Virtual machines

benefits

• multiple flavors - run a different OS in each VM
  – provide for user preference
  – legacy OS/application (run one VM with old OS/app to avoid porting)

• fault isolation / security
  – OS in one VM crashes without taking the whole system down
Virtual machines

benefits

• development/testing
  – develop/test in one VM without affecting others

• migration - move a VM from one physical server to another
  – load balancing
  – fault recovery
Virtual machines

approaches

- bare metals approach

<table>
<thead>
<tr>
<th>user apps</th>
<th>user apps</th>
<th>user apps</th>
<th>user apps</th>
</tr>
</thead>
<tbody>
<tr>
<td>client OS</td>
<td>client OS</td>
<td>client OS</td>
<td>client OS</td>
</tr>
<tr>
<td>VM</td>
<td>VM</td>
<td>VM</td>
<td>VM</td>
</tr>
</tbody>
</table>

Virtual machine monitor (hypervisor)

hardware
Virtual machines

approaches

- hosted approach

| user apps | user apps |
| client OS | client OS |
| VM monitor | user app ... user app |
| hose OS |
| hardware |

also:

- dynamic recompilation of user and target OS code
- modify client OS to work with VMM (paravirtualization)
Virtual machines

VM should mimic I/O, interrupts, etc. - but should run w/ limited privileges

• software simulate the actual machine => slow
• run directly on hardware but each privileged operation is trapped and simulated
• code VM with special assists
Virtual machines

history

• 1964 - work starts on IBM CP-40 (control program for S/360 Model 40)
• 1966 - IBM CP-67 (control program for S/360 Model 67, virtual memory)
• 1972 - IBM VM/370 for S/370 computers
• 1988 - SoftPC for Apple and Sun systems
• 1999 - VMware
Virtual machines

history (cont'd)

• today - Microsoft Virtual PC, VMware, and Xen are but three among many
• AMD provides AMD-V and Intel provides Intel-VT
• (will allow Xen to host Windows)