

aParams

Esta propiedad la tienen todos los controles que puedan ejecutar una llamada tanto a nuestro CGI como a cualquier otro webservice.

Su utilidad es poder añadir a la llamada que se realice información extra para que la reciba el webservice llamado, la información se enviara siempre por POST por pares de Variable y Dato.

aParam es un array de pares

```
:aParams := {"variable1","dato1"}, {"variable2","dato2"}}
```

como pueden ver siempre son cadenas, en caso de que algún dato no sea cadena, Néfele lo convertirá a cadena antes de montar el HTML final.

Tenemos la posibilidad de hacer que entre los datos a devolver nos venga la propiedad de algún control en el momento del envío, para eso como dato pasaremos # y el `cid` del control

```
:AddParam({"variable1", "#nombre:value"})
```

si obviamos la propiedad (:value), por defecto nos retorna **:value**, pero dependiendo del control es posible que necesitemos que nos retorne otra propiedad, como por ejemplo un `wCheckBox`, en este caso lo lógico sería pedir la propiedad **:checked**, ya que si pedimos **:value** siempre nos retornara el valor de la propiedad **cChecked** del control.

Para los `wAttachFile` habrá que pedirle la propiedad **:inputfiles**, este control no tiene por defecto una propiedad que podamos pedirle, así que hemos creado esta que incluirá todo el código JS necesario para recuperar el/los ficheros seleccionados.

También podemos hacer que nos devuelva el resultado de una función o variable JS que lógicamente deberá existir en la página o ser genérica de JS. Para ello utilizaremos **'js:'** y el nombre de la función o variable.

```
:AddParam({"esmovil", "js:IsMovil"})
```

En el caso de que la función o variable JS retorne un Hash deberemos utilizar el indicador **'ap:'** para que Néfele sepa que tiene que concatenar ese Hash con el resto de aParams.

```
:AddParam({"mibrowse", "ap:dt_getValues('browse')"})
```

Podemos también recuperar datos de `LocalStorage` y `SessionStorage` mediante las etiquetas **'ls:'** y **'ss:'** y el nombre del campo.

```
:AddParam({"midata", "ls:midata"})
```

Desde la versión **0.5** podremos recuperar el orden de los `wItemListView` dentro de un `wListView` solicitando el valor de su propiedad **orderlist** y en el caso del `wArrayBrowse` le pediremos el valor de su propiedad **orderarray**, en ambos casos nos retorna un array con los **nrowid** en el orden actual.

Hay que tener en cuenta que nuestro **CGI** al recuperar esta información siempre nos retorna cadenas.

Para recuperar estos parámetros utilizaremos `oCGI:GetValue(cParametro, cDefault)`, con lo que nos devolverá el valor del parámetro indicado (`cParametro`) y en caso de no existir nos devolverá `cDefault`.

Si sabemos que el parámetro en cuestión es un Array podemos utilizar `oCGI:GetArray` para recuperarlo.

Peculiaridades a tener en cuenta

En el control `wForm` los `aParams` se envían al hacer el submit, y estos se añadirán a los valores de los controles input (`wEdit`, `wDatePicker`, `wTimePicker`, etc..) que tengamos dentro del `wForm`, por lo que no es necesario incluirlos en `aParams`.

En los controles que a su vez contienen controles con evento, por ejemplo los `wListView` y sus `wItemListView`, podemos definir `aParams` genéricos que se enviarán en todas las llamadas y `aParams` específicos para cada ítem que se añadirán a los genéricos en el momento de la llamada. Para facilitar el trabajo estos ítem ya enviarán datos que nos pueden ser útiles en nuestro CGI como es el `clid` del control que realizó la llamada, el evento y el `nRowid`, si lo tiene, y que no debemos confundir con el `clid`

El `nRowid` es el id del registro en la BBDD (o el id que le pasáramos en el momento de crear el control y que nos identifica de manera inequívoca el registro sobre el que queremos actuar) y el `clid` es el identificador de cada control dentro de la página.

From:

<https://nefele.dev/wiki/> - **Nefele Project**

Permanent link:

<https://nefele.dev/wiki/propiedades/aparams?rev=1742755408>

Last update: **23/03/2025 18:43**

