Standard Type Services Reference Manual

1.0

Generated by Doxygen 1.3-rc3-20030223

Tue Mar 11 09:59:51 2003

Contents

3 Standard Type Services Module Documentation

3.1 Basic functions

Compounds

- struct xSTCapableReply
  ST Capable reply.
- struct xSTCapableReq
  ST Capable request.
- struct xSTDisableReq
  ST Disabled request.
- struct xSTEnableReq
  ST Enabled request.
- struct xSTInfoReply
  ST Info reply.
- struct xSTInfoReq
  ST Info request.
- struct xSTQueryVersionReply
  ST Query Version reply.
- struct xSTQueryVersionReq
  ST Query Version request.

Defines

- #define **sz_xSTQueryVersionReq** 8
- #define **sz_xSTQueryVersionReply** 32
- #define **sz_xSTCapableReq** 4
- #define **sz_xSTCapableReply** 32
- #define **sz_xSTEnableReq** 4
- #define **sz_xSTDisableReq** 4
- #define **sz_xSTInfoReq** 4
- #define **sz_xSTInfoReply** 32

3.2 XSTTypeEnv Protocol messages

Compounds

- struct xSTTypeEnvCreateFontReply

  ST TypeEnv Create Font reply.

- struct xSTTypeEnvCreateFontReq

  ST TypeEnv Create Font request.

- struct xSTTypeEnvDestroyFontReq

  ST TypeEnv Destroy Font request.

- struct xSTTypeEnvDisposeReq

  ST TypeEnv Dispose request.

- struct xSTTypeEnvFindAllLayoutEnginesReply

  ST TypeEnv Find All Layout Engines reply.

- struct xSTTypeEnvFindAllLayoutEnginesReq

  ST TypeEnv Find All Layout Engines request.

- struct xSTTypeEnvFindAllScalersReply

  ST TypeEnv Find All Scalers reply.

- struct xSTTypeEnvFindAllScalersReq

  ST TypeEnv Find All Scalers request.

- struct xSTTypeEnvFindCreateFontFromURLReq

ST TypeEnv Find or Create Font From URL request.

- struct xSTTypeEnvFindFontFamilyReply
  ST TypeEnv Find Font Family(s) reply.
- struct xSTTypeEnvFindFontFamilyReq
  ST TypeEnv Find Font Family(s) request.
- struct xSTTypeEnvFindFontsByNameReply
  ST TypeEnv Find Fonts By Name reply.
- struct xSTTypeEnvFindFontsByNameReq
  ST TypeEnv Find Fonts By Name request.
- struct xSTTypeEnvFindLayoutEngineReply
  ST TypeEnv Find Layout Engine reply.
- struct xSTTypeEnvFindLayoutEngineReq
  ST TypeEnv Find Layout Engine.
- struct xSTTypeEnvFindScalerReply
  ST TypeEnv Find Scaler reply.
- struct xSTTypeEnvFindScalerReq
  ST TypeEnv Find Scaler request.
- struct xSTTypeEnvGetFontFallbackPolicyReply
  ST TypeEnv Get Font Fallback Policy reply.
- struct xSTTypeEnvGetFontFallbackPolicyReq
  ST TypeEnv Get Font Fallback Policy request.
- struct xSTTypeEnvGetFontFallbacksReply
  ST TypeEnv Get Font Fallbacks reply.
- struct xSTTypeEnvGetFontFallbacksReq
  ST TypeEnv Get Font Fallbacks request.
- struct xSTTypeEnvGetFontFoldersReply
  ST TypeEnv Get Font Folders reply.
- struct xSTTypeEnvGetFontFoldersReq

ST TypeEnv Get Font Folders request.

- struct xSTTypeEnvGetLocationsReply

  ST TypeEnv Get Locations reply.

- struct xSTTypeEnvGetLocationsReq

  ST TypeEnv Get Locations request.

- struct xSTTypeEnvNewCopyReq

  ST TypeEnv New/New Copy request.

- struct xSTTypeEnvSetFontFallbackPolicyReq

  ST TypeEnv Set Font Fallback Policy request.

- struct xSTTypeEnvSetFontFallbacksReq

  ST TypeEnv Set Font Fallbacks request.

- struct xSTTypeEnvSetFontFoldersReq

  ST TypeEnv Set Font Folders request.

- struct xSTTypeEnvSetLocationsReq

  ST TypeEnv Set Locations request.

Defines

- #define **sz_xSTTypeEnvNewCopyReq** 12
- #define **sz_xSTTypeEnvDisposeReq** 8
- #define **sz_xSTTypeEnvSetFontFallbackPolicyReq** 12
- #define **sz_xSTTypeEnvGetFontFallbackPolicyReq** 8
- #define **sz_xSTTypeEnvGetFontFallbackPolicyReply** 32
- #define **sz_xSTTypeEnvSetFontFallbacksReq** 12
- #define **sz_xSTTypeEnvGetFontFallbacksReq** 8
- #define **sz_xSTTypeEnvGetFontFallbacksReply** 32
- #define **sz_xSTTypeEnvFindFontsByNameReq** 24
- #define **sz_xSTTypeEnvFindFontsByNameReply** 32
- #define **sz_xSTTypeEnvCreateFontReq** 12
- #define **sz_xSTTypeEnvCreateFontReply** 32
- #define **sz_xSTTypeEnvFindCreateFontFromURLReq** 16
- #define **sz_xSTTypeEnvDestroyFontReq** 12
- #define **sz_xSTTypeEnvFindAllScalersReq** 8
- #define **sz_xSTTypeEnvFindAllScalersReply** 32

- #define **sz_xSTTypeEnvFindScalerReq** 12
- #define **sz_xSTTypeEnvFindScalerReply** 32
- #define **sz_xSTTypeEnvFindAllLayoutEnginesReq** 8
- #define **sz_xSTTypeEnvFindAllLayoutEnginesReply** 32
- #define **sz_xSTTypeEnvFindLayoutEngineReq** 12
- #define **sz_xSTTypeEnvFindLayoutEngineReply** 32
- #define **sz_xSTTypeEnvFindFontFamilyReq** 20
- #define **sz_xSTTypeEnvFindFontFamilyReply** 32
- #define **sz_xSTTypeEnvGetFontFoldersReq** 8
- #define **sz_xSTTypeEnvGetFontFoldersReply** 32
- #define **sz_xSTTypeEnvSetFontFoldersReq** 12
- #define **sz_xSTTypeEnvGetLocationsReq** 8
- #define **sz_xSTTypeEnvGetLocationsReply** 32
- #define **sz_xSTTypeEnvSetLocationsReq** 12

3.3 XSTScaler Protocol messages

Compounds

- struct xSTScalerGetInfoReply

  ST Scaler Get Info reply.

- struct xSTScalerGetInfoReq

  ST Scaler Get Info request.

Defines

- #define **sz_xSTScalerGetInfoReq** 12
- #define **sz_xSTScalerGetInfoReply** 36

3.4 XSTLayoutEngine Protocol messages

Compounds

- struct xSTLayoutEngineGetInfoReply

  ST LayoutEngine Get Info reply.

- struct xSTLayoutEngineGetInfoReq

  ST LayoutEngine Get Info request.

Defines

- #define **sz_xSTLayoutEngineGetInfoReq** 12
- #define **sz_xSTLayoutEngineGetInfoReply** 32

3.5 XSTFont Protocol messages

Compounds

- struct xSTFontFamilyGetFontsReply

  ST FontFamily Get Fonts reply.

- struct xSTFontFamilyGetFontsReq

  ST FontFamily Get Fonts request.

- struct xSTFontFamilyGetNameReply

  ST FontFamily Get Name reply.

- struct xSTFontFamilyGetNameReq

  ST FontFamily Get Name request.

- struct xSTFontGetBaselinesReply

  ST Font Get Baselines reply.

- struct xSTFontGetBaselinesReq

  ST Font Get Baselines request.

- struct xSTFontGetExtInfoReply

  ST Font Get Extra Info reply.

- struct xSTFontGetExtInfoReq

  ST Font Get Extra Info request.

- struct xSTFontGetFeaturesReply

  ST Font Get Features reply.

- struct xSTFontGetFeaturesReq

  ST Font Get Features request.

- struct xSTFontGetFontFamilyReply

  ST Font Get Family reply.

- struct xSTFontGetFontFamilyReq

  ST Font Get Family request.

- struct xSTFontGetNameStringReply

  ST Font Get Name String reply.

- struct xSTFontGetNameStringReq

  ST Font Get Name String request.

- struct xSTFontGetNameTagsReply

  ST Font Get Name Tags reply.

- struct xSTFontGetNameTagsReq

  ST Font Get Name Tags request.

- struct xSTFontGetTypefaceNameReply

  ST Font Get Typeface Name reply.

- struct xSTFontGetTypefaceNameReq

  ST Font Get Typeface Name request.

Defines

- #define **sz_xSTFontGetNameTagsReq** 12
- #define **sz_xSTFontGetNameTagsReply** 32
- #define **sz_xSTFontGetNameStringReq** 24
- #define **sz_xSTFontGetNameStringReply** 32
- #define **sz_xSTFontGetExtInfoReq** 12
- #define **sz_xSTFontGetExtInfoReply** 128
- #define **sz_xSTFontGetBaselinesReq** 12
- #define **sz_xSTFontGetBaselinesReply** 264
- #define **sz_xSTFontGetFeaturesReq** 12
- #define **sz_xSTFontGetFeaturesReply** 32
- #define **sz_xSTFontGetFontFamilyReq** 12

- #define **sz_xSTFontGetFontFamilyReply** 32
- #define **sz_xSTFontGetTypefaceNameReq** 16
- #define **sz_xSTFontGetTypefaceNameReply** 32
- #define **sz_xSTFontFamilyGetNameReq** 16
- #define **sz_xSTFontFamilyGetNameReply** 32
- #define **sz_xSTFontFamilyGetFontsReq** 12
- #define **sz_xSTFontFamilyGetFontsReply** 32

3.6 XSTGC Protocol messages

Compounds

- struct **xSTGCConcatMatrixReq**
- struct xSTGCGetAlphaRangeReply

  ST GC Get Alpha Range reply.

- struct xSTGCGetAlphaRangeReq

  ST GC Get Alpha Range request.

- struct xSTGCGetColorsReply

  ST GC Get Colors reply.

- struct xSTGCGetColorsReq

  ST GC Get Colors request.

- struct xSTGCGetMatrixReply

  ST GC Get Matrix reply.

- struct xSTGCGetMatrixReq

  ST GC Get Matrix request.

- struct xSTGCGetOutputFormatReply

  ST GC Get Output Format reply.

- struct xSTGCGetOutputFormatReq

  ST GC Get Output Format request.

- struct **xSTGCRotateReq**
- struct **xSTGCScaleReq**
- struct xSTGCSetAlphaRangeReq

ST GC Set Alpha Range request.

- struct xSTGCSetColorsReq

  ST GC Set Colors request.

- struct xSTGCSetMatrixReq

  ST GC Set Matrix request.

- struct xSTGCSetOutputFormatReq

  ST GC Set Output Format request.

- struct **xSTGCSetRegionReq**
- struct **xSTGCShearReq**
- struct **xSTGCTranslateReq**

Defines

- #define **sz_xSTGCSetColorsReq** 48
- #define **sz_xSTGCGetColorsReq** 8
- #define **sz_xSTGCGetColorsReply** 48
- #define **sz_xSTGCSetOutputFormatReq** 16
- #define **sz_xSTGCGetOutputFormatReq** 8
- #define **sz_xSTGCGetOutputFormatReply** 32
- #define **sz_xSTGCSetAlphaRangeReq** 80
- #define **sz_xSTGCGetAlphaRangeReq** 8
- #define **sz_xSTGCGetAlphaRangeReply** 80
- #define **sz_xSTGCGetMatrixReq** 8
- #define **sz_xSTGCGetMatrixReply** 56
- #define **sz_xSTGCSetMatrixReq** 56
- #define **sz_xSTGCConcatMatrixReq** 56
- #define **sz_xSTGCTranslateReq** 24
- #define **sz_xSTGCScaleReq** 24
- #define **sz_xSTGCRotateReq** 16
- #define **sz_xSTGCShearReq** 24
- #define **sz_xSTGCSetRegionReq** 28

3.7 XSTStyle Protocol messages

Compounds

- struct xSTStyleClearReq
  ST Style Clear request.
- struct xSTStyleCompareReply
  ST Style Compare reply.
- struct xSTStyleCompareReq
  ST Style Compare request.
- struct xSTStyleCopyAttributesReq
  ST Style Copy Attributes request.
- struct xSTStyleDisposeReq
  ST Style Dispose request.
- struct xSTStyleGetBaselinesReply
  ST Style Get Baselines reply.
- struct xSTStyleGetBaselinesReq
  ST Style Get Baselines request.
- struct xSTStyleGetDesignMetricsReply
  ST Style Get Design Metrics reply.
- struct xSTStyleGetDesignMetricsReq
  ST Style Get Design Metrics request.
- struct xSTStyleGetEffectsReply
  ST Style Get Effects reply.
- struct xSTStyleGetEffectsReq
  ST Style Get Effects request.
- struct xSTStyleGetFontReply
  ST Style Get Font reply.
- struct xSTStyleGetFontReq
  ST Style Get Font request.

- struct xSTStyleGetLayoutEngineReply
  ST Style Get LayoutEngine reply.
- struct xSTStyleGetLayoutEngineReq
  ST Style Get LayoutEngine request.
- struct xSTStyleGetOptionsReply
  ST Style Get Options reply.
- struct xSTStyleGetOptionsReq
  ST Style Get Options request.
- struct xSTStyleGetScalerReply
  ST Style Get Scaler reply.
- struct xSTStyleGetScalerReq
  ST Style Get Scaler request.
- struct xSTStyleIsEmptyReply
  ST Style Is Empty reply.
- struct xSTStyleIsEmptyReq
  ST Style Is Empty request.
- struct xSTStyleNewCopyReq
  ST Style New Copy request.
- struct xSTStyleOverwriteAttributesReq
  ST Style Overwrite Attributes request.
- struct xSTStyleResetAttributesReq
  ST Style Reset Attributes request.
- struct xSTStyleScalerControlReply
  ST Style Scaler Control reply.
- struct xSTStyleScalerControlReq
  ST Style Scaler Control request.
- struct xSTStyleSetEffectsReq
  ST Style Set Effects request.

- struct xSTStyleSetFontReq

  ST Style Set Font request.

- struct xSTStyleSetLayoutEngineReq

  ST Style Set LayoutEngine request.

- struct xSTStyleSetOptionsReq

  ST Style Set Options request.

- struct xSTStyleSetScalerReq

  ST Style Set Scaler request.

- struct xSTStyleUnderwriteAttributesReq

  ST Style Underwrite Attributes request.

Defines

- #define **sz_xSTStyleNewCopyReq** 20
- #define **sz_xSTStyleDisposeReq** 8
- #define **sz_xSTStyleCompareReq** 16
- #define **sz_xSTStyleCompareReply** 32
- #define **sz_xSTStyleIsEmptyReq** 8
- #define **sz_xSTStyleIsEmptyReply** 32
- #define **sz_xSTStyleClearReq** 12
- #define **sz_xSTStyleSetFontReq** 32
- #define **sz_xSTStyleGetFontReq** 8
- #define **sz_xSTStyleGetFontReply** 32
- #define **sz_xSTStyleSetScalerReq** 24
- #define **sz_xSTStyleGetScalerReq** 8
- #define **sz_xSTStyleGetScalerReply** 32
- #define **sz_xSTStyleSetLayoutEngineReq** 16
- #define **sz_xSTStyleGetLayoutEngineReq** 8
- #define **sz_xSTStyleGetLayoutEngineReply** 32
- #define **sz_xSTStyleScalerControlReq** 12
- #define **sz_xSTStyleScalerControlReply** 32
- #define **sz_xSTStyleSetEffectsReq** 24
- #define **sz_xSTStyleGetEffectsReq** 8
- #define **sz_xSTStyleGetEffectsReply** 32
- #define **sz_xSTStyleResetAttributesReq** 12
- #define **sz_xSTStyleCopyAttributesReq** 16
- #define **sz_xSTStyleOverwriteAttributesReq** 16

- #define **sz_xSTStyleUnderwriteAttributesReq** 16
- #define **sz_xSTStyleGetBaselinesReq** 12
- #define **sz_xSTStyleGetBaselinesReply** 264
- #define **sz_xSTStyleGetDesignMetricsReq** 8
- #define **sz_xSTStyleGetDesignMetricsReply** 32
- #define **sz_xSTStyleGetOptionsReq** 8
- #define **sz_xSTStyleGetOptionsReply** 44
- #define **sz_xSTStyleSetOptionsReq** 44

3.8 XSTText Protocol messages

Compounds

- struct xSTTextAugmentStyleReq

  ST Text Augment Style request.

- struct xSTTextClearReq

  ST Text Clear request.

- struct xSTTextCopyAttributesReq

  ST Text Copy Attributes request.

- struct xSTTextDisposeReq

  ST Text Dispose request.

- struct xSTTextFindMissingCharsReply

  ST Text Find Missing Chars reply.

- struct xSTTextFindMissingCharsReq

  ST Text Find Missing Chars request.

- struct xSTTextGetCommonStyleReply

  ST Text Get Common Style reply.

- struct xSTTextGetCommonStyleReq

  ST Text Get Common Style request.

- struct xSTTextGetControlsReply

  ST Text Get Controls reply.

- struct xSTTextGetControlsReq

  ST Text Get Controls request.

- struct xSTTextGetFontFallbacksReply

  ST Text Get Font Fallbacks reply.

- struct xSTTextGetFontFallbacksReq

  ST Text Get Font Fallbacks request.

- struct xSTTextGetMetricsReply

  ST Text Get Metrics reply.

- struct xSTTextGetMetricsReq

  ST Text Get Metrics request.

- struct xSTTextGetStyleReply

  ST Text Get Style reply.

- struct xSTTextGetStyleReq

  ST Text Get Style request.

- struct **xSTTextGetTextReply**
- struct xSTTextGetTextReq

  ST Text Get Text request.

- struct xSTTextNewCopyReq

  ST Text New Copy request.

- struct xSTTextNewReq

  ST Text New/New Empty request.

- struct xSTTextOverwriteStyleReq

  ST Text Overwrite Style request.

- struct xSTTextResetAttributesReq

  ST Text Reset Attibutes request.

- struct xSTTextSetControlsReq

  ST Text Set Controls request.

- struct xSTTextSetFontFallbacksReq

  ST Text Set Font Fallbacks request.

- struct xSTTextSetMetricsReq

  ST Text Set Metrics request.

- struct xSTTextSetStyleReq

  ST Text Set Style request.

- struct xSTTextSetTextReq

  ST Text Set Text request.

- struct xSTTextUnderwriteStyleReq

  ST Text Underwrite Style request.

- struct xSTTextUpdateReq

  ST Text Update request.

Defines

- #define **sz_xSTTextNewCopyReq** 12
- #define **sz_xSTTextNewReq** 16
- #define **sz_xSTTextDisposeReq** 8
- #define **sz_xSTTextClearReq** 8
- #define **sz_xSTTextSetMetricsReq** 296
- #define **sz_xSTTextGetMetricsReq** 8
- #define **sz_xSTTextGetMetricsReply** 296
- #define **sz_xSTTextSetControlsReq** 36
- #define **sz_xSTTextGetControlsReq** 8
- #define **sz_xSTTextGetControlsReply** 36
- #define **sz_xSTTextSetFontFallbacksReq** 12
- #define **sz_xSTTextGetFontFallbacksReq** 8
- #define **sz_xSTTextGetFontFallbacksReply** 32
- #define **sz_xSTTextCopyAttributesReq** 16
- #define **sz_xSTTextResetAttributesReq** 12
- #define **sz_xSTTextSetTextReq** 12
- #define **sz_xSTTextGetTextReq** 8
- #define **sz_xSTTextGetTextReply** 32
- #define **sz_xSTTextUpdateReq** 20
- #define **sz_xSTTextAugmentStyleReq** 20
- #define **sz_xSTTextOverwriteStyleReq** 20
- #define **sz_xSTTextUnderwriteStyleReq** 20
- #define **sz_xSTTextSetStyleReq** 20

- #define **sz_xSTTextGetStyleReq** 12
- #define **sz_xSTTextGetStyleReply** 32
- #define **sz_xSTTextGetCommonStyleReq** 16
- #define **sz_xSTTextGetCommonStyleReply** 32
- #define **sz_xSTTextFindMissingCharsReq** 8
- #define **sz_xSTTextFindMissingCharsReply** 32

3.9 XSTLine Protocol messages

Compounds

- struct xSTLineAddHighlightReq

  ST Line Add Highlight request.

