X Window System Extensions

XI034

Patrick Mahan

TGV Window Washer

mahan@tgv.com

TGV, Incorporated

10:00 AM, June 8, 1993

X Extensions

Spring DECUS Symposium 1993

- What are X Extensions
- Known X Extensions
- X Extensions supported by Digital

X Extensions

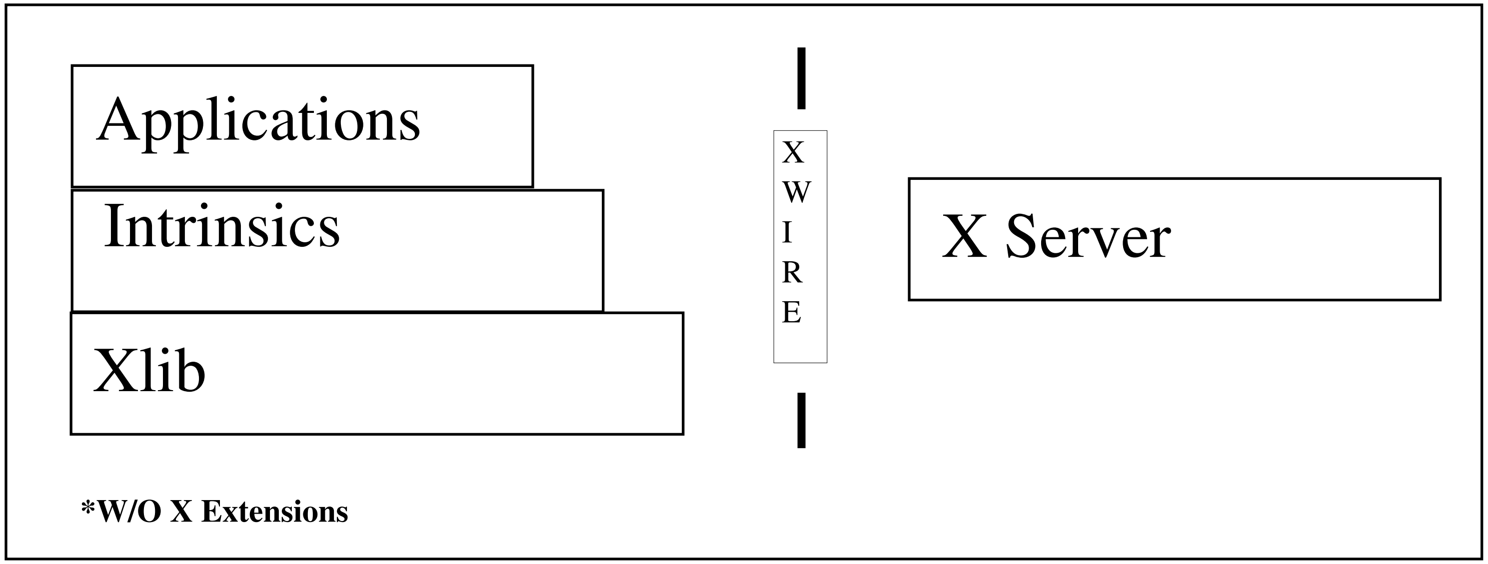
Spring DECUS Symposium 1993

- X Protocol designed with Extensiblity in Mind
- Support provided starting with X11R4
- First extensions also provided with X11R4

