Project: complex, non-routine, one-time effort limited by time, budget, resources, perf specs designed to meet customer needs
Project characteristics: established objective, defined life span with beginning & end, requires participation across the org, typically involves doing sth never done before, has specific time, cost, perf reqs
Project life cycle:
Defining: goals, specs, tasks, responsibilities
Planning: schedules, budgets, resources, risks, staffing
Executing: status reports, changes, quality, forecasts
Closing: train customer, transfer docs, release resources, evaluation, lessons learned
Factors leading to increased use of PM: compression of product life cycle, knowledge explosion, triple bottom line (planet, people, profit), corporate downsizing, increased customer focus, small projects represent big problems
Project Manager: manages temporary, non-repetitive activities, frequently acts independently of formal org, marshals resources for project, linked directly to customer interface, provides direction, coordination & integration to the project team, responsible for performance & success of the project, must induce right people at right time to address the right issues and make right decisions
PM Technical Aspects: scope, WBS, schedules, resource allocation, baseline budgets, status reports
PM Sociocultural Aspects: leadership, problem solving, teamwork, negotiation, politics, customer expectations
Integrated management of projects: strategic alignment, portfolio management, PM, with org culture env wrapped around
Integration of projects with org strategy: use of selection criteria to ensure strategic alignment and project priorities (effective use of org resources); selection process that is systematic, open, consistent & balanced; all selected projects become part of portfolio that balances risk for org; portfolio management ensures only most valuable projects approved & managed across entire org; value of project not only ROI but also strategic fit & best use of org resources
Integrative PM approach benefits: provide senior management with overview of all PM activities, big picure of how org resources used, risk assessment of project portfolio, rough metric of org's improvement in managing projects relative to others in industry, linkages of senior management with actual project execution management
Portfolio Management Functions: oversee project selection, monitor aggregate resource levels & skills, encourage use of best practices, balance projects in portfolio in order to represent risk level appropriate to the organisation, improve communication among all stakeholders, create total ord perspective that goes beyond silo thinking, improve overall management of projects over time
Program: a series of coordinated, related, multiple projects that continue over an extended time and are intended to achieve a goal
Traditional PM: focus on thorough, upfront planning of entire project, requires high degree of predictability to be effective
Agile: relies on incremental, iterative dev cycles to complete less predictable projects, ideal for exploratory projects in which requirements need to be discovered and new tech tested (uncertain ablut how long, what is required, allows change in reqs), focus on active collaboration between project tem & customer reps

Traditional	**Agile**
Design up front	Continuous design
Fixed scope	Flexible
Deliverables	Features/reqs
Freeze design as early as possible	as late as possible
Low uncertainty	high
Avoid change	embrace
Low customer interaction	high
Conventional project teams	self-organised

Agile Details: use iterations to develop workable product that satisfies the customer and other key stakeholders, stakeholders & customers review progress & re-evaluate priorities to ensure alignment with customer needs & company goals, adjustments are made & a different iterative cycle begins that subsumes the work of the previous iterations & adds new capabilities to the evolving product
Agile Advantages: useful in developing critical breakthrough tech or defining essential features; continuous integration, verification & validation of the evolving product; frequent demonstration of progress to increase likelihood that end product will satisfy customer needs; early detection of defects & problems
Agile Limitations: does not satisfy top management's need for budget, scope & schedule control; self-organisation & close collaboration principles can be incompatible with corporate cultures; appears to work best on small project with 5–9 people, requires active customer involvement & cooperation
Agile Principles: focus on customer value, iterative & incremental delivery, experimentation & adaptation, self-organisation, continuous improvement
Project uncertainty dimensions: scope & tech **Scrum**: holistic (interconnected emphasis) approach for use by cross-functional team collaborating to develop new product, defines product features as deliverables & prioritises them by perceived highest value to the customer, re-evaluates priorities after each iteration/sprint to produce fully functional features, phases: analysis, design, build, test
