NTNU
Norwegian University of Science and Technology
Department of Marine Technology

MASTER THESIS NAUTICAL SCIENCE

Title: Application of an Oil Spill Response Model in Ice-bridge Simulator Training	**Delivered:** 11th December 2011
	Availability: Accessible
Student: Ulrich Alain Kounchou Tagne	**Number of pages:** 93

Abstract:

The oil spill emergency response and recovery in the Arctic Ocean has been since a major area of research. The use of previously practiced methods such as oil spill response options – with the implementation of mechanical and non-mechanical recovery systems; the tracking and monitoring systems; and the spill trajectory and weathering systems, has been proven very effective. However these technologies cannot guaranty success when spill emergency occur. A new approach must be considered to increase the level of certainty.

The overall aim of this thesis was the development of an application that can be used for oil spill response training in ice-bridge simulators. This application developed a new approach and focused on real-time objects information sharing and emergency personnel skills.

Actor's goal and behavior was modeled and a comprehensive use case and domain model was proposed following a unified modeling process.

Spatial data objects were used with GIS integration and database system support.

The application was implemented using distributed databases to share spatial data and information between the Trainer and the Trainees during the courses. Several training and test courses scenarios were simulated.

The scenarios form the basis for new training courses on the planned ice-bridge navigation simulator facility at the University of Tromsø.

Key words:

Ice-bridge Simulator Training; Oil Spill Response; Master; Slave; Data Replication; Arctic Ice-covered Water; GIS; Oil Mapping; Networking; Domain Modeling; UML; Spatial Data; Information Sharing; MySQL SAR Radar; Fuzzy Logic System; Database; Trainer; Trainees.

Professor Dr. Egil Pedersen

Address:
NTNU
Department of Marine Technology
N-7491 Trondheim

Location
Marinteknisk Senter
O. Nielsens vei 10

Tel. +47 73 595501
Fax +47 73 595697

NTNU

Master Thesis in Nautical Science

Spring 2011

For

MSc Student Ulrich Alain Kounchou Tagne

Application of an Oil Spill Response Model in Ice-bridge Simulator Training

Ulrich Alain kounchou Tagne

Trondheim, 11.06.2011

NTNU Trondheim
Norwegian University of Science and Technology
Department of Marine Technology

MSc THESIS IN NAUTICAL SCIENCE

SPRING 2011

FOR

STUD. TECHN. Ulrich Tagne

Application of an Oil Spill Response Model in Ice-bridge Simulator Training

Global rise of energy demand and security are likely to intensify oil and gas E&P in the Arctic regions in the years to come. However, the operational complexity in the Arctic environment makes it challenging to operate safely without endangering the stability of the eco-system. Effective response to a blowout spill is required by the authorities and E&P companies. This is a task that is suitable for simulator-based training in suitable facilities.

This thesis shall investigate application of an oil spill response model in an ice-bridge simulator as proposed by the candidate in the project thesis. The aim is to provide knowledge that can form the basis for new training courses on the planned ice-bridge navigation simulator facility at the University of Tromsø with personnel that is to be involved in the response and recovery phases of an oil spill incident in Arctic waters. The work shall include, but is not limited to the following:

- Developing a relational database model for oil spill response in an ice-bridge navigation simulator.
- Filling the database schema with relevant data while keeping the data integrity
- Building a master, slave and client database network communication for simulation of different oil spill scenarios with emphasis on:
    - Managing the overall oil spill response operations while different training users are facing different oil spill scenarios.
    - Testing and evaluating the validity of the application from the viewpoint of users of training courses.
- Propose relevant simulator-based training courses and evaluation procedures with emphasis on efficient response to an oil spill incident in Arctic waters.

NTNU Trondheim
Norwegian University of Science and Technology
Department of Marine Technology

In the thesis the candidate shall present his personal contribution to the resolution of problem within the scope of the thesis work.

Theories and conclusions should be based on mathematical derivations and/or logic reasoning identifying the various steps in the deduction.

The candidate should utilize the existing possibilities for obtaining relevant literature.

The thesis should be organized in a rational manner to give a clear exposition of results, assessments, and conclusions. The text should be brief and to the point, with a clear language. Telegraphic language should be avoided.

The thesis shall contain the following elements: A text defining the scope, preface, list of contents, summary, main body of thesis, conclusions with recommendations for further work, list of symbols and acronyms, reference and (optional) appendices. All figures, tables and equations shall be numerated.

The supervisor may require that the candidate, in an early stage of the work, present a written plan for the completion of the work. The plan should include a budget for the use of computer and laboratory resources that will be charged to the department. Overruns shall be reported to the supervisor.

The original contribution of the candidate and material taken from other sources shall be clearly defined. Work from other sources shall be properly referenced using an acknowledged referencing system.

The project shall be submitted in two copies:

- Signed by the candidate.
- The text defining the scope included.
- In bound volumes.
- Drawings and/or computer prints that cannot be bound should be organized in a separate folder.
- As pdf file.

Supervisor : Professor Egil Pedersen, Department of Marine Technology, NTNU
Co-advisor : Dr. Kensuke Kirimoto, Department of Marine Technology, NTNU
Start : 17th January, 2011
Deadline : 11th June, 2011

Trondheim, 17th January, 2011

Egil Pedersen

Intentionally left blank

Contents at a Glance

Preface

This Master thesis (TMR4925) is written as part of a mandatory academic work in final year graduation for International Master Study Program in Nautical Science at the Norwegian University of Science and Technology, Department of Marine Technology.

The Arctic Ocean region has large quantities of oil and the production it is expected to be start in the near future. Due to the remoteness of the Arctic - low accessibility to human and its fragile natural resources (fauna and flora), any large-scale oil spill event can be disastrous.

Extensive research works related to oil spill response in Arctic Waters have been conducted and still ongoing or on to-do list with respect to detecting and monitoring [3], and applying mechanical or non-mechanical recovery technologies, [19]. Some of the recovery methodologies were studied and elaborated in the preceding project work: "Deepwater Blowout: Operational Aspects of Oil Spill Response in Arctic Ice-covered Waters", [13].

However, there is an urgent need to consider new approach in responding to serious oil spill events: the correlation between these technologies (tracking, monitoring, and recovery) and the on-scene emergency personnel must be kept as strong as possible during the whole spill response. Based on the literature review, it tends to be that this correlation was clearly neglected. Furthermore, it is unfortunate that while the benefits of integrating an oil spill response module in an Ice-bridge simulator training (IBST) are remarkable, they are yet non-existent today. At first, on-scene emergency personnel need to have immediate and real time information access to critical objects - objects they are using or working with, oil, ice, and the surrounding objects. Secondly, these information need to be shared in real-time with onshore base operators. By following this thesis the reader will get understand on the importance of these issues and how to apply them.

Acknowledgements

I would like to take this opportunity to express my sincere gratitude and thanks to my Supervisor Dr. Professor Egil Pedersen and my Co-supervisor Dr. Kensuke Kirimoto at the Norwegian University of Science and Technology, Department of Marine Technology, for their continuous guidance, advices and kindness throughout my master study and the entire process of my thesis work. Many thanks to Boye Høverstad – Research scientist at Sintef Marine Environmental Technology, for his comment and advices related to my thesis. My special thanks to Nick Hughes - Leader of the Ice Service Forecasting Division for Northern Norway. He has provided me with lots of ice data as well as technical advices, which I have used as a solid basis in this thesis.

Executive Summary

Oil spill emergency response and recovery in the Arctic Ocean has been a major and busy area of research. Several spill trajectory systems such as OSCAR, [21], GNOME, [22] and OILMAP, [23] have been proven effective. These systems provide support to various operational methods - both mechanical and non mechanical recovery techniques. The level of importance and relevance in the development of new systems or improving those already existing is still growing strong; especially with oil production in the Arctic ice-covered waters scheduled to get start in the near future.

However, using these technologies may not always bring success unless they are correlated with personnel skills and objects information system. Clearly, the on-scene spill response personnel must be skilled and be in position to have real-time access on all information stored in relevant objects such as oil, ice, water, equipment, sensitive and non-sensitive segment areas, sacrificial zones, and other possible neighbouring objects including shoreline, vessel transit line, pipeline, etc. Additionally, real time communication and objects information sharing between the emergency personnel and onshore base operators must be maintained during the whole response time. This approach will considerably add the significance on the spill response outcome.

Today, there is a lack of application (oil spill response model in ice-bridge simulator) which can be used to train or educate emergency personnel by remotely coupling the spill operational methods with the real time objects information and communication sharing, Fig. 31.

The objective of this thesis is to propose and promote a new feasible approach, namely building such application as a solution proposal and demonstrating its added value on the oil spill emergency response.

The actors' interests and behaviour in using the application is modelled and discussed in details. Simulations showing the Trainees and the Trainer's actions in oil spill response training are made and the results clearly illustrated. The data and pattern generated by the Trainees in oil spill response are retrieved, analyzed, and stored for further improvement.

Although primarily designed for training purposes, this application offers greater room for extensions.

Nomenclature and Conventions

Δx	: [km]	Range swath
Δy	: [km]	Azimuth swath
°N	: [-]	Degree North
°E	: [-]	Degree East
H_{SAR}	: [km]	Flying height (altitude) of the SAR radar
km	: [-]	Kilometer per second
L_X ; L_Y	: [m]	Vertical and horizontal dimension of antenna
λ	: [m]	Wavelength
m/s	: [-]	Meter per second
%	: [-]	Percentage
R_0	: [km]	Distance between the radar and the antenna footprint
r	: [-]	Slant range plane of SAR radar
θ_0	: [^{0}]	Angle of incidence
V_{SAR}	: [m/s]	Velocity of the SAR radar
x	: [-]	Antenna ground range plane
y	: [-]	Antenna Azimuth plane

Abbreviations & Definitions

App.	:	Appendix
ESRI	:	Environmental Systems Research Institute
Fig.	:	Figure
GNOME	:	General NOAA Operational Modeling Environment
GPX	:	GPS Exchange Format
GPR	:	Ground-Penetrating Radar
GPS	:	Global Positioning System
GIS	:	Geographic Information System
IBST	:	Ice-bridge Simulator Training
IBSTraining	:	Ice-Bridge Simulator Training
IBSTesting	:	Ice-Bridge Simulator Testing
NEBA	:	Net Environmental Benefit Analysis
OSCAR	:	Oil Spill Contingency And Response
OILMAP	:	Oil Spill Model and Response System
RUP	:	Rational Unified Process
ROV	:	Remotely Operated Vehicles
SAR	:	Synthetic Aperture Radar
SRID	:	Spatial Reference Identifiers
Tab.	:	Table
UML	:	Unified Modeling Language
WKT	:	Well-Known Text

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

Oil spill incidents at sea are major problem for the environment. Any large scale spill discharged at sea must trigger a prompt and comprehensive cleanup response. The driving factors to carrying out the response are mainly the long term negative impact of oil on the environment, the marine habitat, the ecology and on the socio-economic interests. Most of these spills occur during offshore operations or transportation and are truly categorized as *intentional* or *accidental*. Illegal activities are intentional and include operations such as oily tank cleaning and dumping of residue in the sea. Other types of pollution are *non-intentional* and might result from oil tank leakage, oil pipeline fissure/ ruptures or even from blowout preventer defect on the seabed.

Oil spill pollution has therefore become a great concern for the environmentalist, the government, the offshore production industries, and the general public. The complexity and sensitivity of the Arctic Ocean region bring even more spill awareness compare to open Ocean waters, [13]. Any serious spill emergency response requires skills and good management by the real world decision makers (Marine Coast Guard Authority, on-scene coordinator, oil spill response service companies).

In the current trend, several oil spill response systems are locked on the detection and tracking of marine spill (spill trajectory, weathering and forecasting), and on the implementation of mechanical and non-mechanical recovery systems. However, one can rationally argue that even with the best equipment and technology, there is still some level of uncertainties.

The spill might be detected, monitored and even predicted with great accuracy; the right recovery equipment might be in place, but ultimately the human have to do the cleanup operations.

The feasibility of a successful cleanup outcome will therefore demand additional resources such as gathering real time objects (oil, ice, water, equipment, sensitive and non-sensitive segment areas, sacrificial zones, and other possible neighbouring objects including shoreline, vessel transit line, pipeline, etc) information and communication sharing between the emergency personnel and onshore base operators. The correlation between equipment, response technology and objects information and communication sharing system must be kept as strongly coupled as possible.

Basically, the important factors that could lead to greater certainty in cleanup operations can be formulated as follows:

Equipment + Response Technology + Information System + Human ⇒ Success

The main focus in this thesis will not rest on the static parts of the above formulation (equipment and technology), but rather on the dynamic sections: the information system and the personnel managing available resources.

It is possible to build an oil spill response application model for integration in an overall Ice-bridge navigational interface, taking advantage from the various objects information and transaction, [13].

In real situation, the trained emergency personnel will not embrace the “trial by error” approach, but rather immediately apply knowledge with the support of decision makers.

The motivation for this thesis is therefore the construction of an application for oil spill response which can be integrated as a module in an ice-bridge simulator for training purposes. This application is established upon a comprehensive relational database system with spatial objects (captured by remote sensing imaging with SAR satellite) which makes use of geographical information system (GIS) tools.

1.2 Previous Work

The previous work examines key issues surrounding the driving force of applications of oil spill response in ice-bridge simulator.

Many scientists and institutions have worked on various issues relating to oil spill monitoring and surveillance system based on airborne or space-borne radar system; some data have been captured using a GPR (ground penetrating radar) system. The automatic and semi-automatic oil spills identification from SAR satellite and GIS mapping tools have all gained greater recognition from many institutions - governments, academics, oil service companies, and marine coastal institutions.

These researches often miss the correlations between the relevant technologies and individuals involved in the fight against the oil spill. The management of information in real time during the cleaning is often overlooked.

A reference list of some scientists and their work is presented in Tab. 1.

Table 1: PREVIOUS WORK ON ARCTIC OIL SPILL RESPONSE, [3]

Author(s) / year	Related Work
Mark Reed, Henrik Rye, Ismail Durgut, Øistein Johansen, May Kristin Ditlevsen, Ben Hetland, Marinela Gerea, Boye Høverstad, Kjell Skognes, Ole Morten Aamo, Narve Ekrol, Keith Downing (2010), [21]	“*Oil Spill Contingency and Response (OSCAR) Model System*” Focus on oil spill trajectory and response strategy (Mechanical and Non-mechanical recovery systems) and contingency planning. It utilizes various algorithms to determine the weathering and fate of the oil at sea. The system also assesses the environmental risk and impact associated with the NEBA.
Y. Li, A.J. Brimicombe, M.P. Ralphs (2000)	“*Spatial data quality and sensitivity analysis in GIS and environmental modelling: the case of coastal oil spills*” Focus on the use of spatial data and GIS integration in the case of coastal oil spill model.
F. R. Engelhardt, ENOVA Research Applications, Orleans, ON, K1C 7A9, Canada (1999)	“*Remote sensing for oil spill detection and response*” Focus on the comparison between airborne and space-borne remote sensing technology to capture the spill. It concludes that the space-borne is more suitable in processing the image and should be used in the surveillance system.
William E. Roper and Subijoy Dutta	“*Oil Spill and Pipeline Condition Assessment Using Remote Sensing and Data Visualization Management Systems*” Focus on the detection and monitoring of oil spill by using satellite imagery. Once the spill is detected, an emergency plan can be established to avoid worst case scenario. It also highlights the necessity to include the human expert in the spill classification.
Youcef Smara, Aichouche Belhadj-Aissa and Mostefa Belhadj-Aissa (2005)	“*Application of GIS and Remote Sensing Technologies in Disaster Management in Algeria*” Focus on the importance to choose a digital data capturing technique rather than analogue one. The later is less accurate and time consuming. Incorporating GIS technology

	alongside with remote sensing is a best available alternative for disaster management system.
Yongcun Cheng, Xiaofeng Li, Qing Xu, Oscar Garcia-Pineda, Ole Baltazar Andersen, William G. Pichel (2010)	**“***SAR observation and model tracking of an oil spill event in coastal waters***”** Focus on oil spill detection and tracking with SAR satellite. For spill classification, a neural network algorithm is used. It emphasizes that the ocean current is the most influential factor to model the spill movement. The GNOME software was used to analyze the effect of uncertainty.
Iphigenia Keramitsoglou, Constantinos Cartalis, Chris T. Kiranoudis (2004)	**“***Automatic identification of oil spills on satellite images***”** Focus on the automatic detection of oil spill using SAR satellite with a fuzzy logic classification system. When the SAR detects several black spots, a probability of each spot being an oil spill is calculated and classified in a table. The decision maker – expert will then use the table to advice the emergency personnel.
Camilla Brekke, Anne H.S. Solberg (2004)	**“***Oil spill detection by satellite remote sensing***”** Focus on detecting the oil spill under various environmental conditions using different sensors. Both manual and automatic detection analyse the burden of classifying the look-alikes and the oil spill. It is concluded that SAR imaging is the best alternative as it delivers greater accuracy. It states that the wind speed factor can be problematic on SAR capability, depending also on the oil type and age.

1.3 Present Work

1.3.1 Scope of work

This master's thesis proposes an application to combat oil spills at sea using ice-bridge simulator training.

At first, it provides a solid base on methods for oil spills mapping production using key technologies such as fuzzy logic, GIS, remote sensing and imaging with SAR system. A relational database model is then established based on the interactive Rational Unified Process (RUP). In the RUP, different use cases of key actors (Trainees, Trainer, Administrator, Stakeholder) participating in the application system are modeled with the emphasis on goals and behaviors. Following the use case modeling, a domain model for the application is proposed. To accurately describe the relationship between the use cases of the trainees and the domain model, a logical robustness diagram is constructed.

A further consideration is given to the database server management focusing on the creation of a networked Master and Slave. The network connection is required to establish relevant training courses on the simulator. The validity of the system is evaluated based on its technical performance and operational value.

Finally, a conclusion is drawn on the basis of research results. Some proposals for future work are also presented.

1.3.2 Contributions

The main contribution of this thesis is based on the modelling and implementation of the application. The focus is on the correlation between the mechanical and non-mechanical response technologies and exchange of objects information between Trainees and the Trainer during the training courses.

A distributed database network system connecting the Trainer and the Trainees with different oil spill response scenarios is established. This system is an important support tool for emergency personnel. It allows them to learn new skills and be proactive and ready to combat oil spills

Lots of information are stored and encapsulated in spatial data objects (vector layers) to support emergency personnel during spill response.

1.3.3 Outline Structure of the Thesis

The thesis is organized into six chapters. The chapters are presented thematically rather than chronologically. However, the chapters are related to each other and it is recommended that the reader follows the chapters in chronological order to avoid missing important information. The degree of connection is shown in organigram, Fig. 1.

Chapter 1: Introduction

This chapter provides the reader with information on previous research and the motivation to fight against oil spills at sea. The reasons to work on a new research topic are outlined and justified. These reasons are based on previous research work. The overall research objectives are identified and explained.

Chapter 2: GIS and Oil Spill Response in Arctic Waters

This section examines some of the capabilities of GIS technology with remote sensing imaging - SAR technology. The process of producing an oil spill map is illustrated using a fuzzy classifier system. It is emphasized that this system is at the forefront of today's technology to execute an effective emergency response plan. Information on the treatment of geospatial data in GIS systems is explained.

Chapter 3: Database and Domain Modeling in Oil Spill Response

This section is a solid foundation for the implementation of the oil spill response application in ice-bridge simulator training. The modeling of flow control is established following the UML language. All key actors are classified according to their goal, behavior, and interest. A distributed database server topology is proposed as the basis for the implementation of the training simulations.

