Munsys 14

DEVELOPERS GUIDE

Munsys® Developers Guide

Visit Open Spatial on the internet: http://www.openspatial.com

Contents

Appendix

Chapter 1
Introduction

Overview

Munsys requires an API that will allow third parties to develop custom applications. In addition to custom applications, clients will be able to extend and enhance the Munsys functionality to suit their organization's requirements. The API is a LISP interface that enables users to interact with Munsys Objects, as well as the Oracle database. The API does not cater for COM development, as the C++ API and VB API is beyond the scope of this document.

Chapter 2
Function Definition Specification

Function Naming

The API interface should adhere to the following specification:

The function name should be prefixed by “mun-”, then by function category, and lastly by the actual name of the function. The name of the function should inform the user of its exact use. The function name must always contain, where appropriate, “get” or “set”, depending on whether the function is getting or setting a value. The following is a list of categories for the functions:

- Munsys Spatial Object Functions – spo

  All functions that operate on a spatial object will have the category code of spo whether it sets or gets information from the spatial object.
- Database Functions – db

  Functions that retrieve information from the database or functions that set information in the database will have the category code of db.
- MUNLABEL Functions – munlabel

  MUNLABEL functions are functions that will only operate on MUNLABEL objects.
- MUNPOINT Functions – munpoint

  MUNPOINT functions are functions that will only operate on MUNPOINT objects.
- MUNLINE Functions – munline

  MUNLINE functions are functions that will only operate on MUNLINE objects.
- MUNPOLY Functions – munpoly

  MUNPOLY functions are functions that will only operate on MUNPOLY objects.
- General Utility Functions – utl

  General Utility functions that perform some operation other than what is presented above, will reside under the utl category code.

An example of a function that sets an attribute value of a Munsys Spatial Object would be defined as mun-spo-setattribute

All the functions listed in this document should check that the user is currently connected to the database.

Chapter 3
Global Variables

The following variables are defined in Munsys and should not be overwritten by any means.

Application Variables

Variable	Description
#MSCHEMA	Name of the currently logged in schema
#MUSER	Name of the currently logged in user
#MENVIRO	Name of the database connection environment
#LSP_DIR	Directory path of the Munsys applications routines
#INST_DIR	Directory path of the current installation of Munsys
#MDCL_ID	Global Munsys dialog ID
#APP_LST	List of installed applications
#DWG_DIR	Dwg path of where to look for Catalog and Road Intersection drawings

All application variables that are session bound will have a prefix of # whereas all database variable will have a prefix of * and a suffix of *. For example, to reference the session variable of CMS_PARCEL_LSTATUS you would use the following:

Command: !#CMS_PARCEL_LSTATUS

"C"

To reference the database stored value, you would use the following:

Command: !*CMS_PARCEL_LSTATUS*

"R"

The "Types" are categorized as follows:

C – Character

S – Symbol (T of nil)

R – Real number

I – Integer

A – Angle

Type	Category	Variable	Description
C	CFS	CFS_CBLTYPE	Cable Fiber Type
C	CFS	CFS_CBLGROUP	Cable Fiber Group
R	CFS	CFS_FIBSYM_SCL	Cable Fiber Symbol Scale
R	CFS	CFS_LBLFIBLINK_TSIZE	Cable Fiber Label Height
R	CFS	CFS_DIM_TSIZE	Dimension Note Height
R	CFS	CFS_FIBDIM_OFF	Cable Fiber Dimension Distance
R	CFS	CFS_NOTE_TSIZE	Note Tag Height
R	CFS	CFS_GEOM_TOL	Geometry Tolerance
R	CFS	CFS_GEOM_ARCTOL	Fiber Geom Settings
C	CFS	CFS_GEOM_UNIT	Geometry Unit
S	CFS	CFS_CBLAUTOTERM	Cable Fiber Auto Terminate
S	CFS	CFS_CBLASKDUCT	Cable Fiber Duct Prompt
C	CMS	CMS_BUILDING_TYPE	Building Type
C	CMS	CMS_DENSITY_TYPE	Density Type
R	CMS	CMS_EASEMENT_OFFSET	Easement Offset Distance
R	CMS	CMS_EASEMENT_TSIZE	Easement Tag Height
C	CMS	CMS_EASEMENT_TYPE	Easement Type
C	CMS	CMS_LANDUSE_TYPE	Land Use Type
**R	CMS	CMS_MUNICIPAL_TSIZE	Municipality Tag Height
R	CMS	CMS_NOTE_TSIZE	Note Tag Height
C	CMS	CMS_PARCEL_LSTATUS	Parcel Legal Status
R	CMS	CMS_PARCEL_TSIZE	Parcel Tag Height
C	CMS	CMS_PARCEL_TYPE	Parcel Type
C	CMS	CMS_PARCEL_WSTATUS	Parcel Work Status
R	CMS	CMS_SHORT_OBJECTS	Short Objects
R	CMS	CMS_SRCH	Cadastral Search Tolerance
C	CMS	CMS_STR_FNT	Street Text Font
R	CMS	CMS_STRADDR_TSIZE	Street Address Tag Height

R	CMS	CMS_STRNAME_TSIZE	Street Name Tag Height
*R	CMS	CMS_SUBURB_TSIZE	Suburb Tag Height
*C	CMS	CMS_TOWN_FNT	Township Font
*C	CMS	CMS_TOWN_LSTATUS	Township Legal Status
*R	CMS	CMS_TOWN_TSIZE	Township Tag Height
*C	CMS	CMS_TOWN_WSTATUS	Township Work Status
*R	CMS	CMS_WARD_TSIZE	Ward Tag Height
C	CMS	CMS_ZONING_TYPE	Zoning Type
C	CRS	CRS_CRTTYPE	Cable Route Type
C	CRS	CRS_CRT_OWNER	Cable Route Owner
C	CRS	CRS_CRT_STATUS	Cable Route Status
C	CRS	CRS_CRT_STATE	Cable Route State
R	CRS	CRS_CRTLINK_OFF	Cable Route Default Offset Distance
R	CRS	CRS_CRTSYM_SCL	Cable Route Symbol Scale
R	CRS	CRS_CRTNODE_SCL	Cable Route Node Scale
C	CRS	CRS_CRTNODE_NFUNC	Cable Route Node Function
R	CRS	CRS_SRCH	Cable Route Search Tolerance
R	CRS	CRS_CRTLINK_TOL	Cable Route Tolerance
R	CRS	CRS_CRTNODE_TOL	Cable Route Node Tolerance
R	CRS	CRS_SHORT_OBJECTS	Short Objects
R	CRS	CRS_LBLCRTLINK_TSIZE	Cable Route Label Height
R	CRS	CRS_DIM_TSIZE	Dimension Note Height
R	CRS	CRS_NOTE_TSIZE	Note Tag Height
C	CRS	CRS_INFRASTRUCTURE_TYPE	Infrastructure Type
R	CRS	CRS_CRTINF_VAULTSIZE_W	Vault Size Width
R	CRS	CRS_CRTINF_VAULTSIZE_H	Vault Size Height
R	CRS	CRS_CRTINF_SITESIZE_W	Site Size Width
R	CRS	CRS_CRTINF_SITESIZE_H	Site Size Height
R	CRS	CRS_CRTINF_DBOXSIZE_W	Drawbox Size Width
R	CRS	CRS_CRTINF_DBOXSIZE_H	Drawbox Size Height

R	CRS	CRS_CRTINF_PITSIZE_W	Pit Size Width
R	CRS	CRS_CRTINF_PITSIZE_H	Pit Size Height
R	CRS	CRS_CRTINF_MHSIZE_W	Manhole Size Height
R	CRS	CRS_CRTINF_MHSIZE_H	Manhole Size Height
R	CRS	CRS_CRTINF_POLESIZE_DIA	Pole Size Diameter
R	CRS	CRS_DEF_OFF	Node Default Offset Distance
R	CRS	CRS_CRTSWCH_OFF	Switch Node Offset Distance
R	CRS	CRS_CRTMICWV_OFF	Microwave Tower Node Offset Distance
R	CRS	CRS_CRTDLU_OFF	DLU Node Offset Distance
R	CRS	CRS_CRTMDF_OFF	MDF Node Offset Distance
R	CRS	CRS_CRTPRIMJOINT_OFF	Primary Joint Node Offset Distance
R	CRS	CRS_CRTSDC_OFF	SDC Node Offset Distance
R	CRS	CRS_CRTSECJOINT_OFF	Secondary Joint Node Offset Distance
R	CRS	CRS_CRTSCJOINT_OFF	SC Joint Node Offset Distance
R	CRS	CRS_CRTDP_OFF	DP Node Offset Distance
R	CRS	CRS_CRTDPSTUBBY_OFF	DPS Node Offset Distance
R	CRS	CRS_CRTDPAERIAL_OFF	DPA Node Offset Distance
R	CRS	CRS_CRTDPBUILD_OFF	DPB Node Offset Distance
R	CRS	CRS_CRTDIGI_OFF	DIGICON Terminal Node Offset Distance
R	CRS	CRS_CRTMSAN_OFF	MSAN Node Offset Distance
S	CRS	CRS_NODE_SNAP	Snap Nodes
S	CRS	CRS_NODE_AUTPLACE	Automatic Node Placement
S	CRS	CRS_NODE_AUTROT	Align Nodes with Cadastral
S	CRS	CRS_CRTINF_NAMENUM	Infrastructure Prompt Name/Num
S	CRS	CRS_CRTNODE_ASKNO-DEREF	Node Reference
R	DMS	DMS_CATCH_TSIZE	Catchment Tag Height
R	DMS	DMS_CHNL_BWIDTH	Channel Bottom Width

R	DMS	DMS_CHNL_DEPTH	Channel Depth
C	DMS	DMS_CHNL_MATRL	Channel Material
R	DMS	DMS_CHNL_OFF	Channel Offset Distance
R	DMS	DMS_CHNL_TWIDTH	Channel Top Width
C	DMS	DMS_CHNL_TYPE	Channel Type
R	DMS	DMS_CLVT_DEPTH	Culvert Depth
C	DMS	DMS_CLVT_MATRL	Culvert Material
R	DMS	DMS_CLVT_OFF	Culvert Offset Distance
C	DMS	DMS_CLVT_TYPE	Culvert Type
R	DMS	DMS_CLVT_WIDTH	Culvert Top Width
R	DMS	DMS_DAM_TSIZE	Dam Tag Height
C	DMS	DMS_DAM_TYPE	Dam Type
R	DMS	DMS_DIM_TSIZE	Dimension Note Height
R	DMS	DMS_FLOODL_TSIZE	Floodline Tag Height
C	DMS	DMS_FLOODL_TYPE	Floodline Type
R	DMS	DMS_LBLCHNL_TSIZE	Channel Label Height
R	DMS	DMS_LBLCLVT_TSIZE	Culvert Label Height
R	DMS	DMS_LBLSWPIPE_TSIZE	Pipe Label Height
R	DMS	DMS_NOTE_TSIZE	Note Tag Height
R	DMS	DMS_RIVER_TSIZE	River Tag Height
C	DMS	DMS_RIVER_TYPE	River Type
R	DMS	DMS_SHORT_OBJECTS	Short Objects
R	DMS	DMS_SRCH	Stormwater Search Tolerance
R	DMS	DMS_SWNODE_TOL	Stormwater Node Tolerance
R	DMS	DMS_SWSERV_LEN	Stormwater Serv Length
R	DMS	DMS_SWSERV_TOL	Stormwater Serv Tolerance
C	DMS	DMS_SWSERV_TYPE	Stormwater Serv Type
R	DMS	DMS_SWSYM_SCL	Stormwater Symbol Scale
R	DMS	DMS_SWPIPE_DIA	Stormwater Pipe Diameter
C	DMS	DMS_SWPIPE_MATRL	Stormwater Pipe Material

