Agile Enterprise Data Warehousing
Radical idea or practical concept?

Larissa T. Moss

Method Focus Inc.

methodfocus@earthlink.net

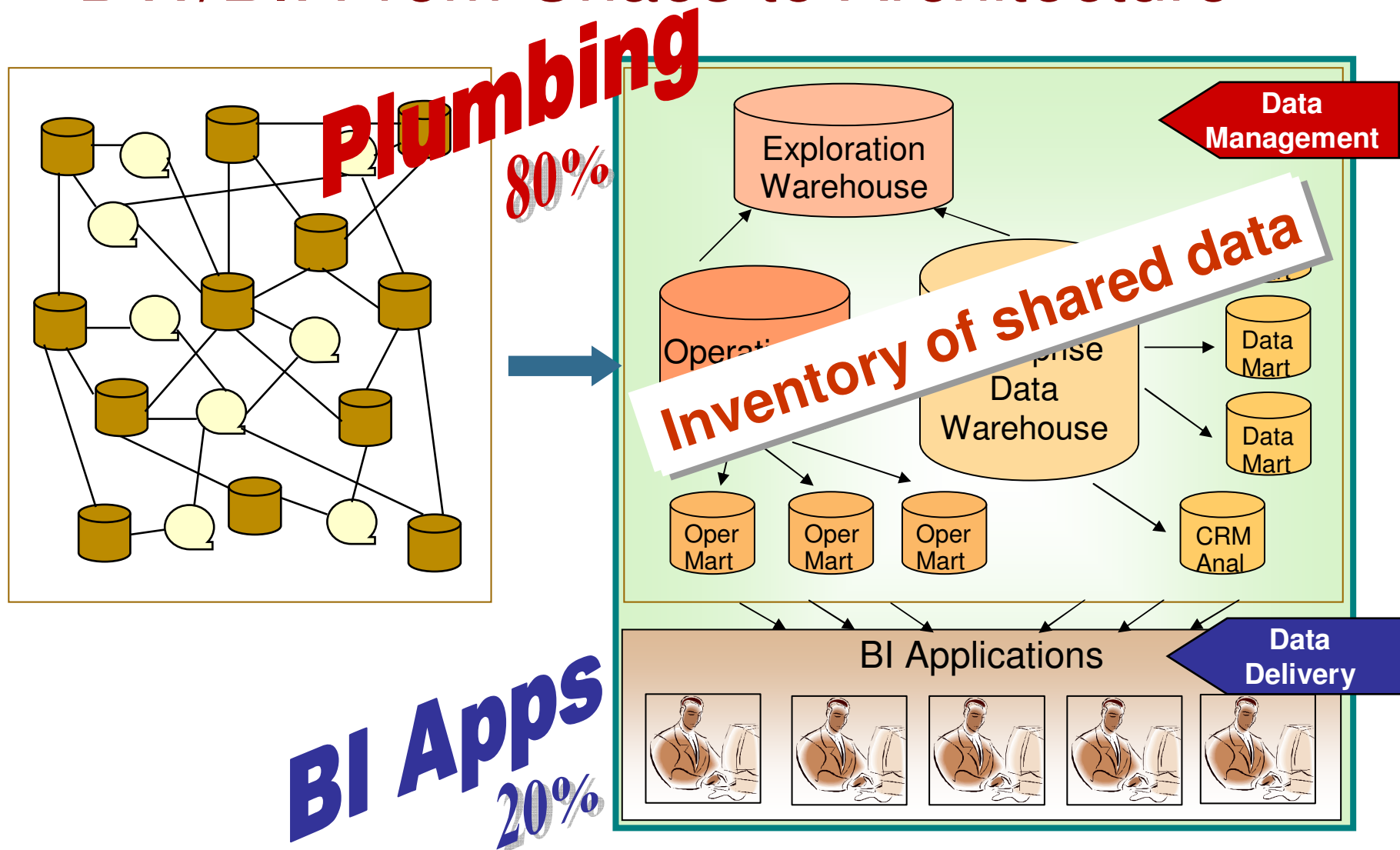TDWI • South Florida Chapter • March 11, 2011

Outline

- Why traditional methodologies don't work on DW/BI projects

- Agile BI versus Agile DW

- What works and what doesn't on DW projects
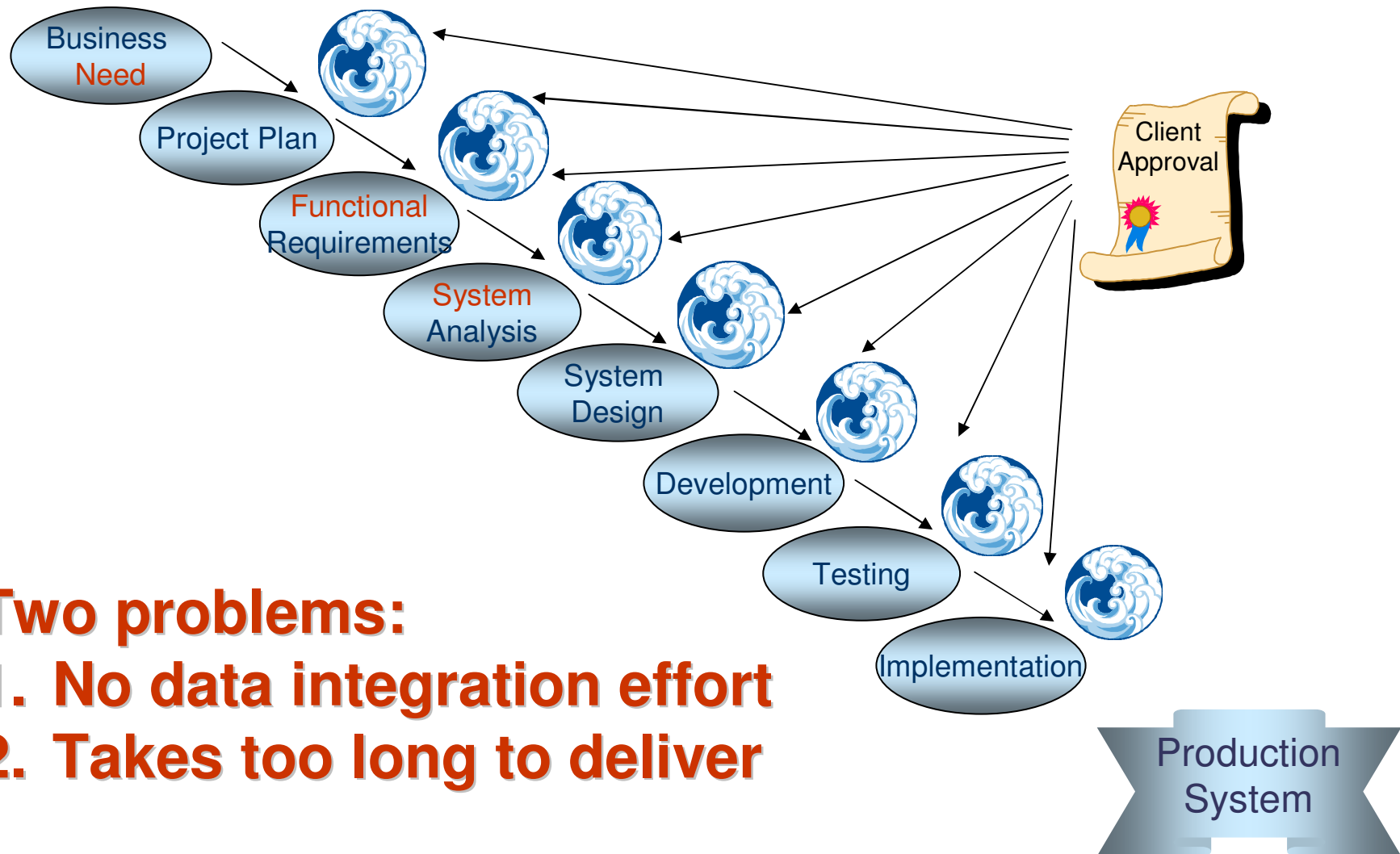
- Extreme Scoping™ Agile Approach to DW/BI

DW/BI: From Chaos to Architecture

Plumbing 80%

Data Management

Exploration Warehouse

Inventory of shared data

Operational ... Enterprise Data Warehouse

Data Mart

Data Mart

Oper Mart

Oper Mart

Oper Mart

CRM Anal

BI Applications

Data Delivery

BI Apps 20%

Waterfall Methodologies

Business Need → Project Plan → Functional Requirements → System Analysis → System Design → Development → Testing → Implementation → Production System

