Enabling the business-based
Internet of Things and Services

(FP7 257852)

D3.1 Enterprises Use Cases

Published by the ebbits Consortium

Dissemination Level: Public

European Commission
Information Society and Media

SEVENTH FRAMEWORK PROGRAMME

Project co-funded by the European Commission within the 7th Framework Programme
Objective ICT-2009.1.3: Internet of Things and Enterprise environments

Document control page

Document file: D3.1_Enterprise_Use_Cases_v2.0.docx
Document version: 2.0
Document owner: Comau

Work package: WP3 – Enterprise Frameworks for Life-Cycle Management
Task: T3.1 – Enterprise and management processes
Deliverable type: R

Document status: ☒ approved by the document owner for internal review
☒ approved for submission to the EC

Document history:

Version	Author(s)	Date	Summary of changes made
0.1	Roberto Checcozzo	2010-10-20	TOC
0.2	Tomas Sabol	2010-11-02	Commented and TOC modified
0.3	Michael Jacobsen	2010-12-20	Use cases added to agricultural domain
0.4	Jozef Glova	2011-01-02	Use cases characterization and methodology
0.5	Roberto Checcozzo	2011-01-03	Use cases added to manufacturing domain
0.6	Viliam Vajda	2011-01-10	Use cases added to generic domain
0.7	Viliam Vajda	2011-01-20	Use cases design
1.3	Tomáš Sabol, Roberto Checcozzo	2011-02-11	General revision
1.9	Ferry Pramudianto Yves Martin	2011-02-18	Peer review
2.0	Roberto Checcozzo, Tomas Sabol	2011-02-28	Final version submitted to the European Commission

Internal review history:

Reviewed by	Date	Summary of comments
Ferry Pramudianto, FIT Fraunhofer	2011-02-20	
Yves Martin, SAP AG	2011-02-25	

Index:

1. Executive Summary 4

2. Introduction 5

2.1 Overview of the ebbits project 5

2.2 Purpose, context and scope of this deliverable 5

3. Use Cases study 6

3.1 General characterization of use cases 6

3.2 Methodology 7

3.2.1 Scenarios 7

3.2.2 UML Use Case diagrams 8

3.2.3 Use Case Narrative 9

4. Use cases 14

5. The Automotive Manufacturing Domain 20

5.1.1 Body welding and assembly (BWA) 20

5.2 Use cases in automotive manufacturing 21

5.2.1 Energy reduction management process 21

5.2.2 Plant production shut down and restart 22

5.2.3 Automatic energy reduction 24

5.2.4 Retrieve production information 25

6. Agricultural Domain 26

6.1 Domain Description 26

6.1.1 Field 26

6.1.2 Feed Production 26

6.1.3 The Farm 27

6.1.4 The Slaughterhouse 29

6.1.5 Retail Store 29

6.2 Use cases in agricultural domain 29

6.2.1 Story for the field area 29

6.2.2 Stories for the feed production 31

6.2.3 Stories for the farm 32

6.2.4 Stories for the slaughterhouse 33

6.2.5 Stores for retail 34

7. Conclusions 35

8. References 36

9. Appendix 1 - Use case diagrams in Automotive Manufacturing Domain 37

10. Appendix 2 - Use case diagrams in Agricultural Domain 50

1. Executive Summary

Ebbits project aims to provide solutions to improve company performances by introducing new technologies, architectures and processes. The document D3.1 describes enterprise and management processes in the manufacturing and in the slaughterhouse domains, which are considered relevant to the application of the ebbits project in the industrial companies involved in the project, providing use cases models for each of these processes.

Use cases provide an easy way to comprehend the functional requirements of the system and also make it easy to identify the various interactions between the users and the systems within an environment. They are descriptive and hence clearly represent the value of an interaction between actors and the system. The use cases clarify system requirements very categorically and systemically making it easier to understand the system and its interactions with the users. They are a powerful tool to understand the system's functionality during the early analysis phase of the project. To provide a standard and easy way to understand model of the use cases, the Unified Modelling Language (UML) has been used. UML is a standardized set of modelling languages developed by Object Management Group (OMG).

The enterprise use cases provided by this document consider and enhance the scenario thinking described in Deliverable 2.1, which has been developed to provide information on the possible scenario of the usage of the ebbits project outcomes. These scenarios were revised and new situations have been considered, so as to enlarge the possible area of actuation of the project development. The use cases reported describe existing, standardized business processes, which can be considered common in the considered domains - the automotive manufacturing domain, directly related to the manufacturing activities of a car body in white, and the agricultural domain section, which describes the process used in the production of pork meat.

Examples of the use cases reported are automatic energy reduction process that could be adopted in an automatic manufacturing plant and integrated in the quality book, the other case is management of the nutrient in soil used in the agricultural domain.

2. Introduction

2.1 Overview of the ebbits project

The ebbits project aims to develop architecture, technologies and processes, which allow businesses to semantically integrate the Internet of Things into mainstream enterprise systems and support interoperable real-world, online end-to-end business applications. It will provide semantic resolution to the Internet of Things and hence present a new bridge between backend enterprise applications, people, services and the physical world, using information generated by tags, sensors, and other devices and performing actions on the real world.

The ebbits platform will support interoperable business applications with context-aware processing of data separated in time and space, information and real-world events (addressing tags, sensor and actuators as services), people and workflows (operator and maintenance crews), optimisation using high-level business rules (energy and cost performance criteria), end-to-end business processes (traceability, lifecycle management), or comprehensive consumer demands (product authentication, trustworthy information, and knowledge sharing).

The ebbits platform will feature a Service-oriented Architecture (SoA) based on open protocols and middleware, effectively transforming every subsystem or device into a web service with semantic resolution. The ebbits platform thus enables the convergence of the Internet of People (IoP), the Internet of Things (IoT) and the Internet of Services (IoS) into the "Internet of People, Things and Services (IoPTS)" for business purposes.

The ebbits platform will be demonstrated in end-to-end business applications featuring connectivity to and online monitoring of a product during its entire lifecycle, i.e. from the early manufacturing stage to its end-of-life. The project will develop, implement and demonstrate two ebbits IoPTS applications. The first application demonstrates real-time optimisation metrics, including energy savings, in manufacturing processes. The second application demonstrates online traceability with the enhanced information on food.

2.2 Purpose, context and scope of this deliverable

Accurate identification of current standard processes, used in industrial manufacturing industries, is a crucial factor for future establishment of new, innovative procedures and processes based on the ebbits platform. Current status within enterprises will be described by using use cases from manufacturing as well as managerial point of view. The precise definition of single roles and procedures, together with decision making processes (process model) within enterprises will allow the semantic decisions models to be created and will also be significant input to the development of new business models to understand value creation in the new environment.

The main objective of this deliverable is to describe such processes, and in addition to the standard enterprise processes, domain specific processes will be described, according to the type, size and focusing of selected enterprises. A set of specific use cases will be collected and described. COMAU will describe the processes and uses cases in manufacturing and TNM in food traceability.

3. Use Cases study

Accurate identification of the current, standard processes, as used in the manufacturing and agriculture domains, is crucial for the future establishment of innovative procedures and processes based on the ebbits platform.

For the purpose to identify and describe the current processes, the use case technique is used to capture system's behavioural requirements by detailing scenario-driven threads through functional requirements.

The use cases in this report are not used for the user requirements elicitation, but rather for description of manufacturing, enterprise, business processes, which potentially in the future can be included also in the ebbits platform-based business models to be developed in WP3.

That helps us to understand the current events contained in the particular processes and thus to understand them within the firm and its environment.

3.1 General characterization of use cases

Use case in system engineering is a description of a system's behaviour as it responds to a request that originates from outside of the system, i.e. a use case describes "*who*" can do "*what*" with the system in question. It naturally describes the things the actors want the system to do, including for example querying the ebbits platform to retrieve data about a specific food in the agriculture domain so that all food would be traceable, or querying the ebbits platform to retrieve the product information relevant to improving the energy efficiency.

According to an alternative definition [8], *use case is behavioural classifier, which specifies behaviour of a subject by describing a set of sequences of actions performed by the system to yield an observable result of some value to one or more actors or other stakeholders of the system* (i.e. each use case describes a unit of complete and useful functionality that the subject provides to its users).

This functionality, which is initiated by an actor, must always be completed for the use case to terminate. Actor is a human user of the designed system, some other system, or hardware using services of the subject.

Each use case focuses on describing how to achieve a goal or a task. This means that a number of use cases are needed to define the scope of a new system.

Use cases and scenarios describe functionality that describes how actors interact with a system. The system could theoretically be anything, but most commonly use cases are used with online or web applications. The term "use case" is often a short version of "use case narrative" or "use case flow of events" and is depicted as an oval in the use case diagram, see Figure 1.

The use case approach is a methodology usually used to managing system requirements, i.e. their identification, clarification and organization [7]. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal. It consists of a group of elements (for example classes and interfaces) that can be used together in a way that will have an effect larger than the sum of the separate elements combined. The use case should contain all system activities that have significance to the users. Use case can be thought of as a collection of possible scenarios related to a particular goal, indeed, the use case and goal are sometimes considered to be synonymous.

Use case (or a set of use cases) has the following characteristics:

- Organizes functional requirements;
- Models goals of the system/actor (user) interactions;
- Records paths (called *scenarios*) from trigger events to goals;
- Describes one main flow of events (also called a *basic course of action*), and possibly other ones, called *exceptional* flows of events (also called *alternate courses of action*);

- Is multi-level, so that one use case can use the functionality of another one.

Use Case technique is used in this report as an essential input to identify roles and deliverables in the specific domains (agriculture and manufacturing). As such it consists of actors - users and systems that interact within the specific domains, and use cases containing sequences of events, through which the actors interact with the domain elements to get their job done.

Altogether, the actors and the use cases describe who is involved in the current activities and how these activities take place.

3.2 Methodology

The following section details the methodology use to describe the current manufacturing/enterprise/business processes. Some of these processes will be included also in the business models to be produced in Task 3.5. The use case technique is used to capture current actors, stakeholders, use cases and interactions among them. To extend the description of the current processes in detail, use scenario, use case narratives and use case diagrams methods will be used.

3.2.1 Scenarios

The use case technique was described above as a technique that is used to capture system's behavioural requirements by detailing scenario-driven threads through functional requirements.

Scenario (often called also *use scenario*) is one path or flow through a use case [9]. Typically a use case has a primary scenario, one or more alternate scenarios, and possibly exception scenarios. For instance, a bank transaction might be completed using a customer's account number (primary path), using the Customer Information system to look up the account number (alternate), or be cancelled (perhaps due to not finding the account number). In this regard, use cases are a collection of related scenarios for accomplishment of a goal.

Principal rules how to develop a use scenario [3, 6, 10] are as follows:

1. Determine the *actors*. The actor has an active role in the scenario. In the case of several actors, more scenarios should be set up;
2. Determine the *goals* the actor has to complete;
3. Determine starting points of the scenario: a *trigger* or an *event*;
4. Identify *stakeholders* and their interests;
5. Determine the number of scenarios that should be created based on the number of actors and their goals;
6. Write the *scenarios*. Work from the starting point towards completing the actor's goals. Be specific about task, sub tasks, context and the actor's motivations to complete the goals.

Actors

Actor is a person, organization, or external system that plays a role in one or more interactions with the system.

Actors interact directly with the system. Actors can be humans, systems, or event/time triggers. For instance in a banking system, actor might be a teller who enters banking transactions and a Customer Information System that supplies and maintains the customer data. They are depicted outside the use case diagram.

System boundary boxes (optional)

A rectangle is drawn around the use cases, called the *system boundary box*, to indicate the scope of the system. Anything within the box represents functionality that is in scope and anything outside the box is not.

Relationships

There are four principal use case *relationships* [12], which are used also in this report:

1. **Extend** - in one form of interaction: a given use case may *extend* another. This relationship indicates that the behaviour of the *extension use case* may be inserted into the *extended use case* under some conditions. Notation is a dashed arrow from the extension to the extended use case, with the label "«extend»". Notes or constraints may be associated with this relationship to illustrate the conditions under which this behaviour will occur.

   Analysts use the «extend» relationship to indicate the use cases that are "optional" to the base use case. Depending on the analyst's approach "optional" may mean "*potentially not executed with the base use case*" or it may mean "*not required to achieve the base use case goal*".

2. **Include** – in another form of interaction: a given use case may provide access to a shared functionality in another use case. The common functionality is housed in a separate use case and is "included" by one or more use cases that share the same functionality. "Include" is a directed relationship between two use cases, implying that the behaviour of the included use case is inserted into the behaviour of the including use case.

   The first use case often depends on the outcome of the included use case. This is useful for extracting truly common behaviours from multiple use cases into a single description. The notation is a dashed arrow from the *including* to the *included use case*, with the label "«include»". This usage resembles a macro expansion where the behaviour of the included use case is placed inline in the *base use* case behaviour. There are no parameters or return values. To specify the location in a flow of events, in which the base use case includes the behaviour of another, you simply write *include* followed by the name of the use case you want to include.

3. **Generalization** – in the third form of relationship among use cases, generalization/specialization relationship exists. The given use case may have common behaviour, requirements, constraints, and assumptions with a more general use case. In this case, describe them once, and the generalization permits to deal with each use case in a common way, describing any differences in the specialized cases. The notation is a solid line ending in a hollow triangle drawn from the specialized to the more general use case (following the standard generalization notation).

4. **Associations** - show which use cases an actor can initiate and which actors a use case can access. Associations between actors and use cases are indicated in the use case diagrams by solid lines. An association exists whenever an actor is involved with an interaction described by the use case. Associations are modelled as lines connecting use cases and actors to one another, with an optional arrowhead on one end of the line. The arrowhead is often used to indicate the direction of the initial invocation of the relationship or to indicate the primary actor within the use case. The arrowheads imply control flow and should not be confused with the data flow.

3.2.2 UML Use Case diagrams

Use Case Diagram is an expression of relations between use cases in a specific system or object and external actors. Use Case expresses functions of the system and how the system functions interact with the external actors. The Use Case Diagrams have four primary elements: actors, use cases, associations, and dependencies [1, 2, 12].

The use case diagrams are used to gather requirements of a system including internal and external influences. These requirements are mostly design requirements. So when a system is analysed to view its functionalities use cases are prepared and actors are identified.

The purpose of the use case diagrams is to:

1. facilitate elicitation of requirements of a system;
2. get an outside view of a system;
3. identify external and internal factors influencing the system;
4. show interactions among the requirements and actors.

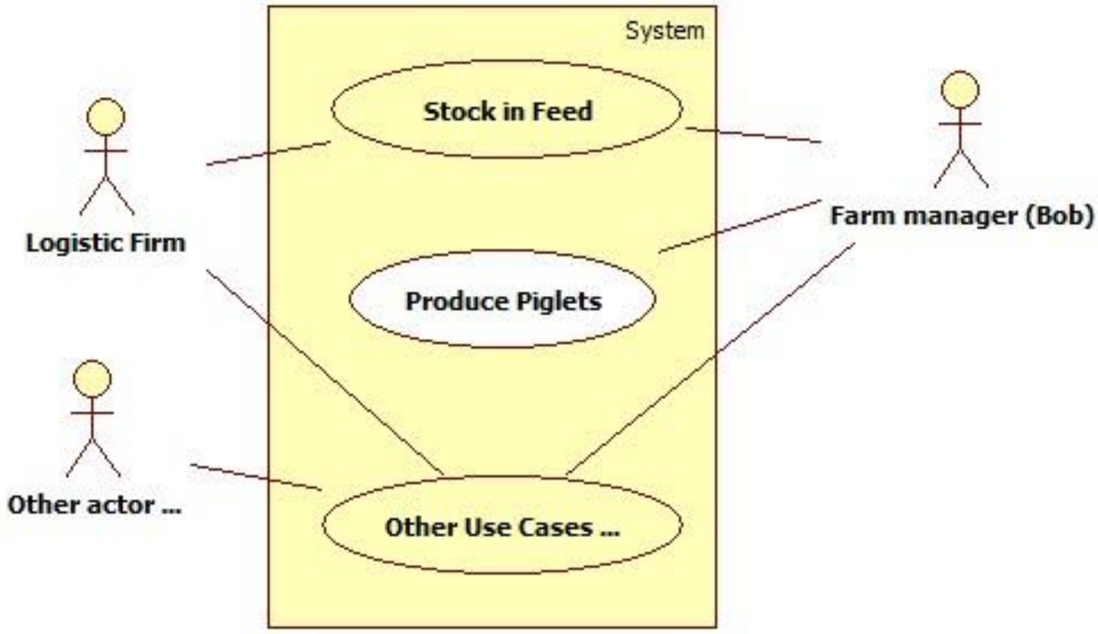


Figure 1. An example of Use Case Diagram

3.2.3 Use Case Narrative

Each scenario is described in a narrative text from operations perspective and contains one or more specific actors. For the structure of Use Case narrative see Figure 2. Use Case and use case diagrams are based on the UML standard. The Use Case Diagrams are used to describe outwardly visible operations of a system and in this Concept of Operations they define the system boundary. The Use Case Diagrams, as already mentioned above, have four primary elements: actors, use cases, associations, and dependencies.