R	DMS	DMS_SWPIPE_OFF	Stormwater Default Offset Distance
R	DMS	DMS_SWPIPE_TOL	Stormwater Pipe Tolerance
C	DMS	DMS_SWPIPE_TYPE	Stormwater Pipe Type
R	DMS	DMS_SWSYM_SCL	Stormwater Symbol Scale
S	EMS	EMS_CABLE_ASKINFO	Cable Info
R	EMS	EMS_CABLE_OFF	Cable Default Offset Distance
R	EMS	EMS_CABLE_TIEIN	Cable Tie-in distance
R	EMS	EMS_CABLE_TOL	Electricity Cable Tolerance
R	EMS	EMS_DCT_FIX	Cable duct trim or extend
R	EMS	EMS_DEF_DST	Default Cable Tie-in distance
C	EMS	EMS_DEF_MATRL	Cable material
R	EMS	EMS_DEF_OFF	Node Default Offset Distance
R	EMS	EMS_DEF_SLK	Node slack
R	EMS	EMS_DIM_TSIZE	Dimension Note Height
R	EMS	EMS_EALINK_DST	Airlink cable tie-in distance
R	EMS	EMS_EALINK_OFF	Airlink symbol offset distance
R	EMS	EMS_EALINK_SLK	Airlink cable slack
R	EMS	EMS_EBBRD_DST	Billboard cable tie-in distance
R	EMS	EMS_EBBRD_OFF	Billboard symbol offset distance
R	EMS	EMS_EBBRD_SLK	Billboard cable slack
R	EMS	EMS_EDB_DST	DB cable tie-in distance
R	EMS	EMS_EDB_OFF	DB symbol offset distance
R	EMS	EMS_EDB_SLK	DB cable slack
R	EMS	EMS_EHMSL_DST	HMSL cable tie-in distance
R	EMS	EMS_EHMSL_OFF	HMSL symbol offset distance
R	EMS	EMS_EHMSL_SLK	HMSL cable slack
R	EMS	EMS_EHVJUNC_DST	HV Junction Box cable tie-in distance
R	EMS	EMS_EHVJUNC_OFF	HV Junction Box symbol offset distance
R	EMS	EMS_EHVJUNC_SLK	HV Junction Box cable slack

R	EMS	EMS_EHVSWITCH_DST	HV Switch cable tie-in distance
R	EMS	EMS_EHVSWITCH_OFF	HV Switch symbol offset distance
R	EMS	EMS_EHVSWITCH_SLK	HV Switch cable slack
R	EMS	EMS_EKIOSK_DST	Kiosk cable tie-in distance
R	EMS	EMS_EKIOSK_OFF	Kiosk symbol offset distance
R	EMS	EMS_EKIOSK_SLK	Kiosk cable slack
R	EMS	EMS_ELVJUNC_DST	LV Junction Box cable tie-in distance
R	EMS	EMS_ELVJUNC_OFF	LV Junction Box symbol offset distance
R	EMS	EMS_ELVJUNC_SLK	LV Junction Box cable slack
R	EMS	EMS_EMSUB_DST	Mini Sub cable tie-in distance
R	EMS	EMS_EMSUB_OFF	Mini Sub symbol offset distance
R	EMS	EMS_EMSUB_SLK	Mini Sub cable slack
R	EMS	EMS_EPOLE_DST	Pole cable tie-in distance
R	EMS	EMS_EPOLE_OFF	Pole symbol offset distance
R	EMS	EMS_EPOLE_SLK	Pole cable slack
R	EMS	EMS_EPYLON_DST	Pylon cable tie-in distance
R	EMS	EMS_EPYLON_OFF	Pylon symbol offset distance
R	EMS	EMS_EPYLON_SLK	Pylon cable slack
R	EMS	EMS_ESL_DST	Streetlight cable tie-in distance
R	EMS	EMS_ESL_OFF	Streetlight symbol offset distance
R	EMS	EMS_ESL_SLK	Streetlight cable slack
R	EMS	EMS_ESLJUNC_DST	SL Junction Box cable tie-in distance
R	EMS	EMS_ESLJUNC_OFF	SL Junction Box symbol offset distance
R	EMS	EMS_ESLJUNC_SLK	SL Junction Box cable slack
R	EMS	EMS_ESLPOLE_DST	Streetlight Pole cable tie-in distance
R	EMS	EMS_ESLPOLE_OFF	Streetlight Pole symbol offset distance
R	EMS	EMS_ESLPOLE_SLK	Streetlight Pole cable slack

R	EMS	EMS_ESSTA_DST	Switch Station cable tie-in distance
R	EMS	EMS_ESSTA_OFF	Switch Station symbol offset distance
R	EMS	EMS_ESSTA_SLK	Switch Station cable slack
R	EMS	EMS_ESSUB_DST	Standard Sub cable tie-in distance
R	EMS	EMS_ESSUB_OFF	Standard Sub symbol offset distance
R	EMS	EMS_ESSUB_SLK	Standard Sub cable slack
R	EMS	EMS_ETRANS_DST	Transformer cable tie-in distance
R	EMS	EMS_ETRANS_OFF	Transformer symbol offset distance
R	EMS	EMS_ETRANS_SLK	Transformer cable slack
R	EMS	EMS_ISCALE	Insertion Scale
S	EMS	EMS_NODE_ASKDESC	Node Descriptions
S	EMS	EMS_NODE_AUTPLACE	Automatic Node Placement
S	EMS	EMS_NODE_AUTROT	Align nodes with cadastral
S	EMS	EMS_NODE_SNAP	Snap nodes
R	EMS	EMS_NODE_TOL	Electricity Node Tolerance
R	EMS	EMS_NOTE_TSIZE	Note Tag Height
S	EMS	EMS_SC_ASKINFO	Service connection Info
R	EMS	EMS_SC_LEN	Service connection length
R	EMS	EMS_SC_TOL	Electricity SC Tolerance
R	EMS	EMS_SRCH	Electricity Search Tolerance
R	EMS	EMS_SYM_SCL	Electricity Symbol Scale
R	EMS	EMS_ZONE_TSIZE	Zone Tag Height
I	GEN	DB_EXTENTS_RESOLUTION	Database Extents Display Resolution
C	GEN	DWG_DIR	Drawing Directory
S	GEN	INTEG_NETWORK_AUTCHECK	Integrity Network Auto Check
R	GEN	INTEG_CIRCSIZE	Integrity Circle Size
I	GEN	PRECISION	Precision

S	GEN	ROTATE_TRANSFORMED	Rotate Tag and Symbol
S	GEN	SCALE_TRANSFORMED	Scale Tag and Symbol
R	GEN	SNAP	Snap Tolerance
A	GEN	TAG_ANGLE	Tag Angle
C	GEN	TAG_FNT	Tag Font
C	GEN	TAG_JUST	Tag Justification
R	GEN	TAG_SIZE	Tag Height
C	MBK	MBK_COORDS_FONT	Font for coordinates
C	MBK	MBK_DATESCALE_FONT	Font for the Date/Scale
C	MBK	MBK_DIR	Map Book Drawing Directory
C	MBK	MBK_DWG_PFIX	Drawing Prefix
C	MBK	MBK_EVN_PFIX	Even Page Prefix
C	MBK	MBK_FONT	Map Book Font
C	MBK	MBK_ODD_PFIX	Odd Page Prefix
C	MBK	MBK_PAGENO_FONT	Font for Page Number
C	MBK	MBK_TITLE_FONT	Font for Title
R	MBK	MBK_XORD_FACT	X Coordinate Multiplication Factor
R	MBK	MBK_YORD_FACT	Y Coordinate Multiplication Factor
R	RMS	RMS_ISCALE	Insertion Scale
R	RMS	RMS_NOTE_TSIZE	Road Note Height
C	RMS	RMS_RDCL_CLASS	Road CL Classification
C	RMS	RMS_RDCL_JURIS	Road CL Jurisdiction
R	RMS	RMS_RDCL_OFF	Road CL Offset Distance
C	RMS	RMS_RDCL_OWNER	Road CL Segment Owner
C	RMS	RMS_RDCL_SURFACE	Road CL Surface Type
R	RMS	RMS_RDCL_TOL	Road CL Tolerance
C	RMS	RMS_RDCL_TRAFDIR	Road CL Traffic Direction
C	RMS	RMS_RDCL_TYPE	Road CL Segment Type
R	RMS	RMS_RDEDGE_OFF	Road Edge Offset Distance

R	RMS	RMS_RDINT_TOL	Road Int Tolerance
R	RMS	RMS_RDINTSYM_SCL	Intersection Marker Symbol Scale
R	RMS	RMS_RDTSYM_SCL	Traffic Symbol Scale
R	RMS	RMS_RDWALK_OFF	Road Walkway Offset Distance
R	RMS	RMS_SHORT_OBJECTS	Short Objects
R	RMS	RMS_SRCH	Road Search Tolerance
R	SMS	SMS_BASIN_TSIZE	Basin Tag Height
R	SMS	SMS_DIM_TSIZE	Dimension Note Height
R	SMS	SMS_GPIPE_DIA	Gravity Pipe diameter
C	SMS	SMS_GPIPE_MATRL	Gravity Pipe Material
R	SMS	SMS_GPIPE_OFF	Gravity Pipe Default Offset Distance
C	SMS	SMS_GPIPE_TYPE	Gravity Pipe Type
R	SMS	SMS_LBLGPIPE_TSIZE	Gravity Pipe Label Height
R	SMS	SMS_LBLRPIPE_TSIZE	Pressure Pipe Label Height
R	SMS	SMS_LBLVPIPE_TSIZE	Vacuum Pipe Label Height
R	SMS	SMS_NOTE_TSIZE	Note Tag Height
R	SMS	SMS_RESPIPE_OFF	Residential Pipe Default Offset Distance
R	SMS	SMS_RPIPE_DIA	Pressure Pipe diameter
C	SMS	SMS_RPIPE_MATRL	Pressure Pipe Material
R	SMS	SMS_RPIPE_OFF	Pressure Pipe Default Offset Distance
C	SMS	SMS_RPIPE_TYPE	Pressure Pipe Type
R	SMS	SMS_SEWNODE_SCL	Sewer Node Scale
R	SMS	SMS_SEWNODE_TOL	Sewer Node Tolerance
R	SMS	SMS_SEWPIPE_TOL	Sewer Pipe Tolerance
R	SMS	SMS_SEWSERV_LEN	Sewer Serv Length
R	SMS	SMS_SEWSERV_TOL	Sewer Serv Tolerance
C	SMS	SMS_SEWSERV_TYPE	Sewer Serv Type
R	SMS	SMS_SEWSYM_SCL	Sewer Symbol Scale

