

文章

[姚鑫](#) · 二月 18, 2021 阅读大约需 6 分钟

## 第四十章 Caché 变量大全 \$ZREFERENCE 变量

### 第四十章 Caché 变量大全 \$ZREFERENCE 变量

包含当前全局变量global引用。

## 大纲

\$ZREFERENCE

\$ZR

## 描述

\$ZREFERENCE包含上次全局引用的名称和下标。这就是所谓裸指针。

注意：最后一个全局引用是最近访问的全局节点。通常，这是对全局的最新显式引用。但是，某些命令可能在内部使用\$ORDER函数遍历全局下标(ZWRITE命令就是一个例子)，或者它们可能在内部引用其他全局脚本。发生这种情况时，\$ZREFERENCE包含上次访问的全局节点，该节点可能不是为命令指定的全局节点。

最后一个全局引用可以是全局(^myglob)或进程专用全局(^||myppg)。\$ZREFERENCE以最初用于该变量的形式返回进程专用全局前缀，而不管随后对该变量使用哪个进程专用全局前缀。在接下来的\$ZREFERENCE描述中，单词“global”指的是这两种类型的变量。

最后一个全局引用是命令或函数最近引用的全局。由于ObjectScript按从左到右的顺序执行操作，因此最后一个全局引用始终是最右侧的全局引用。当命令或函数使用多个参数时，最右侧参数中指定的全局参数是最后一个全局引用。当参数包含多个全局引用时，最右侧指定的全局引用是最后一个全局引用。即使使用圆括号来定义操作顺序，从左到右的严格顺序也是正确的。

当发出显式全局引用时，InterSystems

IRIS会更新\$ZREFERENCE。调用计算结果为全局引用的表达式(如局部变量)不会更新\$ZREFERENCE。

\$ZREFERENCE包含最新的全局引用，即使此全局引用不成功。当命令引用未定义的全局时，会发出<unfined>错误，InterSystems IRIS会将\$ZREFERENCE更新为该全局引用，就像定义了全局一样。此行为不受设置%SYSTEM.Process.Unfined()方法的影响。

\$ZREFERENCE通常包含最新的全局引用，即使命令执行不成功。InterSystems IRIS在引用每个全局变量时更新\$ZREFERENCE。例如，发出<Divide>错误(试图将数字除以0)的命令会将\$ZREFERENCE更新为错误发生前命令中引用的最后一个全局变量。但是，<SYNTAX>错误不会更新\$ZREFERENCE。

## 长全局变量名称

如果全局名称超过31个字符(不包括全局前缀字符，如^)，\$ZREFERENCE将返回缩短为31个字符的全局名称。

## 裸全局变量引用

如果上一个全局引用是裸全局引用，则\$ZREFERENCE包含当前裸全局引用的外部、可读的完整形式。下面的示例演示了这一点：

```
/// d ##class(PHA.TEST.SpecialVariables).ZREFERENCE()  
ClassMethod ZREFERENCE()  
{  
    SET ^MyData(1)="fruit"  
    SET ^MyData(1,1)="apples" ; ??????????  
    SET ^{2)="oranges" ; ????????, ?? ^MyData(1,2)  
    WRITE !,$ZREFERENCE ; Returns "^MyData(1,2)"  
}
```

```
DHC-APP>d ##class(PHA.TEST.SpecialVariables).ZREFERENCE()
```

```
^MyData(1,2)
```

### 扩展局变量引用

扩展局变量引用用于引用当前命名空间以外的命名空间中的全局。如果命令使用扩展全局引用引用全局变量，则\$ZREFERENCE值将包含该扩展全局引用。在以下情况下，InterSystems IRIS返回扩展的全局引用：

- 如果最后一个全局引用使用扩展引用来引用另一个命名空间中的全局。
- 如果最后一个全局引用使用扩展引用来引用当前命名空间中的全局。
- 如果最后一个全局引用是远程引用(远程系统上的全局引用)。

在所有情况下，\$ZREFERENCE都以全部大写字母返回命名空间名称，而不管它在全局引用中是如何指定的。

### 更新\$ZREFERENCE的操作

\$ZREFERENCE特殊变量被初始化为空字符串(“ ”)。更改当前名称空间会将\$ZREFERENCE重置为空字符串。

以下操作将\$ZREFERENCE设置为最近引用的GLOBAL：

- 使用全局变量作为参数的命令或函数。如果它使用多个全局变量，则\$ZREFERENCE被设置为全局变量的最右侧匹配项。(但请注意，\$ORDER除外。)
- 使用全局作为后置条件表达式的命令。
- 在ZWRITE之后，InterSystems IRIS将\$ZREFERENCE设置为指定全局引用的上次访问的下标节点。
- 引用未定义的全局变量的命令或函数，它或者生成<unfined>错误，或者在\$INCREMENT的情况下定义全局变量。

