> OK, I'm lost again. :-(

You should have turned left at Alberquerque.

> I cannot figure out how to pull out the information from
> the CAMPAIGN_ID element for example:
>
>   name: CAMPAIGN_ID
>   value: 00
>   units: n/a
>
> Without knowing in my code that the element CAMPAIGN_ID exists.
>
> I guess my question is this: How can I find all the sub-elements
> of CONFIGDATA in a generic way? (I can find ALL the elements in the
> file, but what I want is the first sub-elements of CONFIGDATA, then
> all of *its* sub-elements, etc.) I find the documentation uh, vague. :-(

So, if I understand your question, you want to walk the node tree
without prior knowledge of what nodes you may encounter on your journey.
   The only way I know to do this is with good ol' recursion. I could
explain the recursion, but an example is much better than an explanation
in this case. In the vague IDL documentation there's a pretty good
example of tree-walking. So, I guess the question is, what about this
doesn't do what you want it to?

 From the IDL Documentation...

The following code traverses a DOM tree using pre-order traversal.

```
PRO sample_recurse, oNode, indent

  ; "Visit" the node by printing its name and value
  PRINT, indent GT 0 ? STRJOIN(REPLICATE(' ', indent)) : '', $
    oNode->GetNodeName(), ':', oNode->GetNodeValue()

  ; Visit children
  oSibling = oNode->GetFirstChild()
  WHILE OBJ_VALID(oSibling) DO BEGIN
    SAMPLE_RECURSE, oSibling, indent+3
    oSibling = oSibling->GetNextSibling()
  ENDWHILE
```

END

```
PRO sample
   oDoc = OBJ_NEW('IDLffXMLDOMDocument')
   oDoc->Load, FILENAME="sample.xml"
   SAMPLE_RECURSE, oDoc, 0
   OBJ_DESTROY, oDoc
END
```

-Mike