The Use Scenarios address specific needs identified during the stakeholder assessment of a project.

Use Case Name
Version
Author

Last update
Assumptions
Pre-conditions
Successful End Condition
Actors
Use Case Initiation
Main flow
Post-Conditions
Notes

Figure 2. Headlines of Use Case Narrative

Use case name

A descriptive name provides a *unique identifier* for the use case. It should be written in verb-noun format, e.g. "action verb + [qualified] object" (for instance Withdraw Cash). The name should also describe an achievable goal (e.g., *Register User* is better than *Registering User*) and should be sufficient for the end user to understand what the use case is about.

Goal-driven use case analysis will name use cases according to the actor's goals, thus ensuring use cases are strongly user centric. Two to three words is the optimum. If more than four words are proposed for a name, there is usually a shorter and more specific name that could be used. Vague verbs (such as *do* or *process*) and database-oriented verbs, so-called CRUD verbs (create, read, update, delete, get or insert) should be also avoided. A project glossary has to define each object in the use case name [4].

Version

Often a version section is needed to inform the reader of the stage a use case has reached. The initial use case developed for business analysis and scoping may well be very different from the evolved version of that use case when the software is being developed. Older versions of the use case may still be in the current documents, because they may be valuable to different user groups.

Author

This section should list when a version of the use case was created and who documented it.

Last update

It should also list and date any versions of the use case from an earlier stage in the development which are still current documents.

Assumption

This section should list the conditions that generally does not change during the execution and should be true to successfully terminate the use case.

Goal (Successful End Condition)

Without a *goal* a use case is useless. There is no need for a use case when there is no need for any actor to achieve a goal. A goal briefly describes what the user intends to achieve within this use case.

Actors (or stakeholders)

Actor is someone or something outside the system that either acts on the system – a *primary actor* – or is acted on by the system – a *secondary actor*. An actor may be a person, a device, another system or sub-system, or time. Actors represent different roles that something outside has in its relationship with the system, functional requirements of which are being specified. An individual in the real world can be represented by several actors if the individual plays several different roles and has multiple goals in regards to the system. These actors interact with the system and do some action on it.

Stakeholder is an individual or department that is affected by the outcome of the use case. Individuals are usually agents of the organization or department for which the use case is being created. A stakeholder might be called on to provide input, feedback, or authorization for the use case. The stakeholder section of the use case can include a brief description of the functions the stakeholder is assigned to fulfil.

Pre-Conditions

Pre-conditions section defines all the conditions that must be met (i.e., it describes the state of the system) for the *trigger* (see below) to meaningfully cause the initiation of the use case. That is, if the system is not in the state described in the preconditions, behaviour of the use case is indeterminate. Note that the preconditions are *not* the same thing as the *triggers* (see below): the mere fact that the preconditions are met does NOT initiate the use case.

However, it is theoretically possible both that a use case should be initiated whenever the condition X is met and that the condition X is the only aspect of the system that defines whether the use case can meaningfully start. If this is true, then the condition X is both the precondition and the trigger, and would appear in both sections. But this is rare, and the analysts should check carefully that they have not overlooked some preconditions which are part of the trigger.

Use Case Initiation (so-called "trigger")

An 'initiation' section describes an event that causes the use case to be initiated. This event can be external, temporal, or internal. If the trigger is not a simple true "event" (e.g. the customer presses a button), but instead "when a set of conditions are met", there will need to be a triggering process that is continually (or periodically) run to test whether the "trigger conditions" are met: the "triggering event" is a signal from the trigger process that the conditions are met now.

There is varying practice over how to describe what to do when the trigger occurs, but the preconditions are not met.

- One way is to handle the "error" within the use case (as an exception). Strictly, this is illogical, because the "preconditions" are not true preconditions at all now (because the behaviour of the use case is determined even when the preconditions are not met).
- Another way is to put all the preconditions in the trigger (so that the use case does not run, if the preconditions are not met) and create a different use case to handle the problem. Note that if this is a common practice, then the use case template theoretically does not need a preconditions section.

Main flow

At minimum, each use case should convey a *primary scenario*, or typical course of events, also called "basic flow", "normal flow," "happy flow" and "happy path". The main basic course of events is often conveyed as a set of usually numbered steps. For example:

1. The system prompts the user to log on,
2. The user enters his name and password,
3. The system verifies the logon information,
4. The system logs user on to the system.

Alternative paths or Exceptions

Use cases may contain secondary paths or alternative scenarios, which are variations on the main theme. Each tested rule may lead to an alternative path. When there are many rules, the permutation of paths increases rapidly and that can lead to very complex documents. Sometimes it is better to use conditional logic or activity diagrams to describe use case with many rules and conditions.

Exceptions, or what happens when things go wrong at the system level, may also be described, not using the alternative paths section, but in a section of their own. Alternative paths make use of the numbering of the basic course of events, to show at which point they differ from the basic scenario, and, if appropriate, where they re-join. The intention is to avoid repeating information unnecessarily. The description of an exception should indicate how the system will respond to it, or (if possible) recover from, the error condition.

An example of an alternative path could be: "The system recognizes a cookie on the user's machine", and "Go to step 4 (Main path)". An example of an exception path would be: "The system does not recognize a user's logon information", and "Go to step 1 (Main path)".

Post-Conditions

The *post-conditions* section describes what the change in the state of the system will be after the use case completes. Post-conditions are guaranteed to be true when the use case ends.

Notes

Experience has shown that however well-designed the use case template is; the analyst will always have some important information that does not fit under a specific heading. Therefore all good templates include a section (e.g. "Notes to Developers") that allows to record less-structured information.

Activity Diagram

Activity Diagram is a special form of the *state-chart diagram* that is suitable for expressing the activity execution flow. The Activity Diagram is commonly used for expressing workflow, and is frequently used for objects like classes, packages, and operations [5].

Activity diagrams describe the workflow behaviour of a system. Activity diagrams are similar to the state diagrams since activities are the state of doing something. The diagrams describe the state of activities by showing a sequence of activities performed. Activity diagrams can show activities that are conditional or parallel.

We use activity diagrams also in the conjunction with the use case diagram techniques. The main reason to use activity diagrams is to model the workflow behind the system being designed. Employment of the activity diagrams is very useful for analysing a use case by describing what actions need to take place and when they should occur.

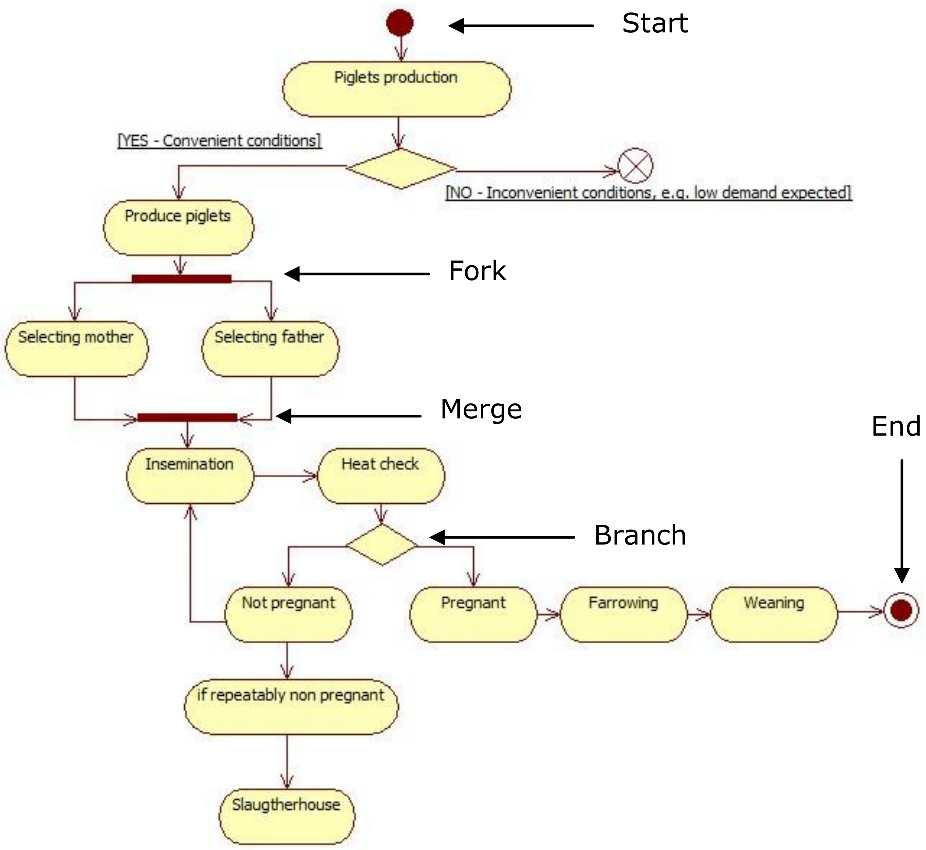


Figure 3. An example of the Activity Diagram

To sum it up, Use Cases are an important and widely used technique for capturing interaction requirements. They provide a structured means to uncover many hidden and detailed requirements, and they can lead to related interface and data requirements.

4. Use cases

Presented Use Cases are divided into:

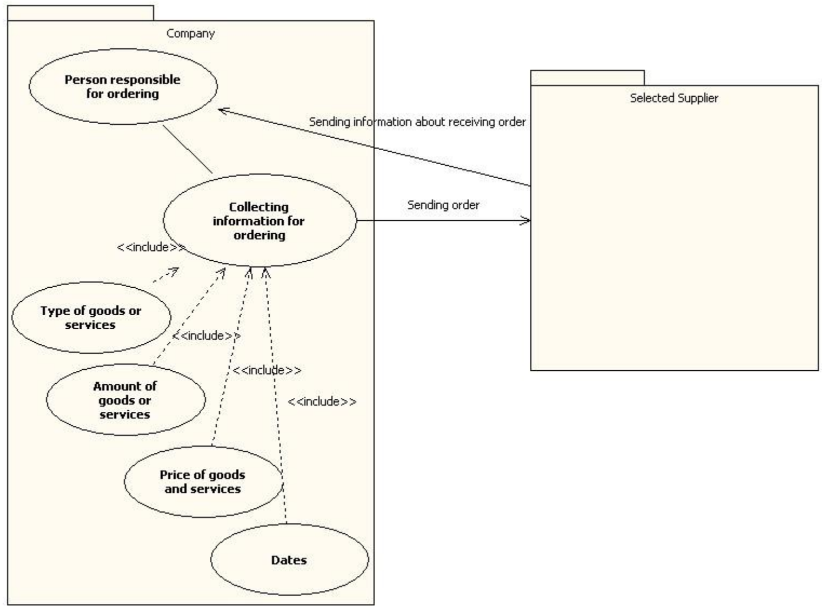
- General Use Cases – for all types of companies/businesses
- Sector specific – specific for ebbits domain

General Use Cases

In the whole economic environment, regardless of sectors, there exist some common processes for all kinds of businesses. Every company, that produces products and/or services, uses standard processes like ordering, invoicing, accounting and payment, and are slighly out of the scope of the ebbits project, as we would like to concentrate to the MES related processes. These, mostly managerial, processes are carried out together with standard business processes in the company. This section will describe the general use cases of managerial processes.

Order goods

Use Case Name	Order goods or services
Version	1.0
Author	TUK
Last update	13/1/2011
Assumptions	
Pre-conditions	Purchasing managers has selected supplier
Successful End Condition	Order sent
Actors	Person responsible for ordering (Manager), Selected Supplier
Use Case Initiation	Use Case starts when the responsible person in the company receives decision about purchasing of goods or services
Main flow	1. Manager collects data for ordering - this data should be stored in management system and usually contain: • Type of goods or services (should contain ID if it is standardised) • Amount of goods or services • Price of goods or services • Dates (including the date of issue, expiry date, or potential delivery date) 2. Manager sends order to the selected supplier 3. Supplier sends acknowledgement of the received order
Post-Conditions	



Note:

The decision about purchasing goods or services depends on the size of the company and the managerial structure.

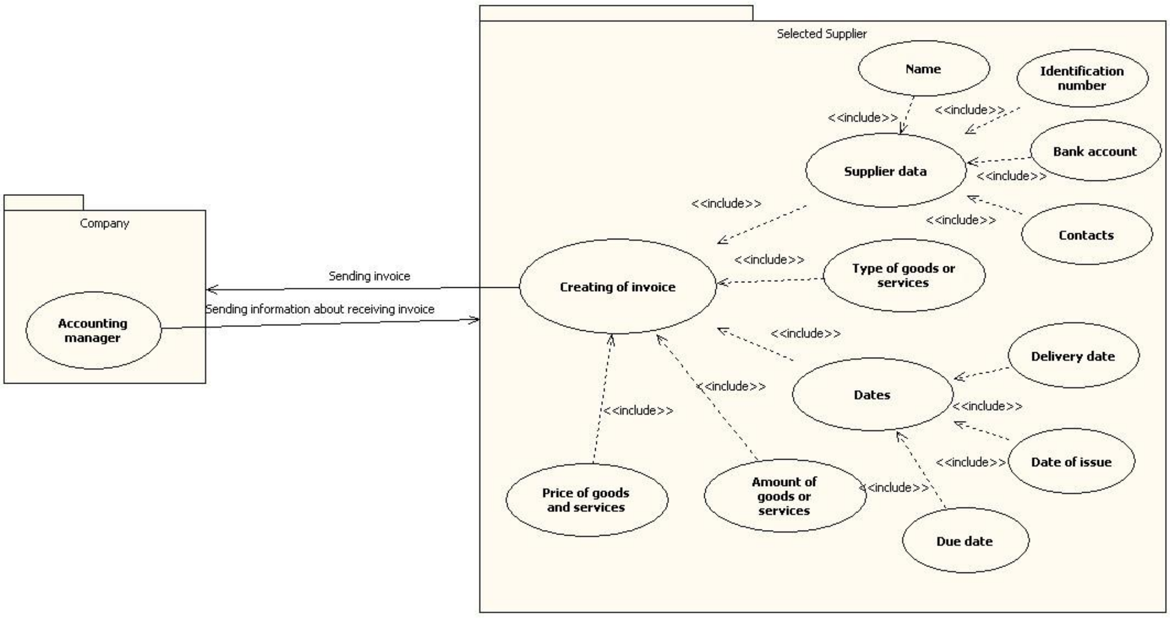
Mostly in the SMEs the decision maker and the person responsible is the same person.

Bigger enterprises usually have specialized procurement departments and the process of decision making and specially process of selecting supplier can differ (depending on the type, size among the companies) and depends on individual internal rules, procedures, methods and practices in the enterprise.

