

Informatik-Projektentwicklung

– Lecture 10 –

Prof. Dr. Peter Müller

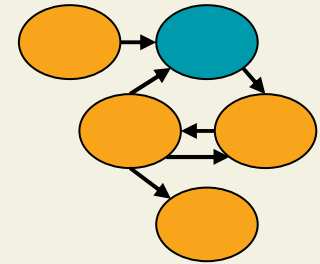
Software Component Technology

Wintersemester 03/04

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

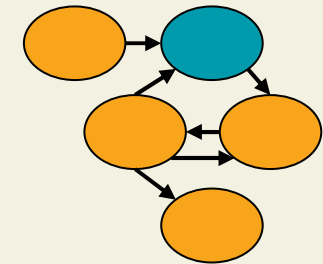
Cost Management



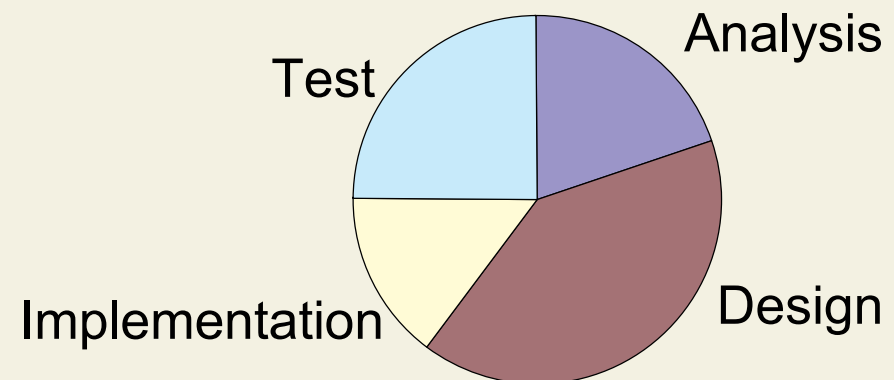
Analogous estimating:

- a. Uses bottom-up estimating techniques
- b. Is used most frequently during the execution phase of the project
- c. Uses top-down estimating techniques
- d. Uses actual detailed historical costs

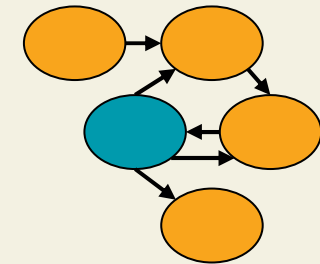
Top-Down Estimates



- Estimation by analogy or comparison
- Estimate is based on experience and historical data
- Characteristics
 - Less accurate than other methods
 - Quicker and less expensive than other methods
 - No WBS needed
- Typical figures for software development
 - Analysis 20%
 - Design 40%
 - Implementation 15%
 - Test 25%



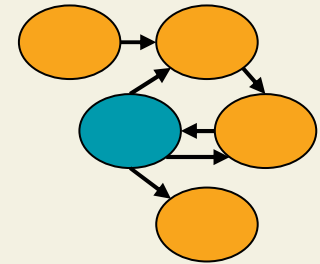
Cost Management



A Cost Performance Index of 0.89 means:

- a. At this time, we expect the total project to cost 89% more than planned
- b. When the project is completed we will have spent 89% more than planned
- c. The project is only progressing at 89% of that planned
- d. The project is only getting 89 cents out of every dollar invested

Cost Performance Index (CPI)



- Compares **budgeted cost** to **actual**
- Indicates the **efficiency** of the project

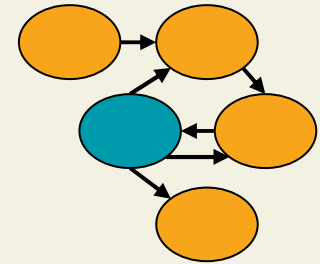
$$CPI = \frac{EV}{AC}$$

- How much do we get out of one Franc we spend?

Activity	PV(t)	AC(t)	EV(t)
Paint wall	800	1000	800
Paint ceiling	400	300	300
Total	1.200	1.300	1.100

$$CPI = \frac{1.100}{1.300} = 85\%$$

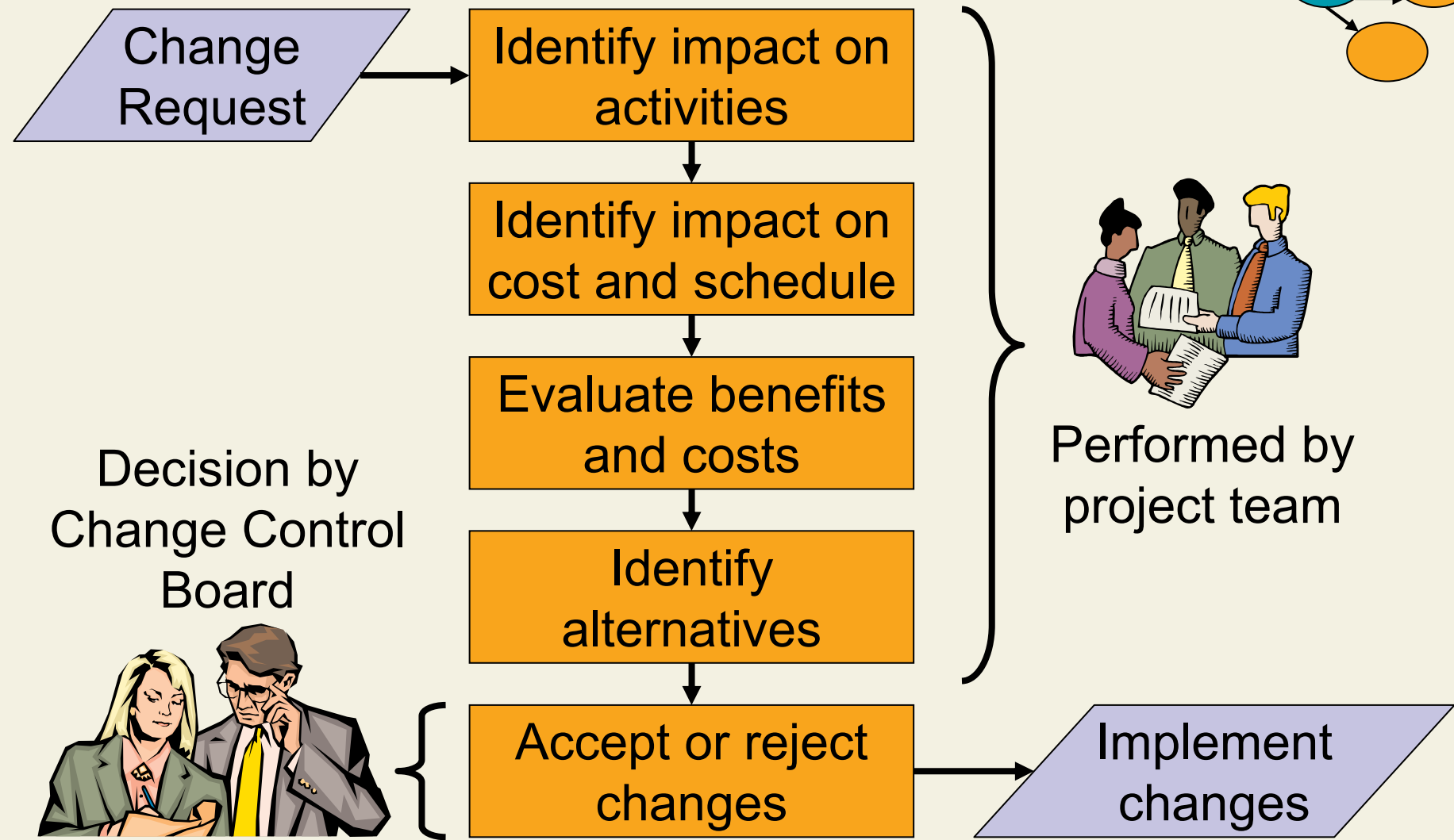
Change Management



The project is mostly complete. However, the customer wants to make a major change to the scope of work. The project manager should:

