Business Analyse

Inhoud

1. Introduction to Business Analysis

A. What is a Business Analysis?

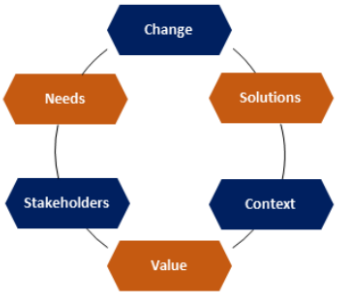
Business Analysis = the practice of enabling change in an enterprise by defining needs and recommending solutions that deliver value to stakeholders.

For Ex.: Need = We don't have a good system to know who I have to contact
Solution = New CRM system
Business Analyst → set up the requirements

I. BACCM

= Business Analyst Core Concept Model

→ 6 core concepts



II. IIBA

= to get a certificate to be a business analyst

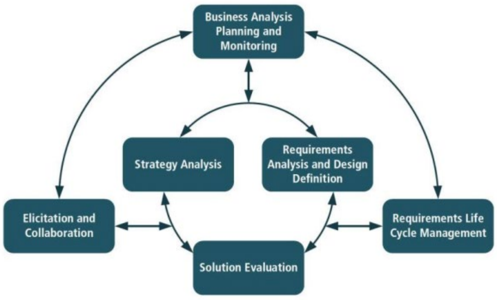
III. BABOK

= Business Analysis Body of Knowledge

→ Collection of concepts, activities, deliverables, competences, principles,….

→ Summary of everything about Business Analysis

→ Knowledge Areas (will look to all of them through the cursus)


Business Analysis Planning and Monitoring
Strategy Analysis
Requirements Analysis and Design Definition
Elicitation and Collaboration
Requirements Life Cycle Management
Solution Evaluation

B. Role of a Business Analyst?

→ A liaison among stakeholder in order to elicit

→ Analyze

→ Communicate

→ Validate requirements for changes to business processes, policies and information systems

→ To make a link between Business and IT

Roles of Business Analyst:

- Problem solver
- Facilitator = positive, continuous discussion and progress
- Negotiator = mediates between clients, stakeholders,…
- Architect
- Planner = Project lifecycle (define, organise, schedule,…)
- Communicator
- Expert = know your business, industry and domain
- Strategist = longterm vision, goals, tactical plans
  = Generalist

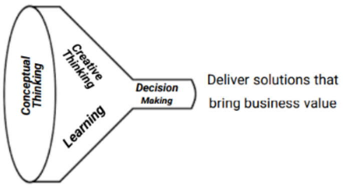
C. Competencies of a Business Analyst

Competence = the capability to apply or use a set of related knowledge, skills, and abilities required to successfully perform "critical work functions" or tasks in a defined work setting.

Areas:

- Business knowledge
  - External business knowledge
    - Business Acumen
      Awareness and knowledge about solutions that have been considered previously in other organisations.
      → Learning by doing and experience
    - Industry Knowledge
      You need a wide perspective of the industry
      → Competitors, current practices, trends, key processes, regulations,….
    - Information Technology
      How they work?, Trends, Technological advances, Methodologies, Development processes,…
  - Internal Business Knowledge
    - Business Model and Strategies
      The way a company creates and delivers value to its customers
    - Organizational Knowledge
      What is the structure?
      Units, departments, power of influence, communication channel

- Analytical thinking
  - Discovering
  - Synthesizing
  - Analysing
  - Identifying
  - Evaluating



- Organizing and time management

- Communication and interaction
  → Being neutral,….

- Tools and techniques
  → A lot!! **Business Process Modelling**

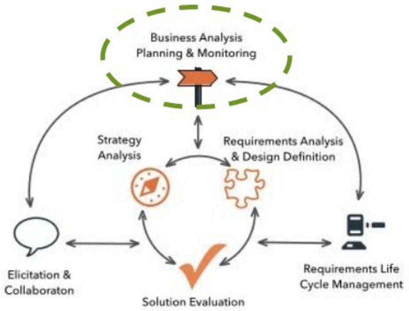
Decision Analysis

  - Decision Modelling

    - Root cause analysis
    - SWOT Analysis

D. BA Planning and Monitoring

Knowledge area


Business Analysis Planning & Monitoring
Strategy Analysis
Requirements Analysis & Design Definition
Elicitation & Collaboraton
Solution Evaluation
Requirements Life Cycle Management

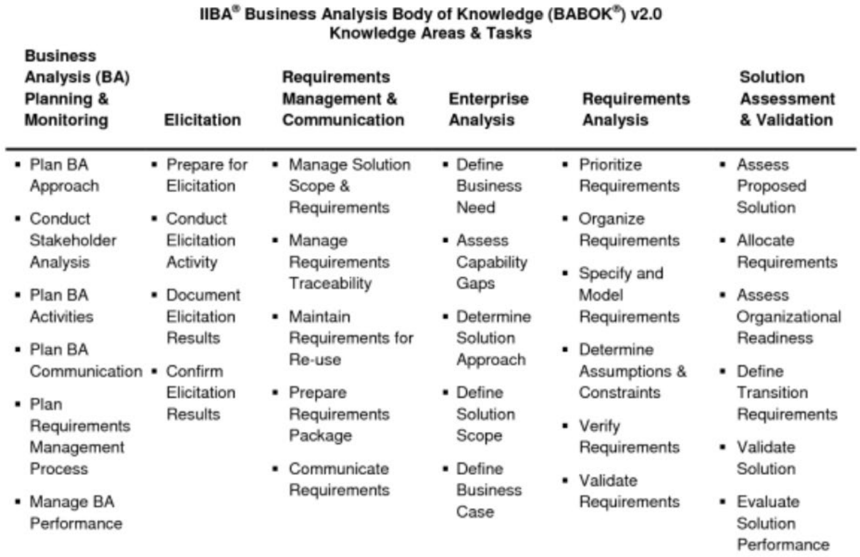
Planning Parameters

- Objective
    - What do they want to deliver at the end of the business analysis work?
- Needs
    - What are the perceptions?
    - Why it is the problem?
    - How is it affecting the business?
    - What are the desired effects or value of the solutions?
- Scope (very important)
    - Determine the boundaries of the project
        - What to consider and what not?
        - Enables planning of activities
- Approach = *aanpak*
    - Determine or follow internal policy of software development method (predictive of agile approach)
    - How to get the results to the context of:
        - The change
        - Involved resources
        - Standards and Methods
        - Restrictions
- Activities
    - What is required to achieve the outcome considering:
        - Activities
        - Tools/Methods
        - Sequential dependency
        - One-time vs iterations
        - Availability of key resources
        - Priority
    - And if applicable:
        - Dependencies
        - Limitations
        - Restrictions
- Complexity
    - What is feasible?
    - Where is complexity affected?
- Risk
    - Can be affected by:
            - Experience with similar problems

        - General sentiment
        - Experience and knowledge of stakeholders

- Approval
    - A plan can be approved by key stakeholders and resources allocated so to begin the business analysis work
        - Time
        - Cost
        - Resources
        - Results

Further knowledge areas and tasks



IIBA® Business Analysis Body of Knowledge (BABOK®) v2.0
Knowledge Areas & Tasks

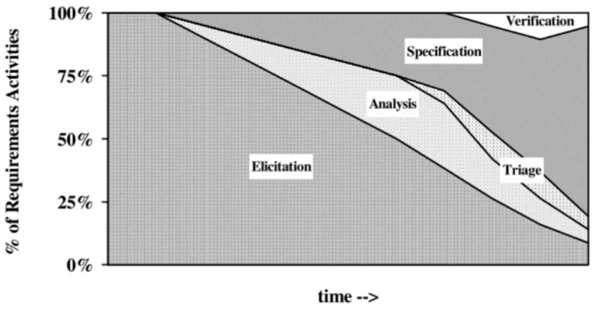
Business Analysis (BA) Planning & Monitoring	Elicitation	Requirements Management & Communication	Enterprise Analysis	Requirements Analysis	Solution Assessment & Validation
▪ Plan BA Approach ▪ Conduct Stakeholder Analysis ▪ Plan BA Activities ▪ Plan BA Communication ▪ Plan Requirements Management Process ▪ Manage BA Performance	▪ Prepare for Elicitation ▪ Conduct Elicitation Activity ▪ Document Elicitation Results ▪ Confirm Elicitation Results	▪ Manage Solution Scope & Requirements ▪ Manage Requirements Traceability ▪ Maintain Requirements for Re-use ▪ Prepare Requirements Package ▪ Communicate Requirements	▪ Define Business Need ▪ Assess Capability Gaps ▪ Determine Solution Approach ▪ Define Solution Scope ▪ Define Business Case	▪ Prioritize Requirements ▪ Organize Requirements ▪ Specify and Model Requirements ▪ Determine Assumptions & Constraints ▪ Verify Requirements ▪ Validate Requirements	▪ Assess Proposed Solution ▪ Allocate Requirements ▪ Assess Organizational Readiness ▪ Define Transition Requirements ▪ Validate Solution ▪ Evaluate Solution Performance

- Elicitation
    - “What do the stakeholder need?”
    - Gathering requirements from various stakeholders
    - Identify tasks, knowledge and techniques for capturing requirements
    - Techniques
        - Brainstorming
        - Focus groups
        - Interviewing
        - Observation
        - Prototyping
        - Requirements workshop
        - Survey
        - Document analysis
        - Interface analysis

- Requirements Management and Communications
    - “Does everyone understand and agree?”
    - Focus on representing and communicating documented requirements to all stakeholders to bring the group to consensus on project scope.
    - Identify and manage change

- Enterprise Analysis
    - “Why are we doing this?”
    - Define business goals the solution must meet
    - Integrate requirements into larger business architecture
    - Support initiatives and long-term planning

- Requirement Analysis
    - “What must the solution do?”
    - Analysing the data
    - Defines methods, tools, techniques to structure raw data collected during elicitation
    - Identifies gaps
    - Defines solution capabilities
    - Foundation for selecting among solution alternatives



- Solution Assessment & Validation
    - “Does the solution do what it is supposed to do?”
    - Focus on ensuring the best approach is chosen
    - Solution will be meet stakeholder objectives
    - Solution is feasible
    - Guides solution “verification”

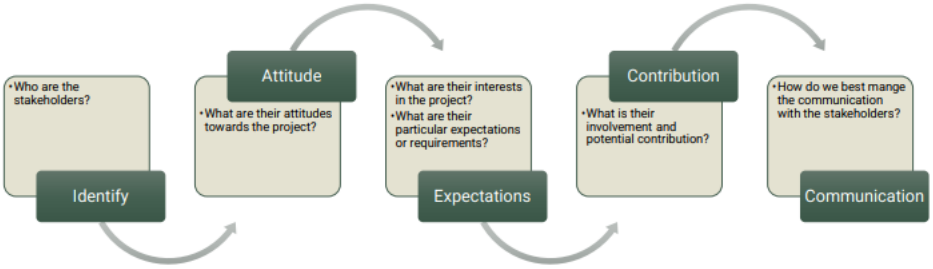
E. Stakeholder Engagement

Stakeholders = a person or group with a relationship to the change or solution

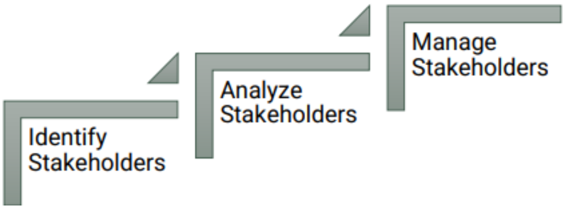
Importance:

- Efficient communication strategy
- Better stakeholder satisfaction
- Limiting scope creep

What we want to know:


•Who are the stakeholders?
Identify
Attitude
•What are their attitudes towards the project?
•What are their interests in the project?
•What are their particular expectations or requirements?
Expectations
Contribution
•What is their involvement and potential contribution?
•How do we best mange the communication with the stakeholders?
Communication

Stakeholder Engagement:



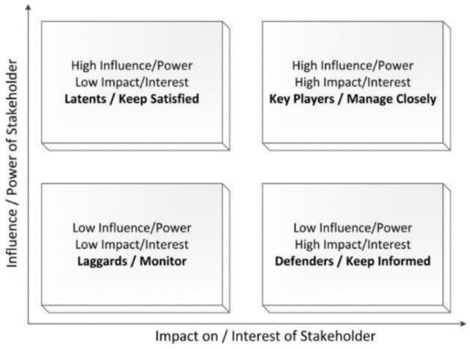
Who are stakeholders?

Internal and external

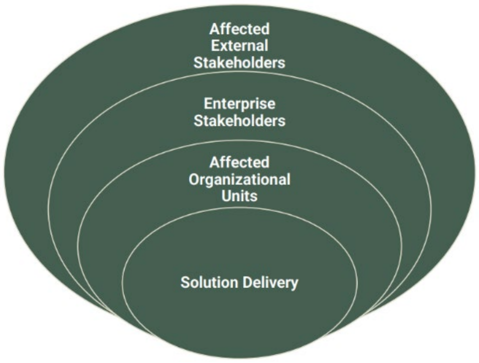
- Business Analyst
- Customers

- Domain Subject Matter Expert (DSME)
- End user
- Implementation Subject Matter Expert (ISME)
- Project managers
- Testers
- Regulator
- Sponsor
- Supplier

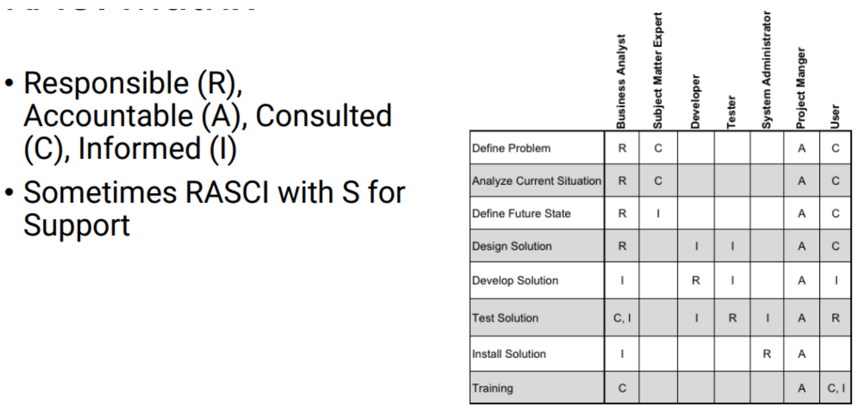
Stakeholder analysis



Union Diagram



RACI matrix



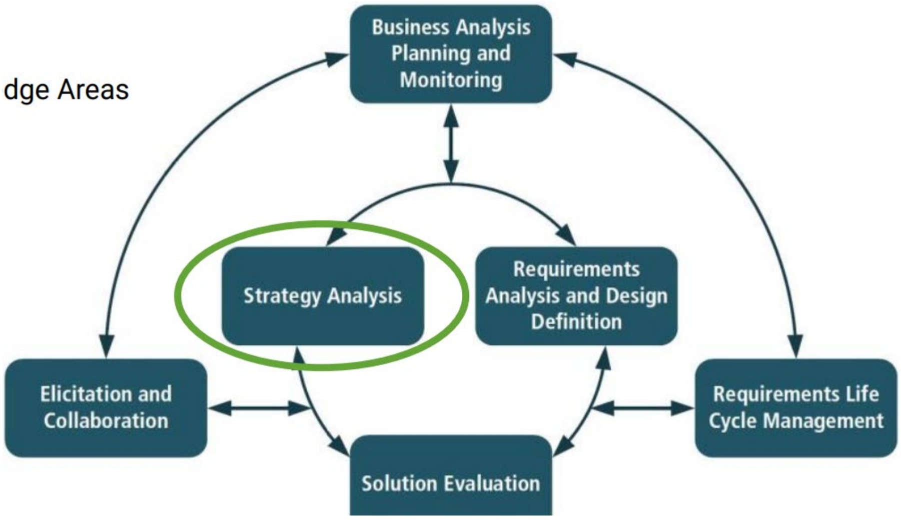
- Responsible (R), Accountable (A), Consulted (C), Informed (I)
- Sometimes RASCI with S for Support

	Business Analyst	Subject Matter Expert	Developer	Tester	System Administrator	Project Manger	User
Define Problem	R	C				A	C
Analyze Current Situation	R	C				A	C
Define Future State	R	I				A	C
Design Solution	R		I	I		A	C
Develop Solution	I		R	I		A	I
Test Solution	C, I		I	R	I	A	R
Install Solution	I				R	A	
Training	C					A	C, I

F. Conclusion

- Business Analysis is a broad domain focused on identifying and developing solutions within organizations for particular needs or changes that can drive business value
- Business Analysts are typically generalists carrying out different roles in a project typically with a varied skill set
- A key success factor for BA projects is proper stakeholder management, focusing on analysis, management and communication

2. Context Analysis

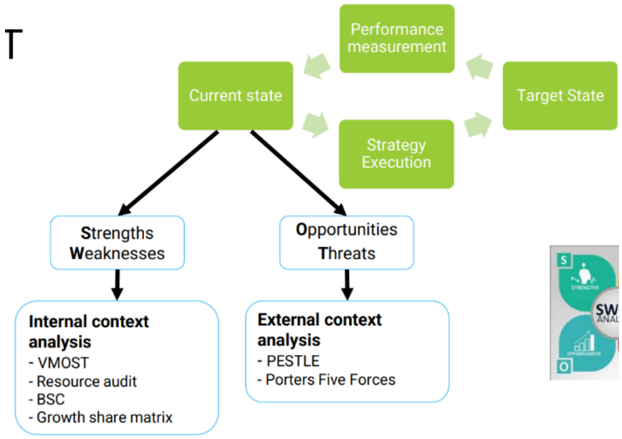

dge Areas
Business Analysis Planning and Monitoring
Strategy Analysis
Requirements Analysis and Design Definition
Elicitation and Collaboration
Requirements Life Cycle Management
Solution Evaluation

A. Business Strategy

Strategy = an integrated set of actions aimed at increasing long-term wellbeing and strength of an enterprise
= 5 P's of strategy (Plan, Ploy, Pattern, Position, Perspective)

Importance → Foundation for success through aligning execution with the context of the internal and external environment

Business analyst: working without reference to strategic context reduces the value of his service.



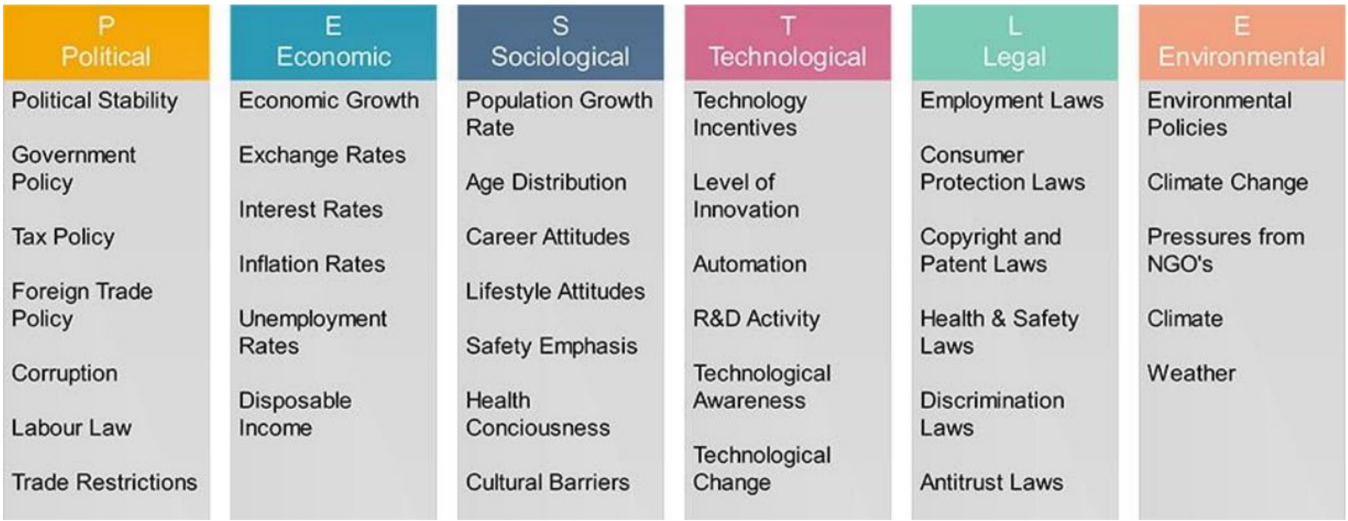
B. External Context Analysis

→ Opportunities and Threats

→ External Perspective

- Economic Growth
- Monetary policy
- Regulations
- Technological Trends

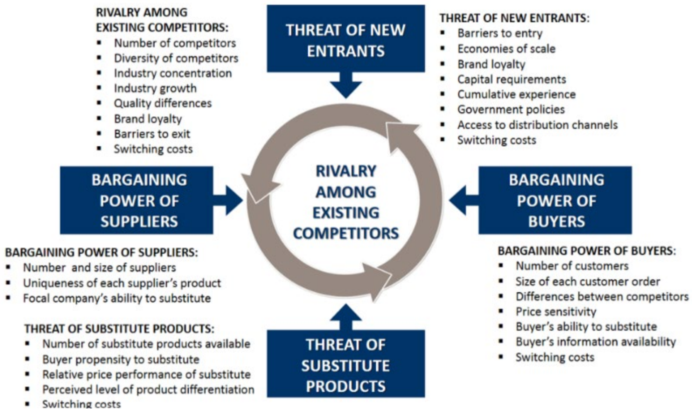
I. PESTLE Analysis



P Political	E Economic	S Sociological	T Technological	L Legal	E Environmental
Political Stability Government Policy Tax Policy Foreign Trade Policy Corruption Labour Law Trade Restrictions	Economic Growth Exchange Rates Interest Rates Inflation Rates Unemployment Rates Disposable Income	Population Growth Rate Age Distribution Career Attitudes Lifestyle Attitudes Safety Emphasis Health Conciousness Cultural Barriers	Technology Incentives Level of Innovation Automation R&D Activity Technological Awareness Technological Change	Employment Laws Consumer Protection Laws Copyright and Patent Laws Health & Safety Laws Discrimination Laws Antitrust Laws	Environmental Policies Climate Change Pressures from NGO's Climate Weather

Extra E for Ethical → GDPR, AI regulatory,…

II. Porters' Five Forces Model


THREAT OF NEW ENTRANTS
BARGAINING POWER OF SUPPLIERS
RIVALRY AMONG EXISTING COMPETITORS
BARGAINING POWER OF BUYERS
THREAT OF SUBSTITUTE PRODUCTS

RIVALRY AMONG EXISTING COMPETITORS:

- Number of competitors
- Diversity of competitors
- Industry concentration
- Industry growth
- Quality differences
- Brand loyalty
- Barriers to exit
- Switching costs

THREAT OF NEW ENTRANTS:

- Barriers to entry
- Economies of scale
- Brand loyalty
- Capital requirements
- Cumulative experience
- Government policies
- Access to distribution channels
- Switching costs

BARGAINING POWER OF SUPPLIERS:

- Number and size of suppliers
- Uniqueness of each supplier's product
- Focal company's ability to substitute

BARGAINING POWER OF BUYERS:

- Number of customers
- Size of each customer order
- Differences between competitors
- Price sensitivity
- Buyer's ability to substitute
- Buyer's information availability
- Switching costs

THREAT OF SUBSTITUTE PRODUCTS:

- Number of substitute products available
- Buyer propensity to substitute
- Relative price performance of substitute
- Perceived level of product differentiation
- Switching costs

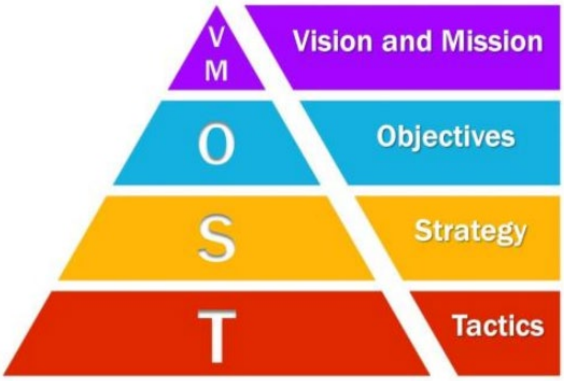
C. Internal Context Analysis

→ Strengths and weaknesses

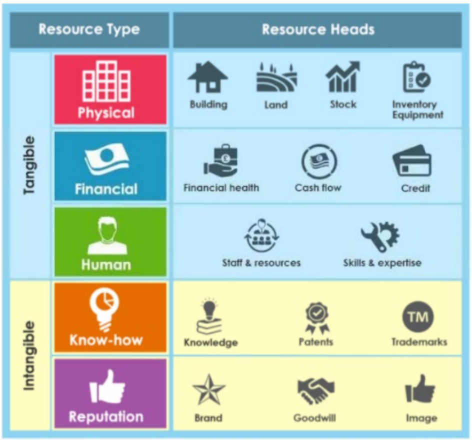
→ Internal Perspective

- Organizational strategy
- Business model
- Capabilities
- Organizational culture and structure

I. VMOST



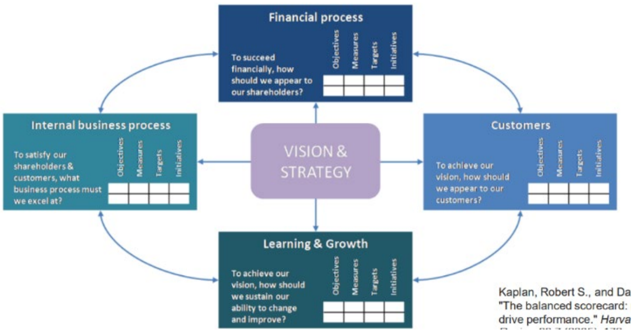
II. Resource audit



→ more specific

→ Identify Strengths and weaknesses per resource type

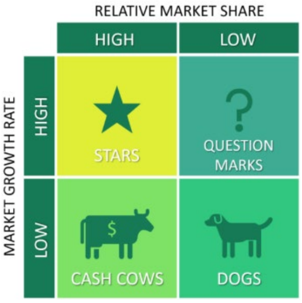
III. BSC = Balanced Scorecard


Financial process
To succeed financially, how should we appear to our shareholders?
Objectives
Measures
Targets
Initiatives
Internal business process
To satisfy our shareholders & customers, what business process must we excel at?
VISION & STRATEGY
Customers
To achieve our vision, how should we appear to our customers?
Learning & Growth
To achieve our vision, how should we sustain our ability to change and improve?