Send invoice

Use Case Name	Send Invoice
Version	1.0
Author	TUK
Last update	13/1/2011
Assumptions	
Pre-conditions	Supplier received order, supplier deliver ordered goods or services
Successful End Condition	Invoice sent
Actors	Supplier, Person responsible for accounting (Accounting manager)
Use Case Initiation	Use Case starts when ordered goods or services are delivered
Main flow	1. Supplier creates invoice according to delivered goods or services including information on: • Supplier data (name, identification number, bank

	account, contacts) • Type of the goods or services • Amount of the goods or services • Price of the goods or services • Dates (including the date of issue, delivery date, due date) 2. Supplier sends invoice to the company 3. Company receives invoice and sends information about receiving invoice
Post-Conditions	

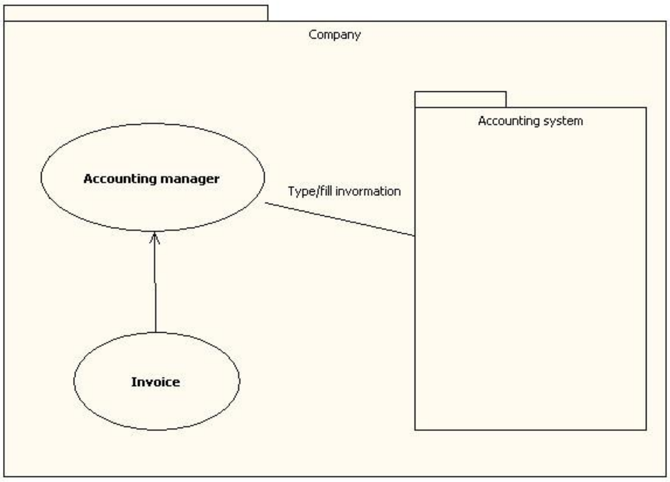


Note:

Record data to the accounting system

Use Case Name	Record data to the accounting system
Version	1.0
Author	TUK
Last update	13/1/2011
Assumptions	
Pre-conditions	Invoice received, Accounting system in the company, Accounting manager has access to the accounting system
Successful End Condition	Information recorded in the accounting system
Actors	Accounting manager, accounting system

Use Case Initiation	Use Case starts when the invoice is received
Main flow	1. Accounting manager logs into the accounting system 2. Accounting manager type/fill all information from the invoice 3. Accounting manager logs out from the system
Post-Conditions	System performs check especially about due dates



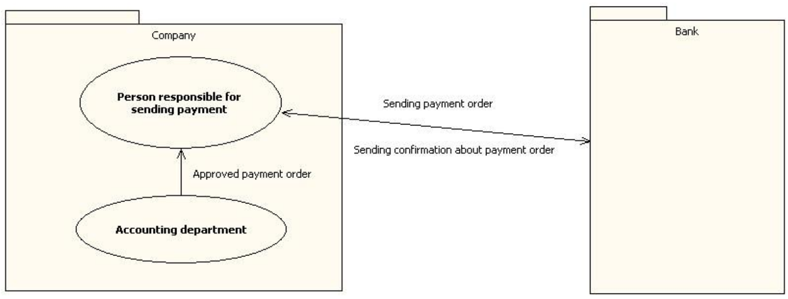
Note:

The accounting system and compulsory information recorded in the accounting system depend on national rules.

Send payment

Use Case Name	Send Payment
Version	1.0
Author	TUK
Last update	13/1/2011
Assumptions	
Pre-conditions	Company has a bank account
Successful End Condition	Payment sent
Actors	Person responsible for sending the payment, Bank
Use Case Initiation	Use Case starts when the responsible person in company receives decision about the sending payment to the supplier
Main flow	1. Person responsible for sending the payment receives the decision about sending payment to the supplier (e.g. from the accounting department) 2. Person responsible for sending the payment send

	order to the bank 3. Bank confirms receiving the payment order
Post-Conditions	Bank transfer amount of money to the account of the supplier



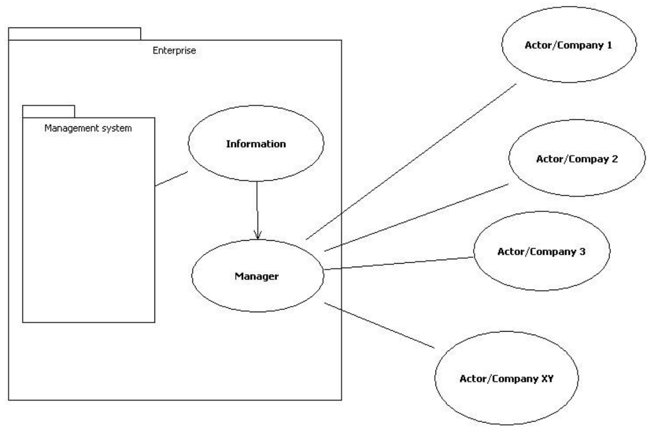
Note:

The decision about sending the payment to the supplier depends on, as in the first use case, the size of company and the managerial structure.

In bigger enterprises usually the accounting department sends info about the due date of delivered goods and services (or prepared payment order) to the responsible person (manager) and the manager approves the payment. That is of course true under „normal operation" (except bankruptcy, etc.).

Request for information

Use Case Name	Request for information (RFI)
Version	1.0
Author	TUK
Last update	13/1/2011
Assumptions	
Pre-conditions	
Successful End Condition	Person responsible for collecting information receives requested information
Actors	Person responsible for collecting information (manager), Actors
Use Case Initiation	Use Case starts when the responsible person in company recognizes that new information is needed
Main flow	1. Manager gathers necessary data 2. Manager selects actors (enterprises) 3. Manager sends data to the selected actors 4. Manager receives requested data
Post-Conditions	Manager works with the received data



Note:

Manager should request information about price, capacity (ability to deliver goods or services) or about delivery dates.

Manager has also several methods how to select actors (enterprises). E.g. he can use internal cataloque or external firm (this could be advisory company, chamber of commerce or another institution).

5. The Automotive Manufacturing Domain

5.1.1 Body welding and assembly (BWA)

Body in white plant mission is to assemble a multitude of panels and braces by welding, bolting or gluing processes all the parts that compose a car body. The following picture depicts the main components that are assembled during the processes performed in the plant.

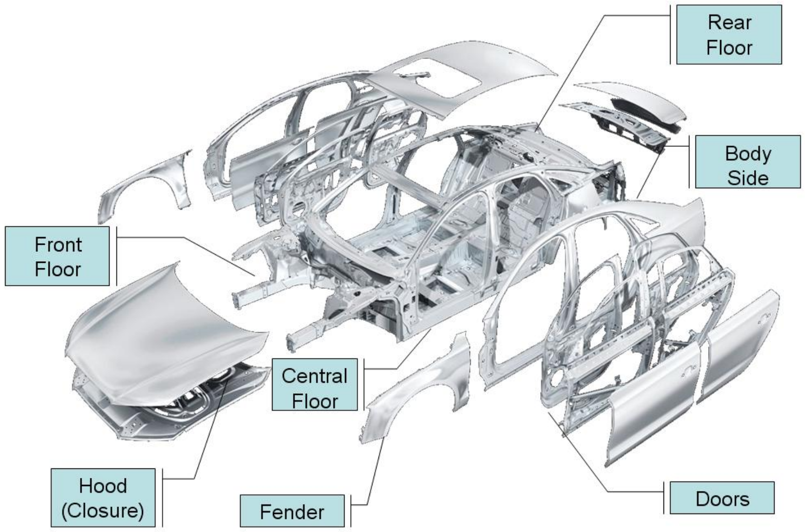


Figure 4. Body in white components

As the forming body moves down the assembly line, held in place by clamping fixtures, the shell of the vehicle is built. First, the left and right quarter panels are robotically disengaged from pre-staged shipping containers and placed onto the floor panel, where they are stabilised with geometric fixtures and welded.

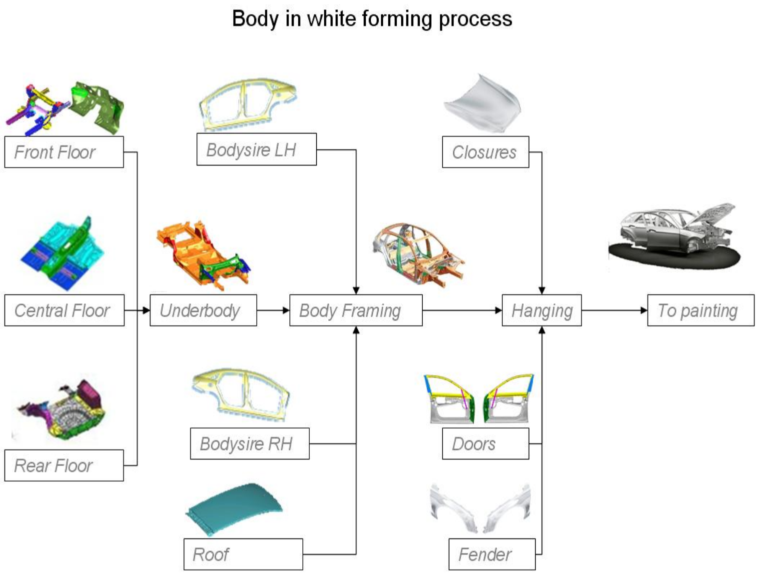


Figure 5. Body in white manufacturing forming process

The front and rear door pillars, roof, and body side panels are assembled in the same fashion. The shell of the automobile assembled in this section of the process lends itself to the use of robots because articulating arms can easily introduce various component braces and panels to the floor panel and perform a high number of welding operations in a timeframe and with a degree of accuracy no human workers could ever approach. Robots can pick and load roof panels and place them precisely in the proper position for welding, assisted by vision devices. For these operations several electronic devices are used, as RFID to identify the part to be worked, electronic controllers to manage the proper sequence of operations, drives and position sensors to perform accurate positioning and more. Considering the harsh environment of this context, the used of wireless devices is not yet widely accepted, as they could not work in a predictable way.

The body is built up on a separate assembly line from the chassis. Robots once again perform most of the welding on the various panels, but human operators are necessary to bolt the parts together. During welding, component pieces are held securely in a jig while welding operations are performed.

As the body moves from the isolated welding area of the assembly line, subsequent body components including fully assembled doors, deck lids, hood panel, fenders, trunk lid, and bumper reinforcements are installed. Although robots help workers place these components onto the body shell, the workers provide the proper fit for most of the bolt-on functional parts using pneumatically assisted tools.

5.2 Use cases in automotive manufacturing

In this section the current standard processes, relevant to the ebbits platform and used in industrial manufacturing industries are identified and described. This activity is crucial to establishment of the future new innovative procedures and processes based on the ebbits platform.

The potential impact of the ebbits platform will be made visible by demonstrating the ability of ebbits IoPTS applications to manage production optimisation with special emphasis on energy consumption.

The ebbits architecture is deployable in many different manufacturing domains characterised by different forms of manufacturing and thus different challenges of optimisation and coordination in the physical world: Process industries (e.g. pharmaceutical), stream industries (e.g. meat production) and assembly industries (e.g. automotive manufacturing). However, all these considered scenarios are characterized by dynamically changing conditions due to mass product customization, constant shortening of product life-cycle as well as plant maintenance and/or modernization. For simplicity, the assembly industry has been selected as the industrial domain to be used for the deployment of prototypes in the ebbits project.

5.2.1 Energy reduction management process

Nowadays, the automotive manufacturing plant considers limited management of energy parameters. Especially, no automatic systems to collect information on energy consumption exist at the device level, thus implying that any analysis to reduce energy consumption is based on data that are not consolidated.

The process is structured as follows:

1) a request of performance improvement on the device X is issued by the maintenance manager to the employee. Example of devices that require particular attention in management of energy could be: spot welding process, metal cutting shaft motor, body in white transportation elevator, laser welding source, etc.;
2) The employee sends a request to an external firm (or performs it himself if he has sufficient technical skills) for an initial assessment of the energy consumption;
3) On the basis of the data collected, the employee writes a strategy to reduce the energy requirement of the device or the process and shares it with the management;
4) The management analyzes the plan and evaluates economically the feasibility and impact, then decides about the proposed improvement;

5) If the proposed improvement is accepted, the employee plans the intervention on the machine and executes it;
6) The employee performs periodical monitoring of the modification performed retrieving the data on consumption and reports these data to the management;
7) The management assesses the data and decides if there is a need to perform other activities.

Figure 6 maps selected activities in the automotive area including managerial activities. Activity diagram corresponds to the use cases 5.2.1.1 – 5.2.1.6 (see Appendix 1) as well as to the general use cases described above.

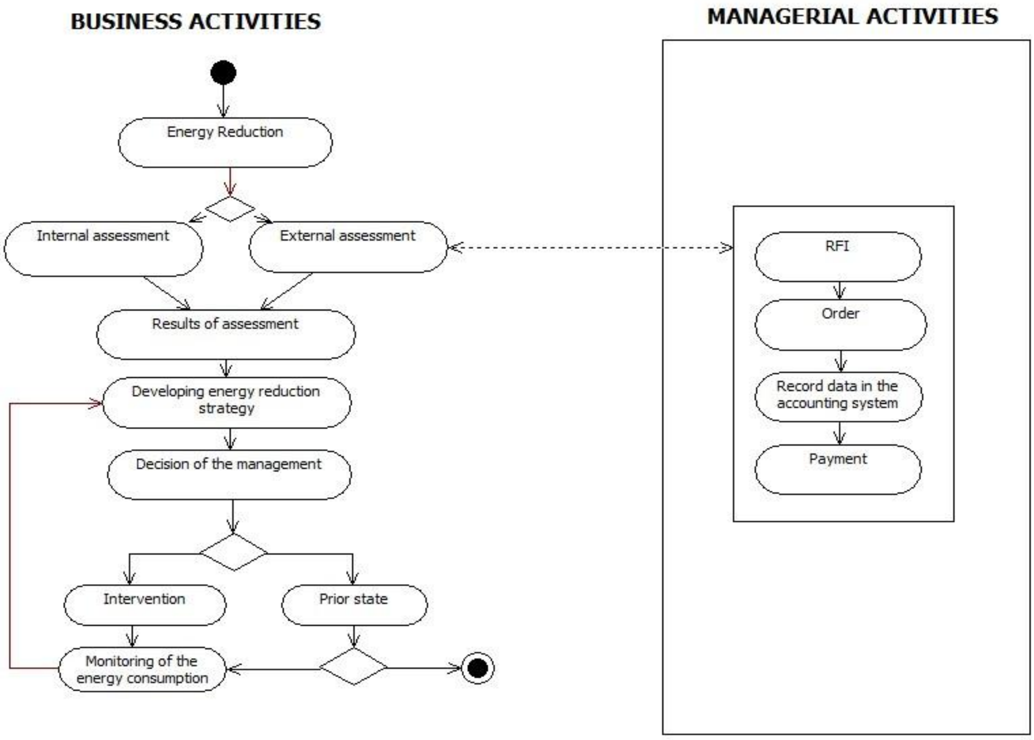


Figure 6. Energy reduction process

5.2.2 Plant production shut down and restart

Eliminating the energy consumption of devices when the plant is not producing requires complex operation to ensure a proper start up, as any error could infer damages at mechanic and electric devices. The process described below is nowadays manually performed by the operator. In the future, with an augmented capacity provided by ebbits platform in term of signals collection and data analysis, most of this process could be automatized.

Shutdown process requires at least the following activities:

1) To empty the automatic line from the part to produce;
2) To verify that all the machines are in the rest position;
3) To close the water valves;
4) To close the compressed air valves and verify if there is some mechanical movement that is changing its position;

5) To verify the efficiency of the back-up batteries of the PLC and Drives;
6) To save the list of the work in progress (typology of part to produce) and the related quality record;
7) To save the program running and the status of the variables for each electronic device used in the automatic line;
8) To shut down the electric power of automatic line;
9) Completely clean the machines.

