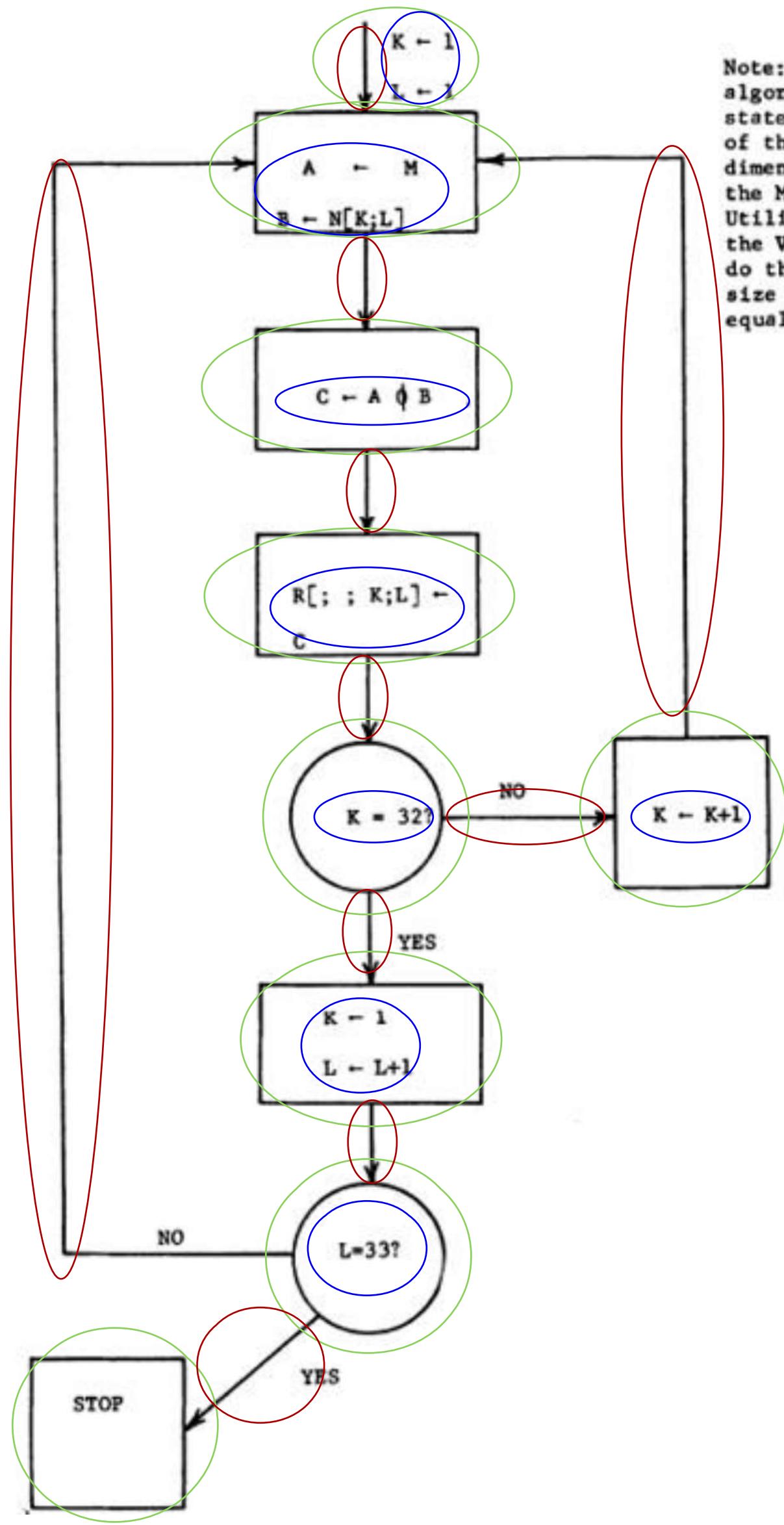
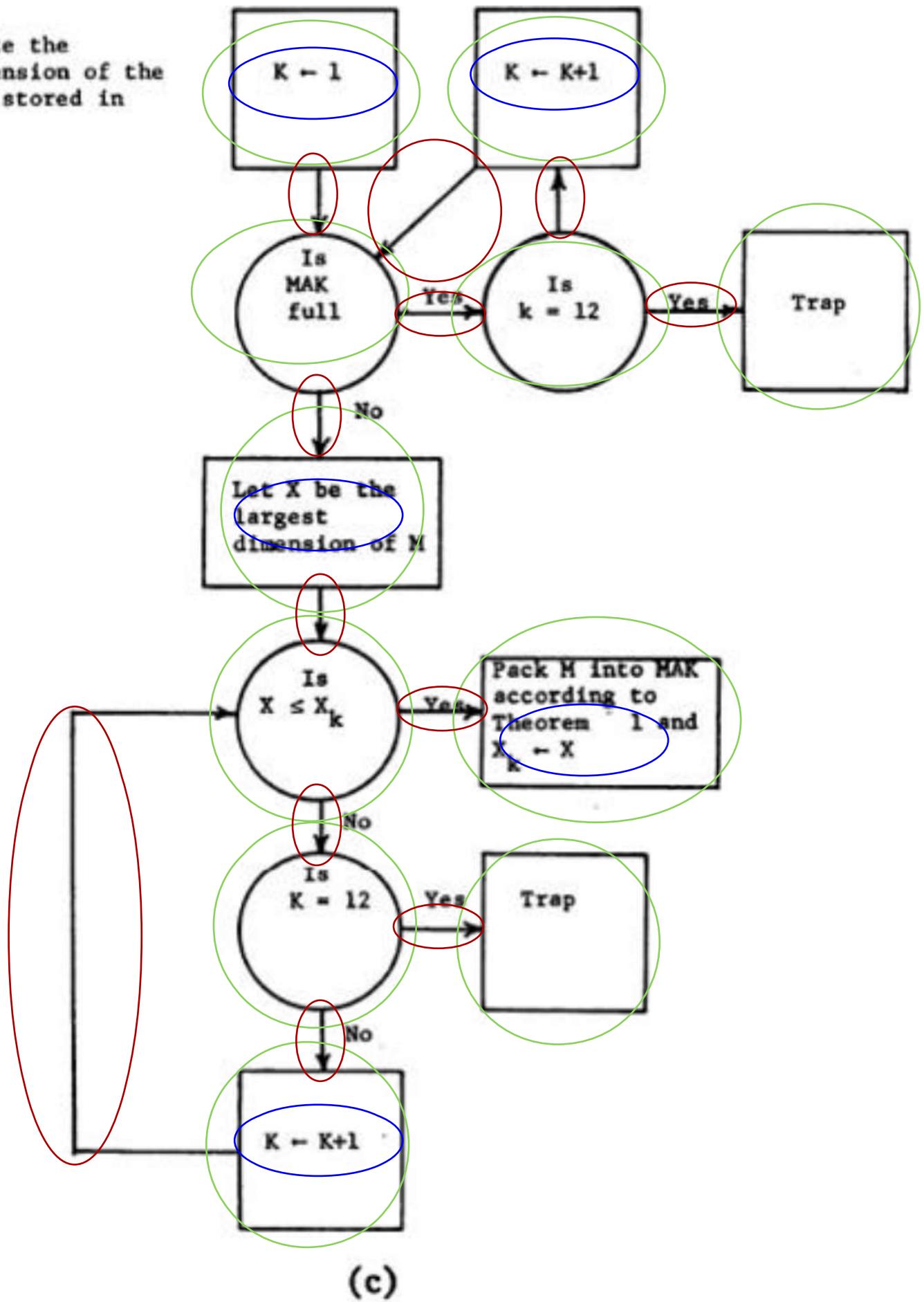


Note: This algorithm is stated in terms of the full dimension of the MLIM; i.e., 32. Utilizing RI, and the VA one can do this for any size less than or equal to 32.

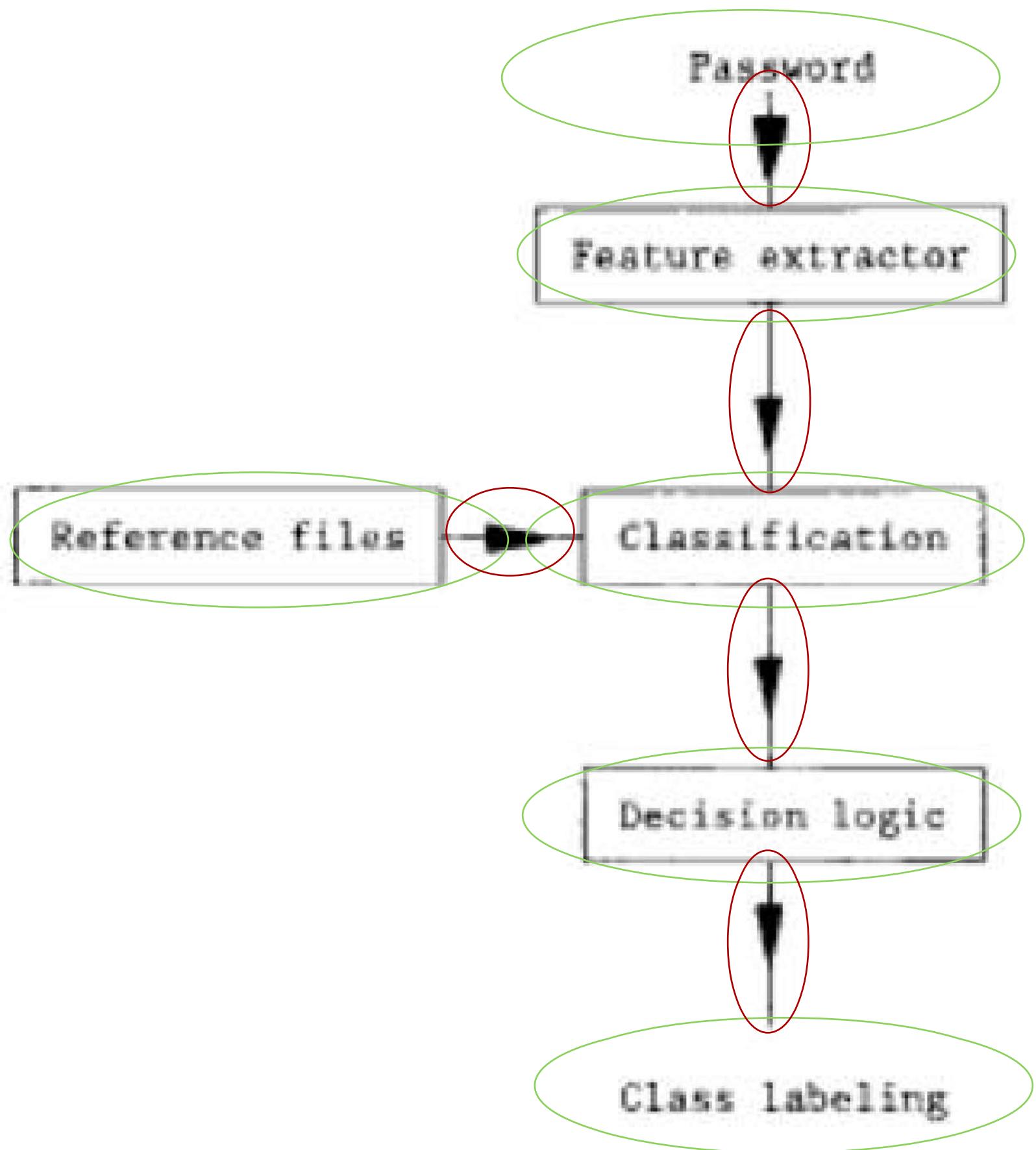
Note: This algorithm is stated in terms of the full dimension of the MLIM; i.e., 32. Utilizing RL and the VA one can do this for any size less than or equal to 32.

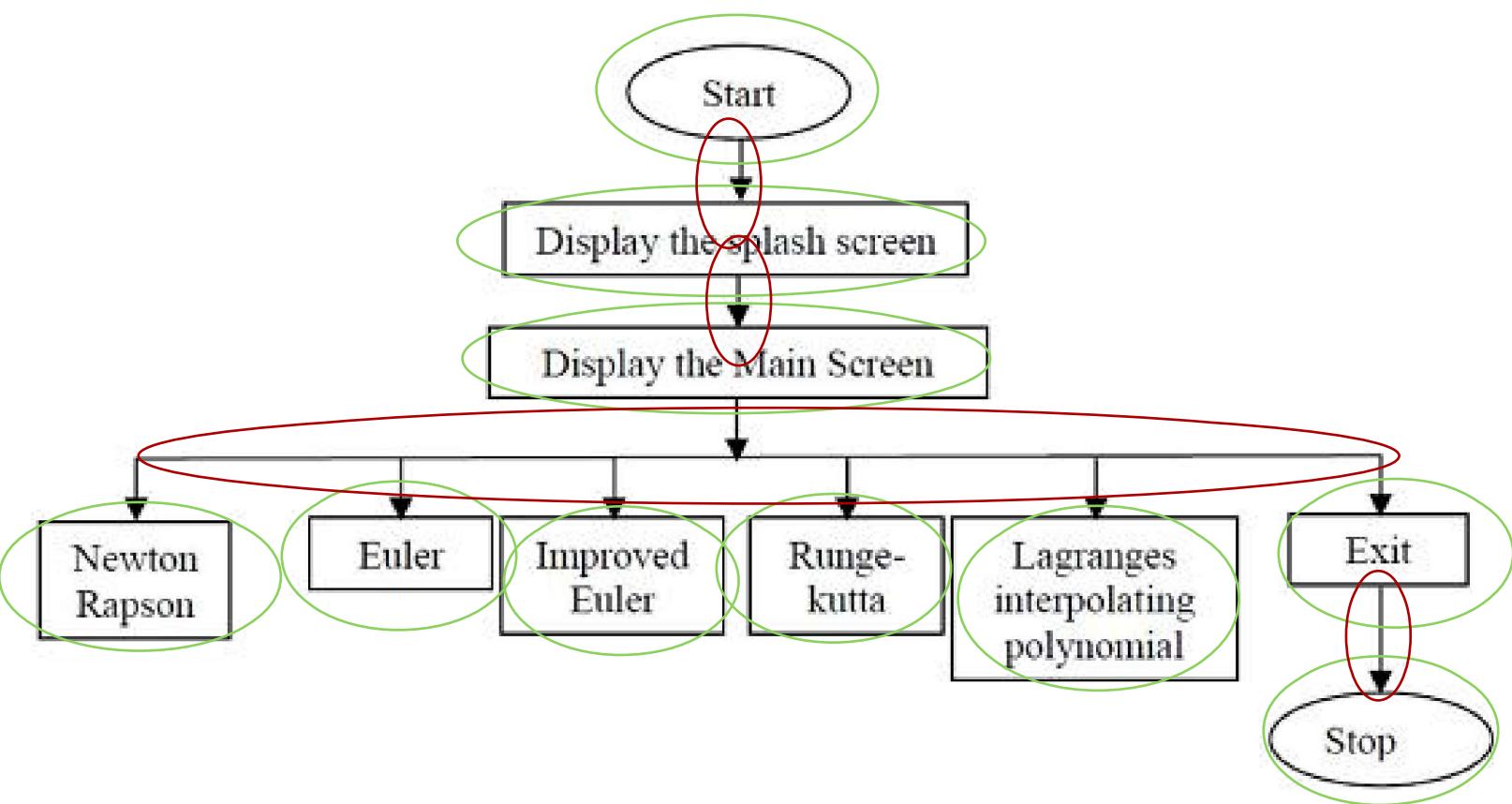


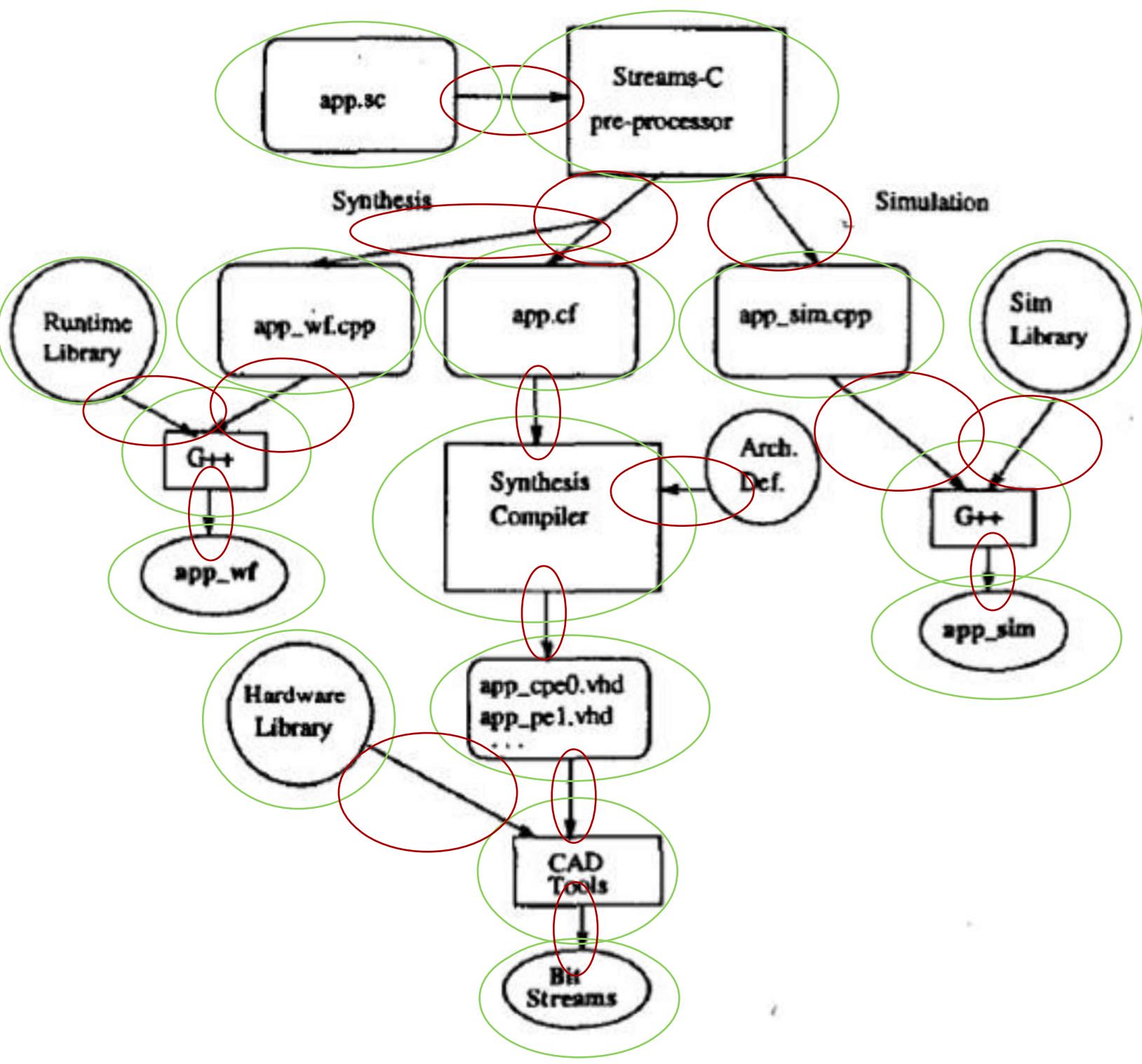
Let X_k denote the largest dimension of the last matrix stored in MAK.

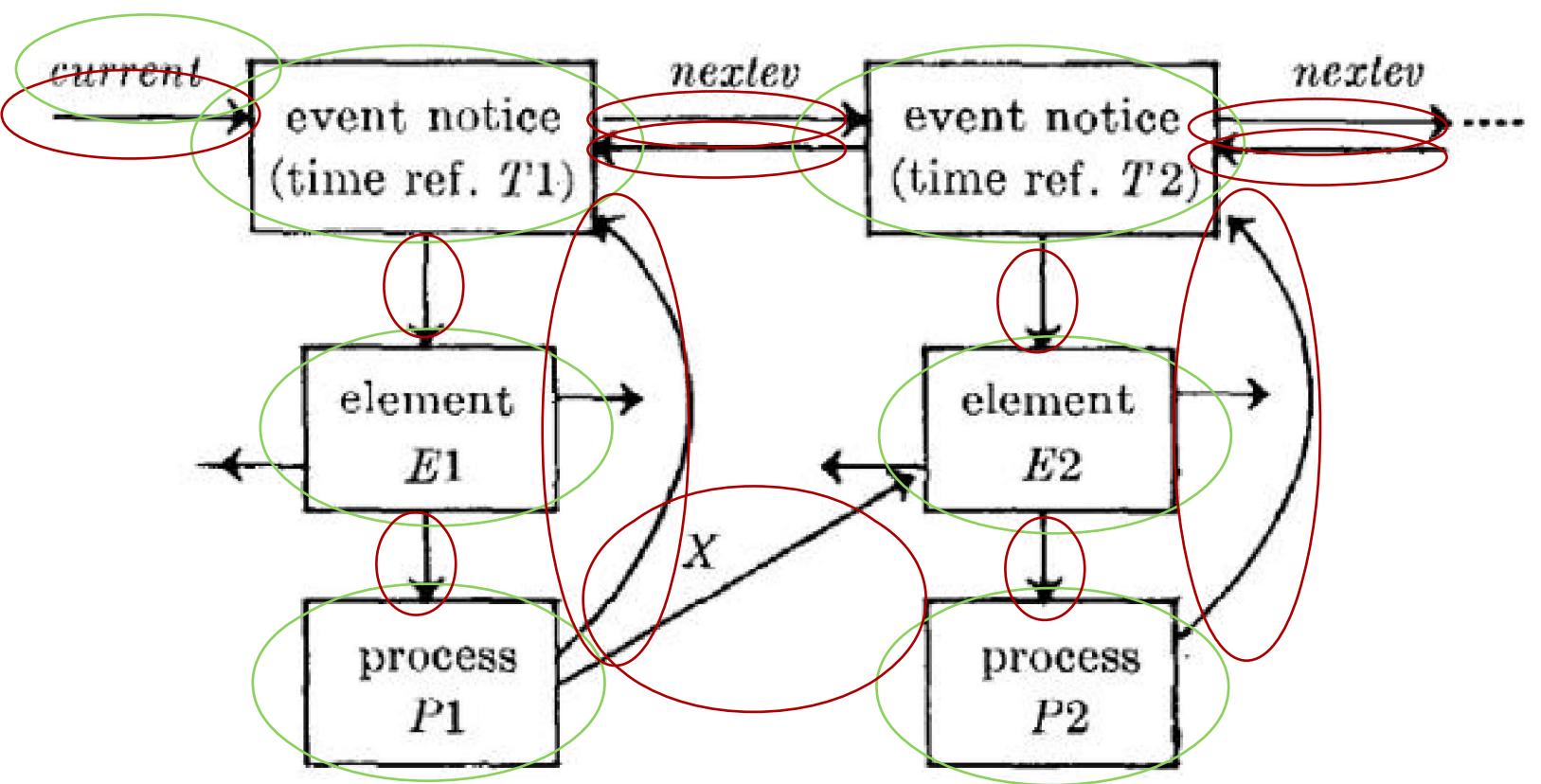


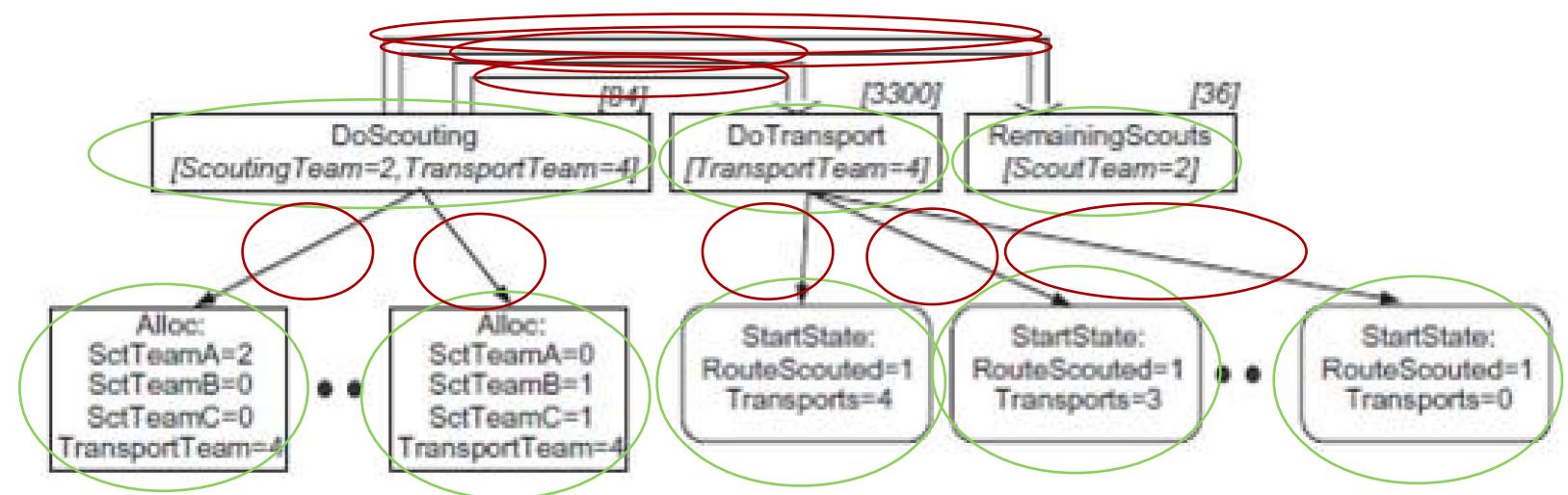
(c)

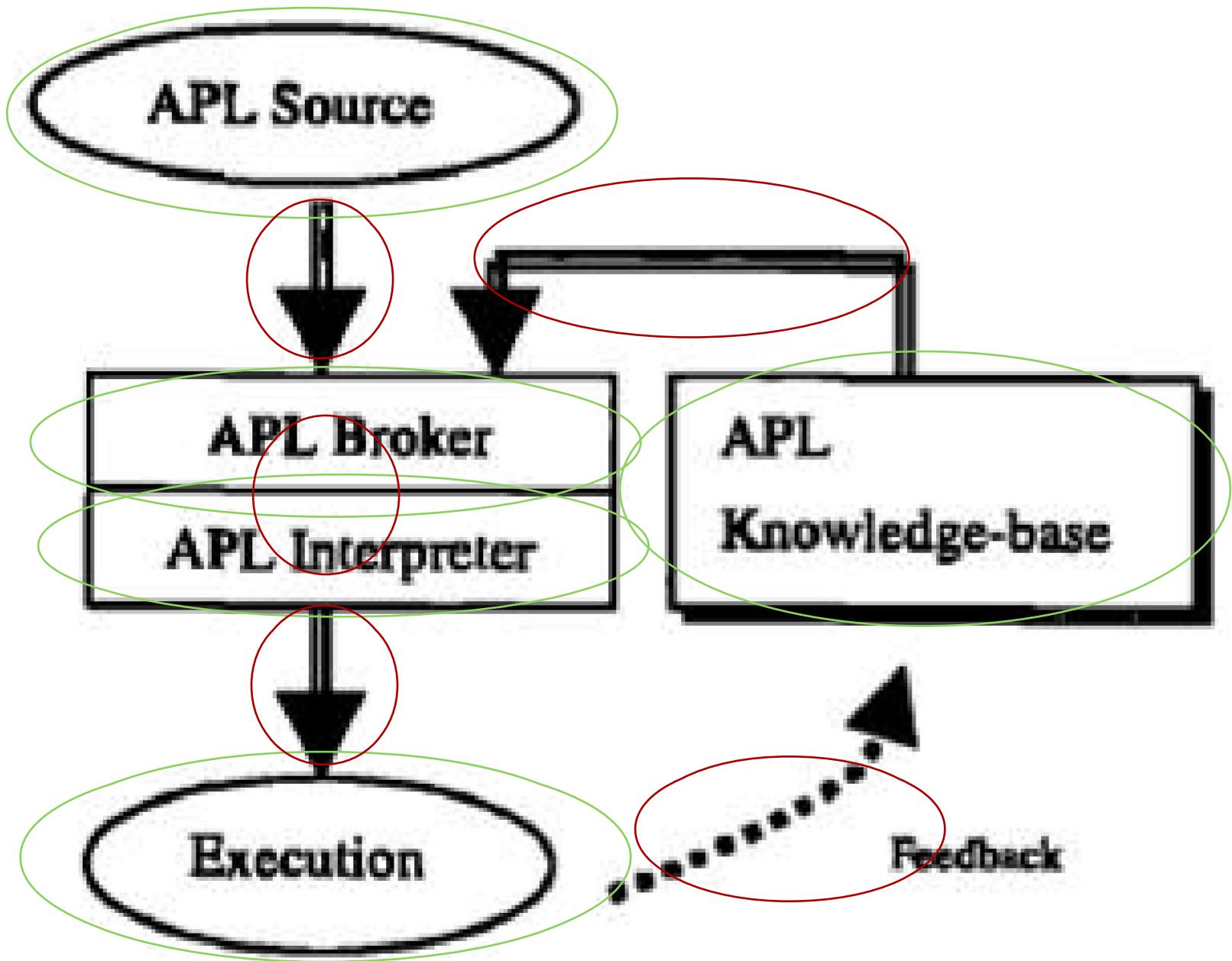


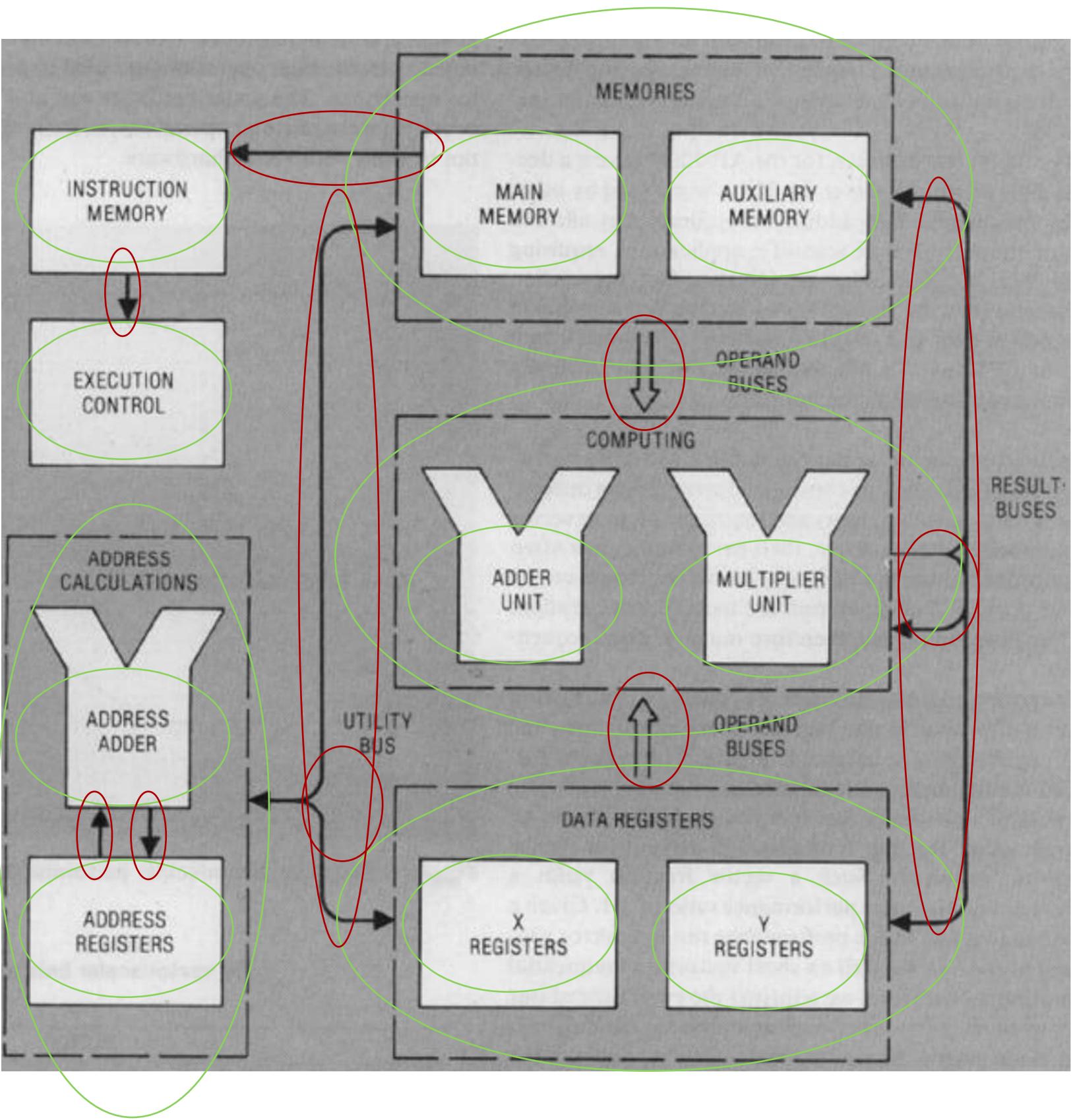


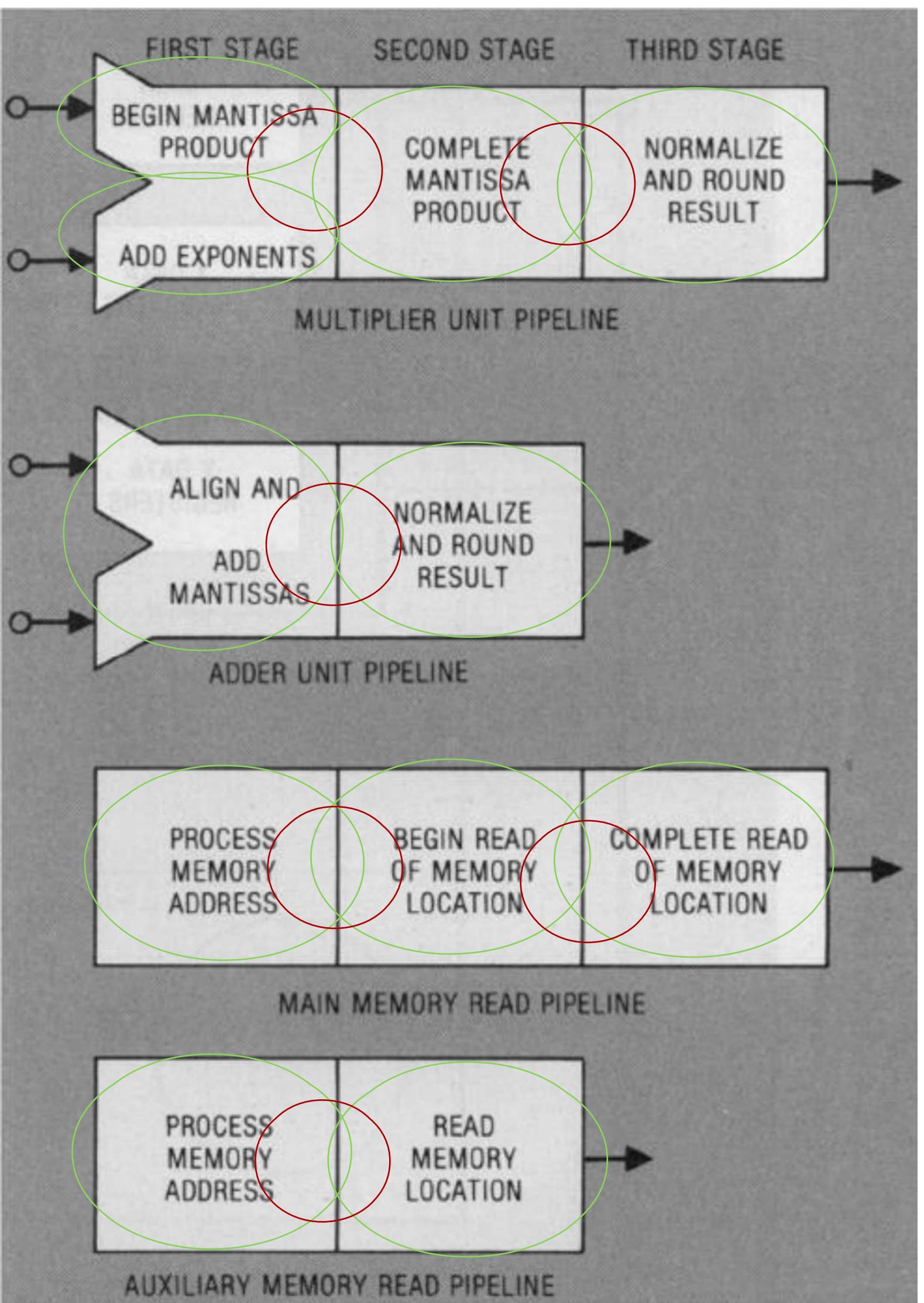


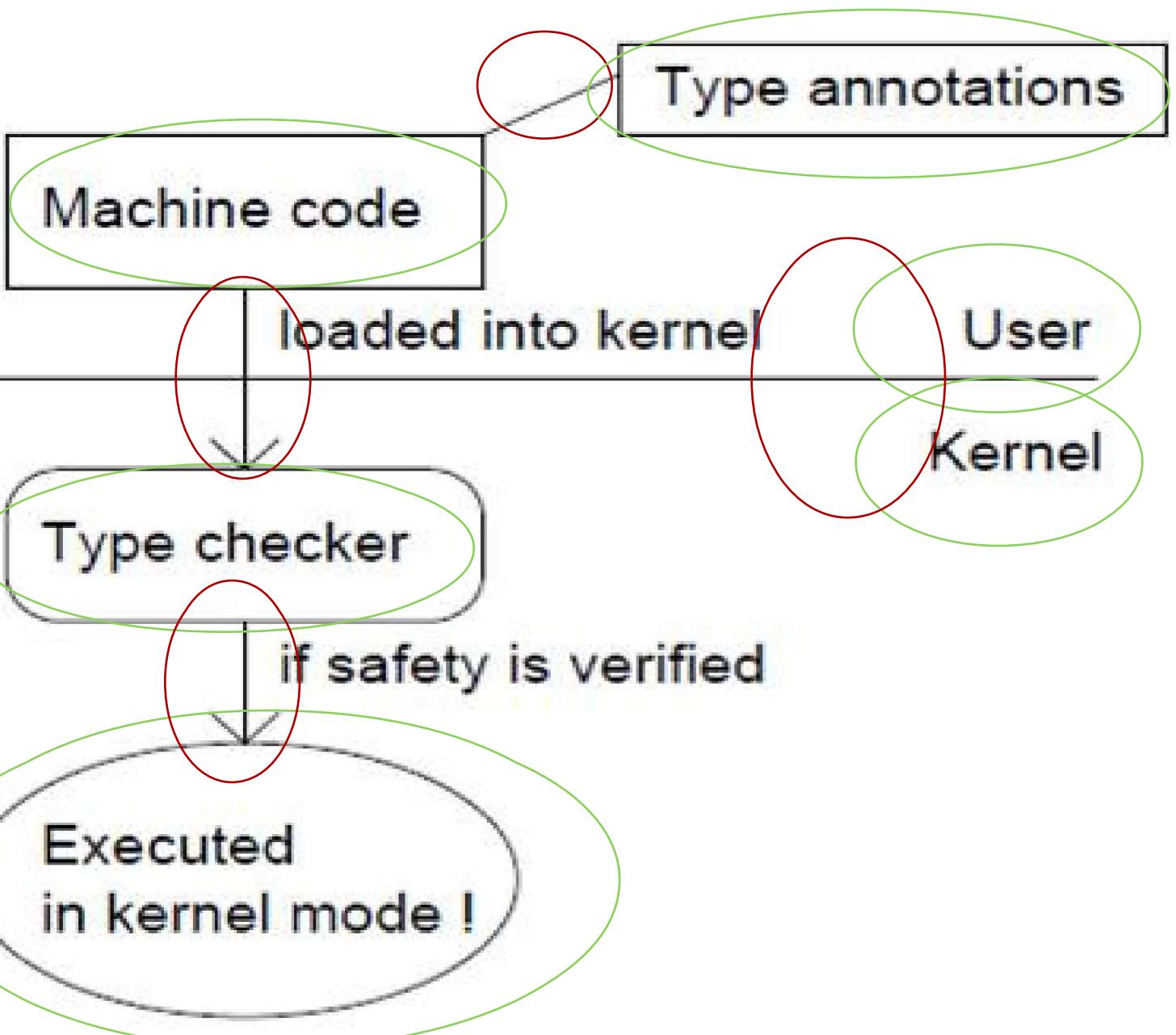




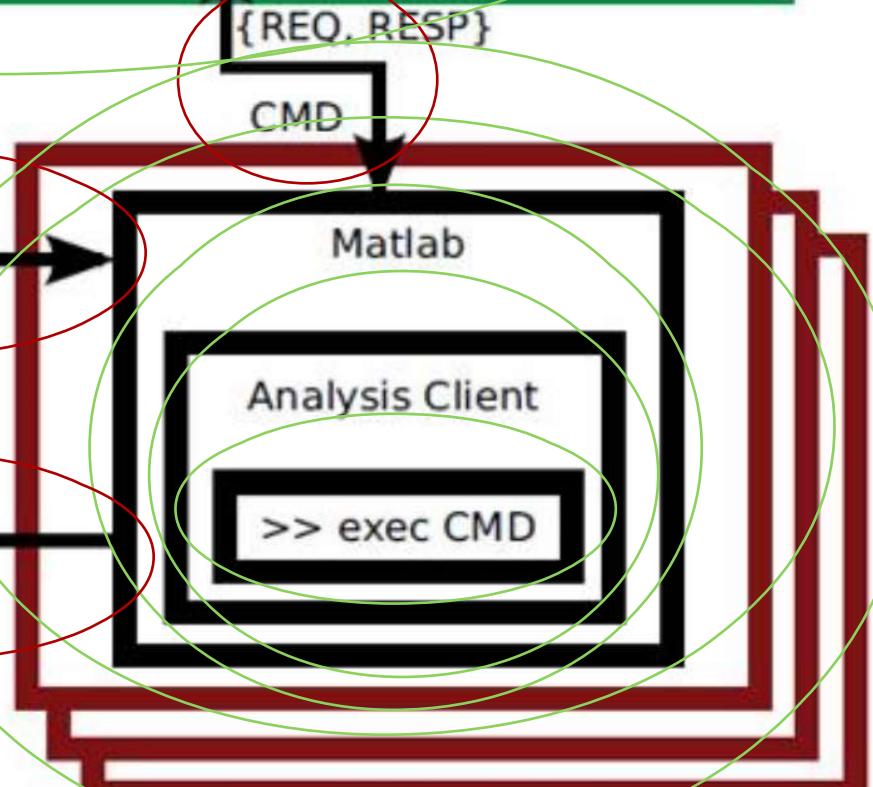
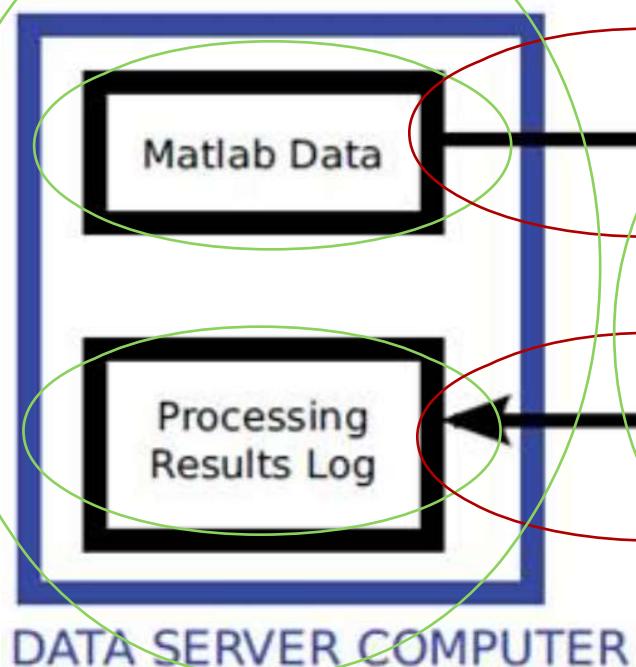
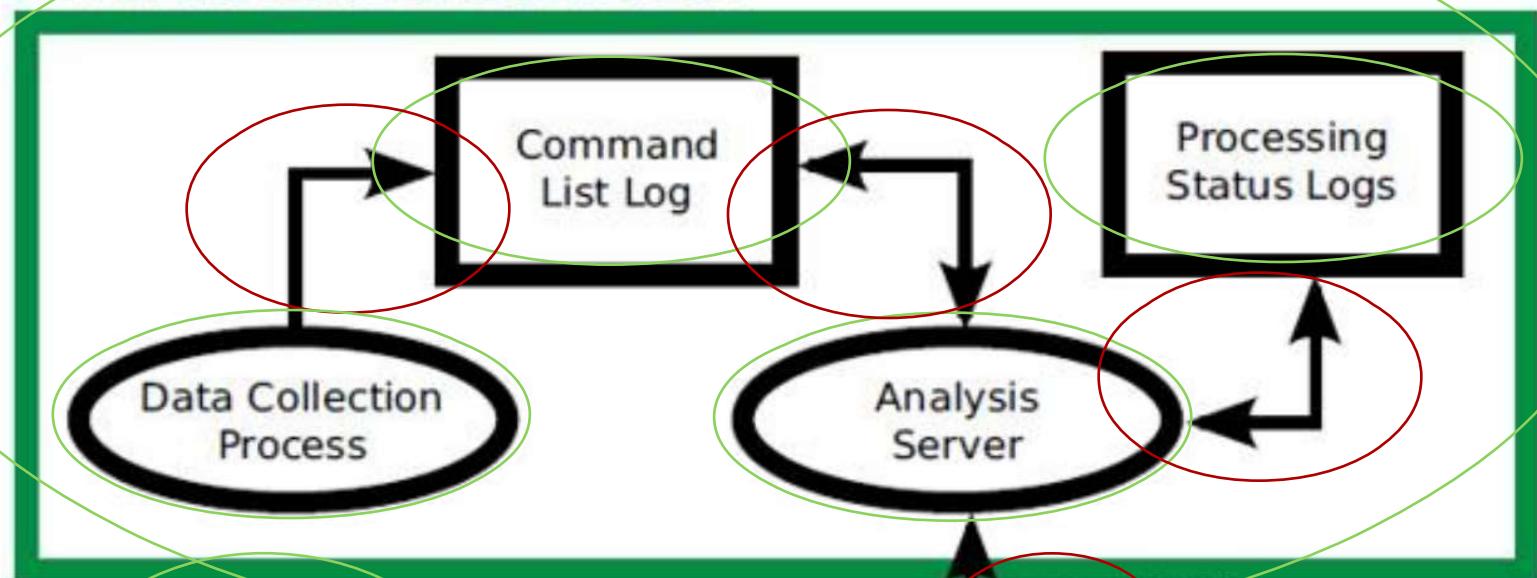