X Extensions

Spring DECUS Symposium 1993

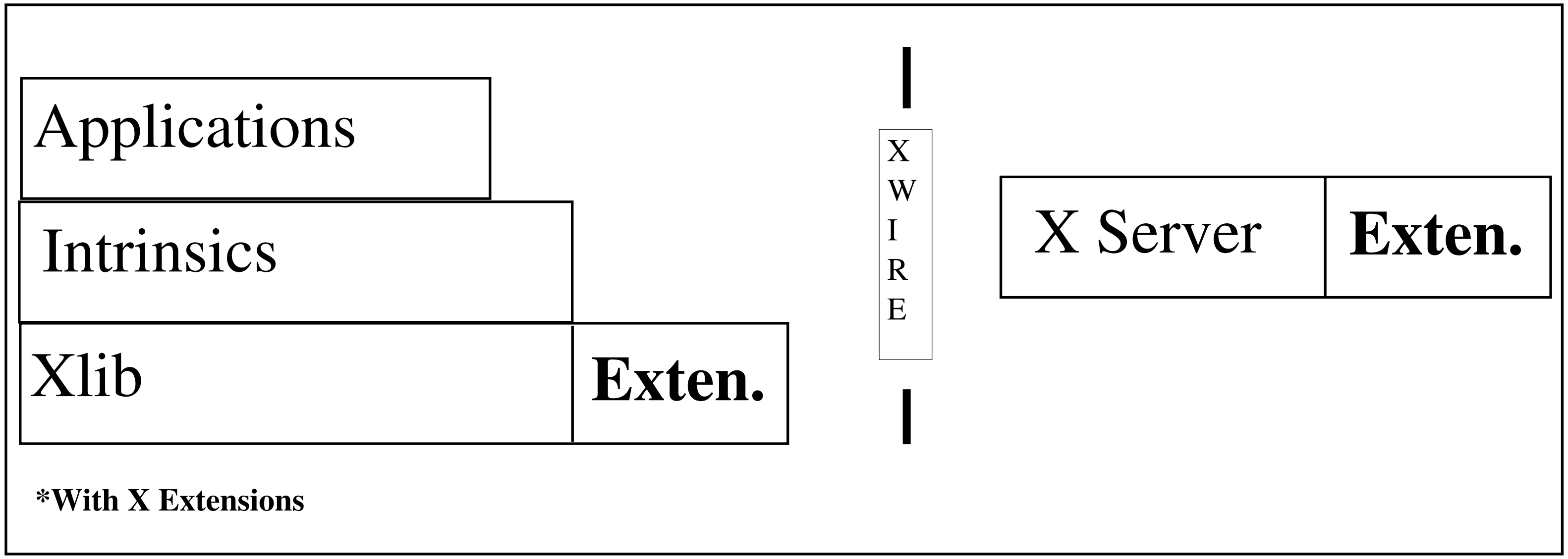
- ## Architecture Overview



X Extensions

Spring DECUS Symposium 1993

- **Architecture Overview**



❑ Available X extensions

- Public
  - SHAPE
  - X Input
  - Multi-Buffering
  - PEX
- Private
  - Adobe DPS
  - DEC XTRAP
  - Xie*

* Currently submitted to X Consortium

Listing Extensions

- XListExtensions()*
- xdpyinfo*

Querying Extensions

- XQueryExtension()*

* Requires X11R4

X Extensions
Spring DECUS Symposium 1993

```
/*
 *   Example program to List Extensions
 */
#include <stdio.h>
#include <X11/Xlib.h>
main(argc, argv)
int  argc;
char *argv[];
{
        int n = 0;
```

```
char **extensions;
Display *dpy;
/*
 * Open a connection to server
 */
dpy = XOpenDisplay(NULL);
/*
 * Get the supported extensions List
 */
extensions = XListExtensions(dpy, &n);
/*
 * Printout what we found
```

```
 */
if (n != 0) {
        register int i;
        fprintf(stdout, "Number of Extensions found: %d\n", n);
        for (i = 0; i < n; i++)
                fprintf(stdout, "\t\"%s\"\n", extensions[i]);
        /*
         * Free the List
         */
        XFreeExtensionList(extensions);
}
}
```

❑ XListExtensions Example

- XListExtensions() returns extensions as an array of NULL terminated strings including the number returned
- The array is allocated by Xlib and is freed by calling XFreeExtensionList()

