L4: Naming systems

Frans Kaashoek

6.033 Spring 2011

http://web.mit.edu/6.033

```java
public interface BidInterface extends Remote {
    public String bid(String, Integer) throws RemoteException;
}
```

```java
public static void main (String[] argv) {
    try {
        BidInterface srvr = (BidInterface)
            Naming.lookup("//xxx.ebay.com/Bid");
        winning = srvr.bid("123", 10);
        System.out.println(winning);
    } catch (Exception e) {
        System.out.println ("BidClient exception: " + e);
    }
  }
```

Names

- LD R0, 0x2020
- 18.7.22.69
- web.mit.edu
- /mit/6.033/www/foo.c
- http://web.mit.edu/6.033
- 6.033-staff@mit.edu
- amsterdam
- foo.c
- .. (as in cd .. or ls ..)
- wc
- (617)253-7149, x37149
- Frans Kaashoek
- 021-84-2030

The "dig" Command

dig csail.mit.edu

"Time to Live" (TTL)

Type of record
A = "Address"
NS = "Name server"

```
;; QUESTION SECTION:
;csail.mit.edu.                         IN      A        The question we asked

;; ANSWER SECTION:
csail.mit.edu.                 5921     IN      A        128.30.2.121  The answer

;; AUTHORITY SECTION:
csail.mit.edu.                 15152    IN      NS       AUTH-
    NS0.csail.mit.edu.
csail.mit.edu.                 15152    IN      NS       AUTH-
    NS1.csail.mit.edu.
csail.mit.edu.                 15152    IN      NS       AUTH-
    NS2.csail.mit.edu.
csail.mit.edu.                 15152    IN      NS       AUTH-
    NS3.csail.mit.edu.
```

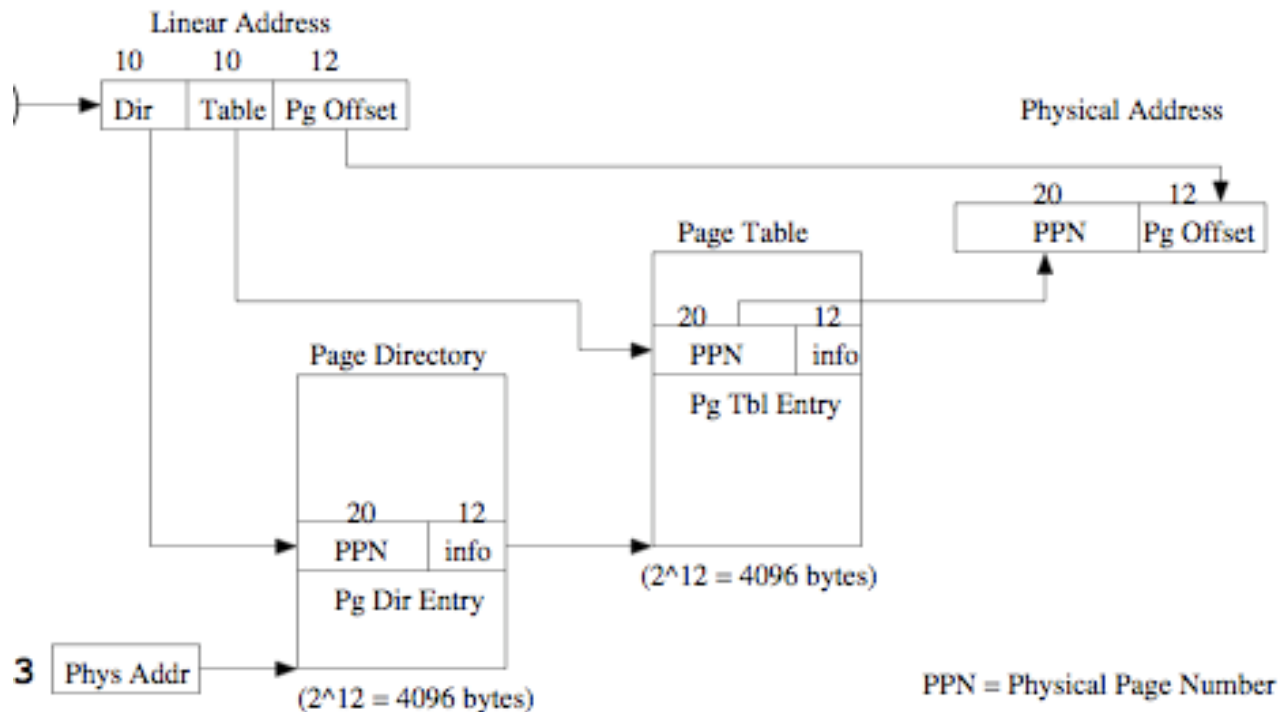*Servers that can answer the question*
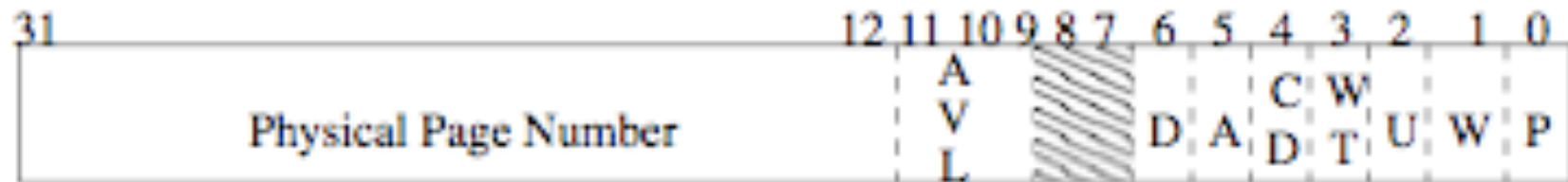
Intel x86-32 two-level page table