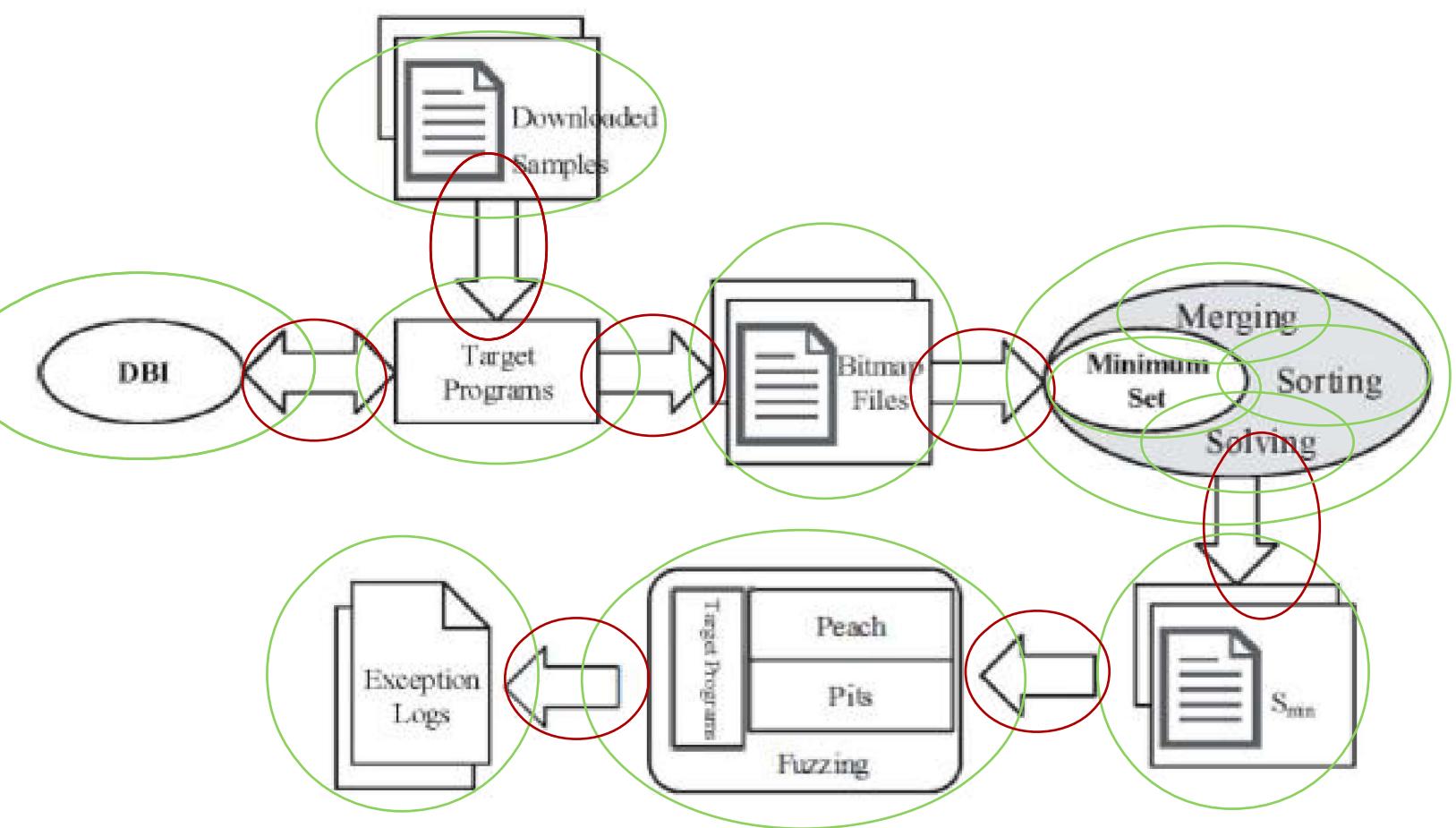


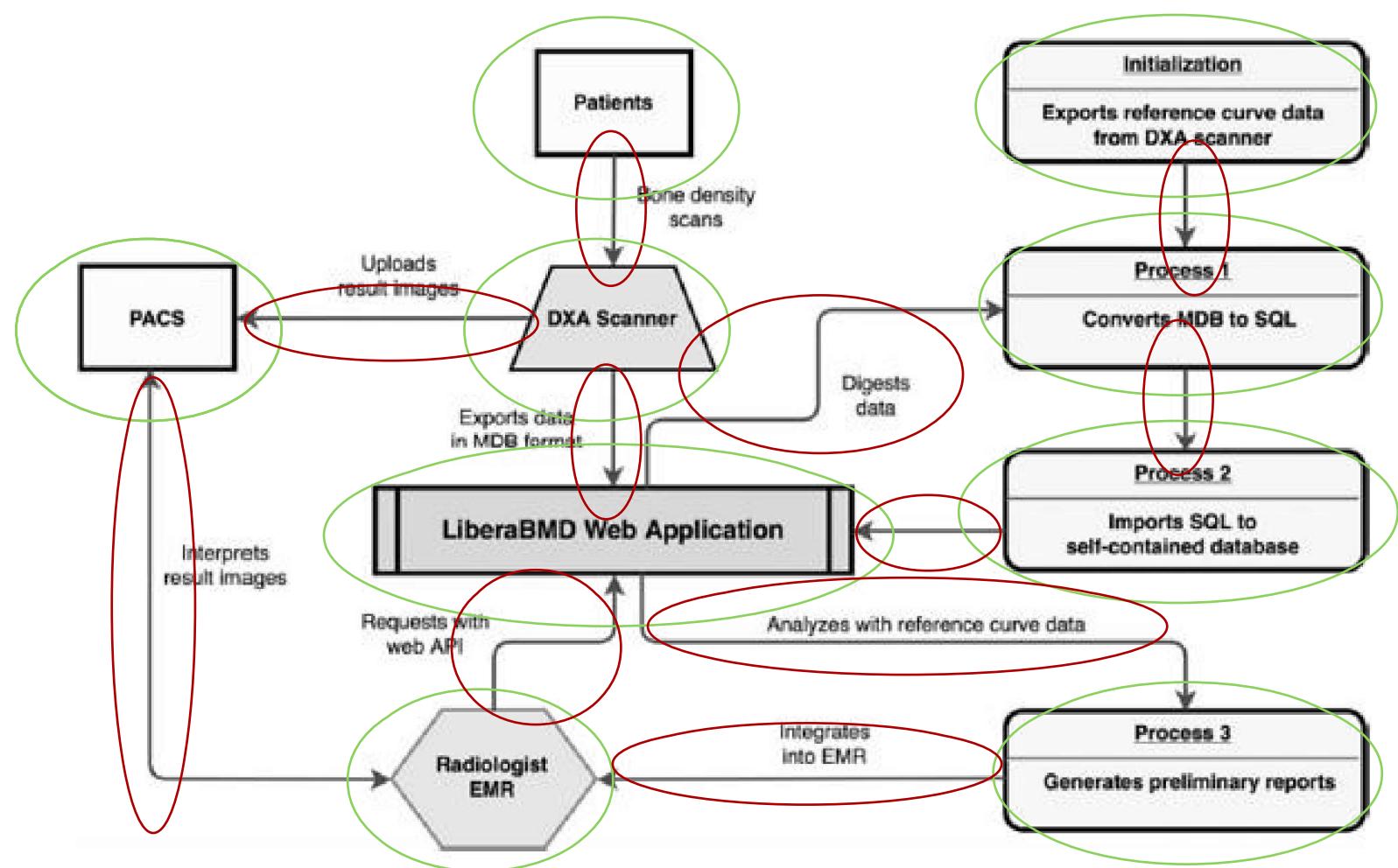
ANALYSIS SERVER COMPUTER

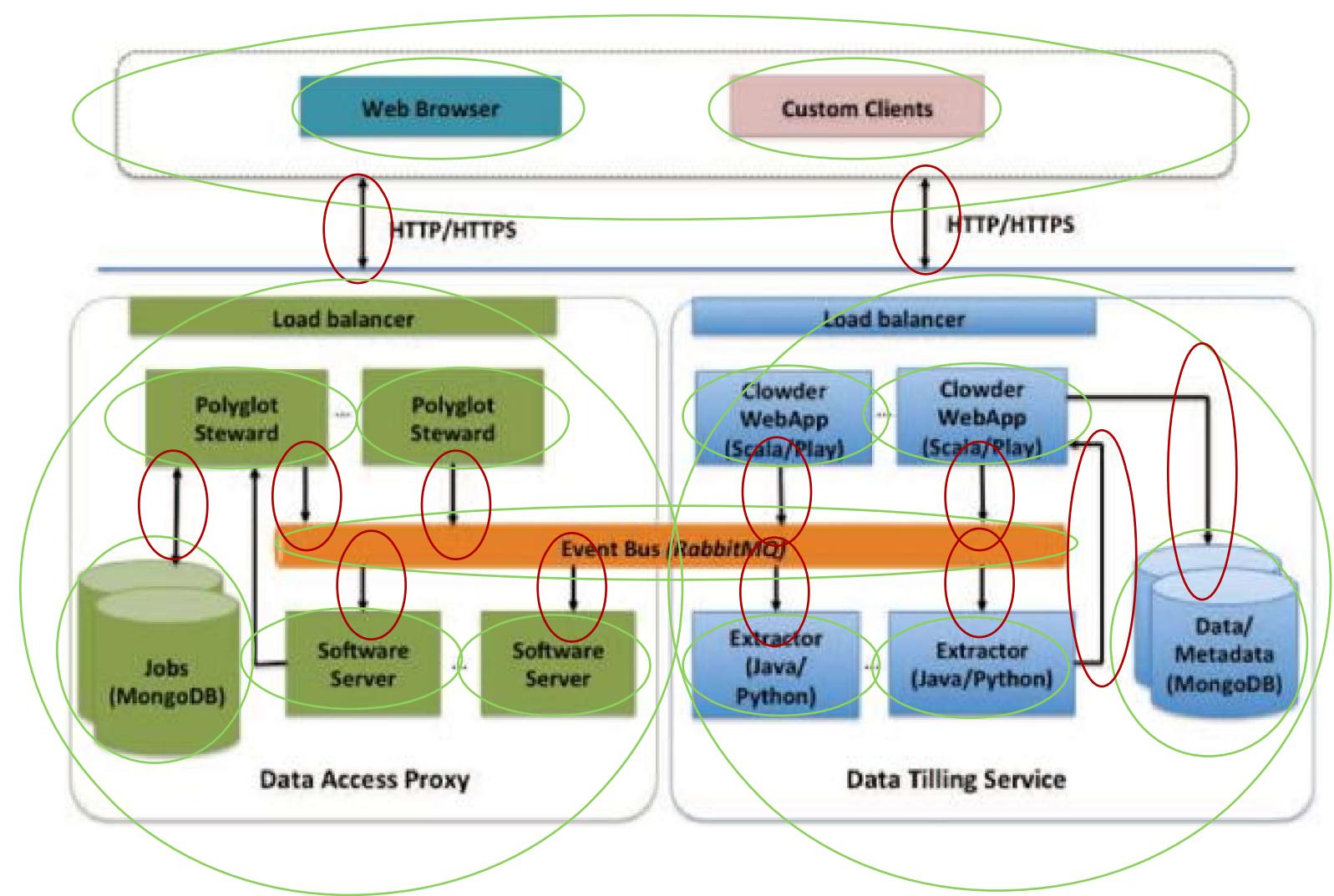


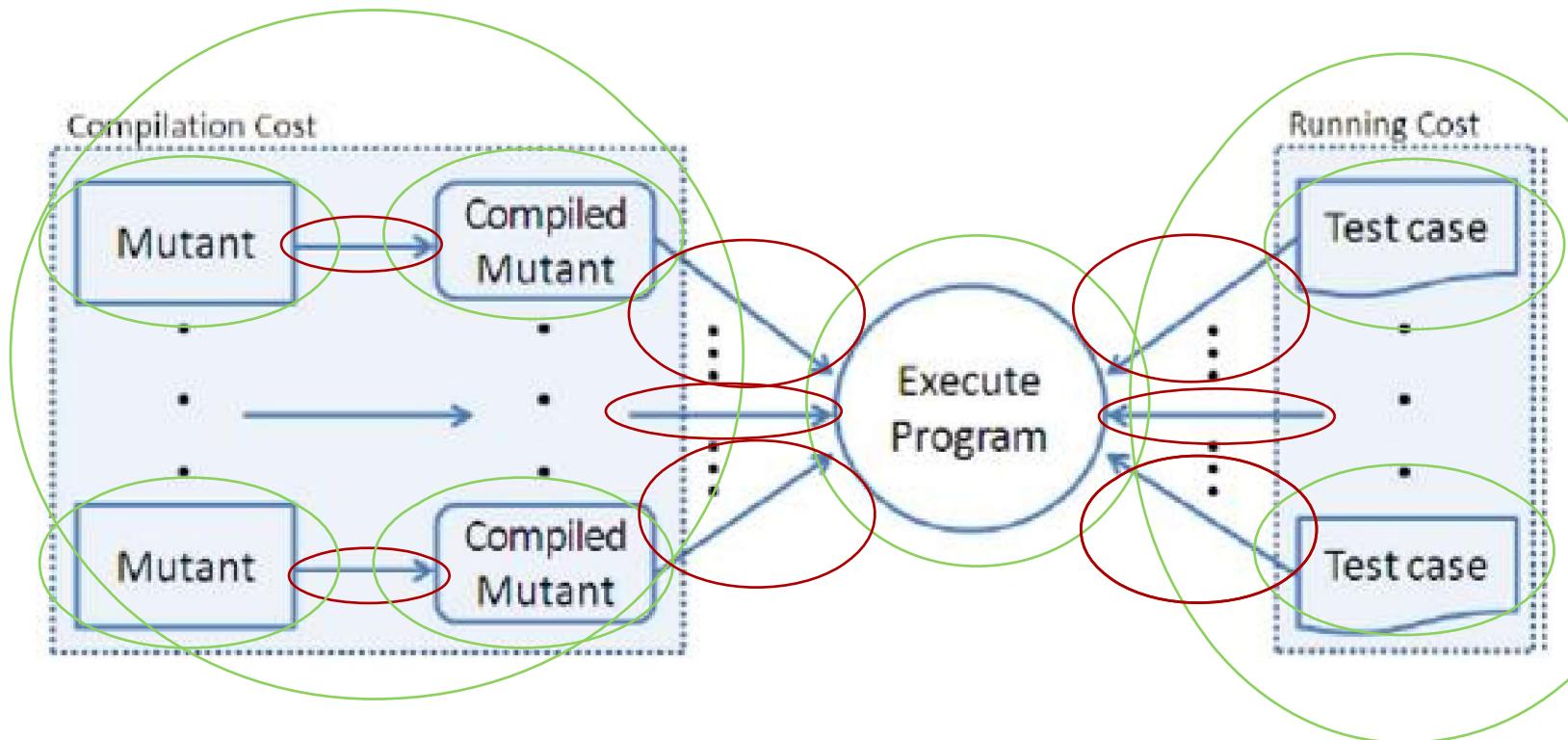
DATA SERVER COMPUTER

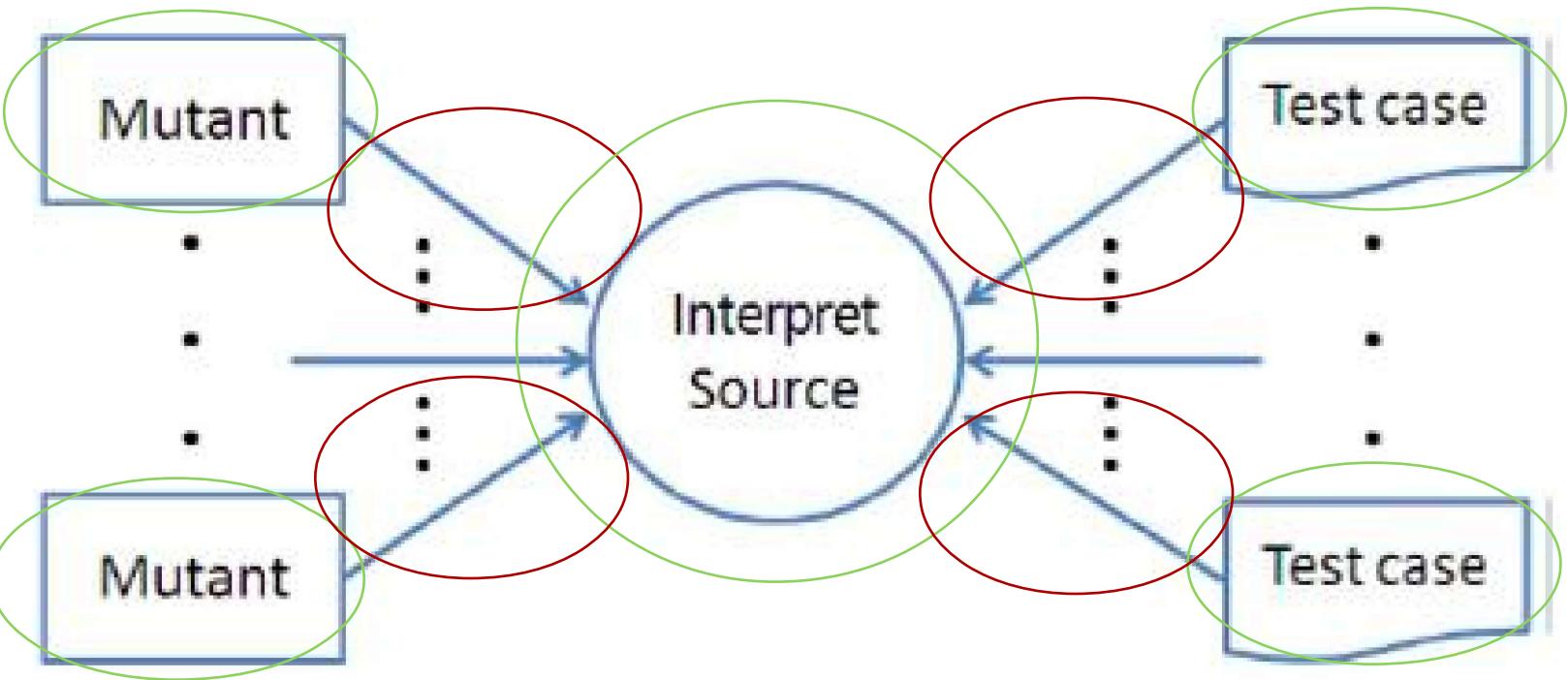
ANALYSIS CLIENT COMPUTERS

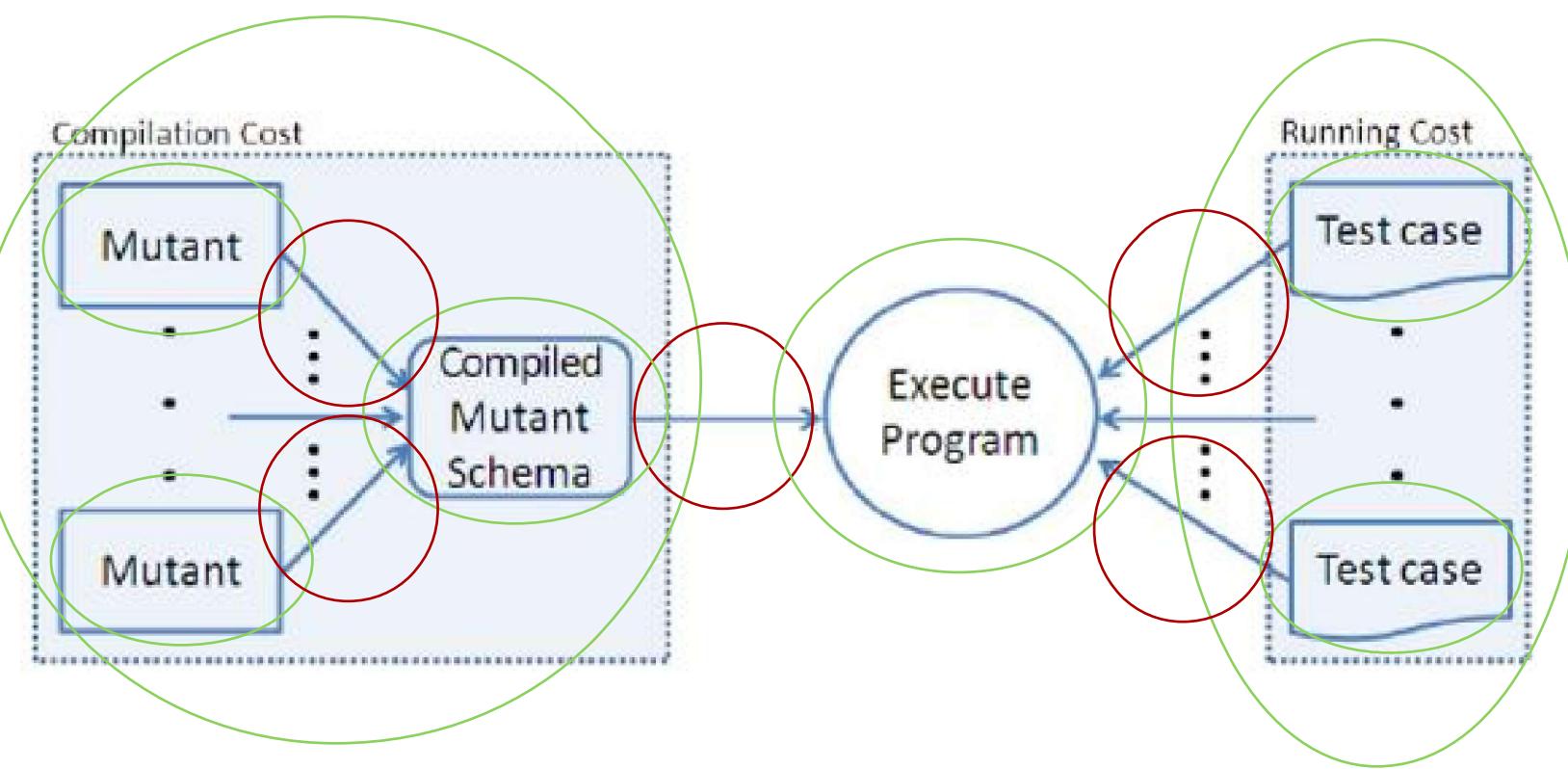


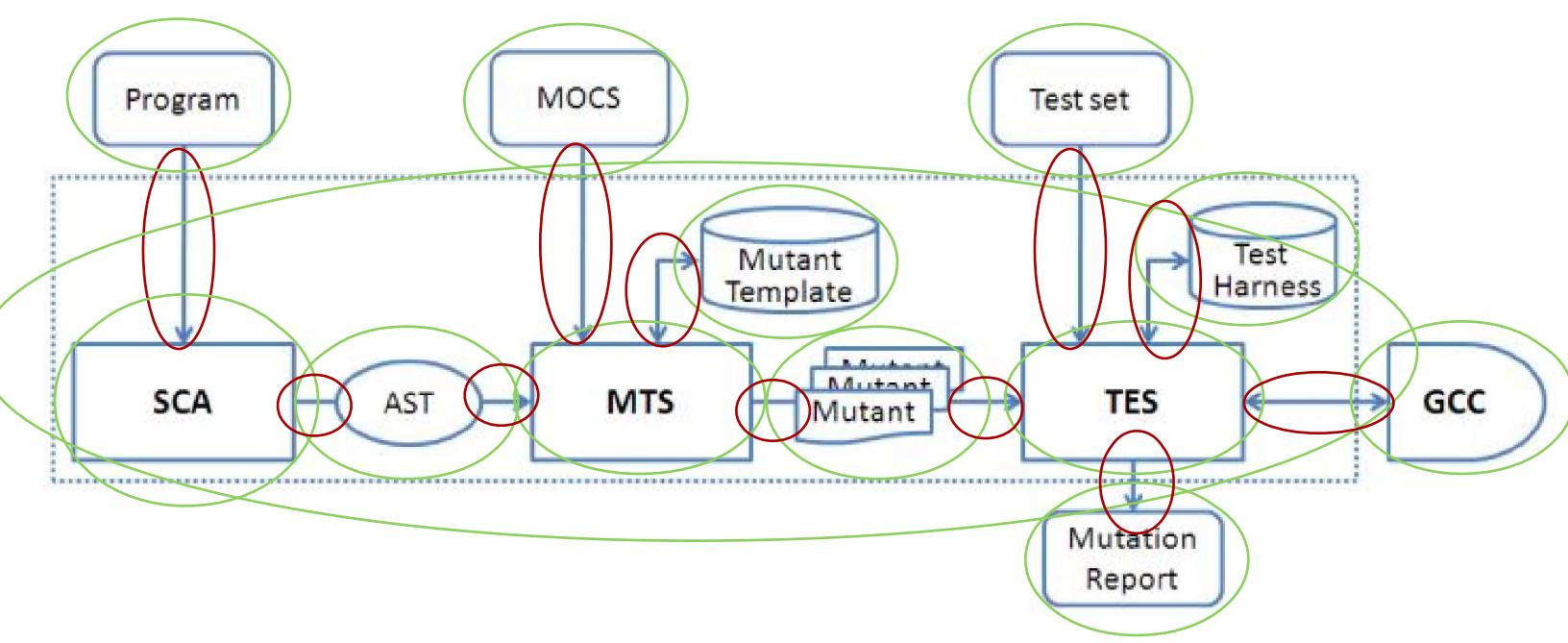


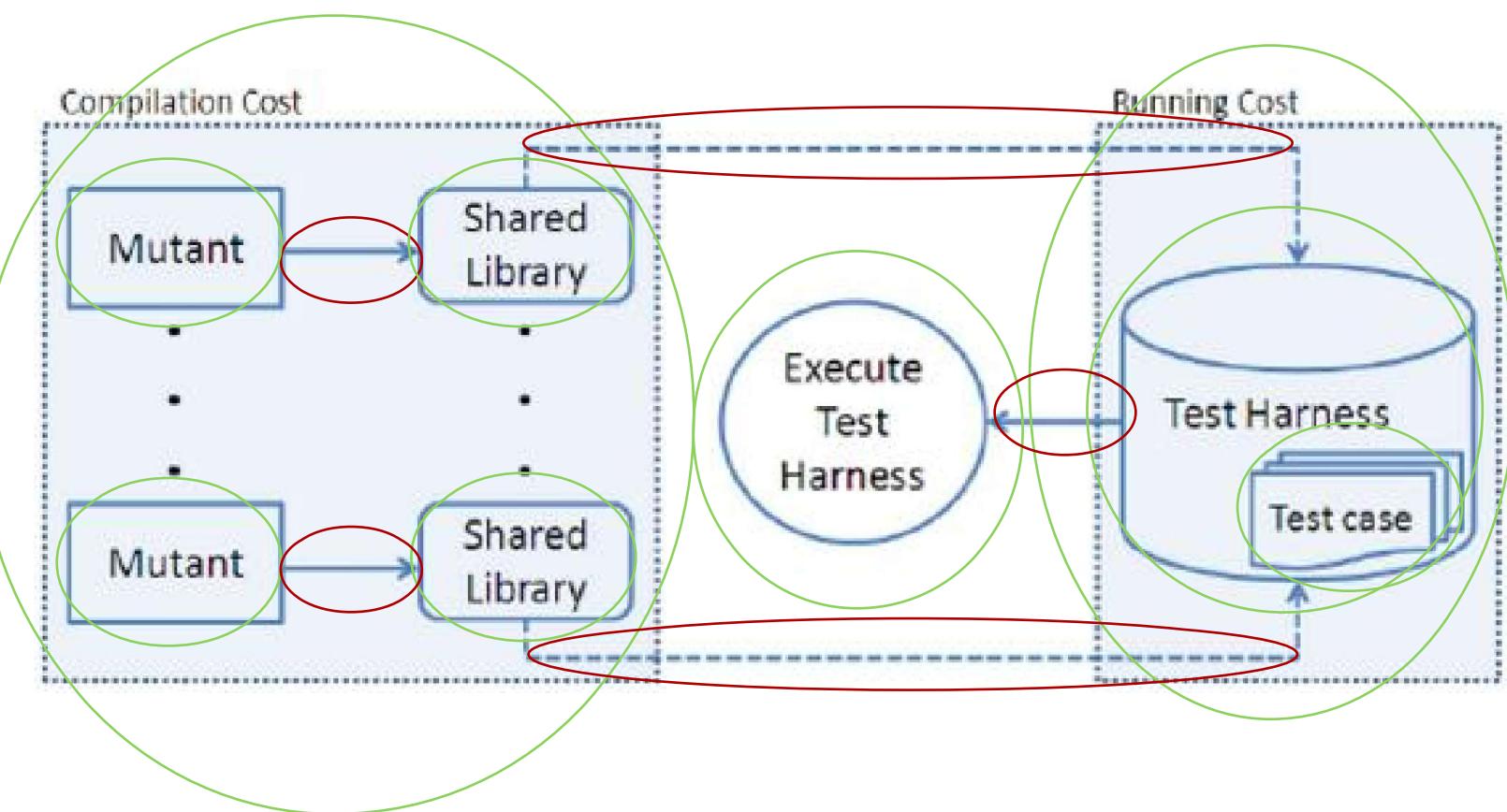


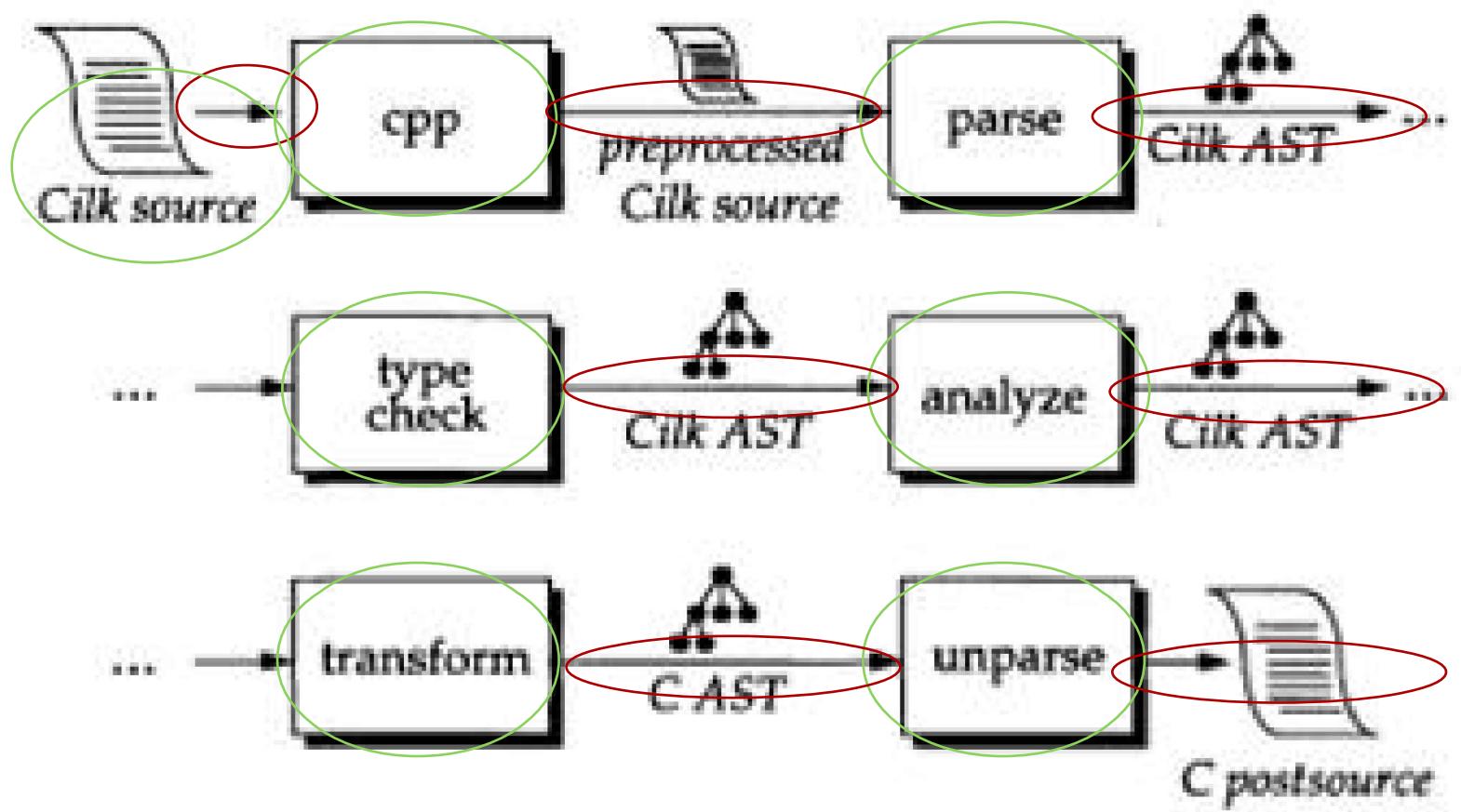


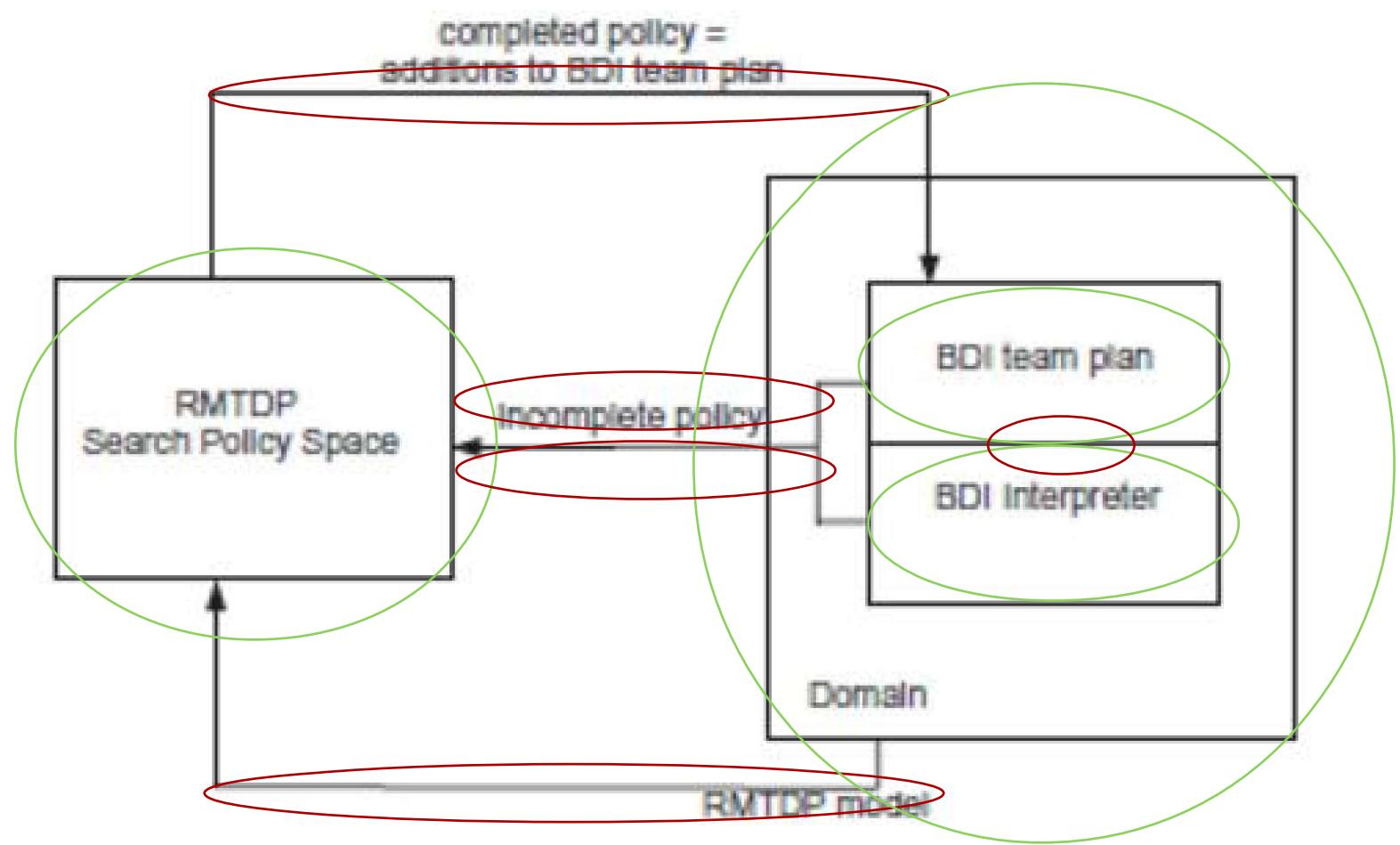


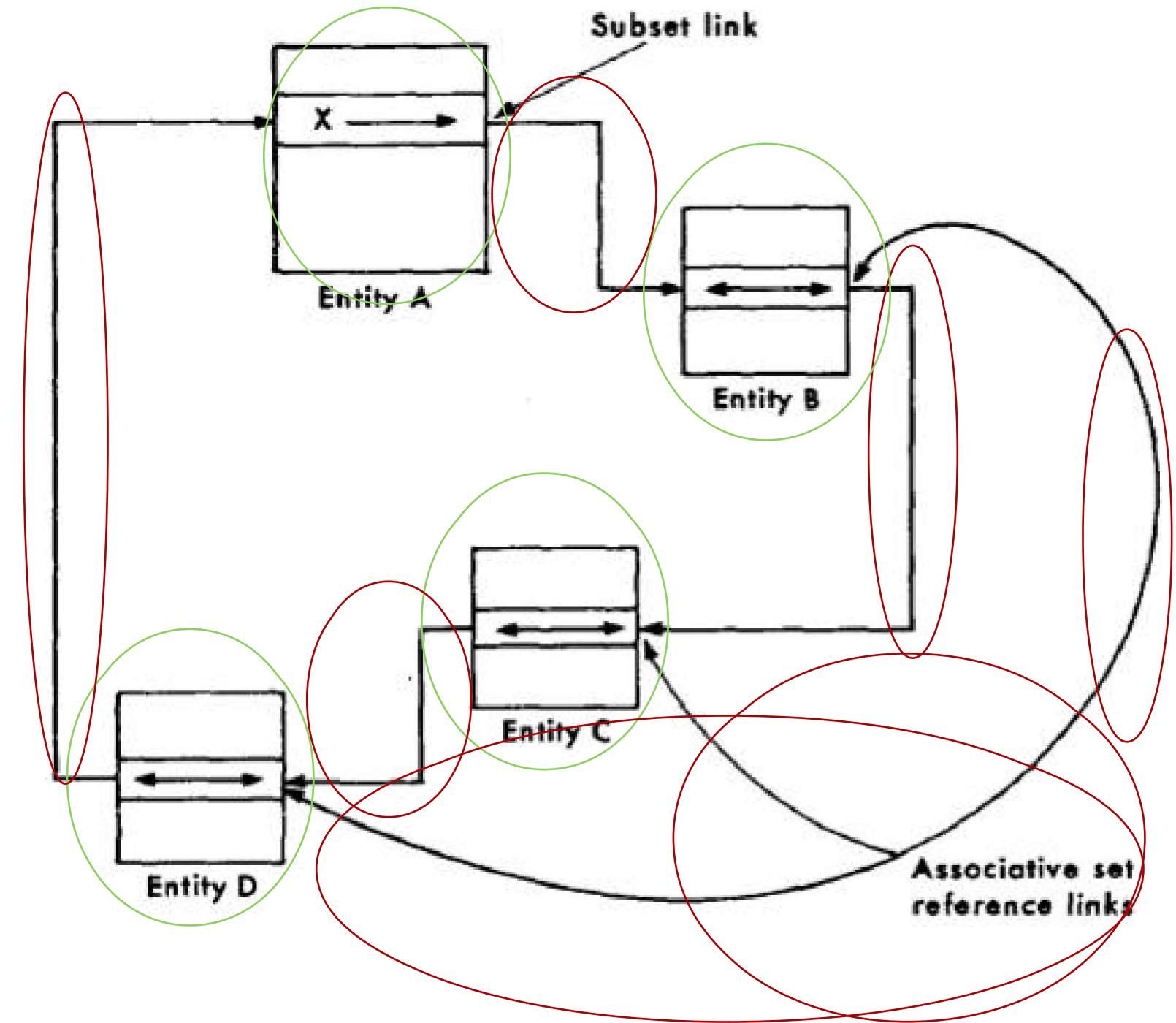


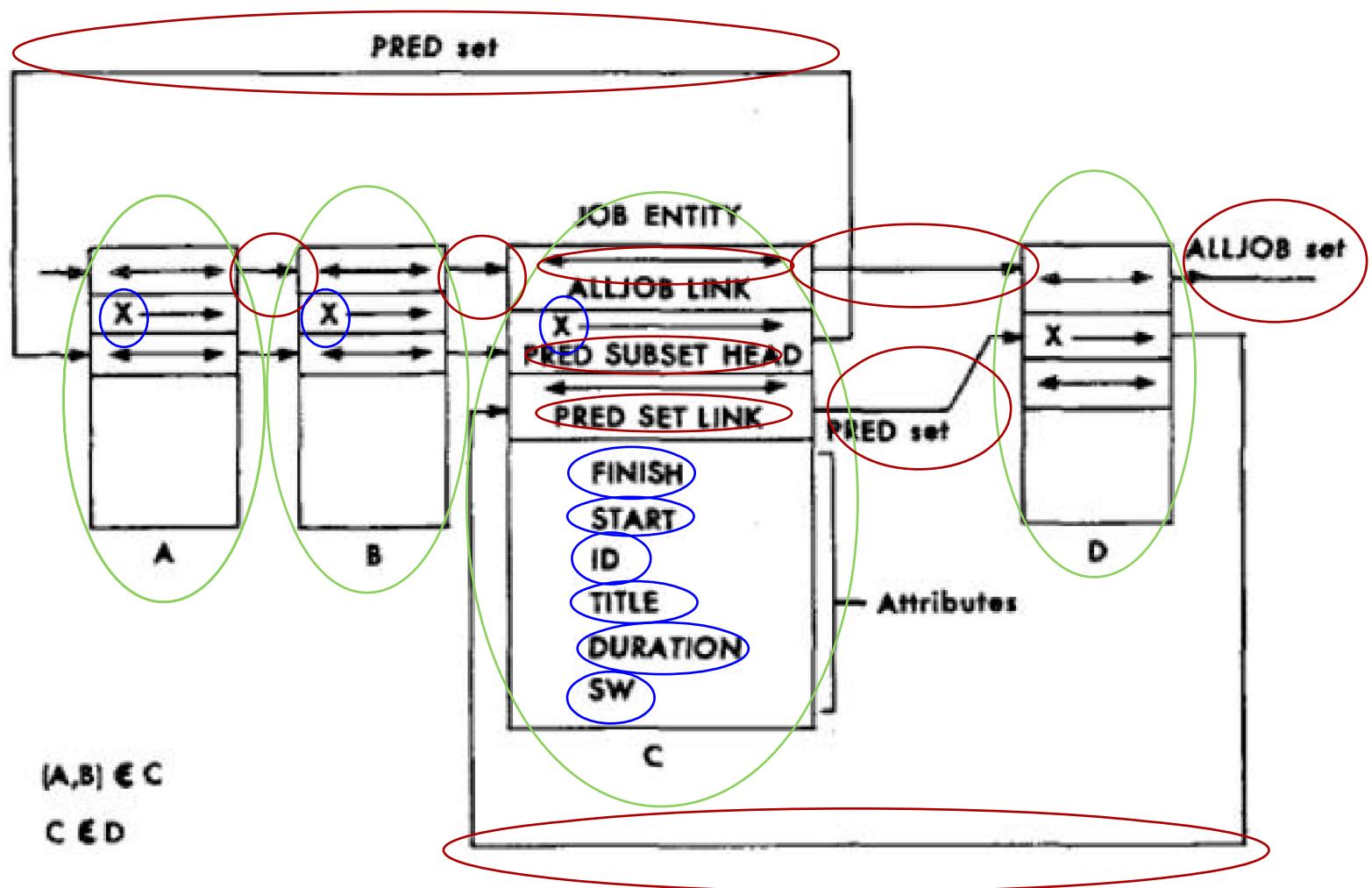












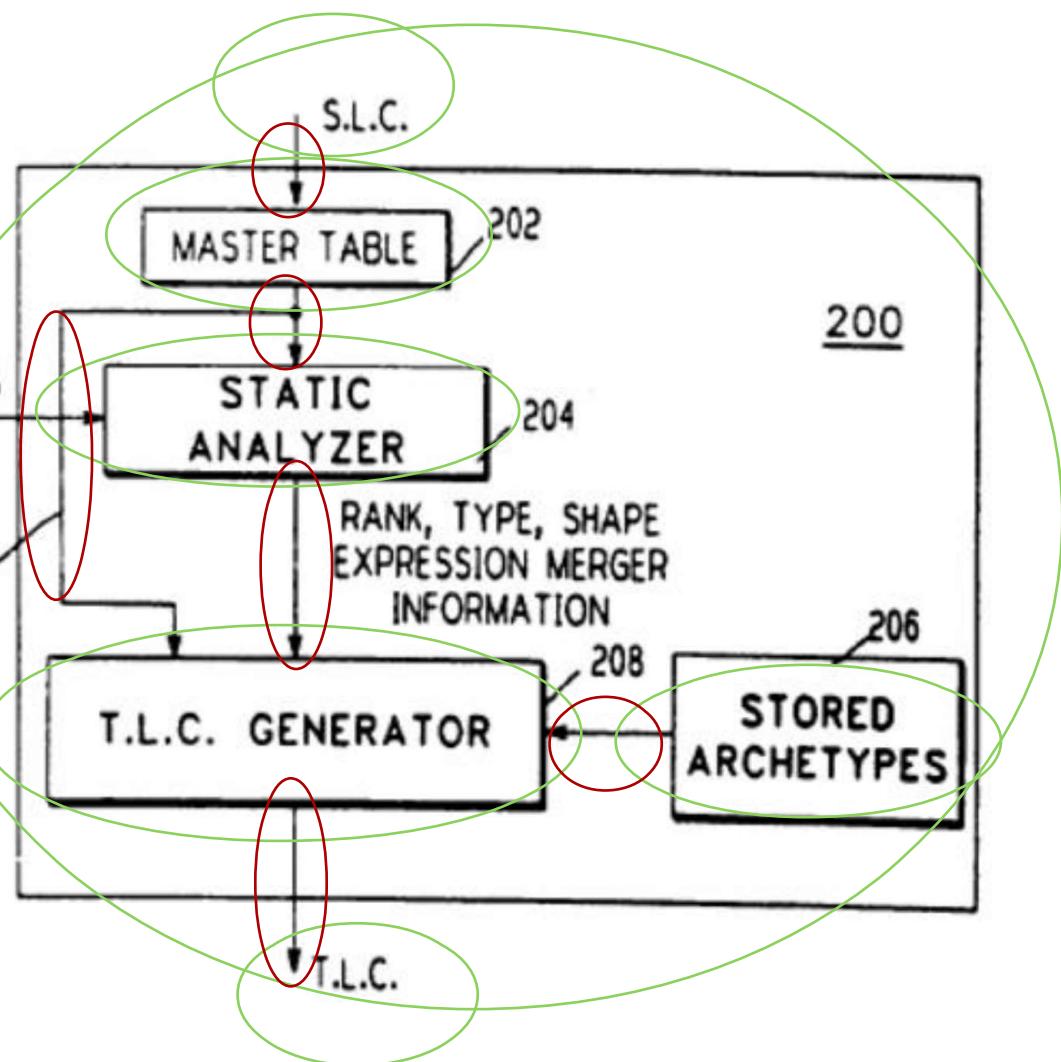
S.L.C.=SOURCE
LANGUAGE CODE

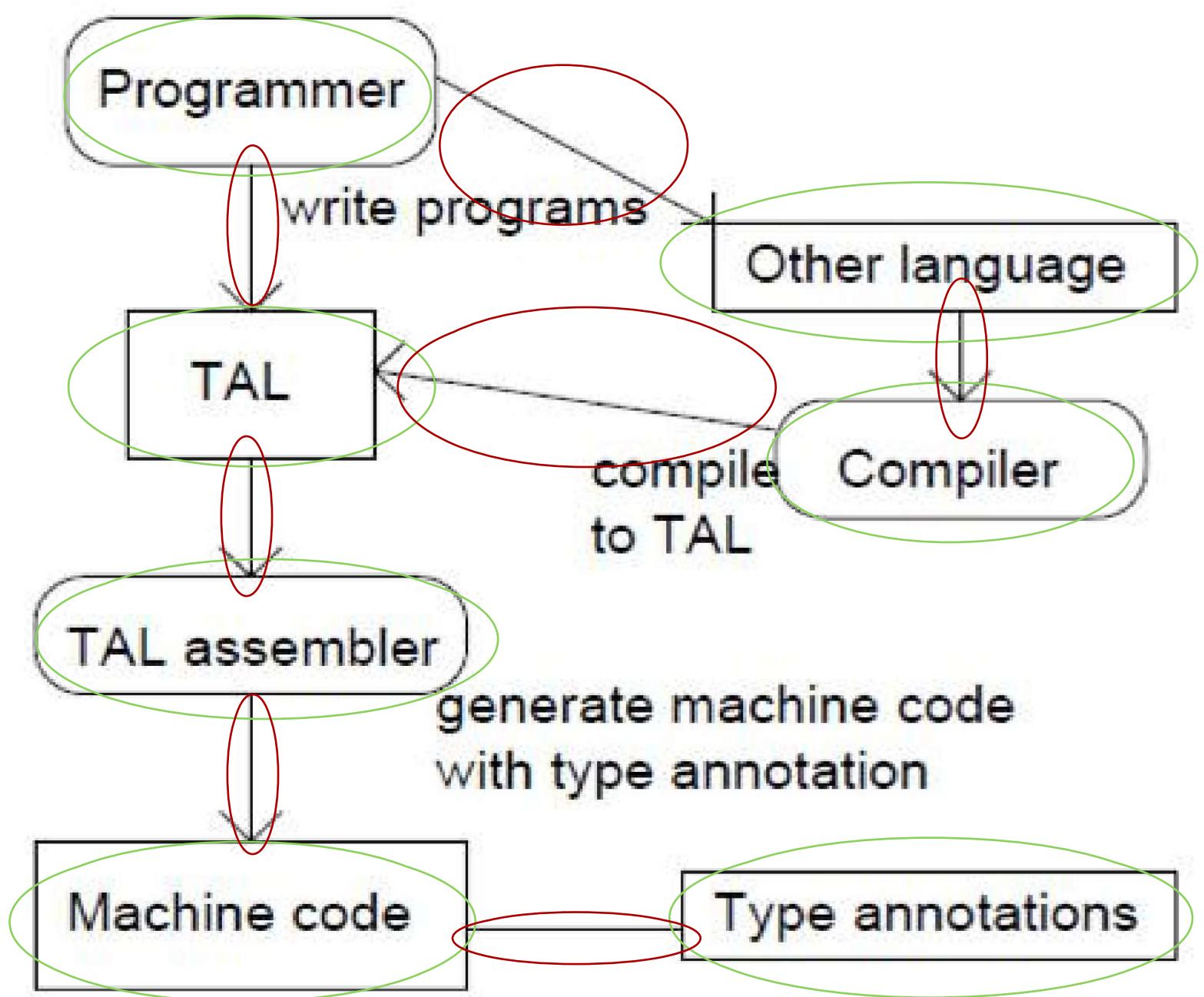
T.L.C.=TARGET
LANGUAGE CODE

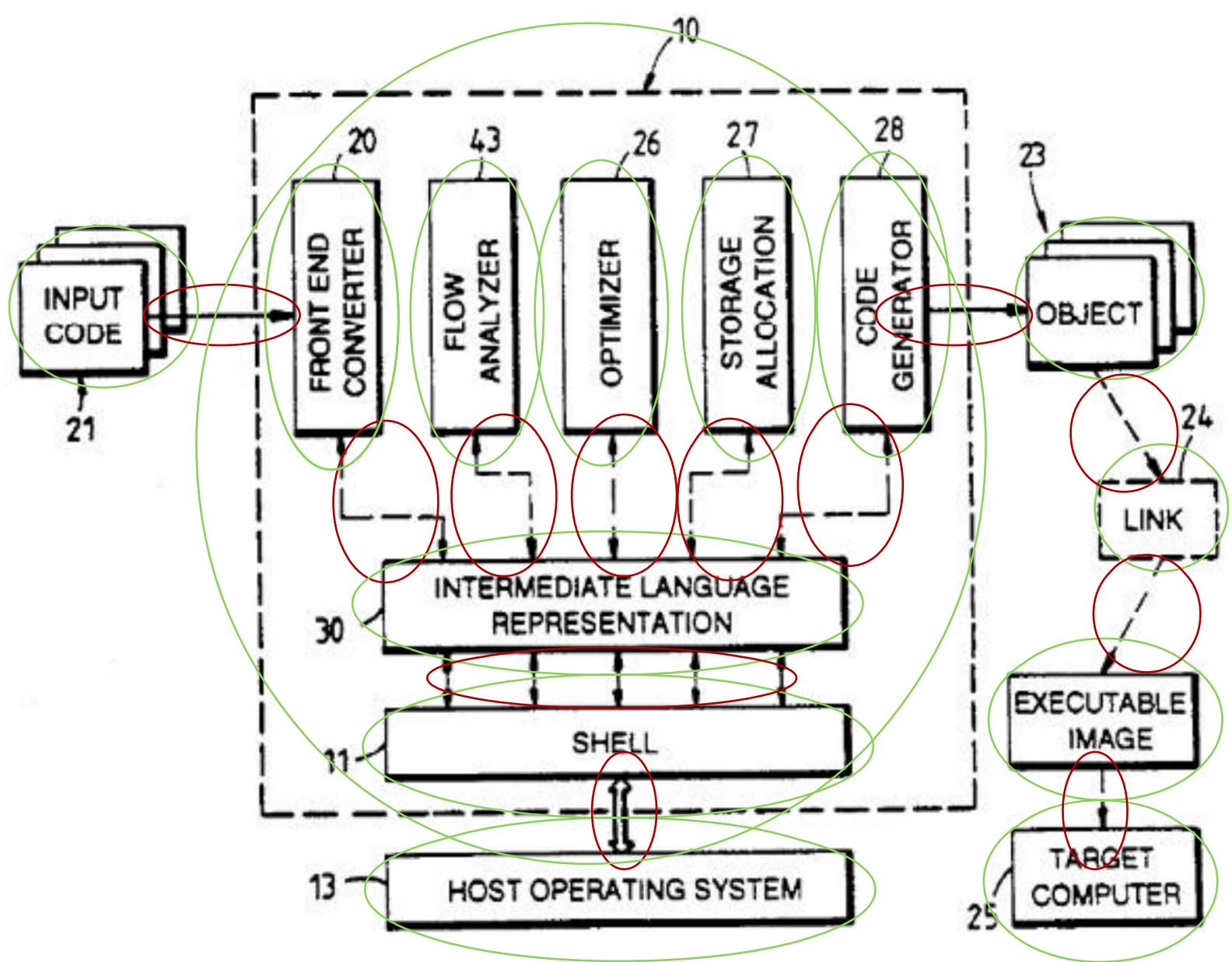
DECLARATIONS

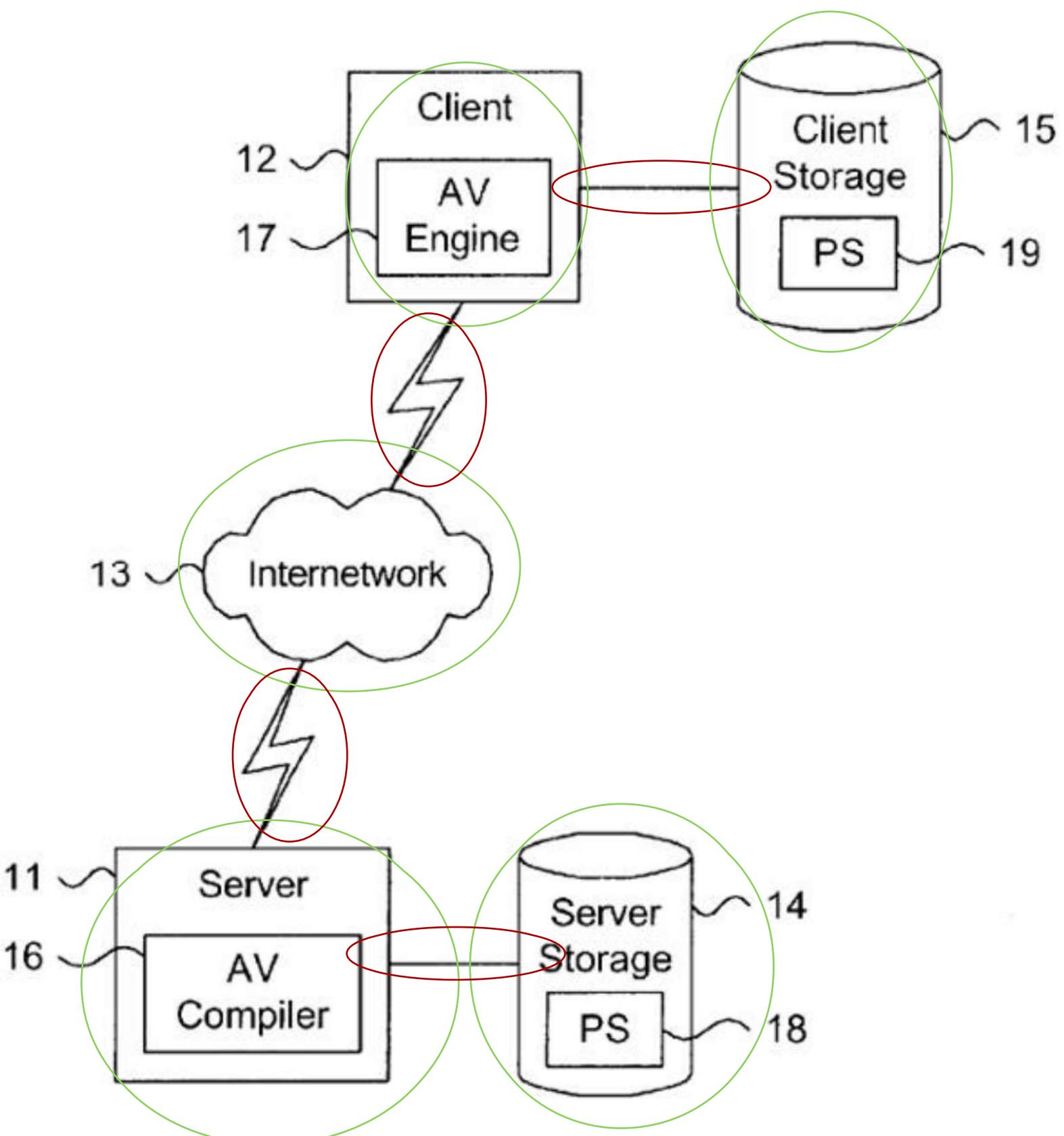
RE-STRUCTURED S.L.C.