- a. Meet with the project team to determine if this change can be made
- b. Ask the customer for a description of the change
- c. Explain that the change cannot be made at this point in the process
- d. Inform management

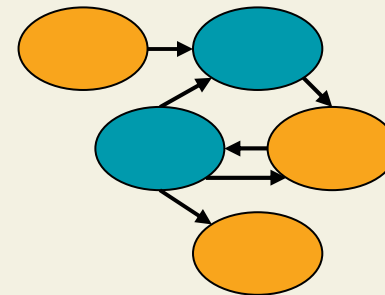
Change Control Process



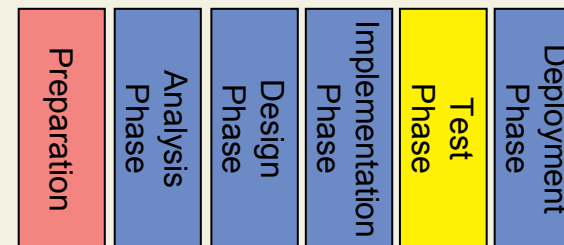
Agenda for Today

10. Risk Management and Test Phase

10.1 Risk Management



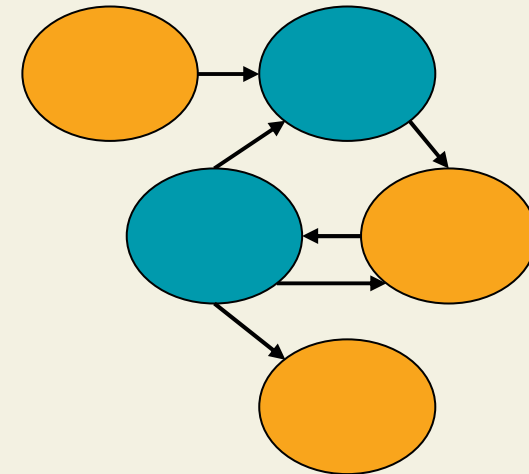
10.2 Test Phase



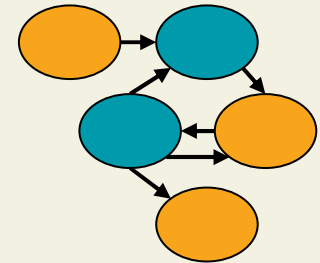
10. Risk Management and Test Phase

10.1 Risk Management

10.2 Test Phase



Risk



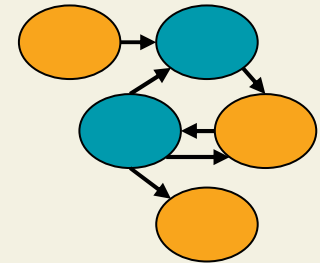
- Definition:

An uncertain event or condition that, if it occurs, has a positive or negative effect on a project objective

- Risks have three components

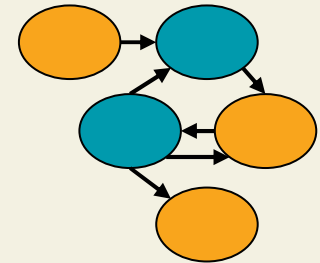
- A possible future event (uncertainty)
- Probability of the occurrence of that event (likelihood)
- Impact of that event (consequence)

Risk Classification



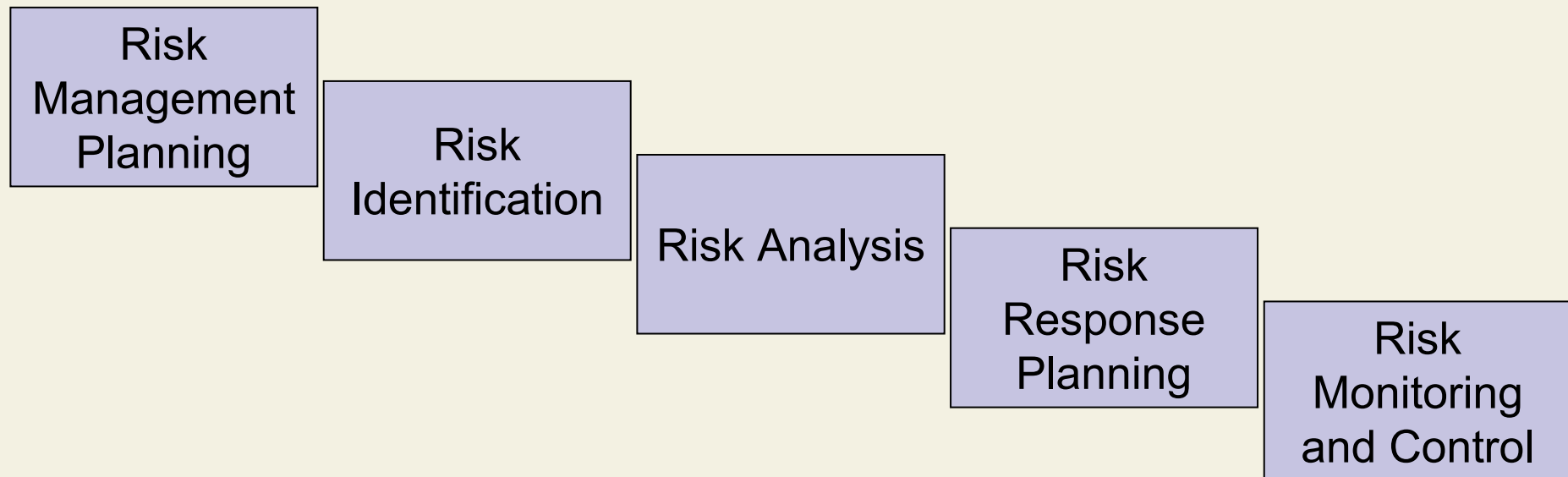
- Known risks
 - Unclear requirements
 - Inexperienced team
- Unknown risks: Foreseen based on experience
 - Difficult communication with customer
 - Fluctuation within team
- Unknowable risks: Cannot be foreseen
 - Half of the team gets fish poisoning at first social event
 - Earthquake wipes out production plant

Risk Management

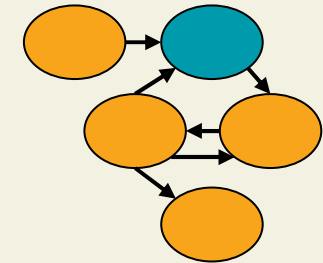


- Definition:

Systematic process of identifying, analyzing, and responding to project risk. It includes minimizing the consequences of adverse events to project objectives.



