X Display Manager Control Protocol

X.Org Standard

Keith Packard, Massachusetts Institute of Technology

X Display Manager Control Protocol: X.Org Standard

© ! "##$ % & ' (

) * * * * + (
) * , -. * + -/ . * + + ((+ ( )
( ) *) (((0 * . * +)
+) . * + *((1 * + 2

% 3) ((*
. * +

%45 .&6%78 5 9. & 9:5: -8. 9.- 79%4&;% 78 8<%= &6 8<= 9<: 5 5.. & 9> ?95: 9< ?;:9<'
@;% <&% ?9>9%5: %& %45 78 8<%95. &6 >5 48<%8@9?9%= 69%<5.. 6& 8 8 %9 ;?8 ; &.5 8<:
<&<9<6 9<'5>5<% 9< <& 5 5<% .48?? %45 & 5< ' &; @5 ?98@?5 6& 8<= ?89> :8>8'5. &
&%45 ?98@9?9%= 745%45 9< 8< 8 %9&< &6 &<% 8 % %& % & &%45 79.5 8 9.9<' 6 &> &;% &6
& 9< &<<5 %9&< 79%4 %45 .&6%78 5 & %45 ;.5 & &%45 :58?9<'. 9< %45 .&6%78 5

5A) * % & ' ((3 +
) (. * + + (+ (B *) % & ' (

) * % & ' (

Table of Contents

('
" &3 3 + * C
C : % $
$ 6) D
D E
E . %)
. :) !
! 5 ""
: 6) "E
> (* (: 9: 6) "
8("!

Chapter 1. Purpose and Goals

% (* : > , :> / 3
(*))) * ( ) (F(3 *)
) @ ( ) (+) * +
* (+ 3 ,6
A) 3) fork/exec F(/
8) ,) (+ * /)
A) * ( ) (

6) * 3 + * (3)) ( ) (
(+) . * (
1(+ +) +)
%) (+ ( ) (4 + 3 +
3) +)(( ) + + , /
9 3) +)))
) + (* + (
7 + ((* + 2

G % * A + (9 (
+ 3) + ((3

G 8 * 3 + (+) *
*) ,6 A) (H *)) (
(3) 3) / 9 (
* * (3 (
) (*

G % * A * 9 (*
3 8> ((*
)) * () * (
B * + (3 , / +
3 (((+ *(3
+ 3

% ((+ (3
3 *))(9 * F(
* 3) 1 + *) (
* (:> ((* :> +
4 + 3 (()
*)
6 (
) 3 + * F( )

% ( ))() , (*
/ F( )
) , + / * (*
F( ))() + +

. ( ))(*
%) (A * :> 2

G 9 (* 3 *))(+
) 3 @ ((+ ,* A)
+ / 3) 3 *

G 9 (* 3 (
B)) (*

G 9 (3) 3 * (3 *
3 3 (B))

G @ (*) * (:>)(
* A ( ) 3 * ( ))

Chapter 2. Overview of the Protocol

:> 3 ( ))
3 *) 8 + + 3 I*9: > I* +
(:>))(+ (*
% + ( ) 9 +) *
* (*) B +)
(*)) + * %
3 (3) 3 *))
*) 0) +

:) (** ( ) 8 A (
(**)(* %)
F(+ 9* (
A))) + %
) , + % /) +
(** 9) 3)) (( )
. H + 3 * 3) (
) * + 3

: (F( ) * (*
& (3 (* F(+ (( )
) *) +
) 3 (* F(6 + + (
))() 3) ( ) +
(() * F(() @(*
) + 9) ()
*) (F(() * (
) 3

% >)(* % 3 >
*)) ((* * F(
>) *) 3
> 3)(+) + 3
*) ((% (
+ + + + > 3 , */
& >)) 3)
, * + / ( )(

Chapter 3. Data Types

:> 3 * 9 3 (+)
* * ,-@ 5 - / 8 :> + (
F(* + ( )
** *)) 8 * + (+

8 :!		8 (
8 : E	"	%+ (
8 :C"	$	6 ((
8 8=!	J"	% (8 : E * + * 8 :! % 3 (* 8 : E * , / * () * 8 :! 3 (* +
8 8= E	"K)J	% 8 :! ,)/ + * () * 8 : E 3 (* +
8 8=C"	$K J	% 8 :! , / + * () * 8 :C" 3 (* +
8 8= *8 8=!	L	% 8 :! + * () * 8 8=! 3 (* +