Chapter 4: Relevant Simulator-Based Training Courses

This section is a key chapter as it deals with the implementation of training courses. It shows how the Trainees use the system and how the system responds to Trainees' actions. A case study is fully implemented to demonstrate the importance of the application. The results obtained are presented and explained in detail. The training is conducted when the Master and Slaves servers are networked. The system also focuses on the dynamic exchange of information between the Trainer (on Master) and Trainees (on Slaves).

Chapter 5: Evaluating the Validity of Application

This section provides an assessment on the validity of the system. The evaluation is based on the technical performance of the application and its operational value

Chapter 6: Concluding Remarks

This section presents the specific objectives of this thesis. It briefly summarizes the driving forces the application in simulator based training. The limits of the application are also mentioned and proposals for future work are made.

Chapter Connection Level

The organigram (Fig. 1) shows the level of dependency between chapters. Connecting link are used to reinforce the dependency.

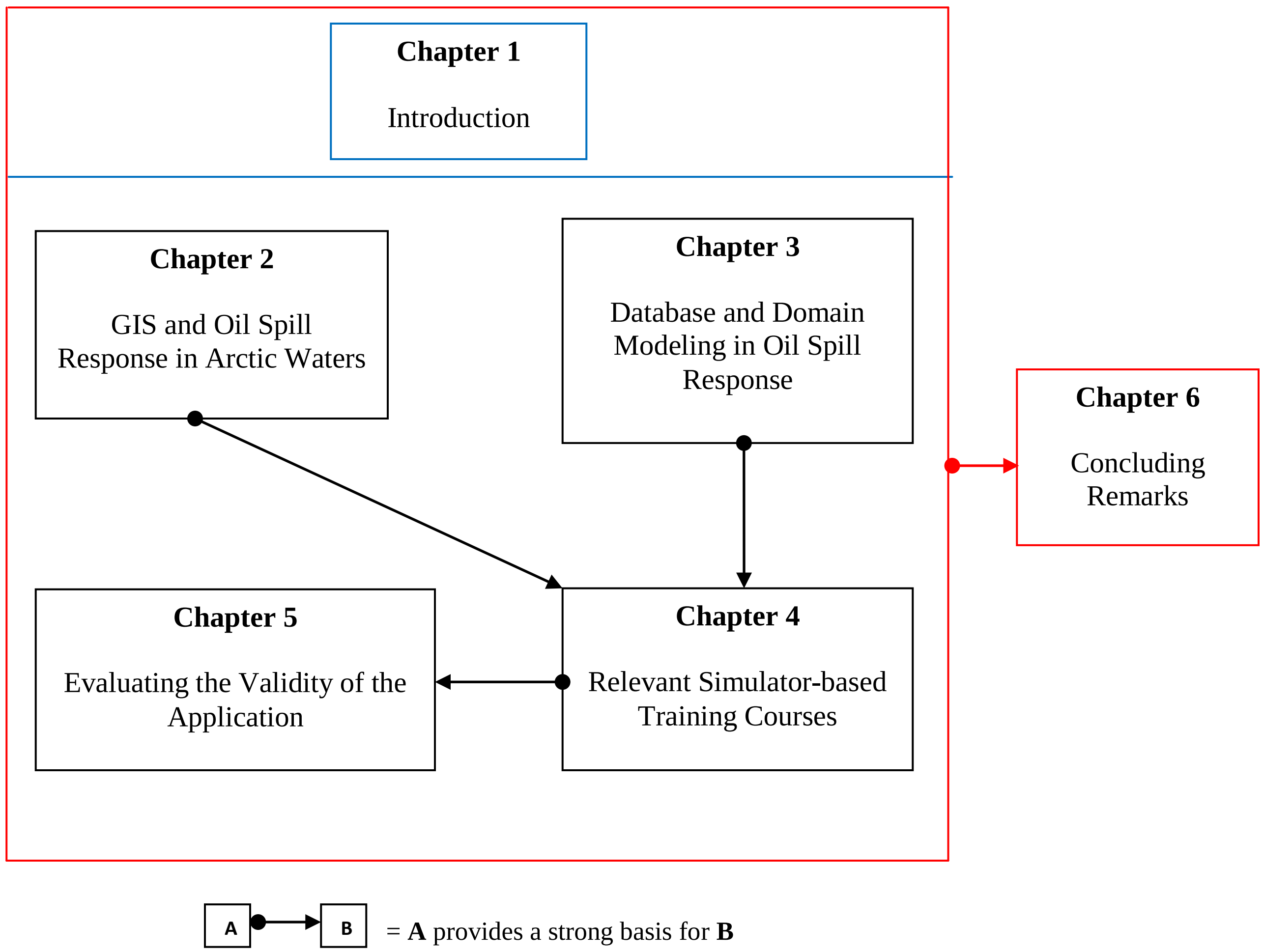


Figure 1: Chapter's Organigram

As chapter 4 receives two connection links, it is a key chapter and meets the objectives of this thesis.

Chapter 2

GIS and Oil Spill Response in Arctic Waters

2.1 SAR System and Oil Spill Recognition with Fuzzy Logic

The value of SAR (synthetic aperture radar) systems in detecting and monitoring oil spill at sea is amazing. GIS (global positioning system) are extensively used in areas such as oil spill mapping or more generally in the field of human resource and environmental management. GIS typically includes several tools such as data management, geometry, geoprocessing, research and analysis. These vectorial tools have several important elements such as data modeling and structuring. One of the main sources of data is the environmental in-situ, imaging with remote sensing and SAR satellite.

Today, SAR systems are one of the most sophisticated systems in capturing images. A SAR satellite does not depend on solar illumination as it can produce its own light. The captured images have a very high spatial resolution. It can operate in all environmental conditions and, consequently, the SAR imaging system is ideal for providing a greater ability to detect and monitor oil spills. As a corollary, using a SAR system will allow a rapid deployment of oil spill emergency response team.

While the SAR radar provides high capacity of spill detection and monitoring, the GIS system on the other hand offers the best tool for oil spill mapping and for displaying relevant information that are stored in vector layers.

The SAR imaging system consists of a transmitter, receiver and antenna. The antenna is used to transmit and receive pulse signals. A SAR capturing raw images from an altitude H_{SAR} and moving at a speed V_{SAR} forms an angle of incidence Θ_0 is illustrated in Fig. 2.

The main reasons for preferring space-borne sensors compare to the airborne is that the former is more accurate and fast due to it constant travel path and stable altitude. It operates independently of solar illumination and cloud coverage. Unlike the former, airborne sensors always have discrepancies mainly due to unstable travel path caused by the aircraft aerodynamics: roll, pitch and yaw angles and speed changes.

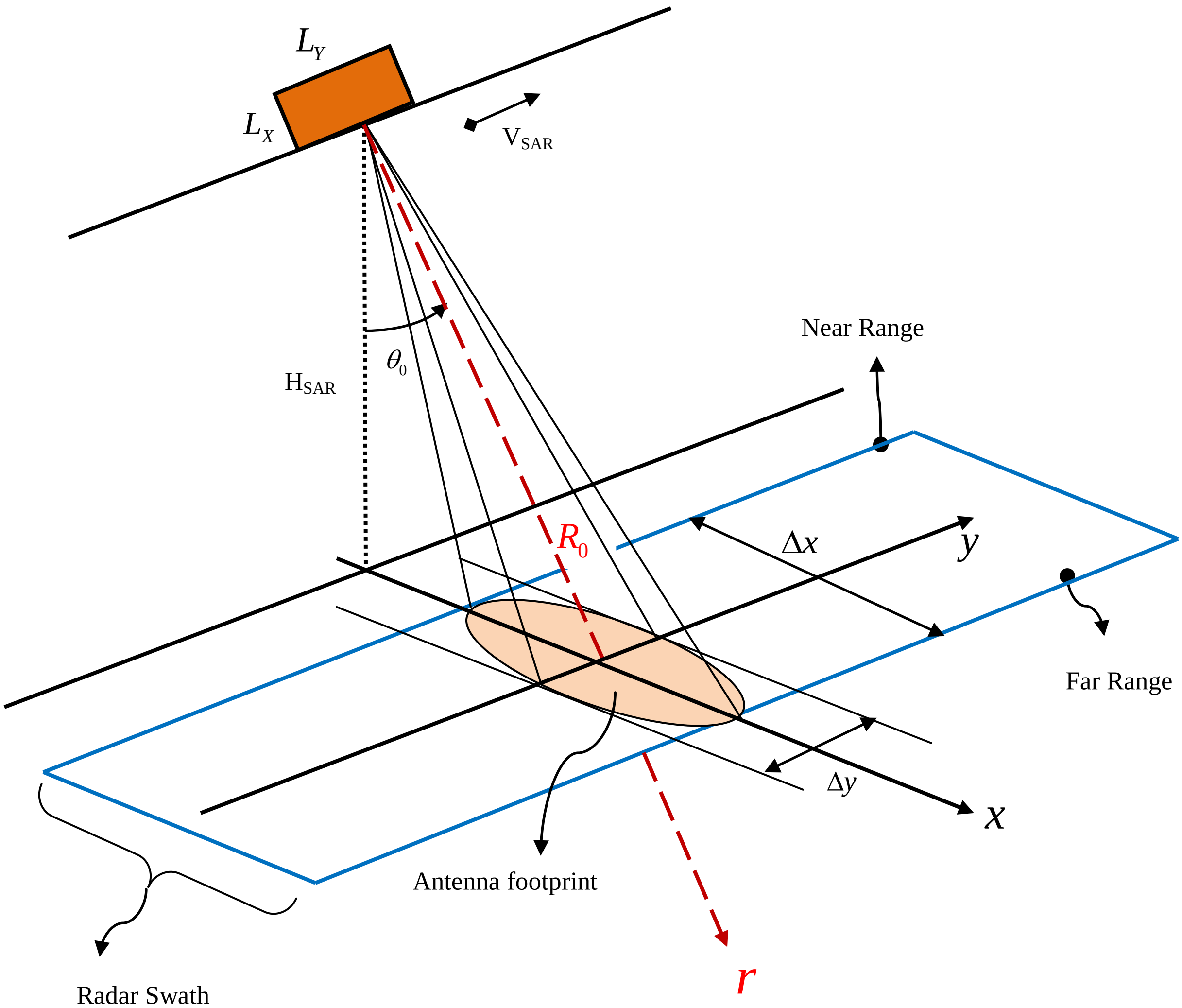


Figure 2: Side-looking SAR Imaging Geometry, [20]

R_0 :	Distance between the radar and the antenna footprint.
L_X ; L_Y :	Vertical and horizontal dimensions of the antenna.
r	Slant range plane of SAR radar
Δx	Range swath
Δy	Azimuth swath
V_{SAR}	Velocity of the SAR radar
θ_0	Angle of incidence
x , y	Antenna ground range plane and azimuth plane, respectively.

It is important to know that the SAR imaging data is first captured as raw data. These raw data include some deficiencies. For these reasons the SAR images must to be pre-processed before it can be plotted in a GIS system. There are different methods of pre-processing remote sensing imagery. The pre-processing algorithms (Tab. 2) are designed to solve the following problems,

[24]:

- Image noise.
- Geometric distortions
- Atmospheric interference
- Varying the geometry of illumination
- Image calibration due to degradation of the sensor
- Correction of image pixels due to topographic effects.

The choice of preprocessing algorithm strongly depends on the type of information one would like to retrieve the remote sensing imagery.

Table 2: IMAGE PROCESSING TECHNIQUES

Image processing Algorithms	Explanations
Range Doppler [1], [2], [3]	"The most straightforward and accurate technique to achieve image formation. The range-Doppler algorithm is the most commonly used algorithm for processing continuously collected SAR data into an image. It is computationally efficient and, for typical space-borne imaging geometries."
Back Projection, [3], [4]	"The most accurate image formation algorithm is the tomographic backprojection. The backprojection algorithm calculates an exact solution for every pixel in the image. However, this approach has very high computational cost. There have been numerous algorithms developed that have acceptable accuracy with much less computational time than the backprojection algorithm."
Omega-K or Wave front Reconstruction [7], [8]	"The most exact form of frequency domain processing algorithms. It is carried out in the 2-D frequency domain and allows the processing of very high azimuth aperture data."

The system presented in Fig. 3 is based on a framework for the detection of oil spills and mapping. It consists of four key systems:

1. ▪ Remote sensing imaging system (airborne, space-borne, vessel, GPR, Diving, ROV, Drilling);
2. ▪ In-situ data system (environmental weather, sea state, ice data);
3. ▪ GIS integration system (cartography, site resources information, bathymetry, coastal hydrographic, sensitivity areas, ship routing, rigs platforms);
4. ▪ Oil spill response system (operational response options and methods).

In SAR image processing, the land masking is used to eliminate all unnecessary objects in the production of sea map. The land masking function in SAR must not discriminate the ice parameters because the oil can be encapsulated in ice.

The smoothing process consists of applying Gaussian blur to remove unwanted information and noise in the image.

The segmentation and thresholding are processes of partitioning digital images into multiple segments. Their main objectives are to simplify the representation of images so that they become much easier to analyze.

Fuzzy Logic Systems

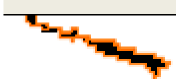
The purpose of the fuzzy logic algorithm is to find the probability of a dark object to be an oil spill. This is to avoid or reduce false alarms. From image processing to segmentation and thresholding, the human expert may still not be able to identify the look-alikes. Therefore, a fuzzy system is used right after the segmentation and thresholding process, Fig. 3.

Dark spots on the map can be presented in a tabular output format showing the probability of being an oil spill. This is because some dark objects may just be look-alikes (other type of spill, for example from algae) rather than oil spills.

The value of probability will mostly allow the expert decision makers to make better judgement on whether an emergency response action needs to be dispatched.

The fuzzy logic system is based on the fuzzification of crisp input (numerical value) into linguistic language (black, gray, incertain). From this stage, the inference checks the linguistic value with a set of knowledge base (data and rules).

Table 3: PROBABILITY OF A DARK OBJECT TO BE AN OIL SPILL, [12]

Dark object on image	Number of objects around	Area (km^2)	Eccentricity	Land distance (km)	Longitude (°N)	Latitude (°E)	Probability to be an oil Spill (%)
							

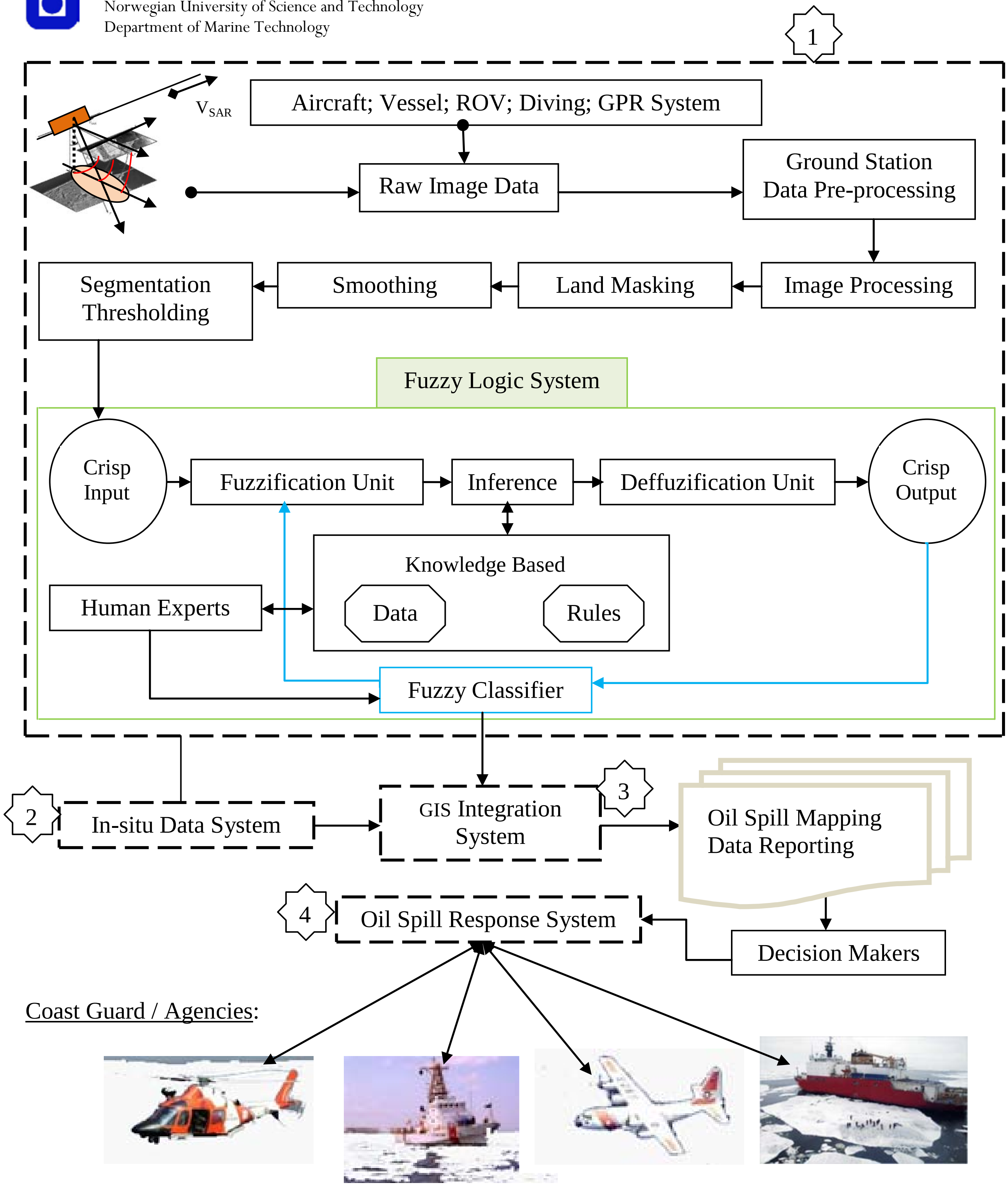


Figure 3: Framework for Oil Spill Response - Detection with Fuzzy and GIS mapping, [26], [27], [28], [5]

When a linguistic value is found, a defuzzification process converts the value into crisp output (numerical value). At this stage, if the object is successfully classified by a fuzzy classifier, it may be plotted in GIS system. If the object has failed in the classification process, the classifier will send the failed numerical value for re-fuzzyfication.

The figure also shows that the human experts can influence the knowledge base system. Actually, they use their own knowledge and experience to build the knowledge base component. They have knowledge on all databases (oil, ice, weather, and ocean), so they can select linguistic values and send the their corresponding numerical value directly to a fuzzy classifier.

The fuzzy logic system is a loop-system and a set of data and rules can be changed over time.

In-situ data System:

The in-situ data system is one of the most critical systems that affect the production of oil spill maps. Although it provides important information on the weather and environmental conditions, it is often a source of creating uncertainty in a model.

GIS Integration System and Oil Mapping:

After the raw data processing, smoothing, thresholding, segmentation and fuzzy classification, an oil spill map output is produced using a GIS tool. Vector layer objects such as oil spill, ice, water can be organized, structured and stored in a database. These data can be easily accessed by emergency personnel.

With a GIS system such as QuantumGIS, it is possible to retrieve the data from a database and plot them directly to a GPS unit (GPSbabel), [25]. For example by using a GPX (GPS exchange format), emergency personnel will be able to plot the boundary of protected and sensitive areas during the spill response. They will also have the ability to process the data on the oil map: Setting waypoints, tracks, routes and add new icons (from symbol manager) as shown in Fig. 4.