R	SMS	SMS_SHORT_OBJECTS	Short Objects
R	SMS	SMS_SRCH	Sewer Search Tolerance
R	SMS	SMS_VPIPE_DIA	Vacuum Pipe diameter
C	SMS	SMS_VPIPE_MATRL	Vacuum Pipe Material
R	SMS	SMS_VPIPE_OFF	Vacuum Pipe Default Offset Distance
C	SMS	SMS_VPIPE_TYPE	Vacuum Pipe Type
R	WMS	WMS_DIM_TSIZE	Dimension Note Height
R	WMS	WMS_LBLWATPIPE_TSIZE	Pipe Label Height
R	WMS	WMS_NOTE_TSIZE	Note Tag Height
R	WMS	WMS_SHORT_OBJECTS	Short Objects
R	WMS	WMS_SRCH	Water Search Tolerance
R	WMS	WMS_WAT_CAT	Water Category
R	WMS	WMS_WATBRCH_OFF	Water branch offset
R	WMS	WMS_WATNODE_SCL	Water Node Scale
R	WMS	WMS_WATNODE_TOL	Water Node Tolerance
R	WMS	WMS_WATPIPE_DIA	Water Pipe Diameter
C	WMS	WMS_WATPIPE_MATRL	Water Pipe Material
R	WMS	WMS_WATPIPE_OFF	Water Default Offset Distance
R	WMS	WMS_WATPIPE_TOL	Water Pipe Tolerance
C	WMS	WMS_WATPIPE_TYPE	Water Pipe Type
R	WMS	WMS_WATSERV_LEN	Water serv Length
R	WMS	WMS_WATSERV_OFF	Water serv Default Offset Distance
R	WMS	WMS_WATSERV_TOL	Water Serv Tolerance
R	WMS	WMS_WATSYM_SCL	Water Symbol Scale
R	WMS	WMS_ZONE_TSIZE	Zone Tag Height

*International version only; **US version only

Munsys System Variables

The following system variables are used by MUN-UTL-SETMUNVAR and MUN-UTL-GETMUNVAR:

Integrity Variables

Variable	Values	Description
MUNINTDELMARKERS	ON/OFF Default: ON	Erase integrity markers at start of operation
MUNINTDLGRESULTS	ON/OFF Default: ON	Show integrity result summary dialog
MUNINTDLGNOTIFY	ON/OFF Default: ON	Notify when objects require network validation

Database Posting Variables

Variable	Values	Description
MUNPSTDLGSUMMARY	ON/OFF Default: ON	Show posting summary dialog
MUNPSTDLGRESULTS	ON/OFF Default: ON	Show posting results dialog
MUNPSTLCKAUTOUNLOCK	ON/OFF Default: ON	Auto-unlock objects after posting
MUNPSTUNDORESET	ON/OFF Default: ON	After posting objects to the database the undo will be cleared to avoid synchronization problems between the content of a drawing and the content in the database. Using the MUNPOSTOBJS command in a lisp routine will cause an error because of the nested function to clear the undo if it is called multiple times. This variable will avoid this error. It should only be switched OFF in a situation where a LISP routine is calling the MUNPOSTOBJS command multiple times.

ToolTip Variables

Variable	Values	Description
MUNTIPSHOW	ON/OFF Default: ON	Show object description
MUNTIPOBJGID	ON/OFF Default: OFF	Show object GID
MUNTIPOBJMODIFY	ON/OFF Default: OFF	Show object modified status
MUNTIPOBJATTR	ON/OFF Default: OFF	Show modified attributes
MUNTIPINTMARKER	ON/OFF Default: ON	Show integrity marker information

Color Variables

Variable	Values	Description
MUNCOLCONSTRUCTION	Integer or string representing color index Default: 6 - Magenta	Construction Color
MUNCOLINTEGRITY	Integer or string representing color index Default: 3 - Green	Integrity Color

Geometry Validation Variables

Variable	Values	Description
MUNINTGEOMVALIDATE	OFF/MODIFIED/ALL Default: MODIFIED	Validate Geometry

Attribute Validation Variables

Variable	Values	Description
MUNINTATTRVALIDATE	OFF/MODIFIED/ALL Default: MODIFIED	Validate Attribute

Network Integrity Variables

Variable	Values	Description
MUNINTTOPOVALIDATE	ON/OFF Default: ON	Validate Topology
MUNINTCONNVALIDATE	ON/OFF Default: ON	Validate Connectivity

Chapter 4
Munsys Objects

DFX Codes

The following dfx codes are specific to the Munsys objects such as MUNLABEL, MUNPOINT, MUNLINE, MUNPOLY.

DXF Group Code	DESCRIPTION
75	Munsys Status Flags
95	Munsys ID
96	Spatial Object GID
97	Attributes – used internally

Chapter 5
Command Line Functions

MUNQRYEXEC

Allows you to run and change a query dynamically from the command line interface.

Query ID to Execute:

- The default queries are as follows. To obtain queries not listed below, please refer to the MUNSYS_QUERY table:

Query ID	Description
*1	Townships
*2	Townships (Archived)
3	Parcels
4	Parcels (Archived)
*5	Suburbs
*6	Block Boundaries
*7	Wards
8	Street Name
9	Street Addresses
10	Easement Lines
11	Easement Line Text
12	TP Zoning
13	TP Density
14	Easement Polygons
15	Easement Polygon Text
16	Buildings
17	Cadastral Notes
**18	Municipalities
19	TP Land Use

20	Sewer Labels
21	Sewer Dimensions
22	Sewer Basins
23	Sewer Nodes
24	Sewer Notes
25	Sewer Pipes (Gravity)
26	Sewer SCs
27	Sewer Pipes (Vacuum)
28	Sewer Pipes (Pressure)
29	Sewer Residential Pipes
33	Sewer Dimension Notes
34	Sewer Symbols
35	Sewer Map Page Grid
41	Water Dimensions
42	Water Zones
43	Water Nodes
44	Water Notes
45	Water Pipes
46	Water SCs
47	Water Residential Pipes
49	Water Dimension Notes
50	Water Symbols
51	Water Map Page Grid
401	Water Dimensions (Potable)
402	Water Dimensions (Reclaimed)
403	Water Dimensions (Abandoned)
404	Water Dimensions (Raw)
405	Water Zones (Potable)
406	Water Zones (Reclaimed)
407	Water Zones (Abandoned)

408	Water Zones (Raw)
409	Water Nodes (Potable)
410	Water Nodes (Reclaimed)
411	Water Nodes (Abandoned)
412	Water Nodes (Raw)
413	Water Notes (Potable)
414	Water Notes (Reclaimed)
415	Water Notes (Abandoned)
416	Water Notes (Raw)
417	Water Pipes (Potable)
418	Water Pipes (Reclaimed)
419	Water Pipes (Abandoned)
420	Water Pipes (Raw)
421	Water SCs (Potable)
422	Water SCs (Reclaimed)
423	Water SCs (Abandoned)
424	Water SCs (Raw)
433	Water Dimension Notes (Potable)
434	Water Dimension Notes (Reclaimed)
435	Water Dimension Notes (Abandoned)
436	Water Dimension Notes (Raw)
437	Water Symbols (Potable)
438	Water Symbols (Reclaimed)
439	Water Symbols (Abandoned)
440	Water Symbols (Raw)
60	Stormwater Labels
61	Stormwater Dimensions
62	Stormwater Catchments
63	Stormwater Nodes
64	Stormwater Notes

65	Stormwater Pipes
66	Stormwater Channels
67	Stormwater Culverts
68	River Lines
69	River Polygons
70	Floodlines
71	Dams
72	Stormwater Pipe Labels
73	Stormwater Channel Labels
74	Stormwater Culvert Labels
75	Stormwater Dimension Notes
76	Stormwater SCs
77	Stormwater Symbols
81	Road Centre Lines
82	Road Intersections
83	Road Walkways
84	Road Areas
85	Road Edges
86	Road Notes
101	Electricity Cables
102	Electricity Nodes
103	Electricity Ducts
104	Electricity Transformer Zones
105	Electricity Notes
106	Electricity Dimensions
107	Electricity SCs
108	Electricity Dimension Notes
109	Electricity Cables (EHV)
110	Electricity Cables (HV)
111	Electricity Cables (LV)

112	Electricity Cables (SL)
113	Electricity Cables (SC)
114	Electricity Nodes (EHV)
115	Electricity Nodes (HV)
116	Electricity Nodes (LV)
117	Electricity Nodes (SL)
118	Electricity Nodes (SC)
145	Cable Route Symbols
143	Cable Route Notes
142	Cable Route Nodes
141	Cable Route Links
146	Cable Route Labels
148	Cable Route Infrastructure
144	Cable Route Dimensions
147	Cable Route Dimension Notes
161	Fiber Cables
162	Fiber Paths
163	Cable Fiber Notes
164	Cable Fiber Dimensions
165	Cable Fiber Symbols
166	Cable Fiber Dimension Notes
167	Cable Fiber Service Connections
171	Qry: Fiber Cable by Type
172	Qry: Fiber Path by Service
173	Qry: Fiber Strand Count (In Use)

*International version only; **US version only

Parameter to Change <none>:

Parameter Name	Parameter Description
Gscmode	A number indicating whether to use a GSC for the current query. (0 = no GSC, 1 = use GSC)
qryTxt	A number indicating whether to query text. (0 = no, 1 = yes)
obJFmt	A number indicating how the object should be formatted. Supports only the following values now: 1 – MUNLABEL 2 – MUNPOINT 3 – MUNLINE 4 – MUNPOLY
obJColor	Indicates the color for the geometry – NULL implies BYLAYER.
obJAngle	Object angle. See Format Options for valid values. (# = default)
obJBulge	Object bulge factor. See Format Options for valid values.
obJElev	Object elevation. See Format Options for valid values.
obJLayer	Object layer. See Format Options for valid values. (# = default)
obJScale	Object scale. See Format Options for valid values. (# = default)
obJThick	Object thickness. See Format Options for valid values.
obJWidth	Object width. See Format Options for valid values.
obJBLock	Indicates whether to query object as a block object (0 = No, 1 = Yes).
txTAngle	Text angle. See Format Options for valid values. (# = default)
txTColor	Indicates the color for the label – NULL implies BYLAYER.
txTElev	Text elevation. See Format Options for valid values.
txTJust	Text justification. See Format Options for valid values. (# = default)
txTLayer	Text layer. See Format Options for valid values. (# = default)
txTHeight	Text size. See Format Options for valid values. (# = default)
txTStyle	Text style. See Format Options for valid values.
txTValue	Text value. See Format Options for valid values. (# = default)
aTTTable	A string indicating the name of an optional attribute table to join for the query.
aTTCond	A string indicating an additional WHERE clause to be used in the query.