Client Approval

Two problems:
1. **No data integration effort**
2. **Takes too long to deliver**

Spiral Data Integration Methodologies

Solve problem 1
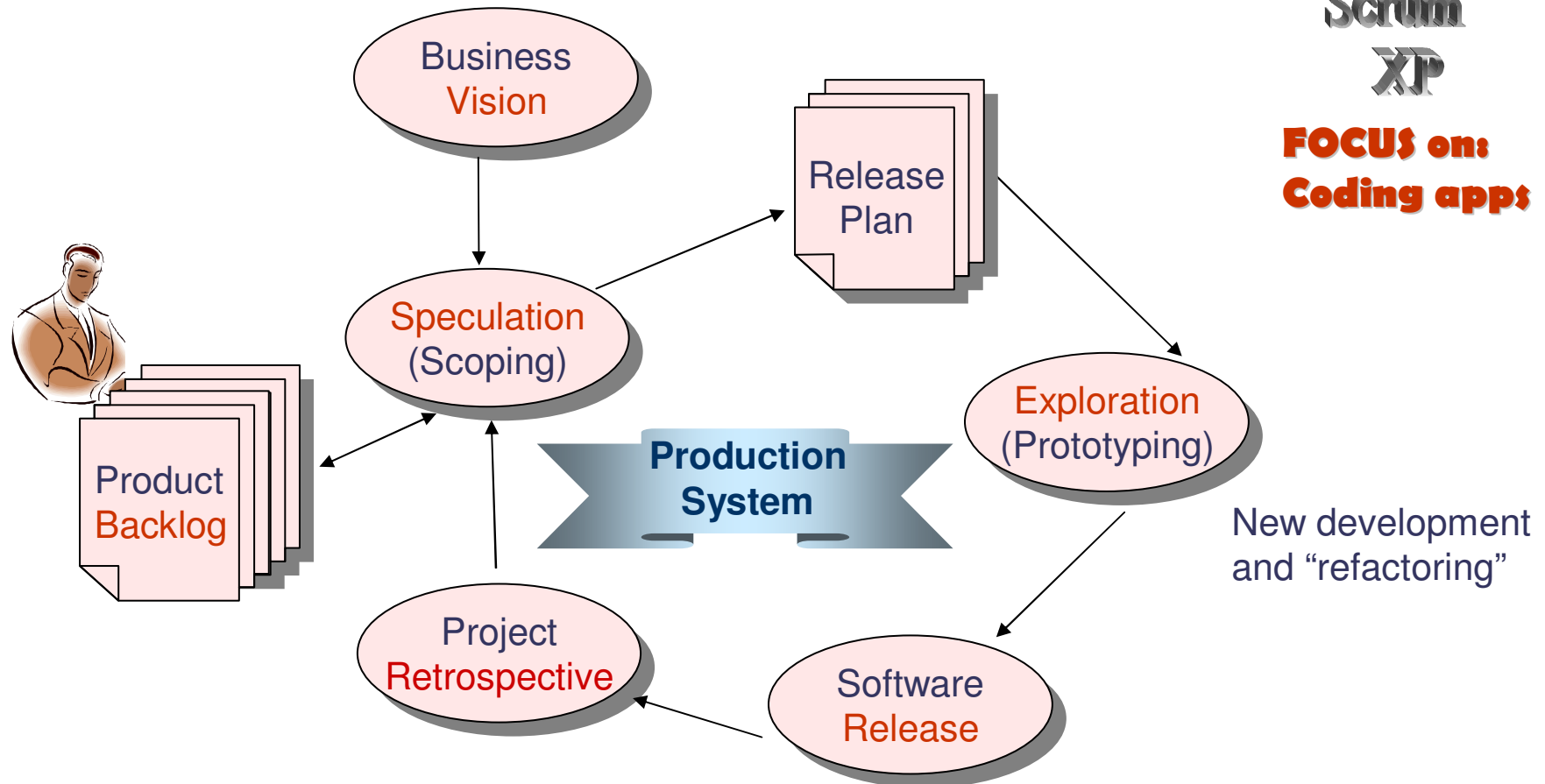

- Business Opportunity
- Assessment & Strategy
- Data Requirements
- Project Planning
- Integration Analysis
- Application Prototyping
- Development
- Testing
- Implementation
- Post-Impl. Review

Evolving **Data Warehouse**

80% **Data Management**

20% **Data Delivery**

**Iterative Development =
One application at a time**

Agile Development Methodologies

Solve problem 2

Scrum
XP

**FOCUS on:
Coding apps**

- Business Vision
- Release Plan
- Speculation (Scoping)
- Product Backlog
- Exploration (Prototyping)
- Production System
- Project Retrospective
- Software Release

New development and "refactoring"

One Application = Multiple Releases

Can Agile be used for BI?

Solve problem 2

- Depends on what you call "BI"
- Depends on what you call "Agile"

Scrum ☑
XP ☑

FOCUS on: Coding apps

Silos

Data Delivery

20%
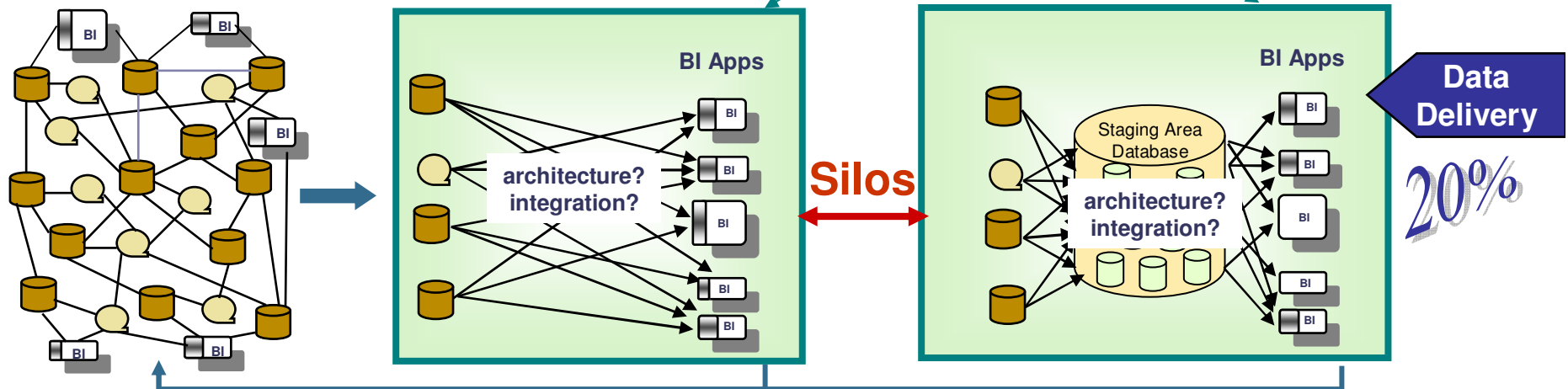
- Independent BI applications
- Directly sourced from operational databases
- Developed by the same or by different BI groups

- Independent BI applications
- Sourced through a central staging area database containing copies of operational databases
- Developed by the same or by different BI groups

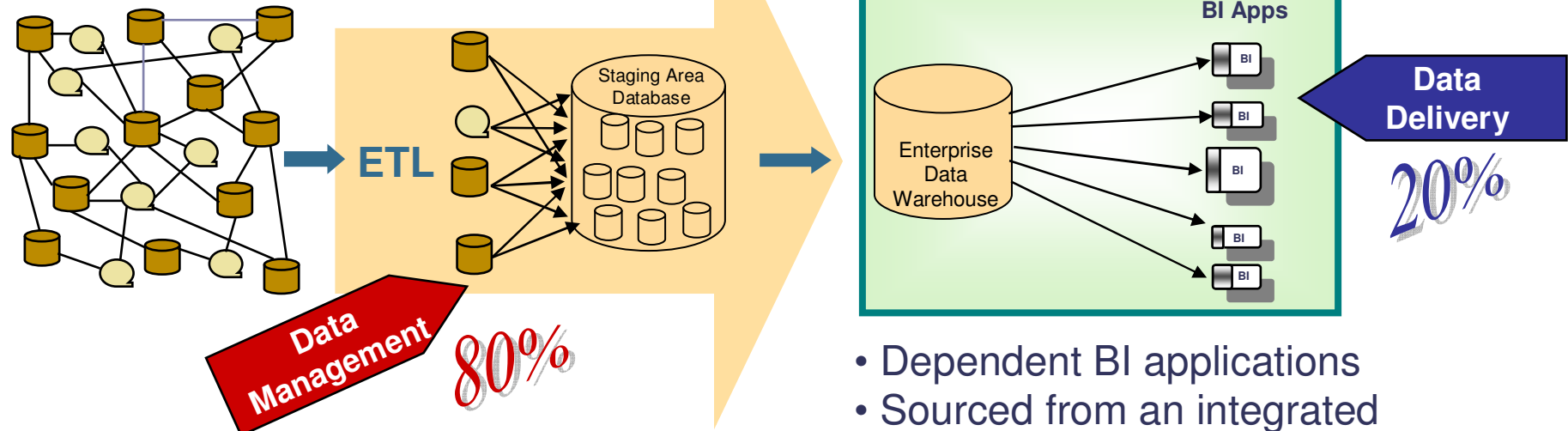**BI without DW**

7

Can Agile be used for BI?