Each icon contains specific information: Place of refuge, resource information - equipment, protected areas and sensitive areas, dangerous areas, anchor place, sacrificial zone, ice type, etc.). The map can be shared between emergency personnel and experts through a network to improve the oil spill response strategy.

Decision Makers:

After mapping the oil, a report document is generated and carefully studied by decision makers onshore. If there is a spill, then an alarm is triggered.

The emergency response system is based on the response options, which includes both the equipment strategic and the equipment tactical logistic features.

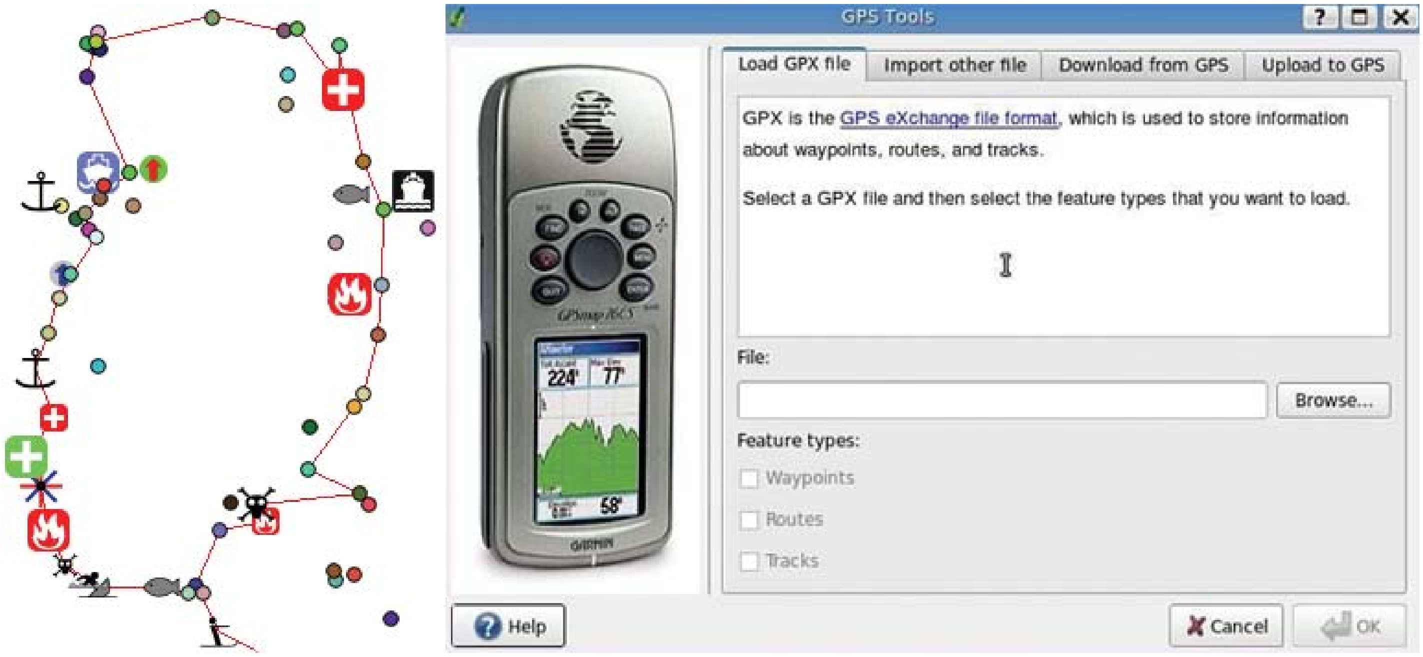


Figure 4: Plotting GPX Data File Sample

- In the ***equipment strategic***, information related to the storage of equipment for future spill is outlined and includes:

Storage type of equipment: Ice-breaker, ice-boom, ROV, chemical dispersant, mechanical recovery devices, airborne craft, coast guard vessel, oil storage tank, etc.

Storage quantity of equipment: Whether the number of necessary equipment are sufficient enough to contain and recover the spill.

Storage location of equipment: Whether the equipment are stored in proximity of the spill or far away. And the time requires to dispatching additional and specific equipment to the spill site.

- In the ***equipment tactical logistic***, information related to the equipment to be dispatched on the spill site is outlined and includes:

Type of equipment to be dispatched: Making decision on which equipment to be dispatched depend on the equipment strategic and on the external weather conditions (freezing temperature, operational visibility, and the locale ice regime).

Quantity of equipment to be dispatched: Coping with resource management during spill events is important. It is not always favorable to bring any equipment available on the spill site. This often requires considerable amount of time and resources in the transportation, especially in a difficult and an isolated environment (Arctic Ocean). Such decisions should be taken by decision makers who must consider the nature of the spill and the net environmental benefits analysis.

Equipment exposure time: This information strongly depends on the external weather conditions. The duration of equipment is calculated in order to prevent them from icing or malfunctions.

2.2 Spatial Data Objects

Spatial objects include the geometry and geography datatypes of environmental or real objects. The basis is that all objects including environmental object (Ice, Oil Spill, Water segment, and Weather conditions) and other types of objects (not limited to) such as Ships, Land, Equipments, Florae and Faunae, can be modeled as spatial data objects depending on a specific problem domain.

When using a GIS system, such spatial objects offer the opportunity to perform operations such as: ***data capturing***, ***data structuring***, ***data manipulating***, ***data analysis***, and ***data presentation*** (Raper and Maguire, 1992), [15]

• Spatial data can be captured manually or automatically. These data can then be inserted into a computer program for geoprocessing. The output may be a raster or vector data format.

• Structuring spatial data is an essential step in building a spatial database. Depending on the complexity of a problem, a database can be structured in an object-oriented or a relational model in the implementation. Structuring of data will provide greater value for data manipulation

• Manipulating spatial data involves the removal of roughness and irregularities around objects. This is achieved by averaging data points and lines to their closest neighbors. The process is named spatial data smoothing. Other types of operations include map scaling and coordinate transformations.

• Analyzing spatial data is a central and essential part of GIS system. The geometrical and geographical data-types properties of spatial objects are very important. The analysis is also made with the purpose of carrying out objects modification to ensure their validity. The need to obtain a clear and simplified geometry is another reason to analyze the data. The need to reduce processing time and achieve high performance is also important because the size of the SAR's raw data is usually large, requiring considerable time to process.

• Presenting spatial data. When all relevant data has been captured and stored in a database in a structured way, they can be retrieved from the database and plotted on computer screen for visualization. These tasks are often achieved by the means of spatial data syndication. The plotted result is a map showing objects such as polygons, linestring, and points with content specifications.

There are several commercial organizations or government institutions that provide information of spatial data in formats that are fully applicable in GIS system. The format used in this thesis is ESRI (Environmental Systems Research Institute) shapefile. A shapefile is a geospatial vector data (with attributes) for GIS system and most commercial spatial data are supplied in this or similar format.

A single shapefile is composed of four file extensions sharing the same data information:

.shp file: this file contains raw geometrical shape data like points, lines or polygons. It is the main file and occupies the largest amount of space comparing to the other file extensions.

.dbf file: this file is the data specification and occupies the second largest amount of space after the .shp file.

.shx file: this file takes care of the shapefile index. It allows seeking the shape forwards and backwards quickly. This file helps improve the performance of spatial data manipulation.

.prj file: this file is a projection format; it gives details about the projection in which the coordinates of the geometry data are represented. The format is as in plain text file WKT (Well-Known-Text). This file also contains information that is required to determine the correct spatial reference identifier (SRID) in geometry.

Chapter 3

Database and Domain Modelling in Oil Spill Response

3.1 Actors and Domain Modeling

In the modelling part of the actors, the emphasis is placed on the optimal objective of the actors and their behaviour. In the modelling part of database architecture, the emphasis is on physical design of the database.

In this thesis, the Ice-bridge simulator training (IBST) comprises two essential parts:

1 – The training of Trainees with six different training scenarios

2 – The testing of Trainees with six different testing scenarios

Each scenario consists of steps that can be enumerated as follows:

1. A message of an actor to the system;
2. A validation or a change of state of the system (due to message received);
3. A message of the system to the actor.

These scenarios are performed by trainees (Trainee1, Trainee2, Trainee3 ... etc.) located in training rooms (Room 1, 2, 3, and so on). All scenarios are managed and controlled by a trainer located in the control room. For simplicity, only two students are taken into account in the explanation. In explanation, trainees are often skewed as Slaves and the trainer as Master.

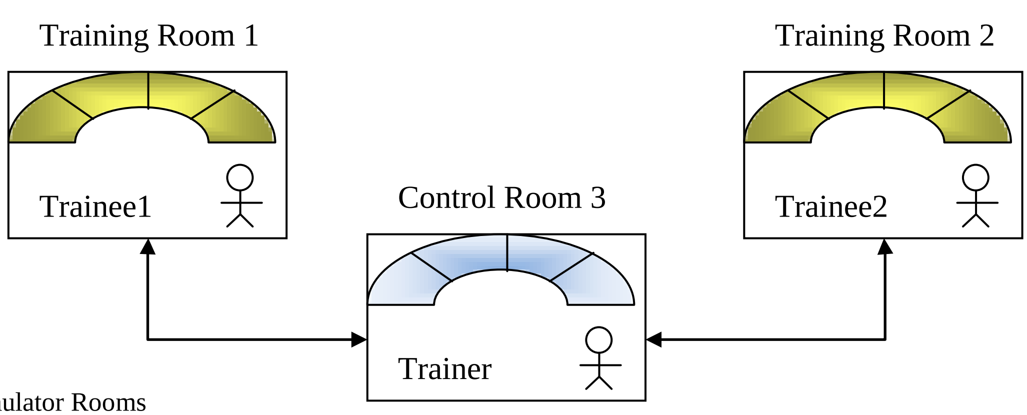


Figure 5: Simulator Rooms

Augmenting the number of trainees will only be a matter of updating the server setting parameters.

During the training and testing scenarios both slave and master servers are networked. Each trainee receives appropriate data and selective courses from the trainer. The total number of scenarios is twelve: six allocated to training and the rest for the test (exam). Both types of scenarios are stored in databases called *IBSTraining* and *IBSTesting*, (App. E) respectively. The implementation is made so that trainees simultaneously receive the same scenario with relevant information on their screen.

The scenarios also give provision for applying operation methods (Mechanical and Non-mechanical systems) in oil spill emergency response. Each scenario produces a unique spill map with the possibility to generate reports as described in the framework system, Fig. 3. The oil spill mapping is being generated as a geospatial vector data. These vector data are suitable for visualization and analysis through GIS application such as ArcGIS, QuantumGIS, etc.

In this thesis the QuantumGIS application is used for plotting the data stored in the IBSTraining and IBSTesting databases. QuantumGIS has geoprocessing capabilities which make it suitable to automate the GIS mapping functions from input data and solve complex problems.

3.1.1 Use Case Modeling

Modelling use case is a process of defining the behavioural requirement of the system. Issues such as how the user will use the system? How the system reacts to user action? What can happen after system response? What are the user's responses vis-à-vis the system response? - As expected, this part of modelling is strongly linked to screens and user interfaces. Thus, the use case modelling is a dynamic system. It emphasizes on the objectives and on the needs of the user and often follows the flow of events and response as shown in Fig. 6.

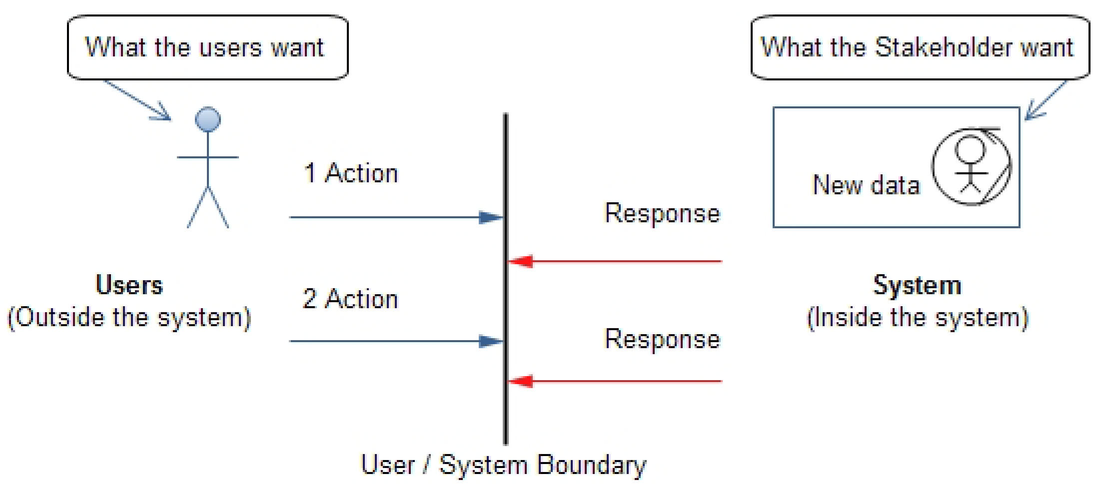


Figure 6: Users and System boundary, [10]

An Actor is defined as the one who benefit from the utilization of the overall system. As can be seen from the Fig. 7, there are two main actives Actors: the User and the Stakeholder. A User is an Actor having behaviour - sending command or request to the system. This User could be a person, an organization, or a combination of them. In this thesis, the User is referring to the ***Trainer***, the ***Trainees*** and the ***Administrator*** all of them acting from outside the system, Fig. 8.

The System is defined as an internal Actor responding to User. Another important Actor is the ***Stakeholder***; Fig. 6, 7, 8, 9. Strictly speaking, this Stakeholder is an external abstract Actor.

The Stakeholder is not a human Actor, but rather an external system interacting with the internal IBST system. Stakeholders are entitled to see their interests protected by the system; their only goals are to ensure that the internal system delivers, i.e. a system with quality training. It does this by providing updated information on environmental or geospatial data which are considered essential for the simulation of course scenarios.

For example, these Stakeholders could be identified as systems (Service Companies) providing organized ready-to-run, easy-to-update and easy-to-reuse geospatial data files (input dataset) from sources such as Arctic Ocean region, environmental weather conditions, bathymetry, ecological and biological resources distribution, and local ice regime. It can also be the data files specifying oil spill map.

In figure 8 and 9, goal oriented behaviour patterns are established. It shows that one Actor has plenty of behaviour vis-à-vis the system that is being used. A Stakeholder has a lot of interests in seeing the system capable of delivering a goal in term of responsibility (its mission). To achieve a goal, the interests of the Stakeholder are protected by private action, which in fact perform control and validation of the input dataset.

Many actors can participate in one interaction. An interaction (a subclass of Action super-class) is an operation massage or a set of sequences transmitting messages across the system to achieve goals. All these activities are the responsibility of the Actor's behaviour.

Due to the abstract participation of stakeholders, one may wonder why to include stakeholders in the modelling, especially when the model must be kept simple, readable, and under control at all times.

Therefore, it is assumed in this thesis work that all relevant data such as oil spill data, ice data, weather data, and Ocean region data are available as real- time data input and ready to be used into the ice-bridge simulator system. This assumption will only reduce the complexity in modelling the Stakeholder's use case: From the conceptual domain point of view, building a complete Stakeholder's use case model is beyond the scope of this thesis.

Each of the four actors mentioned above has its own mission, responsibility and behaviour as it interacts with the system. Tab. 4 shows most of their main tasks, and forms a solid base to build the model.

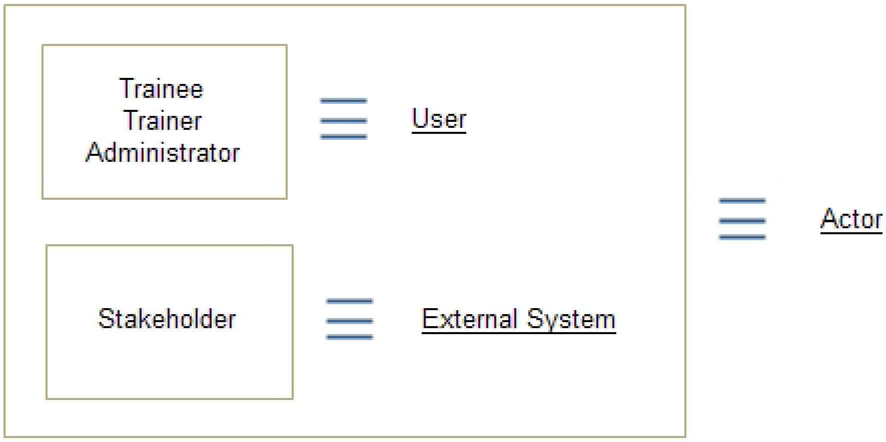


Figure 7: Actor Naming Concept

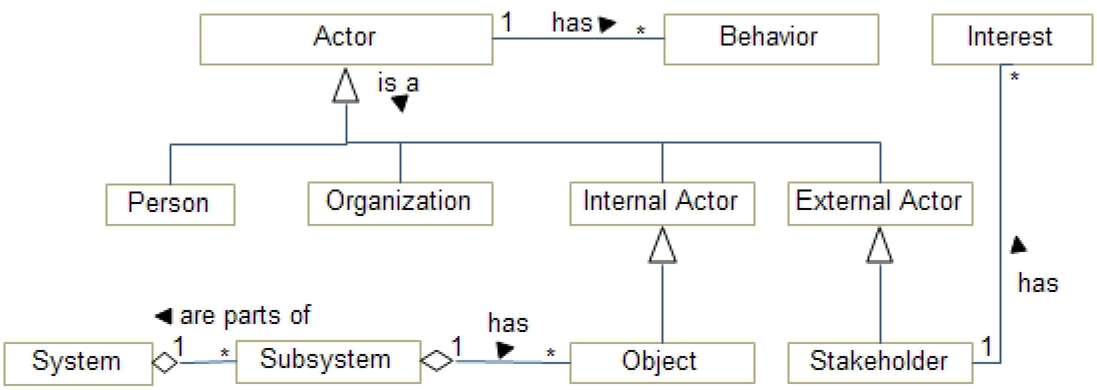


Figure 8: Actor, Stakeholder and System vs. Behaviour and Interest, [6]

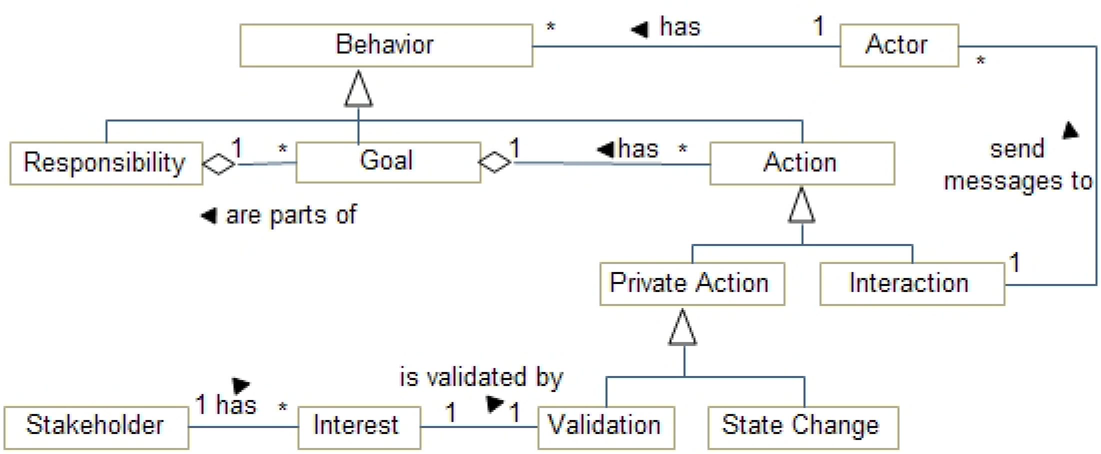


Figure 9: Goal Oriented Behaviour: Responsibility, Goal and Action, [6]

1 = only one. * = many. Reading direction arrow▶

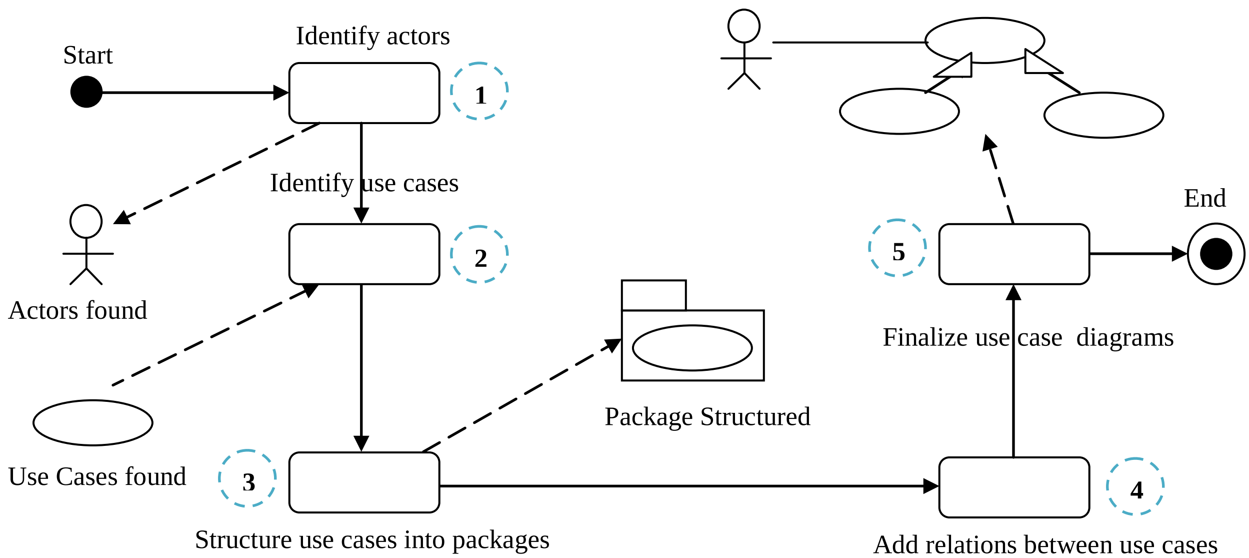


Figure 10: Synoptic Procedure of Use Case Modelling, [9]

One way to make good use case model is to follow a synoptic procedure, as shown in Fig. 10.