Kaplan, Robert S., and Da
"The balanced scorecard:
drive performance." *Harva*

→ To define the KPI's

IV. Boston Box



= Growth share matrix

= BCG matrix

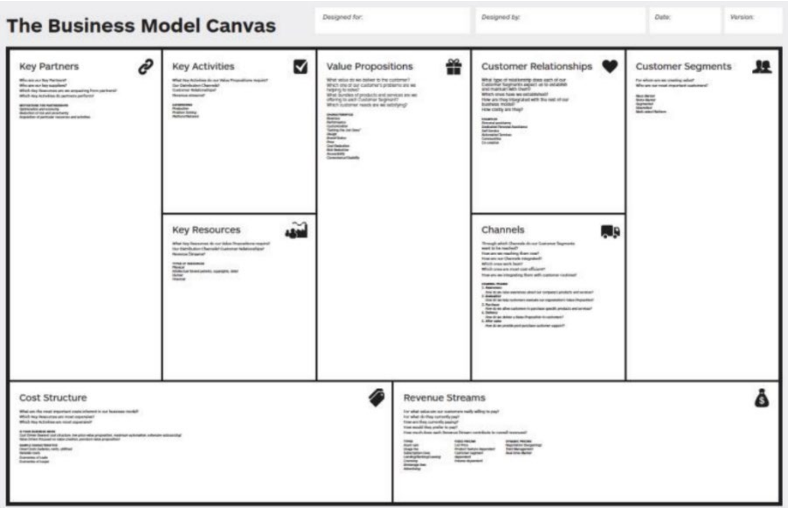
→ for marketing

→ Identify strengths and weaknesses from the product portfolio

D. Strategy Execution

→ Analysing the gap between current and desired target state

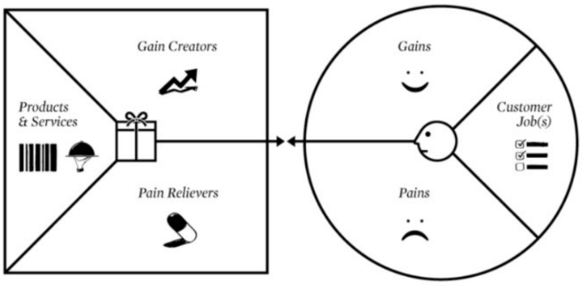
I. Business Model Canvas = BMC



→ How an organisation creates, manages and delivers business value

→ With a financial part

- Customer Segments & Value propositions
  - Who are the organization's important customers?



Value Proposition Canvas

→ There has to be a match
→ Product Market Fit

- Customer jobs
  The things your customers are trying to get done in their work or life
  - Functional jobs
    Performing, completing specific tasks, solving specific problems
  - Social jobs
    Wants to look 'good', gain social value, reputation
    How the customer wants to be perceived by others.
  - Emotional jobs
    Aim to achieving a certain emotional state or feeling
- Customer pains
  Anything that causes the customer be become annoyed before, during or after getting a job done
- Customer gains
  Customer is getting the outcome or benefits they want.

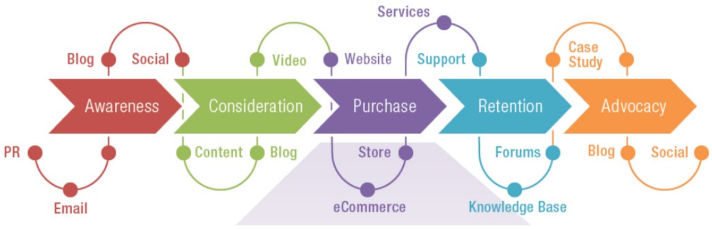
- Products and services
  A list of wat you offer
  - Physical
  - Intangible
  - Digital
  - Financial

- Pain relievers
  Describes how exactly your product or service alleviate specific customer pains

- Gain creators
  Describes how the product or service create gains for the customer

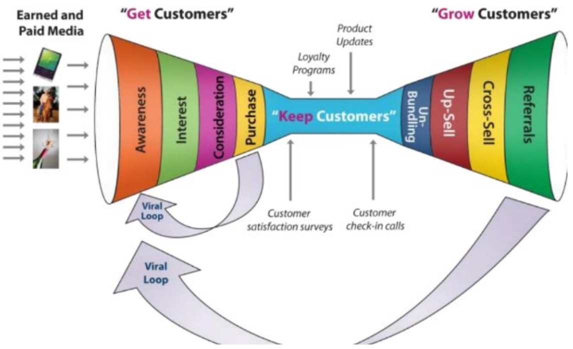
- **Market Segmentation** = dividing a market into different clusters of customers
  Customers are different between but similar within.

- Channels & Customer Relationships
  - Customer journey mapping

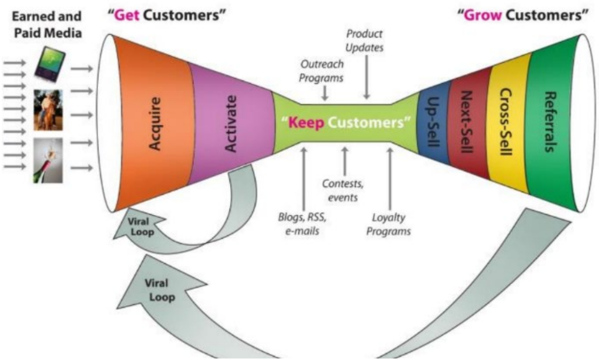


- Channel → how a company communicates with customers

Funnel of Physical products



Virtual products



- Customer Relationships
  - Personal Assistance

    Human interaction
  - Dedicated Personal Assistance
  - Self-Service

    No direct contact, but help themselves
  - Automated Services
  - Communities

    Where users help each other and companies understand their customers better
  - Co-Creation

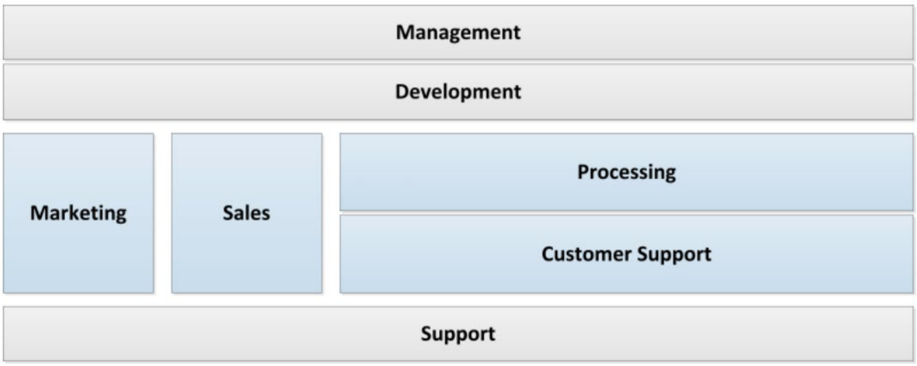
- Key Resources & Key Activities
  - Describes the most important assets required to make a business model work
  - Describes the most important things a company must do to make its business model work

- Cost Structure and revenue streams
  - Profit = revenues – costs

II. Business Capability Model = BCM

Capability = abstract collection of resources, processes and technologies that together in whatever combination, enables an organization to achieve a desired outcome.

- Describes what an organisation does
- Is long lasting
- No duplication



Management

Development

Marketing

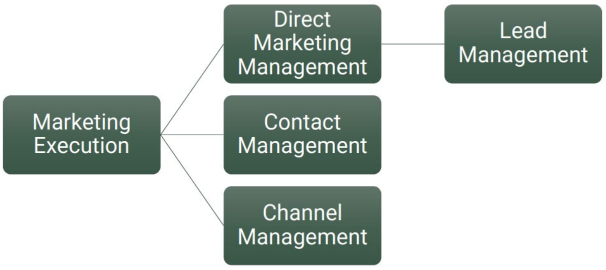
Sales

Processing

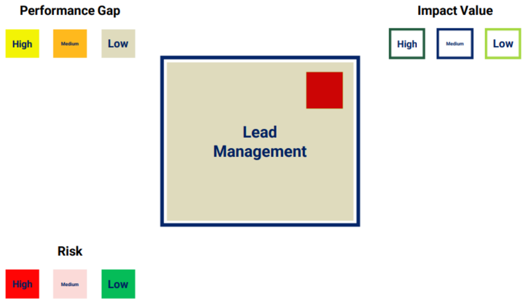
Customer Support

Support

- Levels of capabilities



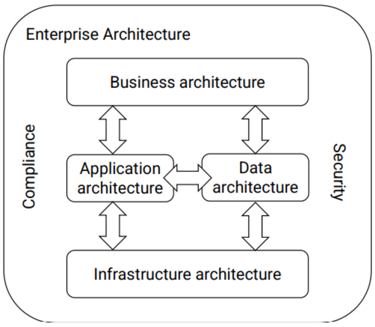
Modelling a capability



- Value of Capability Analysis
  - Awareness of capabilities that initiatives are to support
  - Alignment of business initiatives with strategic goals and directions
  - Identify where to strengthen a capability
  - Build upon strengths for new initiatives
  - Identify gaps and develop needs

!!!! done at high level of management!!!!

E. Link to Enterprise Architecture



Business architecture = blueprint of the enterprise that provides a common understanding of the organisation and is used to align strategic objectives and tactical demands

Data architecture = describe all the data that are being held within the business

Infrastructure architecture = hardware, …

Compliance architecture **vs** **security architecture**

Internal and external Compliance expectations

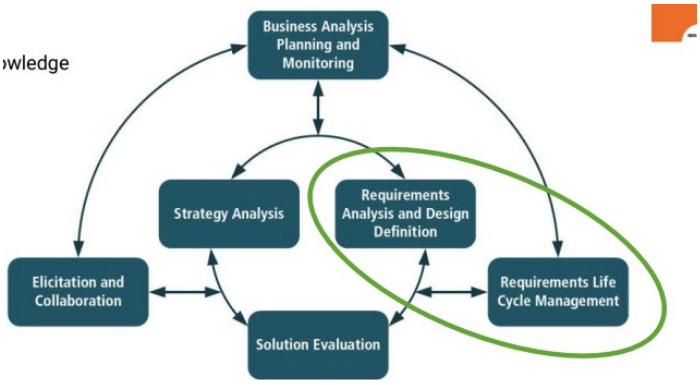
protect assets from harm, loss and danger

Main frameworks: • TOGAF (The Open Group Architectural Framework) • Zachman Framework • Federal Enterprise Architecture • Gartner Methodolog

F. Conclusion

- We scrutinized some frameworks for
  - External context analysis
  - Internal context analysis
  - Strategy execution
- Link to EA
- Business analysts' concern?
  - Strategic awareness allows for improved BA service delivery
  - Strategic alignment is a key success factor for many BA projects
  - Especially important for senior business analysts

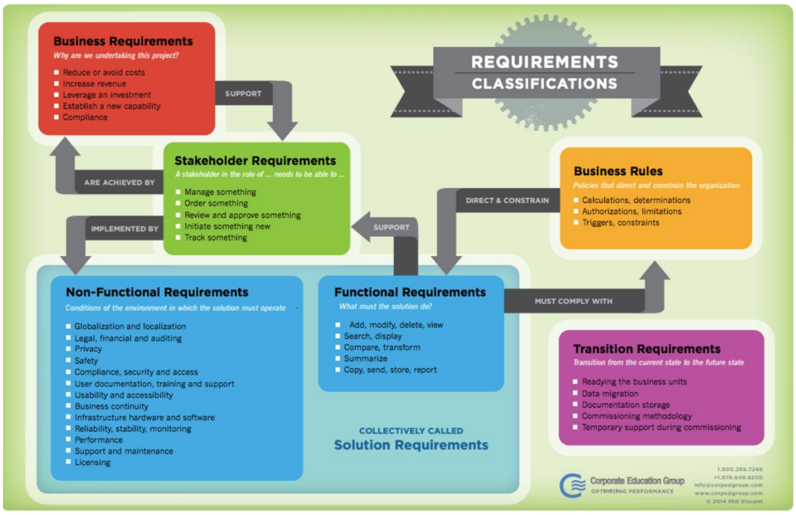
3. Requirements Engineering


›wledge
Business Analysis Planning and Monitoring
Strategy Analysis
Requirements Analysis and Design Definition
Elicitation and Collaboration
Requirements Life Cycle Management
Solution Evaluation

A. Requirements definition and types

Requirement = feature or characteristic that has been requested by a stakeholder and may form part of a solution

- Types of Requirements


REQUIREMENTS
CLASSIFICATIONS
Business Requirements
Why are we undertaking this project?
Reduce or avoid costs
Increase revenue
Leverage an investment
Establish a new capability
Compliance
SUPPORT
ARE ACHIEVED BY
Stakeholder Requirements
A stakeholder in the role of ... needs to be able to ...
Manage something
Order something
Review and approve something
Initiate something new
Track something
IMPLEMENTED BY
SUPPORT
DIRECT & CONSTRAIN
Business Rules
Policies that direct and constrain the organization
Calculations, determinations
Authorizations, limitations
Triggers, constraints
Non-Functional Requirements
Conditions of the environment in which the solution must operate
Globalization and localization
Legal, financial and auditing
Privacy
Safety
Compliance, security and access
User documentation, training and support
Usability and accessibility
Business continuity
Infrastructure hardware and software
Reliability, stability, monitoring
Performance
Support and maintenance
Licensing
Functional Requirements
What must the solution do?
Add, modify, delete, view
Search, display
Compare, transform
Summarize
Copy, send, store, report
MUST COMPLY WITH
COLLECTIVELY CALLED
Solution Requirements
Transition Requirements
Transition from the current state to the future state
Readying the business units
Data migration
Documentation storage
Commissioning methodology
Temporary support during commissioning
Corporate Education Group
OPTIMIZING PERFORMANCE
1.800.288.7246
+1.978.649.8200
info@corpedgroup.com
www.corpedgroup.com
© 2014 Phil Vincent

  - Business Requirements
    - Higher level statements of the goals, objectives or needs of the enterprise

  - Stakeholder Requirements
    - User perspective – needs that a given stakeholder has an dhow that will interact with a solution
    - Bridge between business requirements and various classes of solution requirements

  - Solution Requirements
    - Characteristics of a solution that meet business and stakeholders requirements
    - Two sub-categories

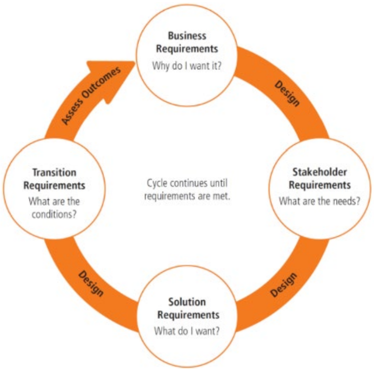
      - Functional Requirements

      - Non-functional Requirements

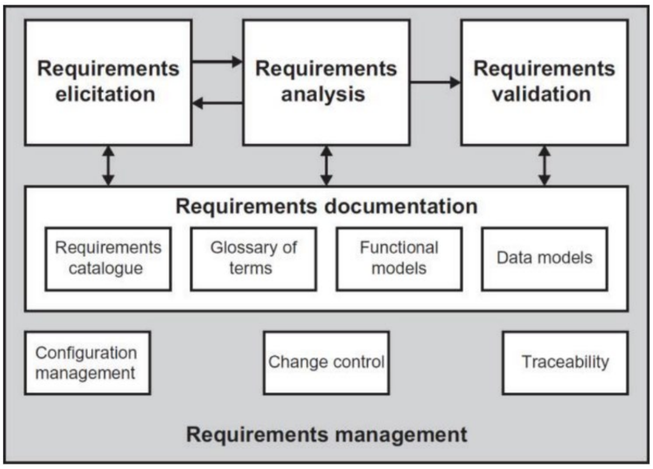
  - Functional Requirements
    - The behaviour and information that he solution will manage

  - Non-functional Requirements
    - Capture conditions that do not directly relate to the behaviour or functionality of the solution, but rather describe environmental conditions under which the solutions must remain effective or qualities that the system must have
    - = quality requirements

        - Can be related to capacity, speed, security, availability and the information structure and presentation of the user interface
    - Transition Requirements
        - If you change anything, fix that this is possible in a minimum time
        - Describe the capabilities that a solution must to facilitate transition from the current state until the future state, but which are not needed once the change is complete



- Requirements engineering



RE Framework

→ Clarifies the activities to be carried out when defining requirements
→ Requirements should be:

- Unambiguous
- Well - structured
- Correct
- Relevant

B. Requirements Elicitation

- Most crucial stage of the RE process
- Uncover, acquire and elaborate requirements
- Different Sources
    - Stakeholders
    - Documentation
    - Existing systems
- Different Techniques

I. Qualitative Elicitation Techniques

- Gaining impressions and opinions
- Types
    - Collaborative
        - Workshop
        - Focusgroup
    - One to one
        - Interviews
        - Meetings
        - Observations
    - Scenarios

    - Prototyping and wireframes (what will happen if you push on this button? ~ Panenco)
    - User Role analysis

II. Quantitative Elicitation Techniques

- Focused on volumes, frequencies
- Types
    - Surveys
    - Questionnaires
    - Document Analysis

C. Requirements Analysis

- Identify requirements
    - That overlap
    - Are in conflict with other requirements
    - Are duplicates
    - Need to be separated into individual requirements because they are too complex

I. Categorizing requirements

II. Defining/accepting requirements

→ Examine elicited requirements , filter, and build a well-formed set

- Evaluating Feasibility
    - Technical
    - Business
    - Financial
- Quality of expression
    - Clear
    - Concise
    - Consistent
    - Relevant
    - Unambiguous
    - Correct
    - Testable
    - Traceable

III. Modelling requirements

IV. Prioritising requirements

Important task, continuous shifting

→ number of collected requirements can be huge, 2 approaches to prioritization

- MoSCoW Technique

    → Helping priority discussions between stakeholders and business analysts
    - Make use of a nominal scale to sort the requirements
    - Extremely simple
    - Stands for
        - Must have

            Highest priority, must satisfy

            Solution without this requirement (at least one of these conditions is true)

            - makes no sense to the stakeholders
            - is illegal
            - is unsafe

        - Should have

            Usable, has an impact, is not absolutely necessary

            A requirement is a should have if: (at least one of these conditions is true)

                - is important but not essential
                - solution remains relevant
                - solution cannot work but a workaround can be used to deal with the gap

        - Could have

            They wish, but has a low impact

            Stakeholder could accept to remove the requirement

        - Would have (or won't have)

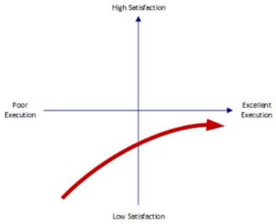
            Lowest priority

            They are interesting to implement over long run

- KANO model
    - Same way as MoSCow model but uses 5 levels
    - Express importance of a requirement based on satisfaction and level of implementation
    - 5 levels
        - Must be

            ~ Must have

            Stakeholders takes this for granted



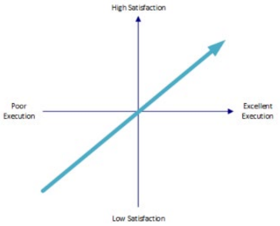
        - One-Dimensional

            ~ Should have

            What the customer demand for the solution to be accepatble

            Satisfaction gets higher as the implementation level increases

            → Proportionally



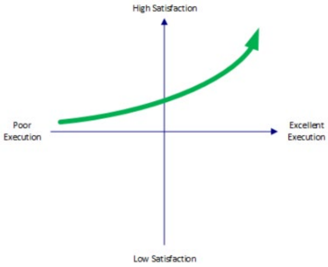
        - Attractive

            ~ Could have

            Stakeholders will appcreciate this greatly if implemented

            Satisfaction gets higher as the implementation level increases

            → Not proportionally

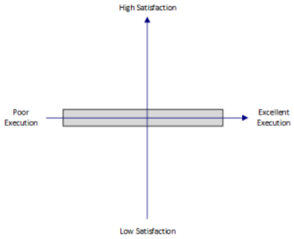


        - Indifferent

            No impact

            Stakeholders do not demand and they are indifferent

            Satisfaction stays steady if the implementation level increases



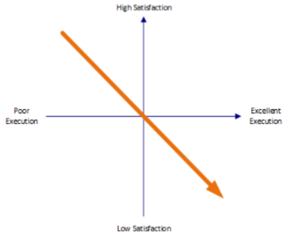
- Reverse

  Negative impact

  What the customer does not want

  Satisfaction gets lower as the implementationlevel increases

  → Proportionally



D. Requirements Validation

- = check that the documented requirements are consistent with actual customer expectations
- Show that requirements define the system that the customer really wants, conducted by external stakeholders
- Very high costs of errors on requirements
- Is not testing = verify that the system begin implemented comply with the specifications, expectations defined in the requirements documents
- Formal vs informal validation
  - Linear project approach →documented requirements
  - Agile project approach → less formal, but still sufficient clarity required to include them
- Criteria
  - Validity

    Does the requirements reflect the actual expectations of the customer?
  - Consistency

    Are there conflicts between requirements?
  - Completeness

    Are all functions required by the customer included?
  - Realism

    Can the requirements be implemented given the budget and available technology?
  - Verifiability

    Can the requirements be checked?
- Techniques
  - Requirements review

    = systematic manual analysis
  - Prototyping

    = executable preliminary model of the solution to check in practice if the requirements are correct
  - Test-case

    = not testing, but think how we are going to test?

