The Rework Cycle
Today’s Agenda

- Overview: Causes of Project Dynamics
- The Rework Cycle
- Integrating Tools in Project Planning
- Simple Model of Project Dynamics, Pt. 1
Typical Behavior Modes on a Project

Project Staffing

Time

Productivity (Normalised)

Time

© 2001 John Wiley & Sons, Ltd. All rights reserved. This content is excluded from our Creative Commons license. For more information, see http://ocw.mit.edu/fairuse.
Drivers of Project Dynamics

- The “rework cycle”
- Feedback effects
  - Negative, controlling
  - Positive, re-enforcing, often “vicious circles”
- Knock-on or domino effects within or between work phases
  - Knock-on or domino effects between projects
Today’s Agenda

- Overview: Causes of Project Dynamics
- The Rework Cycle
- Integrating Tools in Project Planning
- Simple Model of Project Dynamics, Pt. 1
Network Diagram for NMM Case (from 2010 Homework #1)
The Network View of a Program – Task Accomplishment

Task Accomplishment – Task X

Productivity

Staff

Work to Do

Work Being Accomplished

Work Done
Network Diagram for NMM Case

Two stocks and one flow subsumed within each task of network diagram
Aggregate Model of Network View

Initial Work to Do = 1000 tasks
Productivity = 1 task/mo/person
Staff = 40 people
Data Often Tell a More Complex Story

Productivity

Staff

Work to Do

Work Being Accomplished

Work Done

Staff

Work Being Accomplished

Years

PRODUCTIVITY?

OR...
Effort is being spent elsewhere …

(0-1: fraction not to be reworked)
Definitions

- **Productivity --**
  Work accomplished per hour of effort, regardless of completeness or correctness.

- **Fraction Correct & Complete --**
  Fraction of work just accomplished that will not need rework. “Work Quality”

Later we will revisit the iron triangle to consider “delivered quality.”
The Rework Cycle

- **Productivity**
  - Progress
  - Effort Applied
  - Fraction Correct and Complete

- **Original Work to Do**
- **Rework to Do**
- **Undiscovered Rework**
- **Work Done**

**Delay!**

Delay in Discovering Rework
Adding Rework Delays Completion ...

No Rework

With Rework

Work Done: Traditional With Rework
Work Done: Traditional View

Undiscovered Rework: Traditional With Rework

Adding Rework Delays Completion …
So, What Happens on Projects?

TIME

Staff

Effort on Initial Work

Effort on Rework
Rework creates and extended staffing tail

... Staffing experiences an extended tail

© 2001 John Wiley & Sons, Ltd. All rights reserved. This content is excluded from our Creative Commons license. For more information, see http://ocw.mit.edu/fairuse.
... or a second staffing peak

Program Staff, Simulated vs. Data (Equivalent Staff)

- Simulated
- Original Plan
- Actual

Disguised results from actual vehicle project

... A second staffing peak

Work Assignments of Staff to...

- Original Work
- Rework

Disguised results from actual vehicle project

© 2001 John Wiley & Sons, Ltd. All rights reserved. This content is excluded from our Creative Commons license. For more information, see http://ocw.mit.edu/fairuse.
Consequences of Undiscovered Rework

1. Overestimates of Progress
Additional Metrics (Used in Model)

- Fraction Reported to be Complete = \( \frac{\text{Work Done} + \text{Undiscovered Rework}}{\text{Initial Work to Do}} \)
- Fraction of Work Really Complete = \( \frac{\text{Work Done}}{\text{Initial Work to Do}} \)
Undiscovered rework generates progress-measuring errors ...
Rework Cycle Creates “The Lost-Year”...

Design Progress (Percent Complete)

Simulated

Disguised results from actual aerospace project

Source: Ken Cooper

Time

Year 1    Year 2    Year 3    Year 4    Year 5    Year 6    Year 7    Year 8    Year 9
and the “90% Syndrome” ---

Design Progress (Percent Complete)

Disguised results from actual aerospace project
Survey Question 1

Does your organization *measure* rework?