To turn on the line the following steps must be performed:

1) To verify the status of all the mechanical components;
2) To turn on the electronic devices;
3) To turn on the compressed air;
4) To place all the miss-positioned mechanic devices in the rest position;
5) To turn on the water;
6) To perform a dry run of the machine;
7) To verify with the production of one element that everything is working properly;
8) To start the production.

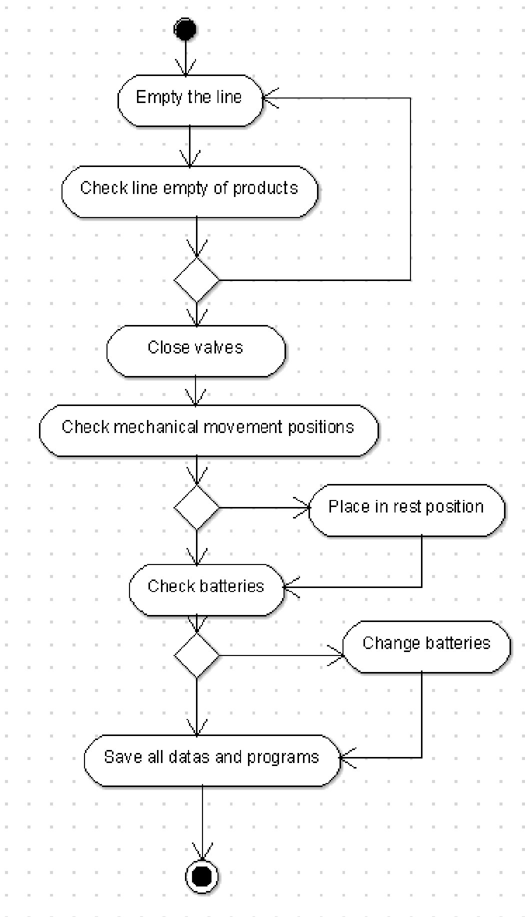


Figure 7. Gracefully shut down of an automatic line

5.2.3 Automatic energy reduction

Reducing energy consumption when the automatic plant is producing at a lower rate requires advanced monitoring and control capacity. On the monitoring side, it requires to retrieve data from the production status of the plant, from the status of the logistic activities, from the situation of the market. All this information can be combined to set-up the speed of the plant in order to reduce the energy consumption per part produced.

Such process requires the following operations:

1) To collect data on the work-in-progress element distributed in the production flow;
2) To collect data on the static and dynamic costs of the production:
    a. Costs of the manpower
    b. Costs of the consumables
    c. Logistic costs
    d. Cost of energy
3) To collect data from the production plan:
    a. number of order requested
    b. number of order already delivered
    c. number of order in work-in-progress
4) To collect data from the logistic status of the plant:
    a. Element and consumables to be worked present in the plant
    b. Element and consumables in transit from the supplier to the plant
    c. Element and consumables ordered to supplier
5) To perform elaboration of data and produce the following output:
    a. Request a reduction of the production speed of the plant
    b. Require a discount on the element or consumables to the supplier
    c. Enable discount to the customer

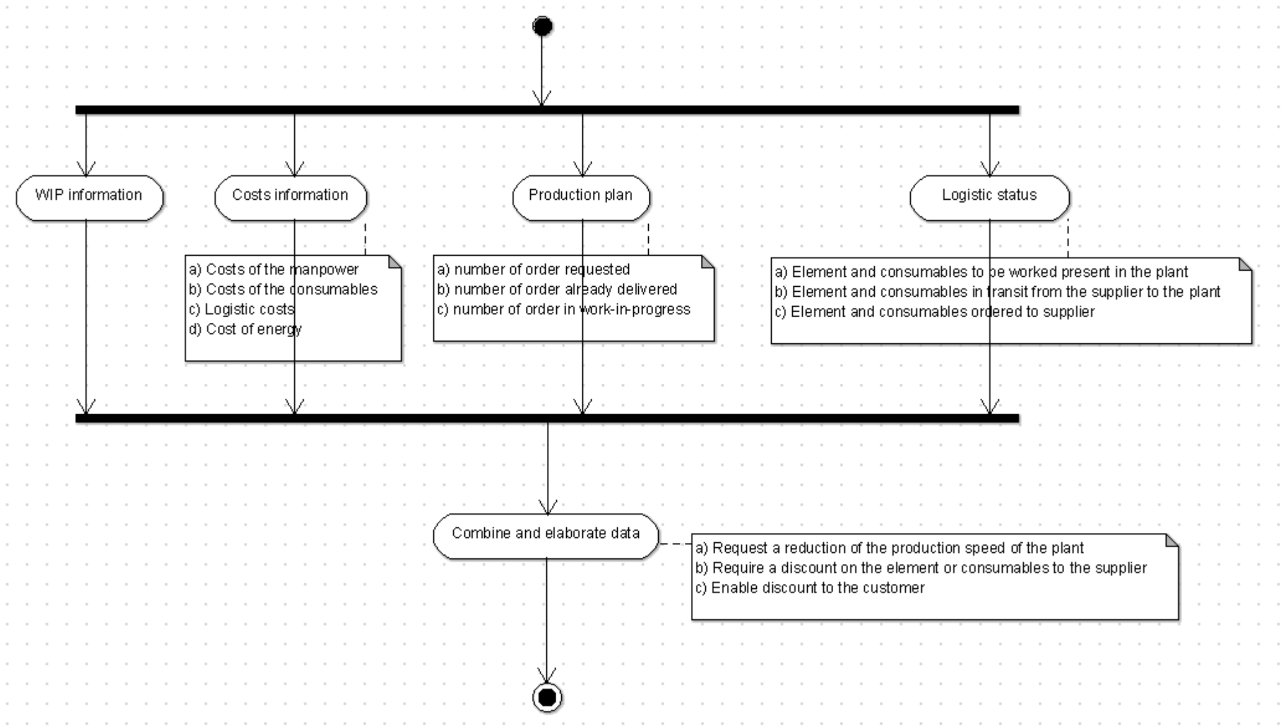


Figure 8. Energy reduction process

Reduction of the speed will reduce significantly the acceleration, the wearing, the consumables, etc. In this way the cost per part produced and the production plant CO2 footprint in the specific situation will be reduced.

On the other hand, the tuning of the production speed will provide the possibility to manage the logistic chain and then require reduction of cost to the supplier.

5.2.4 Retrieve production information

1) Machine operator must fill a form with information related to the machine stop due to:
    a. Faults (mechanical crash, electronic devices failures, improper use of the machine, …),
    b. Machine starved (missing part in input)
    c. Machine blocked (the output buffer is full)
    d. Machine consumable refills (tips, bolts, etc)
    e. Machine planned maintenance activities
    f. Machine unplanned maintenance activities
2) An operator collects each day information on the machine status from the machine operators and the production numbers:
    a. element produced
    b. scrap parts
    c. availability times
3) In the office the data are store manually by the operator in databases of the MES and the system calculates the OEE.
4) The data are placed in Powerpoint and presented to senior management.

6. Agricultural Domain

In order to get a pork chop on the plate a number of steps have taken place. The steps are visualized in Figure 9.

Figure 9. From field to plate.

Crops are grown on the field. The crops are processed to feed at the feed production facility. The feed is used for fattening of pigs. The pigs are slaughtered at the slaughterhouse. Finally the consumer buys a piece of meat at the retail store.

We will go through the entire chain of steps with a strong focus on the farm. Afterwards we will go into details with a number of use cases for each part of the chain.

6.1 Domain Description

6.1.1 Field

Most fields can be harvested once a year. The typical process used to obtain crops is:

1) The field is ploughed and harrowed in order to prepare the soil for seeding. Ploughing turns over the upper layer of the soil which brings fresh nutrients to the surface. Furthermore, weeds and the remains of previous crops are buried, allowing them to break down. It also aerates the soil, and allows it to hold moisture better.
2) The field is seeded with the desired type of crops.
3) The field is fertilized. Both synthetic and organic fertilizers are used.
4) The field is sprayed with pesticides and herbicides.
5) The field may be watered.
6) The crops are harvested.

All of these steps (except the watering) are executed with the help of machines such as tractors and combine harvesters. These machines are often fitted with GPS devices which help optimize the use of fertilizers, pesticides and herbicides.

The type of crop on a particular field is rotated in order to avoid the build-up of pathogens and balancing of nutrients over the years.

6.1.2 Feed Production

The harvested crops are transported to a feed production facility. We will skip discussing the crops that are processed for human consumption, such as the wheat processed into flour.

At the feed production plant crops are mixed with various other components to produce feed with specific nutritional values. The other components can be:

- Residual parts from other food processing such as barley residue from beer brewing;
- Fishmeal;
- Minerals.

6.1.3 The Farm

Farms with animals come in different shapes. We will focus on the pig producing farm and briefly cover the farms producing dairy and poultry.

Pig Production

Pig production can be divided into two areas of expertise:

1) Production of piglets
2) Fattening of piglets

The fattening is sometimes divided into two sub-categories: Fatting from 7 kg to 30 kg and fattening from 30 to 100 kg. However, we will consider the fattening as one category.

Production of Piglets

In order to produce piglets the farmer needs to work intensely with his sows. Figure 10 shows the essential events in the life cycle of a sow on a farm.

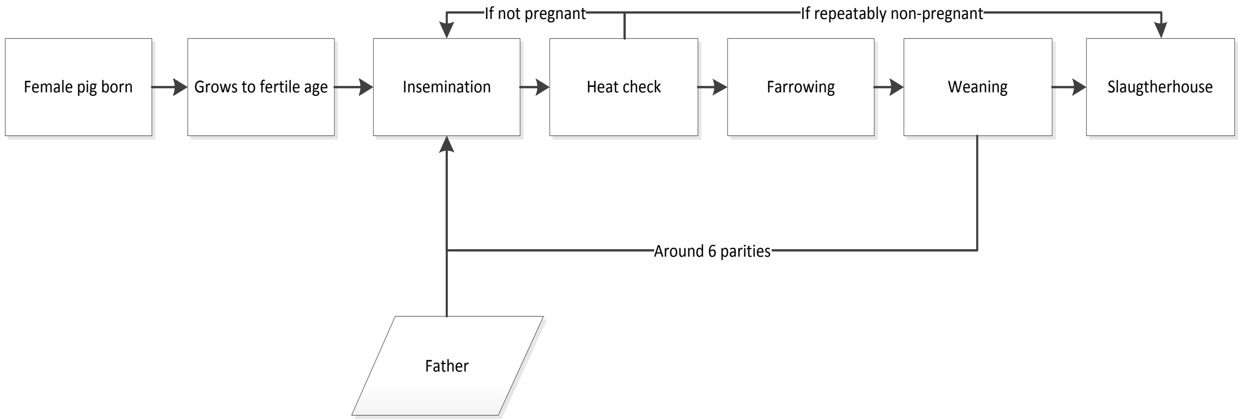


Figure 10. Life cycle of a sow

When a female pig is born and is expected to be a good breeding sow it is allowed to grow to its fertile age. Some farms are specialized in breeding good female pigs which they sell to other farms to use. When the sow is fertile it is inseminated with semen (either naturally or artificial) from a boar which typically is of another race as it yields the best meat. After a number of days the sow is checked to determine whether the insemination turned out to be a success. Then after about 116 days of pregnancy the sow farrows and gives birth to a number of piglets, typically between 10 and 15. The piglets stay with their mother until they age of 4 to 5 weeks where they are weaned from their mother. The piglets now go to the fattening stage.

Figure 10 has omitted the feeding and medication parts. The sow is fed according to a number of parameters:

1) Age and/or weight. If the farmer notices a sow that is thin he can increase the amount of feed given to the sow. Conversely, he can decrease the feed ration if the sow is too thick.
2) Day since insemination. As the embryos grow within the sow it needs more feed. Also it has been shown that decreasing the feed ration before farrowing gives a smoother birth.
3) Number of piglets. When the sow is producing milk for its piglets it needs extra feed if it has more piglets than usual.

As the sow gets older it starts to perform worse, that is, it gets fewer piglets. At some point the farmer chooses to introduce a new sow into production and send the old sow to the slaughterhouse.

Fatting of Pigs

Fattening of pigs is basically a matter of feeding the pigs optimally. However, as Figure 11 shows we can also have factors such as medicine and climate

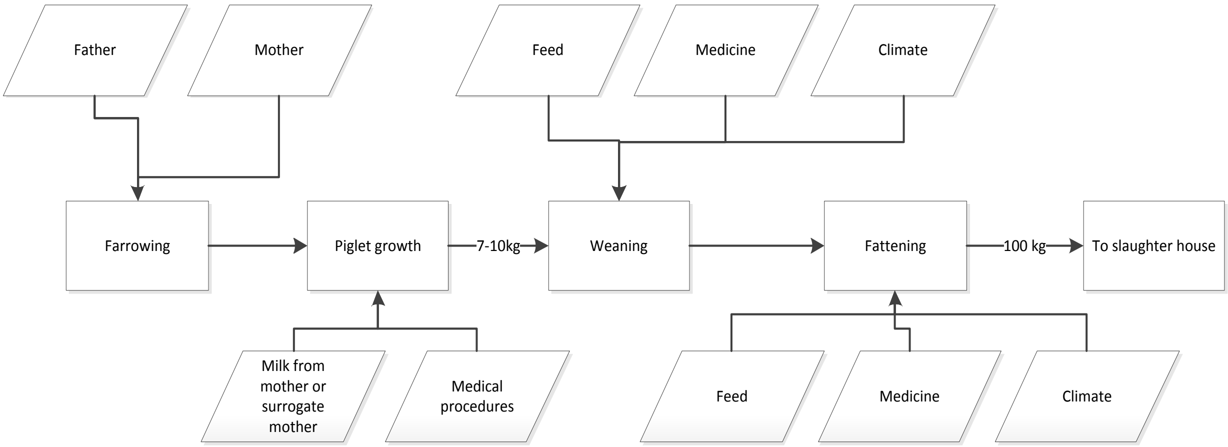


Figure 11. Elements of fattening a pig.

The piglet is born when the sow farrows. By drinking milk from its mother it grows to about 10 kg until it is weaned. After that it is fattened until it reaches a weight of 100 kg where it is send to the slaughterhouse.

The small piglets are sensitive to their climate (temperature, humidity etc.). A bad climate reduces growth, increases the risk of illness and, in the worst case scenario, death. The piglets also undergo a number of medical treatments. Foremostly, they are vaccinated against various diseases. The piglets are also treated with medicine if they catch some disease. Finally, male piglets are often castrated as their testicles produce a hormone which sometimes results in "boar taint" which is a very offensive odour.

When the pig reaches a weight of about 100 kg it is transported to the slaughterhouse.

Dairy

In order to produce milk the dairy farmer upholds a stock of cows. In order to get the cow to produce milk it must be bred and have a calf. A female calf may be raised to go into the dairy production while male calves are fattened in order to produce beef (often by another farmer). The calves are separated from their mother soon after birth and are given controlled amounts of milk/milk-replacement.

After giving birth to a calf the cow is in its lactation period (i.e. it produces milk) for approximately 300 days. It is important to milk the cow twice every day. Otherwise it ends its lactation period and needs to give birth again.

On a modern dairy farm the cows are milked by an automatic milking robot. Each cow is trained to go to the milking machine when its udder is filled with milk. The automatic milking robot is also 'instructed' in where the teats are placed in each of the cows found on the farm. That is, each cow can be recognized by the milking robot from the RFID in its ear tag.

Poultry

The poultry production comes in three categories:

- Egg producing farms that produce eggs for human consumption;
- Breeders that produce fertilized eggs and hatch them;

- Broilers that fattens the chickens produced by the breeders.

6.1.4 The Slaughterhouse

When the animals arrive at the slaughterhouse they are killed and cut into 'manageable', but still relatively large pieces. Animals are inspected and samples are taken in order to detect signs of diseases and infections.

The slaughterhouse may also record the quality (e.g., meat percentage) of the delivered animals which is used to compute the price of the animal. This information is reported back to the farmer.

