Functional Requirements

Describe product's actions

The product shall produce a schedule of all roads upon which ice is predicted to form within a given parameter

Non Functional Requirements

Describe product’s qualities and attributes

The product shall be able to determine “friend or foe” in less then .25 seconds

The product shall provide a pleasing user experience

The product shall be able to be used by travelers in the arrivals hall who do not speak the home language

Constraints

Constraints can be limitations on the project itself or restrictions on the project design

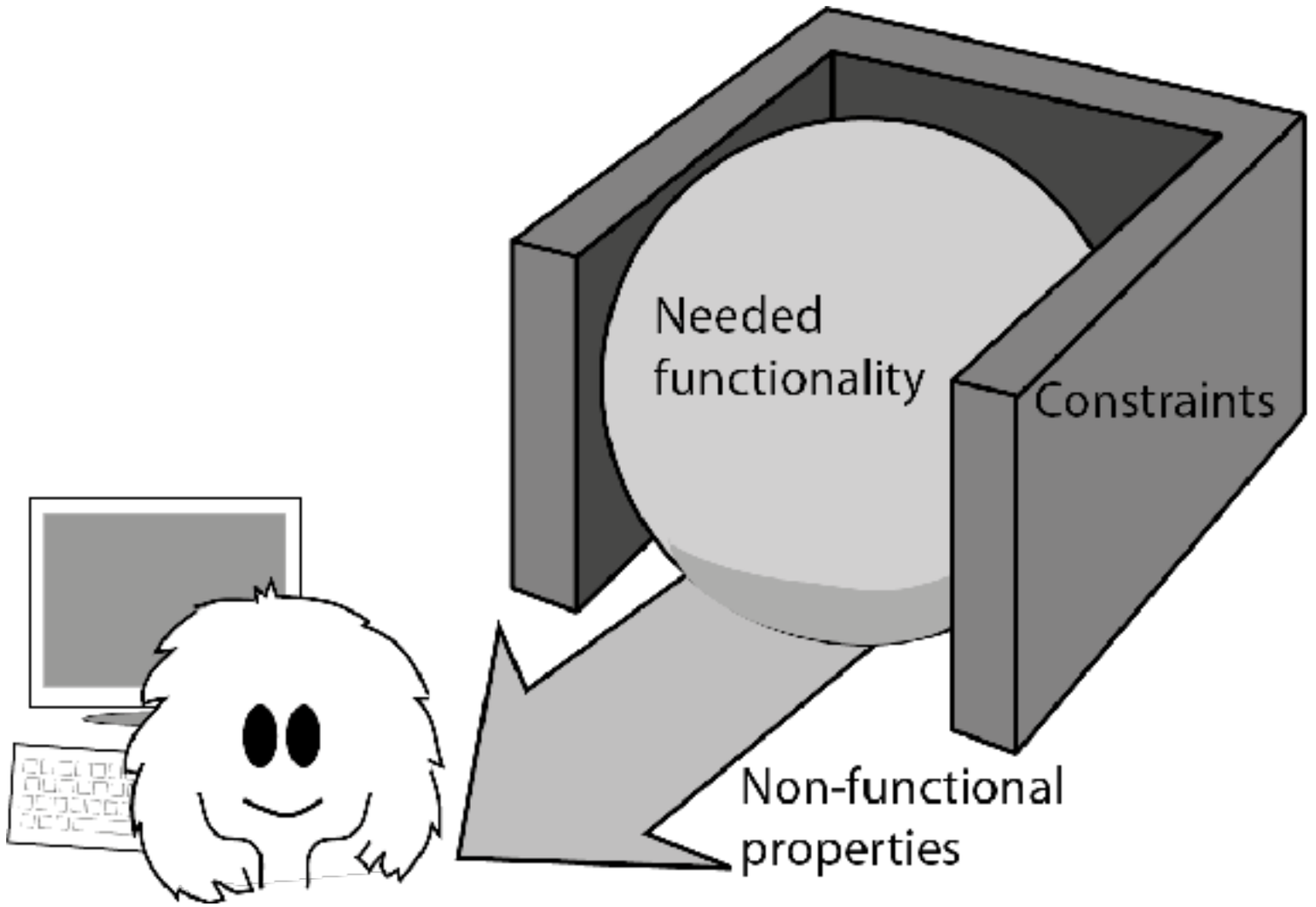
Constrains are simply another type of requirements

Constrains are global issues that shape the requirements

The product must be available at the beginning of the new tax year

The product shall operate as an iPad, iPhone, Android, and Blackberry App

The functionality of the end product is restricted by the constraints. The functionality is to the benefit of its user, but it is the non-functional requirements that “deliver” the functionality by making the product usable and acceptable to the users.