1. Yes, we keep data on how much rework is being *discovered*
2. Yes, we keep data on how much *work being done* is rework (vs original work)
3. Yes, both 1 and 2
4. No, we do not keep data on rework being discovered
Survey Question 2

Does your organization recognize rework in executing the project?

1. Yes, by adjusting progress estimates to reflect expected remaining undiscovered rework
2. Yes, by building rework tasks into the project graph
3. Both 1 and 2
4. No, we only react as rework is discovered
Consequences of Undiscovered Rework

1. Overestimates of Progress
2. Build Errors Into Downstream Work
Survey Question 3

How significant is the “Errors on Errors” feedback on typical projects in your organization?

1. Very
2. Modest
3. Weak
4. None
Units of Work – SD Model

- “Tasks” -- ideally, of uniform size and fungible
- In actual applications, might use things like:
  - Drawings, work packages, lines of code, …
  - Tons of concrete, feet of steel, feet of wiring, ...
  - Parts designed
  - Sometimes %
- Can represent “precedence” constraints
“Quality”

Need to distinguish between:

- Instantaneous “work quality” (fraction correct and complete)
- Delivered “quality” – remaining “bugs” (undiscovered rework) when product shipped
- Product capabilities, “fit and finish” (a part of scope in model)
Fraction Correct and Complete

- Represents unplanned iterations; planned iterations are separate tasks.
- Sources of errors many fold:
  - Mistakes from inexperience, fatigue, ...
  - Technical complexity/uncertainty
  - Work done “correctly” but ultimately needing rework because it builds on
    - Incorrect prior work
    - Assumptions about technology or customer requirements which prove incorrect
Two Types of Iteration

**Planned Iteration**
- Caused by needs to “get it right the first time.”
- We know where these iterations occur, but not necessarily how much.
- Planned iterations should be facilitated by good design methods, tools, and coordination.

**Unplanned Iteration**
- Caused by errors and/or unforeseen problems.
- We generally cannot predict which unplanned iterations will occur.
- Unplanned iterations should be minimized using risk management methods.

*Separate tasks in the rework cycle*

*“Quality” problems in the rework cycle*
Qualitative Insights

- Undiscovered Rework is one of the most important single factors driving schedule and budget overruns
- Most management reporting systems overestimate real progress and discourage reporting of rework
Lessons: Recognize the rework cycle and minimize its consequences

- **Increase**
  - Build Errors Into Downstream Work
  - Overestimates of Progress
  - Fraction Correct and Complete

- **Reduce**
  - Be Realistic
  - Delay in Discovering Rework

- **Rework Generation**
  - Original Work to Do
  - Work Done
  - Rework to Do
  - Undiscovered Rework

- **Productivity**
  - + Progress

- **Effort Applied**
  - +

- **Lessons**
  - Recognize the rework cycle and minimize its consequences
Dynamics of Project Performance

The “rework cycle”
- Fraction correct and complete ✓
- Undiscovered rework ✓

Feedback effects on productivity and fraction correct (Next class)
- Negative, controlling
- Positive, re-enforcing, often “vicious circles”

Knock-on effects between work phases (Next)
- Availability and quality of work products
- Progress to discover upstream rework
Today’s Agenda

- Causes of Project Dynamics
- The Rework Cycle
- Integrating Tools in Project Planning
- Simple Model of Project Dynamics, Pt. 1
Fundamental Approaches

- Network-based (graph theory) methods
  - CPM, PERT, ....
  - Task is a node or an arc

- Matrix-based methods
  - DSM - Tasks are columns and rows
  - Interrelationships are off-diagonal entries

- System Dynamics
  - Feedback loops, causal relationships
  - Stocks and flows simulation
  - Tasks that are done or waiting to be done are stocks – “amount of work”
  - Doing project work causes a “flow”

What have we learned thus far?
Network View of CityCar Project
Network Diagram for NMM Case
(from 2010 Homework #1)

Note: Diesel, Elec Drive, and Software do not show separate “design” and “build” tasks.
Organizing the Tasks by “Metatask”
"Gantt" Chart with Metatasks Shown

Note: dates approximate; JML estimates of Diesel & Electric, Software Design/Build Split