- Depends on what you call "BI"
- Depends on what you call "Agile"

Solve problem 2

Scrum ☑
XP ☑

FOCUS on:
Coding apps

Data Management **80%**

Data Delivery **20%**

Solve problem 1

BI with separate DW

- Dependent BI applications
- Sourced from an integrated and standardized DW
- Developed by the same or by different BI groups
- But, building/enlarging the DW is a different project and a different team
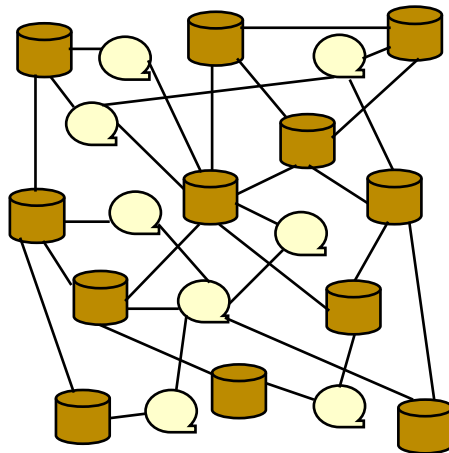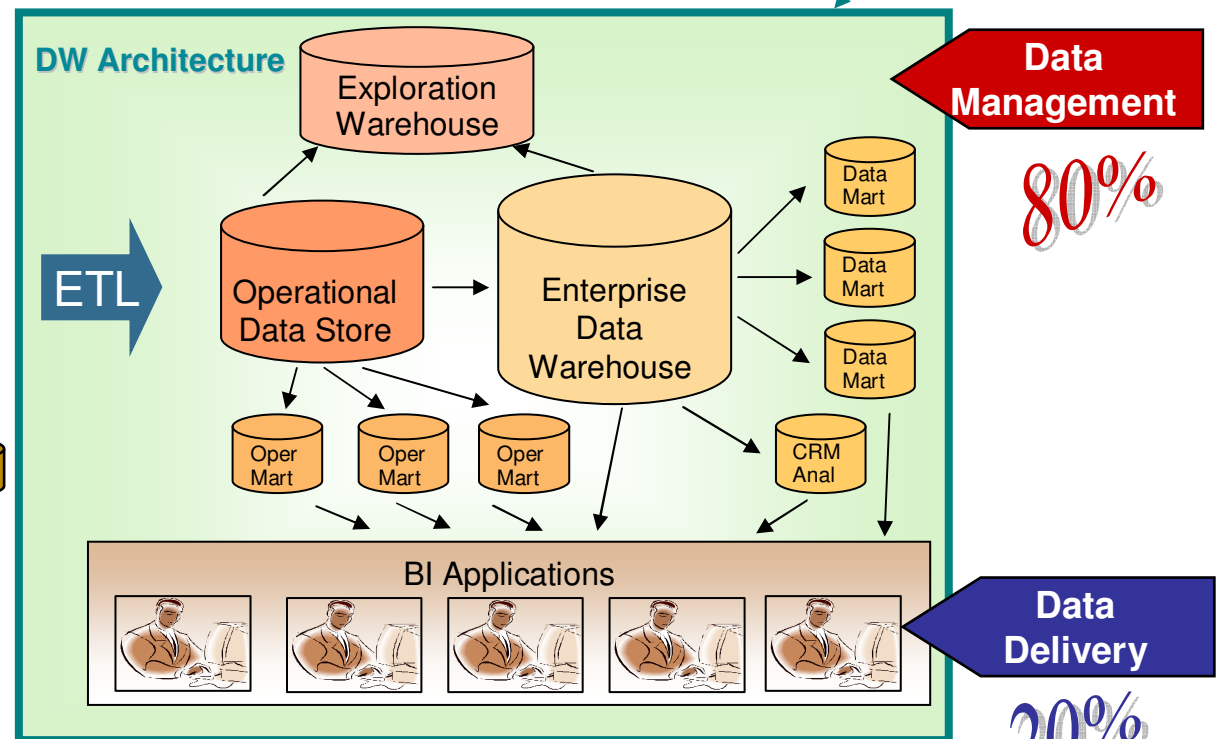
8

Can Agile be used for BI?

- Depends on what you call "BI"
- Depends on what you call "Agile"

Solve problems 1&2

Scrum ☒
XP ☒

DW Architecture

Exploration Warehouse

ETL → Operational Data Store → Enterprise Data Warehouse → Data Mart / Data Mart / Data Mart

Oper Mart / Oper Mart / Oper Mart

CRM Anal

BI Applications

Data Management

80%

Data Delivery

20%

BI includes DW

- **Dependent BI applications**
- **Sourced from an integrated and standardized DW or from dependent data marts**
- **Developed by the same group**
- **Building or enlarging the DW is the same project and the same team**

Can Agile be used for BI?

- Depends on what you call "BI"
- Depends on what you call "Agile"

Solve problems 1&2

Extreme Scoping™ ☑

DW Architecture

Exploration Warehouse

ETL

Data Standardization
Common Data Definitions
Fully Qualified Data Names
Standardized Business Rules
Enterprise Data Modeling
Source Data Analysis
Data Cleansing Rules

FOCUS on: Enterprise data standardization and integration

Not: Coding apps

- **Dependent BI applications**
- **Sourced from an integrated and standardized DW or from dependent data marts**
- **Developed by the same group**
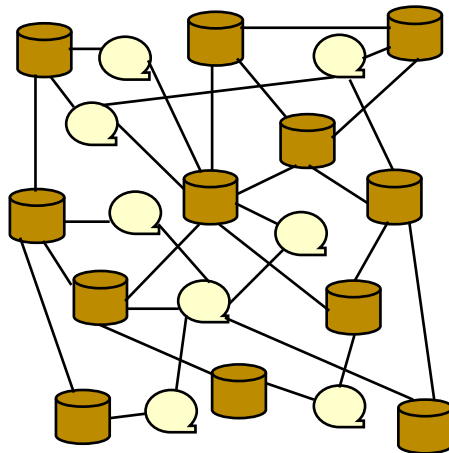- **Building or enlarging the DW is the _same_ project and the _same_ team**
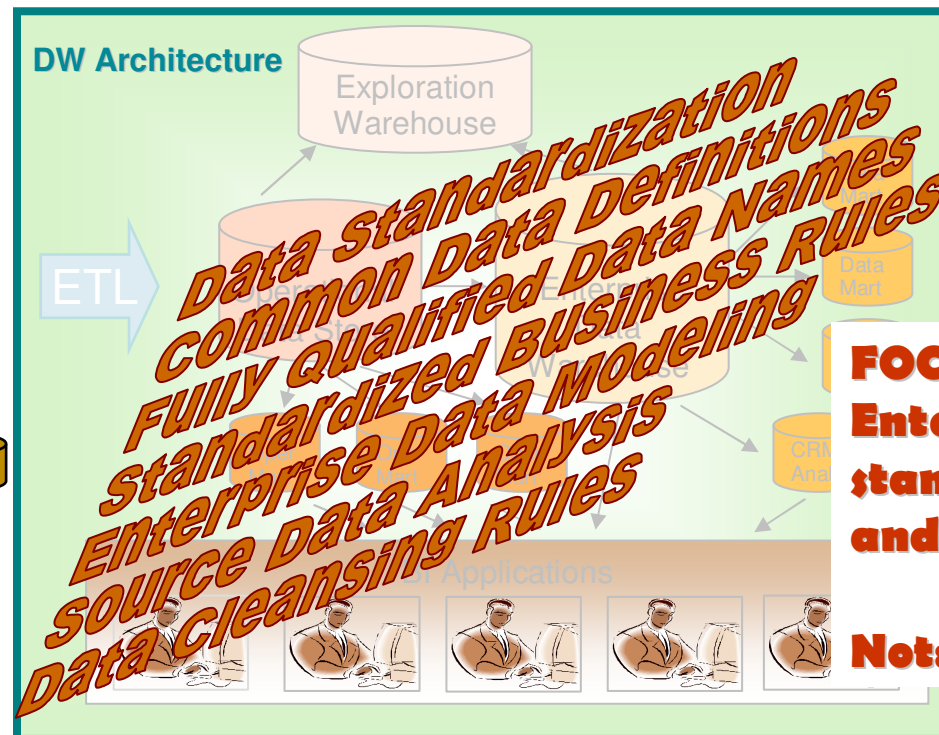
BI includes DW

