get_modifier_children

The **get_modifier_children** message returns all the children of a particular modifier. A client may want a list of all children in order to expand a modifier of the vocabulary tree when a user is browsing through the tree.

Populating Children of Tree Nodes That are Modifiers

The **get_modifier_children** message is used to populate tree nodes that are modifiers within the Ontology Navigate Terms tool. The table / path (root) to search are known.
The sequence of events is as follows:

1. The client sends a message with the following settings:

max = 200 (or higher)
type = core

1. ONT server performs the following steps:
    a. Parses *<parent>* to obtain the table key and path. Queries the table of tables to confirm that the user / role can access the table that is referenced by the table key passed in. If not, return coded error.
    b. If max is set, the database is queried for that number of children associated with the parent and *<applied_path>* that is passed in.
    c. If count < max or no max set, the database is queried for the entire list of children that meets the parent criteria.
    d. If count > max a coded error message is sent back.

1. If no errors, the client receives a list of children and populates the tree.
2. If max exceeded, the client receives an error message with the code "MAX_EXCEEDED". A dialog box is displayed to ask if the user wants to see all nodes. If no – done. If yes – client sends another message with max=empty.

get_modifier_children Request Message

A **get_modifier_children** message implies that the user is passing a key / path for a parent and wants the children returned. The parent tag will tell the service what metadata table / path to search in and for the **get_modifier_children** message must be specified. The structure of a parent is organized as follows:
\\table_key\path
The key (i2b2) plus the path (\Severe) is the parent:
<parent>\\i2b2\Severe{color:#0000ff}</parent>

The *<applied_path>* tag is used to further identify the modifier we are requesting children for. Finally, the *<applied_concept>* tag identifies the concept (key \ path pair) we are obtaining modifier children for. This information is needed in order to properly exclude potential modifier children from the concept we are ultimately linking this modifier to. Both the *<applied_path>* tag and the *<applied_concept>* tag must be included.
The remaining attributes provide information about the results to be returned. If the number of rows found is greater than max, then an error message will returned in the i2b2 header. If max is left out then it is interpreted that there is no max. The hiddens and synonyms attributes tell whether to return hiddens and synonyms. By default hiddens and synonyms are false, so if they are left out it will be false. The type tells which columns to select (default / core / all). By default, type is set to default so you don't have to actually include it if you want the default set of columns returned. Each message will interpret "default" to be a different set of columns. **get_modifier_children's** default set of columns is set to all columns except the blob and the system / date information. If *type = core*, then all columns except the blob and the system / date information will be returned (same as "default" in this case). If *type = all*, then all columns except the blob are returned. The blob attribute indicates whether or not to return the blob along with the default / core / all return columns.
Example:
<message_body>
<get_modifier_children max="200" hiddens="true" synonyms="true" type="default" blob="false">
<parent>\\i2b2\Severe{color:#0000ff}</parent>
<applied_path>\i2b2\Diagnoses%</applied_path>
<applied_concept>\\i2b2\i2b2\Diagnoses\Mental Disorders (290- 319){color:#0000ff}</applied_concept>
</get_modifier_children>
</message_body>

Possible "hiddens" Settings

Some modifiers exist but for various reasons are not displayed in the query tree.

Value	Description
false	Do not return data categorized as "hidden"
true	Include data categorized as "hidden"

Possible "synonyms" Settings

Some modifiers are listed as synonyms for other modifiers.

Value	Description
false	Do not return data categorized as "synonym"
true	Include data categorized as "synonym"

Possible "type" Settings

Value	Description
default	Return all data except system / date information
core	Return all data except system / date information (same as default)
all	Return all data

Possible "blob" Settings

Value	Description	Example
false	Do not return data stored as a blob or clob	xml, comments
true	Return xml and comments	

get_modifier_children Response Message

The request has the following settings:
type=default
blob=false

Response Message:

```xml
<message_body>
<modifiers>
<modifier>
<level>2</level>
<applied_path>\i2b2\Diagnoses%</applied_path>
<key>\\i2b2\Severe\Lethal{color:#0000ff}</key>
<fullname>\Severe\Lethal{color:#0000ff}</fullname>
<name>Lethal</name>
<visualattributes>RA </visualattributes>
<synonym_cd>N</synonym_cd>
<totalnum xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true" />
<basecode>lethal</basecode>
<tooltip>Severe \ Lethal</tooltip>
</modifier>
<modifier>
<level>2</level>
<applied_path>\i2b2\Diagnoses%</applied_path>
<key>\\i2b2\Severe\Type I hypersensitivity{color:#0000ff}</key>
<fullname>\Severe\Type I hypersensitivity{color:#0000ff}</fullname>
<name>Type I hypersensitivity</name>
<visualattributes>RA </visualattributes>
<synonym_cd>N</synonym_cd>
<totalnum xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:nil="true" />
<basecode>typeIhyper</basecode>
<tooltip>Severe \ Type I hypersensitivity</tooltip>
</modifier>
</modifiers>
</message_body>
```