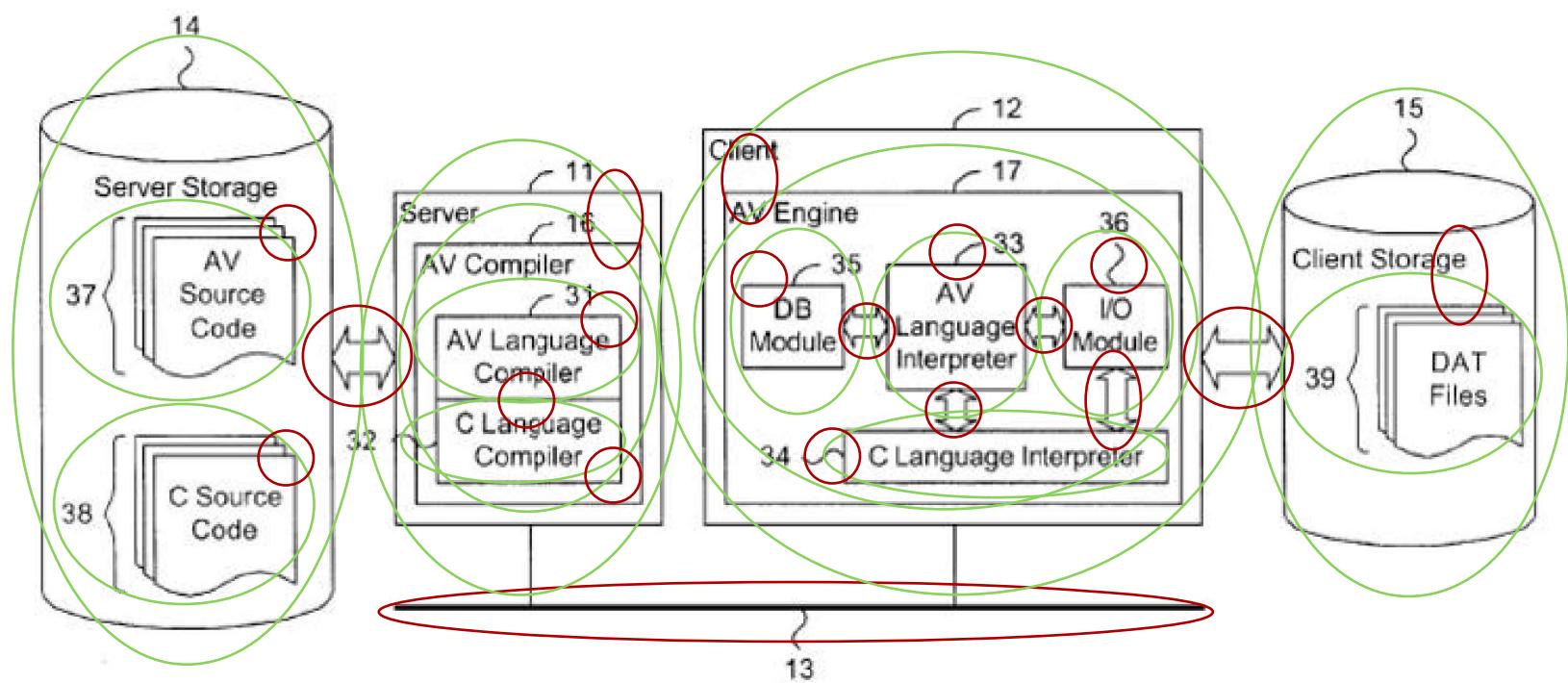
FIG. 3

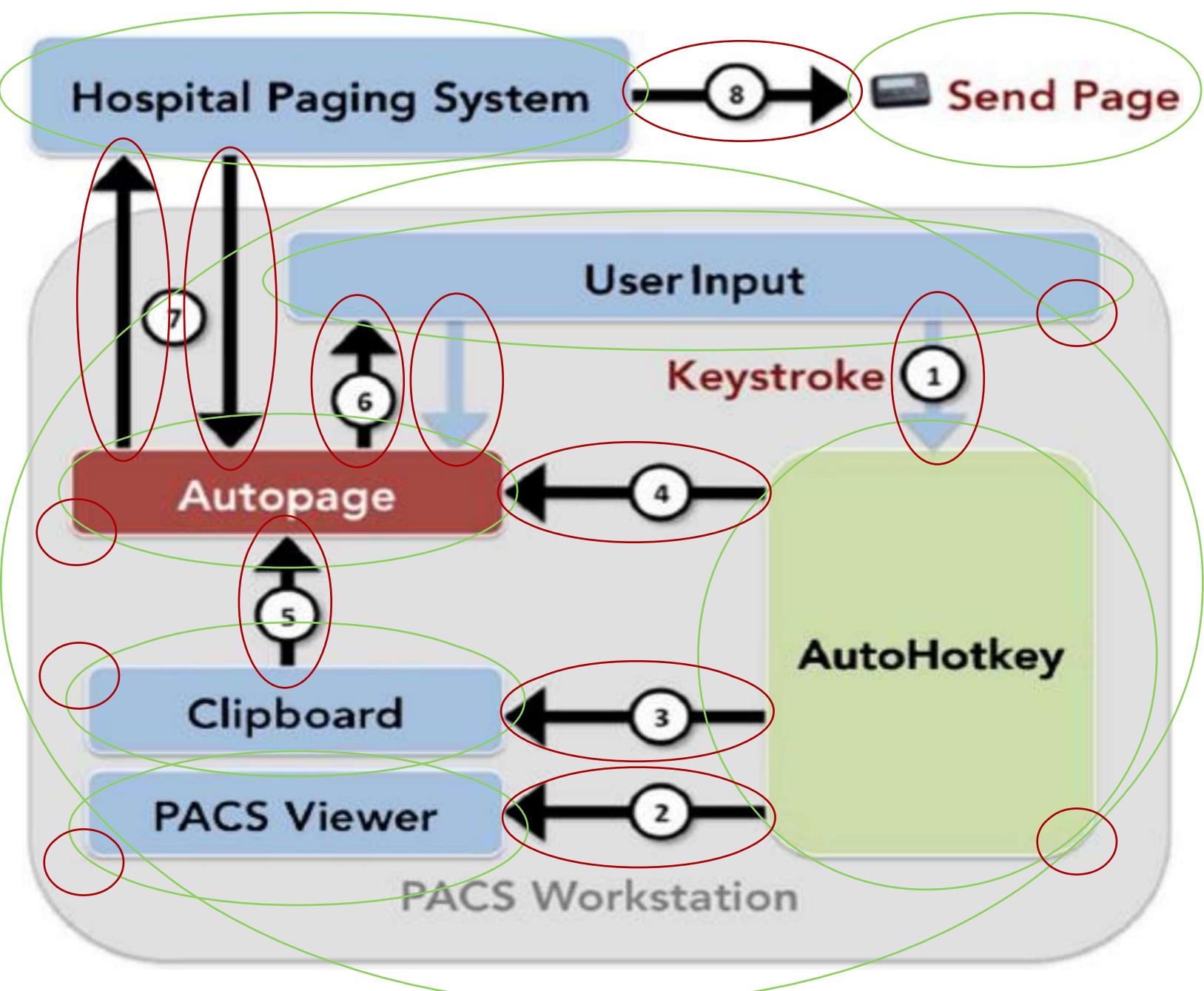


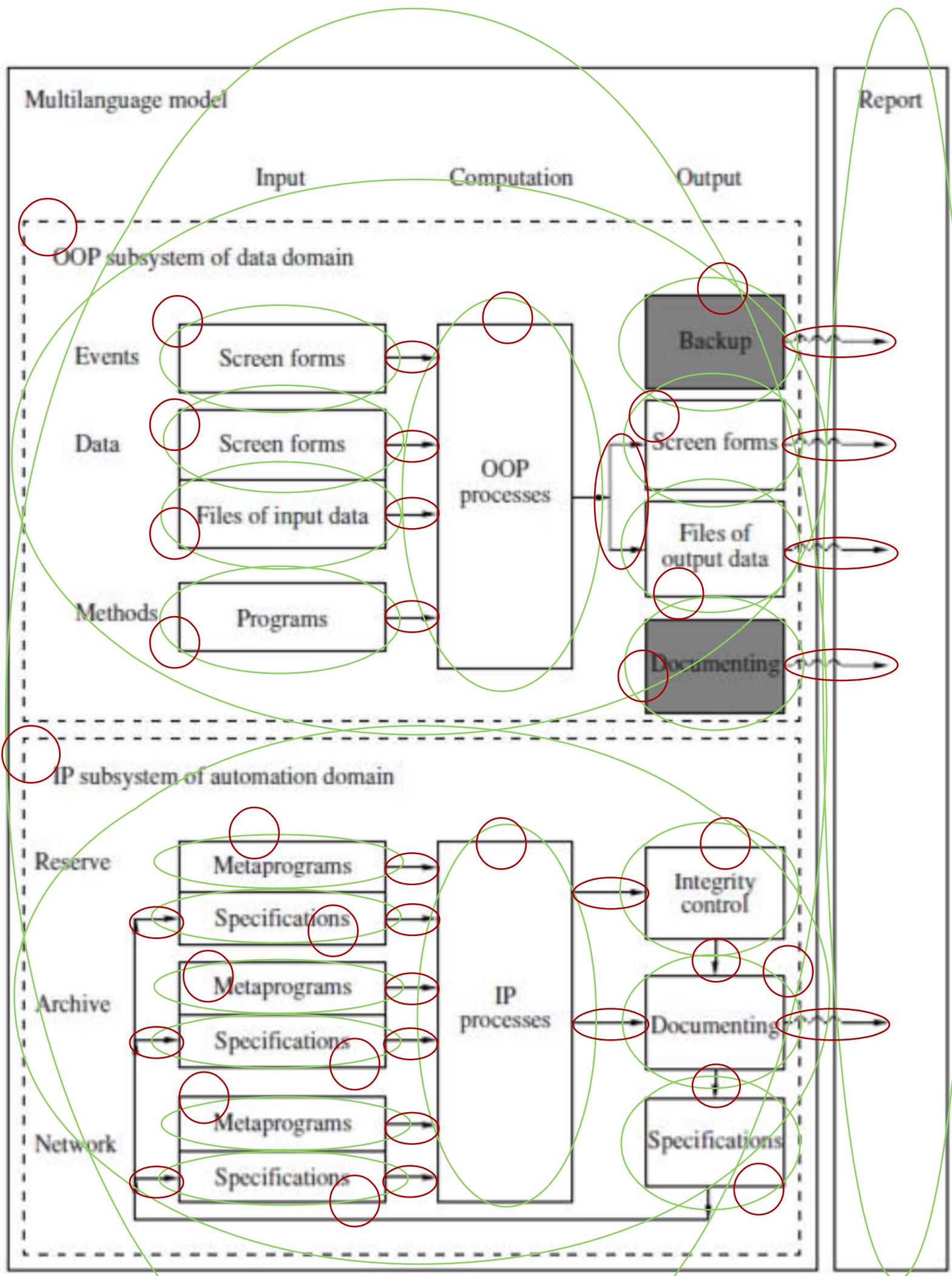


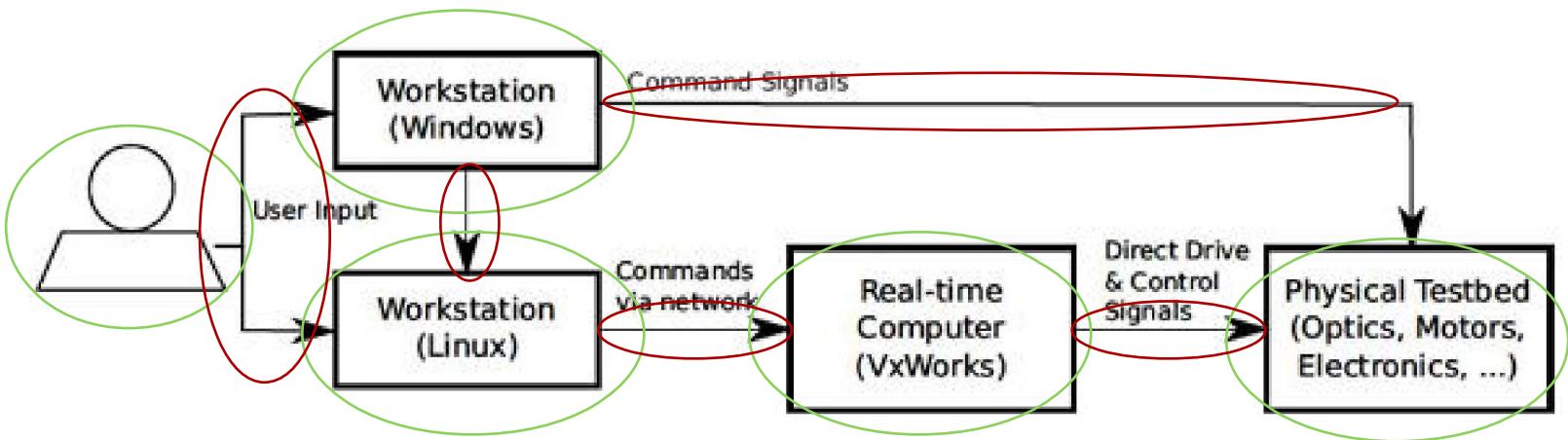


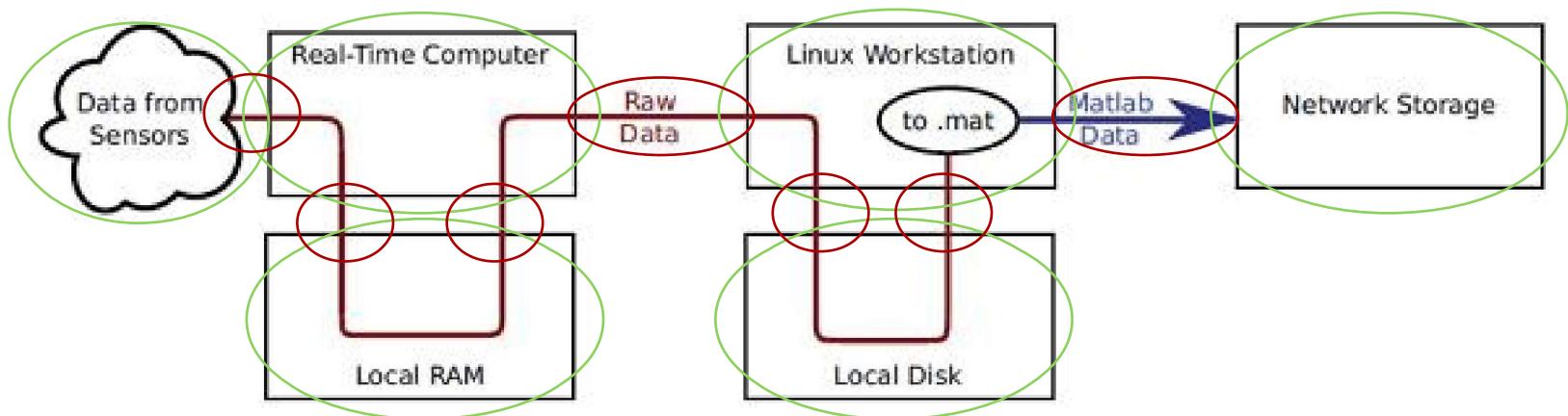


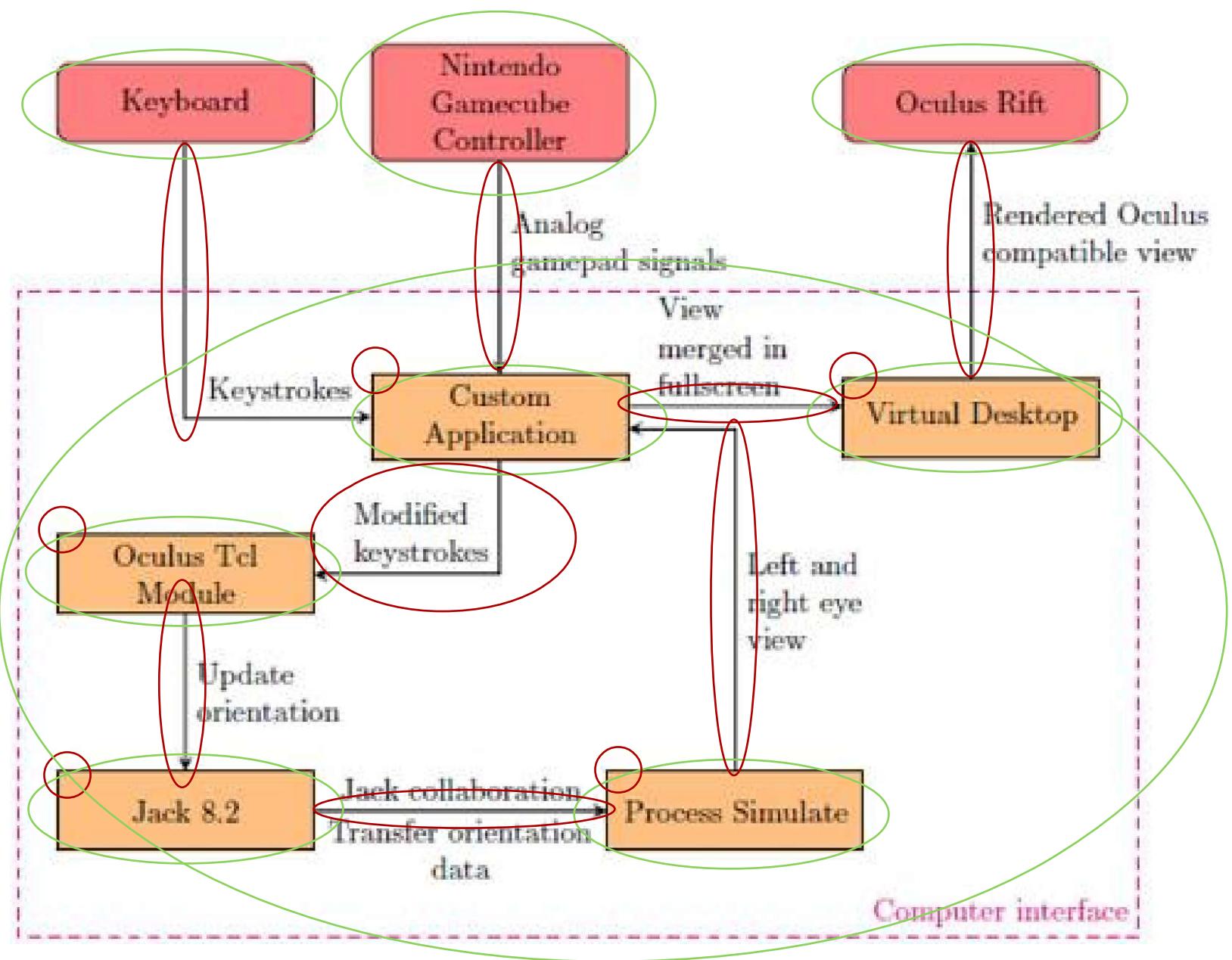


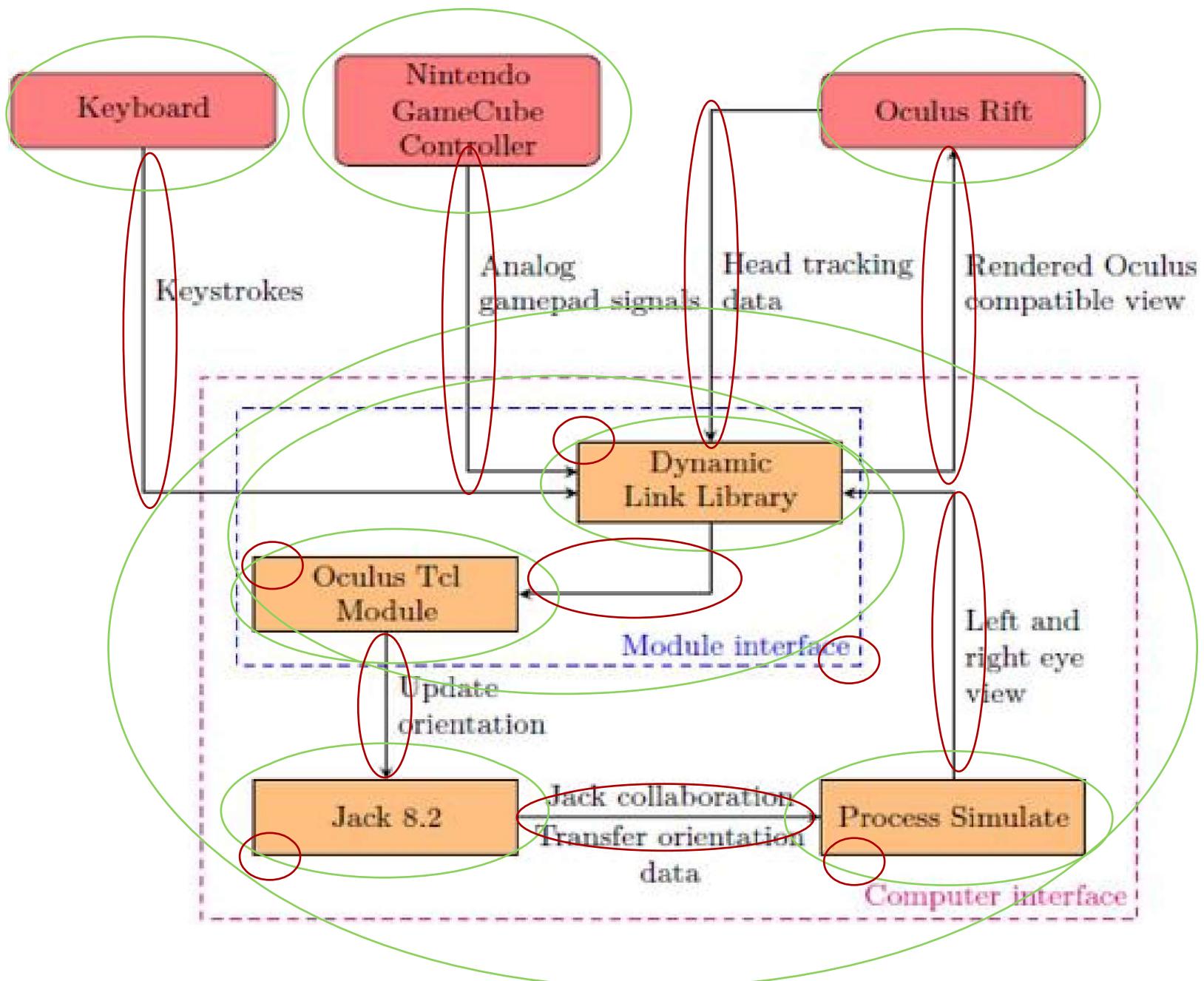


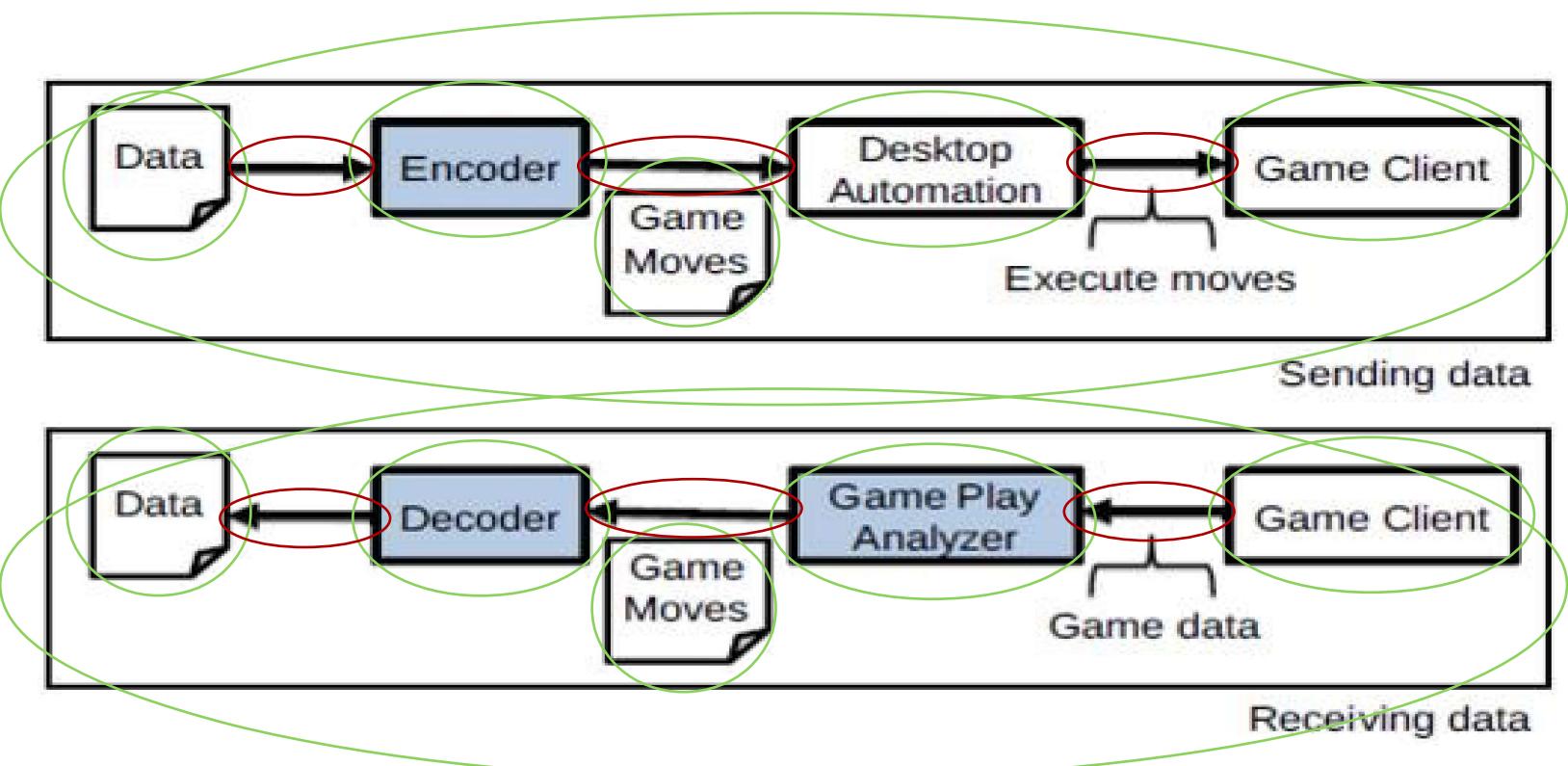


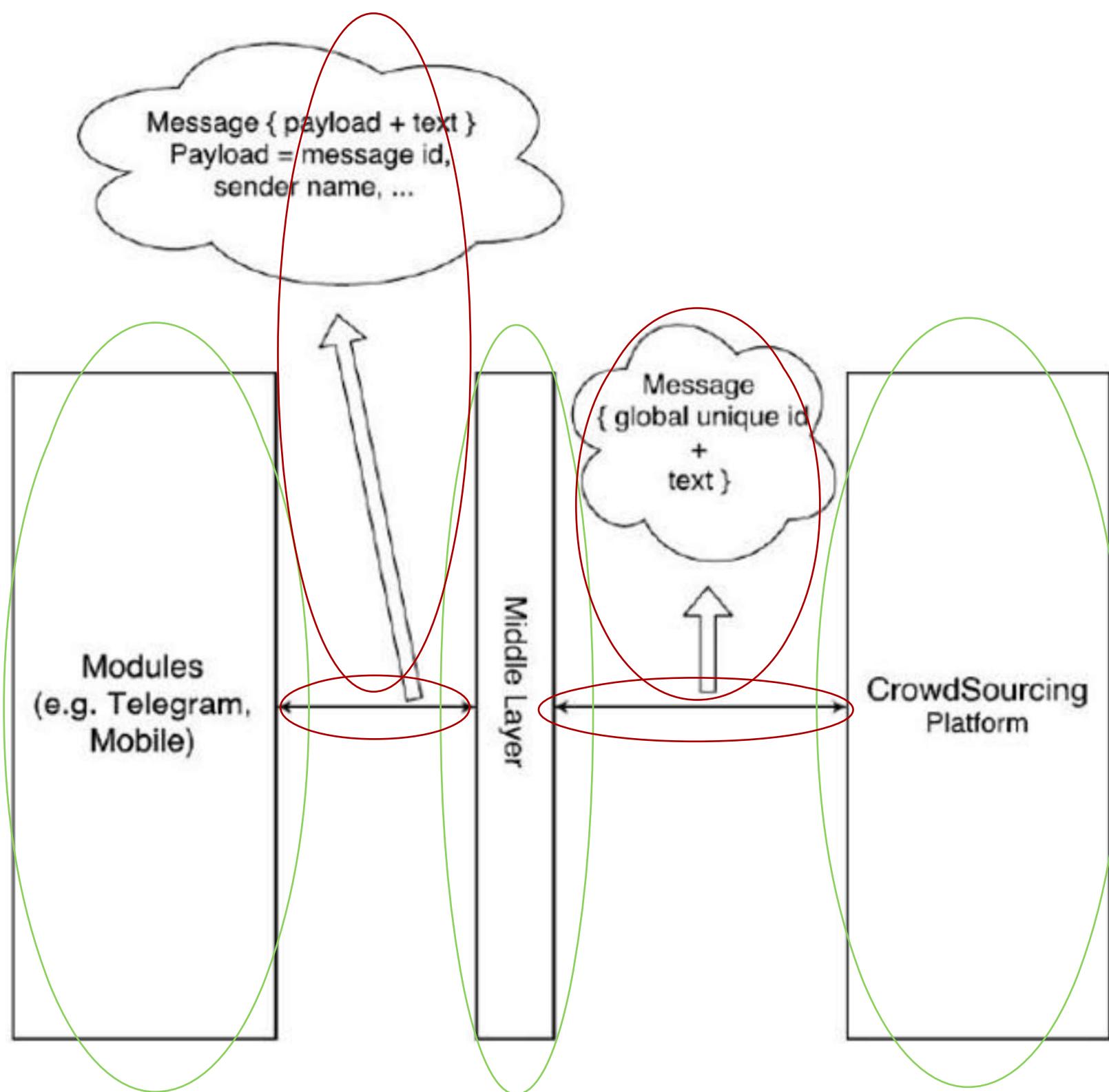


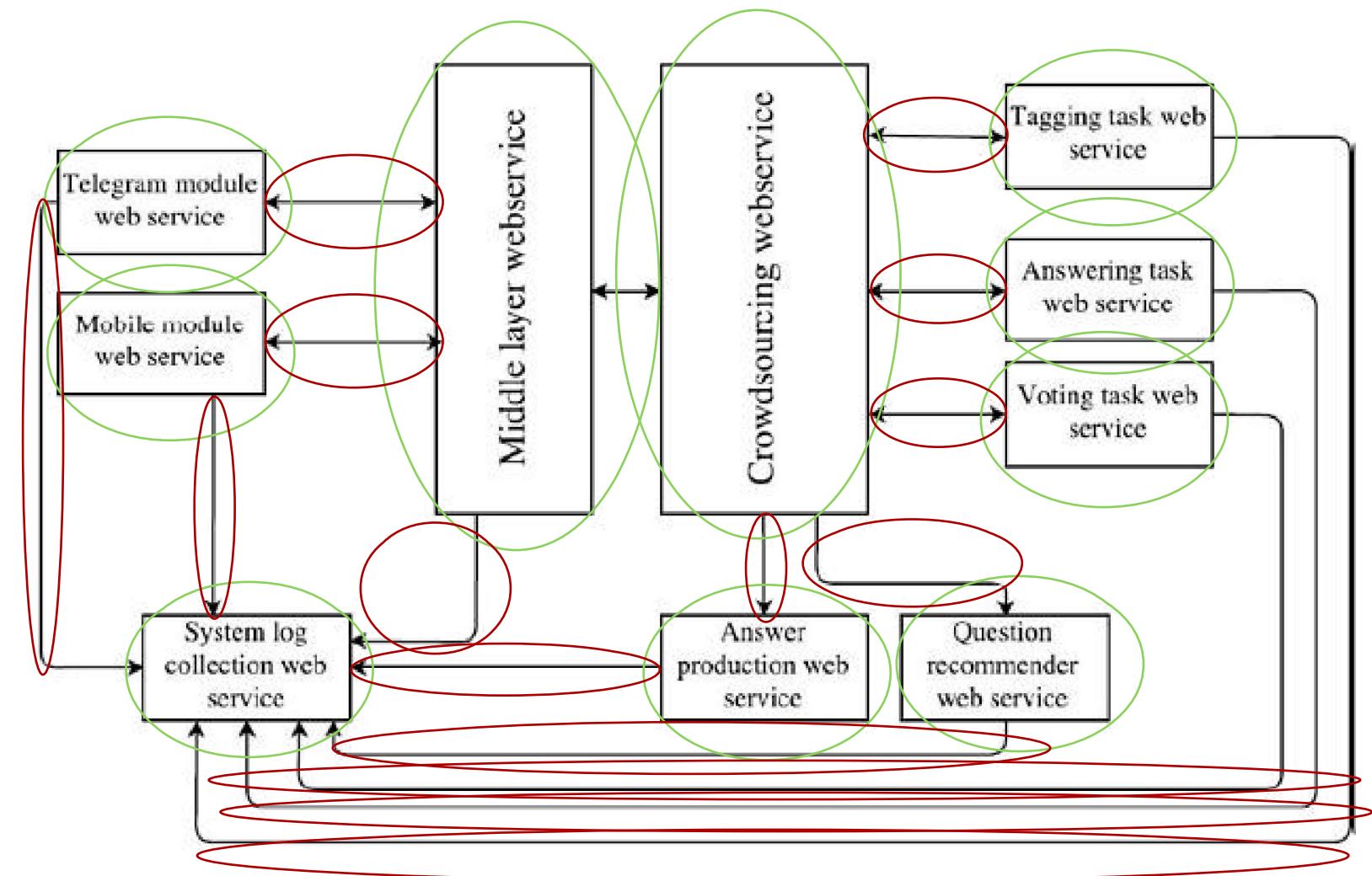


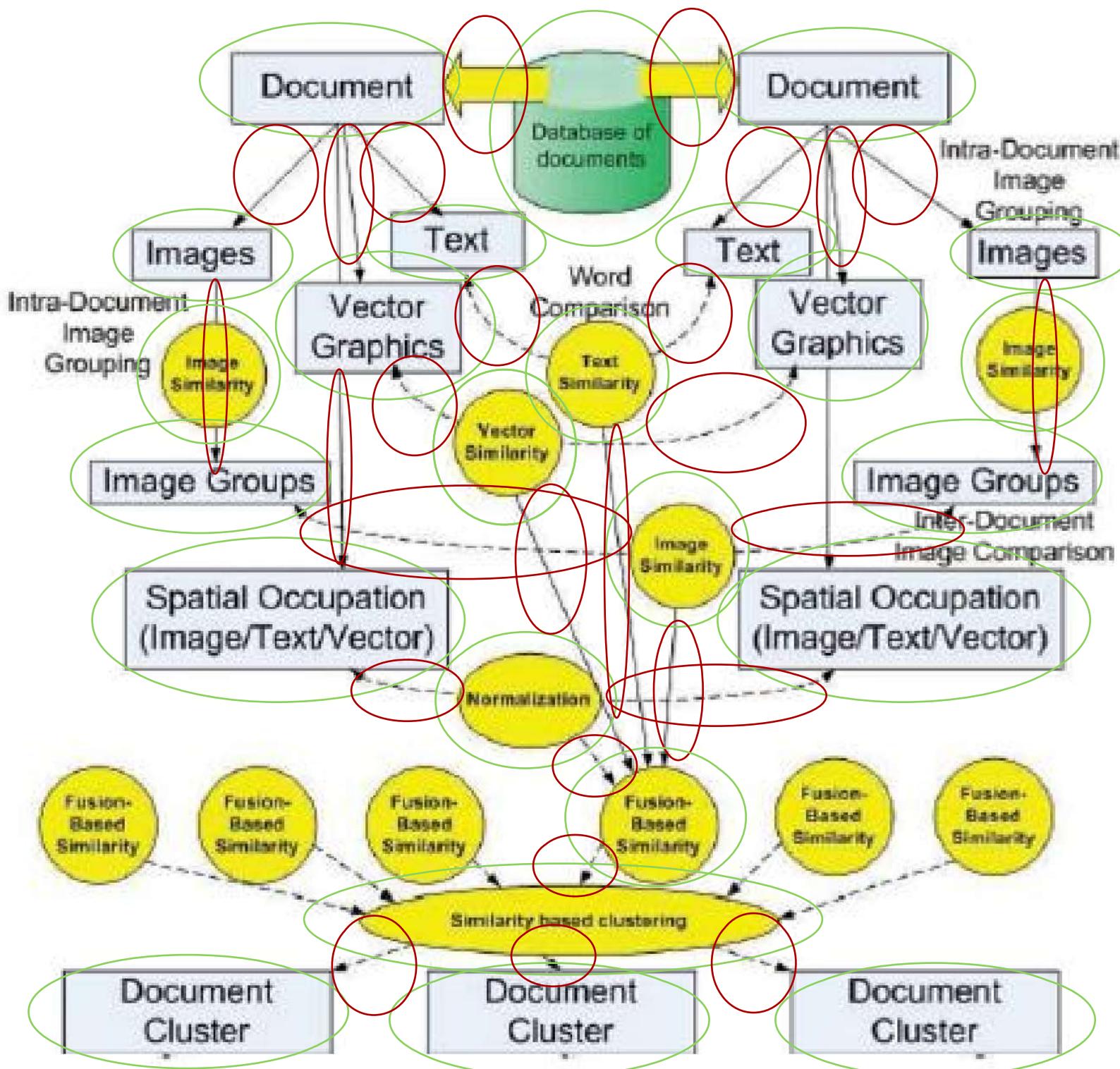


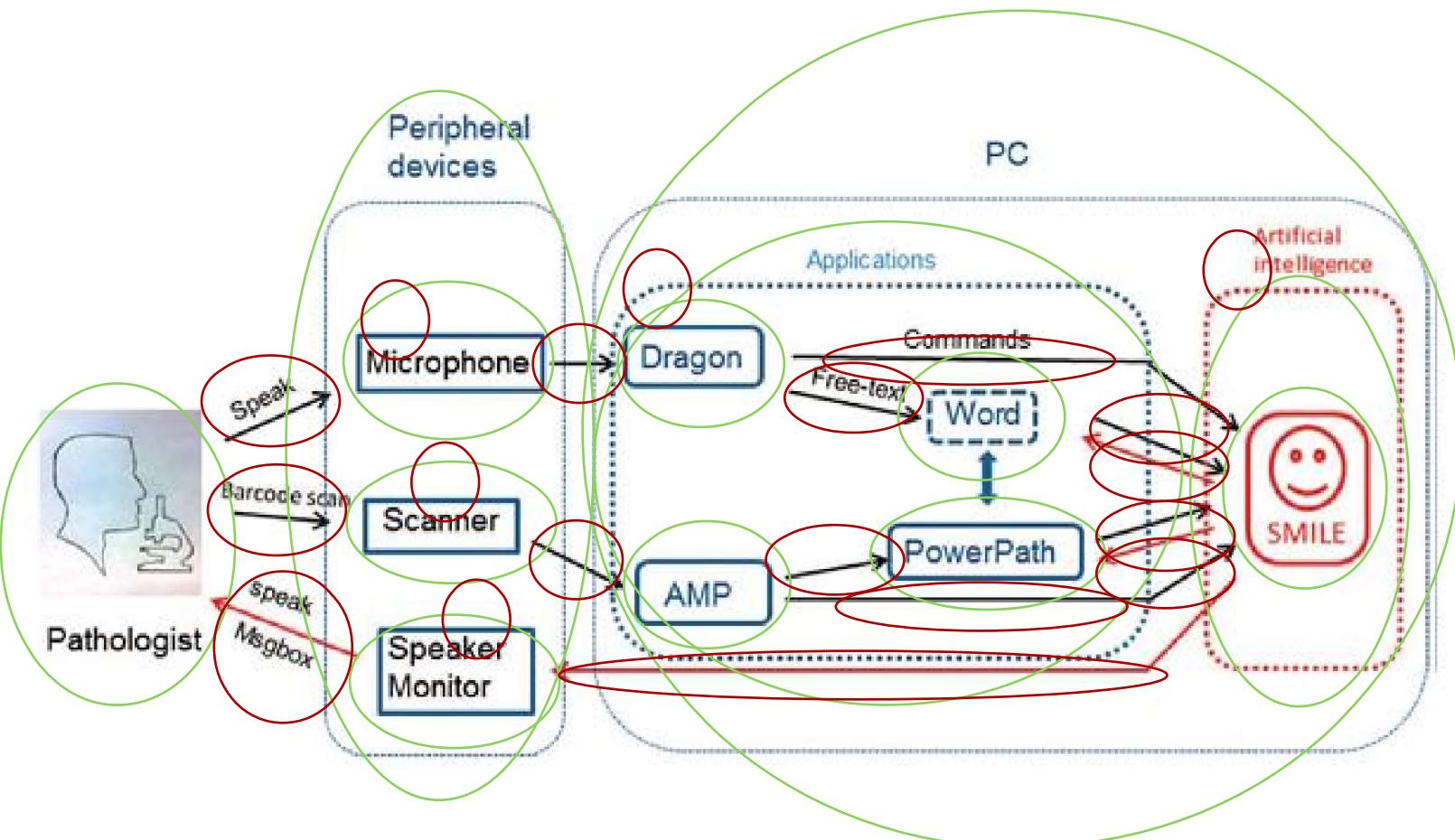


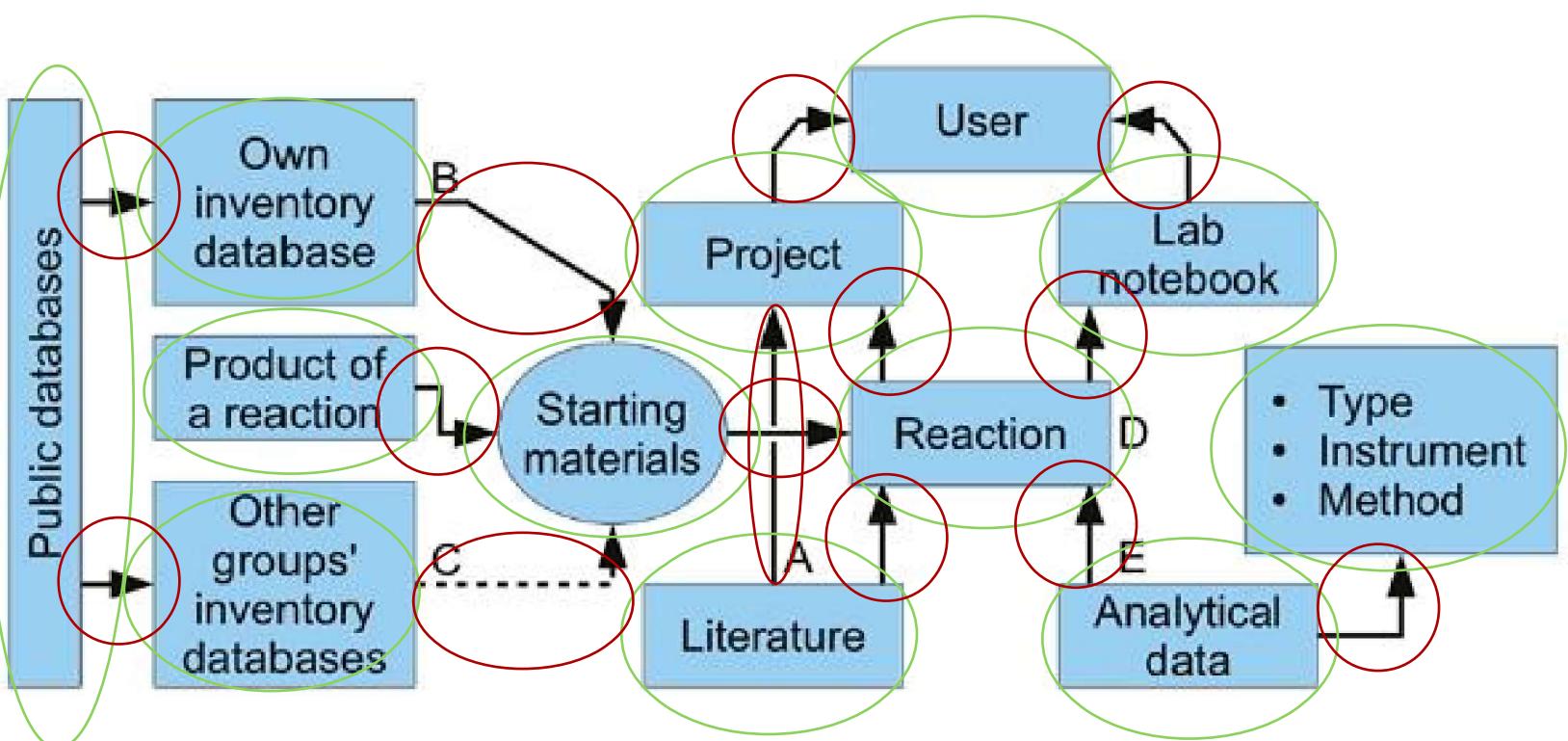


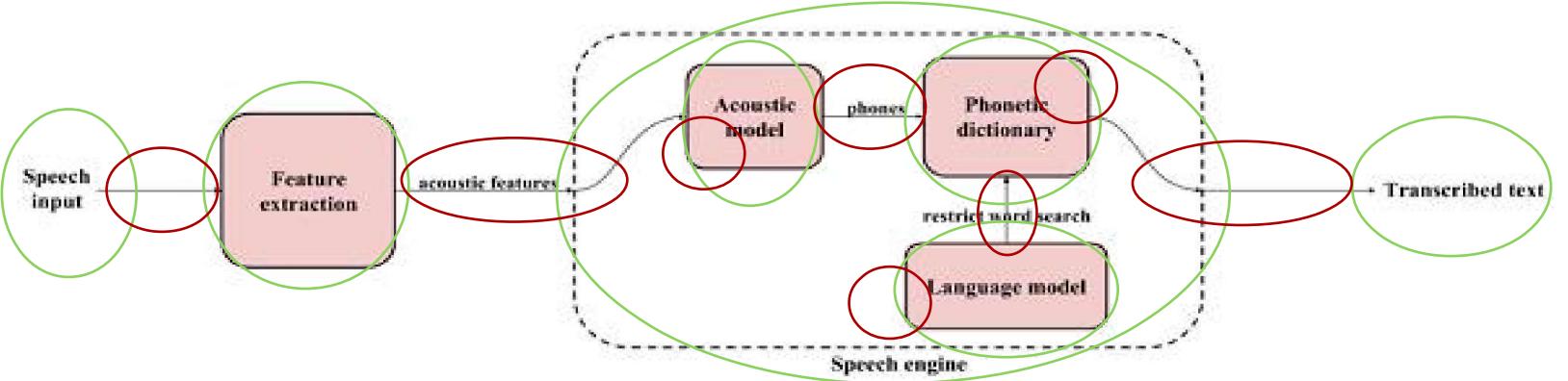


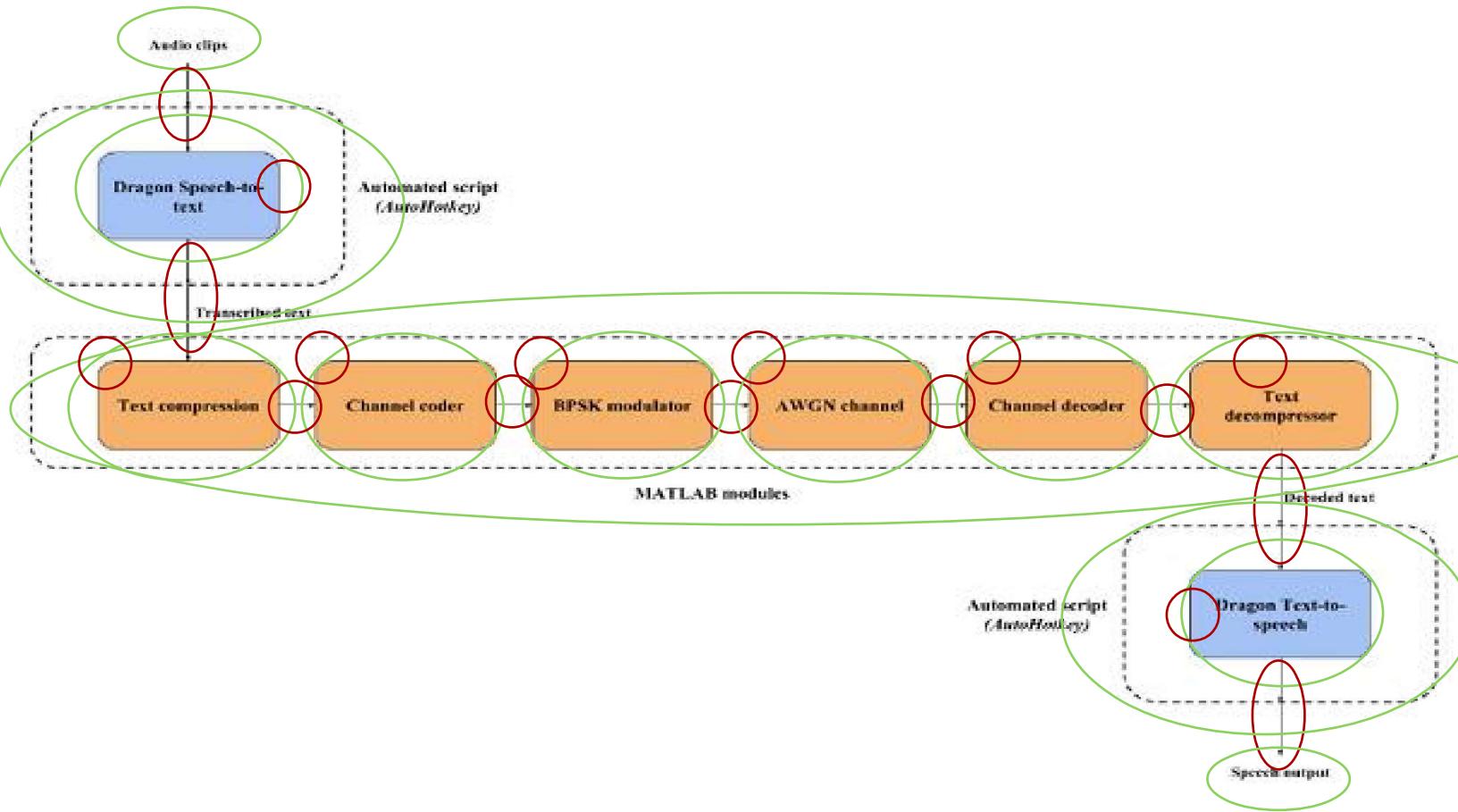


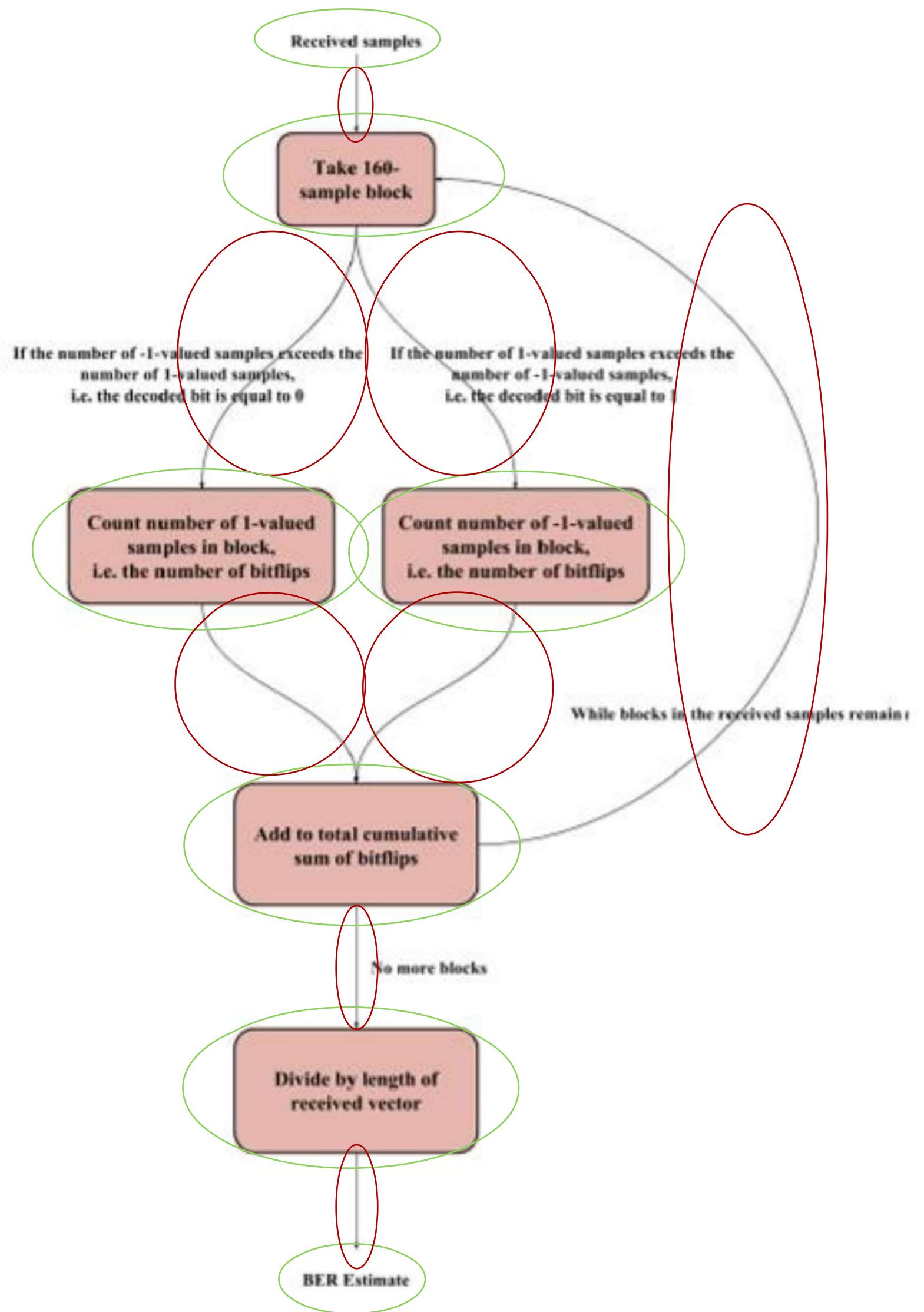


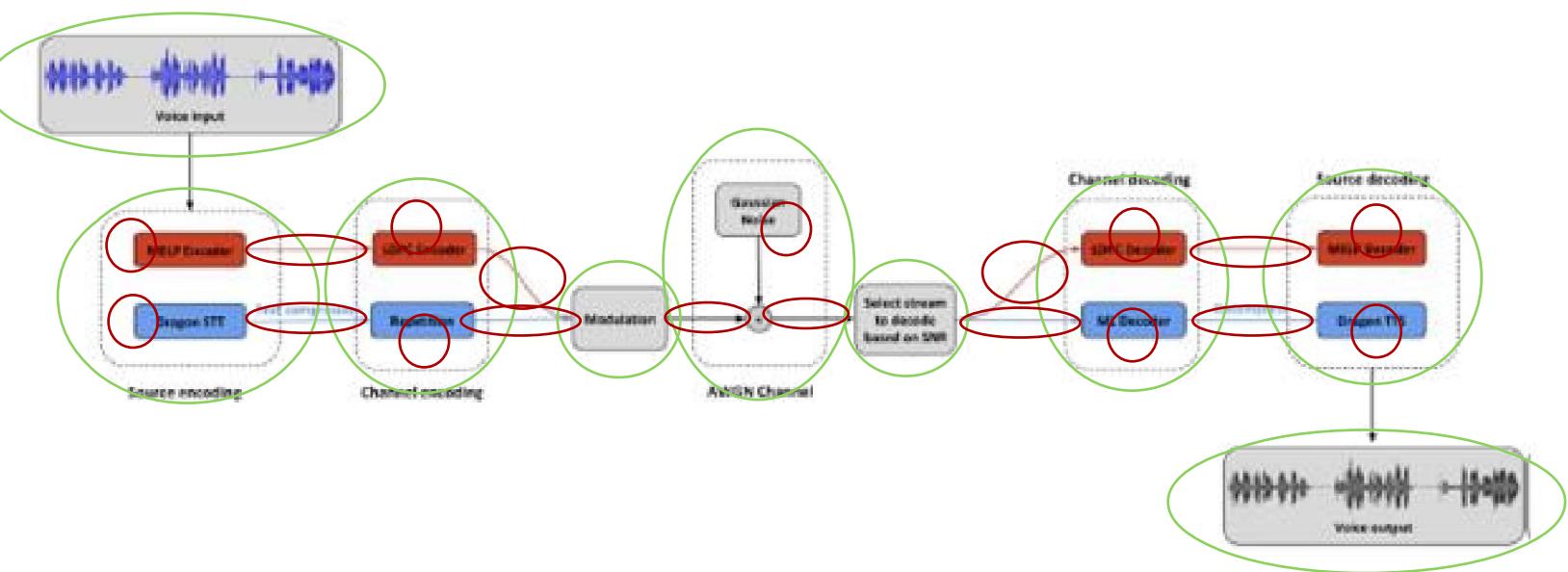


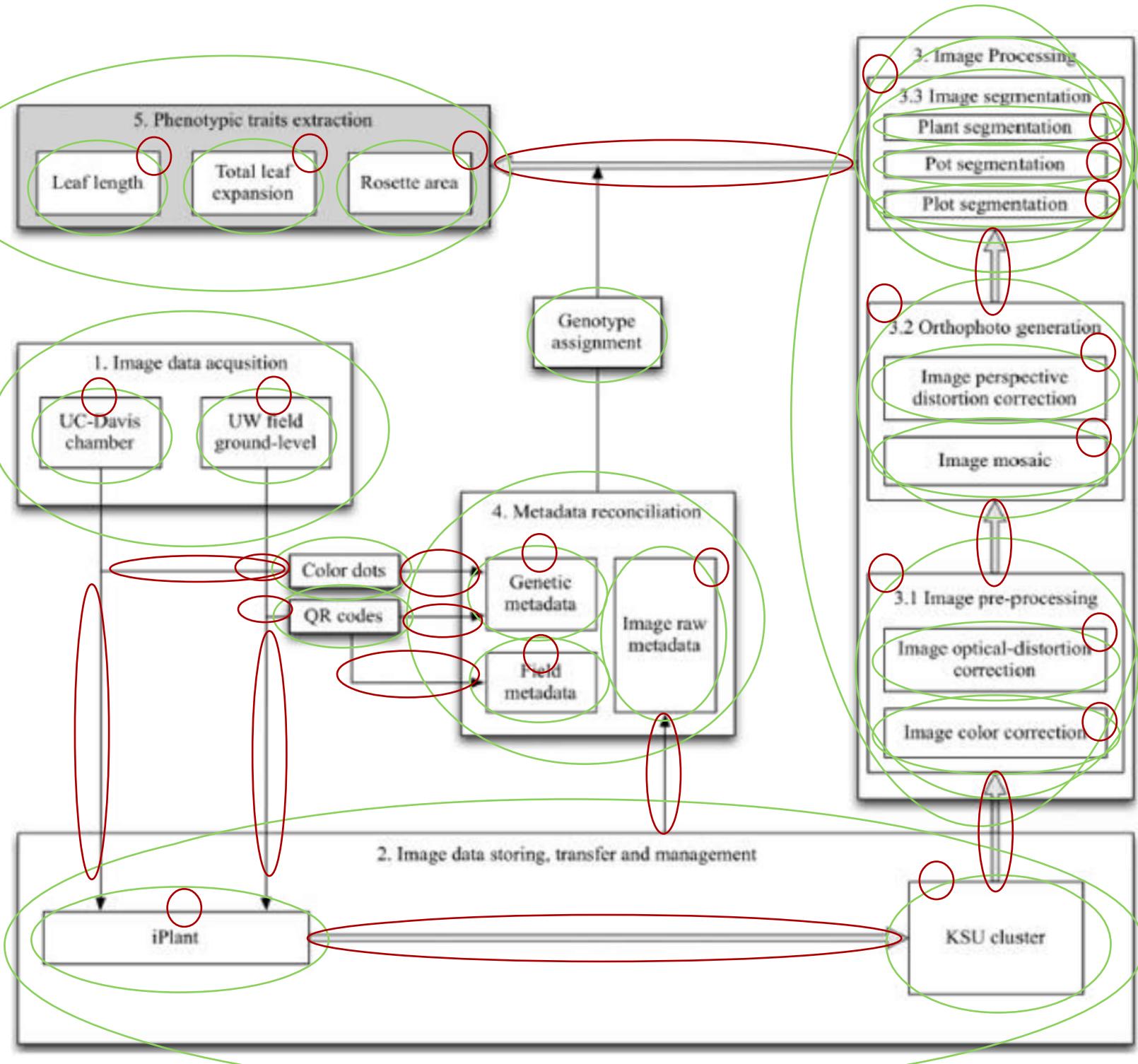


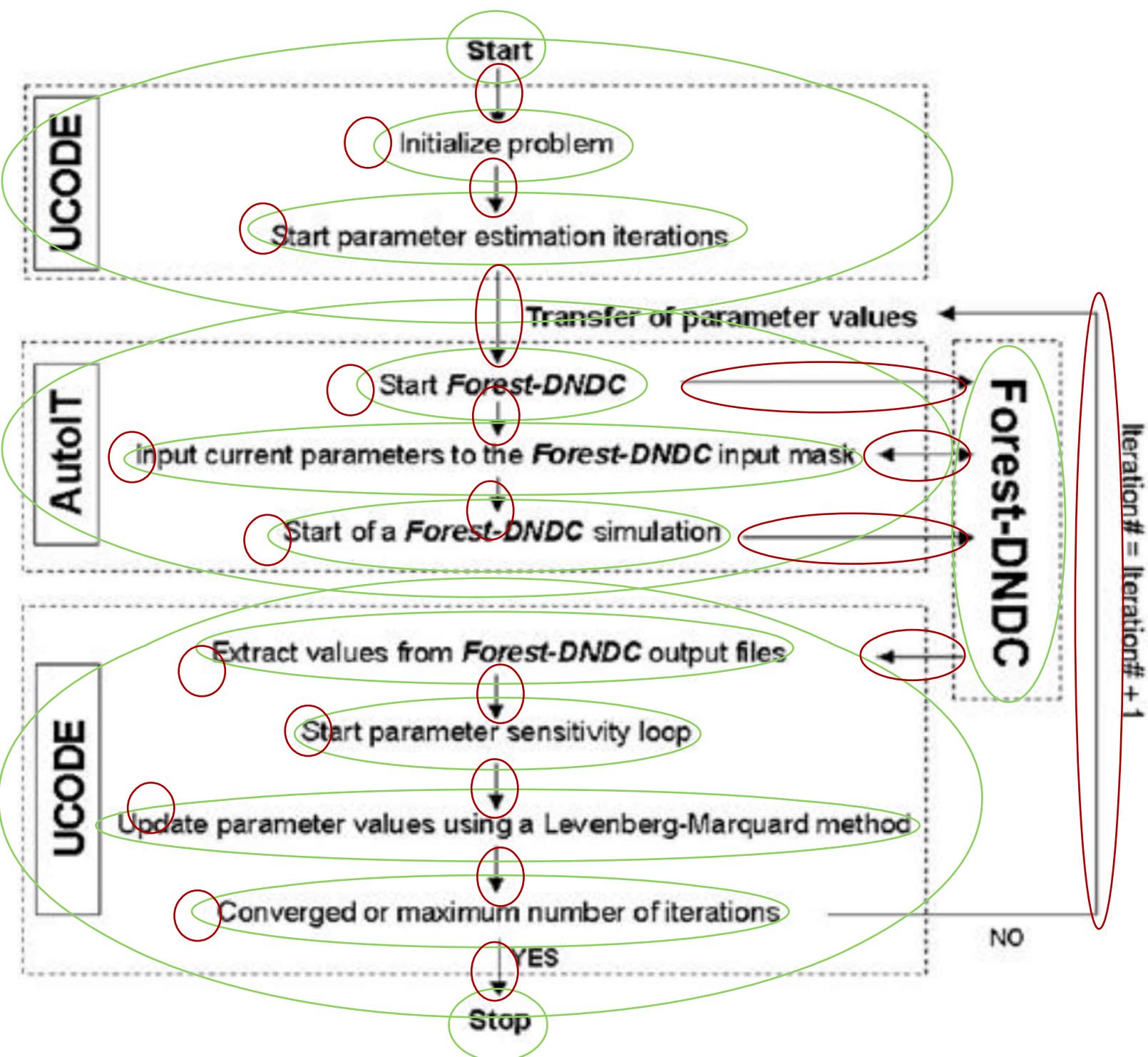


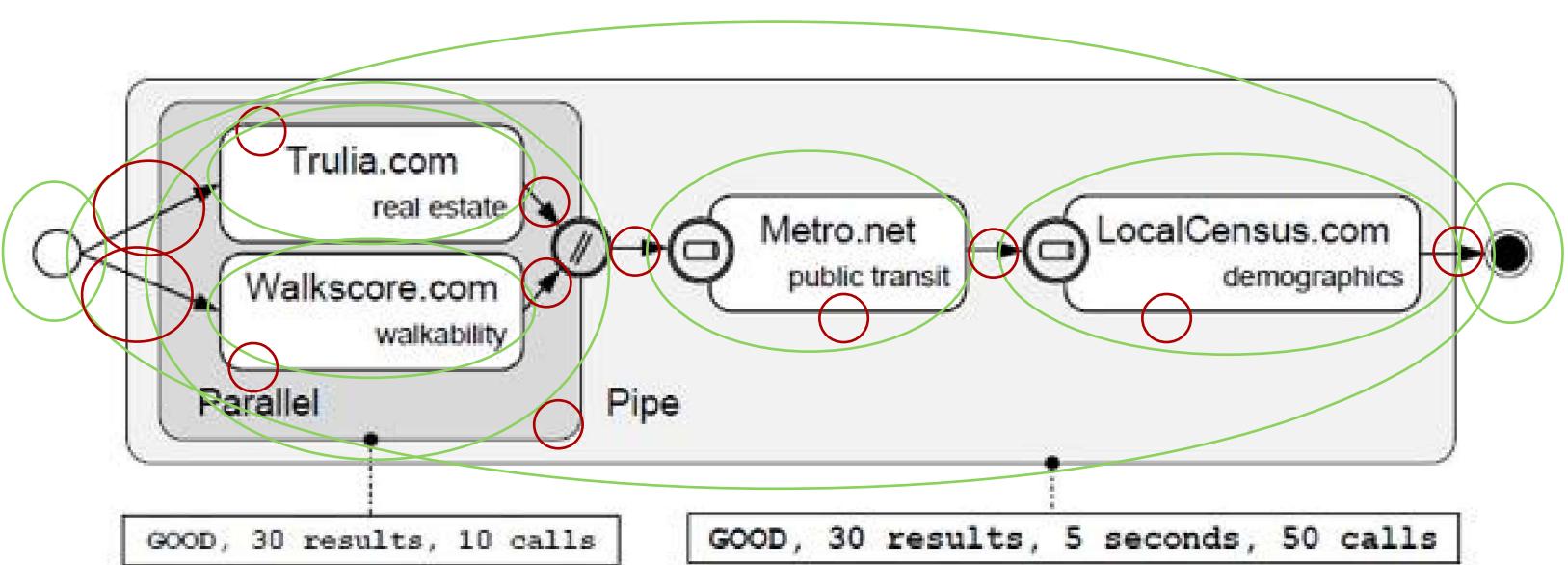






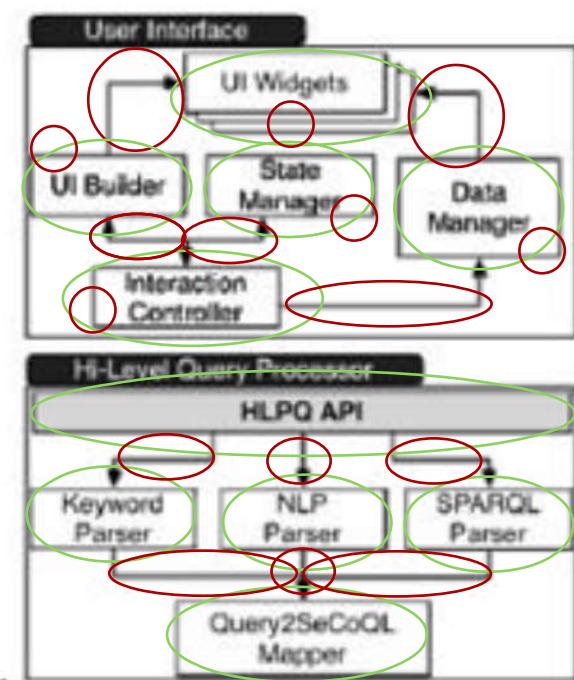






USER INTERACTION

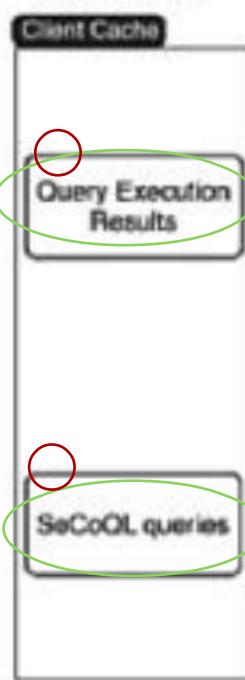
PROCESSING



Client
Server

REPOSITORIES

Transient



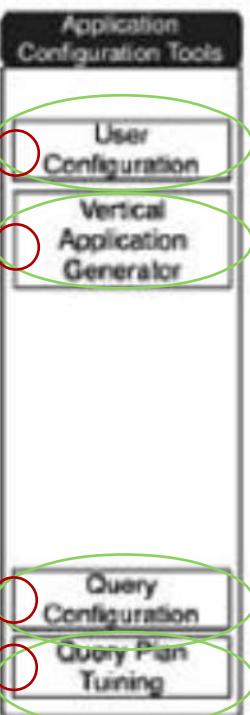
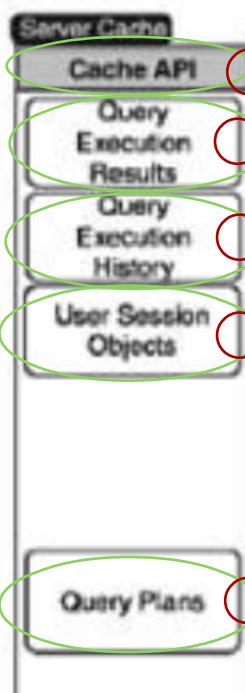
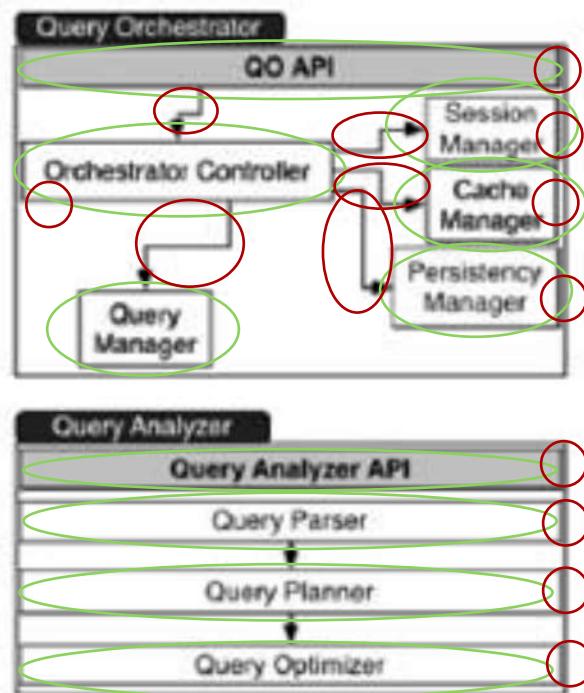
Persistent



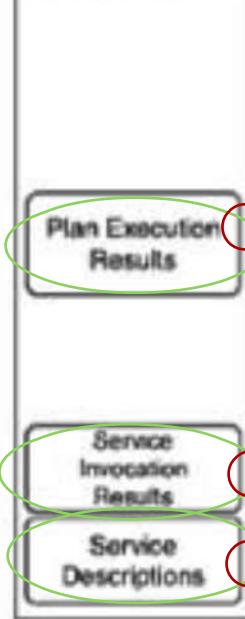
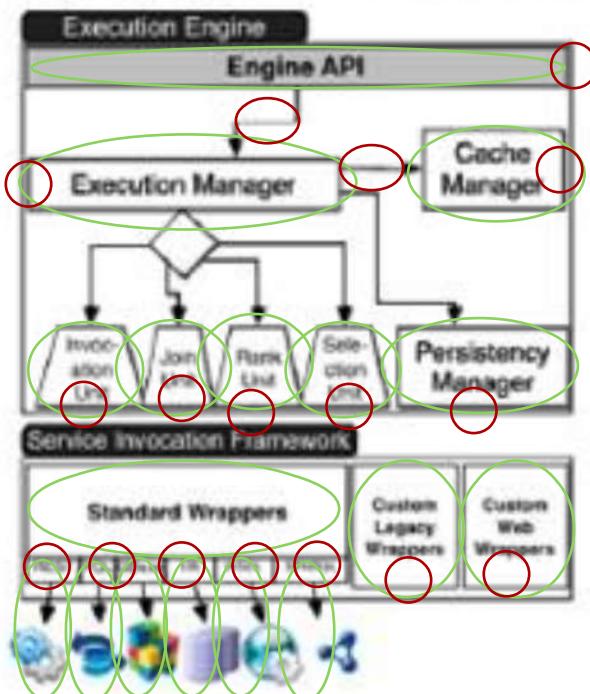
TOOLS