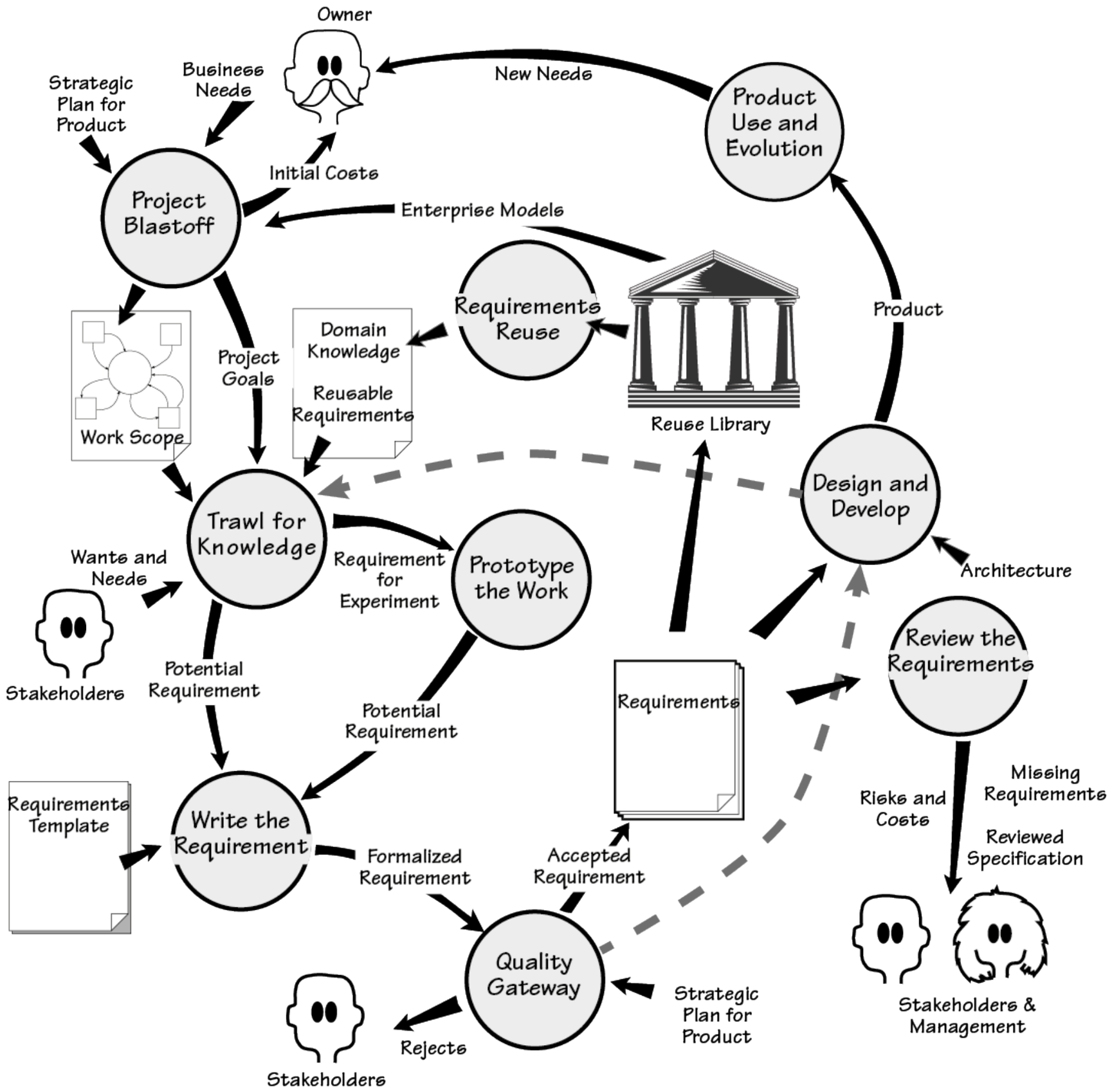


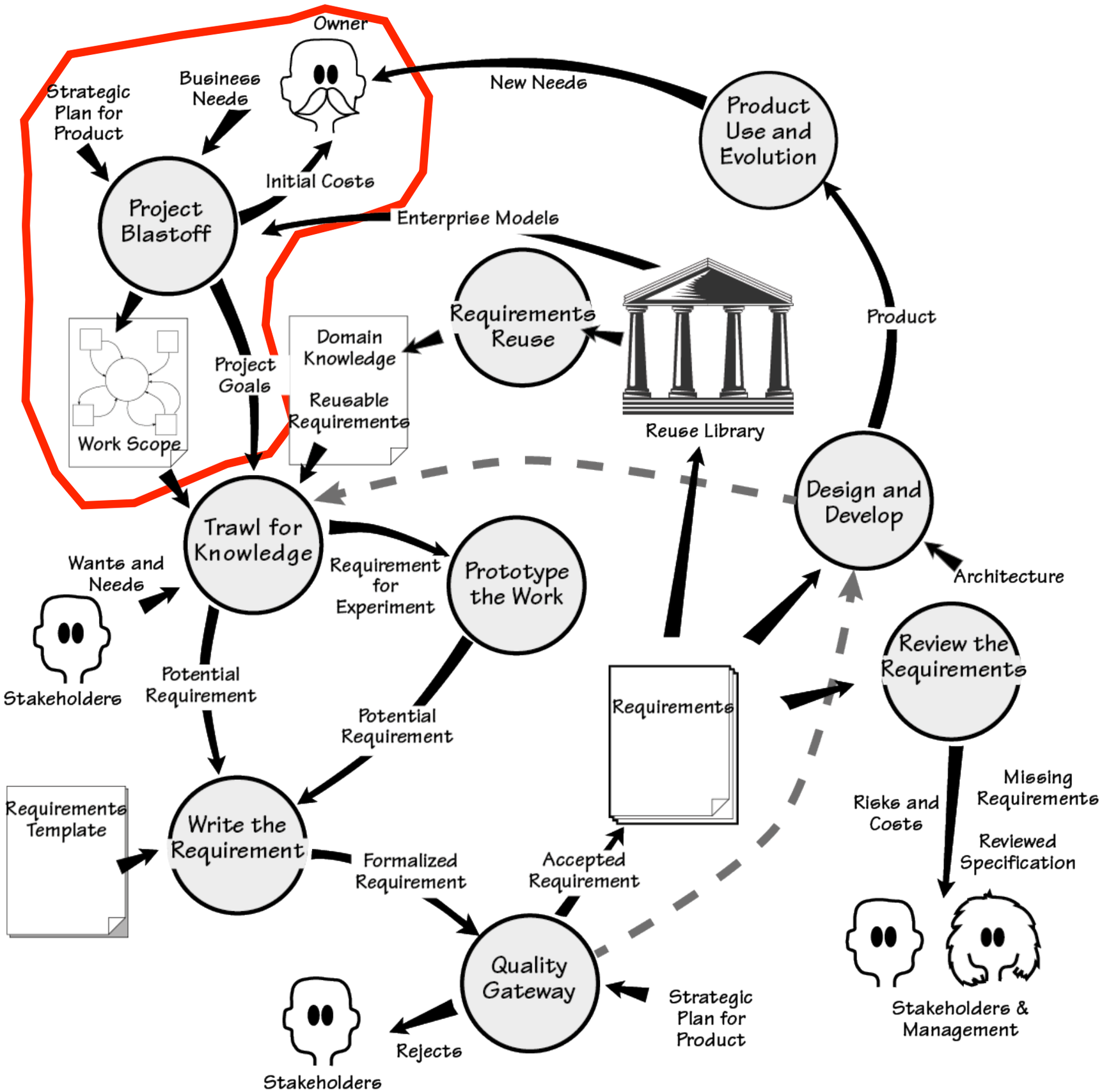
A Case Study

- ✦IceBreaker
  - ✦Predicts when and where ice will form on roads
  - ✦Schedules trucks to treat the roads with de-icing material
  - ✦Will enable road authorities to more accurately predict ice formation, schedule road treatment more precisely and thereby make roads safer.
  - ✦Will also reduce the amount of de-icing material needed, helping both finances and the environment

Figure 2.1

This map of the Volere Requirements Process shows the activities and their deliverables. We have used a stylized data flow notation. Each activity (the bubbles) and its deliverables (named arrows or documents) are explained in the text. The dotted lines represent how this process is used with iterative projects.




Owner
Strategic Plan for Product
Business Needs
Initial Costs
New Needs
Product Use and Evolution
Project Blastoff
Enterprise Models
Requirements Reuse
Reuse Library
Product
Work Scope
Project Goals
Domain Knowledge
Reusable Requirements
Trawl for Knowledge
Wants and Needs
Stakeholders
Requirement for Experiment
Prototype the Work
Potential Requirement
Potential Requirement
Design and Develop
Architecture
Review the Requirements
Requirements
Requirements Template
Write the Requirement
Formalized Requirement
Accepted Requirement
Risks and Costs
Missing Requirements
Reviewed Specification
Quality Gateway
Strategic Plan for Product
Rejects
Stakeholders
Stakeholders & Management

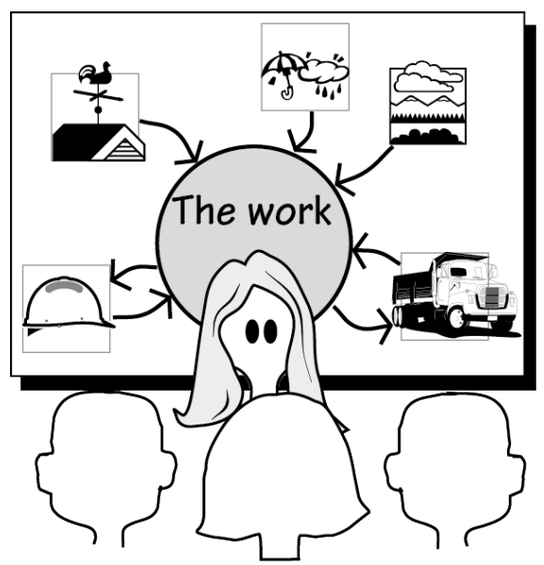
1-Project Blastoff

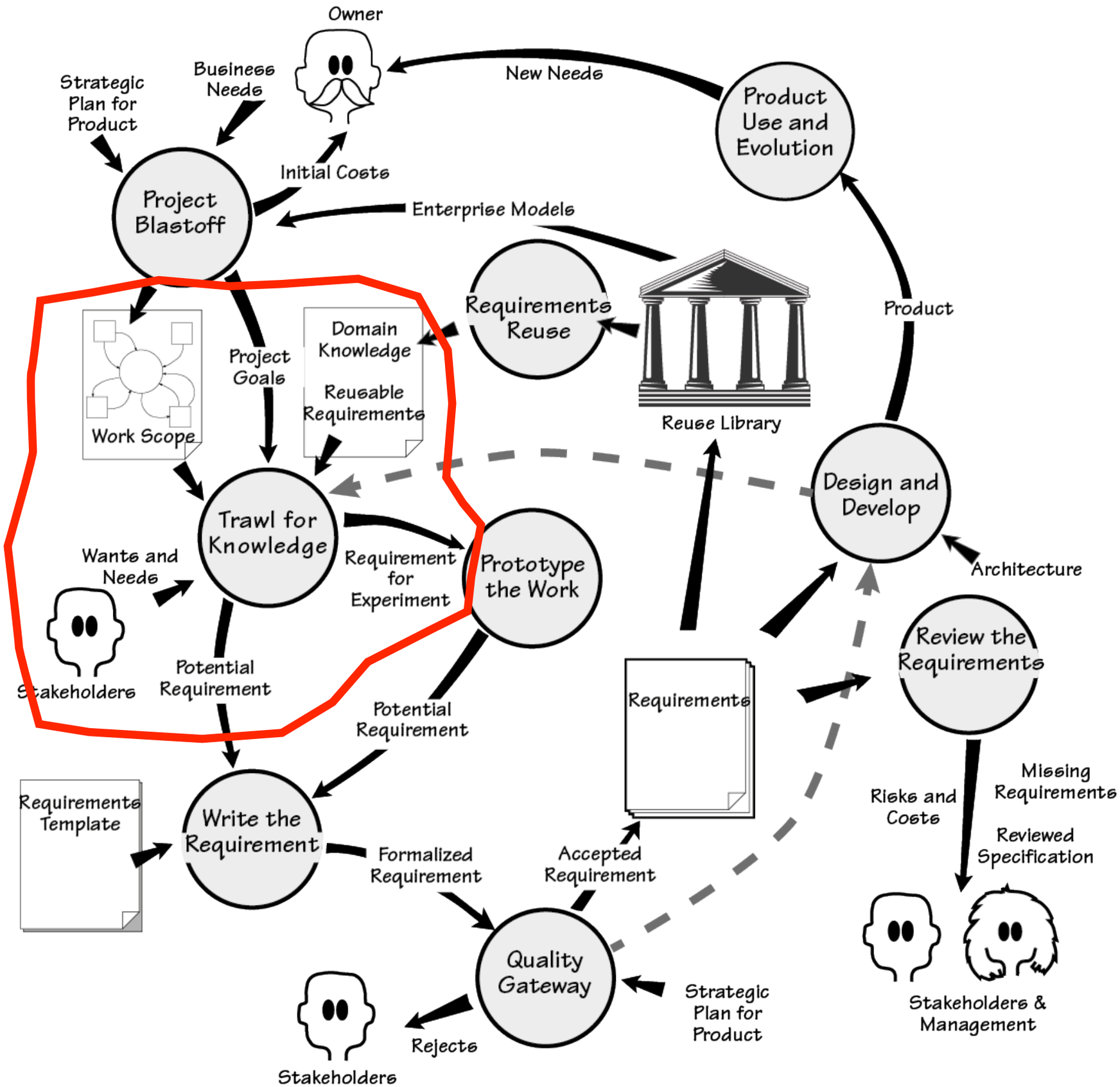
- Build the foundation for the requirements discovery that is to follow.
- Ensure that all components of a successful project are in place.
- Defines the scope of the business problem, and seeks agreement from stakeholder.
- Lead a discussion to define scope, and relationship with everything around it.
- Define Stakeholders.
- Confirms goals.
- Estimate of costs.
- Assessment of Risks.
- Determine Viability,

  a.k.a go/no go

Figure 2.2

The context diagram is used to build a consensus among the stakeholders as to the scope of the work that needs to be improved. The eventual product will be used to do part of this work.




Owner
Strategic Plan for Product
Business Needs
New Needs
Product Use and Evolution
Project Blastoff
Initial Costs
Enterprise Models
Requirements Reuse
Reuse Library
Product
Work Scope
Project Goals
Domain Knowledge
Reusable Requirements
Trawl for Knowledge
Prototype the Work
Wants and Needs
Requirement for Experiment
Stakeholders
Potential Requirement
Potential Requirement
Design and Develop
Architecture
Review the Requirements
Requirements
Requirements Template
Write the Requirement
Formalized Requirement
Accepted Requirement
Risks and Costs
Missing Requirements
Reviewed Specification
Quality Gateway
Strategic Plan for Product
Rejects
Stakeholders
Stakeholders & Management

2-Trawling for Requirements

- ✦Understand the underlying business reason.
- ✦Most stakeholders talk about perceived solution, which may not be the optimal solution.
- ✦Your job is to uncover the *essence* of the system, not the perceived solution!