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a Start</td>
<td>1/1/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>b Requirements</td>
<td>1/1/10</td>
<td>3/25/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>c Diesel Specs</td>
<td>3/25/10</td>
<td>5/6/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>d Elec Drive Specs</td>
<td>3/26/10</td>
<td>5/6/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>g Software Specs</td>
<td>3/26/10</td>
<td>7/5/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>e Layout</td>
<td>3/26/10</td>
<td>6/1/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>k Chassis Design</td>
<td>6/2/10</td>
<td>10/21/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>l Body Design</td>
<td>6/2/10</td>
<td>10/5/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>s Body Chassis Prototype</td>
<td>10/22/10</td>
<td>12/28/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>f Electronics Design</td>
<td>5/7/10</td>
<td>9/9/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>h Software Dev</td>
<td>9/10/10</td>
<td>4/7/11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>p Electronics Delivery</td>
<td>9/10/10</td>
<td>12/20/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>q Electronics Software Int</td>
<td>4/8/11</td>
<td>5/19/11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>i Diesel Dev</td>
<td>5/7/10</td>
<td>1/13/11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>j Elec Drive Dev</td>
<td>5/7/10</td>
<td>9/9/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>n Diesel Delivery &amp; Check</td>
<td>1/14/11</td>
<td>1/31/11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>o Integration</td>
<td>2/1/11</td>
<td>4/25/11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>r Charging Assembly</td>
<td>4/26/11</td>
<td>5/11/11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>t Power Vehicle Int</td>
<td>5/20/11</td>
<td>8/11/11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>m Internal Fittings</td>
<td>10/22/10</td>
<td>12/28/10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>u Final Assembly</td>
<td>8/12/11</td>
<td>9/22/11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>v Dynamo Lab Test</td>
<td>9/23/11</td>
<td>11/3/11</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>w Road Test</td>
<td>11/4/11</td>
<td>2/13/12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x Car Show Setup</td>
<td>11/4/11</td>
<td>1/5/12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>y Finish</td>
<td>2/13/12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
What Messages Come From DSM and SD regarding the network/Gantt plan?

- DSM?
- SD?
- How/when is rework discovered?
DSM Identifies Coupled Tasks & Areas of Unplanned Iterations (from Homework #2)

Coupled Tasks – Possible planned Iterations (in design)
Unplanned Iterations (why not back to design phase?)
Observation/Question: no coupling between major design phases?
Network Diagram With “Build” Steps Added and Major Work Phases Highlighted
In an SD Model, Each Phase of Work Could Be Represented by a Rework Cycle

- **Original Work to Do**
- **Rework to Do**
- **Original Work Done Correctly**
- **Work Done**

**Staff**
- Staff on Original Work
- Staff on Rework

**Productivity**
- Original Work Being Accomplished
- Rework Being Accomplished

**Fraction Correct and Complete**
- Rework Generation on Original Work
- Rework Generation on Rework

**Undiscovered Rework**
- Rework Discovery
Assumptions:

**Requirements**
- Scope = 100 Tasks
- Staff = 4
- Productivity = 2 tasks/month/person
- Duration = 12.5 months

**Design**
- Scope = 1000 tasks
- Staff = 20
- Productivity = 4 tasks/month/person
- Duration = 12.5 months

**Build/Test**
- Scope = 1000 tasks
- Staff = 40
- Productivity = 1 tasks/month/person
- Duration = 25 months
Simulation of project assuming no rework ...
Sources of Rework -- Categories

1. Classical “Quality” or design mis-execution.
3. Work done “correctly” but ultimately needing rework because it builds on
Classical “Quality”

There are two (or more) important sources of these errors –

- “People” factors such as fatigue, inexperience, skill mismatches, etc.
- “Coupled” tasks that require shared information but which are done independently
- Organizational size & complexity?

These types of errors can in theory be discovered by further “design” work, such as doing downstream design work, QA, design reviews, planned iteration, etc.
Technical complexity/uncertainty; customer uncertainty.

- Technical: “novelty” of the project
- Customer: Same logic might apply to novel/new “uncertain customer requirements”? I.e., where say a software product is being developed and the customer may not know what they want until they see it in practice.]