Risk Mgmt. Planning: Summary

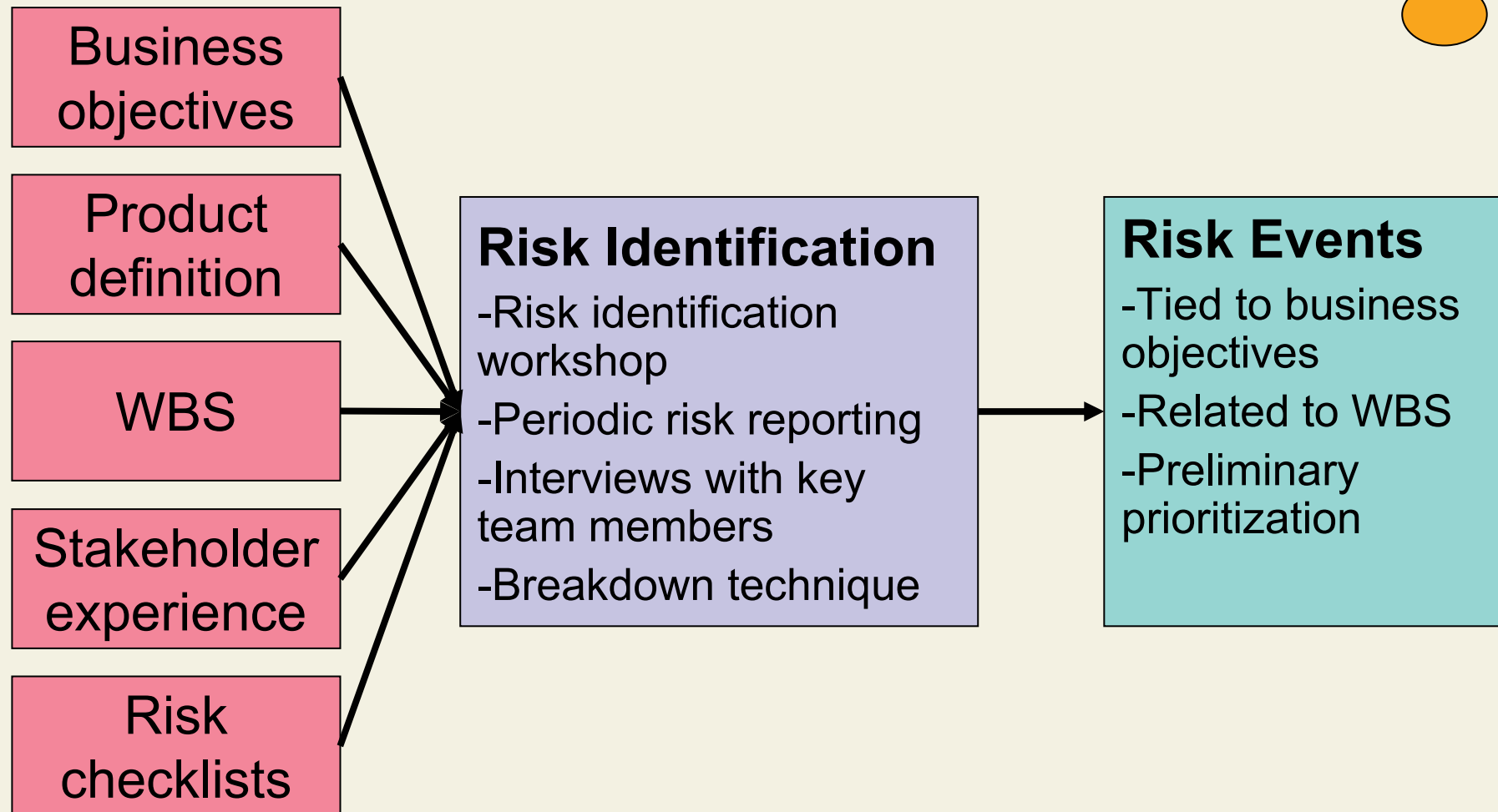
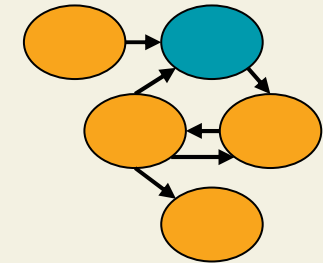


■ Purpose

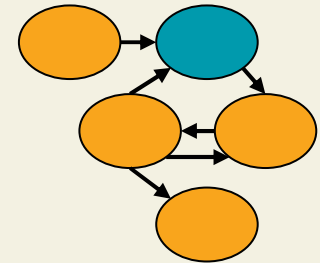
- To decide how to approach and plan the risk management activities for a project

Inputs	Tools & Techniques	Outputs
<ol style="list-style-type: none">1. Organization's risk mgmt. policies2. Defined roles and responsibilities3. Stakeholder risk tolerance4. WBS	<ol style="list-style-type: none">1. Planning meetings	<ol style="list-style-type: none">1. Methodology2. Timing3. Scoring and interpretation4. Thresholds5. Reporting formats