Chapter 4. Packet Format

8 :> 3 * + *) 2

"	8 : E	3 ()
"	8 : E	
"	8 : E	M *)
	LLL	*

% * * + 2

() % * 3 * :>

F(:)) (
3 *) % * +
, /

& % * + *
(* *
+ 3 (, 3 +/2

! " # $ %
& # ' (%) %)

? * % * * *) * +
* E 5 ** *
) +
3 (@ (3)
A))
3)(
(

( ) (* * + 2

%)(3

" % *) (() * *
3 ( ))

C % (A , *) 3 /

$ 9* * & # ' . 9: ( ) 3 (
%

Chapter 5. Protocol

5 * + @ (3 3
+ + > *
3 *) (3 %
*) +
(*) (

T : N >
T 8 6 2
T T 2 8 8=*8 8=!
T T T . * * ( )
(%) + *
(!
T T .)
T T T 8 Query *) *
* + 3))
3 % (+ Will-
ing * + 3 Unwilling *

8 BroadcastQuery) Query
A 3
+ , (+ / 4 + 3 (Query
F(+ 3
( ) BroadcastQuery F(

8 IndirectQuery + +)
* + F(*
) (ForwardQuery 9
+ *)
(+ (O (*
) ( )) 3 3
%))) Willing

5 **) 2
T T T T % Query *
9* (Query)(
) +)(Query
% Query)
*)) Willing Unwilling

% BroadcastQuery)
5) 3
+ + Unwilling

% IndirectQuery
) + F(F(
* + *) (
ForwardQuery % A
) (
) 3 3 % *(*
) BroadcastQuery except
that BroadcastQuery * +

T T T 2

T T Willing Unwilling

T T T) 0. (2

T T)2

T T T <) 3 F(

9 2

T T T T < * BroadcastQuery IndirectQuery +
* (3 Willing

T T T . (2

T T T T + + *
( )

T T %) (0) 2

T T T 8 A **) ((
(+ *
. " ** * * " C"
( ) * "E
% ( ) (+ ((
9 + + + (+
((

ForwardQuery

T) > N . >

8 6 2

T T 2 8 8=!

T T T . * + *

T T 2 8 8=!

T T T . * * *

T T 2 8 8=*8 8=!

T T T 9 (* 8(<) +
3 IndirectQuery

T .) 2

T T 7)) 3 IndirectQuery
* ForwardQuery
*) 3 3 (
) + IndirectQuery
+ 3 *) % 8 *
)( ) (

( ) + 5
) Willing * +
3 3

ForwardQuery) BroadcastQuery
) + 3 (
(Unwilling

T 2

T T Willing

T) 0. (2

T T 9 BroadcastQuery

T %) (0) 2

T T ? *)) (3
)

Willing

T > N :

8 6 2

T T 2 8 8=!

T T T . * ( ) *)
** Query , BroadcastQuery , Indi-
rectQuery) A
((F(Request %
( ) *
)(Query (8(
<) *) Willing 3

% *) F(
(** 3 * (

T T 2 8 8=!

T T T 9 () *)
+ + % *
* *

T T 2 8 8=!

T T T 9 () (*
% ((3 0 () *(
*) % *
* *

T .) 2

T T 8 Willing)) 3
*) 9 Query ,
BroadcastQuery , ForwardQuery ( ))
)) 3 3 ,* A))
(/

T) 0. (2

T T)2

T T T Willing 3

9 2

T T T T < * BroadcastQuery IndirectQuery +
* (3 Willing .