- These types of errors can only be discovered by doing build and test work. Fixing these errors may just require redoing tasks, but may also require new tasks to make the technology work (e.g., and additional controller).
“Knock-on” Rework

Work done “correctly” but ultimately needing rework because it builds on

- Incorrect prior work
- Assumptions about technology or customer requirements which prove incorrect.
- Exogenous changes in requirements, scope, etc.

These errors could be discovered in design or build/test, depending on the source.
How does Design Affect Build?

- Design Fraction
  - Correct and Complete
  - Reported Complete

- Build/Test Fraction
  - Correct and Complete
  - Reported Complete
How is Design Rework Discovered?
Simulation Incorporating Rework and Rework Discovery

Notes: (1) Details of simulation do not correspond to NMM Case; (2) so far, we do not represent project control actions and effects, so staff/hours worked are constant.
Design Original Work to Do: Three Phase Four Stock V1
Design Undiscovered Rework: Three Phase Four Stock V1
Design Rework to Do: Three Phase Four Stock V1
Design Work Done: Three Phase Four Stock V1
Progress and Rework Discovery

Design Fraction Complete

Fraction Design Rework Discovered

Discovery by design

Discovery by build
Rework Increases Cost and Delays Finish

With Rework

Without Rework
Revised Network/Gantt to reflect rework discovery (by build, not design)

Insert some Rework Tasks in Design, more in Build/Test

<table>
<thead>
<tr>
<th>2010</th>
<th>2011</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apr</td>
<td>May</td>
</tr>
<tr>
<td>Electronics Design</td>
<td>90</td>
</tr>
<tr>
<td>Software Dev</td>
<td>150</td>
</tr>
<tr>
<td>Electronics Delivery</td>
<td>72</td>
</tr>
<tr>
<td>Electronics Software Int</td>
<td>30</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Electronics &amp; Software -- Rework Discovered in Build/Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>f Electronics Design</td>
</tr>
<tr>
<td>h Software Dev</td>
</tr>
<tr>
<td>p Electronics Delivery</td>
</tr>
<tr>
<td>q Electronics Software Int</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Design</th>
<th>Rework</th>
<th>Build/Test</th>
<th>Coupled</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>R</td>
<td>R R R R R R R R</td>
<td></td>
</tr>
<tr>
<td>R</td>
<td>R</td>
<td>R R R R R R R R</td>
<td></td>
</tr>
<tr>
<td>R</td>
<td>R</td>
<td>R R R R R R R R</td>
<td></td>
</tr>
</tbody>
</table>
Management Issues

- Planned Iterations (vs dealing with unplanned iterations later) – How many iterations?
- When to start build phases – how much overlap with design phases.
- Factors to Consider:
  - Type of project (determines amount of rework discoverable in design, amount of rework, number of tasks which must be repeated for each iteration)
  - Relative cost of design vs build/test (determines the relative cost of spending more in design)

More on these later in term ...
Role of Tools

- **SD:**
  - Evaluate the macro tradeoffs in terms of impact on cost and schedule
  - Coming up with a good staffing plan

- **DSM:**
  - Determine where design iterations are needed and how many tasks would need to be repeated per iteration

- **Network:**
  - Guide operational task planning and resource scheduling
Dynamics of Project Performance

The “rework cycle”
- Fraction correct and complete ✓
- Undiscovered rework ✓

Feedback effects on productivity and fraction correct (Next class)
- Negative, controlling
- Positive, re-enforcing, often “vicious circles”

Knock-on effects between work phases
- Availability and quality of work products ✓
- Progress to discover upstream rework ✓
Knock-on Between Phases: A system dynamics model usually represents several phases of work ...

Productivity and Fraction Correct Effects (availability and quality of upstream work)

Notes: Suppliers might be a “phase”
Why Represent Separate Phases?

- Correspond to gate reviews and deliverables
- Different units of work (and therefore data)
- Different types of labor
- Disproportionate increase in costs of fixing errors as move downstream
- Address issues of overlap/concurrency
- Opportunity to adjust plans, delay start, between phases
Today’s Agenda

- Overview: Causes of Project Dynamics
- The Rework Cycle
- Integrating Tools in Project Planning
- Simple Model of Project Dynamics, Pt. 1
Hard tools force us to be more explicit, and accurately simulate the consequences of our models ... 

