

Artículo

[Alberto Fuentes](#) · 25 ago, 2022 · Lectura de 4 min

Recoger datos de rendimiento mientras se ejecutan pruebas unitarias

Hace varios años, estaba enseñando los conocimientos básicos de nuestro *framework* %UnitTest durante la clase de Fundamentos de Caché (ahora llamada [Developing Using InterSystems Objects and SQL](#)). Un alumno preguntó si era posible recoger estadísticas de rendimiento mientras se ejecutan pruebas unitarias. Unas semanas más tarde, añadí un código adicional a los ejemplos de %UnitTest para responder a esa pregunta. Ahora lo comparto con la Comunidad.

La clase %SYSTEM.Process ofrece varias métricas que puedes recoger para un proceso (además de Duration).

- Duration
- Lines Executed
- Global References
- System CPU Time
- User CPU Time
- Disk Read Time

Para permitir a **cualquier** prueba unitaria recoger esas estadísticas, crea una subclase de %UnitTest.TestCase y añade propiedades.

```
Class Performance.TestCase Extends %UnitTest.TestCase
{
Property Duration As %Time;
Property Lines As %Integer;
Property Globals As %Integer;
Property SystemCPUTime As %Integer;
Property UserCPUTime As %Integer;
Property DiskReadTime As %Integer;
}
```

Cualquier clase de unit test que crees debería heredar de tu nueva subclase en vez de %UnitTest.TestCase.

En la subclase, usa OnBeforeOneTest() para iniciar la recogida de estadísticas para cada prueba unitaria. Para todo excepto DiskReadTime, el código inicia la propiedad con el valor actual.

```
/// initialize performance stats
Method OnBeforeOneTest(testname As %String) As %Status
{
    // initialize with current values
    set ..Duration = $zh
    set ..Lines = $system.Process.LinesExecuted()
    set ..Globals = $system.Process.GlobalReferences()
    set ..SystemCPUTime = $piece(CPUTime, ",", 1)
    set ..UserCPUTime = $piece(CPUTime, ",", 2)
    // reset disk read time to 0 and start counting
    do $system.Process.ResetDiskReadTiming()
    do $system.Process.EnableDiskReadTiming()
    return $$$OK
}
```

Usa `OnAfterOneTest()` para terminar la recogida de estadísticas para cada prueba unitaria. Para todo excepto `DiskReadTime`, el código elimina el valor inicial desde el valor actual.

```
/// Finalize performance stats
/// This is where you could also add code to save the counters to a separate table for analysis.
Method OnAfterOneTest(testname As %String) As %Status
{
    set ..Duration = $zh - ..Duration
    set ..Lines = $system.Process.LinesExecuted() - ..Lines
    set ..Globals = $system.Process.GlobalReferences() - ..Globals
    set CPUTime = $system.Process.GetCPUTime()
    set ..SystemCPUTime = $piece(CPUTime, ",", 1) - ..SystemCPUTime
    set ..UserCPUTime = $piece(CPUTime, ",", 2) - ..UserCPUTime
    // get disk read time and stop counting
    set ..DiskReadTime = $system.Process.DiskReadMilliseconds()
    do $system.Process.DisableDiskReadTiming()
    // add message to unit test log
    set msg = "Performance: " _ "Duration: " _ ..Duration _
    ", Lines: " _ ..Lines _
    ", Globals: " _ ..Globals _
    ", System CPU Time: " _ (..SystemCPUTime / 1000) _
    ", User CPU Time: " _ (..UserCPUTime / 1000) _
    ", Disk Read Time: " _ (..DiskReadTime / 1000) _
    do $$$LogMessage(msg)
    return $$$OK
}
```

Hay otro pequeño truco. Puede que quieras ejecutar tus pruebas unitarias **con o sin** estadísticas de recogida. Así que, el código en el que estás invocando tus pruebas unitarias debe tomar un argumento (podría ser un %Boolean 1 o 0) y de alguna manera pasarlo. Los métodos que realmente ejecutan las pruebas (como `RunTest()` u otro de los métodos `Run*`()) toman un array como el 3er argumento, pasado por referencia. Este es un ejemplo:

```
// create an array to hold the logging argument (1 or 0) and pass it by reference
set p("logging") = logging
do ##class(%UnitTest.Manager).RunTest(test, qualifiers, .p)
```

Al valor que pasas en la matriz puede accederse en `OnBeforeOneTest()` y `OnAfterOneTest()`. Añade esto como la primera línea en ambos métodos:

```
if (..Manager.UserFields.GetAt("logging") = 0) { return $$$OK }
```

¡Y ya está!

Espero vuestras preguntas, comentarios e ideas.

[#Code Snippet](#) [#Mejores prácticas](#) [#Prueba](#) [#Rendimiento](#) [#Caché](#) [#InterSystems IRIS](#)

URL de

fuentes: <https://es.community.intersystems.com/post/recoger-datos-de-rendimiento-mientras-se-ejecutan-pruebas-unitarias>