No "Silo" and No "Big-Bang" Development

Solve problems 1 & 2

Merge Spiral with Agile!
= Extreme Scoping™

- Business Vision
- Business Opportunity
- Data Requirements
- Release Plan
- Project Planning
- Speculation (Scoping)
- Exploration (Prototyping)
- Evolving Data Warehouse
- Requirements
- Po...
- Implementation
- Project Retrospective
- Software Release
- Application Prototyping
- New development and "refactoring"
- Testing
- Development

Agile principles that _work_ for DW projects

- Business vision instead of final requirements

- Speculation instead of estimating

- Exploration (_prototyping_) instead of development

- Self-organizing project team (no interference)

- Daily stand-up meetings (course corrections)

- Get physical as quickly as possible (deliver partial functionality)

- Time-boxed increments

- (_Data_) Quality before quantity

- Refactoring (refinement = system _evolution_)

4 E's:
Experimental
Experiential
Educational
Evolutionary

Agile principles that _don't work_ for DW projects

- Freezing the scope of an increment
- Scrum Master (project manager)
- Product owner (single user)
- Product backlog managed by product owner
- Effort estimate based on code complexity
- Cadence = development rhythm (every 10 or 29 days)
- Burn-down charts

Step1: Speculation and Scouting
(Trademark of Larissa T. Moss, Method Focus Inc.)

1. Speculate on the total effort for the new DW/BI request

"Speculation"

Development Steps

1. Business Case Assessment .... Enterprise-view
2.A Technical Infrastructure Evaluation .... Enterprise-view
2.B Non-Technical Infrastructure Evaluation ... Enterprise-view
3. Project Planning .... *Project-specific*
4. Requirements Definition .... Enterprise-view
5. Data Analysis .... Enterprise-view
6. Application Prototyping .... *Project-specific*
7. Meta Data Repository Analysis .... Enterprise-view
8. Database Design .... Enterprise-view
9. ETL Design .... Enterprise-view
10. Meta Data Repository Design .... Enterprise-view
11. ETL Development .... Enterprise-view
12. Application Development .... *Project-specific*
13. Data Mining .... Enterprise-view
14. Meta Data Repository Development .... Enterprise-view
15. Implementation .... *Project-specific*
16. Release Evaluation .... Enterprise-view

From ***Business Intelligence Roadmap***, Moss Larissa T. and Shaku Atre)

describes

Selected things to consider
- activities
- dependencies
- deliverables
- roles

exposes

Unknown conditions (Questions)
- data
- architecture
- inter-dependencies
- resources

time boxed

"Scouting"

14

Speculation and Scouting Example

Development Steps

1. ~~Business Case Assessment~~ ....
2.A ~~*Technical Infrastructure Evaluation*~~ ....
2.B Non-Technical Infrastructure Evaluation ... 5 days
3. Project Planning .... **?**
4. Requirements Definition .... 15 days
5. Data Analysis .... **?**
6. *Application Prototyping* .... 10 day
7. Meta Data Repository Analysis .... 1 day
8. Database Design ....22 days
9. ETL Design .... 15 days
10. Meta Data Repository Design .... 6 days
11. ETL Development ....45 days**?**
12. Application Development .... 10 days
13. ~~Data Mining~~ ....
14. Meta Data Repository Development .... 10 days
15. *Implementation* .... 8 days
16. Release Evaluation .... 1 day

"Speculation"

Selected things to consider
- activities
- dependencies
- deliverables
- roles

~ 148 days + ?

- Identify data sources
- Identify data owners
- Profile data sources
- Create dirty data report
- Write cleansing specs

Unknown conditions (Questions)
- data
- architecture
- inter-dependencies
- resources

time boxed

~ 10 days

"Scouting"

Speculation and Scouting Example

Development Steps

1. ~~Business Case Assessment~~ ....
2.A ~~Technical Infrastructure Evaluation~~ ....
2.B Non-Technical Infrastructure Evaluation ... 5 days
3. Project
4. Require
5. Data Analysis ....
6. *Application Prototyping* .... 10 day
7. Meta Data Repository Analysis .... 1 day
8. Database Design ....22 days
9. ETL Design .... 15 days
10. Meta Data Repository Design .... 6 days
11. ETL Development .... 60 45 days **?**
12. Application Development .... 10 days
13. ~~Data Mining~~
14. Meta Data Repository Development .... 10 days
15. *Implementation* .... 8 days
16. Release Evaluation .... 1 day

~ 60 elapsed weeks days

"Speculation"

Selected things to consider
- activities
- dependencies
- deliverables
- roles

~ 148 days + 17
163

- Identify data sources 1 day
- Identify data owners   1 day
- Profile data sources   8 days
- Create dirty data report 2 days
- Write cleansing specs 5 days

Unknown conditions (Questions)
- data
- architecture
- inter-dependencies
- resources

time boxed
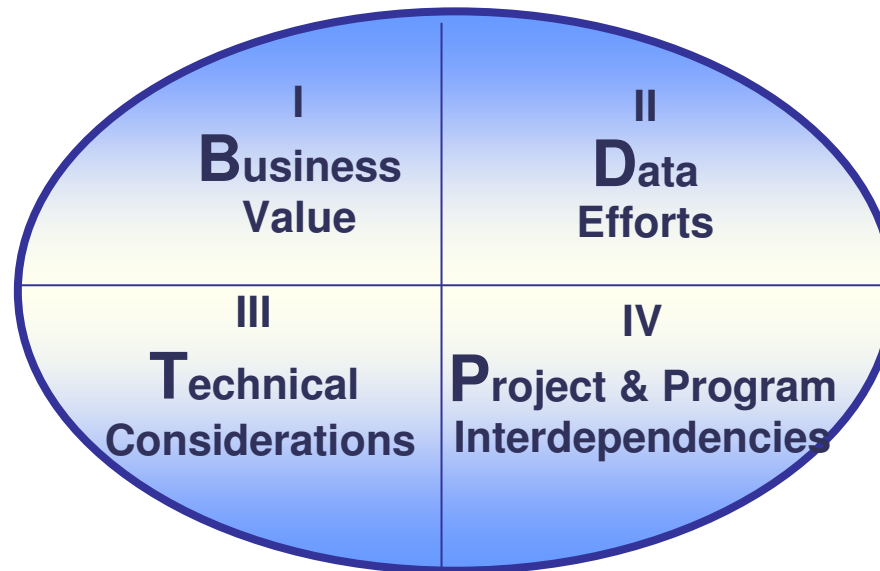
~ 10 days

"Scouting"

Step 2: Extreme Scoping™ Applied
(Trademark of Larissa T. Moss, Method Focus Inc.)