X Extensions
Spring DECUS Symposium 1993

```
/*
 *   Example program to Query an Extension
 *
 *   NOTE: Extension names ARE case sensitive
 */
#include <stdio.h>
#include <X11/Xlib.h>
```

```
static char *Extension="SHAPE";




main(argc, argv)
int  argc;
char *argv[];
{
        int major_opcode, first_event, first_error;
        Display *dpy;
        /*
         *      Open a connection to server
         */
```

```
dpy = XOpenDisplay(NULL);
/*
 *      Get the requested extension
 */
if (XQueryExtension(dpy, Extension, &major_opcode,
                    &first_event, &first_error)) {
    fprintf(stdout, "\"%s\" Extension exists!\n", Extension);
    if (major_opcode)
      fprintf(stdout, "Major Opcode: 0x%X ", major_opcode);
    if (first_event)
      fprintf(stdout, "Base Event Code: 0x%X ", first_event);
    if (first_error)
```

```
            fprintf(stdout, "Base Error Code: 0x%X ", first_error);
        fprintf(stdout, "\n");
    } else
        fprintf(stdout, "\"%s\" Extension doesn't exist :-(\n",
            Extension);
}
```

XQueryExtension Example

- XQueryExtension() returns a 1 if extension is found
- If an extension is found the following information maybe returned
  - Major Opcode (assigned by X Server)
  - First Event Code (assigned by extension)
  - First Error Code (assigned by extension)

