What is a computer program?

“A combination of computer instructions and data definitions that enable computer hardware to perform computational or control functions”

- Instructions
- Data
- What is missing here? User Interface!

![Diagram of computer operations]

- Memory access
- CPU resource
- Disk I/O
- Network I/O
- …
How Do You Know It is Chicken that You are Eating?

- Answer 1:: hmm, the machine says it is chicken
- Answer 2:: well, it tastes like chicken

Similarly, how can an end-user tell if a program is running locally or remotely; whether the data is fetched from a remote location or on a hard disk?

- Assuming, resources are abundant
- E.g. Unix domain socket for inter-process communication
Transparency to Mobility

Mobility of hardware, data and software in computer applications

- Network-centric
  - Link layer handoff
    - Example?
  - Routing (mobile IP)
  - Transport layer migration
    Connectivity between client host machine to the rest of world

- User-centric
  - Mobile agent
  - Thin client approaches
The “True” Transparency

- Decouple computing session from computing devices
  - End-user devices, be it a window machine, Linux box, PDA etc.
    - Just provide a user interface
  - Hosting server (where the computation is carried out)
    - No necessarily restricted to a physical entity
    - A close example would be computing cluster

- Benefits
  - High availability and reliable application services
  - On-demand access to application/computational resources
  - Simple end-user devices may bridge the gap between haves and have-nots

Question: adaptation vs. transparency?
Key elements

- **Display virtualization**
  - Display user interface remotely

- **Process migration**
  - Session can be check-pointed, move from one computer to another
  - Examples app. Load balancing, system maintenance
  - But, only among the same operating systems

- **Persistent connectivity**
  - Process can have access to network I/O “independently”
  - Continuous operations even when the physical point of attachment changes
  - e.g., two HTTP sessions both on port 80
Display Virtualization

- **Application**
  - Operation and management of graphical elements
  - e.g. window manager

- **Middleware**
  - A hardware-independent abstraction of the display hardware to meet the requirements of the display system and its application
  - e.g. XFree86

- **Hardware**
  - Frame buffer

- Open a new browser window

- Draw a bunch of boxes, circles, text using certain font

- Display raw pixel data at a given location & size
Example Implementation

- Virtual network computing (VNC)
  - Due to ATT Cambridge Labs (ORL)
  - Primitive: *Put a rectangle of pixel data at a given x, y position*
  - Also support adaptive update
  - Support raw encoding, jpeg encoding
  - No state maintained at client -- this is in contrast to X-Window systems
  - Client pull model
MobiDesk: Mobile Virtual Desktop Computing

- A proxy based approach

![Diagram of MobiDesk Architecture]
Architecture Overview

- Layer 7 proxy exposes a single entry point to clients
  - Proxy handles client requests and dispatch to backend application server
  - Back-end application servers host virtualized environment
  - Storage server provides persistent file storage
  - Users access a completely private and mobile environment using a thin-client session viewer

- MobiDesk virtualizes three resources
  - Display
  - Operating system
  - Network
Display Virtualization in MobiDesk

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RAW</td>
<td>Display raw pixel data at a given location and size</td>
</tr>
<tr>
<td>COPY</td>
<td>Copy frame buffer area to specified coordinates</td>
</tr>
<tr>
<td>SFILL</td>
<td>Fill an area with a given pixel color value</td>
</tr>
<tr>
<td>BITMAP</td>
<td>Fill a region using a bitmap image</td>
</tr>
<tr>
<td>PFILL</td>
<td>Tile a pixmap rectangle in a given region</td>
</tr>
</tbody>
</table>

- Provide a virtual video driver
Display Virtualization

- Client hardware resources are exported to the server and used if possible to reduce latency
  - Direct video support
  - Cursor drawing support
  - Resizing to client’s display

- Latency reduction techniques
  - Server push
  - Display command scheduling: shortest remaining size first

- All client state is soft; all session state is stored at the respective session server
  - Client’s state destroyed upon disconnect
  - Allow multiple clients for the same session
Comparison between VNC & MobiDesk

Video Quality

Data size