2-Trawling for Requirements

Learning and understanding the functionality of the work

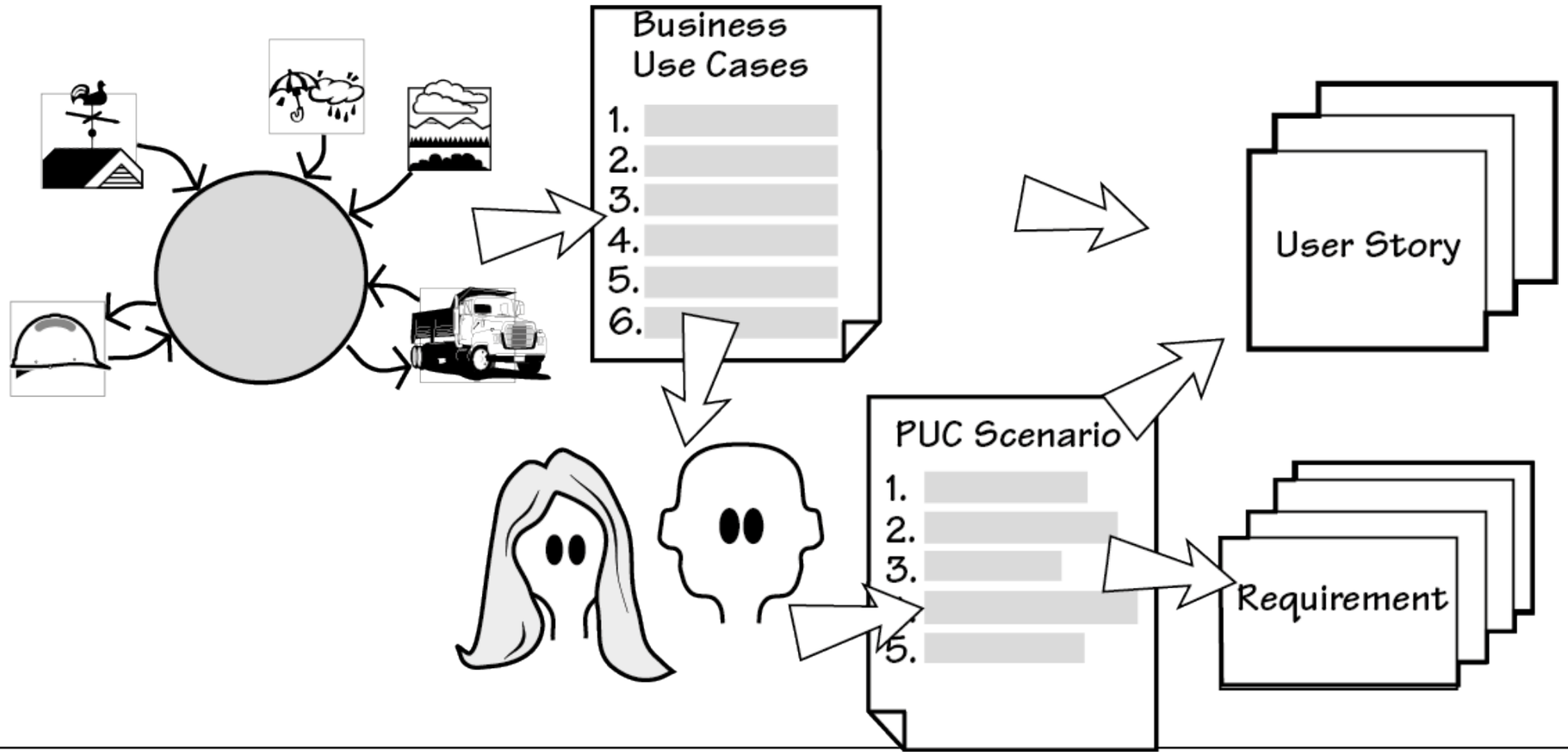
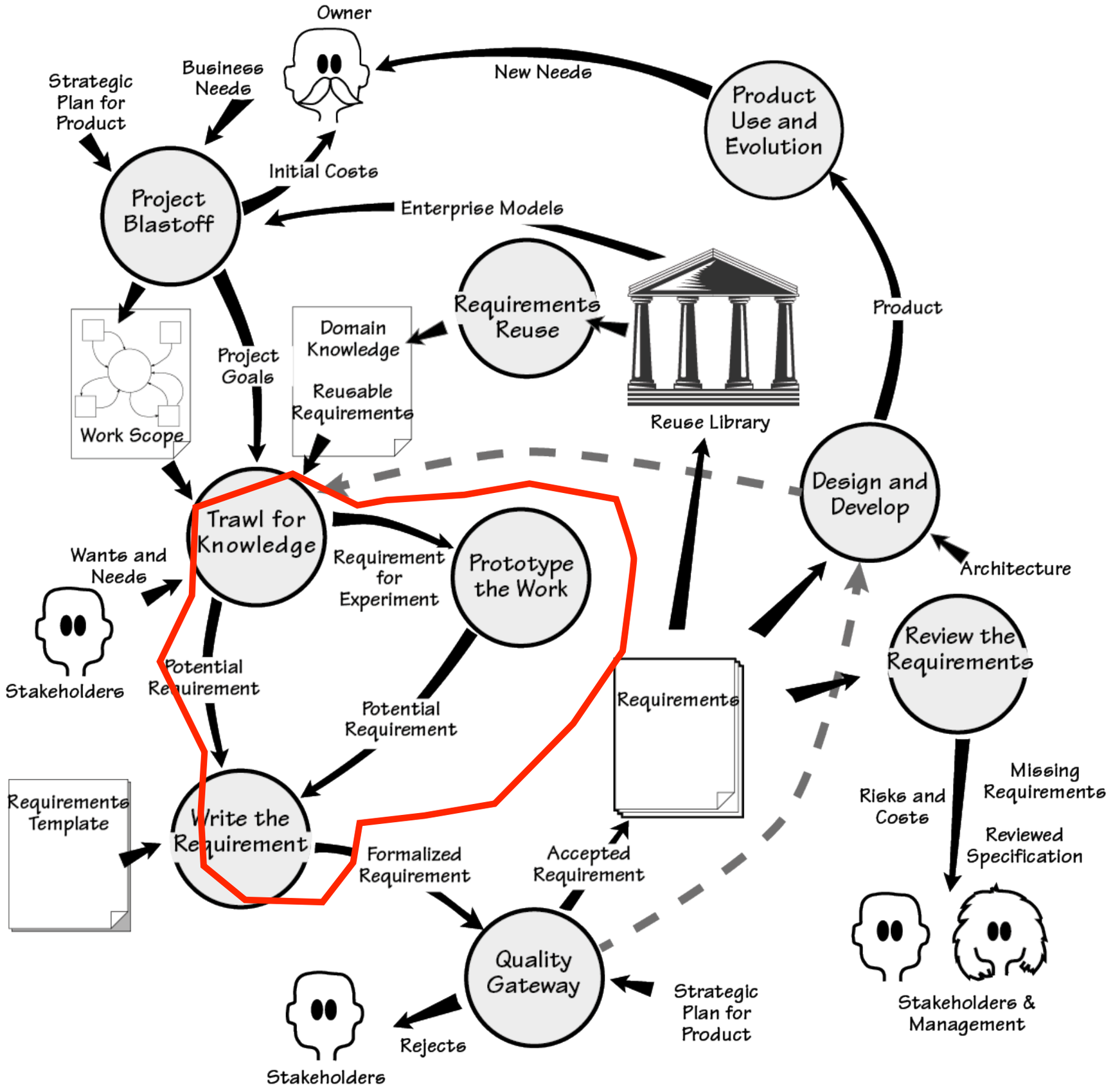


Figure 2.3

The blastoff determines the scope of the work to be improved. The business use cases are derived from the scope. Each of the business use cases is studied by the requirements analysts and the relevant stakeholders to discover the desired way of working. When this is understood, the appropriate product can be determined (the PUC scenario) and requirements or user stories written from it.


Owner
Strategic Plan for Product
Business Needs
New Needs
Product Use and Evolution
Project Blastoff
Initial Costs
Enterprise Models
Requirements Reuse
Reuse Library
Product
Work Scope
Project Goals
Domain Knowledge
Reusable Requirements
Trawl for Knowledge
Prototype the Work
Design and Develop
Wants and Needs
Requirement for Experiment
Architecture
Review the Requirements
Stakeholders
Potential Requirement
Potential Requirement
Requirements
Requirements Template
Write the Requirement
Formalized Requirement
Accepted Requirement
Risks and Costs
Missing Requirements
Reviewed Specification
Quality Gateway
Strategic Plan for Product
Stakeholders & Management
Rejects
Stakeholders

Quick and Dirty Modelling

Models, sometimes referred to as prototypes, can be used at any time during the specification life cycle

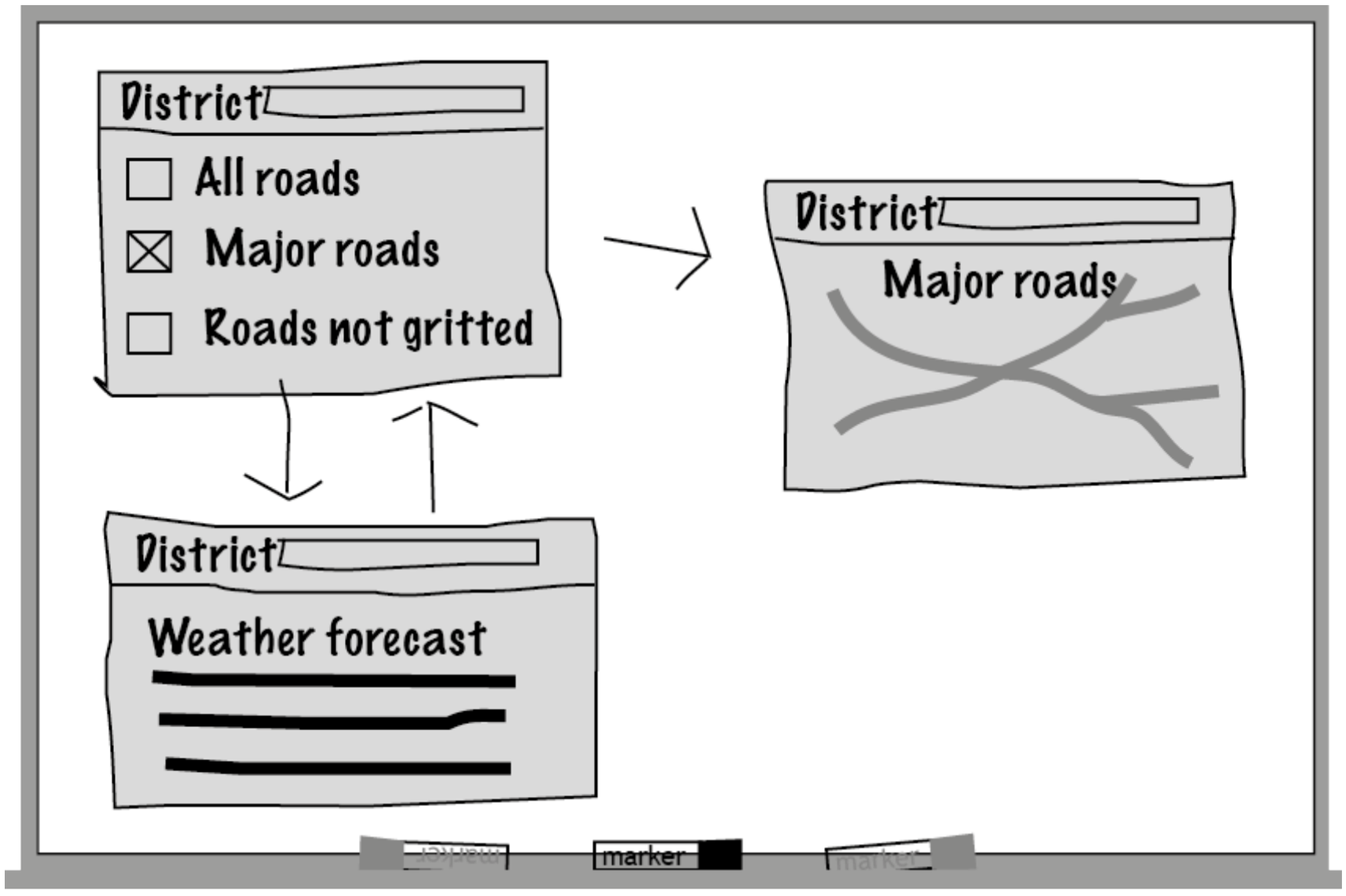


Figure 2.4

A quick and dirty prototype built on a whiteboard to provide a rapid visual explanation of how some of the requirements might be implemented, and to clarify misunderstood or missing requirements.

