Week 3:
Project roles and staffing

What are the major project roles?

How are they specified?

How do we fill them?

Roles versus jobs

<table>
<thead>
<tr>
<th>Job (or position)</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>relatively permanent</td>
<td>exists for duration of an activity (Project roles are always temporary.)</td>
</tr>
<tr>
<td>individual occupies exactly one at a time</td>
<td>Individual may take on multiple roles</td>
</tr>
<tr>
<td>recognized and administered by H.R.; tied to salary, rank, status, etc.,</td>
<td>needn't be official -- a management tool at any level</td>
</tr>
</tbody>
</table>

Some Project Roles
- Project manager
- Project sponsor (user, customer)
- Systems analyst
- Designer / Chief programmer
- Programmer
- Database manager
- Tester
- Quality reviewer

Do we know what each of them does?
Who are the project team members?
Who are the "developers"?

Project Role Categories
- Project team members
  - Project manager
  - Systems analyst
  - System designer
  - Programmer
- Project consultants / advisors
  - Database manager
  - Quality reviewer
  - Methodology administrator
  - Systems integrator
- Project sponsors
  - Manager / funding authority
  - Prospective user

COMP 370 Fall, 2014
Conrad Weisert
Defining a role

- To successfully manage people in the various project roles, we must understand exactly what each role does.
- The incumbent should understand his or her own role in exactly the same way as management does.
- The role definition, then, is like a contract between the incumbent and his or her boss.

Formal Role Definition

- Much more concrete than the typical organization's job descriptions
- Avoids vague activity terms, such as:
  - participates in . . .
  - assists with . . .
  - contributes to . . .
  - advises on . . .
  - concurs in . . .

Accountability for results

- More important than the activities an individual engages in are the results he or she is accountable for producing.
- Therefore a rigorous (and useful) role definition must specify measures of accountability.

Role Definition structure

- A successful form of role definition is the so-called RAA structure, consisting of three sections:
  - Responsibility -- The activities the incumbent performs
  - Accountability -- The results the incumbent is expected to produce (also called "measures of performance")
  - Authority -- What the incumbent may do without specific higher approval.
Accountability versus annual objectives
- Many organizations require a list of goals an individual commits to accomplish during the coming year.
- An accountability list is a list of results the individual commits to produce.

How are they different?
Let's look at an example:
Senior programmer

A critically important role: programmer
- Origin and evolution
- Role in a modern I.S. project
- Titles and status
- Productivity

1950s origins
- A programmer (a technically competent problem solver) would confer with a problem sponsor (user with funds and authority) on requirements for a new or modified application.

What happened next?

Why didn't that work for complicated business applications?

Changing role of the programmer: the 1950's
- Given a problem, implement a computer-based solution to it a prestigious job!
1960s crises
- Growing demand from business led to
  - larger project teams with > 4 programmers
  - hordes of poorly trained, underqualified programmers.
  - major project fiascos
- Desperate organizations "promoted" their most mature and knowledgeable programmers to **systems analysts**.

  *What did those systems analysts do?*

1970's maturing
- The "structured revolution" (~1975-1985) established a clear distinction between programming and systems analysis.
  - The **systems analyst** would rigorously **define** the problem to be solved
  - The **programmer** would **solve** a well-defined problem, usually using computer software.

Changing role of the programmer: the 1960's
- Given a **detailed design** (usually flowcharts & record layouts) code and test a program to implement it.

  *Who created that design?*

  - Explosion of demand (why?) --
  - Hordes of poorly qualified trainees.

The role of the modern programmer
- Given a **well-defined problem**, to design and implement a computer-based solution.
  - That solution may or may not require writing new software (programming)
  - A competent professional programmer is aware of alternatives and tries to **avoid** writing programs.

**Programming**
- design
- coding
- testing

  *What about documenting?*
Role of the Systems Analyst

- A highly knowledgeable professional, who prepares the specification of exactly what a proposed new application system will do (not how it will do it).
- Is that a higher-level activity than programming?
- Should an analyst know how to program?

See COMP320 for details.

Choosing programmers

- Studies repeatedly show a 20-to-1 range in productivity among experienced programmers.
- Their range in salary is at most 3-to-1
- Furthermore, the most productive programmers usually produce higher quality programs than the least productive.

What strategy does that suggest?

Offshore programming

- You can often outsource a programming job to a country with lower labor rates
  - Many Indian (etc.) programmers are highly competent
  - Their hourly rates are a small fraction of those of U.S. programmers

When is outsourcing a wise choice?

- Can we outsource systems analysis, too?
Offshore programming (continued)

- Other countries are competing for that business.

  What are some of them?.
  What do they offer?

The bottom line on project roles

- You can call the roles anything you like, as long as:
  - Every result that a project must produce is on the accountability list of some project role
  - Every incumbent has both the resources and the authority to produce the results he or she is accountable for.
  - No two roles have conflicting authority.

Roles for software testing (validation / verification)

<table>
<thead>
<tr>
<th>Testing stage</th>
<th>Main Responsibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit</td>
<td>Programmer</td>
</tr>
<tr>
<td>Integration</td>
<td>Programmers</td>
</tr>
<tr>
<td>System</td>
<td>Testers, analysts</td>
</tr>
<tr>
<td>Volume</td>
<td>Testers</td>
</tr>
<tr>
<td>Acceptance</td>
<td>Users, management</td>
</tr>
</tbody>
</table>

Which ones are we interested in? Why?

What about alpha and beta testing?