    = preparation for testing purposes

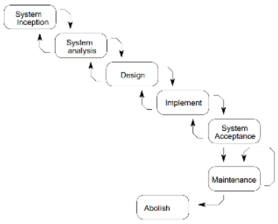
E. Requirements Documentation

- Importance
  - Enables communication
  - Basis for ensuring requirements consistency
  - Provides a firm basis for validating that there is an accurate record of what the solution should provide
  - Essential for further activities
- Documentation Style
  - Text based
    - Requirements catalogue
    - User Story
  - Diagrammatic
    - Use case model
    - Data model
    - Business process model

- Linear vs agile
  - Linear project → more formal, high documentation
    - Waterfall

      1 analysis
    - Requirements catalogue

      - Document for defining requirements
      - Key characteristics



  - Agile project → more ad-hoc, requirements aren't steady
    - Scrum

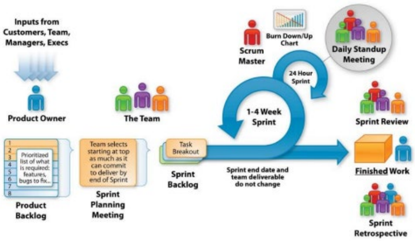
      Continuous carrying out
    - User Stories

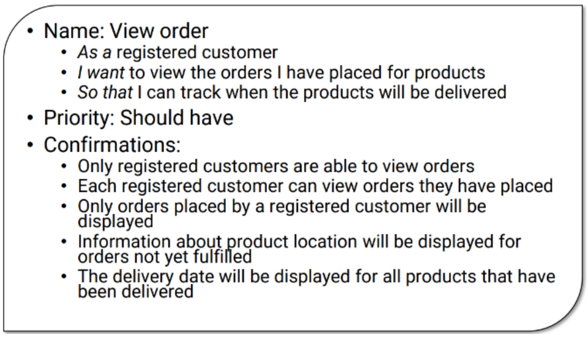
      - Backlog of user stories
      - Define the features actors require from a system
      - Actor or user role perspective
      - Quick to develop
      - Outline the identified requirement

      → 3Cs framework

        1. Card
        2. Conversation
        3. Confirmation

      → Standard format


Inputs from Customers, Team, Managers, Execs
Product Owner
The Team
Scrum Master
Burn Down/Up Chart
Daily Standup Meeting
24 Hour Sprint
1-4 Week Sprint
Sprint Review
1
2
3
4
5
6
7
8
Prioritized list of what is required: features, bugs to fix...
Product Backlog
Team selects starting at top as much as it can commit to deliver by end of Sprint
Sprint Planning Meeting
Task Breakout
Sprint Backlog
Sprint end date and team deliverable do not change
Finished Work
Sprint Retrospective

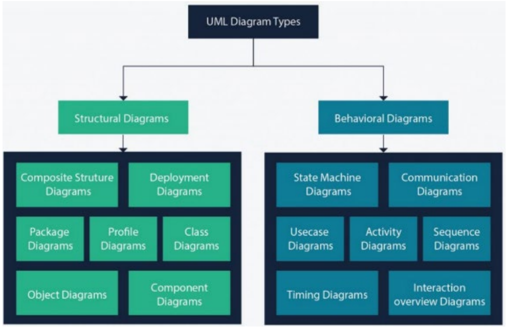


- Name: View order
  - *As a* registered customer
  - *I want* to view the orders I have placed for products
  - *So that* I can track when the products will be delivered
- Priority: Should have
- Confirmations:
  - Only registered customers are able to view orders
  - Each registered customer can view orders they have placed
  - Only orders placed by a registered customer will be displayed
  - Information about product location will be displayed for orders not yet fulfilled
  - The delivery date will be displayed for all products that have been delivered

F. Requirements Modelling

- Diagrammatic models
  - ○ Describing requirements in textual format is often difficult
    - ▪ Ambiguity
    - ▪ Poor precision
    - ▪ Unclear
    - ▪ No holistic perspective
  - ○ Models are ideal instrument

    → UML = Unified Modelling Language



I. Modelling the business perspective

- Excellent for business stakeholders to represent their view of the solution
- Solution still remains a black box
- Different levels of detail
  - ○ Context diagram
  - ○ Business use case diagram
  - ○ System use case diagram

    →UML Use Case Diagrams

    - Actors

        User roles, external systems,…

    - Use case

        Actor wants the system to do

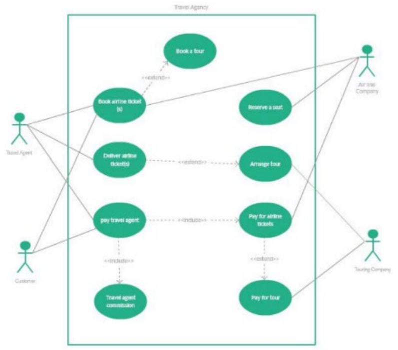
    - System boundary

        Large box around the use cases

    - Associations

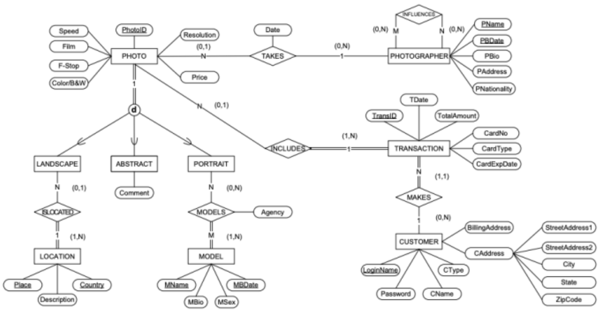
        Linking actors and use cases

    - Include <-> exclude

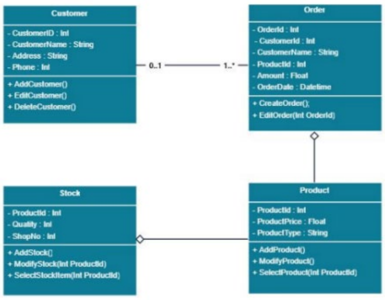


II. Modelling the data perspective

- Allows the stakeholders who use the system to agree the data that is to be recorded and accessed
- 2 standard techniques
  - ○ Entity relationship diagrams (ERDs)



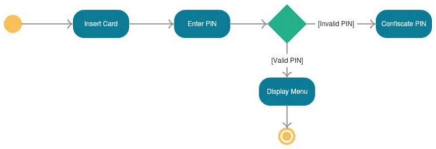
  - ○ UML Class Diagrams



III. Modelling the process perspective

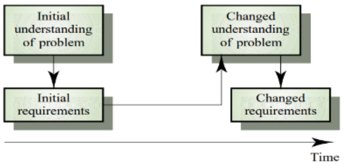
- Key perspective for defining and documenting functional requirements
- Techniques
  - BPMN

    Task 1 of the group assignment
  - UML activity Diagrams



G. Requirements Management

- Overall process of managing changing requirements
- Begins at the same time of the project, but continues after the end of the project
- Requirements change because:
  - Different stakeholders have different requirements
  - New requirements
  - Experience of user
  - Environment changes


Initial understanding of problem
Changed understanding of problem
Initial requirements
Changed requirements
Time

- Levels
  - Sustainable Requirements

    Stable requirements derived from core activity

  - Volatile Requirements

    Requirements change during development
    - Mutable Requirements

      If the environment changes
    - Emerging Requirements

      Emerge as the understanding of the system develops
    - Consequential Requirements

      Impact of another project
    - Compatibility Requirements

      Depend on other systems of processes

- Traceability

  Relationship between requirements
  - Horizontal Traceability → from inception to delivery
    - Backwards from traceability

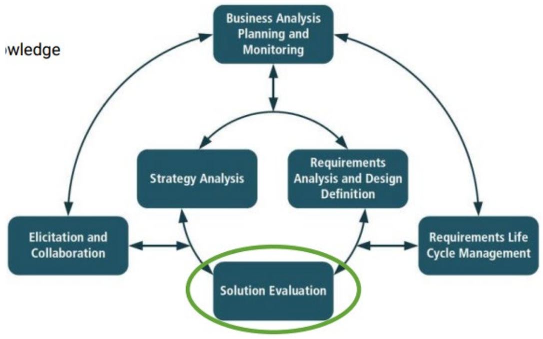
      "What was the source for this requirement and who raised it?"
    - Forwards to traceability

      " What happened to this requirement?"
  - Vertical Traceability → up or down the hierarchy

H. Conclusion

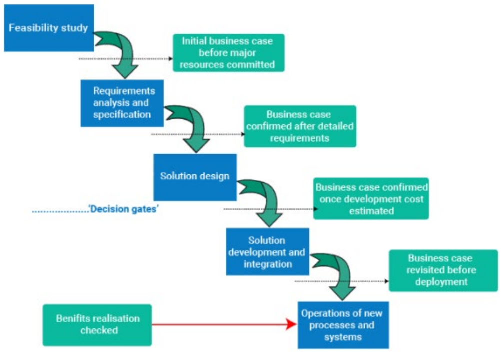
- Requirements are a fundamental concept in conducting a business analysis project
- Requirements Engineering entails elicitation, analysis, modelling, documentation, validation, and management
- Requirements documentation is a crucial aspect within a business analysis project
- Business and especially functional requirements are often subject to diagrammatic modelling
    - Providing clarity, consistency, sufficient detail, alignment, etc.
    - Most important viewpoints:
        - Business: use case diagrams
        - Process: business process models
        - Data: data models

4. Delivery of Solution and Project Management


wledge
Business Analysis Planning and Monitoring
Strategy Analysis
Requirements Analysis and Design Definition
Elicitation and Collaboration
Requirements Life Cycle Management
Solution Evaluation

A. Making the Business Case

Business Case = presents and evaluates one or more courses of action that will address a problem or enable the organisation to grasp a business opportunity
→ Supports decision making


Feasibility study
Initial business case before major resources committed
Requirements analysis and specification
Business case confirmed after detailed requirements
Solution design
'Decision gates'
Business case confirmed once development cost estimated
Solution development and integration
Business case revisited before deployment
Benifits realisation checked
Operations of new processes and systems

- When?
  - Directly after preliminary investigation
  - Living document – ongoing review
  - Projects should pass certain tests before they can proceed to a next stage
- Structure
  - Introduction
  - Executive summary
  - Description of the current situation
  - Option considered
    - Option description
    - Analysis of costs and benefits

      → Tangible and intangible costs and benefits

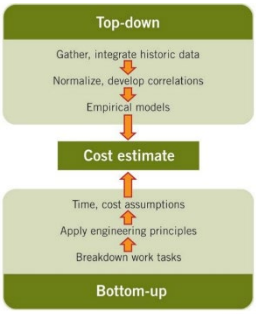
      → Techniques:

      - Payback (break-even)
      - Discounted Cash Flow (DCF) / Net present value (NPV)
      - Internal Rate of Return (IRR)
    - Impact Assessment

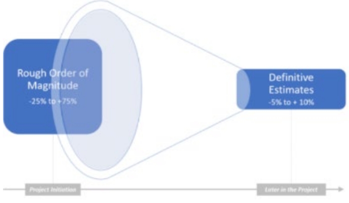
      - Organisation structure
      - Interdepartmental relations
      - Working practices
      - Management style
      - Recruitment
      - Appraisal and promotion
      - Supplier relations
    - Risk Assessment

      Description + Impact + Probability + Countermeasures + Ownership
  - Recommendations
- Cost of the solution

- Top-down vs. Bottom-up
    - Top-down
        - Fast
        - Less costly
        - Less accurate
    - Bottom-up
        - More accurate
        - Detailed solution description required

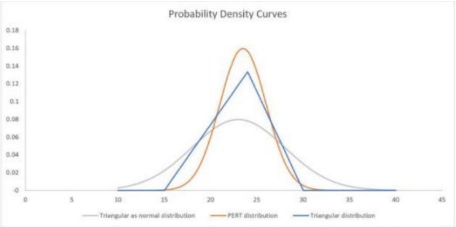

Top-down
Gather, integrate historic data
Normalize, develop correlations
Empirical models
Cost estimate
Time, cost assumptions
Apply engineering principles
Breakdown work tasks
Bottom-up

- Rough Order of Magnitude (ROM)
    - Provide stakeholders with a rough idea of the projects costs
    - Upper boundary = ROM_estimate x 1,75
    - Lower Boundary = ROM_estimate x 0,75



Project-Management.info

- PERT estimation (more advanced)
    - 3 different estimates
        - Optimistic estimate
        - Pessimistic estimate
        - Most likely estimate



Project-Management.info

B. Solution Development Approaches

- Delivery Style
    - Numerous methods, standards and lifecycles that may be used when developing solutions to fulfil defined requirements
    - Factors
        - Roles

          Key roles to be performed during the project
        - Deliverables

          Artefacts to be delivered
        - Context

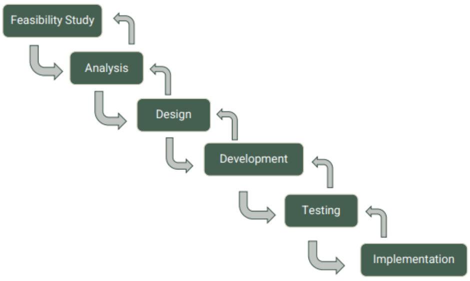
          Characteristics of the business and project
        - Lifecycle

          The process adopted for development and implementation

- Context

  It is very important to know in which context you are
    - Culture and philosophy
    - Business context
    - Constraints
    - Prioritised business needs
    - Project drivers

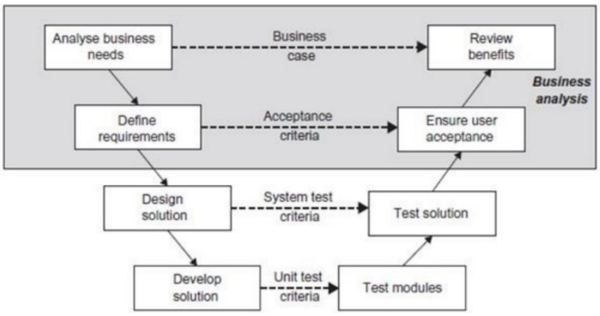
- Delivery Lifecycles
  - A clear basis for conducting development projects
  - Sets out a sequence of stages
  - Main SDLCS = Software Development Lifecycle
    - Waterfall



→ Strict separation of the stages
→ Linear approach

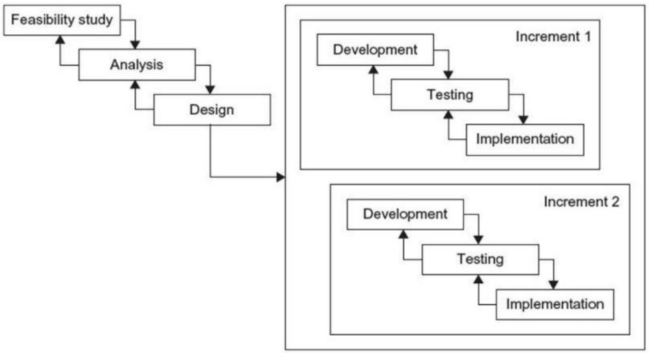
+ Strong basis for firm and clear project management
+ Support delivery of high-quality solution
- High risk for projects delays due to quality focus
- Not enable adaption and change well

    - V model



→ Mix and match the different stages
→ Variant of waterfall model
→ Same benefits and drawbacks

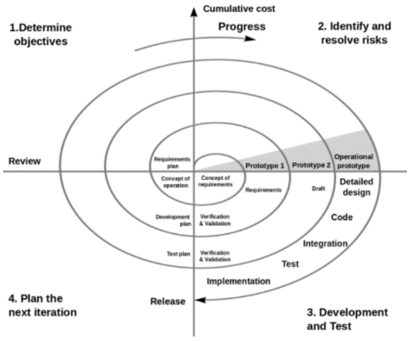
    - Incremental lifecycle



→ Recognize the difference in importance of certain requirements
→ Developing and delivering the solution in a series of increments
- High priority requirements first
- Lower priority requirements deferred

- Higher costs
- High level analysis and design
- Less flexible

- Iterative lifecycle



→ Most popular
→ More flexible for changing requirements
→ Starting from the middle
→ Basis for agile approaches
  - DSDM
  - Scrum

→ Agile principles:

Customer satisfactions
Changing requirements
Frequent delivery
Communicate regularly
Support team member
Face-to-face communication
Measure work progress
Development process
Good design
Measure progress
Continue seeking result
Reflect and adjust regularly

→ Main Characteristics

Collaborative working
Prioritised requirements
Timeboxed iterations
Evolutionary development
Empowered teams
Incremental delivery
Continuous testing
(DSDM)
Experiential learning

→ Agile Methodologies

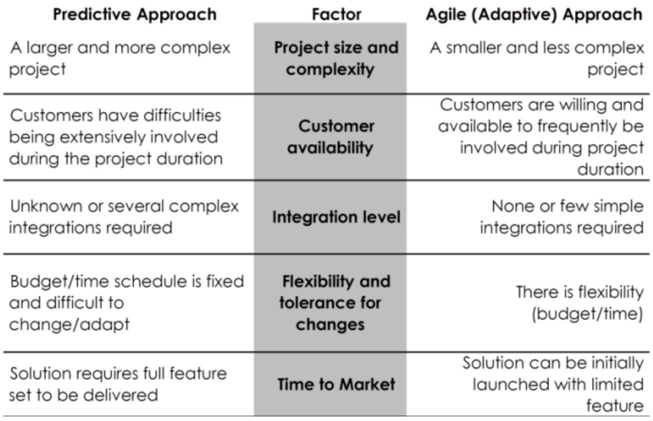
Scrum
Lean
Kanban
Crystal
Extreme Programming (XP)
Feature Driven Development (FDD)
Dynamic System Development Method

- Lifecycles summary

SDLC	Predictive/Adaptive	Linear/Evolutionary
Waterfall	Predictive	Linear
V model	Predictive	Linear
Incremental lifecycle	Predictive	Evolutionary
Agile	Adaptive	Evolutionary

Predictive → business stakeholders know exactly what they want at the outset of a project

- Selecting an approach



Predictive Approach	Factor	Agile (Adaptive) Approach
A larger and more complex project	**Project size and complexity**	A smaller and less complex project
Customers have difficulties being extensively involved during the project duration	**Customer availability**	Customers are willing and available to frequently be involved during project duration
Unknown or several complex integrations required	**Integration level**	None or few simple integrations required
Budget/time schedule is fixed and difficult to change/adapt	**Flexibility and tolerance for changes**	There is flexibility (budget/time)
Solution requires full feature set to be delivered	**Time to Market**	Solution can be initially launched with limited feature

C. Testing the Solution

→ There is a solution, we have to test this solution

→ Important part of implementation phase

- Testing is
  - Trying to demonstrate that the systems works fine
  - Running the system with the purpose of finding errors
  - Finding the differences between the desired results and the real results
  - Measuring software quality
- Test levels
  - Unit/module testing – Lowest level
  - Integration testing – system as a whole
  - System testing – specific aspects
    - Safety test
    - Volume test
    - Performance test
    - Stress test
  - Acceptance testing – highest level

- Software testing lifecycle
  - Unit tests – Validate that each unit of the software performs as designed
    - Static testing (no execution)

      - Desk checking
      - Structured walkthroughs
      - Control flow & reachability
      - Data flow

- Dynamic testing

  Execution with test cases

  - Black Box testing

    = Functional testing

    → Independent form the code

    → Random testing

  - White Box testing

    = Content testing

    → Look at the module content

    → Selecting input values that cover as much as possible

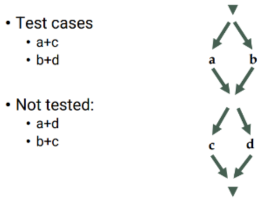
    → Criteria

    - ❖ Path coverage

      Testing each possible path through the model

    - ❖ Branch coverage

      Test at least each branch



- Test cases
  - a+c
  - b+d
- Not tested:
  - a+d
  - b+c

    - ❖ Structured testing

      Boundary cases of loops

    - ❖ Special values testing

      Exceptions, limits

- Integration tests – Check that modules work together in combination
  - Bottom-up

    Empty control module calling the module to be tested
  - Top-down

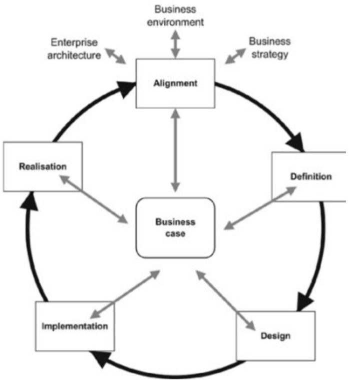
    Empty modules at lower level
  - Bing-bang

    Combine all modules and test entire system

- System tests – Evaluate the system's compliance with the specified requirements

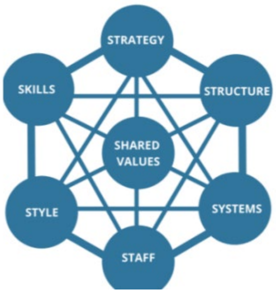
- Acceptance tests – Evaluate the system's compliance with the business requirements, no technology
  - Involve customers and stakeholders!

D. Delivering the Solution

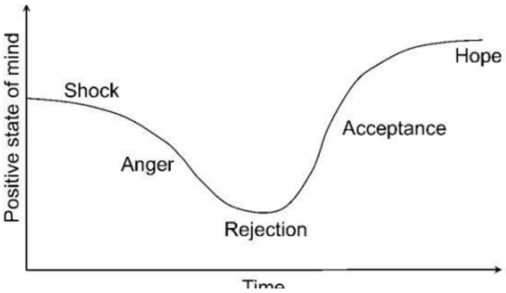


I. Implementation stage

- Requires planning and careful execution
- 3 major aspects
  - Business Readiness assessment
    - Can this solution work?
    - Is the business area prepared to accept and operate the new ways of working?

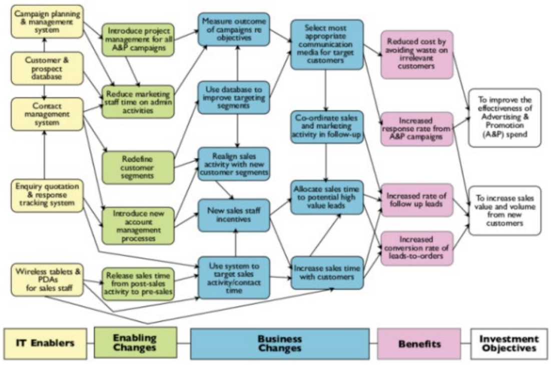


  - Transition and migration
    - Data migration
    - Training sessions
    - Creation of users guides, procedure descriptions,…
    - Deciding on implementation strategy
      - Direct changeover
      - Parallel running
      - Pilot running
      - Phased implementation
  - People's response to change
    - SARAH curve



II. Realisation stage

- How the expected business benefits are to be achieved
- Aspects
  - Benefits plan
  - Benefits dependency network
    - You can map the overall high level benefits to the lowest level technically received
  - Benefits review/business case management


Campaign planning & management system
Customer & prospect database
Contact management system
Enquiry quotation & response tracking system
Wireless tablets & PDAs for sales staff
Introduce project management for all A&P campaigns
Reduce marketing staff time on admin activities
Redefine customer segments
Introduce new account management processes
Release sales time from post-sales activity to pre-sales
Measure outcome of campaigns re objectives
Use database to improve targeting segments
Realign sales activity with new customer segments
New sales staff incentives
Use system to target sales activity/contact time
Select most appropriate communication media for target customers
Co-ordinate sales and marketing activity in follow-up
Allocate sales time to potential high value leads
Increase sales time with customers
Reduced cost by avoiding waste on irrelevant customers
Increased response rate from A&P campaigns
Increased rate of follow up leads
Increased conversion rate of leads-to-orders
To improve the effectiveness of Advertising & Promotion (A&P) spend
To increase sales value and volume from new customers
IT Enablers
Enabling Changes
Business Changes
Benefits
Investment Objectives

E. Conclusion

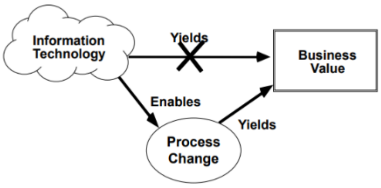
- A lot of focus in business analysis projects is on the analysis and solution design
- However, delivering the solution and managing the way in which the solution is developed (and tested) are crucial for success as well
- Key elements
  - The business case
  - The delivery style and project management lifecycle
  - Solution testing
  - Solution delivery

1. Introduction to BPM

A. The world of Business Process Management

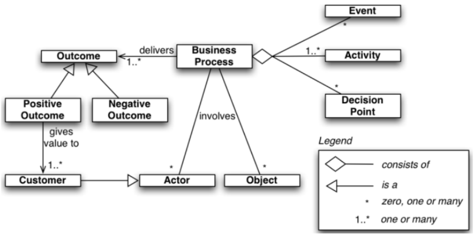
- What is it?
    - Body of principles, methods and tools to design, analyse, execute and monitor business processes

- Why?
    - To improve a business process
    - Automation applied to an efficient operation will magnify the efficiency <-> automation applied to an inefficient operation will magnify the inefficiency



- How to engage?
    - Continuous Process Improvement (CPI)
        - Make small changes
        - Does not look at the current process structure
    - Business Process Re-Engineering (BPR)
        - Fundamental changes
        - Look at the fundamental assumptions and principles of the existing process structure

- Business processes
    - Collection of related events, activities and decisions that involve a number of actors and objects and that collectively lead to an outcome that is of value to an organization or its customers.



-
    - The core elements of a process
        - Activities
            - Active elements
            - State-changing
        - Events
            - Passive elements
            - Represents conditions
            - Instantaneous
        - Business objects = data
            - Physical or electronic information
        - Actors
            - Performing process activities' and generating events
            - Human and systems

        → How to combine?

        1. Control Flow

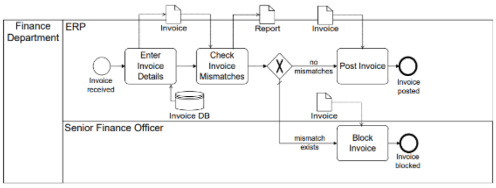
            "What needs be done and when?"