Scenarios

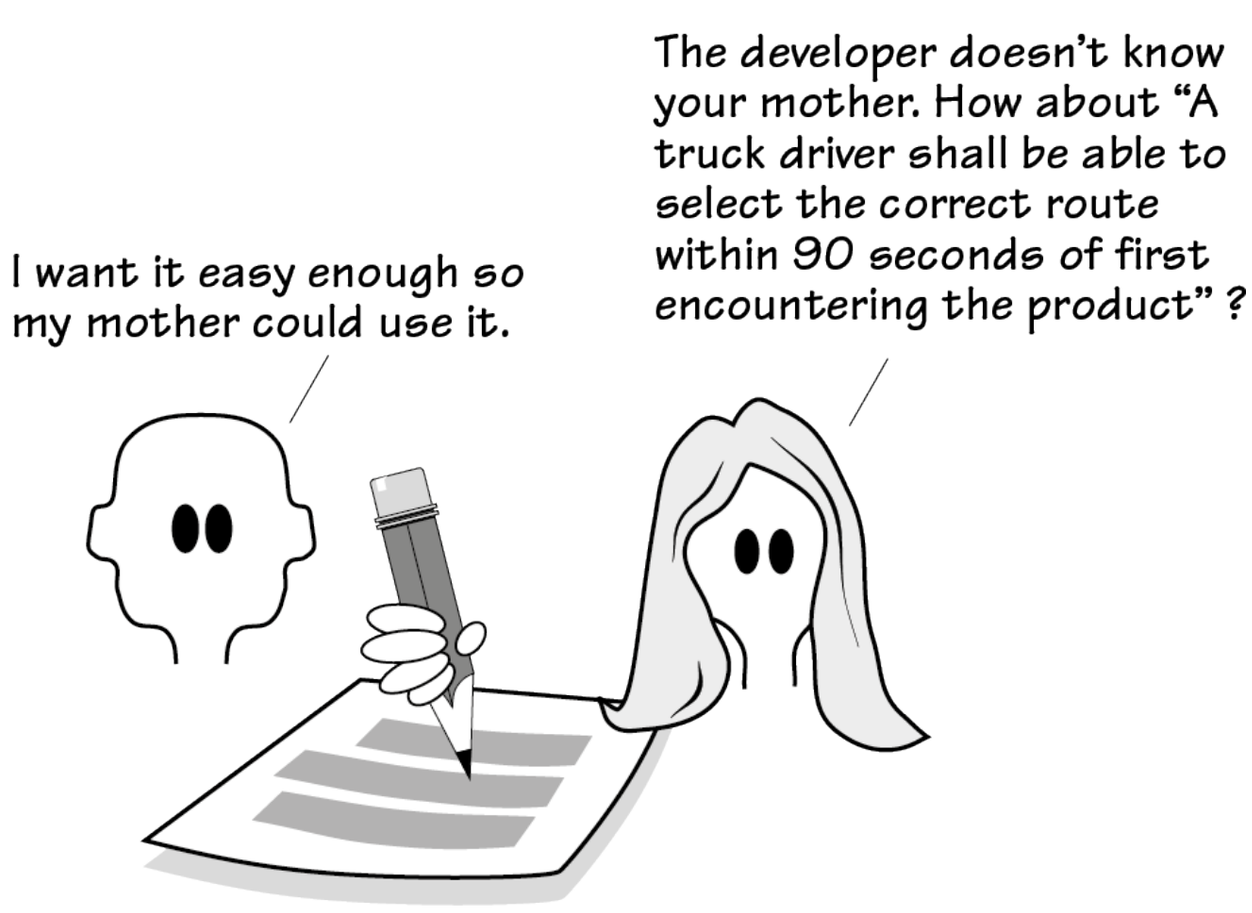
Show functionality of a business process by breaking it down into a series of easily recognizable steps, written in English, so they are accessible to all stakeholders.

Writing Requirements

Analysts must write requirements in an unambiguous and testable manner, and at the same time ensure that the originating stakeholder understands and agrees with the written requirement before it's passed to developers

Figure 2.5

The requirements are captured in written form to facilitate communication between the stakeholders, the analysts, and the developers (and anyone else who has an interest). By writing the requirements carefully, the team ensures that the correct product is built.



Project Blastoff
(Feasibility Study)
Chapter 3

Recap of definitions

- **Requirement**: an action that the system (a.k.a. product) is capable of doing, or an attribute/quality that the system must have.
- You must discover the requirements before you start building the product.

Functional Requirements

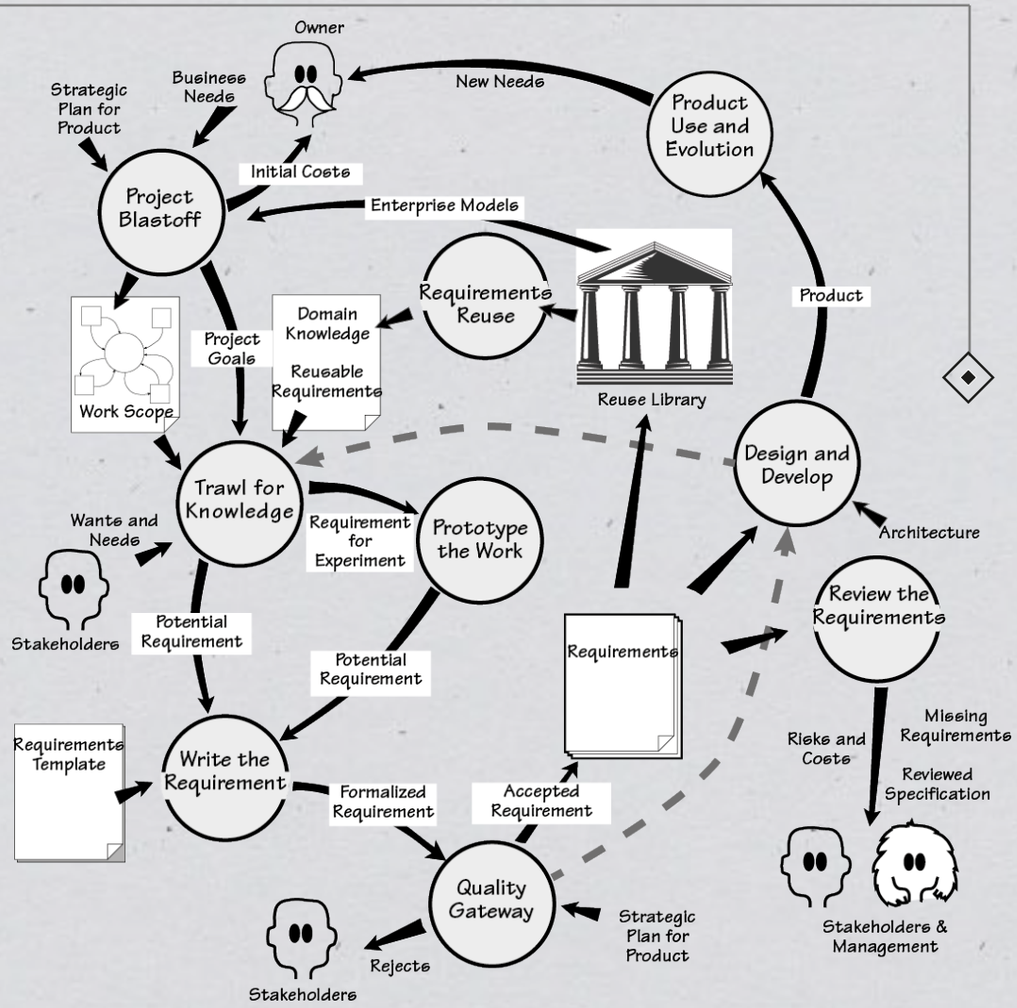
- Specify the **actions** that the system/product must take to contribute to the goal of the project.
- They are not qualities.
- Example:
  - The *Airline Passenger Service System shall* automatically assign a passenger to a seat on her chosen flight.

Non-Functional Requirements

- ✦Qualities or attributes that the system/product must have.
- ✦Example: Performance (e.g., speed), Look and Feel, Security, etc.
- ✦Example: The *Airline Passenger Service System shall* automatically *assign* a passenger to a seat on her chosen flight within 0.25 second.

Constraints

- ✦Global issues that shape the requirements.
- ✦They set limitations/restrictions on the product.
- ✦Budgetary constrains, Design solutions, platform choices, etc.
- ✦Example:
  - ✦The budget available to develop the system is $1.2 million.
  - ✦The system has to be developed within 90 days.
  - ✦The system shall run on Android and iOS.



- This figure is an overview of different requirement activities.
- We will cover each activity in detail throughout the course.
- This figure can be overwhelming, so I decided to summarize it in text in the next two slides..