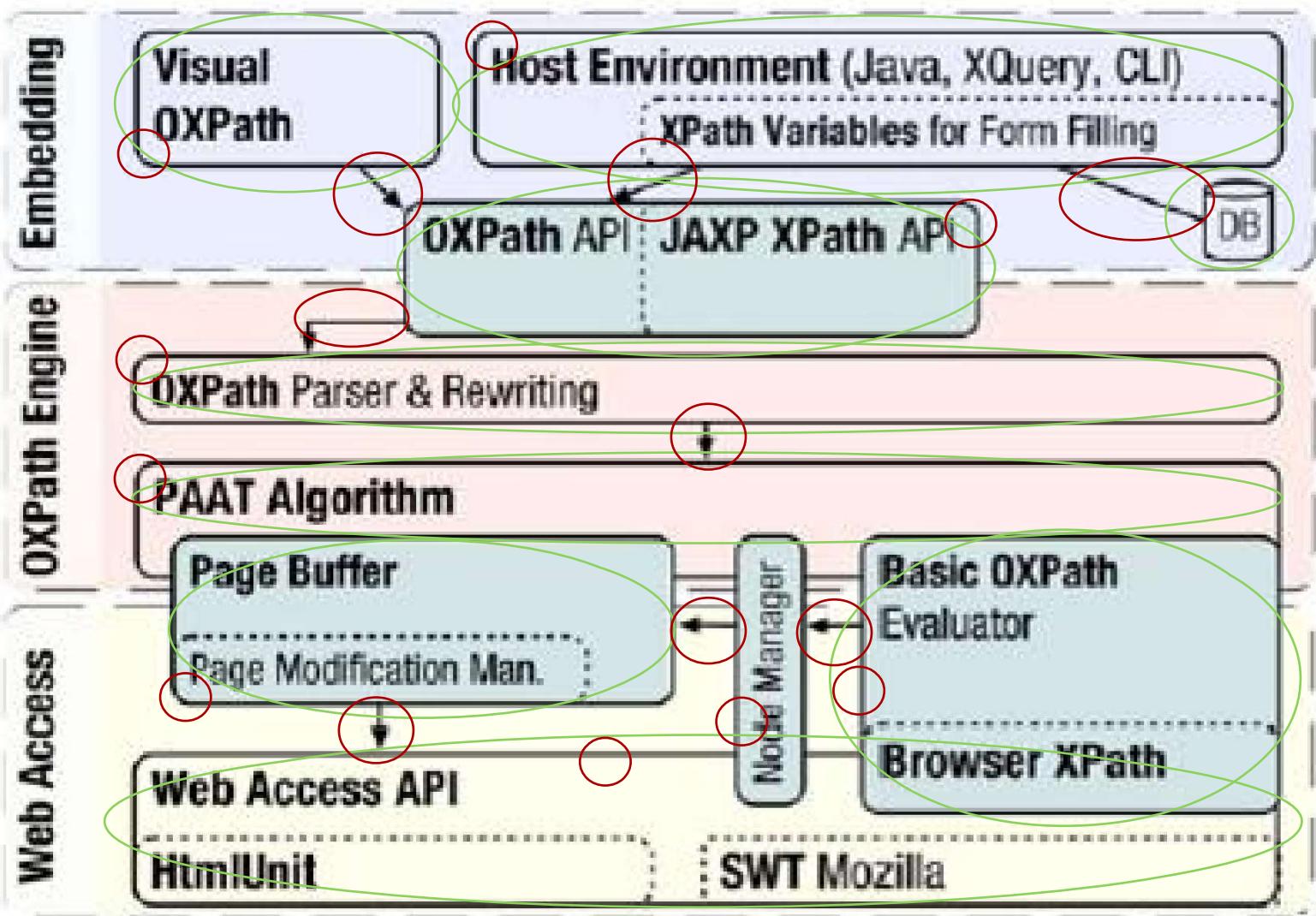


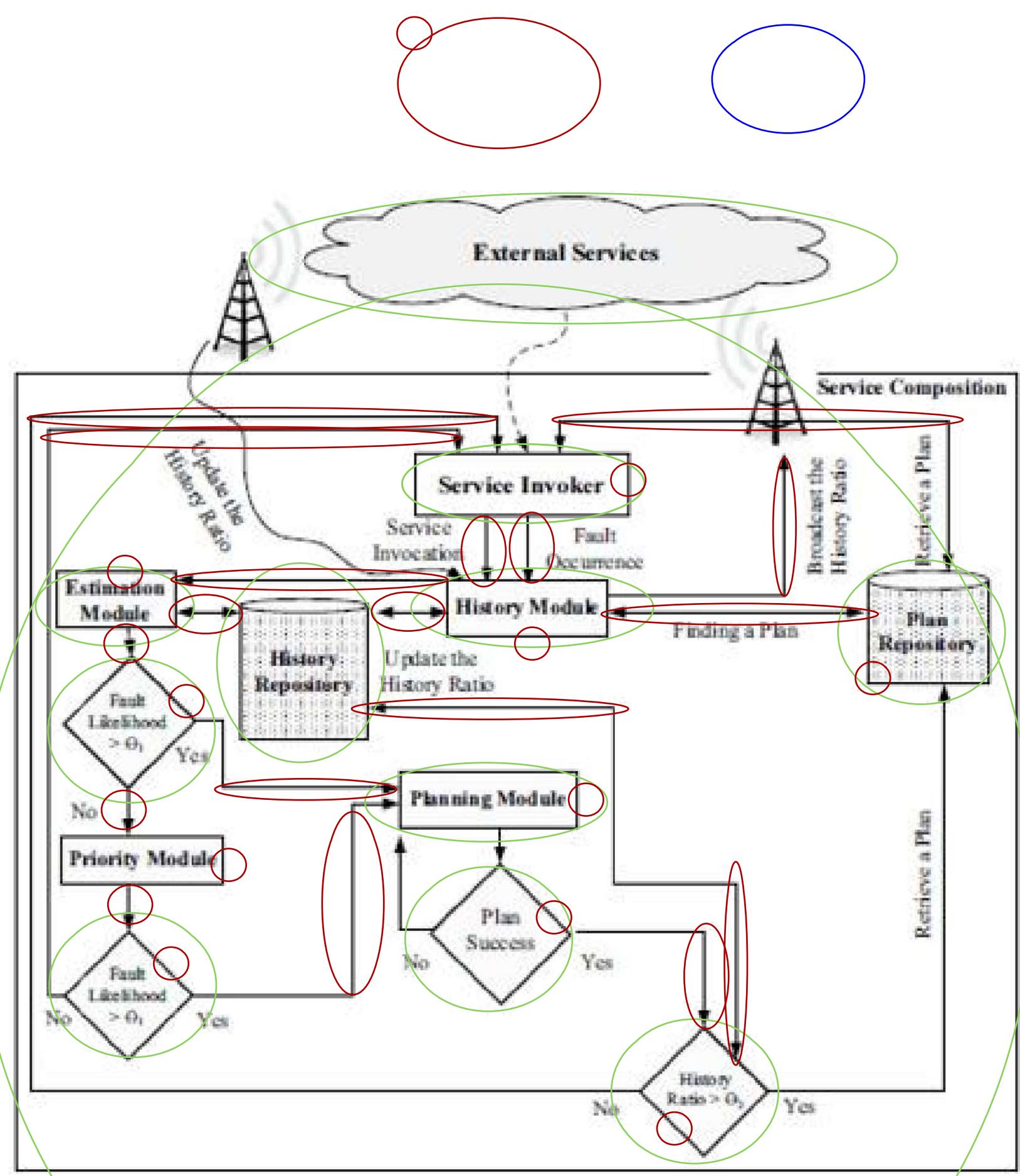
CONTROL

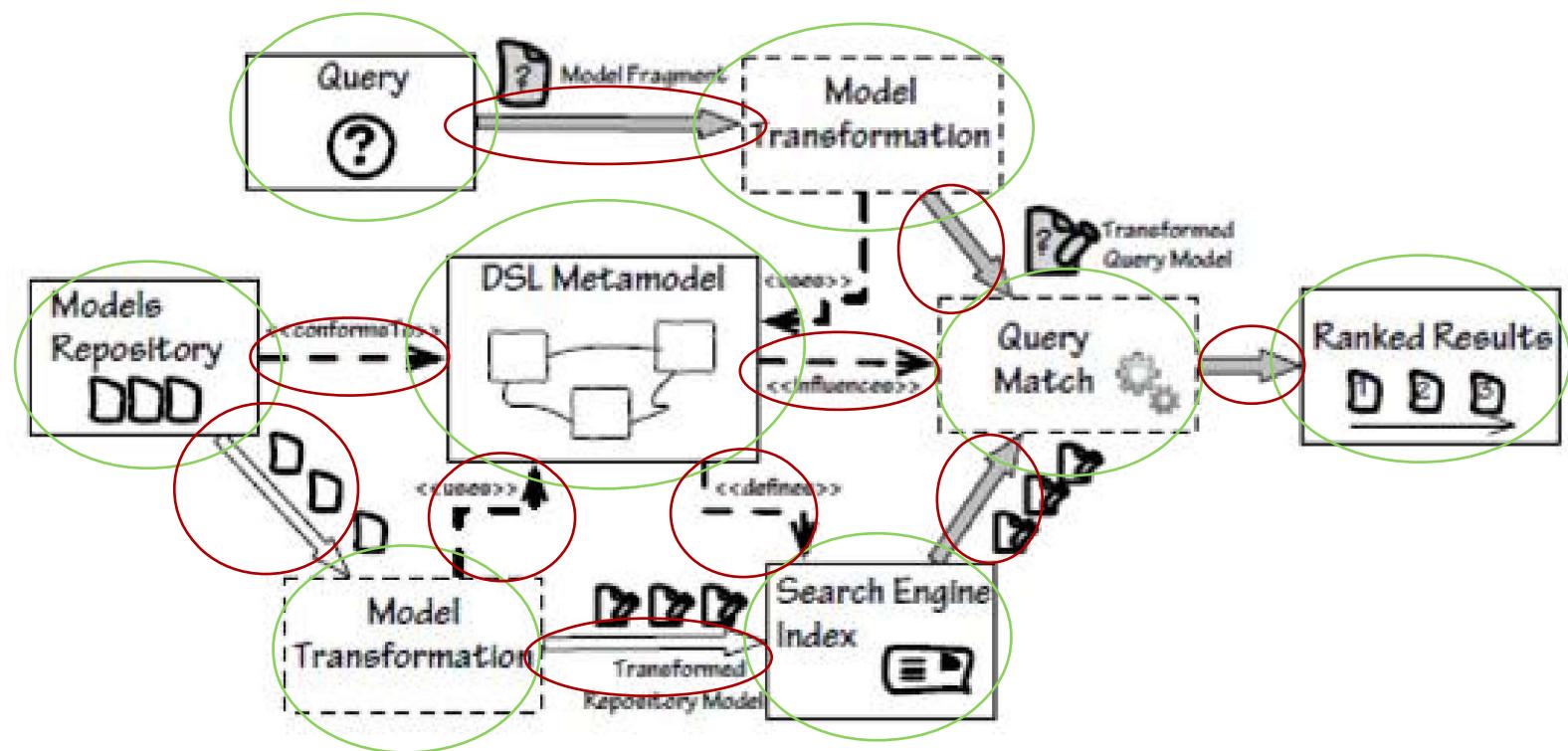


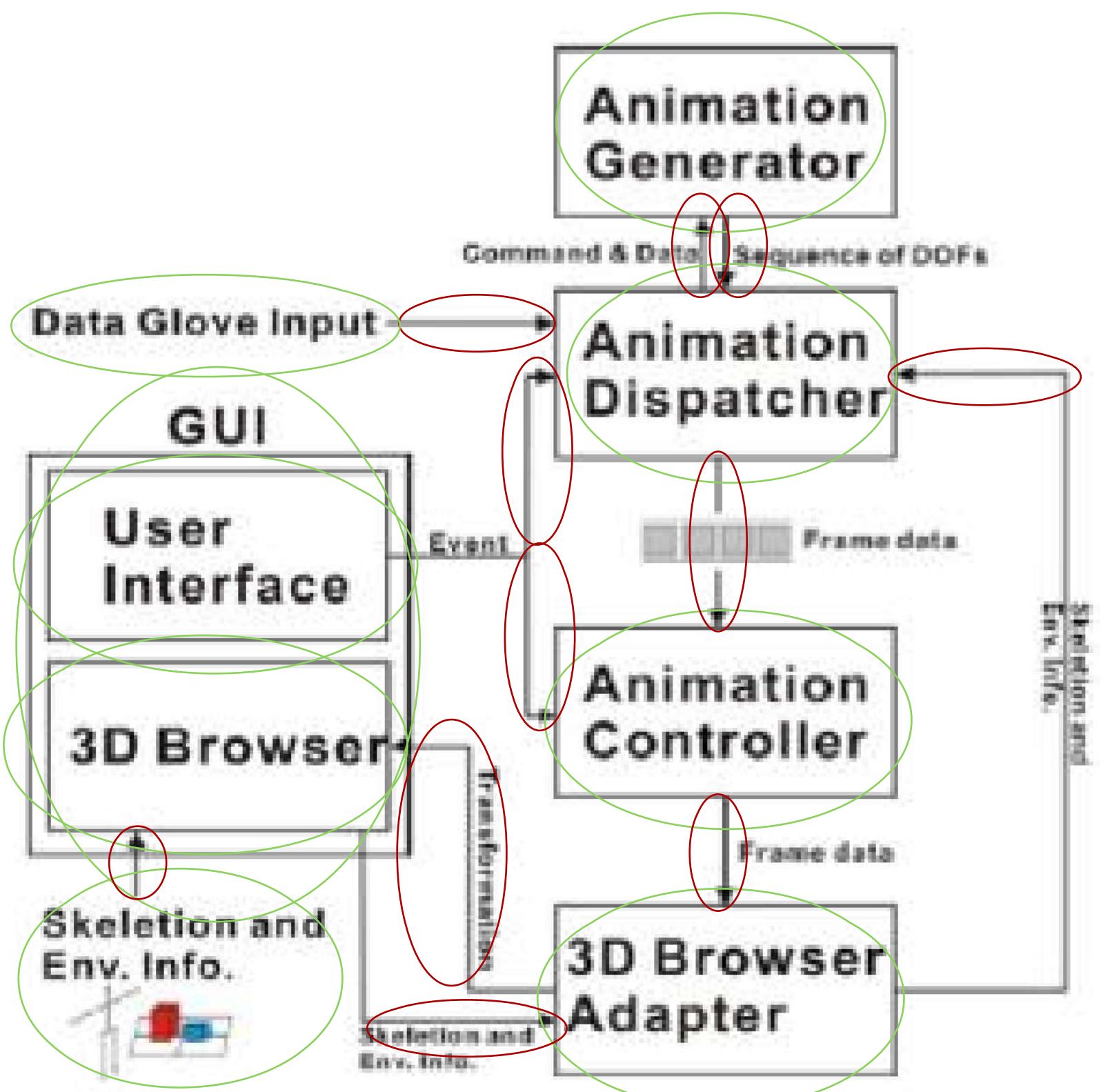
EXECUTION

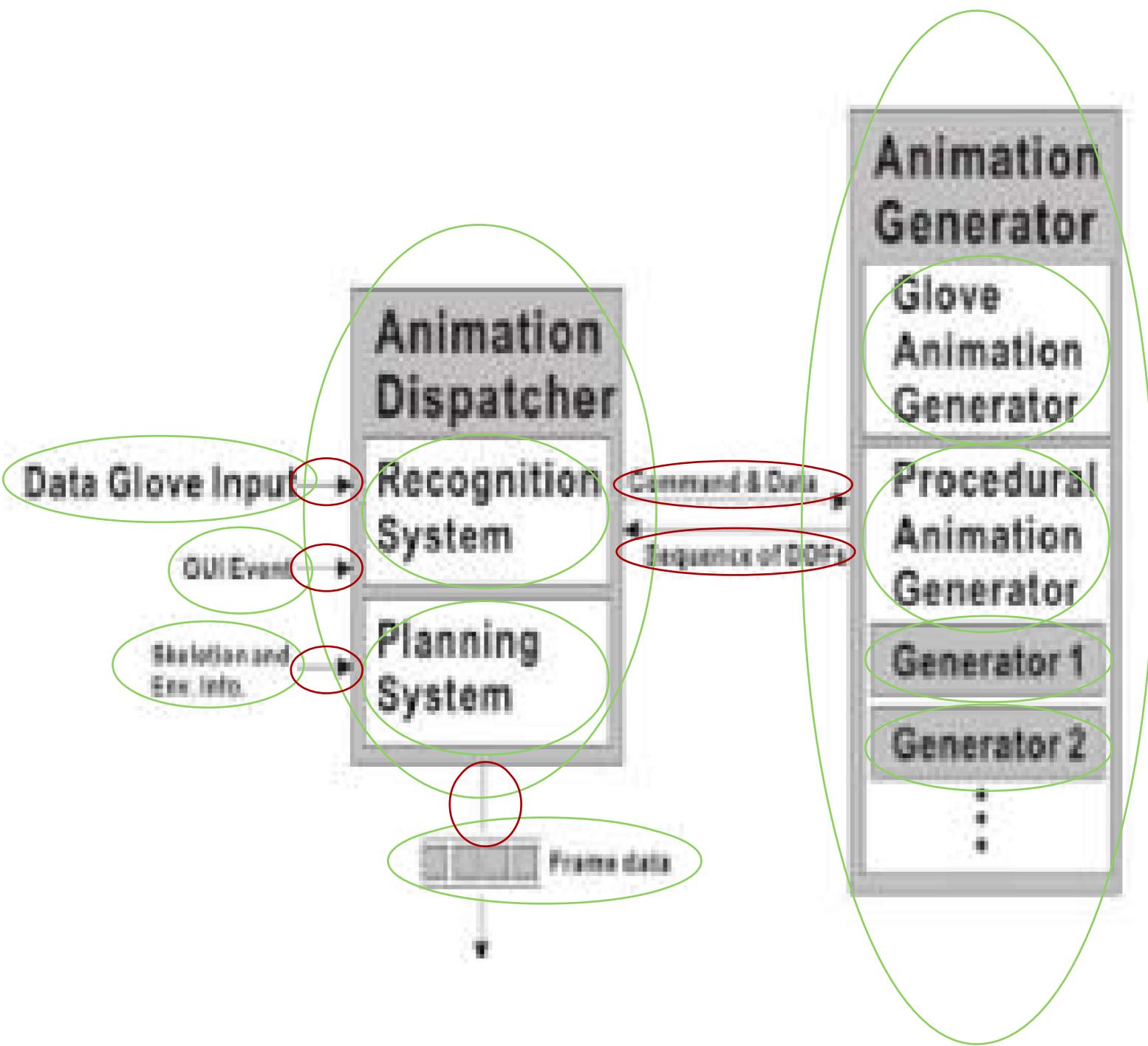


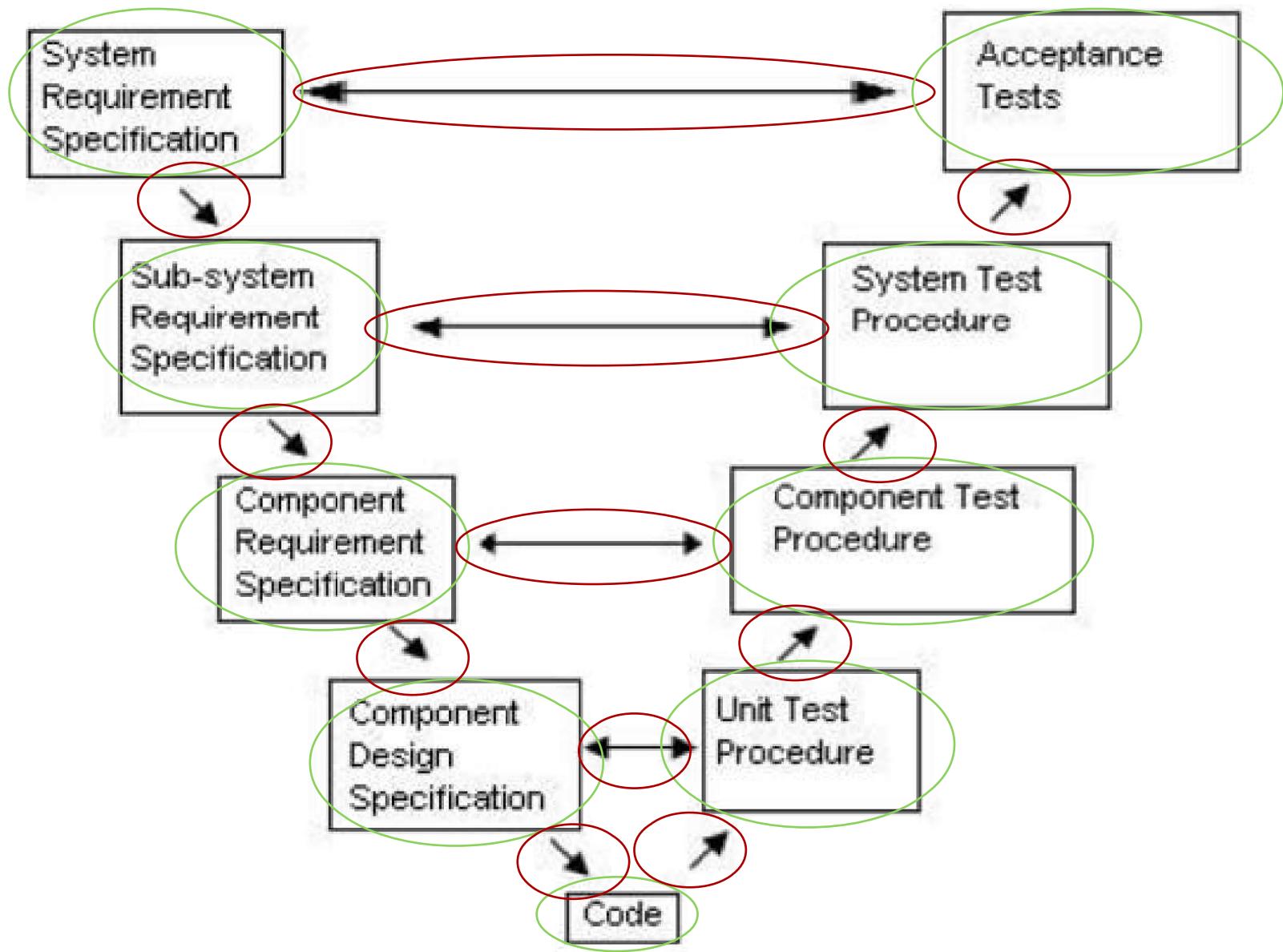


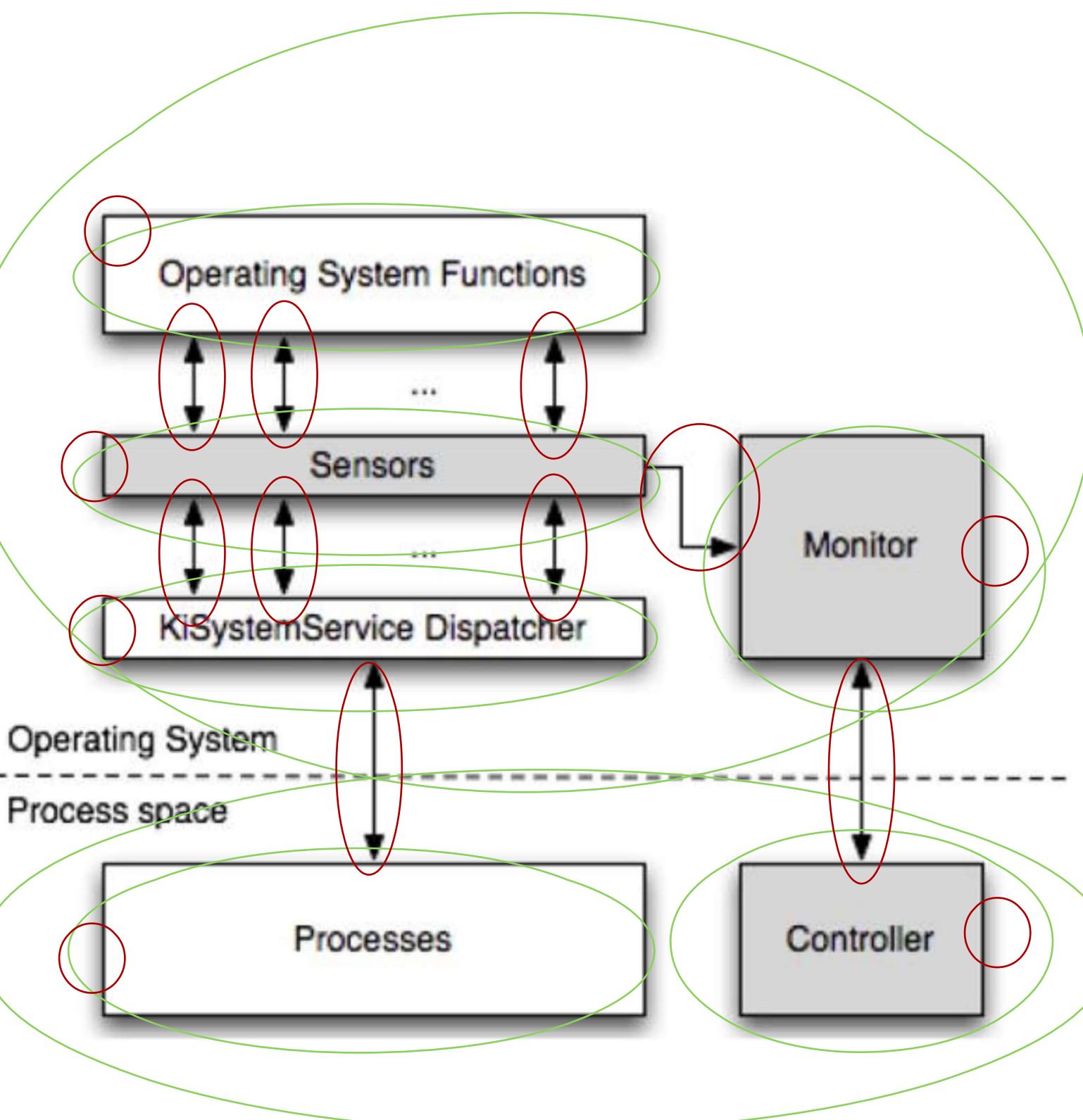


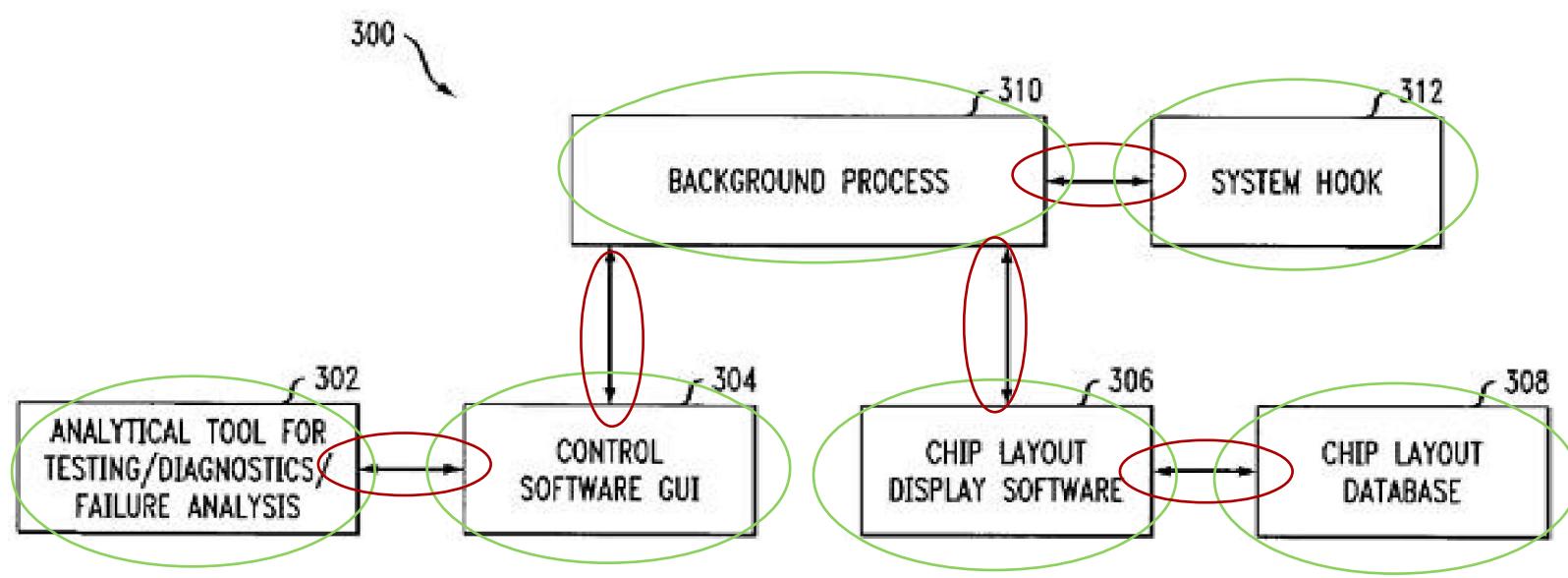












400

402

START THE ANALYTICAL
TOOL SOFTWARE GUI

404

START THE LAYOUT
DISPLAY SOFTWARE

406

START THE
BACKGROUND SOFTWARE

408

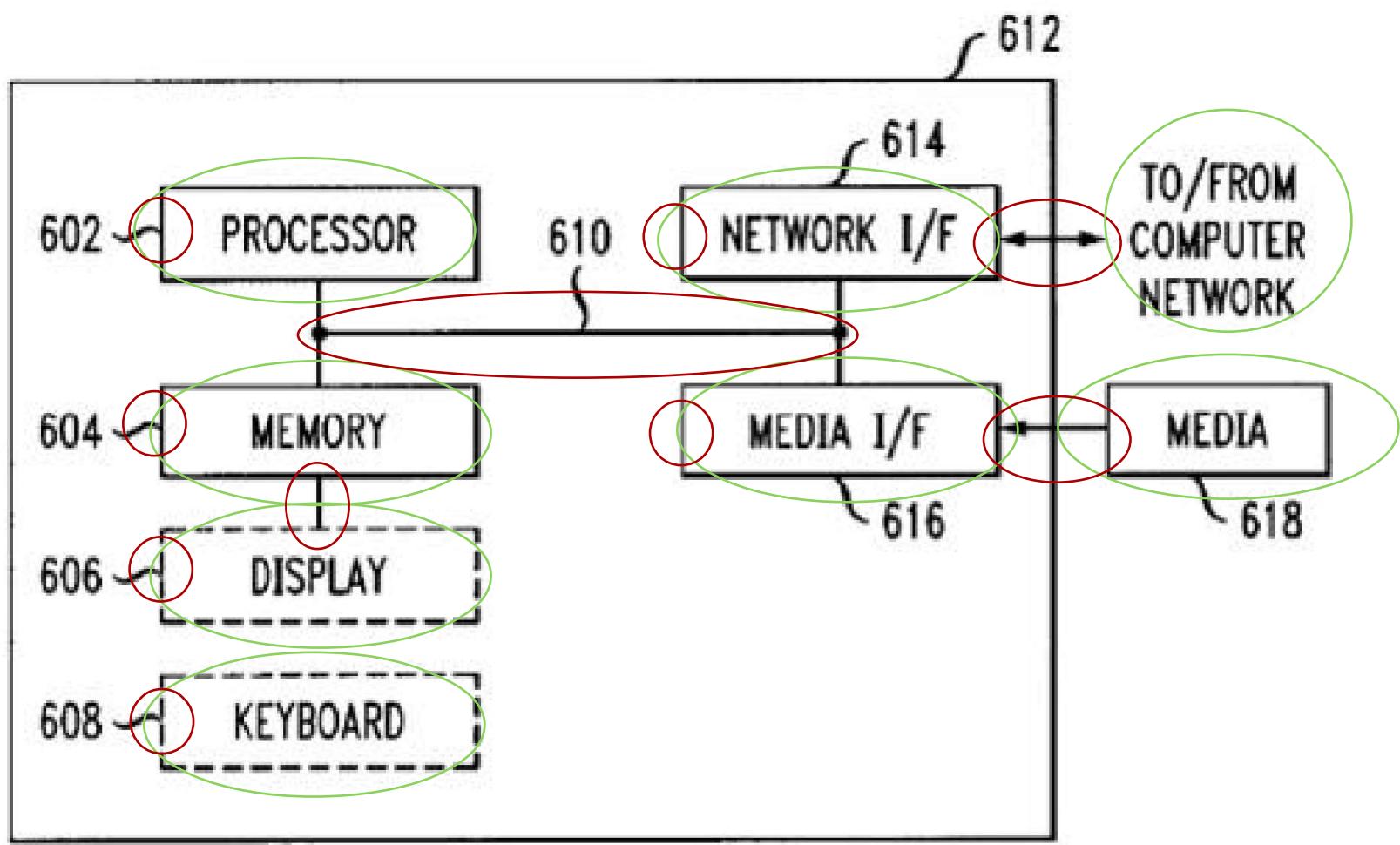
INSTALL ONE OR
MORE SYSTEM HOOKS

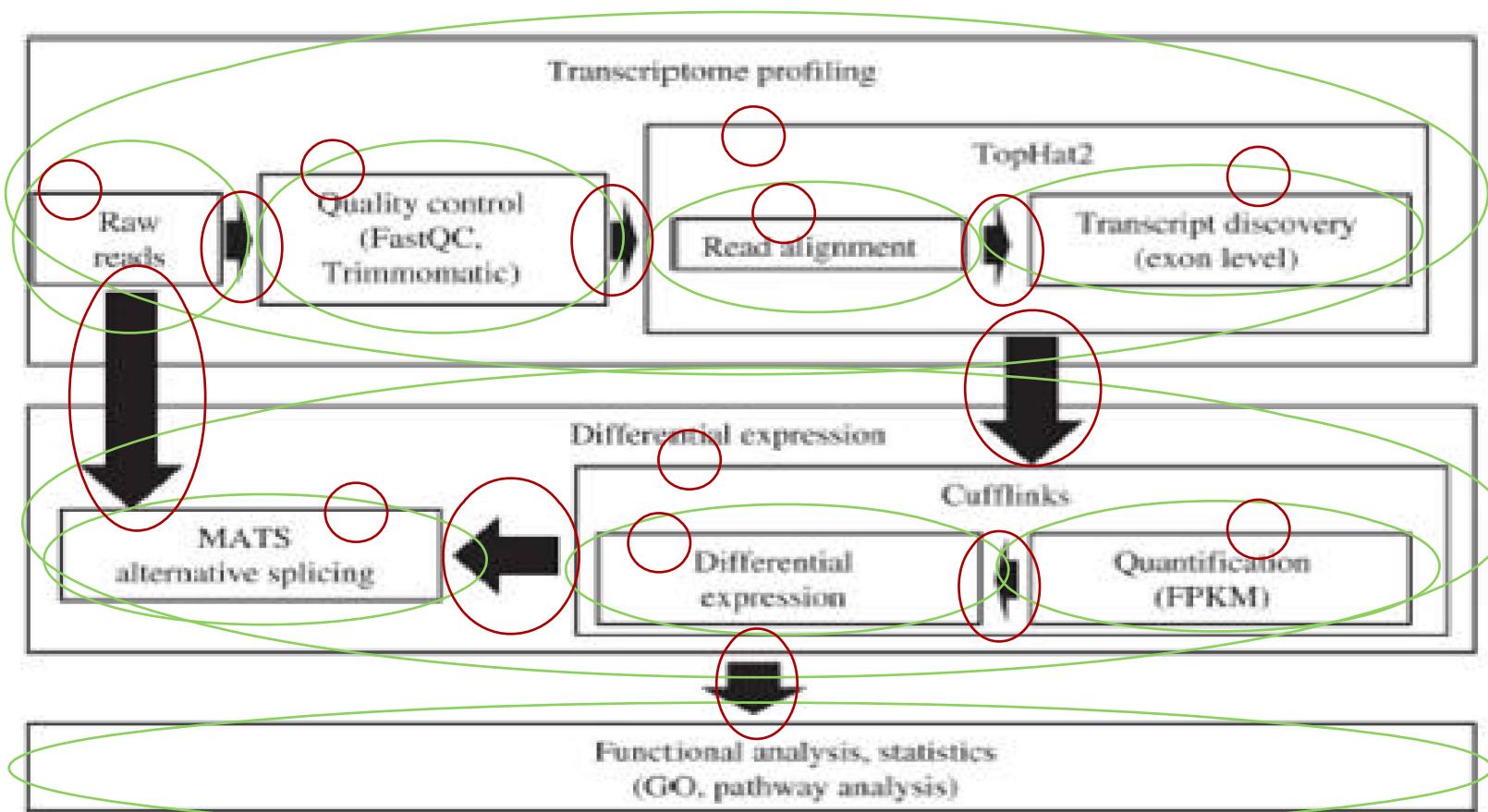
410

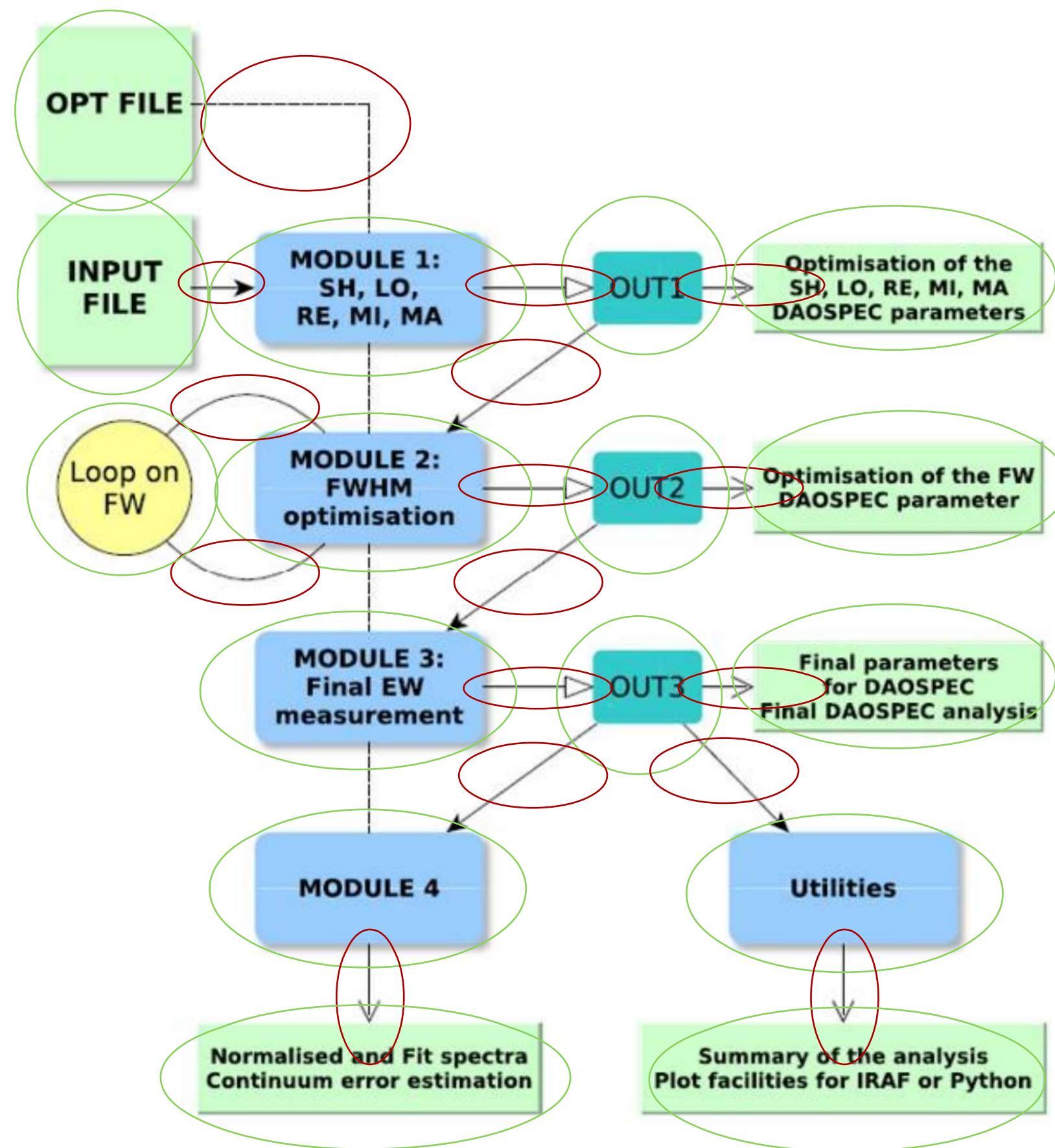
ENTER A STANDBY MODE
WAITING FOR THE SYSTEM
HOOK TO BE TRIGGERED

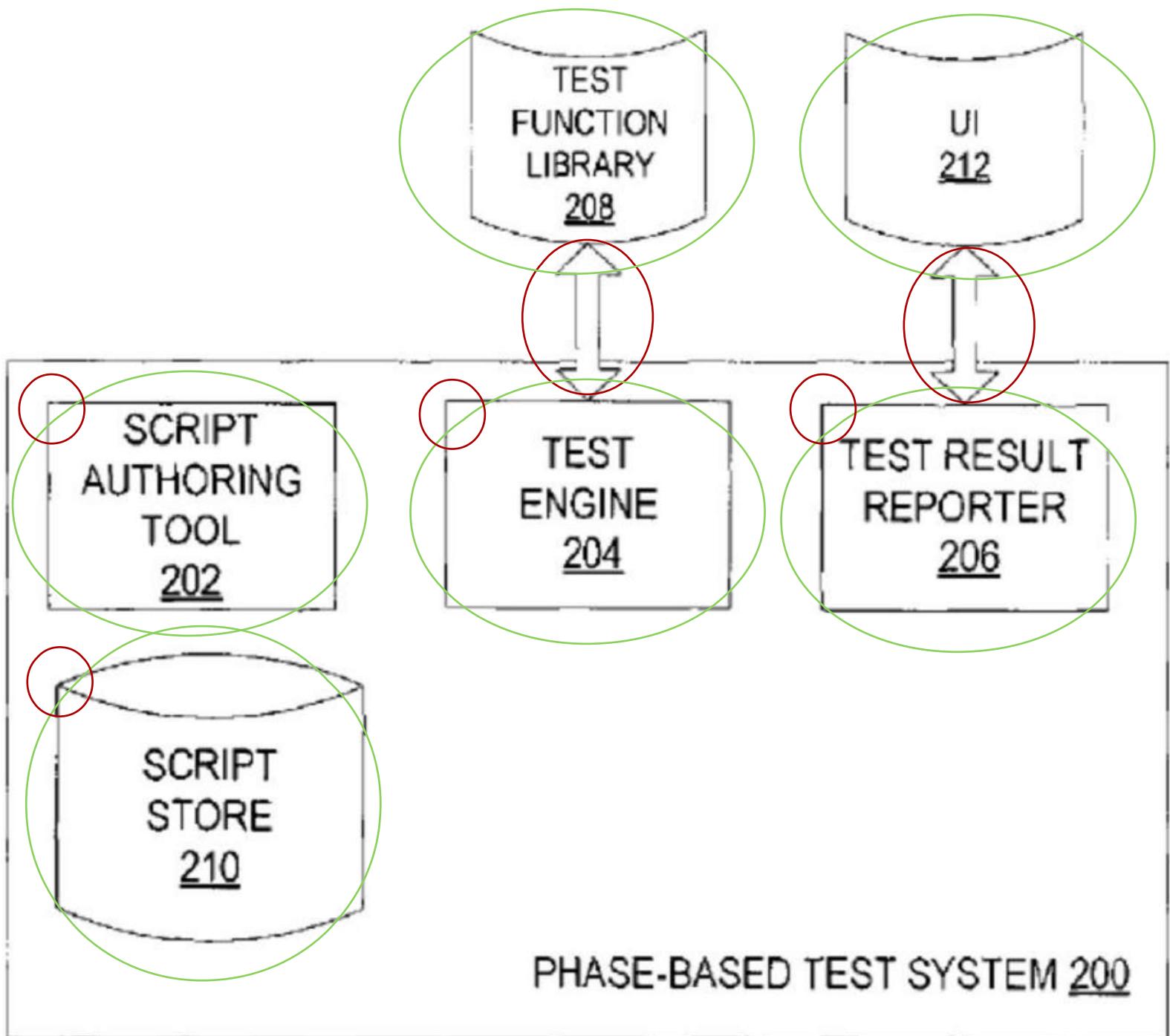
412

OPERATOR HAS CONTROL







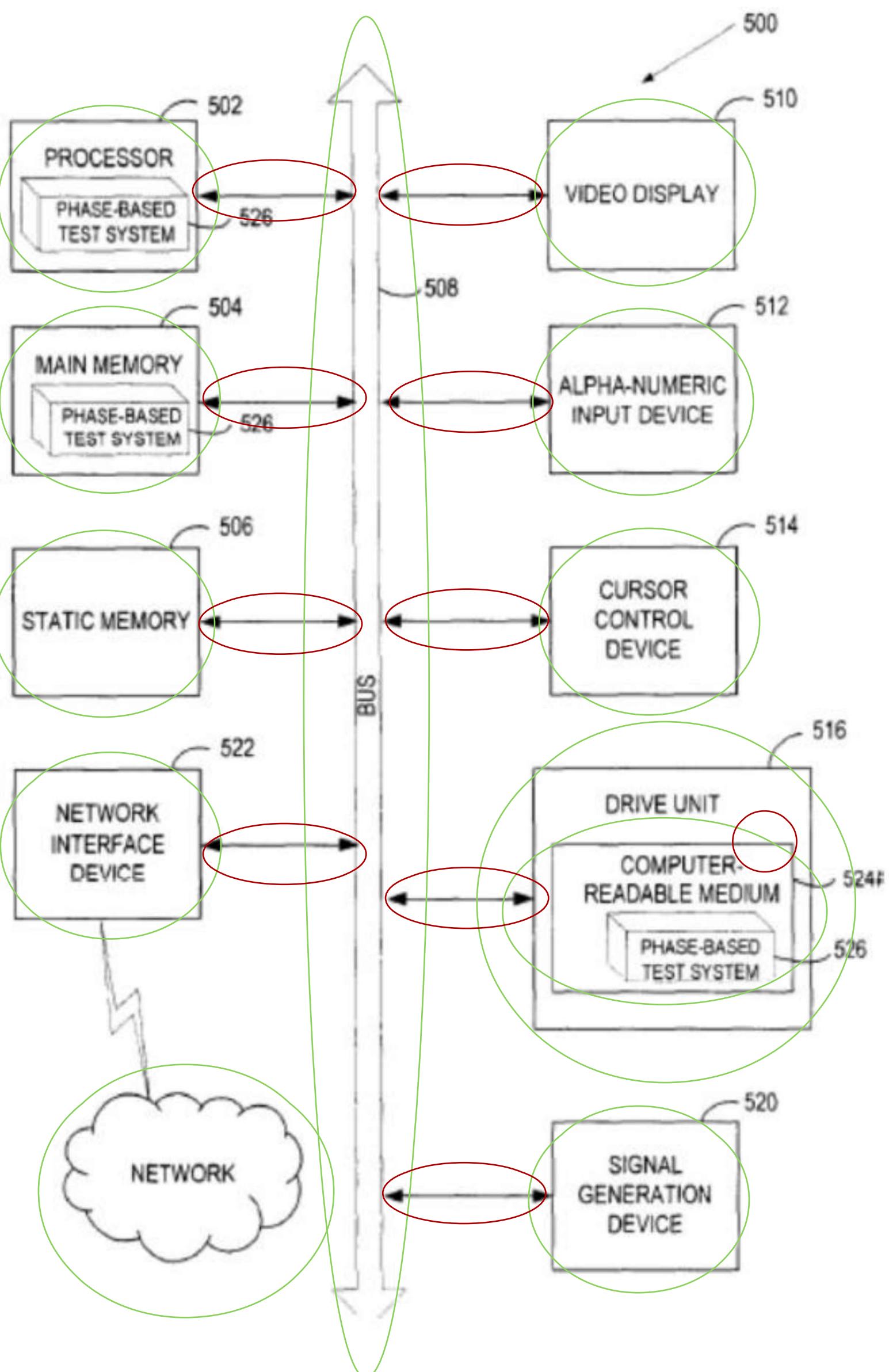


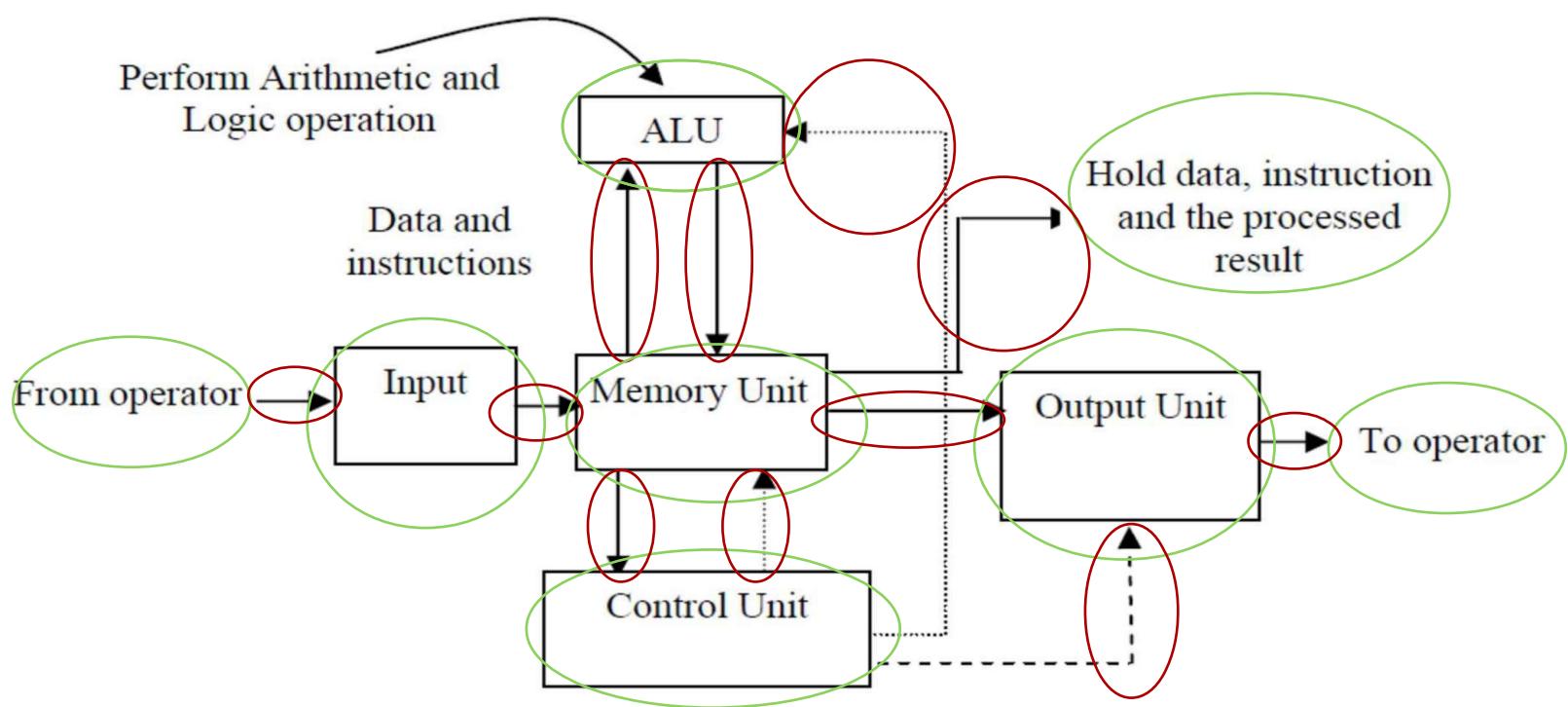
**PHASE-BASED
TEST SYSTEM 104**

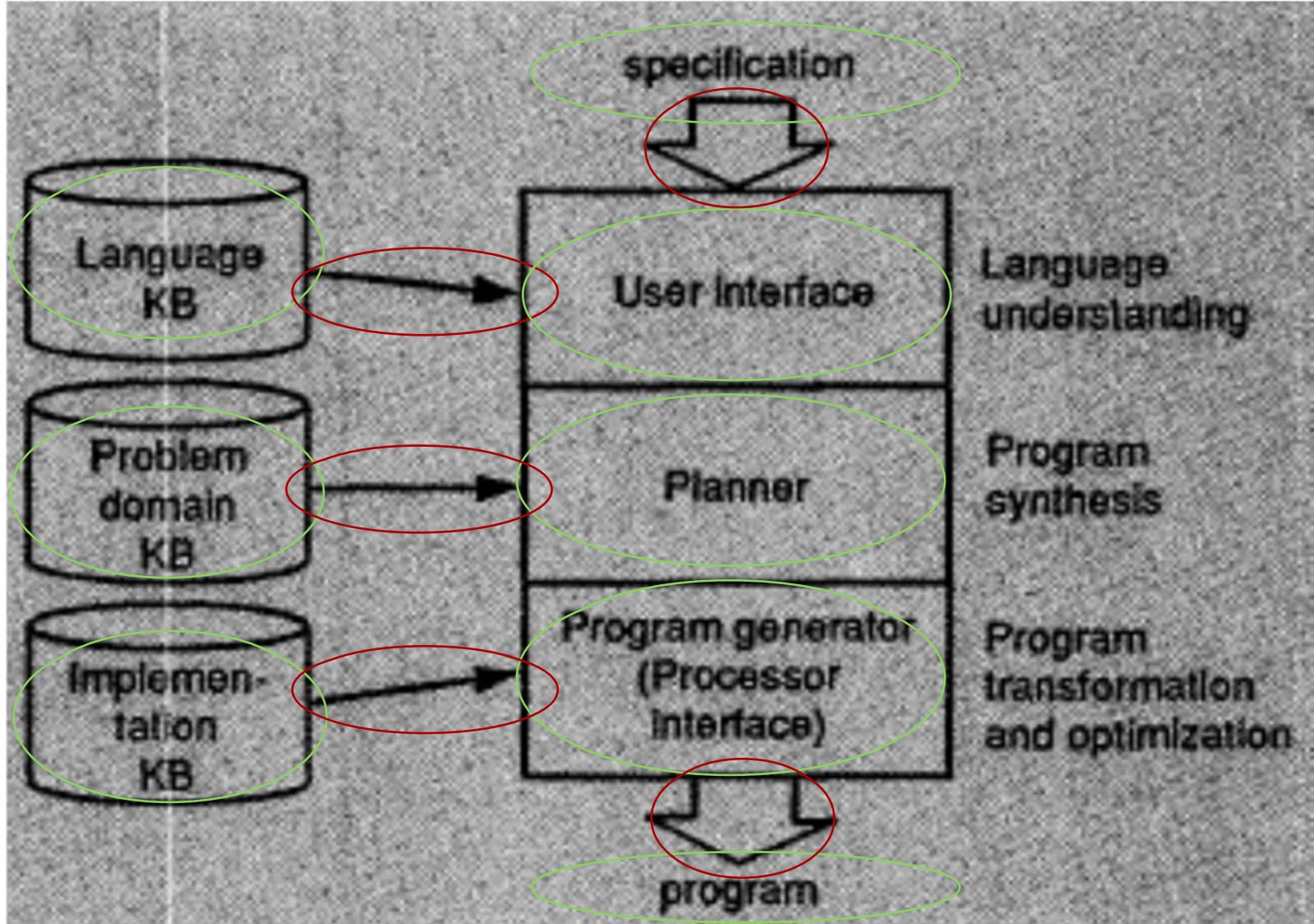
SCRIPT 106

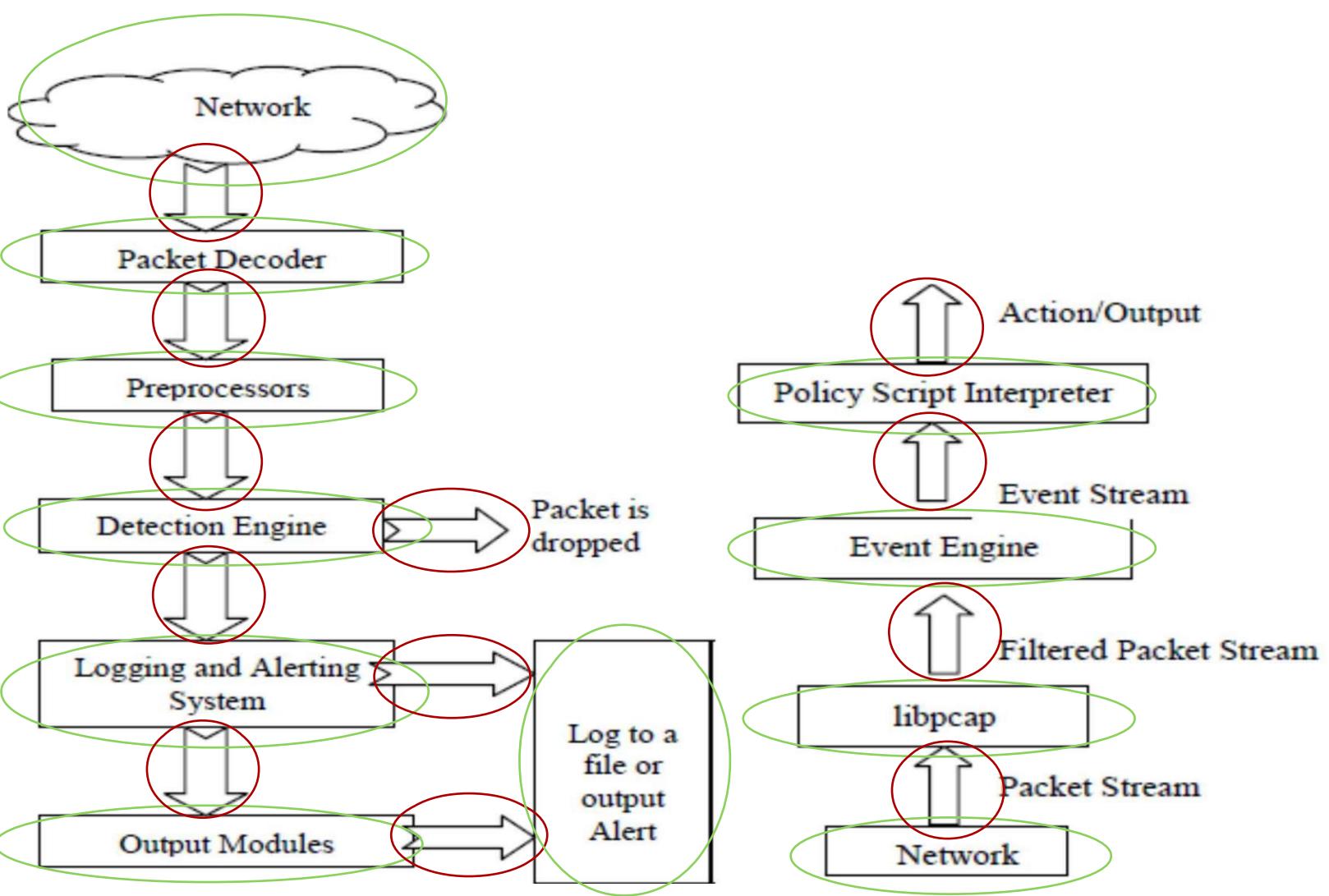
TEST ENVIRONMENT 102

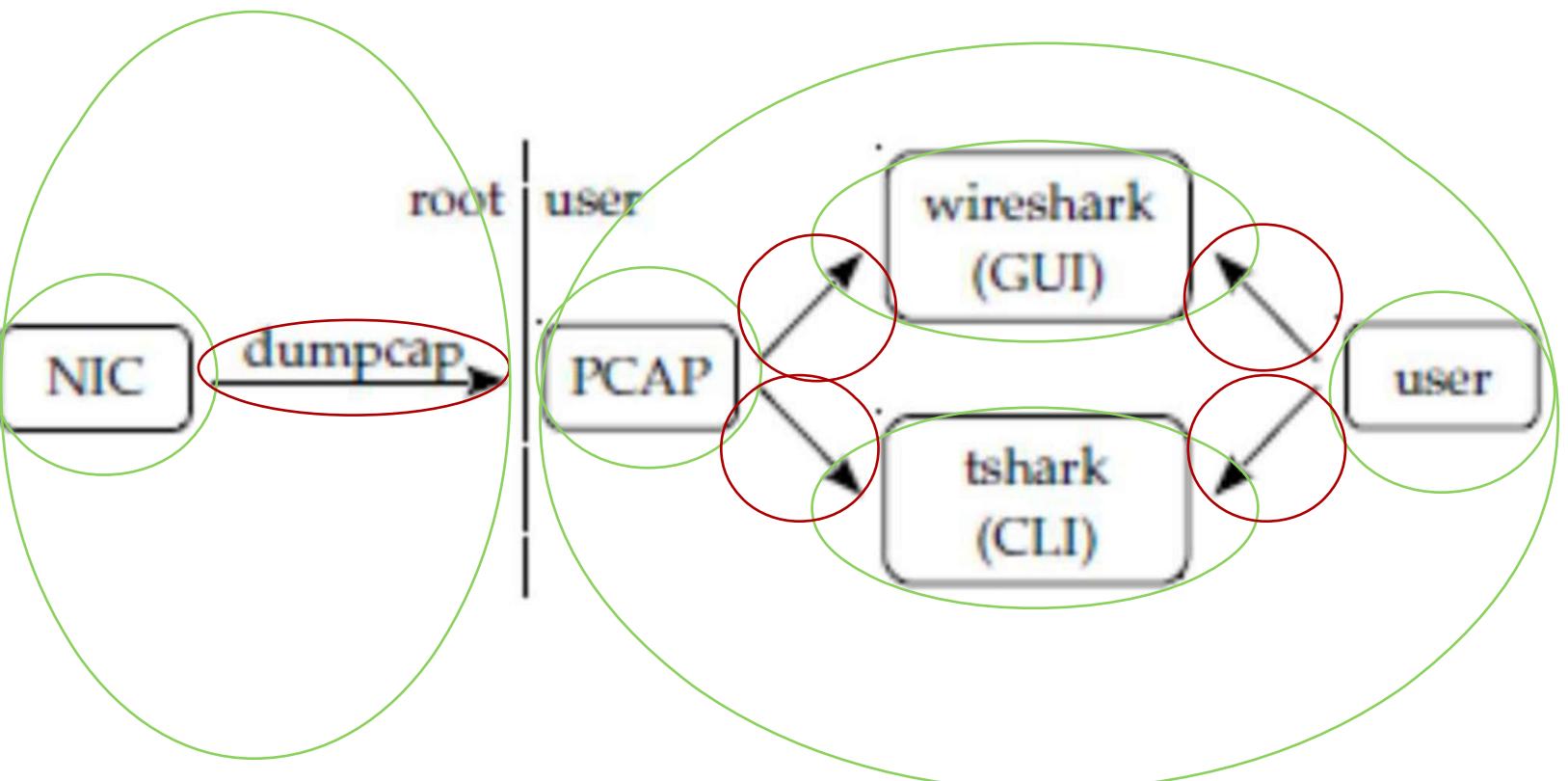
COMPUTER SYSTEM 100

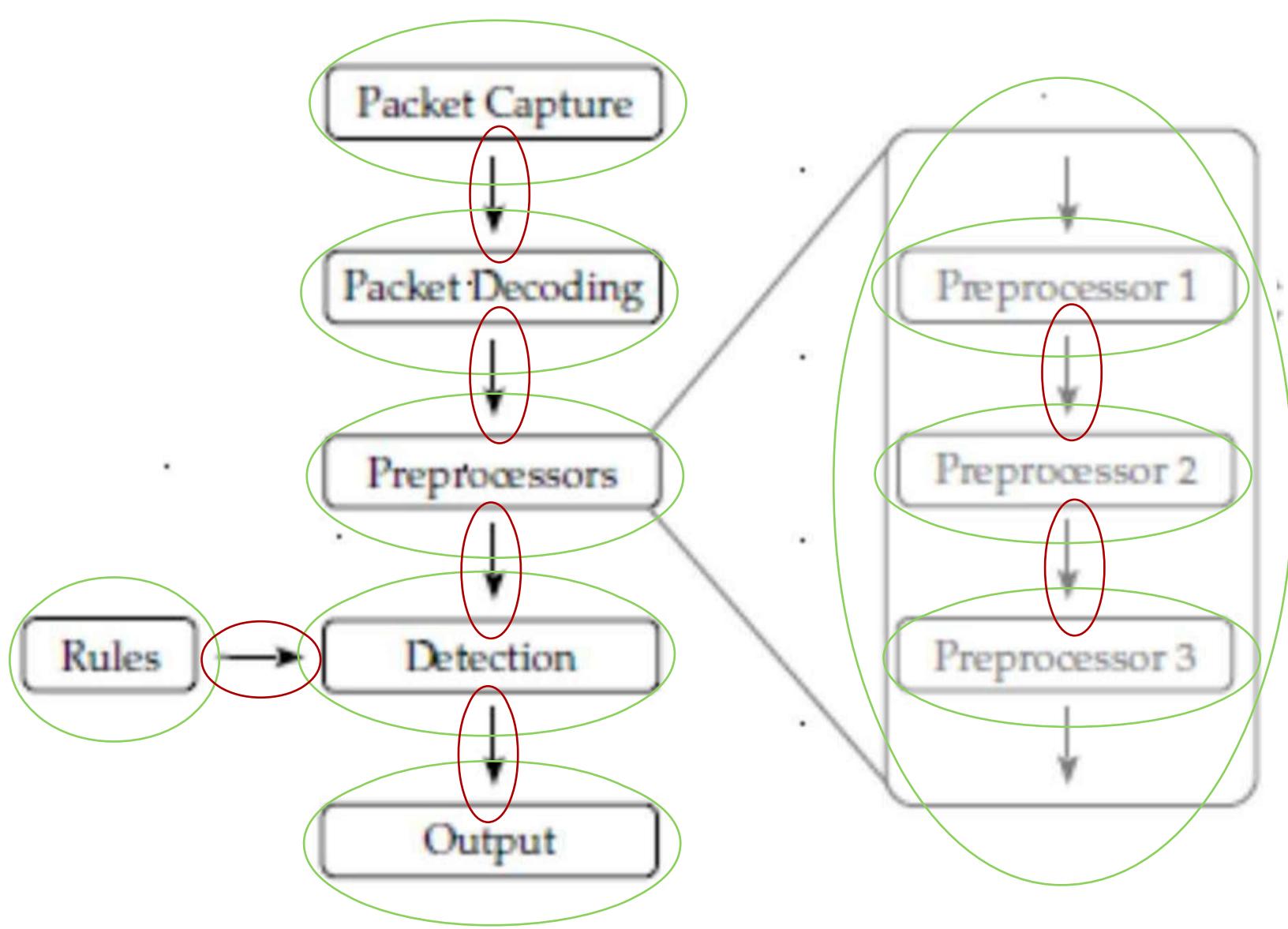


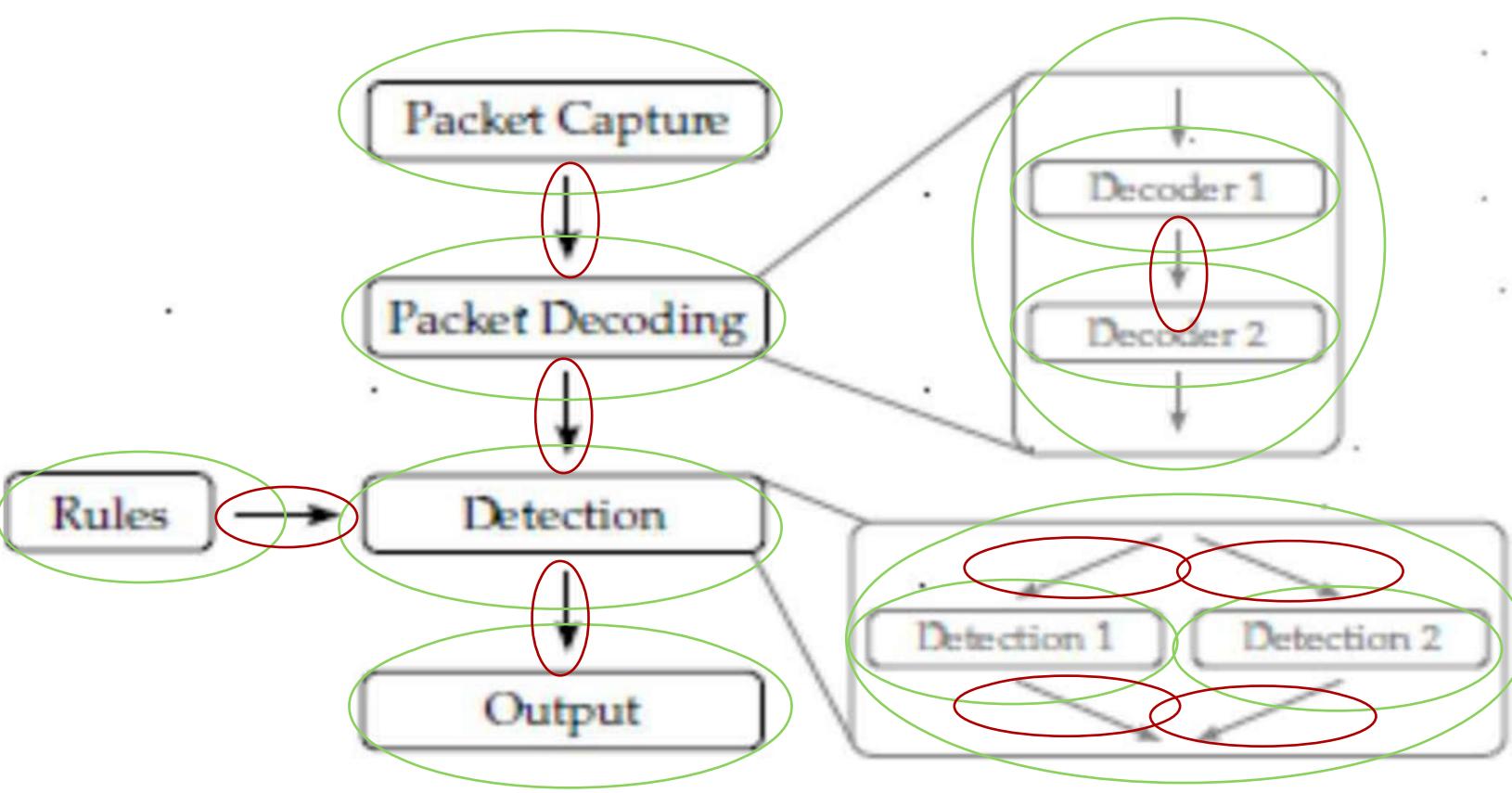


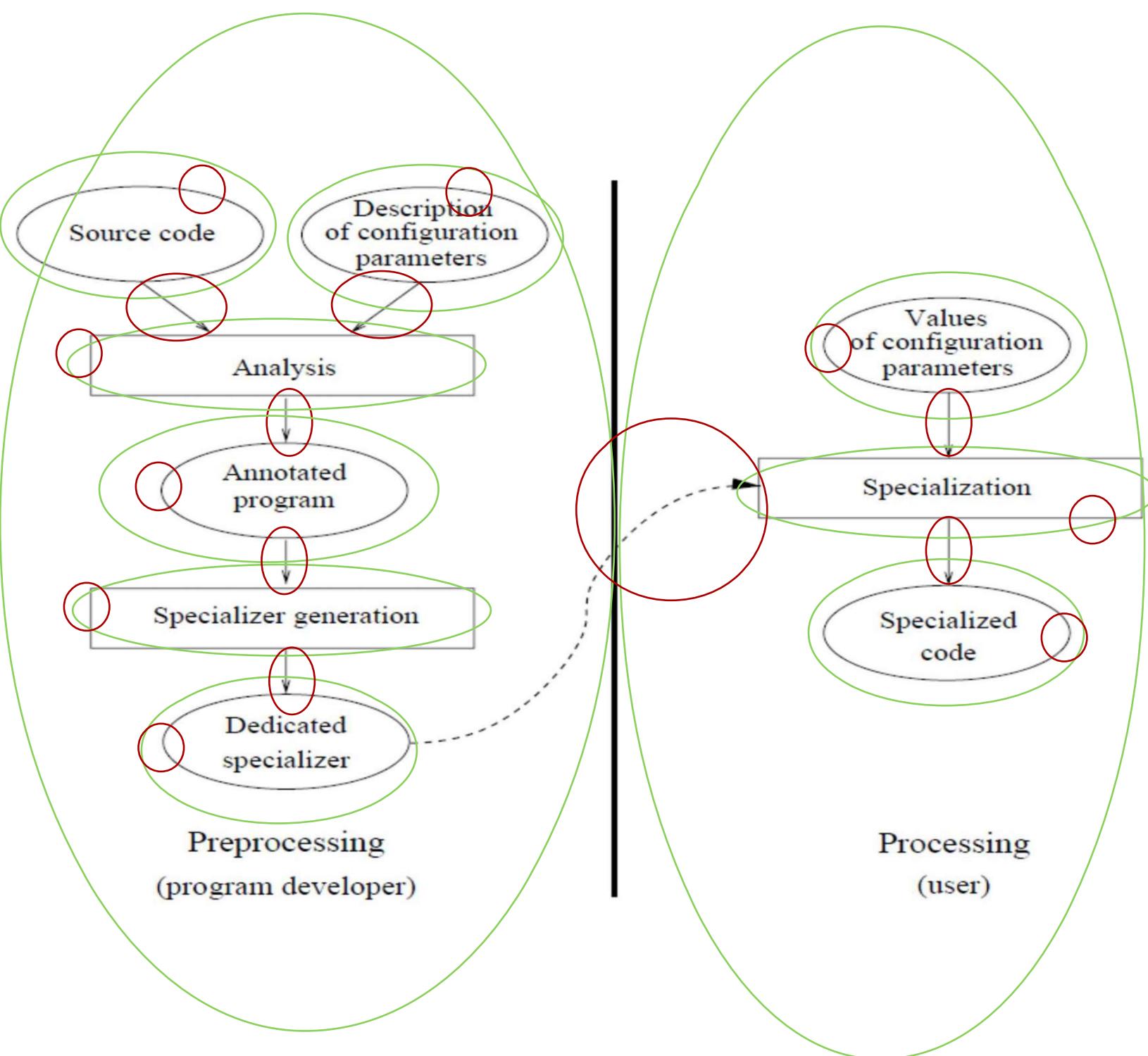


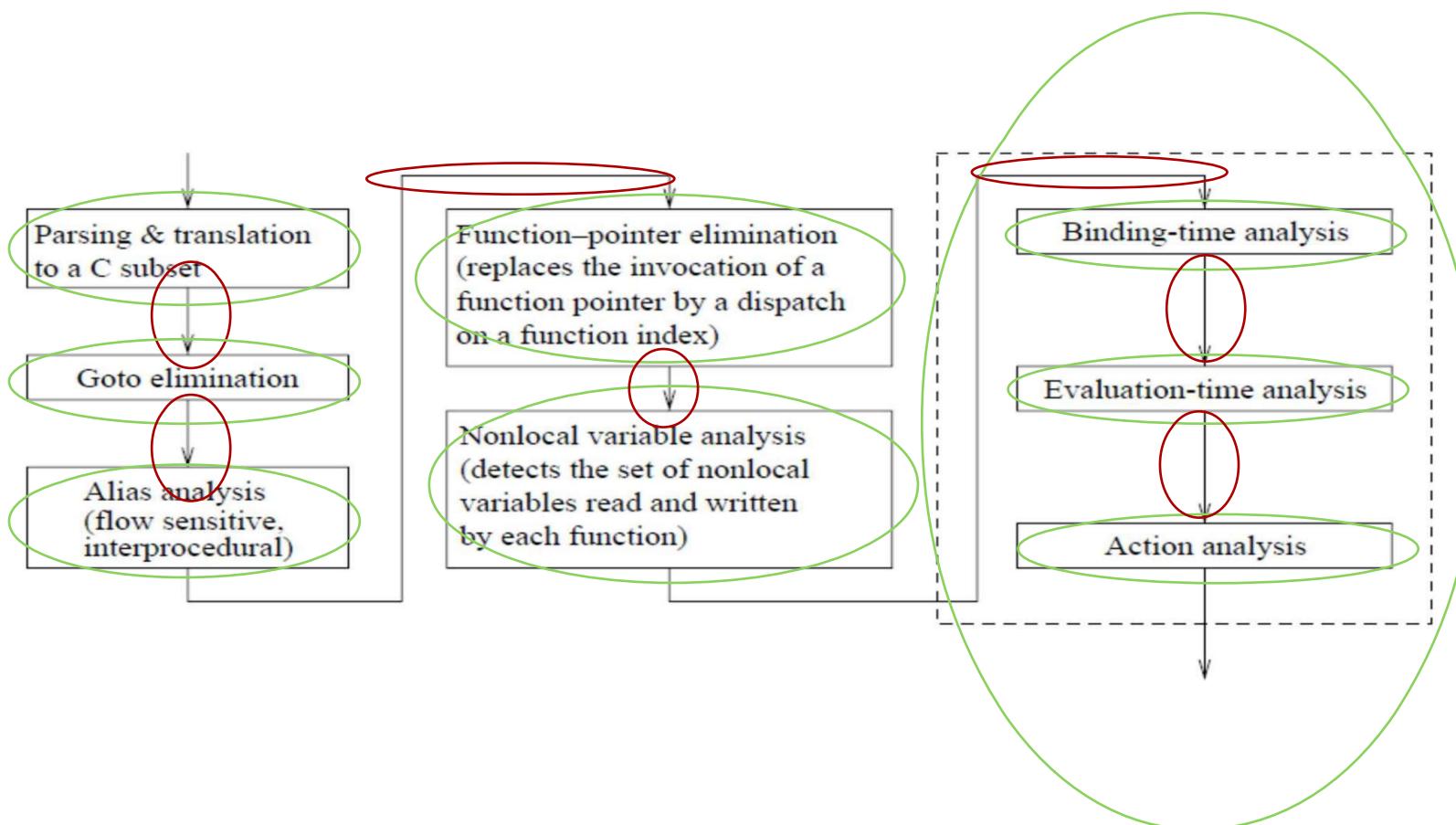


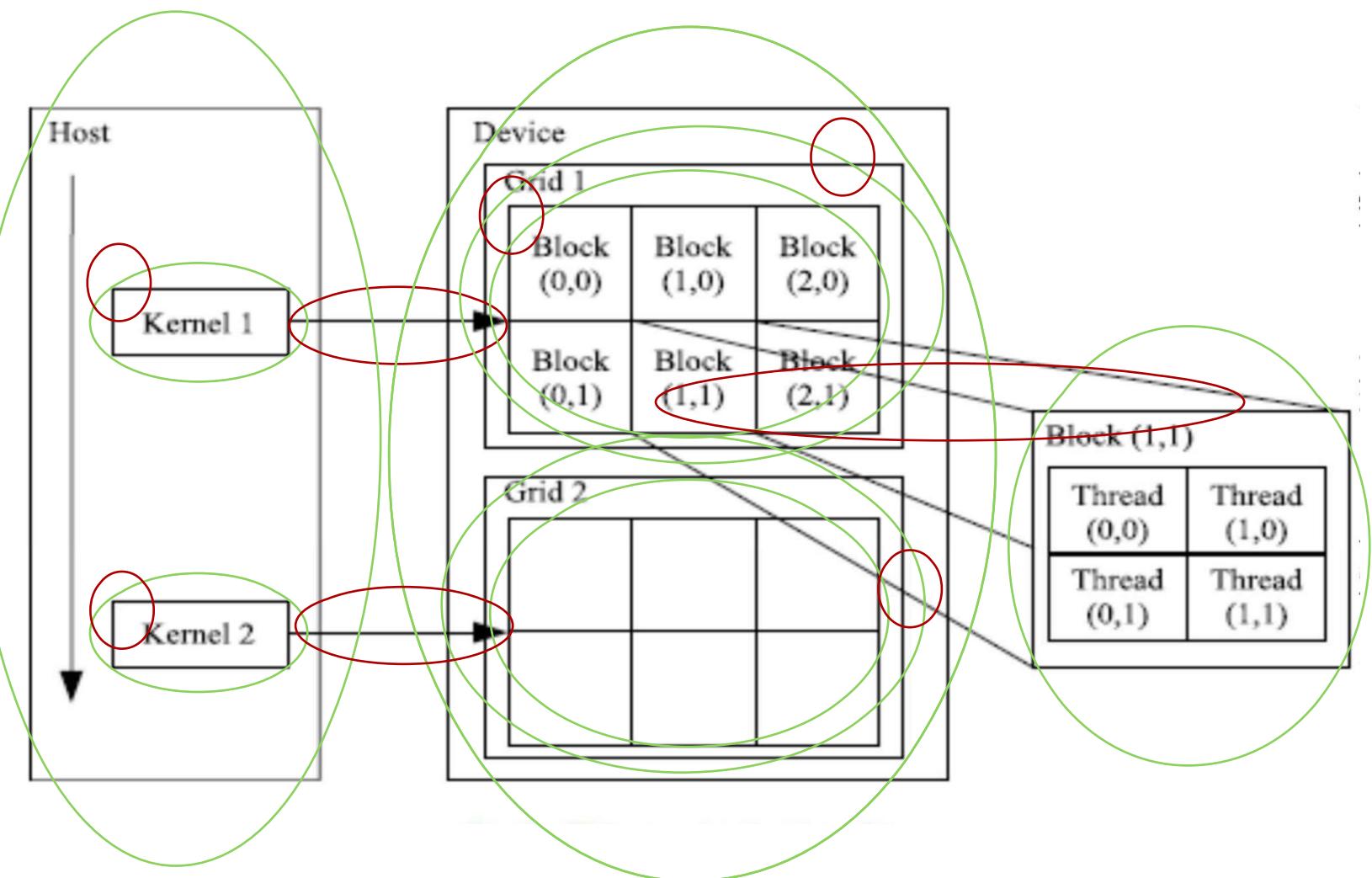


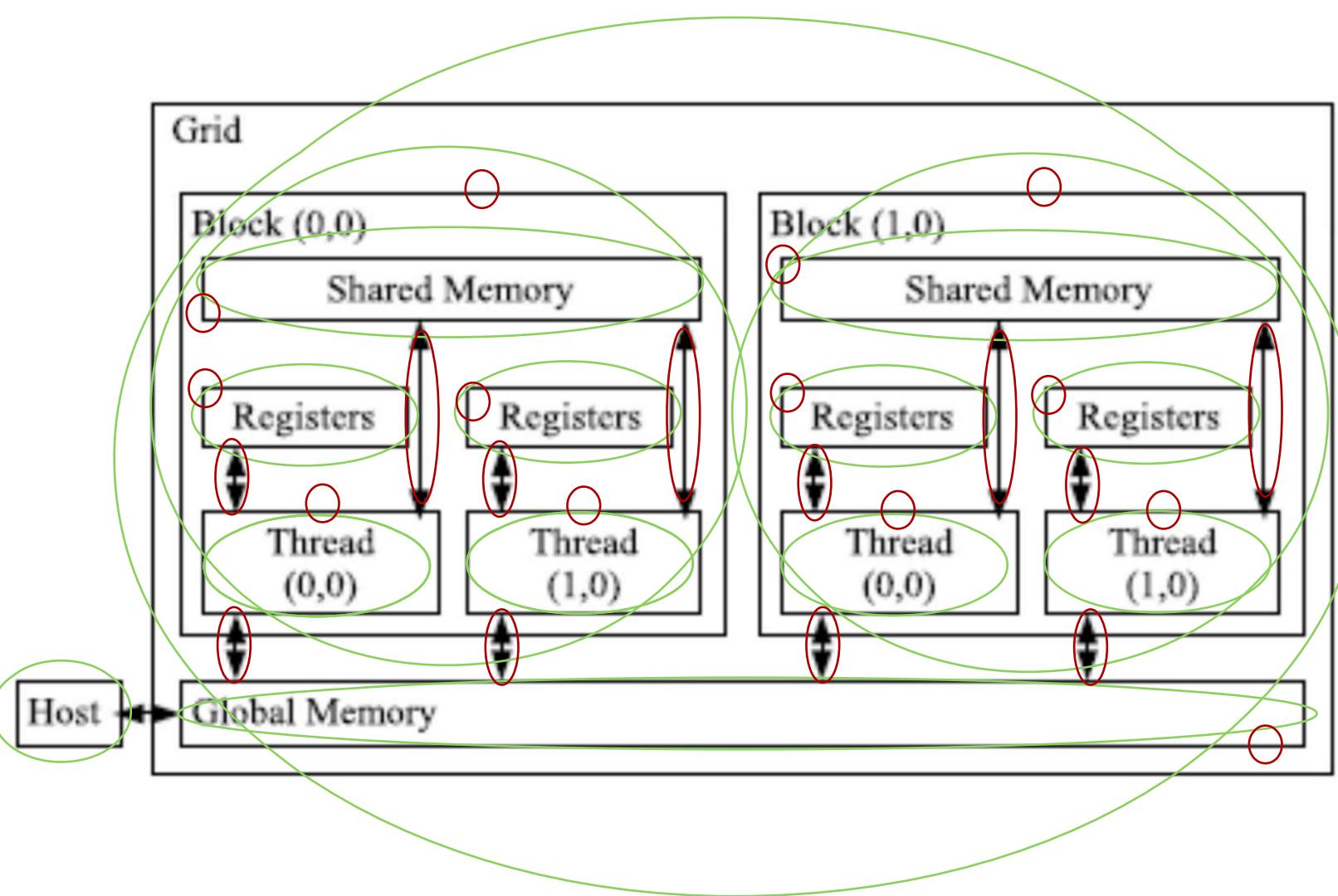


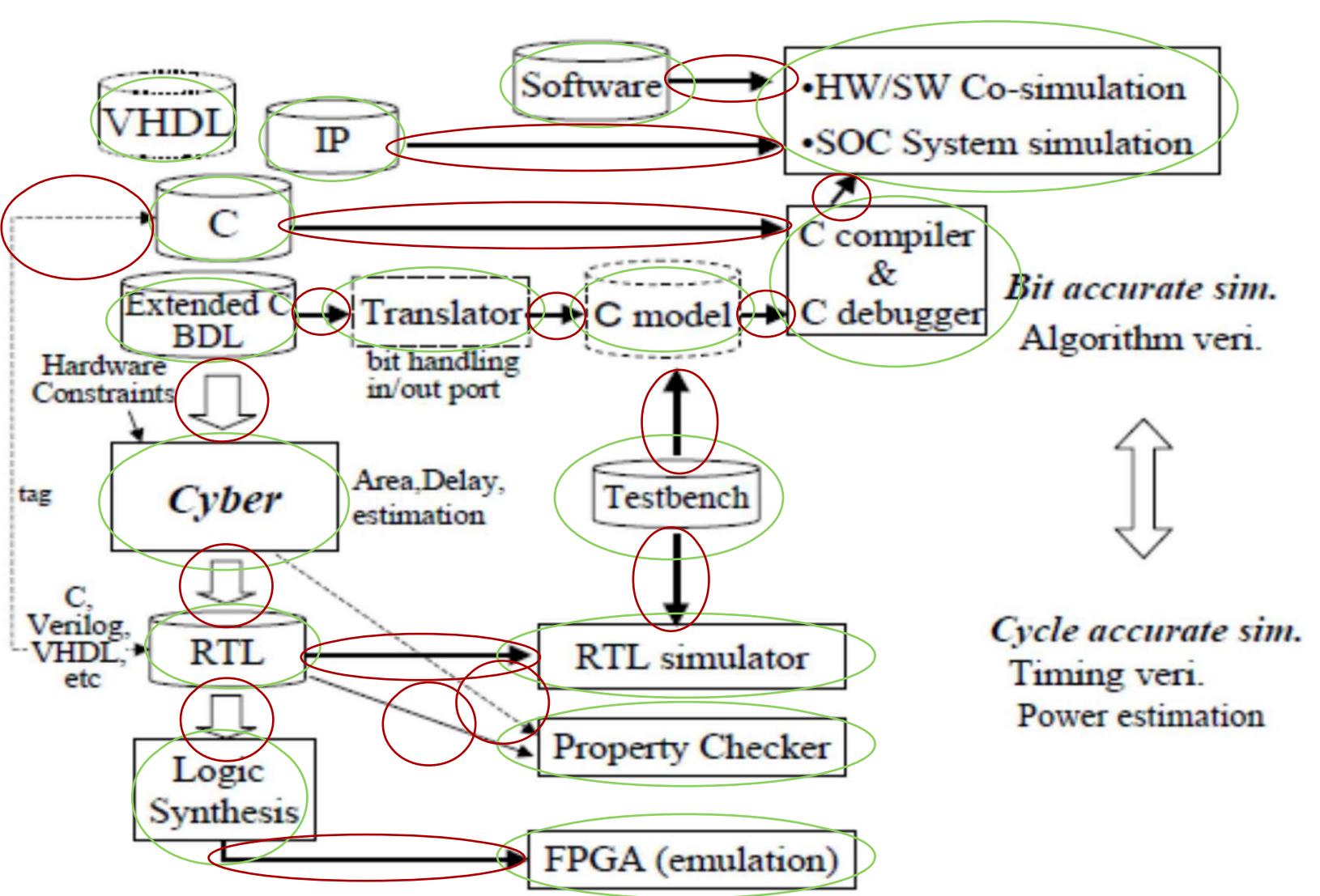


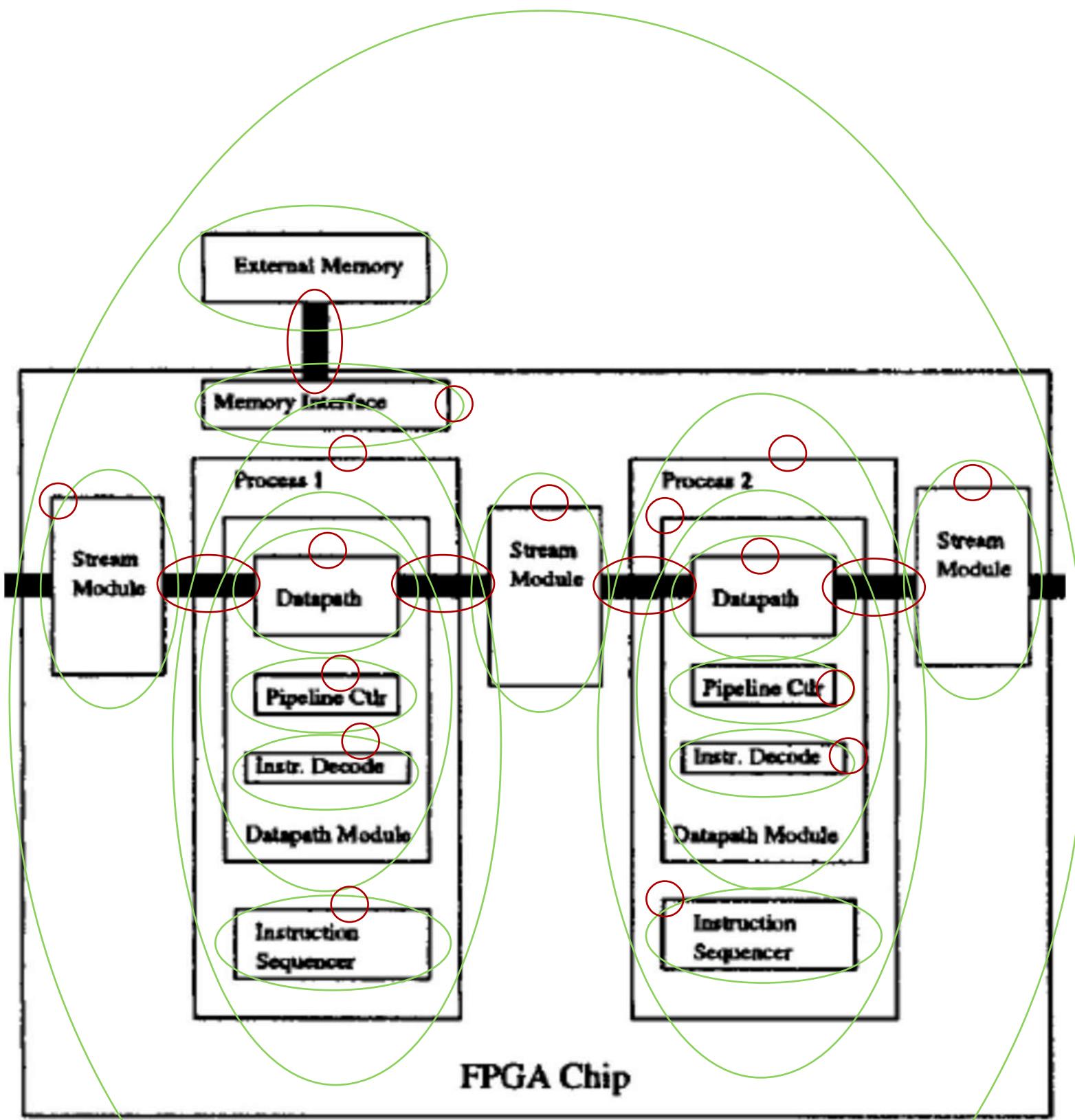


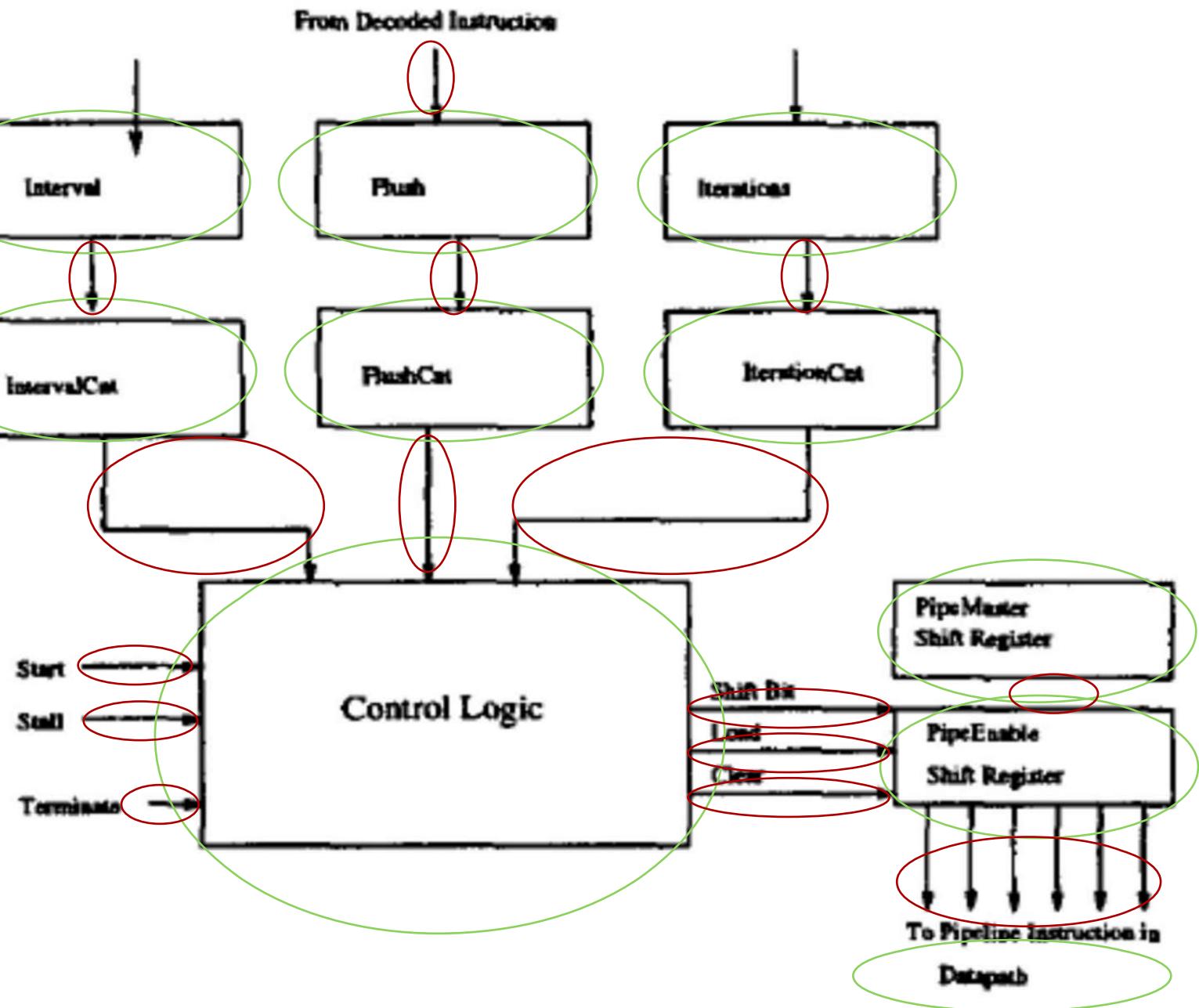


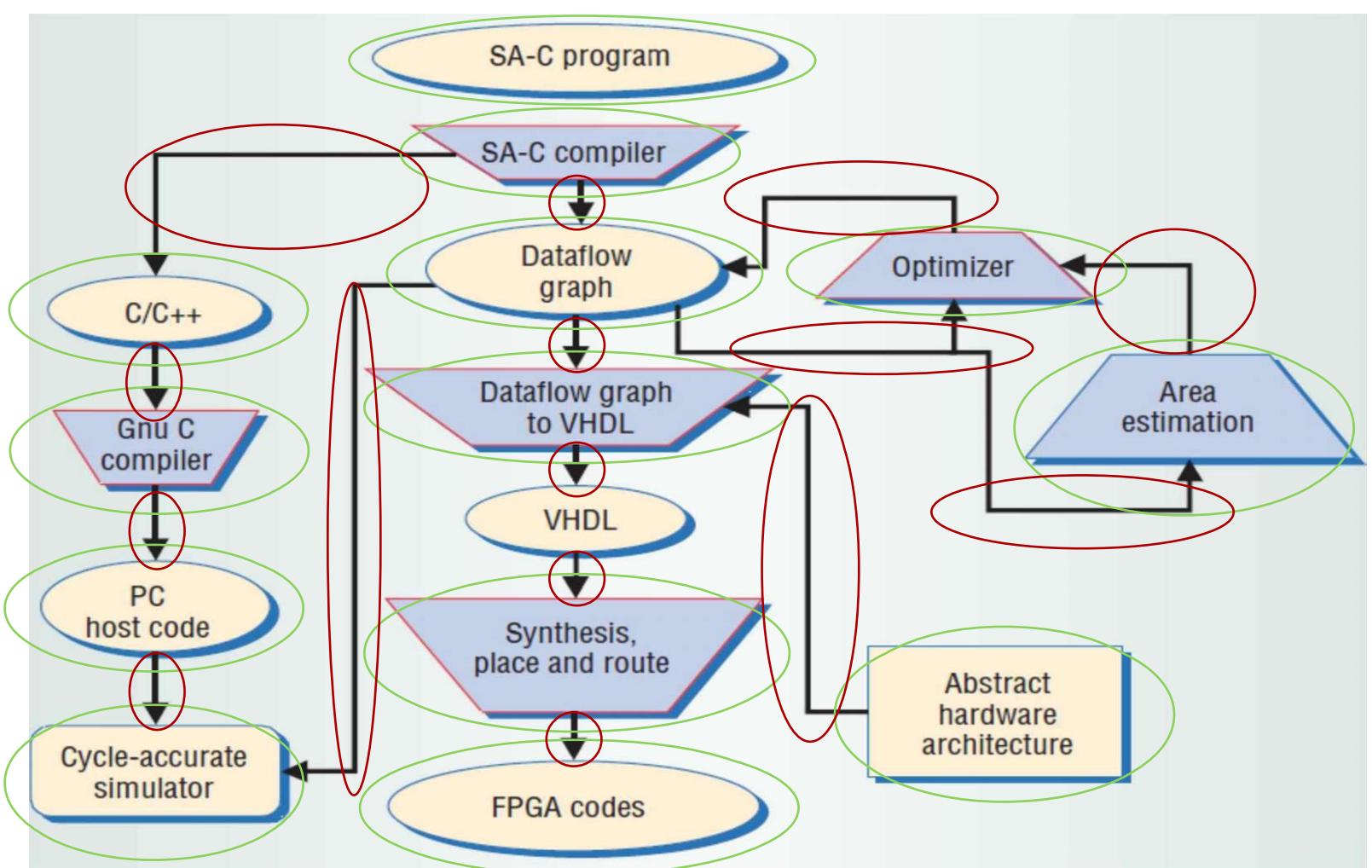


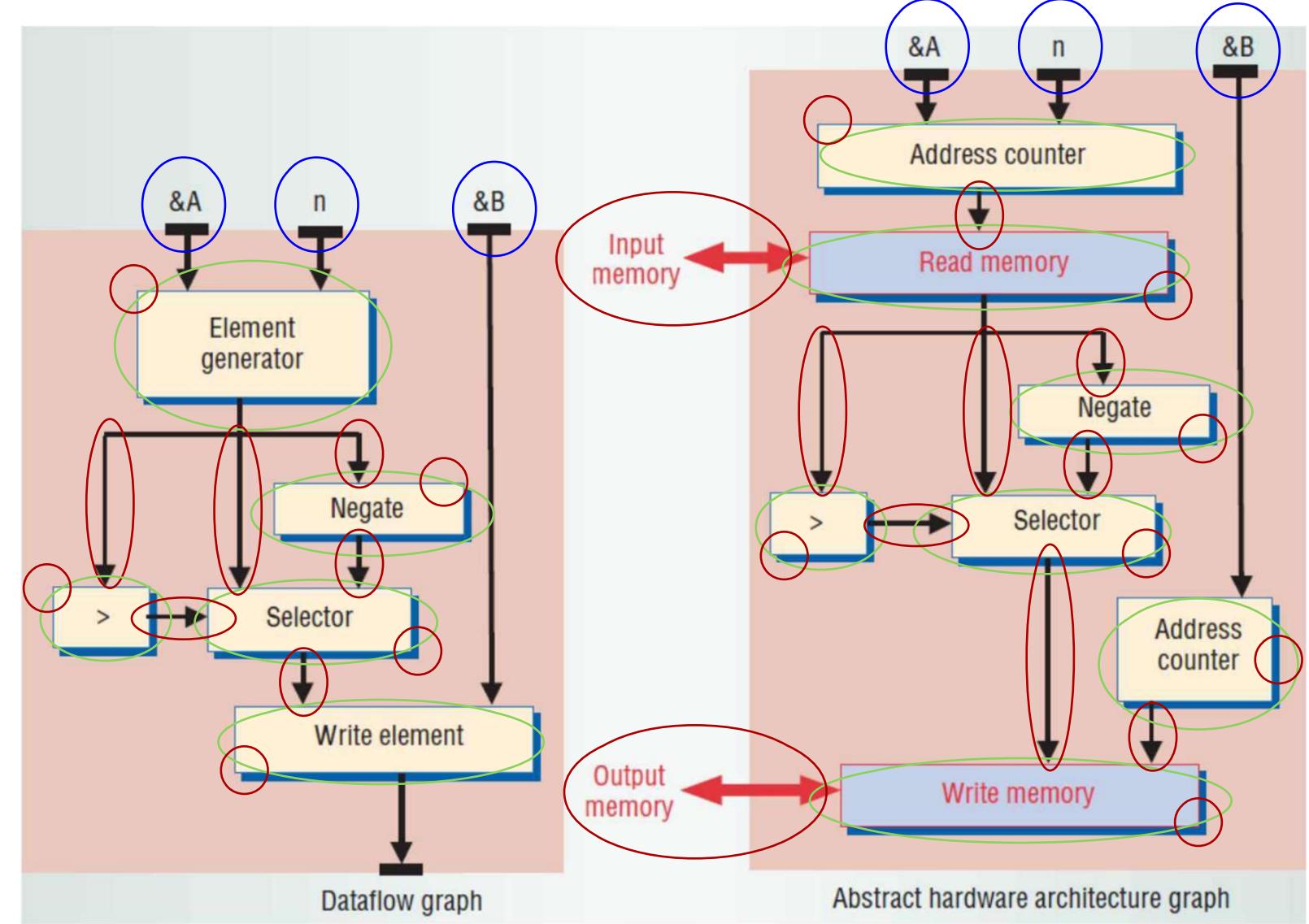


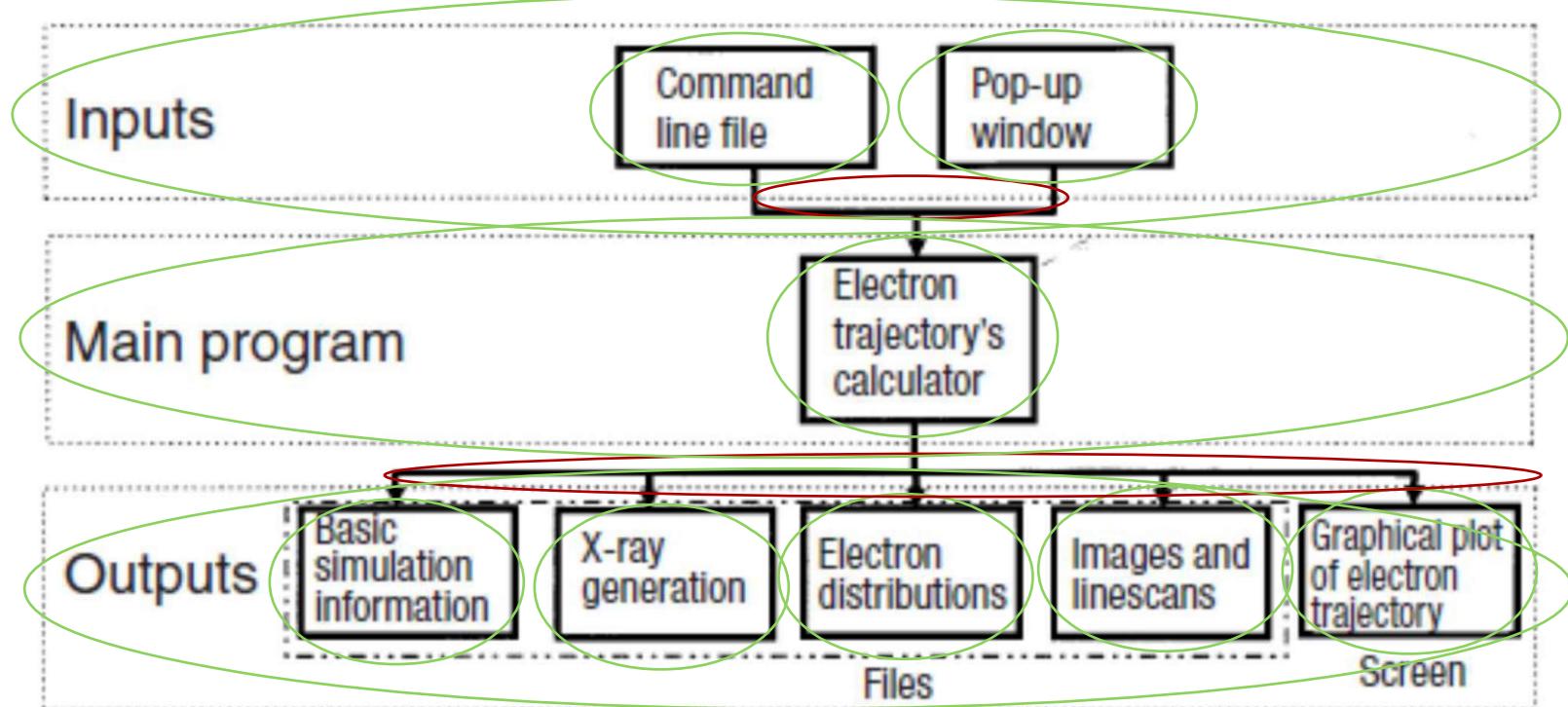


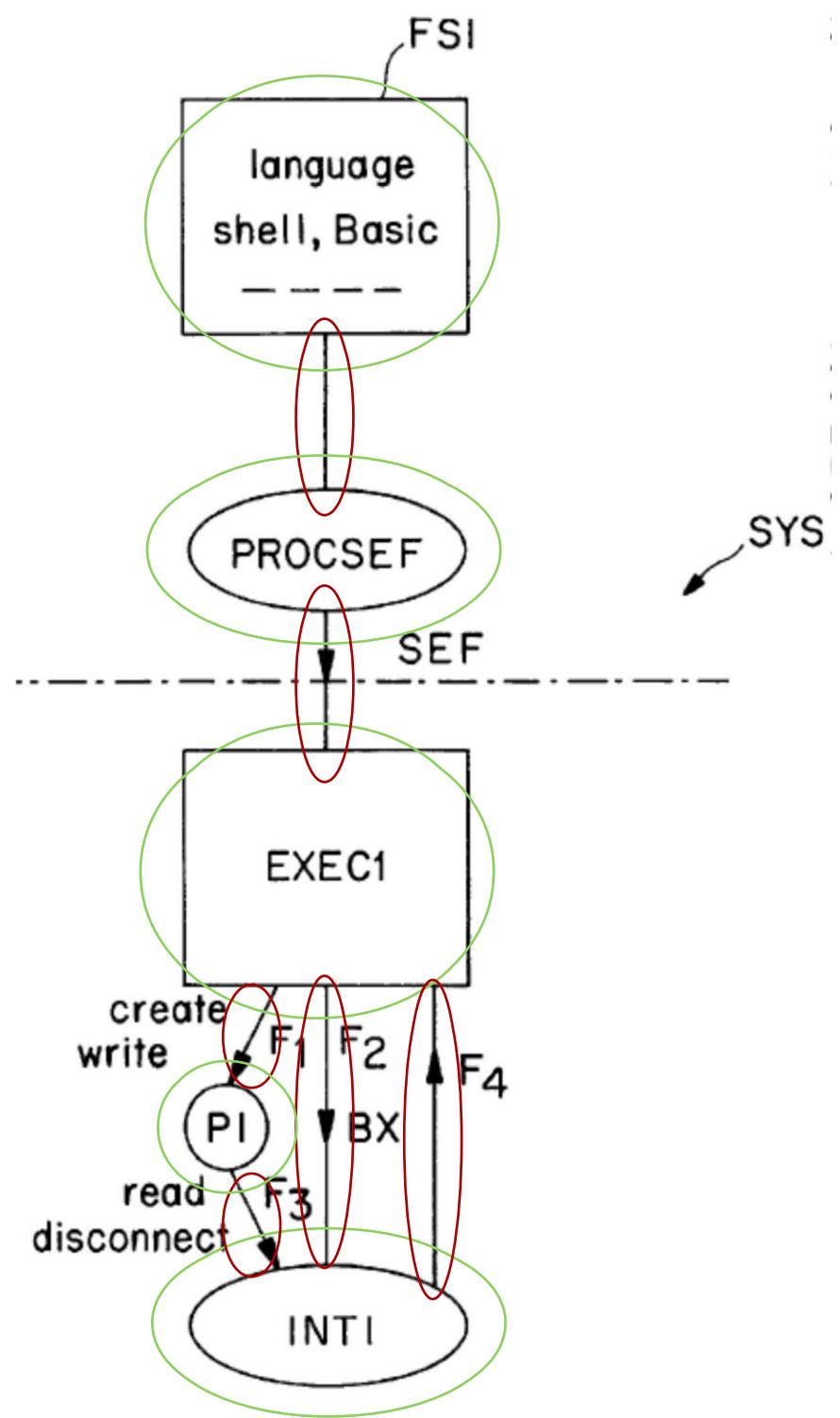


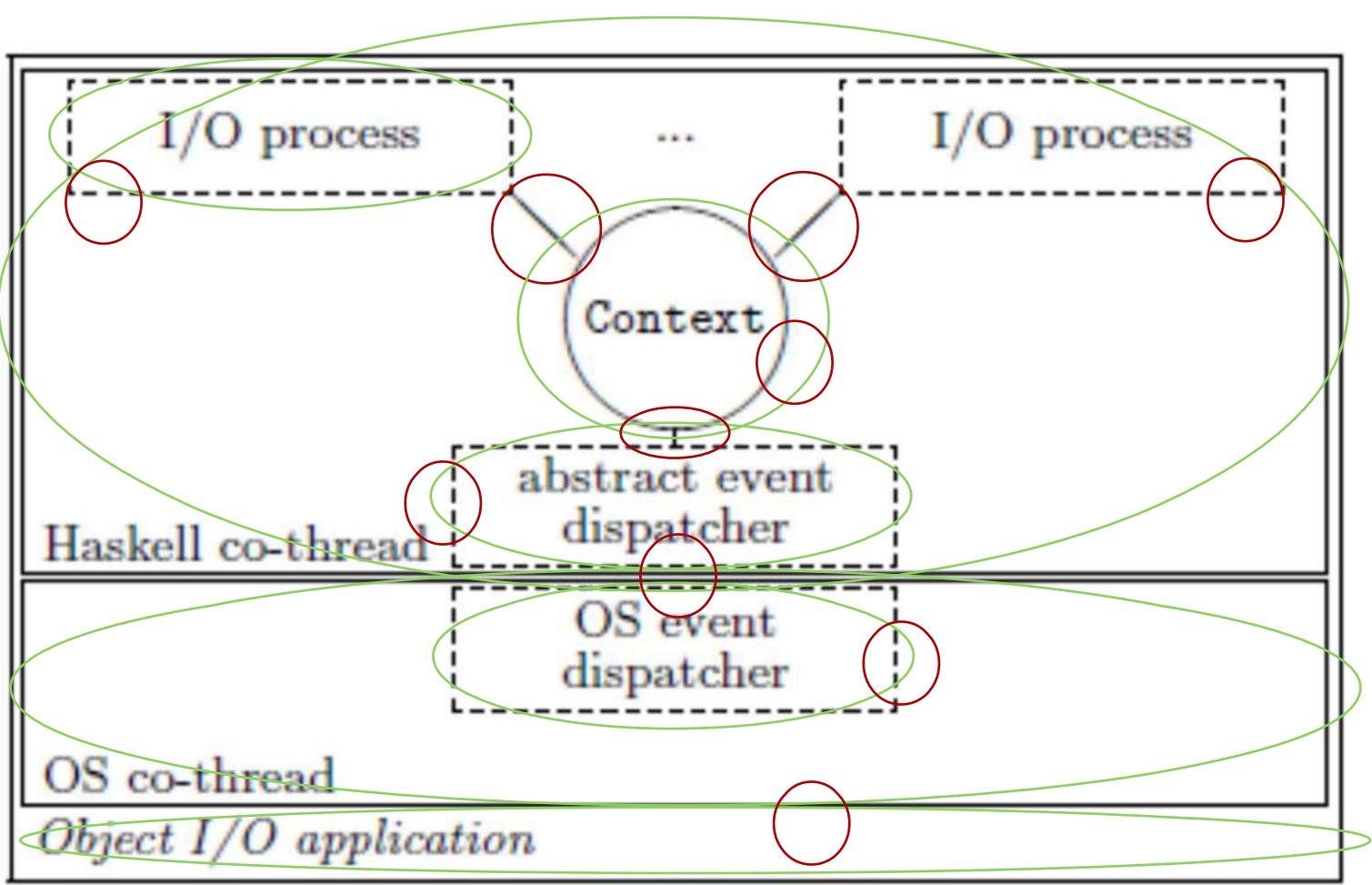


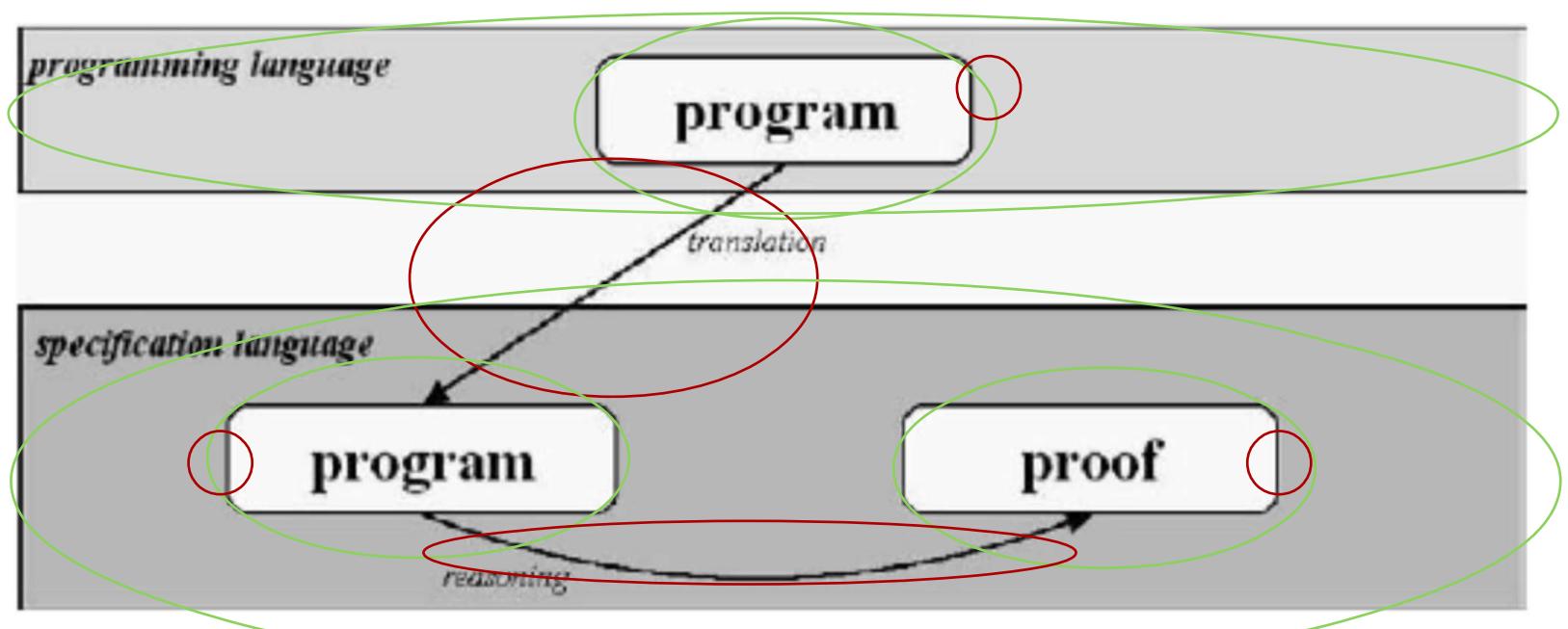


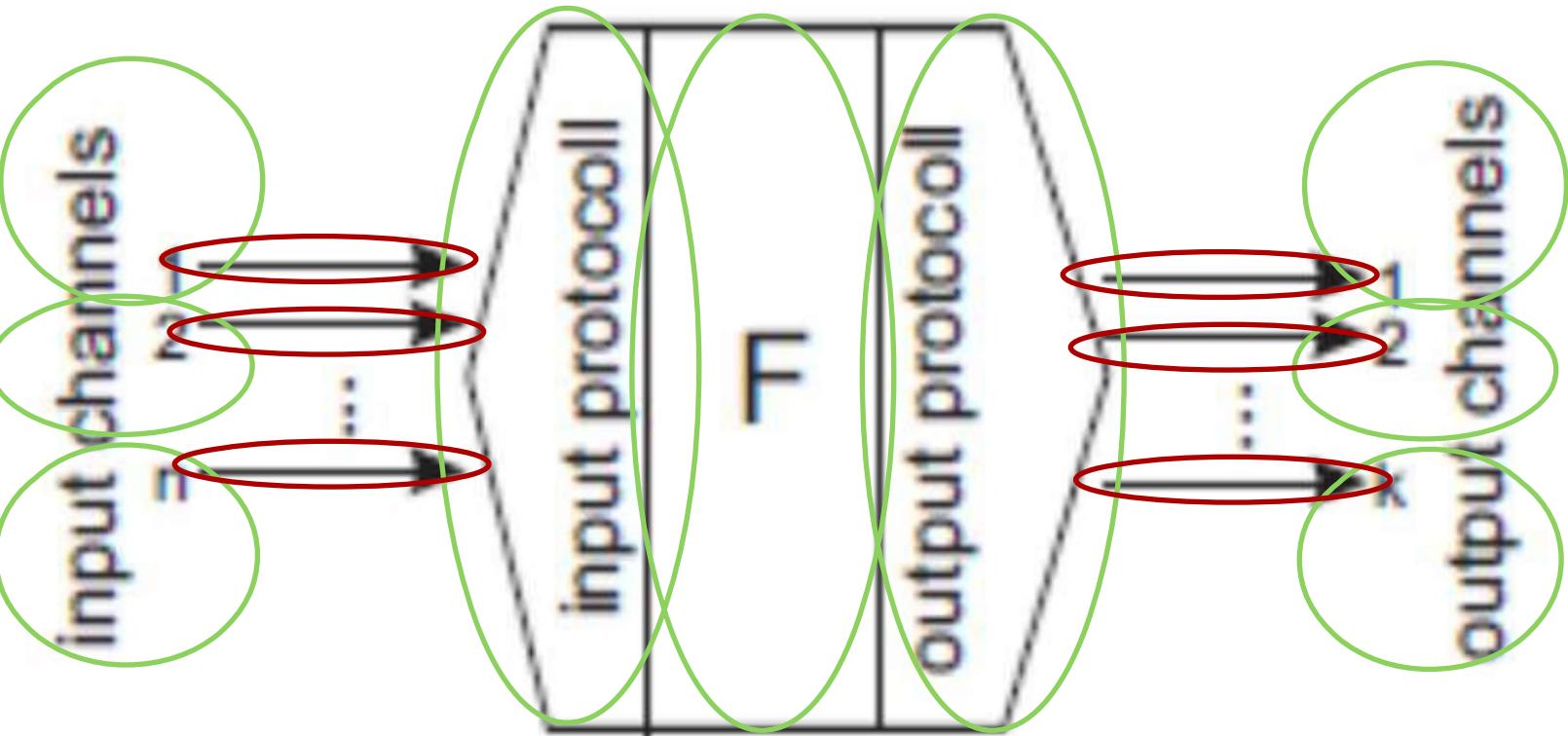


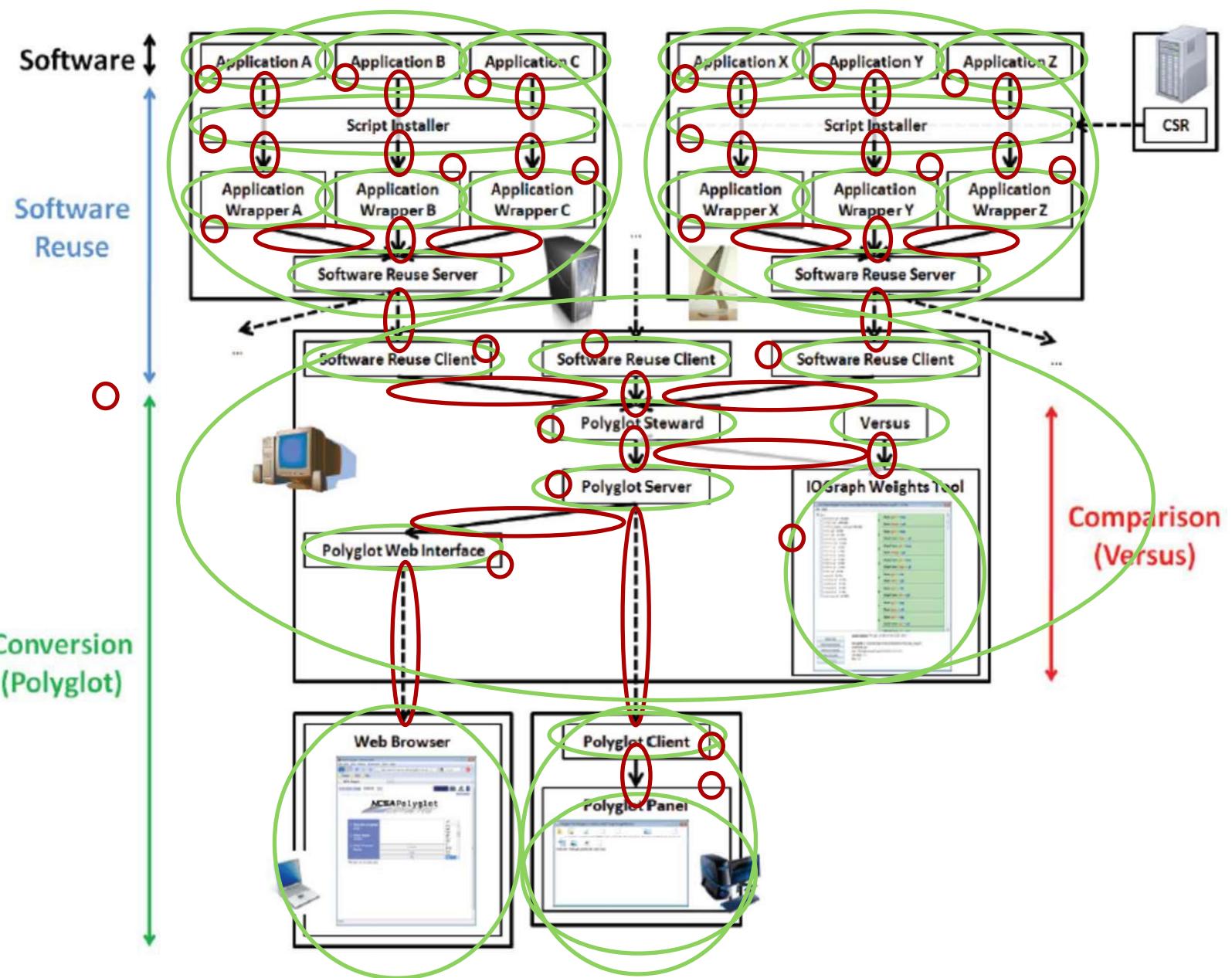


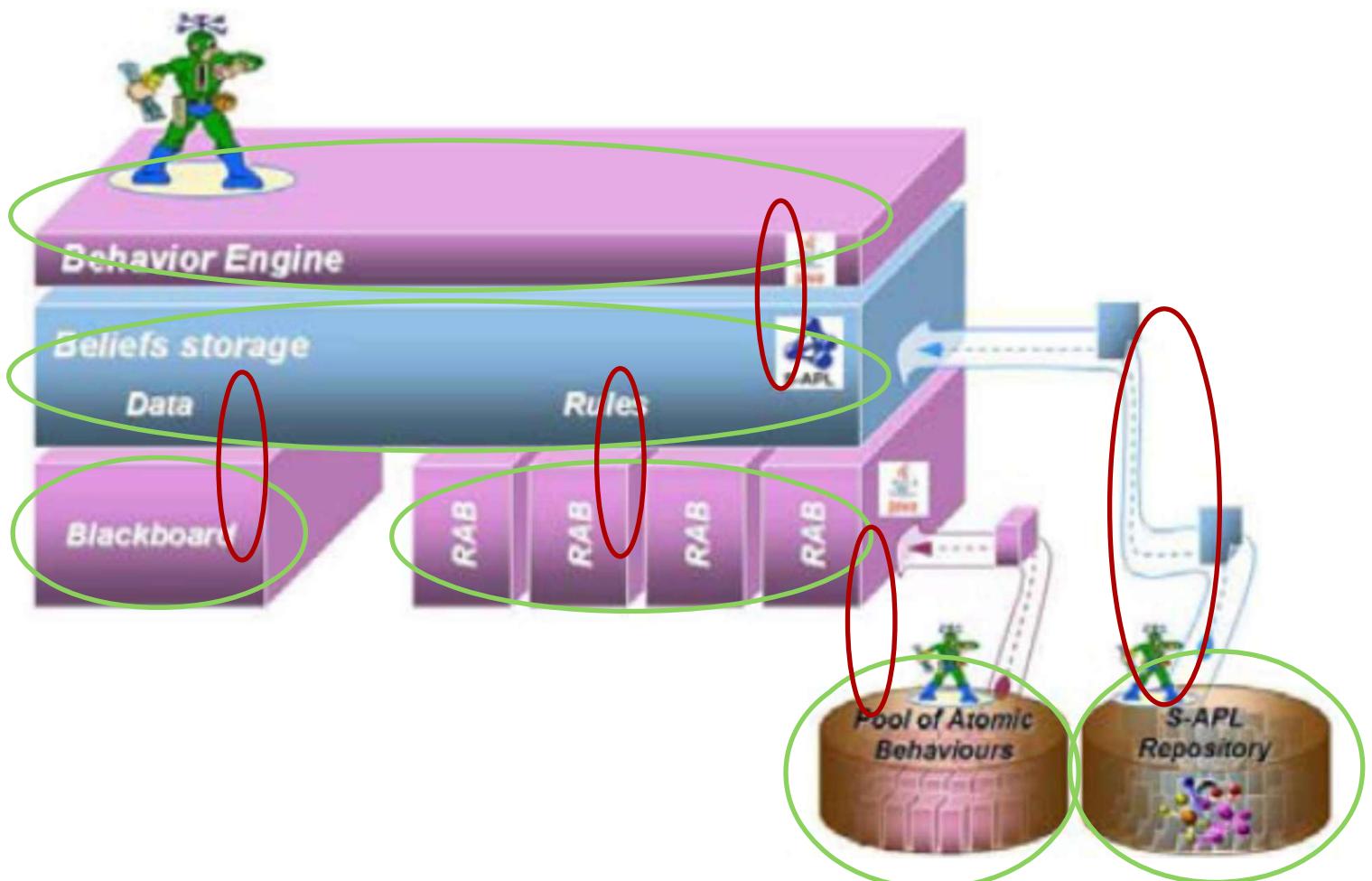


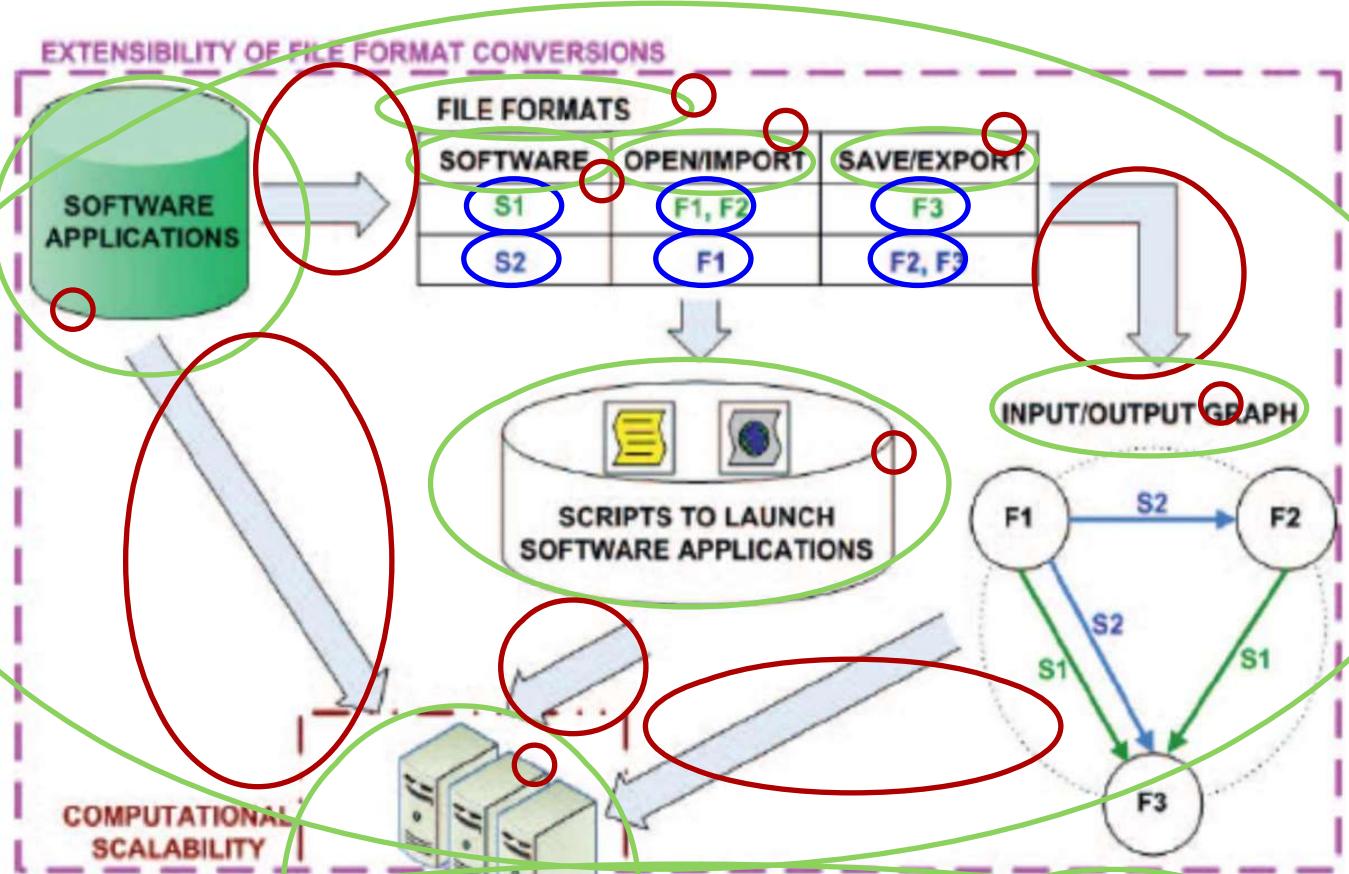




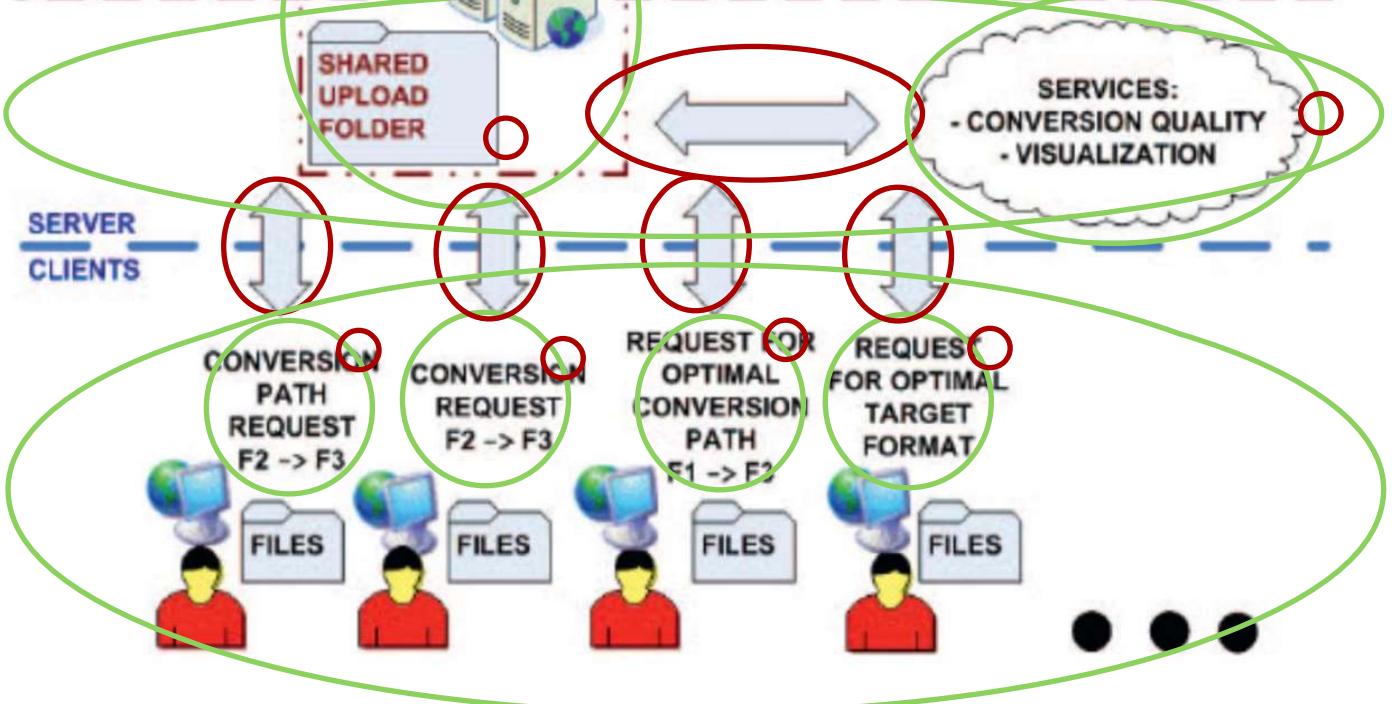


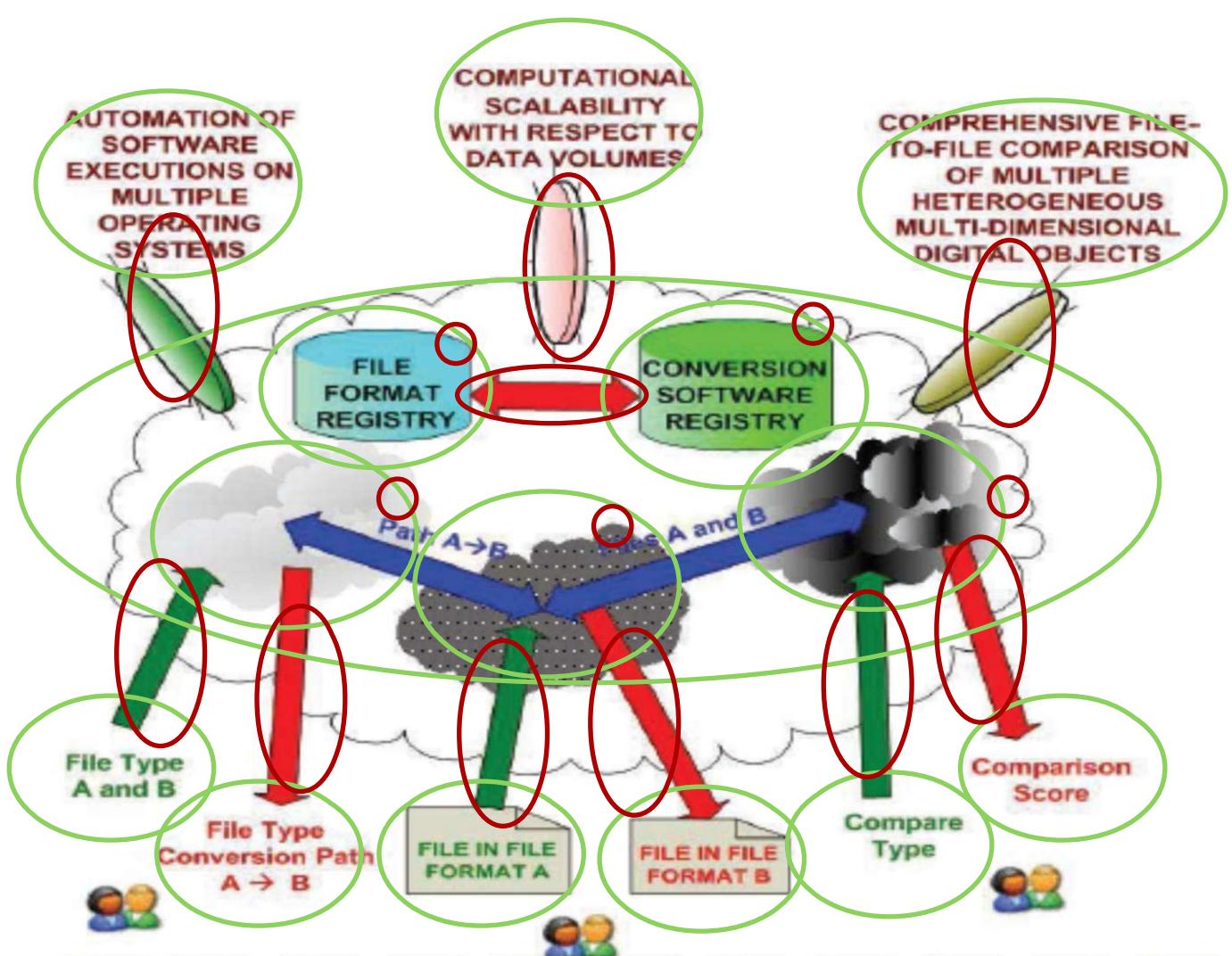




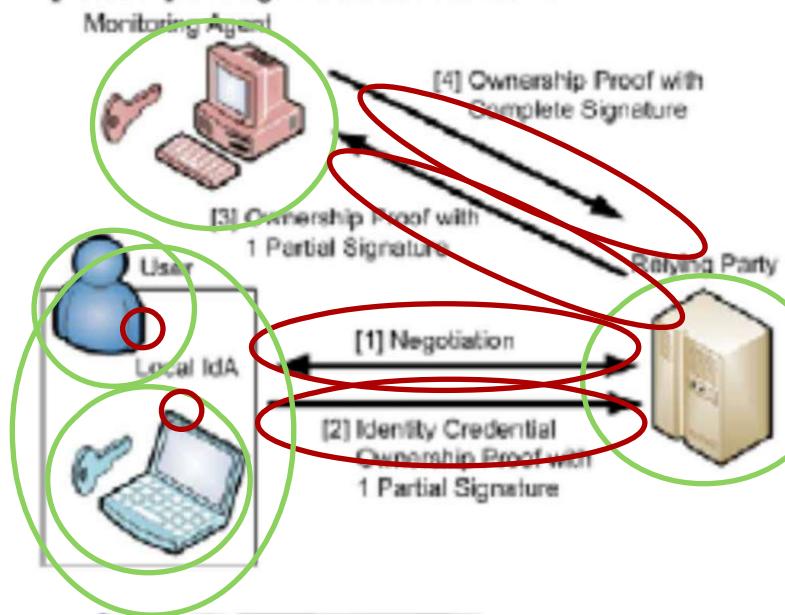


COMPUTATIONAL SCALABILITY

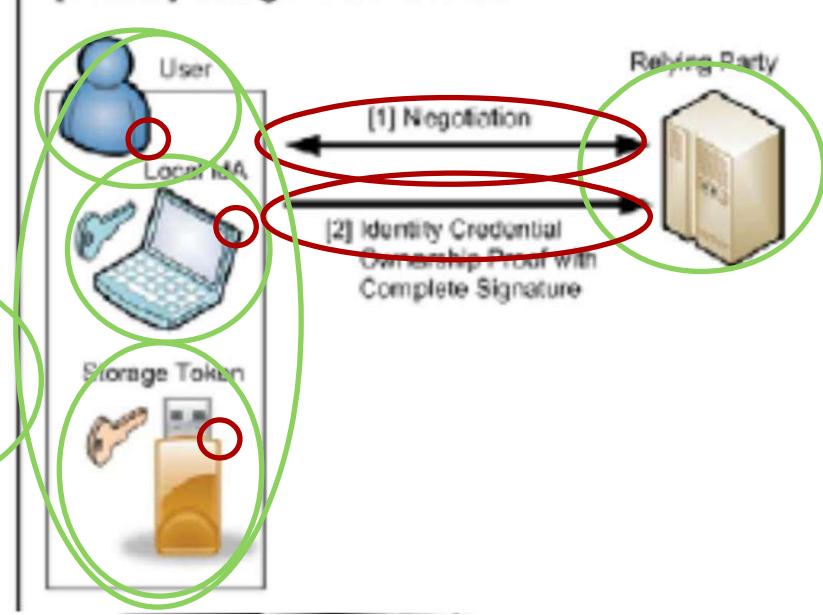


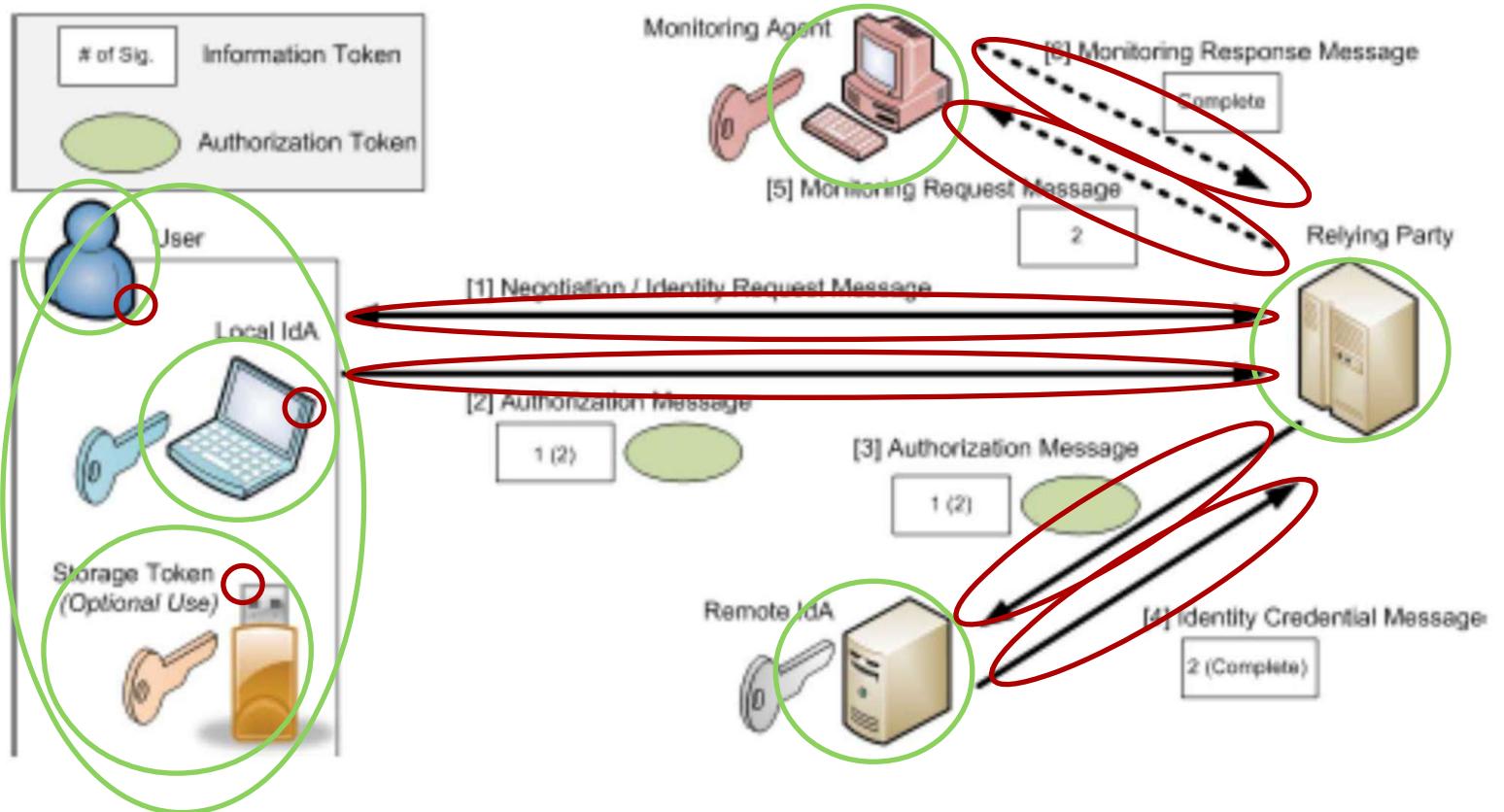


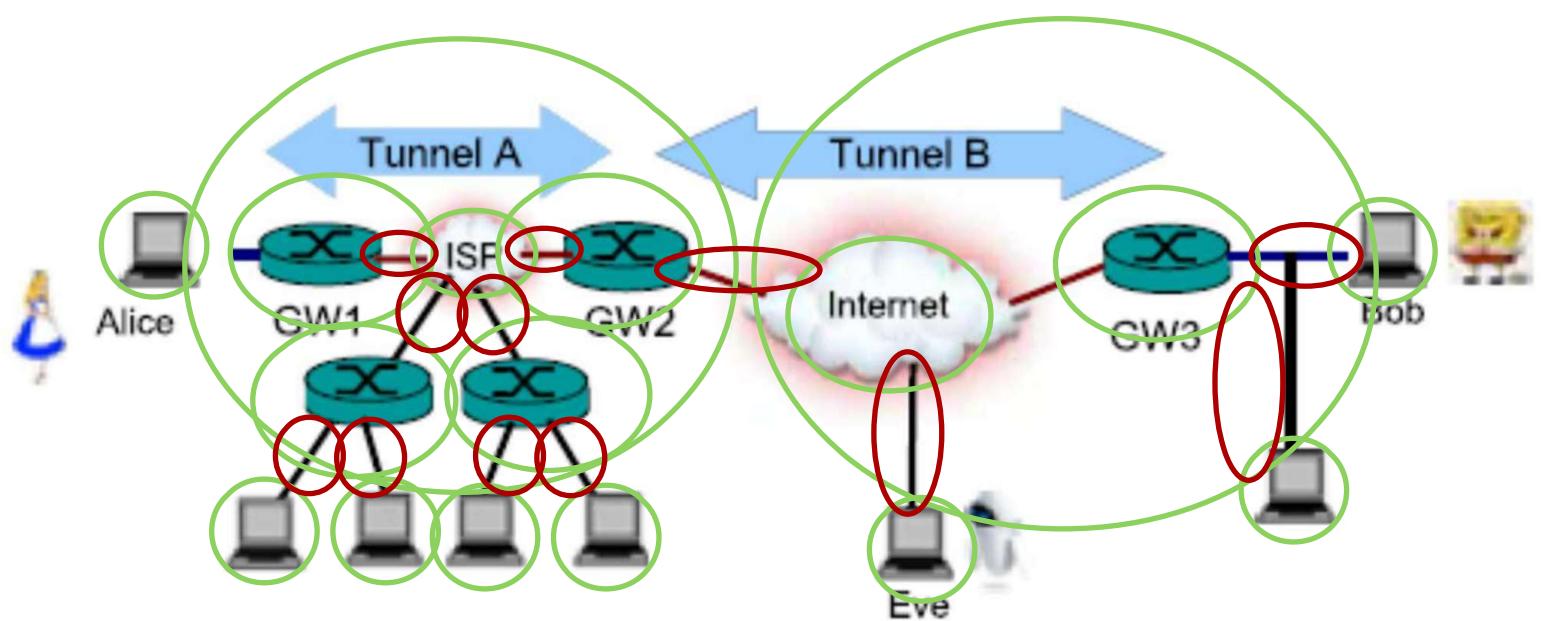