6.1.5 Retail Store

The retail store further processes the meat into consumer sized packages which goes into the store. Here the consumer can pick and buy the meat.

6.2 Use cases in agricultural domain

This section contains a number of use cases for the entire chain. However, we have put emphasis on describing use cases on the farm. During the production a number of actors comes into play. The following list contains the most important actors which we will be using in the use cases.

- The farmer that manages the farm. The actor "farmer" also covers any employees that he might have. The farmer has a number of systems to assist his work
    - One or more management systems (field, pigs, cows/dairy etc)
    - Feeding systems
    - Climate systems
- The farm consultant that assists the farmer in making decisions.
- The feed production manager that operates the feed production plant.
- The slaughterhouse manager that operates the slaughterhouse
- The veterinarian that monitors animals at the farm and in the slaughterhouse.
- The retail management that operates the supermarkets.

6.2.1 Story for the field area

The farmer needs to decide which type of crop to produce on a particular field. To do this he needs to know the history of the field, that is, which types of crops have been produced there the previous years. He gets this information from his management system. He also needs to take into account the type of soil. Finally, the price of the seed and the expected revenue are taken into account.

In order to optimize the yield of the field the farmer puts fertilizers onto the field. He needs to choose which type of fertilizer and how much to use. To do this he needs information such as:

1) Type of soil (sandy, silt, clay);
2) Existing nutrients in soil which can be determined by an analysis in a laboratory. Multiple samples may be taken from the fields as nutrient content can vary;
3) The production type, that is, whether the field is farmed after organic principles or not. An organic field may only use certain types of fertilizers;
4) The price of the fertilizer;
5) Availability of manure (from his own or nearby farms) and its content of nutrients;
6) Environmental regulations.

Yet another way of optimizing yield is to use herbicides and pesticides. Similarly to the case of fertilizers, the farmer must take into account the price and expected gain to determine the best option.

Fertilizers, herbicides, and pesticides are sprayed/distributed with the use of a tractor. The most advanced setups are able to utilize GPS to adjust amounts sprayed depending on the position on the field as some areas need less fertilizer than others.

The crops on the field are soon reaching a state where it is to be harvested. The farmer needs to decide when exactly to do initiate the harvest. To do the planning he needs to take a number of things into account:

1) The weather needs to be suitable. That is, it should be dry such that crops are less humid and prone to rotting. The farmer uses weather forecasts to predict a good harvesting time.
2) The crops need to be as close to the optimal maturity level. The farmer uses manual inspection and experience to determine and estimate the maturity level.
3) The availability of harvesting equipment. The farmer may own his own equipment but otherwise (or as a complement) he may rent harvesting equipment to get the harvesting done (faster).
4) Availability of man power. The harvesting equipment needs to be operated and as many farmers wishes to harvest at the same time there is a high demand for man power during the harvesting season.

Figure 12 maps selected activities in the farm production area including managerial activities. Activity diagram corresponds/reacts with use cases 6.2.1.1 – 6.2.1.5 (see Appendix 2) as well as with general use cases described above.

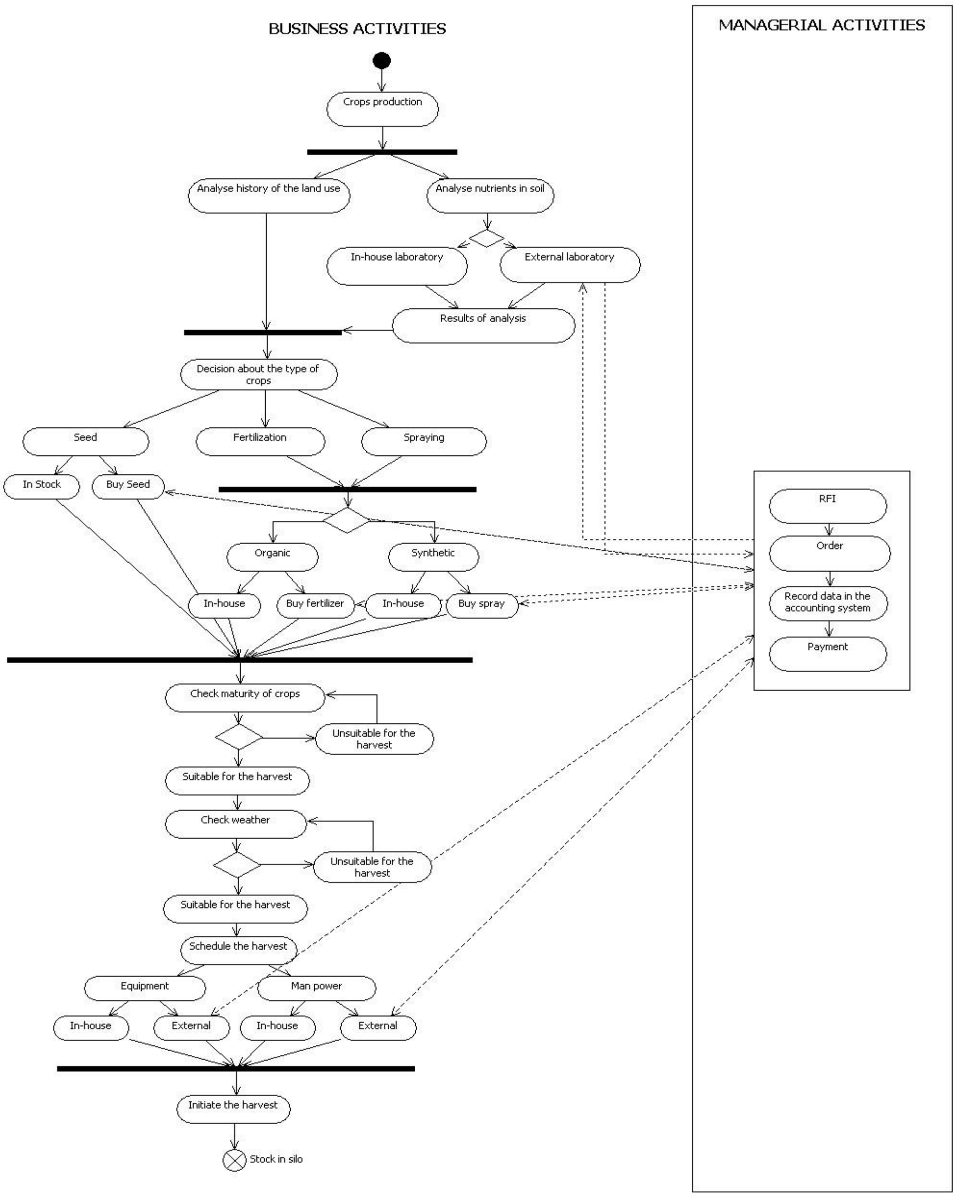


Figure 12. Activity Diagram for Crops production

6.2.2 Stories for the feed production

Extra production capacity

The feed production management system contains all orders for feed stuff. However, the manager can see that the production facility has left over capacity when the existing orders are combined. In this case he might consider producing extra of a standard feed stuff type, so that he is prepared to deliver feed at a later time and he can optimize his production flow.

Silo empty

The silo containing soybean meal is running empty. However, the feed factory is committed to delivering feed stuff with specific nutritional values and the usual feed mix uses soybean meal. The production manager chooses to produce an alternative feed mix using fish meal to provide the same nutritional value. He chooses fish meal as it is the cheapest alternative with similar characteristics.

6.2.3 Stories for the farm

Replacing a sow

The yield of each sow is monitored by recording the number of piglets it can produce and days where the sow is "unproductive". It is unproductive when it is neither pregnant nor feeding piglets. For example, if the sow is inseminated but does not get pregnant it gets 14 days of unproductivity as the pregnancy test (heat check) is performed after 14 days.

The farmer considers both the number of piglets at birth but more importantly the number and weight of weaned piglets, that is, the number of piglets that are able to survive their first month of life and seem fit to be fattened. The information needed is collected by the farmer and put into the management system. Finally, the management system provides an analysis of the sows. If this quality metric is to low (e.g. compared to other sows at the farm) the farmer might decide to replace the sow.

The farmer now has two options; to use an animal raised by himself or buy one raised by another farmer specialized in breeding and raising good mother animals. In any case he needs to put the price of each option up against the expected quality of a given animal.
The quality of an animal is currently estimated by a computation that takes a number of factors, such the performance of the family of the animal, see http://www.danbredint.dk/view.asp?ID=3900 into account.

Inseminating a sow

When a sow is fertile it needs to be inseminated. The management system assists the farmer in determining the sows that are fertile based on their age and last weaning.

The farmer needs to decide whether it should be with semen from a boar of his own or semen from a supplier. He can order different qualities from the supplier. Depending on the aim of his production he chooses the race of the boar. If he is producing animals for breeding he has constraints on the race, while animals for meat production are more likely to mixed race animals as they yield better results. The insemination itself is a manual process.

Illness in the herd

If the farmer detects illness within his herd he needs to treat the animals. Detection of illness is currently mostly done by visual inspection of the animal behaviour but analyses of e.g. drinking behaviour have been used to assist the farmer in detecting the illness at an earlier stage. Depending on the type of illness the farmer chooses to medicate the animals. This choice is often (or must be) assisted by the veterinarian.

Depending on the type of medication the animal may be put into a sales quarantine. That is, it may not be allowed to sell the animal for slaughtering within the 14 days after medication such that the medicine cannot be detected in the meat at the slaughterhouse. When the management system is told which animals are treated with medicine it is also able to prevent sales of animals with medicine still in the body. The slaughterhouse fines the farmer if they detect medicine in the meat.

Weaning sows

When the piglets have grown to about 7-10 kg the piglets are taken from their mother. The management system helps the farmer find the sows that should have piglets of the appropriate age. If the piglets vary greatly in size the farmer may choose to wean only the biggest piglets. The number of weaned piglets is reported to the management system.

When the piglets are put into a pen the feeding system is told the number and weight of the pigs. It needs the information in order to feed the appropriate amount of feed. The climate system may also be given this information as it can optimize the climacteric conditions.

Selling pigs to the slaughterhouse

When the fattened pigs reach the weight that the slaughterhouse prefers (around 100 kg) the farmer sends the animals to the slaughterhouse. The farmer estimates the number of pigs he wants to sell with the help of the management system. The management system can tell where pigs of the proper age are located but the farmer needs to manually sort the animals as they do not grow with the same rate.

The pigs are transported to the slaughterhouse and slaughtered. During the slaughtering the pigs are weighed and various parameters are such as thickness of fat tissue is measured. These parameters form the basis for the pay to the farmer. If the pig is too big or too small there is a price deduction. If the meat contains too much fat there is also a deduction in price. The data is gathered in a report which is given to the farmer along with the computed price for the animals. The farmer may use the report to make adjustments to his production.

Figure 13 maps selected activities in the farm production area including managerial activities. Activity diagram corresponds/reacts with use cases 6.2.3.1 – 6.2.3.8 (see Appendix 2) as well as with general use cases described above.

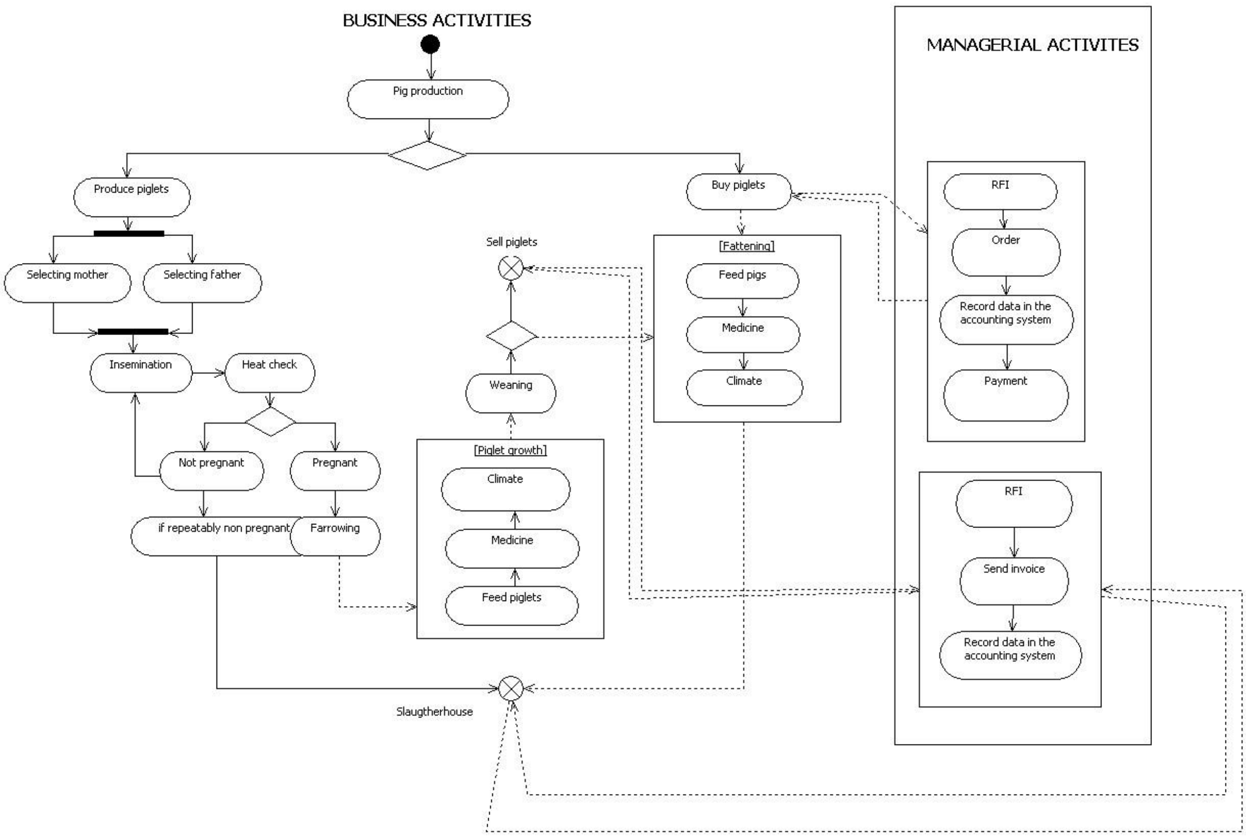


Figure 13. Activity Diagram for Pigs production

6.2.4 Stories for the slaughterhouse

Find the payment to the farmer

A transport with a number of pigs from multiple farmers arrives at the slaughterhouse. Each pig has been tattooed with an ID of the farm from which it comes. The tattoo is read visually by a human. Hereafter, it is tracked as an animal from that farm. If the tattoo is unreadable the slaughterhouse employee must use logic to determine which farm the animal comes from, that is, "I have counted 13 from that farm but they loaded 14 animals implying that this last animal must be from that farm". Because of this extra effort the farmer is paid less for the animal. An RFID system could help avoid such kind of situation and the relative economic damage.

The animal is now cut into halves by robots and the guts are removed. The carcass is weighed and the thickness of the fat below the skin is measured by ultrasound. The weight and fat thickness are two of the important parameters used to compute the price paid to the farmer for the pig. If the pig is small or larger than 80 kilograms the price per kilogram is lower as a standardized size is easier to handle later in the process.

Processing a new batch of animals

The manager needs to decide how the pigs awaiting slaughtering are to end up. The slaughterhouse management system tells what orders it has got and helps him choose the proper cuttings for the next batches of animals.

6.2.5 Stores for retail

The sales department of the supermarket chain is preparing the special deals for a coming week. They decide to have a special offer on pork loins and thus they need to determine the amount of meat to order from the slaughterhouse. To do so they look up sales of earlier similar special offers. Finally, the sales department arranges with the slaughterhouse to get the estimated amount of meat.

7. Conclusions

The deliverable presents use cases relevant to the ebbits project in the domains of manufacturing and agriculture. The use cases are modelled using standardized modelling language (UML).

Modelling the use cases of the two considered domains is the first step that will be further refined in the following deliverable, in order to provide detailed models of manufacturing and agricultural relevant processes. In the following deliverables other methodologies, like class diagrams, object diagram or more appropriate tools, to provide more details will be considered.