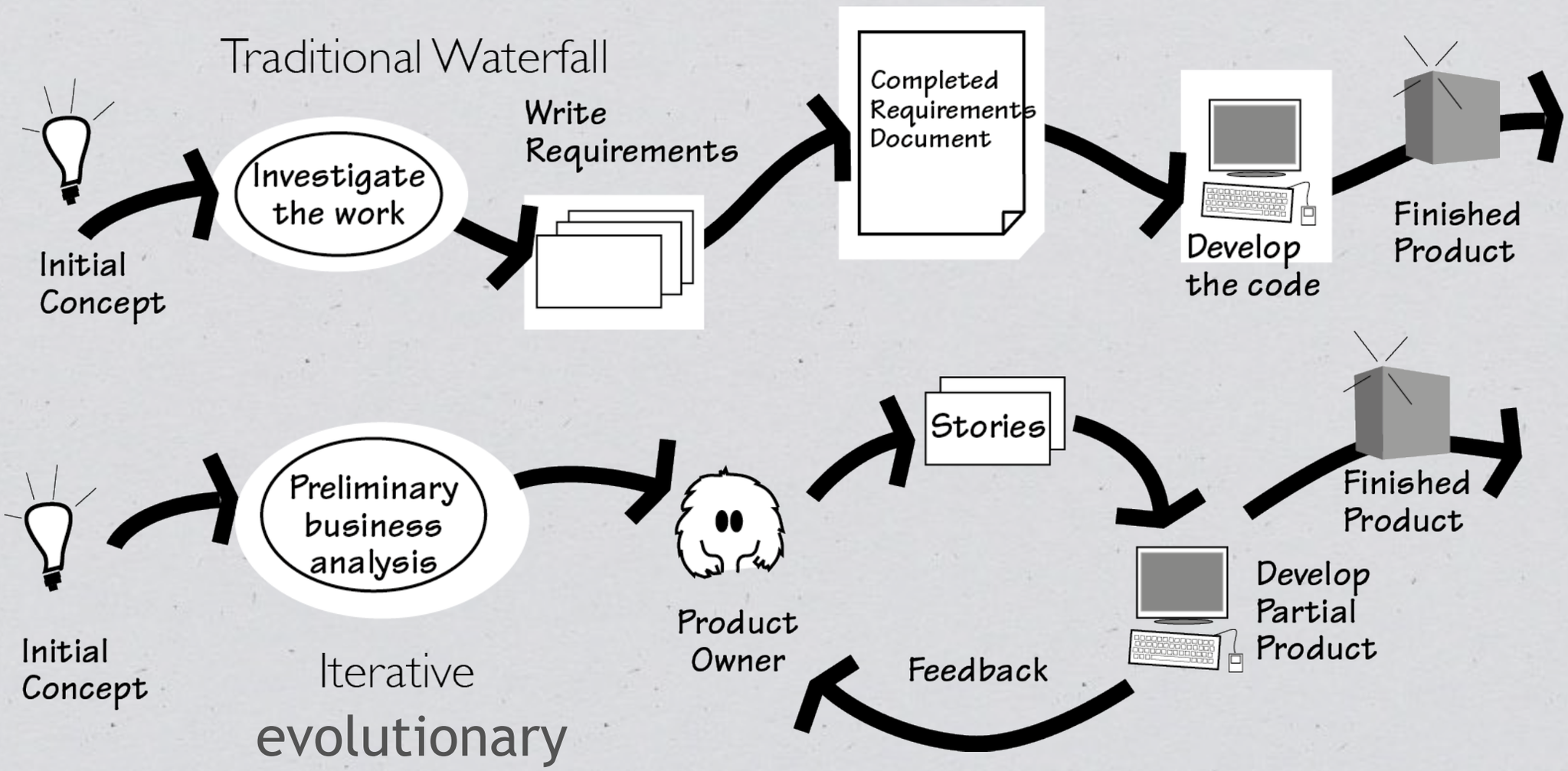
Requirement activities

1. Project Blastoff/Inception/Feasibility study: the business analyst meets with the client and everyone who has interest in the project.
   Decide whether the project is feasible or not.
2. Trawl for Knowledge: Investigate client's work and his current way of doing business.
3. Prototype the business: use sketches/models to clarify client's work.
4. Write the Requirements: in an unambiguous language.

Requirement activities

5. Quality gateway: test the correctness of the requirements. Usually done by the lead requirement engineer and the tester.
6. Reusing Requirements: reuse similar requirements from previous projects.
7. Reviewing the Requirements: check for missing requirements. Check if the requirements are consistent and resolve conflicting requirements.

Software Development Process



Waterfall Model

- ✦Useful if you want to outsource your project.
- ✦Usually used for development of large systems.
- ✦Difficult to add new changes. Does not respond to changing customer requirements.

Evolutionary Model

- The cost of accommodating changing customer requirements is reduced.
- Easy to get feedback from clients.
- Rapid delivery and deployment
- System's structure tend to degrade as new changes are added.
- Incorporating further changes becomes increasingly costly and difficult.

Evolutionary Model

e.g. extreme-programming, scrum

Feasibility Study Typical Deliverables

- ✦Purpose of the project (Goals)
- ✦Scope of the Work
- ✦Stakeholders
- ✦Constraints
- ✦Terminology
- ✦Relevant Facts & Assumptions
- ✦Estimated Cost
- ✦Risks
- ✦Go/no go Decision

You can follow Appendix A in the textbook..

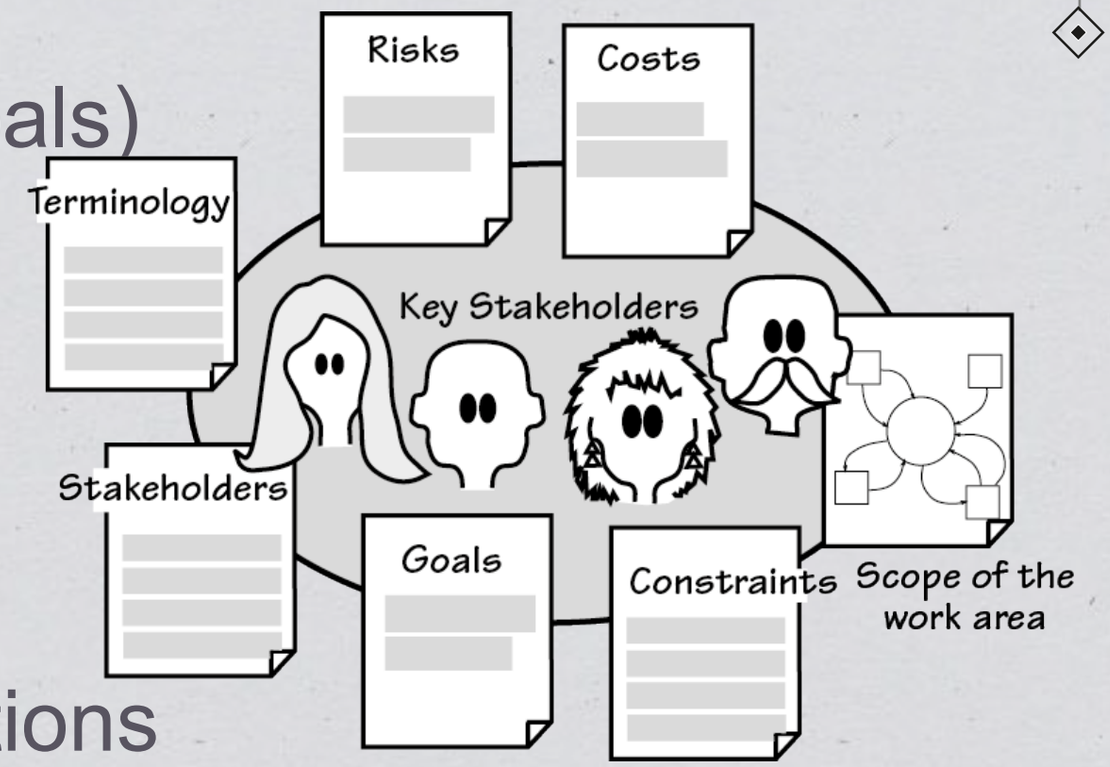


Figure 3.2

The blastoff activity assembles enough information to ensure that the project can proceed smoothly. It also verifies that the project is viable and worthwhile. Most of the outputs serve as the foundation for the trawling activity about to come; the risks and costs are used by project management.

Case Study: IceBreaker

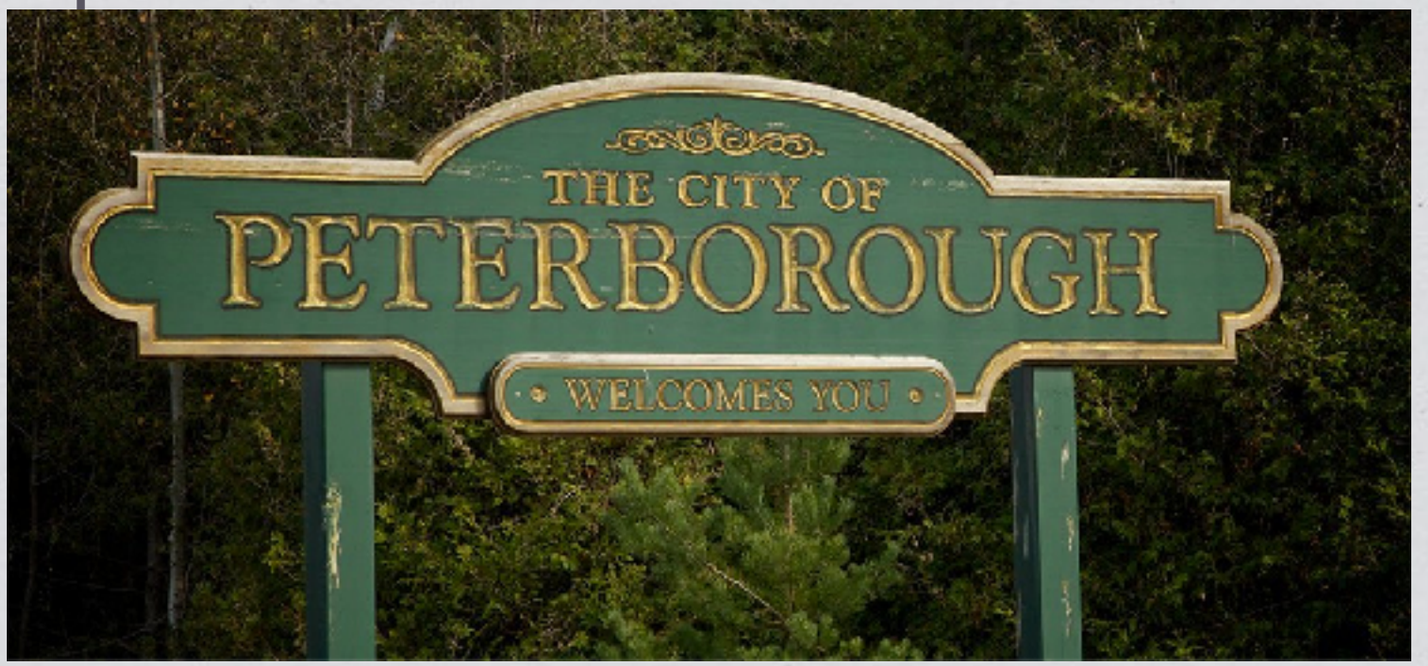
- You work for a software company, and you a responsible for producing the requirements specification.
- The client is an ice breaker company called Saltworks Systems.

Ice Breaker

- IceBreaker uses data from the environment to predict when ice will form on roads.
- It then schedules trucks to treat the roads with de-icing material (a salt compound) before the roads can become dangerous.