To achieve readability and avoid confusion among actors, specials and standard icons are used (Fig. 11) following the recommendation of Ivar Jacobson – a pioneer of UML modelling.

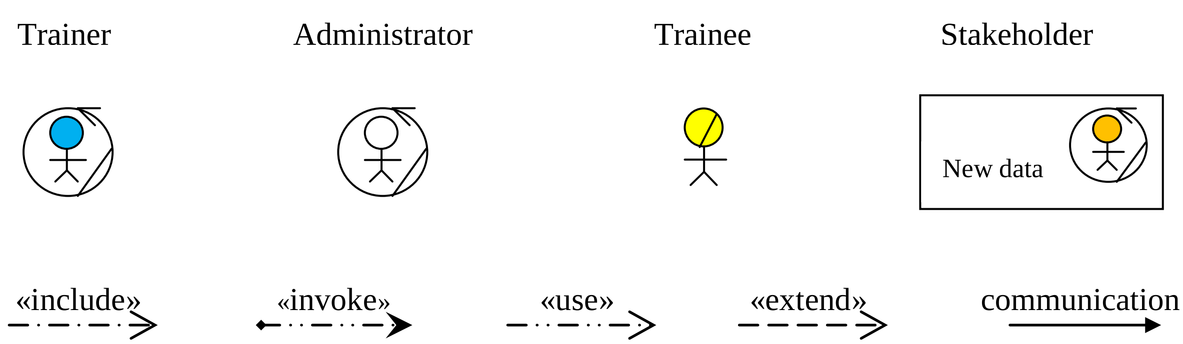


Figure 11: Actor and Arrowhead Symbols in Use Case Diagrams, [9]

Refer to App. D for symbols definitions.

.Table 4: ACTORS OPTIMAL NEEDS AND BEHAVIOUR

Administrator	**Trainer**	**Trainee**	**Stakeholder**
Login / logout Manage user account (system) Request help Provide technical help Manage training program: [update databases and tables; manage new data; validate data] Maintain database (backup, recovery, debug) Manage the network connection (connect / disconnect Master and Slaves interaction) Monitor the system performance Evaluate technology Generate technology report Provide recommendations	Login / logout Request help Provide help Manage database training contents: [update oilspill weathering; update Arctic oilSpillMap; update response methods; update risk factors; update training dataset (ice, oil spill, weather)] Manage database test contents: [update test dataset (ice, oil spill, weather)] Manage database / table (select, alter {add, delete, populate, update, lock, unlock}) Plot oilSpillMap Assess situation Monitor trainees' response action and procedure Evaluate training: [check trainee response report card; check training program; check training objectives; check trainee feedback; check technology report] Make recommendations to Trainees based on their test responses Manage trainee account (training /test): [grant access; revoke access]	Login / logout Request help Perform training: [study arctic oil spill weathering; study arctic oilSpillMap; study arctic regime; study cleanup response methods; compare methods] View training report Perform test: [retrieve dataset; plot oilSpillMap; assess situation; do nothing; take response action; select response option; select best response method; protect sensitive areas edit response; stop response] Submit the test View test report Give feedback	Supply new data: [digital data files (ice, oil spill, wind, oil Spill region, weather, GIS map)] Provide help

3.1.2 Use Case Diagramming

The use case diagram is considered as a high level requirement analysis for the system. Its shows the relationships levels between Actors and use cases.

The Actors' important activities are represented in terms of use cases as established in Fig. 12, 13, 14, 15, 16. From those figures, it is for example shown that the association between the user and the use case "Request help" demonstrates the transmission of information between two actors. It is also noticed that one use case can trigger a direct participation of other use cases and an actor can participate in several use cases.

An extension association is used to illustrate the dependency between two use cases. It also indicates which element started the interaction. For the extension association, the initiator always remains at the end of the arrowhead.

For example, based on the Trainee's own situation assessment, he/she can decide to take action (Fig. 13). The "*Assess situation*" use case is the initiator while the "*Take action*" is the executor.

In figure 14, there is a Stakeholder (a non-human Actor) responsible to supply new data.

The Stakeholder represented in the model is a software-intensive system and reveals its optimal goal.

From the application point of view, one might be interested to know the functional priority level (high, medium, low) among the use cases, and what are the risks level (high, medium, low) in implementing each use case.

Indeed, the classification of use cases depends on their degree of importance balanced with their difficulties in implementation. Merging the use cases' functional priority with risk assessment will lead to a reasonable classification of the iteration.

A shortened list of use cases' priority and risk levels for the ice-bridge simulator training are classified in Tab. 5.

High Level Use Case for Ice-bridge Simulator Training (IBST):

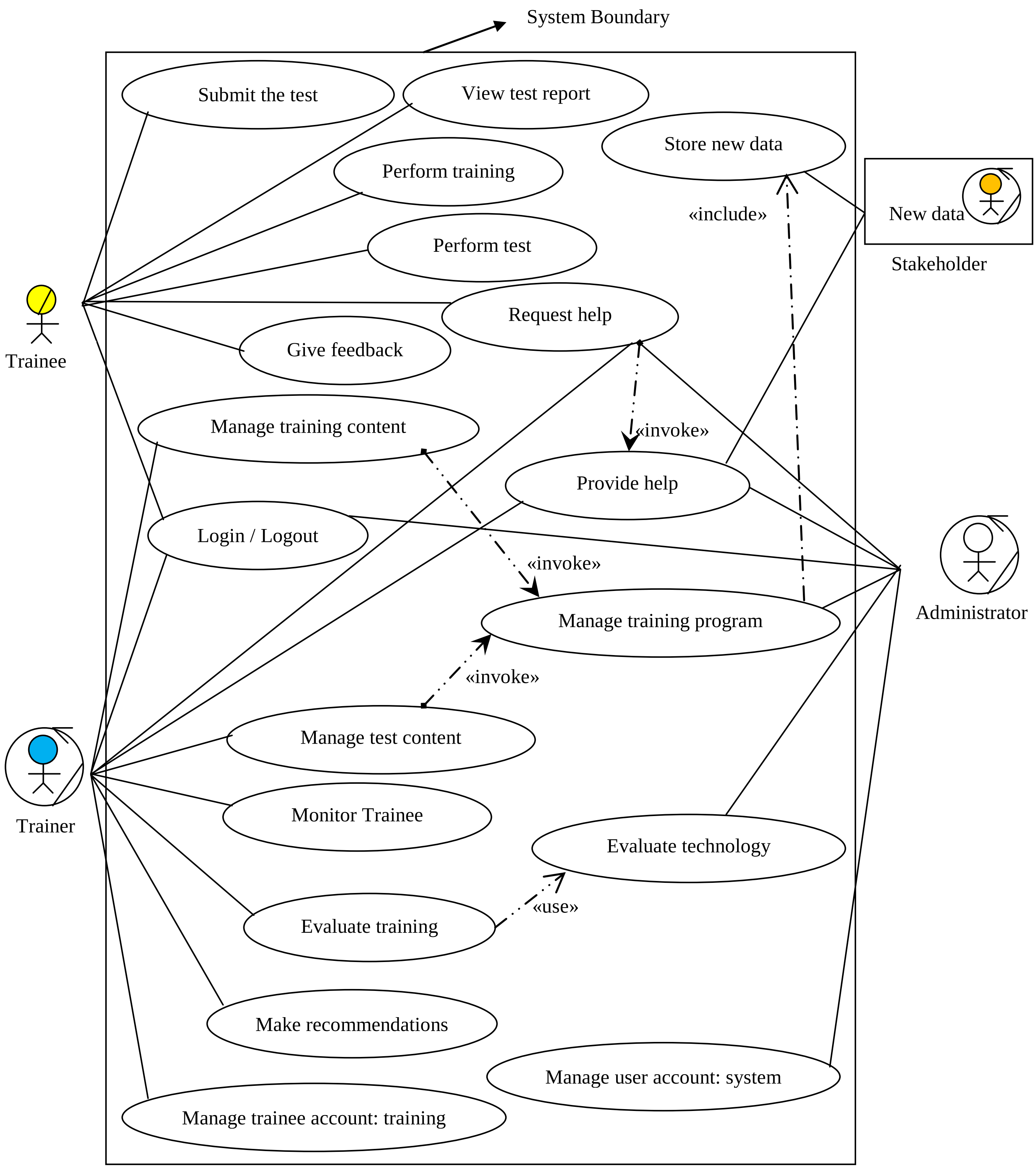


Figure 12: High Level Use Case Diagram for Ice-Bridge Simulator Training (IBST)

Use Case for Trainee:

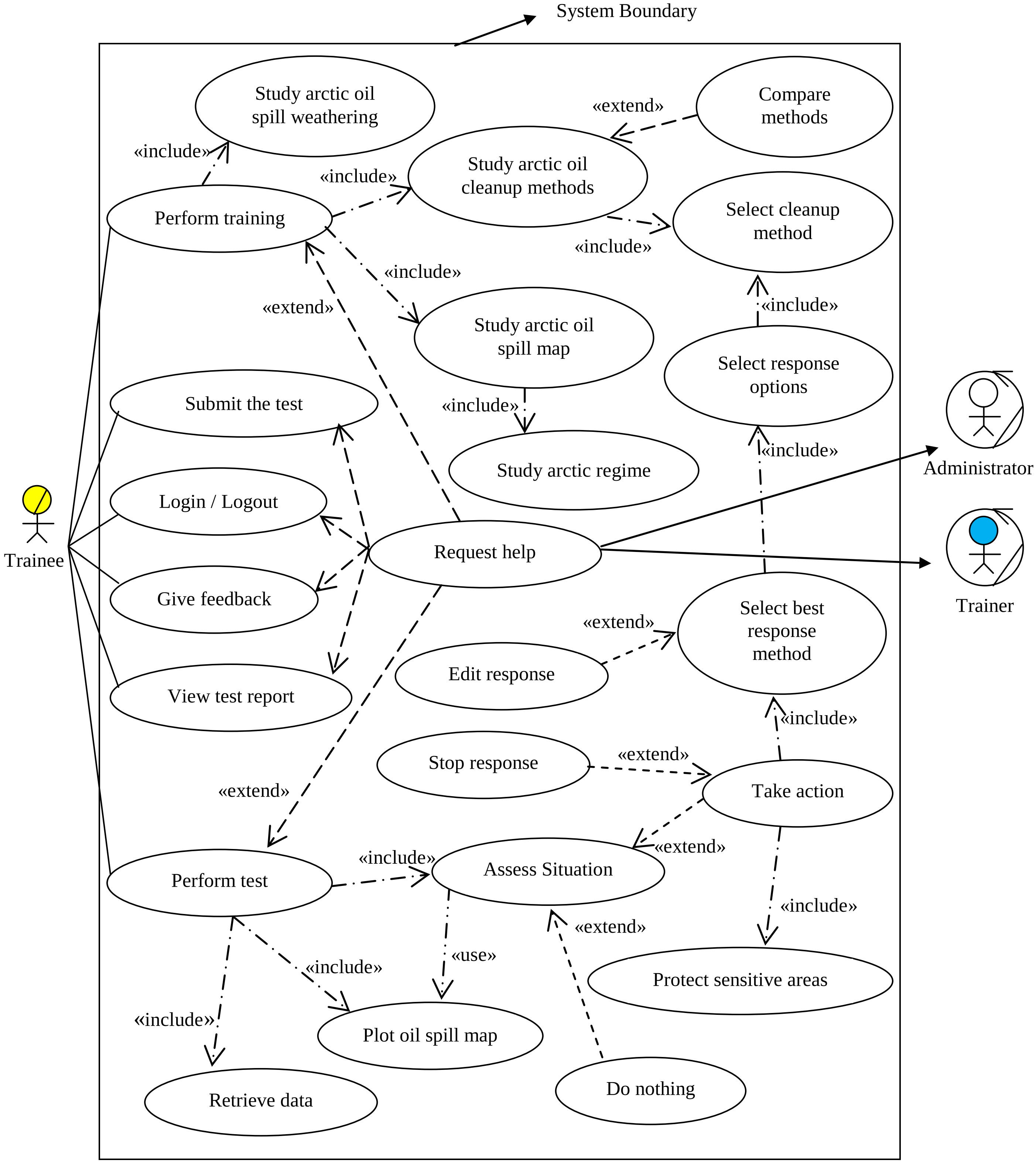


Figure 13: Use Case Diagram for Trainee

Use Case for Trainer:

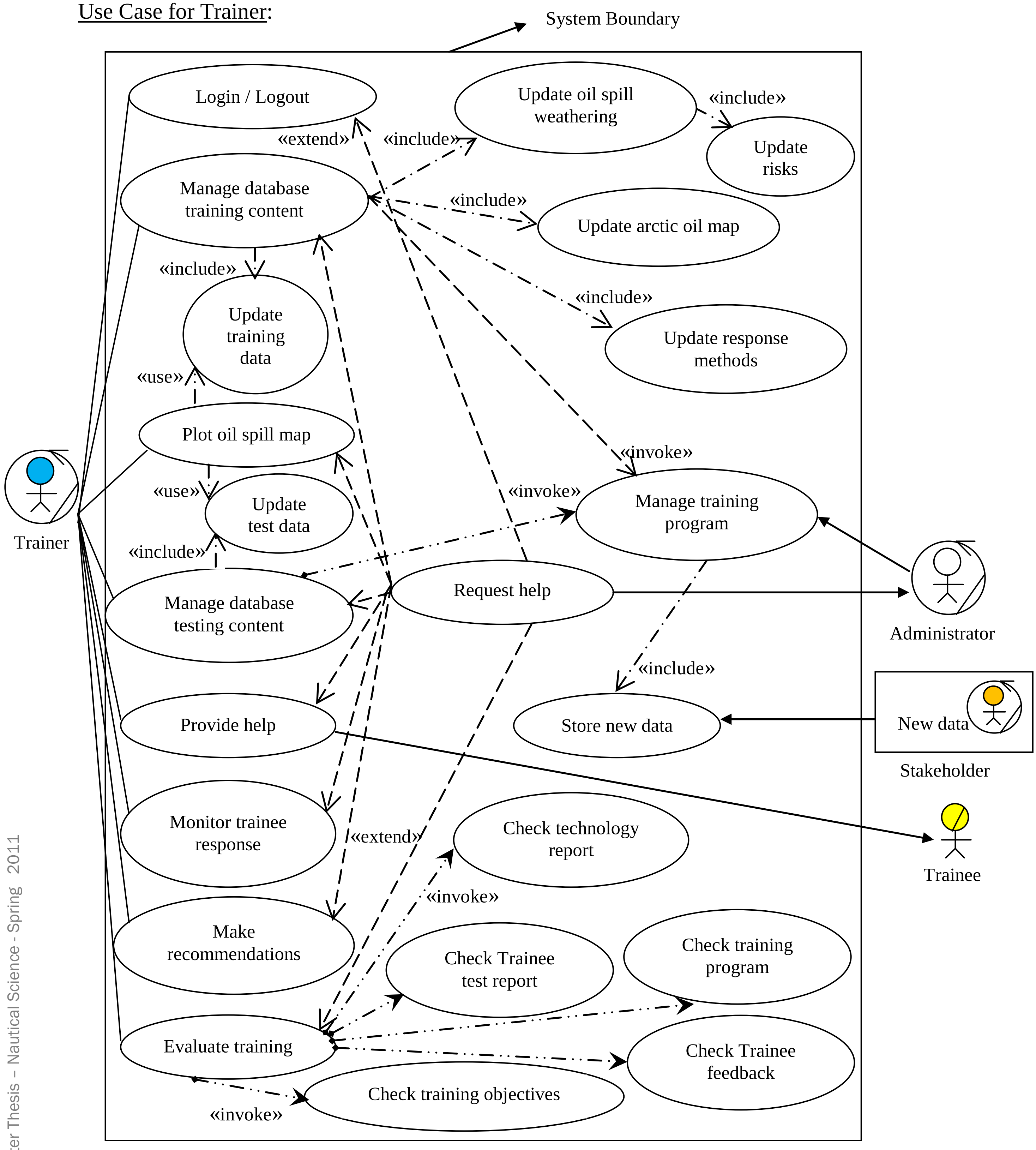


Figure 14: Use Case Diagram for Trainer

Use Case for Administrator:

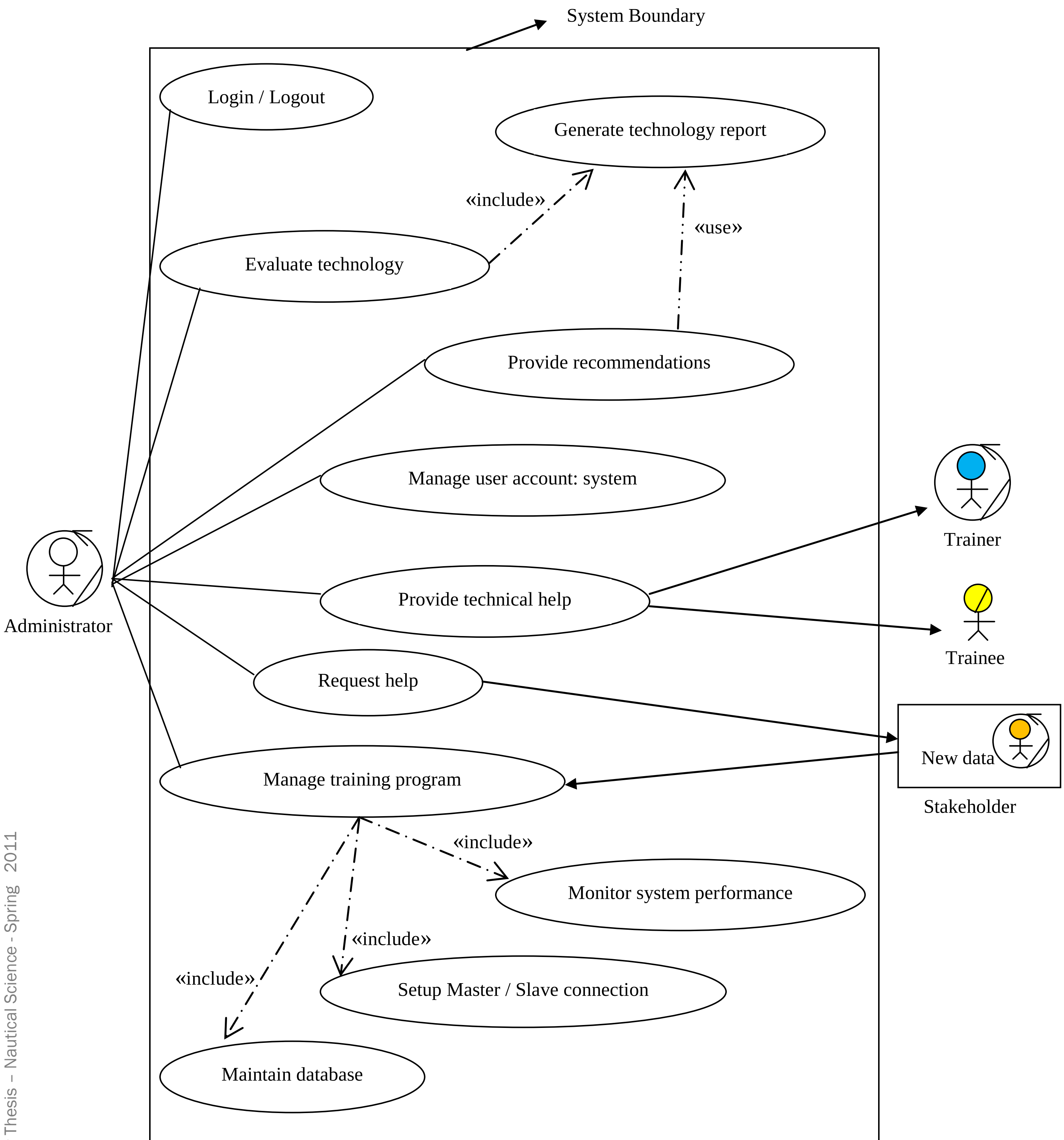


Figure 15: Use Case Diagram for Administrator

Use Case for Stakeholder:

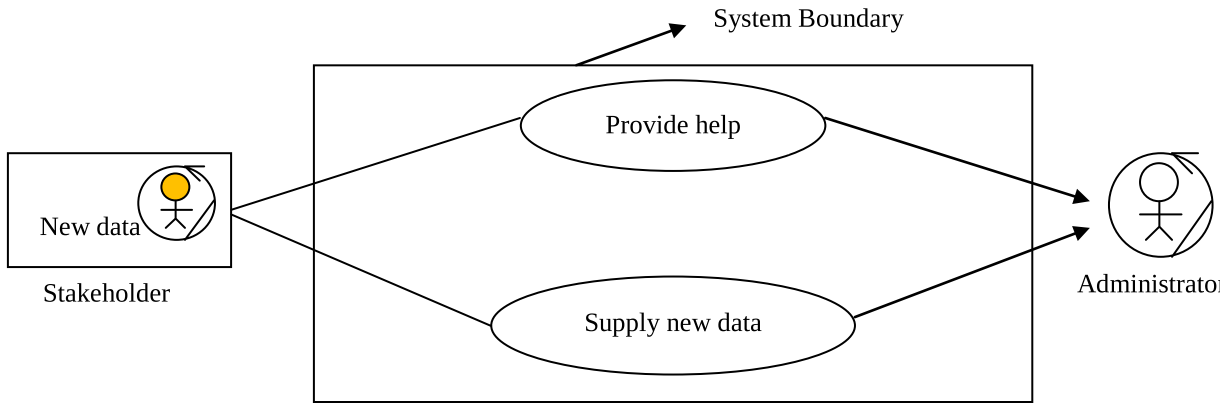


Figure 16: Use Case Diagram for Stakeholder

Table 5: USE CASES CLASSIFICATION

Use Case	Priority	Risk
Trainee		
Login / Logout from system	Medium	Low
Retrieve data from database	High	Medium
Plot oil spill map	High	Low
Assess situation	High	Low
Select response method	Medium	Low
Perform clean up response	High	Medium
Submit the test	High	Medium
Submit feedback	High	Medium
View test result report card	Low	Low
Request help	Low	Low
Trainer / Administrator		
Provide help	Low	Low
Create database	High	Low
Manage training / test database	High	High
Plot oil spill map	Low	Low
Read/Write from database	High	Medium
Request data / help	High	Medium
Monitor / Manage the system	High	Medium
Setup Master / Slave	High	High
Monitor Trainee	High	Low
Give Recommendations		
Retrieve data from Trainee		
Submit test results		
Manage Trainee access account	Low	Low
Generate reports	Low	Low
Stakeholder		
Supply data	High	High
Provide help	Low	Low

3.1.3 Domain Modeling

The reasons for building a domain model are the identification of objects that are important for the application. These objects are essential tools to ensure good communication. It requires the identification of real world objects (oil spills, ice, protected areas, etc.) that are accessible by programs.

The generic modeling process is illustrated in Fig. 17, and the results in Fig. 18.

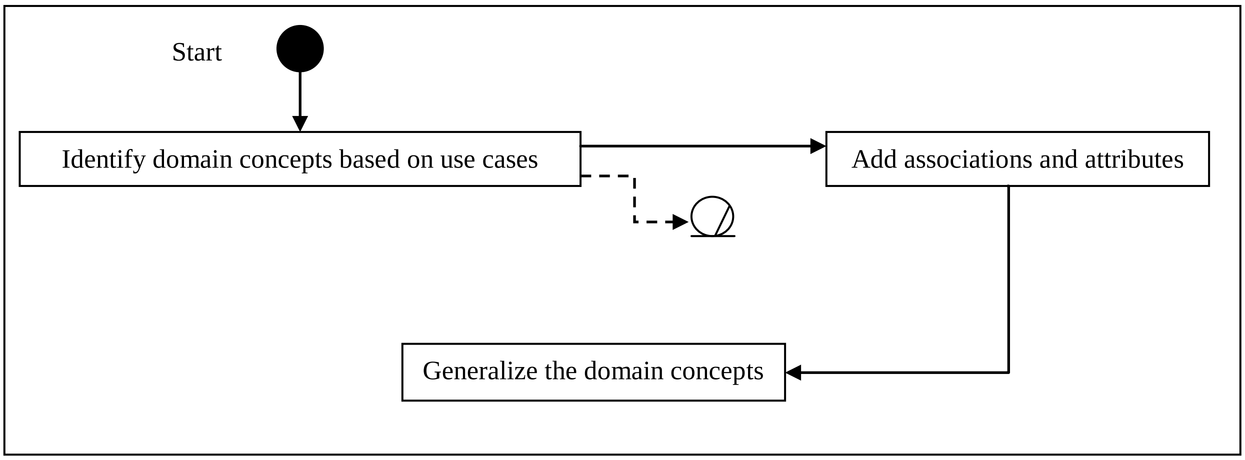
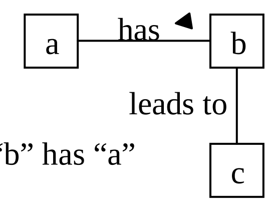


Figure 17: Framework to Building a Domain Model, [9]

Concepts Reading Convention for Fig. 18:



1. Read a concept from left to right class: ex: “a” has “b”.
2. Read a concept from top to bottom class: ex: “b” leads to “c”
3. If a directed arrow is used, then it overwrite the above conventions: “b” has “a”

The symbols used in the diagram are explained in App. D.

The figure 18 represents a domain model for the IBST. It shows that an Actor is composed of a Trainee, a Trainer, an Administrator and a Stakeholder; each having one account (with different access privileges) and managed by the Administrator. One or several Trainees attend one or several courses given by one or several Trainers.

The courses which the Trainees attend are included in a *course catalog*. Within the course catalog there are educational *training catalogs* and *test catalogs*. The Trainees are allowed to participate in the training session before taking the examination test. But they are free to choose and enroll for the test only (for example, return candidates). The training catalog also includes a *map catalog*, an *environmental catalog*, a *response method catalog* and an *equipment catalog*.

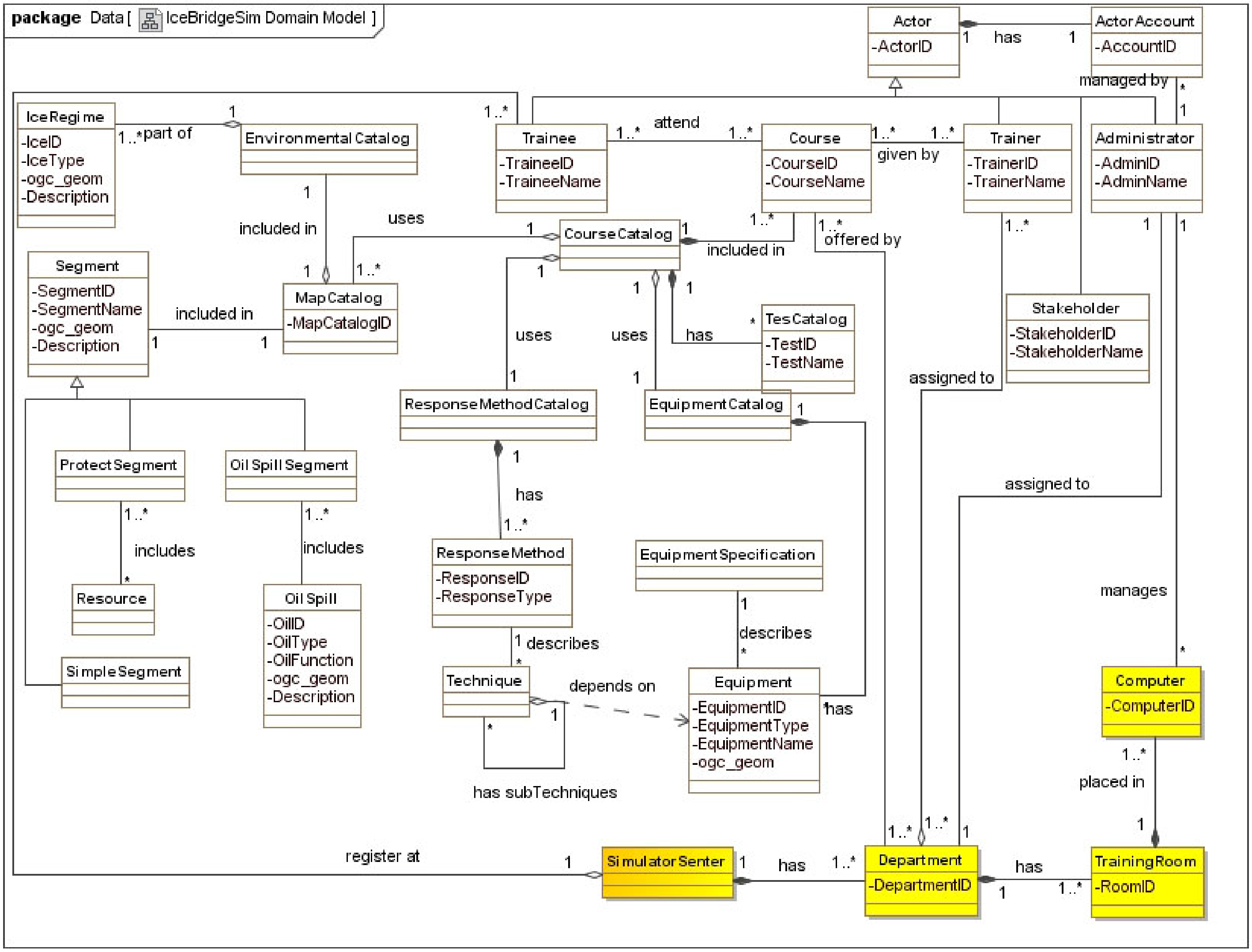


Figure 18: Ice-Bridge Simulator Training Domain Model Diagram

<u>Training Catalog</u>:

The map catalog is a GIS map over an area of interest. A complete map includes a segment location. The segment can include: a protected area (environmental sensitivity) accommodating various resources; an oil spill area with spill information (volume, depth, weathering, rate, viscosity and profile); a simple area (sacrificial zone) or a combination of them. The map catalog also projects the environmental conditions (wind, air / water temperature, wave height, current, ice, snow).

<u>Response Method Catalog</u>:

The response method catalog is a catalog that includes different response techniques: Mechanical recovery systems (skimmers, Booms, Ice-booms, weirs, pumps, ice management, etc) and Non-mechanical recovery systems (in-situ burning, chemical dispersant, etc). Each of these methods describes different implementation techniques knowing that one technique has several sub-techniques. The uses of any of these methods rely upon the equipment strategic and tactical logistic, as mentioned in paragraph 2.1.

Prior the response action, a situation assessment is conducted by the Trainees based on the plotted oil map. Decision has to be made whether any action should be taken or not. In case of action, a response policy and procedure is followed taking into account the severity levels of the spill, the coastal proximity, the proximity and vulnerability of a protected area. The decision making process can be supported remotely by the experts (Trainer).

Test Catalog:

The test catalog is the module based on the course catalog. After completing the training, the Trainees are allowed to demonstrate their problem solving skills by taking the test. The testing includes assessing the output map, and choosing the best response cleanup method based on the available equipment and the weather conditions. The emphasis is placed on the trainee's ability to compile multiple sources of information and find an effective response.

The test dataset are securely stored in a MySQL database server – MySQL has the capability to load any spatial vector data format such as shapefile. Fig. 26 shows the geometry objects support in MySQL.

Both the Trainer and the Administrator are assigned to a department in the Ice-bridge Simulator Centre. This department has several computers managed by the Administrator.

3.1.4 Robustness Diagram – Trainees Point of View

A robustness diagram originally developed by Ivar Jacobsen is a dynamic diagram that links the use cases together with the domain model and user interfaces.

The main objective of building a logical (not forcibly sequential) robustness diagram is to ensure the validity, consistency and strength of the use cases model analysis. It illustrates the logical flow of behaviour and interaction between the user and the system.

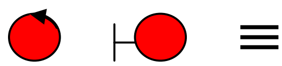
Figure 19: Robustness Symbols

Actors: Actors' symbol: Trainer; Trainees; Administrator; Stakeholder.

Boundary objects: these are screens used by actors to communicate with the system.

Entity objects: these are entity objects from the conceptual domain model.

Controllers: these are class functions controlling the interaction between the boundaries and entities.

 Boundary object and controller in failure mode, respectively

Robustness Communication Rules:

1. Actors are only entitled to interact with boundary objects.
2. Boundaries are only entitled to interact with controllers and actors.
3. Entities are only entitled to interact with controllers.
4. Controllers are only entitled to interact with themselves, boundaries and entities.

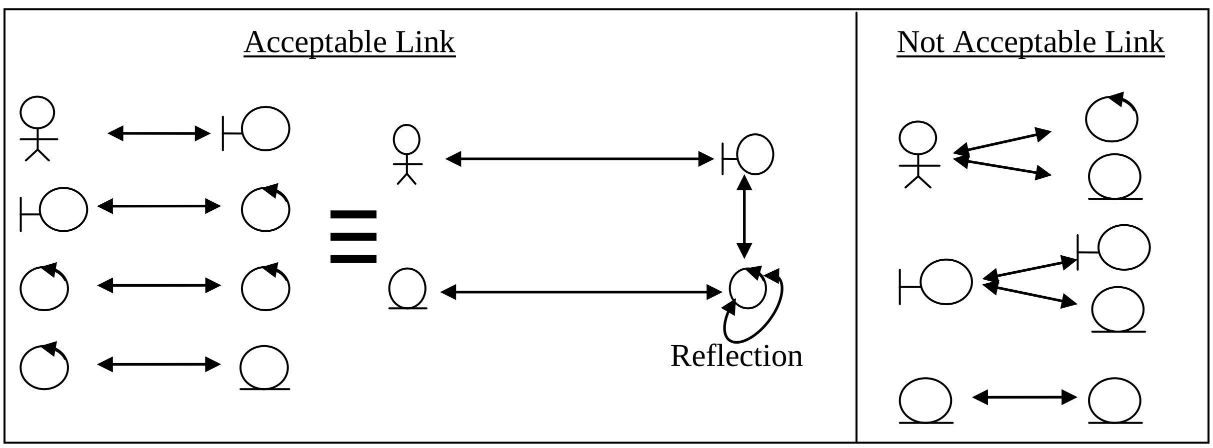


Figure 20: Communication Rules for Robustness Diagram, [11]

Login / Logout from System

BASIC COURSE:
The Trainee startup the system and the system displays the access control window. The Trainee enters its user credentials (password) into the system.
The system controller checks the validity of the persistent account in the account list.
If the Trainee exists, the system then checks the password.
If the login password is ok, then the system retrieve the account information and display the welcome window back to the Trainee.

ALTERNATE COURSES:
Trainee forgot his password:
When the Trainee enters wrong credentials the system does not recognized him and an error message is displayed back to the login screen. The Trainee can then try again.

Trainee cannot remember his password or credentials:
The Trainee cannot remember his account information.

The Trainee can then invoke assistance from the Trainer (administrator).