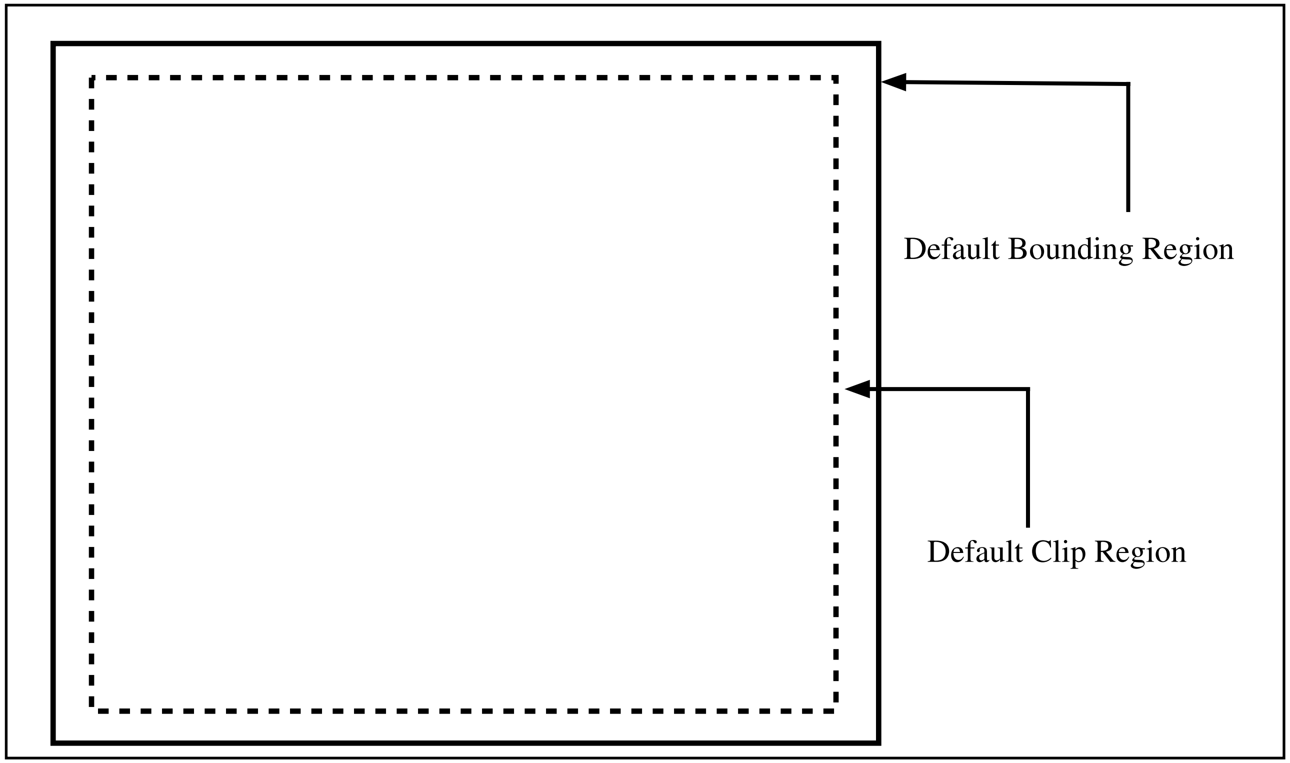
Common X Extensions

- SHAPE
- X Input
- Multi-Buffering
- PEX

SHAPE Extension

- Allows programmer to create a non-rectangular window
- Based on the fact that a window is defined by two regions
  - Default Bounding Region
  - Default Clip Region