Ice Breaker

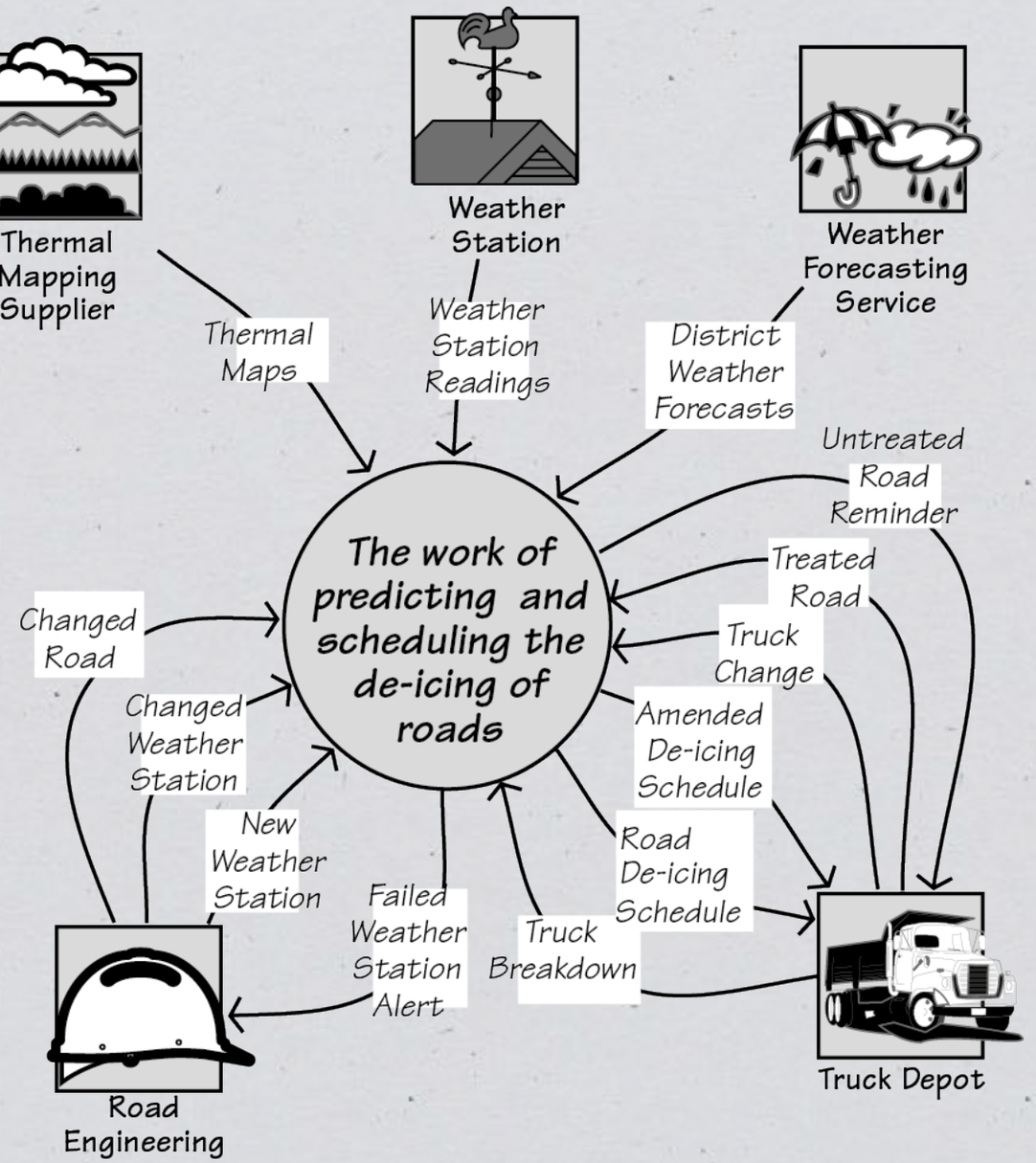
- The first customer for the IceBreaker product is the Peterborough Transportation Department.
- The Transportation Department is responsible for keeping the roads free of ice that is likely to cause accidents; it has agreed to provide expertise and information for you to build the optimally valuable product for the department.



Scope of the Work

Figure 3.5

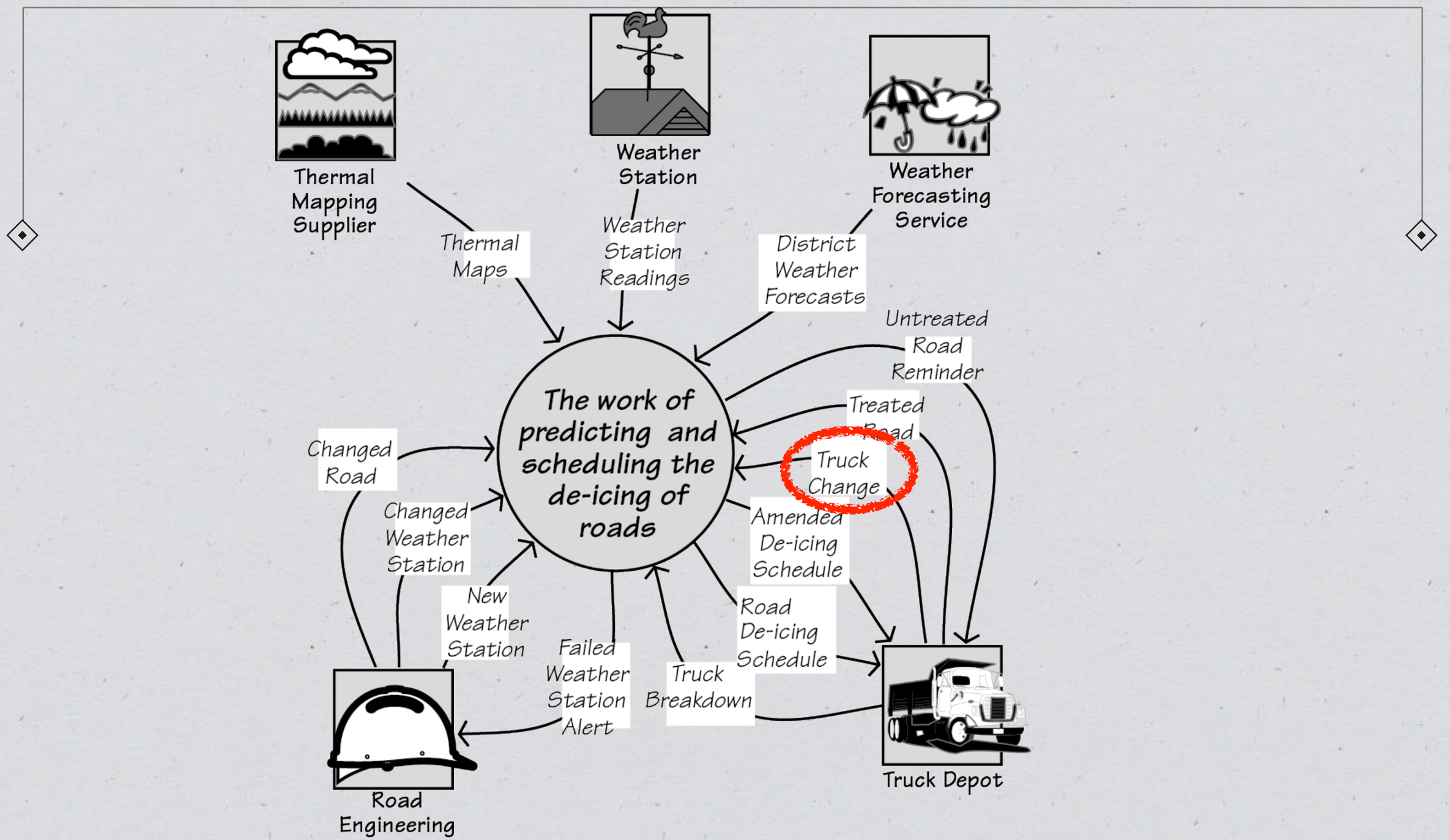
The work context diagram identifies the scope of the work to be studied. It shows the work as a single activity, surrounded by the adjacent systems. The named arrows represent the data that flows between the work and the adjacent systems. The adjacent systems are the activities that you have decided not to study.



Work Context Diagram

Work Context Diagram

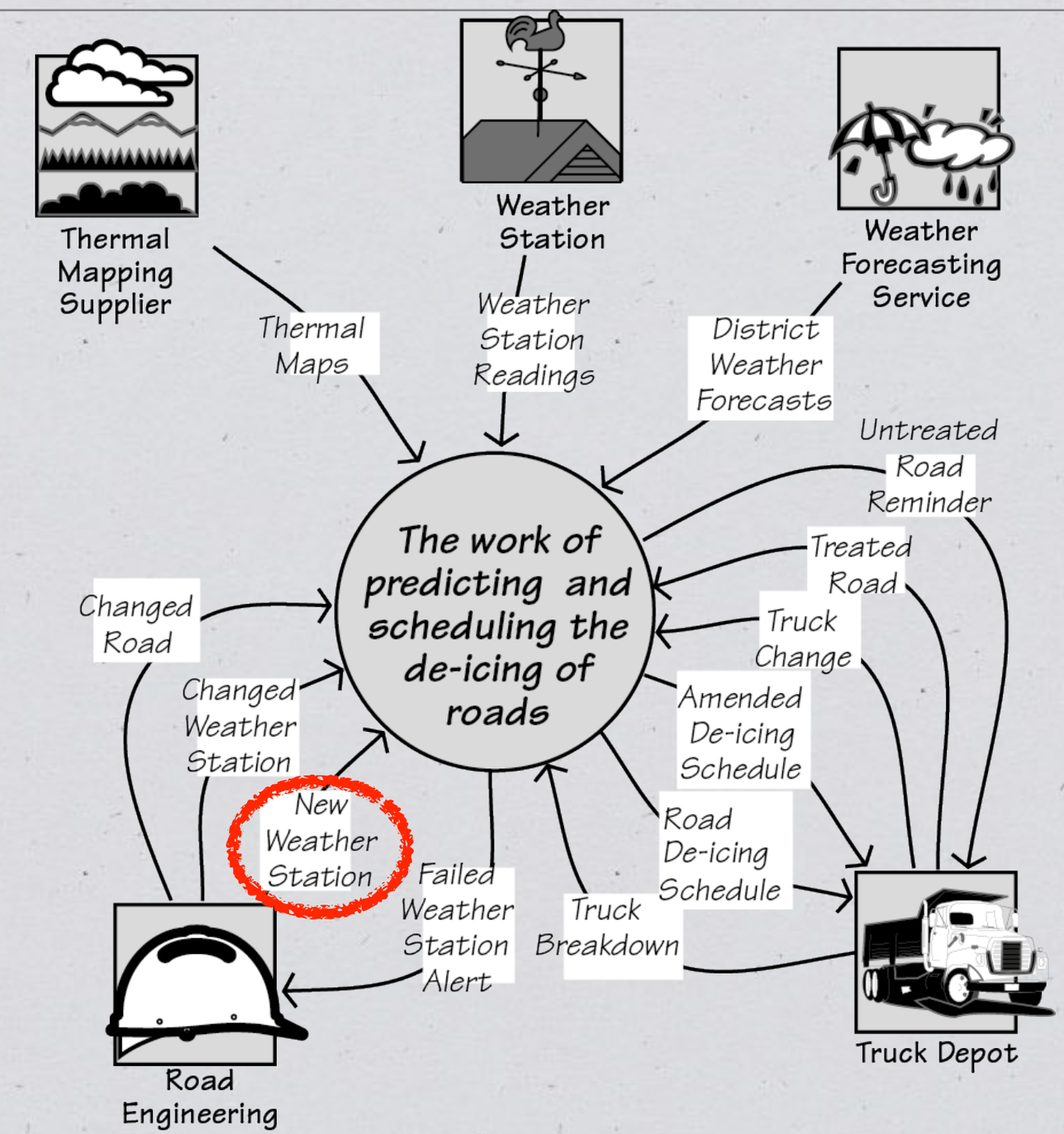
- Shows the scope of the work to be studied.
- Shows the activities that you choose not to study, i.e., the *adjacent systems*.
- The arrows show the data flow between the work and the adjacent systems.
- The context diagram defines the responsibilities of the work and the responsibilities of the adjacent systems.
- We will study the context diagram in more detail in the next lecture..