sPJoincol	***The column to join the spatial table on. This will be included in the WHERE clause when joining the tables.
aTTJoincol	***The column to join the attribute table on. This will be included in the WHERE clause when joining the tables.
fiLLMode	0 – no fill 1 – apply a fill (only when obJFmt = 4)
fiLLType	Indicates the pattern type either Custom or Predefined. Used only if fiLLMode = 1 and fiLLPatt refers to a hatch pattern (not a solid)
fiLLColor	Indicates the color for the fill – NULL implies BYLAYER. Used only if fiLLMode = 1.
fiLLPatt	Name of fill pattern – NULL implies SOLID. Used only if fiLLMode = 1.
fiLLAngle	Hatch angle in radians. Used only if fiLLMode = 1 and fiLLPatt does not refer to a solid.
fiLLSCale	Hatch scale. Used only if fiLLMode = 1 and fiLLPatt refers to a hatch pattern (not a solid).
fiLLSPace	Indicates the size of the hatch spacing. Used only if fiLLMode = 1 and fiLL-Patt refers to a hatch pattern (not a solid)
fiLLDouble	Can either be Double or Cross. Used only if fiLLMode = 1 and fiLLPatt refers to a hatch pattern (not a solid)
ranGEMode	Specifies how to create entity layer: 0 specifies to use the name as from obJLayer 1 to create a range of layers using parameters from the ran parameters
ranGEColumn	Column being evaluated for range of layers. Used when ranGEMode = 1.
ranGEGroups	Number of groups to split range into
ranGEDec	Digits after decimal point – used to convert numeric value to a string
ranGEMIN	Indicates Start of Range
ranGEMAX	Column being evaluated for range of layers. Used when ranGEMode = 1.Indi-cates End of Range

Chapter 6
Format Options

Format Options

The following format options can be used for queries and expressions in Munsys.

= default

$ = SQL expression. This will enable the user to define SQL expressions to evaluate as in - $'PARCEL'||'_'||PRCL_TYPE

Important There is a 128 character limit when passing strings from lisp to the MUNQRYEXEC command. To solve this limitation, simply pass the string as a variable prefixed with "!" , as seen in the example below:

If the string is less than 128 characters, then you can use a normal lisp variable:
(command "munqryexec" "3" "Gscmode" 0 "atTCond" #QRYCOND "")

If the string is more than 128 characters, then you will have to use:
(command "munqryexec" "3" "Gscmode" 0 "atTCond" "!#QRYCOND" "")

Chapter 7
API Functions

SPO FUNCTIONS

MUN-SPO-CLRINTEGFLAG

Clears the integrity flag for a Munsys object.

(mun-spo-clrintegflag ename)

Arguments

ename

Name of the entity whose integrity flag you wish to clear.

Return Values

None

Examples

Command: (setq ename (entsel))
Command: (mun-spo-clrintegflag (car ename))

MUN-SPO-GETATTRIBUTE

Retrieves column values for object.

(mun-spo-getattribute ename column_list)

Arguments

ename

Name of the entity whose attributes you wish to retrieve.

column_list

List of column name whose values you wish to retrieve.

Return Values

List of column values, as strings, corresponding to the *column_list* otherwise returns nil.

Examples

Command: (mun-spo-getattribute (car (entsel))
(list "GID" "TAG_VALUE"))
("13535" "FERNDALE")

MUN-SPO-GETGID

Gets the GID for a Munsys object.

(mun-spo-getgid ename)

Arguments

ename

Name of the entity whose GID you wish to retrieve.

Return Values

GID of object as an integer number otherwise returns 0 if no GID.

Examples

Command: (mun-spo-getgid (car (entsel)))
12537

MUN-SPO-GETMUNID

Gets the MUNID for a Munsys object.

(mun-spo-getmunid ename)

Arguments

ename

Name of the entity whose MUNID you wish to retrieve.

Return Values

MUNID of object as an integer number otherwise returns 0 if no MUNID.

Examples

Command: (mun-spo-getmunid (car (entsel)))
3

MUN-SPO-GETTAGANGLE

Gets the tag angle for a Munsys object. The value will be retrieved from the object in the drawing and not from the TAG_ANGLE column in the database.

(mun-spo-gettagangle ename)

Arguments

ename

Entity name of object to retrieve tag angle from.

Return Values

Returns angle for the tag as a real number in radians.

Examples

Command: (setq ename (car (entsel)))
Command: (mun-spo-gettagangle ename)
1.579

MUN-SPO-GETTAGJUST

Gets the tag justification for a Munsys object. The value will be retrieved from the object in the drawing and not from the TAG_JUST column in the database.

(mun-spo-gettagjust ename)

Arguments

ename

Entity name of object to retrieve tag justification from.

Return Values

Returns justification for the tag as a string. Valid values are:

L – Left

C – Center

M – Middle

R – Right

TL – Top Left

TC – Top Center

TR – Top Right

ML – Middle Left

MC – Middle Center

MR – Middle Right

BL – Bottom Left

BC – Bottom Center

BR – Bottom Right

Examples

Command: (setq ename (car (entsel)))
Command: (mun-spo-gettagjust ename)
"MC"

MUN-SPO-GETTAGSIZE

Gets the tag size for a Munsys object. The value will be retrieved from the object in the drawing and not from the TAG_SIZE column in the database.

(mun-spo-gettagsize ename)

Arguments

ename

Entity name of object to retrieve tag size from.

Return Values

Returns size for the tag as a real number.

Examples

Command: (setq ename (car (entsel)))
Command: (mun-spo-gettagsize ename)
2.0

MUN-SPO-GETTAGSTYLE

Gets the tag style for a Munsys object. The value will be retrieved from the object in the drawing.

(mun-spo-gettagstyle ename)

Arguments

ename

Entity name of object to retrieve tag style from.

Return Values

Returns style for the tag as a string.

Examples

Command: (setq ename (car (entsel)))
Command: (mun-spo-gettagstyle ename)
"STANDARD"

MUN-SPO-GETTAGVALUE

Gets the tag value for a Munsys object. The value will be retrieved from the object in the drawing and not from the TAG_VALUE column in the database.

(mun-spo-gettagvalue ename)

Arguments

ename

Entity name of object to retrieve tag value from.

Return Values

Returns the tag value as a string.

Examples

Command: (setq ename (car (entsel)))
Command: (mun-spo-gettagvalue ename)
"MY TAG"

MUN-SPO-GETTAGXY

Gets the tag xy coordinate point for a Munsys object.

(mun-spo-gettagxy ename)

Arguments

ename

Entity name of object to retrieve tag xy point from.

Return Values

XY Coordinate as a point

Examples

Command: (setq ename (car (entsel)))
Command: (mun-spo-gettagxy ename)
(-77588.5 -2.88402e+006 0.0)

MUN-SPO-REPLACEGEOMETRY

This function replaces the geometry of a Munsys object with that of a CAD entity that is of the same compatible type.

The list of compatible types is as follows:

- MUNPOLY—Closed LWPOLYLINE, MPOLYGON
- MUNLINE—LWPOLYLINE, LINE
- MUNPOINT—POINT, TEXT, BLOCK
- MUNLABEL—TEXT

(mun-spo-replacegeometry ename cad_ename)

Arguments

ename

Name of the entity whose geometry you wish to replace.

cad_ename

Name of the CAD entity whose geometry you wish to use.

Return Values

T if successful or nil otherwise

Examples

Command: (setq ename (car (entsel)))
Command: (setq cad_ename (car (entsel)))
Command: (mun-spo-replacegeometry ename cad_ename)
T

MUN-SPO-SETATTRIBUTE

Attaches column values to object for posting to database.

(mun-spo-setattribute ename column_names column_values)

Arguments

ename

Name of the entity to update.

column_names

List of strings containing the columns to update.

column_values

List of strings containing the values to update the column_names with.

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (setq #FLST (list "PIPE_TYPE"))
Command: (setq ENT (car (entsel)))
Command: (setq #DLST (list "PRESSURE"))
Command: (mun-spo-setattribute ENT #FLST #DLST)
T

MUN-SPO-SETTAGANGLE

Sets the tag angle for a Munsys object. Please note that the column TAG_ANGLE will be attached to the object as an attribute update. The object tag angle in the drawing will also be updated

(mun-spo-settagangle ename tag_angle)

Arguments

ename

Name of the entity whose tag angle you wish to update.

tag_angle

New angle for tag, as a real number in radians.

Return Values

Returns T if successful or nil otherwise

Examples

Command: (mun-spo-settagangle (car (entsel)) 1.579))
T

MUN-SPO-SETTAGJUST

Sets the tag justification for a Munsys object.

(mun-spo-settagjust ename tag_justification)

Arguments

ename

Name of the entity whose tag justification you wish to update.

tag_justification

New justification for tag, as a string. Valid values are:

L – Left

C – Center

M – Middle

R – Right

TL – Top Left

TC – Top Center

TR – Top Right

ML – Middle Left

MC – Middle Center

MR – Middle Right

BL – Bottom Left

BC – Bottom Center

BR – Bottom Right

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-spo-settagjust (car (entsel)) "MC")
T

MUN-SPO-SETTAGSIZE

Sets the tag size for a Munsys object. Please note that the column TAG_SIZE will be attached to the object as an attribute update. The object tag size in the drawing will also be updated.

(mun-spo-settagsize ename tag_size)

Arguments

ename

Name of the entity whose tag size you wish to update.

tag_size

New size for tag, as a real number.

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-spo-settagsize (car (entsel)) 2.0))
T

MUN-SPO-SETTAGSTYLE

Sets the tag style for a Munsys object

(mun-spo-settagstyle ename tag_style)

Arguments

ename

Name of the entity whose tag style you wish to update.

tag_style

New style for tag, as a string.

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-spo-settagstyle (car (entsel)) "STANDARD"))
T

MUN-SPO-SETTAGVALUE

Sets the tag value for a Munsys object. Please note that the column TAG_VALUE will be attached to the object as an attribute update. The object tag value in the drawing will also be updated.

(mun-spo-settagvalue ename tag_value)

Arguments

ename

Name of the entity whose tag value you wish to update.

tag_value

New value for tag, as a string.

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-spo-settagvalue (car (entsel)) "MY TAG"))
T

DB FUNCTIONS

MUN-DB-COMMIT

Commits any open transactions to the database.

(mun-db-commit)

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-db-commit)
T

MUN-DB-GETLOOKUP

Display lookup dialog and returns selected value/s.

