Low Bandwidth X Extension

X Consortium Standard

D. Converse
J. Fulton
D. Lemke
R. Mor
K. Packard
R. Tice
D. Tonogai

Protocol Version 1.0

! "# " !
! ! !!
! ! !!
# #
$!! %

& ' !! ! !!
! # (

&)* #+,&-./* 0# /+102*2 ".# 0#" -0&)+3& -.//.4&5 +, .45
6042 * /*## +/ 07 80*2 04 832049 :3& 4+& 8070&*2 &+ &)*
-.//.4&0*# +, 7*/).4&.:080&5 ,0&4*## ,+/ . ./&0 38./
3/ +#* .42 4+404,/049*7*4&(04 4+ *1*4& #).88 &)*
+4#+/&037 :* 80.:8* ,+/ .45 8.07 2.7.9*# +/ +&)*/ 80.:080
&5 -)*&)*/ 04 .4 . &0+4 +, +4&/. & &+/& +/ +&)*/-0#*
./0#049 ,/+7 +3& +, +/ 04 +44* &0+4 -0&) &)* #+,&-./* +/
&)* 3#* +/ +&)*/ 2*.8049# 04 &)* #+,&-./*(

*; !!
' ! !
< (

Table of Contents

0 (((B
2 (((B
2 ,! (((C
& (((C
# (((D
9 / (((E
7 ' (((E
*' #= (((E

7 ! ((E
7 ! ! ; ! ((E
(((
2 ! (((
(((
. ! ((F
8 0 ((F
. ! 8 0 ((F
; 8 0 ((J
! ((J
' & ((J
* ((G
/ = ((G
*' ((CG
/ ((CC
.! 4 ((CC
* ((CC
* ((CD
/ = ((CD
*' ((EG
/ *' ((EB
/ ((EC

Introduction

8 : 8: > ! -
! ' ! !
! ! ! ! > (0 '
= ' !
' (0 ; ! ! ;
! ' ! ? !
; ' (

& ; ! @ , ! 2 ' 8 > 6
> 2 ! & !! 4 > 2 ' 4 2 (6
6 $ ' ; ./ ! (/ ! 7
! ! (2 '
!
! (/ & !'
(: # ! ! (

& ; "8: "(

Description

& 8: ; !
' (& ; = ' !
!! ! (. !! ;
' ' ! ! ! ! (& !
; > (. ; ! ! ! !
' ' ' ! ! ' (
& !! ; ! ! ; ! ! ! ! !
' (

7 8: ! ; (&
! ' !
= ' A ! (

8: ! ' ! (3
! ! '
! !!(8: ! ' ; 8: (

Data Flow

& 8: ! %

(! =

B(/ 8: !

C(/ =

D(!

E(7 ! ! ; ! =

(2 ! !

F(#

&

(#

B(2 !

C(2 ! ! ; ! =

D(!

E(/

(/ =

& ' ' ! ' (

Tags

& ! ; =
! ! (# > =
! ! ! (/
! (& ; ' = =
! (&
>
(

& ! (& ;
! ;(& ' '
; A 0 ' ! & ' ! (
&
(& ' (

& ' > > > ;(& ; ' !
! (& ;
! (& ; ' !
! ' (

0 ' ' '
= ; ' ;
(

Tag Substitution in Requests

7 = ' ! !! ! > !(
, = ! !
! G(0 ! ! !
" !" ! (

Property Tags

> ! (.
! (0 ! ; !
' > ' ' (
= > =
(& ! =
(0 ! !! '
' ' (0
' = (& ;
! $ ((
! (# ; ' !
> ' ' = (

Short-circuiting

! (& ! ! ?
/9: !! (. ! ?/9:
! ' (! !
(# ! = =
! (

. ' ! 0 7 (+ ; >
= =
' (

! ?/9: ;
! = = (

! ! !! ! ! !!
; > /9: ' ! ! ! (&
; H = !'
!! ' ! !!H (″
(

, !! ! ; !! !! ' =
= = (&
$ % (

Graphics Re-encoding

Motion events

Event Squishing

Master Client

Multiplexing of Clients

& ' ! !; ! ; +
. + (& !
! > ! > ! (& ;
! > ' ! ; ! !;
! ; ! ' (

/ ! ! !; ' ;
! !
(

Swapping

! ; ' (# ;
! !!
! ! !! ' ; ;
; (& ' ! !
! (

Delta cache

8: > ' ' !
' ! (, ; ! / ++ '
' / ++ ' $ (: $
" ! " ' !! (
2 ! = ; !! !
' ' (

& ' ; > ; = (&
' ! (.!! '
! J
! ! (& !
; ' < = (

& 8: = ' = !
0 + - , - -#
1 + " " *2*
& % + & % + *1* & % +
"+ = (& '
- *1* & + + (&
! = (7
= ; G ; !
(& =
! ; (

0 ! !
! ! ! !
! (

Stream Compression

: !
! (& !
8 ;# ; K ;(LBGJFG M& ; ' =

(. ; ! '
! N80: ! (

& N80: !
8: (& > ;
< ! > (* ! > <! (G 9 !! O. !
> (
& !! %

```
out[0] = (length & 0xfff) >> 8 | ((compflag) ? 0x80 : 0);
out[1] = length & 0xff;
```

Authentication Protocols

& ' 8: ! !
! ;(& !
= = !
! , - = !
, - (+ ; ! ! P3*/5#* 3/0&5 (

C Library Interfaces

& 8 8: ; ! (& !
! " 8 ;("(

Application Library Interfaces

0 ; ' ! !! >
' 8: (! ' 8: ;
' (3 !
; # 8 0 K ;(LCCC M

XLbxQueryVersion

& ' 8: ' !! *3 0 +* (

Bool **XLbxQueryVersion**(*display*, *major_version_return*, *minor_version_return*);

! # ' (

$ Q' Q / ; $ ' (

Q' Q / ; ' (

& *3 0 +* 8: ; (0
; *3 0 +* ' + R
(0 ; *3 0 +* $
' ; ' (

Proxy Library Interfaces

& !! ;(

XLbxQueryExtension

& !! ; !
3 1 + 8: *3 1 +* (

```
Bool XLbxQueryExtension(display, major_opcode_return,
first_event_return, first_error_return);
```

! # '(

$ Q Q / $ (

Q ' Q / ' (

Q Q / (

& *3 1 +* 8: ; (0
; *3 1 +* ' + R
(0 ; *3 1 +*
$; $ Q Q '
Q ' Q Q Q R
'! (

XLbxGetEventBase

& ' ! *3 1 +*
8: *3 1 2 +* (

```
int XLbxGetEventBase(display);
```

! # '(

& *3 1 2 +* ' ;
R (

Protocol

Syntactic Conventions and Common Types

! - # !
' ! (

& !! ! ; %

%
G ST ST . E OO U V . E T G;E W ?
. E
. GE ST ST . C G OO U V . E T G; F W
? . E
. GG ST ST . E OO U V . E T G; W
? . E
. C G S ST . GE OO U V . E T G; J W
? . E
. C G S ST . C G KGM T XX JR K M T

K; % 8: 04&
% 8: ./2
! ! B% 8: .498*M

- ! ; ! '
! (

%
G;GGGG ST S G;GG,G ./2J
G;GG,G ST S G; G,G KGM T G;,G Y G;,G XX
J
K M T G;,G

K ; ! ;% ./2D M

! % 2/.-.:8* ?Z !
-
TT G Z?
% 9 M ?Z ! TT G Z?

