People and Organizations

Anne Banks Pidduck
Adapted from John Musser

Today

- Project Roles & Team Structure
- Project Control
- Status Reporting

Project Roles

- Programmers (system engineers)
  - Technical lead, architect, programmer
- Quality Assurance (QA) engineers and testers
  - QA Manager, QA Lead, QA staff
- DBAs
  - DB Administrator, DB Programmer, DB Modeler
- Configuration management engineers
  - build engineers
- Network engineers, System Administrators

Project Roles ..2

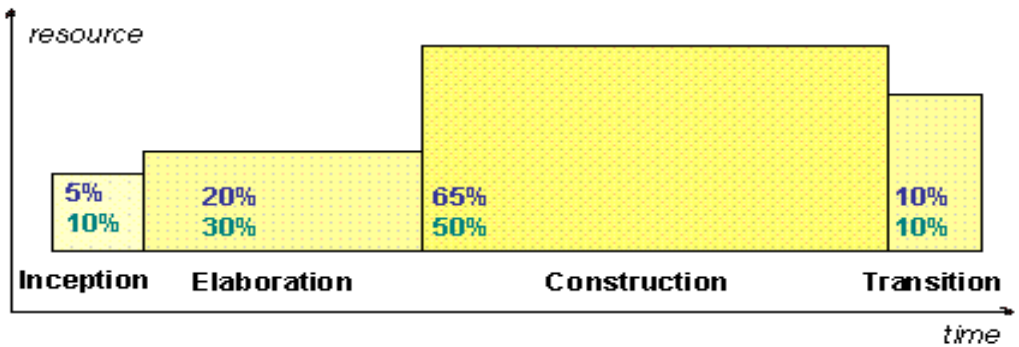
- Analysts and business analysts
- User Interface Designers
- Information Architects
- Documentation writers
  - editors, documentation specialist
- Project manager
- Other
  - Security specialist, consultants, trainer

Project Roles

- You need to decide which of these are necessary
- Depends on what you're building
  - How big is it?
  - Is it UI intensive? Data intensive?
  - Are you installing/managing hardware?
  - Do you need to run an operations center?
  - Is it in-house, contract, COTS, etc?
- Depends on your budget

Staffing Profile

- Projects do not typically have a ‘static team size’
- Who and how many varies as needed



Legend:
Actual Effort (% of project total)
Schedule (% of project total)

Roll-on & Roll-off

- PM must have a plan as to how & when
- Roll-on
  - Hiring or ‘reserving’ resources
  - Ramp-up time
    - Learning project or company
- Roll-off
  - Knowledge transfer
  - Documentation
  - Cleanup

Staffing Management Plan

- Part of Software Development Plan
- Includes
  - What roles needed, how many, when, who
  - Resource assignments
  - Timing: Start/stop dates
  - Cost/salary targets (if hiring)
- Project Directory
  - Simply a list of those involved with contact info.
- Team size: often dictated by budget

Team Structure

What is the team's objective?

- Problem resolution
  - Complex, poorly-defined problem
  - Focuses on 1-3 specific issues
  - Ex: fixing a showstopper defect
  - Sense of urgency
- Creativity
  - New product development
- Tactical execution
  - Carrying-out well-defined plan
  - Focused tasks and clear roles

Team Models

- Two early philosophies
  - Decentralized/democratic
  - Centralized/autocratic
- Variation
  - Controlled Decentralized

Team Models

- Business Team
  - Most common model
  - Technical lead plus team (rest team at equal status)
  - Hierarchical with one principal contact
  - Adaptable and general
  - Variation: Democratic Team
    - All decisions made by whole team
    - See Weinberg's "egoless programming" model

Team Models

2. Chief-Programmer Team

- From IBM in 70's
  - See Brooks and Mythical Man-Month
- a.k.a. 'surgical team'
- Puts a superstar at the top
  - Others then specialize around him/her
    - » Backup Programmer
    - » Co-pilot or alter-ego
    - » Administrator
    - » Toolsmith
    - » "Language lawyer"
- Issues
    - » Difficult to achieve
    - » Ego issues: superstar and/or team
- Can be appropriate for creative projects or tactical execution

Team Models

3. Skunkworks Team
   - Put a bunch of talented, creative developers away from the mother ship
     - Off-site literally or figuratively
   - Pro: Creates high ownership & buy-in
   - Con: Little visibility into team progress
   - Applicable: exploratory projects needing creativity
     - Not on well-defined or narrow problem

Team Models

4. SWAT Team

- Highly skilled team
- Skills tightly match goal
- Members often work together
- Ex: security swat team, Oracle performance team

Team Models

- Large teams
  - Communication increases multiplicatively
    - Square of the number of people
    - 50 programmers = 1200 possible paths
    - Communication must be formalized
  - Always use a hierarchy
  - Reduce units to optimal team sizes
    - Always less than 10

Team Size

- What is the optimal team size?
  - 4-6 developers
    - Tech lead + developers
  - Small projects inspire stronger identification
  - Increases cohesiveness
  - QA, operations, and design on top of this

Hiring

- “Hire for Trait, Train for Skill”
- Look for: “Smart, Gets Things Done”
- Balance the team

Responsibility Assignment Matrix

- A resource planning tool
- Who does What
- Can be for both planning and tracking
- Identify authority, accountability, responsibility
- Who: can be individual, team or department
- Can have totals/summary at end of row or column (ex: total Contributors on a task)

Simple RAM

Item	WBS	Description		Sponsor	Developer	Developer	QA	Customer
1	1	Initiate Project		A				
2	1.1	PMP Signoff		A				R
3	1.2	Initial UI			L	C		R
4	1.3	DB Model			C	L		
5	1.4	Start Test					L	
	Legend							
	A	Approval						
	L	Lead						
	S	Secondary						
	C	Contributor						
	R	Reviewer						

Sample RAM With Stakeholders

Item		Development	Customer A	Customer B	Mgmt	QA	
Unit Test		A	S	S	R	A	
Systems Test		P	R	R	R	R	
Beta Test		P	R	R	P	R	
User Acceptance Test		A	S	S	S	S	
Accountable	A						
Participant	P						
Reviewer	R						
Sign-off Required	S						

Skills Matrix

- Another resource planning tool
- Resources on one axis, skills on other
- Skills can high level or very specific
- Cells can be X's or numeric (ex: level, # yrs.)

	Analyst	Developer (Java)	Developer (HTML)	QA Tester	Database Design
Dilbert	7	2			
Larry			8		4
Sarah	4	4			
Boss				4	
Fred					5