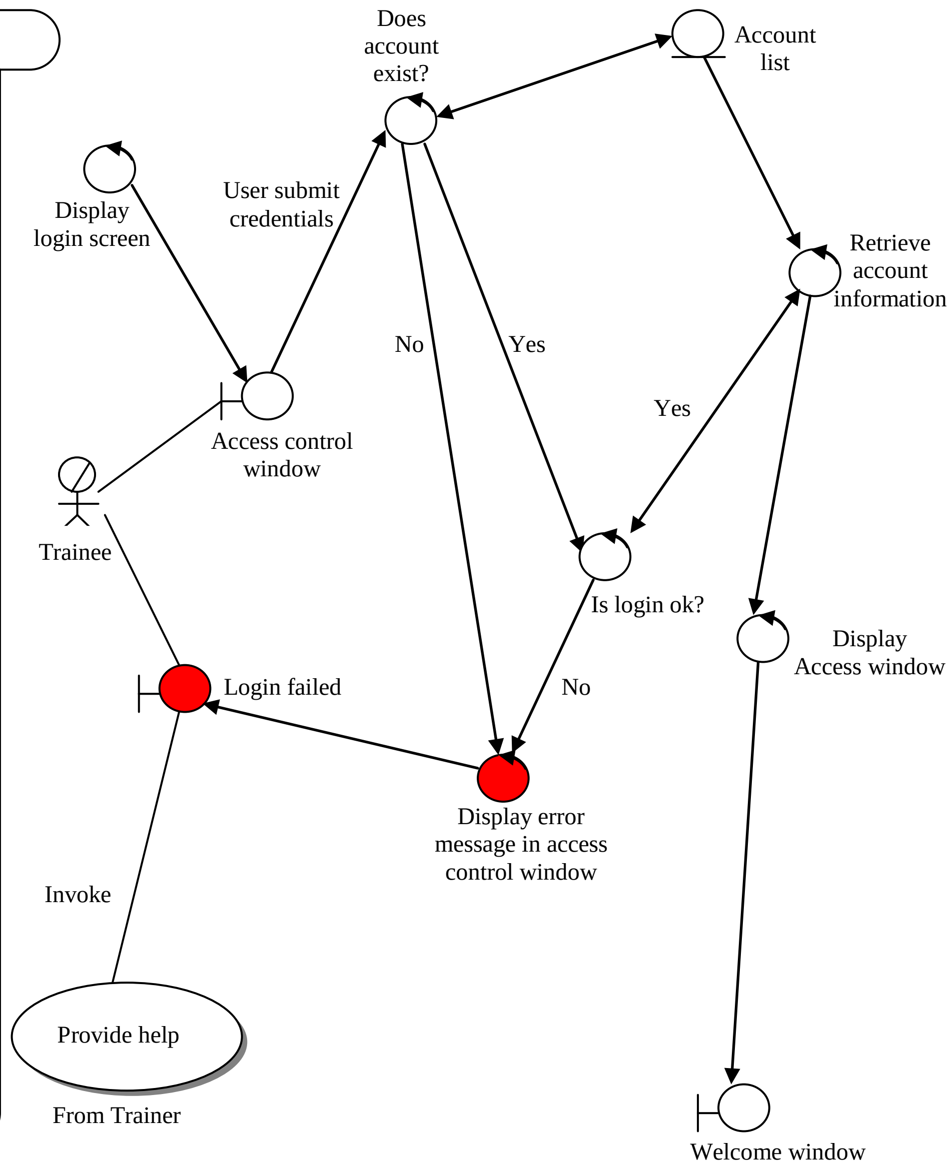


Figure 21: Trainee - Login / Logout from System

Retrieving Scenarios Data from Database

BASIC COURSE:
The Trainee startup the GIS application system and the system displays an empty screen window. The Trainee enters the database to retrieve the scenario. The system controller checks the validity of the persistent account and retrieves the scenario list window. The Trainee cancels the selection and returns to the empty GIS screen or proceeds in the selection and click ok. The system controller checks the selection validity. The controller checks whether the selection makes a full scenario or not. When the selection for a scenario is completed, the controller displays the scenario on the GIS screen window.

ALTERNATE COURSES:
<u>Trainee forgot to make selection before clicking ok</u>:
When the Trainee clicks ok without making selection, the controller returns an empty GIS screen to him.

<u>Trainee cannot view the list of scenarios</u>:
When the scenario list is empty, the Trainee cannot retrieve any data from the database. The controller returns an empty GIS screen as a consequence.
The Trainee can thus request help from the Trainer.

<u>Trainee retrieve incomplete vector layer for a scenario</u>:
When the Trainee does not select enough vector layers which constitute a full scenario from the selection list, the controller validate the choice of the Trainee and displays an incomplete scenario on the GIS screen.

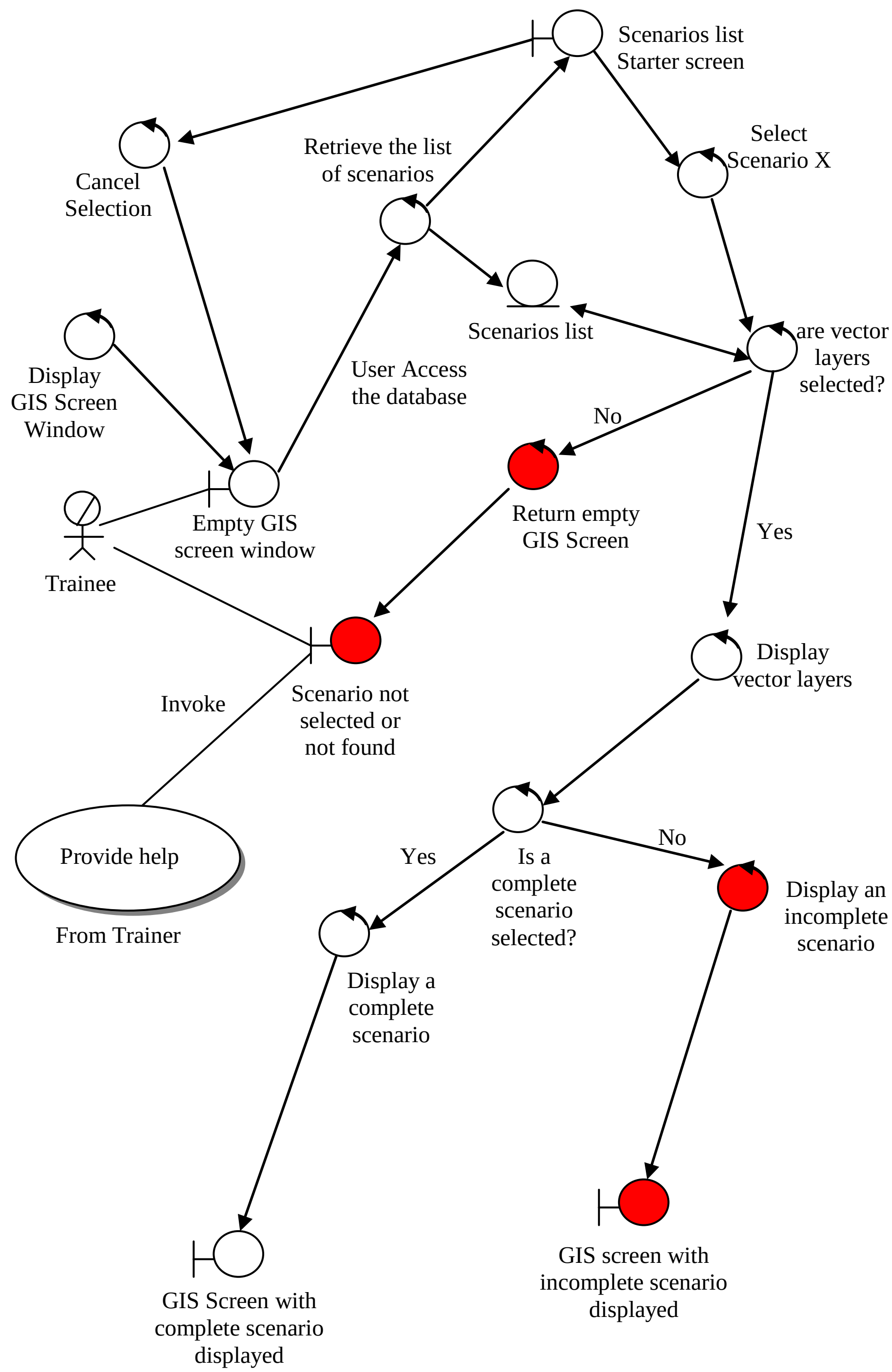


Figure 22: Trainee Retrieving Scenarios from Database

Performing the training

BASIC COURSE:
The Trainee load the scenario X in the GIS application. He tries to retrieve the information X corresponding to the scenario X. the system open the information list window. The Trainee selects the information X and read its content.
As another basis course, the Trainee retrieve the vector layer properties to look for further information attached to the layer. The controller retrieves the symbology and classifies the layer. A GIS main screen is presented with vector layer classified. The Trainee can also cancel the process at any time and return back to the GIS main window.

ALTERNATE COURSES:
<u>Trainee cannot view the information X in the list:</u>
When the information X for scenario X is not available, the Trainee can return back to the GIS main window. The controller closes the window and presents a GIS screen window with scenario X.

<u>Trainee cannot find the layer properties</u>:
When the Trainee wants to view the layer properties, the controller checks for those properties stored in the database and attached to the layers. When no layer property is found, the Trainee then is allowed to view the GIS screen main window with scenario X but without classified layers.

The Trainee can then invoke assistance from the Trainer (administrator).

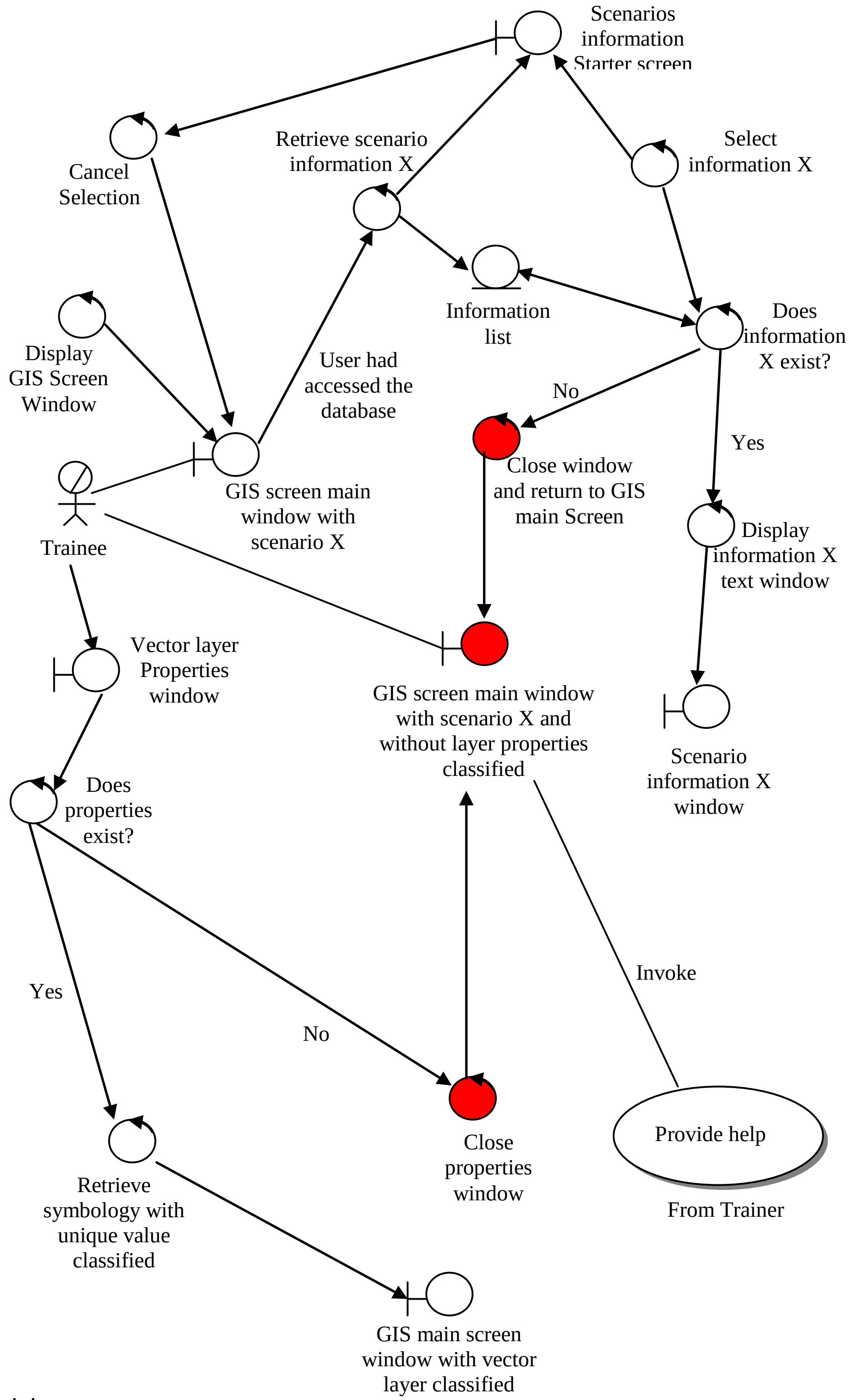


Figure 23: Trainee Performing the Training

Performing and Submitting the Test

BASIC COURSE:
The Trainee requests the scenario test information as a primary step. The controller verifies whether the related information exists in the database. On confirmation, the information is retrieved and displayed to the Trainee.
The Trainee performs the test by editing the object layer and exports the result in a WGS84 geographic coordinate system.
The Trainee can then submit his/her work by uploading the file to the database.
The controller checks the insert command syntaxes with necessary privileges. On confirmation, the controller displays a message stating that the request has been successful.

ALTERNATE COURSES:
Trainee cannot view the test information in the list:
When the test information for a scenario is not available, the Trainee can return back to the GIS main window. The controller closes the window and presents a GIS screen window with the scenario. at this moment, the Trainee does not know what to do.

Trainee cannot submit (upload) his/her work:
The Trainee cannot submit his/her work due to syntax errors in the commando, or the Trainee lacks the necessary privileges to perform such operations.

The Trainee can then invoke assistance from the Trainer (administrator).

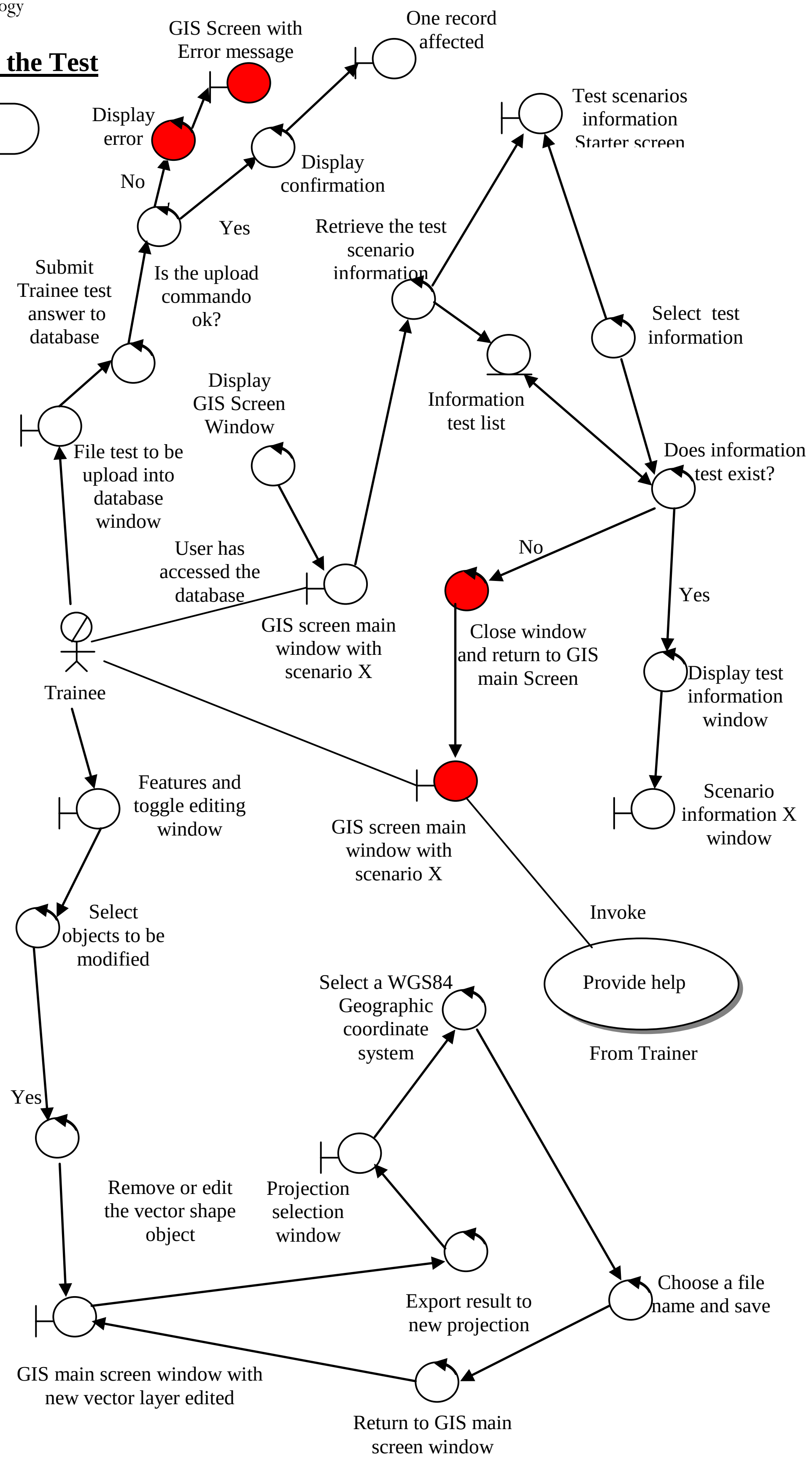


Figure 24: Trainee Performing

Submitting the feedback and View Test Result Report

BASIC COURSE:
The Trainee opens the MySQL application command line and access the database with credentials.
The system validates the access account and displays the confirmation window.
The Trainee writes the feedback and loads it into the database.
The system checks the feedback format and displays a confirmation window, stating that the feedback has been stored in the database.
The Trainee may also view the result of the submitted test using the server command line. The controller then checks whether the Trainee has submitted his/her test. On positive confirmation, the controller retrieves the test result report.

ALTERNATE COURSES:
Trainee sends a wrong feedback format:
When the Trainee sends a feedback with an unrecognized format, the system displays an error message window to the Trainee. The Trainee can then try again or request help from the Trainer.

Trainee cannot view the test result report:
The Trainee is unable to access his/her test result because he/she has not submitted the corresponding test. Or the Trainer has not yet made the test result available for retrieval.

The Trainee can thus request assistance from the Trainer.

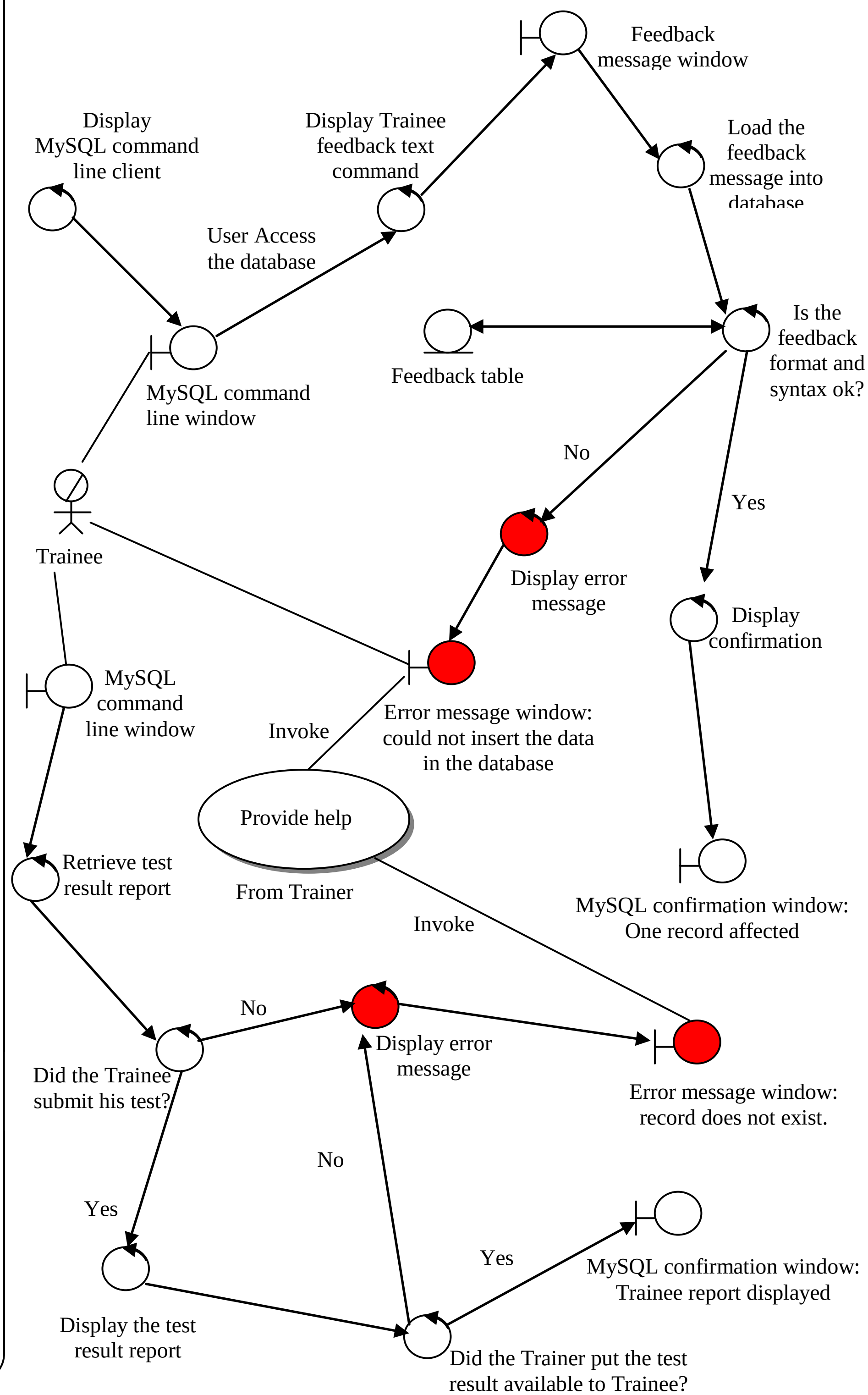


Figure 25: Trainee Submitting Feedbacks and Viewing Test Results

3.2 MySQL Database Server Architecture and Behavior

MySQL is the database server used in this thesis. MySQL is a multithreaded database management system. Many features are stored in and it can operates on objects containing geometric and geospatial data, Fig. 26.