Scrum roles & responsibilities:
Product owner: acts on behalf of customer to represent interests, responsible for product backlog priorities & process selection
Development team: 5–9 people with cross-functional skillsets responsible for delivering product, sets own goals, organises itself, makes decisions
Scrum master: facilitates scrum process and resolves impediments at the team & org level by acting as buffer between team & outside interference
Scrum practices:
Sprint: time-controlled mini-project that implements specific portion of a system, 30 day time box with specific goals & deliverables, frozen scope defined from sprint backlog
Daily Scrum: daily meeting of all team members to report progress (15 min max), also called standup
Sprint final half-day review meeting: review & identify changes needed for following sprints
Sprint meetings: sprint planning, daily scrum, sprint review, sprint retrospective
Product backlog: customer's prioritised list of desired key features for the completed project, can only be changed by product owner
Sprint backlog: amount of work team commits to complete during the next sprint, developed & controlled by team
Scaling: using several teams to work on different features of large scale project at same time
Staging: upfront planning to manage interdependencies of the different features to develop, involves developing protocols & defining roles to coordinate efforts & assure compatibility & harmony
Strategy Importance: Project managers must respond to changes to organisation mission and strategy appropriately, if understand strategy can become effective advocates of projects aligned with firm's mission
Mistakes by not understanding role of projects in accomplishing strategy: focus on problems/solutions with low strategic priority, focus on immediate customer rather than whole marketplace & value chain, over-emphasising tech that results in projects that pursue exotic tech that does not fit strategy or customer need, trying to solve customer issues with product/service rather than focusing on 20% with 80% of value (**Pareto's law**), engaging in never-eding search of perfection that only team really cares about
Strategic management: requires every project to be clearly linked to strategy; provides theme & focus of organisational future direction (responding to changes in external env — env scanning, allocating scarce resources of firm to improve competitive position — internal responses to new programs); requires strong links among mission, goals, objectives, strategy, impl
Strategic management activities:
Review & define org mission: identify & communicate purpose of org to stakeholders, identify scope of org in terms of product/service, provides focus for decision making, used for eval org perf
Set long-range goals & objectives: translate mission to specific, concrete & measurable terms; sets targets for all levels of org in a cascaded manner; where is org headed and when it will get there; focus managers on where org should move to
Analyse & formulate strategies to reach objectives: focus on what needs to be done to reach objectives, relaistic view of past & current position, SWOT analysis, alternatives generated & assessed, strategy formulation & assignation
Implement strategies through projects: focus on how strategies will be realised with resources, maintain link between strategy (what) & impl (how), requires resource allocation, action & completion of tasks, prioritisation
SMART objectives: Specific, Measurable (indicators of progress), Assignable (to one person for completion), Realistic (what can realistically be done with avail resources), Time related (state when objective can be achieved)
SWOT analysis: internal (strengths, weaknesses) & external (opportunities, threats) analysis
Scenario planning: longer term, steps: clarifying core business & assessing drivers of change in industry env, dev potential scenatios & assess impact of STEEP factors, dev potential contingency strategies & best future strategic options, identifying early indicators & establishing triggers for strategic action
STEEP factors: social, tech, env, economic, political
Project portfolio management benefits: build discipline to project selection process, link project selection to strategic metrics, prioritise project proposal across common set of criteria rather than politics/emotion, allocate resources to projects that align with strategic direction, balance risk across all projects, justifies stopping projects that don't support strategy, improves comms & supports agreement on project goals
Project portfolio management problems:
Implementation gap: lack of understanding & consensus on strategy among to management & middle-level (functional) managers who independently implement strategy
Org politics: project selection based on persuasiveness & power of people advocating projects
Resource conflicts & multitasking: Multiproject env creates interdependency relationships of shared resources which results in starting, stopping & restarting of projects