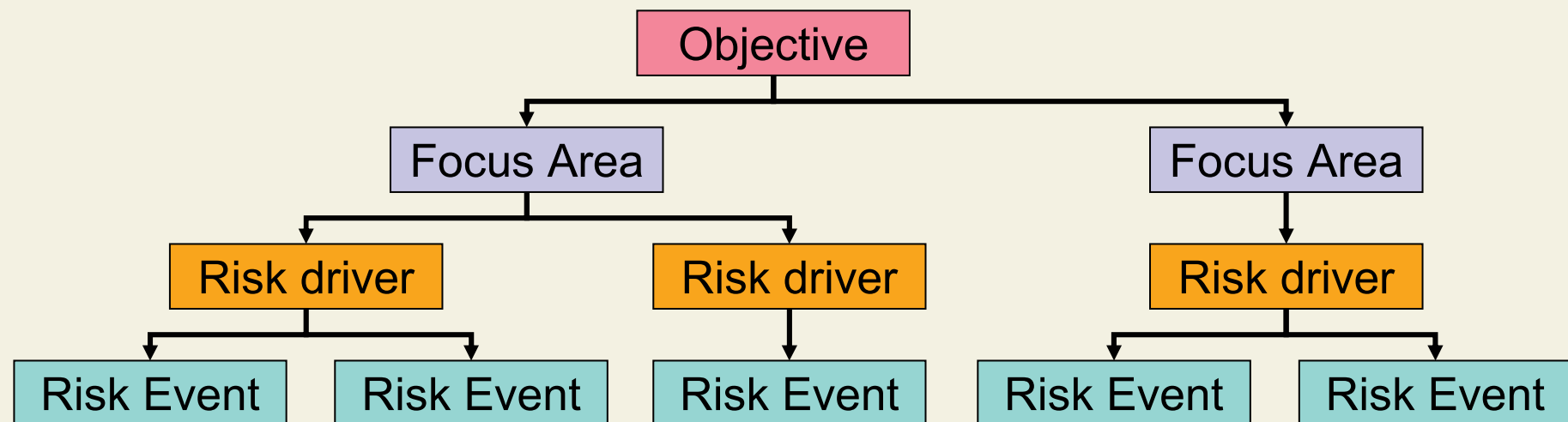
Risk Identification



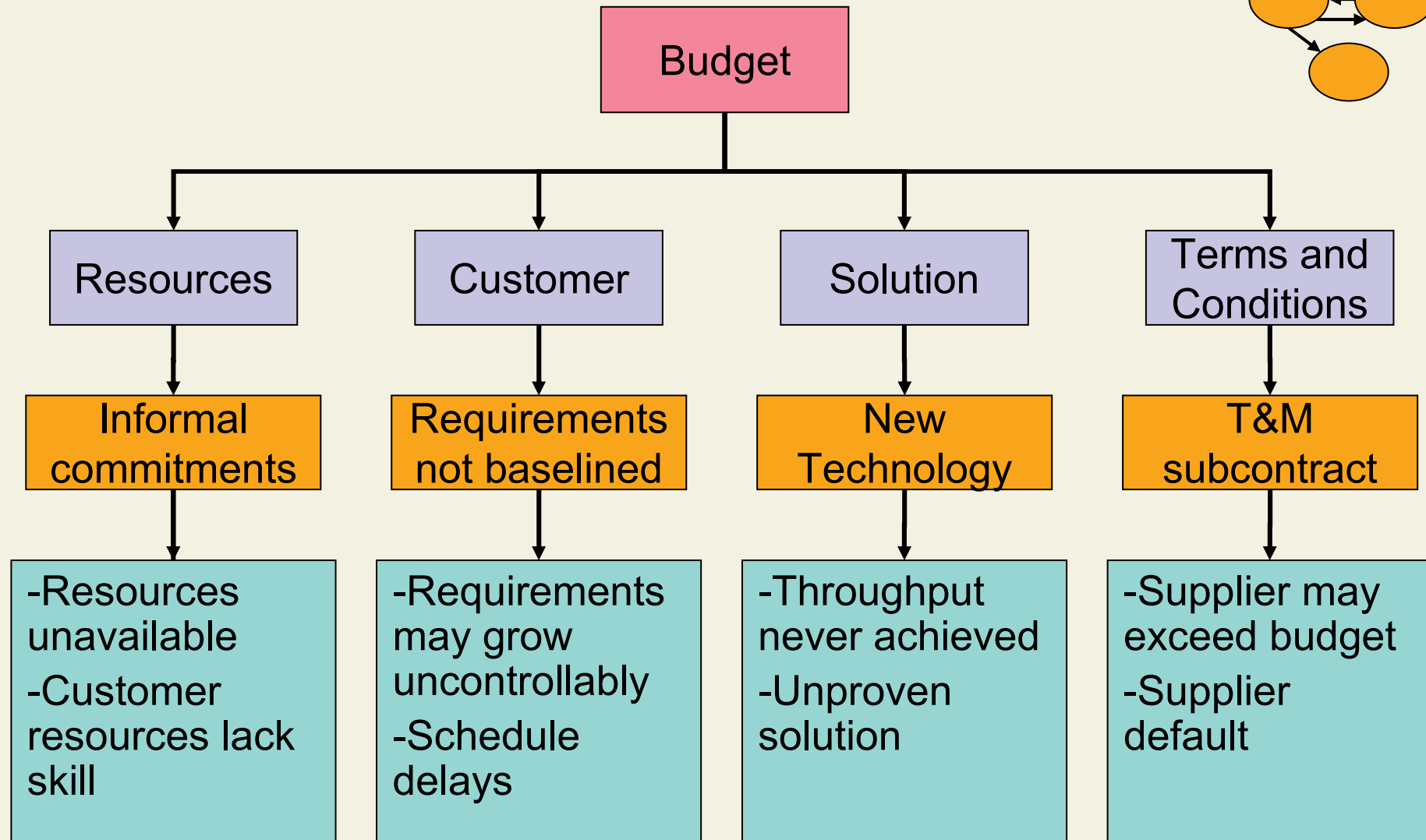
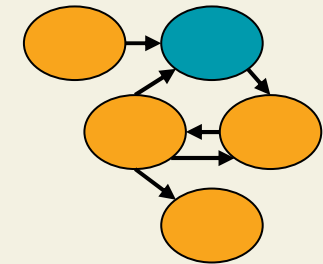
Breakdown Technique



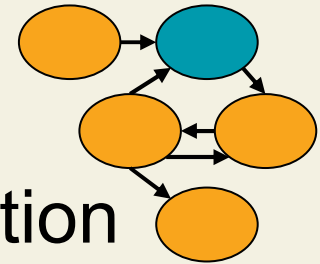
- Identify risks systematically
 - Project objectives: Win, Budget, Satisfy
 - Focus area: A breakdown of the project's potential sources of risk
 - Risk driver: A condition that increases the probability that a risk event will be present



Breakdown Example



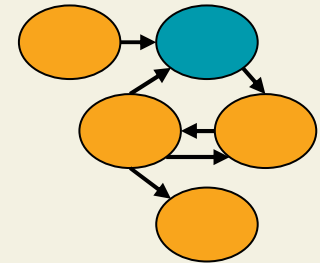
Risk Analysis



- Often called risk assessment or risk evaluation
- Determine
 - **Probability** of the risk to occur
 - **Impact** on the project objectives in case the risk occurs
 - **Severity** (Severity = Probability x Impact)
- Identify risks to be mitigated

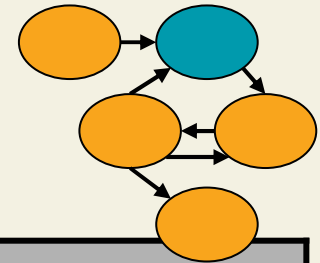
- Qualitative analysis
- Quantitative analysis
 - Based on estimates and simulations

Probability Criteria



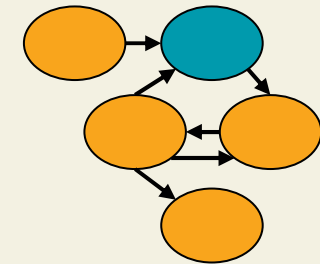
Qualitative Rating	Quantitative Rating	Description
Very High	>84%	Almost assured to happen
High	60-84%	Likely to happen
Medium	35-59%	Somewhat likely to happen
Low	10-34%	Not very likely to happen

Impact Criteria



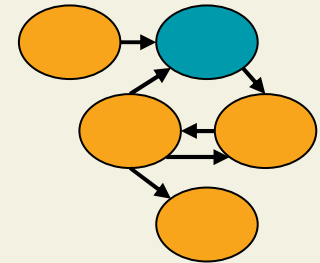
Risk Rating	Description
Very High	Has potential to cause cancellation of the project
High	Likely to cause significant serious disruption to schedule, increase in cost, or degradation of performance
Medium	Has potential to cause some disruption to schedule, increase in cost, or degradation of performance
Low	Has little potential to cause some disruption to schedule, increase in cost, or degradation of performance

Severity of Individual Risks



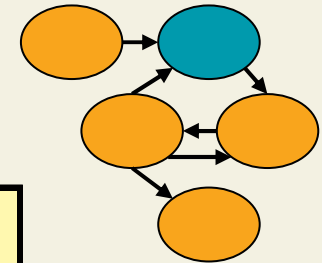
Severity		Impact			
		Very High	High	Medium	Very High
Probability	Very High	Unacceptable	Very High	High	High
	High	Very High	High	High	Medium
	Medium	High	High	Medium	Medium
	Low	High	Medium	Medium	Low

Risk Ranking



- Prioritize risk according to
 - Severity
 - Timing
 - Time required to mitigate (preliminary estimate)
 - Etc.
- “Top 10” Approach
 - Develop mitigation strategies for top 10 risks
 - Use the top 10 as an agenda item for regular project meetings

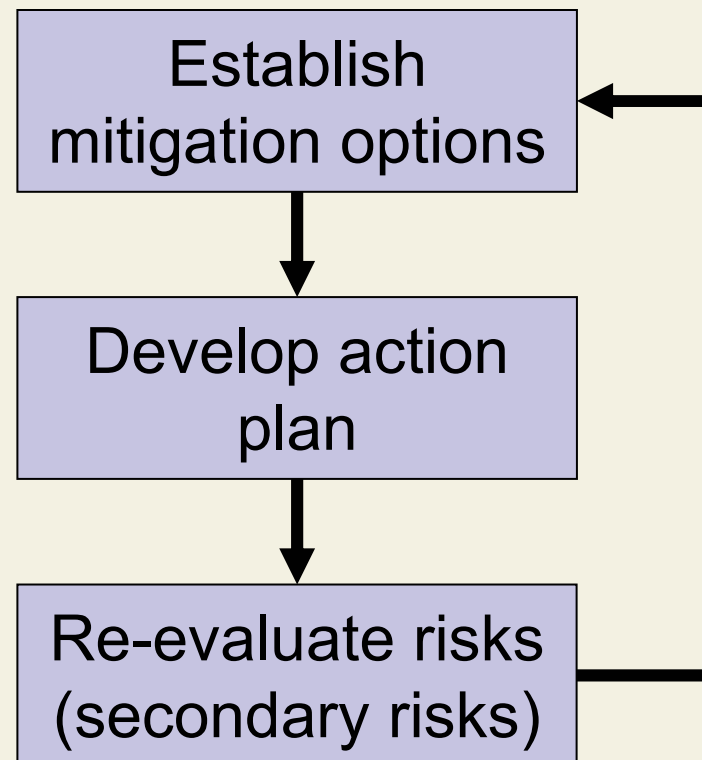
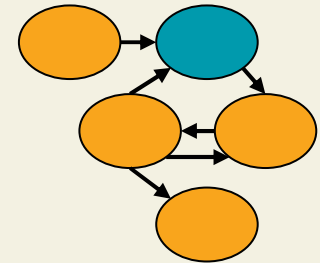
Risk Documentation