The integration of GIS in MySQL is still under development. Currently, MySQL does not support all features offered in OpenGIS. For example, MySQL only supports 2D vector data. Fortunately, MySQL's GIS functions are largely sufficient for carrying out various tasks, including those to be implemented (simulations of oil spill response training scenarios) in this thesis.

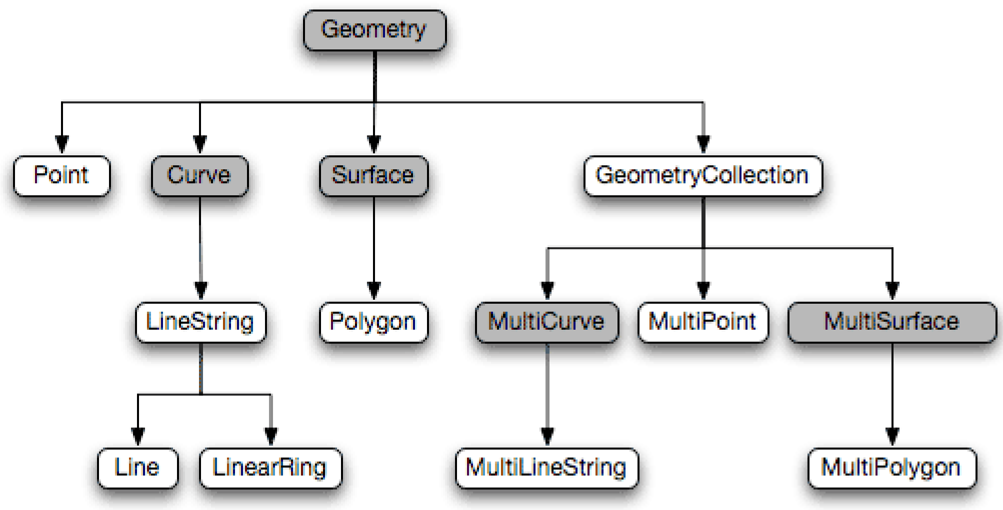


Figure 26: Geometric Objects Support in MySQL, [14]

MySQL Server Logical Architecture:

The MySQL is a high level architecture relational database management system. The conceptual architecture model reposes on a 3- tier layer, Fig. 27:

1. The ***Application layer***,
2. The ***logical layer***,
3. The ***physical layer***

Application Layer:

The application layer is the computer interface where users interact with the system. It is responsible for the look and feel of the application. It corresponds to boundary objects.

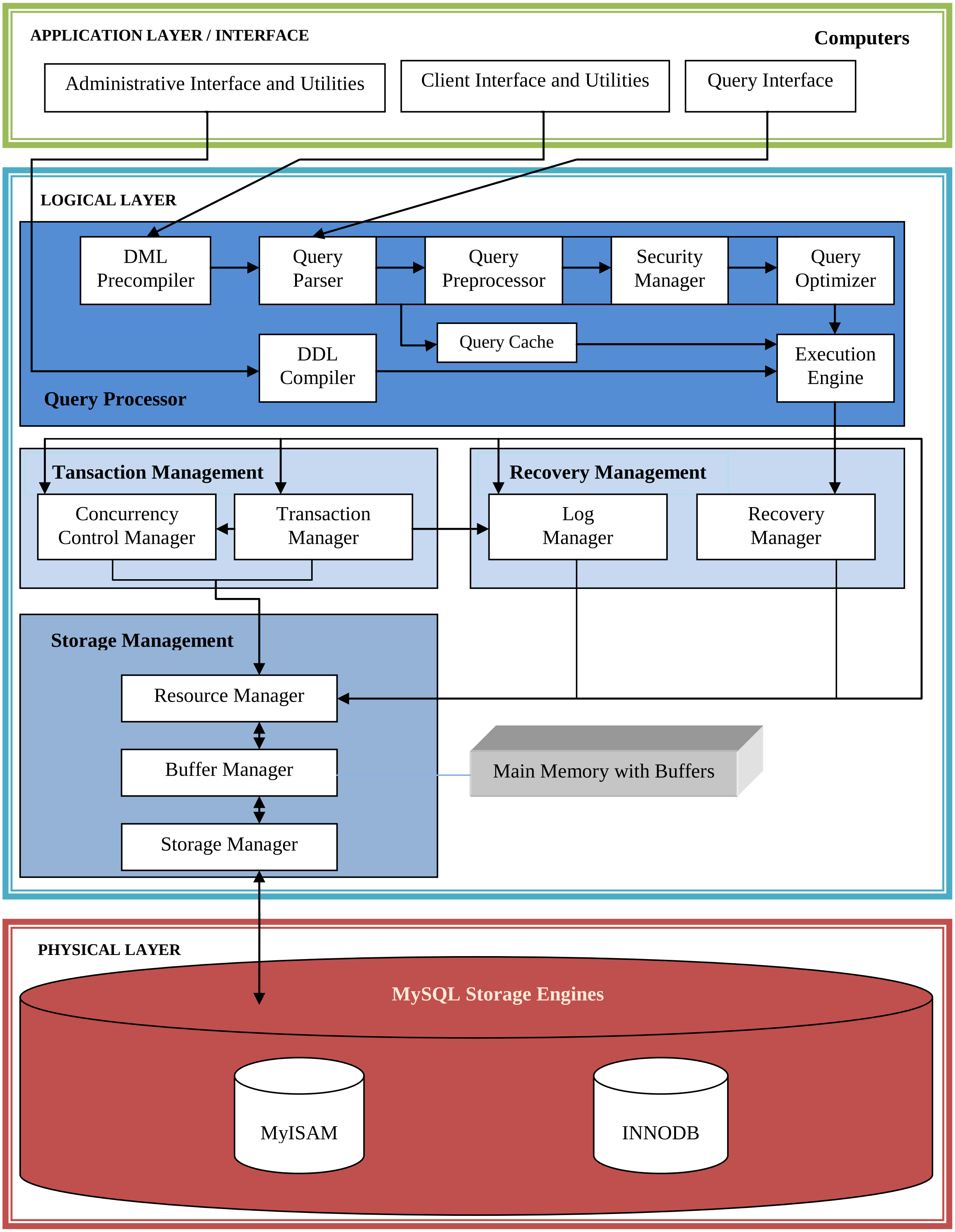


Figure 27: 3-Tier MySQL System Overview, [17]

Logical Layer:

The logical layer is the central control system. All requests from users of the system must be processed in the logical layer. It is composed of four different elements: query processor, transaction management, recovery management and storage management. It corresponds to controllers.

Physical Layer:

There are several types of MySQL storage engines in the physical layer and each has its strengths and weaknesses. The choice of an appropriate storage engine depends largely on the type of operations to implement, the type of data to store, if the data will be constantly changed or logged, whether there will be multiple statements transaction control, and data replication.

The implementation of the application in this thesis requires an active use of MyISAM storage engine - MyISAM is an engine favourable for data replication.

3.2.1 Distributed Data Replication Topology

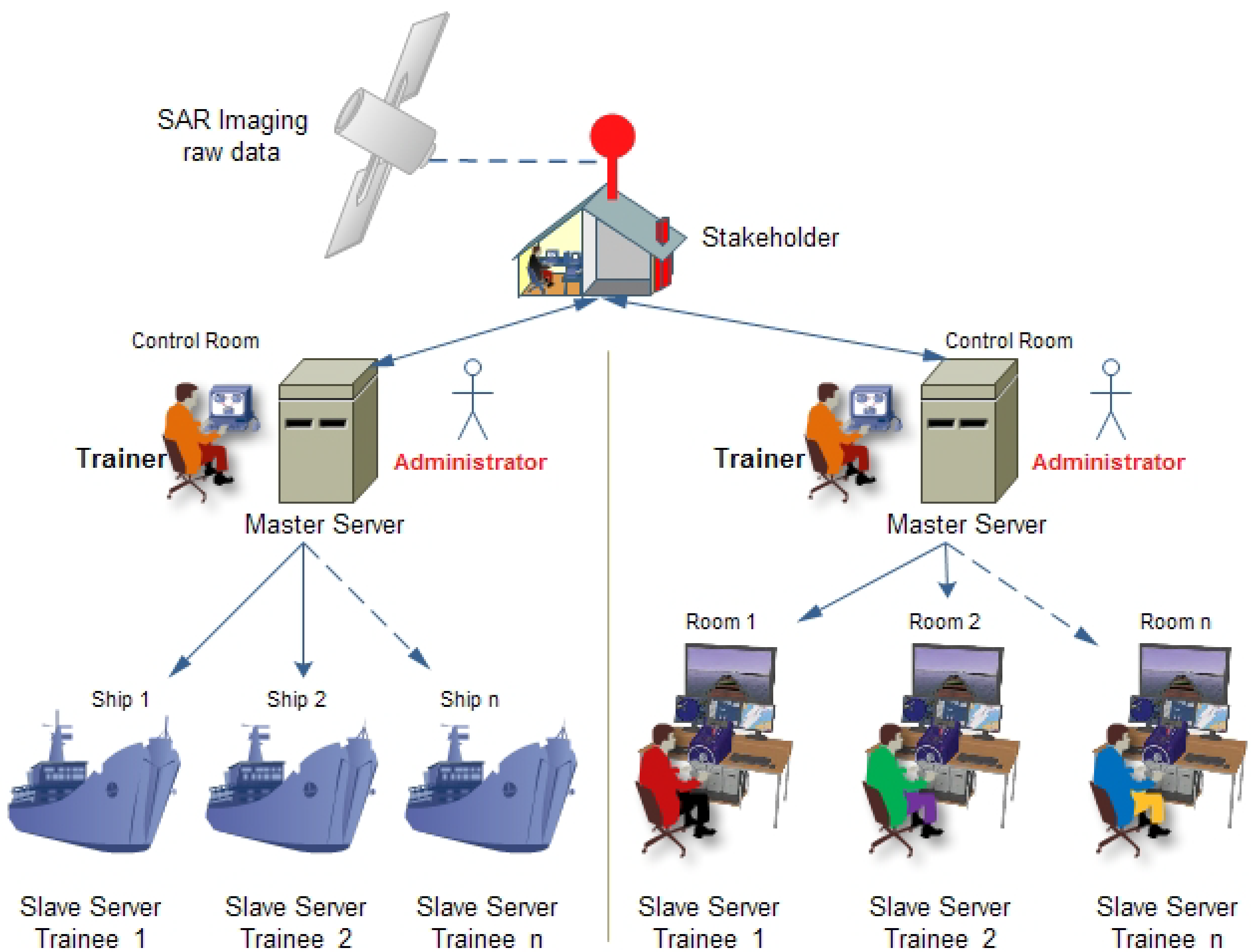


Figure 28: Master - Multi Slaves Replication in Ship Bridge Simulator

Replication is a database technology used to transfer data from one server to another server. The replication topology used in this thesis is a one Master to many Slaves connection, Fig. 28.

The Trainer is the client on the Master server and the Trainees are clients on Slave servers. The Trainer is assigned database administrative role while the Trainees only have limited database privileges. As can be seen from the figure, the interaction between different Slaves is prohibited. One Slave can only have one Master, but a Master can have many Slaves. A Slave must be promoted as Master to be able to communicate with other Slaves.

For many reasons, this types of replication configuration fit very well with the oil spill response training and test scenarios.

The Slaves are placed in their respective training's rooms and without interacting with each other. They are only connected to the Master. Moreover, the simulator training configuration supports more reading and few writing from the database: There are many Slaves all are mainly engaged in reading the data sent by the master; On the other hand, there is a Master who manages the data (read / write) in the database.

In a situation where a large number of slaves are required to participate at the same time, the master server can become overloaded or too heavy. This can lead to latency. The consequence of this state is a high speed parachute of the application performance and availability (the slaves are heavily dependent on the Master during training). This event is unacceptable in terms of training point of view. The probability of such scenario can be minimized by using a *relay server* to ease the burden of replication on the master server. The topology is shown in Fig. 29.

It is important to note that the master can be located on a mobile base station at sea or onshore. All servers should not necessarily be located in the same geographical area or in close proximity

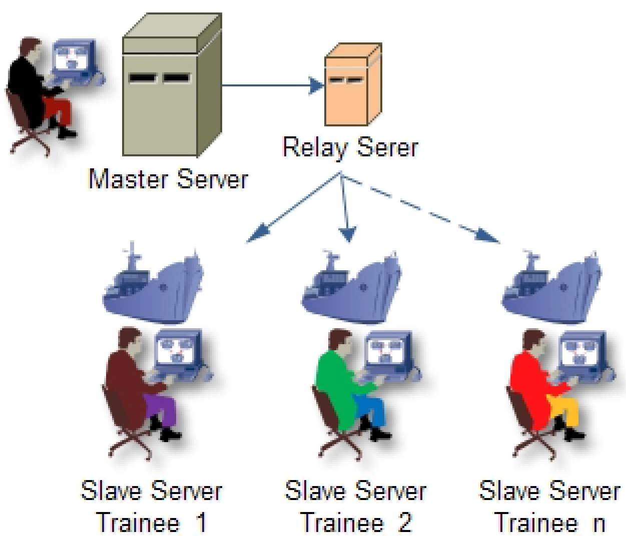


Figure 29: Master, Relay and Multiple Slaves Replication

in order to carry out the training. The only two necessary requirements are to make sure that the Internet Protocol is enabled on each server and that each computer has an interface that fully support the functionality required to conduct the training.

This technology is robust and very promising to terms of improving the oil spill emergency response.

It is believed that one of the most important tasks in dealing with oil spill is the ability to monitor the cleanup response (not only the spill itself). For example, one can follow the pattern of data generated from the spill site. The pattern here refers to:

- The positioning and motion path of the equipment being used during the oil spill response
- The positioning and weathering of the oil, and ice

When these data are retrieved from a base station, an oil spill response analysis can be performed with greater accuracy. The expert system can use these types of recorded data to give recommendations while the response cleanup is still in progress. For example, a set of priorities can be established on a segment area; the next cleanup starting or abandon point can be decided, etc.

Chapter 4

Relevant Simulator-Based Training Courses

4.1 Testing the Ice-bridge Simulator Training Application

This application is examined throughout the previous chapters and is ideally designed, Fig. 28. The main steps that are considered during the implementation of the application are listed as:

a. Establishing the Master and Slaves servers connection
b. Trainees performing the training – case scenario
c. Trainees taking the test examination – case scenario
d. Trainees submitting the test answer
e. Trainer retrieving the Trainees' data and submitting the corrected examination
f. Trainees submitting their feedbacks or requests.

a. ESTABLISHING THE MASTER AND SLAVE SERVERS CONNECTION

The Figure 30 shows a high level of reads and writes processes between a master and a slave server. The basic but fundamental communication scenarios are listed below:

1. The Master and the Slave server are initialized (connected).

2. The Master sends the data to the Slave by recording the data in its binary log events

3. The Slave read the Master's binary log by using its I/O thread (constantly checking the binary log status).

4. If the I/O thread observes changes, it writes the events into the Slave relay log.

5. The SQL thread then reads and executes the events from the relay log. From this step, the Slave is à jour with the Master.

Another important point to mention is that the Master is also able to retrieve and read all data from the Slave database. For example, when a Trainee wants to submit a feedback or a test, he/she will only need to store the data in his/her own computer database.

The Trainer will automatically retrieve those data from his/her own Master server. The Trainer will then plot the Trainee's data, analyse them and possibly make corrections or recommendations accordingly. To deliver the test correction, the Trainer will simply need to dump the corrected test results into each Trainee's computer database.

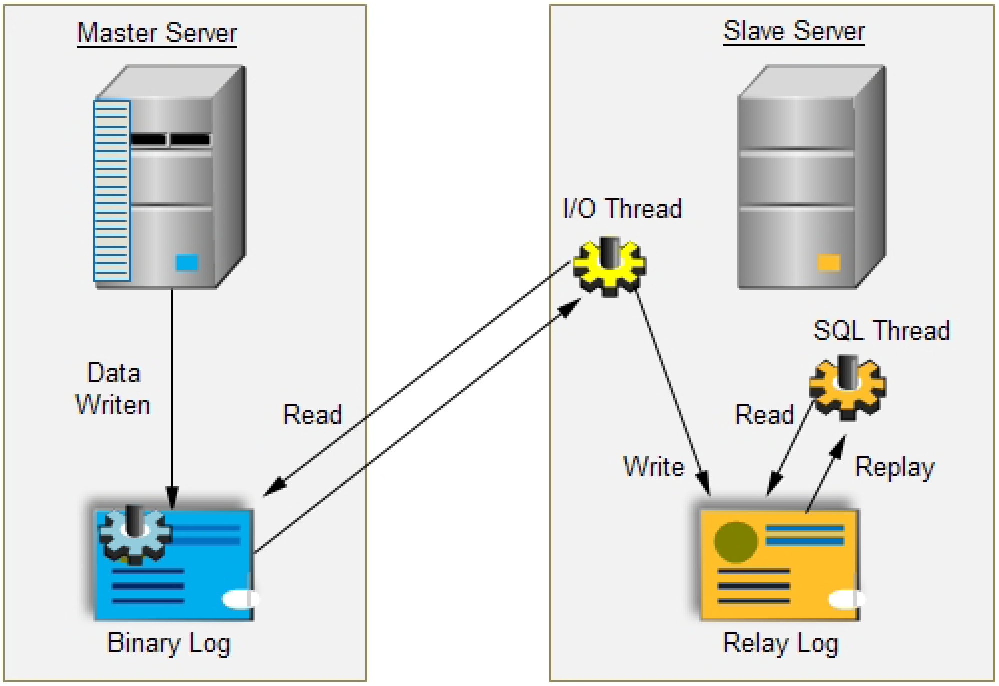


Figure 30: High Levels Read / Write Data Replication Process, [16]

Refer to App. C for the step-by-step technical setup of Master / Slave server connection and procedures for the Training and the Testing scenarios.