Thus the use cases considered are specific to two extremely different domains, through the further analysis of the use cases, the project team will be able to identify their commonalities and especially how to apply the ebbits outcomes to these different areas. This exercise will allow the project team to understand a basic mechanism that will enable the developed ebbits platform to be adapted to the different environments considered in the project, and thus this will also significantly help to the project team to extend the project results to different areas of actuation.

8. References

[1] Agile Modeling, UML2 Use Case Diagrams, available under http://www.agilemodeling.com/artifacts/useCaseDiagram.htm (state on 20th December 2010).

[2] Carnegie Mellon community, UML Use Case Diagrams: Tips and FAQ, available under http://www.andrew.cmu.edu/course/90-754/umlucdfaq.html (state on 20th December 2010).

[3] Cockburn A., Writing Effective Use Cases, Addison Wesley, 2000.

[4] Gottesdiener, E., Use Cases: Best Practices, A technical discussion on Use Case Best Practices, Rational.software, IBM,2006, available under http://www.eg.bucknell.edu/~cs475/F04-S05/useCases.pdf (state on 20th December 2010).

[5] Kennesaw State University, Activity Diagrams, available under http://atlas.kennesaw.edu/~dbraun/csis4650/A&D/UML_tutorial/activity.htm (state on 20th December 2010).

[6] Leffingwell, D., Widrig, D., Managing Software Requirements: A Use Case Approach, Second Edition, Addision Wesley, 2003.

[7] Rosenberg, D., Stephens, M., Use Case Driven Object Modeling with UML – Theory and Practice, Apress, 2007.

[8] Use Case Diagrams, available under http://www.uml-diagrams.org/use-case-diagrams.html (state on 20th December 2010)

[9] Watermark Learning, Scenarios and Use Cases – Useful Techniques, available under http://www.watermarklearning.com/blog/scenarios-and-use-cases/ (state on 20th December 2010).

[10] West Pole, Use Case and Interviewing Techniques for Focused Requirements Capture, available under http://www.westpole.com/pdf/use-cases.pdf (state on 20th December 2010).

[11] Wikipedia, Use case, available under http://en.wikipedia.org/wiki/Use_case (state on 20th December 2010).

[12] Wikipedia, Use case diagram, available under http://en.wikipedia.org/wiki/Use_case_diagram (state on 20th December 2010).

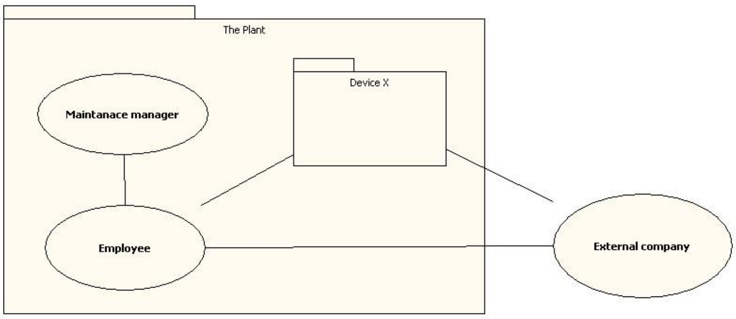
[13] Object Management Group under http://omg.org

9. Appendix 1 - Use case diagrams in Automotive Manufacturing Domain

5.2.1 Energy reduction management process

Use case 5.2.1.1 - Initial assessment of energy consumption

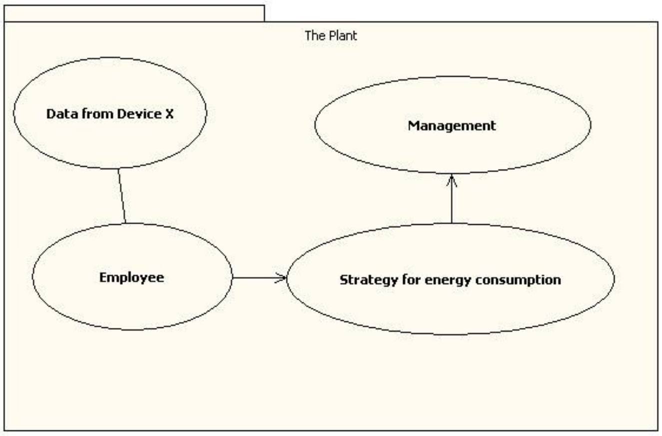
Use Case Name	5.2.1.1 Initial assessment of energy consumption
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	
Successful End Condition	Data on energy consumption collected
Actors	Device X, Employee, External firm
Use Case Initiation	Use Case starts on the maintenance manager's demand
Main flow	1. The maintenance manager sends requests for assessment of the energy consumption of a selected device 2. The employee initiates assessment or requests to an external company 3. The employee collects the data from the assessment
Post-Conditions	



Note:

Use case 5.2.1.2 - Develop a strategy for energy reduction

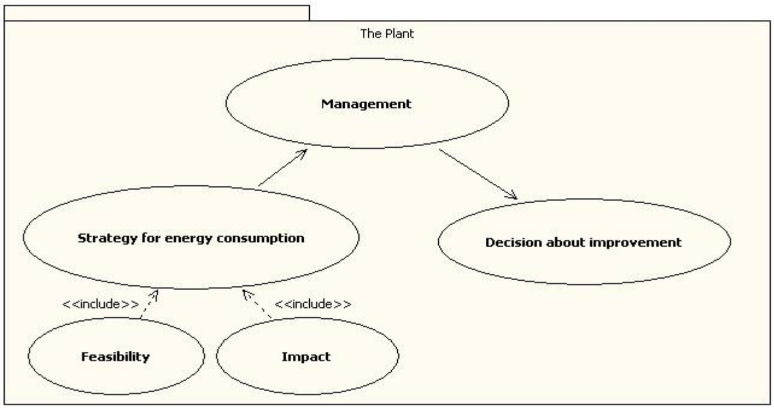
Use Case Name	5.2.1.2 Develop a strategy for energy reduction
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	Collected data of energy consumption
Successful End Condition	Strategy is developed and shared with company management
Actors	Employee, Management
Use Case Initiation	Use Case starts when the use case 5.2.1.1 is completed
Main flow	1. The employee analyses the collected data 2. The employee develops a strategy for energy reduction 3. The employee shares the strategy with management
Post-Conditions	



Note:

Use case 5.2.1.3 - Decision about improvement of the device

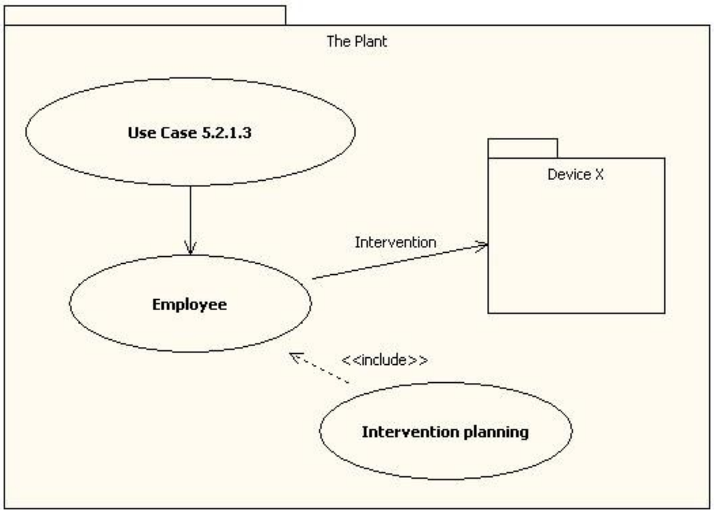
Use Case Name	5.2.1.3 Decision about improvement of the device
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	Collected data on the energy consumption, strategy of the energy consumption
Successful End Condition	Decision about improvement of the energy consumption
Actors	Employee, Management
Use Case Initiation	Use Case starts when the use case 5.2.1.2 is completed
Main flow	1. Management analyses plan and evaluate the feasibility and impact of the strategy 2. Management makes decision to implement the improvement of the energy consumption
Post-Conditions	



Note:

Use case 5.2.1.4 - Intervention on the device

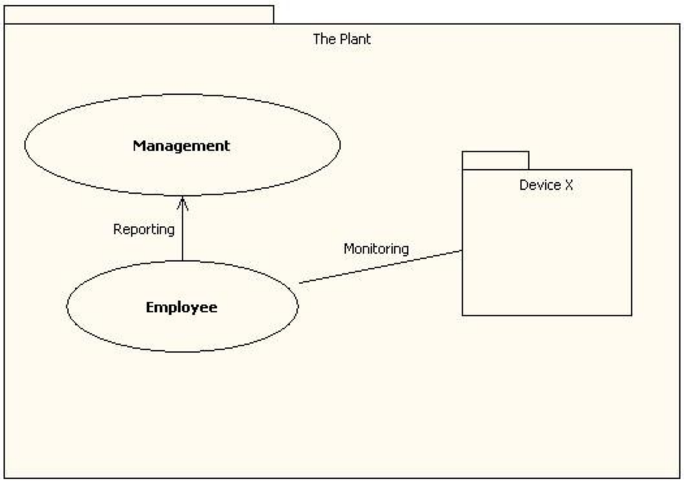
Use Case Name	5.2.1.4 Intervention on the device
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	Improvement of the energy consumption is accepted
Successful End Condition	Intervention on the device X is performed
Actors	Employee, Management, Device X
Use Case Initiation	Use Case starts when the use case 5.2.1.3 is completed
Main flow	1. The employee plans an intervention on the Device X 2. The employee executes intervention
Post-Conditions	



Note:

Use case 5.2.1.5 - Monitoring of the performed modification

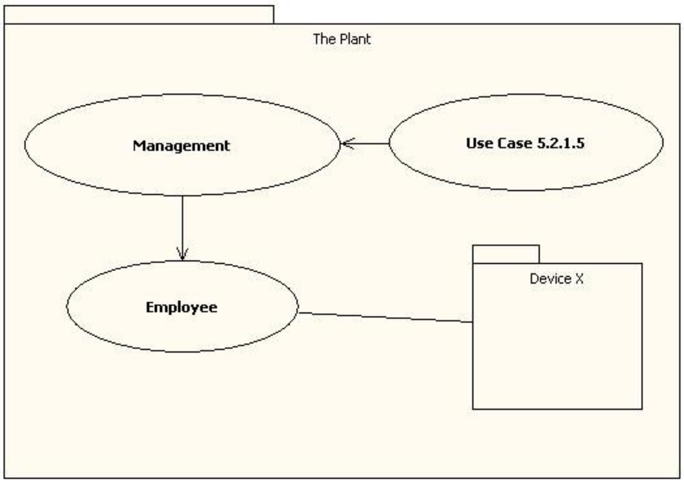
Use Case Name	5.2.1.5 Monitoring of the performed modification
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	Performed intervention on the Device X
Successful End Condition	Data collected
Actors	Employee, Management, Device X
Use Case Initiation	Use Case starts when the use cases 5.2.1.1 - 5.2.1.4 are completed
Main flow	1. The employee periodically monitors the performed modification on the Device X 2. The employee collects data on the consumption 3. The employee reports data to the management
Post-Conditions	



Note:

Use case 5.2.1.6 - Decision about additional improvement

Use Case Name	5.2.1.6 Decision about additional improvement
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	Monitoring of the performed modification on the Device X
Successful End Condition	Intervention on the device X is or is not performed
Actors	Employee, Management, Device X
Use Case Initiation	Use Case starts when the use case 5.2.1.5 is completed
Main flow	1. The Management assets data of energy consumption (of the modified Device X) 2. The Management decides to perform the improvement or continue with the current modification 3. The Management sends decision to the employee
Post-Conditions	

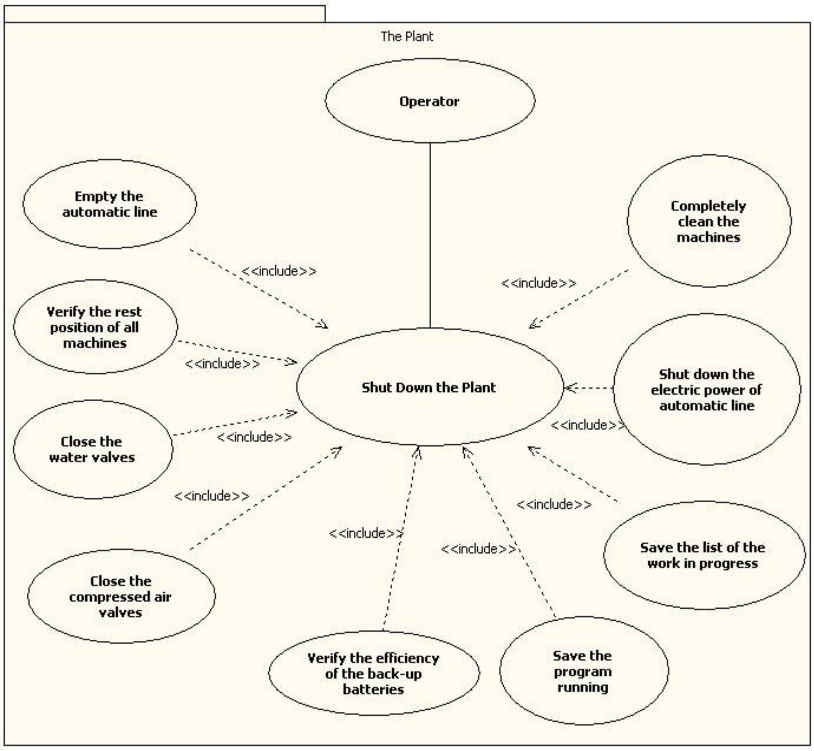


Note:

5.2.2 Plant production shut down and restart

Use case 5.2.2.1 - Shut down of the plant production

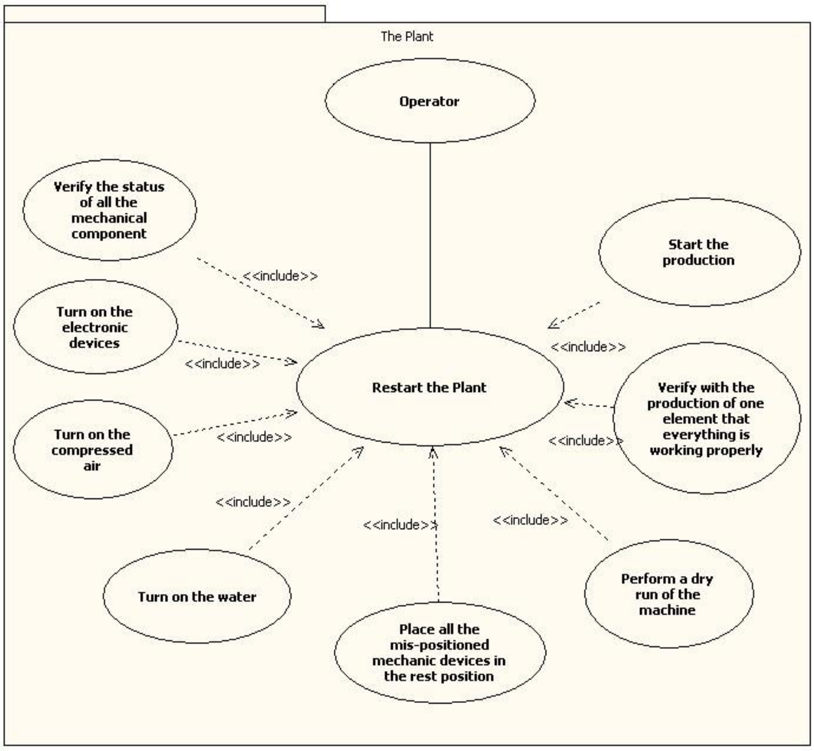
Use Case Name	5.2.2.1 Shut down of the plant production
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	Plant is running
Successful End Condition	Shut down the Plant
Actors	Operator
Use Case Initiation	Use cases starts when management makes decision that plant needs to be shut down
Main flow	1. The operator performs the following activities including: • Empty the automatic line from the part • Verify that all the machines are in the rest position • Close the water valves • Close the compressed air valves and verify if there is some mechanical movement that is changing its position • Verify the efficiency of the back-up batteries • Save the list of the work in progress (typology of part to produce) and the related quality record • Save the program running and the status of the variables for each electronic device used in the automatic line • Shut down the electric power of automatic line • Completely clean the machines
Post-Conditions	