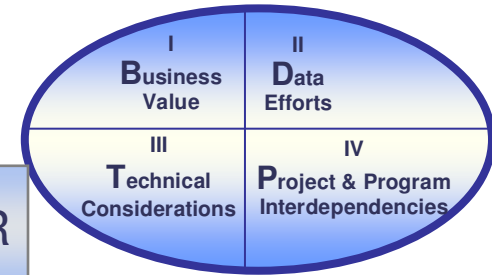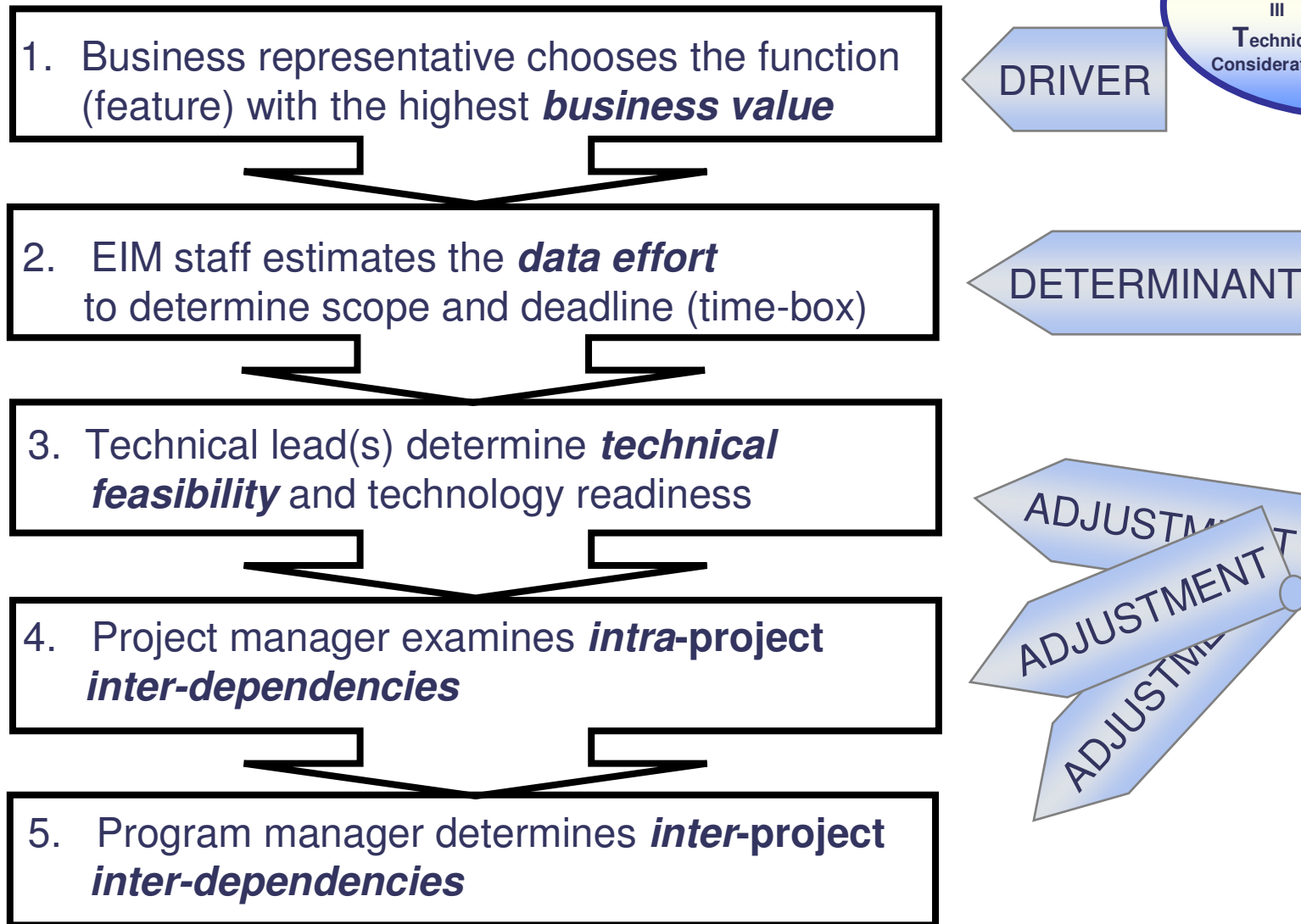> 1. Speculate on the total effort for the new DW/BI request

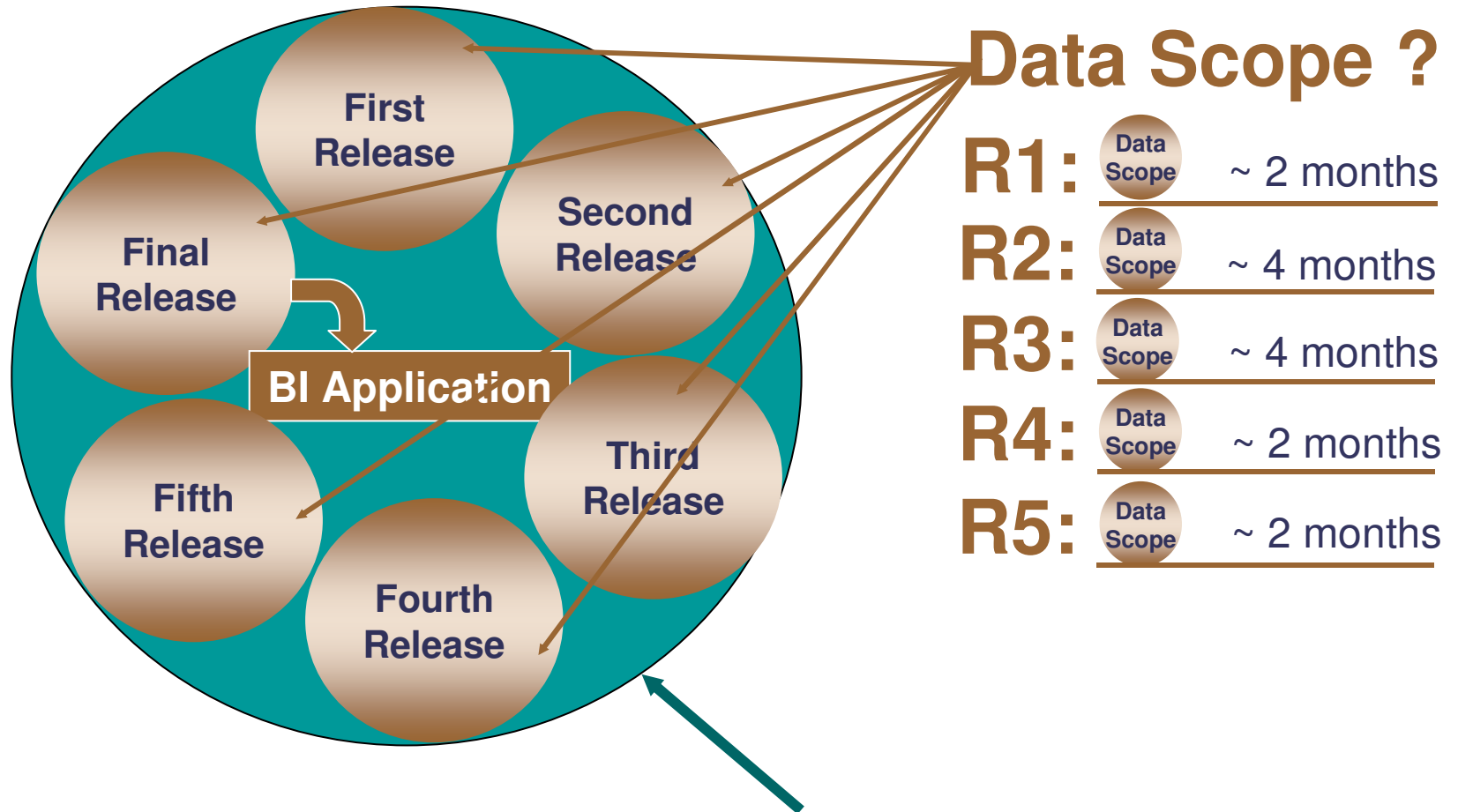> 2. Break DW/BI application request into software releases

BDTP™ Balance!

I **B**usiness Value	II **D**ata Efforts
III **T**echnical Considerations	IV **P**roject & Program Interdependencies

BDTP Balance™

I Business Value	II Data Efforts
III Technical Considerations	IV Project & Program Interdependencies

1. Business representative chooses the function (feature) with the highest **business value**

DRIVER

2. EIM staff estimates the **data effort** to determine scope and deadline (time-box)

DETERMINANT

3. Technical lead(s) determine **technical feasibility** and technology readiness

ADJUSTMENT

4. Project manager examines **intra**-project inter-dependencies

ADJUSTMENT

5. Program manager determines **inter**-project inter-dependencies

ADJUSTMENT

How Many Software Releases?

Data Scope ?

R1: Data Scope — ~ 2 months

R2: Data Scope — ~ 4 months

R3: Data Scope — ~ 4 months

R4: Data Scope — ~ 2 months

R5: Data Scope — ~ 2 months

First Release

Second Release

Final Release

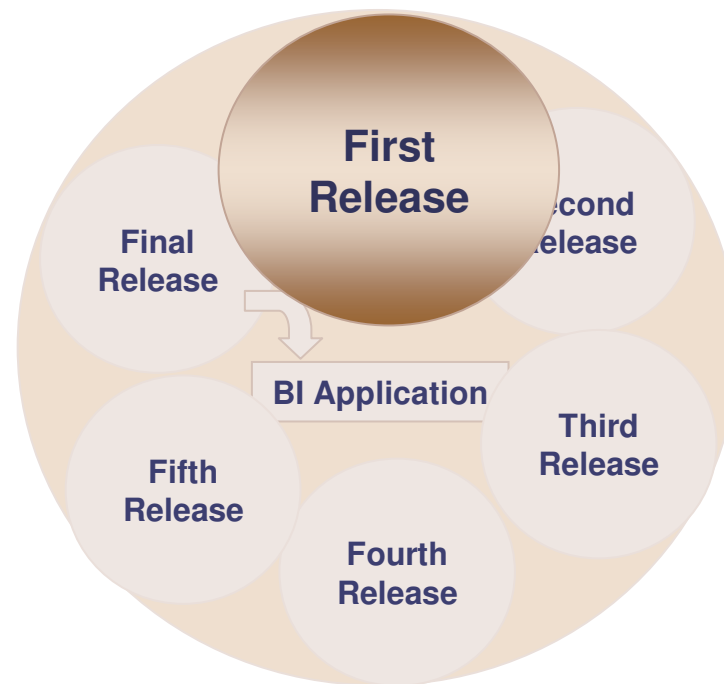BI Application

Fifth Release

Third Release

Fourth Release

180 effort days = 60 elapsed weeks (~14 months)

Step 3: Planning the First Release

(Trademark of Larissa T. Moss, Method Focus Inc.)

