The XIM Transport Specification

Revision 0.1

Takashi Fujiwara, FUJITSU LIMITED

The XIM Transport Specification: Revision 0.1

© ! "#$ %" &$'$ ()
© ! *

Abstract

+ , + - + , + - $' % . +
* . + + + /0$/)(- +1

/ * - , , + + , , - + - - +
* , 2 3% , 45 - % , + + - *
+ * - , * - + -0 + , % , - *
* % , , - - + , + - 6

. + + - * + + - - + ,
% , 1

7(%8 9: ($% / 8 $)() 3:% $%4 9$ 78" 9: :; < 8 :;< =$;) (/ (%% 8 $'/&$() $; &")$;>
?" ;8 &$'$ () 8 7(9: :; $(% 8 '(7:; :?$&$ < $;(%% 8 : /: $ "&: /" /8%(:;)
;8;$; $;>('(; 1 $; ;8 ((; %7:&& 7(8;%8 $"' ?(&$:?&(8 :;< &:$'):':>(% 8
8 7(&$:?$&$ < 97(7($; :; : $8; 8 8; : 8 8 8 7(9$%(: $%$;> 8' 8" 8
8 $; 8;;($8; 9$ 7 7(%8 9: (8 7("%(8 8 7( )(:&$;>% $; 7(%8 9: (1

(@+ + - + * , 8 > - -.
* - % , A , * *1

9 - % * - * , 8 > 1

Table of Contents

1 % + , + 1111111111

List of Tables

1 1 & + 1111111111111111111111111111111111111111111111111111111111111111111111111111111111 B
1B1 ' $'% - 1 1111111111111111111111111111111111111111111111111111111111111111
1C1 ' $' % . 1 1111111111111111111111111111111111111111111111111111111111111111111111
1!1 -0 * - - * 0* . 11111111111111111111 G
1D1 ' . E , * 2, * -- 5 11111111111111111111111111111111111111111
1F1 ' . E , * 2 5 111111111111111111111111111111111111111111111
1 1 / . E , * 1111111111111111111111111111111111111111111111111111111111111111111111
1G1 ' . E , * - : * , 11111111111111111111 H
1 1 ' . E , * 2, * -- 5 111111111111111111111111111111111111 H
1 H1 ' . E , * 2 5 1111111111111111111111111111111111111111
1 1 / . E , * 1111111111111111111111111111111111111111111111111111111111111111111111
1 B1 ' . E , * - : * , 11111111111111111111 B

Chapter 1. X Transport Specification

Introduction

$' * * - , + +
, + - 1 , *
+ - - , * * 1 (+ , + ,
6

* * . , + , $' - + , +
, + - + ** + $' % . 1

* * , , *
+ - . - * *
- - , , + 1

- + - + ** + $' % . 1 $
- , . , + * - 1

+ , + - + , + - + *
. + ** + + + + /0$/)(
% (:' +1 - . - , * - - , --
1 $ -- * * * , +
+ - + - + !1

Initialization

Registering structure to initialize

+ - %9 + ,
+ , + * * 1

```
typedef struct {
      char *transport_name;
      Bool (*config);
} TransportSW;
```

* ,
+ , , +

, I $ ' -/ + 6 : - @ ?

: * , 6

```
TransportSW _XimTransportRec[] = {
/*     char *:
 *     transport_name,     Bool (*config)()
 */
      "X",              _XimXConf,
      "tcp",            _XimTransConf,
```

```
        "local",          _XimTransConf,
        "decnet",         _XimTransConf,
        "streams",        _XimTransConf,
        (char *)NULL,     (Bool (*)())NULL,
};
```

Initialization function

, , + + - + + , , +
1

```
Bool (*config)( im,  *transport_data);
```

% +, $' + -- 1

% +, - +,+ $'
% . -- 1

, + * , + 1

+ , + + $'% . + + *
* + +, - _XimTransportRec J
+,+** , */ + - * A -
, + 1

$, +, - + , - ++ , , +
1 : . (, + , , + ,1

+ */ + , , + 6

```
Bool (*connect)();                /* Open connection */
Bool (*shutdown)();               /* Close connection */
Bool (*write)();                  /* Write data */
Bool (*read)();                   /* Read data */
Bool (*flush)();                  /* Flush data buffer */
Bool (*register_dispatcher)();    /* Register asynchronous data handler */
Bool (*call_dispatcher)();        /* Call dispatcher */
```

, + + - - +** + $'% . 1
, + * + +- - + - 1

The interface/transport layer functions

, + -, , +1

5 5

	!	"
K * +	+ +	C1
K *% -	-	C1B

, I $ ' -/ + 6: -@?

	!	"
K *9		C1C
K * -	-	C1!
K *	,	C1D
K *) +	K- +	C1F
K *) +	+ K- +	C1

/ + + . , + : . (,
+ * 1 * * - , */ * * , +
+ * 1 : . (. - - * * *
/ + & 1

Opening connection

9 XOpenIM + - , , + + - + + $'
% . 1

```
Bool (*connect)( im);
```

% + , $' + -- 1

, + * + + $' % . 1 $, + +
- ++ , , + 1 : . (,
, + 6

```
Bool _XimConnect( im);
```

% + , $' + -- 1

Closing connection

9 XCloseIM + - , , + + - - + + + +
$' % . 1 : . (, , + 6

```
Bool (*shutdown)( im);
```

% + , $' + -- 1

, + * + + + $' % . 1 $, + + + -
++ , , + 1 : . (, , + 6

```
Bool _XimShutdown( im);
```

% + , $' + -- 1

Writing data

, , + + - - - $' % . 1

```
Bool _XimWrite( im, len, data);
```

% + , $' + -- 1

% + , , - 1

% + , - 1

, + $' % . - , + 1 *
, - 1 - - 1 $, - +
+ , , + 1 , I $ ' - / + I ,
+ , - 1 : . (, , + 6

```
Bool _XimWrite( im,  len,  data);
```

% + , $' + -- 1

% + , , - 1

% + , - 1

Reading data

, , + + - , , * $' .
+ 1

```
Bool  _XimRead( im,  read_buf,  buf_len,  *ret_len);
```

% + , $' + -- 1

% + , „ - 1

% + , A ,

% + , , -- 1

, + -- + A + , - 1
A , - 1 , + , - - *
- - + * -1

: . (, , + 6

```
Bool  _XimRead( im,  *ret_len,  buf,  buf_len,  (*predicate)(),
predicate_arg);
```

% + , $' + -- 1

% + , A , „ 1

% + , „ - 1

% + , , 1

% + , - + , $' - 1

% + , - + + , + - 1

- + + - - + , $' 1
, + -- + A + , - 1 A ,
- 1 $, , + 1 $,
+ - + + , + 1

+- - * 6

```
void (*predicate)( im, len, data, predicate_arg);
```

% +, $' + -- 1

% +, A , „ 1

% +, „ - 1

% +, -+ +,+- 1

Flushing buffer

, , + + - - , - 1

```
void (*flush)( im);
```

% +, $' + -- 1

, + * , - - „ 1
$, - , +* - , + 1 : . (,
, + 6

```
void _XimFlush( im);
```

% +, $' + -- 1

Registering asynchronous data handler

- - + , * $' % . 1 + *
, $' - ++ + . 1

- - - - + + -
- + 1 + - + 1
- + * * - + 1 , + *
, * - , + , - + _XimCallDispatcher1

9 * - + - + - L -
, . , + - 1 K- + -+ -
, 6

```
Bool (*register_dispatcher)( im, (*dispatcher)(), call_data);
```

% +, $' + -- 1

% +, - + , + 1

% +, * , 1

- + , + , , 6

```
Bool (*dispatcher)( im, len, data, call_data);
```

% +, $' + -- 1

% +, A , „ 1

% +, „ - 1

% +, * - K- + 1

- + .-- + 1 + - + , .
+ - - , 1 $, - - * 1
* 1

$, - + , + & * -
+ - 1 : . (, , + 6

```
Bool _XimRegisterDispatcher( im, (*dispatcher)(), call_data);
```

% +, $' + -- 1

% +, - + , + 1

% +, * , 1

Calling dispatcher

, , + - + -- + , +
+ , * $' % . . -1

```
Bool (*call_dispatcher)( im, len, data);
```

% +, $' + -- 1

% +, A , „ 1

% +, „ - 1

+ K- + * + - + , + - , 1
- - - K- + 1

. + + - + . + - , , - ** -
, . 1

$ - - -- $' / + +
1 : . (, , + 6

```
Bool _XimCallDispatcher( im, len, call_data);
```

Sample implementations for the Transport Layer

% * * * * , + + - + -
1

X Transport

: , + + , $' * + *
- „ - * + - $' $' % .

+ - $' % . @+ $'
' . -9 - / 1$, - + -
, - I+ + ** + - I -
- - + - $' % . , - I$'% + ** +
- I1

Connection

$ - + + + ** + - + -1 : '
, . E , * - , $'K%((
+ + $' % . + -1

, I $ ' - / + I , $'K%((*1

5 # 5 $ % & %" ' ('

" %	$
	'
K	% 9 - % *
? -K .	% 9 - % *
) M-	- + + +
9 - -	$'% 9 - $)
: * * K	$: *2- IK $'K 8;;(I , 5
, *	CB
- 1 NHO	+ + ** + - $)
- 1 N O	+ * . 2M 5
- 1 NBO	+ * . 2M 5

$ - + + 2 , $' % . + ** +
- 5 $' % . - ' , . E , *
+ + ** + - 1

5) 5 $ % & % " *

" %	$
	'
K	% 9 - % *
? -K .	% 9 - % *
) M-	- + + +
9 - -	$'% 9 - $)
: * * K	$: *2- IK $'K 8;;(I , 5
, *	CB
- 1 NHO	+ + ** + - $)
- 1 N O	+ * . 2M 5
- 1 NBO	+ * . 2M 5
- 1 NCO	- . - A ' - / 2MB5

2M 5 * 0* .

-0 * - -+- - +* ,* 0* .
, 6

5 +5 (,' ((- ,
*****

*		(,'
-		
H	H	'P/ '
		'P* '
	B	'P* 'P/ '
	H	/ ; ,
B	H	'P/ ; ,
		'P* 'P/ ; ,

```
only-CM             :    data is sent via a ClientMessage
multi-CM            :    data is sent via multiple ClientMessages
Property-with-CM    :    data is written in Property, and its Atom
                         is send via ClientMessage
PropertyNotify      :    data is written in Property, and its Atom
                         is send via PropertyNotify
```

* - -+- * 0* . , 6

Q + - H * 0* . $' % . 1 +
* * - ! C1 + * - *
* 0* . * - . , 1

Q $' % . - * 0* . * + 1
+ - - * - +, - $' % . 1

Q $, * 0* . * . - - H1

2MB5 - . - A ' - /

$, - . , * ' - / +, - . - A
' - / 1 - + * A .
* '2 '5 - - + A . /
1

read/write

- , - . ' 9 - / 9
- % *1

Format for the data from the Client to the IM Server

$ % &

$, - . ' . , * , 6

5 . 5 $ % & * / ((

" %	$
	'
K	% 9 - % *
? -K .	% 9 - % *
) M-	- + + +
9 - -	$'% 9 - $)
: * * K	$: *2- IK $'K'8 (): : I 5
, *	G
+ - 1 NBHO	2 -0): : 6 BH 5

5 0 5 $ % & * /

" %	$
	'
K	% 9 - % *
? -K .	% 9 - % *
) M-	- + + +
9 - -	$'% 9 - $)
: * * K	$: *2- IK $'K/ 8 8 8&I 5
, *	G
+ - 1 NBHO	2 -0): : 6 ': BH 5

$, - * BH - . - * H1

!

$ + , - - . 9 - / , ,, + +1
, * - , / - .
- + - - + * - -1

Q / , + - - + + ** +
- -: * , -- , - $' % . . '
. 1

Q / , + - - + + ** +
- -: * , -- , - $' % . . / ; ,
. 1

* , / , 6

5 1 5 $ & ! * /

&	$
) M-	- + + +
9 - -	$'% + ** + - $)

&	$
: *	-0 : * 2M 5
, *	G
* -	/ ' - : -
K+ M-	-0): :
*	,): :

2M 5 -0 : 8' + , XInternAtom1
IK+ I

+ + * - , / ' - : - - $'
% . - - * - 1 1 2- R 51

$, : * , - . ' . , * , '
, 6

5 2 5 $ % & * / (

" %	$
	'
K	% 9 - % *
? -K .	% 9 - % *
) M-	- + + +
9 - -	$'% 9 - $)
: * * K	$: *2- IK $'K/ 8 8 8&I 5
, *	G
- 1 NHO	, -0 : *
- 1 N O	-0 : *

Format for the data from the IM Server to the Client

$ % &

, * , ' , 6

5 3 5 $ % & * / ((

" %	$
	'
K	% 9 - % *
? -K .	% 9 - % *
) M-	- + + +
9 - -	$'% 9 - $)
: * * K	$: *2- IK $'K'8 (): :I 5
, *	G

" %	$
+ - 1 NBHO	2 -0): : 6 BH 5

5 4 5 $ % & * /

" %	$
	'
K	% 9 - % *
? -K .	% 9 - % *
) M-	- + + +
9 - -	$'% 9 - $)
: * * K	$: *2- IK $'K/ 8 8 8&I 5
, *	G
+ - 1 NBHO	2 -0): : 6 ': BH 5 2M 5

2M 5 $, - A * BH - . - *
H1

!

$ + , - - . 9 - / , ,, + +1
, * - , / - .
- + - - + * - -1

Q / , + - - $'% + ** +
- - : * , . ' . 1

Q / , + - - $'% + ** +
- - : * , . / ; , . 1

* , / , 6

5 5 $ &! * /

&	$
) M-	- + + +
9 - -	$'% + ** + - $)
: *	-0 : * 2M 5
, *	G
* -	/ ' - : -
K+ M-	-0): :
*	,): :

2M 5 -0 : 8' + * + + -
+ XInternAtom1

$' % . + * - , / ' - : - -
+ - - * - 1 1 2- R 51

$, : * , - . ' . , * , '
, 6

5 # 5 $ % & * / (

" %	$
	'
K	% 9 - % *
? -K .	% 9 - % *
) M-	- + + +
9 - -	$'% 9 - $)
: * * K	$: *2- IK $'K/ 8 8 8&I 5
, *	G
- 1 NHO	, -0 : *
- 1 N O	-0 : *

Closing Connection

$, + - + + $' % . - , + - , + *
* + - - +11

References

N O ' ; - 7 - 7