

**NAME**

**XtCreateSelectionRequest**, **XtSendSelectionRequest**, **XtCancelSelectionRequest** – bundle multiple selection conversion requests into a single request using **MULTIPLE** target

**SYNTAX**

```
#include <X11/Intrinsic.h>
```

```
void XtCreateSelectionRequest(Widget requestor, Atom selection);
```

```
void XtSendSelectionRequest(Widget requestor, Atom selection, Time time);
```

```
void XtCancelSelectionRequest(Widget requestor, Atom selection);
```

**ARGUMENTS**

*requestor* Specifies the widget making the request. Must be of class **Core** or a subclass thereof.

*selection* Specifies the particular selection desired.

*time* Specifies the timestamp to be used in making the request.

**DESCRIPTION**

When **XtCreateSelectionRequest** is called, subsequent calls to **XtGetSelectionValue** and **XtGetSelectionValueIncremental** with the requestor and selection as specified to **XtCreateSelectionRequest** will be bundled into a single selection conversion request with multiple targets. The request is actually initiated by calling **XtSendSelectionRequest**.

When **XtSendSelectionRequest** is called with a value of *requestor* and *selection* matching a previous call to **XtCreateSelectionRequest**, a selection conversion request is actually sent to the selection owner. If a single target request is queued, that request is made. If multiple targets are queued they are bundled into a single request with the target **MULTIPLE** using the specified timestamp. As the conversions are made, the callbacks associated with each **XtGetSelectionValue** and **XtGetSelectionValueIncremental** are invoked in turn.

Multi-threaded applications should lock the application context before calling **XtCreateSelectionRequest** and release the lock after calling **XtSendSelectionRequest** to ensure that the thread assembling the request is safe from interference by another thread assembling a different request naming the same widget and selection.

When **XtCancelSelectionRequest** is called, any requests queued since the last call to **XtCreateSelectionRequest** are cleaned up. Subsequent calls to **XtGetSelectionValue**, **XtGetSelectionValues**, **XtGetSelectionValueIncremental**, and **XtGetSelectionValuesIncremental** will not be deferred.

**SEE ALSO**

*X Toolkit Intrinsics – C Language Interface*

*Xlib – C Language X Interface*