T T T . (2

T T T T % ((F(
(3

T %) (0) 2

T T ? *))
(3)

Unwilling

T > N:

8 6 2

T T % 4) . (* Willing %
. (* ((* *(*
3

T .) 2

T T 8 Unwilling)
Query F(, BroadcastQuery Indirect-
Query F(/ *) + F(*)
) %) +
3 ( ) () *

T)0. (2

T T)2

T T T Unwilling 3

9 2

T T T T : * 3 Unwilling .

T T T . (2

T T T T % ((Query)
(3

T %) (0) 2

T T ? *))
(3)

Request

T : N>

8 6 2

T T 2 8:E

T T T . * A * (3 *
+ % 3 (+ B
*) ( )(

T T 2 8 8=E

T T T . *) 3
9* (

B +
*)

T T 2 8 8=*8 8=!

T T T 6 3 (
+
* 3

T T 2 8 8=!

2 8 8=!

T T T . * (A
) 3 * + % 8(
: A)
+) * ((*

T T ! 2 8 8=*8 8=!

T T T . * + * (B (
%)) 1 +
+ *) (B

T T " # $ 2 8 8=!

T T T ( )) +
8(: * .
+ > (* (: 9: 6)

T .) 2

T T 8 Request *
F(9: *
9*) + 3
((Accept + 3 9:
(* (F(Manage F(& +
((Decline

T 2

T T Accept , Decline

T) 0. (2

T T)2

T T T F(3)

T T T 9 2

T T T T :) (+ *

T T T . (2

T T T T : Request)

T T T)2

T T T T > 3 (*)

T T T 9 2

T T T T <

T T T . (2

T T T T 5) Request)
. 9: + A
Accept . 9* A

) +) +
) . 9: 9*
) + + + . 9:O
+ Accept +

T %) (0) 2

T T %) (* " A ** C" 8*
) "E 3 (
(

Accept

T > N :

8 6 2

T T $ 2 8 :C"

T T T 9 *)

T T 2 8 8=!

2 8 8=!

T T T 9 (
) 9* 8(: 3 (A
( )
(

T T ! 2 8 8=!

! 2 8 8=!

T T T 9 * (
B) + (*
XOpenDisplay * Manage 3

T .) 2

T T 8 Accept) Request
*) + *
% . 9: (* *)
+ ((
F(Manage % . 9: C" ()
)) Accept)(
(F(3 *)

9* (*) 3 Decline +
(+ Status)

T) 0. (2

T T)2

T T T Accept Decline 3

T T T 9 2

T T T T :) (+ * Re-
quest .

T T T . (2

T T T T : Request)

T T)2

T T T > 3 (*

T T T 9 2

T T T T : 3 Accept * Manage

T T T . (2

T T T T : Accept) *
Manage)

T %) (0) 2

T T ? *))
(3)

Decline

T > N:

8 6 2

T T 2 8 8=!

T T T 9 () *
*(* 3

T T 2 8 8=!

2 8 8=!

T T T 9 (
) 9* 8(: 3 (A
( )
(

T .) 2

T T 8 Decline) Request
*) (+ *
% + 3 *)
Willing 3 (F(

T) 0. (2

T T .) * Accept .

T %) (0) 2

T T ? *))
(3)

Manage

T : N>

8 6 2

T T $ 2 8 :C"

T T T . (B 9: (
Accept

T T 2 8 : E

T T T >( ) 3 (3 (Request

T T 2 8 8=!

T T T . * * . :
6) + (*) * *

T .) 2

T T 8 Manage)
9* . 9:)
(O + (+
Refuse Failed (. 9:) (
((*(
(3 ( ) 9
Manage (% + +
) 3 3 (
3 *) 3 * (
,+ + 3 (Failed
/

T 2

T T + (* Refuse , Failed .