1. Speculate on the total effort for the new DW/BI request

2. Break DW/BI application request into software releases

3. Create work breakdown structure for first software release

Select Appropriate Tasks or Subtasks

592	Write query script ...	587
593	~~Write access interface programming specifications~~	~~588~~
594	~~Write online help function programming specifications~~	~~589~~
595	**Step 11: Extract/Transform/Load Development**	
596	**Build and unit test the ETL process**	**542, 549**
597	~~Code the ETL programs~~	
598	If using an ETL tool, write instructions for ETL tool modules	
599	~~Capture the ETL technical metadata for the metadata repository~~	
600	Write code to produce reconciliation totals, quality metrics, and load statistics	
601	Unit test each individual program module	597, 600
602	If using an ETL tool, unit test each ETL tool module	598, 600
603	Write the scripts to execute the ETL programs and the sort, merge, and load utilities	601, 602
604	**Integration or regression test the ETL process**	**596**
605	Create a test plan with test cases for the ETL process	
606	Create test data for the ETL programs	
607	Integration or regression test the entire ETL process	606, 606
608	~~Log the actual test results and document any test issues~~	

from Business Intelligence Roadmap
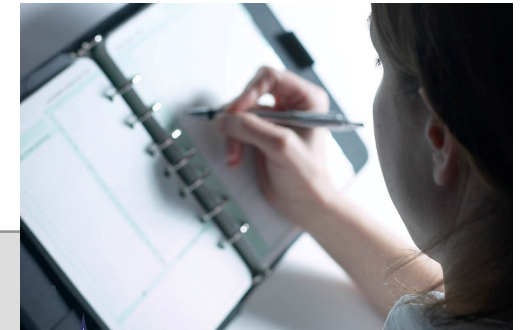*(Moss & Atre)*

Step 4: Weekly milestones
(Trademark of Larissa T. Moss, Method Focus Inc.)

1. Speculate on the total effort for the new DW/BI request

2. Break DW/BI application request into software releases

3. Create work breakdown structure for first software release

4. Create milestones from DDD to YAH for first release

Describing the milestones

Week 1: - Potential source files identified and profiled
         - Data cleansing specifications identified and reviewed with users
         - ETL architectural design changes identified
         - 8 report designs prototyped and demonstrated to users

Week 2: - New data modeled and data disputes among users documented
         - New table created and 3 existing databases modified
         - Undisputed data mapped from source to target
         - One new extract program coded and 5 ETL programs modified

Week 3: - All data disputes among users resolved
         - Data cleansing specifications modified
         - All ETL programs unit tested
         - 5 new reports coded based on approved designs using new table
         - 3 old reports from existing databases modified

Week ddd-n: - …

Week ddd-2: - UAT completed and code frozen
         - Operations signed off
         - Job scheduler created and tested
         - Production environment created
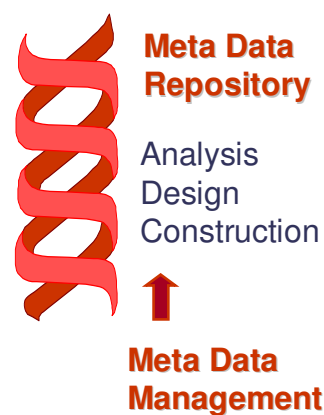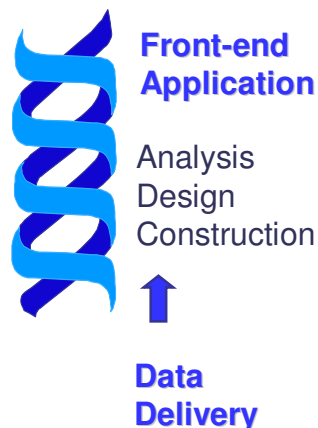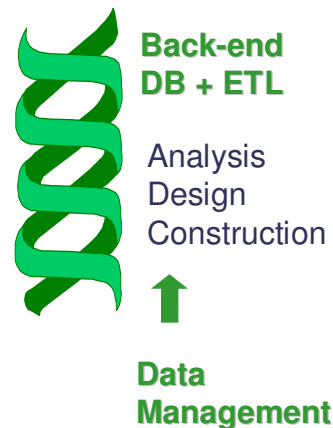
Week ddd-1: - All programs migrated into production environment
         - Load files ready to run in production
         - Job scheduler modified, tested, and signed off
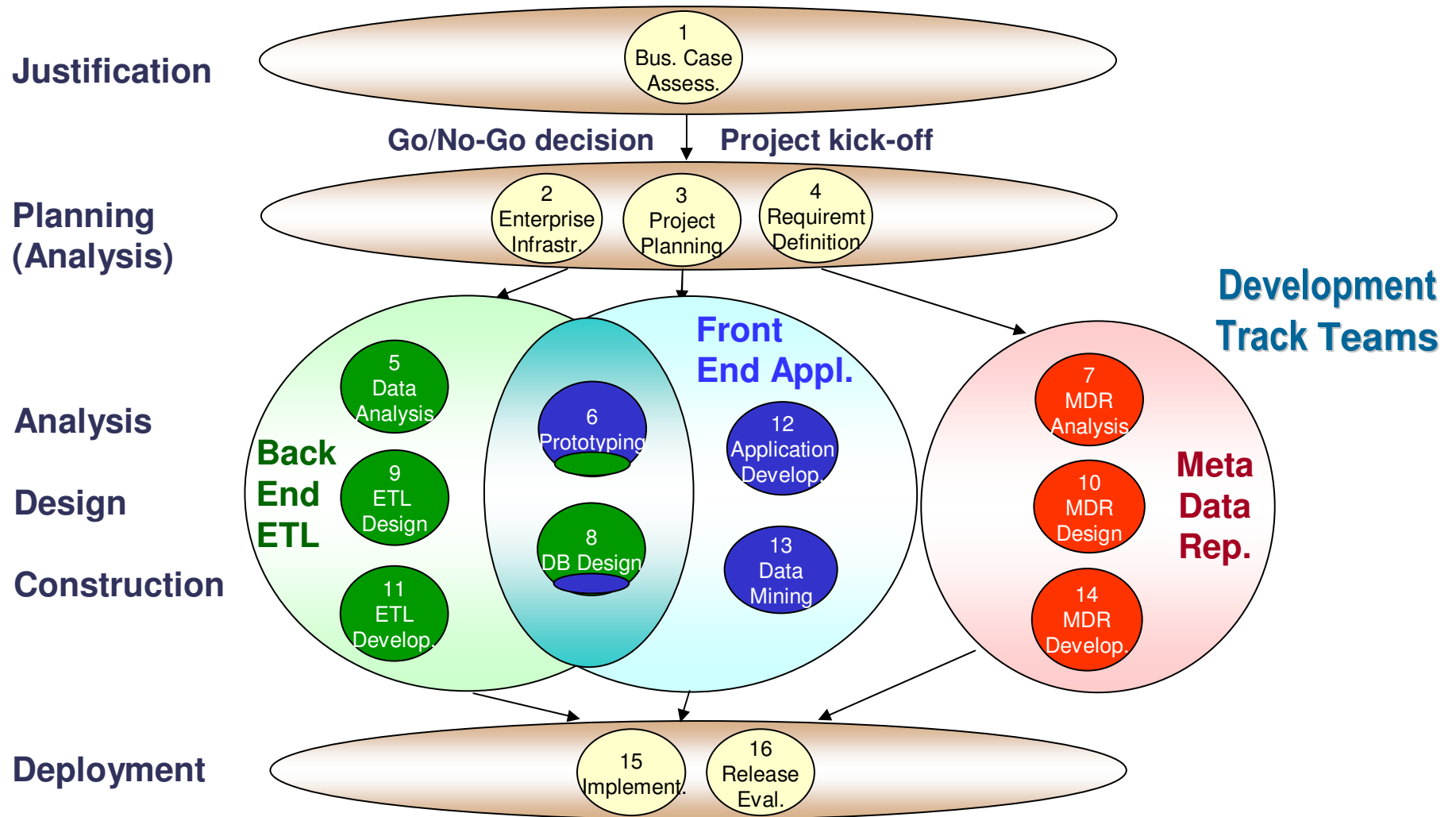
Step 5: Parallel Development Tracks