% 8: 9 .422/.-*4&
% 8: 9 .422/.-3 2.&*

% 04&J B :5&*
%
G;,F G ST S G;,, G KGM T G;JG Y W G;FG XX
J O G;G,
K M T W G;FG
G;,, G ST S G;GGJG ./2J
G;GGJG ST S G;GJJG KGM T G;JG Y G;JG XX
J O G;G,
K M T G;JG

% ./2J B :5&* ?Z
!! ' Z?
%
G;,*GG ST S G;GGGG KGM T G;,G Y W G; GGG
XX J O G;G,
K M T W G; GGG
G;GGGG ST S G;GG,G ./2J
G;GG,G ST S G;G*,G KGM T G;,G Y G;,G XXJ
K M T G;,G

% K; % 8: 04& M
- ! ! ;
! ' ! (

%
K; % 8: 04&
% 8: ./2 M

- ! ! ! ; ! '
! ! (

7.#6% ./2J

Errors

. ! = =
! ; (

& (& ! ! 8:
= ' ! ; A ' !
; = (

Requests

& = ; ! ! % *0* +

& = ; ! ; %
1 + (

& !! = ; ! ;
; % , - -
+ # $ % -#
(

.!! = ; 8: '
= ! (& ! = (

Requests Initiated by the Proxy or by the Client

!"#$ %& '( )

TXR

!"#$ %& '( )
$ 1 % ./2
1 % ./2

& = $ ' 8: !(

& = # 8 ;P 1 K! ;(LCD M(

Requests Initiated or Substituted by the Proxy

!"#$ % ")'( )
+ % ./2CB
" % #&/049J
TX
= % ./2J
% :++8
$ % ./2J
' % ./2J
% ./2J
! >% 80#& 7.#6 ?Z ! Z?
' >%80#& 7.#6 ?Z ! Z?
* %

& = ! *1* + = ! !
! ! ! (- ! *1* + =
; !! *1* + = (

& = ; (0 $
; (+ < (.
= ; (0 ;
' !' ! ' ' (+ <
(& ' ; (0 ;
' !' ! (+ <
(& ! ; (

0 = ; (0
= < ' ! ! ! >
' > !! ! !(& ! >
; = (& !
= ; ; =
(* ! ! ! ;
! ! ! (3
< (& = = !
! < (0 ' >
; = (.
= ! ! ;
' (0 ! > ! ' > !! ! (

& = # 8 ;P *; K! ;(LCF F M(

Control Requests Initiated by the Proxy

!"* + "%

+ % 80#& + &0+4

TX

% 80#&)+0 *

* %

%

+ &0+4 K % ./2J

! % + &8*4

% # # # ; + Ƙ ;(LCEDDD MM

)+0 * K % ./2J

! % + &8*4

% # # # ; + Ƙ ;(LCEDDD MM

+!, : -:* + "% . ( )'

. / 0	**. ( )**	**/1 (/**	**0 2+$,**
! ;	2*8&.+ &	2*8&.)+0 *	T ;! T D
! '	2*8&.+ &	2*8&.)+0 *	T ;! T D
	80#& 4.7*2+ &	042* *2)+0 *	4
	80#& #&/049J	80#& 042* * 2+ &	4
;	80#& 0 7. 7*&)+2	80#& 0 7.)+0 *	4
=	:++8	:++8	&
	:++8	:++8	&
!	80#& #&/049J	042* *2)+0 *	4 ! 9
;	4.7*2+ &	042* *2)+0 *	*; 2 !

2*8&.+ & K 4 ;4 4% ./2J

7 ;7 8 ;7 ;7 8 7 ;7 8 % ./2JM

2*8&.)+0 * K ;! % ./2JM

042* *2)+0 * K ;% ./2J

% 80#& :5&*M

042* *2+ & K ; % ./2JM

4.7*2+ & K % #&/049J

!% 80#& :5&*M

+ &8*4 C ./2J

%

G S ST G;,, % T

G ST ST G;,,,, % KGM T GR K M T XX JR KBM T O G;,,R

0 7. 7*&)+2 K % #&/049J

>% :0&7.#6

% 80#& ./2JM

0 7.)+0 * K ; % ./2J

>% :0&7.#6

% 80#& ./2JM

& = 8: ! ; '
! 8: !(

& ; ' ! = (& '
' ; (& !
; (0
!! ! (

& ' ! ; !
! (# # 2 ! K ;(LBBE E M ! (
& ! + (&
; ; '!
" ,4 " ,4 $, !
" # #+ 4 " # #+ 4 $# #+ (& ' A !
! + " '! (& '!
;(& ' ! + '! G
! ! (& ! D (

& ! ! + " " (
# # # K ;(L E M
* ! !! ' # .!
4 K ;(L CEFG M & ;
! ' ! ! (, ! ;
" ! ! !
! (& ! G ;
! ! ! !!
! (

: " " (& ;
! ' ! ! ! ' ! ! ! (
, ! ! G ; ! !
! ! ' '! = (&
! !! ' # .! 4 K ;(L CEFG M

; " " (& ;
! ' ! ! ! (, ! ! !
> ! (&
' ! ! ! (, ; ! !
! G ; ! ; !
= ! > '! !

' ! (.! !! ' # .! 4
K ;(L CEFG M(

#= = (0 ; =
' ! (& ! ' =
!! = ! + +% + = (

& # & K ;(L G J M
(0 ; ' ! (& !
' !! ! +
= (

& ! ! ! !!
; ; = !! ; !
! (& ! ; !
' ! ! ! ! (& ! G
; ! ! ! !!
! (0 ! ! !!
″ & + ″ = !!
(

& ; ! ; 8: (& ;
; ! ! (& ;
; !! ; (& ;
!! ' # .! 4 K ;(L CEFG M(
& ; ! ! ;
! ! ; (& ! ;
< ; ;
(& ; !! ; (& '
; < (

. ! !
8: ; ' = (

& = # 8 ;# ; K ;(LBFDEB M(

!"* . "%

* %

& = ; '
! ;(

& = # 8 ;# ; K ;(LBCDF M(

. = ! 8: ;(

!" 3 ,( )

% ./2J
% ./2CB
″ 5 + % ./2
″ + 6 ./2
7 ″ % #&/049J
7 % #&/049J

TX

!" 3 ,( )

! $

+/

% :++8

% [4 2 ! 4 ! ! 2 ! . 9 2 ! \

! $ ' % ./2

! ' % ./2

% ./2CB

! % ./2

% +404,+ +420, +420,/++&

%

+404,+% " ! " !

+420,% K % ./2CB

> % 80#& #*& *1*4&M

+420,/++&% K % ./2CB

% -042+-

' !% 10#3.802

! ! % +8+/7.

; ! ! > ; !% ./2CB

> % 80#& #*& *1*4&M

* % 8 ; ! .!!

& = ; ' !

' ! ' (

7 , - = ! !

! !(

, , - = ! = 8: ! (, , -

! ! = 8:

8: ! ! (

& ; ' ! =

! , - = (& ! 8: !

! - = - *1* (

! G ' ; ! (. !!

! , - = ! G !

