

INFORMATION

ASSIGNMENT # 1 is available at the
WEB SITE OF THE COURSE + 2 CHAPTERS
(SEE NEWSGROUP)
covering - Basics of
CORBA
- Adaptor

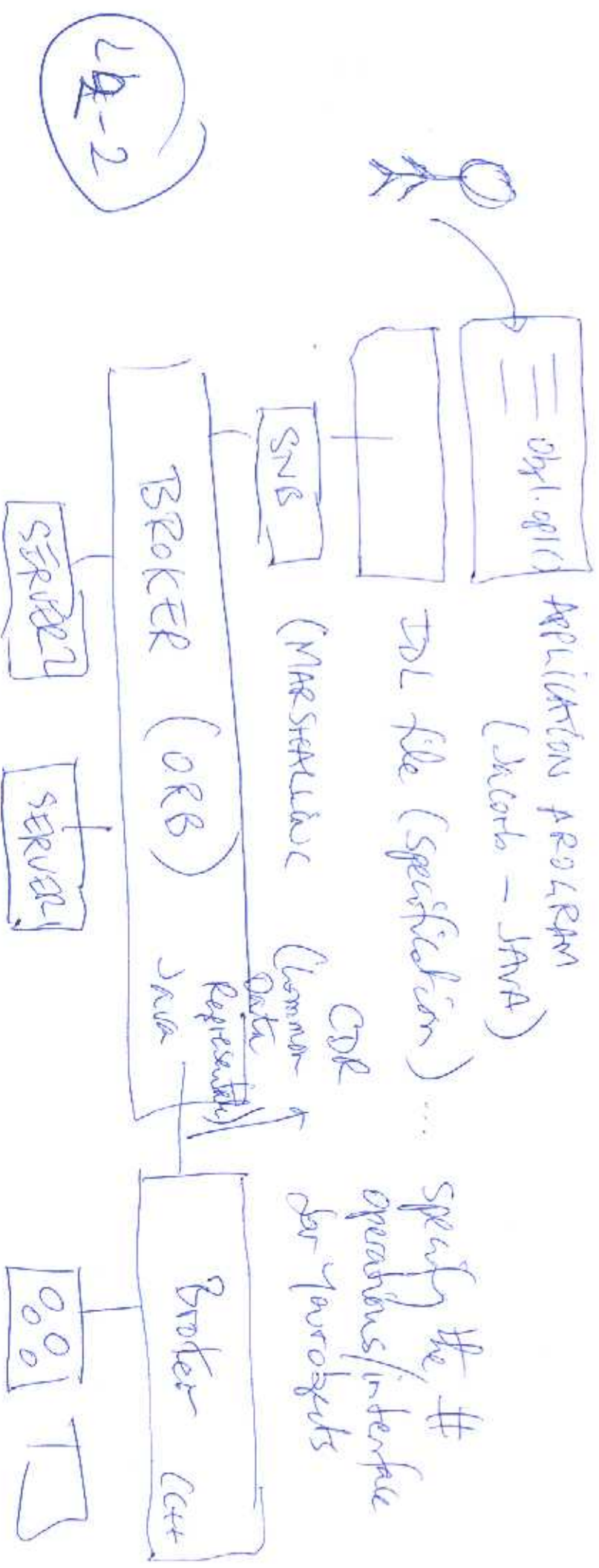
READ NEWSGROUP (as often as you can !!!)
rmit.cs, Adv Client Server Architecture

E-mail James Broberg (Tutor) for consultation /
information

jbroberg@cs.mit.edu

LAST WEEK

BASICS OF CORBA
"SIMPLE" BINDING (of client requests to server objects)



Types

SIMPLE TYPES: String, Integer

COMPLEX TYPES:

- tuple type []
- set type { }
set

tuple constructor

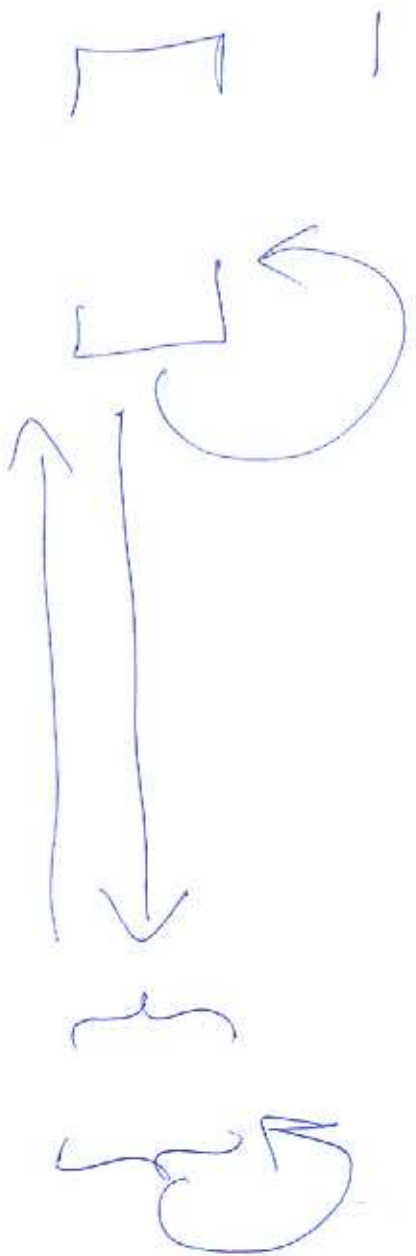
Person1: [Name: string; age: integer]

Person2: [Name: string; address: [code: integer; street: string]]

{ Person1 }
tuple type

Person3: [Name: string; age: integer; address: string] (2-3)

ORTHOGONALITY



[] : you define "tuples of tuples"
current

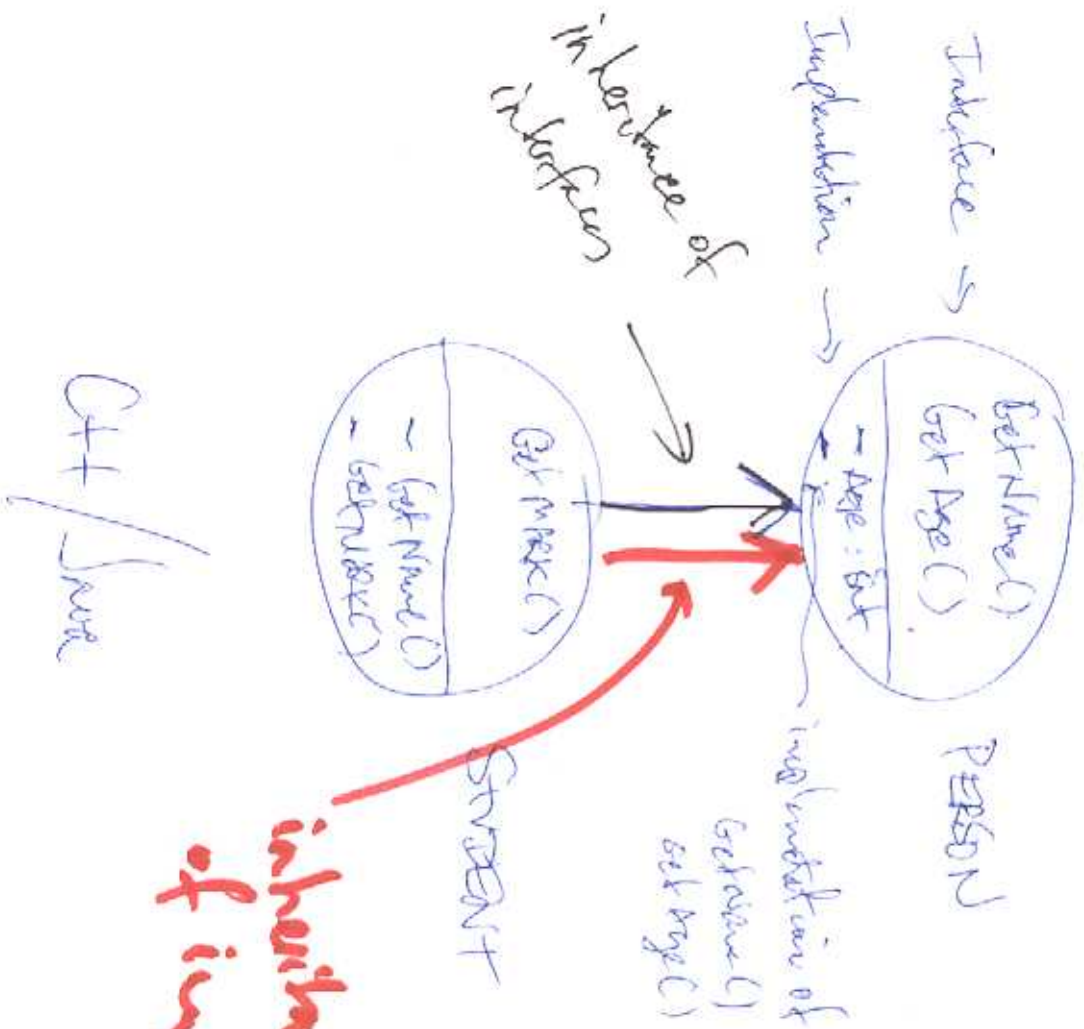
SET : "Set" of objects
where there is no ordering