(mun-db-getlookup table_name dialog_description selection_num selected_column selected_row [is_numeric])

Arguments

table_name

Name of lookup table as a string.

dialog_description

Description of the dialog as a string. This is the text that is displayed in the title bar of the lookup dialog.

selection_num

This value determines how many values the user is allowed to select in the lookup dialog as an integer number.

selected_column

If the dialog must preselect a value when opening the dialog, then this value is the value of the column whose value is passed in selected_row as a integer value. Columns are 0 based indexed.

selected_row

This is the value of the preselected row to display as a string value.

is_numeric

Optional parameter indicating whether the value of the preselected row is a numeric value. Valid values for this parameter are either T or nil.

Return Values

List of values selected.

Examples

Command: (mun-db-getlookup "LU_CMS_PRCLTYPE" "My Lookup" 1 1 "")
(("R") ("Registered"))

Command: (mun-db-getlookup "LU_SMS_PIPEDIA" "My Lookup" 1 0 "4" T)
(("4") (" 4"))

MUN-DB-GETLOOKUPVALS

Returns lookup table values as 2 separate lists. First list consisting of the LCODE values and the second list consisting of the LVALUE values.

(mun-db-getlookupvals table_name)

Arguments

table_name

Name of lookup table as a string.

Return Values

List consisting of 2 sublists – ((lcode values)(lvalue values))

Examples

Command: (mun-db-getlookupvals "LU_CMS_SUBNAME")
(("1981" "0680" "0500" "0200" "0180" "0150" "0050" "0020" "0015" "0010" "0007"
"0006" "0004" "0003" "0002" "0001" "0000") ("KLIPFONTEIN 35-IR" "NIRVANA"
"WESTENBURG" "HAMBERG" "TREESBANK A/H" "ROCKY CREST" "FAIRVIEW" "OLYMPIA" "HOCHLAND PARK" "CONSTANTIA A/H" "RANDPARK" "GREYMONT" "KENSINGTON" "IVYDALE" "ISANDO" "CHARTWELL A/H" "UNKNOWN"))

MUN-DB-GETNEXTSEQ

Gets next sequence number from the database.

(mun-db-getnextseq sequence_name)

Arguments

sequence_name

Sequence name, as a string, of the sequence whose value you wish to retrieve.

Return Values

Next sequence number as an integer value or nil if sequence does not exist.

Examples

Command: (mun-db-getnextseq "SMS_SEWNODE")
6285

MUN-DB-GETSQL

Executes SQL statement and returns results as a list.

(mun-db-getsql sql_statement)

Arguments

sql_statement

String containing SQL statement to execute on database.

Return Values

List consisting of results.

Examples

Command: (mun-db-getsql (strcat "select * from "#MSCHEMA ".LU_CMS_-SUBNAME"))

(("0000" "UNKNOWN") ("0001" "CHARTWELL A/H") ("0002" "ISANDO") ("0003"

"IVYDALE") ("0004" "KENSINGTON") ("0006" "GREYMONT") ("0007" "RANDPARK") ("0010" "CONSTANTIA A/H") ("0015" "HOCHLAND PARK") ("0020" "OLYMPIA") ("0050""FAIRVIEW") ("0150" "ROCKY CREST") ("0180" "TREESBANK A/H") ("0200" "HAMBERG") ("0500" "WESTENBURG") ("0680" "NIRVANA") ("1981" "KLIPFONTEIN 35-IR"))

MUN-DB-INEXTENTS

Determines if a list of points fall within the database extents.

(mun-db-inextents point_lst)

Arguments

point_list

List of points or this can be a single point.

Return Values

T if all the points fall within the database extents or nil otherwise.

Examples

Command: (mun-db-inextents (list (list -75877.706 -2882642.686 0.0) (list -75771.080 -2882578.395 0.0)))

T

MUN-DB-ISCONNECTED

Determines if user is connected to the database or not.

(mun-db-isconnected)

Return Values

T if connected to the database or nil otherwise.

Examples

Command: (mun-db-isconnected)
T

MUN-DB-RESET

Resets application session variables

(mun-db-reset)

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-db-reset)
T

MUN-DB-SETSQL

Executes SQL statement on database – SQL statement usually consists of insert or delete statements

(mun-db-setsql sql_statement)

Arguments

sql_statement

String containing SQL statement to execute on database

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-db-setsql (strcat "delete from "#MSCHEMA ".LU_CMS_-SUBNAME"))

T

UTL FUNCTIONS

MUN-UTL-ACADANGLE

Converts true radian angle to the current AutoCAD Setting units.

(mun-utl-acadangle angle)

Arguments

angle

Angle in radians as a real value.

Return Values

Returns angle as a string value.

Examples

Command: (mun-utl-acadangle 1.57)
"89.954"

MUN-UTL-BOUNDARYOFFSET

Offsets boundary segments and constructs polyline object.

(mun-utl-boundaryoffset multiply_value)

Arguments

multiply_value

This is the distance the default offset distance is multiplied by.

Return Values

Returns entity name of newly constructed object.

Examples

Command: (mun-utl-boundaryoffset 1.0)
Select offset segment:
Specify point on side to offset:
Multiply offset by <1.000>:
Select offset segment:
Done selecting offset segments [Yes/No] <Yes>:
<Entity name: 7efdec40>

MUN-UTL-CAPNODE

Captures node and breaks entity if requested.

(mun-utl-capnode mun_id ename point insertion_type break_entity insertion_angle symbol_name symbol_scale)

Arguments

mun_id

Munsys Spatial ID as an integer. This is the Munsys ID of the Node that is being inserted.

ename

Name of the entity that you wish to insert the node on.

point

Point entity that you wish to insert the node on.

insertion_type

Valid values are, as string values:

F – Freehand

E - Endpoint

break_entity

Indicates whether you wish to break the entity being inserted on as a string value.

Y - Yes

N - No

insertion_angle

Indicates whether to align the node to the object inserted on. Valid values are:

Y – Yes

N - No

symbol_name

Name of the symbol to insert as a string value.

symbol_scale

Scale of the symbol as an integer value.

Return Values

List consisting of the newly inserted node, the new first broken entity, the second broken entity.

Examples

Command: (setq ename (entsel))
Command: (mun-utl-capnode 43 (car ename) (cadr ename) "F" "Y" "Y" "_WGTVALVE" 1.0)

MUN-UTL-CHECKDB

Checks whether user is connected to database. If not, then displays logon screen and resets application settings.

(mun-utl-checkdb)

Return Values

None

Examples

Command: (mun-utl-checkdb)

MUN-UTL-CHECKLOCK

Checks if object is locked – if not then automatically locks object.

(mun-utl-checklock ename)

Arguments

ename

Name of the entity that you wish to lock.

Return Values

1 if successful.

Examples

Command: (setq ename (car (entsel)))
Command: (mun-utl-checklock ename)
1

MUN-UTL-CHECKLOCKSSET

Checks if selection set is locked – if not then automatically lock objects in selection set.

(mun-utl-checklocksset selset locktype)

Arguments

selset

Name of selection set.

locktype

If this is set to “LOCKALL” then this will prompt you if all object in selection set could not be locked.

If this is set to any other string value then you will not be prompted if all objects could not be locked.

Return Values

T always if locktype is set to any other string.

OR

T if locktype is set to “LOCKALL” and all objects could be locked otherwise returns nil.

Examples

Command: (setq selset (ssget))
Command: (mun-utl-checklocksset selset "")
T

MUN-UTL-CLEARMOP

Clears the last MOP (Munsys Objects Processed) list for any Munsys commands

(mun-utl-clearmop)

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-utl-clearmop)

MUN-UTL-CLRERRORMSG

Clears all error messages

(mun-utl-clrerrormsg)

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-utl-clrerrormsg)

MUN-UTL-D2R

Converts degrees to radians.

(mun-utl-d2r angle)

Arguments

angle

Angle in degrees as a real value to be converted to radians.

Return Values

Real value, converted angle in radians.

Examples

Command: (mun-utl-d2r 90.0)
1.5708

MUN-UTL-DISPLAYANGLE

Sets angle to display text/symbol correctly.

(mun-utl-displayangle angle)

Arguments

angle

Angle in radians as a real value.

Return Values

Returns angle, in degrees, as a string value.

Examples

Command: (mun-utl-displayangle 3.24)
"5.638"

MUN-UTL-DRAWAPPROXLINE

Constructs an approximate line and offsets line according to offset_distance.

(mun-utl-drawapproxline offset_distance)

Arguments

offset_distance

This is the default distance to offset the constructed line.

Return Values

Returns entity name of newly constructed object.

Examples

Command: (mun-utl-drawapproxline 1.0)
Specify approximate start point:
Specify approximate end point:
Select segment for offset:
Specify point on side to offset:
Specify offset distance <1.000>:
<Entity name: 7efdec28>

MUN-UTL-DRAWPOLYLINE

Constructs a freehand and prompts for offset is so specified.

(mun-utl-drawpolyline offset offset_distance)

Arguments

offset

Strng indicating whether to prompt for offset or not, valid values are “Y” to offset and “N” to not offset.

offset_distance

This is the default distance to offset the constructed polyline.

Return Values

Returns entity name of newly constructed object.

Examples

Command: (mun-utl-drawpolyline "Y" 1.0)
Specify points or [Offset segment]:
Specify next point:
Specify next point:
Specify next point:
Specify point on side to offset:
Specify offset distance <1.000>:
<Entity name: 7efdec10>

MUN-UTL-EDITBOX

Displays edit dialog box

(mun-utl-editbox text_value dialog_description text_type edit_width)

Arguments

text_value

Default value to display in edit box as a string value.

dialog_description

Description of dialog as a string value.

text_type

Type of text to check, valid values are:

ALPHA - Alphanumeric

NUM - Number

edit_wdith

Maximum number of characters to check for as an integer value.

Return Values

String of the current value listed in the edit box.

Examples

Command:

(mun-utl-editbox "Current Value" "My Dialog" "ALPHA" 10)

"1234567890"

MUN-UTL-ENTSEL

Selects entity based on filter.

(mun-utl-entsel element_no element_value lock_entity prompt_string error_string)

Arguments

element_no

Element number to filter on as integer value.

element_value

Element value to filter on as a list, string, integer or real value.

lock_entity

Indicates whether or not to automatically lock object. Valid values are:

T - True (to lock object)

nil – nil (to not lock object)

prompt_string

Displays alternative prompt as a string value.

error_string

Displays alternative prompt as the error prompt as a string value.

Return Values

List consisting of the entity name and point on entity selected or nil otherwise.

Examples

Command: (mun-utl-entsel 95 45 nil "\nSelect water pipe: "
"\nObject selected is not a water pipe...")
(<Entity name: 7efd8458> (-75834.1 -2.88238e+006 0.0))
Command: (mun-utl-entsel 0 (list "LINE" "LWPOLYLINE" "MUNLINE"
"MUNPOLY" "MPOLYGON") nil "\nSelect first segment for dimension: "
"\nCannot determine segment from object selected..."))
(<Entity name: 7efd8458> (-75834.1 -2.88238e+006 0.0))

MUN-UTL-FILEEXISTS

Checks whether file exists.