### 设置\$ZREFERENCE

可以使用set命令设置此特殊变量，如下所示：

- 设置为空字符串(“ ”)。这样做会删除裸指示器。如果下一个全局引用是裸全局引用，则InterSystems IRIS会发出<naked>错误。
- 设置为有效的全局参照(已定义或未定义)。这会导致后续的裸引用使用设置的值，就好像它是最后一个实际的全局引用一样。

不能使用SET命令以其他方式修改\$ZREFERENCE。尝试这样做会导致<SYNTAX>错误。

### 示例

下面的示例返回最后一个全局引用：

```
/// d ##class(PHA.TEST.SpecialVariables).ZREFERENCE1()  
ClassMethod ZREFERENCE1()  
{  
    SET ^a(1,1)="Hello" ; ??????????  
    SET ^a(2)=" world!" ; ??????????  
    WRITE $ZREFERENCE  
}
```

```
DHC-APP>d ##class(PHA.TEST.SpecialVariables).ZREFERENCE1()  
^a(1,2)
```

下面的示例从几个不同的命令返回全局引用。请注意，WRITE和ZWRITE设置同一全局引用的不同表示形式。

```
/// d ##class(PHA.TEST.SpecialVariables).ZREFERENCE2()  
ClassMethod ZREFERENCE2()  
{  
    SET (^barney,^betty,^wilma,^fred)="flintstone"  
    WRITE !,"1: "_$ZREFERENCE  
    KILL ^flies  
    WRITE !,"2: "_$ZREFERENCE  
    WRITE !,^fred  
    WRITE !,"3: "_$ZREFERENCE,!  
    ZWRITE ^fred  
    WRITE !,"4: "_$ZREFERENCE  
}
```

```
DHC-APP>d ##class(PHA.TEST.SpecialVariables).ZREFERENCE2()  
  
1:^fred // ??????????????????????????  
2:^flies // KILL?????????????????????  
flintstone  
3:^fred  
^fred="flintstone"  
  
4:^fred("")
```

下面的示例返回扩展的全局引用。请注意，名称空间名称始终以大写字母返回：

```
/// d ##class(PHA.TEST.SpecialVariables).ZREFERENCE3()  
ClassMethod ZREFERENCE3()  
{  
    SET ^["samples"]a(1,1)="Hello"  
    SET ^a(2)=" world!"  
    WRITE $ZREFERENCE  
    QUIT  
}
```

```
DHC-APP>d ##class(PHA.TEST.SpecialVariables).ZREFERENCE3()  
^[ "SAMPLES" ]a(1,2)
```

下面的示例返回最后一个全局引用。在本例中，它是`^a(1)`，用作`$LENGTH`函数的参数：

```
/// d ##class(PHA.TEST.SpecialVariables).ZREFERENCE4()  
ClassMethod ZREFERENCE4()  
{  
    SET ^a(1)="abcdefghijklmnopqrstuvwxyz"  
    SET ^b(1)="1234567890"  
    SET x=$LENGTH(^a(1))  
    WRITE $ZREFERENCE  
    QUIT  
}
```

```
DHC-APP>d ##class(PHA.TEST.SpecialVariables).ZREFERENCE4()  
^a(1)
```

下面的示例返回为`$ZREFERENCE`设置的值，就好像它是最后一个全局引用一样。在本例中，它是`^a(1,1)`。

```
/// d ##class(PHA.TEST.SpecialVariables).ZREFERENCE5()  
ClassMethod ZREFERENCE5()  
{  
    SET ^a(1,1)="abcdefghijklmnopqrstuvwxyz"  
    SET ^b(1,1)="1234567890"  
    WRITE !,^a(1) ; ??????????????????????  
    SET $ZREFERENCE="^a(1,1)"  
    WRITE !,$ZREFERENCE  
    WRITE !,^a(1) ; ????????$ZREFERENCE?????????????????.  
}
```

```
DHC-APP>d ##class(PHA.TEST.SpecialVariables).ZREFERENCE5()  
  
1234567890  
^a(1,1)  
abcdefghijklmnopqrstuvwxyz
```

下面的示例设置扩展的全局引用。请注意双引号：

```
/// d ##class(PHA.TEST.SpecialVariables).ZREFERENCE6()  
ClassMethod ZREFERENCE6()  
{  
    KILL ^x  
    WRITE !,$ZREFERENCE  
    SET $ZREFERENCE="^[ "samples" ]a(1,2)"  
    WRITE !,$ZREFERENCE  
}
```

```
DHC-APP>d ##class(PHA.TEST.SpecialVariables).ZREFERENCE6()
```

^x

^[ "SAMPLES" ]a(1,2)

[#Caché #InterSystems IRIS #InterSystems IRIS for Health](#)

---

源

URL:

<https://cn.community.intersystems.com/post/%E7%AC%AC%E5%9B%9B%E5%8D%81%E7%AB%A0-cach%C3%A9-%E5%8F%98%E9%87%8F%E5%A4%A7%E5%85%A8-zreference-%E5%8F%98%E9%87%8F>