文章
姚 鑫 · 六月 21, 2022 阅读大约需 3 分钟

第六章 操作位和位串（二）

第六章 操作位和位串（二）

将位序列存储为整数

如果要将一系列布尔参数传递给方法，一种常见的方法是将它们作为编码为单个整数的位序列传递。

例如，Security.System.ExportAll() 方法用于从 IRIS
实例中导出安全设置。如果查看此方法的类引用，将看到它的定义如下：

```
classmethod ExportAll(FileName As %String = "SecurityExport.xml",
ByRef NumExported As %String, Flags As %Integer = -1) as %Status
```

第三个参数 Flags 是一个整数，其中每个位代表一种可以导出的安全记录。

```
Flags - What type of records to export to the file, -1 = ALL
Bit 0 - System
Bit 1 - Events
Bit 2 - Services
Bit 4 - Resources
Bit 5 - Roles
Bit 6 - Users
Bit 7 - Applications
Bit 8 - SSL Configs
Bit 9 - PhoneProvider
Bit 10 - X509Credential
Bit 11 - OpenAMIdentityService
Bit 12 - SQL privileges
Bit 13 - X509Users
Bit 14 - DocDBs
Bit 15 - LDAPConfig
Bit 16 - KMIPServer
```

存储为整数的位串中的位 0 表示 20，位 1 表示 2^1，依此类推。如果要导出与位 5、6、7、8、10、11 和 13
对应的类型的安全记录，可以通过将 Flags 设置为 2^5 +2^6 + 2^7+ 2^8 + 2^10 + 2^11 + 2^13 = 11744 来完成.

在 ObjectScript 中，这可能看起来像：

```
set flags = (2**5) + (2**6) + (2**7) + (2**8) + (2**10) + (2**11) + (2**13)
set status = ##class(Security.System).ExportAll("SecurityExport.xml", .numExported, f
lags)
```

一些 API 定义了宏以使代码更易于阅读。一种这样的情况是在 DataMove 实用程序中，其中 DataMove 对象是使用

DataMove.Data.CreateFromMapEdits() 方法创建的。无需过多介绍细节，该方法在类参考中定义如下：

```
classmethod CreateFromMapEdits(Name As %String, ByRef Properties As %String,
ByRef Warnings As %String, ByRef Errors As %String) as %Status
```

它有以下参数：

- Name - 要创建的 DataMove 对象的名称。
- Properties - 用于创建对象的属性数组。可以选择指定以下属性。
- Properties("Description")- 数据移动操作的描述，默认 =""。
- Properties("Flags") - 描述操作的标志，默认 = 0。
- Properties("LogFile")- 日志文件的目录和文件名，默认 = \iris\mgr\DataMovename.log。

为了使 Properties("Flags") 更容易定义，这些宏可供使用：

控制数据移动的位标志。

- $$$BitNoSrcJournal - 允许不记录源数据库
- $$$BitNoWorkerJobs - 在复制数据期间不要使用 “ worker ” 作业
- $$$BitBatchMode - 在 “ 批处理 ” 模式下运行复制作业
- $$$BitCheckActivate - 在 Activate() 期间调用 $$CheckActivate^ZDATAMOVE()

这些宏定义为特定位的计算值，允许设置正确的位，而无需记住哪个位代表哪个标志：

#;DataMove 标志属性的定义
- #define BitNoSrcJournal 1
- #define BitNoWorkerJobs 512
- #define BitBatchMode 2048
- #define BitCheckActivate 4096

在代码中，可以使用此代码片段来设置标志并创建一个 DataMove 对象：

```
// Set properties("Flags") to 6657
set properties("Flags") = $$$BitNoSrcJournal + $$$BitNoWorkerJobs + $$$BitBatchMode +
 $$$BitCheckActivate
set status = ##class(DataMove.Data).CreateFromMapEdits("dm", .properties, .warnings,
.errors)
```

#SQL #Caché

源 URL: https://cn.community.intersystems.com/post/%E7%AC%AC%E5%85%AD%E7%AB%A0-%E6%93%8D%E4%BD%9C%E4%BD%8D%E5%92%8C%E4%BD%8D%E4%B8%B2%EF%BC%88%E4%BA%8C%EF%BC%89