Project portfolio sys design: classification of project, selection criteria depending upon classification, sources of proposals, evaluating proposals, ranking proposals, managing portfolio of projects
Project types: compliance (must-do, incl emergency, meet regulations, usually have penalties if not impl), strategic (directly support long-run mission, increase revenue/market share, ex: new products, R&D), operational (support current ops, improve perf, reduce product cost, improve efficiency of delivery sys, ex: upgrade building green rating)
Financial Selection Criteria: payback, NPV, IRR (internal rate of return, inverse of payback)
Payback model: measures time project takes to recover investment; uses more desirable, shorter paybacks; emphasises cash flows (key factor in business)
Payback limitations: ignores time value of money, assumes casf inflow only for investment period, does not consider profitability
Net Present Value: $I_0 + \sum_{t=1}^{n} \frac{F_t}{(1+k)^t}$, I_0 is initial investment (negative), F_t is net cash inflow for period t, k is required rate of return, want positive
Non-financial strategic criteria: capture larger market share, make it difficult for competitors to enter the market, develop enabler product which by interduction will increase sales in more profitable products, develop core tech to be used in next-gen products, reduce dependency on unreliable suppliers, prevent government intervention & regulation
Multicriteria selection models:
Checklist model: use list of questions to review potential projects & to determine accept/reject, fails to answer relative importance/value of potential project & doesn't allow for comparison with others
Multiweighted scoring model: use several weighted qualitative and/or quantitative selection criteria to evaluate project proposals, can use for comparison
Selection model advantages: bring projects to closer alignment with org strategic goals, reduce number of wasteful projects, help identify proper goals for projects, help everyone involved understand how & why project is selected
Project relativity matrix: 2 dimensions (technical feasibility, NPV), white elephant (low, low, showed promise at one time but are no longer viable), oyster (low, high, technological breakthroughs with high commercial payoffs), bread-and-butter (high, low, evolutionary improvements to current products & services), pearl (high, high, revolutionary commercial opportunities using proven tech advances)
Challenges to organising projects: need to balance needs of project with org, uniqueness & short duration of projects relative to ongoing longer term org activities, multidisciplinary & cross-functional nature of projects creates authority & responsibility dilemmas
Functional org: different segments of project delegated to functional units, coordination maintained through normal management channels, used when interest of 1 functional area dominates project or has dominant interest in project success
Functional +: no structural change, flexibility, in-depth expertise, easy post-project transition
Functional -: lack of focus, poor integration, slow, lack of ownership
Dedicated project teams: teams operate as separate units under leadership of full-time project manager, in projectised org where projects are dominant form of business functional depts are responsible for providing support to teams
Dedicated +: simple, fast, cohesive, cross-functional integration
Dedicated -: expensive, internal strife, limited tech expertise, difficult post-project transition
Hybrid/Matrix: overlaid on normal functional structure, 2 chains of command (functional & project), project participants report simultaneously to both functional & project managers, optimise use of resource (allows pariticipation on multiple projects while performing normal functional duties)
Matrix +: efficient, strong project focus, flexible, easy post-project transition
Matrix -: dysfunctional conflict, infighting, slow, stressful
Weak matrix: authority of functional manager predominates, project manager has indirect authority
Balanced matrix: the project manager sets overall plan & the functional manager determines how work is to be done
Strong matrix: project manager has broader control, functional departments act as subcontractors to project
Matrix division of responsibilities:
Project manager: what has to be done, when should the task be done, how much money is available to do the task, how well has the total project been done
Functional manager: how will it be done, how will project involvement impact normal functional activities, how well has the functiona input been integrated
Negotiated issues: who will do the task, where will the task be done, why will the task be done, is the task satisfactorily completed
Choosing the appropriate project management structure:
Organisational considerations: how important is the project to the firm's success, what percentage of core work involves projects, what level od resources (human & physical) are available
Project considerations: size of project, strategic importance, novelty & need for innovation, need for integration (number of depts involved), environmental complexity (number of external interfaces), budget & time constraints, stability of resource reqs