Risk ID	WBS Number	Risk Event	Owner	Area of Impact (W/B/S)
1	2.04.05	Requirements will grow uncontrollably	PM	B/S

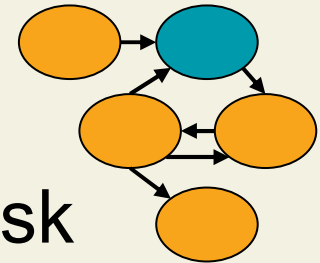
Probable Impact Date	Risk Probability	Risk Impact	Severity	Rank
20.07.2004	High	Very high	Very high	1

Risk Response Planning



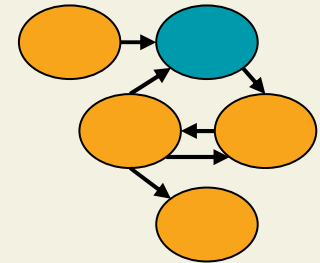
- Risk Response Planning is often called **risk mitigation**

Mitigation Strategies



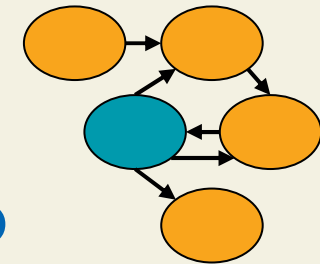
- **Avoid** the path or project to eliminate the risk
- **Ignore / Accept** the risk and its consequences of it occurs
- **Transfer** all or part of the risk to another party
- **Contain** the risk by specific actions to lower the probability and / or impact
- **Establish contingency**: Set funds aside to be used if the risk occurs or when later containment is deemed appropriate

Extended Documentation



Risk ID	Mitigation Strategy	Mitigation Task	Responsible	Status
1	Contain	Use phased approach	PM	in progress

Execution and Post Mitigation

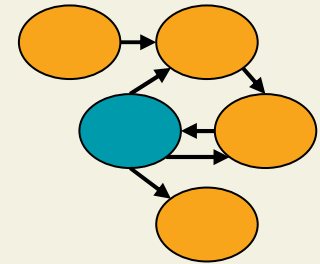
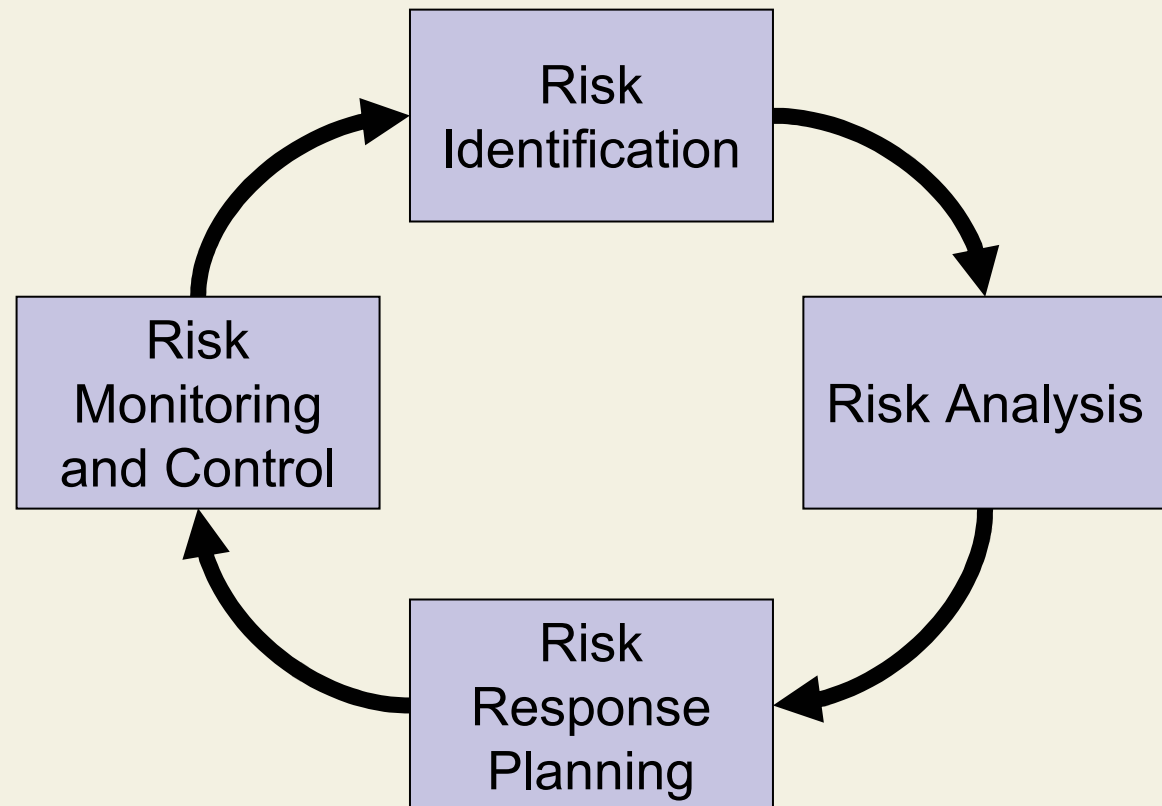


- Mitigation tasks have to be **integrated into project plan** (WBS)
- Execution has to be **closely monitored**
- Risk has to be **re-assessed** to check whether mitigation is successful