T) 0. (2

T T)2

T T T Manage 3)

T T T 9 2

T T T T :) (+ *

T T T . (2

T T T T : Manage)

T T)2

T T T Manage 3 (*)

T T T 9 2

T T T T . +) .
9:

T T T . (2

T T T T Manage

T T T 9 2

T T T T . 9:) A . 9:

T T T . (2

T T T T Refuse)

T T)2

T T T :)

T T T 9 2

T T T T : (*

T T T . (2

T T T T Failed) (()

T T)2

T T T : (*)
3 (*) ((

T T T 9 2

T T T T Manage 3 + . 9:)
)

T T T . (2

T T T T Manage 8)
+ (+ + (
3) + 3 (3
(* Failed
O Manage

T %) (0) 2

T T %) (* " A ** C" 8*
) "E 3 (
(

Refuse

T > N:

8 6 2

T T $ 2 8 :C"

T T T . (. 9: 3 Manage

T .) 2

T T 8 Refuse) + . 9:
3 Manage) (.
9: % (() 3 Accept
(Request

T) 0. (2

T T)2

T T T 5)

T T T 9 2

T T T T :) (+ * +
Refuse Failed .

T T T . (2

T T T T : Manage)

T %) (0) 2

T T ? *))
(3)

Failed

T > N:

8 6 2

T T $ 2 8 :C"

T T T . (. 9: 3 Manage

T T 2 8 8=!

T T T 9 () * *
(

T .) 2

T T 8 Failed) +)
Manage

T) 0. (

T T .) * Refuse .

KeepAlive

T : N >

8 6 2

T T 2 8 : E

T T T . A *

T T $ 2 8 :C"

T T T . (. 9: 3 Manage
(* (

T .) 2

T T 8 KeepAlive) (
3 *) (%)
(+ Alive + 3 3 *

% + 3 +)
(8 F(KeepAlive
(* * Alive
F(*) *

% A (*) 3
+) + * % (
*) 3 *)
) (KeepAlive + (
))

T 2

T T Alive

T) 0. (2

T T)2

T T T > 3
3

T T T 9 2

T T T T < Alive (

T T T . (2

T T T T) + A
**O " ()
(* C#

Alive

T > N :

8 6 2

T T % & 2 8 :!

T T T 9 * . 9: (
3 % 3 (B * 3
* 3

T T $ 2 8 :C"

T T T . * 9: * ((O *
7 3 * (B

T .) 2

T T 8 Alive KeepAlive F(9*
(3) (
. 9: % (*)
) (*)

Chapter 6. Session Termination

7 3 + ,
+ Manage /+) 9*
+ 3 (
(9*)(3)((
* +)
(& + (
+ 3) ,
/

%))) *)
KeepAlive % A) (*
KeepAlive * ** (
* A + +) + ((KeepAlive
) + *)

Chapter 7. State Diagrams

% * +) 3 *
) 8 3 (* F(+

: 2

2 ; F(N '

; F( ) N

; F(N

' 2 . Query N '

' 2 3 Willing N

3 Unwilling N

%) (N '

2 . BroadcastQuery

N '

' 2 3 Willing N &

; F(N

%) (N

& 2 8 + (

N '

2 . IndirectQuery

N '

' 2 3 Willing N &

; F(N

%) (N

& 2 8 + (

N '

2 . Request

N '

' 2 3 Accept N &

3 Decline N

%) (N

& 2 .3 . 9:

. Manage + . 9:

N &

& 2 3 XOpenDisplay : N

3 Refuse +) . 9: N

3 Failed +) . 9: N

%) (N &

2 : (*) (

N

2 : KeepAlive N ( )

+ * *

N

( ) 2 . KeepAlive + (. 9:

N)

) 2 3 Alive +) . 9: N

3 Alive +) . 9:
68?.5 . (N

6) (+ (3 Alive N

%) (N ( )

2 , * / N
+

? N

N

> 2

2 3 Query N '