1. Speculate on the total effort for the new DW/BI request

2. Break DW/BI application request into software releases

3. Create work breakdown structure for first software release

4. Create milestones from DDD to YAH for first release

5. Organize and assign parallel development tracks

Back-end DB + ETL

Analysis
Design
Construction

Data Management

Front-end Application

Analysis
Design
Construction

Data Delivery

Meta Data Repository

Analysis
Design
Construction

Meta Data Management

Development Track Teams

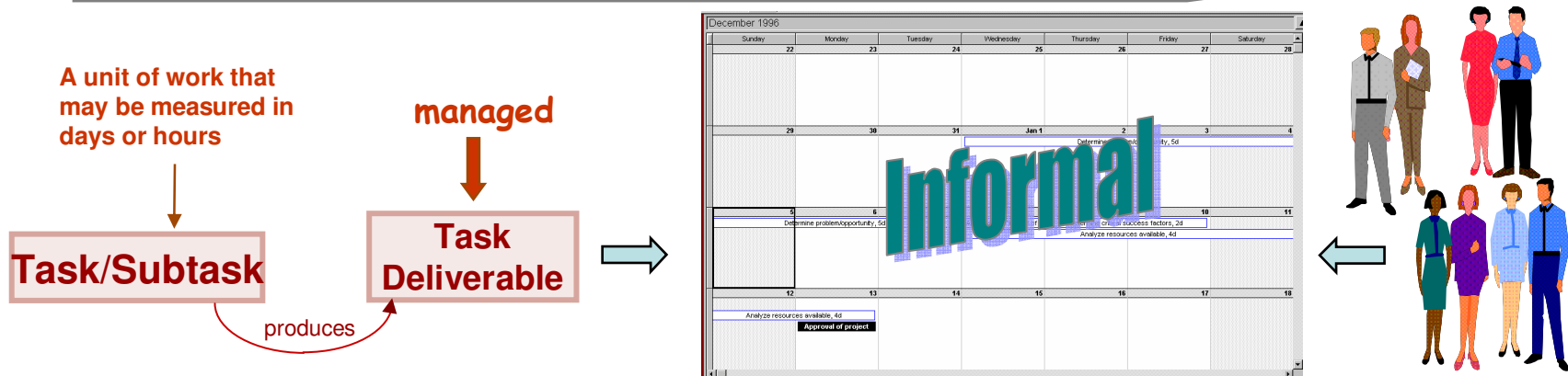# Parallel Development Activities

25

Step 6: Informal Internal Micro Plan
(Trademark of Larissa T. Moss, Method Focus Inc.)

1. Speculate on the total effort for the new DW/BI request

2. Break DW/BI application request into software releases

3. Create work breakdown structure for first software release

4. Create milestones from DDD to YAH for first release

5. Organize and assign parallel development tracks

6. Create detailed work assignments for weekly milestones

A unit of work that may be measured in days or hours

Task/Subtask

produces

managed

Task Deliverable

26

Example: Task Board

27

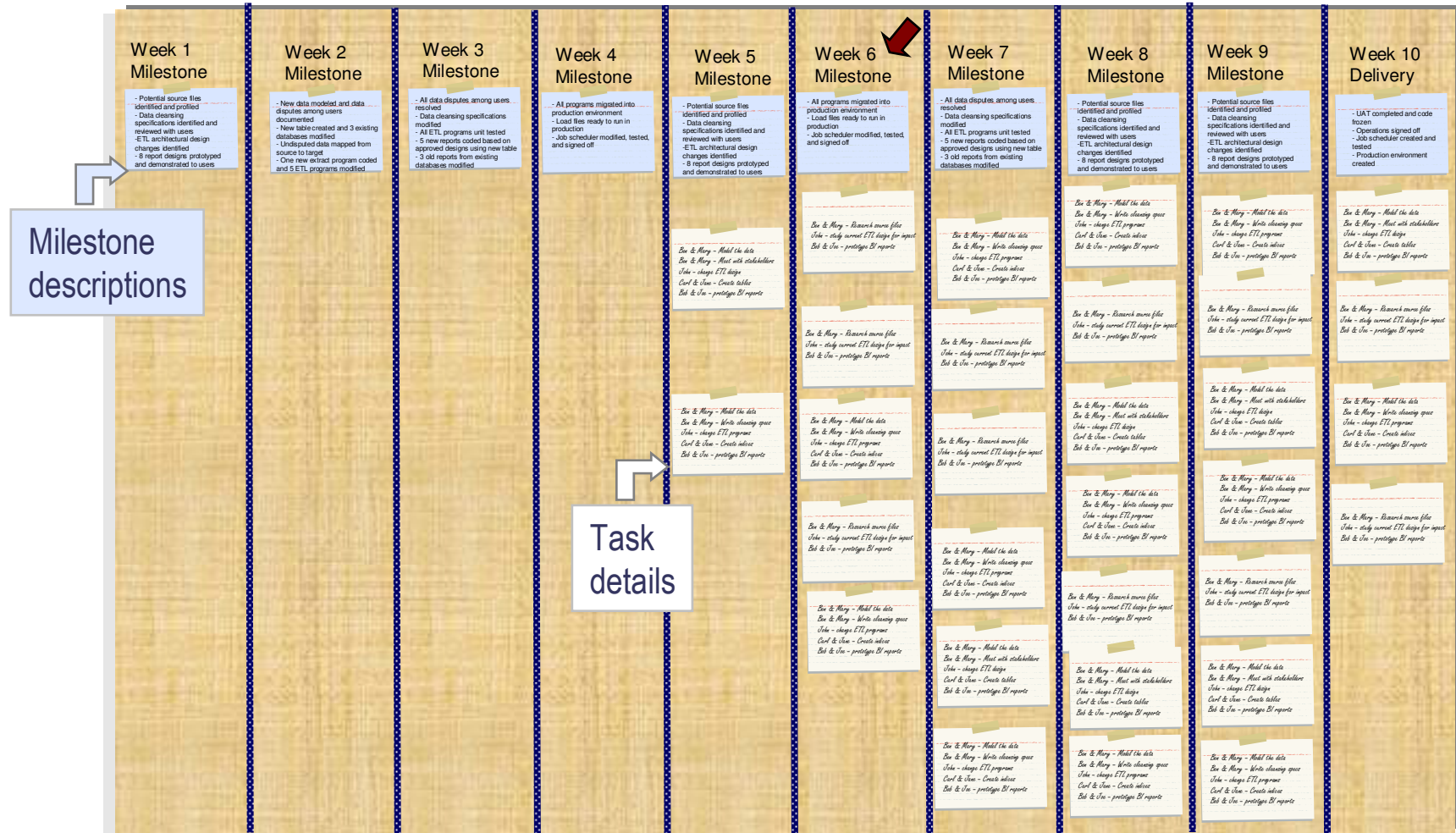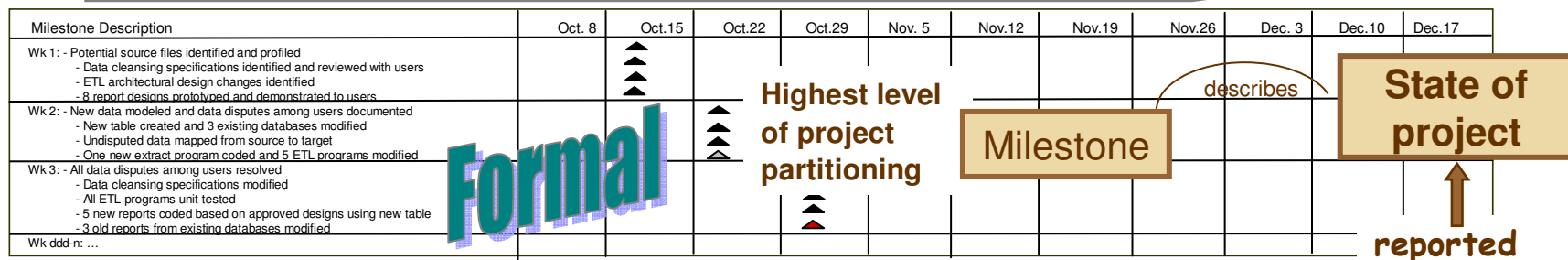# Step 7: Formal Management Macro Plan

(Trademark of Larissa T. Moss, Method Focus Inc.)