Org culture: system of shared norms, beliefs, values & assumptions that bind people together, thereby creating shared meanings; personality of org that sets it apart from other orgs
Org culture benefits: provides sense of identity to members, helps legitimise management system of org, clarifies & reinforces standards of behaviour, helps create social order
Diagnosing org culture: study physical characteristics (architecture, office layout, decor, attire), read about org (annual reports, internal newsletters, vision statements), observe how people interact within org (pace, lang, meetings, issues discussed, decision-making style, comm patterns, rituals), interpret stories & folklore surrounding org (anecdotes, heroines, heroes, villains)
Org culture dimensions: member identity (job, org), team emphasis (individual, group), management focus (task, people), unit integration (independent, interdependent), control (loose, tight), risk tolderance (low, high), reward criteria (performance, other), conflict tolerance (low, high), means-ends orientation (means, ends), open-system focus (internal, external, degree to which org monitors & responds to changes in external env)
Culture challenges for structuring projects: interacting with culture & subcultures of parent org, interacting with project clients or customer orgs, interacting with other orgs connected to project
Mechanisms for sustaining org culture: formal statement of principles, top management behaviour, reactions to org crises, allocation of rewards & status, rituals, stories, symbols
Defining the project: defining project scope, establishing project priorities, creating WBS, integrating WBS with org, coding WBS for information sys
Project scope: definition of end result or mission of project — a product/service for clinet/customer — in specific, tangible & measurable terms
Scope statement: statement of work (SOW)
Scope statement purpose: clearly define deliverables for end user, focus project on successful completion of its goals, to be used by project owner & participants as planning tool & measuring project success
Project scope checklist: project objective; deliverables; milestones; technical reqs; limits & exclusions; reviews with customer
Project charter: can contain expanded version of scope statement, document authorising project manager to initiate & lead project
Scope creep: tendency for project scope to expand over time due to changing requirements, specs, priorities
Priority matrix: budget/cost, schedule/time, performance/scope, constrain, enhance (optimise), accept
Work Breakdown Structure: hierarchical outline (map) that identifies products & work elements involved in project, defines relationship of final deliverable to subdeliverables & in turn their relationships to work packages, best suited for design & build projects that have tangible outcomes rather than process-oriented projects
WBS Hierarchy: project, deliverable, sub-deliverable, lowest sub-deliverable (lowest management responsibility level), cost account (group of work packages for monitoring progress & responsibility), work package
WBS benefits for project manager: facilitates evaluation of cost, time & technical perf of org on project; provides management with info appropriate to each org level; helps in dev of OBS, which assigns project responsibilities to org units & individuals; help manage plan, schedule & budget; define comm channels & assists in coordinating various project elements
Work Package: defines work (what), identifies time to complete, time-phased budget to complete (cost), resources needed to complete (how much), person responsible for units of work, monitoring points/milestones for measuring success (how well)
Org Breakdown Structure: how company organised to discharge work responsibility for project
OBS details: provides framework to summarise org work unit perf, identifies org units responsible for work packages, ties org units to cost control accounts
Intersection of WBS & OBS: project control point/cost account
WBS coding system: defines levels & elements of WBS, org elements, work packages, budget & cost info, allows reports to be consolidated at any level in org structure
Responsibility Matrix: linear responsibility chart, summarises tasks to be accomplished & who is responsible for what on the project
RM details: list project activities & participants, clarifies critical interfaces between units & individuals that need coordination, provides means for all participants to view responsibilities & agree on assignments, clarifies extent/type of authority that can be exercised by each participant
Project communication plan: what info needs to be collected & when, who will receive info, what methods will be used to gather & store info, what are limits on who as access to certain kinds of info, when will info be communicated, how will it be communicated
Comm plan steps: stakeholder analysis, info needs, soruces of info, dissemination modes, responsibility & timing
Information needs: project status reports, deliverable issues, changes in scope, team status meetings, getting decisions, accepted request changes, action items, milestone reports
