
Question

[Timothy Leavitt](#) · Dec 11, 2017

Validating XML

Problem:

I want to be able to import XML (say, using [%XML.Reader](#)) to Caché objects permissively (ignoring invalid/unexpected tags/attributes), but also to validate the same XML and list any/all invalid tags/attributes that it contains.

Possible path to a solution:

XML-enabled classes (see [documentation](#)) inherit [XMLIGNOREINVALIDTAG](#) and [XMLIGNOREINVALIDATTRIBUTE](#) parameters that can customize the treatment of unexpected tags/attributes in XML import. One heavy-handed option would be to have parallel sets of classes representing the same XML document, one with both parameters set to 0 and one with both parameter set to 1. (And different XML namespaces, of course.) But this seems like a lot of work, and I strongly suspect that %XML.Reader would give up at the first error, which doesn't help the user much if there are lots of problems in the XML document. Ideally there would be some runtime option to make a list of the times XMLIGNOREINVALIDTAG=1/XMLIGNOREINVALIDATTRIBUTE=1 made a difference.

Any thoughts/recommendations? Has anyone done this before in Caché?

[#Object Data Model](#) [#XML](#) [#Caché](#)

Source URL: <https://community.intersystems.com/post/validating-xml>