(mun-utl-fileexists filename)

Arguments

filename

File name to check for as a string value.

Return Values

Returns T if file exists or nil otherwise.

Examples

Command: (mun-utl-fileexists "c:\\test.txt")
nil

MUN-UTL-FILEOPEN

Checks if file is already open.

(mun-utl-fileopen filename)

Arguments

filename

File name to check for as a string value.

Return Values

Returns T if file is open or nil otherwise.

Examples

Command: (mun-utl-fileopen "c:\\test.txt")
nil

MUN-UTL-GETCOLUMNPREC

Returns the column precision for a specific column in a table.

(mun-utl-getcolumnprec table_name column_name)

Arguments

table_name

Table name as a string value.

column_name

Column name as a string value whose precision you wish to retrieve.

Return Values

Precision of column as an integer.

Examples

Command: (mun-utl-getcolumnprec "SP_WATNODE" "NODE_ELEV")
3

MUN-UTL-GETCOLUMNWIDTH

Returns the column width for a specific column in a table.

(mun-utl-getcolumnwidth table_name column_name)

Arguments

table_name

Table name as a string value.

column_name

Column name as a string value whose precision you wish to retrieve.

Return Values

Width of column as an integer.

Examples

Command:

(mun-utl-getcolumnwidth "SP_WATNODE" "COMMENTS")

150

MUN-UTL-GETCURRENTHELP

Returns the current application help file – used in conjunction with MUN-UTL-SETCURRENTAPP.

(mun-utl-getcurrenthelp)

Return Values

Returns the current application help file as a string value.

Examples

Command: (mun-utl-getcurrenthelp)
"munrms.chm"

MUN-UTL-GETELEMENT

Gets element value from an entity.

(mun-utl-getelement element_id ename)

Arguments

element_id

Element number to retrieve value for.

ename

Entity name to retrieve value from.

Return Values

Current value of element number.

Examples

This example retrieves the layer name of the selected entity
Command: (setq ename (car (entsel)))
Command: (mun-utl-getelement 8 ename)
"PARCEL_C"

MUN-UTL-GETMOP

Gets the MOP (Munsys Objects Processed) list that was handled by the last Munsys command.

(mun-utl-getmop)

Return Values

Returns a list containing a list of ((created)(modified)(deleted)) objects processed by the last Munsys command.

Examples

Command: (mun-utl-getmop)
((<Entity name: 7e909230> <Entity name: 7e909228>)(nil)(<Entity name: 7e909548>))

MUN-UTL-GETMUNVAR

Returns the value of a Munsys system variable.

(mun-utl-getmunvar system_var)

Arguments

system_var

Name of the system variable whose value you wish to retrieve.

Return Values

Returns the value of the system variable as a string value.

Examples

Command: (mun-utl-getmunvar "MUNTIPSHOW")
"ON"

MUN-UTL-GETPLEN

Gets the total length for a given list of points.

(mun-utl-getplen point_list)

Arguments

point_list

List consisting of points.

Return Values

Length of point_list (this is the total geometry length and not the number of elements).

Examples

Command: (setq ename (car (entsel)))
Command: (setq ptlst (mun-utl-getplst ename))
Command: (mun-utl-getplen ptlst)
14.6197

MUN-UTL-GETPLST

Gets a point list from either a LINE or LWPOLYLINE object.

(mun-utl-getplst ename)

Arguments

ename

Entity name whose points you wish to retrieve.

Return Values

List of points.

Examples

Command: (setq ename (car (entsel)))
Command: (setq ptlst (mun-utl-getplst ename))
((-75832.8 -2.8824e+006 0.0) (-75828.3 -2.88239e+006 0.0) (-75823.7 -2.8824e+006 0.0) (-75821.5 -2.88239e+006 0.0)
(-75819.5 -2.88239e+006 0.0))

MUN-UTL-GETREGVALUE

Gets the registry key value located under Munsys registry path [[HKEY_CURRENT_USER\Software\Open Spatial\Munsys[version]\Applications\Options key]]

(mun-utl-getregvalue reg_subpath reg_key)

Arguments

reg_subpath

The sub key path to retrieve registry key value located under the Munsys registry path.

reg_key

Registry key to retrieve.

Return Values

Returns the registry key value as a string.

Examples

Command: (mun-utl-getregvalue "" "InitEnvironment")
"1"

MUN-UTL-GETSUBSEGMENT

Gets the sub segment of an object closest to the point selected. The only valid objects are LINE, LWPOLYLINE, MUNLINE, MUNPOLY,MPOLYGON.

(mun-utl-getsubsegment ename point_on_segment)

Arguments

ename

Name of the entity whose segment you wish to retrieve.

point_on_segment

Point on the segment for which you want to retrieve.

Return Values

List consisting of start point of segment, end point of segment and angle of segment in radians or nil otherwise

Examples

Command: (setq ename (entsel))
Command: (mun-utl-getsubsegment (car ename) (cadr ename))
((-75807.3 -2.88248e+006 0.0) (-75753.0 -2.88254e+006 0.0) 5.46795)

MUN-UTL-GETTABLENAME

Returns the spatial table name that corresponds to the mun_id.

(mun-utl-gettablename mun_id)

Arguments

mun_id

Munsys ID of the table name you wish to retrieve.

Return Values

Returns spatial table name as a string.

Examples

Command: (mun-utl-gettablename 3)
"SP_PARCEL"

MUN-UTL-GETTEXTJUST

Gets the text justification for a text object.

(mun-utl-gettextjust ename)

Arguments

ename

Name of the entity whose tag justification you wish to retrieve.

Return Values

Returns one of the following values as a string or nil otherwise:

L – Left

C – Center

M – Middle

R – Right

TL – Top Left

TC – Top Center

TR – Top Right

ML – Middle Left

MC – Middle Center

MR – Middle Right

BL – Bottom Left

BC – Bottom Center

BR – Bottom Right

Examples

Command: (mun-utl-gettextjust (car (entsel)))
"TL"

MUN-UTL-GETVERSIONKEY

Gets the current Munsys version number.

(mun-utl-getversionkey)

Arguments

None

Return Values

Current version number as a string value of the major version concatenated with the minor version number.

Examples

Command: (mun-utl-getversionkey)
"14.2"

MUN-UTL-GETWORKCOLOR

Returns the current working color.

Important This function was deprecated in Munsys 9.2 and replaced with the Munsys system variable MUNCOLINTEGRITY - see MUN-UTL-GETMUNVAR and MUN-UTL-SETMUNVAR for getting and setting this variable respectively.

(mun-utl-getworkcolor)

Return Values

Working color as an integer value.

Examples

Command: (mun-utl-getworkcolor)
3

MUN-UTL-GSCREFRESH

Refreshes the GSC settings with latest values in database. Function is only necessary if the GSC values are updated via SQL and these updated settings needs to be reflected in the current session.

(mun-utl-gscrefresh)

Arguments

None

Return Values

T if successful or Nil otherwise.

Examples

Command: (mun-utl-gscrefresh)
T

MUN-UTL-MUNBREAK

Breaks Munsys object and returns entity names of the 2 new objects.

(mun-utl-munbreak ename point_on_ent)

Arguments

ename

Name of the entity that you wish to break.

point_on_ent

Point on entity - also indicates break point.

Return Values

List consisting of 2 new entity names.

Examples

Command: (setq ename (entsel))
Command: (mun-utl-munbreak (car ename) (cadr ename))
(<Entity name: 7eb88968> <Entity name: 7eb88970>)

MUN-UTL-PROMPT-CLEAR

Clears the current prompt values.

(mun-utl-prompt-clear)

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-utl-prompt-clear)
T

MUN-UTL-PROMPT-SET

Sets the prompt values for any Munsys commands.

(mun-utl-prompt-set prompt_type prompt_value)

Arguments

prompt_type

The type of prompt to set as a string value. Valid options are:

VALUE – value for the prompt

PROMPT – alternative prompt

ENTSEL – selection entity

prompt_value

Value of the prompt. The following types are valid:

VALUE – string,real,integer,point

PROMPT – string

ENTSEL – (list consisting of entity name and point on entity)

e.g. (<Entity name: 7eb2bda8> (-75787.7 -2.8825e+006 0.0))

Return Values

None

Examples

Command: (mun-utl-prompt-clear)
Command: (mun-utl-prompt-set "ENTSEL" (list CENT (list 0.0 0.0 0.0)))
Command: (mun-utl-prompt-set "VALUE" "Join")
Command: (mun-utl-prompt-set "PROMPT" "Specify join mode [Extended/Linked]: ")
Command: (command "mungedit" PAUSE PAUSE PAUSE "eXit" "")

MUN-UTL-R2D

Converts radians to degrees.

(mun-utl-r2d angle)

Arguments

angle

Angle in radians as a real value to be converted to degrees.

Return Values

Real value, converted angle in degrees.

Examples

Command: (mun-utl-r2d 1.57)
89.9544

MUN-UTL-SETCURRENTAPP

Sets the internal current application setting. This function is mainly used with MUN-UTL-GETCURRENTHELP in order to specify which is the current application and which help file is now appropriate.

(mun-utl-setcurrentapp app_name)

Arguments

app_name

Application name of the current application.

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-utl-setcurrentapp "CMS")
T

MUN-UTL-SETCURRENTLAYER

Sets the current layer – if layer does not exists then it automatically creates a new layer with the default color of layer_color.

(mun-utl-setcurrentlayer layer_name layer_color)

Arguments

layer_name

Name of the layer to set or create as a string value.

layer_color

Color of new layer as an integer value.

Return Values

None

Examples

Command: (mun-utl-setcurrentlayer "MYLAYER" 1)

MUN-UTL-SETDEBUG

Sets the SQL debug statement. This allows you to view all SQL statements executed on the database.

(mun-utl-setdebug debug_state)

Arguments

debug_state

Integer – 1 for on, 0 for off

Return Values

Returns T if successful or nil otherwise.

Examples

Command: (mun-utl-setdebug 1)
Debug Mode switched On.T

MUN-UTL-SETELEMENT

Sets element value for an entity with new value.

(mun-utl-setelement element_id ename new_value)

Arguments

element_id

Element number to set value for.

ename

Entity name to set value for.

new_value

New value for element number.

Return Values

An association list containing the updated entity definition of *ename*.

Examples

Command: (entmod (mun-utl-setelement 8 ename "PARCEL"))

MUN-UTL-SETMUNVAR

Sets the value of a Munsys system variable.

(mun-utl-setmunvar system_var value)

Arguments

system_var

Name of the system variable whose value you wish to retrieve.

value

The value to set the system variable to as a string value.

Return Values

T if successful or nil otherwise.