- struct xSTLineDisposeReq

  ST Line Dispose request.

- struct xSTLineGetDesignMetricsReply

  ST Line Get Design Metrics reply.

- struct xSTLineGetDesignMetricsReq

  ST Line Get Design Metrics request.

- struct xSTLineGetGlyphBoundsReply

  ST Line Get Glyph Bounds reply.

- struct xSTLineGetGlyphBoundsReq

  ST Line Get Glyph Bounds request.

- struct xSTLineGetHighlightsReply

  ST Line Get Highlights reply.

- struct xSTLineGetHighlightsReq

  ST Line Get Highlights request.

- struct xSTLineGetMetricsReply

  ST Line Get Metrics reply.

- struct xSTLineGetMetricsReq

  ST Line Get Metrics request.

- struct xSTLineGetOffsetReply
  ST Line Get Offset reply.
- struct xSTLineGetOffsetReq
  ST Line Get Offset request.
- struct xSTLineGrowReq
  ST Line Grow request.
- struct xSTLineHitTestReply
  ST Line Hit Test reply.
- struct xSTLineHitTestReq
  ST Line Hit Test request.
- struct xSTLineMeasureTextImageReply
  ST Line Measure Text Image reply.
- struct xSTLineMeasureTextImageReq
  ST Line Measure Text Image request.
- struct xSTLineMeasureTextReply
  ST Line Measure Text reply.
- struct xSTLineMeasureTextReq
  ST Line Measure Text request.
- struct xSTLineMoveCaretReply
  ST Line Move Caret reply.
- struct xSTLineMoveCaretReq
  ST Line Move Caret request.
- struct xSTLineNewForWidthReq
  ST Line New For Width request.
- struct xSTLineNewReply
  ST Line New/New For Width reply.
- struct xSTLineNewReq
  ST Line New request.

- struct xSTLinePositionToCaretReply

  ST Line Position to Caret reply.

- struct xSTLinePositionToCaretReq

  ST Line Position To Caret request.

- struct xSTLineRemoveHighlightReq

  ST Line Remove Highlight request.

- struct xSTLineRenderReq

  ST Line Render request.

- struct xSTLineSetMetricsReq

  ST Line Set Metrics request.

- struct xSTLineShrinkReq

  ST Line Shrink request.

Defines

- #define **sz_xSTLineNewForWidthReq** 24
- #define **sz_xSTLineNewReq** 20
- #define **sz_xSTLineNewReply** 32
- #define **sz_xSTLineDisposeReq** 8
- #define **sz_xSTLineGrowReq** 16
- #define **sz_xSTLineShrinkReq** 16
- #define **sz_xSTLineGetOffsetReq** 8
- #define **sz_xSTLineGetOffsetReply** 32
- #define **sz_xSTLineSetMetricsReq** 296
- #define **sz_xSTLineGetMetricsReq** 8
- #define **sz_xSTLineGetMetricsReply** 296
- #define **sz_xSTLineMeasureTextReq** 8
- #define **sz_xSTLineMeasureTextReply** 40
- #define **sz_xSTLineMeasureTextImageReq** 8
- #define **sz_xSTLineMeasureTextImageReply** 40
- #define **sz_xSTLineGetGlyphBoundsReq** 20
- #define **sz_xSTLineGetGlyphBoundsReply** 32
- #define **sz_xSTLineHitTestReq** 24
- #define **sz_xSTLineHitTestReply** 32
- #define **sz_xSTLinePositionToCaretReq** 16

- #define **sz_xSTLinePositionToCaretReply** 80
- #define **sz_xSTLineMoveCaretReq** 20
- #define **sz_xSTLineMoveCaretReply** 32
- #define **sz_xSTLineRenderReq** 16
- #define **sz_xSTLineAddHighlightReq** 16
- #define **sz_xSTLineGetHighlightsReq** 8
- #define **sz_xSTLineGetHighlightsReply** 32
- #define **sz_xSTLineRemoveHighlightReq** 16
- #define **sz_xSTLineGetDesignMetricsReq** 8
- #define **sz_xSTLineGetDesignMetricsReply** 40

3.10 XSTGlyphVector Protocol messages

Compounds

- struct xSTGlyphVectorAdjustPositionsReq

  ST GlyphVector Adjust Positions request.

- struct xSTGlyphVectorDisposeReq

  ST GlyphVector Dispose request.

- struct xSTGlyphVectorGetBoundsReply

  ST GlyphVector Get Bounds reply.

- struct xSTGlyphVectorGetBoundsReq

  ST GlyphVector Get Bounds request.

- struct xSTGlyphVectorGetGlyphsReply

  ST GlyphVector Get Glyphs reply.

- struct xSTGlyphVectorGetGlyphsReq

  ST GlyphVector Get Glyphs request.

- struct xSTGlyphVectorMeasureReply

  ST GlyphVector Measure reply.

- struct xSTGlyphVectorMeasureReq

  ST GlyphVector Measure request.

- struct xSTGlyphVectorNewCopyReq

ST GlyphVector New Copy request.

- struct xSTGlyphVectorNewReq

  ST GlyphVector New request.

- struct xSTGlyphVectorRenderReq

  ST GlyphVector Render request.

- struct xSTGlyphVectorReplaceGlyphsReq

  ST GlyphVector Replace Glyphs request.

- struct xSTGlyphVectorSetGlyphsReq

  ST GlyphVector Set Glyphs request.

Defines

- #define **sz_xSTGlyphVectorNewReq** 16
- #define **sz_xSTGlyphVectorNewCopyReq** 12
- #define **sz_xSTGlyphVectorDisposeReq** 8
- #define **sz_xSTGlyphVectorSetGlyphsReq** 12
- #define **sz_xSTGlyphVectorGetGlyphsReq** 16
- #define **sz_xSTGlyphVectorGetGlyphsReply** 32
- #define **sz_xSTGlyphVectorReplaceGlyphsReq** 20
- #define **sz_xSTGlyphVectorAdjustPositionsReq** 32
- #define **sz_xSTGlyphVectorMeasureReq** 16
- #define **sz_xSTGlyphVectorMeasureReply** 40
- #define **sz_xSTGlyphVectorGetBoundsReq** 28
- #define **sz_xSTGlyphVectorGetBoundsReply** 32
- #define **sz_xSTGlyphVectorRenderReq** 24

4 Standard Type Services Class Documentation

4.1 xSTCapableReply Struct Reference

ST Capable reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- BOOL capable
  Flag for ST Capable.
- CARD8 pad1
  Not used.
- CARD16 pad2 B16
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.
- CARD32 pad6 B32
  Not used.
- CARD32 pad7 B32
  Not used.

4.1.1 Detailed Description

ST Capable reply.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.2 xSTCapableReq Struct Reference

ST Capable request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STCapable.

- CARD16 length B16

  Request Length in Bytes.

4.2.1 Detailed Description

ST Capable request.

The X_STCapable request returns a Boolean flag which indicates whether the target XServer is capable of performing XST functions.

BadLength may be returned if the protocol message is not of the proper length.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.3 xSTDisableReq Struct Reference

ST Disabled request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STDisable.
- CARD16 length B16
  Request Length in Bytes.

4.3.1 Detailed Description

ST Disabled request.

The X_STDisable request is used to turn XST capabilities in the server OFF.

BadLength may be returned if the protocol message is not of the proper length.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.4 xSTEnableReq Struct Reference

ST Enabled request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType

ST Request: X_STEnable.

- CARD16 length B16

  Request Length in Bytes.

4.4.1 Detailed Description

ST Enabled request.

The X_STEnable request is used to turn XST capabilities in the server ON.

BadLength may be returned if the protocol message is not of the proper length.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.5 xSTFontFamilyGetFontsReply Struct Reference

ST FontFamily Get Fonts reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 count B32

  Number of XSTFont IDs returned.

- CARD32 pad1 B32

Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.5.1 Detailed Description

ST FontFamily Get Fonts reply.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.6 xSTFontFamilyGetFontsReq Struct Reference

ST FontFamily Get Fonts request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STFontFamilyGetFonts.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

XSTTypeEnv XID to use.

- CARD32 FontFamily B32

  Font ID to use in getting the info.

4.6.1 Detailed Description

ST FontFamily Get Fonts request.

The X_STFontFamilyGetFonts request attempts to retrieve the Font IDs for a specified font family in a particular STTypeEnv. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol FontFamily is the specific font family from which the user wants to obtain the list of font IDs. The XServer calls STFontFamilyGetFonts and passes the STTypeEnv structure, and the FontFamily ID. The function returns a list of font IDs along with the number of fonts in the list.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.7 xSTFontFamilyGetNameReply Struct Reference

ST FontFamily Get Name reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 count B32

  Number of characters in the returned string.

- CARD32 bytes B32

  Number of bytes in the returned string.

- CARD32 LanguageID B32

  Language of returned string.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

4.7.1 Detailed Description

ST FontFamily Get Name reply.

The protocol response count is filled in with the number of characters in the string. The protocol response bytes is filled in with the number of bytes used by the string. The font family name string is appended onto the end of the protocol response.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.8 xSTFontFamilyGetNameReq Struct Reference

ST FontFamily Get Name request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STFontFamilyGetName.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Env B32
  XSTTypeEnv XID to use.
- CARD32 FontFamily B32
  Font ID to use in getting the info.
- CARD32 LanguageID B32
  Language type requested for string.

4.8.1 Detailed Description

ST FontFamily Get Name request.

The X_STFontFamilyGetName request attempts to retrieve the Font Family name for a specified font family in a particular STTypeEnv. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol FontFamily is the specific font family from which the user wants to obtain the font family name string. The protocol LanguageID is the language and region for which to obtain the name. The XServer calls STFontFamilyGetName and passes the STTypeEnv structure, the FontFamily ID and the LanguageID. The function returns a string along with it's length in characters and the language that the name was returned using. The XServer calls STIntCountUTF16Bytes to get the number of bytes that the string occupies.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.9 xSTFontGetBaselinesReply Struct Reference

ST Font Get Baselines reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD64 base01
  Baseline value.
- CARD64 base02
  Baseline value.
- CARD64 base03
  Baseline value.
- CARD64 base04
  Baseline value.
- CARD64 base05
  Baseline value.
- CARD64 base06
  Baseline value.
- CARD64 base07
  Baseline value.
- CARD64 base08
  Baseline value.

- CARD64 base09

  Baseline value.

- CARD64 base10

  Baseline value.

- CARD64 base11

  Baseline value.

- CARD64 base12

  Baseline value.

- CARD64 base13

  Baseline value.

- CARD64 base14

  Baseline value.

- CARD64 base15

  Baseline value.

- CARD64 base16

  Baseline value.

- CARD64 base17

  Baseline value.

- CARD64 base18

  Baseline value.

- CARD64 base19

  Baseline value.

- CARD64 base20

  Baseline value.

- CARD64 base21

  Baseline value.

- CARD64 base22

  Baseline value.

- CARD64 base23
  Baseline value.
- CARD64 base24
  Baseline value.
- CARD64 base25
  Baseline value.
- CARD64 base26
  Baseline value.
- CARD64 base27
  Baseline value.
- CARD64 base28
  Baseline value.
- CARD64 base29
  Baseline value.
- CARD64 base30
  Baseline value.
- CARD64 base31
  Baseline value.
- CARD64 base32
  Baseline value.

4.9.1 Detailed Description

ST Font Get Baselines reply.

The function returns the baseline values available for the font. The protocol response baseXX is filled in with the appropriate baseline value.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.10 xSTFontGetBaselinesReq Struct Reference

ST Font Get Baselines request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STFontGetBaselines.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 Font B32

  Font ID to use in getting the info.

4.10.1 Detailed Description

ST Font Get Baselines request.

The X_STFontGetBaselines request attempts to retrieve baseline information from a specified font in a particular STTypeEnv. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Font is the specific font from which the user wants to obtain the baseline information. The XServer calls STFontGetBaselines and passes the STTypeEnv structure and the Font. The function returns the baseline values available for the font.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.11 xSTFontGetExtInfoReply Struct Reference

ST Font Get Extra Info reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 FontType B32
  Font Format: TrueType, Type1, etc.
- CARD32 Flags B32
  Mask of font capabilities: IE embedded bitmaps.
- CARD64 hAscent
  Metric value returned from the font.
- CARD64 hDescent
  Metric value returned from the font.
- CARD64 hLeading
  Metric value returned from the font.
- CARD64 vAscent
  Metric value returned from the font.
- CARD64 vDescent
  Metric value returned from the font.
- CARD64 vLeading
  Metric value returned from the font.

- CARD64 italicAngle

  Metric value returned from the font.

- CARD64 xMin

  Metric value returned from the font.

- CARD64 yMin

  Metric value returned from the font.

- CARD64 xMax

  Metric value returned from the font.

- CARD64 yMax

  Metric value returned from the font.

- CARD32 SbitCount B32

  Number of embedded bitmaps available in font.

- CARD32 widthClass B32

  Width of the font: IE Condensed, Expanded.

- CARD32 weightClass B32

  Weight of the font: IE Thin, Bold, etc.

- CARD32 fNameCount B32

  Number of bytes in the Font name.

- CARD32 pNameCount B32

  Number of bytes in the Postscript font name.

- CARD32 pad1 B32

  Not used.

4.11.1 Detailed Description

ST Font Get Extra Info reply.

The protocol response FontType is set to the font type. The protocol response Flags is set to the flags value representing the font's capabilities. The protocol responses h-Ascent, hDescent, hLeading, vAscent, vDescent, vLeading, ItalicAngle, xMin, yMin, xMax, yMax, widthClass, and weightClass are all set to the values stored in the font

metrics. The protocol response SbitCount is set to the number of Sbit elements in the Sbit array. The protocol response fNameCount is set to the number of characters in the full font name. The protocol response pNameCount is set to the number of characters in the Postscript name. The array of Sbits is appended to the end of the response, followed by the full font name string, followed by the postscript name string.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.12 xSTFontGetExtInfoReq Struct Reference

ST Font Get Extra Info request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STFontGetExtInfo.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 Font B32

  Font ID to use in getting the info.

4.12.1 Detailed Description

ST Font Get Extra Info request.

The X_STFontGetExtInfo request attempts to retrieve several items of extra information from a specified font in a particular STTypeEnv. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Font is the specific font from which the user wants to obtain the extra information. The

XServer calls STFontGetExtInfo and passes the STTypeEnv structure and the Font. The function returns the full font name, its length, the postscript name, and its length, along with the font type, flags indicating capabilities, an array of Sbits and the number of elements in the array, and the font's metrics.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.13 xSTFontGetFeaturesReply Struct Reference

ST Font Get Features reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 count B32

  Number of Font Folders returned.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.13.1 Detailed Description

ST Font Get Features reply.

The protocol response count is filled in with the number of features available for the font. The list of features is appended onto the end of the response.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.14 xSTFontGetFeaturesReq Struct Reference

ST Font Get Features request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STFontGetFeatures.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 Font B32

  Font ID to use in getting the info.

4.14.1 Detailed Description

ST Font Get Features request.

The X_STFontGetFeatures request attempts to retrieve the available features for a specified font in a particular STTypeEnv. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Font is the specific font from which the user wants to obtain the features information. The XServer calls STFontGetFeatures and passes the STTypeEnv structure and the Font. The function returns a list of features available for the font.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.15 xSTFontGetFontFamilyReply Struct Reference

ST Font Get Family reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

Length of Reply beyond normal in Bytes.

- CARD32 FontFamily B32
  XSTFontFamily ID.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.15.1 Detailed Description

ST Font Get Family reply.

The protocol response FontFamily is filled in with the FontFamily ID.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.16 xSTFontGetFontFamilyReq Struct Reference

ST Font Get Family request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STFontGetFontFamily.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 Font B32

  Font ID to use in getting the info.

4.16.1 Detailed Description

ST Font Get Family request.

The X_STFontGetFontFamily request attempts to retrieve the FontFamily ID value for a specified font in a particular STTypeEnv. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Font is the specific font from which the user wants to obtain the FontFamily ID. The XServer calls STFontGetFontFamily and passes the STTypeEnv structure and the Font. The function returns the FontFamily ID.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.17 xSTFontGetNameStringReply Struct Reference

ST Font Get Name String reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 count B32

  Number of characters in the non UTF8 string.

- CARD32 bytes B32

  Number of bytes in the UTF8 string.

- CARD32 LanguageID B32

  Language of the string.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

4.17.1 Detailed Description

ST Font Get Name String reply.

The protocol response count is set to the number of bytes in the non UTF8 string. The protocol bytes is set to the number of bytes returned by STIntCountUTF16Bytes. The string is appended onto the end of the response. The protocol response LanguageID is the language of the returned string.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.18 xSTFontGetNameStringReq Struct Reference

ST Font Get Name String request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STFontGetNameString.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Env B32
  XSTTypeEnv XID to use.
- CARD32 Font B32
  Font ID to use in getting the tags.
- CARD16 PlatformID B16
  Platform ID tag.
- CARD16 EncodingID B16
  Encoding ID tag.
- CARD32 LanguageID B16
  Language ID tag.
- CARD16 NameID B16
  Name ID tag.
- CARD16 Format B32
  Platform ID tag.

4.18.1 Detailed Description

ST Font Get Name String request.

The X_STFontGetNameString request attempts to retrieve a particular string from a specified font in a particular STTypeEnv. The string should be uniquely represented by a NameTag whose values are included in the request.

This protocol message is used for three seperate calls. It handles the XSTFontGetNameString, XSTFontGetUnicodeNameString and XSTFontGetURL functions.

The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Font is the specific font from which the user wants to obtain a name string. For the XSTFontGetNameString function, the protocol PlatformID, EncodingID, LanguageID, and NameID are values that represent a unique NameTag. The protocol Format is set to 0. The XServer calls STFontGetNameString and passes the STTypeEnv structure, Font, and the NameTag structure built of the four NameTag values. The function returns a string along with its length. For the XSTFontGetUnicodeNameString function, the protocol PlatformID, EncodingID, LanguageID, and NameID are values that represent a unique NameTag. The protocol Format is set to 1. The XServer calls STFontGetUnicodeName and passes the STTypeEnv structure, Font and the NameTag structure built of the four NameTag values. The function returns a string along with its length. The XServer also calls STIntCountUTF16Bytes to get the number of bytes used in the string. For the XSTFontGetURL function, the protocol PlatformID, EncodingID, LanguageID, and NameID are not used and are set to 0. The protocol Format is set to 2. The XServer calls STFontGetURL and passes the STTypeEnv structure, and the Font. The function returns a string.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for the string cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.19 xSTFontGetNameTagsReply Struct Reference

ST Font Get Name Tags reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 count B32

  Number of Font Folders returned.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.19.1 Detailed Description

ST Font Get Name Tags reply.

The protocol response count is set to the number of tags. The rest of the tags are appended onto the end of the response.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.20 xSTFontGetNameTagsReq Struct Reference

ST Font Get Name Tags request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STFontGetNameTags.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Env B32
  XSTTypeEnv XID to use.
- CARD32 Font B32
  Font ID to use in getting the tags.

4.20.1 Detailed Description

ST Font Get Name Tags request.

The X_STFontGetNameTags request attempts to retrieve all of the NameTags available in a particular font in a particular STTypeEnv. A NameTag is comprised of a PlatformID, EncodingID, LanguageID, and NameID. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Font is the specific font from which the user wants to obtain NameTags. The XServer calls STFontGetNameTags and passes both the STTypeEnv structure and Font. The function returns a list of NameTags and the number of tags found in the font.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for the NameTags cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.21 xSTFontGetTypefaceNameReply Struct Reference

ST Font Get Typeface Name reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of characters in the returned string.
- CARD32 bytes B32
  Number of bytes in the returned string.
- CARD32 weight B32
  Weight of font: IE: Compressed, Expanded.
- CARD32 width B32
  Width of font: IE: Thin, Bold, etc.
- CARD32 style B32
  Style of font: IE: Regular, Italic.
- CARD32 LanguageID B32
  Language of returned string.

4.21.1 Detailed Description

ST Font Get Typeface Name reply.