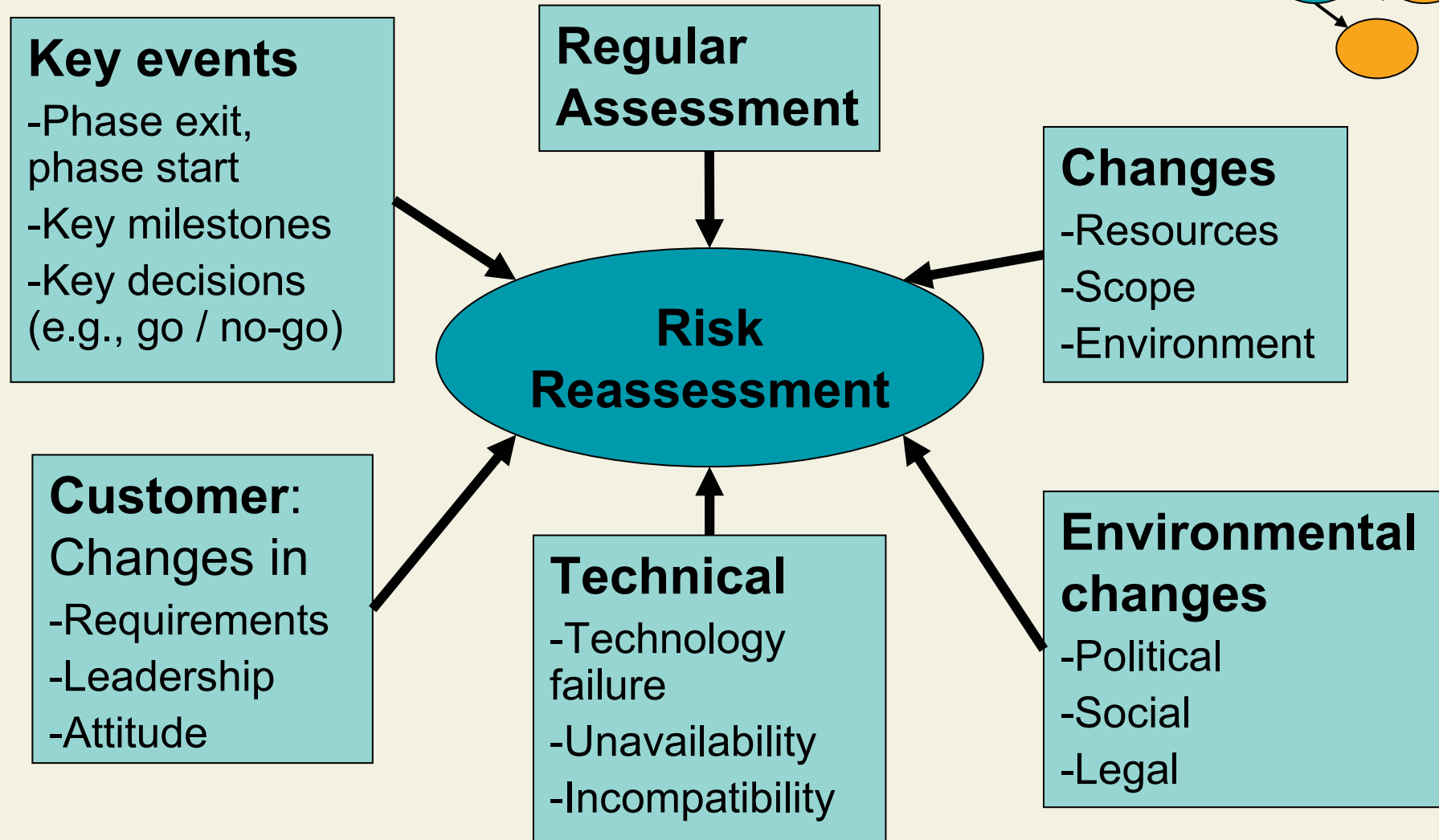
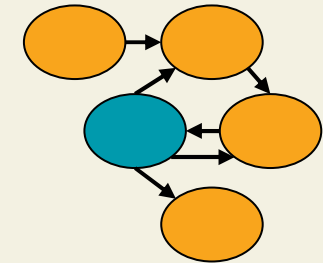
Severity		Impact			
		Very High	High	Medium	Very High
Probability	Very High	Very High	Very High	High	High
	High	Very High	High	High	Medium
	Medium	High	High	Medium	Medium
	Low	High	Medium	Medium	Low

Risk Monitoring

- **Implement, track, and reassess** mitigation strategies
- **Communicate** risk plan status to stakeholders
- **Update** documents



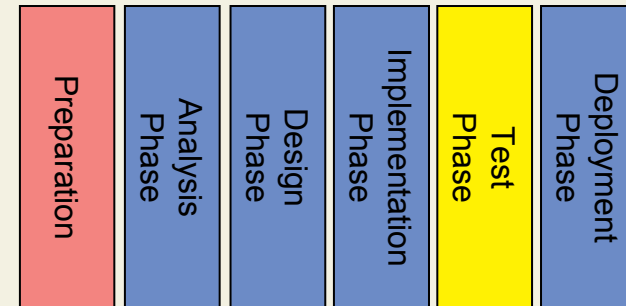
Triggers for Risk Reassessment



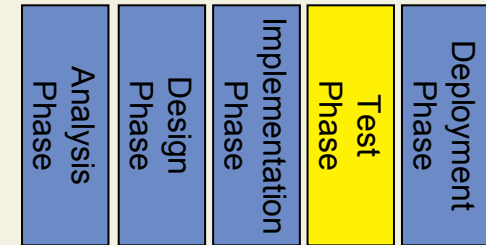
10. Risk Management and Test Phase

10.1 Risk Management

10.2 Test Phase

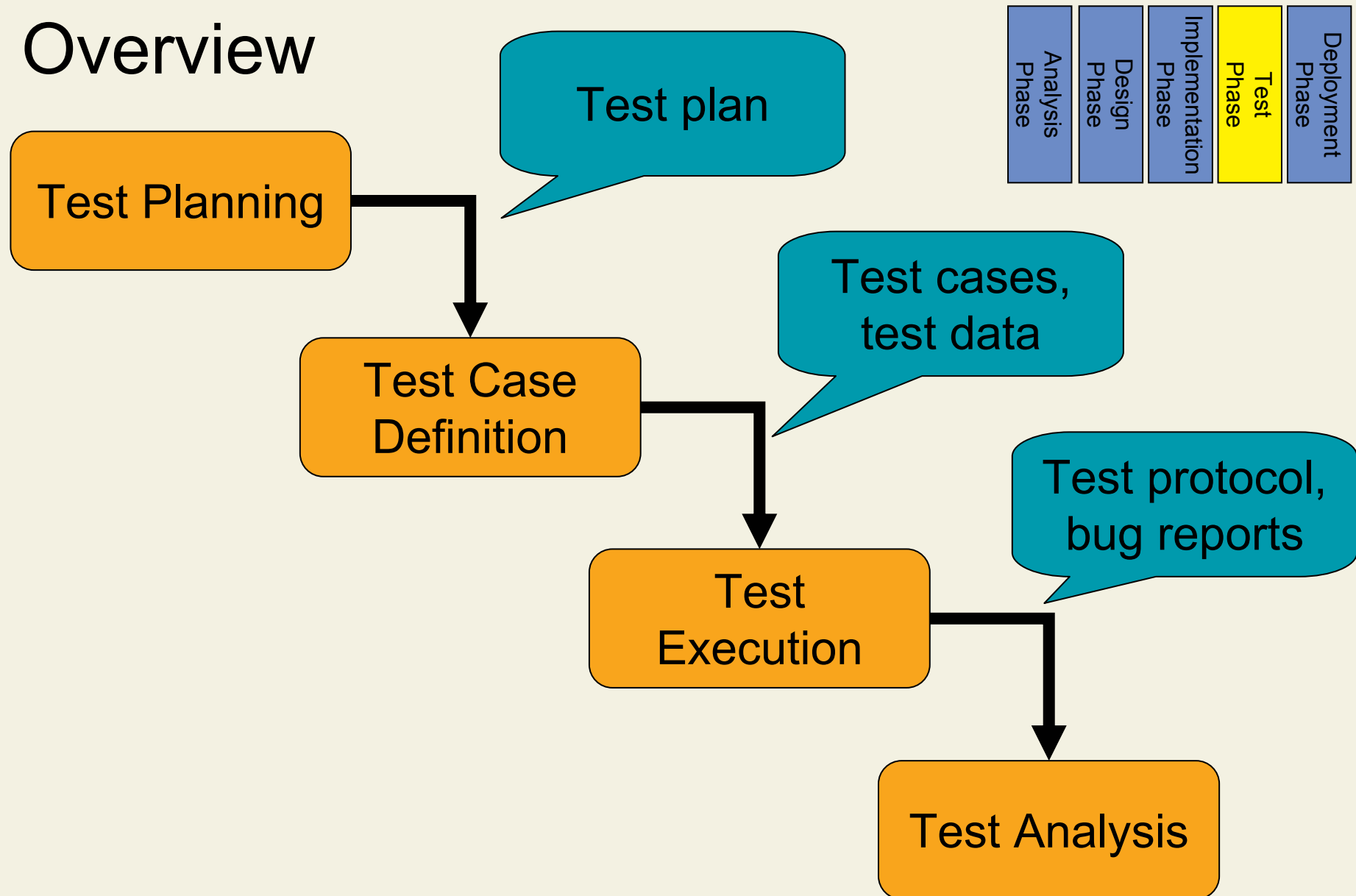


Objectives

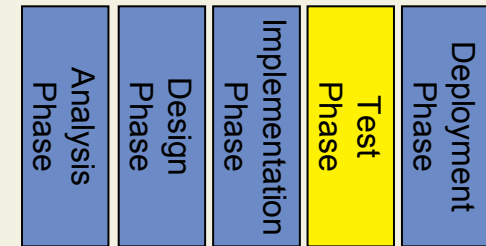


- To dynamically verify the behavior of a software system on a finite set of test cases against the specified expected behavior
- Effectiveness
 - Check as many different executions of a program as possible with the available effort
- Efficiency
 - Suitably select test cases from the usually infinite executions domain
- Reproducibility
 - Detected bugs must be reproducible to enable fixing
 - Prerequisite for regression test