        2. Data

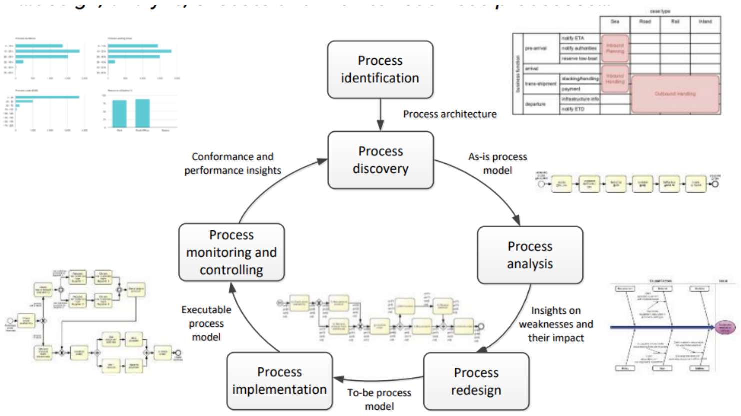
            "What do we need to work on?"

        3. Resources

            "Who's doing the work?"



B. The BPM Lifecycle

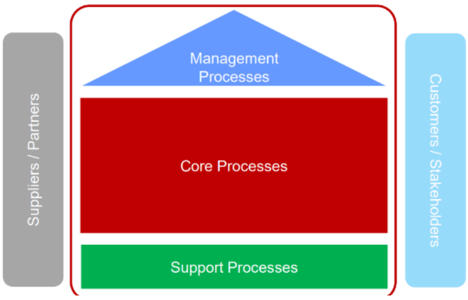


I. Process Identification

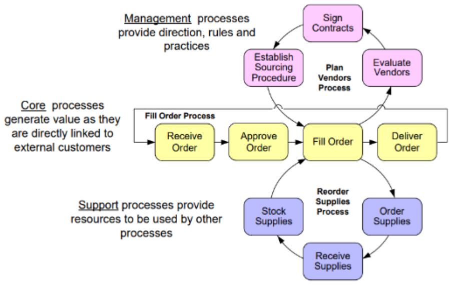
- Combination of all different processes
- Over the whole organization
- Steps
  - Designation phase
    - Enumerate main processes
      - Not so easy
      - Most business have 3 core processes:

        Sell stuff, deliver stuff and making sure you have stuff to sell and deliver

- Porter: Types of processes



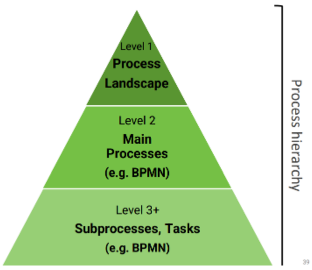
- Relations between core, support and management processes


Management processes provide direction, rules and practices
Sign Contracts
Establish Sourcing Procedure
Plan Vendors Process
Evaluate Vendors
Core processes generate value as they are directly linked to external customers
Fill Order Process
Receive Order
Approve Order
Fill Order
Deliver Order
Support processes provide resources to be used by other processes
Reorder Supplies Process
Stock Supplies
Order Supplies
Receive Supplies

- Determine process scope
  - Processes are independent → interrelations
    - Specialization – General
    - Horizontal – Upstream – Downstream

      When are we in a new step of our process?

      1. Change of key business object
      2. Change of granularity of a main business object
      3. Change in frequency/time
      4. Change in intermediate outcome/resolution/objective
    - Vertical – Main – Sub-processes
      - Value chains
      - Processes
      - Subprocesses
      - Process tasks



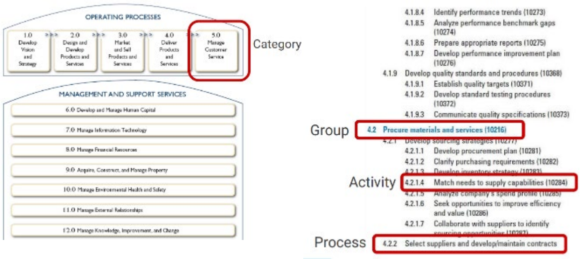
→**Process Architecture**

- Value chains
- Groups

- Via reference models
  - APQC Process Classification Framework (PCF)

    Four levels:

    1. Categories
    2. Process group
    3. Process
    4. Activity

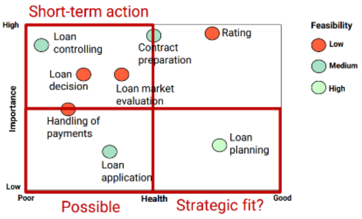

OPERATING PROCESSES
1.0 Develop Vision and Strategy
2.0 Design and Develop Products and Services
3.0 Market and Sell Products and Services
4.0 Deliver Products and Services
5.0 Manage Customer Service
Category
MANAGEMENT AND SUPPORT SERVICES
6.0 Develop and Manage Human Capital
7.0 Manage Information Technology
8.0 Manage Financial Resources
9.0 Acquire, Construct, and Manage Property
10.0 Manage Environmental Health and Safety
11.0 Manage External Relationships
12.0 Manage Knowledge, Improvement, and Change
4.1.8.4 Identify performance trends (10273)
4.1.8.5 Analyze performance benchmark gaps (10274)
4.1.8.6 Prepare appropriate reports (10275)
4.1.8.7 Develop performance improvement plan (10276)
4.1.9 Develop quality standards and procedures (10368)
4.1.9.1 Establish quality targets (10371)
4.1.9.2 Develop standard testing procedures (10372)
4.1.9.3 Communicate quality specifications (10373)
Group
4.2 Procure materials and services (10216)
4.2.1.1 Develop procurement plan (10281)
4.2.1.2 Clarify purchasing requirements (10282)
Activity
4.2.1.4 Match needs to supply capabilities (10284)
4.2.1.6 Seek opportunities to improve efficiency and value (10286)
4.2.1.7 Collaborate with suppliers to identify
Process
4.2.2 Select suppliers and develop/maintain contracts

- Evaluation phase

  = Process Selection

  - Prioritize process based on
    - Importance
    - Health
    - F easibility

  → **Prioritized Process Portfolio**



C. Process discovery: as-is process modelling

I. Process Discovery

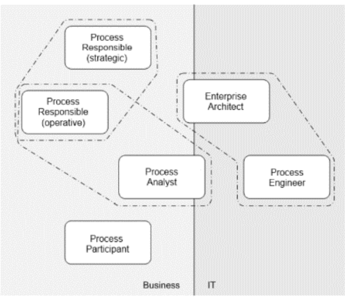
1. Defining the setting

   → Assembling a team that will be responsible for working on the process
2. Gathering Information

   → Building understanding of the process
3. Conducting the modelling task

   → Organizing the creation of the process model, gives guidance for mapping out the process in a systematic way.

- Who is involved?
  - Domain expert
  - Process analyst



- Challenges
  - Fragmented process knowledge
    - Most important people are those who are in the process, they know their own part bud don’t have an overview of the process
  - Domain experts think on instance level
  - Knowledge about process modelling is rare
    - What is a good language to model a process?

- Elicitation techniques
  1. Document analysis
     - Documentation point to existing roles, activities and business objects
     - Formal documentation
     - Forms
     - Work instructions

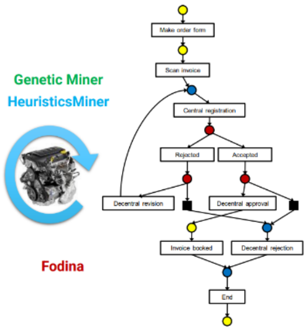
     - In real life is not the same as on papers

2. Observation
   - What people do at their workplace
   - Inspect the work environment

- Observation bias
- Time consuming
- Few cases to observe

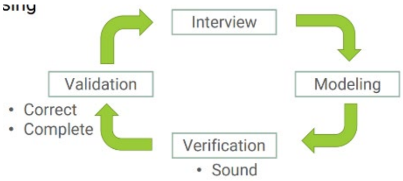
3. Process Mining: automated process discovery
   - Observe digital cases
   - Objective + detailed

- Quality of data can be very low



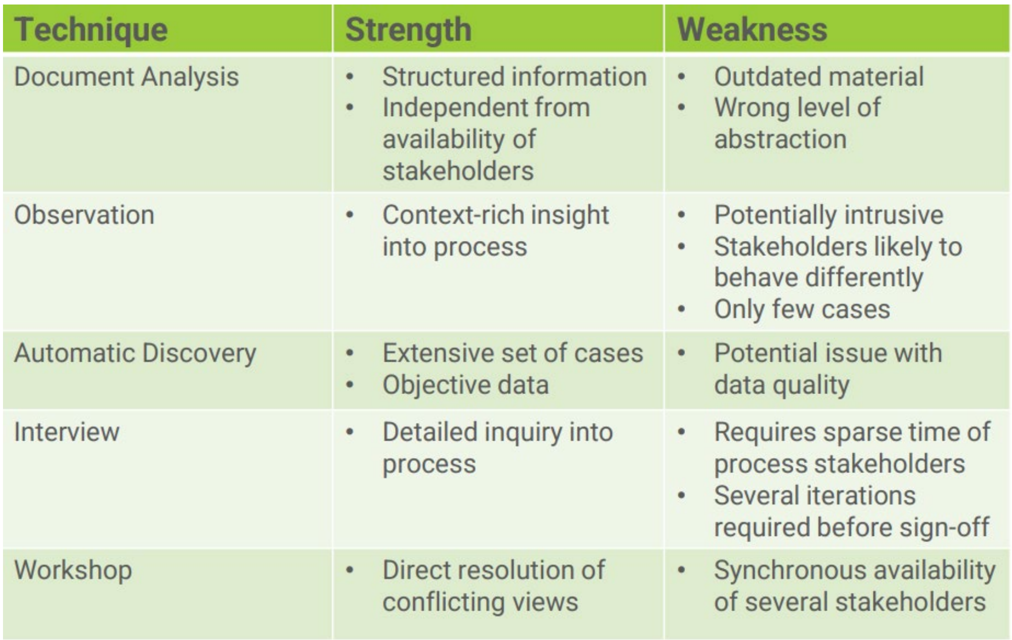
4. Interview-based
   - Structured vs unstructured
   - Analyst and stakeholder share terminology
   - Identifying deviations from standard processing

+ You can ask for information or start a discussion where you are looking for
- Time consuming
- Are they telling the truth?



5. Workshop-based
   - Gather all stakeholders together
   - Point for discussions

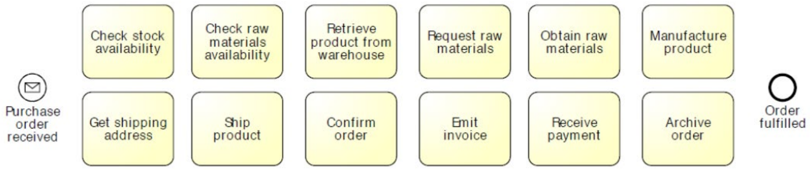
+ Efficient
+ Direct conflict resolution
- Extreme difficult to plan



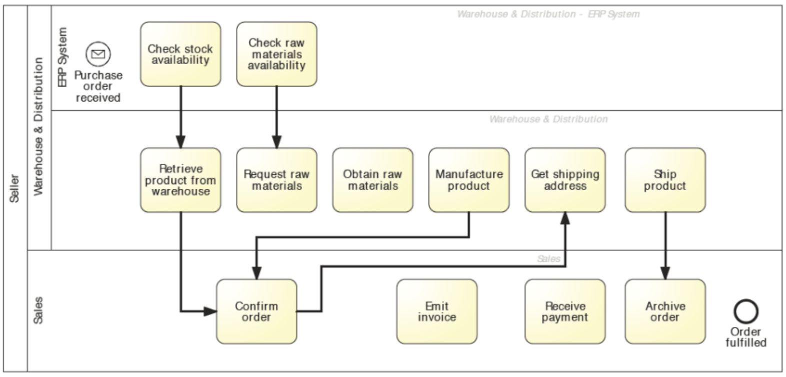
Technique	Strength	Weakness
Document Analysis	• Structured information • Independent from availability of stakeholders	• Outdated material • Wrong level of abstraction
Observation	• Context-rich insight into process	• Potentially intrusive • Stakeholders likely to behave differently • Only few cases
Automatic Discovery	• Extensive set of cases • Objective data	• Potential issue with data quality
Interview	• Detailed inquiry into process	• Requires sparse time of process stakeholders • Several iterations required before sign-off
Workshop	• Direct resolution of conflicting views	• Synchronous availability of several stakeholders

- Organizing the gathered material and conducting the modelling task
  1. Identify the process boundaries
     - Under which condition does the process start?
     - With which result does it end?
     - Which perspective do you assume?

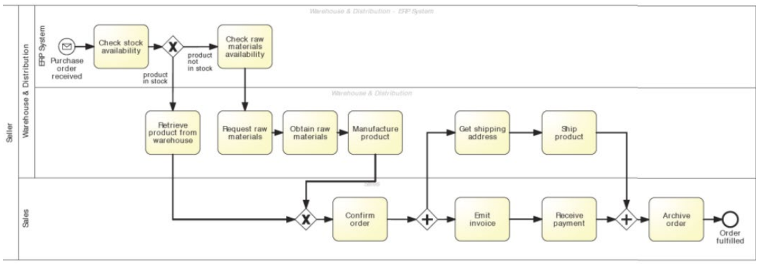
  2. Identify activities and events



  3. Identify resources and their handovers



  4. Identify the control flow

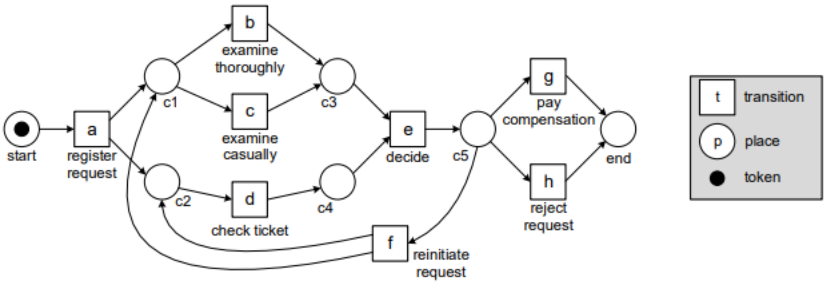


  5. Identify additional elements

2. Essential Process Modelling

A. Process modelling languages

- Petri nets
  - Automatically control if the model is correct
  - Not so user-friendly



$$P = \{start, c1, c2, c3, c4, c5, end\}$$

$$T = \{a, b, c, d, e, f, g, h\}$$

$$F = \{(start, a), (a, c1), (a, c2), (c1, b), (c1, c), \ldots\}$$

$$m_0 = [start^1, c1^0, c2^0, c3^0, c4^0, c5^0, end^0] \quad or \quad m_0 = [start]$$

  - Execution semantics
    - A transition can fire when it contains at least one token in each of its input places
  - Workflow nets
    - One input place = source place

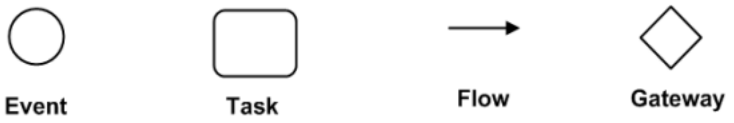
      One place with only outgoing arcs
    - One output place = sink place

      One place with only incoming arcs
    - The net is strongly connected

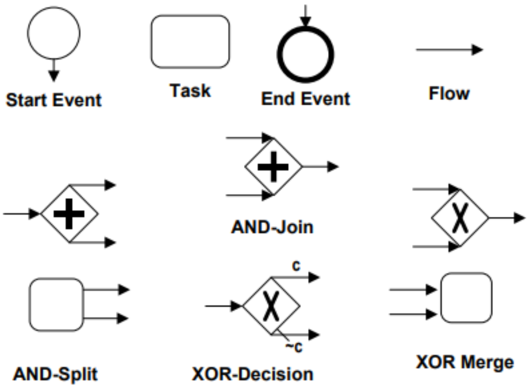
      There is a directed path between any pair of nodes

→ Strongly connected if there is a directed path between any pair of nodes

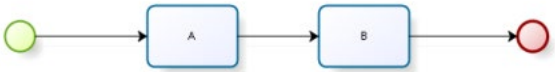
- BPMN
  - Elements



  - Basic Flow elements

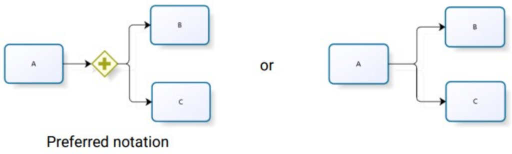


    - Basic routing patterns
        - Sequential

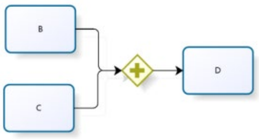


        - Parallel

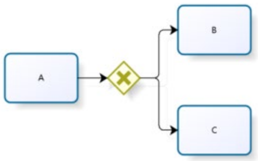
- AND-split



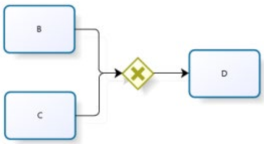
- AND-join



- XOR-split



- XOR-join



        - Choice
        - Iteration

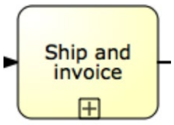
- Other languages
    - YAWL
    - EPCs
    - Transition sytems

B. Process modelling with BPMN: Control-flow

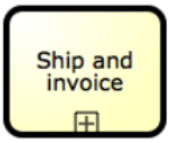
(basics)

C. Process modelling with BPMN: sub processes

- If the process is not easy to read → try to make some subprocesses



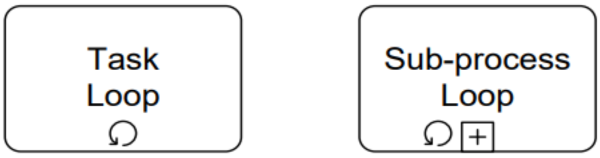
- Call activities
    - Shared subprocess, subprocess is needed at multiple places in the model



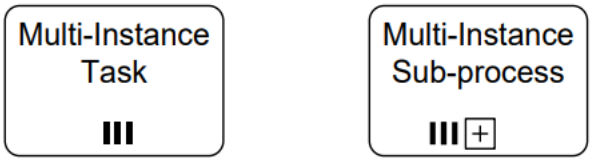
- Rules for subprocesses
    - Start with one start event
    - End with one end event
    - Sequence flows CANNOT cross subprocess boundaries
    - Message flows can cross subprocess boundaries

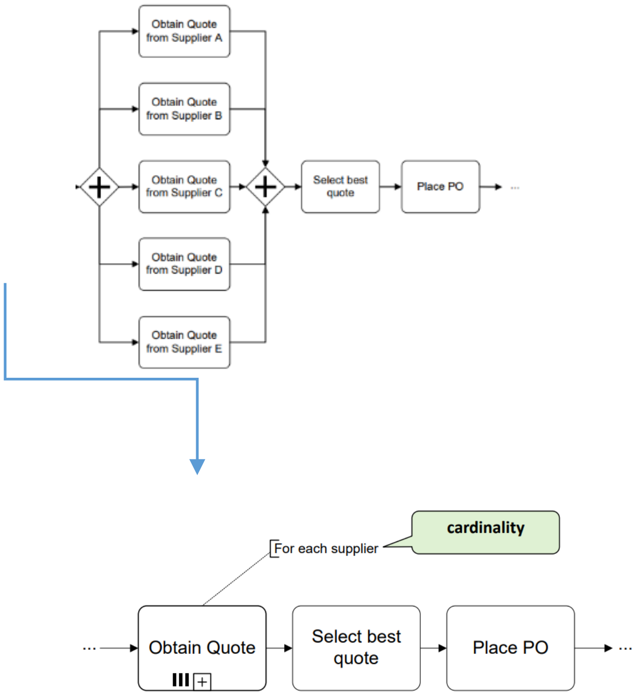
D. Process modelling with BPMN: repetition

- Sequential repetition

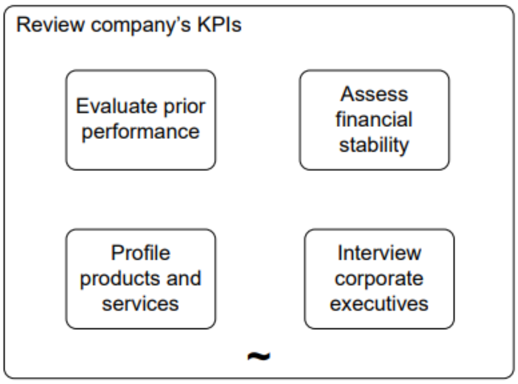


- Parallel repetition




Obtain Quote from Supplier A
Obtain Quote from Supplier B
Obtain Quote from Supplier C
Obtain Quote from Supplier D
Obtain Quote from Supplier E
Select best quote
Place PO
...
cardinality
For each supplier
...
Obtain Quote
Select best quote
Place PO
...

- Uncontrolled repetition: ad hoc subprocesses


Review company's KPIs
Evaluate prior performance
Assess financial stability
Profile products and services
Interview corporate executives
~

3. Advanced Process Modelling

A. Process Modelling with BPMN: events

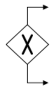
- Event types
    - Start vs intermediate vs end

    - Typed vs untyped

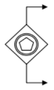
    - Throwing vs catching

    - Interrupting vs non-interrupting

- Event-based decisions
    - Xor-split gateway
        - Data-driven
        - Event-driven



data-driven XOR-split



event-driven XOR split

- Boundary events
    - Interrupting
        - Double border
        - Follow this task and stop the normal flow

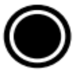
    - Non-interrupting
        - Trigger a task in parallel to the normal flow
        - Double dashed border

- Event sub-processes
    - Process attached to parent process
    - Alternative of putting a boundary non-interrupting event around the parent process

B. Process Modelling with BPMN: exception handling

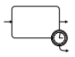
I. Terminate event

- If there is a negative outcome
- It forces the whole process to abort
- All tokens left will stop

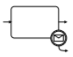


II. Exception handling with boundary events

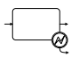
- Stopping an activity and performing a special activity
- Types



    - Timeout
        - Activity takes too long and must be interrupted



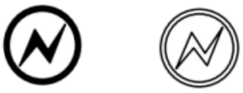
    - External
        - Something goes wrong outside the process, current activity must be interrupted



    - Internal
        - Something goes wrong inside an activity, whose execution must thus be interrupted

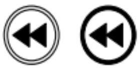
→ Stop enclosing event but start an exception handling routine = interrupting

- Error events
  - End error event
    - Stops the enclosing subprocess exection
  - Intermediate error event
    - Process execution will continue after the error

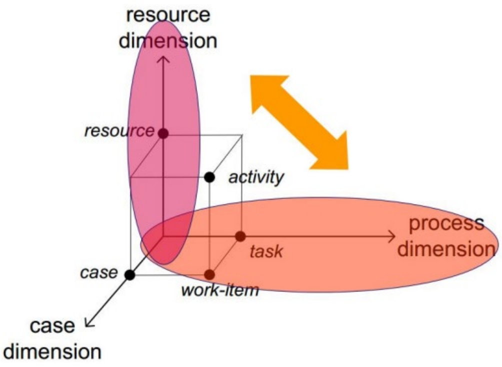


III. Compensation handling

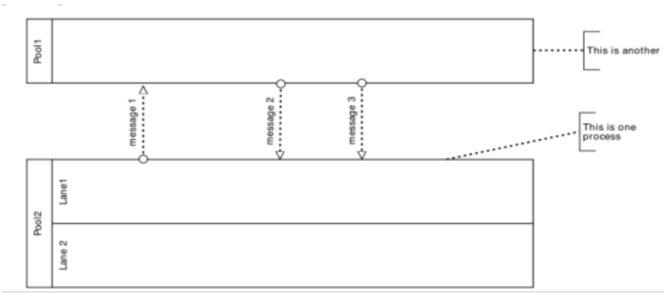
- Part of an exception handling procedure
- Rollback of completed process activities



C. Resource modelling

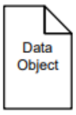


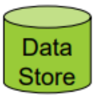
- Making abstractions
  - Resource: human actor that is required to perform an activity
  - Resource class: set of resources
- A resource class
  - Role: skill, competence, qualification
  - Group: department, team, official, organizational unit
  - Pools – independed organizational entities
  - Lanes – resource classes in the same space and sharing common systems

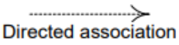


D. Data modelling

- Data objects
  - How data is required or produced by activities
- Data stores
  - Containers of data objects that need be persisted beyond the duration of a process instance
- Associations
  - Used to link artifacts such as data objects and data stores with flow objects

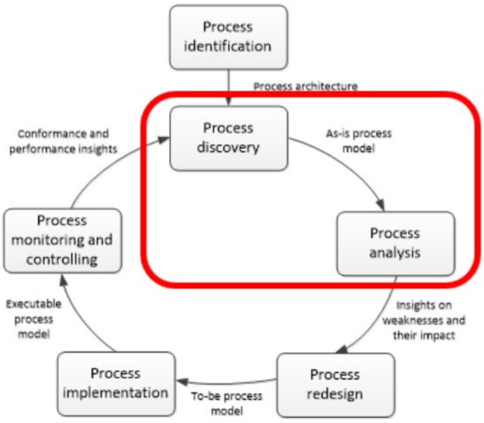


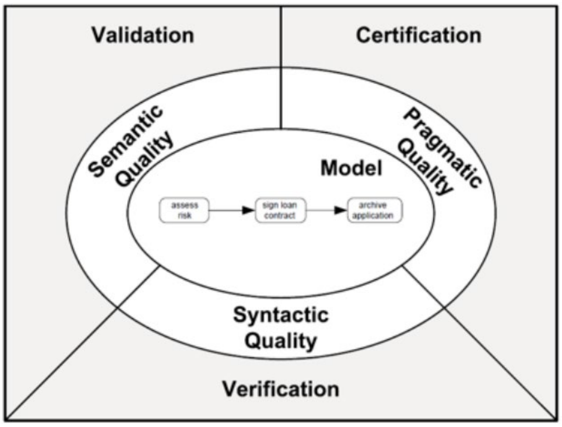




4. Process Model Verification, Validation and Certification

A. Process model quality assurance





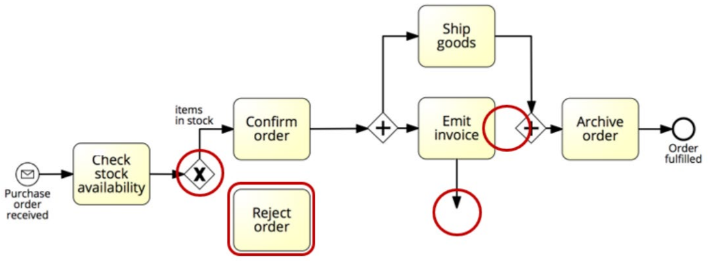
B. Syntactic quality – Verification

- Syntactic quality – are the rules of the modelling language correctly followed?
- 2 problem types
  - Behavioural problems – how the control flow constructs are combined
    - Deadlock

      XOR and then an AND
    - Livelock

      There is an ongoing loop
    - Lack of Synchronization

      A token is left at anywhere in the model after reaching the end
    - Potential Lack of Synchronization
  - Structural problems – direct mistakes against the modelling language's rules



- Verification: soundness of Workflow nets
  - Three requirements for soundness
    1. Option to complete

       There has to be an option to reach the end of the process
    2. Proper completion

       If one token in the end, then the other task are empty
    3. No dead transitions

       Can we execute every path through the Workflow net?