The protocol response count is filled in with the number of characters in the string. The protocol response bytes is filled in with the number of bytes used by the string. The protocol response LanguageID is filled in with the LanguageID that most nearly matches the requested LanguageID. The protocol response weight tells the weight of the font from light to heavy. The protocol response width tells the width of the font from narrow to expanded. The protocol response style tells the style of the font from Regular to Italic. The Typeface name string is appended onto the end of the protocol response.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.22 xSTFontGetTypefaceNameReq Struct Reference

ST Font Get Typeface Name request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STFontGetTypefaceName.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Env B32
  XSTTypeEnv XID to use.
- CARD32 Font B32
  Font ID to use in getting the info.
- CARD32 LanguageID B32
  Language type requested for string.

4.22.1 Detailed Description

ST Font Get Typeface Name request.

The X_STFontGetTypefaceName request attempts to retrieve the available features for a specified font in a particular STTypeEnv. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Font is the specific font from which the user wants to obtain the typeface name string. The protocol LanguageID is the language and region for which to obtain the name. The XServer calls STFontGetTypefaceName and passes the STTypeEnv structure, the Font and the LanguageID. The function returns a string along with it's length in characters and the language that the typeface was returned using. The XServer calls STIntCountUTF16Bytes to get the number of bytes that the string occupies.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.23 xSTGCGetAlphaRangeReply Struct Reference

ST GC Get Alpha Range reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD64 TminAlpha

  Text minimum Alpha value.

- CARD64 TmaxAlpha

  Text maximum Alpha value.

- CARD64 HminAlpha

  Highlight minimum Alpha value.

- CARD64 HmaxAlpha

  Highlight maximum Alpha value.

- CARD64 UminAlpha

  Underline minimum Alpha value.

- CARD64 UmaxAlpha

  Underline maximum Alpha value.

- CARD64 SminAlpha

  Strikethrough minimum Alpha value.

- CARD64 SmaxAlpha

  Strikethrough maximum Alpha value.

- CARD32 mask B32

  Mask indicating which Alpha values are set.

- CARD32 pad1 B32

  Not used.

4.23.1 Detailed Description

ST GC Get Alpha Range reply.

The protocol response TminAlpha is a double that represents the minimum alpha value used for the Text. The protocol response TmaxAlpha is a double that represents the maximum alpha value used for the Text. The protocol response HminAlpha is a double that represents the minimum alpha value used for the background highlight. The protocol HmaxAlpha is a double that represents the maximum alpha value used for the background highlight. The protocol response UminAlpha is a double that represents the minimum alpha value used for the underline. The protocol response UmaxAlpha is a double that represents the maximum alpha value used for the underline. The protocol

response SminAlpha is a double that represents the minimum alpha value used for the strikethrough. The protocol response SmaxAlpha is a double that represents the maximum alpha value used for the strikethrough. The ∗min values are defaulted to 0 while the ∗max values are defaulted to 1.0.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.24 xSTGCGetAlphaRangeReq Struct Reference

ST GC Get Alpha Range request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGCGetAlphaRange.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 gc B32

  Graphics Context.

4.24.1 Detailed Description

ST GC Get Alpha Range request.

The X_STGCGetAlphaRange request is used to get alpha ranges for the text, highlight, underline, and strikethrough. The protocol gc is the XID value referring to the gc structure to check.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the GC to use is not valid.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.25 xSTGCGetColorsReply Struct Reference

ST GC Get Colors reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 TextColor B32
  Color of text.
- CARD32 HLTextColor B32
  Color of highlighted text.
- CARD32 HLBGColor B32
  Color of highlight.
- CARD32 STColor B32
  Color of strikethrough.
- CARD32 HLSTColor B32
  Color of highlighted strikethrough.
- CARD32 ULColor1 B32
  Color of first underline.

- CARD32 HLULColor1 B32

  Color of highlighted first underline.

- CARD32 ULColor2 B32

  Color of second underline.

- CARD32 HLULColor2 B32

  Color of highlighted second underline.

- CARD32 mask B32

  Mask indicating which colors have been set.

4.25.1 Detailed Description

ST GC Get Colors reply.

The protocol response TextColor is an ARGB value used for the non highlighted text. The protocol response HLTextColor is an ARGB value used for the highlighted text. The protocol response HLBGColor is an ARGB value used for the highlighted background color. The protocol response STColor is an ARGB value used for the strikethrough color. The protocol response HLSTColor is an ARGB value used for the highlighted strikethrough color. The protocol response ULColor1 is an ARGB value used for the first underline color. The protocol response HLULColor1 is an ARGB value used for the first highlighted underline color. The protocol response ULColor2 is an ARGB value used for the second underline color. The protocol response HLULColor2 is an ARGB value used for the second highlighted underline color. The protocol response mask indicates which of the color values have been set.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.26 xSTGCGetColorsReq Struct Reference

ST GC Get Colors request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGCGetColors.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 gc B32

  Graphics Context.

4.26.1 Detailed Description

ST GC Get Colors request.

The X_STGCGetColors request is used to get colors for the text, highlight, underline, and strikethrough. The protocol gc is the XID value referring to the gc structure to check. The XServer calls STGraphicsGetColors and passes the STGraphics object obtained from the GC. The function returns those colors that have been set along with a mask indicating which colors those are.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the GC to use is not valid. BadImplementation may be returned if the STGraphics to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.27 xSTGCGetMatrixReply Struct Reference

ST GC Get Matrix reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD64 xx

  xx portion of matrix

- CARD64 xy

  xy portion of matrix [xx][xy]

- CARD64 yx

  yx portion of matrix [yx][yy]

- CARD64 yy

  yy portion of matrix [tx][ty]

- CARD64 tx

  tx portion of matrix

- CARD64 ty

  ty portion of matrix

4.27.1 Detailed Description

ST GC Get Matrix reply.

The protocol response xx is a double representing the xx portion of the matrix. The protocol response xy is a double representing the xy portion of the matrix. The protocol response yx is a double representing the yx portion of the matrix. The protocol response yy is a double representing the yy portion of the matrix. The protocol response tx is a double representing the tx portion of the matrix. The protocol response ty is a double representing the ty portion of the matrix.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.28 xSTGCGetMatrixReq Struct Reference

ST GC Get Matrix request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGCGetMatrix.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 gc B32

  Graphics Context.

4.28.1 Detailed Description

ST GC Get Matrix request.

The X_STGCGetMatrix request is used to get the transformation matrix. The protocol gc is the XID value referring to the gc structure to use. The XServer calls STDeviceGetMatrix and passes the STDevice object obtained from the GC. The function returns the transformation matrix.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the GC to use is not valid. BadImplementation may be returned if the STDevice to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.29 xSTGCGetOutputFormatReply Struct Reference

ST GC Get Output Format reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 Mode B32
  Format of the output.
- CARD32 Type B32
  Output Type.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.

4.29.1 Detailed Description

ST GC Get Output Format reply.

The protocol response Mode indicates which output device is being used (B&W, Grayscale, LCD, etc.). The protocol response Type indicates whether the format is Raster or Vector.

4.29.2 Member Data Documentation

4.29.2.1 CARD32 Type xSTGCGetOutputFormatReply::B32

Output Type.

IE: Raster or Vector device

4.29.2.2 CARD32 Mode xSTGCGetOutputFormatReply::B32

Format of the output.

IE: B&W, LCD, Grayscale

The documentation for this struct was generated from the following file:

- XSTstr.h

4.30 xSTGCGetOutputFormatReq Struct Reference

ST GC Get Output Format request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STGCGetOutputFormat.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 gc B32
  Graphics Context.

4.30.1 Detailed Description

ST GC Get Output Format request.

The X_STGCGetOutputFormat request is used to get the output device, either a raster or vector device, and to get the mode the output is tailored to. The protocol gc is the

XID value referring to the gc structure to modify. The XServer calls STGraphicsGetDevice and passes the STGraphics object obtained from the GC.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the GC to use is not valid. BadImplementation may be returned if the STGraphics to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.31 xSTGCSetAlphaRangeReq Struct Reference

ST GC Set Alpha Range request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGCSetAlphaRange.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 gc B32

  Graphics Context.

- CARD64 TminAlpha

  Text minimum Alpha value.

- CARD64 TmaxAlpha

  Text maximum Alpha value.

- CARD64 HminAlpha

  Highlight minimum Alpha value.

- CARD64 HmaxAlpha

  Highlight maximum Alpha value.

- CARD64 UminAlpha

  Underline minimum Alpha value.

- CARD64 UmaxAlpha

  Underline maximum Alpha value.

- CARD64 SminAlpha

  Strikethrough minimum Alpha value.

- CARD64 SmaxAlpha

  Strikethrough maximum Alpha value.

- CARD32 mask B32

  Mask indicating which Alpha values are set.

- CARD32 pad1 B32

  Not used.

4.31.1 Detailed Description

ST GC Set Alpha Range request.

The X_STGCSetAlphaRange request is used to set alpha ranges for the text, highlight, underline, and strikethrough. The protocol gc is the XID value referring to the gc structure to modify. The protocol TminAlpha is a double that represents the minimum alpha value to use for the Text. This value is normally 0. Values higher than 0 will result in the bounding box containing the text to become visible. The protocol TmaxAlpha is a double that represents the maximum alpha value to use for the Text. This value is normally 1.0. Values lower than 1.0 will result in semi-transparent text. The protocol HminAlpha is a double that represents the minimum alpha value to use for the background highlight. The protocol HmaxAlpha is a double that represents the maximum alpha value to use for the background highlight. Values lower than 1.0 will result in a semi-transparent highlighting effect. The protocol UminAlpha is a double that represents the minimum alpha value to use for the underline. The protocol UmaxAlpha is a double that represents the maximum alpha value to use for the underline. Values lower than 1.0 will result in a semi-transparent underline. The protocol SminAlpha is a double that represents the minimum alpha value to use for the strikethrough. The protocol SmaxAlpha is a double that represents the maximum alpha value to use for the strikethrough. Values lower than 1.0 will result in a semi-transparent strikethrough. No call to ST is made for this command. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the GC to use is not valid or if the ∗min is greater than the

∗max value. BadAlloc may be returned if the STGraphics associated with the GC cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.32 xSTGCSetColorsReq Struct Reference

ST GC Set Colors request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STGCSetColors.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 gc B32
  Graphics Context.
- CARD32 mask B32
  Mask indicating which colors to set.
- CARD32 TextColor B32
  Color of text.
- CARD32 HLTextColor B32
  Color of highlighted text.
- CARD32 HLBGColor B32
  Color of highlight.
- CARD32 STColor B32
  Color of strikethrough.

- CARD32 HLSTColor B32
  Color of highlighted strikethrough.
- CARD32 ULColor1 B32
  Color of first underline.
- CARD32 HLULColor1 B32
  Color of highlighted first underline.
- CARD32 ULColor2 B32
  Color of second underline.
- CARD32 HLULColor2 B32
  Color of highlighted second underline.

4.32.1 Detailed Description

ST GC Set Colors request.

The X_STGCSetColors request is used to set colors for the text, highlight, underline, and strikethrough. The protocol gc is the XID value referring to the gc structure to modify. The protocol TextColor is an ARGB value used for the non highlighted text. The protocol HLTextColor is an ARGB value used for the highlighted text. The protocol HLBGColor is an ARGB value used for the highlighted background color. The protocol STColor is an ARGB value used for the strikethrough color. The protocol HLSTColor is an ARGB value used for the highlighted strikethrough color. The protocol ULColor1 is an ARGB value used for the first underline color. The protocol HLULColor1 is an ARGB value used for the first highlighted underline color. The protocol ULColor2 is an ARGB value used for the second underline color. The protocol HLULColor2 is an ARGB value used for the second highlighted underline color. The protocol mask indicates which of the color values to set. The XServer calls STGraphicsSetColors and passes the STGraphics object obtained from the GC along with the colors and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the GC to use is not valid. BadImplementation may be returned if the STGraphics to use is NULL. BadAlloc may be returned if an STGraphics cannot be allocated to associate with the GC.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.33 xSTGCSetMatrixReq Struct Reference

ST GC Set Matrix request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STGCSetMatrix.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 gc B32
  Graphics Context.
- CARD64 xx
  xx portion of matrix
- CARD64 xy
  xy portion of matrix [xx][xy]
- CARD64 yx
  yx portion of matrix [yx][yy]
- CARD64 yy
  yy portion of matrix [tx][ty]
- CARD64 tx
  tx portion of matrix
- CARD64 ty
  ty portion of matrix

4.33.1 Detailed Description

ST GC Set Matrix request.

The X_STGCSetMatrix request is used to set the transformation matrix held in the GC and the STDevice object. The protocol gc is the XID value referring to the gc structure to modify. The protocol xx is a double representing the xx portion of the matrix. The protocol xy is a double representing the xy portion of the matrix. The protocol yx is a double representing the yx portion of the matrix. The protocol yy is a double representing the yy portion of the matrix. The protocol tx is a double representing the tx portion of the matrix. The protocol ty is a double representing the ty portion of the matrix. The XServer calls STDeviceSetMatrix and passes the STDevice object obtained from the GC along with the matrix information. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the GC to use is not valid. BadImplementation may be returned if the STDevice to use is NULL. BadAlloc may be returned if the STGraphics cannot be allocated to associate with the GC or the STDevice cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.34 xSTGCSetOutputFormatReq Struct Reference

ST GC Set Output Format request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGCSetOutputFormat.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 gc B32

  Graphics Context.

- CARD32 Mode B32

  Format of the output.

- CARD32 Type B32

  Output Type.

4.34.1 Detailed Description

ST GC Set Output Format request.

The X_STGCSetOutputFormat request is used to set the output to go to a raster or vector device, and to set the mode to tailor the output to be B&W, Grayscale or LCD optimized. The protocol gc is the XID value referring to the gc structure to modify. The protocol Mode indicates which output device is being used (B&W, Grayscale, LCD, etc.). The protocol Type indicates whether the format is Raster or Vector. The XServer calls STGraphicsSetDevice and passes the STGraphics object obtained from the GC along with the STDevice created based on the Type value. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the GC to use is not valid or the Type value is not Raster or Vector. BadImplementation may be returned if the STGraphics to use is NULL. BadAlloc may be returned if the STGraphics cannot be allocated to associate with the GC or the STDevice cannot be allocated.

4.34.2 Member Data Documentation

4.34.2.1 CARD32 Type xSTGCSetOutputFormatReq::B32

Output Type.

IE: Raster or Vector device

4.34.2.2 CARD32 Mode xSTGCSetOutputFormatReq::B32

Format of the output.

IE: B&W, LCD, Grayscale

The documentation for this struct was generated from the following file:

- XSTstr.h

4.35 xSTGlyphVectorAdjustPositionsReq Struct Reference

ST GlyphVector Adjust Positions request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STGlyphVectorAdjustPositions.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 GlyphVector B32
  XSTGlyphVector XID of STGlyphVector to use.
- CARD32 position B32
  Position of first GlyphVector to shift.
- CARD32 GlyphCount B32
  Number of GlyphVectors to shift.
- CARD64 x
  Amount of shift GlyphVectors along the X axis.
- CARD64 y
  Amount of shift GlyphVectors along the Y axis.

4.35.1 Detailed Description

ST GlyphVector Adjust Positions request.

The X_STGlyphVectorAdjustPositions request is used to shift the position of a portion of the GlyphVectors by a specified X, Y amount. The protocol GlyphVector represents the XSTGlyphVector XID of the STGlyphVector object to use. The protocol position indicates the position of the first GlyphVector to shift. The protocol GlyphCount

indicates the number of GlyphVectors to shift. The protocol x indicates the amount to shift the selected GlyphVectors by along the X axis. The protocol y indicates the amount to shift the selected GlyphVectors by along the Y axis. The XServer calls STGlyphVectorAdjustPositions and passes the STGlyphVector, the position, the number of GlyphVectors to shift, and the X and Y values. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTGlyphVector to use is not valid. BadImplementation may be returned if the STGlyphVector to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.36 xSTGlyphVectorDisposeReq Struct Reference

ST GlyphVector Dispose request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGlyphVectorDispose.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 GlyphVector B32

  XSTGlyphVector XID of STGlyphVector to use.

4.36.1 Detailed Description

ST GlyphVector Dispose request.

The X_STGlyphVectorDispose request is used to remove and free the resources used by the STGlyphVector as well as remove the XID value from the XServer. The protocol GlyphVector represents the XSTGlyphVector XID of the STGlyphVector object

to dispose. The XServer calls STGlyphVectorDispose. There is no response to the XSTGlyphVectorDispose command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTGlyphVector to use is not valid. BadImplementation may be returned if the STGlyphVector to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.37 xSTGlyphVectorGetBoundsReply Struct Reference

ST GlyphVector Get Bounds reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 count B32

  Number of bounding boxes included in reply.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.37.1 Detailed Description

ST GlyphVector Get Bounds reply.

The procotol response count indicates the number of bounding boxes returned.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.38 xSTGlyphVectorGetBoundsReq Struct Reference

ST GlyphVector Get Bounds request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGlyphVectorGetBounds.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 GlyphVector B32

  XSTGlyphVector XID of STGlyphVector to use.

- CARD32 gc B32

  Graphics Contect containing transformation.

- CARD32 position B32

  Position of first GlyphVector to use.

- CARD32 GlyphCount B32

  Number of GlyphVectors to use.

- CARD32 MaxBounds B32

  Maximum number of bounding boxes to return.

- CARD32 BoundsType B32

  Type of bounding box to obtain.

4.38.1 Detailed Description

ST GlyphVector Get Bounds request.

The X_STGlyphVectorGetBounds request is used to get the transformed trapezoidal bounding areas around the selected GlyphVectors. The protocol GlyphVector represents the XSTGlyphVector XID of the STGlyphVector object to use. The protocol position indicates the position of the first GlyphVector to use. The protocol GlyphCount indicates the number of GlyphVectors to use. The protocol MaxBounds indicates the maximum number of bounding areas to return. The protocol BoundsType indicates the type of bounding areas to use. The XServer calls STGlyphVectorGetBounds and passes the STGlyphVector object, the position and the number of GlyphVectors to include, and the bounds max and type to use. It returns a list of STTrapezoids.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTGlyphVector to use is not valid. BadImplementation may be returned if the STGlyphVector to use is NULL. BadAlloc may be returned if space for the protocol response cannot be allocated.

4.38.2 Member Data Documentation

4.38.2.1 CARD32 BoundsType xSTGlyphVectorGetBoundsReq::B32

Type of bounding box to obtain.

IE: caret

The documentation for this struct was generated from the following file:

- XSTstr.h

4.39 xSTGlyphVectorGetGlyphsReply Struct Reference

ST GlyphVector Get Glyphs reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of GlyphVectors returned.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.39.1 Detailed Description

ST GlyphVector Get Glyphs reply.

The protocol response count indicates the number of GlyphVectors appended onto the end of the response.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.40 xSTGlyphVectorGetGlyphsReq Struct Reference

ST GlyphVector Get Glyphs request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGlyphVectorGetGlyphs.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 GlyphVector B32

  XSTGlyphVector XID of STGlyphVector to use.

- CARD32 position B32

  Position of first GlyphVector to return.

- CARD32 GlyphCount B32

  Number of GlyphVectors to return.

4.40.1 Detailed Description

ST GlyphVector Get Glyphs request.

The X_STGlyphVectorGetGlyphs request is used to return the list of GlyphVectors associated with the specified STGlyphVector. The protocol GlyphVector represents the XSTGlyphVector XID of the STGlyphVector object to use. The protocol position indicates the position of the first GlyphVector to return. The protocol GlyphCount indicates the number of GlyphVectors to return. The XServer calls STGlyphVectorGetGlyphs and passes the STGlyphVector, the position and the number of GlyphVectors to retrieve. The function returns a list of GlyphVectors.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTGlyphVector to use is not valid. BadImplementation may be returned if the STGlyphVector to use is NULL. BadAlloc may be returned if space for the protocol response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.41 xSTGlyphVectorMeasureReply Struct Reference

ST GlyphVector Measure reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD64 ax

  Upper left X value.

- CARD64 ay

  Upper left Y value.

- CARD64 bx

  Lower right X value.

- CARD64 by

  Lower right Y value.

4.41.1 Detailed Description

ST GlyphVector Measure reply.

The protocol response ax represents the upper left X value of the bounding rectangle.
The protocol response ay represents the upper left Y value of the bounding rectangle.
The protocol response bx represents the lower right X value of the bounding rectangle.
The protocol response by represents the lower right Y value of the bounding rectangle.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.42 xSTGlyphVectorMeasureReq Struct Reference

ST GlyphVector Measure request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGlyphVectorMeasure.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 GlyphVector B32

  XSTGlyphVector XID of STGlyphVector to use.