List [...]
Set { ... }

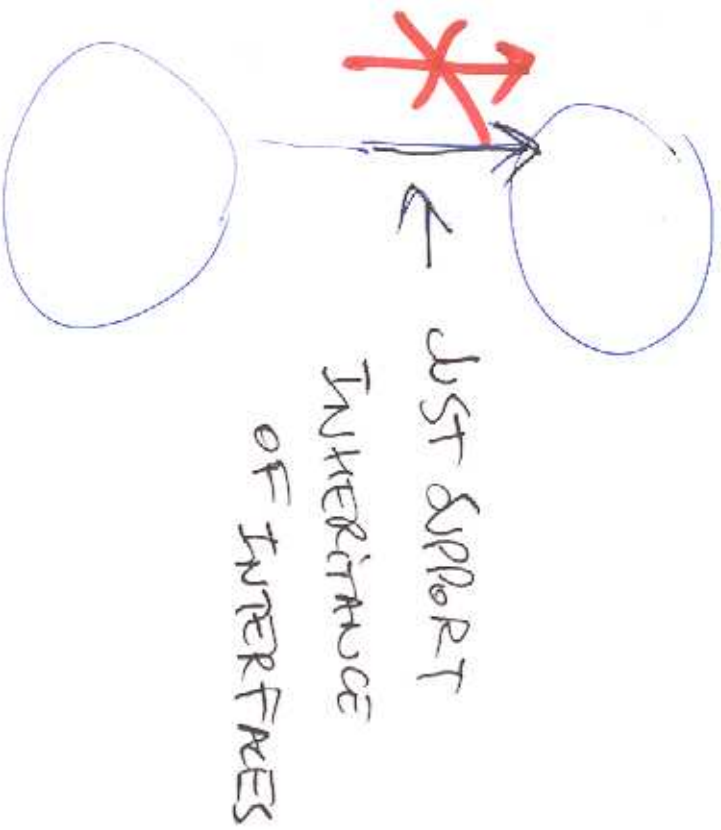
12-4

INHERITANCE

"DISTRIBUTED OBJECT MODEL"