Responsibilities are defined by the flow of data.

Truck Depot advises the Work of a Truck Change.

Tracking truck changes is the responsibility of Truck Depot



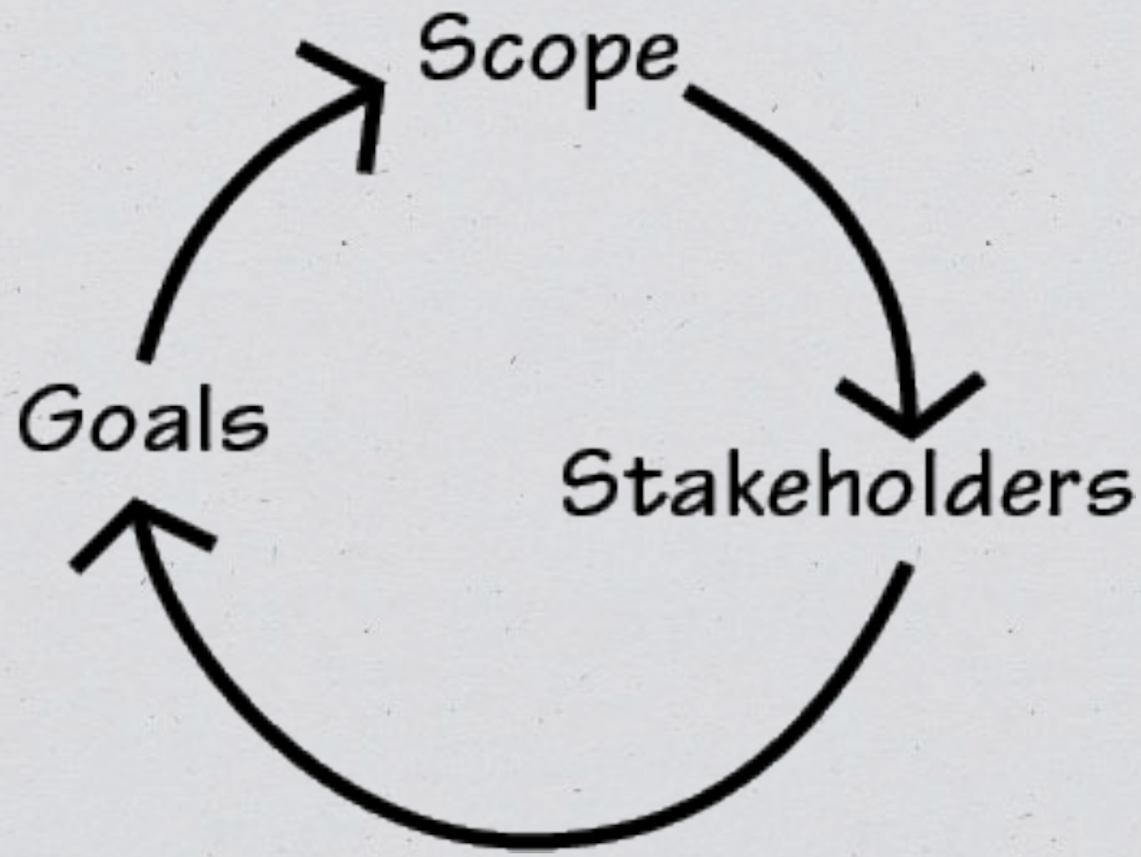
The Road Engineering advises the work of a new weather station.

Tracking of a new weather station is the responsibility of the Road Engineering.

Scope, Stakeholders & Goals

Figure 3.6

The scope, stakeholders, and goals are not decided in isolation from one another. Rather, the scope of the work indicates the stakeholders who have an interest in the work; the stakeholders, in turn, decide what they want the goals of the project to be.



Stakeholders: Sponsor

- **Client (or Owner)**: Someone who pays for the product.
  - Many products do not have direct access to the client/owner
  - Sponsor represents owner's interests.

The sponsor of the project is Mack Andrews, the CEO of Saltworks Systems. He has said that it is his goal to develop this product to appeal to a broader range of markets in other countries, including airports and their runways.

Stakeholders: The Customer

- The customer buys the product. You have to know this person well enough to understand what he finds valuable and, therefore, what he will buy.
- You must understand what appeals to your customers, and what they value. What will they find useful? What will they pay for? Understanding your customer correctly makes a huge difference to the success of your product.

The customer for the product is the Peterborough Transportation Department, represented by director Jane Shaftoe.

Potential customers for the product include all cities in the North America, United Kingdom, and northern Europe and Scandinavia. A summary of the requirements specification will be presented to the Transportation Department managers of selected counties, states, and countries for the purpose of discovering additional requirements.

Stakeholders: The Customer

- Sometimes Customer can be as same as the client/ owner. *Example*: The City of Peterborough wants to upgrade its website. **Client/Owner**: City of Peterborough. **Customer**: City of Peterborough.

Stakeholders: Users

- ✦People who will ultimately be hands on operators of the product
  - You have to bring about a product that your users are both able to use and want to use.
- ✦When developing consumer products or mass market software, consider using a persona
- ✦Record identified users - IceBreaker:
  - Qualified Road Engineers
  - Clerks in Truck Department
  - Managers

- For each category of user, describe the attributes of the users
  - Subject-matter experience: How much help do they need
  - Technological experience: Can they operate the product? Which technical terms should be used?
  - Intellectual abilities: Should tasks be made simpler? Or broken down to a lower level?
  - Attitude toward the job: What are the users' aspirations?
  - Education: What can you expect your user to know?
  - Linguistic skills: Not all your users will speak or read the home language.
  - And most importantly, what is it about their work that they most wish to improve?

- Also identify the particular attributes your product must cater to
  - People with disabilities: Consider all disabilities. This, in some cases, is a legal requirement.
  - Nonreaders: Consider people who cannot read and people who do not speak the home language.
  - People who need reading glasses
  - People who cannot resist changing things like fonts, styles, and so on.
  - People who will probably be carrying luggage, large parcels, or a baby.
  - People who do not normally use a computer.
  - People who might be angry, frustrated, under pressure, or in a hurry.

Other Stakeholders

- ✦Consultants: people who have expertise you need, e.g., security expert.
- ✦Management: e.g., product/program managers.
- ✦Subject Matter Experts: e.g., domain analysts, business consultants.
- ✦Core Team: developers, testers, system architects.
- ✦Inspectors: safety inspectors, technical inspectors, auditors, government inspectors.
- ✦Market Forces: Marketing department representing the marketplace.

Other Stakeholders

- ✦Legal Experts
- ✦Negative Stakeholders: hackers, defrauders.
- ✦Industry Standard Setters: codes of conduct by professional bodies.
- ✦Public Opinion: user groups, polling data.
- ✦Government: for reporting purposes.
- ✦Special Interest Groups: environmental bodies, people with disability.

Other Stakeholders

- Technical Experts: hardware people, usability experts.
- Cultural Interests: political, ethnic, religious interests.
- Adjacent Systems: adjacent systems on your work context diagram are the systems, people, or work areas that directly interact with the work you are studying. From the context diagram find: who are the managers/project leaders of the adjacent systems?

Purpose of the Project: Problem Statement

"Roads freeze in winter, and icy conditions cause road accidents that may potentially kill people. Predictions at the moment rely largely on guesswork, experience, and phoned-in reports from motorists and the police. Trucks do not always get to the icy roads on time to prevent accidents, or they may arrive far too early, which results in the de-icing material being dispersed by the time the road freezes. Road treatment is sometimes indiscriminate, and this wastes de-icing material and causes environmental damage."

Purpose of the Project: Goals PAM

- ✦**Purpose**: What should the product do?
- ✦**Advantage**: Which business advantage does it provide?
- ✦**Measurement**: How do you measure the advantage?

Purpose of the Project: Goals PAM

Purpose:
To accurately forecast road freezing and schedule de-icing treatment.

Advantage:
To reduce road accidents by eliminating icy road conditions.

Measurement:
Accidents attributed to ice shall be no more than one accident per 10,000 vehicle-miles traveled on all roads in the districts covered by the product.

Aspects of Project Goal

- Viable: Given what you understand about the constraints, is it possible for the product to achieve the business advantage?
- Feasible: Given what you have learned from the blastoff, is it generally possible to build a product to achieve the measure?
- Achievable: Does your organization have (or can it acquire) the skills to build the product and operate it once built?

Purpose: To save money on winter road de-icing costs.

Advantage: Reduced de-icing and road maintenance costs.

Measurement:
The cost of de-icing shall be reduced by 25 percent of the current cost of road treatment, and damage to roads from ice shall be reduced by 50 percent.

Advantage: To reduce damage to the environment by unnecessary application of de-icing compounds

Measurement: The amount of de-icing chemicals needed to de-ice the authority's roads shall be reduced to 70 percent of current usage.
Supporting Materials: Thornes, J. E. "Salt of the Earth." Surveyor Magazine, December 8, 1994, pp. 16–18.

Constraints

✦Solution Constraints

- Your specification should describe any mandated designs or solutions.
- Any partner or collaborative applications, such as databases, reporting systems, or web-based systems should also be brought to light and recorded at this time.
- Off-the-shelf and open-source applications, if they are to be used, or interacted with, are recorded under the “Constraints” heading as well.

✦Project Constraints

- Project constraints describe the time and financial budgets for the project

Naming Conventions (Terminology)

- Every project has names that are particular to it, and this terminology should be recorded to make communication easier, and future understanding more reliable.
- Starting to define terminology at scoping time has a distinct advantage: You make the words visible. The stakeholders are able to discuss and change them to reflect the consensus of the meaning.
- Later in the project, this will evolve into a *data dictionary*.

Weather station: Hardware capable of collecting and transmitting road temperature, air temperature, humidity, and precipitation readings. Weather stations are installed in eight locations in Peterborough.

Relevant Facts and Assumptions

- Relevant facts are external factors that have an effect on the product and may not be included in the requirement specifications.

- Relevant fact provides background information for specific readers.
- Example: One ton of de-icing material will treat three miles of single-lane roadway.
- Example: The existing application is 10,000 lines of C code.

Relevant Facts and Assumptions

- Business rules might have an impact on the work/ business/domain that is the source of the requirements.

- Example: The maximum length of a truck driver's shift is 8 hours.
- Example: The engineers maintain the weather stations once a week.

Relevant Facts and Assumptions

- Assumptions: the intention is to make people declare the assumptions that they are making

- Example: Roads that have been treated will not need treatment again for at least 2 hours.