Estimating: process of forecastine/approximating time & cost of completing project deliverables, task of balancing exepctations of stakeholders& need for control while project is implemented
Estimating importance: support good decisions, schedule work, determine how long project should take & cost, determine whether project worth doing, develop cash flow needs, determine how well the project is progressing, develop time=phased budgets & establish project baseline
Estimation accuracy factors: planning horizon, project duration, people, project structure & org, padding estimates, org culture, other non-project factors
Estimating guidelines: have people familiar with tasks make estimate; use several people to make estimates; base estimates on normal conditions, efficient methods & normal level of resources; use consistent time units; treat each task as independent; don't make allowances for contingencies, adding risk assessment helps avoid surprises to stakeholders
Top-down estimates: derived from someone who uses experience and/or info to determine the project duration & total cost, are made by top managers who have little knowledge of the processes used to complete the project, time & costs are not considered, grouping tasks may lead to omissions & unrealistic times & costs, accuracy -20% to +60%, cost 0.1–0.3%
Conditions for top-down: strategic decision making; high uncertainty; internal, small project, unstable scope
Top-down intended use: feasibility/conceptual phase, rough time/cost estimate, fund reqs, resource capacity planning
Bottom-up approach: can serve as a check on cost elements in WBS by rolling up work packages & associated cost accounts to major deliverables at work package level, more accurate but takes more time, accuracy level may not be required for some projects, accuracy -10% to +30%, cost 0.3–1%

Conditions for bottom-up: cost & time important, fixed-price contract, customer wants details
Bottom-up intended use: budgeting, scheduling, resource reqs, fund timing
Preferred estimating approach: rough top-down estimates, dev WBS/OBS, make bottom-up estimates, dev schedules & budgets, reconcile diffs between top-down & bottom-up estimates
Top-down approaches:
Consensus: use xp of senior and/or mid managers to estimate total project duration & cost; typically involve meeting where experts discuss, argue & ultimately reach decision for best guessestimate
Delphi: about likelihood that certain events will occur, ask experts, then return summary of opinions (anon), encourage to reconsider/change based on others' opinions, repeat 2–3x, median will move toward 'correct' estimate, avoid ego, domineering, bandwagon, halo effect, no need for physical contact
Ratio: use cost/time per area/capacity size/features/complexity
Apportion: extension of ratio, use if projects closely follow past projects in features & cost, pay contractor by completion of parts or split costs based on deliverables in WBS (each has percent allocated)
Function point: for software & system projects, take several elements (input, output, inquiries, files, interfaces), rate complexity, multiply number of each with complexity, total is estimate
Learning curve: take number of units & improvement rate
Bottom-up approaches:
Template: start wirh standard task cost/time estimates then adjust specifics
Parametric applied to specific tasks: need to do X work, 1 person can do Y work in Z time
Range estimates for work packages: low, average, high for each, useful if work packages have significant uncertainty
Phase estimating: hybrid top-down & bottom-up, macro long-term (rest of project) & micro short-term (current phase, need, specs, design, produce, deliver), for projects with high uncertainty, customers may be able to change features & re-evaluate project at each stage, but customers & clients typically want form estimates of time & overall cost up front
Level of detail: varies in WBS with project complexity, each management level can focus on what they need; excessive detail is costly, fosters focus on departmental outcpmes & create unproductive paperwork but insufficient detail is also costly, fosters lack of focus on goals & leads to wasted effort on non-essential activities
Cost types:
Direct: clearly chargeable to specific work package, ex: labour, materials, equipment
Direct (project) overhead: directly tied to identifiable deliverable/work package, ex: salary, rents, supplies, specialised machinery
General & administrative overhead: indirectly linked to specific package apportioned to project, carried for project duration, ex: ads, accounting, senior management
Cost views: committed, scheduled budget, actual cost
Adjusting estimates: done for specific activities as risks, resources & situation particulars become more actively defined, mitigate risks by recognising mistakes can occur (ex: include independent testers to check design), allowing for difficult conditions in contracts affecting scope (if excessive ground water adjust foundation estimates)