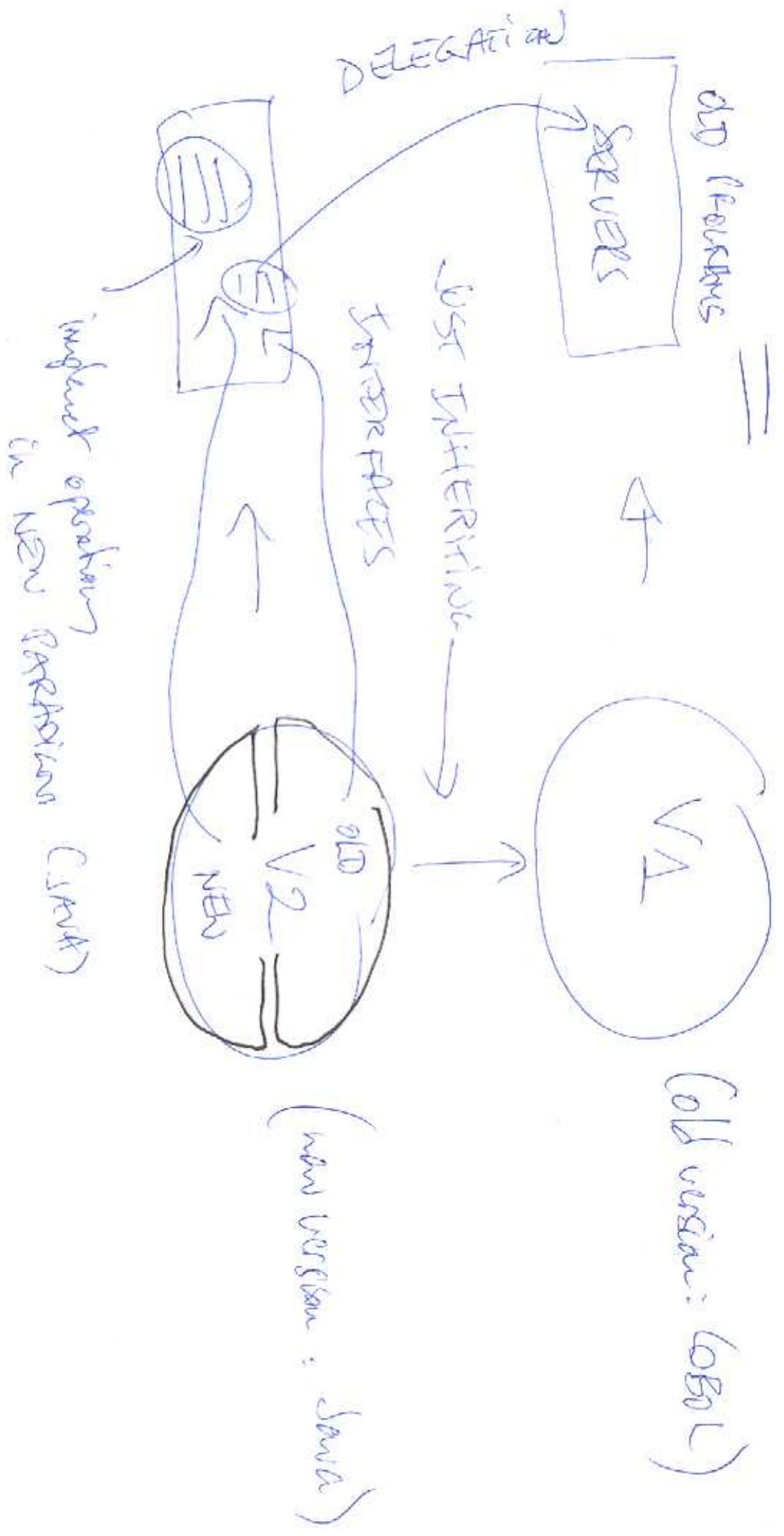


inheritance of implementation



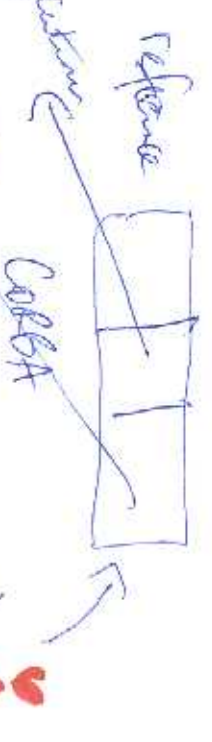
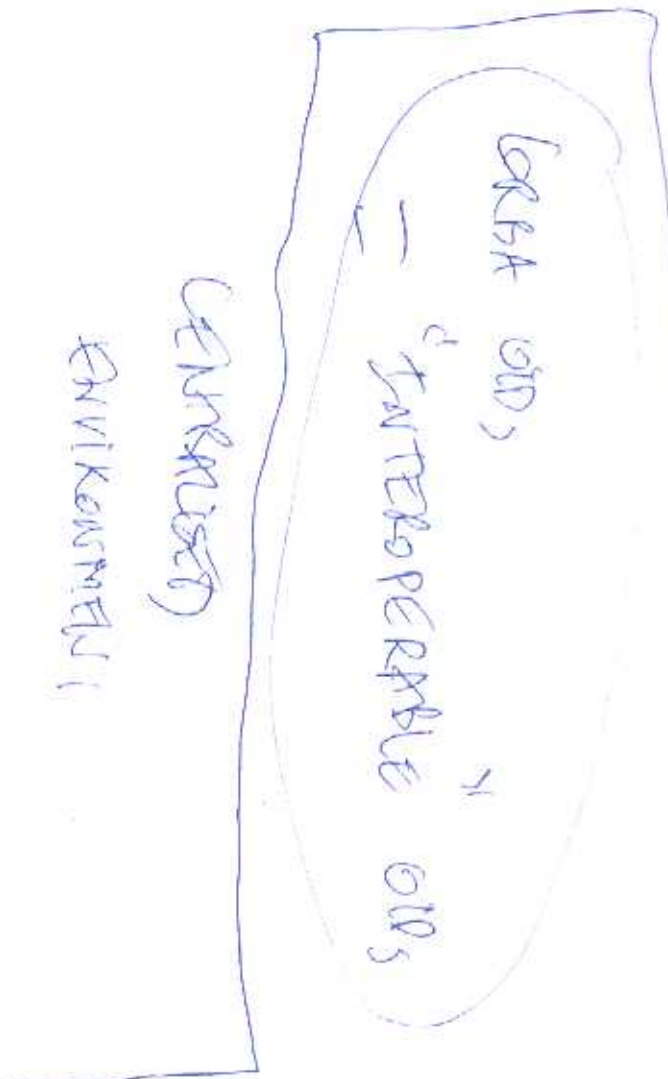
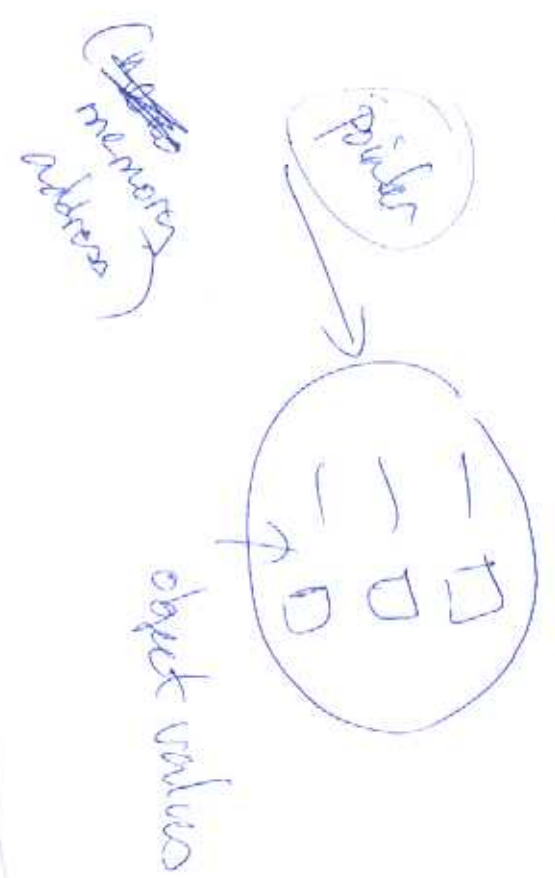
Distributed Object Model

VERSIONING



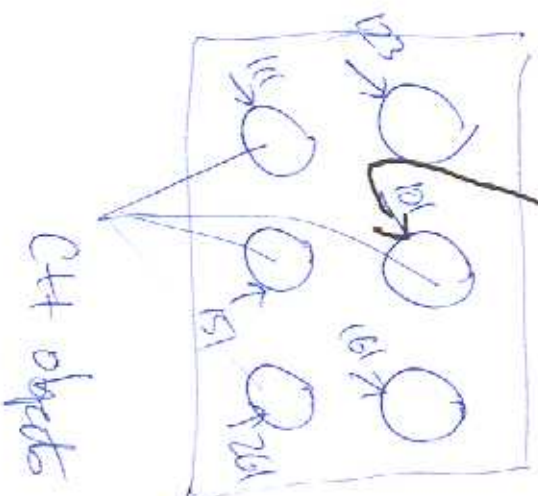
L9-6

OBJECTS IDENTIFY (oid) → (points)



oid (virtual)

LINK & CORBA
OID with
CORBA oid



SERVER



DISTRIBUTED ENVIRONMENT

19-7

STATIC

~~BRINDANT~~
INVOCATION

(IT USES -SUB)

IT KNOWS HOW TO MASSAGE OPERATIONS KNOWN

@ compile time (i.e. they are defined in the

IDL - Interface Definition Language)

EXTEND (EXERCISE)

SMART SUB

(extension of SUB with actions)