C. Semantic quality – Validation

- 2 Aspects
  - Correctness
  - Completeness

- Validation techniques
  - Interviews
  - Workshops
  - Process mining

D. Pragmatic quality – Certification

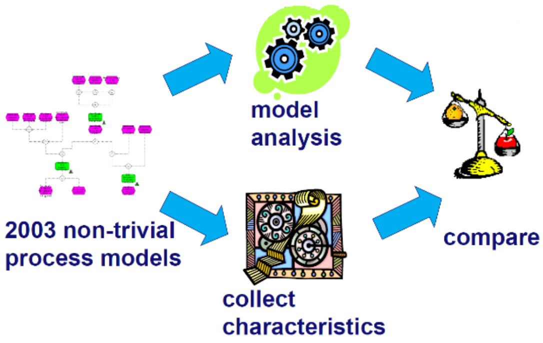
- Dimensions
  - Understandability
  - Maintainability
  - Learning

- Influencing factors
  - Size
  - Structural complexity
  - Graphical layout

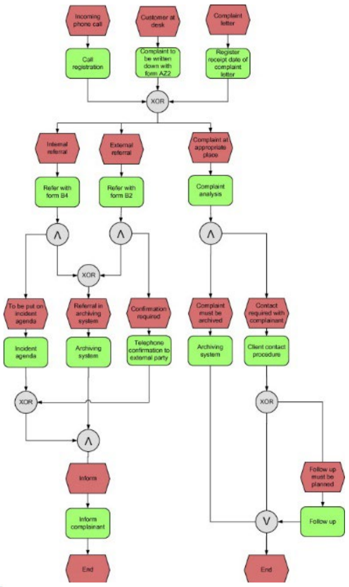
- Formulate labels adequately
  - Activities as verb-object
  - Events as object-passive-participle
  - Conditions with reference to object

- Seven Process Modelling Guidelines (7PMG)
  - G1: Use as few elements in the model as possible
  - G2: Minimize the routing paths per element
  - G3: Use one start and one end event
  - G4: Model as structured as possible
  - G5: Avoid OR routing elements
  - G6: Use verb-object activity labels
  - G7: Decompose a model with more than 50 elements

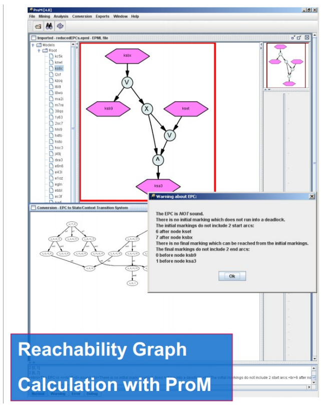
E. Case study: Predicting errors in process models



SAP reference model



- Model analysis
  - EPC Soundness

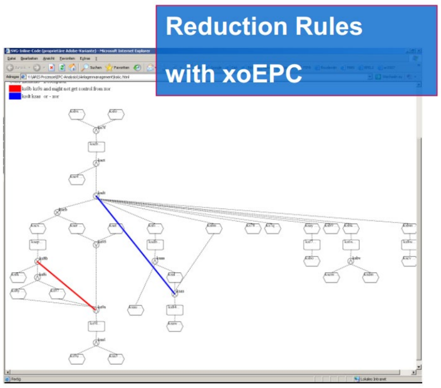


Warning about EPC:

The EPC is *NOT* sound.
There is no initial marking which does not run into a deadlock.
The initial markings do not include 2 start arcs:
6 after node kset
7 after node ksbx
There is no final marking which can be reached from the initial markings.
The final markings do not include 2 end arcs:
0 before node ksb9
1 before node ksa3

Ok

Reachability Graph Calculation with ProM


SVG-Inline-Code (proprietäre Adobe-Variante) - Microsoft Internet Explorer

🟥 ks8b ks9s and might not get control from xor
🟦 ksdt ksas or - xor

Reduction Rules with xoEPC

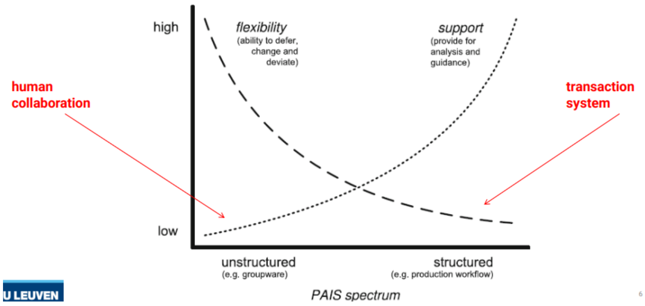
- Model characteristics
  - Structuredness
    - Compares reduced and unreduced graph
    - More structure implies less complexity
  - Connector Heterogeneity
    - Entropy over the different connector types
    - More heterogeneity implies more complexity

5. Declarative Process Modelling

A. Introduction

Business Process Models = collection of related structured activities or tasks that produce a specific service or product for a customer

- Balancing Flexibility and Control

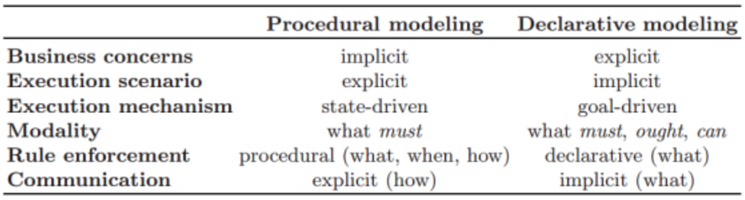


- Procedural or imperative business process modelling paradigm

  = Defining an activity sequence that will result in obtaining the related corporate goal

  - Potential problems
    - Inflexible
    - Overspecification
    - Maintainability
    - Difficulties with demonstrating compliance

- Procedural vs declarative approach



	Procedural modeling	Declarative modeling
Business concerns	implicit	explicit
Execution scenario	explicit	implicit
Execution mechanism	state-driven	goal-driven
Modality	what *must*	what *must*, *ought*, *can*
Rule enforcement	procedural (what, when, how)	declarative (what)
Communication	explicit (how)	implicit (what)

B. Declarative process modelling

→ Focus on capturing and defining regulatory or internal directives in constraints

- Dynamic, goal driven execution
- Assumption bias

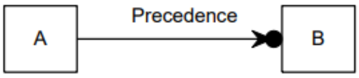
  →Overspecified by choosing one particular sequence

- Languages
  - CMMN
  - DCR - More complex then CMMN
  - Declare – most robust language for declarative models

C. Declare

Declare = body of process model constraints

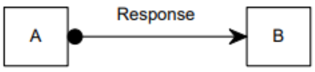
- Different constraints
  - Precedence
    - Activity B, it it occurs, has to be preceded by activity A



Ex: ACAAB, A, CCC, AA

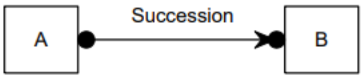
NOT: BACC, BB, CB, CBA

- Response
    - When activity A is executed, activity B has to be executed afterwards



Ex: ACAAAB, ABB, AAAB, ACCBAB, BB
NOT: AABBA, A

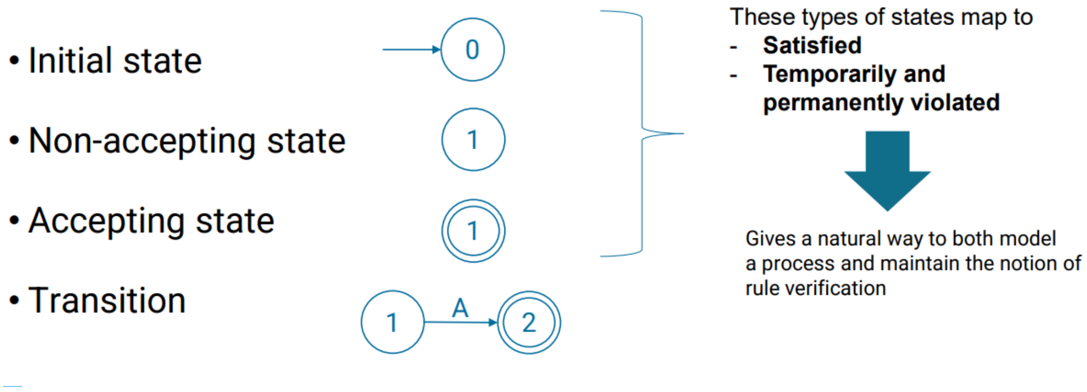
- Succession
    - Combination of Response and Precedence
    - When activity A is executed, activity B has to be executed afterwards eventually.
    - Activity B, if it occurs, has to be preceded by at least one activity A.



Ex: CCC, ABBBAB, AB, AAB, ABB, CACCABB
NOT: ACC, AABBA, ACBCA, BBA, A,B

D. Declare constraints and execution

I. Execution



- Initial state
- Non-accepting state
- Accepting state
- Transition

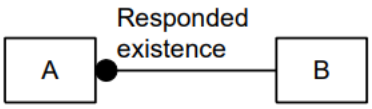
These types of states map to
- **Satisfied**
- **Temporarily and permanently violated**

Gives a natural way to both model a process and maintain the notion of rule verification

II. Constraints

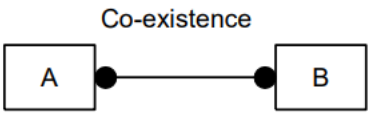
- Body of constraints: structured and hierarchical
- Unary constraints
    - Position-related
    - Numerosity-based
        - Absence(A,n+1): A can occur at most n times
        - Existence(A,n): A has to occur at least n times
        - Exactly(A,n): A has to occur exactly n times (Existence+Absence)
- Binary constraints
    - Position agnostic
        - Responded existence

          When A occurs, B has to occur as well



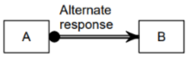
        - Co-existence

          When A occurs, B has to occur and vice versa

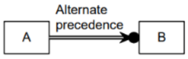


- Simple ordered
  (see above)

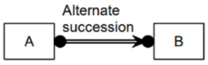
- Alternating ordered
  - Alternate response
    - Every occurrence of A has to be followed by a new occurrence of B
      Ex: ABAB, BBABBAB, ACCCBAB, CCB
      NOT: AABB, CA, A, ACAABB



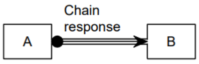
  - Alternate precedence
    - B can occur only after the next occurrence of A
      Ex: AAA, CCC, AC, ABAA, ABACCBA, C
      NOT: ABB, CB, ABBA



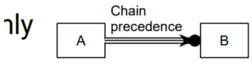
  - Alternate succession
    - Both alternate precedence and alternate response hold
      Ex. CCC, AB, ACCB, ACCBCCACCB
      NOT: ACC, AABBA, ACBCA, BBA, A, B



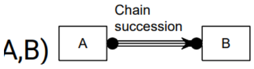
- Chain ordered
  - Chain Response
    - Right after an occurrence of A, only B can occur
      Ex. ABABBB, BB, CC, ABCBCCABB
      NOT: ACB, CA, CCAC



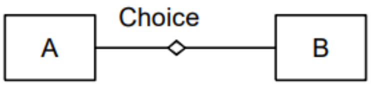
  - Chain Precedence
    - B can occur only right after A
      Ex. AAA, CCC, AC, ABAA, C
      NOT: AABB, CB, B, ACAABB, ABAAACCBA



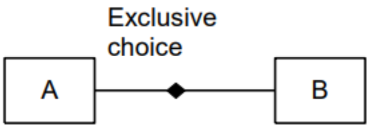
  - Chain Succession
    - Both Chain precedence and chain response
      Ex. CCC, AB, CCABC
      NOT: ACC, AABBA, ACBCA, BA, A, B



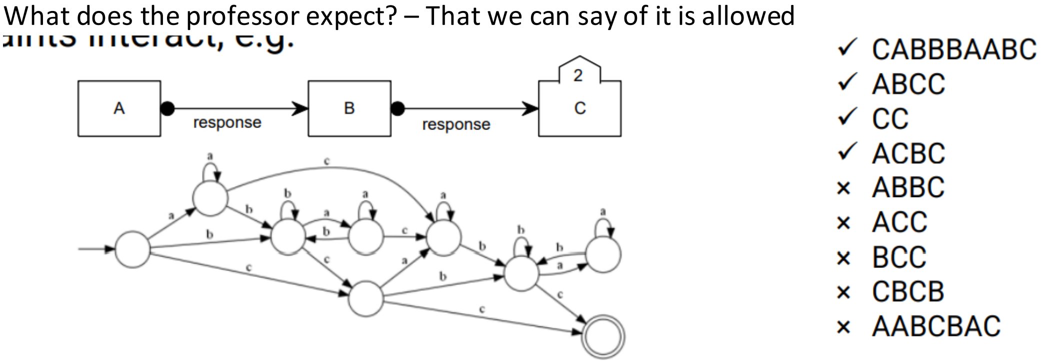
- Choice
  - Choice
    - Either A or B has to occur, or both
      Ex. ABAA, ABAAB, CCA, B, A, BC
      NOT: CCC



  - Exclusive Choice
    - A and B cannot occur together, but at least one has to occur
      Ex. AAA, AC, CBBB, B
      NOT: AABB, BA, ACABB, CC



- What does the professor expect? – That we can say of it is allowed



✓ CABBBAABC
✓ ABCC
✓ CC
✓ ACBC
× ABBC
× ACC
× BCC
× CBCB
× AABCBAC

E. Conclusion

- Two different business process modelling paragdigms
  - Procedural vs. declarative
- When flexibility is required, the declarative approach can be preferred
  - Case management systems
  - Knowledge Intensive Processes
- Several declarative modelling languages exist
- Adoption in practice has been slow

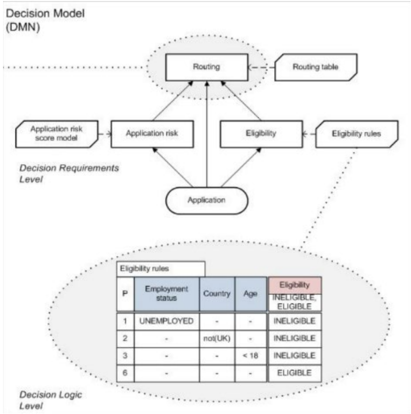
6. Decision Modelling with DMN

A. Process modelling and decisions

- Operational decisions
  - = Daily, high-volume, standard procedures
  - Made frequently
  - Non-trivial
  - Made rapidly
  - Made consistently
  - High volume
  - Measurable business impact
  - Deterministic
  - Frequent change
  - Comprehensibility
  - Automated or manual

- Decisions in processes
  - Inside a knowledge-intensive activity
  - As shown by a simple gateway
  - As shown by a cascade of gateways

→ separate the business rule from the process → has deciding logica → DMN



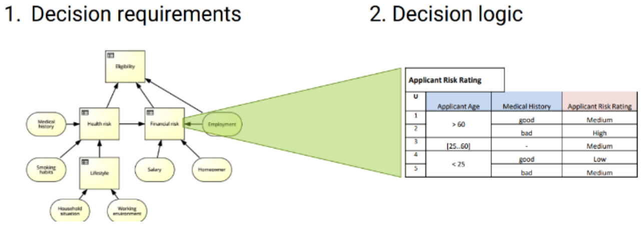
Eligibility rules				
P	Employment status	Country	Age	Eligibility INELIGIBLE, ELIGIBLE
1	UNEMPLOYED	-	-	INELIGIBLE
2	-	not(UK)	-	INELIGIBLE
3	-	-	< 18	INELIGIBLE
6	-	-	-	ELIGIBLE

B. Decision Model and Notation (DMN)

= modelling language for the precise specification of business decisions and business rules

- Can be automated using a Business Rules Management System (BRMS), they should be:
  - Well-designed
  - Correct
  - Consistent
  - Explainable
  - Understandable by the business
  - Easy to change
  - Maintained by the business

- Two modelling levels
  - Decisions Requirements level
  - Decision Logic level


1. Decision requirements

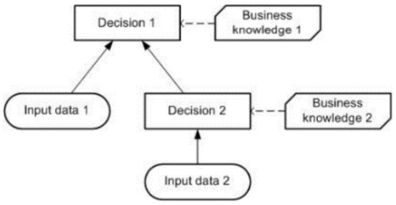
2. Decision logic

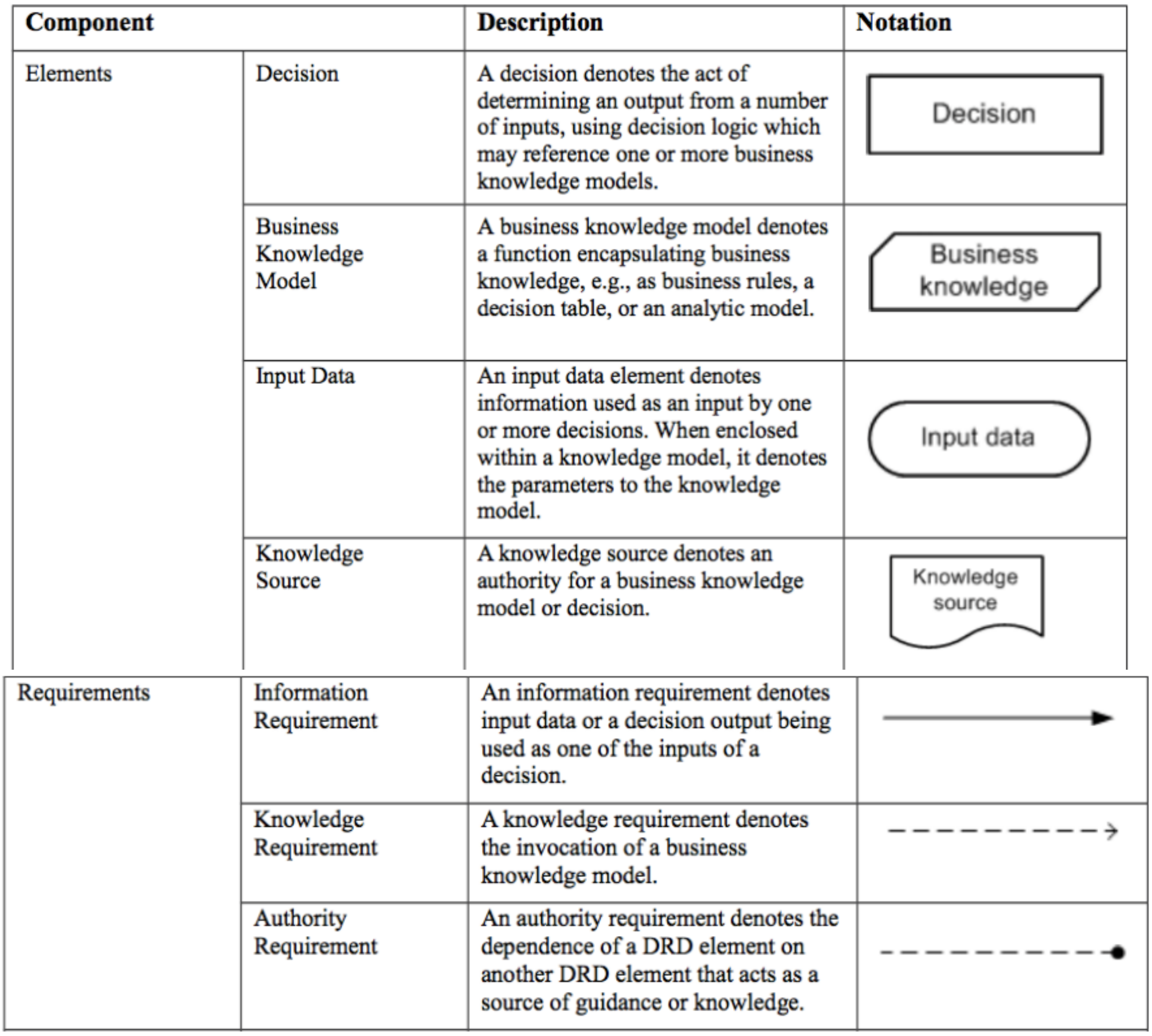
Applicant Risk Rating			
U	Applicant Age	Medical History	Applicant Risk Rating
1	> 60	good	Medium
2		bad	High
3	[25..60]	-	Medium
4	< 25	good	Low
5		bad	Medium

- Major elements of DMN
  - Decision Requirements Diagram
    - What do we need to make a decision: goal oriented requirements
  - The Decision rules
    - Logic behind each decision
  - The FEEL
    - How to define functions, operators and expressions

C. Decision requirements modelling

- **WHAT** do we need in order to make a decision?
  - Decision Requirements Diagam (= DRD)





Component		Description	Notation
Elements	Decision	A decision denotes the act of determining an output from a number of inputs, using decision logic which may reference one or more business knowledge models.	Decision
	Business Knowledge Model	A business knowledge model denotes a function encapsulating business knowledge, e.g., as business rules, a decision table, or an analytic model.	Business knowledge
	Input Data	An input data element denotes information used as an input by one or more decisions. When enclosed within a knowledge model, it denotes the parameters to the knowledge model.	Input data
	Knowledge Source	A knowledge source denotes an authority for a business knowledge model or decision.	Knowledge source
Requirements	Information Requirement	An information requirement denotes input data or a decision output being used as one of the inputs of a decision.	──────►
	Knowledge Requirement	A knowledge requirement denotes the invocation of a business knowledge model.	- - - - - - - - ->
	Authority Requirement	An authority requirement denotes the dependence of a DRD element on another DRD element that acts as a source of guidance or knowledge.	- - - - - - - - - ●

- Decisions Require
  - Input data
    - Transactions
    - Master data
    - External data
  - Decision logic
    - Rules, knowledge
    - Policies
    - Analytics
  - Outcome of other decisions
    - Reusability

D. DMN decision logic

- Boxed expressions
    - Implementation of the value expression
    - 3 types
        - Decision tables