Reasons for adjusting estimates: interaction costs hidden in estimates, normal conditions do not apply, things go wrong in projects, changes in project scope & plans
Estimating DB: estimated & actuals on labour, costs, equipment, benchmarking ratios, code of accounts for various project types
Project network: flow chart that graphically depicts sequence, interdependencies, start & finish times of project job plan of activities
Critical path: longest activity paths through network that allows for completion of all activities; shortest expected time in which entire project can be completed; 0 slack, also consider deps caused by resource constraints
Project network benefits: provides basis for scheduling labour & equipment, enhance comms among project participants, provides estimate of project duration, provides basis for budgeting cash flow, highlights 'critical' activities that cannot be delayed, highlights activities that can be compressed to meet deadline, help managers get & stay on plan
Late finish: latest activity can finish & not delay following activity, LS + DUR
Late start: latest activity can start & not delay following activity, LF — DUR
Early finish: earliest an activity can finish if all preceding activities are finished by early finish, ES + DUR
Early start: earliest an activity can start, largest early finish of all predecessors, EF — DUR
Activity: project element that requires time
Merge: activity with 2 or more preceding activities on which it depends
Parallel/concurrent: can occur independently & if desired not at the same time
Path: sequence of connected, dependent activities
Event: point in time when activity started/completed, does not consume time
Burst: activity with more than one activity immediately following it
Rules for dev project network: flow left to right, activity cannot begin until all predecessors complete, arrows on networks indicate precedence & flow, each activity should have unique id number, id must be larger than predecessors, no loops & conditionals, use common start & stop nodes
Total slack: amount of time activity can be delayed & not delay project, time activity can exceed early finish date without affecting project end date/imposed completion date, LS — ES or LF — EF
Sensitivity: likelihood original critical paths will change once project initiated
Free slack (float): amount of time activity can be delayed after start of longer parallel activities, how long activity can exceed EF without affecting ES of successors, allows flexibility in scheduling scarce resources
Laddering: activities broken into segments so following activity can begin sooner & not delay work
Lags: minimum amount of time dependent activity must be delayed to begin/end, lengthy activities broken down to reduce delay in start of successor (if successor only dep on part finished early), with lags start and finish can have different slacks
Lag types:
Finish-to-start: order materials, 1 day to place order & 19 days to receive goods, can use for laddering
Start-to-start: can also be used for laddering, reduce network detail & project delays, often used in concurrent engineering (instead of completely sequential, can start next stage once part of predecessor complete)
Finish-to-finish: test cannot be completed any earlier than 4 days after prototype complete, cannot be f-s because subcomponent test does not qualify as complete sys test, which takes 4 days
Start-to-finish: system doc cannot end until 3 days after test start, since relevant info is generated after 3 days of testing
Hammock activity: spans over segment of project, used to aggregate sections of project to facilitate getting right amount of detail for specific sections of project, used to identify fixed resources/costs over segment of project (inspection services, consultants, construction management)
Risk: uncertain/chance events that planning cannot overcome/control, focus on future, deals with probabilities, tends to emphasise negative consequences
Threat: risk event external to org (inflation, market acceptance, laws), not within project manager/team's responsibility area, normally considered before decision to proceed with project, if project initiated contingency funds placed in management reserve budget
Risk project life cycle: high chance of risk occurring initially (defining & planning) but low cost to fix, both swap by delivering
Risk anatomy: cause (how & why), event (what can go wrong, occurrence, outcome), effects (consequence)
Certainty: knowns, decision-maker aware of alternatives & outcomes
Uncertainty: unknown unknowns, future unknowable to probabilities & consequences unknown
Risk: known unknowns, situation where future can be analysed & planned for
Risk management attributes: is a decision-making process (informs decisions), should have structure & formality (helps effective management), has to have continuity through the project (iterative, continuous monitoring), has a project focus (for project performance & outcomes, such as time, cost, perf)
Risk management benefits: proactive rather than reactive approach, reduce surprises & negative consequences, prepares project manager to take advantage of appropriate risks, provides better control over future, improves chances of reaching project perf objectives within budget & on time