[CASE 1] Storage Token is not used.

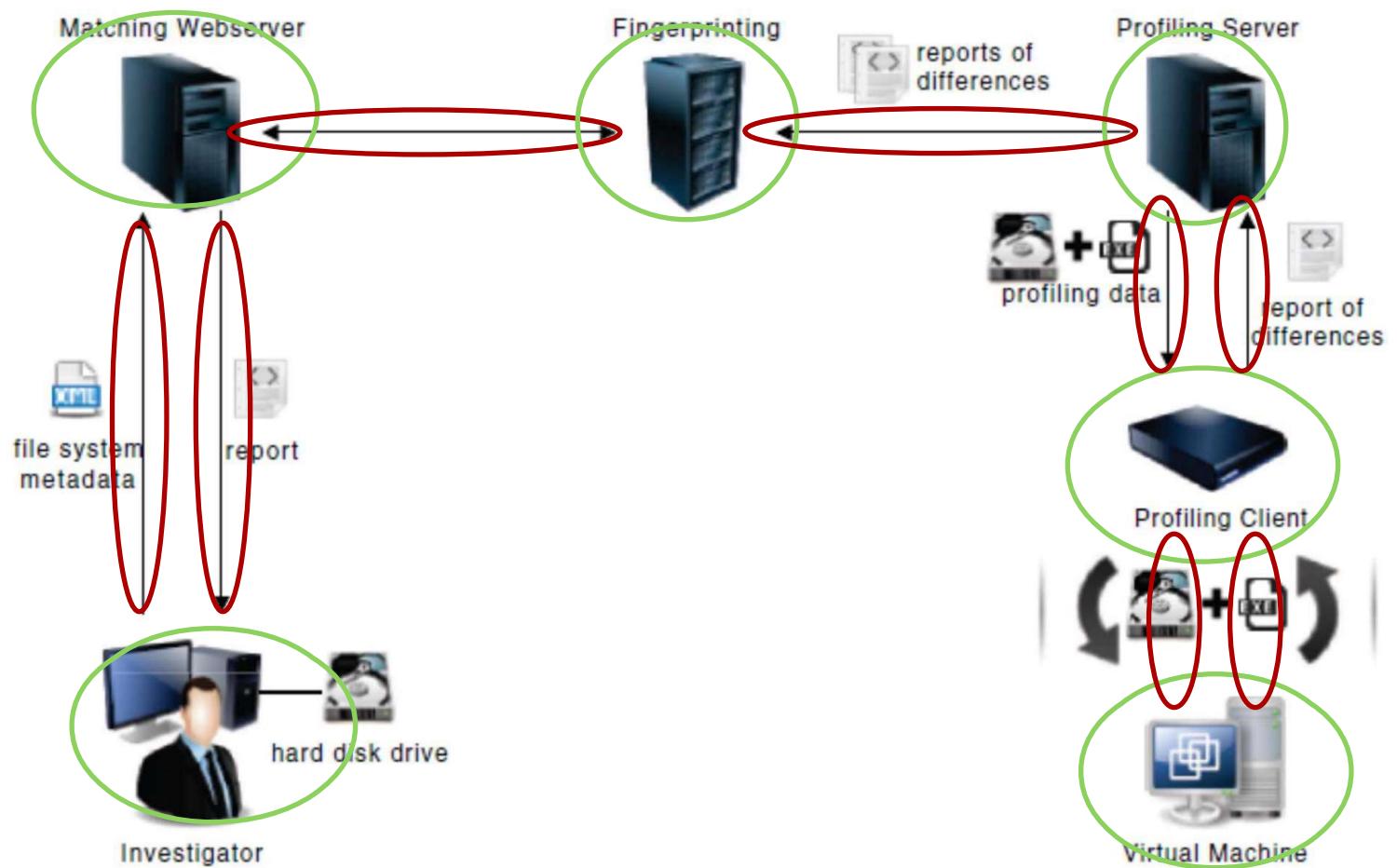


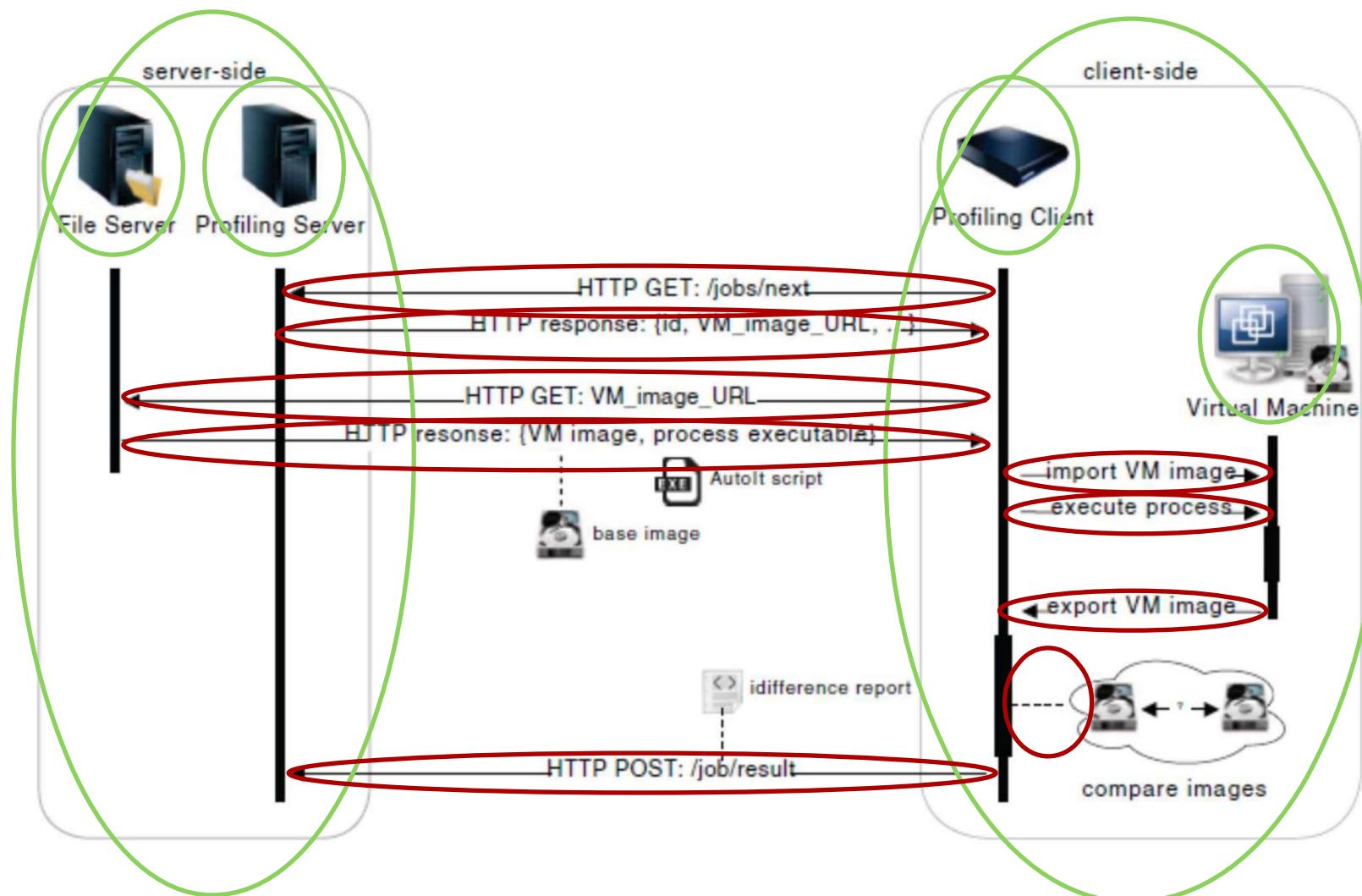
[CASE 2] Storage Token is used.

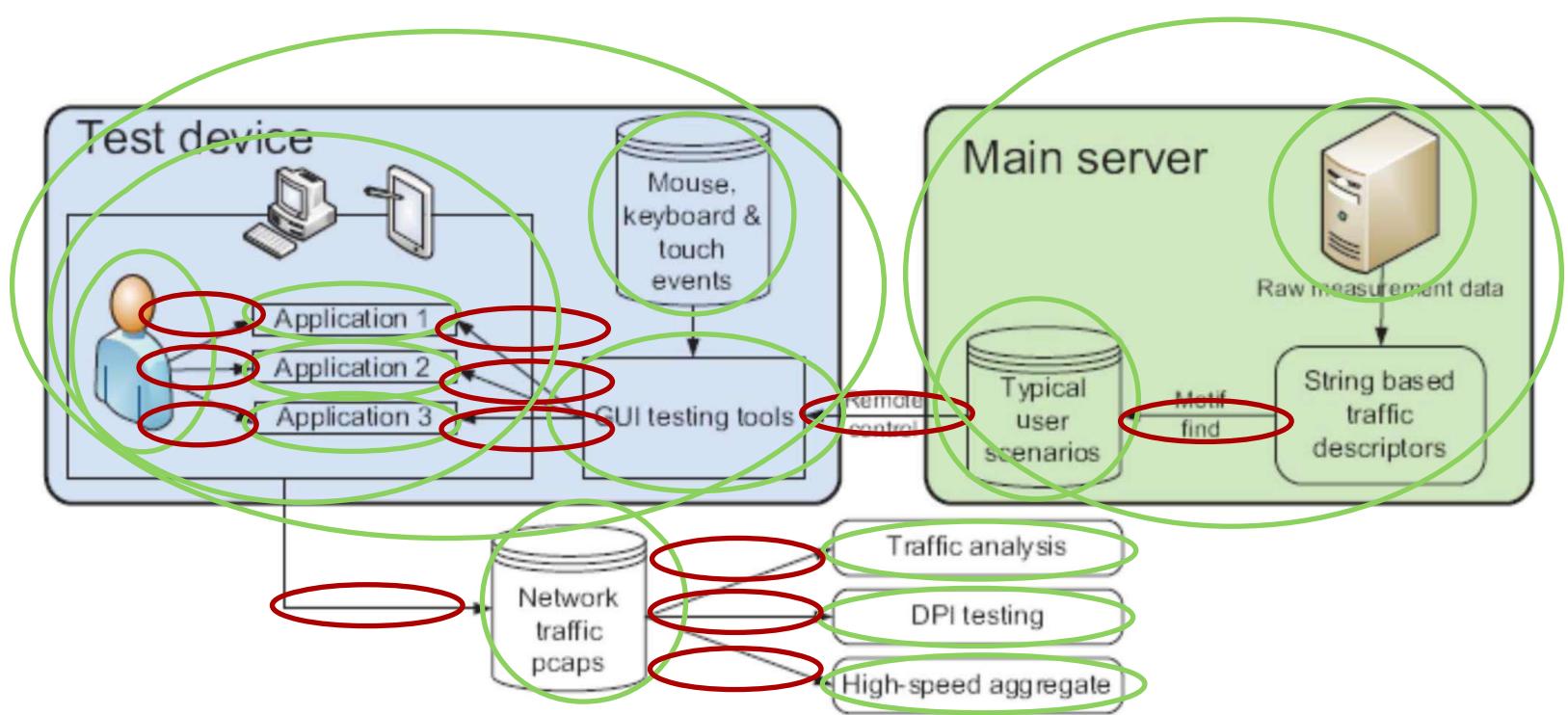


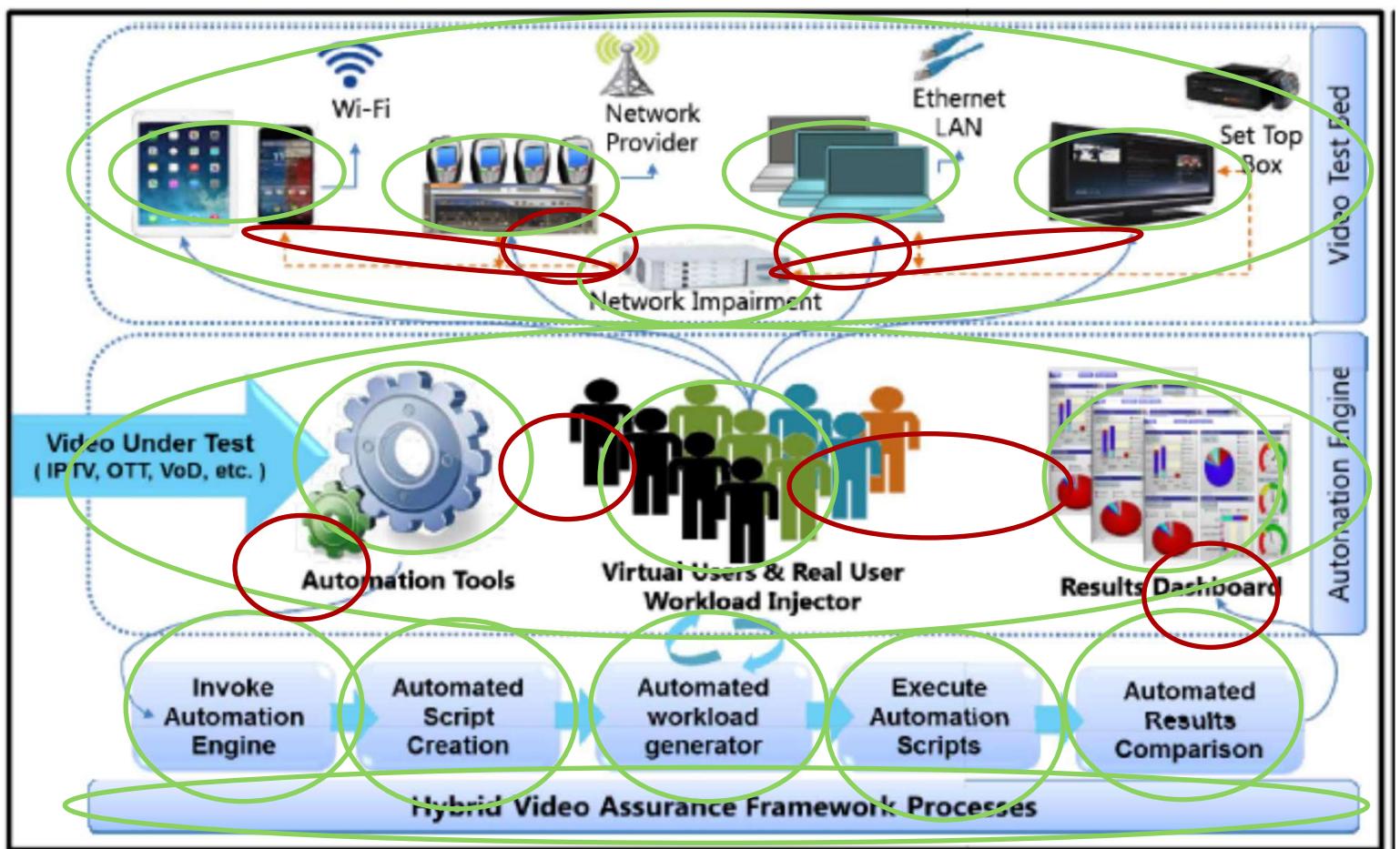


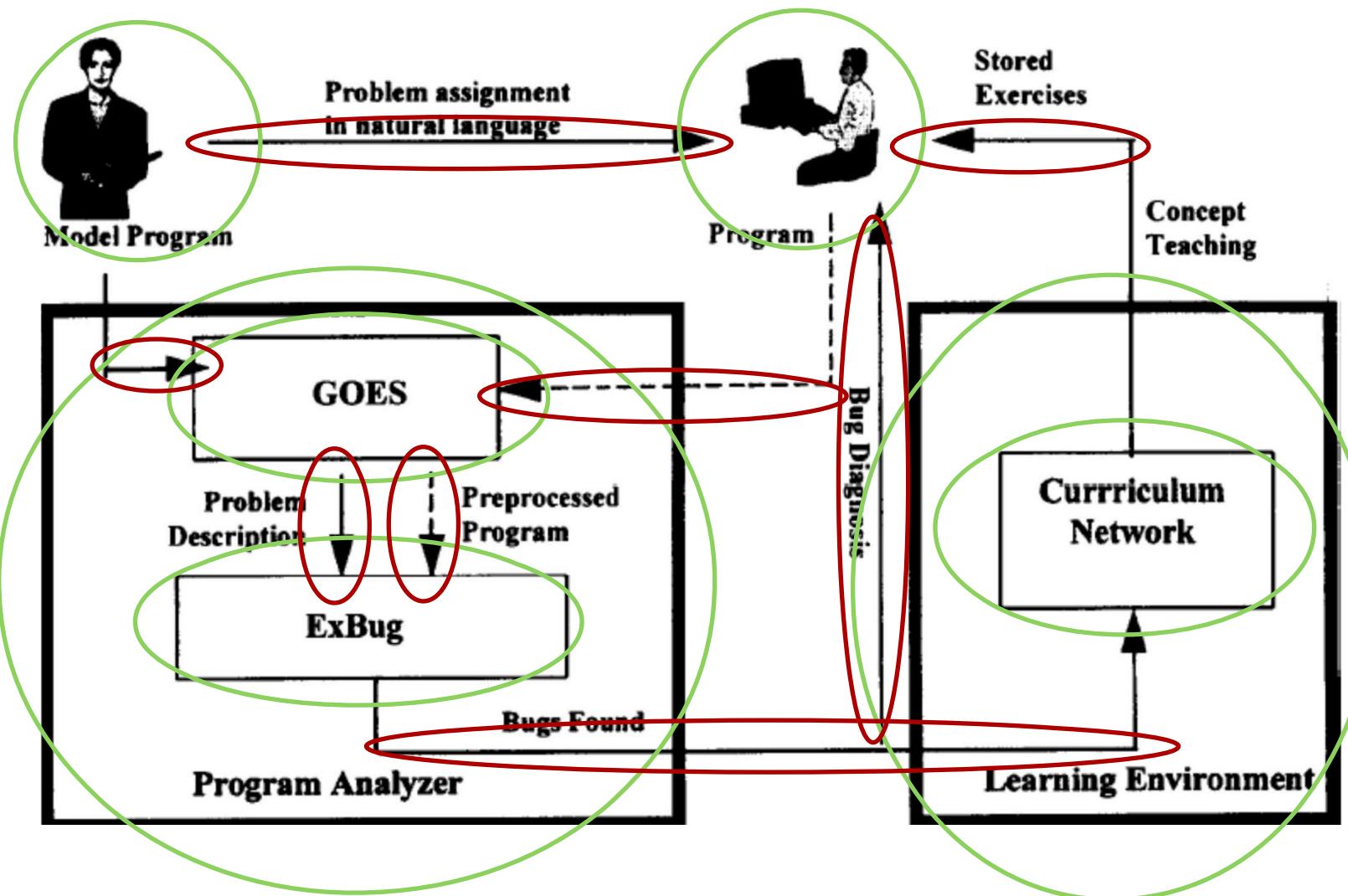


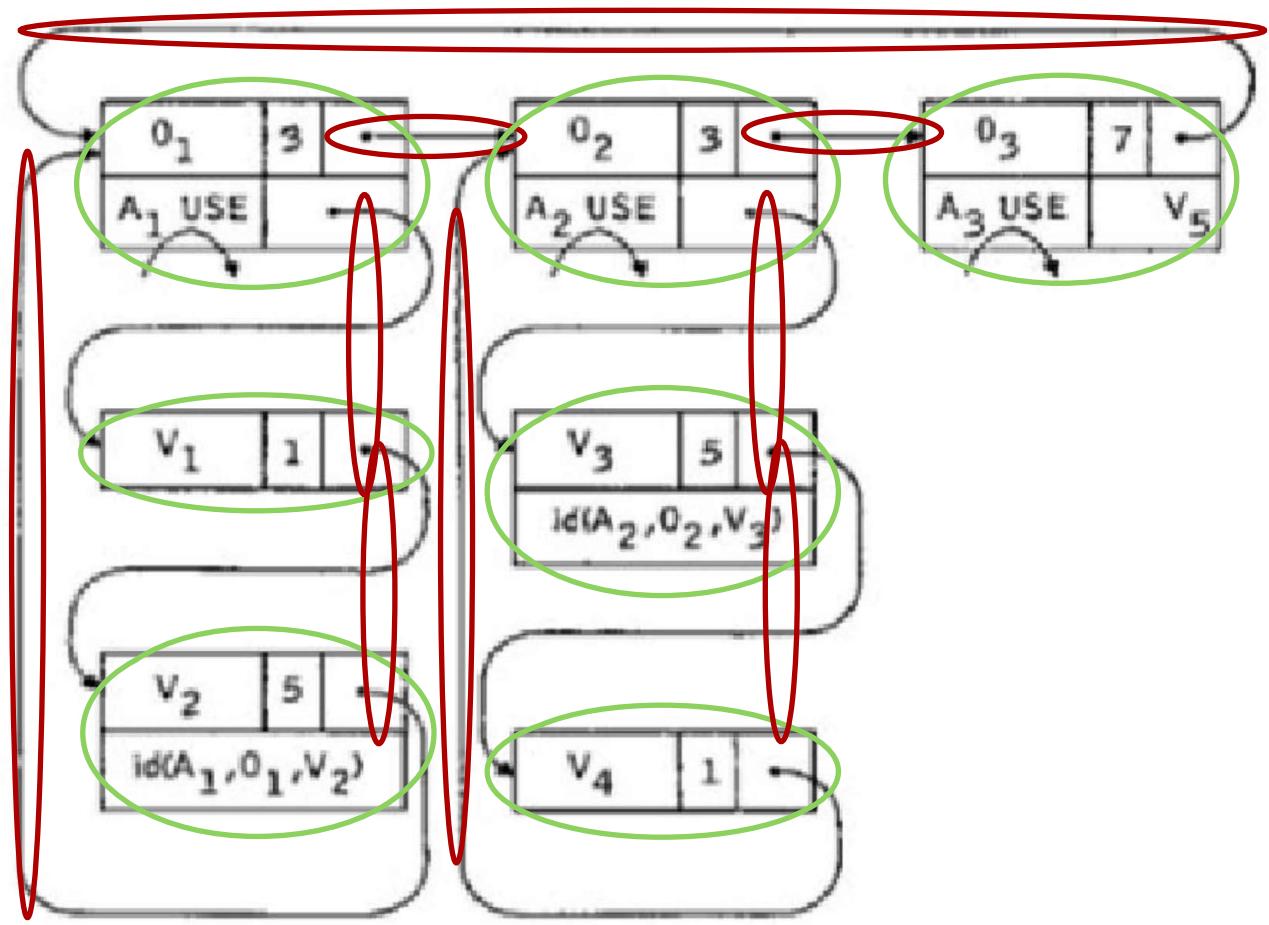


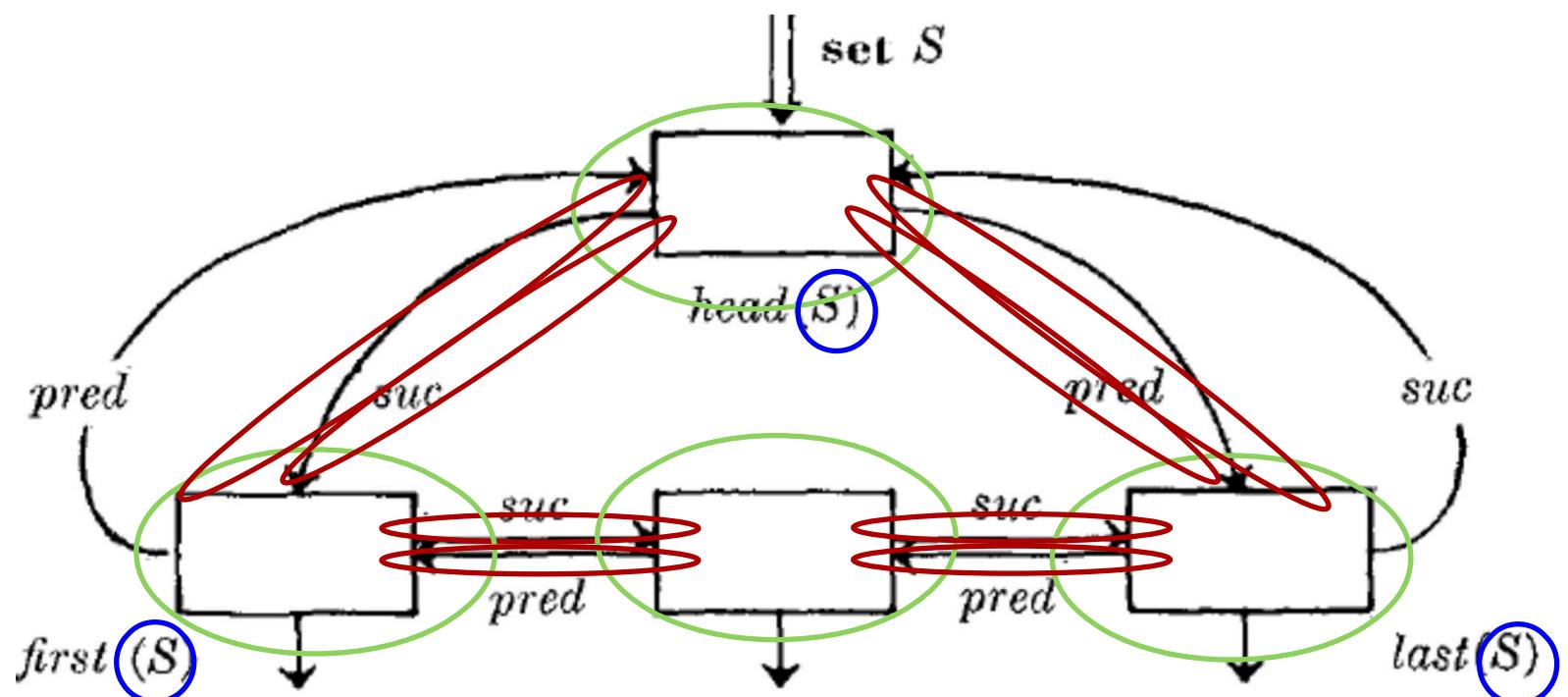


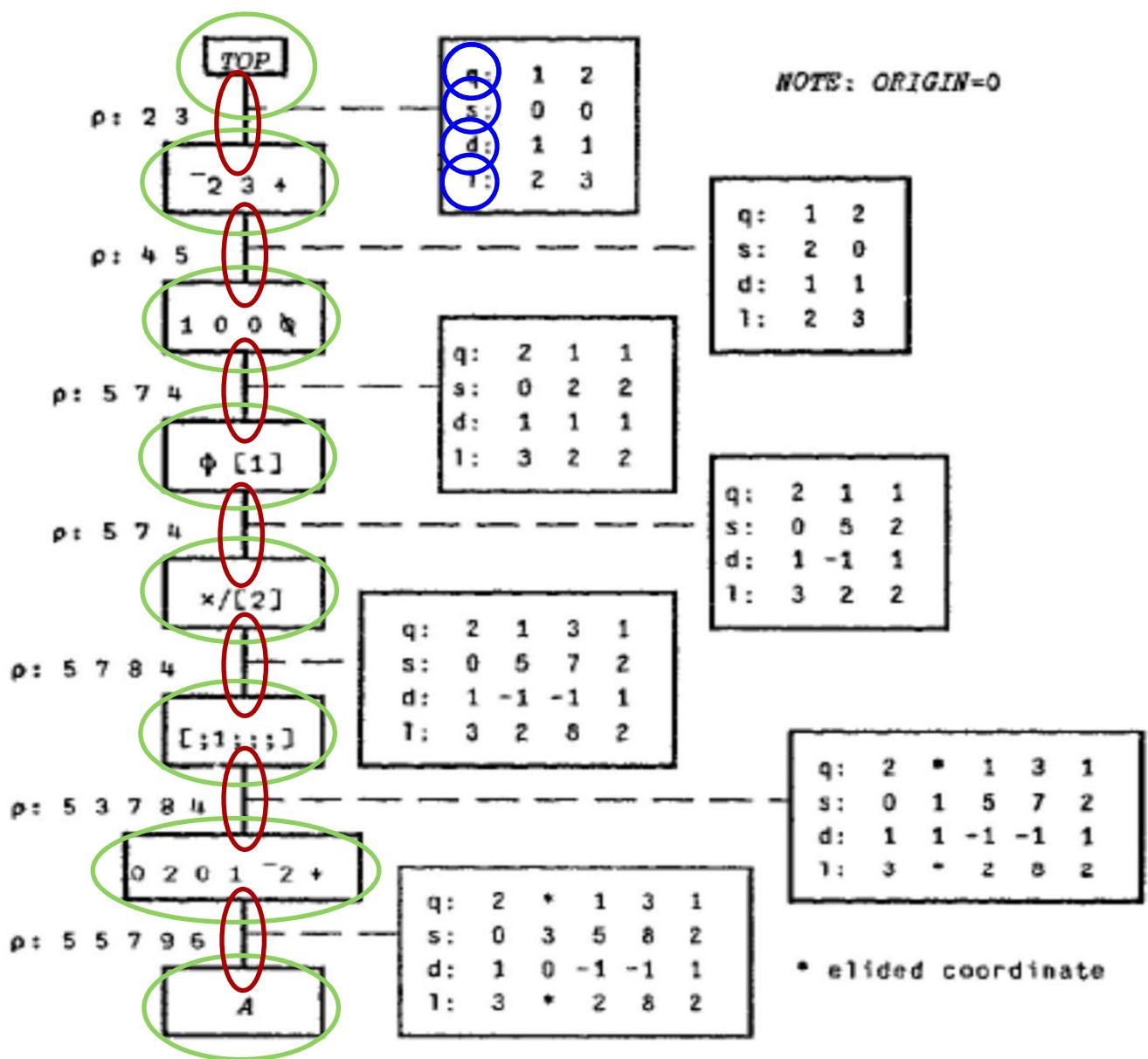


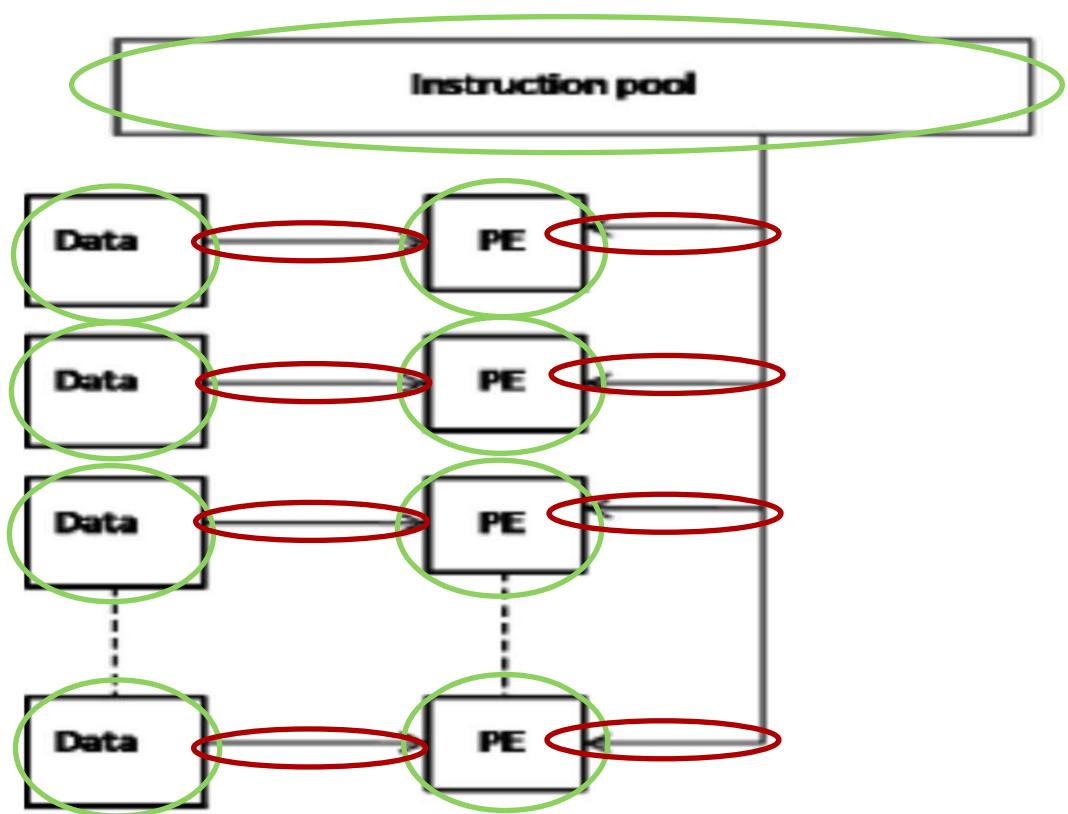


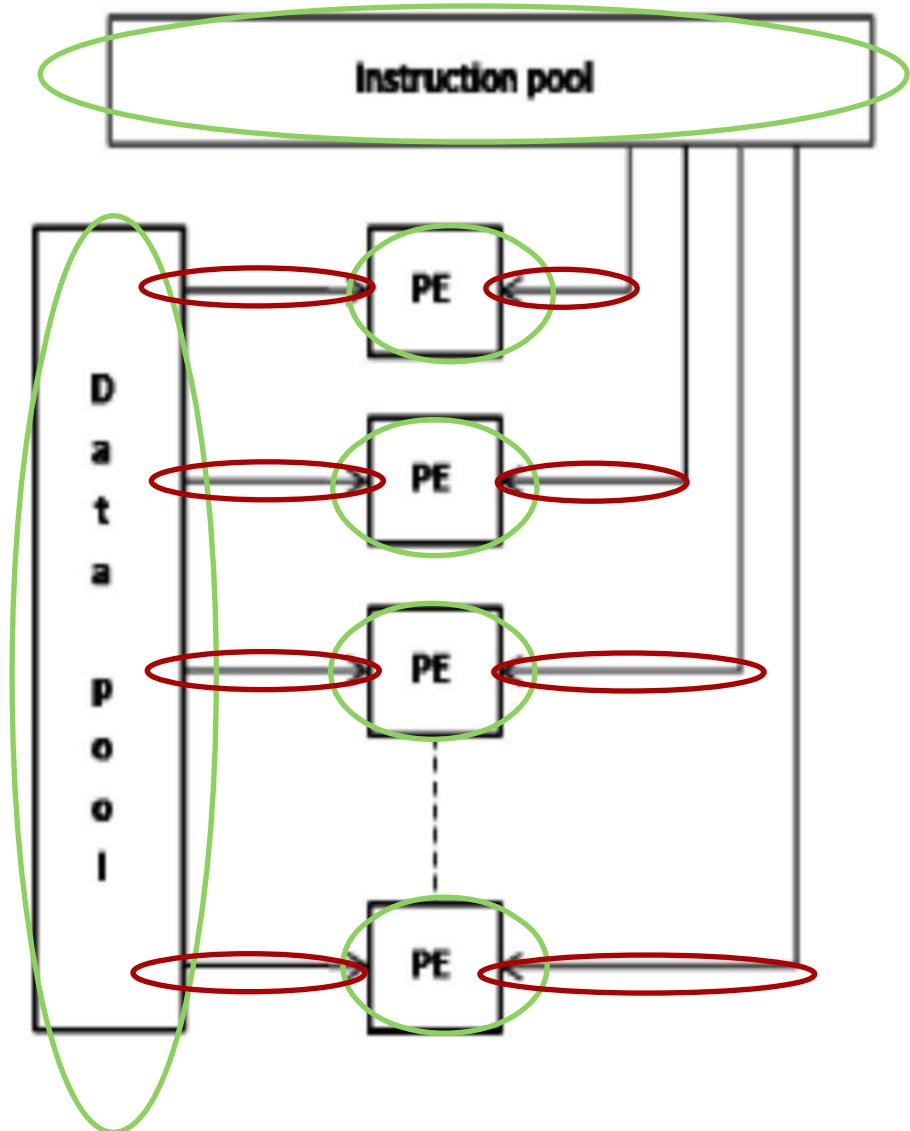


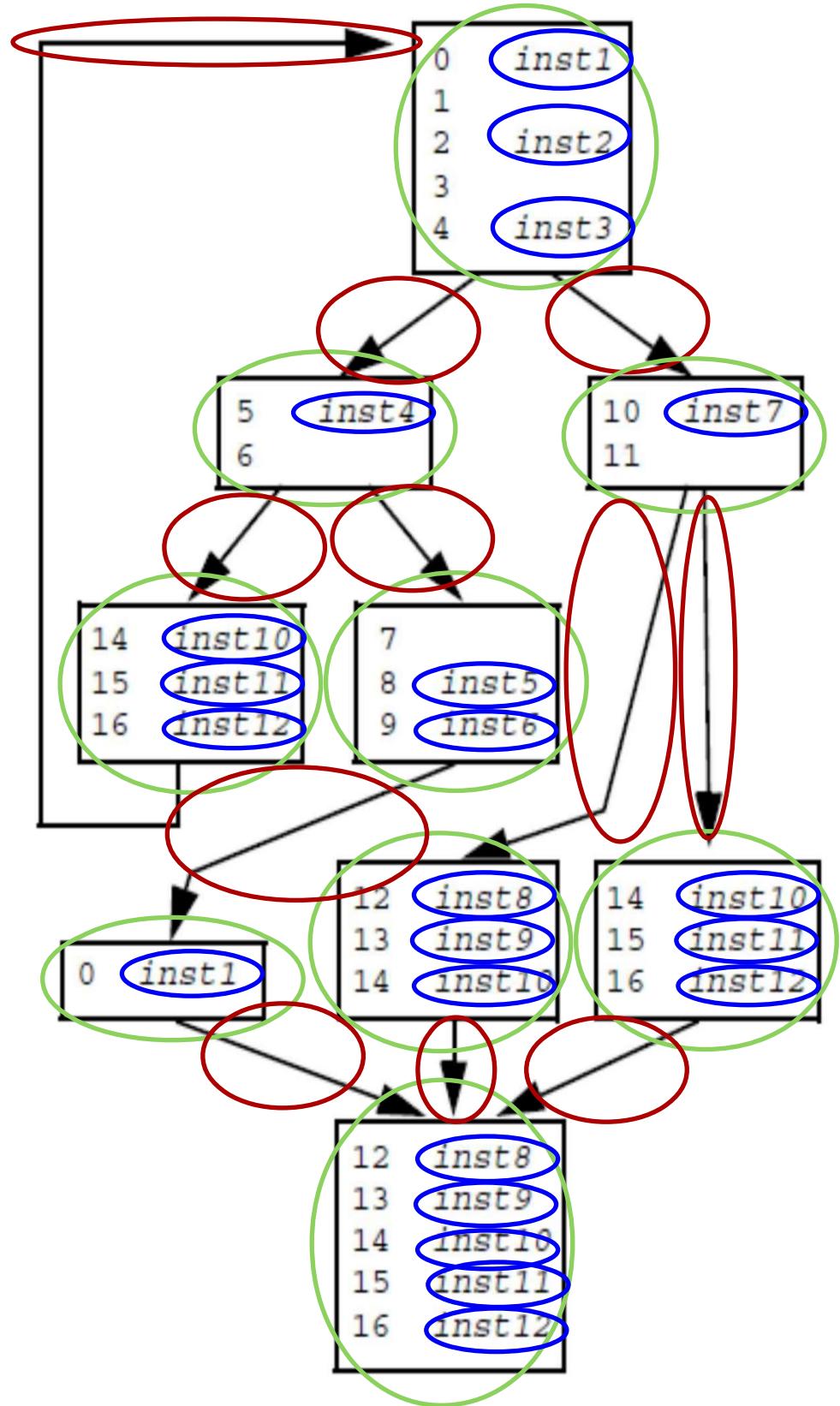


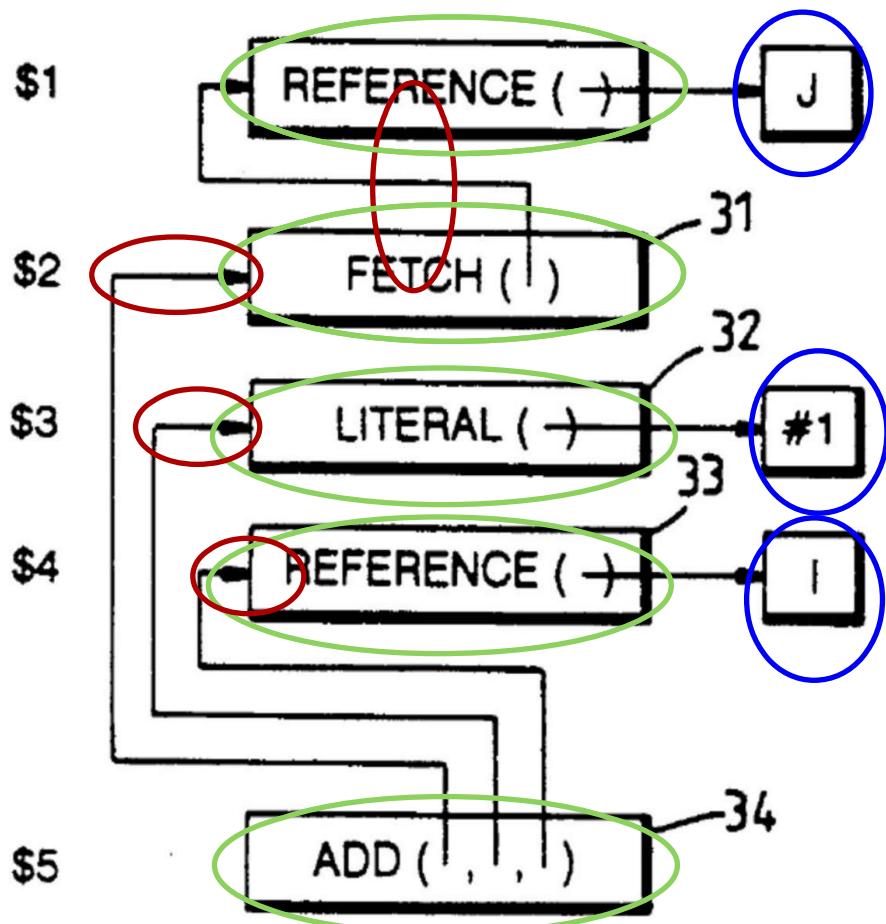












TCL

```
> set a 4  
> engine create eng  
-nosplash -nojvm  
> eng set b a  
> eng eval c=b^2  
> set d [eng get c]  
> engine destroy  
eng
```