The Plant
Operator
Empty the automatic line
Completely clean the machines
Verify the rest position of all machines
Shut Down the Plant
Shut down the electric power of automatic line
Close the water valves
Close the compressed air valves
Save the list of the work in progress
Verify the efficiency of the back-up batteries
Save the program running
<<include>>

Note:

Use case 5.2.2.2 - Restart of the plant production

Use Case Name	5.2.2.2 Restart of the plant production
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	Plant is not running
Successful End Condition	Restart of the Plant
Actors	Operator
Use Case Initiation	Use cases starts when management makes decision that plant needs to be restarted
Main flow	The operator performs activities including: • Verify the status of all the mechanical component • Turn on the electronic devices • Turn on the compressed air • Place all the mis-positioned mechanic devices in the rest position • Turn on the water • Perform a dry run of the machine • Verify with the production of one element that everything is working properly • Start the production
Post-Conditions	

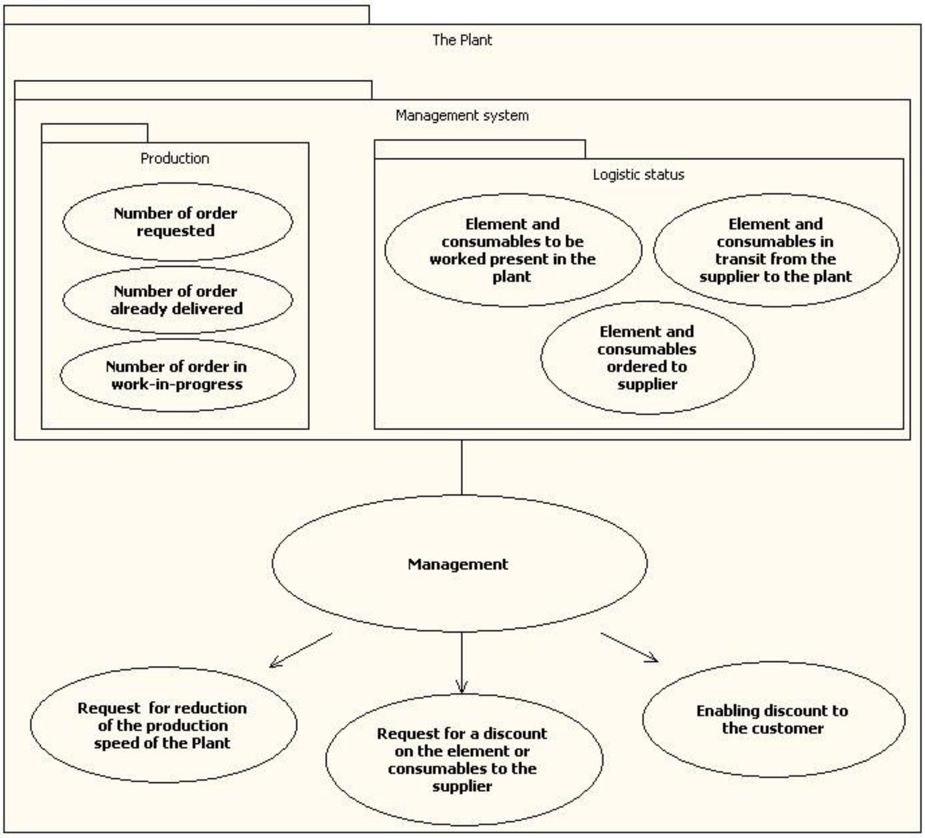


Note:

5.2.3 Automatic energy reduction

Use case 5.2.3.1 - Decision about the energy reduction

Use Case Name	5.2.3.1 Decision about the energy reduction
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	Management system at the Plant
Successful End Condition	Speed optimisation of the production line in the Plant
Actors	Management, Management system
Use Case Initiation	Use Case is carried out on the running basis when the management requires production optimisation
Main flow	1. The Management receives data about the production from management system including: • The number of orders requested • The number of orders already delivered • The number of orders in work-in-progress 2. The Management receives data about the logistic status from the management system including: • Element and consumables to be worked present in the plant • Element and consumables in transit from the supplier to the plant • Element and consumables ordered to supplier 3. The Management analyses data and makes decision including: • Request for reduction of the production speed of the Plant • Request for a discount on the element or consumables to the supplier • Enabling discount to the customer
Post-Conditions	

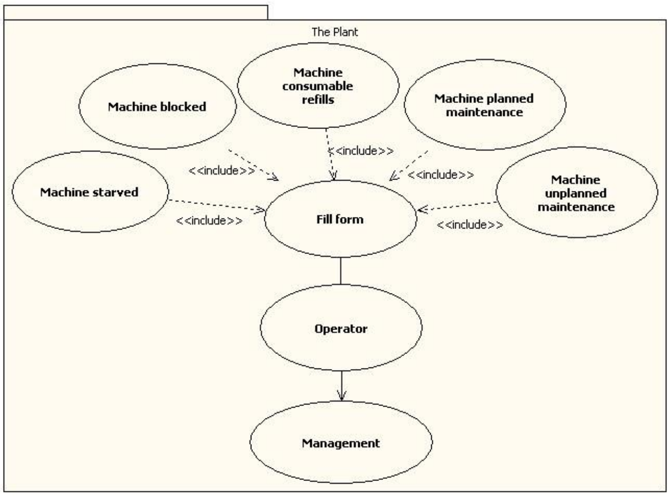

The Plant
Management system
Production
Number of order requested
Number of order already delivered
Number of order in work-in-progress
Logistic status
Element and consumables to be worked present in the plant
Element and consumables in transit from the supplier to the plant
Element and consumables ordered to supplier
Management
Request for reduction of the production speed of the Plant
Request for a discount on the element or consumables to the supplier
Enabling discount to the customer

Note:

5.2.4 Retrieve production information

Use case 5.2.4.1 - Filling in a form about an accident

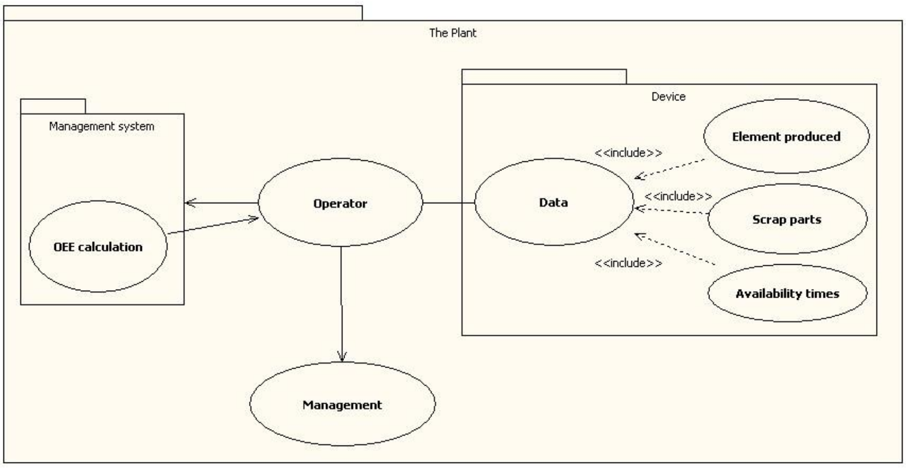
Use Case Name	5.2.4.1 Filling in a form about an accident
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	
Successful End Condition	Production accident information
Actors	Operator
Use Case Initiation	Use case starts immediately after the machine stops due to an accident
Main flow	1. The operator fills in a form with information related to the machine stop due to faults including: • Machine starved (missing part on the input) • Machine blocked (the output buffer is full) • Machine consumable refills (tips, bolts, etc) • Machine planned maintenance activities • Machine unplanned maintenance activities 2. The operator informs management
Post-Conditions	



Note:

Use case 5.2.4.2 - Collect the production data

Use Case Name	5.2.4.2 Collect the production data
Version	1.0
Author	TUK
Last update	28/1/2011
Assumptions	
Pre-conditions	
Successful End Condition	Information on the production accident
Actors	Operator
Use Case Initiation	Use Case is carried out on the daily basis
Main flow	1. The operator collects daily information on the machine status from the machine operators and the production numbers: • element produced • scrap parts • availability times 2. The operator stores data to the management system 3. The management system performs OEE calculation 4. The operator presents results to management
Post-Conditions	



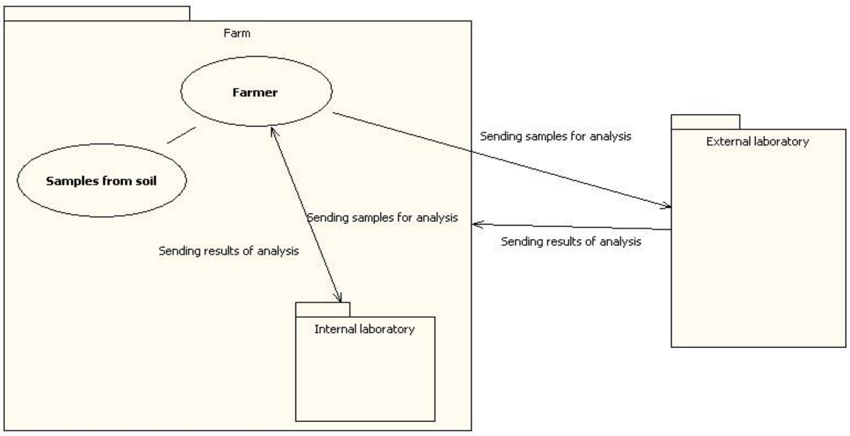
Note:

10. Appendix 2 - Use case diagrams in Agricultural Domain

6.2.1 Field area

Use case 6.2.1.1 - Analyse nutrients in soil (optional)

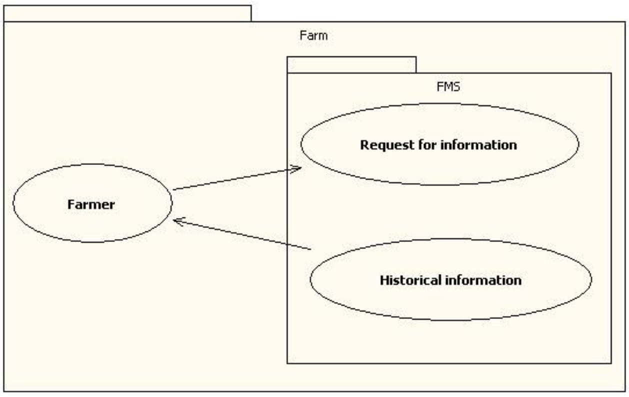
Use Case Name	6.2.1.1 Analyse nutrients in soil (optional)
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	
Successful End Condition	Information on nutrients in the soil
Actors	Person responsible for decision making (Farmer), Laboratory (external or in-house)
Use Case Initiation	Use Case starts before the sowing season (continuously during the whole year)
Main flow	1. Farmer collects samples of the soil 2. Farmer sends samples to the laboratory (or perform analysis in his laboratory) 3. Laboratory performs analysis 4. Laboratory sends result of the analysis to the farmer
Post-Conditions	



Note:
In the case that farmer uses an external firm all general use cases are performed at managerial level.

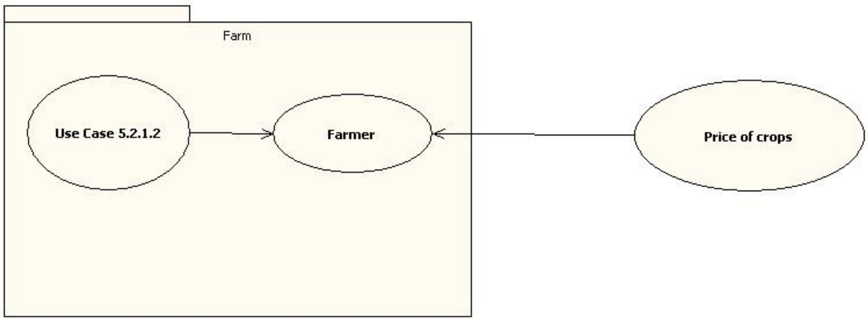
Use case 6.2.1.2 - Analysis of the history of the land use

Use Case Name	6.2.1.2 Analysis of the history of the land use
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Farm´s management system (FMS)
Successful End Condition	Received information of history of the land use
Actors	Person responsible for decision making (Farmer), Farm´s management system (FMS)
Use Case Initiation	Use Case starts before the sowing season (continuously during the whole year)
Main flow	1. Farmer logs in the FMS 2. Farmer obtains data about the history of the specific field (which types of crops have been produced there in the previous years) 3. Farmer logs out from the FMS
Post-Conditions	

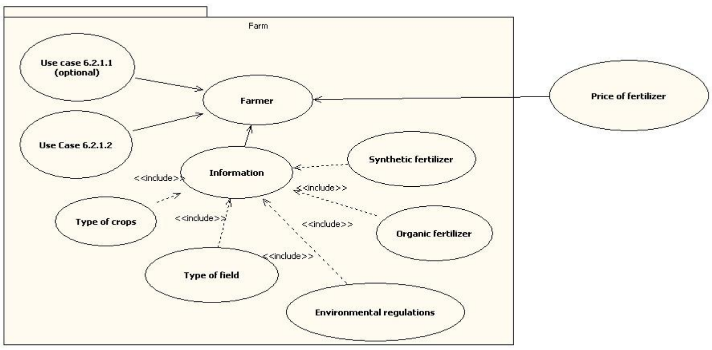


Note:

Use case 6.2.1.3 - Decision about the type of crops

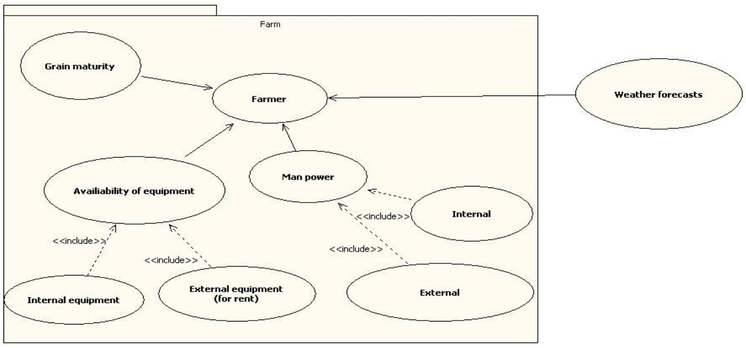
Use Case Name	6.2.1.3 Decision about the type of crops
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Farm´s management system (FMS), Completed use case 6.2.1.1 (optional) and use case 6.2.1.2
Successful End Condition	Decision about type of crops
Actors	Person responsible for decision making (Farmer), Farm´s management system (FMS), Suppliers
Use Case Initiation	Use Case starts before sowing season, when the use case 6.2.1.2 is completed
Main flow	1. Farmer obtains all information from the FMS (Use case 5.2.1.2 2. Farmer collects all information about the price of crops 3. Farmer decides which type of the crops is suitable for the selected type of field
Post-Conditions	Responsible person initiates Use case RFI and other managerial use cases
	
Note:	

Use case 6.2.1.4 - Decision about fertilization

Use Case Name	5.2.1.4 Decision about fertilization
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	
Successful End Condition	Decision about type of fertilizer
Actors	Person responsible for decision making (Farmer), Farm´s management system (FMS), Suppliers
Use Case Initiation	Use Case starts before sowing season, when the use case 6.2.1.3 is completed
Main flow	1. Farmer obtains information from the FMS (use case 6.2.1.1, use case 6.2.1.2) 2. Farmer selects a type of the fertilizer including information on: • type of crops, • type of field • organic or inorganic (synthetic) focusing in production • price of fertilizer
Post-Conditions	After the decision about selecting proper fertilizer Farmer has to check availability of the manure and its content of nutrients Responsible person initiates Use case RFI and other managerial use cases
	