- CARD32 position B32

Position of first GlyphVector to use.

- CARD32 GlyphCount B32

  Number of GlyphVectors to use.

4.42.1 Detailed Description

ST GlyphVector Measure request.

The X_STGlyphVectorMeasure request is used to determine the bounding box around the selected GlyphVectors. The protocol GlyphVector represents the XSTGlyphVector XID of the STGlyphVector object to use. The protocol position indicates the position of the first GlyphVector to use. The protocol GlyphCount indicates the number of GlyphVectors to use. The XServer calls STGlyphVectorMeasure and passes the STGlyphVector object, the position and the number of GlyphVectors to include. It returns an STRectangle indicating the upper left and lower right corners of the bounding rectangle.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTGlyphVector to use is not valid. BadImplementation may be returned if the STGlyphVector to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.43 xSTGlyphVectorNewCopyReq Struct Reference

ST GlyphVector New Copy request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGlyphVectorNewCopy.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 OldVector **B32**
- CARD32 NewVector **B32**

4.43.1 Detailed Description

ST GlyphVector New Copy request.

The X_STGlyphVectorNewCopy request is used to copy one STGlyphVector object to another new STGlyphVector object. The protocol OldVector represents the XSTGlyphVector XID used by the XServer to get the STGlyphVector object. The protocol NewVector represents the XSTGlyphVector XID of the new STGlyphVector to be created by STSF. The XServer calls STGlyphVectorNewCopy and passes the existing STGlyphVector object. It returns a pointer the new STGlyphVector object. This command has no response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for STGlyphVector data structures cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.44 xSTGlyphVectorNewReq Struct Reference

ST GlyphVector New request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGlyphVectorNew.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

XSTTypeEnv XID to use.

- CARD32 count B32

  Number of glyph vectors to put in object.

- CARD32 GlyphVector B32

  XSTGlyphVector XID for new STGlyphVector.

4.44.1 Detailed Description

ST GlyphVector New request.

The X_STGlyphVectorNew request is used to create a new STGlyphVector containing any glyphs appended to the request. A GlyphVector is similar to an STLine except that there is no layout that occurs on the GlyphVector. The client is responsible for poisitioning all glyphs. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol count indicates the number of GlyphVector items appended onto the end of the request. The protocol GlyphVector is the XID that will be used to refer to the STGlyphVector object. The XServer calls STGlyphVectorNew and passes the STTypeEnv object, and the list of GlyphVectors. It returns a pointer to the new STGlyphVector object. This command has no response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for STGlyphVector data structures cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.45 xSTGlyphVectorRenderReq Struct Reference

ST GlyphVector Render request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGlyphVectorRender.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 GlyphVector B32

  XSTGlyphVector XID of STGlyphVector to use.

- CARD32 drawable B32

  Window or Pixmap to draw on.

- CARD32 gc B32

  Graphics Context.

- CARD32 position B32

  Position of first GlyphVector to render.

- CARD32 GlyphCount B32

  Number of GlyphVectors to render.

4.45.1 Detailed Description

ST GlyphVector Render request.

The X_STGlyphVectorRender request is used to render an STGlyphVector object referred to by the XSTGlyphVector XID. STGlyphVectorRender requires data values to be set, including the callback for Rendering. The protocol GlyphVector is the XID value referring to the STGlyphVector object to use. The protocol gc is GC containing the transformation matrix and STGraphics objects to use. The protocol drawable is the destination window or pixmap to draw on. The protocol position indicates the position of the first GlyphVector to use. The protocol GlyphCount indicates the number of GlyphVectors to use. The XServer calls STGlyphVectorRender and passes the STGlyphVector object referred to by GlyphVector, and the STGraphics object found in the GC. The function has no response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTGlyphVector, GC, or Drawable to use is not valid. BadImplementation may be returned if the STGlyphVector to use is NULL. BadAlloc may be returned if space for STGlyphVectorRender data structures cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.46 xSTGlyphVectorReplaceGlyphsReq Struct Reference

ST GlyphVector Replace Glyphs request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGlyphVectorReplaceGlyphs.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 GlyphVector B32

  XSTGlyphVector XID of STGlyphVector to use.

- CARD32 position B32

  Position of first GlyphVector to replace.

- CARD32 GlyphCount B32

  Number of GlyphVectors to replace.

- CARD32 count B32

  Number of GlyphVectors to insert.

4.46.1 Detailed Description

ST GlyphVector Replace Glyphs request.

The X_STGlyphVectorReplaceGlyphs request is used to replace a subset of the GlyphVectors in an STGlyphVector object. The protocol GlyphVector represents the XSTGlyphVector XID of the STGlyphVector object to use. The protocol position indicates the position of the first GlyphVector to replace. The protocol GlyphCount indicates the number of GlyphVectors to replace. The protocol count indicates the number of GlyphVectors appended to the request that will replace the selected area. The XServer calls STGlyphVectorReplaceGlyphs and passes the STGlyphVector, the

position, the number of GlyphVectors to replace, and the list of GlyphVectors to put in their place. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTGlyphVector to use is not valid. BadImplementation may be returned if the STGlyphVector to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.47 xSTGlyphVectorSetGlyphsReq Struct Reference

ST GlyphVector Set Glyphs request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STGlyphVectorSetGlyphs.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 GlyphVector B32

  XSTGlyphVector XID of STGlyphVector to use.

- CARD32 GlyphCount B32

  Number of GlyphVectors appended after request.

4.47.1 Detailed Description

ST GlyphVector Set Glyphs request.

The X_STGlyphVectorSetGlyphs request is used to associate a supplied set to GlyphVectors with the STGlyphVector object. The protocol GlyphVector represents the XSTGlyphVector XID of the STGlyphVector object to use. The protocol GlyphCount

indicates the number of GlyphVectors appended onto the end of the request. The XServer calls STGlyphVectorSetGlyphs and passes the STGlyphVector and the list of GlyphVectors. There is no response to the XSTGlyphVectorSetGlyphs command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTGlyphVector to use is not valid. BadImplementation may be returned if the STGlyphVector to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.48 xSTInfoReply Struct Reference

ST Info reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- BOOL state
  ST enabled or disabled.
- CARD8 pad1
  Not used.
- CARD16 pad2 B16
  Not used.
- CARD32 pad3 B32

Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

- CARD32 pad6 B32

  Not used.

- CARD32 pad7 B32

  Not used.

4.48.1 Detailed Description

ST Info reply.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.49 xSTInfoReq Struct Reference

ST Info request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STInfo.

- CARD16 length B16

  Request Length in Bytes.

4.49.1 Detailed Description

ST Info request.

The X_STInfo request is used to get information on the state of ST within the XServer.

BadLength may be returned if the protocol message is not of the proper length.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.50 xSTLayoutEngineGetInfoReply Struct Reference

ST LayoutEngine Get Info reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 Tag B32
  ???
- CARD32 Version B32
  Major Number/Minor Number of version.
- CARD32 LEFlags B32
  Mask of layout engine abilities.
- CARD32 ShortNameLen B32

Number of bytes in Short Name.

- CARD32 LongNameLen B32

  Number of bytes in Long Name.

- CARD32 NoticeLen B32

  Number of bytes in Notice.

4.50.1 Detailed Description

ST LayoutEngine Get Info reply.

The protocol response Tag represents the ???. The protocol response Version indicates the version number of the scaler. The top 16 bits are the major number, while the bottom 16 bits are the minor number. The protocol response LEFlags is a mask representing the abilities of the scaler. The protocol response ShortNameLen indicates the length of the standard name of the scaler. The protocol response LongNameLen indicates the length of the full name of the scaler. The protocol response NoticeLen indicates the length of the notice string. Each of the strings is appended onto the end of the response message. The ShortName is first, followed by the LongName, followed by the Notice.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.51 xSTLayoutEngineGetInfoReq Struct Reference

ST LayoutEngine Get Info request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLayoutEngineGetInfo.

- CARD16 length B16

Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 LayoutEngine B32

  XSTLayoutEngine XID to use to get info.

4.51.1 Detailed Description

ST LayoutEngine Get Info request.

The X_STLayoutEngineGetInfo request is used to get detailed information about a specific layout engine. The protocol Env is an XSTTypeEnv XID value referring to the STTypeEnv structure. The protocol LayoutEngine is an XSTLayoutEngine XID representing the STLayoutEngine value. The XServer calls STLayoutEngineGetInfo and passes the STTypeEnv object and the STLayoutEngine value. The function returns the name strings, version numbers and capability flags.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.52 xSTLineAddHighlightReq Struct Reference

ST Line Add Highlight request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineAddHighlight.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

- CARD32 Position **B32**
- CARD32 CharCount **B32**

4.52.1 Detailed Description

ST Line Add Highlight request.

The X_STLineAddHighlight request is used to select a portion of an STLine object referred to by the XSTLine XID. The protocol Line is the XID value referring to the STLine object to use. The protocol Position is the character position to start highlight within the line. The protocol CharCount is the number of characters to highlight from the starting point within the line. The XServer calls STLineAddHighlight and passes the STLine object referred to by Line, and the Position and CharCount values. The function has no response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.53 xSTLineDisposeReq Struct Reference

ST Line Dispose request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineDispose.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to new STLine to use.

4.53.1 Detailed Description

ST Line Dispose request.

The X_STLineDispose request is used to free an existing STLine object and its XSTLine XID. The protocol Line is the XID value referring to the STLine object to dispose of. The XServer calls STLineDispose and passes the STLine object referred to by Line. The function has no response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.54 xSTLineGetDesignMetricsReply Struct Reference

ST Line Get Design Metrics reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD64 Ascent

  Maximum ascent of line in design units.

- CARD64 Descent

  Maximum descent of line in design units.

- CARD64 Leading

  Maximum leading of line in design units.

- CARD64 LineWidth

  Maximum width of line in design units.

4.54.1 Detailed Description

ST Line Get Design Metrics reply.

The protocol response LineWidth is a double that indicates the width of the line in points. The protocol response Ascent is a double that indicates the maximum ascent of any character in the line. The protocol response Descent is a double that indicates the maximum descent of any character in the line. The protocol response Leading is a double that indicates the leading value to be used after this line.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.55 xSTLineGetDesignMetricsReq Struct Reference

ST Line Get Design Metrics request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

ST Request: X_STLineGetDesignMetrics.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

4.55.1 Detailed Description

ST Line Get Design Metrics request.

The X_STLineGetDesignMetrics request is used to get the design metrics for STLine object referred to by the XSTLine XID. Design metrics differ from regular metrics in that they are imposed by the font rather than the client application. The protocol Line is the XID value referring to the STLine object to use. The XServer calls STLineGetDesignMetrics and passes the STLine object referred to by Line. The function returns the metric information about the line that was set by the font.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.56 xSTLineGetGlyphBoundsReply Struct Reference

ST Line Get Glyph Bounds reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 BoundsCount

  Number of bounding trapezoids returned.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.56.1 Detailed Description

ST Line Get Glyph Bounds reply.

The protocol response BoundsCount is the number of trapezoids that will be returned. The trapezoids are appended onto the end of the response message.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.57 xSTLineGetGlyphBoundsReq Struct Reference

ST Line Get Glyph Bounds request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineGetGlyphBounds.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

- CARD32 gc B32

  Graphics Context.

- CARD32 MaxBoundsCount B32

  Maximum number of bounds to return.

- CARD32 Bounds B32

  Type of bounding box to obtain.

4.57.1 Detailed Description

ST Line Get Glyph Bounds request.

The X_STLineGetGlyphBounds request is used to get a bounding rectangle around an STLine object referred to by the XSTLine XID. The protocol Line is the XID value referring to the STLine object to use. The protocol gc is the GC containing the transformation matrix to use. The protocol MaxBoundsCount indicates the maximum number of bounding boxes to return. The protocol BoundsType indicates the type of bounding boxes to obtain. The XServer calls STLineGetGlyphBounds and passes the STLine object referred to by Line, the STGraphics held in the GC, the MaxBoundsCount, and the BoundsType. The function returns a list of trapezoidal bounding boxes around the text and the number of trapezoids returned.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine or GC to use is not valid. BadImplementation may be returned if the STLine to use is NULL. BadAlloc may be returned if the response cannot be allocated.

4.57.2 Member Data Documentation

4.57.2.1 CARD32 Bounds xSTLineGetGlyphBoundsReq::B32

Type of bounding box to obtain.

IE: caret

The documentation for this struct was generated from the following file:

- XSTstr.h

4.58 xSTLineGetHighlightsReply Struct Reference

ST Line Get Highlights reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of highlighted regions returned.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.

- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.58.1 Detailed Description

ST Line Get Highlights reply.

The protocol response count is the number of highlighted regions in the STLine.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.59 xSTLineGetHighlightsReq Struct Reference

ST Line Get Highlights request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STLineGetHighlight.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Line B32
  XSTLine XID referring to the STLine to use.

4.59.1 Detailed Description

ST Line Get Highlights request.

The X_STLineGetHighlight request is used to get the portions of an STLine object, referred to by the XSTLine XID, that are highlighted. The protocol Line is the XID value referring to the STLine object to use. The XServer calls STLineGetHighlight and passes the STLine object referred to by Line. The function returns a list of Position, CharCount values that is appended onto the end of the response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.60 xSTLineGetMetricsReply Struct Reference

ST Line Get Metrics reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD64 LineWidth
  Maximum width of STLine in points.
- CARD64 Ascent
  Maximum Ascent of any character in text.

- CARD64 Descent
  Maximum Descent of any character in text.
- CARD64 Leading
  Leading between line.
- CARD64 base01
  Available baseline.
- CARD64 base02
  Available baseline.
- CARD64 base03
  Available baseline.
- CARD64 base04
  Available baseline.
- CARD64 base05
  Available baseline.
- CARD64 base06
  Available baseline.
- CARD64 base07
  Available baseline.
- CARD64 base08
  Available baseline.
- CARD64 base09
  Available baseline.
- CARD64 base10
  Available baseline.
- CARD64 base11
  Available baseline.
- CARD64 base12
  Available baseline.

- CARD64 base13
  Available baseline.
- CARD64 base14
  Available baseline.
- CARD64 base15
  Available baseline.
- CARD64 base16
  Available baseline.
- CARD64 base17
  Available baseline.
- CARD64 base18
  Available baseline.
- CARD64 base19
  Available baseline.
- CARD64 base20
  Available baseline.
- CARD64 base21
  Available baseline.
- CARD64 base22
  Available baseline.
- CARD64 base23
  Available baseline.
- CARD64 base24
  Available baseline.
- CARD64 base25
  Available baseline.
- CARD64 base26
  Available baseline.

- CARD64 base27
  Available baseline.
- CARD64 base28
  Available baseline.
- CARD64 base29
  Available baseline.
- CARD64 base30
  Available baseline.
- CARD64 base31
  Available baseline.
- CARD64 base32
  Available baseline.

4.60.1 Detailed Description

ST Line Get Metrics reply.

The protocol response LineWidth is a double that indicates the width of the line in points. The protocol response Ascent is a double that indicates the maximum ascent of any character in the line. The protocol response Descent is a double that indicates the maximum descent of any character in the line. The protocol response Leading is a double that indicates the leading value to be used after this line. The protocol response baseXX values are doubles that indicate the baseline values for the line.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.61 xSTLineGetMetricsReq Struct Reference

ST Line Get Metrics request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineGetMetrics.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

4.61.1 Detailed Description

ST Line Get Metrics request.

The X_STLineGetMetrics request is used to get the metrics for STLine object referred to by the XSTLine XID. These metrics need to have been set with the XSTLineSetMetrics command previously. The protocol Line is the XID value referring to the STLine object to use. The XServer calls STLineGetMetrics and passes the STLine object referred to by Line. The function returns the metric information about the line that was set previously.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.62 xSTLineGetOffsetReply Struct Reference

ST Line Get Offset reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 Position B32

  First character location within STText.

- CARD32 CharCount B32

  Number of characters in STLine.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

4.62.1 Detailed Description

ST Line Get Offset reply.

The protocol response Position is set to the first character in the line. The protocol response CharCount is the number of characters in the STLine.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.63 xSTLineGetOffsetReq Struct Reference

ST Line Get Offset request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STLineGetOffset.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Line B32
  XSTLine XID referring to the STLine to use.

4.63.1 Detailed Description

ST Line Get Offset request.

The X_STLineGetOffset request is used to find which characters are in an existing STLine object referred to by an XSTLine XID. The protocol Line is the XID value referring to the STLine object to check. The XServer calls STLineGetOffset and passes the STLine object referred to by Line. The function returns the position of the first character in the line, and the number of characters in the line.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.64 xSTLineGrowReq Struct Reference

ST Line Grow request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineGrow.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

- CARD32 Position B32

  Flag to indicate Left or Right side of STLine.

- CARD32 CharCount B32

  Number of characters to add to STLine.

4.64.1 Detailed Description

ST Line Grow request.

The X_STLineGrow request is used to increase the number of characters in an existing STLine object referred to by an XSTLine XID. The protocol Line is the XID value referring to the STLine object to grow. The protocol Position indicates the beginning or ending of the line of text. The protocol CharCount is the number of characters to add to the STLine. The XServer calls STLineGrow and passes the STLine object referred to by Line, the Position and CharCount. The function has no response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.65 xSTLineHitTestReply Struct Reference

ST Line Hit Test reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 POffset B32
  Primary Offset into STText.
- CARD32 IsLeading B32
  Flag Indicates the type of Caret produced.
- CARD32 SOffset B32
  Secondary offset int STText.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.

4.65.1 Detailed Description

ST Line Hit Test reply.

The protocol response POffset is the primary character offset into the text. The protocol response IsLeading indicates which offset to use depending on the bidi status of the line. The protocol response SOffset is the secondary character offset into the text and is used for bidi text.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.66 xSTLineHitTestReq Struct Reference

ST Line Hit Test request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineHitTest.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

- CARD64 X

  X Position.

- CARD64 Y

  Y Position.

4.66.1 Detailed Description

ST Line Hit Test request.

The X_STLineHitTest request is used to get a character position within an STLine object referred to by the XSTLine XID based on an X, Y coordinate. The protocol Line is the XID value referring to the STLine object to use. The protocol X is the X value that is within the scope of the line. The protocol Y is the Y value that is within the scope of the line. The XServer calls STLineHitTest and passes the STLine object referred to by Line, and the X and Y values. The function returns the primary offset,

secondary offset, and the IsLeading flag. The protocol response POffset is the primary character offset into the text.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.67 xSTLineMeasureTextImageReply Struct Reference

ST Line Measure Text Image reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD64 ax
  Upper left X value.
- CARD64 ay
  Upper left Y value.
- CARD64 bx
  Lower right X value.
- CARD64 by
  Lower right Y value.

4.67.1 Detailed Description

ST Line Measure Text Image reply.

The protocol response ax is a double that is the upper left X value. The protocol response ay is a double that is the upper left Y value. The protocol response bx is a double that is the lower right X value. The protocol response by is a double that is the lower right Y value.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.68 xSTLineMeasureTextImageReq Struct Reference

ST Line Measure Text Image request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineMeasureTextImage.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

4.68.1 Detailed Description

ST Line Measure Text Image request.

The X_STLineMeasureTextImage request is used to get a bounding rectangle around an STLine object referred to by the XSTLine XID. This command differs from X_STLineMeasureText because it uses the imposed justification, flush factor and other values that affect the output positioning. The protocol Line is the XID value referring to the STLine object to use. The XServer calls STLineMeasureTextImage and passes

the STLine object referred to by Line. The function returns a rectangular bounding box around the text.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.69 xSTLineMeasureTextReply Struct Reference

ST Line Measure Text reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD64 ax
  Upper left X value.
- CARD64 ay
  Upper left Y value.
- CARD64 bx
  Lower right X value.
- CARD64 by
  Lower right Y value.

4.69.1 Detailed Description

ST Line Measure Text reply.

The protocol response ax is a double that is the upper left X value. The protocol response ay is a double that is the upper left Y value. The protocol response bx is a double that is the lower right X value. The protocol response by is a double that is the lower right Y value.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.70 xSTLineMeasureTextReq Struct Reference

ST Line Measure Text request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineMeasureText.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

4.70.1 Detailed Description

ST Line Measure Text request.