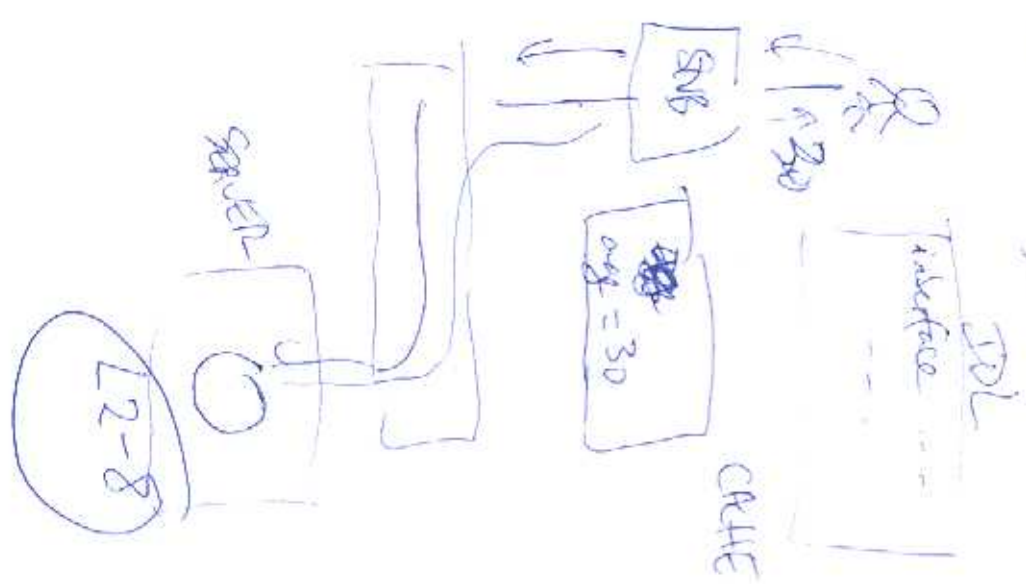
e.g.: org of TOM [1st invocation]

TOM.org()

↳ org = 30

Directly access [2nd invocation]

code

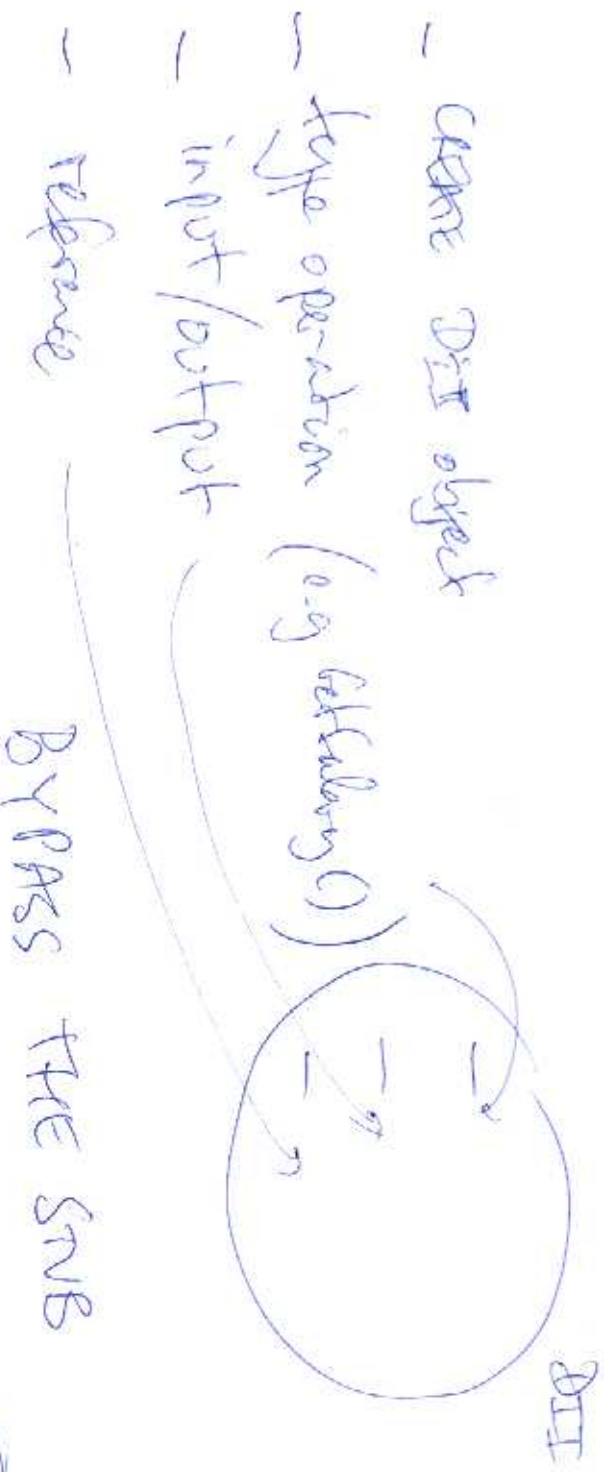


DYNAMIC INVOCATION

... FOR OPERATIONS THAT ARE NOT KNOWN @ COMPILE TIME

↳ SUB ANNOT MARKERS

↳ DII (Dynamic Interface Invocation)



Stick figure icon [~~Service~~ APPLICATIONS (PROBLEMS)]

① — OBTAIN A REFERENCE (bind()) in JARs)

USING A NAMING SERVICE or TRADING SERVICE.