Note:	

Use case 6.2.1.5 - Decision about harvesting time

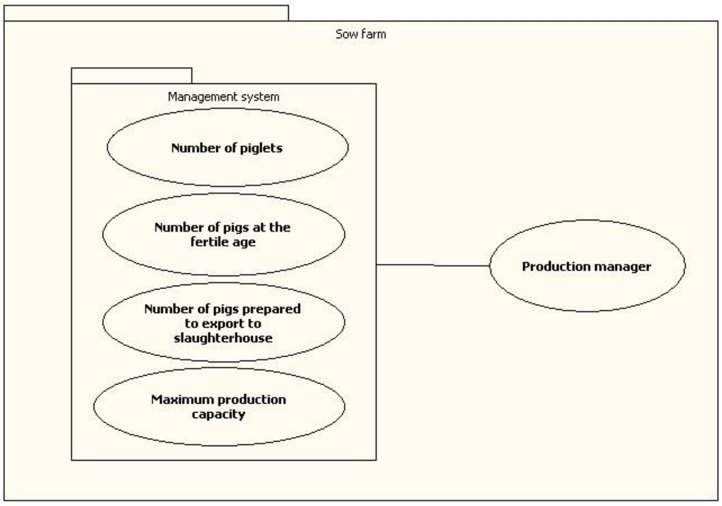
Use Case Name	6.2.1.5 Decision about harvesting time
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	
Successful End Condition	Decision about the start of harvesting
Actors	Person responsible for decision making (Farmer), Farm´s management system (FMS), Suppliers
Use Case Initiation	Use Case starts in the harvesting season
Main flow	1. Farmer manually checks grain maturity 2. Farmer obtains information necessary for harvesting including: • weather information (external forecasts) • availability of harvesting equipment • availability of man power
Post-Conditions	Responsible person initiates Use case RFI Responsible person initiates other managerial use cases (for external equipment and man power)



Note:

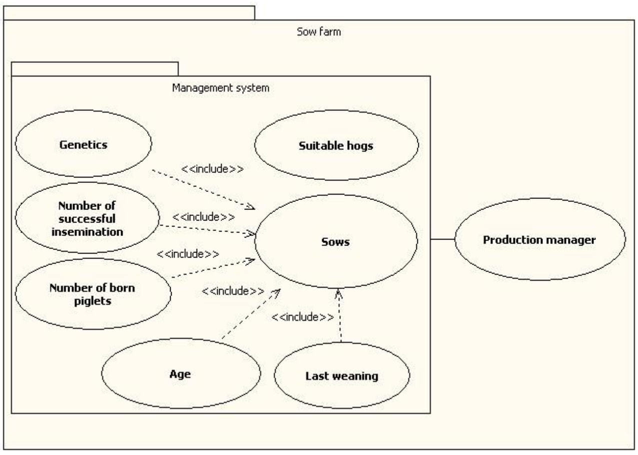
6.2.3 Farm production

Use case 6.2.3.1 - Check conditions for production

Use Case Name	6.2.3.1 Check conditions for production
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Sow farm management system (SFMS) Responsible person (production manager) has access to SFMS
Successful End Condition	Received information about the conditions
Actors	Sow farm management system (SFMS), Responsible person (production manager)
Use Case Initiation	Use Case is on the running basis
Main flow	1. The Production manager logs into the SFMS 2. The Production manager receives information on the production including: • the number of the piglets • the number of pigs at the fertile age • the number of pigs prepared to export to slaughterhouse • maximum capacity for production 3. The Production manager logs out from the SFMS
Post-Conditions	
	
Note:	

Use case 6.2.3.2 - Decision about insemination

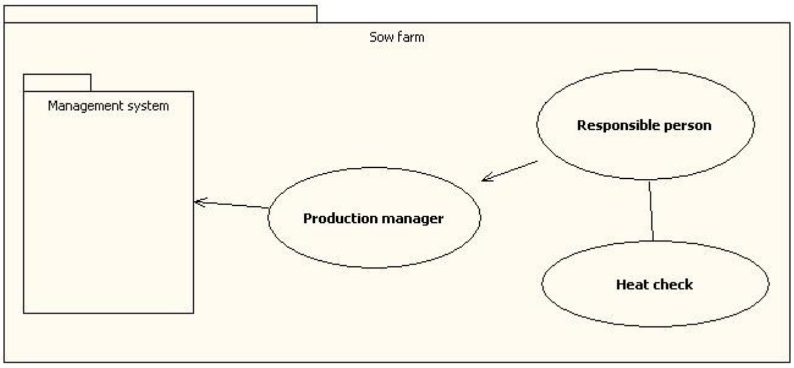
Use Case Name	6.2.3.2 Decision about insemination
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Sow farm management system (SFMS) Responsible person (production manager) has access to SFMS
Successful End Condition	Decision about insemination
Actors	Sow farm management system (SFMS), Responsible person (production manager)
Use Case Initiation	Use Case is carried out on the running basis, when the use case 6.2.3.1 is completed
Main flow	1. The production manager logs into the SFMS 2. The production manager selects sows suitable for inseminating including historic information • Genetics of the sow • Age of the sow • Last weaning • The number of successful insemination • The number of born piglets 3. The production manager selects hogs suitable for the process of insemination 4. The production manager logs out from the SFMS
Post-Conditions	Physical insemination is performed



Note:

Use case 6.2.3.3 - Check heat

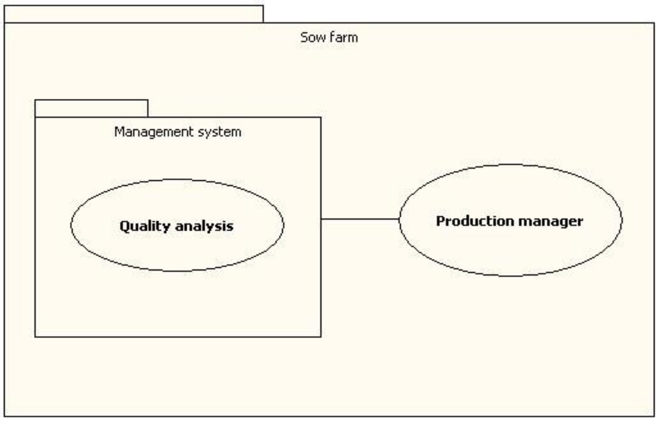
Use Case Name	6.2.3.3 Check heat
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Completed use case 6.2.3.2 Physical insemination, Sow farm management system (SFMF)
Successful End Condition	Updating data in the Sow farm management system (SFMF)
Actors	Sow farm management system (SFMS), Responsible person for heat check, Production manager
Use Case Initiation	Use Case starts usually 14 days after insemination
Main flow	1. Responsible person physically checks heat of the inseminated sow 2. Production manager logs into the SFMS 3. Production manager updates information about sows 4. Production manager logs out the SFMS
Post-Conditions	If the insemination is not successful, production manager will repeat use case 6.2.3.2



Note:

Use case 6.2.3.4 - Replace a sow

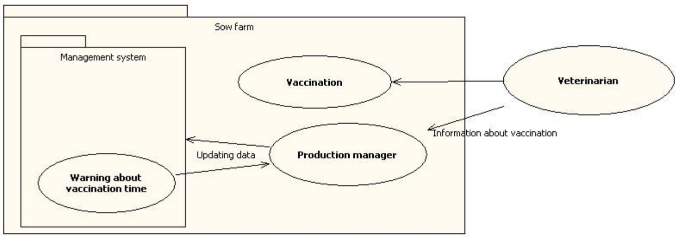
Use Case Name	6.2.3.4 Replace a sow
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Sow farm management system (SFMS)
Successful End Condition	Decision about replacing a sow
Actors	Sow farm management system (SFMS), Production manager
Use Case Initiation	Use Case starts when management system indicates that, according to the metric used, the quality is too low
Main flow	1. The production manager logs into the SFMS 2. The management system performs analysis 3. The production manager makes decision about replacing a sow (when the quality is too low – e.g. compared to the other sows at the farm) 4. The production manager logs out the SFMS
Post-Conditions	



Note:
The manager should replace the selected sow by an animal raised at the farm or buy one raised by another farmer specialized in breeding and raising good quality mother animals.

Use case 6.2.3.5 - Vaccinate

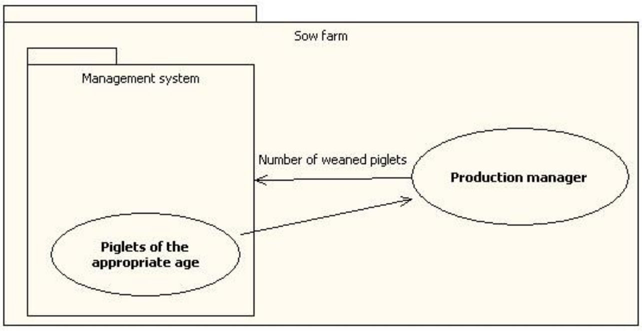
Use Case Name	6.2.3.5 Vaccinate
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Sow farm management system (SFMS)
Successful End Condition	Updated information in the SFMS
Actors	Sow farm management system (SFMS), Production manager, Responsible person/veterinarian
Use Case Initiation	Use Case starts when piglets reach the defined age
Main flow	1. The production manager logs into the SFMS 2. The production manager determines time of vaccination 3. The production manager contacts a responsible person/veterinarian 4. Responsible person/veterinarian physically performs vaccination 5. The production manager updates information in the SFMS 6. The production manager logs out the SFMS
Post-Conditions	



Note:

Use case 6.2.3.6 - Wean piglets

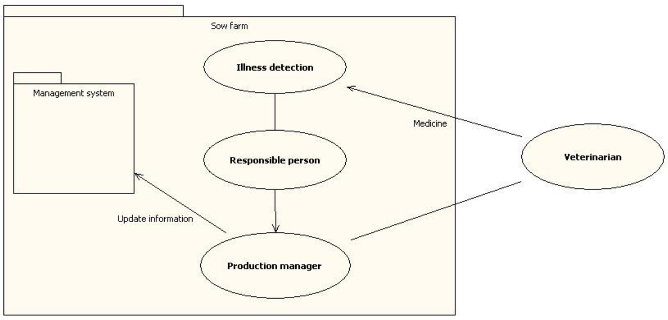
Use Case Name	6.2.3.6 Wean sows
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Sow farm management system (SFMS)
Successful End Condition	Updated information in the SFMS
Actors	Sow farm management system (SFMS), Production manager
Use Case Initiation	Use Case starts when piglets reach the proper age/weight
Main flow	1. The production manager logs into the SFMS 2. The production manager selects sows that should have piglets of the appropriate age 3. The selected piglets are weaned 4. The production manager updates information about the number of weaned piglets in the SFMS 5. The production manager logs out the SFMS
Post-Conditions	



Note:

Use case 6.2.3.7 - Medical intervention

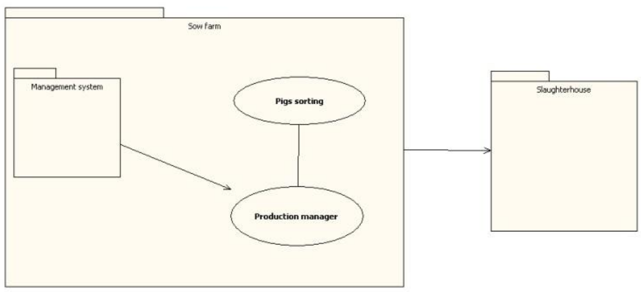
Use Case Name	6.2.3.7 Medical intervention
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Sow farm management system (SFMS)
Successful End Condition	Updated information in the SFMS
Actors	Sow farm management system (SFMS), Production manager, Responsible person/veterinarian
Use Case Initiation	Use Case starts when the responsible person detects illness
Main flow	1. The responsible person continuously checks animals 2. Responsible person detects illness and contacts veterinarian 3. Veterinarian checks situation and decides about the treatment 4. Production manager updates information about the medicine used in the SFMS
Post-Conditions	Management system prevents sale of animals with a medicine still in the body



Note:

Use case 6.2.3.8 - Selling pigs to slaughterhouse

Use Case Name	6.2.3.8 Selling pigs to slaughterhouse
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Sow farm management system (SFMS)
Successful End Condition	Selected pigs are delivered to slaughterhouse
Actors	Sow farm management system (SFMS), Production manager,
Use Case Initiation	Use Case starts when pigs reach the proper age/weight
Main flow	1. The production manager logs into the SFMS 2. The production manager estimates the number of pigs he wants to sell 3. The management system provides information about the location of pigs with the proper age 4. Responsible person manually sorts animals 5. The production manager sends selected animals to slaughterhouse 6. The production manager updates information in the SFMS 7. The production manager logs out the SFMS
Post-Conditions	

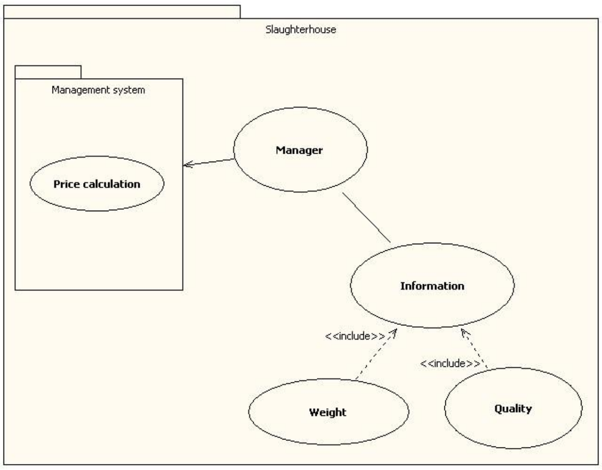


Note:

6.2.4 Slaughterhouse

Use case 6.2.4.1 - Specification of payment to the farm

Use Case Name	6.2.4.1 Specification of payment to the farm
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Slaughterhouse management system (SMS)
Successful End Condition	Determine the amount of money to the farm
Actors	Slaughterhouse management system (SMS), Person responsible for pricing (Slaughterhouse manager)
Use Case Initiation	Use Case starts pigs are delivered to slaughterhouse
Main flow	1. Pigs are counted and slaughtered 2. The animals are cut into halves (guts are removed) 3. Halves are weighted and thickness of the fat is measured 4. Slaughterhouse manager update information in the management system and calculate price
Post-Conditions	Slaughterhouse sends payment to the farm

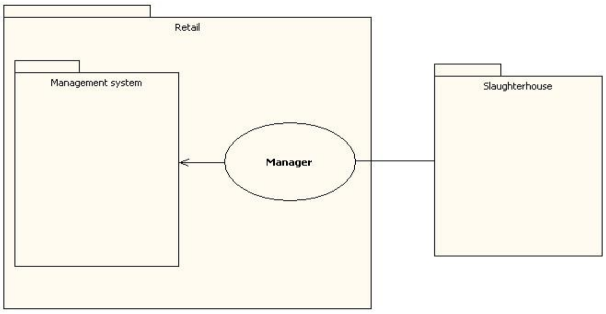


Note:

6.2.5 Retail

Use case 6.2.5.1 - Order specification

Use Case Name	6.2.5.1 Order specification
Version	1.0
Author	TUK
Last update	10/1/2011
Assumptions	
Pre-conditions	Retail management system (RMS), Manager has access to RMS
Successful End Condition	Determine the amount of meat from the slaughterhouse
Actors	Retail management system (RMS), Person responsible for ordering (Retail manager)
Use Case Initiation	Use Case is on the running basis
Main flow	1. Retail manager logs into the RMS 2. Information from the Management system helps to determine the type and amount of meat 3. Retail manager prepares the order to slaughterhouse 4. Retail manager logs out form the RMS
Post-Conditions	



Note: