

MIA (2 6 8 13 14 23 25 30)
EM OK EM OK

NEXT MTG:
 QUIZ LIKELY
 (? TREEWALK?)

READ CH 1-3
 CH 1 WHAT IS SW?

① INTRO QUESTIONS.

1. WHY SO LONG DEV TIME?
 - BUGS

- PROCESS INFO
 = 1M TO 1T BITS
 & USE IT TO CONTROL
 EXECUTION PATHS.

- FACTOR OF
 1000 TO 1B
 MORE COMPLEX
 THE OTHER KINDS
 OF ENGR'G.

HOW TO FIX?

- ① SIMPLIFY EXEC PATHS.
 (AVOID SURPRISE)
- ② REDUCE COMPLEXITY
 OF DATA FLOWING
 BETWEEN "BOXES"
- ③ REDUCE # OF
 DATA PATHS

2. WHY DEV COSTS HIGH?

- HARD TO PREDICT/
 ESTIMATE EFFORT
 DUE TO COMPLEXITY
 (UNFORESEEN BUGS)
- LABOR INTENSIVE

★ MOST EFFORT IN
 FIND/FIX RT BUGS

★ BIG % OF PROJECTS
 FAIL

3. W

Bu

PG

-

(

7

2. WHY DEV COSTS HIGH?

- HARD TO PREDICT/
ESTIMATE EFFORT
DUE TO COMPLEXITY
($\frac{1}{2}$ UNFORESEEN BUGS)
- LABOR INTENSIVE
- ★ MOST EFFORT IN
FIND/FIX RT BUGS

E) LEXITY
VS
ES

★ BIG % OF PROJECTS
FAIL

3. WHY ARE THERE BUGS IN COMPLETED PGMS?

- # OF PATHWAY
COMBINATIONS
TOO BIG TO TEST,
EVER.

4. WHY DO WE MAINTAIN EXISTING (AKA "LEGACY") PGMS FOR SO LONG?

(AKA "CASH COWS")

5. WHY IS IT HARD TO MEASURE DEV PROGRESS?

- WBS HELPS A LOT.

- "90-90" RULE.

↳ (80% OVERRUN)

★ MOST METRICS ARE POOR
& HAVE (HIDDEN)
GIANT "ERROR BAR"

3. WHY ARE THERE
BUGS IN COMPLETED
PGMS?

- # OF PATHWAY
COMBINATIONS
TOO BIG TO TEST,
EVER.

ECTS

4. WHY DO WE MAINTAIN
EXISTING (AKA "LEGACY")
PGMS FOR SO LONG?

(AKA "CASH COWS")

5. WHY IS IT HARD TO
MEASURE DEV PROGRESS?

- WBS HELPS A LOT.

- "90-90" RULE.

↳ (80% OVERRUN)

★ MOST METRICS ARE POOR
& HAVE (HIDDEN)
GIANT "ERROR BAR"

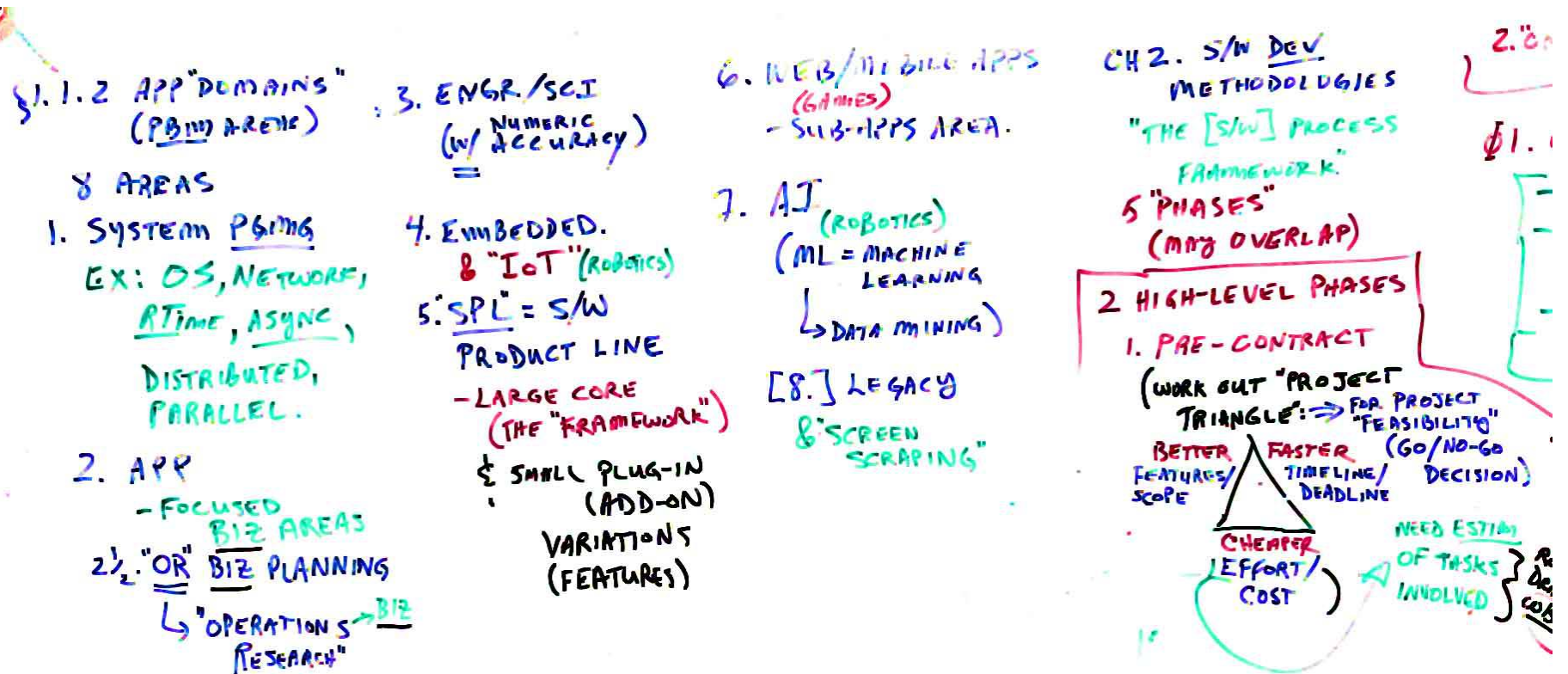
[6.] WHEN IS NEW-DEV
EASY & LOW RISK?

- USE SPECIALTY-AREA
"FRAMEWORK" (OR "LIBRARY")
WHERE YOUR "PBM" IS
REDUCED TO SIMPLE

"BIZ LOGIC".

1. YOU WRITE 10-15%
OF THE TOTAL CODE.
2. FRAMEWORK IS "PROVEN".

3. NEED YOUR PBM TO BE
VERY CLOSE TO SPECIALTY-
AREA.



6. WEB/MOBILE APPS
(GAMES)
- SUB-APPS AREA.

7. AI (ROBOTICS)
(ML = MACHINE LEARNING)
↳ DATA MINING

[8.] LEGACY
& "SCREEN SCRAPING"

CH 2. S/W DEV
METHODOLOGIES
"THE [S/W] PROCESS
FRAMEWORK"

5 "PHASES"
(MAY OVERLAP)

2 HIGH-LEVEL PHASES

1. PRE-CONTRACT

(WORK OUT "PROJECT
TRIANGLE": → FOR PROJECT
"FEASIBILITY"
(GO/NO-GO
DECISION)

BETTER
FEATURES/
SCOPE

FASTER
TIMELINE/
DEADLINE

CHEAPER
EFFORT/
COST

NEED ESTIM
OF TASKS
INVOLVED

"WORK BREAKDOWN
STRUCTURE"
ROUGH
DRAFT
COBS

2. "ON CONTRACT"

Φ 1. COMM/REQTS

- CUST = \$\$\$

- USER = PLAYS
WITH
PGM

- SME =
SUBJECT MATTER
EXPERT

ARCH

Φ 2. PLANNING

- (NBS)

