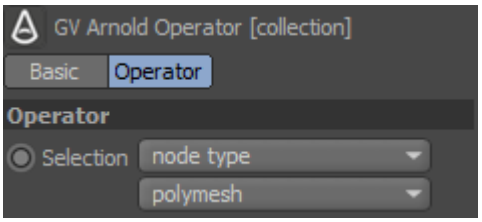
Collection



This creates a selection expression that can be referred to by other operator nodes in the operator graph.

Selection

An expression to filter the nodes that will be part of the collection. The expression syntax is described in the selection expression documentation, with some examples. Note that if the *operator* is connected to a *procedural* the selections are assumed to be relative to the procedural's namespace.

Collection

The name that is used to reference this collection from other selections and collections.

Selection expressions can reference collections which have been defined in the graph using a collection operator. A collection is referenced using a # symbol as a prefix to the collection name, e.g. *#my_collection*. Collections have some nice properties such as:

- Collections can themselves reference one or more collections.
- Collections form a single operand in a selection expression so they can be included as part of a bigger expression.
- Collections can be used to both include and exclude nodes from selections.

The combination of these properties enables building collections from other collections, including exclusion collections to remove parts of other collections as shown in the following example.

```
collection
{
  selection "..."
  collection "big_collection"
}

collection
{
  selection "#big_collection or /scene/chars/*.(role == dowser)"
  collection "custom_collection"
}

collection
{
  selection "*/trees/*acacia*.(look_variant == look2)"
  collection "exclude_selection"
}

# Here we create a smaller collection using an exclusion set
collection
{
  selection "#custom_collection and not #exclude_selection"
  collection "smaller_collection"
}

# Here we increase the scale of the shapes in the smaller collection
transform_shape
{
  selection "#smaller_collection"
  scale 2.0 2.0 1.5
  order "srt" # scale rotate translate
}
```