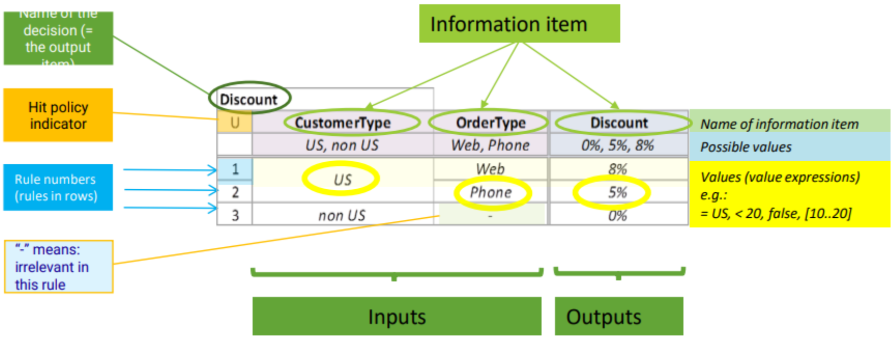
          Logic based on rules
        - Literal expressions

          Formulas such as arithmetic
        - Invocations

          Leveraging decision logic form elsewhere

- How to write decision logic?
    - Natural language
    - Logic
    - Structured English rules
    - Decisions trees, tables, graphs, diagrams
    - Object Contract Language
        - UML

E. Decision tables



Name of the decision (= the output item)
Hit policy indicator
Rule numbers (rules in rows)
"-" means: irrelevant in this rule
Information item
Name of information item
Possible values
Values (value expressions) e.g.: = US, < 20, false, [10..20]
Inputs
Outputs

Discount

U	CustomerType	OrderType	Discount
	US, non US	Web, Phone	0%, 5%, 8%
1	US	Web	8%
2		Phone	5%
3	non US	-	0%

"-" means: does not matter in this rule (irrelevant)

- Problems with lists of rules
    - Order of information items is not the same in all rules
    - Name of information items is repeated in many rules
    - Connectors can be and, or, with parentheses,…
    - Rules are complex, hard to understand and hard to validate
- Decision tables will solve these problems
    - Order of information items will be the same in all rules
    - The name of information items is only written once
    - Connectors will only be AND
    - The collection of rules will be easy to validate

- Hit policies
    - Single hit – Return 1 rule with outcome
        - Default: rules are non-overlapping = unique hit (U)
        - Recognize others: rules are overlapping, 1 rule has to be selected

          - **A**ny: outcomes are equal

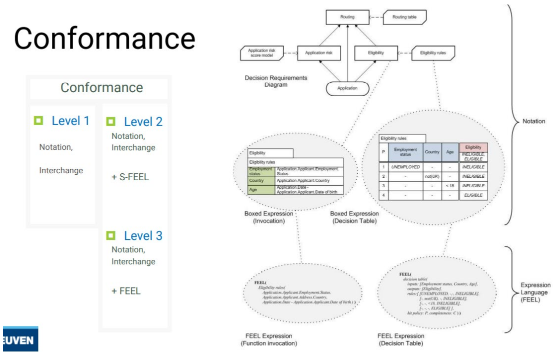
          - **F**irst: first hit by rule order is returned

          - **P**riority: outcome with highest output value priority
    - Multiple hit – Return a list of rules

- What makes a good decision table?
    - Completeness
    - Consistency
    - Avoid subsumption

F. Expression language for decision logic (FEEL)

- S-Feel = basic subset of FEEL designed to cover the essential requirements of Decision Table based DMN models
- Similar to functions in excel



Conformance

Conformance

Level 1
Notation,
Interchange

Level 2
Notation,
Interchange
+ S-FEEL

Level 3
Notation,
Interchange
+ FEEL

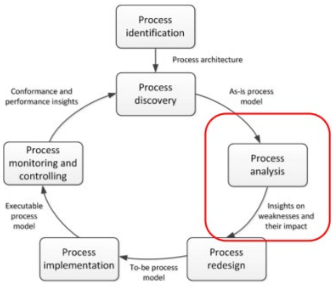
Routing
Routing table
Application risk score model
Application risk
Eligibility
Eligibility rules
Decision Requirements Diagram
Application
Eligibility
Eligibility rules
Employment status
Application.Applicant.Employment.Status
Country
Application.Applicant.Country
Age
Application.Date - Application.Applicant.Date of birth
Boxed Expression (Invocation)
Eligibility rules
P
Employment status
Country
Age
Eligibility
INELIGIBLE, ELIGIBLE
1
UNEMPLOYED
INELIGIBLE
2
not(UK)
INELIGIBLE
3
< 18
INELIGIBLE
4
ELIGIBLE
Boxed Expression (Decision Table)
Notation
FEEL(
Eligibility rules(
Application.Applicant.Employment.Status,
Application.Applicant.Address.Country,
Application.Date - Application.Applicant.Date of birth))
FEEL(
decision table(
inputs: [Employment status, Country, Age],
outputs: [Eligibility],
rules:[[UNEMPLOYED, -, -, INELIGIBLE],
[-, not(UK), -, INELIGIBLE],
[-, -, <18, INELIGIBLE],
[-, -, -, ELIGIBLE]],
hit policy: P, completeness: C))
Expression Language (FEEL)
FEEL Expression (Function invocation)
FEEL Expression (Decision Table)
EUVEN

G. Conclusion

- Issues DMN solves
    - Separating decisions and processes
    - Separating decision structure and decision logic
    - Decision modelling methodology
        - Consistency
        - Completeness
        - Correctness
    - Decision tables types
- Application areas & tools
    - Decisions
    - 'Simple' operational decisions
    - Enumerating and evaluating

7. Qualitative Process Analysis

A. Introduction



- Process Analysis Techniques
  - Qualitative Analysis

    More subjective

    - Value-Added Analysis & Waste Analysis
    - Issue Documentation

      - Issue Register
      - Root-Cause Analysis
      - Pareto Analysis – PICK charts
  - Quantitative Analysis

    More objective

    - Quantitative Flow Analysis
    - Queuing Theory
    - Process Simulation

B. Value-added analysis & waste analysis

I. Value-added Analysis

- Decorticate the process into steps
  - Steps performed before a task
  - The task itself
  - Steps performed after a task

- Classify each step into:
  - Value adding (VA)
    - Produce value or satisfaction to the customer
  - Business value adding (BVA)
    - Necessary or useful for the business to operate
  - Non value adding (NVA)
    - Everything else besides VA and BVA
    - Activities the customer would be unwilling to pay for

II. Waste Analysis

- Muda: Eliminating waste
- 7+1 sources of waste

1. Unnecessary Transportation (send, receive)
2. Motion (drop-off, pick-up, go to)

Move

3. Inventory (large work-in-process)
4. Waiting (waiting time between tasks)

Hold

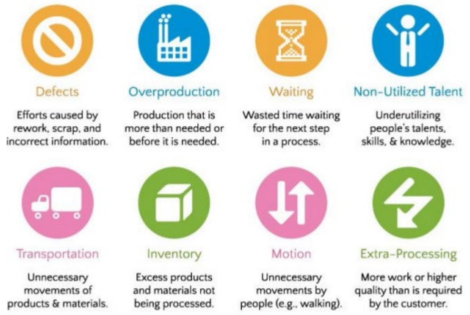
5. Over-Processing (performing what is not yet needed or might not be needed)
6. Over-Production (unnecessary cases)
7. Defects (rework to fix defects)

Over-do

8. Resource underutilization (waste of intellect)

D.O.W.N.T.I.M.E

C.

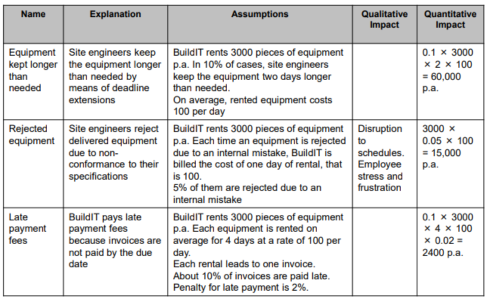

Defects
Efforts caused by rework, scrap, and incorrect information.
Overproduction
Production that is more than needed or before it is needed.
Waiting
Wasted time waiting for the next step in a process.
Non-Utilized Talent
Underutilizing people's talents, skills, & knowledge.
Transportation
Unnecessary movements of products & materials.
Inventory
Excess products and materials not being processed.
Motion
Unnecessary movements by people (e.g., walking).
Extra-Processing
More work or higher quality than is required by the customer.

Issue

documentation

I. Issue Register

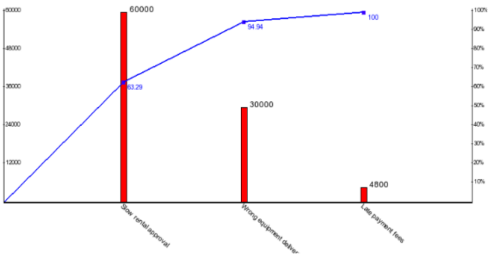
- Categorise identified issues as part of as-is process modelling
- Table with following columns:
  - Issue number
  - Name
  - Description
  - Impact (qualitative vs quantitative)
  - Possible solution



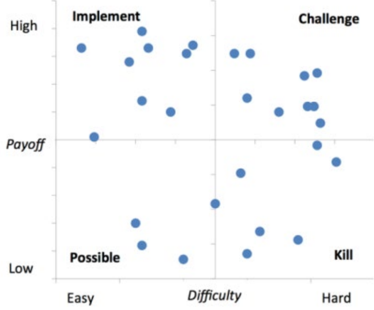
Name	Explanation	Assumptions	Qualitative Impact	Quantitative Impact
Equipment kept longer than needed	Site engineers keep the equipment longer than needed by means of deadline extensions	BuildIT rents 3000 pieces of equipment p.a. In 10% of cases, site engineers keep the equipment two days longer than needed. On average, rented equipment costs 100 per day		0.1 × 3000 × 2 × 100 = 60,000 p.a.
Rejected equipment	Site engineers reject delivered equipment due to non-conformance to their specifications	BuildIT rents 3000 pieces of equipment p.a. Each time an equipment is rejected due to an internal mistake, BuildIT is billed the cost of one day of rental, that is 100. 5% of them are rejected due to an internal mistake	Disruption to schedules. Employee stress and frustration	3000 × 0.05 × 100 = 15,000 p.a.
Late payment fees	BuildIT pays late payment fees because invoices are not paid by the due date	BuildIT rents 3000 pieces of equipment p.a. Each equipment is rented on average for 4 days at a rate of 100 per day. Each rental leads to one invoice. About 10% of invoices are paid late. Penalty for late payment is 2%.		0.1 × 3000 × 4 × 100 × 0.02 = 2400 p.a.

II. Pareto chart

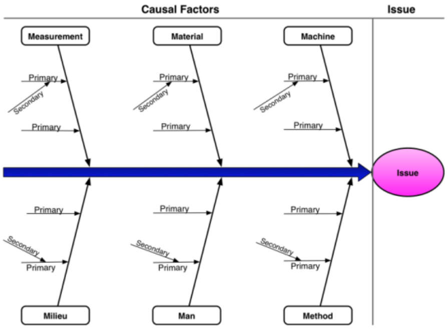
- Useful to prioritize a collection of issues or factors behind an issue
- Y-axis represents the cumulative percentage impact



III. PICK Chart – two dimensional prioritization

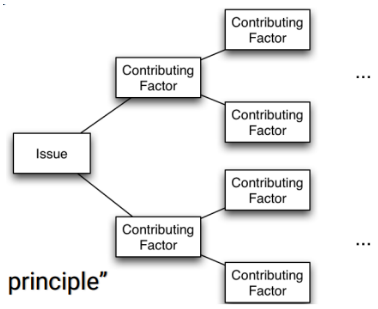


D. Root cause analysis



- Six Ms
1. Machine
2. Method
3. Material
4. Man
5. Measurement
6. Milieu

- Why-why diagram



- 5 levels deep

E. Conclusion

- Qualitative process analysis includes
    - Segregate value-adding, business value-adding and non-value-adding steps (value-added analysis)
    - Identify waste (waste analysis)
    - Collect and systematically organize issues, assess their impact (issue register, Pareto chart, PICK chart)
    - Analyze root causes of issues (fishbone diagram, why-why diagram)

8. Quantitative Process Analysis

A. Introduction

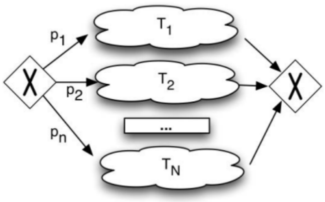
(Same as introduction of qualitative analysis)

- Process performance measures
    - Cost
        - Cost per execution
        - Resource utilization
        - Waste
    - Time
        - Cycle time
        - Waiting time/time spent in non-value added tasks
    - Quality
        - Error rates
        - Missed promise

B. Flow Analysis

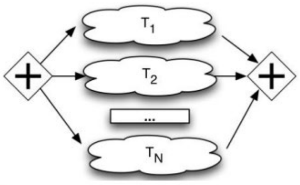
I. Cycle Time analysis

- **Cycle time (CT)** = difference between a job's start and end time
- **Cycle time analysis** = calculating the average cycle time for an entire process
- Cycle time for:
    - Alternative paths



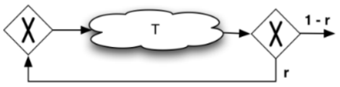
$$CT = p_1T_1+p_2T_2+\ldots+p_nT_n = \sum_{i=1}^{n} p_i T_i$$

-
    - Parallel paths



$$CT_{parallel} = \mathrm{Max}\{T_1, T_2, \ldots, T_M\}$$

-
    - Rework



➡ $CT = T/(1-r)$

II. Cycle time efficiency

$$\text{Cycle Time Efficiency} = \frac{\text{Theoretical Cycle Time}}{CT}$$

- Theoretical cycle time (TCT) = cycle time if we only counted value-adding activities excluded any waiting time
  → Count only processing times

III. Work-in-Process: Little’s Law

$$\mathbf{WIP = \lambda \cdot CT}$$

- WIP = Work in process = number of cases that are running
- λ = arrival rate = number of new cases per time unit
- CT = cycle time

IV. Flow analysis: other angles and limitations

→Limitation 1: not all models are structured
→Limitation 2: Fixed load + fixed resource capacity
Cycle time analysis does not consider waiting times due to resource contention

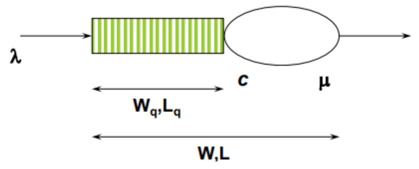
C. Queuing analysis

- Why is queuing analysis important?
  - Capacity problems are very common →need to balance the cost of increased capacity against the gains of increased productivity and service
  - Important in service systems → large costs of waiting and lost sales due waiting

- Delay is caused by job interference
  - Deterministic traffic
    - Each activity has the same time length
  - Variable but spaced apart traffic
    - Each activity has another time length but there is no overlap

- Causes of job interference
  - Bursty interarrival intervals
  - Job size variation
  - High utilization
    System close to 100% is unstable

- Queuing theory
  - Basic concepts



    - λ = mean arrival rate = average number of arrivals per time unit
    - μ = mean service rate = average number of jobs handled by one server per time unit
    - c = number of servers
  - Given above parameters we can calculate:
    - ρ = occupation rate
    - W_q = average time in queue
    - W = average time in system
    - L_q = average number in queue
    - L = average number in system average

- Systems
  - M/M/1
    - Times between arrivals and service time follow negative exponential distribution
    - Poisson process

      Times between arrivals are independent and identically distributed and exponential
    - 1 server
    - FIFO

$$\rho = \frac{\text{Capacity Demand}}{\text{Available Capacity}} = \frac{\lambda}{\mu}$$

$$L=\rho/(1-\rho) \qquad W=L/\lambda=1/(\mu-\lambda)$$

$$L_q=\rho^2/(1-\rho)=L-\rho \qquad W_q=L_q/\lambda=\lambda/(\mu(\mu-\lambda))$$

  - M/M/c
    - C servers → expected capacity per time unit is c * μ

$$\rho = \frac{\text{Capacity Demand}}{\text{Available Capacity}} = \frac{\lambda}{c*\mu}$$

    - You need tools to calculate the rest

- Limitations of basic queuing models
  - Cannot be used to analyse cost or quality measures
  - Not suitable for analysing end-to-end processes
  - Not applicable when system includes parallel activities
  - Assumes steady state

D. Simulation

Process simulation = run a large number of process instances and calculate statistics from the output

- Steps in evaluating a process with simulation
  1. Model the process
  2. Enhance the process model with simulation info
  3. Run the simulation
  4. Analyse the simulation outputs
  5. Repeat for alternative scenarios

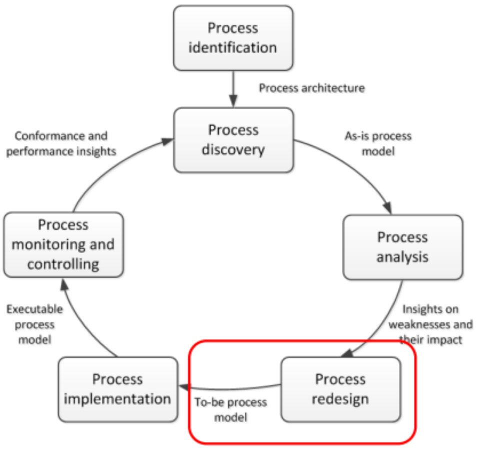
- Elements of a simulation model
  - The process model including:
    - Events, activities, control-flow relations
    - Resource classes
  - Resource assignment
    - Mapping from activities to resource classes
  - Processing times
    - Per activity or per activity resource pair
  - Costs
    - Per activity and/or per activity resource pair
  - Arrival rate of process instances
  - Conditional branching probabilities

- Difficulties of simulation
  - Numerous choices to be made
  - Simplifying assumptions necessary
  - Reliability is largely dependent on accuracy of the inputs
  - Multiple validation loops necessary with process stakeholders to verify validity
  - Sensitivity analysis

E. Conclusion

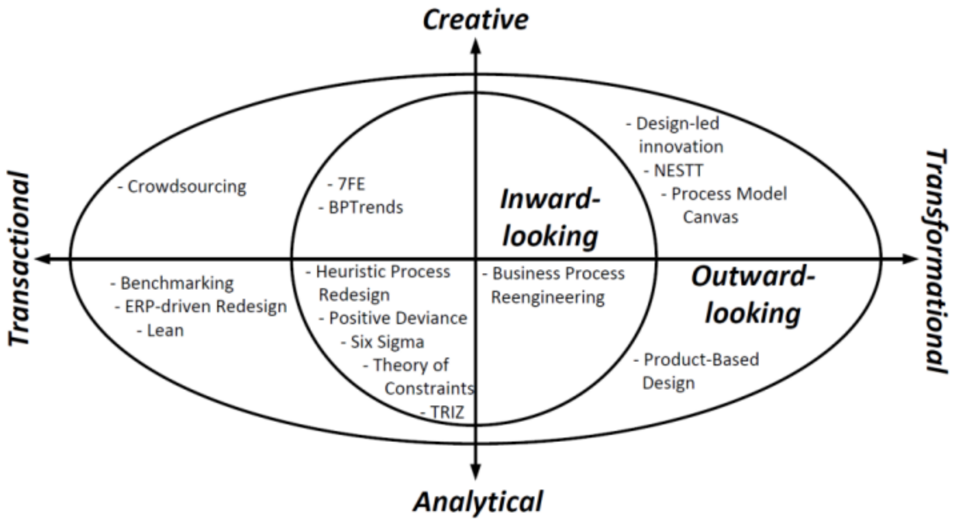
- Processes can be studied from a quantitative perspective
    - Flow analysis
    - Queuing theory
    - Simulation
- All these methods have advantages and disadvantages
- More versatile simulation techniques and tools available
    - Discrete Event Simulation
        - Arena
        - CPN Tools

9. Process Redesign



A. Introduction

- Process redesign
    - Identify possibilities for improving the design of a process
    - “as is” → “to be”
    - Methods



- Explorative redesign = transformational
    - Fundamental assumptions
    - Start from scratch
- Exploitative redesign = transactional
    - Identify problems and solve those problems
    - Current process stays the same

B. Business Process Reengineering (BPR)

- Analytical + Transformational
- Example of Ford
    - First Accounts payable had to check and compare 3 documents
    - With a central database, she just have to take the data she needs
- Some principles of BPR
    - Capture information once and at the source
        - Shared data store
            - All process workers access the same data
        - Self-service
            - Customers capture data themselves

- Subsume information-processing work into the real work that produces the information
  - Evaluated receipt settlement

- Have those who use the output of the process drive the process
  - Vendor-managed inventory
  - Scan-based trading
  - Push work to the actor that has the incentive to do it

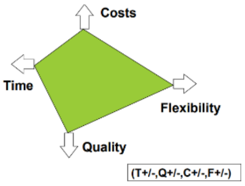
- Put the decision point where the work is performed, and build control into the process
  - Empower the process workers
  - Provide them with information needed to make decisions
  - Replace back-and-forth handovers

- Treat geographically dispersed resources as though they were centralized
  - Integrate and share work where people are doing the same function on different locations
  - Larger resource pools → less waiting times even with high utilization

C. Heuristic process redesign

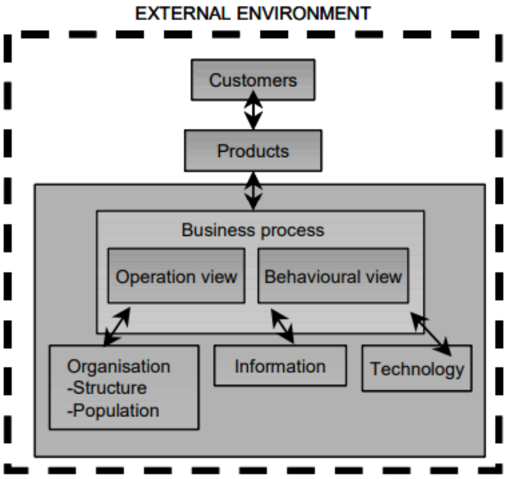
- Transactional
- Inward-looking
- Analytical

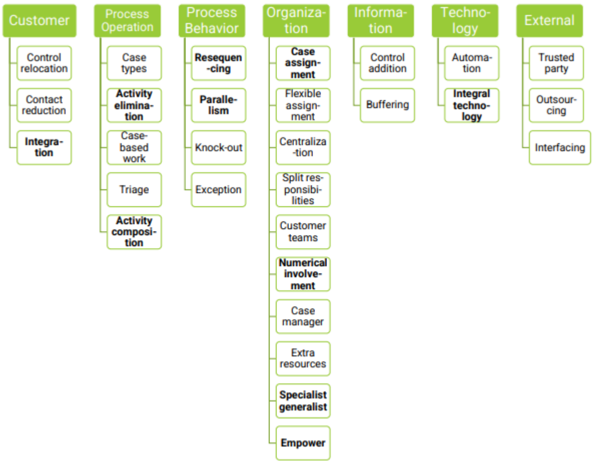
Devil's Quadrangle



- Cost
  - Fixed or variable
  - Human, system or external
  - Processing, management or support
- Time
  - Service time
  - Transport time
  - Waiting time
- Quality
  - External – Satisfaction of the customer
    - Product
    - Process (service level)
  - Internal – Conditions of work
    - Challenging
    - Varying
    - Controlling
- Flexibility
  - The ability to react to changes
    - Resources
    - Process
    - Management
    - Organization

- Heuristic redesign framework





- Most popular redesign heuristics

  You have to know them and be able to make an analyses

  It is not a low, but a logica/expectations

1. Task elimination
   - T+, C+, Q-

2. Integral technology
   - Applying new technology
     - Purchase, development, training might be costly
     - Workers might be reluctant → less quality
   - T+, C-

3. Task composition
   - Small tasks into composite tasks
     - Bigger blocks → less flexible to optimal structure
   - Large tasks into workable tasks
   - T+, C+, F-

4. Parallelism
   - One of the most important ones
   - More parallelism → improved performance
     - More complex to manage
     - Higher costs if there is a knock-out
   - T+, C-, F-

5. Specialist – generalist
   - More specialists
     - Faster processing of activities
     - Higher quality

       → T+, Q+, F-
   - More generalists
     - More flexibility
     - Better utilization

       → T+,Q-, F+

6. Resequencing
    - Order tasks on cost/effect
    - T+, C+

7. Integration
    - More integrated workflows improves efficiency
    - Flexibility may decrease because of mutual dependence
    - T+, C+, F-

8. Empower
    - Gives workers most of the decision making authority
        - Smoother operations with lower time
        - If bad decisions, more rework, increasing cost
    - T+, Q-, F+

9. Numerical Involvement
    - Most used
    - Minimize number of departments
        - Less coordination problems
        - Fewer resources
    - T+,F-