ASSUME REF1 (is the reference needed)

② — INVOKES AN OPERATION ON REF1

NOTE: OPERATIONS MUST BE DEFINED IN IDL

REF1.getMARKED().

③ LOCAL ORS REMOVED

WILL FORWARD TO THE

ORs THAT

RUN IN 11235

L2-10



PORT#
11235

④ REMOTE ORB receives the request

REF1. get MARK()

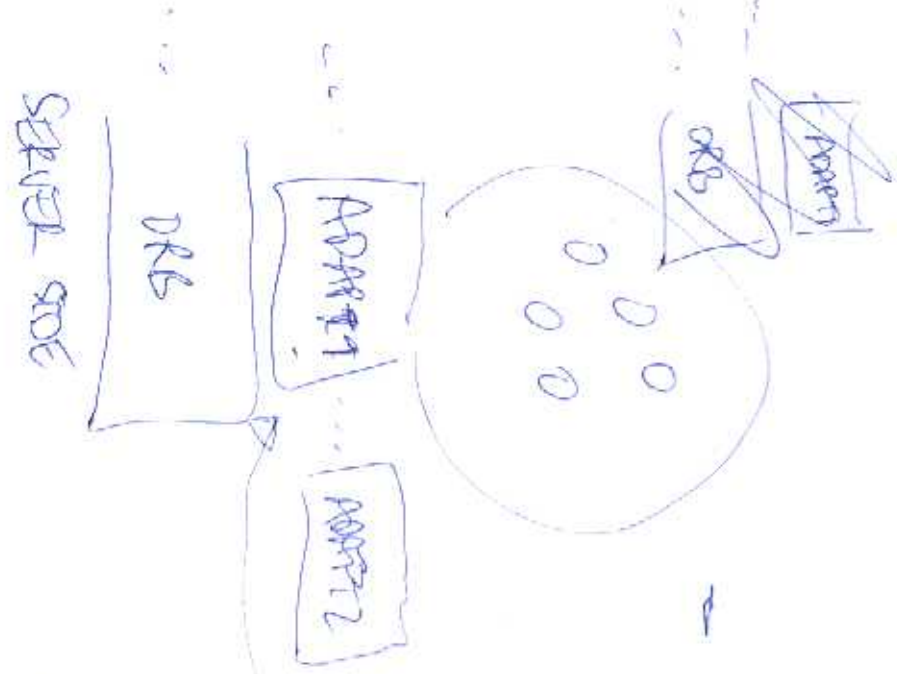
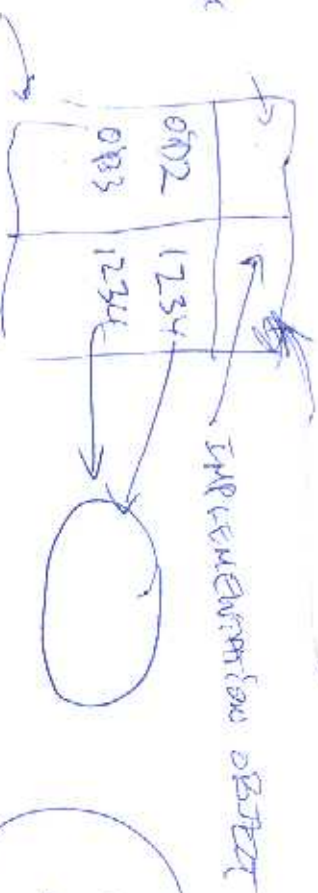
FORWARDS IT TO THE APPROPRIATE ADAPTER
 (ie ADAPT1 embedded in REF1)

ADAPTER "ADAPT1" NEEDS TO MAP

VIRTUAL OBJECT (ie 0102) INTO

"IMPLEMENTATIONS" DID (eg 1234)

ADAPTERA KEEPS "MAP TABLES"



(12-11)