Risk management process:
Planning & context: defines factors (internal/external) to take into account, risk management plan
Risk identification: identify potential risks & causes (list of risks)
Risk analysis & evaluation: analyse risk likelihood & potential consequences, risk evaluation for management
Risk treatment: strategies
Implementation & control: implement, monitor, control, review, happens throughout
Risk management plan: objectives, methodology, roles & responsibilities, budgeting, timing, risk categories, scoring interpretation, tolerance thresholds, reporting formats, tracking
Context types:
External: environment, such as political, social, legal, financial, geographical
Organisational: culture, values, governance, capabilities, policies, processes, strategic objectives
Project: fill set of objectives & project outcomes
Risk Identification Tools: personal xp, individual pondering, group processes, structured interviews, project info, checklists, risk breakdown structure
Risk Identification Method: generate list of possible risks through brainstroming, problem identification & risk profiling, focus on macro risks (affect whole project) then specific events, use core risk team and/or stakeholders, typically occurs in project planning phase, focus on actual events that could produce consequences rather than objectives (instead of 'fail to meet deadline' focus on possible causes)
Risk breakdown structure: split into categories (technical, external, organisational, PM), then split to subcategories
Risk evaluation: need to do anything?, classify (how acceptable), determine risk tolerance, know risk appetite
Risk consequence matrix: effect & probability, contents are how important it is (insignificant almost certain is moderate)
Risk assessment form/matrix: columns are risk events, likelihood, impact, detection difficulty, when
Risk severity matrix: impact & likelihood, with zones (red zone for high likelihood & impact)
Failure Mode & Effects Analysis: add detection to severity matrix, risk value does not differentiate between what part contributes most
Risk treatment options:
Avoidance: change plan to eliminate threat, refuse to accept risk
Reduction/Mitigation: reduce likelihood/consequences pre/post risk, contingency plans
Retention: accept with no further action, often for low risk
Transfer: shift responsibility & consequences to another party (contract/insurance) though it still exists
Contingency plan: alt plan that will be used if possible foreseen risk occurs, plan of actions to reduce/mitigate consequences of risk event, having no plan may slow managerial response, decisions made under pressure can be potentially dangerous & costly
Technical Risks: backup strategies if chosen tech fails, assessing whether tech uncertainties can be resolved
Schedule risks: use of slack increase risk of late project finish, imposed duration dates (absolute project finish date), compression of project schedules due to shortened project duration date (crash or shortening project duration using contingency funds, run activities concurrently/laddered)
Costs risks: costs increase then problem take longer to solve than expected (time/cost dependency links), price protection risks increase for long projects, evaluate item by item for cost sensitive projects
Funding risks: changes in supply of funds can dramatically affect likelihood of implementation/successful completion of project
Risk response matrix: columns are risk event, response, contingency plan, trigger, who is responsible
Principles for selection for treatment:
Practicality: realistic, achievable, easy to implement
Effectiveness: rating comparative effectiveness of options
Acceptability: agreement & commitment of stakeholders
Cost: balancing cost of treatment option against benefit
Capability: effective allocation for responsibility
Timeliness: implemented at the time to be successful
Precautions: need to take action as risk event has serious consequences
Time buffers: amounts of time used to compensate for unplanned delays in project schedule, allocate at critical project times (activities with severe risk, merge activities that may become late due to predecessors being late, noncritical activities to stop them from becoming critical, activities that require scarce resources to ensure adequate time to get the resources), if overall schedule is uncertain could add at end of project but requires top management & project owner authorisation
Implementation & control: use rpogress meetings & risk audits to evaluate identified risks remain valid, any changes in level of risk, implementation process, new treatments identified, new risks identified
Risk register: columns are number, risk, probability, consequences, rating, treatment, residual probability, residual consequences, residual rating, who, when, cost, status, contingency reseves through float (schedule) & reserves (financial) to cover/reduce risk, updated at status meeting