1. Speculate on the total effort for the new DW/BI request

2. Break DW/BI application request into software releases

3. Create work breakdown structure for first software release

4. Create milestones from DDD to YAH for first release

5. Organize and assign parallel development tracks

6. Create detailed work assignments for weekly milestones

7. Create milestone chart for progress reporting

Milestone Description	Oct. 8	Oct.15	Oct.22	Oct.29	Nov. 5	Nov.12	Nov.19	Nov.26	Dec. 3	Dec.10	Dec.17
Wk 1: - Potential source files identified and profiled - Data cleansing specifications identified and reviewed with users - ETL architectural design changes identified - 8 report designs prototyped and demonstrated to users											
Wk 2: - New data modeled and data disputes among users documented - New table created and 3 existing databases modified - Undisputed data mapped from source to target - One new extract program coded and 5 ETL programs modified											
Wk 3: - All data disputes among users resolved - Data cleansing specifications modified - All ETL programs unit tested - 5 new reports coded based on approved designs using new table - 3 old reports from existing databases modified											
Wk ddd-n: ...											

Highest level of project partitioning

Formal

describes

Milestone

State of project

reported

28

Example: Milestone Chart

Milestone Description	Oct. 8	Oct.15	Oct.22	Oct.29	Nov. 5	Nov.12	Nov.19	Nov.26	Dec. 3	Dec.10	Dec.17
Wk 1: - Potential source files identified and profiled	▲										
- Data cleansing specifications identified and reviewed with users	▲										
- ETL architectural design changes identified	▲										
- 8 report designs prototyped and demonstrated to users	▲										
Wk 2: - New data modeled and data disputes among users documented		▲									
- New table created and 3 existing databases modified		▲									
- Undisputed data mapped from source to target		▲									
- One new extract program coded and 5 ETL programs modified		△ (gray)									
Wk 3: - All data disputes among users resolved			△								
- Data cleansing specifications modified			▲								
- All ETL programs unit tested			▲								
- 5 new reports coded based on approved designs using new table			▲								
- 3 old reports from existing databases modified			▲ (red)								
Wk ddd-n: …											
Wk ddd-2: - UAT completed and code frozen										△	
- Operations signed off										△	
- Job scheduler created and tested										△	
- Production environment created										△	
Wk ddd-1: - All programs migrated into production environment											△
- Load files ready to run in production											△
- Job scheduler modified, tested, and signed off											△

Legend:
▲ Completed on time
△ (gray) Completed with significant delay
△ Not completed yet
▲ (red) Major modification to milestone (see detailed explanation on next page)

Planning the Next Software Release

(Trademark of Larissa T. Moss, Method Focus Inc.)

1. Review the total effort for the new DW/BI request

2. Review and adjust DW/BI application software releases

3. Create work breakdown structure for next software release

4. Create milestones from DDD to YAH for next release

5. Organize and assign parallel development tracks

6. Create detailed work assignments for weekly milestones

7. Create milestone chart for progress reporting

Second Release

Third Release

Fourth Release

Nth Release

Final Release

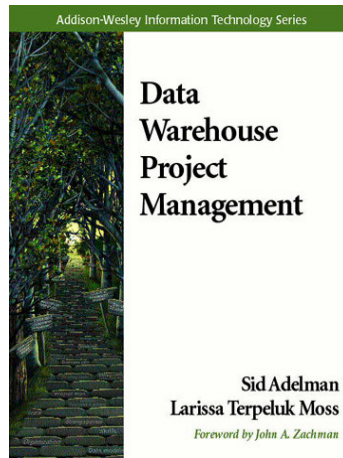# Agile Enterprise Data Warehousing
Radical idea or practical concept?

- Depends on what you call "BI" (see BI Maturity Model)
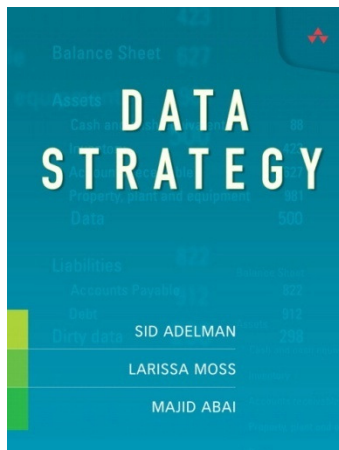- Depends on what you call "Agile"

Scrum ☑
XP ☑
CODE agile

"Data Marts"

AGILE

"Data Warehouses"

Extreme Scoping™ ☑
DATA agile

FOCUS on: Coding apps

FOCUS on: Enterprise data standardization and integration

"Spreadmarts"

"Production Reporting"

"Enterprise DW"

"Analytic Services"

| 1. Prenatal | 2. Infant | 3. Child | 4. Teenager | 5. Adult | 6. Sage |
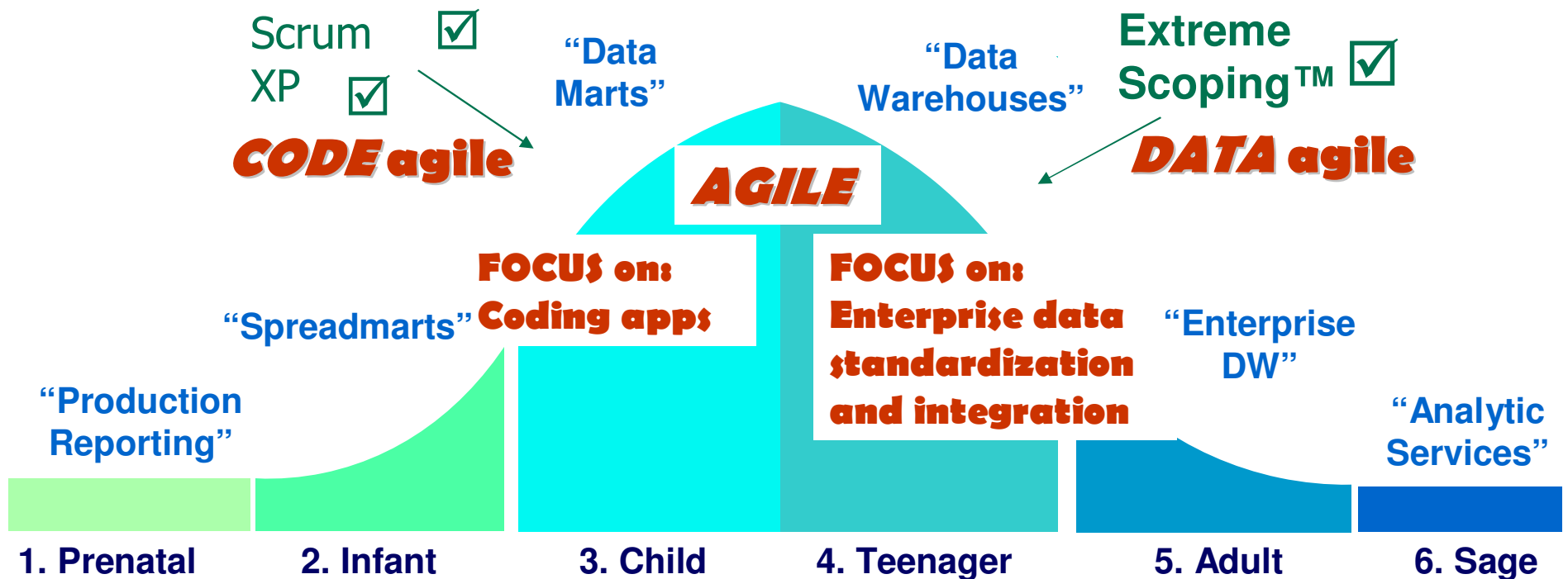
Source: BI Maturity Model, Wayne Eckerson, TDWI

Thank You

ISBN 0-201-61635-1
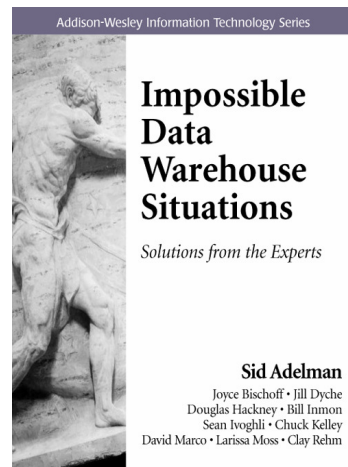
ISBN 0-201-76033-9

ISBN 0-201-78420-3

Coming soon ...

ISBN 0-321-24099-5

Larissa T. Moss
Method Focus, Inc.
methodfocus@earthlink.net