Appendix D. Query Functions

When performing a series of operations in TGrid, you may want to list all the zones of a particular type, group, or containing a particular text string. TGrid allows you to query zones through the TUI based on a regular expression and a specified variable. You can identify face/cell zones closest to a specified point using the query functions. You can also use the query functions to determine the zones created during a particular operation by performing Boolean operations on lists returned by specific query functions.

You can use the `eval-expr` function to evaluate the lists returned by the query functions in order that they may be used as input in the text user interface commands. Refer to Section D.2: Examples for examples on the use of query functions.

The query functions available in TGrid are described in the following table:

Query Function	**Description**	**Examples**
`(get-face-zone-at-location '(x y z))`	Returns the face zone at or closest to the specified location	—
`(get-cell-zone-at-location '(x y z))`	Returns the cell zone at or closest to the specified location	—
`(get-zones-of-type 'type)`	Returns a list of zones of the specified default face zone type	`(get-zones-of-type 'symmetry)`
	Returns a list of zones of the specified default edge zone type	`(get-zones-of-type 'boundary-edge)`
	Returns a list of zones of the specified default cell zone type	`(get-zones-of-type 'fluid)`
	Returns a list of zones of the specified default node zone type	`(get-zones-of-type 'boundary-node)`

Query Function	Description	Examples
(get-zones-of-group ’group)	Returns a list of the face zones in the specified default face zone group	(get-zones-of-group ’geometry)
	Returns a list of the edge zones in the specified default edge zone group	(get-zones-of-group ’boundary-edge)
	Returns a list of the cell zones in the specified default cell zone group	(get-zones-of-group ’fluid)
	Returns a list of the node zones in the specified default node zone group	(get-zones-of-group ’boundary-node)
	Returns a list of face or edge (as appropriate) zones in the specified user-defined group	(get-zones-of-group ’inlets) returns a list of all inlets in the geometry, where **inlets** is the user-defined group comprising all inlets
(get-face-zones-of-filter ’filter)	Returns a list of the face zones whose names contain the specified filter string	(get-face-zones-of-filter ’prism-cap*)
(get-cell-zones-of-filter ’filter)	Returns a list of the cell zones whose names contain the specified filter string	(get-cell-zones-of-filter ’prism-cells*)
(get-edge-zones-of-filter ’filter)	Returns a list of the edge zones whose names contain the specified filter string	(get-edge-zones-of-filter ’prism-cap*)

Query Function	Description	Examples
`(get-node-zones-of-filter 'filter)`	Returns a list of the node zones whose names contain the specified filter string	`(get-face-zones-of-filter 'boundary-node*)`
`(get-wrapped-zones)`	Returns a list of the wrapped face zones	—

D.1 Using Boolean Operations with Query Functions

You can also perform Boolean operations on lists returned by the query functions. The following Boolean operations can be performed:

- **Union of lists:** (list-union list-1 list-2 . . .)
- **Intersection of lists:** (list-intersection list-1 list-2 . . .)
- **Subtraction of lists:** (list-subtract list-1 list-2)

  Note: *Only two lists can be used as arguments for a subtraction operation.*

  For example,

list-1	=	'(1 2 3 4)
list-2	=	'(1 5 6)
list-3	=	'(1 4 7)

  then,

(list-union list-1 list-2 list-3)	=	'(1 2 3 4 5 6 7)
(list-intersection list-1 list-2 list-3)	=	'(1)
(list-subtract list-1 list-2)	=	'(2 3 4)

D.2 Examples

Some examples of using query functions are:

- Deleting the geometry while retaining the wrapper surface after wrapping operations

  1. Use the command
     ```
     (define initial-zones (get-zones-of-group 'boundary))
     ```
     to obtain a list of the boundary zones.
  2. Perform the wrapping operations as required.
  3. Use the command
     ```
     (define final-zones (get-zones-of-group 'boundary))
     ```
     to obtain a list of the boundary zones after the wrapping operations.
  4. Use the command
     ```
     /boundary/manage/delete (eval-expr '(list-intersection initial-zones
     final-zones))
     ```
     to delete the geometry.

- Smoothing the prism-cap zone created during the prism creation operation

  1. Use the command
     ```
     (define initial-zones (get-face-zones-of-filter 'prism-cap*))
     ```
     to obtain a list of the zones named prism-cap*.
  2. Apply appropriate prism parameters and create prisms.
  3. Use the command
     ```
     (define final-zones (get-face-zones-of-filter 'prism-cap*))
     ```
     to obtain a list of the zones named prism-cap* after the prism creation operation.
  4. Use the command
     ```
     /boundary/improve/smooth (eval-expr '(list-subtract final-zones
     initial-zones))
     ```
     to smooth the recently created prism-cap zone.