- Page size is 4,096 bytes
  - 1,048,576 pages in 2^{32}
  - Two-level structure to translate

x86 page table entry


```
31                                    12 11 10 9 8 7  6  5  4  3  2  1  0
┌──────────────────────────────────────┬─┬──────┬──┬──┬─┬─┬──┬──┬──┐
│                                      │A│//////│  │  │C│W│  │  │  │
│     Physical Page Number             │V│//////│D │A │D│T │U │W │P │
│                                      │L│//////│  │  │ │ │  │  │  │
└──────────────────────────────────────┴─┴──────┴──┴──┴─┴─┴──┴──┴──┘
```

- ## W: writable?
  - Page fault when W = 0 and writing
- ## U: user mode references allowed?
  - Page fault when U = 0 and user references address
- ## P: present?
  - Page fault when P = 0

Enforcing modularity with page tables
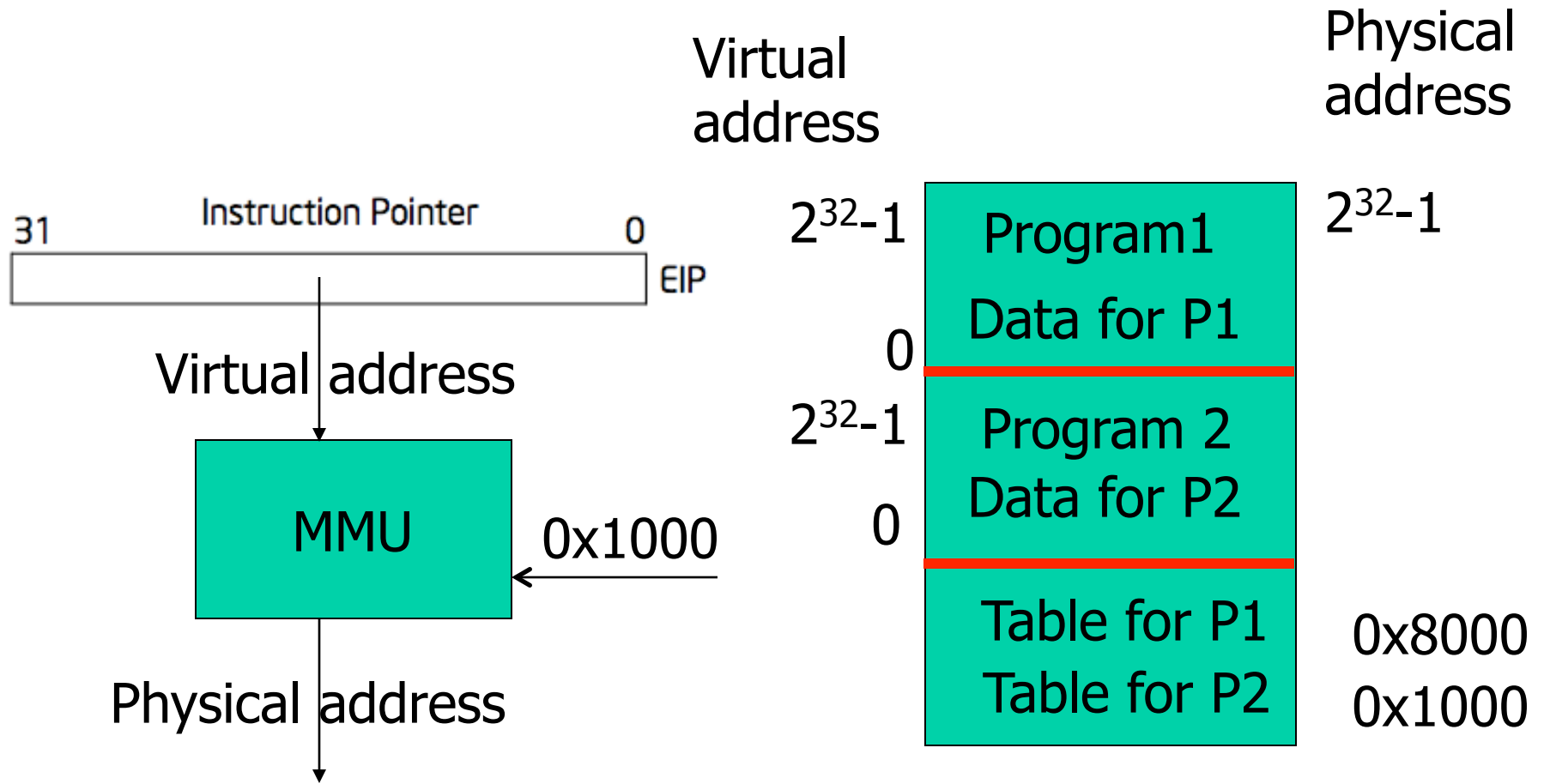
- Each running program has its own page table

Summary

- Naming systems are the glue to connect modules

- Naming systems are every where

- Understanding a naming system:
  - What is the syntax for name?
  - What are the values?
  - What is the naming resolution algorithm?
  - Where does a name's context from?