Contingency funds: funds to cover project risks — identified & unknown, size of funds reflects overall risk of project, use needs to be closely monitored, independent of original time/cost estimates (risk may not occur so not included in baseline, if occurs then draw and add to baseline, if not take out from reserve)
Contingency fund types:
Budget reserves: linked to identified risks of specific work packages, allocated to specific work packages/activities, communicated to project team but allocated by PM if required
Management reserves: large funds used to cover major unforeseen risks (ex: change in scope), allocated to entire project, established after budget reserves identified & funded, controlled by PM & project owner (internal/external), may contain technical reserves for project involving highly innovative process/product as fallback plan in case process/product is unsuccessful
Opportunity management tactics:
Exploit: seek to eliminate uncertainty associated with opportunity to ensure it definitely happens
Share: allocating some/all of ownership of opportunity who is best able to capture it for benefit of project
Enhance: take action to increase probability and/or positive impact of opportunity
Accept: be willing to take advantage if it occurs but not taking action to pursue
Resource scheduling: used to assign time-phased costs to provide project budget baseline
Time-phased baseline: cost baseline derived from WBS & rpoject schedule, budgeted costs are distributed to mirror schedule
Time-phased budgets: planned costs broken down by distinct time periods for work package, provides better cost control
Resource problem:
Resources & priorities: project network times not a schedule until resources assigned, implicit assumption is that resources will be available in required amounts when needed, adding new projects requires making realistic judgments of resource availability & project durations, cost estimates not a budget until they have been time-phased
Resource smoothing/leveling: involves attempting to even out varying demands on resources by using slack (delay non-critical activities) to manage resource utilisation when resources are adequate over life of project
Resource-constrained scheduling: duration of project may be increased by delaying late start of some activities if resources are not adequate to meet peak demands
Project constraint types:
Technical/logic: constraints related to networked sequence in which project activities must occur
Physical: activities that cannot occur in parallel or are affected by contractual or environmental conditions
Resource: absence, shortage or unique interrelationship and interaction characteristics of resources that require a particular sequencing of project activities, types: people, materials, equipment
Classification of scheduling problems: time-constrained & resource-constrained
Resource allocation assumptions:
Limiting: splitting activities not allowed (once started carried to completion), level of resources used for activity cannot be changed
Risk: activities with most slack pose least risk, reduction of flexibility does not increase risk, nature of activity (easy, complex) doesn't increase risk
Time-constrained projects: must be completed by imposed date, require use of leveling techniques that focus on balancing/smoothing resource demands, use positive slack (delay noncritical activities) to manage resource utilisation over duration of project
Resource leveling advantages: peak resource demands reduced, resources over life of project reduced, fluctuation in resource demand minimised
Resource leveling disadvantages: loss of flexibility that occurs from reducing slack, increases criticality of all activities
Time-constrained heuristic: minimum slack, smallest/least duration, lowest ID
Impacts of resource-constrained scheduling: reduces delay & flexibility, increase criticality of events, increase scheduling complexity, may make traditional critical path no longer meaningful, can break sequence of events, may cause parallel activities to become sequential & critical activities with slack to become non-critical
Splitting: scheduling technique used to get better project schedule and/or increase resource utilisation, pause activity to use resources on sth else, feasible when startup & shutdown costs low, considered major reason why projects fail to meet schedule
Benefits on scheduling resources: leave time for consideration of reasonable alternatives (cost-time tradeoffs, priority change), provide info for time-phased work package budgets to assess impact of unforeseen events & amount of flexibility in available resources
Managing multiproject scheduling: create project offices/depts to oversee scheduling of resources across projects, use project priority queuing system (FCFS for resources), centralise PM (treat all projects as part of megaproject), outsource projects to reduce number of projects handled internally
Multiproject scheduling problems:
Overall project slippage: delay on one create delays for others
Inefficient resource application: peaks & valleys of resource demands create scheduling problems & delays
Resource bottlenecks: critical resource shortage for multiple projects