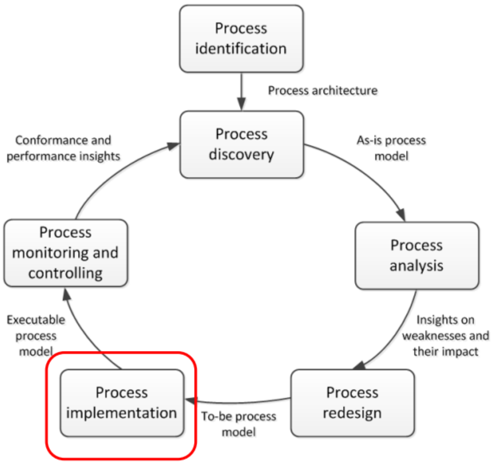
10. Case assignment
    - Let workers perform as many steps as possible
        - Decrease set-up time
        - Improve quality
        - Flexibility reduce
    - Q+, F-

D. Conclusion

- Redesign is difficult, never final, but crucial for many organizations
    - No silver bullet!
- Redesign inherently means “making trade-offs”

10. Process Automation

A. Introduction



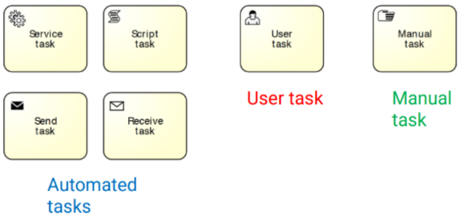
= the well-known gap → find a translation

- Conceptual ‘to-be’ process models <-> Executable process models

Conceptual ‘to-be’ process models	Executable process models
- Domain experts	- IT experts
- Understandable	- BPMS
- Basis for communication with stakeholders	- Machine readable
- Intuitive	- Unambiguous
- Purely a relevant set of process info	- Further details relevant to implement

B. Bridging gap: a five step approach

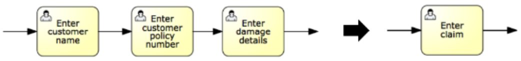
1. Identify the automation boundaries
   - Not all processes can be automated
   - Task type:
     - Automated tasks
       - Service task → invoke externally
       - Descript task → invoke internally
     - User tasks
     - Manual tasks



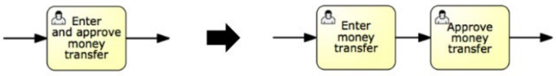
2. Review manual tasks
   - If it can’t be seen by the BPMS, it doesn’t exist
   - Find ways to support manual task via IT
     → Isolate them and automate the rest

3. Complete the process model
   - Exceptions are the rule
   - No data = no decisions, no tasks handover

4. Adjust task granularity
   - BPMSs add value if they coordinate handovers of work between resources
     - Aggregate any two consecutive tasks assigned to the same resource



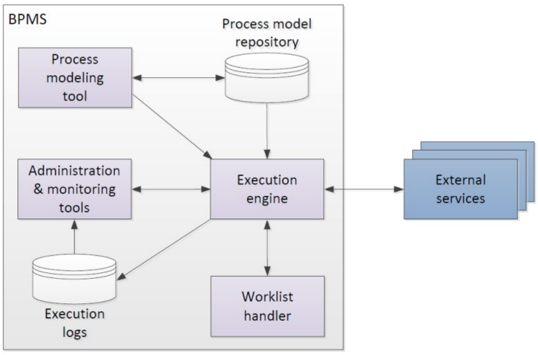
     - Refine tasks that are too coarse-grained



5. Specify execution properties
    - Process variables, messages, signals, errors
    - Task and event variables and their mappings to process variables
    - Service details
    - Code snippets
    - Participant assignment rules and user interface structure
    - Task, event and sequence flow expressions
    - BPMS-specific: work queues, forms, connectors…

C. Business Process Management Systems (BPMS)

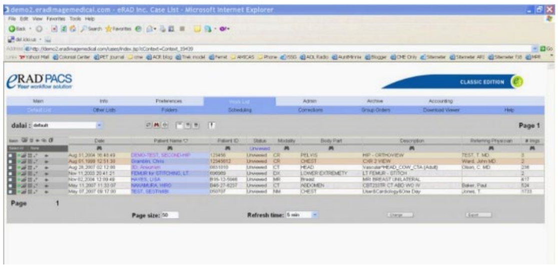
- General architecture



I. Execution Engine

- Instantiates executable process models
- Logs execution data
- Stays in the same order
- Who will do the tasks?

II. Worklist handler


demo2.eradimagemedical.com - eRAD Inc. Case List - Microsoft Internet Explorer

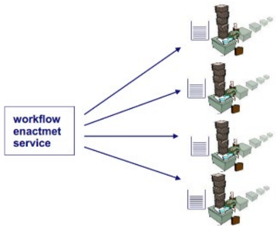
dalai : default   Page 1

Date	Patient Name	Patient ID	Status	Modality	Body Part	Description	Referring Physician	# Imgs
Aug 31,2004 16:48:49	DEMO-TEST, SECOND-HIP	123456	Unviewed	CR	PELVIS	HIP - ORTHOVIEW	TEST, T. MD	3
Aug 01,1999 12:51:38	Grambin, Chris	12345612	Unviewed	CR	CHEST	CXR 2 VIEW	Ward, John MD	2
Aug 28,2007 02:12:00	3D, Anuurism	0831010	Unviewed	CT	HEAD	Vascular^HEAD_COW_CTA (Adult)	Olson, C. MD	236
Nov 11,2003 20:41:21	FEMUR for STITCHING, LT	696969	Unviewed	DX	LOWER EXTREMETY	LT FEMUR - STITCH		2
Nov 02,2004 12:09:49	HAYES, LISA	B16-12-5046	Unviewed	MR	Breast	MRI BREAST UNILATERAL		417
May 11,2007 11:33:07	NAKAMURA, HIRO	B46-27-8237	Unviewed	CT	ABDOMEN	CBT233TR CT ABD WO IV	Baker, Paul	524
May 07,2007 09:17:00	TEST, SESTIMIBI	050707	Unviewed	NM	CHEST	Users\Cardiology\One Day	Jones, T	1733

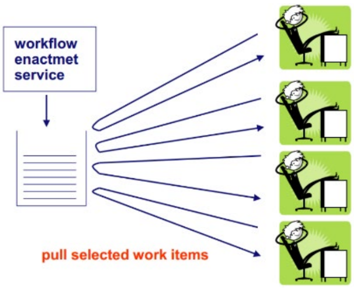
Page 1

Page size: 50   Refresh time: 5 min

- "inbox"
- Offers work items to process participants
- Handles participants' work queues and work item priorities
- Can have social media capabilities
- Push vs pull



- Push: select a 'victim'
    - Everything is decided by the systems, resources cannot say anything



- Pull: resources decide
    - Decide by people

III. Administration & Monitoring Tools

- To manage automation solutions
- To configure access to system components
- To monitor participants availability and performance of process

IV. External Services

- Expose a service interface with which the engine can interact

→ BPMS classification according to BPMN support

1. Pure BPMN

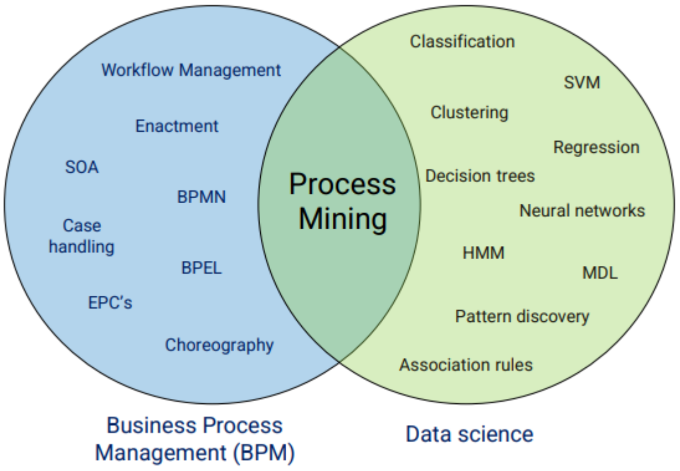
   Designed from the ground to follow the spec to the letter → exact BPMN
2. Adapted BPMN

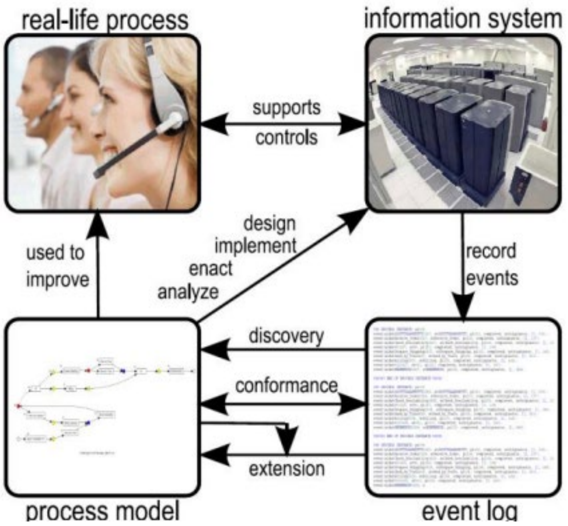
   Use a BPMN skin but rely on internal representation
3. Non BPMN

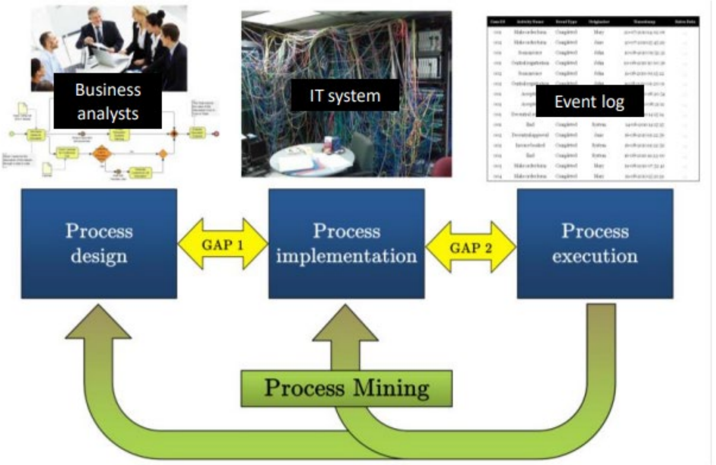
   Other language

11. Process Mining Process Discovery

A. Introduction

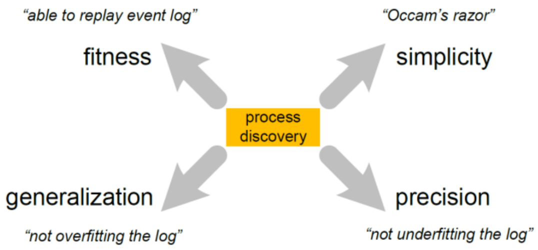






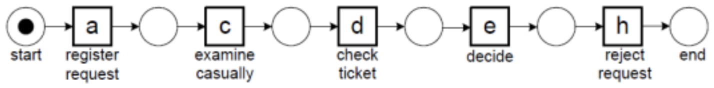
B. Process discovery

- Quality Trade-off: four dimensions



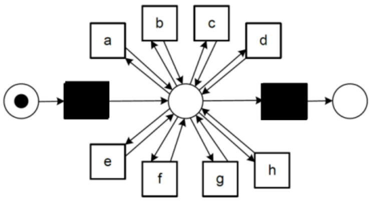
"able to replay event log"
fitness
"Occam's razor"
simplicity
process discovery
generalization
"not overfitting the log"
precision
"not underfitting the log"

  - Fitness; everything is covered in the model
  - Generalization; allow new cases that were not in the training data – avoid overfitting



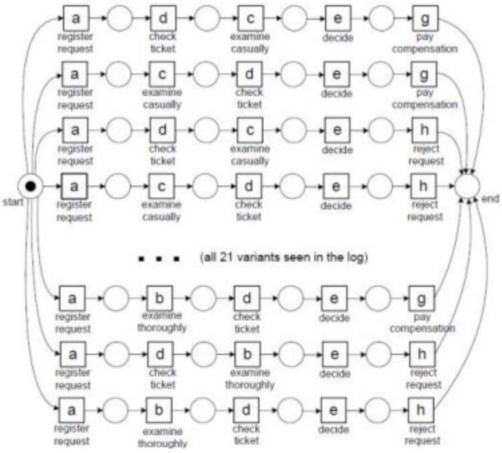
    - Nonfitting with the outcome → no generalization

  - Precision; avoid underfitting



- Underfitting
- There is no flow
  → no precision

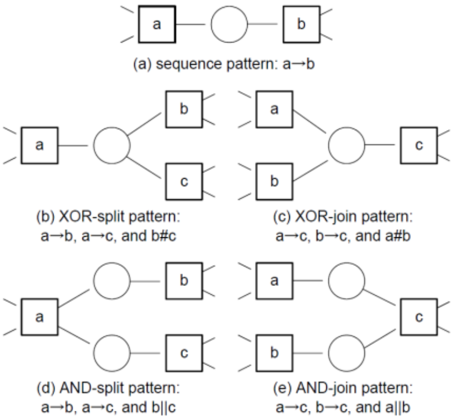
  - Simplicity; as simple as possible


a register request, d check ticket, c examine casually, e decide, g pay compensation
a register request, c examine casually, d check ticket, e decide, g pay compensation
a register request, d check ticket, c examine casually, e decide, h reject request
start
a register request, c examine casually, d check ticket, e decide, h reject request
end
(all 21 variants seen in the log)
a register request, b examine thoroughly, d check ticket, e decide, g pay compensation
a register request, d check ticket, b examine thoroughly, e decide, h reject request
a register request, b examine thoroughly, d check ticket, e decide, h reject request

- Too overfitting model
- Too complicated
- Not fitting with the outcome → bad generalization

C. The alpha algorithm

1. Read the log
2. Get the set of tasks
3. **Infer the ordering relations**
   - Direct succession: x > y
   - Causality: x → y
     Iff x > y and NOT y > x
   - Parallel: x II y
     Iff x > y and y > x
   - Unrelated: x#y
     Iff NOT x > y and NOT y > x
4. Build the net based on inferred relations
5. Output the net



(a) sequence pattern: a→b

(b) XOR-split pattern: a→b, a→c, and b#c

(c) XOR-join pattern: a→c, b→c, and a#b

(d) AND-split pattern: a→b, a→c, and b||c

(e) AND-join pattern: a→c, b→c, and a||b

- Limitations of the alpha-algorithm
    - Representational bias problems
        - No discovery of loops of length one

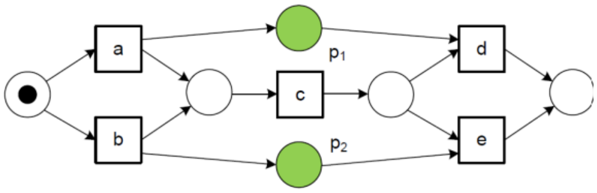
          B > B and NOT B > B implies B→B

          → impossible

        - No discovery of loops of length two

          B > C and C > B implies B II C and C II B instead of B→C and C→B

        - No discovery of non-local dependencies



→ Green places are not discovered!!

        - No discovery of duplicate tasks

        - No discovery of silent/invisible activities
    - Noise
    - Incompleteness

D. Heuristic process discovery

- Heuristics Minder
    - To deal with noise and incompleteness
    - To have a better representational bias than the alpha-algorithm
        - Skips
        - Non-local independencies
        - OR-splits and -joins
    - Uses different process model representation
        - First version: Heuristic nets
        - New version: Causal nets or C-nets
        - Petri net

- Dependency measure
    1. Counting number of times two activities directly follow each other

$$\boxed{\begin{array}{l}|a > b| = \text{the number of times activity } a \text{ is directly followed by activity } b \\ |b > a| = \text{the number of times activity } b \text{ is directly followed by activity } a\end{array}}$$

    2. Calculating 'dependency measure' between task a and b

• If $a \neq b$: Dependency measure $(\mathrm{a}, \mathrm{b}) = \frac{|a>b|-|b>a|}{|a>b|+|b>a|+1}$

• If $a = b$: Dependency measure $(\mathrm{a}, \mathrm{a}) = \frac{|a>a|}{|a>a|+1}$

    - Lower threshold

      → for example: 2 direct successions + at least a dependency of 0,7

      → < 0.70 is not showed in the model
    - Higher threshold

      → for example: 5 direct successions + at least a dependency of 0,9

- Can we now deal with noise?
  → if negative dependency → we do not include this relation in the model
  → the alphamodel would include the relation

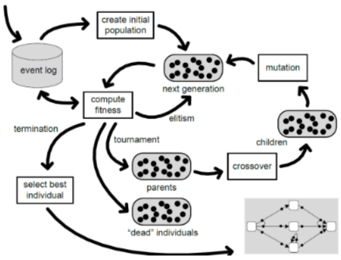
- Conclusion
  + Robust
  + Noices
  + Log completeness
  + Computationally tractable
  + Very fast
  + Can deal with many representational bias problems

  - No duplicate tasks
  - Parameter settings

E. Other process discovery approaches

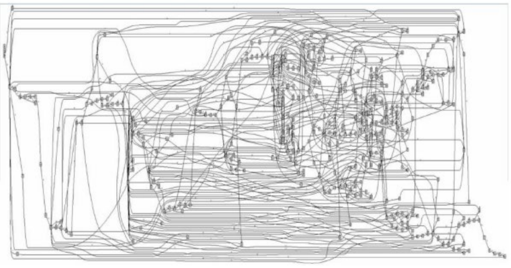
I. Genetic process mining

"Survival of the fittest"



- Design decisions
  - Representation of individuals
  - Initialization
  - Fitness function
  - Selections strategy
  - Crossover
  - Mutation
- Characteristics
  - Requires a lot of computing power
  - Can deal with noise, infrequent behaviour, duplicate tasks, invisible tasks

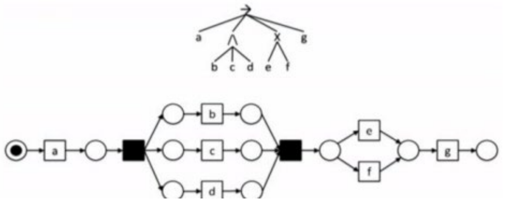
II. Process maps – Fuzzy Miner



- Spaghettimodels
- Simplification by abstraction + filtering
- Conclusion:
  + Can derive process models from highly complex event logs
  + Uses advanced simplification techniques
  + Filtering
  + Abstraction
  - Uses proprietary process modelling language
  - No conversation to Petri nets
    - Difficult to calculate quality metrics
    - Difficult to compare with other process discovery techniques

III. Inductive miner

- Goal: discover sound and block-structured process model
- You can make trade-offs between evaluation dimensions
- Process tree as underlying model



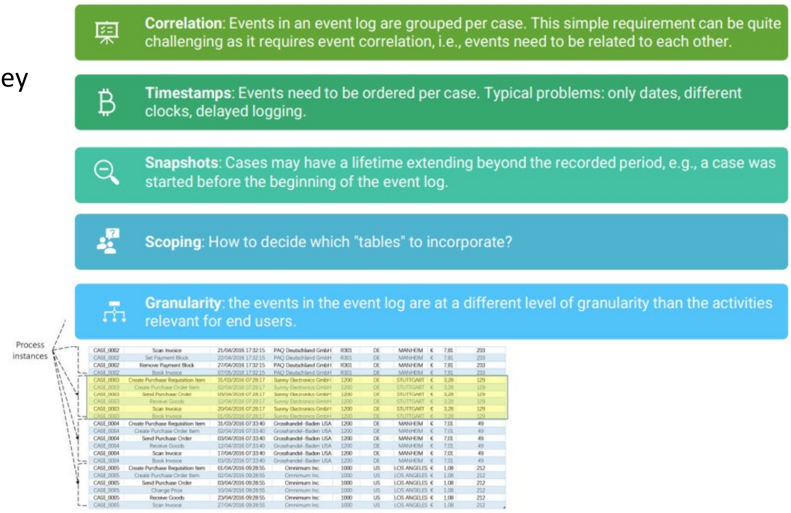
IV. Split miner

- BPMN model discovery tool
- Functionalities
  - Hierarchical process
  - Default representation in BPMN

F. Event data

- Preparation
  - Data sourcing and preprocessing is a key aspect
  - Takes at least 80% of the entire effort
  - Limited and fragmented

- Event logs
  - Case ID
  - Activity Name
  - Timestamp



Correlation: Events in an event log are grouped per case. This simple requirement can be quite challenging as it requires event correlation, i.e., events need to be related to each other.

Timestamps: Events need to be ordered per case. Typical problems: only dates, different clocks, delayed logging.

Snapshots: Cases may have a lifetime extending beyond the recorded period, e.g., a case was started before the beginning of the event log.

Scoping: How to decide which "tables" to incorporate?

Granularity: the events in the event log are at a different level of granularity than the activities relevant for end users.

CASE_0002	Scan Invoice	21/04/2016 17:32:15	PAQ Deutschland GmbH	R301	DE	MANHEIM	€	7,81	233
CASE_0002	Set Payment Block	22/04/2016 17:32:15	PAQ Deutschland GmbH	R301	DE	MANHEIM	€	7,81	233
CASE_0002	Remove Payment Block	27/04/2016 17:32:15	PAQ Deutschland GmbH	R301	DE	MANHEIM	€	7,81	233
CASE_0002	Book Invoice	07/05/2016 17:32:15	PAQ Deutschland GmbH	R301	DE	MANHEIM	€	7,81	233
CASE_0003	Create Purchase Requisition Item	31/03/2016 07:28:17	Sunny Electronics GmbH	1200	DE	STUTTGART	€	3,28	129
CASE_0003	Create Purchase Order Item	02/04/2016 07:28:17	Sunny Electronics GmbH	1200	DE	STUTTGART	€	3,28	129
CASE_0003	Send Purchase Order	05/04/2016 07:28:17	Sunny Electronics GmbH	1200	DE	STUTTGART	€	3,28	129
CASE_0003	Receive Goods	12/04/2016 07:28:17	Sunny Electronics GmbH	1200	DE	STUTTGART	€	3,28	129
CASE_0003	Scan Invoice	20/04/2016 07:28:17	Sunny Electronics GmbH	1200	DE	STUTTGART	€	3,28	129
CASE_0003	Book Invoice	01/05/2016 07:28:17	Sunny Electronics GmbH	1200	DE	STUTTGART	€	3,28	129
CASE_0004	Create Purchase Requisition Item	31/03/2016 07:33:40	Grosshandel-Baden USA	1200	DE	MANHEIM	€	7,01	49
CASE_0004	Create Purchase Order Item	02/04/2016 07:33:40	Grosshandel-Baden USA	1200	DE	MANHEIM	€	7,01	49
CASE_0004	Send Purchase Order	03/04/2016 07:33:40	Grosshandel-Baden USA	1200	DE	MANHEIM	€	7,01	49
CASE_0004	Receive Goods	12/04/2016 07:33:40	Grosshandel-Baden USA	1200	DE	MANHEIM	€	7,01	49
CASE_0004	Scan Invoice	17/04/2016 07:33:40	Grosshandel-Baden USA	1200	DE	MANHEIM	€	7,01	49
CASE_0004	Book Invoice	03/05/2016 07:33:40	Grosshandel-Baden USA	1200	DE	MANHEIM	€	7,01	49
CASE_0005	Create Purchase Requisition Item	01/04/2016 09:28:55	Omnimum Inc.	1000	US	LOS ANGELES	€	1,08	212
CASE_0005	Create Purchase Order Item	02/04/2016 09:28:55	Omnimum Inc.	1000	US	LOS ANGELES	€	1,08	212
CASE_0005	Send Purchase Order	03/04/2016 09:28:55	Omnimum Inc.	1000	US	LOS ANGELES	€	1,08	212
CASE_0005	Change Price	10/04/2016 09:28:55	Omnimum Inc.	1000	US	LOS ANGELES	€	1,08	212
CASE_0005	Receive Goods	23/04/2016 09:28:55	Omnimum Inc.	1000	US	LOS ANGELES	€	1,08	212
CASE_0005	Scan Invoice	27/04/2016 09:28:55	Omnimum Inc.	1000	US	LOS ANGELES	€	1,08	212

- Additional data attributes
  - Case attributes: do not change
  - Event attributes: are particular to a step in the process
  - Usage
    - Filtering
    - Contextual information
    - Dedicated mining techniques

- Event types
  - An event can represent transactional information
  - 2 defaults for process mining
    - Atomic activities → completed events
    - Activities having a duration → started + completed events

- JavaScript Object Notation (JSON)
  - Each event is a JSON line
  - Representing real-time server-to-browser communication

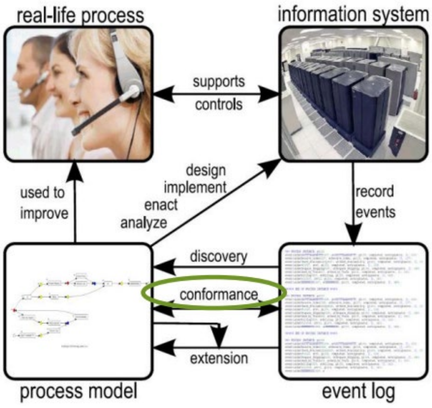
- XES = eXtensible Event Stream
  - Event log storage

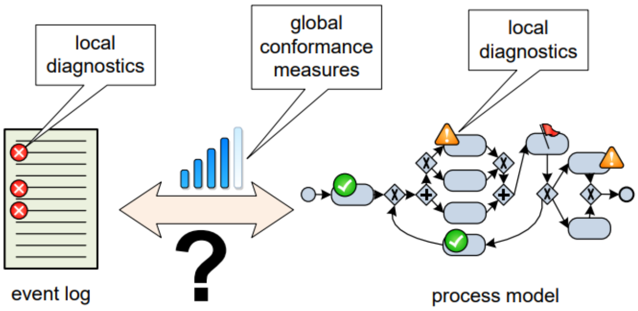
G. Conclusion

- Process discovery is the main innovation driving the process mining field
- A variety of algorithms exist to automatically infer process models from event logs, ranging from the original alpha algorithm to the most recent techniques Fodina, Inductive Miner and Split Miner
- Yet, there is still ample opportunities to improve algorithmic techniques given the complexity of the problem
- Event data comes with important challenges in terms of quality, sourcing, etc. before process discovery can be successfully applied at all

12. Process Mining-Conformance Checking

1. Introduction to Conformance Checking

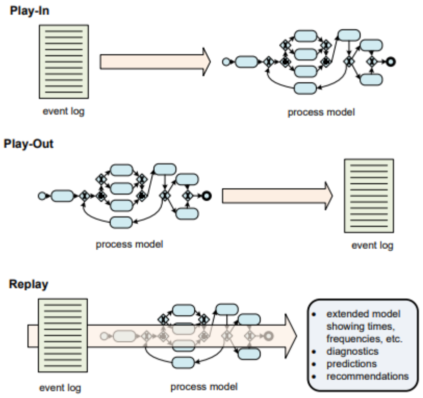




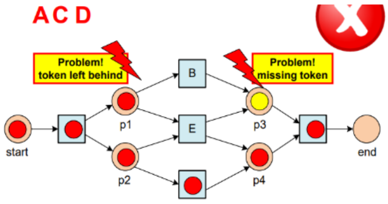
- Using conformance checking
  - Analyse of difference between 2 sorts of behaviour
  - Are there deviations?