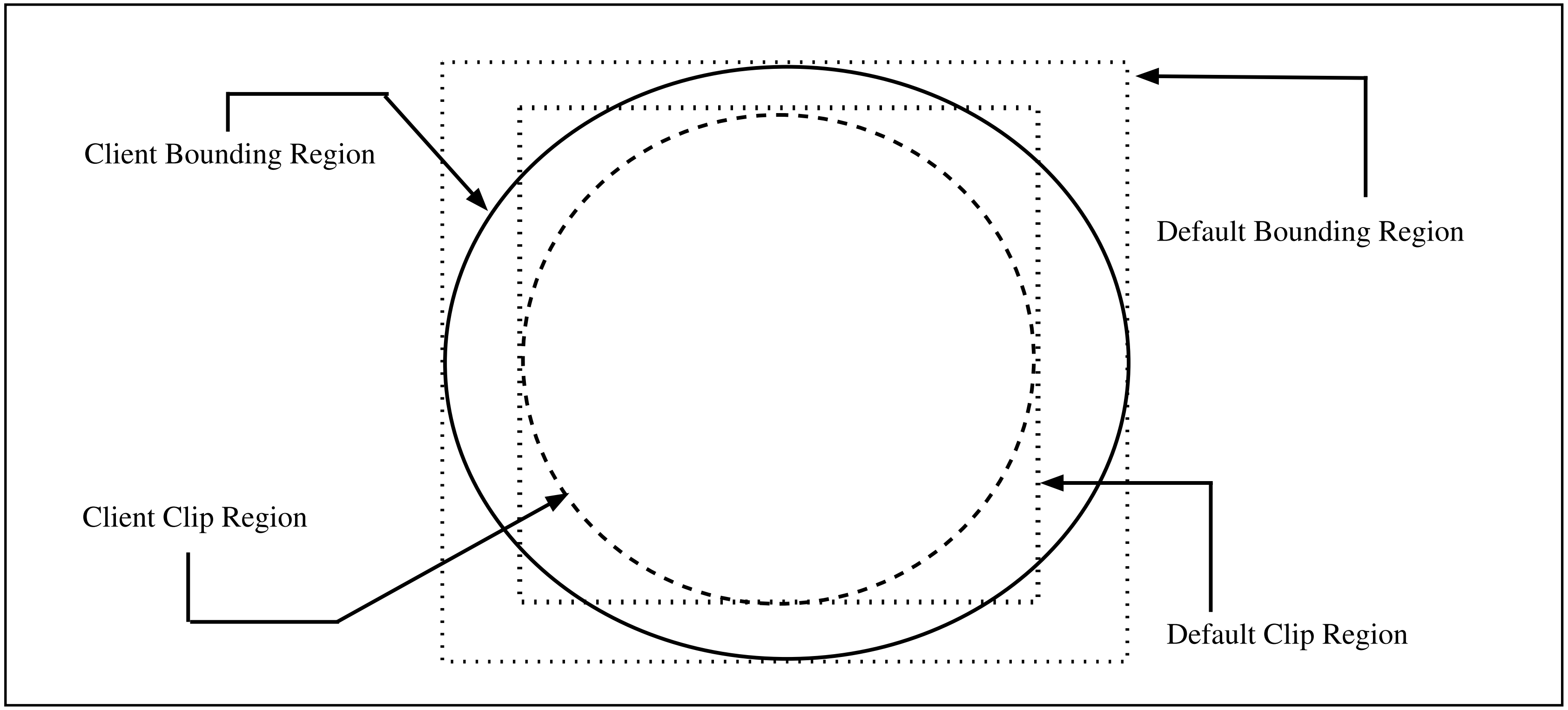
- ## SHAPE Extension



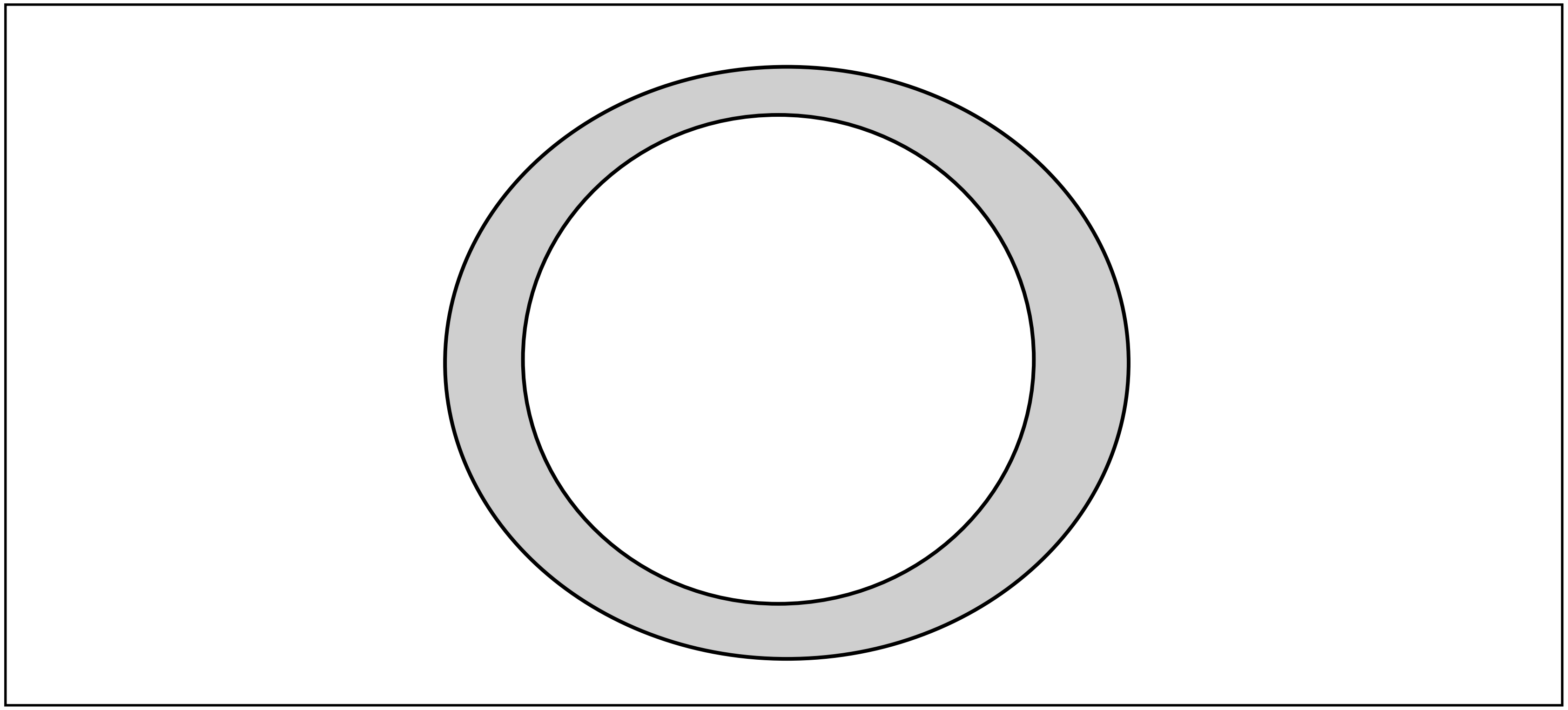
SHAPE Extension

- Introduces two new regions
  - Client Bounding Region
  - Client Clip Region

- **SHAPE Extension**



❑ SHAPE Extension



SHAPE Extension

- Allows different methods of specifing shape
  - a list of rectangles
  - a bitmap
  - existing bounding or clip region from the window
- Digital currently supports the SHAPE extension only on the Client side for OpenVMS systems and on both the Client and Server for OpenVMS AXP systems

❑ X Input Extension

- ♦ Added as a draft standard with X11R4
- ♦ Created to service other input methods than the standard keyboard and mouse
- ♦ Currently allows for the use of dial/button boxes or sketch pads
- ♦ Became a Consortium Standard with the release of X11R5
- ♦ Digital only provides support for the Client

❑ Multi-Buffering Extension

- ♦ Grew out of a need for operations that must be performed in a given time frame
- ♦ Clients needed a method to produce smooth animation
- ♦ Server uses an approximate real-time model to implement the buffering

Multi-Buffering Extension

- Clients create a series of buffers (usually pixmaps) containing the data needed for display
- Clients can specify a minimum and a maximum time allowed between consecutive buffers
- Digital provides support for Multi-Buffering in both the Client and Server on all architectures

❑ PHIGS Extension

- ♦ Also known as PEX
- ♦ Currently a Consortium draft standard
- ♦ Developed primarily
  - Provide 3-D graphics for X
  - Provide a migration environment for a large number of PHIGS base applications
- ♦ Provides three methods of operation
  - Server based database (Batch mode)
  - Client based database (Immediate mode)
  - Shared database between Client and Server*

* Digital is the only vendor to currently support this option

