Virtualization

CPU
- Mech
- Policies

Memory
- Mech
- Policies

Abstraction:
- Process
- "Address Space"

Address Space

P1

Stack
- heap
- s. data
- code

n-1 higher addr.
1. code
2. heap
3. stack.

lower addr
Assumptions:

1. Process AS is contiguous in phy. mem.
2. A AS of a process < phy. mem.
3. All processes have the same size of AS.
void foo() {
    int x = 100;
    x = x + 10;
}

128: movl 0x1004, %eax
132: addl $10, %eax
135: movl %eax, 0x1004

<table>
<thead>
<tr>
<th>VA</th>
<th>PA</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>32K+128</td>
</tr>
<tr>
<td>20000</td>
<td>32K+20000</td>
</tr>
<tr>
<td>132</td>
<td>32K+132</td>
</tr>
<tr>
<td>135</td>
<td>u+135</td>
</tr>
<tr>
<td>20000</td>
<td>u+20000</td>
</tr>
</tbody>
</table>

```c
void foo() {
    int x = 100;
    x = x + 10;
}
```
1. $1024 < 16k$

2. $PA = 32k + 1024$

$VA: \ 1024$

$PA = Base + VA$

Base and Bounds Relocation

Dynamic

HW-based.
\[ P_1 \quad \text{\(X = 0x1000\)} \]

\[ P_2 \quad \text{\(0x300\)} \]

\[ \text{code} \quad \text{\(P_2\)} \]

\[ \text{stack} \quad \text{\(P_1\)} \]

\[ \text{waste} \quad \text{\(N_1\)} \]

\[ \text{OS} \quad \text{\(0x1000\)} \]

\[ \text{Phy. mem.} \]

\[ \text{internal fragmentation} \]

\[ P_1 - 4KB \]

\[ P_2 - 2KB \]

\[ P_3 - 4KB \]

\[ \rightarrow 10KB \]
<table>
<thead>
<tr>
<th>Base</th>
<th>Bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>28K</td>
</tr>
<tr>
<td></td>
<td>4K</td>
</tr>
<tr>
<td>Heap</td>
<td>48K</td>
</tr>
<tr>
<td></td>
<td>2K</td>
</tr>
<tr>
<td>Stack</td>
<td>20K</td>
</tr>
<tr>
<td></td>
<td>2K</td>
</tr>
</tbody>
</table>

**Code**

\[ \text{V.A.: } 1024 = 1K \text{ (Code)} \]

\[ 28K + 1K = 29K \]

**Heap**

\[ \text{V.A.: } 4200 \text{ (Heap)} \]

\[ \text{offset} = 4200 - 4096 \]

\[ = 104 \]

\[ \text{P.A.} \]

\[ 48K + 104 \]
Size (V.A. Space) = 16KB = 16 \times 2^{10} \text{ bytes} = 4 \times 2^{10} = 2^{14} \text{ bytes}

V.A.: 00 01 00 00

1024 = 0x0400

4 = 2^{2} \text{ bytes}

2 MSBs | Seg
---|---
00 | Code
01 | Heap
11 | Stack

16

4200

2062 - 8

16 16 - 6

1 - 0

0 0 0 0 0 0 1 1 0 1 0 0 0 1 0

0 6

4200 = 0x1068

16KB
12KB
8KB
4KB

Stack
Heap
Code
Unused
$$4\text{ GB} = 4 \times 2^{30} = 2^{32}$$

Stack

\[
\begin{array}{c}
1 & 1 \\
\hline
2^8 & 2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\
\hline
& 8 & 4 & 2 & 1 \\
\end{array}
\]

\[
\begin{array}{c}
1 & 1 & 0 & 0 \\
\hline
2^8 & 2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\
\hline
& 0 & 0 & 0 & 0 \\
\end{array}
\]

\[
\begin{array}{c}
1 & 1 & 0 & 0 \\
\hline
2^8 & 2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\
\hline
& 0 & 0 & 0 & 0 \\
\end{array}
\]

$$15\text{ KB} = 0 \times \text{3C00}$$

$$\text{PA} = \text{Base} - \text{max size of stack} + \text{initial offset}$$

$$= 20\text{ K} - 4\text{ K} + 3\text{ K}$$

$$= 20\text{ K} - 1\text{ K}$$

$$= 19\text{ K}$$
<table>
<thead>
<tr>
<th>Seg</th>
<th>Base</th>
<th>Bounds</th>
<th>Growing up?</th>
<th>Perm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Code</td>
<td>1</td>
<td>YX</td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>1</td>
<td>YN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>0</td>
<td>YW</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$4GB \rightarrow 32 \text{ bits } V.A.$

Page size $= 4KB$

$$\text{# pages} = \frac{4GB}{4KB} = \frac{4 \times 2^{30}}{4 \times 2^{10}} = 2^{20} = 1 \text{ million V. Pages}.$$
$P_2 \cap C = 8 \text{KB}$
\( V_{A0} = 10 \)

16 bytes = 2^4 bytes

\[
\begin{array}{c}
\text{VPN} \\
1 & 0 \\
8 & 4 & 2 & 1
\end{array}
\]

Page size = 4 bytes = 2^2 bytes

\( \text{VPN} = 2 \)

Addr Trans (using P.T.)

\( \text{PFN} = 12 \)

\[
\begin{array}{c}
P \cdot A = \\
1 & 1 & 0 & 0 & \underline{1} & 0
\end{array}
\]

PFN (4 bits)

\[
\begin{array}{c}
8 & 4 & 2 & 1
\end{array}
\]

Offset