- Process mining for audit
  - Obligated to make a rapport for each problem
  - Better manner to sample cases

- Replay
  - Connecting events to model elements is essential for process mining



  - Can detect conformance problems
    - One token left behind
    - Missing token



2. Measuring Fitness

- Conformance: four dimensions → FITNESS; token-based fitness alignments
- Token based fitness (f)
  - Occurs when
    - One or more tokens are missing during replay (m_i)
    - One or more tokens are remaining after replay (r_i)

$$f_{trace} = \frac{1}{2}\left(1 - \frac{m}{c}\right) + \frac{1}{2}\left(1 - \frac{r}{p}\right)$$

m = number of missing tokens
c = number of consumed tokens
r = number of remaining tokens
p = number of produced tokens

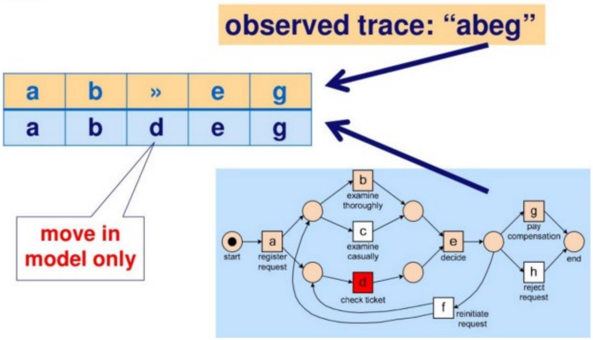
- Fitness over an entire log L

$$f_L = \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times m_\sigma}{\sum_{\sigma \in L} L(\sigma) \times c_\sigma}\right) + \frac{1}{2}\left(1 - \frac{\sum_{\sigma \in L} L(\sigma) \times r_\sigma}{\sum_{\sigma \in L} L(\sigma) \times p_\sigma}\right)$$

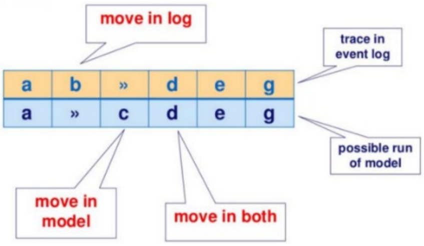
$$\begin{aligned} L(\sigma) &= \text{frequency of trace } \sigma \\ m_\sigma &= \text{number of missing tokens for trace } \sigma \\ c_\sigma &= \text{number of consumed tokensfor trace } \sigma \\ r_\sigma &= \text{number of remaining tokensfor trace } \sigma \\ p_\sigma &= \text{number of produced tokensfor trace } \sigma \end{aligned}$$

→ some important exercises!!!

- Alignment based conformance checking
  - Try to find the best sequence of transition through the model matching a trace as closely as possible
  - Best solution = solution with the lowest cost
  - Optimization can be performed using alpha-algorithm



→ there is no indication to carry out D, but the alignment find an optimal solution to carry out D where there is no observation

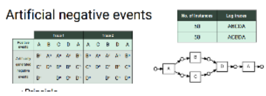


+ Avoids forcing transition to fire
+ Does not introduce tokens that can skew replay
+ Nice optimization problem: tree-based algorithm to find min cost path
+ Costs can be user-modified

- Very time-consuming
- Model-only move hard to justify
- Setting costs is nice but what if optimization routes around your set costs?

3. Measuring precision

- Conformance: four dimensions → PRECISION



I. Advanced Behavioural Appropriateness

- One of the first precision metrics
- Compare behaviour in the model with the behaviour in the event log

- 2 metrics compare two types of relationship between activities
  - Follows relationship
    - A_F = Always Follows
    - N_F = Never Follows
    - S_F = Sometimes Follows
  - Precedes relationship
    - A_P = Always Precedes
    - N_P = Never Precedes
    - S_P = Sometimes Precedes

$$a'_B = \frac{|S_F^l \cap S_F^m|}{2\times|S_F^m|} + \frac{|S_P^l \cap S_P^m|}{2\times|S_P^m|}$$

= 1 → perfect precision

< 1 → underfitting – too much behaviour

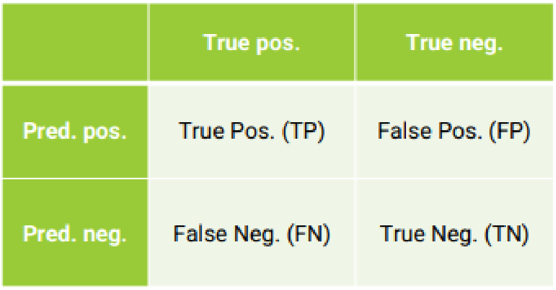
II. Negative Event-based behavioural precision

- Artificial negative events
  - Principle: At each positive event, it is checked whether another event appearing in the log could also occur, given its prefix. Therefore, it is tested whether in the log there exists a similar prefix sequence with this other event occurring at the same position of the positive event. When this is not the case, a negative event can be injected into the log at the position of the positive event.

- Recall (fitness) and precision
  - Confusion matrix after log replay → eventbased

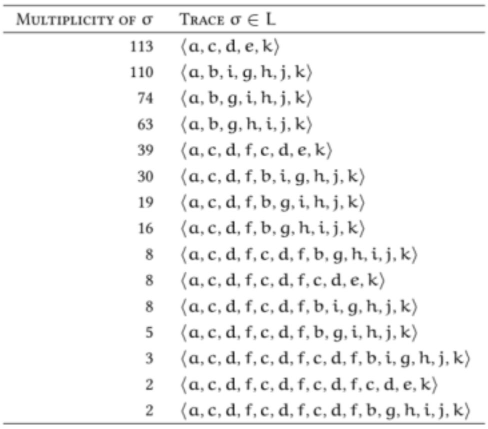
- Recall: $$r_B^p = \frac{\sum_{i=1}^{k} n_i TP_i}{\sum_{i=1}^{k} n_i TP_i + \sum_{i=1}^{k} n_i FN_i}$$

- Precision: $$p_B = \frac{\sum_{i=1}^{k} n_i TP_i}{\sum_{i=1}^{k} n_i TP_i + \sum_{i=1}^{k} n_i FP_i}$$



	True pos.	True neg.
Pred. pos.	True Pos. (TP)	False Pos. (FP)
Pred. neg.	False Neg. (FN)	True Neg. (TN)

- The basic algorithm



Multiplicity of σ	Trace σ ∈ L
113	⟨a, c, d, e, k⟩
110	⟨a, b, i, g, h, j, k⟩
74	⟨a, b, g, i, h, j, k⟩
63	⟨a, b, g, h, i, j, k⟩
39	⟨a, c, d, f, c, d, e, k⟩
30	⟨a, c, d, f, b, i, g, h, j, k⟩
19	⟨a, c, d, f, b, g, i, h, j, k⟩
16	⟨a, c, d, f, b, g, h, i, j, k⟩
8	⟨a, c, d, f, c, d, f, b, g, h, i, j, k⟩
8	⟨a, c, d, f, c, d, f, c, d, e, k⟩
8	⟨a, c, d, f, c, d, f, b, i, g, h, j, k⟩
5	⟨a, c, d, f, c, d, f, b, g, i, h, j, k⟩
3	⟨a, c, d, f, c, d, f, c, d, f, b, i, g, h, j, k⟩
2	⟨a, c, d, f, c, d, f, c, d, f, c, d, e, k⟩
2	⟨a, c, d, f, c, d, f, c, d, f, b, g, h, i, j, k⟩

[a, c, d, e, k]

○ First position, all events that not started are added as negative events to position 1.

Result: [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, c, d, e, k]

○ Second position, prefix [a]

Result: : [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, (a-, d-, e-, f-, g-, h-, i-, j-, k-), c, d, e, k] → not b because it is observed after prefix [a]

○ Third position prefix [a,c]

Result: : [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, (a-, d-, e-, f-, g-, h-, i-, j-, k-), c, (a-, b-, c-, e-, f-, g-, h-, i-, j-, k-), d, e, k]

- Fourth position, prefix [a, c, d]
  Result: [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, (a-, d-, e-, f-, g-, h-, i-, j-, k-), c, (a-, b-, c-, e-, f-, g-, h-, i-, j-, k-), d, (a-, b-, c-, d-, g-, h- , i-, j-, k-), e, k] → no f, because it is observed after the prefix

- Fifth position, prefix [a,c,d,e]
  Result: [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, (a-, d-, e-, f-, g-, h-, i-, j-, k-), c, (a-, b-, c-, e- , f-, g-, h-, i-, j-, k-), d, (a-, b-, c-, d-, g-, h-, i-, j-, k-), e, (a-, b-, c-, d-, e-, f-, g-, h-, i-, j-), k]

  → Computing recall and precision
  [(b-, c-, d-, e-, f-, g-, h-, i-, j-, k-), a, (a-, d-, e-, f-, g-, h-, i-, j-, k-), c, (a-, b-, c-, e- , f-, g-, h-, i-, j-, k-), d, (a-, b-, c-, d-, g-, h-, i-, j-, k-), e, (a-, b-, c-, d-, e-, f-, g-, h-, i-, j-), k]
  - → True positive
  - → False positive
  - → False negative
  - → True negative

4. Measuring generalization/simplicity

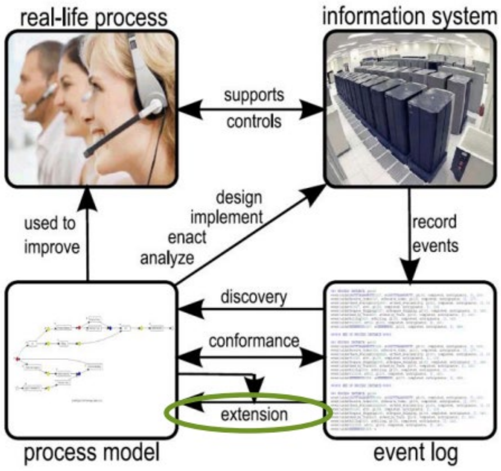
- Conformance: four dimensions → SIMPLICITY
- Based on topology, layout, "visual" of the model
  <-> generalization
  - Too specified

5. Conclusion

- Conformance checking deals with comparing model behavior with observed behavior in the event log
- Conformance checking algorithms (replay, alignments) can automatically carry out such a comparison
- Comparing modelled an observed behavior can yield valuable insights, e.g. for auditors
- Several metrics exist to quantify conformance dimensions, most importantly fitness and precision metrics

13. Process Mining – Extension and Tooling

A. Extension Techniques

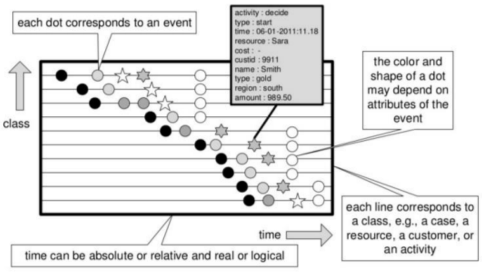


- Process are not just about control-flow
- A process consist of cases
- A case consist of events
- Events are ordered within a case
- Events can have attributes

I. Performance analysis

- Using time dimensions
- During replay → information of wait time and lead time
- Identify bottlenecks

II. Dotted charts

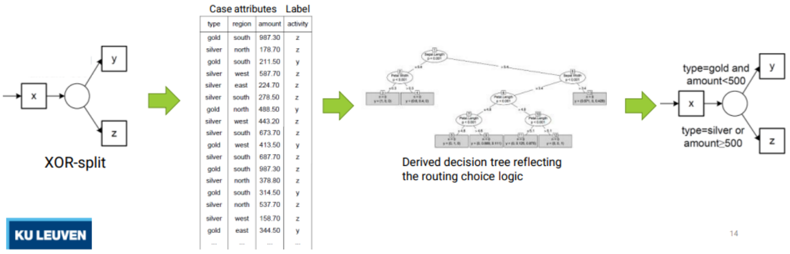

each dot corresponds to an event
activity : decide
type : start
time : 06-01-2011:11.18
resource : Sara
cost : -
custid : 9911
name : Smith
type : gold
region : south
amount : 989.50
the color and shape of a dot may depend on attributes of the event
class
each line corresponds to a class, e.g., a case, a resource, a customer, or an activity
time
time can be absolute or relative and real or logical

- Upfront analysis
- Helicopter view
- Every line is a case
- Every shape is an event

→ lot of flexibilities

III. Decisions mining

- Find rules that explain routing choices in terms of the characteristics of a case
- XOR-split
- Methodology
  - Find the features that influence the routing decisions
  - Decision tree learning can be used
  - Only information form the past


x
y
z
XOR-split
Derived decision tree reflecting the routing choice logic
type=gold and amount<500
type=silver or amount≥500
KU LEUVEN
14

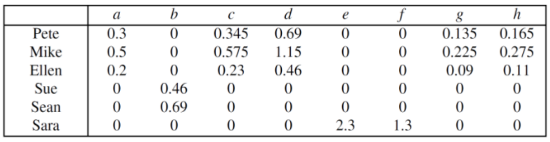
Case attributes Label

type	region	amount	activity
gold	south	987.30	z
silver	north	178.70	z
gold	south	211.50	y
silver	west	587.70	z
silver	east	224.70	z
silver	south	278.50	z
gold	north	488.50	y
silver	west	443.20	z
silver	south	673.70	z
gold	west	413.50	y
silver	south	687.70	z
gold	south	987.30	z
silver	north	378.80	z
gold	south	314.50	y
silver	north	537.70	z
silver	west	158.70	z
gold	east	344.50	y
...	...	...	...

IV. Organisational mining

- Resource-activity matrix

  = mean number of times a resource performs an activity per case



	a	*b*	*c*	*d*	*e*	*f*	*g*	*h*
Pete	0.3	0	0.345	0.69	0	0	0.135	0.165
Mike	0.5	0	0.575	1.15	0	0	0.225	0.275
Ellen	0.2	0	0.23	0.46	0	0	0.09	0.11
Sue	0	0.46	0	0	0	0	0	0
Sean	0	0.69	0	0	0	0	0	0
Sara	0	0	0	0	2.3	1.3	0	0

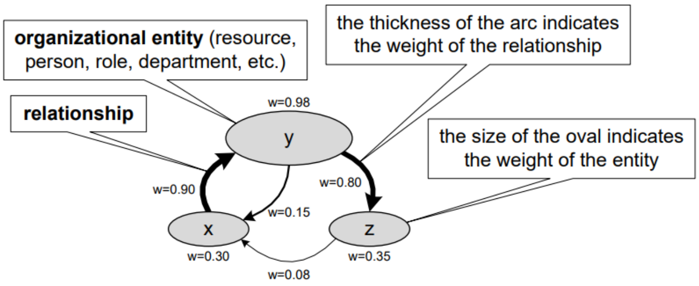
→ Mike is a generalist

→ Sue is an expert

→ Sara is a manager

→ Activity a is executed exactly once for each case (hence the sum of the first column is 1). Pete, Mike, and Ellen are the only ones executing this activity. In 30% of the cases, a is executed by Pete, 50% is executed by Pete, and 20% is executed by Ellen. Activities e and f are always executed by Sara. Activity e is executed, on average, 2.3 times per case.

→ Social network:



organizational entity (resource, person, role, department, etc.)

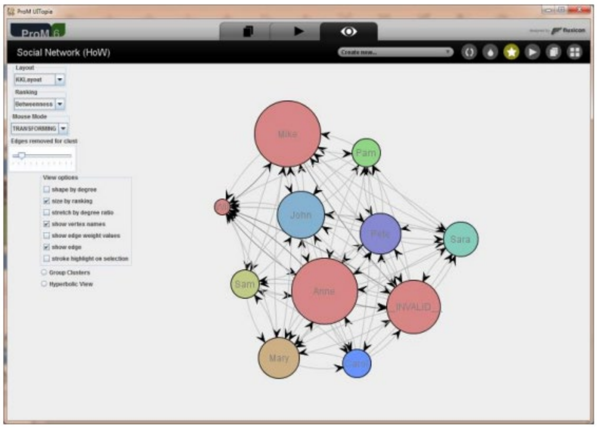
the thickness of the arc indicates the weight of the relationship

relationship

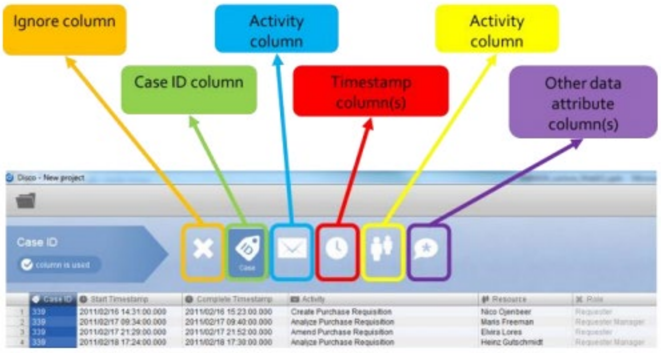
the size of the oval indicates the weight of the entity

→ Merging process and social views → handover matrix

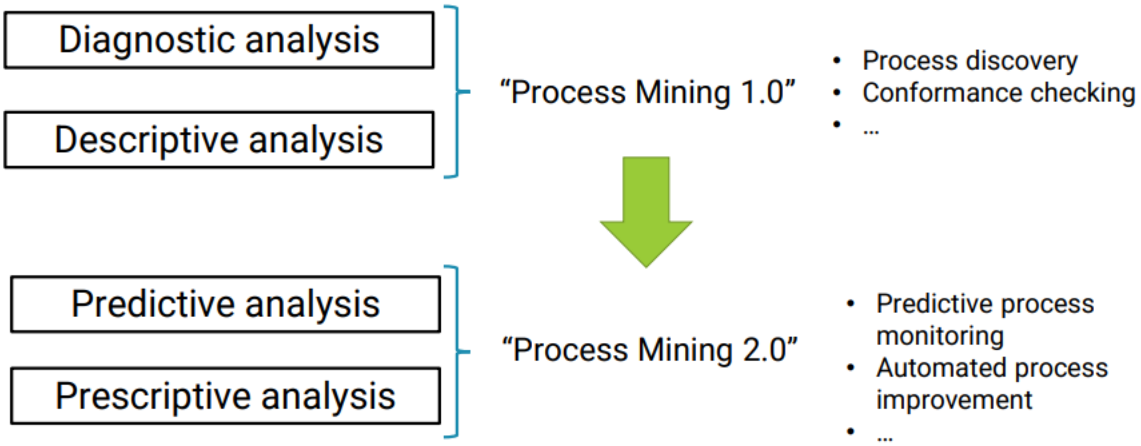
→ ProM



→ Disco



B. Predictive Process Mining



Diagnostic analysis

Descriptive analysis

"Process Mining 1.0"

- Process discovery
- Conformance checking
- ...

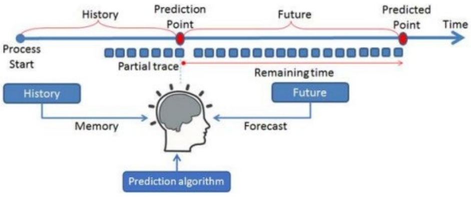
Predictive analysis

Prescriptive analysis

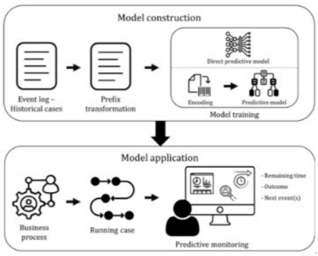
"Process Mining 2.0"

- Predictive process monitoring
- Automated process improvement
- ...

- From descriptive to predictive
  - Supervised applications are becoming more important
  - Use event data for predictive goals
    - Next event
    - Remaining suffix
    - Remaining time
    - Outcome

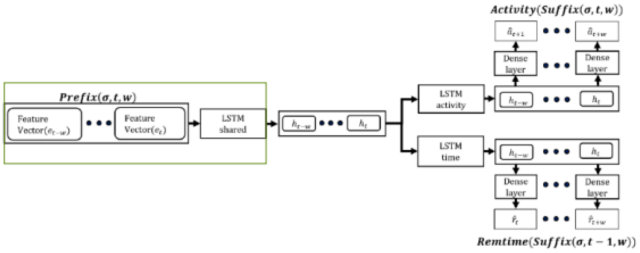


- Predictive process monitoring

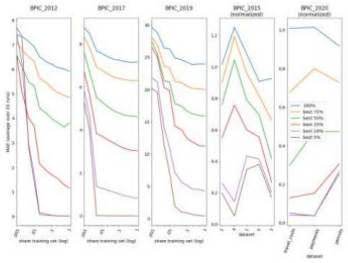


- Machine Learning driving new developments
  - Decision trees
  - Recurrent neural networks
  - General adversarial neural networks
  - Convolutional neural networks

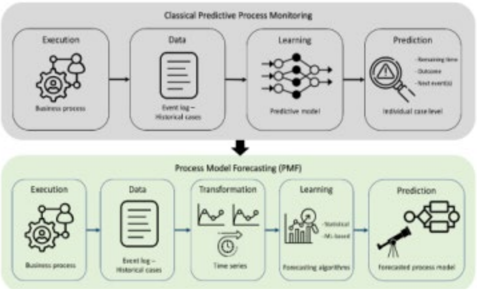
- Our research
  - Seq2Seq-LSTMs for complete suffix and remaining time prediction
    - Processing of luggage at Brussels Airport, trajectory and execution times of luggage influenced by:
      - Case features
      - Time features
      - Event features
    - Research: develop a model architecture that can natively incorporate all relevant information in order to predict the remaining suffixes and runtimes of bags



  - Uncertainty-based deep learning
    - 2 types of uncertainty
      - Epistemic uncertainty
      - Aleatoric uncertainty
    - Technique to estimate these uncertainties
      - Better for smaller datasets
      - Improvement adoption strategies



- Process model forecasting
    - From operational predictions to tactical and strategic insights
    - Full model → forecast future model



C. Tool support

I. Open-source tools

- ProM
    - Load + apply data
- Apromore
- ….

II. Commercial process mining tools

- Celonis
- Disco
    - Focus on data-mining
    - More userfriendly then Celonis
    - Commercial tool

      + Easy to use
      + Fast
      + Profound insights into event log
      + Advanced filtering techniques

    - Case-level statistics
        - Overall event log statistics
        - Various graphs
        - Case-specific information
    - Activity-level statistics
        - General statistics
        - Several graphs
        - Activity-specific information; frequency and duration
    - Resource-level statistics
        - Similar to activity level
    - Process maps
    - Filtering → highly advanced