Examples

Command: (mun-utl-setmunvar "MUNTIPSHOW" "ON")
T

MUN-UTL-SETREGVALUE

Sets the registry key value located under Munsys registry path [[HKEY_CURRENT_USER\Software\Open Spatial\Munsys[version]\Applications\Options key]]

(mun-utl-setregvalue reg_subpath reg_key new_value)

Arguments

reg_subpath

The sub key path to set the registry key value located under the Munsys registry path.

reg_key

Registry key to set.

new_value

The new registry value to set as a string value.

Return Values

T if successful or nil otherwise.

Examples

Command: (mun-utl-setregvalue "" "InitEnvironment" "1")
"T"

MUN-UTL-SSGET

Select entities and locks objects if required.

(mun-utl-ssget filter_list munsys_filter lock_entity prompt_string)

Arguments

filter_list

ssget filter list as standard autocad filter list.

munsys_filter

The munsys_filter allows you to filter on values not available with the standard (ssget) function such as the symbol/block names on MUNPOINT objects, for example:

(mun-utl-ssget '((0. "MUNLABEL,MUNPOINT,MUNLINE,MUNPOLY")(95 . 23))(list 2 (list "_SOVALVE" "_SCLVALVE")) T "\nSelect open or closed valves...")

lock_entity

Indicates whether or not to automatically lock objects. Valid values are:

T - True (to lock object)

nil – nil (to not lock object)

prompt_string

Displays alternative prompt as a string value.

Return Values

Selection set or nil otherwise.

Examples

Command: (mun-utl-ssget '((0 . "MUNLINE")(95 . 46)) nil T "\nSelect service connections...")

MUN-UTL-SYSDATE

Gets the current date according to the regional settings.

(mun-utl-sysdate)

Arguments

None

Return Values

Date as a string value formatted according to the regional setting.

Examples

Command: (mun-utl-sysdate)
"16 March 2005"

MUN-UTL-SYSTIME

Gets the current time according to the regional settings.

(mun-utl-systime)

Arguments

None

Return Values

Time as a string value formatted according to the regional setting.

Examples

Command: (mun-utl-systime)
"9:25:14 AM"

MUN-UTL-ZOOMENTITY

Zooms to the extents of an entity.

(mun-utl-zoomentity ename scale)

Arguments

ename

Name of the entity to zoom extents to.

scale

Zoom scale factor to use as a real value.

Return Values

None

Examples

Command: (mun-utl-zoomentity (entlast) 1.0)

MUNPOINT FUNCTIONS

MUN-MUNPOINT-GETNUMOFPOINTS

Gets the number of points in a MUNPOINT object.

(mun-munpoint-getnumofpoints ename)

Arguments

ename

Entity name of object to retrieve number of points from.

Return Values

Number of points as an integer value.

Examples

Command: (setq ename (car (entsel)))
Command: (mun-munpoint-getnumofpoints ename)
1

MUN-MUNPOINT-GETPOINT

Gets the insertion point for the specified point of a MUNPOINT object.

(mun-munpoint-getpoint ename point_num)

Arguments

ename

Entity name of object to retrieve point from.

point_num

Point position – 0 based index.

Return Values

Point

Examples

Command: (setq ename (car (entsel)))
Command: (mun-munpoint-getpoint ename 0)
(-75804.1 -2.88235e+006 0.0)

MUN-MUNPOINT-GETSYMANGLE

Gets the symbol angle for a MUNPOINT object.

(mun-munpoint-getsymangle ename)

Arguments

ename

Entity name of object to retrieve symbol angle from.

Return Values

Symbol angle as a real value in radians.

Examples

Command: (setq ename (car (entsel)))
Command: (mun-munpoint-getsymangle ename)
0.79

MUN-MUNPOINT-GETSYMNAME

Gets the symbol name for a MUNPOINT object.

(mun-munpoint-getsymname ename)

Arguments

ename

Entity name of object to retrieve symbol name from.

Return Values

Symbol Name as a string value.

Examples

Command: (setq ename (car (entsel)))
Command: (mun-munpoint-getsymname ename)
"_WGTVALVE"

MUN-MUNPOINT-GETSYMSCALE

Gets the symbol scale for a MUNPOINT object.

(mun-munpoint-getsymscale ename)

Arguments

ename

Entity name of object to retrieve symbol scale from.

Return Values

Symbol scale as a real value.

Examples

Command: (setq ename (car (entsel)))
Command: (mun-munpoint-getsymscale ename)
1.0

MUN-MUNPOINT-SETSYMANGLE

Sets the symbol angle for a MUNPOINT object.

(mun-munpoint-setsymangle ename symbol_angle)

Arguments

ename

Entity name of object to update.

symbol_angle

Symbol angle as a real value in radians.

Return Values

None

Examples

Command: (setq ename (car (entsel)))
Command: (mun-munpoint-setsymangle ename 1.57)

MUN-MUNPOINT-SETSYMNAME

Sets the symbol name for a MUNPOINT object.

(mun-munpoint-setsymname ename symbol_name)

Arguments

ename

Entity name of object to update.

symbol_name

Symbol name as a string value.

Return Values

None

Examples

Command: (setq ename (car (entsel)))
Command: (mun-munpoint-setsymname ename "_WGTVALVE")

MUN-MUNPOINT-SETSYMSCALE

Sets the symbol scale for a MUNPOINT object.

(mun-munpoint-setsymscale ename symbol_scale)

Arguments

ename

Entity name of object to update.

symbol_scale

Symbol scale as a real value.

Return Values

None

Examples

Command: (setq ename (car (entsel)))
Command: (mun-munpoint-setsymscale ename 1.0)

MUNLINE FUNCTIONS

MUN-MUNLINE-CHANGEDIRECTION

Changes the direction of a MUNLINE object.

(mun-munline-changedirection ename)

Arguments

ename

Entity name of object to update.

Return Values

T if successful or nil otherwise.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-changedirection (car ename))
T

MUN-MUNLINE-ENDPOINT

Gets the end point of a MUNLINE object.

(mun-munline-endpoint ename)

Arguments

ename

Entity name of object to retrieve end point from.

Return Values

End point of MUNLINE as a point.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-endpoint (car ename))
(-76117.4 -2.88262e+006 0.0)

MUN-MUNLINE-GETNUMOFSEGMENTS

Gets the number of segments of a MUNLINE object.

(mun-munline-getnumofsegments ename)

Arguments

ename

Entity name of object to retrieve number of segments from.

Return Values

Returns an integer value indicating the number of segments in a MUNLINE object.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-getnumofsegments (car ename))
1

MUN-MUNLINE-GETSEGMENT

Gets the segment number of the selected segment on a MUNLINE object.

(mun-munline-getsegment ename point_on_segment)

Arguments

ename

Entity name of object to retrieve segment from.

point_on_segment

Point on segment to retrieve.

Return Values

Returns 0 based segment number as an integer value.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-getsegment (car ename) (cadr ename))
0

MUN-MUNLINE-MOVEPOINT

Moves a point on a MUNLINE object to a new specified point.

(mun-munline-movepoint ename segment_number point_to_move new_point)

Arguments

ename

Entity name of object to update.

segment number

Segment number containing point to move - 0 based index.

point_to_move

Point closest to point that you wish to move.

new_point

Position of new point.

Return Values

None

Examples

Command: (setq ename (entsel))
Command: (setq oldpt (getpoint "\nSpecify point to move: "))
Command: (setq newpt (getpoint "\nSpecify new point: "))
Command: (mun-munline-movepoint (car ename) 3 oldpt newpt)

MUN-MUNLINE-SEGMENT-DISTFROMEND

Gets the distance that a point lying on a segment is from the end point of that segment. The distance calculated is the distance along the segment from the point indicated to the end of the specified segment.

(mun-munline-segment-distfromend ename
segment_number point_on_segment)

Arguments

ename

Entity name of object.

segment_number

Number of segment that point lies on – 0 based Index.

point_on_segment

Point on segment.

Return Values

Distance from end point of segment as a real value.

Examples

Command: (setq ename (entsel))
Command: (setq segnum (mun-munline-getsegment (car ename) (cadr ename)))
Command: (mun-munline-segment-distfromend (car ename) SEGNUM (osnap (cadr ename) "_nea"))
51.459

MUN-MUNLINE-SEGMENT-DISTFROMSTART

Gets the distance that a point lying on a segment is from the start point of that segment.

(mun-munline-segment-distfromstart ename segment_number

point_on_segment)

Arguments

ename

Entity name of object.

segment_number

Number of segment that point lies on.

point_on_segment

Point on segment.

Return Values

Distance from start point of segment as a real value.

Examples

Command: (setq ename (entsel))
Command: (setq segnum (mun-munline-getsegment (car ename) (cadr ename)))
Command: (mun-munline-segment-distfromstart (car ename) SEGNUM (osnap (cadr ename) "_nea"))
51.459

MUN-MUNLINE-SEGMENT-ENDPOINT

Gets the end point of a segment

(mun-munline-segment-endpoint ename segment_number)

Arguments

ename

Entity name of object whose segments end point you wish to retrieve.

segment_number

Number of segment whose end point you wish to retrieve – 0 based index.

Return Values

End point of segment as a point.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-segment-endpoint (car ename) 0)
(-75748.0 -2.88256e+006 0.0)

MUN-MUNLINE-SEGMENT-GETPOINTAT

Returns a specific point within a given segment.

(mun-munline-segment-getpointat ename segment_number point_num)

Arguments

ename

Entity name of object containing point to retrieve.

segment_number

Number of segment containing point to retrieve – 0 based index.

point_num

Position of point within segment – 0 based index.

Return Values

Point at given position.

Examples

Example returns the first point of the first segment:

Command: (setq ename (entsel))
Command: (mun-munline-segment-getpointat (car ename) 0 0)
(-75884.1 -2.88254e+006 0.0)

MUN-MUNLINE-SEGMENT-GETVERTICES

Gets the vertices for a MUNLINE object.

(mun-munline-segment-getvertices ename segment_number)

Arguments

ename

Name of the entity whose vertices you wish to retrieve.

segment_number

Number of the segment you wish to retrieve the vertices for. The segment_number parameter has a 0 based index.

Return Values

List containing a list of vertices and a list of bulges.

Examples

Command: (setq ename (entsel))
Command: (setq segment_number (mun-munline-getsegment (car ename) (cadr ename)))
Command: (mun-munline-segment-getvertices (car ename) segment_number)
(((-76328.718 -2883468.056 0.0)(-76313.518 -2883472.736 0.0) (-76284.888 - 2883445.536 0.0)(-76306.118 -2883429.676 0.0)) (0.0 1.25 0.0 0.0))

MUN-MUNLINE-SEGMENT-LENGTH

Gets the geometry length of a MUNLINE segment.

(mun-munline-segment-length ename segment_number)

Arguments

ename

Entity name of object whose segment length you wish to retrieve.

segment_number

Number of segment whose length you wish to retrieve – 0 based index.

