da comunidade InterSystems, que permite usar arquivos armazenados no sistema de arquivos local e armazenamento de objetos em nuvem, como o AWS S3, como tabelas SQL.



Ele pode ser encontrado no GitHub https://github.com/intersystems-community/IRIS-ExternalTable, Open Exchange https://openexchange.intersystems.com/package/IRIS-External-Table e está incluído no InterSystems Package Manager, ZPM.

Para instalar o External Table a partir do GitHub, use:

```
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)
```

Para instalar usando o ZPM Package Manager:

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

Trabalhando com arquivos locais

Vamos criar um arquivo simples parecido com este:

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

Abra seu editor favorito e crie o arquivo ou apenas use uma linha de comando no linux/mac:

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

No IRIS SQL, crie uma tabela para representar este arquivo:

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

Converta a tabela para usar o armazenamento externo:

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

E finalmente, consulte a tabela:

```
select * from test
```

Se tudo funcionar conforme o esperado, você verá uma saída como esta:

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

Agora volte ao editor, altere o conteúdo do arquivo e execute novamente a consulta SQL. Uau!!! Você está lendo os novos valores de seu arquivo local no SQL.

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

Lendo dados a partir do S3

Em https://covid19-lake.s3.amazonaws.com/index.html você pode obter acesso a dados atualizados constantemente sobre o COVID, armazenados pela AWS no data lake público.

Vamos tentar acessar uma das fontes de dados neste data lake: s3://covid19-lake/rearc-covid-19-nyt-data-in-usa/csv/us-states

Se você tiver a ferramenta de linha de comando AWS instalada, pode repetir as etapas abaixo. Caso contrário, vá direto para a parte SQL. Você não precisa de usar um AWS específico instalado em sua máquina para acompanhar a parte 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
```

Portanto, temos um arquivo com uma estrutura bastante simples. Com cinco campos delimitados.

Para expor esta pasta S3 como um External Table, primeiro, precisamos criar uma tabela "regular" com a estrutura desejada:

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

Observe que alguns nomes de campo como “Date” são palavras reservadas no IRIS SQL e precisam ser colocados entre aspas duplas. Em seguida, precisamos converter esta tabela “regular” para a tabela “externa”, com base no bucket AWS S3 e tipo CSV.

```
 -- convert table to external storage
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
    }'
)
```

Se você observar com atenção, os argumentos dos procedimentos EXT.ExternalTable são o nome da tabela e a string JSON, contendo vários parâmetros, como localização para procurar por arquivos, adaptador, delimitador, etc. Além da AWS S3, o External Table oferece suporte ao armazenamento BLOB do Azure, Cloud Buckets e o sistema de arquivos local. O GitHub Repo contém referências para a sintaxe e as opções suportadas em todos os formatos.

E finalmente, consulte a tabela:

```
-- query the table
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    state   fips    cases   deaths
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
```

Compreensivelmente, leva mais tempo para consultar dados da tabela remota, do que na tabela "IRIS nativa" ou com base global, porém, ela é completamente armazenada e atualizada em nuvem e está sendo puxada para o IRIS "nos bastidores".

Vamos explorar mais alguns recursos do External Table.

%PATH e tabelas, com base em vários arquivos

Em nossa pasta de exemplo, o bucket contém apenas um arquivo. Mais frequentemente, ele teria vários arquivos com a mesma estrutura, onde nome do arquivo identifica o carimbo de data/hora ou deviceid de algum outro atributo que desejaremos usar em nossas consultas.

O campo %PATH é adicionado automaticamente a cada External Table e contém o caminho completo para o arquivo de onde a linha foi recuperada.

```
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
```

Você pode usar este campo %PATH em suas consultas SQL como em qualquer outro campo.

Dados ETL para "Tabelas Regulares"

Se sua tarefa é carregar dados do S3 em uma tabela IRIS, você pode usar o External Table como uma ferramenta ETL. Apenas faça:

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

No nosso caso, se quisermos copiar os dados sobre COVID do S3 para a tabela 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 - tabela nativa e externa. Consultas federadas

External Table é uma tabela SQL. Ele pode ser unido a outras tabelas, usado em subseleções e UNIONs. Você pode até combinar a tabela IRIS "Regular" e duas ou mais tabelas externas de fontes diferentes na mesma consulta SQL.

Tente criar uma tabela regular, como os nomes de estado correspondendo com códigos de estado, como por exemplo, Washington – WA. E junte-a com nossa tabela baseada em 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 stat
e=name
```

Altere 'join' para 'left join' para incluir linhas para as quais o código de estado não foi definido. Como você pode ver, o resultado é uma combinação de dados do S3 e sua tabela IRIS nativa.

Acesso seguro aos dados

A AWS Covid Data Lake é público. Qualquer pessoa pode ler dados dele sem qualquer autenticação ou autorização. Na vida real, você desejará acessar seus dados de forma segura, evitando que estranhos espiem seus arquivos. Os detalhes completos da AWS Identity and Access Management (IAM) estão fora do escopo deste artigo. Mas o mínimo que você precisa saber é que você precisa de pelo menos a chave de acesso e a chave secreta da conta da AWS para acessar dados privados em sua conta.

AWS usa autenticação de chave/segredo de conta para assinar solicitações.

https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html#ac...

Se você estiver executando o IRIS External Table na instância EC2, a maneira recomendada de lidar com a autentica ç ão é usar as fun ç ões da instância EC2https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/iam-roles-for-amazon.... O IRIS External Table será capaz de usar as permissões dessa fun ç ão. Nenhuma configura ç ão extra é necessária.

Em uma instância local/não EC2, você precisa especificar o AWS_ACCESS_KEY_ID e AWS_SECRET_ACCESS_KEY, especificando variáveis de ambiente ou instalando e configurando o cliente AWS CLI.

```
export AWS_ACCESS_KEY_ID=AKIAEXAMPLEKEY
export AWS_SECRET_ACCESS_KEY=111222333abcdefghigklmnopqrst
```

Certifique-se de que a variável de ambiente esteja visível em seu processo IRIS. Você pode verificá-lo executando:

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

Ela deve retornar o valor da chave.

ou instale o AWS CLI, seguindo as instru ç ões aquihttps://docs.aws.amazon.com/cli/latest/userguide/install-cliv2-linux.html e execute:

```
aws configure
```

O External Table poderá, então, ler as credenciais dos arquivos de configura ç ão do aws cli. Seu shell interativo e o processo IRIS podem estar sendo executados em contas diferentes. Certifique-se de executar aws configure na mesma conta do seu processo IRIS.

#Analytics #CSV #Interoperabilidade #Nuvem #SQL #InterSystems IRIS
Confira o aplicativo relacionado no InterSystems Open Exchange

URL de origem: https://pt.community.intersystems.com/post/lendo-dados-do-covid-com-aws-s3-como-tabela-sql-no-iris