A
Selection Compatibility Attributes

This appendix lists all the selection compatibility attributes in alphabetical order. Appendix B, *Selection Compatibility Procedures and Macros*, lists the selection compatibility procedures. XView Version 3 supports a newer selection mechanism that implements selections using the `SELECTION` package. The selection mechanism that was available in older versions of XView is supported in XView Version 3; its attributes are shown in this appendix and the mechanism is described in Appendix A, *The Selection Service*, in the *XView Programming Manual*. All of these compatibility attributes use the `SELN_`-prefix. These attributes do *not* work with the XView `SELECTION` package.

Each selection attribute's description is in the format below.

Selection Compatibility Attributes

INTRODUCTION

This field provides a brief description of the attribute.

Argument: This field shows the first programmer-supplied value associated with the attribute. If an attribute has multiple values, then the type of each value is shown in multiple Argument fields.

Default: The default field shows the default value for the attribute.

Procs: The procedures field shows the procedures that are valid for the attribute. Note: These attributes are *not* valid for `Selection` objects.

SELN_REQ_BYTESIZE

Specifies the number of bytes in the selection.

Argument: `int`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_COMMIT_PENDING_DELETE

Instructs the replier to delete any secondary selection made in pending delete mode.

Argument: None
Default: None

SELN_REQ_CONTENTS_ASCII

Specifies a NULL-terminated list of 4-byte words containing the selection's ASCII contents. If the last word of the contents is not full (including NULL terminator for the string), it is NULL-padded.

Argument: `char *`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_CONTENTS_PIECES

Specifies that the value is a NULL-terminated list of 4-byte words containing the selection's contents described in the `textsw`'s piece-table format.

Argument: `char *`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_DELETE

Instructs the holder of the selection to delete the contents of the selection from its window (used only by text subwindows).

Argument: `void`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_END_REQUEST

Returns an error for failed or unrecognized requests.

Argument: `void`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_FAKE_LEVEL

Gives a level to which the selection should be expanded before processing the remainder of this request. The original level should be maintained on the display, however, and restored as the true level on completion of the request.

Argument: `int`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_FILE_NAME

Specifies a NULL-terminated list of 4-byte words. Contains the name of the file which holds the *shelf* selection.

Argument: `char *`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_FIRST

Gives the number of bytes that precede the first byte of the selection.

Argument: `int`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_FIRST_UNIT

Gives the number of units of the selection's current level (line, paragraph, etc.) which precede the first unit of the selection.

Argument: `int`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_LAST

Gives the byte index of the last byte of the selection.

Argument: `int`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_LAST_UNIT

Gives the unit index of the last unit of the selection at its current level.

Argument: `int`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_LEVEL

Gives the current level of the selection.

Argument: `int`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_RESTORE

Instructs the replier to restore the selection referred to in this request, if it has maintained sufficient information to do so.

Argument: No value
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_SET_LEVEL

Gives a level to which the selection should be set. This request should affect the true level.

Argument: `int`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`

SELN_REQ_YIELD

Requests the holder of the selection to yield it. `SELN_SUCCESS`, `SELN_DIDNT_HAVE`, and `SELN_WRONG_RANK` are legitimate responses. The latter comes from a holder asked to yield the primary selection when it knows a function key is down.

Argument: `Seln_result`
Default: None
Procs: `selection_ask()` `selection_init_request()` `selection_query()`