The X_STLineMeasureText request is used to get a bounding rectangle around an STLine object referred to by the XSTLine XID. The protocol Line is the XID value referring to the STLine object to use. The XServer calls STLineMeasureText and passes the STLine object referred to by Line. The function returns a rectangular bounding box around the text.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.71 xSTLineMoveCaretReply Struct Reference

ST Line Move Caret reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 Offset B32
  New position of the caret.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32

Not used.

- CARD32 pad5 B32

  Not used.

4.71.1 Detailed Description

ST Line Move Caret reply.

The protocol response Offset stores the new position of the caret.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.72 xSTLineMoveCaretReq Struct Reference

ST Line Move Caret request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineMoveCaret.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

- CARD32 Offset B32

  Current cursor position within STLine.

- CARD32 Direction B32

  Direction to move cursor.

- CARD32 Movement B32

  Amount to move cursor.

4.72.1 Detailed Description

ST Line Move Caret request.

The X_STLineMoveCaret request is used to move the caret around an STLine object referred to by the XSTLine XID. The protocol Line is the XID value referring to the STLine object to use. The protocol Offset is the character that the cursor is position at within the line. The protocol Direction is the direction to move the caret within the line. The protocol Movement is the distance to move the caret. The XServer calls STLine-MoveCaret and passes the STLine object referred to by Line, the Offset, the Direction and Movement values. The function returns the new position within the Line.

BadLength may be returned if the protocol message is not of the proper length. Bad-Value may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

4.72.2 Member Data Documentation

4.72.2.1 CARD32 Movement xSTLineMoveCaretReq::B32

Amount to move cursor.

IE: clustor, character

The documentation for this struct was generated from the following file:

- XSTstr.h

4.73 xSTLineNewForWidthReq Struct Reference

ST Line New For Width request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineNewForWidth.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the new STLine.

- CARD64 Width

  Width of STLine in points.

- CARD32 Text B32

  XSTText XID referring to the STText to use.

- CARD32 Position B32

  Position of the first character within STText.

4.73.1 Detailed Description

ST Line New For Width request.

The X_STLineNewForWidth request is used to create a new STLine object and return an XSTLine XID value to the client along with the length of the new STLine. The new STLine object is to use the most characters that will fit in Width points. The protocol Text is the XID value referring to the STText object for which to create the line. The protocol Line is the XID value the client is requesting. The protocol Width is the width in points of the new Line. The protocol Position is the first character position of the new STLine object. The XServer calls STLineNewForWidth and passes the STText object referred to by Text, the Position and the Width values. The function returns an STLine object along with the number of characters that were able to fit into the STLine.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL. BadAlloc may be returned if the new STLine object cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.74 xSTLineNewReply Struct Reference

ST Line New/New For Width reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 Line B32
  XSTLine XID associated with new STLine object.
- CARD32 CharCount B32
  Number of characters placed in STLine object.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.

4.74.1 Detailed Description

ST Line New/New For Width reply.

The protocol response Line is the XSTLine XID associated with the new STLine object. The protocol response CharCount is the number of characters placed in the STLine.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.75 xSTLineNewReq Struct Reference

ST Line New request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineNew.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the new STLine.

- CARD32 Text B32

  XSTText XID referring to the STText to use.

- CARD32 Position B32

  Position of the first character within STText.

- CARD32 CharCount B32

  Number of characters contained in STLine.

4.75.1 Detailed Description

ST Line New request.

The X_STLineNew request is used to create a new STLine object and return an XSTLine XID value to the client along with the length of the new STLine. The protocol Text is the XID value referring to the STText object for which to create the line. The protocol Line is the XID value the client is requesting. The protocol Position is the first character position of the new STLine object. The protocol CharCount is the number of characters requested to be in the line. The XServer calls STLineNew and passes the STText object referred to by Text, the Position and the CharCount values. The function returns an STLine object along with the number of characters that were able to fit into the STLine. In general, the number of characters added to the line will equal the number of characters requested, unless there are not enough characters in the STText to satisfy the request.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL. BadAlloc may be returned if the new STLine object cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.76 xSTLinePositionToCaretReply Struct Reference

ST Line Position to Caret reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

Length of Reply beyond normal in Bytes.

- CARD64 SCax

  Strong Caret upper X value.

- CARD64 SCay

  Strong Caret upper Y value.

- CARD64 SCbx

  Strong Caret lower X value.

- CARD64 SCby

  Strong Caret lower Y value.

- CARD64 WCax

  Weak Caret upper X value.

- CARD64 WCay

  Weak Caret upper Y value.

- CARD64 WCbx

  Weak Caret lower X value.

- CARD64 WCby

  Weak Caret lower Y value.

- CARD32 SplitCenter B32

  Split Caret.

- CARD32 pad1 B32

  Not used.

4.76.1 Detailed Description

ST Line Position to Caret reply.

The protocol response SCax and SCay are one of the points of the Strong Caret. The protocol response SCbx and SCby are the second point of the Strong Caret. The protocol response WCax and WCay are the first point of the Weak Caret. The protocol response WCbx and WCby are the second point of the Weak Caret. The protocol response SplitCenter indicates that the caret splits the center of a compound glyph.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.77 xSTLinePositionToCaretReq Struct Reference

ST Line Position To Caret request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLinePositionToCaret.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

- CARD32 Position B32

  Position within text to get Caret.

- CARD32 IsLeading B32

  Type of Caret to return.

4.77.1 Detailed Description

ST Line Position To Caret request.

The X_STLinePositionToCaret request is used to get a caret based on a character position within an STLine object referred to by the XSTLine XID. The protocol Line is the XID value referring to the STLine object to use. The protocol Position is the character that the cursor is position at within the line. The protocol IsLeading is the type of caret to produce. The XServer calls STLinePositionToCaret and passes the STLine object referred to by Line, and the Position and IsLeading values. The function returns the two Strong Caret points, and the two Weak Caret points.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.78 xSTLineRemoveHighlightReq Struct Reference

ST Line Remove Highlight request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineRemoveHighlight.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

- CARD32 Position B32

  First position to begin removing highlight.

- CARD32 CharCount B32

  Number of characters to remove highlight.

4.78.1 Detailed Description

ST Line Remove Highlight request.

The X_STLineRemoveHighlight request is used to de-select a portion of an STLine object referred to by the XSTLine XID that was selected previously. The protocol Line is the XID value referring to the STLine object to use. The protocol Position is

the character position to start removing highlight from within the line. The protocol CharCount is the number of characters to remove highlighting from the starting point within the line. The XServer calls STLineRemoveHighlight and passes the STLine object referred to by Line, and the Position and CharCount values. The function has no response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.79 xSTLineRenderReq Struct Reference

ST Line Render request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STLineRender.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 gc B32
  Graphics Context.
- CARD32 drawable B32
  Window or Pixmap to draw on.
- CARD32 Line B32
  XSTLine XID referring to the STLine to use.

4.79.1 Detailed Description

ST Line Render request.

The X_STLineRender request is used to render an STLine object referred to by the XSTLine XID. STLineRender requires data values to be set, including callbacks for Rendering, Highlighting, Underlining, and Strikethrough. The protocol Line is the XID value referring to the STLine object to use. The protocol gc is GC containing the transformation matrix and STGraphics objects to use. The protocol drawable is the destination window or pixmap to draw on. The XServer calls STLineRender and passes the STLine object referred to by Line, and the STGraphics object found in the GC. The function has no response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine, GC, or Drawable to use is not valid. BadImplementation may be returned if the STLine to use is NULL. BadAlloc may be returned if space for STLineRender data structures cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.80 xSTLineSetMetricsReq Struct Reference

ST Line Set Metrics request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineSetMetrics.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

- CARD64 LineWidth

Maximum width of STLine in points.

- CARD64 Ascent

  Maximum Ascent of any character in text.

- CARD64 Descent

  Maximum Descent of any character in text.

- CARD64 Leading

  Leading between line.

- CARD64 base01

  Available baseline.

- CARD64 base02

  Available baseline.

- CARD64 base03

  Available baseline.

- CARD64 base04

  Available baseline.

- CARD64 base05

  Available baseline.

- CARD64 base06

  Available baseline.

- CARD64 base07

  Available baseline.

- CARD64 base08

  Available baseline.

- CARD64 base09

  Available baseline.

- CARD64 base10

  Available baseline.

- CARD64 base11

Available baseline.

- CARD64 base12
  Available baseline.
- CARD64 base13
  Available baseline.
- CARD64 base14
  Available baseline.
- CARD64 base15
  Available baseline.
- CARD64 base16
  Available baseline.
- CARD64 base17
  Available baseline.
- CARD64 base18
  Available baseline.
- CARD64 base19
  Available baseline.
- CARD64 base20
  Available baseline.
- CARD64 base21
  Available baseline.
- CARD64 base22
  Available baseline.
- CARD64 base23
  Available baseline.
- CARD64 base24
  Available baseline.
- CARD64 base25

Available baseline.

- CARD64 base26
  Available baseline.
- CARD64 base27
  Available baseline.
- CARD64 base28
  Available baseline.
- CARD64 base29
  Available baseline.
- CARD64 base30
  Available baseline.
- CARD64 base31
  Available baseline.
- CARD64 base32
  Available baseline.

4.80.1 Detailed Description

ST Line Set Metrics request.

The X_STLineSetMetrics request is used to set the metrics for STLine object referred to by the XSTLine XID. These metrics will override the metrics imposed by the STText object. The protocol Line is the XID value referring to the STLine object to modify. The protocol LineWidth is a double that indicates the width of the line in points. The protocol Ascent is a double that indicates the maximum ascent of any character in the line. The protocol Descent is a double that indicates the maximum descent of any character in the line. The protocol Leading is a double that indicates the leading value to be used after this line. The protocol baseXX values are doubles that indicate the baseline values for the line. The XServer calls STLineSetMetrics and passes the STLine object referred to by Line, along with the metric values. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.81 xSTLineShrinkReq Struct Reference

ST Line Shrink request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STLineShrink.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Line B32

  XSTLine XID referring to the STLine to use.

- CARD32 Position B32

  Flag to indicate Left or Right side of STLine.

- CARD32 CharCount B32

  Number of characters to remove from STLine.

4.81.1 Detailed Description

ST Line Shrink request.

The X_STLineShrink request is used to decrease the number of characters in an existing STLine object referred to by an XSTLine XID. The protocol Line is the XID value referring to the STLine object to shrink. The protocol Position indicates the beginning or ending of the line of text. The protocol CharCount is the number of characters to remove from the STLine. The XServer calls STLineShrink and passes the STLine object referred to by Line, the Position and CharCount. The function has no response.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTLine to use is not valid. BadImplementation may be returned if the STLine to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.82 xSTQueryVersionReply Struct Reference

ST Query Version reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD16 majorVersion B16
  Major Version of software.
- CARD16 minorVersion B16
  Minor Version of software.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32

Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.82.1 Detailed Description

ST Query Version reply.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.83 xSTQueryVersionReq Struct Reference

ST Query Version request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STQueryVersion.

- CARD16 length B16

  Request Length in Bytes.

- CARD16 majorVersion B16

  Major Version number: check for this num.

- CARD16 minorVersion B16

  Minor Version number: check for this num.

4.83.1 Detailed Description

ST Query Version request.

The X_STQueryVersion request returns version information about the XST extension.

The protocol version numbers returned indicate the version of the XST extension of the target X server. The version numbers can be compared to constants defined in the XST.h header file. Each version is a superset of the previous versions.

BadLength may be returned if the protocol message is not of the proper length.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.84 xSTScalerGetInfoReply Struct Reference

ST Scaler Get Info reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 Tag B32
  ???
- CARD32 Version B32
  Major Number/Minor Number of version.
- CARD32 SFlags B32

Mask of scaler abilities.

- CARD32 FFlags B32

  Mask of font types supported by scaler.

- CARD32 ShortNameLen B32

  Number of bytes in Short Name.

- CARD32 LongNameLen B32

  Number of bytes in Long Name.

- CARD32 NoticeLen B32

  Number of bytes in Notice.

4.84.1 Detailed Description

ST Scaler Get Info reply.

The protocol response Tag represents the ???. The protocol response Version indicates the version number of the scaler. The top 16 bits are the major number, while the bottom 16 bits are the minor number. The protocol response SFlags is a mask representing the abilities of the scaler to perform hinting, embedded bitmaps, etc. The protocol response FFlags is a mask representing the font formats that the scaler can use. The protocol response ShortNameLen indicates the length of the standard name of the scaler. The protocol response LongNameLen indicates the length of the full name of the scaler. The protocol response NoticeLen indicates the length of the notice string. Each of the strings is appended onto the end of the response message. The ShortName is first, followed by the LongName, followed by the Notice.

4.84.2 Member Data Documentation

4.84.2.1 CARD32 SFlags xSTScalerGetInfoReply::B32

Mask of scaler abilities.

IE: hinting

The documentation for this struct was generated from the following file:

- XSTstr.h

4.85 xSTScalerGetInfoReq Struct Reference

ST Scaler Get Info request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STScalerGetInfo.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Env B32
  XSTTypeEnv XID to use.
- CARD32 Scaler B32
  XSTScaler ID to use to get info.

4.85.1 Detailed Description

ST Scaler Get Info request.

The X_STScalerGetInfo request is used to get detailed information about a specific scaler. The protocol Env is an XSTTypeEnv XID value referring to the STTypeEnv structure. The protocol Scaler is an XSTScaler representing the STScaler value. The XServer calls STScalerGetInfo and passes the STTypeEnv object and the STScaler value. The function returns the name strings, version numbers and capability flags.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.86 xSTStyleClearReq Struct Reference

ST Style Clear request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STStyleClear.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Style B32
  XSTStyle XID to use.
- CARD32 mask B32
  Mask indicating the fields to clear.

4.86.1 Detailed Description

ST Style Clear request.

The X_STStyleClear request is used to clear the attribute fields of an STStyle object. The protocol Style is the XID value referring to the STStyle object to clear. The protocol mask is a bitmask indicating which fields to clear. The XServer calls STStyleIsClear and passes the STStyle object. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.87 xSTStyleCompareReply Struct Reference

ST Style Compare reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 compare B32
  Result of comparison.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.87.1 Detailed Description

ST Style Compare reply.

The protocol response compare is the returned value indicating the equality of the STStyle objects.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.88 xSTStyleCompareReq Struct Reference

ST Style Compare request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STStyleCompare.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Style1 B32
  XSTStyle XID to compare.
- CARD32 Style2 B32
  XSTStyle XID to compare.
- CARD32 mask B32
  Mask identifying fields to compare.

4.88.1 Detailed Description

ST Style Compare request.

The X_STStyleCompare request is used to compare two STStyle objects. It tells whether they are equal, subset, superset, or unequal. The protocol Style1 is the XID value referring to the first STStyle object to compare. The protocol Style2 is the XID value referring to the second STStyle object to compare. The protocol mask is a bit-mask indicating which fields to compare. The XServer calls STStyleCompare and passes the two STStyle objects. The function returns a value indicating how equal the styles are.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if either XSTStyle to use is not valid. BadImplementation may be returned if either STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.89 xSTStyleCopyAttributesReq Struct Reference

ST Style Copy Attributes request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleCopyAttributes.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 ToStyle B32

  XSTStyle XID to copy to.

- CARD32 FromStyle B32

  XSTStyle XID to copy from.

- CARD32 mask B32

  Mask indicating fields to copy.

4.89.1 Detailed Description

ST Style Copy Attributes request.

The X_STStyleCopyAttributes request is used to copy specific attributes from an STStyle object to another. The protocol ToStyle is the XID value referring to the STStyle object to modify. The protocol FromStyle is the XID value referring to the STStyle object to copy. The protocol mask indicates which of the values to copy from the STStyle object to the other. The XServer calls STStyleCopyAttributes and passes the two STStyle objects and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL

The documentation for this struct was generated from the following file:

- XSTstr.h

4.90 xSTStyleDisposeReq Struct Reference

ST Style Dispose request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STStyleDispose.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Style B32
  XSTStyle XID to use.

4.90.1 Detailed Description

ST Style Dispose request.

The X_STStyleDispose request is used to free an existing STStyle object and its associated XID value. The protocol Style is the XID value referring to the STLine object to dispose. The XServer calls STStyleDispose and passes the STStyle object. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.91 xSTStyleGetBaselinesReply Struct Reference

ST Style Get Baselines reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD64 base01

  Baseline value.

- CARD64 base02

  Baseline value.

- CARD64 base03

  Baseline value.

- CARD64 base04

  Baseline value.

- CARD64 base05

  Baseline value.

- CARD64 base06

  Baseline value.

- CARD64 base07

  Baseline value.

- CARD64 base08

  Baseline value.

- CARD64 base09

  Baseline value.

- CARD64 base10

  Baseline value.

- CARD64 base11

  Baseline value.

- CARD64 base12

  Baseline value.

- CARD64 base13

  Baseline value.

- CARD64 base14

  Baseline value.

- CARD64 base15

  Baseline value.

- CARD64 base16

  Baseline value.

- CARD64 base17

  Baseline value.

- CARD64 base18

  Baseline value.

- CARD64 base19

  Baseline value.

- CARD64 base20

  Baseline value.

- CARD64 base21

  Baseline value.

- CARD64 base22

  Baseline value.

- CARD64 base23

  Baseline value.

- CARD64 base24

  Baseline value.

- CARD64 base25

  Baseline value.

- CARD64 base26

  Baseline value.

- CARD64 base27

  Baseline value.

- CARD64 base28

  Baseline value.

- CARD64 base29

  Baseline value.

- CARD64 base30

  Baseline value.

- CARD64 base31

  Baseline value.

- CARD64 base32

  Baseline value.

4.91.1 Detailed Description

ST Style Get Baselines reply.

The protocol response baseXX is filled in with the appropriate baseline value.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.92 xSTStyleGetBaselinesReq Struct Reference

ST Style Get Baselines request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleGetBaselines.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

- CARD32 BaselineFlag B32

  Flag indicating which font baseline to use.

4.92.1 Detailed Description

ST Style Get Baselines request.

The X_STStyleGetBaselines request attempts to retrieve baseline information from a specified STStyle.

The protocol Style is an XID referring to the specific STStyle from which the user wants to obtain the baseline information. The XServer calls STStyleGetBaselines and passes the STStyle structure and BaselineFlag information. The function returns the baseline values available for the font associated with the style.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.93 xSTStyleGetDesignMetricsReply Struct Reference

ST Style Get Design Metrics reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD64 Ascent
  Maximum ascent of line in design units.
- CARD64 Descent

Maximum descent of line in design units.

- CARD64 Leading

  Maximum leading of line in design units.

4.93.1 Detailed Description

ST Style Get Design Metrics reply.

The protocol response Ascent refers to the ascent of the text in design units. The protocol response Descent refers to the descent of the text in design units. The protocol response Leading refers to the leading between lines in design units.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.94 xSTStyleGetDesignMetricsReq Struct Reference

ST Style Get Design Metrics request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleGetDesignMetrics.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

4.94.1 Detailed Description

ST Style Get Design Metrics request.

The X_STStyleGetDesignMetrics request attempts to retrieve metric information about the font associated with a specified STStyle.

The protocol Style is an XID referring to the specific STStyle from which the user wants to obtain the metrics information. The XServer calls STStyleGetDesignMetrics and passes the STStyle structure. The function returns the metrics values in design units associated with the font in the STStyle.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.95 xSTStyleGetEffectsReply Struct Reference

ST Style Get Effects reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 StyleEffects B32
  Mask of style effects.
- CARD32 STEffects B32

Mask of strikethrough effects.

- CARD32 ULEffects B32

  Mask of underline effects.

- CARD32 mask B32

  Mask indicating which fields to set.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

4.95.1 Detailed Description

ST Style Get Effects reply.

The protocol response StyleEffects represents a mask of style effects in use such as Ligature and Kerning settings. The protocol response STEffects is a mask indicating the type of strikethrough in use. The protocol response ULEffects is a mask that indicates the type of underline in use. The protocol response mask indicates which of the values were set in the STStyle object.

4.95.2 Member Data Documentation

4.95.2.1 CARD32 ULEffects xSTStyleGetEffectsReply::B32

Mask of underline effects.

IE: dash, thick

4.95.2.2 CARD32 STEffects xSTStyleGetEffectsReply::B32

Mask of strikethrough effects.

IE: double

4.95.2.3 CARD32 StyleEffects xSTStyleGetEffectsReply::B32

Mask of style effects.