Return Values

Length of segment as a real value.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-segment-length (car ename) 0)
91.1138

MUN-MUNLINE-SEGMENT-MIDPOINT

Gets the mid point of a segment.

(mun-munline-segment-midpoint ename segment_number)

Arguments

ename

Entity name of object whose segments mid point you wish to retrieve.

segment_number

Number of segment whose mid point you wish to retrieve – 0 based index.

Return Values

Mid point of segment as a point.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-segment-midpoint (car ename) 0)
(-75748.0 -2.88256e+006 0.0)

MUN-MUNLINE-SEGMENT-NUMOFPOINTS

Gets the number of points within a MUNLINE segment.

(mun-munline-segment-numofpoints ename segment_number)

Arguments

ename

Entity name of object.

segment_number

Number of segment whose points you wish to count – 0 based Index.

Return Values

Number of points on a segment as an integer value.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-segment-numofpoints (car ename) 0)
2

MUN-MUNLINE-SEGMENT-STARTPOINT

Gets the start point of a segment.

(mun-munline-segment-startpoint ename segment_number)

Arguments

ename

Entity name of object whose segments start point you wish to retrieve.

segment_number

Number of segment whose start point you wish to retrieve – 0 based index.

Return Values

Start point of segment as a point.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-segment-startpoint (car ename) 0)
(-75748.0 -2.88256e+006 0.0)

MUN-MUNLINE-SETENDPOINT

Sets a new end point for a MUNLINE object.

(mun-munline-setendpoint ename segment_number point)

Arguments

ename

Entity name of object to update.

segment_number

Segment number of end point to set - 0 based index.

point

New end point.

Return Values

None

Examples

Command: (setq ename (entsel))
Command: (setq pt (getpoint "\nSpecify new end point: "))
Command: (mun-munline-setendpoint (car ename) 0 pt)

MUN-MUNLINE-SETSTARTPOINT

Sets a new start point for a MUNLINE object.

(mun-munline-setstartpoint ename segment_number point)

Arguments

ename

Entity name of object to update.

segment_number

Segment number of start point to set - 0 based index.

point

New start point.

Return Values

None

Examples

Command: (setq ename (entsel))
Command: (setq pt (getpoint "\nSpecify new start point: "))
Command: (mun-munline-setstartpoint (car ename) 0 pt)

MUN-MUNLINE-STARTPOINT

Gets the start point of a MUNLINE object.

(mun-munline-startpoint ename)

Arguments

ename

Entity name of object to retrieve start point from.

Return Values

Start point of MUNLINE as a point.

Examples

Command: (setq ename (entsel))
Command: (mun-munline-startpoint (car ename))
(-75839.5 -2.88239e+006 0.0)

MUNPOLY FUNCTIONS

MUN-MUNPOLY-COPYOBJECT

Copies a MUNPOLY object to a new Munsys Spatial Type based on the Munsys ID.

(mun-munpoly-copyobject ename mun_id)

Arguments

ename

Entity name of object to copy.

mun_id

Munsys ID of new spatial type to create for new object as an integer value.

Return Values

None

Examples

Command: (setq ename (entsel))
Command: (mun-munpoly-copyobject (car ename) 1)

MUN-MUNPOLY-GETBOUNDARY

Returns the boundary number to the point specified on the boundary.

(mun-munpoly-getboundary ename point_on_boundary)

Arguments

ename

Entity name of object containing boundary number to retrieve.

point_on_boundary

Point on the boundary.

Return Values

Boundary number as an integer value – 0 based index.

Examples

Command: (setq ename (entsel))
Command: (mun-munpoly-getboundary (car ename) (cadr ename))
0

MUN-MUNPOLY-GETNUMOFPOLYS

Gets the number of boundary polygons of a MUNPOLY object.

(mun-munpoly-getnumofpolys ename)

Arguments

ename

Entity name of object.

Return Values

Number of outer boundaries as an integer value.

Examples

Command: (setq ename (entsel))
Command: (mun-munpoly-getnumofpolys (car ename))
1

MUN-MUNPOLY-GETNUMTOTALPOLYS

Gets the total number of polygons including islands of a MUNPOLY object.

(mun-munpoly-getnumtotalpolys ename)

Arguments

ename

Entity name of object.

Return Values

Number of all boundaries as an integer value.

Examples

Command: (setq ename (entsel))
Command: (mun-munpoly-getnumtotalpolys (car ename))
1

MUN-MUNPOLY-GETVERTICES

Gets the vertices for a MUNPOLY object.

(mun-munpoly-getvertices ename poly_num)

Arguments

ename

Name of the entity whose vertices you wish to retrieve.

poly_num

Number of the polygon(s) you wish to retrieve the vertices for. The poly_num parameter has a 0 based index.

Return Values

List containing a list of vertices and a list of bulges.

Examples

Command: (setq poly_num (mun-munpoly-getboundary ename (getpoint)))
Command: (mun-munpoly-getvertices ename poly_num)
(((-76328.718 -2883468.056 0.0)(-76313.518 -2883472.736 0.0) (-76284.888 -2883445.536 0.0)(-76306.118 -2883429.676 0.0))(0.0 1.25 0.0 0.0))

MUN-MUNPOLY-ISPOINTINSIDE

Determines whether a point lies within a MUNPOLY object. This function only checks whether the point lies within the disjoint boundaries and not the islands within the disjoint boundaries.

(mun-munpoly-ispointinside ename point)

Arguments

ename

Entity name of object to that contains point to check.

point

Point to check if it lies within a MUNPOLY object.

Return Values

T if point lies within otherwise nil.

Examples

Command: (setq ename (entsel))
Command: (setq pt (getpoint "\nSpecify point: "))
Command: (mun-munpoly-ispointinside (car ename) pt)
T

Appendix

Munsys commands

The following table contains a list of the Munsys commands:

Command	Description
CMSCHKINTEGRITY	Cadastral Object Integrity
DMSCHKINTEGRITY	Drainage Object Integrity
DMSCHKNETWORK	Drainage Network Integrity
EMSCHKINTEGRITY	Electricity Object Integrity
EMSCHKNETWORK	Electricity Network Integrity
CRSCHKINTEGRITY	Cable Route Object Integrity
CRSCHKNETWORK	Cable Route Network Integrity
CFSCHKINTEGRITY	Cable Fiber Object Integrity
MUNABOUT	Show about dialog
MUNADDSPVIEW	Adds a spatial table as a view
MUNANNOTATE	Annotate
MUNAPSETTINGS	Application Settings
MUNATMANAGE	Create and maintain attribute templates
MUNBREAK	Break
MUNBROWSE	Table Browse
MUNBROWSELOCK	Browse locked objects
MUNCHAMFER	Chamfer
MUNCHPROP	Change Munsys display properties
MUNCONNECT	Connect
MUNCONVERT	AutoCAD objects to Munsys objects
MUNDEL	Delete objects
MUNDISCONNECT	Disconnect

MUNEDITLTATTR	Edit linked table attributes
MUNEDITSPATTR	Edit spatial attributes
MUNFILLET	Fillet
MUNFRACTURE	Fracture
MUNGEDIT	Geometry Edit
MUNGOOGLEEARTH	Creates a KML file to display information in Google Earth
MUNGOOGLEMAPS	Creates a KML file to display information in Google Maps
MUNGSCADDOBJ	Adds objects to a GSC
MUNGSCCREATE	Create GSC
MUNGSCSETTINGS	GSC Settings
MUNGSCSHOW	Show GSC
MUNGSCTOCAD	Convert the current GSC to a polyline or point object.
MUNGSCZOOM	Zoom to GSC
MUNHELMERT	Helmert Transformation
MUNIMGBOUNDARY	Associates images/PDF/DWG with boundary objects
MUNIMGGEOREF	Allows an image/PDF/DWG to be selected, scaled or rotated
MUNIMGLOAD	Loads an image/PDF/DWG associated with a Geo-scan boundary object
MUNIMGUNLOAD	Unloads an image/PDF/DWG associated with a Geo-scan boundary object
MUNIMGHYPERLINK	Attaches a hyperlink to a selected Geo-scan boundary object
MUNIMGREPOSITION	Repositions a selected Geo-scan boundary object
MUNIMGRESYNC	Re-syncs a selected Geo-scan boundary object
MUNINFOPALETTE	Shows the Munsys Info Palette
MUNINFOSELECT	Toggles the behavior of the auto-select option in the Info Palette
MUNINTEGMODE	Integrity Color Mode Use/Not Use
MUNINTERRDEL	Erase Integrity Markers
MUNINTERRVIEW	Integrity Error View
MUNLBLPLACE	Place Label
MUNLEGEND	Legend
MUNLOCK	Lock objects

MUNLTMANAGE	Create and maintain link templates
MUNMDEXPORT	Export metadata associated with one or more spatial tables
MUNNOTEPLACE	Place Note
MUNOBJECT	Convert object to Munsys object
MUNOPTIONS	Munsys Options
MUNPASSWORD	Change current Oracle password.
MUNPOLYBLDCAD	Build CAD polygon
MUNPOLYBLDNEW	Build New Munsys Polygon
MUNPOLYREBUILD	Rebuild Existing Munsys Polygon
MUNPOLYVALIDATE	Validate Polygon
MUNPOSTOBJS	Post objects to database
MUNQRYEXEC	Query Run (Execute on command line)
MUNQRYEXECGRP	Runs all Queries in Category (Execute on command line)
MUNQRYMULTIPLE	Runs checked queries from the Query Palette
MUNQRYOPTIONS	Query Preferences Options
MUNQRYPALETTE	Shows the Query Palette
MUNQRYREFRESH	Refresh queried objects
MUNQRYRUN	Query Run
MUNREMOVESPVIEW	Removes a spatial view
MUNSETCOORDSYS	Set Coordinate System
MUNSHOWDBEXTENTS	Show database extents
MUNSHOWINFO	Show Info Dialog
MUNSHOWLLGRID	*Show LL Grid (Unsupported)*
MUNSHOWLOCK	Show locked objects
MUNTAG	Place Tag
MUNTEDIT	Tag Edit (Command line)
MUNTMOVE	Move Tag
MUNTRIM	Trim
MUNTVALUE	Edit Tag Value
MUNUNDEL	Undelete Object

MUNUNLOCK	Unlock objects
MUNWEED	Weed
MUNZOOMDBEXTENTS	Zoom to database extents
MUNZOOMINTEGCIRC	Zoom to Integrity Circles
RMSCHKINTEGRITY	Roads Object Integrity
RMSCHKNETWORK	Roads Network Integrity
SDMCAPSELECT	Spatial Data Manager, Convert Objects to Spatial Objects
SDMCHKINTEGRITY	Spatial Data Manager Object Integrity
SMSCHKINTEGRITY	Sewer Object Integrity
SMSCHKNETWORK	Sewer Network Integrity
WMSCHKINTEGRITY	Water Object Integrity
WMSCHKNETWORK	Water Network Integrity

Table 1 Munsys commands