Create Session

Send
Variable

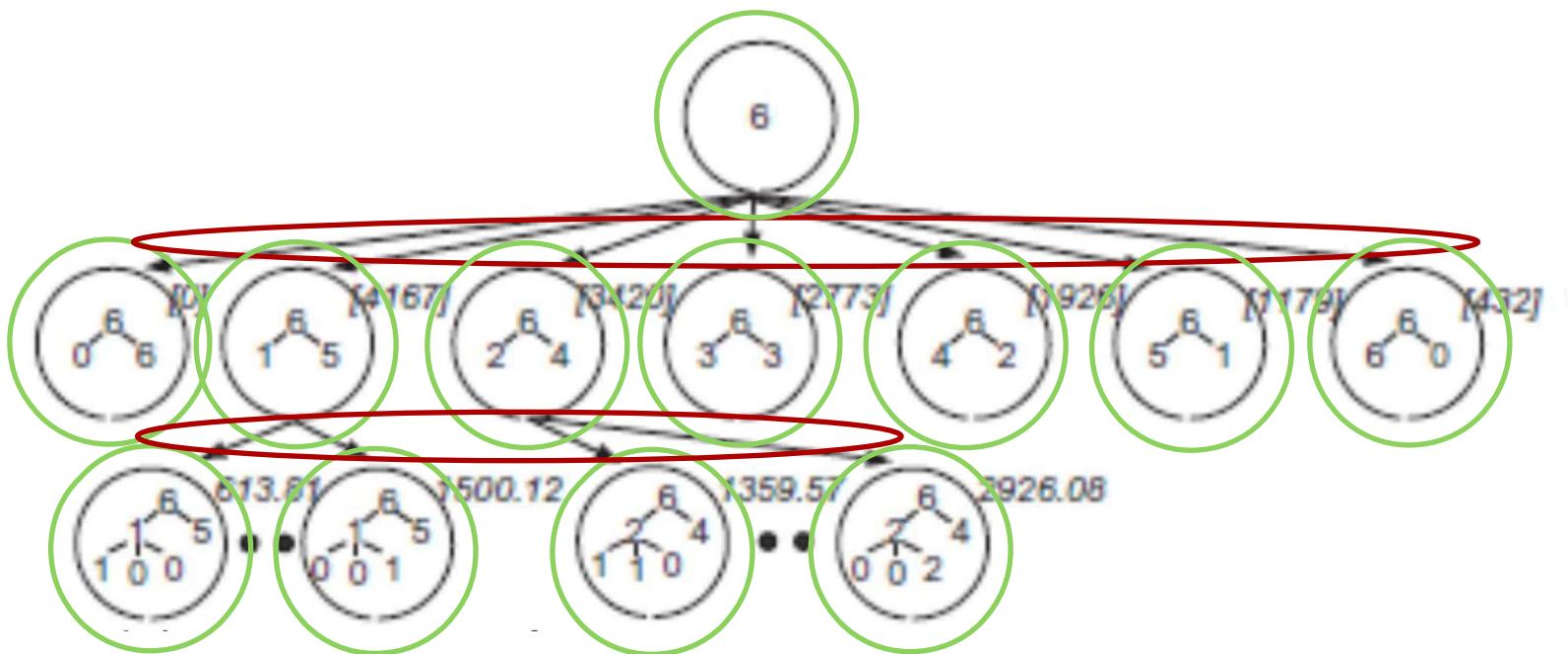
Evaluate
Function

Get
Variable

Destroy Session

Matlab

```
>> b = 4  
>> c = b^2  
>>
```



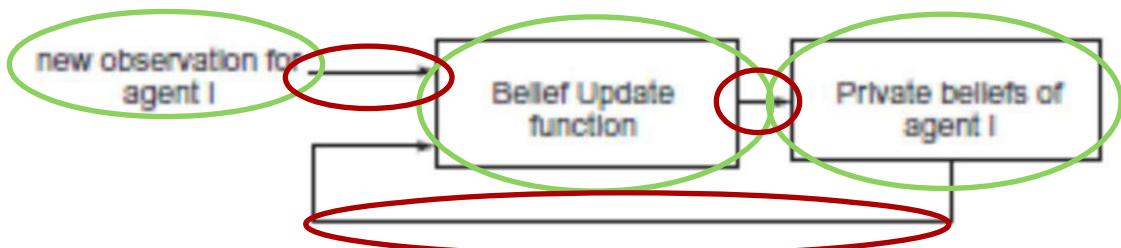
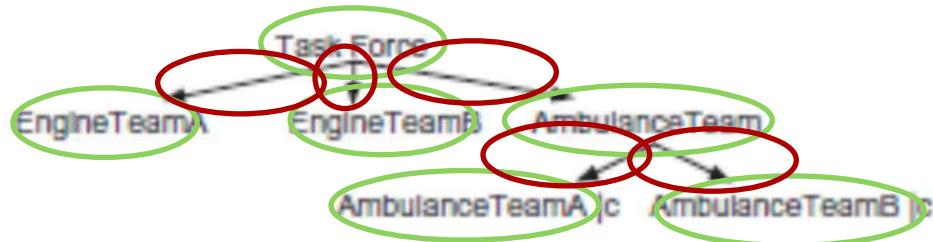
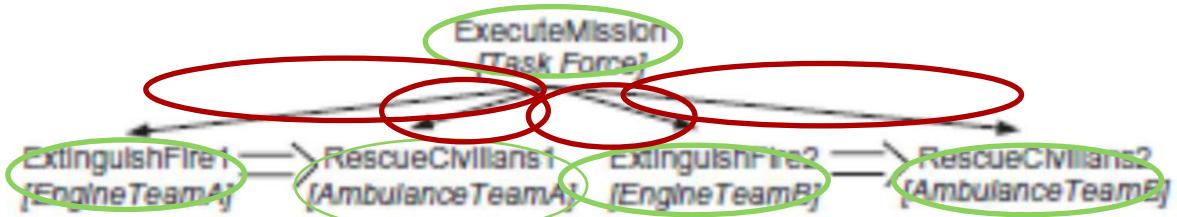


Fig. 6. Mapping of observations to beliefs



(a)



$A \leftarrow 11 \ 12$

$2 \bullet$
11 12

11 12

$B \leftarrow < A$

2
11 12

11 12

$C \leftarrow 2 \ 0 p 0$

2,0 •

$D \leftarrow 1 \ 1 \ 1 p < C$

1,1,1
2,0

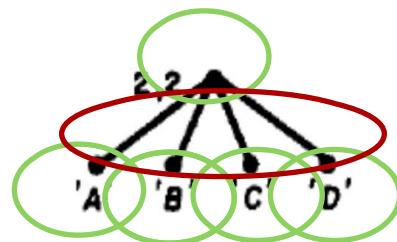
□

$E \leftarrow 2 \ 2 p 'ABCD'$

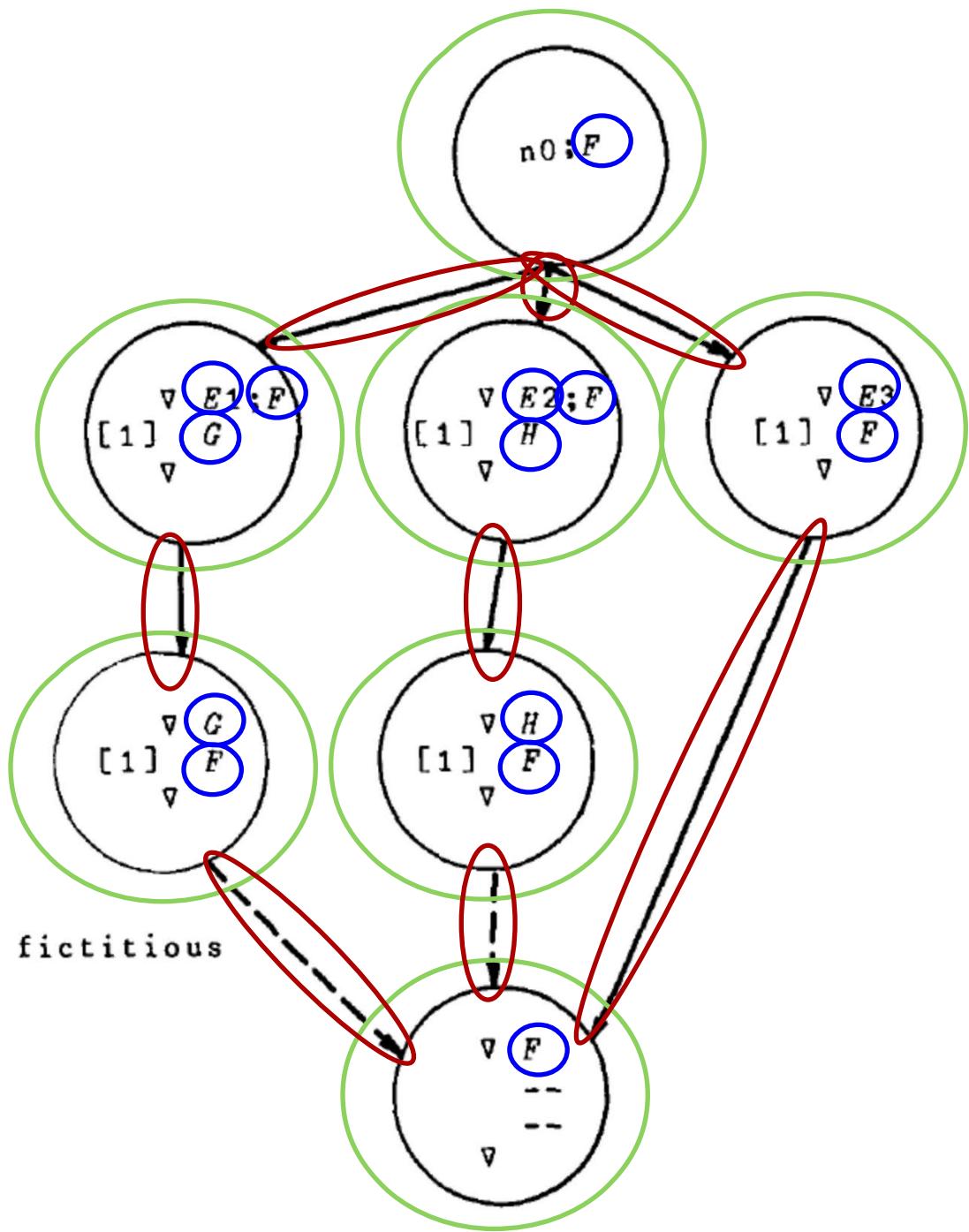
2,2 •
'ABCD'

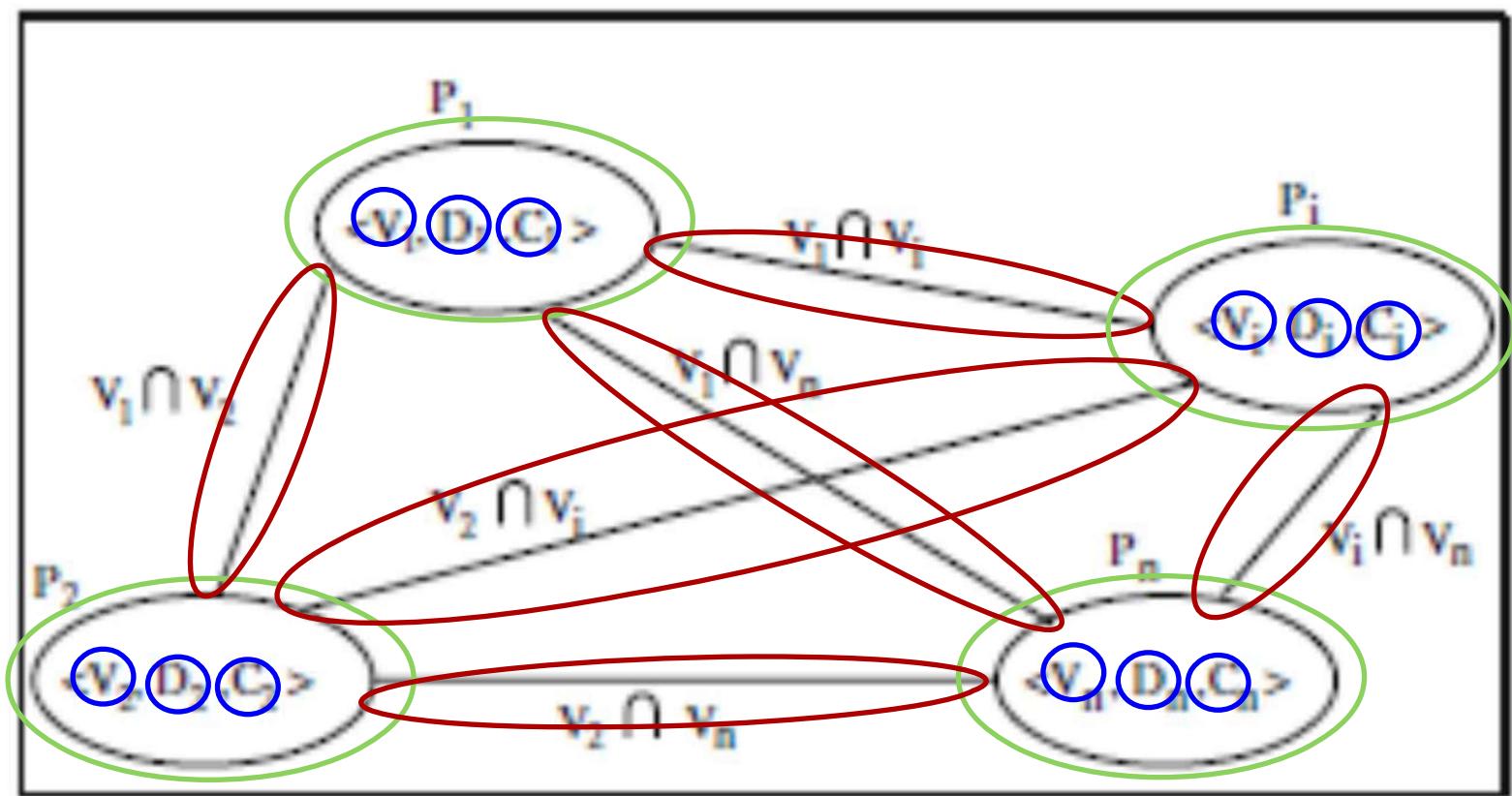
AB
CD

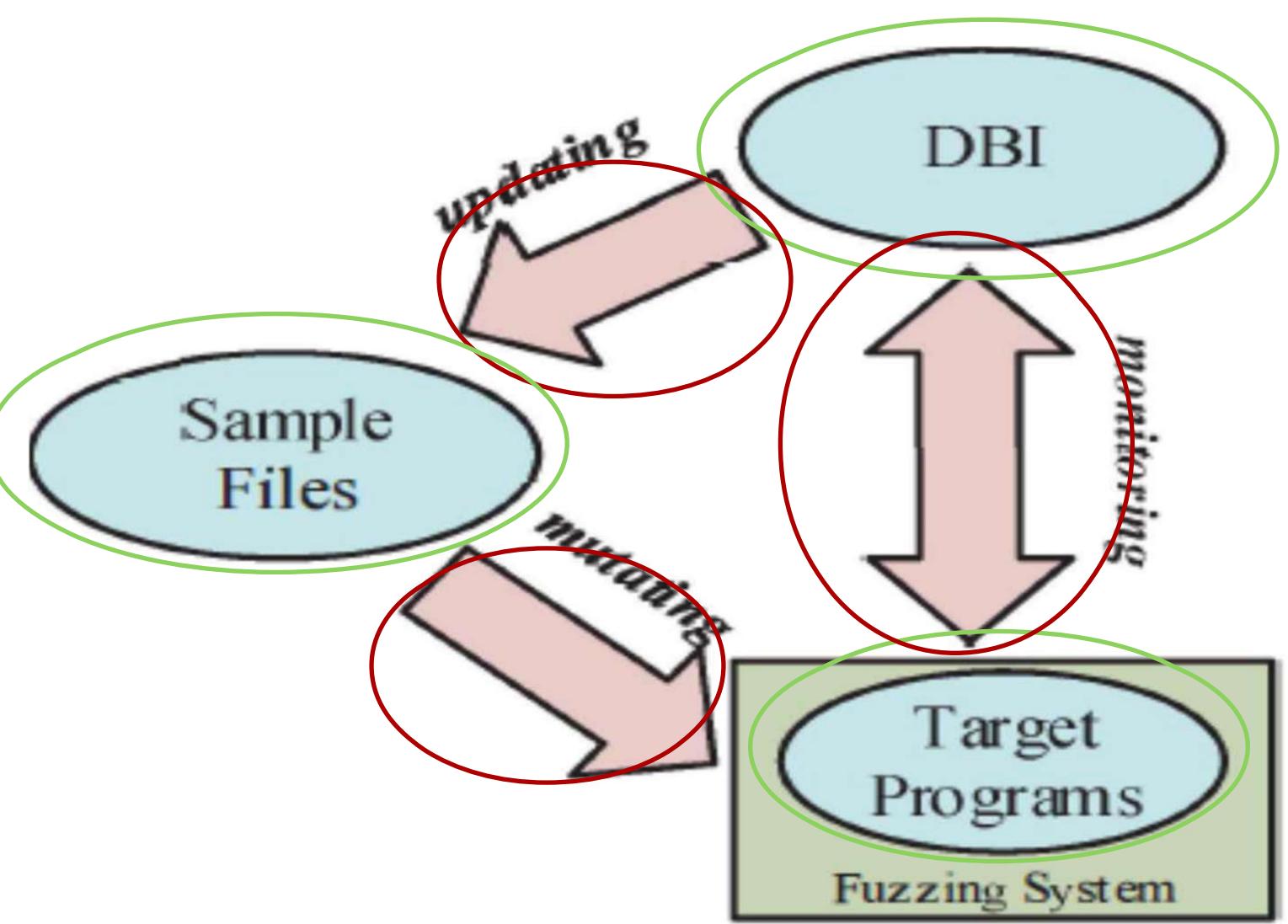
$F \leftarrow \downarrow E$

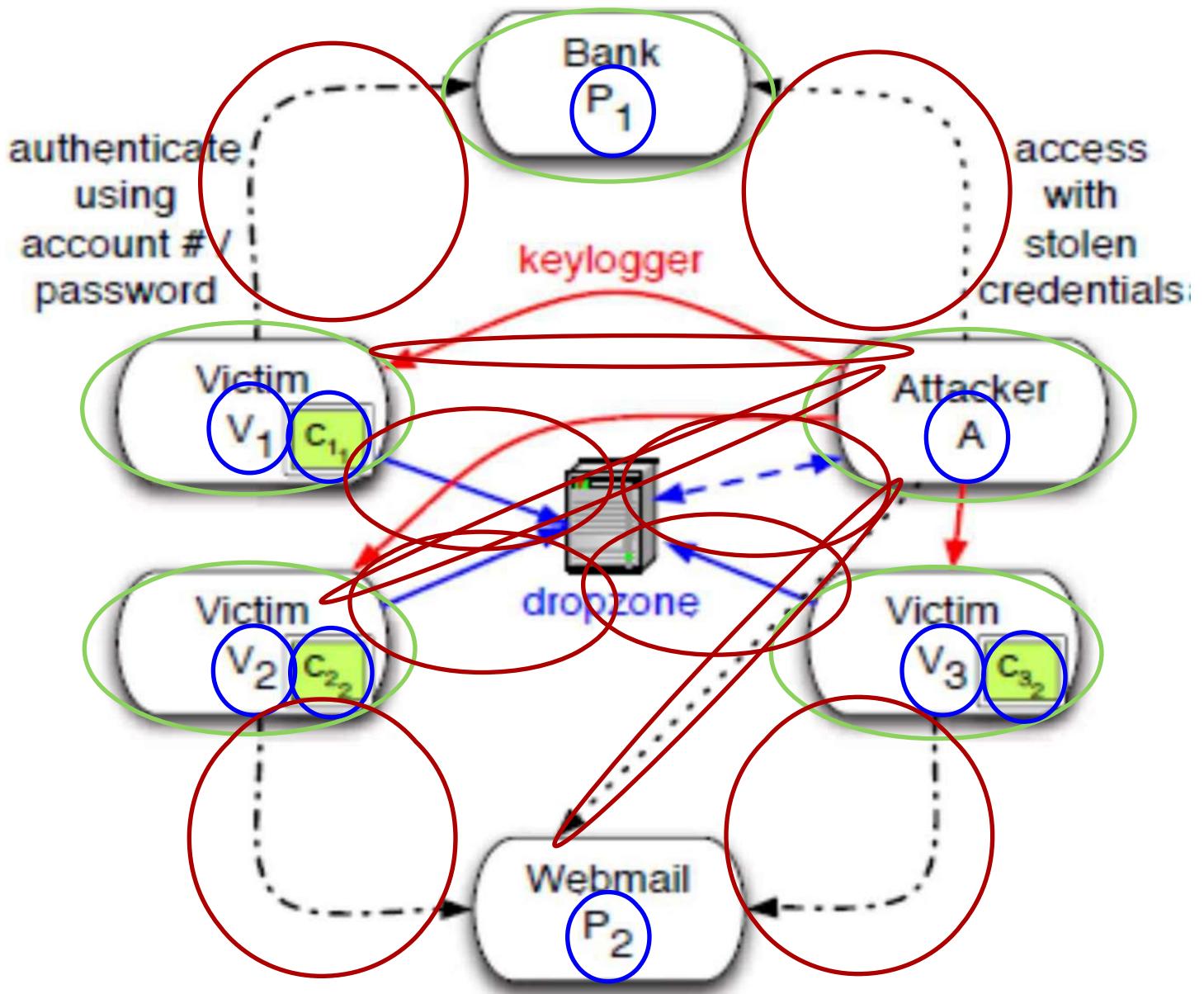


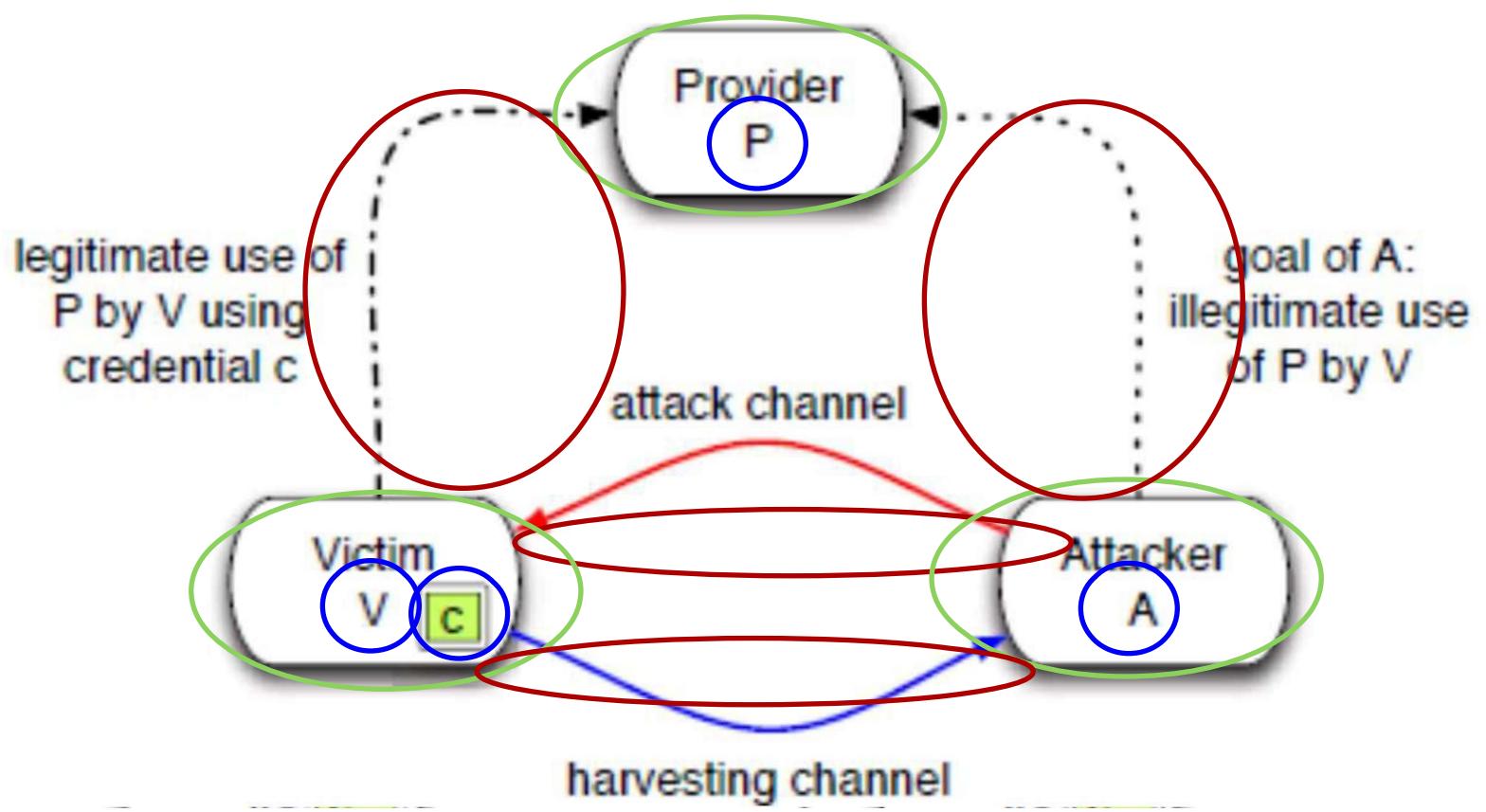
A B
C D

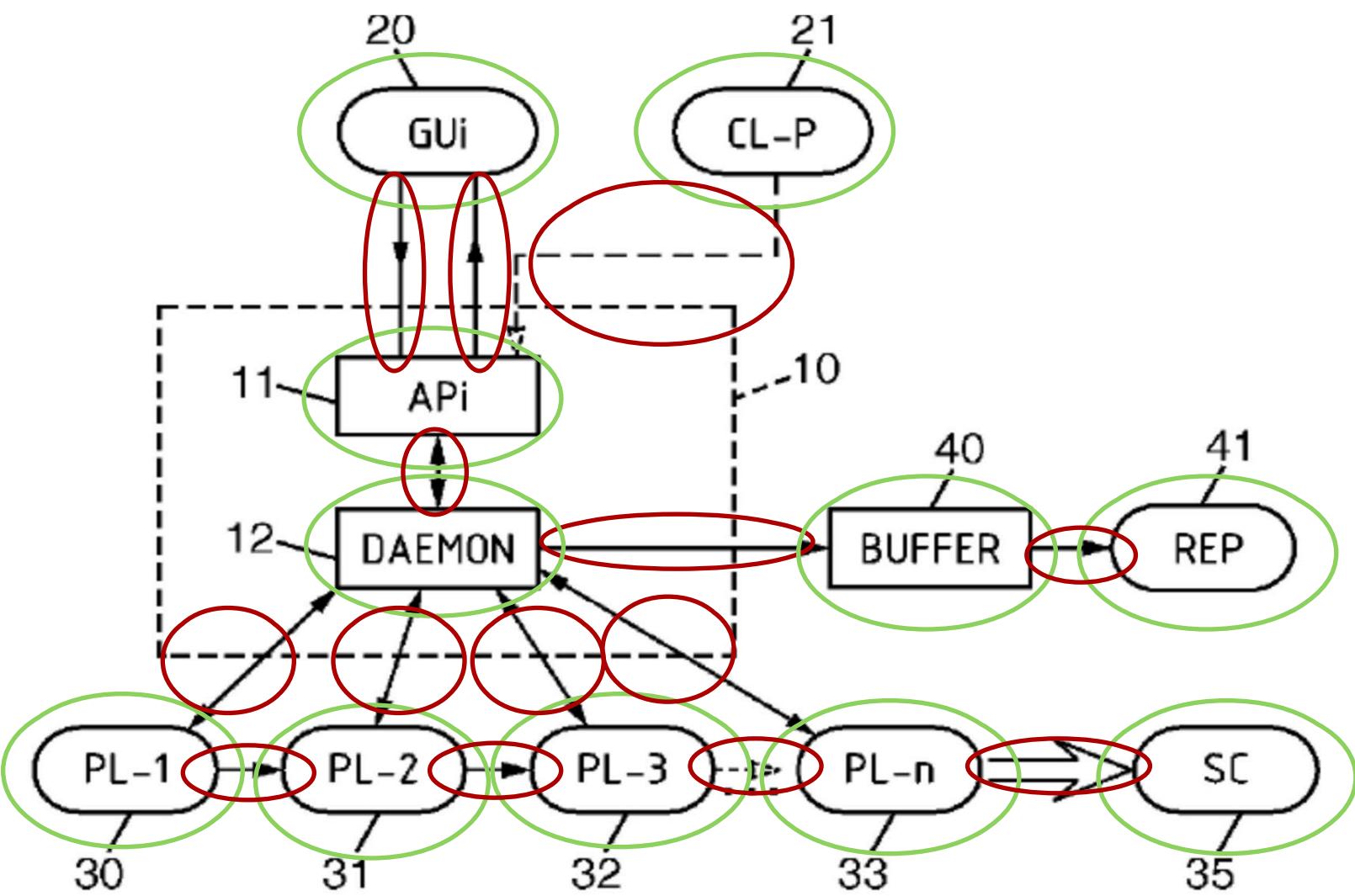




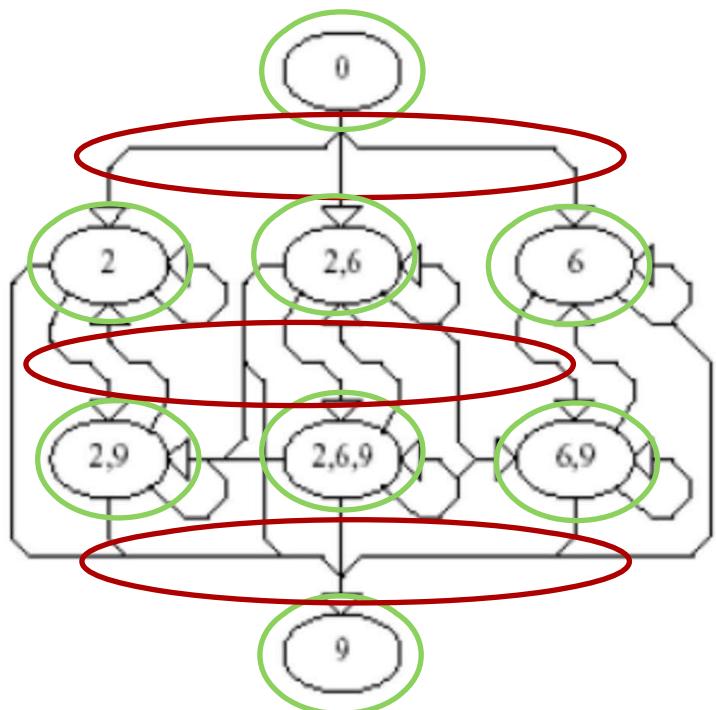


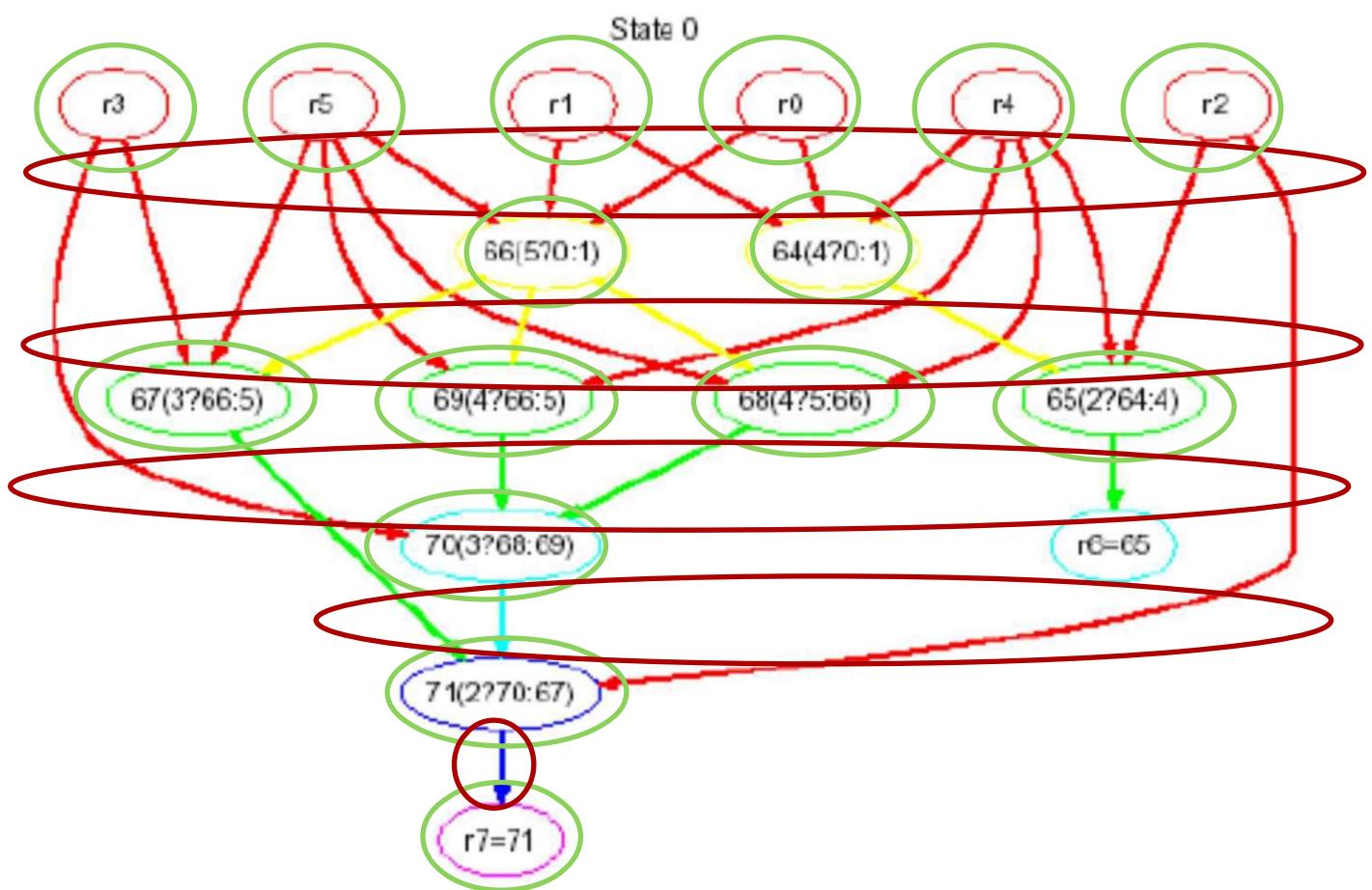






```
if (A) {                                <-- State 0
    do {B} while (C);                   <-- State 2
} else {
    do {D} while (E);                   <-- State 6
}
F                                         <-- State 9
```



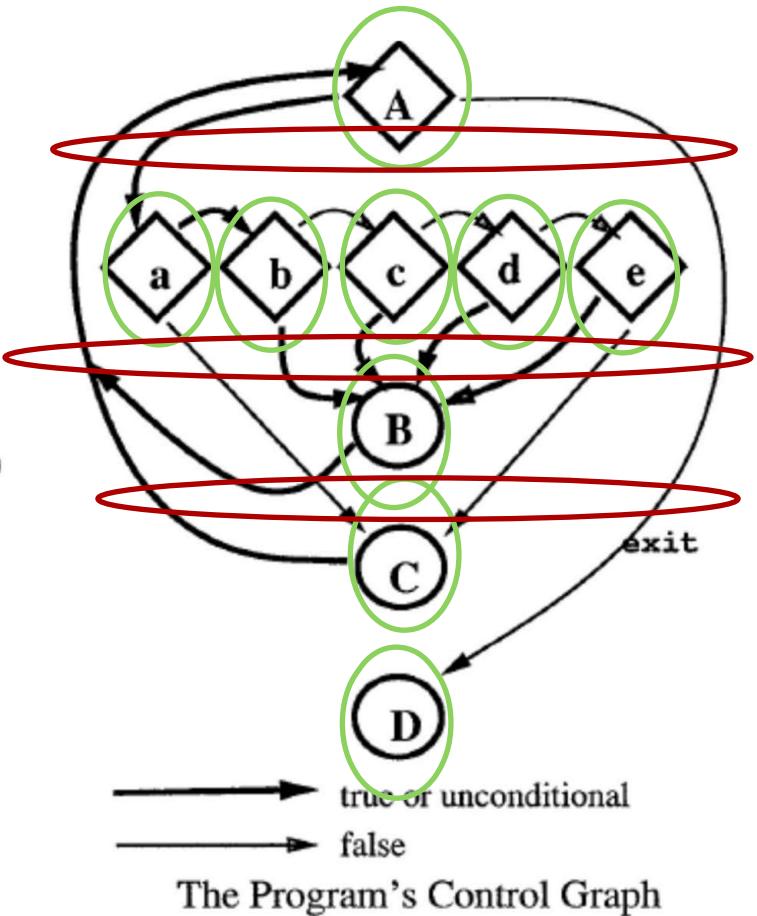


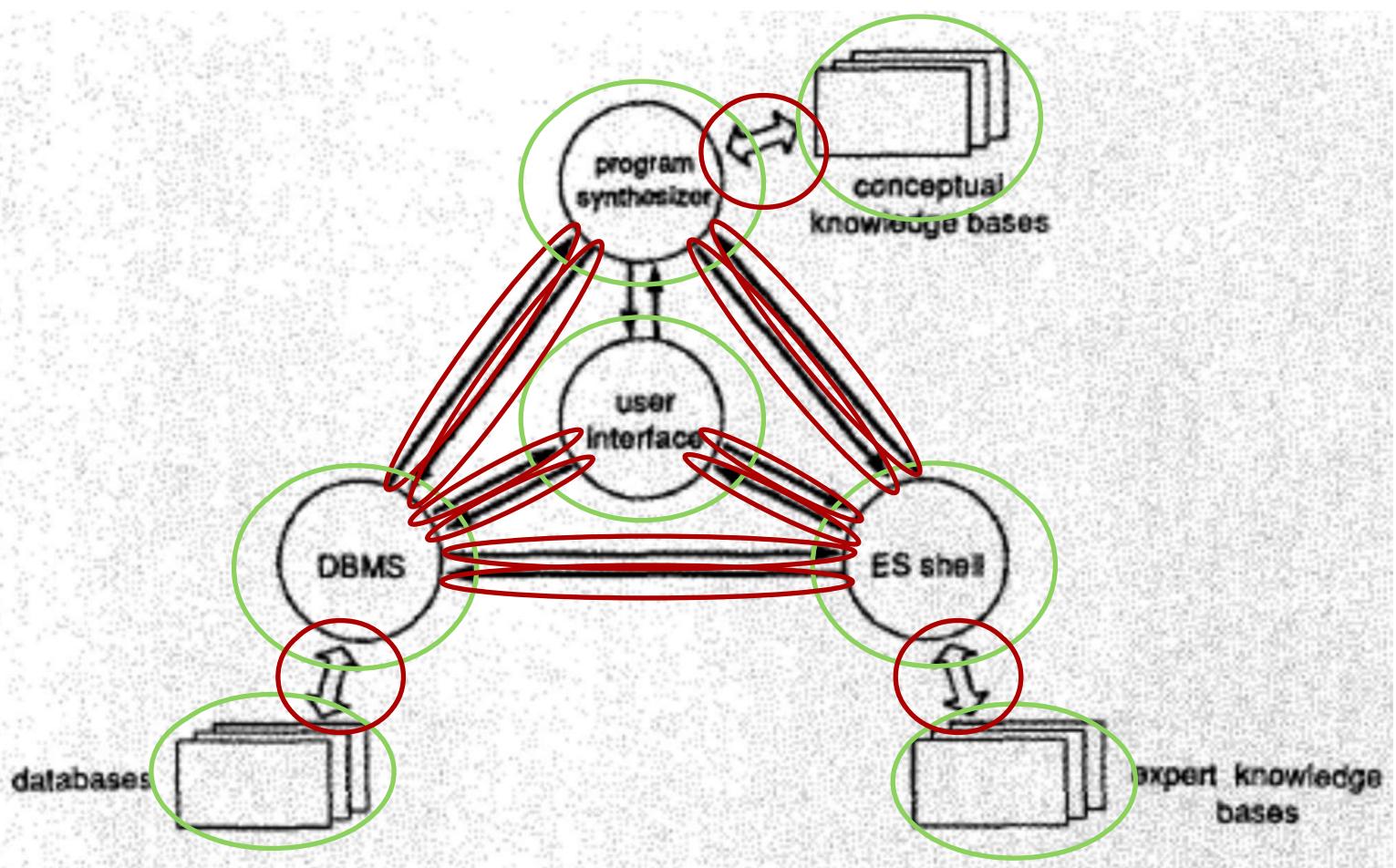
```

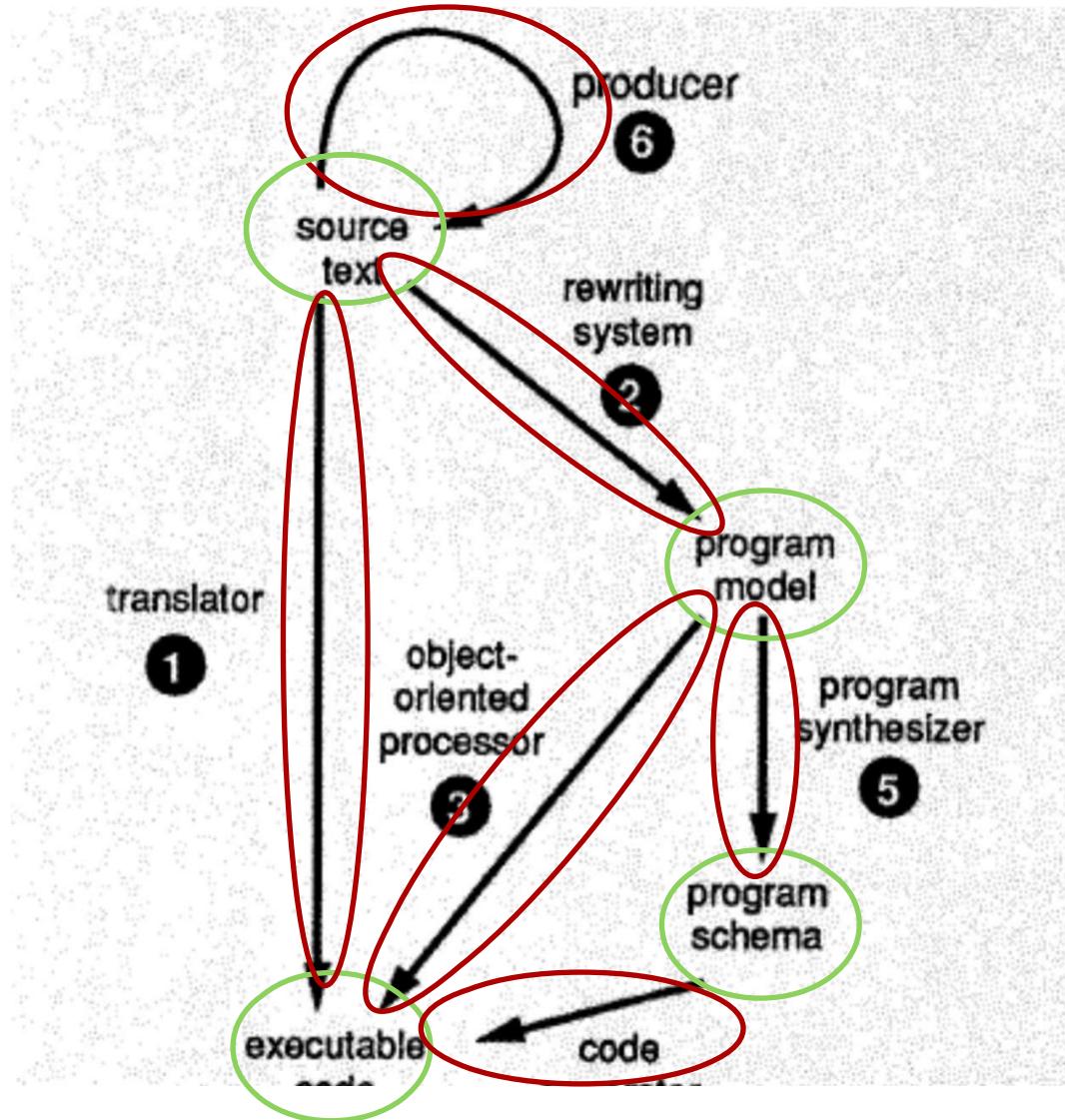
vis(fp)
FILE *fp;
{
    int c;
    while ((c = getc(fp)) != EOF) A
        if (isascii(c) && (isprint(c) ||
            (c=='\n' || c=='\t' || c==' ')) B
            putchar(c);
        else C
    } exit(0); D

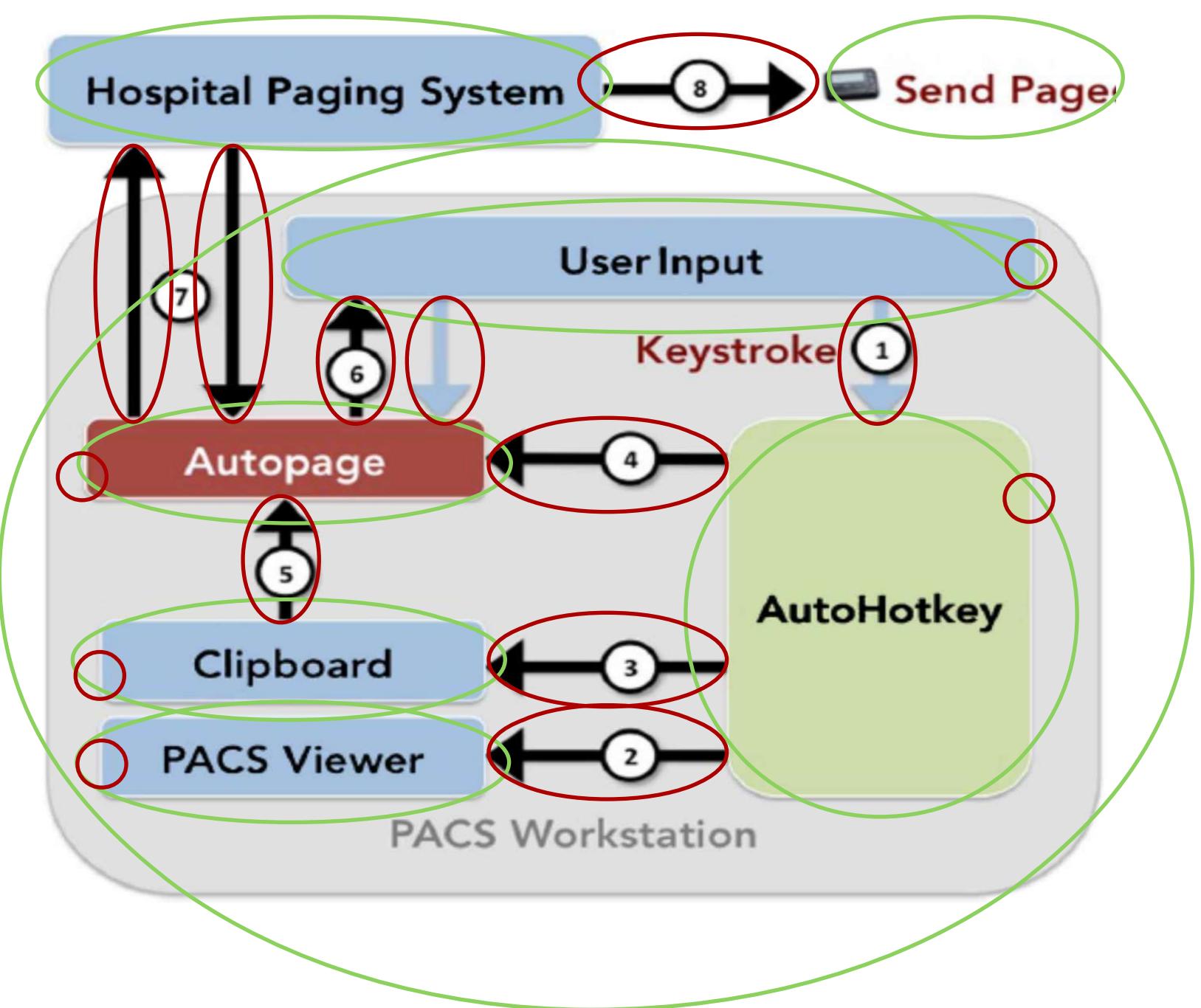
```

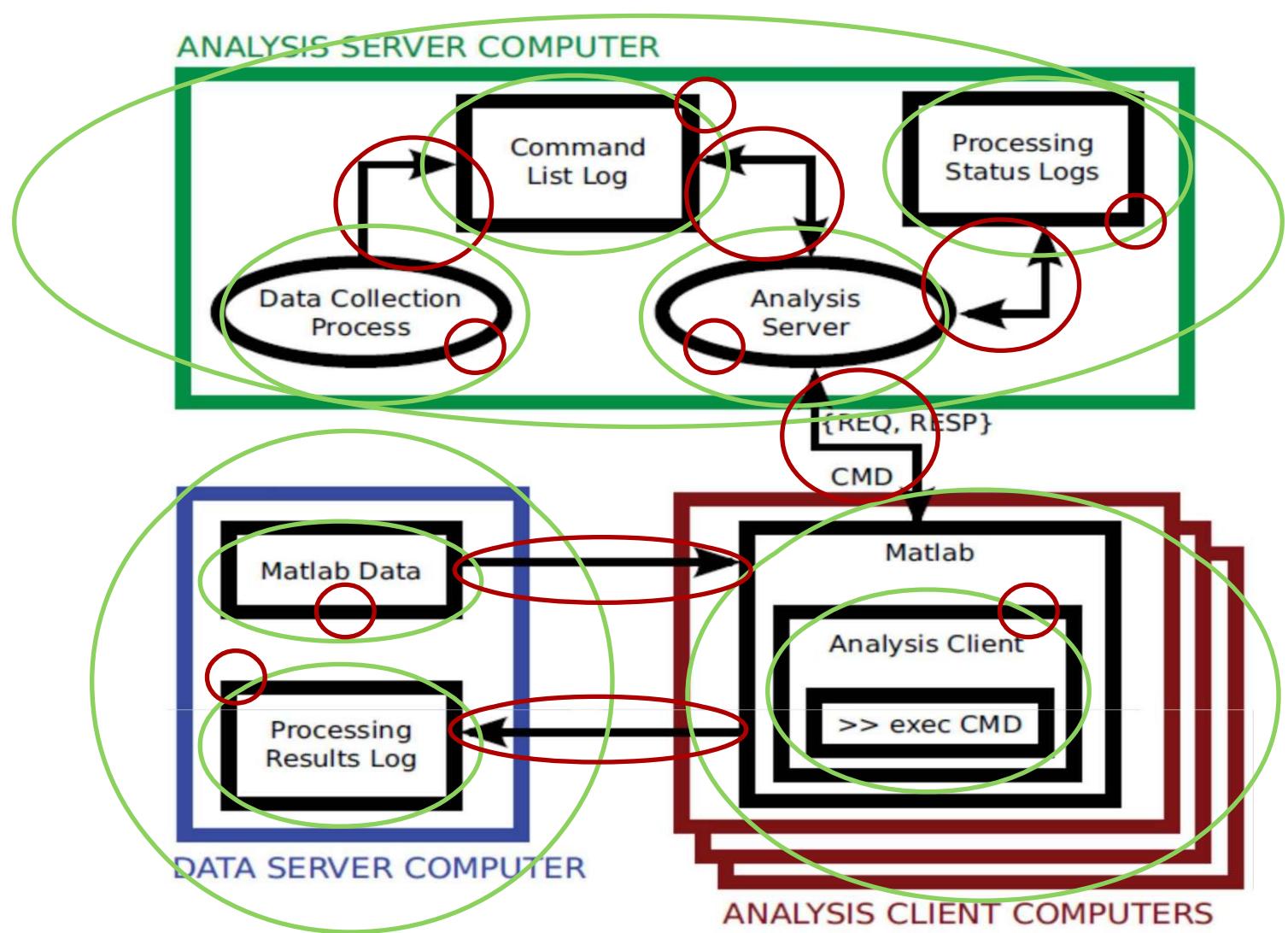
A-D, a-e are basic blocks
A Program in C











Arquitectura