Overview

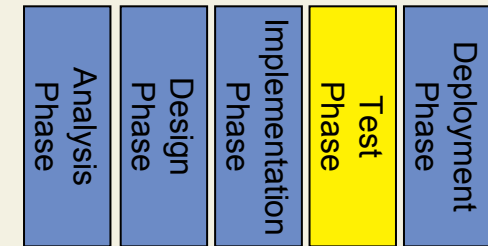


Test Planning



- Test subject
 - Components, interfaces, GUIs, etc.
- Criteria
 - Functionality, performance, load, usability, etc.
- Test strategy
 - Black box test (functional test): Test cases are developed from the functional requirements and do not consider program structure
 - White box test: Test cases take program structure into account (e.g., invoke each method at least once)

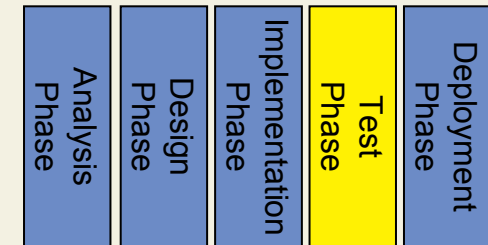
Test Case Definition



- Test goal
 - Example: No bugs at 90% coverage
- Test environment: Software and hardware
- Test cases
 - Test data and action performed on the data
 - Expected results and checks
- Test data
 - Manual development of test data
 - Subsets of existing production data
 - Automatic generation of test data
- Schedule for test execution

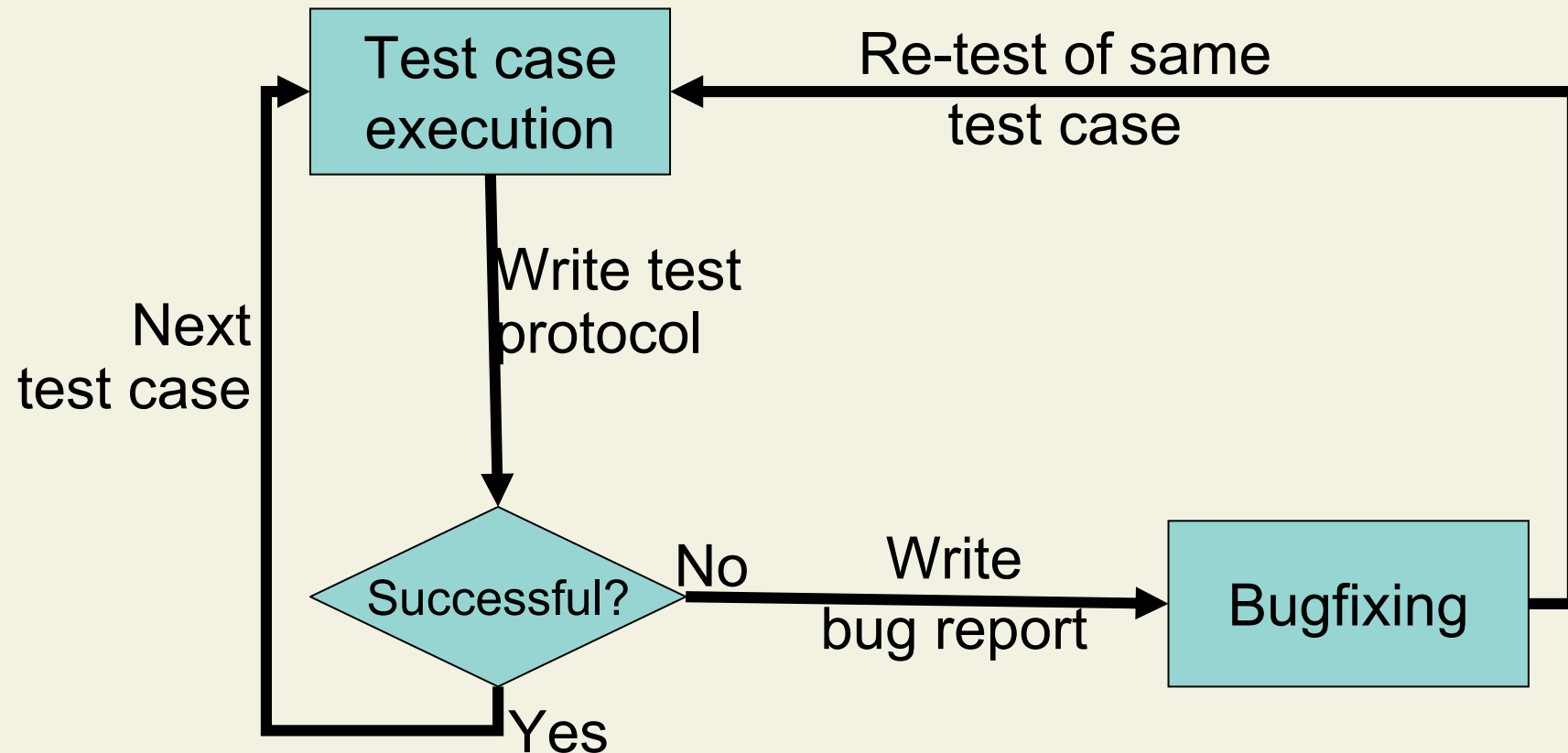
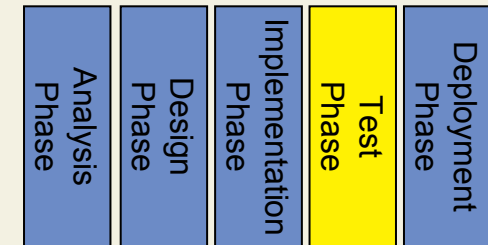


Test Case Selection



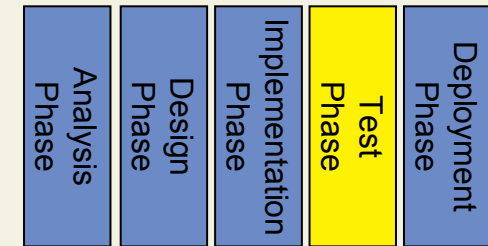
- Test cases can be
 - Defined manually
 - Generated from specifications
 - Randomized data
- Tests have to include
 - positive test cases (normal functionality)
 - negative test cases (error cases)
- Test cases usually represent an equivalence class
 - All cases of one equivalence class are supposed to behave alike
 - Example: One test case “book domestic flight”