! (

0 ' $ ' ! '

! ! ;(& ' 8:

!(

0 ! , +
+404,+ ! !
!(0 < ;
' ! (& ;
' ! ; ! ' ' (

- ! , +
' ! ;(&
! < ' ! ! (.
<
; ' (

0 ! , ″ +
+420, (& ' ! +420,
!! ! (

0 ! +
+420,/++& (& + $ ″
! < !
(& - ! ! !
$ ″ ! ! < (- $ ″
! ! < - ! (&
″ +/+ ! > ! (&
″ +/+ > '
*; (& ' !
″ +/+ (&
! (

& = # & =
C(K ;(L E M

!" , ' ,( )
% ./2CB
* %

& = ' !
; A ! (0 ! A ' !
' , - = ' !! (

& = # & =
B(K ;(LB B M

!"*3(/1
% ./2CB
* %

& = ' = =
' ! (

0 ! > = ; !
= ' ! , - =
(

& = # 8 ;# K ;(LC F G M

!"*%)/

TX

& = ' ! !! =
= ' (

& ! # 8 ;# K' ;(LB J M(

!" 0(2%* 4$)/

5 + % ./2CB

* % 4

& = ' = ' ! !
(& ; # $ % = '
! ! = ! =
' (

& ! # & = C(
K' ;(L G DG M(

!" „ 3 ( )

" % ./2CB

* % 4

& = ! !' ! '
# 7 ' K' ;(L EEGC M(& !
!! ' (

& = # & =
D(K' ;(L J F M(

!")5+,(0+ +6

% ./2CB

& 8: ; '
! ' !
' ! ' (

& = # 8 ;0 ' ! & K' ;(LCFEDE M(

!" +6 + +

% ./2CB

% ./2CB

% 80#& :5&*

& = ' ! (0
% ' *1*
! ; ! ' '
! ' ' (&
! ' !! ! = (

& = # 8 ;& 2 K' ;(LCF FD M(

!" +! 7+.

″ % !

TX

+″ % :++8

6 :++8 ?Z ! Z?

+ *6* :++8 ?Z ! Z?

+ % :++8 ?Z ! Z?

+ *6* ./2D ?Z ! Z?

+ % 80#&).4 ?Z ! Z?

%

).4% 80#& 8: 0 *8

8: 0 *8% 0 *8 /01.&* 0 *8 /01.&*/.49*

0 *8.88+ 0 *8.88+ /.49*

0 *8% ./2J ./2

0 *8 /01.&*% K ; !% 0 *8 M

0 *8 /01.&*/.49*% K ; ! ! ; !% 0 *8M

0 *8.88+ % K ; !% 0 *8

! % +8+/#0498* +8+/&/0 8*M

0 *8.88+ /.49*% K ; ! ! ; !% 0 *8

! % 80#& +8+/#0498* 80#& +8+/&/0 8*M

+8+/#0498*% K ' ! % ./2J ./2 M

+8+/&/0 8*% K % +8+/#0498*M

* % ″

& = > ' ! !! ! !!

! (& ' ! ! = (0

' ! ; ! +″

! ! ! ! (+

! ! ! !

(

+ ; ' ! =

!! ! !! ! (&

; !! ! ! ″ =

! ! (& ' ! !! ' !

;(

& ″ = ! ! ' !

! ! ; =

(

& ' ; !! ! !! !

! ! ;(0 ' ;

!! ! !! ' 8 ; & + ″ '
; ! & + ″ = (

& ' ; ! !! !
;(& ' !! ′ + ' ; !
! !! G(

0 ! ' !
; A ! ! '
; ″ = !
= ! (

0 ! '! ! ! !
; ″ (

& ! ! ' ' !
' ! ! ! !!
! (

& & 0 *8 !! !
./2 = ./2J(& + ! &
! ' ! ; A ! !
' (

0 + , ! ! ! ! ! '!
+8+/&/0 8* ! ! (. !
! ' ! (

0 + & ! ! ! !
! (! '!
! +8+/#0498* 0 *8 '!
/9: ! !
> > ! > ' !(

& + '! ! '! ! ' !
! ! (0 '! F ! +8+/#0498* '!
' ! ./2J ! (+ '!
./2 ! (

& ! ! !! !!
%

. 8: 0 *8 /01.&* ; ! ! ' ! !
!! (

. 8: 0 *8 /01.&*/.49* ; !
' ! ! !! (& $ + + ! +
(

. 8: 0 *8.88+ ; ! ! !!
! ; !(& ! ! ; !(

. 8: 0 *8.88+ /.49* ; !
!! !(& $ + + ! +
(& ! !! ! +8+/#0498* +8+/&/0 8*
' ! + (

. 4* &).44*8 ; ! ! !!
(

. 80#&*42 ! (

.!! ; ! ! !! (

& = # 8 ;9 K ;(L F J M

!" , +' 7+.
" % !

& = ! ! (0 " !
! 2 " (

& ; ! & + "
= ; "
= ! ! (

& = # 8 ;/ ! K ;(L DF M

!")) 7'
% ./2
" +6 $ & , (
TX
"+ % 80#& .&+7
* %

& = !! ; ! (&
' !! ; (

& = # 8 ;0 . K ;(LCD DG M

Substitution Requests

!" „ / ,
" % !
% ./2CB
% ./2

& = ; ' ! !!
! !! ! (& ; =
, " = (

& ! ! !! !! (&
! ! ' ! !
= (& ! ' ! ;
! ! (

, ! ' ! " =
8: (

0 " ! ! 2 ++
(0 ! ; ! !
; ! ! 2 (

& = # 8 ;.!! ! K ;(LBJDB M

!")/ 7) (" ,
" % +8+/7.
% ./2CB
* % 4

& = ! = ! ; ! ! !!
! (0 ' ! !
= ! !! ; ! (& ! =
= (

& = # & =
D(K ;(LCJGEC M

!" , +
% ./2J
% ./2J
$$+ % 80#& 20,,0&*7

& = ! ! ' !
= (& = (
& = ; !
; ' (

& = # & =
J(K ;(LC JCJ M

!" 0(2(+..()6
TX
! + " % ./2J
% ./2CB
! + % 80#& 6*5 +2* ?Z ! Z?

& = ! # $ # =
!(# # & # / =
K ;(LB ECD M ! ! ! (

& = # 8 ;9 7 7 K ;(LDGGEF M

!" 8 %! + 0 +..()6
$ + / % 6*5 +2*
% ./2J
TX
! + ! % ./2J

!" 8 %! + 0 +..()6

% ./2CB

*! + "+ % 80#& 6*5#57 ?Z ! Z?*

** % 0*

& = ! / # ! =

!(# # & # /=

K ;(LB ECD M ! ! ! (

& = # 8 ;9 6 7 K ;(LB FGB M

!" ())0 7

- - % -042+-

TX

' !% 10#3.802

! % [0 + 0 + ! \

' % :0&9/.10&5

' % -049/.10&5

> % [4 3 ! - 7 .! \

> ! % ./2CB

> ; !% ./2CB

' % :++8

! % +8+/7. 4

!! % :++8

% [3 3 ' ! 1 ! \

!! ' > ' >% #*& *1*4&

>% #*& 2*10 **1*4&

' % :++8

% -042+-

% ./2J

; % 04&

% ./2

** % 8 -*

8 - + " = !

!(*8* " !! ; =

(

& = # 8 ;9 - . . 9 K ;(LD DDG M

!"#$ %)

$ % ,+4&.:8*

TX

% :++8

% ./2CB

!"#$ %)

% ,+4&04,+ ?Z ! Z?

% 80#&).04,+ 80#& 8:).04,+ ?Z ! Z?

%

8:).04,+% K! % 04&

% 04&F

% 04&

% 04&

% 04&FM

* % ' *4*

& = ! ' = !

(

& # / = K' ;(LB ECD M

! ! ! (

& " ++ ! & $ + ! 8:

).04,+(

& !! 8:).04,+ '

!

F ! !

! ;Q $ 9 $! (

& = # 8 ;P , K' ;(LBDE F M(

!" 1+)6 . %

- - % -042+-

% .&+7

% .&+7

$ " % [G J CB\

" % [/ ! . \

: + % ./2CB

TX

% ./2CB

& = ' !

= ;(& < =

! (& ' !

;(& ;

' ' ! ' (&

= ; ; 8 ; = (0

' ! > ' !

' ' ! ! > ' ! ! (

0 " ! !

(

0 ! < '
; (& ; !
' !! ' > (

& = # 8 ; K ;(L JG C M

!" . %
- - % -042+-
% .&+7
% .&+7 . &
$$+ % ./2CB
% ./2CB
% ./2J
TX
% .&+7 4
% [G J CB\
% ./2CB
0 % ./2CB
% ./2CB
' ! % 80#& 04&J 80#& 04& 80#& 04&CB

& = ;
= (0 !! ! > !
' ! ! > ! ! ! ! (

& =
= (& ! !! !
! ! ! ″+ (

0 ! < '
! ;
(7 ! ' !
″+ <
! ! ! ! ' ! (& ;
(

0 = '
> ; !
! ! (0
! ! ! < < ! ! ' ! (

0 ' !
! ! ! < <
! ! ! ! ' ! (

& = # 8 ;9 K ;(L CJ C M

!" ,% ()
- 6 8: 9 .422/.-.:8*
+ % 80#& 8: +04&

!" ,% ()

* % ' = (

& = ! = (4 !! =
= (

& ; !! ' ! ' '
' !(

& = # 8 ; ! K ;(LB F M

!" ,% ()

- *6* 8: 9 .422/.-.:8*

\+ % 80#& 8: +04&

* % ' = (

& = ! = (4 !! =
= (

& ; !! ' ! ' '
' !(

& = # & =
B (K ;(LC GJ M

!" ,%* 67)

- *6* 8: 9 .422/.-.:8*

\+ " + % 80#& 8: #*97*4&

%

8: #*9*7*4&R K; ;B B% 8: 04& M

* % ' = (

& = ! " = (4 !! " =
" = (

, = K; M ! ' K;
M ' (, !! K;B BM ! ' A
K; M(

& = # 8 ; ! # K ;(LBFEBJ M

!" ,% / +)6,

- *6* 8: 9 .422/.-.:8*

\+ % 80#& 8: /* &.498*

* % ' = (

& = ! & = (4 !! & =
& = (

& = # & =
BB(K ;(LCC BJ M

!" ,% /
- 6 8: 9 .422/.-.:8*
+ % 80#& 8: ./
* % ' = (

& = ! = (4 !! =
= (

& = # 8 ; ! . K ;(LBEJEE M

!" ,% (,, / +)6,
- 6 8: 9 .422/.-.:8*
+ % 80#& 8: /* &.498*
* % ' = (

& = ! ′ & = (4 !! ′ &
= ′ & = (

& = # 8 ; ! , !!/ ! K ;(LB C M

!" ,% (,, /
- 6 8: 9 .422/.-.:8*
+ % 80#& 8: ./
* % ' = (

& = ! ′ = (4 !! ′ =
′ = (

& = # & =
BB(K ;(L GJ M

!" (,, ,%
- 6 8: 9 .422/.-.:8*
+ % :5&*
+ % 80#& 8: +04&
* % ' = (

& = ! ′ = (4 !! ′ =
′ = (

& ; !! ' ! ' '
' !(

& = # 8 ;, !! ! K ;(LBD J M

!" .% +
+ % ./2J ?Z ! Z?

!" .% +

- 6 8: 9 .422/.-.:8*

+ - % ./2CB

+ % 8: 04&

+ % 8: 04&

- % 8: ./2

% 8: ./2

+ % 8: 04&

+ % 8: 04&

* % & ' = (

& = ! = = (

& = # 8 ; . K ;(L GBC M

!" .% ,+)

% ./2CB

+ % ./2J ?Z ! Z?

- 6 8: 9 .422/.-.:8*

+ - % ./2CB

+ % 8: 04&

+ % 8: 04&

- % 8: ./2

% 8: ./2

+ % 8: 04&

+ % 8: 04&

* % & ' = (

& = ! = = (

& = # 8 ; ! K ;(L JJDF M

!" ,% "

- 6 8: 9 .422/.-.:8*

% 8: 04&

% 8: 04&

"+ % 80#& &* &0&*7J

* % ' = (

& = ! (= = (

& = # & =
BC(K ;(LC DG M

!" ,% "

- 6 8: 9 .422/.-.:8*

!" ,% "

6 8: 04&

% 8: 04&

"+ % 80#& &* &0&*7

* % ' = (

& = !)* = = (

& = # & =

BD(K ;(LCB CD M(

!" 7+6 "

- *6* 8: 9 .422/.-.:8*

\+ % ./2J

% 8: 04&

% 8: 04&

\+ % #&/049J

* % ' = (

& = ! " (= =

(

& = # & =

BD(K ;(L FG J M(

!" 7+6 "

\+ % ./2J

- *6* 8: 9 .422/.-.:8*

;% 8: 04&

% 8: 04&

\+ % #&/049

* % ' = (

& = ! ")* = =

(

& = # & =

BD(K ;(LBC G M(

!" $ 7+6

" ++ " % ./2J

$ " % [*2* " *3;* " < " \ ?Z > Z?

- *6* 8: 9 .422/.-.:8*

\- % 8: ./2

\+ + % 8: 04&

% ./2J ?Z > Z?

!" $ 7+6

$ % ./2J ?Z > Z?

\+ % ./2J ?Z > Z?

%80#& :5&*

* % *0*

\- = !! = ! "

= (& " ++ "

!(& + '

!! (.!!

= (0

< ' *0* (

& = # 8 ; 0 K' ;(L BB J M(

!" 7+6

\- % 2/.-.:8*

% 04&

\- % ./2

" +! % ./2CB

$ " % [5 ; N ; \

TX

% ./2J

; ! % ./2CB

' !% 10#3.802 4

% ./2J

% 80#& :5&*

* % *4# 40*

& = ! " = (& !

!! ; (

& " ++ " !

!(& ! !(&

! ! ' ! D

(

. *0* < ' (. #

"

= (

& = # 8 ;9 0 K' ;(L GG M(

!" 6() + 6 4$ '

% + % ./2CB

* %

& = ! 8 ; & % + 8 ; *1* & % +

= ! = (& !! <

! = !! ! !; ! ! !
' (& ! ! !; '
' ! (

. 2 & % + = ! ! & % +
= !! ! = !!
1 & % + = (& ! = ! ! ;
! ! ! = ;
= ' (

& = # & =
BE(K ;(LBBG C M

!" + 6 4$ ' + +
% 80#& :5&*
* %

& = ! =
2 & % + (&
= ! ! D ! (0
& % + 2 & % +
2 (

& = # & =
B (K ;(LC D M

!")0 + 6 4$ '
* % 4

. 2 & % + 1 & % +
! & % + = (0 ! !
& % + =
! = ! ! 2 & % + =
(0 1 & % +
2 & % + 2 (& = ;
! = =
1 & % + (

& = # 8 ;* 8 / = K ;(LC GCF M

Events

!"*3(/1 5)
% ./2CB

4 ; = ! ' ! '
! (

& ' # 8 ;# *' K ;(L FCDJ M

!" , ' 5)
% ./2CB

4 ; ! H ' ! (

& ' # & ' BF(
K ;(LD J D M

!")5+,(0+ +6 5)

% ./2CB

% [7 6 , 0 \

& ; '
! ! (& !!
' ! % # " / "
' $ (

& ' # 8 ;0 ' ! & *' K ;(LCDDG M

!"*)0 +6 + + 5)

% ./2CB

% [\

& ' ' ; =
;(& = ' !
' (& ; !
= ' (& !! ' ! %
(

& ' # 8 ;# & 2 *' K ;(LBBCEC M

!" ('))

% ./2CB = *$$$$$$$$*

- ' + . ;(. !
; ! =
' (

- ! ' ; ; !!
! ! !
' = ! ;(& !
' (

- ! ' ; !
! ' !! = *$$$$$$$$*&& ; !! '
= ! (& ; !! !
! ' (

& ' !! + ; ' (&
' ! !(

& ' # & ' BF(
K ;(L J CG M

8 ;8 & .!!

4 ; ' ;
!! ! = ' (

& ' # & ' BF(
K ;(LCG G M

!"#$(/9 ( ) , + 5)
" % ./2J
3 % 04&J
; % 04&J

& ' ! # , $ ' ! (
& ! ! # , $ '
# , $ ' *!# 1* # *1* (4
' # , $ ' *!# 1* (

& ' # 8 ;P >7 2 ! *' K ;(LCEB C M

!" ( ) , + 5)
3 % 04&J
; % 04&J
" % ./2
% % ./2

& ' ! # , $ ' ! (
& ! ! # , $ '
# , $ ' *!# 1* # *1* (4
' # , $ ' # *1* (

& ' # 8 ;7 2 ! *' K ;(LCEC G M

!" , +' 7+. 5)
" % !

& ' ; ! ! '
/ ! = (# 8 ;/ ! K ;(LCD FE M

& ' # 8 ;/ ! *' K ;(L DGEB M

!" „' 5)
" % !
4 1 % ./2CB

& ' + ' ; !
; !! <
' ;(& ! ;(
& ; ! ! !
!! ; !
(# 8 ;9 K ;(L G BB M

& ; !# ; !* ! ' !!
(

& ' # 8 ;, !! *' K' ;(L DFC M(

Responses

/ ' ; ! >
' ! (

!" , + '.)'
% ./2J
% ./2J
$$+ % 80#& 20,,0&*7

& ' ! ! '
! (& ! ;
(* $$+ ' ! ' !
(

& ' # 8 ;2 ! / K' ;(L F GG M(

Algorithm Naming

& ' ! ! ! !
; !! !! %

* ! = #&/049J !! %

S < X S ' X

& < ! ' <
/ ' ' (&
' ! ' A ; (

. ; ! <! !
!! N80:(

Encoding

& ; > ! (
- 8: ! (

. ! '
!! ! ! ! ' !
' ! !! ' ' !
! ; ' !' ' ! (
; ! (&
' ! ! $! = ! '
(

, %

4

0 ' ! !! %

T *

* ; * *
! ! (

* T D * D D

0 ! ' ! ! ! (&
' ! 8 D
= (# ! ! ' ! 8
= D (

, ! 480#& ,++%

NLISTofFOO

```
1       m       num items
m       LISTofFOO       items
```

, 9 2 ! %

D G 2/.-:8* ! !
D G 9 ! 9

J 8: 9 .422/.-*4&
J
8: 9 .422/.-3 2.&* ! 9 2 !

Errors

LbxClient

```
1       0           Error
1       CARD8       error-base + 0
2       CARD16      sequence number
4                   unused
2       CARD16      lbx opcode
1       CARD8       major opcode
21                  unused
```

Requests

LbxQueryVersion

```
1       CARD8       opcode
1       0           lbx opcode
2       1           request length
=>
1       1           Reply
1                   unused
2       CARD16      sequence number
4       0           reply length
2       CARD16      major version
2       CARD16      minor version
20                  unused
```

& = # 8 ;P 1 K ;(L JF M

LbxStartProxy

```
1       CARD8                       opcode
1       1                           lbx opcode
2       L                           request length
n       NLISTofOPTION-REQUEST       options
p                                   unused, p=pad(n)
```

OPTION-REQUEST

```
1       OPTCODE                     option-code
m       OPTLEN                      option-request-byte-length, (b=m+a+1)
a       DELTAOPT or                 option
        NLISTofNAMEDOPT or
        NLISTofSTR or
        NLISTofPIXMAPMETHOD or
        BOOL
```

& ! (# # # ;
+ K ;(LCEDDD M

```
1       OPTCODE                             option-code
0       LbxOptionDeltaProxy
1       LbxOptionDeltaServer
2       LbxOptionStreamCompression
3       LbxOptionBitmapCompression
4       LbxOptionPixmapCompression
5       LbxOptionMessageCompression             /* also known as squishing */
6       LbxOptionUseTags
7       LbxOptionColormapAllocation
255     LbxOptionExtension
```

+ &8*4 ! < '! %

OPTLEN

```
1       CARD8               b (0 < b <= 255)
```

OPTLEN

```
1       0                   long length header
1       c                   length0, c = b >> 8
```

```
1      d                  length1, d= b & #xff

DELTAOPT
1      CARD8      min-cache-size
1      CARD8      max-cache-size
1      CARD8      preferred-cache-size
1      CARD8      min-message-length
1      CARD8      max-message-length (in 4-byte units)
1      CARD8      preferred-message-length

NAMEDOPT
f      STR             type-name
1      g+1             option-data-length
g      LISTofBYTE      option-data (option specific)

PIXMAPMETHOD
h      STR              name
1      BITMASK          format mask
1      j                depth count
j      LISTofCARD8      depths

=>
=>

1      1                 Reply
1      CARD8             count

0xff      options in request cannot be decoded
2      CARD16            sequence number
4      (a+p-32)/4        reply length
a      LISTofCHOICE      options-reply
p            unused, if (n<24) p=24-n else p=pad(n)

CHOICE
1      CARD8                request-option-index
b      OPTLEN               reply-option-byte-length
c      DELTACHOICE or       choice
      INDEXEDCHOICE or
      NLISTofINDEXEDOPT or
      NLISTofPIXMAPCHOICE or
      BOOL or
      INDEXEDCHOICE

&              !             ( #                              # #   ;
+      K! ;(  LCEDDD     M(


DELTACHOICE
1      CARD8      preferred cache size
1      CARD8      preferred message length in 4-byte units

INDEXEDCHOICE
1      CARD8           index
d      LISTofBYTE      data
```

PIXMAPCHOICE

```
1       CARD8           index
1       CARD8           opcode
1       BITMASK         format mask
e       NLISTofCARD8    depths
```

& = # 8;# ; K';(LBGJFG M(

LbxStopProxy

```
1       CARD8       opcode
1       2           lbx opcode
2       1           request length
```

& = # 8;# ; K';(LBFDEE M(

LbxSwitch

```
1       CARD8       opcode
1       3           lbx opcode
2       2           request length
4       CARD32      client
```

& = # 8;# K';(LCCEGG M(

LbxNewClient

```
1       CARD8       opcode
1       4           lbx opcode
2       L           request length
4       CARD32      client
```

The remaining bytes of the request are the core connection setup.
=>
If the connection is rejected, a core connection reply is sent. Otherwise the reply has the form:

```
1       BOOL        success
1                   change type
      0             no-deltas
      1             normal-client-deltas
      2             app-group-deltas
2       CARD16      major version
2       CARD16      minor version
2       1 + a       length
4       CARD32      tag id
```

& '! ! (

, ! " ! " !(
T ! ! D = (

, ! ! ! ! ' ! T
W %

```
4       CARD32                  resource id base
```

4b LISTofSETofEVENT root input masks

, ! ! ' !! !
T W D %

4 CARD32 resource id base
4 WINDOW root id base
4 VISUALID visual
4 COLORMAP colormap
4 CARD32 white pixel
4 CARD32 black pixel
4c LISTofSETofEVENT root input masks

& = # 8;4 ! K';(L FJ G M(

LbxCloseClient
1 CARD8 opcode
1 5 lbx opcode
2 2 request length
4 CARD32 client

& = # 8;! ! K';(LB BE M(

LbxModifySequence
1 CARD8 opcode
1 6 lbx opcode
2 2 request length
4 CARD32 offset to sequence number

& = # 8;7 #= K';(LC C M(

LbxAllowMotion
1 CARD8 opcode
1 7 lbx opcode
2 2 request length
4 CARD32 number of MotionNotify events

& = # 8;.!! 7 K';(L EJ E M(

LbxIncrementPixel
1 CARD8 opcode
1 8 lbx opcode
2 3 request length
4 COLORMAP colormap
4 CARD32 pixel

& = # 8;0 ;! K';(LBFBBF M(

LbxDelta
1 CARD8 opcode

```
1       9                   lbx opcode
2       1+(2n +p+2)/4       request length
1       n                   count of diffs
1       CARD8               cache index
2n       LISTofDIFFITEM     offsets and differences
p                           unused, p=pad(2n + 2)
```

& = # 8;2! K';(LB JEF M(

LbxGetModifierMapping

```
1       CARD8       opcode
1       10          lbx opcode
2       1           request length
=>
1       1           Reply
1       n           keycodes-per-modifier
2       CARD16      sequence number
4       2n          reply length
4       CARD32      tag
20                  unused
8n      LISTofKEYCODE       keycodes
```

& = # 8;9 7 7 K';(LCF JF M(

LbxInvalidateTag

```
1       CARD8       opcode
1       12          lbx opcode
2       2           request length
4       CARD32      tag
```

& = # 8;0'! & K';(L BE E M(

LbxPolyPoint

```
1       CARD8       opcode
1       13      lbx opcode
2       1+(m+n+p)/4       request length
m       LBXGCANDDRAWABLE       cache entries
n       LISTofLBXPOINT      points (n is data-dependent)
p       0       unused, p=Lpad
```

& = # 8;! K';(LCF F M(

LbxPolyLine

```
1       CARD8       opcode
1       14      lbx opcode
2       1+(m+n+p)/4       request length
m       LBXGCANDDRAWABLE       cache entries
n       LISTofLBXPOINT      points (n is data-dependent)
p       0       unused, p=Lpad
```

& = # 8;!8 K';(L EFD M(

LbxPolySegment

```
1       CARD8       opcode
1       15       lbx opcode
2       1+(m+n+p)/4       request length
m       LBXGCANDDRAWABLE       cache entries
n       LISTofLBXSEGMENT       segments (n is data-dependent)
p       0       unused, p=Lpad
```

& = # 8 ; !# K' ;(LB GFF M(

LbxPolyRectangle

```
1       CARD8       opcode
1       16       lbx opcode
2       1+(m+n+p)/4       request length
m       LBXGCANDDRAWABLE       cache entries
n       LISTofLBXRECTANGLE       rectangles (n is data-dependent)
p       0       unused, p=pad(m+n)
```

& = # 8 ; !/ ! K' ;(LDG EJ M(

LbxPolyArc

```
1       CARD8       opcode
1       17       lbx opcode
2       1+(m+n+p)/4       request length
m       LBXGCANDDRAWABLE       cache entries
n       LISTofLBXARCS       arcs (n is data-dependent)
p       0       unused, p=Lpad
```

& = # 8 ; !. K' ;(L EC F M(

LbxFillPoly

```
1       CARD8       opcode
1       18       lbx opcode
2       1+(3+m+n+p)/4       request length
1       LBXGCANDDRAWENT       cache entries
1             shape
0       Complex
1       Nonconvex
2       Convex
1       p       pad byte count
m       LBXGCANDDRAWUPDATE       optional gc and drawable
n       LISTofLBXPOINT       points (n is data-dependent)
p       0       unused, p=Lpad
```

& = # 8 ;, !! ! K' ;(LCEF M(

LbxPolyFillRectangle

```
1       CARD8       opcode
1       19       lbx opcode
2       1+(m+n+p)/4       request length
```

```
m       LBXGCANDDRAWABLE      cache entries
n       LISTofLBXRECTANGLE      rectangles (n is data-dependent)
p       0       unused, p=Lpad
```

& = # 8 ; ! , !!/ ! K' ;(LBEE M(

LbxPolyFillArc

```
1       CARD8       opcode
1       20       lbx opcode
2       1+(m+n+p)/4       request length
m       LBXGCANDDRAWABLE      cache entries
n       LISTofLBXARC       arcs (n is data-dependent)
p       0       unused, p=Lpad
```

& = # 8 ; ! , !!. K' ;(LDB J M(

LbxGetKeyboardMapping

```
1       CARD8       opcode
1       21       lbx opcode
2       2       request length
1       KEYCODE       first keycode
1       m       count
2             unused
=>
1       1       Reply
1       n       keysyms-per-keycode
2       CARD16       sequence number
4       nm       reply length (m = count field from the request)
4       CARD32       tag
20            unused
4nm       LISTofKEYSYM       keysyms
```

& = # 8 ;9 6 7
K' ;(LCCF M(

LbxQueryFont

```
1       CARD8       opcode
1       22       lbx opcode
2       2       request length
4       FONTABLE       font
=>
1       1       Reply
1       BOOL       compression
2       CARD16       sequence number
4       L       reply length
4       CARD32       tag
20            unused
All of the following is conditional:
12       CHARINFO       min-bounds
4             unused
12       CHARINFO       max-bounds
4             unused
```

```
2       CARD16      min-char-or-byte2
2       CARD16      max-char-or-byte2
2       CARD16      default-char
2       n       number of FONTPROPs in properties
1             draw-direction
0       LeftToRight
1       RightToLeft
1       CARD8       min-byte1
1       CARD8       max-byte1
1       BOOL       all-chars-exist
2       INT16       font-ascent
2       INT16       font-descent
4       m       number of elements in char-infos
8n       LISTofFONTPROP       properties
and either
12m       LISTofCHARINFO       char-infos
or
m       LISTofLBXCHARINFO       char-infos
```

& = # 8;P , K';(L JJ J M(

LbxChangeProperty

```
1       CARD8       opcode
1       23       lbx opcode
2       6       request length
4       WINDOW       window
4       ATOM       property
4       ATOM       type
1       CARD8       format
1             mode
0       Replace
1       Preprend
2       Append
2             unused
4       CARD32       length of data in format units
             (= n for format = 8)
             (= n/2 for format = 16)
             (= n/4 for format = 32)
=>
1       1       Reply
1             unused
2       CARD16       sequence number
4       0       reply length
4       CARD32       tag
20             unused
```

& = # 8; K';(LDGG J M(

LbxGetProperty

```
1       CARD8       opcode
1       24       lbx opcode
2       7       request length
4       WINDOW       window
```

```
4       ATOM      property
4       ATOM      type
0       AnyPropertyType
1       CARD8      delete
3             unused
4       CARD32      long-offset
4       CARD32      long-length
=>
1       1      Reply
1       CARD8      format
2       CARD16      sequence number
4       CARD32      reply length
4       ATOM      type
0       None
4       CARD32      bytes-after
4       CARD32      length of value in format units
             (= 0 for format = 0)
             (= n for format = 8)
             (= n/2 for format = 16)
             (= n/4 for format = 32)
4       CARD32      tag
8             unused
```

& = # 8;9 K';(LC C F M(

LbxTagData

```
1       CARD8      opcode
1       25      lbx opcode
2       3+(n+p)/4      request length
4       CARD32      tag
4       CARD32      length of data in bytes
n       LISTofBYTE      data
p             unused, p=pad(n)
```

& = # 8;& 2 K';(L F JF M(

LbxCopyArea

```
1       CARD8      opcode
1       26      lbx opcode
2       L      request length
1       CARD8      source drawable cache entry
1       LBXGCANDDRAWENT      cache entries
4 or 0      DRAWABLE      optional source drawable
b       LBXGCANDDRAWUPDATE      optional gc and dest drawable
c       LBXPINT16      src-x
d       LBXPINT16      src-y
e       LBXPINT16      dst-x
f       LBXPINT16      dst-y
g       LBXCARD16      width
h       LBXCARD16      height
p             unused, p=Lpad
```

& = # 8; . K';(L DG M(

LbxCopyPlane

```
1       CARD8       opcode
1       27       lbx opcode
2       L       request length
4       CARD32       bit plane
1       CARD8       source drawable cache entry
1       LBXGCANDDRAWENT       cache entries
4 or 0       DRAWABLE       optional source drawable
b       LBXGCANDDRAWUPDATE       optional gc and dest drawable
c       LBXPINT16       src-x
d       LBXPINT16       src-y
e       LBXPINT16       dst-x
f       LBXPINT16       dst-y
g       LBXCARD16       width
h       LBXCARD16       height
p             unused, p=Lpad
```

& = # 8; ! K';(LC FFB M(

LbxPolyText8

```
1       CARD8       opcode
1       28       lbx opcode
2       L       request length
1       LBXGCANDDRAWENT       cache entries
a       LBXGCANDDRAWUPDATE       optional gc and drawable
b       LBXPINT16       x
c       LBXPINT16       y
n       LISTofTEXTITEM8       items
p             unused, p=Lpad
```

& = # 8; !&;J K';(LBCBG M(

LbxPolyText16

```
1       CARD8       opcode
1       29       lbx opcode
2       L       request length
1       LBXGCANDDRAWENT       cache entries
a       LBXGCANDDRAWUPDATE       optional gc and drawable
b       LBXPINT16       x
c       LBXPINT16       y
2n       LISTofTEXTITEM16       items
p             unused, p=Lpad
```

& = # 8; !&; K';(L CBBJ M(

LbxImageText8

```
1       CARD8       opcode
1       30       lbx opcode
2       L       request length
1       LBXGCANDDRAWENT       cache entries
a       LBXGCANDDRAWUPDATE       optional gc and drawable
```

```
b       LBXPINT16       x
c       LBXPINT16       y
n       STRING8       string
p             unused, p=Lpad
```

& = # 8;0 &;J K';(L G G M(

LbxImageText16

```
1       CARD8       opcode
1       31       lbx opcode
2       L       request length
1       LBXGCANDDRAWENT       cache entries
a       LBXGCANDDRAWUPDATE       optional gc and drawable
b       LBXPINT16       x
c       LBXPINT16       y
2n       STRING16       string
p             unused, p=Lpad
```

& = # 8;0 &; K';(LC EJD M(

LbxQueryExtension

```
1       CARD8       opcode
1       32       lbx opcode
2       2+(n+p)/4       request length
4       n       length of extension name
n       STRING8       extension name
p             unused, p=pad(n)
=>
1       1       Reply
1       n       number of requests in the extension
2       CARD16       sequence number
4       0 or 2*(m + p)       reply length, m = (n+7)/8
1       BOOL       present
1       CARD8       major opcode
1       CARD8       first event
1       CARD8       first error
20            unused
m       LISTofMASK       optional reply-mask
p             unused, p=pad(m)
m       LISTofMASK       optional event-mask
p             unused, p=pad(m)
```

& = # 8;P *; K';(LC B M(

LbxPutImage

```
1       CARD8       opcode
1       33       lbx opcode
2       L       request length
1       CARD8       compression method
1       LBXGCANDDRAWENT       cache entries
a       PIPACKED            bit-packed
b       LBXGCANDDRAWUPDATE       optional gc and drawable
```

```
c       LBXCARD16       width
d       LBXCARD16       height
e       LBXPINT16       x
f       LBXPINT16       y
n       LISTofBYTE       compressed image data
p             unused, p=Lpad
```

0 ! ! = ! 0 . 6*2
!! %

PIPACKED

```
1       #x80 | (format << 5) | ((depth -1) << 2)
```

+ 0. 6*2 %

PIPACKED

```
1       (depth -1) << 2)
1       (format << 5) | left-pad
```

& = # 8 ; 0 K! ;(LB B J M(

LbxGetImage

```
1       CARD8       opcode
1       34       lbx opcode
2       6       request length
4       DRAWABLE       drawable
2       INT16       x
2       INT16       y
2       CARD16       width
2       CARD16       height
4       CARD32       plane mask
1       CARD8       format
3             unused
=>
1       1       Reply
1       CARD8       depth
2       CARD16       sequence number
4       (n+p)/4       reply length
4       (m+p)/4       X reply length; if uncompressed, m=n
4       VISUALID       visual
0       None
1             compression method
15             unused
n       LISTofBYTE       data
p             unused, p=pad(n)
```

& = # 8 ;9 0 K! ;(LB J M(

LbxBeginLargeRequest

```
1       CARD8       opcode
1       35       lbx opcode
```

```
2       2       request length
4       CARD32      large request length
```

& = # 8;: 8 /= K';(LC BG M(

LbxLargeRequestData

```
1       CARD8       opcode
1       36      lbx opcode
2       1+n      request length
4n      LISTofBYTE      data
```

& = # 8;8 /= 2 K';(LC JB M(

LbxEndLargeRequest

```
1       CARD8       opcode
1       37      lbx opcode
2       1       request length
```

& = # 8;* 8 /= K';(LC JD M(

LbxInternAtoms

```
1       CARD8       opcode
1       38      lbx opcode
2       1+(2+m+n+p)/4       request length
2       m       num-atoms
n       LISTofLONGSTR       names
p             pad p=Lpad
=>
1       1       Reply
1             unused
2       CARD16      sequence number
4       a       reply length, a = MAX(m - 6, 0)
4*m       LISTofATOM      atoms
p             pad p = MAX(0, 4*(6 - m))

LONGSTR
2       c       string length
c       STRING8       string
```

& = # 8;0 . K';(LB C M(

LbxGetWinAttrAndGeom

```
1       CARD8       opcode
1       39      lbx opcode
2       2       request length
4       CARD32      window id
=>
1       1       Reply
1             backing store
0       NotUseful
1       WhenMapped
```

```
2       Always
2       CARD16      sequence number
4       7       reply length
4       VISUALID        visual id
2             class
1       InputOutput
2       InputOnly
1       BITGRAVITY      bit gravity
1       WINGRAVITY      window gravity
4       CARD32      backing bit planes
4       CARD32      backing pixel
1       BOOL      save under
1       BOOL      map installed
1             map state
0       Unmapped
1       Unviewable
2       Viewable
1       BOOL      override
4       COLORMAP      colormap
4       SETofEVENT      all events mask
4       SETofEVENT      your event mask
2       SETofDEVICEEVENT      do not propagate mask
2             unused
4       WINDOW      root
2       INT16      x
2       INT16      y
2       CARD16      width
2       CARD16      height
2       CARD16      border width
1       CARD8      depth
1             unused
```

& = # 8;9 - . . 9 K';(LC CJB M(

LbxGrabCmap

```
1       CARD8       opcode
1       40       lbx opcode
2       2       request length
4       COLORMAP       colormap
=>

0             !     !! %


1       1       Reply
1       #x80       flags
2       CARD16       sequence number
4       0       reply length
24             unused

If smart-grab is false, the reply is as follows:

1       1       Reply
1             flags (set of)
```

```
       #x40      auto-release
       #x20      three-channels
       #x10      two-byte-pixels
lower four bits specifies bits-per-pixel
2      CARD16      sequence number
4      L      reply length
m      CHAN or CHANNELS      cells (CHAN if !three-channels)
p      0      pad(m)

CHANNELS
a      CHAN      red
1      5            next channel
b      CHAN      green
1      5      next channel
c      CHAN      blue
1      0      list end

CHAN
d      LISTofLBXPIXEL

LBXPIXEL
e      PIXELPRIVATE or
      PIXELPRIVATERANGE or
      PIXELALLOC or
      PIXELALLOCRANGE

PIXELPRIVATE
1      1      pixel-private
f      PIXEL      pixel

PIXEL
f      CARD8 or CARD16      (CARD8 if !two-byte-pixels)

PIXELPRIVATERANGE
1      2      pixel-private-range
f      PIXEL      fist-pixel
f      PIXEL      last-pixel

PIXELALLOC
1      3      pixel-private
f      PIXEL      pixel
g      COLORSINGLE or COLORTRIPLE      color       (COLORSINGLE if
three-channels)

COLORSINGLE
h      CARD8 or CARD16      value (CARD8       if bits-per-rgb =< 7)

COLORTRIPLE
h      COLORSINGLE      red
h      COLORSINGLE      green
h      COLORSINGLE      blue

PIXELALLOCRANGE
1      4      pixel-private
```

```
f       PIXEL       first-pixel
f       PIXEL       last-pixel
j       LISTofCOLORSINGLE or        color        (COLORSINGLE if three-channels)
      LISTofCOLORTRIPLE
```

& = # 8;9 K';(L G BB M(

LbxReleaseCmap

```
1       CARD8       opcode
1       41       lbx opcode
2       2       request length
4       COLORMAP       cmap
```

& = # 8;/! K';(LCD FE M(

LbxAllocColor

```
1       CARD8       opcode
1       42       lbx opcode
2       5       request length
4       COLORMAP       colormap
4       CARD32       pixel
2       CARD16       red
2       CARD16       green
2       CARD16             blue
2             unused
```

& = # 8;.!! ! K';(L GDD M(

LbxSync

```
1       CARD8       opcode
1       43       lbx opcode
2       1       request length
=>
1       1       Reply
1       n       unused
2       CARD16       sequence number
4       0       reply length
24              unused
```

& = # 8;# K';(LCGF M(

Events

LbxSwitchEvent

```
1       base + 0       code
1       0       lbx type
2       CARD16       sequence number
4       CARD32       client
24              unused
```

& ' # 8;# *' K';(LCCFDJ M(

LbxCloseEvent

```
1       base + 0       code
1       1       lbx type
2       CARD16       sequence number
4       CARD32       client
24              unused
```

& ' # 8;! *' K';(L FB B M(

LbxInvalidateTagEvent

```
1       base + 0       code
1       3       lbx type
2       CARD16       sequence number
4       CARD32       tag
4             tag-type
1       LbxTagTypeModmap
2       LbxTagTypeKeymap
3       LbxTagTypeProperty
4       LbxTagTypeFont
5       LbxTagTypeConnInfo
20              unused
```

& ' # 8;0'! & *' K';(LBCG M(

LbxSendTagDataEvent

```
1       base + 0       code
1       4       lbx type
2       CARD16       sequence number
4       CARD32       tag
4             tag-type
3       LbxTagTypeProperty
20              unused
```

& ' # 8;# & 2 *' K';(LBGCFC M(

LbxListenToOne

```
1       base + 0       code
1       5       lbx type
2       CARD16       sequence number
4       CARD32       client
#xFFFFFFFF
a client not managed by the proxy
24              unused
```

& ' # 8;8 &+ K';(LBEBG M(

LbxListenToAll

```
1       base + 0       code
1       6       lbx type
2       CARD16       sequence number
```

```
28              unused
```

& ' # 8;8 &.!! K';(L G E M(

LbxQuickMotionDeltaEvent

```
1       base + 1        code
1       CARD8       delta-time
1       INT8        delta-x
1       INT8        delta-y
```

& ' CB (

& ' # 8;P >7 2 ! *'
K';(LDGB J M(

LbxMotionDeltaEvent

```
1       base + 0        code
1       7                   lbx type
1       INT8        delta-x
1       INT8        delta-y
2       CARD16        delta-time
2       CARD16        delta-sequence
```

& ' CB (

& ' # 8;7 2 ! *' K';(LCGGCC M(

LbxReleaseCmapEvent

```
1       base + 0        code
1       8                   lbx type
2       CARD16      sequence number
4       COLORMAP        colormap
24              unused
```

& ' # 8;/ ! *' K';(L B M(

LbxFreeCellsEvent

```
1       base + 0        code
1       9                   lbx type
2       CARD16      sequence number
4       COLORMAP        colormap
4       PIXEL       pixel start
4       PIXEL       pixel end
16              unused
```

& ' # 8;, !! *' K';(LCJGD M(

Re-encoding of X Events

& ! = !! ' CB (& 8: '
' ; '

' (& '
(& ; !! ' CB
! (

8: ' !! < =
! %

```
KeyOrButton             32
EnterOrLeave            32
Keymap                  32
Expose                  20
GraphicsExposure        24
NoExposure              12
VisibilityNotify        12
CreateNotify            24
DestroyNotify           12
UnmapNotify             16
MapNotify               16
MapRequest              12
Reparent                24
ConfigureNotify         28
ConfigureRequest        28
GravityNotify           16
ResizeRequest           12
Circulate               20
Property      Notify    20
SelectionClear          20
SelectionRequest        28
SelectionNotify         24
Colormap      Notify    16
MappingNotify           8
ClientMessage           32
Unknown                 32
```

Responses

LbxDeltaResponse

```
1       event_base + 0      event code
1       2       lbx type
2       1+(2+2n+p)/4        request length
1       n       count of diffs
1       CARD8       cache index
2n       LISTofDIFFITEM       offsets and differences
p             unused, p=pad(2n)
```

& # 8 ;2 ! / K' ;(LCDGDB M