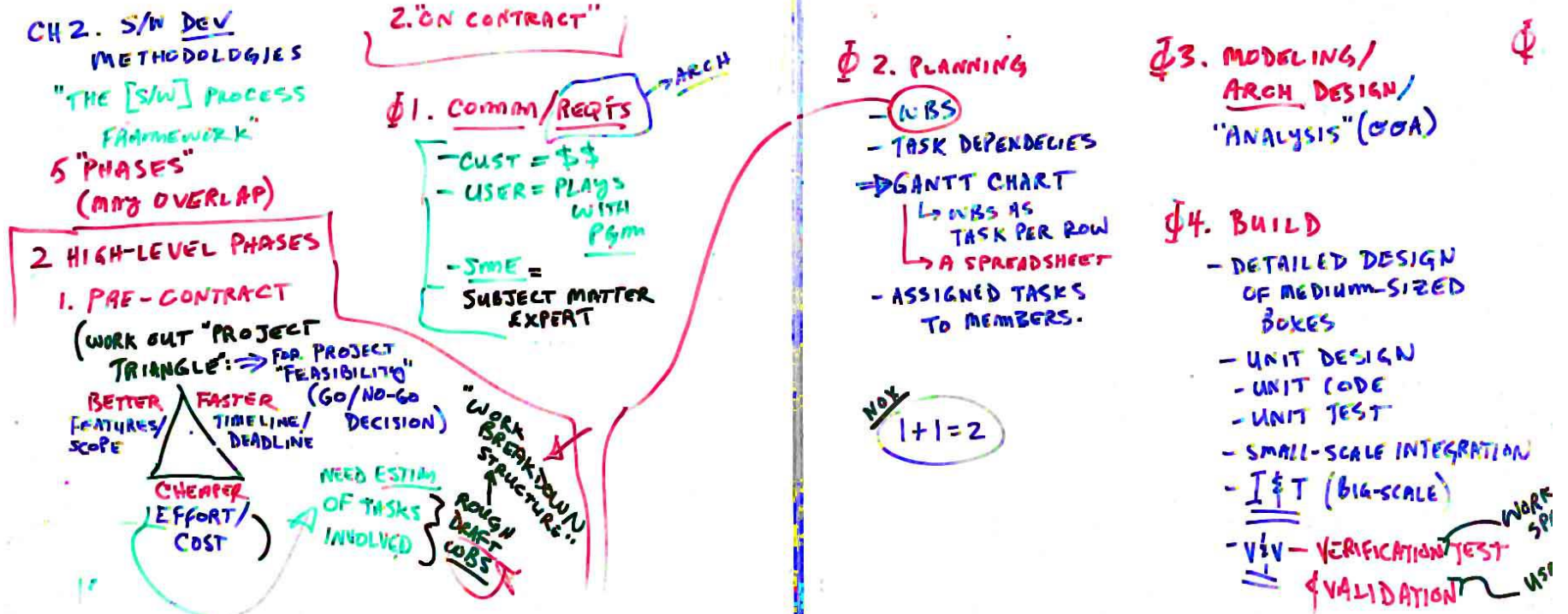
- TASK DEPENDENCIES

→ GANTT CHART

↳ NBS AS
TASK PER ROW
→ A SPREADSHEET

- ASSIGNED TASKS
TO MEMBERS.

NOX
1+1=2



Φ 2. PLANNING

- WBS
- TASK DEPENDENCIES

⇒ GANTT CHART

↳ WBS AS
TASK PER ROW
→ A SPREADSHEET

- ASSIGNED TASKS
TO MEMBERS.

~~NOX~~
 $1 + 1 = 2$

Φ 3. MODELING/ ARCH DESIGN/ "ANALYSIS" (COA)

Φ 4. BUILD

- DETAILED DESIGN
OF MEDIUM-SIZED
BOXES

- UNIT DESIGN
- UNIT CODE
- UNIT TEST
- SMALL-SCALE INTEGRATION
- I & T (BIG-SCALE)

- V & V - VERIFICATION TEST
- VALIDATION

Φ 5. DEPLOY/SHIP

- PACKAGE
- INSTALL TEST
- ON-SITE
"ACCEPTANCE"
TEST
- MANUALS
- TRAINING

WORKS PER
SPEC (REQ'S)
USERS LIKE IT.