Test Execution



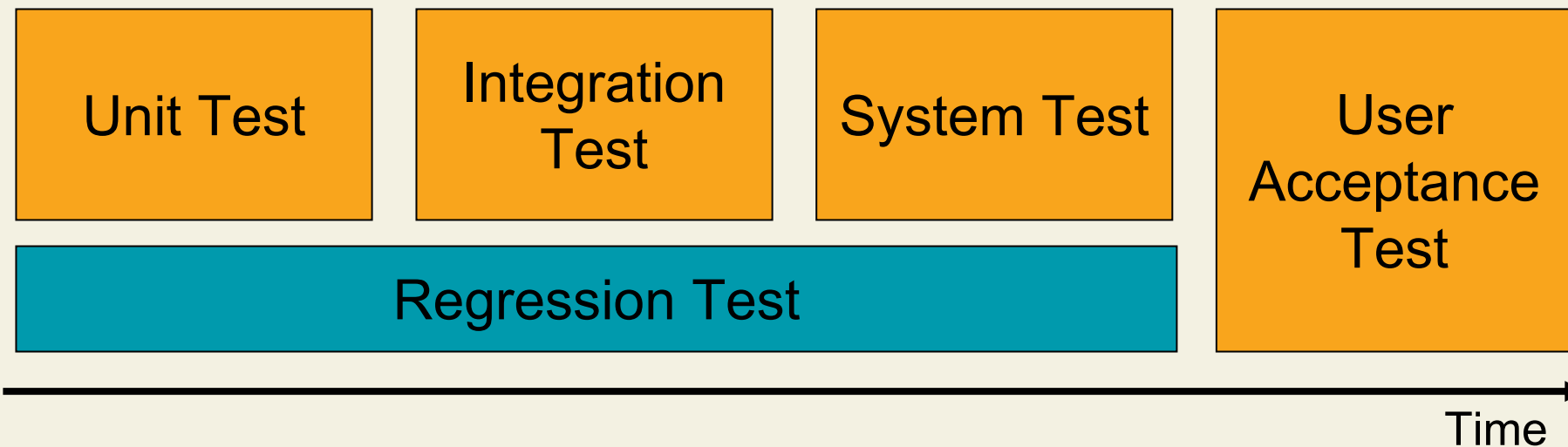
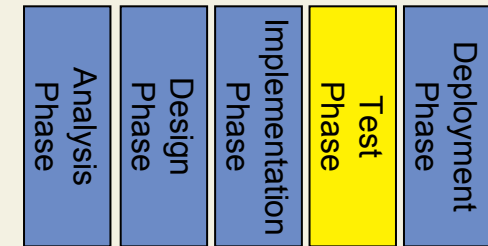
Repeat until test goal is reached

Roles and Responsibilities

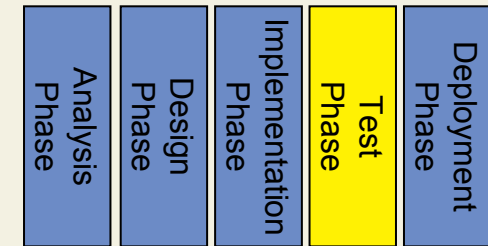


- Developer and tester have to be different persons!
- Developer is biased (often unconsciously)
 - Tendency to avoid most critical test cases
- Motivation
 - Developers aim showing that their program works
 - Testers aim at finding bugs
- Different persons reduce the risk of misunderstood requirements
- Exception: Unit test

Test Stages

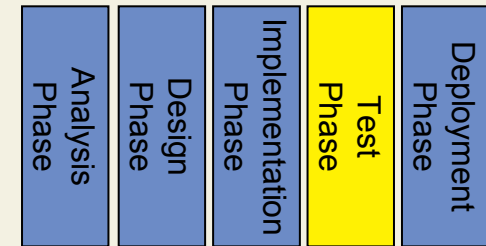


Unit Test



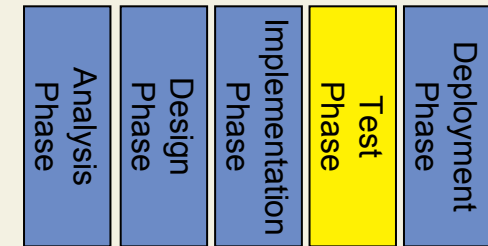
- Purpose
 - Test of individual components, independently of their environment
- Approach
 - White box test
 - Usually performed on development environment
 - Performed by developers
- Often involves development of stubs to simulate peer components
- Typical bugs
 - Logical errors (wrong branch conditions, etc.)
 - Missing cases (special cases)

Integration Test



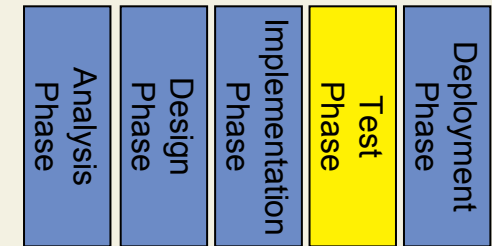
- Purpose
 - Test of assembly of all components of a system
- Approach
 - White box test
 - Usually performed on a designated test environment
 - Performed by testers
- Components should be integrated and tested incrementally
 - Big bang approach makes bug tracking difficult
- Typical bugs
 - Errors in interfaces between components (format and protocol)

System Test



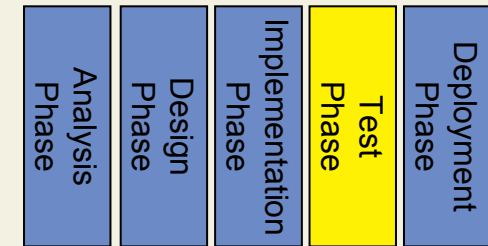
- Purpose
 - Test of external behavior of system
- Approach
 - Black box test
 - Performed on a designated test environment
 - Performed by testers
- Involves in particular
 - Load tests, robustness test, interoperability tests, etc.
- Typical bugs
 - Errors in external interfaces

Regression Test



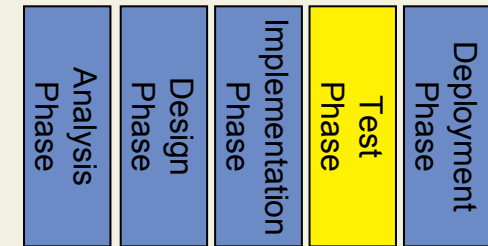
- Purpose
 - Re-run of certain test cases
- Performed by developers or testers
- Regression test is necessary
 - To check that behavior of the system is not changed by adaptations
 - To check that fixing one bug does not bring up others
- Typical bugs
 - All of the above
 - Side effects of new functionality on existing functionality

Acceptance Test



- Purpose
 - Test whether system meets business requirements
- Approach
 - Black box test
 - Alpha test: Test in environment of the developer
 - Beta test: Test in environment of the customer
 - Performed by customers
- Basis for acceptance of the product and formal closing of the project
- Typical bugs
 - Misinterpretation of requirements

Test Phase: Summary



- Purpose
 - To dynamically verify the behavior of a software system on a finite set of test cases against the specified expected behavior
- Main Deliverables
 - Test cases, test protocols, bug fixes
- Main actors
 - Testers, developers, customers
- Tools and techniques
 - Black box test, white box test
 - Test case database, bug tracking tool

	Initiating	Planning	Executing	Controlling	Closing
Integration		Project Plan Dev.	Project Plan Execution	Integr. Change Ctrl	
Scope	Initiation	Scope Planning Scope Definition		Scope Change Ctrl	
Time		Act. Definition, Act. Sequencing, Schedule Dev.			
Cost		Resource Planning Cost Estimating Cost Budgeting			
HR		Organizational Pl. Staff Acquisition	Team Development		
Comm.				Reporting	
Risk		Identification Analysis Response Planning		Monitoring and Control	
Procurement		Procurement Pl. Solicitation Pl.	Solicitation Source Sel. Contr. Admin.		Contract Closeout