IE: Ligature, Kerning

The documentation for this struct was generated from the following file:

- XSTstr.h

4.96 xSTStyleGetEffectsReq Struct Reference

ST Style Get Effects request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleGetEffects.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

4.96.1 Detailed Description

ST Style Get Effects request.

The X_STStyleGetEffects request is used to get the style effects such as Ligature and Kerning use, and to get the effects for Strikethrough and Underline for an STStyle object. The protocol Style is the XID value referring to the STStyle object to check. The XServer calls STStyleGetEffects and passes the STStyle object. The function returns the StyleEffects, the STEffects and ULEffects flags, and the mask indicating which of these effects have been set.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.97 xSTStyleGetFontReply Struct Reference

ST Style Get Font reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD64 Size
  Point size of the font.
- CARD32 Font B32
  XST Font ID of the style.
- CARD32 LanguageID B32
  Language used within style.
- CARD32 BaselineFlag B32
  Baseline set within style.
- CARD32 mask B32
  Mask indicating the fields that have been set.

4.97.1 Detailed Description

ST Style Get Font reply.

The protocol response Size is a double representing the size of the font. The protocol response Font is the STFont value. The protocol response LanguageID represents the language in use. The protocol response BaselineFlag indicates which baseline is used. The protocol count indicates the number of Features appended onto the end of the response message. The protocol mask tells which values are set within the STStyle object.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.98 xSTStyleGetFontReq Struct Reference

ST Style Get Font request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STStyleGetFont.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Style B32
  XSTStyle XID to use.

4.98.1 Detailed Description

ST Style Get Font request.

The X_STStyleGetFont request is used to get the font, size, various flags, and features of an STStyle object. The protocol Style is the XID value referring to the STStyle object to check. The XServer calls STStyleGetFont and passes the STStyle object. The

command returns the Font, its Size, language used during the style, the baseline being used, and any features set for the style. It also returns a mask that shows which of these values was set in the STStyle object.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.99 xSTStyleGetLayoutEngineReply Struct Reference

ST Style Get LayoutEngine reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 LayoutEngine B32

  XSTLayoutEngine XID associated with Style.

- CARD32 mask B32

  Mask indicating the fields that have been set.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

4.99.1 Detailed Description

ST Style Get LayoutEngine reply.

The protocol LayoutEngine is the XID value referring to the STLayoutEngine object associated with the STStyle. The protocol mask is the mask bits indicating which fields have been set.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.100 xSTStyleGetLayoutEngineReq Struct Reference

ST Style Get LayoutEngine request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleGetLayoutEngine.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

4.100.1 Detailed Description

ST Style Get LayoutEngine request.

The X_STStyleGetLayoutEngine request is used to get the layout engine in use. The protocol Style is the XID value referring to the STStyle object to use.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.101 xSTStyleGetOptionsReply Struct Reference

ST Style Get Options reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 ImposeWidth B32
  Width imposed on the Style.
- CARD32 BeforeWSS B32
  Amount to shift before the stream.
- CARD32 AfterWSS B32
  Amount to shift after the stream.

- CARD32 CrossSS B32

  Cross stream shift.

- CARD32 Tracking B32

  Space between characters.

- CARD32 HangingIF B32

  Hanging inhibit factor.

- CARD32 KerningIF B32

  Kerning inhibit factor.

- CARD32 DecompositionIF B32

  Decomposition inhibit factor.

- CARD32 mask B32

  Mask indicating which fields have been set.

4.101.1 Detailed Description

ST Style Get Options reply.

The protocol response ImposeWidth is the width imposed on the style. The protocol response BeforeWSS is the amount to shift before the stream. The protocol response AfterWSS is the amount to shift after the stream. The protocol response CrossSS is the cross stream shift. The protocol response Tracking is the amount of space between the characters. The protocol response HangingIF is the hanging inhibit factor. The protocol response KerningIF is the kerning inhibit factor. The protocol response DecompositionIF is the decomposition inhibit factor. The protocol response mask indicates which of the values to were previously modified.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.102 xSTStyleGetOptionsReq Struct Reference

ST Style Get Options request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleGetOptions.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

4.102.1 Detailed Description

ST Style Get Options request.

The X_STStyleGetOptions request is used to retrieve special kerning values from an STStyle object. The protocol Style is the XID value referring to the STStyle object to access. The XServer calls STStyleGetOptions and passes the STStyle object. The function returns several values along with a mask indicating the values that have been previously set.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.103 xSTStyleGetScalerReply Struct Reference

ST Style Get Scaler reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 Scaler B32

  XSTScaler ID to use.

- CARD32 Hints B32

  Hinting mode to use.

- CARD32 Sbits B32

  Flag to use embedded bitmaps.

- CARD32 mask B32

  Mask indicating the fields that have been set.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

4.103.1 Detailed Description

ST Style Get Scaler reply.

The protocol response Scaler is the STScaler value currently set. The protocol response Hints is a flag indicating the type of hinting the scaler is using. The protocol response Sbits indicates if the scaler is using embedded bitmaps. The protocol response $*$ mask indicates which of the values are set in the STStyle object.

4.103.2 Member Data Documentation

4.103.2.1 CARD32 Hints xSTStyleGetScalerReply::B32

Hinting mode to use.

IE: Apply Hints, Autohint

The documentation for this struct was generated from the following file:

- XSTstr.h

4.104 xSTStyleGetScalerReq Struct Reference

ST Style Get Scaler request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STStyleGetScaler.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Style B32
  XSTStyle XID to use.

4.104.1 Detailed Description

ST Style Get Scaler request.

The X_STStyleGetScaler request is used to get the scaler in use, the hinting method in use, and whether embedded bitmaps are used, in an STStyle object. The protocol Style is the XID value referring to the STStyle object to check. The XServer calls STStyleGetScaler and passes the STStyle object. The function returns which scaler is being used, which hinting method is used, whether embedded bitmaps are enabled, and a mask indicating which of these fields was previously set.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.105 xSTStyleIsEmptyReply Struct Reference

ST Style Is Empty reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 isempty B32
  Result of test.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32

Not used.

- CARD32 pad5 B32

  Not used.

4.105.1 Detailed Description

ST Style Is Empty reply.

The protocol response isempty is the returned value.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.106 xSTStyleIsEmptyReq Struct Reference

ST Style Is Empty request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleIsEmpty.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

4.106.1 Detailed Description

ST Style Is Empty request.

The X_STStyleIsEmpty request is used to check if an STStyle object is empty. The protocol Style is the XID value referring to the STStyle object to check. The XServer calls STStyleIsEmpty and passes the STStyle object. The function returns a value indicating whether the STStyle is empty.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.107 xSTStyleNewCopyReq Struct Reference

ST Style New Copy request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleNewCopy.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 OldStyle B32

  XSTStyle XID to copy.

- CARD32 NewStyle B32

  New XSTStyle XID to associate with STStyle.

- CARD32 Type B32

  Command Type to perform.

4.107.1 Detailed Description

ST Style New Copy request.

The X_STStyleNewCopy request is used to create a new STStyle object. This command is used for creating an empty STStyle, a default STStyle or to copy an existing STStyle. The protocol Env is the XID value referring to the STTypeEnv object to associate the new STStyle object with. The protocol OldStyle is the XID value referring to the STStyle object to copy. This value is zero (0) if the command is performing a NewEmpty or NewDefault operation. The protocol NewStyle is the XID value the client is requesting for the new XSTStyle. The protocol Type is not used for performing a copy operation. It indicates which NewEmpty or NewDefault operation to perform. The XServer calls STStyleNewCopy on a copy request, passing the STTypeEnv object, and the STStyle object referred to by OldStyle XID. For the NewEmpty request, the XServer calls STStyleNewEmpty and passes the STTypeEnv object. For the NewDefault request the XServer calls STStyleNewDefault and passes the STTypeEnv object. The functions each return an STStyle object.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv or XSTStyle (OldStyle) to use is not valid. BadImplementation may be returned if the STTypeEnv or STStyle to use is NULL. BadAlloc may be returned if the new STStyle object cannot be allocated.

4.107.2 Member Data Documentation

4.107.2.1 CARD32 Type xSTStyleNewCopyReq::B32

Command Type to perform.

NewStyle, NewCopy, etc

4.107.2.2 CARD32 OldStyle xSTStyleNewCopyReq::B32

XSTStyle XID to copy.

0 if not copying

The documentation for this struct was generated from the following file:

- XSTstr.h

4.108 xSTStyleOverwriteAttributesReq Struct Reference

ST Style Overwrite Attributes request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleOverwriteAttributes.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 ToStyle B32

  XSTStyle XID to overwrite to.

- CARD32 FromStyle B32

  XSTStyle XID to overwrite from.

- CARD32 mask B32

  Mask indicating fields to overwrite.

4.108.1 Detailed Description

ST Style Overwrite Attributes request.

The X_STStyleOverwriteAttributes request is used to copy specific attributes from an STStyle object to another. The protocol ToStyle is the XID value referring to the STStyle object to modify. The protocol FromStyle is the XID value referring to the STStyle object to copy. The protocol mask indicates which of the values to copy from the STStyle object to the other. The XServer calls STStyleOverwriteAttributes and passes the two STStyle objects and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use

The documentation for this struct was generated from the following file:

- XSTstr.h

4.109 xSTStyleResetAttributesReq Struct Reference

ST Style Reset Attributes request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STStyleResetAttributes.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Style B32
  XSTStyle XID to use.
- CARD32 mask B32
  Mask indicating fields to reset.

4.109.1 Detailed Description

ST Style Reset Attributes request.

The X_STStyleResetAttributes request is used to reset specific attributes for an STStyle object. The protocol Style is the XID value referring to the STStyle object to modify. The protocol mask indicates which of the values to reset in the STStyle object. The XServer calls STStyleResetAttributes and passes the STStyle object and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.110 xSTStyleScalerControlReply Struct Reference

ST Style Scaler Control reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of bytes of data returned from scaler.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.110.1 Detailed Description

ST Style Scaler Control reply.

The protocol response count is the number of bytes returned. The data is appended onto the end of the response message.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.111 xSTStyleScalerControlReq Struct Reference

ST Style Scaler Control request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleScalerControl.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

- CARD32 count B32

  Number of bytes of data to send to the scaler.

4.111.1 Detailed Description

ST Style Scaler Control request.

The X_STStyleScalerControl request is used to send the scaler associated with an STStyle object a string of data and get a response. The protocol Style is the XID value referring to the STStyle object to use. The protocol count indicates the number

of bytes appended onto the end of the request to send to the scaler. The XServer calls STStyleScalerControl and passes the STStyle object along with the data and its length. The function returns a string of data and its length.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL. BadAlloc may be returned if the response could not be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.112 xSTStyleSetEffectsReq Struct Reference

ST Style Set Effects request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleSetEffects.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

- CARD32 StyleEffects B32

  Mask of style effects.

- CARD32 STEffects B32

  Mask of strikethrough effects.

- CARD32 ULEffects B32

  Mask of underline effects.

- CARD32 mask B32

  Mask indicating which fields to set.

4.112.1 Detailed Description

ST Style Set Effects request.

The X_STStyleSetEffects request is used to set the style effects such as Ligature and Kerning use, and to set the effects for Strikethrough and Underline for an STStyle object. The protocol Style is the XID value referring to the STStyle object to modify. The protocol StyleEffects represents a mask of style effects to use such as Ligature and Kerning settings. The protocol STEffects is a mask indicating the type of strikethrough to use. The protocol ULEffects is a mask that indicates the type of underline to use. The protocol mask indicates which of the values to set in the STStyle object. The XServer calls STStyleSetEffects and passes the STStyle object, the StyleEffects, the STEffects and ULEffects flags, and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

4.112.2 Member Data Documentation

4.112.2.1 CARD32 ULEffects xSTStyleSetEffectsReq::B32

Mask of underline effects.

IE: dash, thick

4.112.2.2 CARD32 STEffects xSTStyleSetEffectsReq::B32

Mask of strikethrough effects.

IE: double

4.112.2.3 CARD32 StyleEffects xSTStyleSetEffectsReq::B32

Mask of style effects.

IE: Ligature, Kerning

The documentation for this struct was generated from the following file:

- XSTstr.h

4.113 xSTStyleSetFontReq Struct Reference

ST Style Set Font request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STStyleSetFont.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Style B32
  XSTStyle XID to use.
- CARD64 Size
  Point Size of Font for length of style.
- CARD32 Font B32
  XSTFont ID for length of style.
- CARD32 LanguageID B32
  Language used for length of style.
- CARD32 BaselineFlag B32
  Baseline to use for length of style.
- CARD32 mask B32
  Mask indicating the fields to clear.

4.113.1 Detailed Description

ST Style Set Font request.

The X_STStyleSetFont request is used to set the font, size, various flags, or features of an STStyle object. The protocol Style is the XID value referring to the STStyle object to modify. The protocol Size is a double representing the size of the font to use. The protocol Font is the STFont value. The protocol LanguageID is the language associated with this style. The protocol BaselineFlag indicates which baseline to use. The protocol count indicates the number of Features appended onto the end of the request message to set in the font. The protocol mask tells which values to set within the STStyle object. The XServer calls STStyleSetFont and passes the STStyle object, the STFont, its Size, Flags and Features along with the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.114 xSTStyleSetLayoutEngineReq Struct Reference

ST Style Set LayoutEngine request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleSetLayoutEngine.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

- CARD32 LayoutEngine B32

  XSTLayoutEngine XID to use.

- CARD32 mask B32

  Mask indicating which fields to set.

4.114.1 Detailed Description

ST Style Set LayoutEngine request.

The X_STStyleSetLayoutEngine request is used to set the layout engine to use. The protocol Style is the XID value referring to the STStyle object to set. The protocol LayoutEngine is the XID value referring to the STLayoutEngine object to associate with the STStyle. The protocol mask is the mask bits indicating which fields to set.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.115 xSTStyleSetOptionsReq Struct Reference

ST Style Set Options request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STStyleSetOptions.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Style B32

  XSTStyle XID to use.

- CARD32 ImposeWidth B32

Width imposed on the Style.

- CARD32 BeforeWSS B32
  Amount to shift before the stream.
- CARD32 AfterWSS B32
  Amount to shift after the stream.
- CARD32 CrossSS B32
  Cross stream shift.
- CARD32 Tracking B32
  Space between characters.
- CARD32 HangingIF B32
  Hanging inhibit factor.
- CARD32 KerningIF B32
  Kerning inhibit factor.
- CARD32 DecompositionIF B32
  Decomposition inhibit factor.
- CARD32 mask B32
  Mask indicating which fields to set.

4.115.1 Detailed Description

ST Style Set Options request.

The X_STStyleSetOptions request is used to set special kerning values for an STStyle object. The protocol Style is the XID value referring to the STStyle object to access. The protocol ImposeWidth is the width imposed on the style. The protocol BeforeWSS is the amount to shift before the stream. The protocol AfterWSS is the amount to shift after the stream. The protocol CrossSS is the cross stream shift. The protocol Tracking is the amount of space between the characters. The protocol HangingIF is the hanging inhibit factor. The protocol KerningIF is the kerning inhibit factor. The protocol DecompositionIF is the decomposition inhibit factor. The protocol mask indicates which of the values to were previously modified. The XServer calls STStyleSetOptions and passes the STStyle object and the options. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.116 xSTStyleSetScalerReq Struct Reference

ST Style Set Scaler request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STStyleSetScaler.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Style B32
  XSTStyle XID to use.
- CARD32 Scaler B32
  XSTScaler ID to use.
- CARD32 Hints B32
  Hinting mode to use.
- CARD32 Sbits B32
  Flag to use embedded bitmaps.
- CARD32 mask B32
  Mask indicating the fields to set.

4.116.1 Detailed Description

ST Style Set Scaler request.

The X_STStyleSetScaler request is used to set the scaler to use, the hinting method to use, and whether to use embedded bitmaps or not, for an STStyle object. The protocol Style is the XID value referring to the STStyle object to modify. The protocol Scaler is the STScaler value to set. The protocol Hints is a flag indicating the type of hinting the scaler is to user. The protocol Sbits indicates if the scaler is to use embedded bitmaps. The protocol mask indicates which of the values to set in the STStyle object. The XServer calls STStyleSetScaler and passes the STStyle object, the scaler, the hints and Sbits flags, and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

4.116.2 Member Data Documentation

4.116.2.1 CARD32 Hints xSTStyleSetScalerReq::B32

Hinting mode to use.

IE: Apply Hints, Autohint

The documentation for this struct was generated from the following file:

- XSTstr.h

4.117 xSTStyleUnderwriteAttributesReq Struct Reference

ST Style Underwrite Attributes request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

ST Request: X_STStyleUnderwriteAttributes.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 ToStyle B32

  XSTStyle XID to underwrite to.

- CARD32 FromStyle B32

  XSTStyle XID to underwrite from.

- CARD32 mask B32

  Mask indicating fields to underwrite.

4.117.1 Detailed Description

ST Style Underwrite Attributes request.

The X_STStyleUnderwriteAttributes request is used to copy specific attributes from an STStyle object to another. The protocol ToStyle is the XID value referring to the STStyle object to modify. The protocol FromStyle is the XID value referring to the STStyle object to copy. The protocol mask indicates which of the values to copy from the STStyle object to the other. The XServer calls STStyleUnerwriteAttributes and passes the two STStyle objects and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle to use is not valid. BadImplementation may be returned if the STStyle to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.118 xSTTextAugmentStyleReq Struct Reference

ST Text Augment Style request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextAugmentStyle.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to the STText to use.

- CARD32 Style B32

  XSTStyle XID referring to the STStyle to use.

- CARD32 FirstChar B32

  Starting character of change.

- CARD32 CharCount B32

  Number of characters associated with change.

4.118.1 Detailed Description

ST Text Augment Style request.

The X_STTextAugmentStyle request is used to add fields set in the supplied style to any styles present in the region of characters. The protocol Text is the XID value referring to the STText object to modify. The protocol Style is the supplied XSTStyle XID referring to an STStyle object. The protocol FirstChar is the first character whose STStyle is to be augmented. The protocol CharCount is the number of characters whose STStyle objects are to be augmented.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid or the FirstChar is greater than the current length of the text, or CharCount is greater than the current length of the text. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.119 xSTTextClearReq Struct Reference

ST Text Clear request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextClear.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to STText to clear.

4.119.1 Detailed Description

ST Text Clear request.

The X_STTextClear request is used to clear the attribute values of the STText object referred to by the XSTText XID. The protocol Text is the XID value referring to the STText object to clear. The XServer calls STTextClear and passes the STText value referred to by Text. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.120 xSTTextCopyAttributesReq Struct Reference

ST Text Copy Attributes request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextCopyAttributes.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 FromText B32

  XSTText XID referring to STText to copy from.

- CARD32 ToText B32

  XSTText XID referring to STText to copy to.

- CARD32 mask B32

  Mask identifying fields to copy.

4.120.1 Detailed Description

ST Text Copy Attributes request.

The X_STTextCopyAttributes request is used to copy the metrics, controls, and fallbacks from one STText object to another. Both STText objects are referred to by the XSTText XIDs. The protocol FromText is the XID value referring to the STText object to copy. The protocol ToText is the XID value referring to the STText object to modify. The protocol mask dictates the portions of the FromText object to copy over to the ToText object. The XServer calls STTextCopyAttributes and passes the two STText objects, and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.121 xSTTextDisposeReq Struct Reference

ST Text Dispose request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STTextDispose.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Text B32
  XSTText XID referring to STText to dispose.

4.121.1 Detailed Description

ST Text Dispose request.

The X_STTextDispose request is used to free up memory used by an STText object and free the resource ID used by the XSTText ID. The protocol Text is the XID value referring to the STText object to dispose. The XServer calls STTextDispose and passes the STText value referred to by Text. There is no response to the XSTTextDispose command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.122 xSTTextFindMissingCharsReply Struct Reference

ST Text Find Missing Chars reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of offset, length pairs returned.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.122.1 Detailed Description

ST Text Find Missing Chars reply.

The protocol response count is the number of pairs returned. The pairs are appended onto the end of the response message.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.123 xSTTextFindMissingCharsReq Struct Reference

ST Text Find Missing Chars request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextFindMissingChars.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to the STText to use.

4.123.1 Detailed Description

ST Text Find Missing Chars request.