- **Other MIT provided X extensions (Not supported by Digital)**
  - MIT Shared Memory Extension
    - Provides supported method of allowing a Client and Server to share memory*
  - MIT Miscellaneous extension
    - Used by the X Development staff for minor changes and extensions
    - Currently only used to enable/disable server bug compatibility

X Extension Support Under DECwindows/Motif

- Client library support available with DECwindows/Motif 1.1
- No support for MIT Shared memory and only supports MIT Miscellaneous on OpenVMS AXP
- OpenVMS X Server (X11R4 based) currently only supports the following extensions
  - DEC-Server-Mgmt-Extension
  - ServerManagementExtension

- Adobe-DPS-Extension
- X3D_PEX
- Xie
- DEC-XTRAP
- Multi-Buffering

♦ OpenVMS AXP X Server (X11R5 based) currently only supports the following extensions
  - DEC-Server-Mgmt-Extension
  - ServerManagementExtension
  - SHAPE
  - MIT-SUNDRY-NONSTANDARD
  - Xie
  - DEC-XTRAP

♦ Client side is provided through the Run-Time Library SYS$SHARE:DECW$XEXTLIBSHR.EXE

❑ Available documentation

- All documentation is currently only available from the X Consortium
  - SHAPE Extension: X11 Nonrectangular Window Shape Extension, v1.0 by Keith Packard
  - X Input Extension: X11 Input Extension Protocol Specification, V1.0 and X11 Input Extension Library Specification, v1.0 by Mark Patrick and George Sachs
  - Multi-Buffering Extension: Extending X for Double-Buffering, Multi-Buffering and Stereo, v3.2 by Jeffrey Friedberg, Larry Seller and Jeff Vroom
  - PHIGS Extension: PEX Protocol Specification edited by Sally Barry

❑ Example Programs

- ♦ XDPYINFO* (List of available extensions)
- ♦ ICO* (Usage of Multi-Buffering)
- ♦ XEYES* (Usage of SHAPE Extension)

* Available from DECUS Library as part of the X11R4 Clients and Demos v2.0

- **Questions and Answers**