3 BroadcastQuery N

3 IndirectQuery N

3 ForwardQuery N # 3

Request N ′

3 Manage N &

8 3) N #

3 KeepAlive N)

N

′ 2 9* +) N &

N &

2 9* +) N &

N

2 . ForwardQuery)

9* +) N &

N

2 : * +) N
&

N

& 2 . Willing

N

& 2 . Unwilling N

′ 2 9*) + + N

N

2 ' . 9: 3 . 9:
()) +

. Accept

N

2 . Decline

N

& 2 9* . 9:) 3 . 9: N

9* . 9:) . 9: * *
((3 N

N #

2 . Refuse

N

2 %)

XOpenDisplay

& (N

N #

2 . Failed

N

2 . +

N

2 XCloseDisplay

N

) 2 . Alive ((

N

Chapter 8. Protocol Encoding

7 :>)) * 9 ; :) ,;: /
() (7 (;: 3 9 3$ @ P(
3 ;: 7 (;: 3 9 3E @ P(
3)(98<8 :>)(
* 66# 2#2#2#2#2#2#2 "@ ,+ 3 /
)(% 3 () +
E

* +	,
@ P(	
P(	"
9 P(	C
6 + P(	$
7	D
; +	E
F(	
8	!
:	
>	#
*(	
6	"
8 3	C
8 3	$

8 3 (3	*	()	3	*	%)	(%)
8 3 (3	*	()	3	*	%)	(%)

*) * + 2

```
Query BroadcastQuery IndirectQuery


     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Query, BroadcastQuery or IndirectQuery)
     2     CARD16     length
     1     CARD8      number of Authentication Names sent (m)
     2     CARD16     length of first Authentication Name (m1)
     m1    CARD8     first Authentication Name
     ...            Other Authentication Names
```

< A * *

```
ForwardQuery
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always ForwardQuery)
     2     CARD16     length
     2     CARD16     length of Client Address (m)
```

```
     m     CARD8     Client Address
     2     CARD16     length of Client Port (n)
     n     CARD8     Client Port
     1     CARD8     number of Authentication Names sent (o)
     2     CARD16     length of first Authentication Name (o1)
     o1     CARD8     first Authentication Name
     ...           Other Authentication Names


Willing
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Willing)
     2     CARD16     length (6 + m + n + o)
     2     CARD16     Length of Authentication Name (m)
     m     CARD8      Authentication Name
     2     CARD16     Hostname length (n)
     n     CARD8      Hostname
     2     CARD16     Status length (o)
     o     CARD8      Status


Unwilling
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Unwilling)
     2     CARD16     length (4 + m + n)
     2     CARD16     Hostname length (m)
     m     CARD8      Hostname
     2     CARD16     Status length (n)
     n     CARD8      Status


Request
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Request)
     2     CARD16     length
     2     CARD16     Display Number
     1     CARD8      Count of Connection Types (m)
     2xm   CARD16     Connection Types
     1     CARD8      Count of Connection Addresses (n)
     2     CARD16     Length of first Connection Address (n1)
     n1     CARD8      First Connection Address
     ...               Other connection addresses
     2     CARD16     Length of Authentication Name (o)
     o     CARD8      Authentication Name
     2     CARD16     Length of Authentication Data (p)
     p     CARD8      Authentication Data
     1     CARD8      Count of Authorization Names (q)
     2     CARD16     Length of first Authorization Name (q1)
     q1     CARD8      First Authorization Name
     ...               Other authorization names
     2     CARD16     Length of Manufacturer Display ID (r)
     r     CARD8      Manufacturer Display ID


Accept
```

```
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Accept)
     2     CARD16     length (12 + n + m + o + p)
     4     CARD32     Session ID
     2     CARD16     Length of Authentication Name (n)
     n     CARD8      Authentication Name
     2     CARD16     Length of Authentication Data (m)
     m     CARD8      Authentication Data
     2     CARD16     Length of Authorization Name (o)
     o     CARD8      Authorization Name
     2     CARD16     Length of Authorization Data (p)
     p     CARD8      Authorization Data


Decline
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Decline)
     2     CARD16     length (6 + m + n + o)
     2     CARD16     Length of Status (m)
     m     CARD8      Status
     2     CARD16     Length of Authentication Name (n)
     n     CARD8      Authentication Name
     2     CARD16     Length of Authentication Data (o)
     o     CARD8      Authentication Data


Manage
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Manage)
     2     CARD16     length (8 + m)
     4     CARD32     Session ID
     2     CARD16     Display Number
     2     CARD16     Length of Display Class (m)
     m     CARD8      Display Class


Refuse
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Refuse)
     2     CARD16     length (4)
     4     CARD32     Session ID


Failed
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Failed)
     2     CARD16     length (6 + m)
     4     CARD32     Session ID
     2     CARD16     Length of Status (m)
     m     CARD8      Status


KeepAlive
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always KeepAlive)
```

```
     2     CARD16     length (6)
     2     CARD16     Display Number
     4     CARD32     Session ID


Alive
     2     CARD16     version number (always 1)
     2     CARD16     opcode (always Alive)
     2     CARD16     length (5)
     1     CARD8      Session Running (0: not running 1: running)
     4     CARD32     Session ID (0: not running)
```

Chapter 9. Display Class Format

% : * * Manage ( )
)) *) (% * *
9.& ?8%9< * + *) 2

ManufacturerID-ModelNumber

@) *)(A (* Q - . : * ?
* + R % > (* (9: (+
() % > <() * *
+) (* (H (% (()
( ) (* (3 (*
(3 (A * (

Chapter 10. Manufacturer Display ID Format

% ( )) + 3 %
))(3 + (* (3 %
> (* (: 9: * * Request * (
%) +) + ()
% * (F(*
*)2

-Ethernet-8:0:2b:a:f:d2

9 * *)2

ManufacturerID-ModelNumber-SerialNumber

% > (* (9: > <() . <() (9.&
?8%9< A (Q – . * ? * + R

7 ( ) ((> (* (
: 9: 3 () % *) (
) * (

Chapter 11. Authentication

9 3) + (:> (
3) 8(<) 8(
: * Request 9) +) (
* & ( ) 3

9 ((3))(3 * (*
3 (( ) % + (
*) 8 A) * ( ) * + (
- :> 8;%45<%9 8%9&< - ( ) %) (DE
3 E$ * (8 A)
(B - :> 8;%4& 9S8%9&< - + (% DE
E$ () + , / %)
* + B 7) E$
3 (! * + + , / %
+) (F(+
*

8 () 2

%) 3 % ( ))
) (* (+ (9)(
3 *) * ((+ 3 ( ) *)
+ %)) + (*) *
)

" %) (()

.) * * 2

G Q R' M * A : ,

G Q- RK M * A - + ,

G . M 3)

G / M E$) ()

G *0* M (:>

G *1* M 3)

G *2* M (B

5 + (: 5 . ,:5. 69 . $E C/O
E$ + B * E$ @ E$
+ (2

Q R' M Q R' Q " A Q R' R'

% * (Request 2

0 F(M Q/ R'

6 Accept)) (0_8 2

/ M Q*0* F(RK

0_8 M Q / J R

% Accept (B * (3
8 * (B - :> 8;%4& 9S8%9&< - * +

% Accept (B) - :> 8;%4& 9S8%9&< -
% (B 2

2_8 M Q*1* R

% (B *) * (+ 3 (
:> 8;%4& 9S8%9&< (B) (* + 2

M *

M ( ) ,C" /

2 M Q*/* R[1]

6 % $! C" 9 3$ *
* + E () * 6) *
)(% (3 (*2* " * (B
(3 % 3 3
% * +)(2

G /)( ) 3 (*) :>

G)(+ "## *) 9*)
3 * ( )

G <) , / 3 3 "##
,"#) (/