The X_STTextFindMissingChars request is used to find which character positions within an STText object do not have coverage by the font contained in the STStyle that is associated with it, or by one of the Font Fallbacks in the STText or STTypeEnv objects. The command returns a collection of offset, length pairs indicating the characters that cannot be displayed. The protocol Text is the XID value referring to the STText object to check. The XServer calls STTextFindMissingChars and passes the STText object referred to by Text. The function returns a list of offset, length pairs and the number of pairs.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.124 xSTTextGetCommonStyleReply Struct Reference

ST Text Get Common Style reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 Style B32
  XSTStyle XID referring to STStyle found.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.124.1 Detailed Description

ST Text Get Common Style reply.

The protocol response Style is the XSTStyle XID that refers to the new STStyle object returned.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.125 xSTTextGetCommonStyleReq Struct Reference

ST Text Get Common Style request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextGetCommonStyle.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to the STText to use.

- CARD32 Position B32

  Character position within text to get STStyle.

- CARD32 CharCount B32

  Number of characters in text to create STStyle.

4.125.1 Detailed Description

ST Text Get Common Style request.

The X_STTextGetCommonStyle request is used to get an XSTStyle XID, referring to a new STStyle object, that represents the Style common to a section of text in the STText object, referred to by the XSTText XID. This new style may include information from several styles, or a single style. The protocol Text is the XID value referring to the STText object to check. The protocol Position indicates the first character location in the text string to check. The protocol CharCount indicates the number of characters in the text string to check. The XServer calls STTextGetCommonStyle and passes the STText object referred to by Text, the Position to begin the check and CharCount for the number of characters to check. The function returns an STStyle value that contains all of the elements of the styles found that are the same.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL. BadAlloc may be returned if space for the new STStyle cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.126 xSTTextGetControlsReply Struct Reference

ST Text Get Controls reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 Direction B32

  Direction of text flow.

- CARD32 Justification B32

  Justification of the text.

- CARD32 FlushFactor B32

  Degree to which text is flush with the margins.

- CARD32 Language B32

  Language of the text.

- CARD32 Policy B32

  Font Fallback Policy.

- CARD32 LayoutOptions B32

  ICU Layout options.

- CARD32 mask B32

  Mask identifying which fields have been set.

4.126.1 Detailed Description

ST Text Get Controls reply.

The protocol response Direction represents the direction of text layout. The protocol response Justification represents the justification used on the text. The protocol response FlushFactor represents the degree to which the text is Flush with the margins. The protocol response Language represents the language ID value referring to the input method to use for this text region. The protocol response Policy represents the font fallback policy override for the text region. The protocol response LayoutOptions represents the ICU layout options for the text region. The protocol response mask indicates which of these controls was previously set.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.127 xSTTextGetControlsReq Struct Reference

ST Text Get Controls request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextGetControls.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to STText to use.

4.127.1 Detailed Description

ST Text Get Controls request.

The X_STTextGetControls request is used to get the attribute values of the STText object referred to by the XSTText XID. These attributes pertain to direction of the text, it's justification, the language present in the text and Font Fallback overrides. The protocol Text is the XID value referring to the STText object to modify. The XServer calls STTextGetControls and passes the STText value referred to by Text. The command returns the controls along with a mask indicating which of the controls have been set for this STText object.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.128 xSTTextGetFontFallbacksReply Struct Reference

ST Text Get Font Fallbacks reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of XSTFont IDs in fallback list.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.128.1 Detailed Description

ST Text Get Font Fallbacks reply.

The protocol response count represents the number of font fallback Font IDs. These Font IDs are appended onto the end of the response.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.129 xSTTextGetFontFallbacksReq Struct Reference

ST Text Get Font Fallbacks request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextGetFontFallbacks.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to STText to use.

4.129.1 Detailed Description

ST Text Get Font Fallbacks request.

The X_STTextGetFontFallbacks request is used to get the override the STTypeEnv Font fallback values for the STText object referred to by the XSTText XID. The protocol Text is the XID value referring to the STText object to modify. The XServer calls STTextGetControls and passes the STText value referred to by Text. The command returns a list of Font Fallback Font IDs, and the number of Font IDs.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.130 xSTTextGetMetricsReply Struct Reference

ST Text Get Metrics reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD64 TextWidth

  Width of text in points.

- CARD64 Ascent

  Maximum Ascent of any character in text.

- CARD64 Descent

  Maximum Descent of any character in text.

- CARD64 Leading

  Leading between line.

- CARD64 base01

  Available baseline.

- CARD64 base02

  Available baseline.

- CARD64 base03

  Available baseline.

- CARD64 base04

  Available baseline.

- CARD64 base05

  Available baseline.

- CARD64 base06
  Available baseline.
- CARD64 base07
  Available baseline.
- CARD64 base08
  Available baseline.
- CARD64 base09
  Available baseline.
- CARD64 base10
  Available baseline.
- CARD64 base11
  Available baseline.
- CARD64 base12
  Available baseline.
- CARD64 base13
  Available baseline.
- CARD64 base14
  Available baseline.
- CARD64 base15
  Available baseline.
- CARD64 base16
  Available baseline.
- CARD64 base17
  Available baseline.
- CARD64 base18
  Available baseline.
- CARD64 base19
  Available baseline.

- CARD64 base20
  Available baseline.
- CARD64 base21
  Available baseline.
- CARD64 base22
  Available baseline.
- CARD64 base23
  Available baseline.
- CARD64 base24
  Available baseline.
- CARD64 base25
  Available baseline.
- CARD64 base26
  Available baseline.
- CARD64 base27
  Available baseline.
- CARD64 base28
  Available baseline.
- CARD64 base29
  Available baseline.
- CARD64 base30
  Available baseline.
- CARD64 base31
  Available baseline.
- CARD64 base32
  Available baseline.

4.130.1 Detailed Description

ST Text Get Metrics reply.

The protocol response TextWidth is a double that represents the width of the text in points. The protocol response Ascent is a double that represents the maximum Ascent of any character in the text string. The protocol response Descent is a double that represents the maximum Descent of any character in the text string. The protocol response Leading is a double that represents the leading between text regions. The protocol response baseXX are doubles that represent the available baselines for this text string.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.131 xSTTextGetMetricsReq Struct Reference

ST Text Get Metrics request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STTextGetMetrics.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Text B32
  XSTText XID referring to STText to use.

4.131.1 Detailed Description

ST Text Get Metrics request.

The X_STTextGetMetrics request is used to get the metric attribute values of the STText object referred to by the XSTText XID. This metric information is imposed information

set by the XSTTextSetMetrics command. The protocol Text is the XID value referring to the STText object to check. The XServer calls STTextSetMetrics and passes the STText value referred to by Text. The command returns the metric information stored in the STText.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.132 xSTTextGetStyleReply Struct Reference

ST Text Get Style reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 Style B32

  XSTStyle XID referring to STStyle found.

- CARD32 FirstChar B32

  First character in STText where STStyle starts.

- CARD32 CharCount B32

  Number of characters in STText STStyle covers.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

4.132.1 Detailed Description

ST Text Get Style reply.

The protocol response Style is the XSTStyle XID that refers to the STStyle object returned. The protocol response FirstChar marks the first character of the STText object that the STStyle object applies to. The protocol response CharCount indicates the number of characters of the STText object that the STStyle object applies to.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.133 xSTTextGetStyleReq Struct Reference

ST Text Get Style request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextGetStyle.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

XSTText XID referring to the STText to use.

- CARD32 Position B32

  Character position within text to get STStyle.

4.133.1 Detailed Description

ST Text Get Style request.

The X_STTextGetStyle request is used to get an XSTStyle XID, referring to the STStyle object, that applies to a specific location of text in the STText object, referred to by the XSTText XID. The protocol Text is the XID value referring to the STText object to check. The protocol Position indicates the character location in the text string to check. The XServer calls STTextGetStyle and passes the STText object referred to by Text, and the Position to check. The function returns an STStyle value that includes the Position within it's coverage of the STText object along with it's start and length.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.134 xSTTextGetTextReq Struct Reference

ST Text Get Text request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextGetText.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to STText to use.

4.134.1 Detailed Description

ST Text Get Text request.

The X_STTextGetText request is used to get the text string of an STText object referred to by the XSTText XID. The protocol Text is the XID value referring to the STText object to query. The XServer calls STTextSetText and passes the STText object referred to by Text. The command returns a pointer to the text string, and the number of characters in the string. The XServer calls STIntCountUTF16Bytes to get the number of bytes used by the string.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.135 xSTTextNewCopyReq Struct Reference

ST Text New Copy request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextNewCopy.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 OldText B32

XSTText XID referring to STText to copy.

- CARD32 NewText B32

  XSTText XID referring to new STText.

4.135.1 Detailed Description

ST Text New Copy request.

The X_STTextNewCopy request is used to create a new STText object and the XSTText XID value associated with it by copying an existing STText object. The protocol OldText is the XID value referring to the existing STText object. The protocol NewText is the XID value requested for the STText structure. The XServer calls STTextNewCopy and passes the STTypeEnv structure and the STText value referred to by OldText. The function returns an STText object.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if the Text object cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.136 xSTTextNewReq Struct Reference

ST Text New/New Empty request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextNew.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 Text B32

  XSTText XID referring to new STText.

- CARD32 count B32

  Number of utf16 characters in the string.

4.136.1 Detailed Description

ST Text New/New Empty request.

The X_STTextNew request is used to create a new STText object and the XSTText XID value associated with it. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Text is the XID value requested for the STText structure. The protocol count contains the number of characters in the text string. For XSTTextNewEmpty, count is set to 0. The text string is appended onto the end of the request message in UTF-16 format. The XServer calls STTextNew and passes the STTypeEnv structure, the text string, and the number of characters in the string. The function returns an STText object.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if the Text object cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.137 xSTTextOverwriteStyleReq Struct Reference

ST Text Overwrite Style request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType
  ST Request: X_STTextOverwriteStyle.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Text B32
  XSTText XID referring to the STText to use.
- CARD32 Style B32
  XSTStyle XID referring to the STStyle to use.
- CARD32 FirstChar B32
  Starting character of change.
- CARD32 CharCount B32
  Number of characters associated with change.

4.137.1 Detailed Description

ST Text Overwrite Style request.

The X_STTextOverwriteStyle request is used to change fields in the covered STStyle objects that are set in the supplied STStyle object. The protocol Text is the XID value referring to the STText object to modify. The protocol Style is the supplied XSTStyle XID referring to an STStyle object. The protocol FirstChar is the first character whose STStyle is to be changed. The protocol CharCount is the number of characters whose STStyle objects are to be changed.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid or the FirstChar is greater than the current length of the text, or CharCount is greater than the current length of the text. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.138 xSTTextResetAttributesReq Struct Reference

ST Text Reset Attibutes request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextResetAttributes.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to STText to use.

- CARD32 mask B32

  Mask identifying fields to reset.

4.138.1 Detailed Description

ST Text Reset Attibutes request.

The X_STTextResetAttributes request is used to reset the metrics, controls, and fallbacks of an STText object referred to by the XSTText XID. The protocol Text is the XID value referring to the STText object to reset. The protocol mask dictates the portions of the Text object to reset. The XServer calls STTextResetAttributes and passes the STText object referred to by Text, and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.139 xSTTextSetControlsReq Struct Reference

ST Text Set Controls request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STTextSetControls.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Text B32
  XSTText XID referring to STText to use.
- CARD32 Direction B32
  Direction of text flow.
- CARD32 Justification B32
  Justification of the text.
- CARD32 FlushFactor B32
  Degree to which text is flush with the margins.
- CARD32 Language B32
  Language of the text.
- CARD32 Policy B32
  Font Fallback Policy.
- CARD32 LayoutOptions B32
  ICU Layout options.
- CARD32 mask B32
  Mask identifying which fields to set.

4.139.1 Detailed Description

ST Text Set Controls request.

The X_STTextSetControls request is used to set the attribute values of the STText object referred to by the XSTText XID. These attributes pertain to direction of the text, it's justification, the language present in the text and Font Fallback overrides. The protocol

Text is the XID value referring to the STText object to modify. The protocol Direction represents the direction of text layout. The protocol Justification represents the justification used on the text. The protocol FlushFactor represents the degree to which the text is Flush with the margins. The protocol Language represents the language ID value referring to the input method to use for this text region. The protocol Policy represents the font fallback policy override for the text region. The protocol LayoutOptions represents the ICU layout options for the text region. The protocol mask tells ST which of the controls to set. The XServer calls STTextSetControls and passes the STText value referred to by Text, the controls, and the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.140 xSTTextSetFontFallbacksReq Struct Reference

ST Text Set Font Fallbacks request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextSetFontFallbacks.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to STText to use.

- CARD32 count B32

  Number of XSTFont IDs in fallback list.

4.140.1 Detailed Description

ST Text Set Font Fallbacks request.

The X_STTextSetFontFallbacks request is used to set the override the STTypeEnv Font fallback values for the STText object referred to by the XSTText XID. The protocol Text is the XID value referring to the STText object to modify. The protocol count represents the number of font fallback Font IDs are appended onto the end of the request. The XServer calls STTextSetControls and passes the STText value referred to by Text, the Font Fallback Font IDs, and the number of Font IDs. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.141 xSTTextSetMetricsReq Struct Reference

ST Text Set Metrics request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextSetMetrics.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to STText to set.

- CARD64 TextWidth

  Width of text in points.

- CARD64 Ascent

  Maximum Ascent of any character in text.

- CARD64 Descent

  Maximum Descent of any character in text.

- CARD64 Leading

  Leading between line.

- CARD64 base01

  Available baseline.

- CARD64 base02

  Available baseline.

- CARD64 base03

  Available baseline.

- CARD64 base04

  Available baseline.

- CARD64 base05

  Available baseline.

- CARD64 base06

  Available baseline.

- CARD64 base07

  Available baseline.

- CARD64 base08

  Available baseline.

- CARD64 base09

  Available baseline.

- CARD64 base10

  Available baseline.

- CARD64 base11

  Available baseline.

- CARD64 base12
  Available baseline.
- CARD64 base13
  Available baseline.
- CARD64 base14
  Available baseline.
- CARD64 base15
  Available baseline.
- CARD64 base16
  Available baseline.
- CARD64 base17
  Available baseline.
- CARD64 base18
  Available baseline.
- CARD64 base19
  Available baseline.
- CARD64 base20
  Available baseline.
- CARD64 base21
  Available baseline.
- CARD64 base22
  Available baseline.
- CARD64 base23
  Available baseline.
- CARD64 base24
  Available baseline.
- CARD64 base25
  Available baseline.

- CARD64 base26
  Available baseline.
- CARD64 base27
  Available baseline.
- CARD64 base28
  Available baseline.
- CARD64 base29
  Available baseline.
- CARD64 base30
  Available baseline.
- CARD64 base31
  Available baseline.
- CARD64 base32
  Available baseline.

4.141.1 Detailed Description

ST Text Set Metrics request.

The X_STTextSetMetrics request is used to set the metric attribute values of the STText object referred to by the XSTText XID. The protocol Text is the XID value referring to the STText object to set. The protocol TextWidth is a double that represents the width of the text in points. The protocol Ascent is a double that represents the maximum Ascent of any character in the text string. The protocol Descent is a double that represents the maximum Descent of any character in the text string. The protocol Leading is a double that represents the leading between text regions. The protocol baseXX are doubles that represent the available baselines for this text string. The XServer calls STTextSetMetrics and passes the STText value referred to by Text, and the metric information. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.142 xSTTextSetStyleReq Struct Reference

ST Text Set Style request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STTextSetStyle.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Text B32
  XSTText XID referring to the STText to use.
- CARD32 Style B32
  XSTStyle XID referring to the STStyle to use.
- CARD32 FirstChar B32
  Starting character of change.
- CARD32 CharCount B32
  Number of characters associated with change.

4.142.1 Detailed Description

ST Text Set Style request.

The X_STTextSetStyle request is used to associate an STStyle, referred to by the XSTStyle XID, with a section of STText, referred to by the XSTText XID. The protocol Text is the XID value referring to the STText object to modify. The protocol Style is the XID value referring to the STStyle object to use. The protocol FirstChar marks the first character of the STText object that the STStyle object applies to. The protocol CharCount indicates the number of characters of the STText object that the STStyle object applies to. The XServer calls STTextSetStyle and passes the STText object referred to by Text, the STStyle object referred to by Style, the offset and length. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTStyle or XSTText to use is not valid. BadImplementation may be returned if the STStyle or STText to use is

The documentation for this struct was generated from the following file:

- XSTstr.h

4.143 xSTTextSetTextReq Struct Reference

ST Text Set Text request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextSetText.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to STText to use.

- CARD32 CharCount B32

  Number of UTF16 characters to set in text.

4.143.1 Detailed Description

ST Text Set Text request.

The X_STTextSetText request is used to set the text string of an STText object referred to by the XSTText XID. The STText object only contains a pointer to the text. The protocol Text is the XID value referring to the STText object to modify. The protocol CharCount represents the number of characters attached onto the end of the request message. The XServer calls STTextGetText and frees the old text string. The XServer then calls STTextSetText and passes the STText object referred to by Text, the string

of characters, and the number of character in the string. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.144 xSTTextUnderwriteStyleReq Struct Reference

ST Text Underwrite Style request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextUnderwriteStyle.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to the STText to use.

- CARD32 Style B32

  XSTStyle XID referring to the STStyle to use.

- CARD32 FirstChar B32

  Starting character of change.

- CARD32 CharCount B32

  Number of characters associated with change.

4.144.1 Detailed Description

ST Text Underwrite Style request.

The X_STTextUnderwriteStyle request is used to change fields in the covered STStyle objects that are not set in the supplied STStyle object. The protocol Text is the XID value referring to the STText object to modify. The protocol Style is the supplied XSTStyle XID referring to an STStyle object. The protocol FirstChar is the first character whose STStyle is to be changed. The protocol CharCount is the number of characters whose STStyle objects are to be changed.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid or the FirstChar is greater than the current length of the text, or CharCount is greater than the current length of the text. BadImplementation may be returned if the STText to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.145 xSTTextUpdateReq Struct Reference

ST Text Update request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTextUpdate.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Text B32

  XSTText XID referring to STText to use.

- CARD32 Change B32

  Type of change being performed.

- CARD32 TextOffset B32

  Starting character of change.

- CARD32 TextLength B32

  Number of characters associated with change.

4.145.1 Detailed Description

ST Text Update request.

The X_STTextUpdate request is used to change the text string of an STText object referred to by the XSTText XID. The protocol Text is the XID value referring to the STText object to modify. The protocol Change indicates the type of change to perform, Insert, Remove, Exchange. The protocol TextOffset marks the beginning character of the text to change. The protocol TextLength indicates the amount of text to change. The XServer calls STTextUpdate and passes the STText object referred to by Text, the offset and length, and the string that is modified. There is no response to this command. On some occasions, if the reallocation of space for the string does not result in a string at the same memory position, then the X-server call STTextSetText instead.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTText to use is not valid or the TextOffset is greater than the current length of the text, or TextLength is greater than the current length of the text and a Remove or Exchange request is performed. BadImplementation may be returned if the STText to use is NULL. BadAlloc may be returned if space for the additional text cannot be allocated.

4.145.2 Member Data Documentation

4.145.2.1 CARD32 Change xSTTextUpdateReq::B32

Type of change being performed.

IE: delete

The documentation for this struct was generated from the following file:

- XSTstr.h

4.146 xSTTypeEnvCreateFontReply Struct Reference

ST TypeEnv Create Font reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of XSTFont IDs in list.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.146.1 Detailed Description

ST TypeEnv Create Font reply.

The protocol response Count is set the the number of STFont IDs returned by STSF. The XSTFont ID CARD32s are appended to the end of the response message and returned to the X client.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.147 xSTTypeEnvCreateFontReq Struct Reference

ST TypeEnv Create Font request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvCreateFont.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 DataLength B32

  Length of data appended to request in bytes.

4.147.1 Detailed Description

ST TypeEnv Create Font request.

The X_STTypeEnvCreateFont request is used to create a local font for the X client application to use. This font is local to just the STTypeEnv and is not available to any

other application connected to the STSF server. The font(s) to be created is a string of binary data that is sent to XST client library and forwarded through the protocol. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol DataLength represents the amount of binary data in bytes appended onto the protocol request. This binary data contains the font(s) to be created. The XServer calls STTypeEnvCreateFont to create the font(s) from the binary data. This function returns a list of one or more STFont IDs.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for the return font IDs cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.148 xSTTypeEnvDestroyFontReq Struct Reference

ST TypeEnv Destroy Font request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvDestroyFont.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 Font B32

  XSTFont ID to destroy.

4.148.1 Detailed Description

ST TypeEnv Destroy Font request.