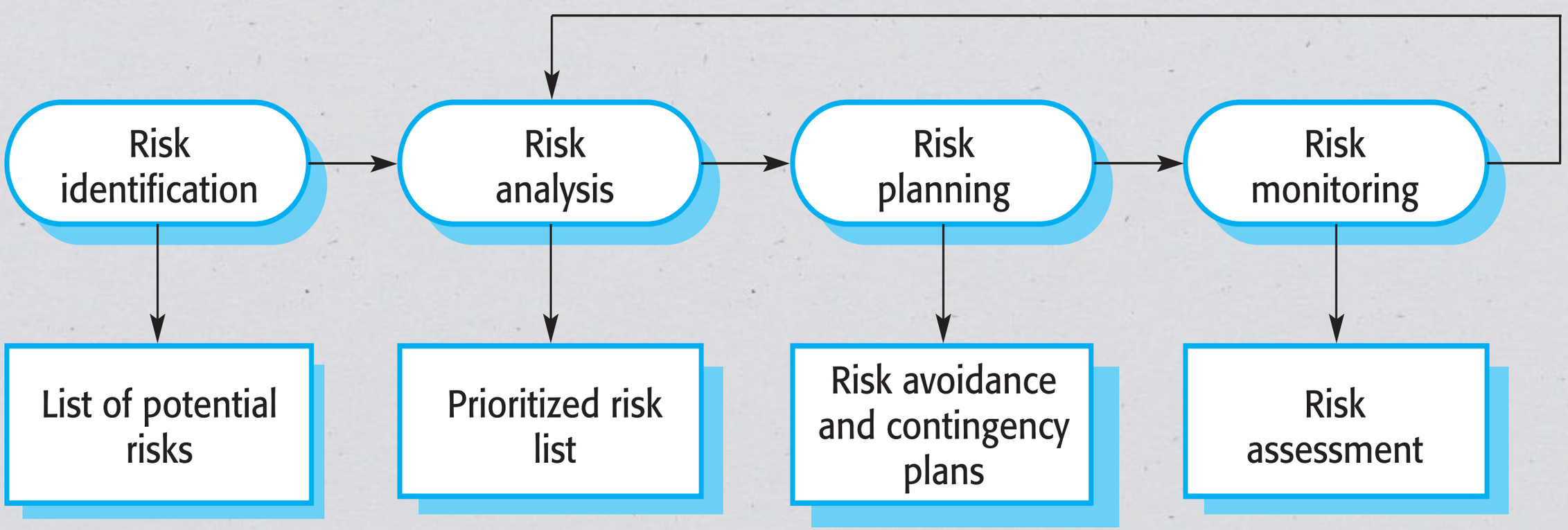
Risks

- This stage should also include an initial risk assessment
- The job is to assess both those risks that are most likely to happen and those risks that will have the greatest impact if they do, in fact, become problems.
- For each identified risk, the assessor determines the probability of it becoming a problem, along with its cost or schedule impact
- Also determines the early-warning signs

The risk management process

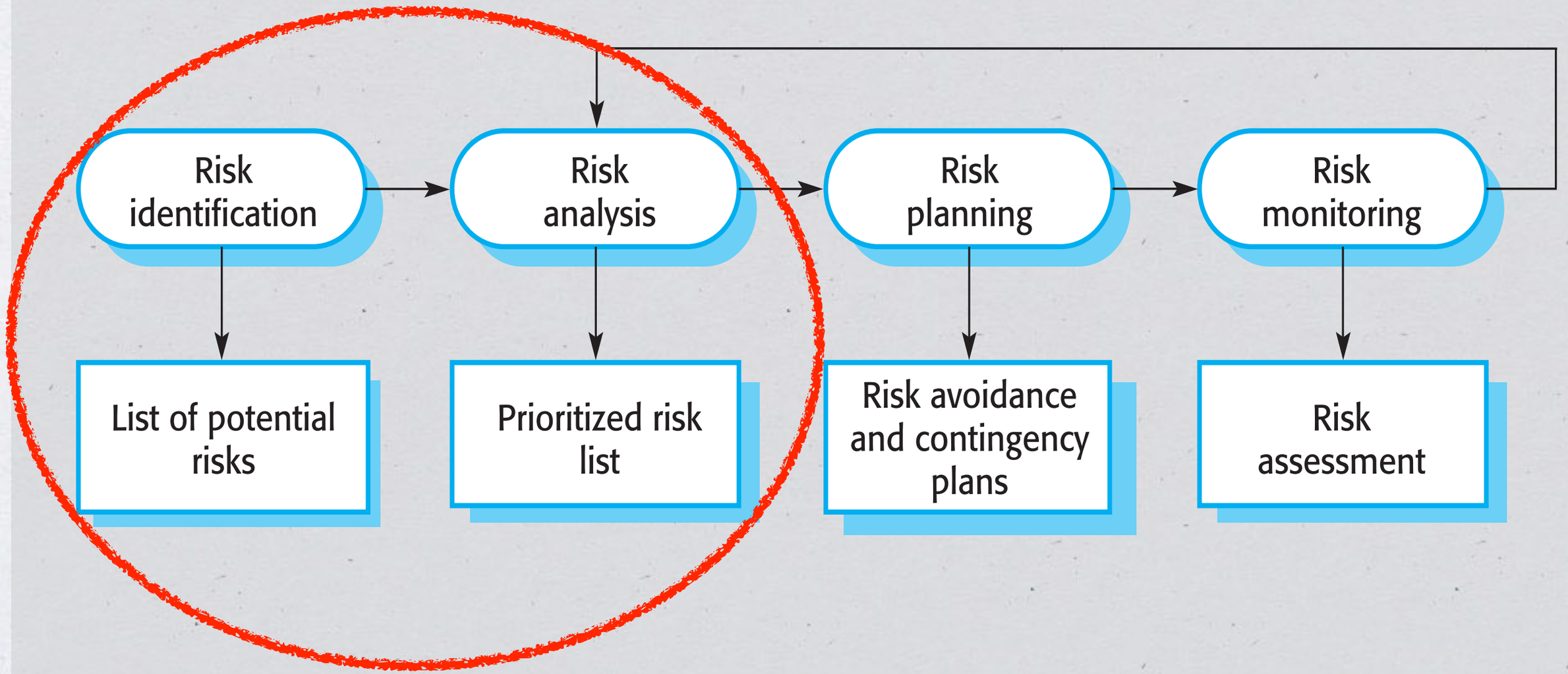
- ✦Risk identification
  - Identify project, product and business risks;
- ✦Risk analysis
  - Assess the likelihood and consequences of these risks;
- ✦Risk planning
  - Draw up plans to avoid or minimize the effects of the risk;
- ✦Risk monitoring
  - Monitor the risks throughout the project;

The risk management process



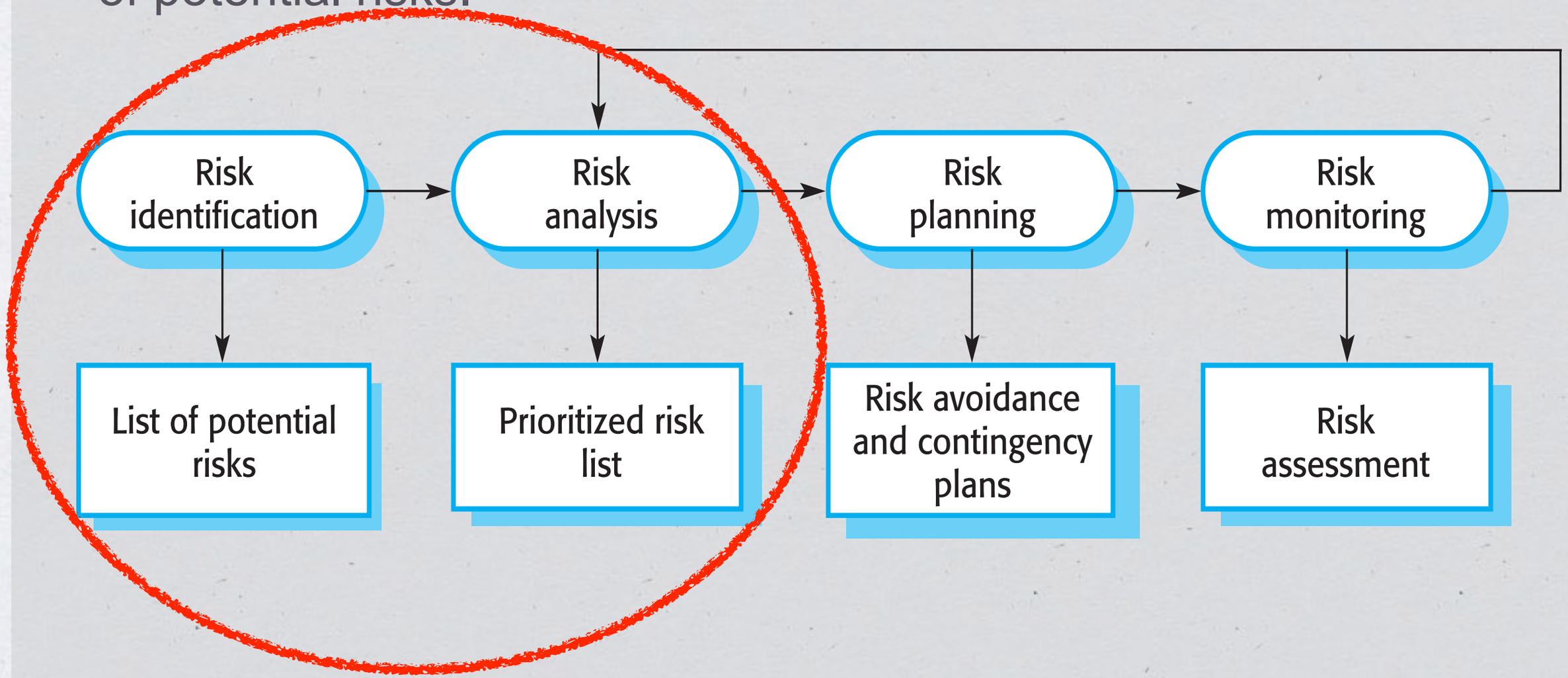
The risk management process

We will focus only on these two activities of risk management



The risk management process

In the feasibility study, you should come up with a prioritized list of potential risks.



Risk identification

- ✦Maybe a team activity or based on the individual project manager's experience.
- ✦A checklist of common risks may be used to identify risks in a project. They can be categorized into the following risk types:
  - Technology risks.
  - People risks.
  - Organizational risks.
  - Tools risks.
  - Requirements risks.
  - Estimation risks.

Examples of different risk types

Risk type	Possible risks
Technology	The database used in the system cannot process as many transactions per second as expected. (1) Reusable software components contain defects that mean they cannot be reused as planned. (2)
People	It is impossible to recruit staff with the skills required. (3) Key staff are ill and unavailable at critical times. (4) Required training for staff is not available. (5)
Organizational	The organization is restructured so that different management are responsible for the project. (6) Organizational financial problems force reductions in the project budget. (7)
Tools	The code generated by software code generation tools is inefficient. (8) Software tools cannot work together in an integrated way. (9)
Requirements	Changes to requirements that require major design rework are proposed. (10) Customers fail to understand the impact of requirements changes. (11)
Estimation	The time required to develop the software is underestimated. (12) The rate of defect repair is underestimated. (13) The size of the software is underestimated. (14)