“Soft” tools --
- behavior-over-time graphs
- cause-effect diagramming
- mental simulation

“Hard” tools --
- computer models
- computer simulation
- calibration to data
- sensitivity and what-if analyses

Tools for describing dynamics
Tools for quantifying dynamics
We will use two models ...

- Simple rework cycle model with project control feedbacks
  - HW#3 – develop simple model without feedbacks
  - Feedbacks added in class, given in HW#5

- Full rework cycle model with two phases of work
  - No project control feedback
  - Model given to you for HW#3 and HW#5
Development of “Simple” Project Model 1

- Rework cycle model (HW#3)
  - Three stocks
  - Variable rework discovery time
  - “Errors Build Errors” Feedback

- Project control & Side Effects (HW#5)
  - Work Intensity/Schedule Pressure & “Haste Makes Waste”
  - Staffing & Experience Dilution
  - Slip Schedule
Two Views of the Rework Cycle

Complete Model

Simplified Version

The simplified version assumes that rework tasks require the same effort as original tasks, and that it is not important to distinguish between original work and rework.
Simple Rework Cycle Model

Steps 1-4 of Homework #3
Rework Discovery Depends on Progress

Graph Lookup - Fraction of Rework Discovered

Fraction Really Complete

Delay in Discovering Rework
This reflects the average delay in discovering discoverable rework, such as from QA activities or downstream work.

Fraction of Rework Discovered
This reflects the fraction of undiscovered rework that is discoverable at any point in the project based on the activities taking place.

Rework Discovery

<Project Finished>
“Errors on Errors” Feedback

Complete Simple Model 1

- Work to Do
- Rework Generation
- Undiscovered Rework
- Work Done

- Fraction of Work Believed Done Correct and Complete
- Effect of Undiscovered Rework on Fraction Correct
- Initial Work to Do
- Rework Discovery
- Delay in Discovering Rework
- Fraction of Rework Discovered

- Minimum Time to Finish a Task
- Potential Work Rate Based on Staff
- Work Being Accomplished

- Fraction Correct and Complete
- Normal Fraction Correct and Complete
- Sensitivity of Fraction Correct to Undiscovered Rework
- Maximum Effect of Undiscovered Rework on Fraction Correct

- Normal Staff
- Effort Expended
- Cumulative Effort Expended

- Project Finished
- Fraction Complete to Finish

- Work Believed to Be Done
- Work Done
- Work Done Correctly
- Rework Generation
- Rework Discovery

- Maximum Work Rate Based on Tasks Available

- Fraction Reported Complete
- Fraction Really Complete

- Cumulative Work Done
- Rate of Doing Work

- Normal Staff

- Minimum Time to Finish a Task

- This reflects the average delay in discovering discoverable rework, such as from QA activities or downstream work.

- This reflects the fraction of undiscovered rework that is discoverable at any point in the project based on the activities taking place.

- The effects of undiscovered rework on the fraction of work believed done correctly and complete.

- Sensitivity of the fraction of work being accomplished to undiscovered rework.

- Maximum effect of undiscovered rework on the fraction of work being accomplished.

- Normal fraction correct and complete.

- Sensitivity of the normal fraction correct and complete to undiscovered rework.

- Fraction reported correct.

- Normal work rate based on tasks available.

- Effort expended.

- Cumulative effort expended.

- Project finished.

- Fraction complete to finish.

- Work believed to be done.

- Work done.

- Work done correctly.

- Rework generation.

- Rework discovery.

- Maximum work rate based on tasks available.

- Minimum time to finish a task.

- Delay in discovering rework.
The effect of undiscovered rework on fraction correct is assumed to be proportional -- an error in past work creates an error in current work. Given that in this simple model fraction correct represents several effects of work errors, this strong relationship may be reasonable.
Work Flows & Staffing in “Simple” Two Phase Model
ESD.36 System Project Management
Fall 2012

For information about citing these materials or our Terms of Use, visit: http://ocw.mit.edu/terms.