The X_STTypeEnvDestroyFont request is used to remove a created font from the STTypeEnv structure. The font will be removed and their resources freed. Note that fonts not created with the STTypeEnvCreateFont command cannot be destroyed. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Font represents the XSTFont ID. The XServer calls STTypeEnvDestroyFont and passes the STFont ID. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.149 xSTTypeEnvDisposeReq Struct Reference

ST TypeEnv Dispose request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvDispose.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to dispose.

4.149.1 Detailed Description

ST TypeEnv Dispose request.

The X_STTypeEnvDispose request is used to remove and free the resources used by the STTypeEnv as well as remove the XID value from the XServer. The protocol Env represents the XSTTypeEnv XID of the STTypeEnv structure to dispose. The XServer calls STTypeEnvDispose. There is no response to the XSTTypeEnvDispose command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to dispose is not valid. BadImplementation may be returned if the STTypeEnv to dispose is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.150 xSTTypeEnvFindAllLayoutEnginesReply Struct Reference

ST TypeEnv Find All Layout Engines reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of Layout Engine IDs returned.
- CARD32 pad1 B32
  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.150.1 Detailed Description

ST TypeEnv Find All Layout Engines reply.

The protocol response count is set to the number of STLayoutEngine IDs returned by STSF. The XSTLayoutEngine ID CARD32s are appended to the end of the response message and returned to the X client.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.151 xSTTypeEnvFindAllLayoutEnginesReq Struct Reference

ST TypeEnv Find All Layout Engines request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvFindAllLayoutEngines.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

4.151.1 Detailed Description

ST TypeEnv Find All Layout Engines request.

The X_STTypeEnvFindAllLayoutEngines request attempts to get all of the Layout engines available to STSF. XSTScaler IDs will be returned. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The XServer calls STTypeEnvFindAllLayoutEngines and passes the Env.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for the return layout engine IDs cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.152 xSTTypeEnvFindAllScalersReply Struct Reference

ST TypeEnv Find All Scalers reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 count B32

  Number of XSTScaler IDs in list.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.152.1 Detailed Description

ST TypeEnv Find All Scalers reply.

The protocol response Count is set to the number of STScaler IDs returned by STSF. The XSTScaler ID CARD32s are appended to the end of the response message and returned to the X client.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.153 xSTTypeEnvFindAllScalersReq Struct Reference

ST TypeEnv Find All Scalers request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvFindAllScalers.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

4.153.1 Detailed Description

ST TypeEnv Find All Scalers request.

The X_STTypeEnvFindAllScalers request is used to find all of the available STScaler IDs and return them to the X client application. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The XServer calls STTypeEnvFindAllScalers which returns a list of STScaler IDs.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for the return scaler IDs cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.154 xSTTypeEnvFindCreateFontFromURLReq Struct Reference

ST TypeEnv Find or Create Font From URL request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvFindCreateFontFromURL.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 URLCount B32

  Number of bytes in URL string.

- CARD32 find B32

  Find the Font, or Create the font.

4.154.1 Detailed Description

ST TypeEnv Find or Create Font From URL request.

The X_STTypeEnvCreateFontFromURL request is used to create a local font for the X client application to use. This font is local to just the STTypeEnv and is not available to any other application connected to the STSF server. The font(s) to be created is a stored The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol DataLength represents the amount of binary data in bytes appended onto the protocol request. This binary data contains the font(s) to be created. The XServer calls STTypeEnvCreateFont to create the font(s) from the binary data. This function returns a list of one or more STFont IDs.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for the return font IDs cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.155 xSTTypeEnvFindFontFamilyReply Struct Reference

ST TypeEnv Find Font Family(s) reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of FontFamily IDs returned.
- CARD32 LanguageID B32
  Language ID associated with single font family.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.

4.155.1 Detailed Description

ST TypeEnv Find Font Family(s) reply.

The protocol response count is set to one (1) if the FontFamily was found. The protocol response LanguageID is set to the language and region that the name matches. The FontFamily ID is appended onto the end of the response.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.156 xSTTypeEnvFindFontFamilyReq Struct Reference

ST TypeEnv Find Font Family(s) request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvFindFontFamily.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 LanguageID B32

  Language ID associated with single font to find.

- CARD32 Max B32

  Maximum number of fonts to return.

- CARD32 NameLength B32

  Number of bytes in font family name to find.

4.156.1 Detailed Description

ST TypeEnv Find Font Family(s) request.

The STTypeEnvFindFontFamily request attempts to find a particular font family, or all font families, available to STSF by sending a name string and a Language value. If the Name is found by STSF, then the FontFamily ID value will be returned.

The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol Max specifies whether this is searching for a single font family (1), or all font families (0).

The remaining protocol items vary depending on whether the call comes as a request for one font matching the parameters, or all fonts. For finding a single font: The protocol LanguageID is used to identify the language and region that the string is to match.

The protocol NameLength indicates the length of the string appended onto the end of the request. The XServer calls STTypeEnvFindFontFamily and passes the STTypeEnv, the Language ID, the string and it's length. The function returns the FontFamily ID if it was found. For finding all fonts: The protocol LanguageID is 0. It is not used. The protocol NameLength is 0. It is not used. The XServer calls STTypeEnvFindAll-FontFamiles and passes the STTypeEnv. The function returns a list of all FontFamily IDs.

BadLength may be returned if the protocol message is not of the proper length. Bad-Value may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.157 xSTTypeEnvFindFontsByNameReply Struct Reference

ST TypeEnv Find Fonts By Name reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of XSTFont IDs in list.
- CARD32 pad1 B32
  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.157.1 Detailed Description

ST TypeEnv Find Fonts By Name reply.

The protocol response count is set to one if a font was found, or zero if no font matches. The XSTFont ID CARD32 is appended to the end of the response message and returned to the X client.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.158 xSTTypeEnvFindFontsByNameReq Struct Reference

ST TypeEnv Find Fonts By Name request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvFindFontsByName.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD16 NameID B16

  ID Tag to search.

- INT16 PlatformID B16

  Platform.

- INT16 EncodingID B16

  Encoding.

- INT16 LanguageID B16

  Language.

- CARD32 Max B32

  Maximum number of Font IDs to return.

- CARD32 NameLength B32

  Number of bytes in string.

4.158.1 Detailed Description

ST TypeEnv Find Fonts By Name request.

The X_STTypeEnvFindFontsByName request is used for all four of the find font commands. The protocol values are changed for each of the commands to indicate to the XServer which STSF command is to be called. For the XSTTypeEnvFindFontByName command, STSF is instructed to find a particular font matching the name appended to the end of the protocol request. For the XSTTypeEnvFindAllFonts command, STSF is instructed to find all available fonts. For the XSTTypeEnvFindFontByPlatformName, STSF is instructed to find a particular font matching the tag value information. For the XSTTypeEnvFindFontsByPlatformName, STSF is instructed to return all fonts matching the platform information. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol NameLength is set to the number of bytes used to specify the name of the font to find. This is the number of bytes appended onto the end of the protocol message in UTF-16 format. For the XSTTypeEnvFindFontByName command, the following protocol values are used: NameID is 0. It is not used. PlatformID is 0. It is not used. EncodingID is 0. It is not used. LanguageID is 0. It is not used. Max is 1. We only want one XSTFont ID in the response. The XServer calls STTypeEnvFindFont and receives an STFont ID. For the XSTTypeEnvFindAllFontByName command, the following protocol values are used:

NameID is 0. It is not used. PlatformID is 0. It is not used. EncodingID is 0. It is not used. LanguageID is 0. It is not used. Max is 0. We want all XSTFont IDs in the response. The XServer calls STTypeEnvFindAllFonts and receives an STFont ID. For the XSTTypeEnvFindFontByPlatformName command, the following protocol values are used: NameID is the field number to compare to the name string. PlatformID is the type of platform the font is designed for. EncodingID is the encoding type of the field. LanguageID is the language type of the field. Max is 1. We only want one XSTFont ID in the response. The XServer calls STTypeEnvFindFontByPlatform and receives an STFont ID. For the XSTTypeEnvFindAllFontsByPlatformName command, the following protocol values are used: NameID is the field number to compare to the name string. PlatformID is the type of platform the font is designed for. EncodingID is the encoding type of the field. LanguageID is the language type of the field. Max is 0. We want all XSTFont IDs in the response. The XServer calls STTypeEnvFindAllFontByPlatform and receives an STFont ID.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for the return font IDs cannot be allocated.

4.158.2 Member Data Documentation

4.158.2.1 INT16 LanguageID xSTTypeEnvFindFontsByNameReq::B16

Language.

IE:

4.158.2.2 INT16 EncodingID xSTTypeEnvFindFontsByNameReq::B16

Encoding.

IE:

4.158.2.3 INT16 PlatformID xSTTypeEnvFindFontsByNameReq::B16

Platform.

IE:

4.158.2.4 CARD16 NameID xSTTypeEnvFindFontsByNameReq::B16

ID Tag to search.

IE: Font name, license, etc.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.159 xSTTypeEnvFindLayoutEngineReply Struct Reference

ST TypeEnv Find Layout Engine reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 LayoutEngine B32
  Layout Engine ID returned.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.159.1 Detailed Description

ST TypeEnv Find Layout Engine reply.

The protocol response LayoutEngine is set to the XSTLayoutEngine ID returned by STSF.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.160 xSTTypeEnvFindLayoutEngineReq Struct Reference

ST TypeEnv Find Layout Engine.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvFindLayoutEngine.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 LETag B32

  Layout Engine Tag uniquely identifying engine.

4.160.1 Detailed Description

ST TypeEnv Find Layout Engine.

The X_STTypeEnvFindAllLayoutEngines request attempts to get a specific Layout engine available to STSF. XSTScaler ID will be returned. The protocol Env represents

the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol LETag specifies the unique identifier for the Layout Engine. The XServer calls STTypeEnvFindLayoutEngine and passes the Env and the LETag.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.161 xSTTypeEnvFindScalerReply Struct Reference

ST TypeEnv Find Scaler reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

  Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 Scaler B32

  XSTScaler ID returned.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

Not used.

- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.161.1 Detailed Description

ST TypeEnv Find Scaler reply.

The protocol response Scaler is set to the XSTScaler ID value.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.162 xSTTypeEnvFindScalerReq Struct Reference

ST TypeEnv Find Scaler request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STTypeEnvFindScaler.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Env B32
  XSTTypeEnv XID to use.
- CARD32 ScalerTag B32
  Unique Scaler ID value.

4.162.1 Detailed Description

ST TypeEnv Find Scaler request.

The X_STTypeEnvFindScaler request attempts to find a particular scaler available to STSF by sending a scaler tag value. If the scaler is found by STSF, then the STScaler ID value will be returned. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol ScalerTag is used to help identify the specific scaler the X client application is interested in finding. The XServer calls STTypeEnvFindScaler and passes the ScalerTag. The function returns the STScaler ID if it was found.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.163 xSTTypeEnvGetFontFallbackPolicyReply Struct Reference

ST TypeEnv Get Font Fallback Policy reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 Policy B32
  Policy value.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.163.1 Detailed Description

ST TypeEnv Get Font Fallback Policy reply.

The protocol response Policy is set and returned to the Xclient application. The policy value returned is a member of the XSTFontFallbackPolicy group.

4.163.2 Member Data Documentation

4.163.2.1 CARD32 Policy xSTTypeEnvGetFontFallbackPolicyReply::B32

Policy value.

See XSTFontFallbackPolicy

The documentation for this struct was generated from the following file:

- XSTstr.h

4.164 xSTTypeEnvGetFontFallbackPolicyReq Struct Reference

ST TypeEnv Get Font Fallback Policy request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvGetFontFallbackPolicy.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

4.164.1 Detailed Description

ST TypeEnv Get Font Fallback Policy request.

The X_STTypeEnvGetFontFallbackPolicy request is used to request the fallback policy to be used for all characters not present in their STStyle specified font. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The XServer calls STTypeEnvGetFontFallbackPolicy to get the appropriate response to return to the X client.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.165 xSTTypeEnvGetFontFallbacksReply Struct Reference

ST TypeEnv Get Font Fallbacks reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type

Reply Type: always X_Reply.

- CARD8 pad0

  Not used.

- CARD16 sequenceNumber B16

  Sequence Number to associate to request.

- CARD32 length B32

  Length of Reply beyond normal in Bytes.

- CARD32 count B32

  Number of XSTFont IDs in list.

- CARD32 pad1 B32

  Not used.

- CARD32 pad2 B32

  Not used.

- CARD32 pad3 B32

  Not used.

- CARD32 pad4 B32

  Not used.

- CARD32 pad5 B32

  Not used.

4.165.1 Detailed Description

ST TypeEnv Get Font Fallbacks reply.

The response is returned along with a number, equal to count, of additional CARD32s. Each additional CARD32 contains an XSTFont ID.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.166 xSTTypeEnvGetFontFallbacksReq Struct Reference

ST TypeEnv Get Font Fallbacks request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvGetFontFallbacks.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

4.166.1 Detailed Description

ST TypeEnv Get Font Fallbacks request.

The X_STTypeEnvGetFontFallbacks request is used to get the font fallback list to be used for all characters not present in their STStyle specified font within each STText object associated with the STTypeEnv object the operation is performed upon. This may be overridden if the STText object has its own font fallback list. (See XSTTextGetFontFallbacks for information on the local font fallbacks.) The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The XServer calls STTypeEnvGetFontFallbacks and receives a list of STFont IDs. Count represents the number of STFont IDs received by the XServer.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if space for the fallbacks cannot be obtained.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.167 xSTTypeEnvGetFontFoldersReply Struct Reference

ST TypeEnv Get Font Folders reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 count B32
  Number of Font Folders returned.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.
- CARD32 pad5 B32
  Not used.

4.167.1 Detailed Description

ST TypeEnv Get Font Folders reply.

The protocol response count is the number of returned directory strings. The directory strings are appended onto the end of the response message.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.168 xSTTypeEnvGetFontFoldersReq Struct Reference

ST TypeEnv Get Font Folders request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STTypeEnvGetFontFolders.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Env B32
  XSTTypeEnv XID to use.

4.168.1 Detailed Description

ST TypeEnv Get Font Folders request.

The X_STTypeEnvGetFontFolders request returns a list of user specified directories that are searched for additional fonts. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The XServer calls STTypeEnvGetFontFolders and passes the STTypeEnv. The function returns an array of directory strings along with the number of directories.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation

may be returned if the STTypeEnv to use is NULL. BadAlloc may be returned if the response cannot be allocated.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.169 xSTTypeEnvGetLocationsReply Struct Reference

ST TypeEnv Get Locations reply.

`#include <XSTstr.h>`

Public Attributes

- BYTE type
  Reply Type: always X_Reply.
- CARD8 pad0
  Not used.
- CARD16 sequenceNumber B16
  Sequence Number to associate to request.
- CARD32 length B32
  Length of Reply beyond normal in Bytes.
- CARD32 mask B32
  Mask of font locations used by STSF server.
- CARD32 pad1 B32
  Not used.
- CARD32 pad2 B32
  Not used.
- CARD32 pad3 B32
  Not used.
- CARD32 pad4 B32
  Not used.

- CARD32 pad5 B32

  Not used.

4.169.1 Detailed Description

ST TypeEnv Get Locations reply.

The protocol response mask is set to the returned mask.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.170 xSTTypeEnvGetLocationsReq Struct Reference

ST TypeEnv Get Locations request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvGetLocations.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

4.170.1 Detailed Description

ST TypeEnv Get Locations request.

The X_STTypeEnvGetLocations request returns a mask indicating which directory locations are being used when searching for fonts. The protocol Env represents the

XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The XServer calls STTypeEnvGetLocations and passes the STTypeEnv. The function returns a mask indicating which directory locations are being used.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.171 xSTTypeEnvNewCopyReq Struct Reference

ST TypeEnv New/New Copy request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvNewCopy.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 OldEnv B32

  Old XSTTypeEnv XID to copy.

- CARD32 NewEnv B32

  New XSTTypeEnv XID to create.

4.171.1 Detailed Description

ST TypeEnv New/New Copy request.

The X_STTypeEnvNewCopy request is used to create a new STTypeEnv and associated XSTTypeEnv XID value. The protocol request and reply used are the same to perform

both an XSTTypeEnvNew and XSTTypeEnvNewCopy. The difference between the two functions is that the OldEnv field is set to 0 when creating a new XSTTypeEnv.

BadLength may be returned if the protocol message is not of the proper length. BadAlloc may be returned if unable to allocate the resource. BadValue may be returned if the XSTTypeEnv to copy is not valid. BadImplementation may be returned if the STTypeEnv to copy is NULL.

4.171.2 Member Data Documentation

4.171.2.1 CARD32 OldEnv xSTTypeEnvNewCopyReq::B32

Old XSTTypeEnv XID to copy.

0 if no copy

The documentation for this struct was generated from the following file:

- XSTstr.h

4.172 xSTTypeEnvSetFontFallbackPolicyReq Struct Reference

ST TypeEnv Set Font Fallback Policy request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType
  Request Type: always STRCode.
- CARD8 stReqType
  ST Request: X_STTypeEnvSetFontFallbackPolicy.
- CARD16 length B16
  Request Length in Bytes.
- CARD32 Env B32
  XSTTypeEnv XID to set.

- CARD32 Policy B32

  Font Fallback policy to use.

4.172.1 Detailed Description

ST TypeEnv Set Font Fallback Policy request.

The X_STTypeEnvSetFontFallbackPolicy request is used to set the fallback policy to be used for all characters not present in their STStyle specified font within each STText object associated with the STTypeEnv object the operation is performed upon. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. Policy represents the font fallback policy to use. It is a member of the XSTFontFallbackPolicy group. The XServer calls STTypeEnvSetFontFallbackPolicy to set the appropriate policy for the STTypeEnv. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to set is not valid. BadImplementation may be returned if the STTypeEnv to set is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.173 xSTTypeEnvSetFontFallbacksReq Struct Reference

ST TypeEnv Set Font Fallbacks request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvSetFontFallbacks.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

XSTTypeEnv XID to set.

- CARD32 count B32

  Number of Fallback Fonts in list.

4.173.1 Detailed Description

ST TypeEnv Set Font Fallbacks request.

The X_STTypeEnvSetFontFallbacks request is used to set the font fallback list to be used for all characters not present in their STStyle specified font within each STText object associated with the STTypeEnv object the operation is performed upon. This may be overridden if the STText object has its own font fallback list. (See XSTTextSetFontFallbacks for information on the local font fallbacks.) The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol count represents the number of XSTFont IDs appended onto the end of the protocol request. Each XSTFont ID is a CARD32. The XServer calls STTypeEnvSetFontFallbacks and sends the list of STFont IDs. There is no response given to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.174 xSTTypeEnvSetFontFoldersReq Struct Reference

ST TypeEnv Set Font Folders request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvSetFontFolders.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 FontFoldersCount B32

  Number of Font Folder strings appended.

4.174.1 Detailed Description

ST TypeEnv Set Font Folders request.

The X_STTypeEnvSetFontFolders request sets the list of directories that are to be searched for extra fonts. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol FontFoldersCount indicates the number of directory strings appended onto the end of the request. The XServer calls STTypeEnvSetFontFolders and passes the STTypeEnv along with the array of directory strings and the number of strings. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h

4.175 xSTTypeEnvSetLocationsReq Struct Reference

ST TypeEnv Set Locations request.

`#include <XSTstr.h>`

Public Attributes

- CARD8 reqType

  Request Type: always STRCode.

- CARD8 stReqType

  ST Request: X_STTypeEnvSetLocations.

- CARD16 length B16

  Request Length in Bytes.

- CARD32 Env B32

  XSTTypeEnv XID to use.

- CARD32 Locations B32

  Location mask to tell STSF where to find fonts.

4.175.1 Detailed Description

ST TypeEnv Set Locations request.

The STTypeEnvSetLocations request allows the user to specify which directory locations are to be used when searching for fonts. The protocol Env represents the XSTTypeEnv XID used by the XServer to get the STTypeEnv structure. The protocol mask indicates which directory locations to use for getting fonts. The XServer calls STTypeEnvSetLocations and passes the STTypeEnv along with the mask. There is no response to this command.

BadLength may be returned if the protocol message is not of the proper length. BadValue may be returned if the XSTTypeEnv to use is not valid. BadImplementation may be returned if the STTypeEnv to use is NULL.

The documentation for this struct was generated from the following file:

- XSTstr.h