Output Result of the Master Status while in Connection with a Slave:

```
mysql> show master status;
+------------------------+----------+--------------------+------------------------+
| File                   | Position | Binlog_Do_DB | Binlog_Ignore_DB |
+------------------------+----------+--------------------+------------------------+
| mysql-bin.000015 |     98     | IBSTraining      |                              |
+------------------------+----------+--------------------+------------------------+
```

Output Result of a Slave Status while in Connection with the Master:

```
mysql> show slave status\G
************************** 1. row **************************
          Slave_IO_State          : Waiting for master to send event
            Master_Host           : 78.91.65.28
            Master_User           : Trainee1
            Master_Port           : 3306
          Connect_Retry           : 60
         Master_Log_File          : mysql-bin.000015
      Read_Master_Log_Pos         : 98
          Relay_Log_File          : mysql-relay-bin.000002
          Relay_Log_Pos           : 235
    Relay_Master_Log_File         : mysql-bin.000015
        Slave_IO_Running          : Yes
       Slave_SQL_Running          : Yes
        Replicate_Do_DB           : IBSTraining
```

```
Replicate_Ignore_DB          :
Replicate_Do_Table           :
Replicate_Ignore_Table       :
Replicate_Wild_Do_Table      :
Replicate_Wild_Ignore_Table  :
Last_Errno                   : 0
Last_Error                   :
Skip_Counter                 : 0
Exec_Master_Log_Pos          : 98
Relay_Log_Space              : 235
Until_Condition              : None
Until_Log_File               :
Until_Log_Pos                : 0
Master_SSL_Allowed           : No
Master_SSL_CA_File           :
Master_SSL_CA_Path           :
Master_SSL_Cert              :
Master_SSL_Cipher            :
Master_SSL_Key               :
Seconds_Behind_Master        : 0
```

Output Result of the Master Processlist while in Connection with a Slave:

```
mysql> show processlist\G;
*************************** 1. row ***************************
Id        : 1
User      : Trainee1
Host      : dhcp-065142.wlan.ntnu.no:49166
Db        : NULL
Command   : Binlog Dump
Time      : 191
State     : Has sent all binlog to slave; waiting for binlog to be updated
Info      : NULL
```

Output Result of a Slave Processlist while in Connection with the Master:

```
mysql> show processlist\G;
*************************** 2. row ***************************
Id        : 2
User      : system user
Host      :
Db        : NULL
Command   : Connect
Time      : 114
State     : Waiting for master to send event
Info      : NULL
*************************** 3. row ***************************
Id        : 3
User      : system user
Host      :
Db        : NULL
Command   : Connect
Time      : 113
State     : Has read all relay log; waiting for the slave I/O thread to update it
Info      : NULL
```

Table 6: EXPLANATION OF SLAVE STATUS OUTPUT

File Name	Meaning
Slave_IO_State	Status of the Slave I/O thread
Master_Host	IP address of the Master
Master_User	The User on the Slave server connected to the Master
Master_Port	Connection channel used by the I/O thread
Connect_Retry	In case of disconnection, the I/O thread will retry the given timeout parameter
Master_Log_File	The Master binary log file read by the I/O thread and executed by SQL thread
Read_Master_Log_Pos	The position where the Slave I/O thread is currently on
Relay_Log_File	The Slave log file where the SQL thread is currently on
Relay_Log_Pos	The Slave log position where the SQL thread is currently on
Relay_Master_Log_File	The Slave log file where the I/O thread has written data for the SQL thread
Slave_IO_Running	Showing whether the I/O thread is running or not
Slave_SQL_Running	Showing whether the SQL thread is running or not
Replicate_Do_DB	The name of the database in the replication process
Replicate_Ignore_DB	The name of the database ignored in the replication process
Replicate_Do_Table	The name of the database table involved in the replication process
Replicate_Ignore_Table	The name of the database table ignored in the replication process
Replicate_Wild_Do_Table	Showing which tables are being replicated in the process
Replicate_Wild_Ignore_Table	Showing which tables are being ignored in the replication in the process
Last_Errno	The last error code that caused the replication process to stop.
Last_Error	The last error message that caused the replication process to stop.
Skip_Counter	In case of error, the number of events that the SQL thread may skip
Exec_Master_Log_Pos	The Master log position corresponding to the SQL thread executing position
Relay_Log_Space	The disk space in bytes occupied by the relay log
Until_Condition	None: if no UNTIL clause was specified; Master: if the slave is reading until a given position in the master's binary log; Relay: if the slave is reading until a given position in its relay log
Until_Log_File	Log file name defining coordinates at which the SQL thread stops executing
Until_Log_Pos	Log file position defining coordinates at which the SQL thread stops executing
Master_SSL_Allowed	Indicating whether the I/O thread should use SSL for connection with Master
Master_SSL_CA_File	Path name to the certificate authority file the Slave must use if SSL is required
Master_SSL_CA_Path	Path name to a directory containing trusted SSL CA certificates
Master_SSL_Cert	Path name to the certificates file
Master_SSL_Cipher	The SSL cipher to be used
Master_SSL_Key	The SSL key to be used
Seconds_Behind_Master	Indicating how late (in seconds) the Slave is with respect to the Master.

Verifying the status and the process lists of the connected servers is an essential step to do before going further with the transaction of data. All participating servers must be properly connected and preferably synchronized to ensure consistency between the Master and Slaver servers.

The synchronization is achieved when a thread on the Master is immediately blocked after its commit statement. It will then be in a "wait modus" until the all Slaves' acknowledgment on all events has completed. In this way, the Master takes into account the readiness of the Slave before writing again on the binary log.

b. TRAINEE PERFORMING THE TRAINING: CASE SCENARIO

When the connection between the Master and Slaves are established, the Trainees are then able to fully follow both the training course and the examination. At this stage the Trainees have been granted all the necessary database privileges to retrieve, update and upload the data into their respective database. Real-time information and data sharing is activated!

The Fig. 31 shows a data plotting result of a training scenario perform by a Trainee. From this figure the Trainee is advised on how to deal with this type of oil spill scenario. The Trainee can open each vector object filled with specific information.

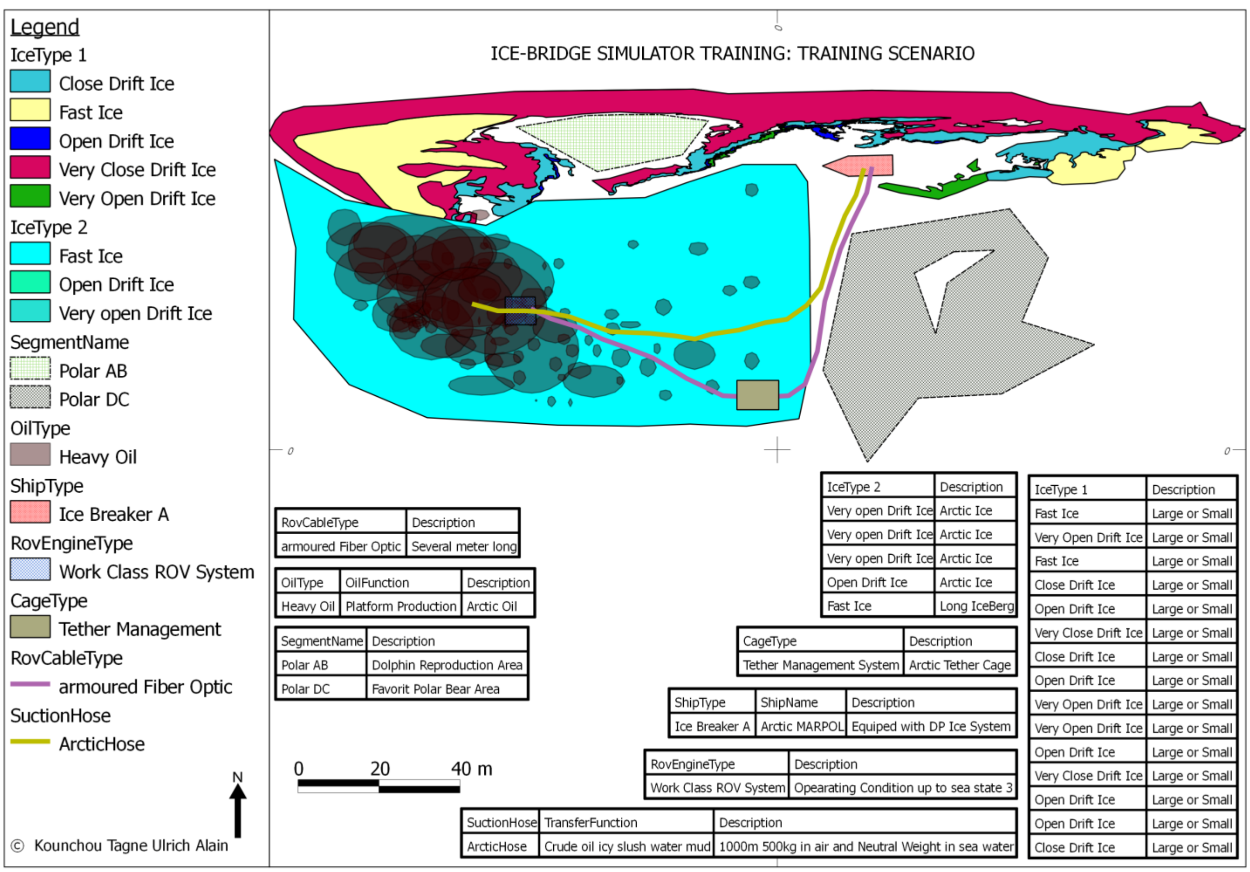


Figure 31: Plotting Result of a Training Scenario

For example, he/she can know the type of oil spill involved; know which segment areas are protected; what is protected; the type of ice in vicinity; and read the full description of the equipment that is being used (ship types, boom types, cable types, etc).

Additionally, the Trainee can open an information file specially attached to the scenario, Fig. 32.

After gaining knowledge of all objects presented in the training, the Trainee can now try different response techniques by repositioning objects such as ships, oil boom, or any relevant equipment available for oil spill cleanup.

These kinds of operation are performed within a GIS tool. In this thesis, the QuantumGIS is used.

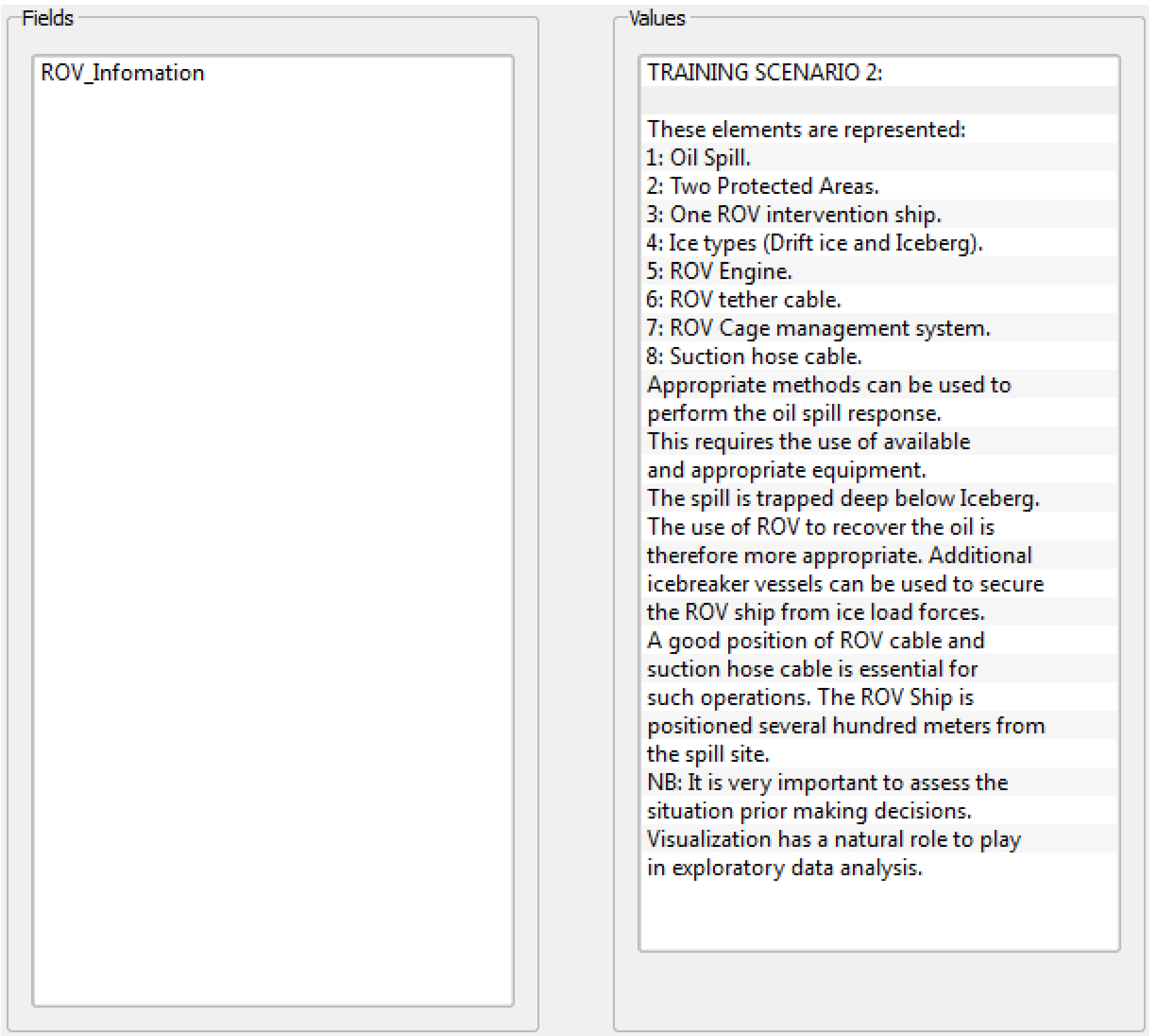


Figure 32: Training Course: Example of Information Attached to Scenario

There are in total six different Training scenarios and for each scenario the Trainee follows the same procedure, but each scenario has a content which is specific.

Other type of Training scenarios (number: 1, 2, 3, 4, 5, 6) are presented in App. A.

After completing all the scenarios, the Trainees are allowed to take an examination.

c. TRAINEE TAKING THE TEST EXAMINATION: CASE SCENARIO

During the Test, the Trainees are only allowed to utilize the available equipment as displayed at the bottom of the Fig. 33.

They still have the privileges to retrieve all information attached to each vector object, but no information related to which types of response methods they should go for is being provided. In addition, they are not allowed to modify the objects such as “protected segment area”, “Iceberg”, “fast ice”.

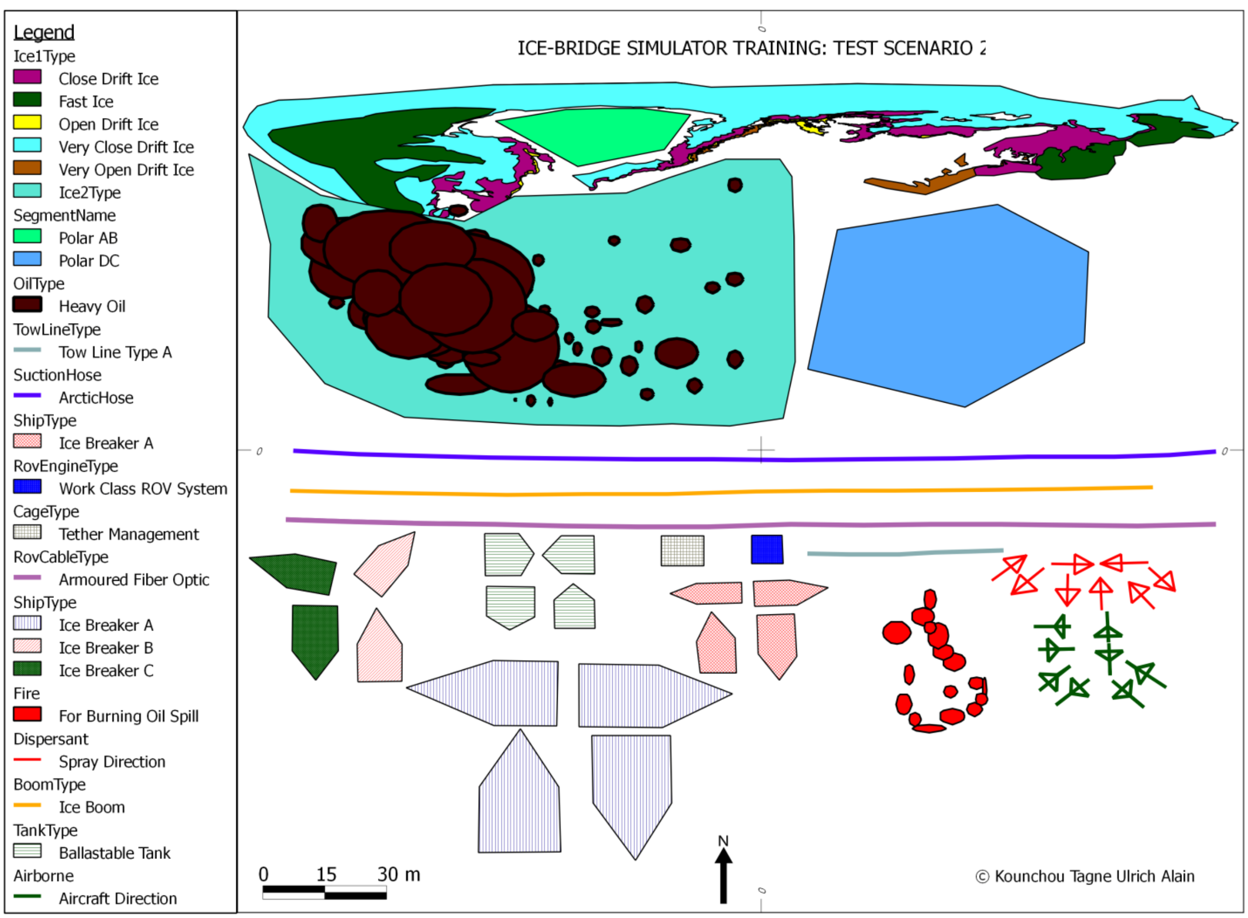


Figure 33: Plotting Result: Trainee taking a Test Scenario- at Sart

d. TRAINEE SUBMITTING THE TEST ANSWER

When the Trainees have finished the test, the next step is to load the result directly to their MySQL database. This is an easiest and fastest way of submitting. Alternatively, they can save their result as shapefile in local drive (for example in C:/ directory) and then upload their results to their own respective database server. Trainees can be forced to follow a special filename to deliver their answer.

The filename used in this thesis for submitting the test is as follow: ***Trainee1_Svar_Scenario_X_Tab***; Meaning that the user (Trainee1) has submitted his /her test answers for scenario number X. Following this arrangement the Trainer will easily classify the results considering the number of scenarios and the number of trainees.

The following command is used by the Trainees to perform the database uploading operation:

```
ogr2ogr -f "MySQL" MySQL:"IBSTraining, user=Trainee1, host= Slave_Host, password=Slave_Password,
port=3306" -nln "Trainee1_Svar_Scenario_X_Tab" -a_srs "EPSG:4326" -lco engine=MYISAM
C:/Trainee1_Svar_Scenario_X_Tab.shp
```

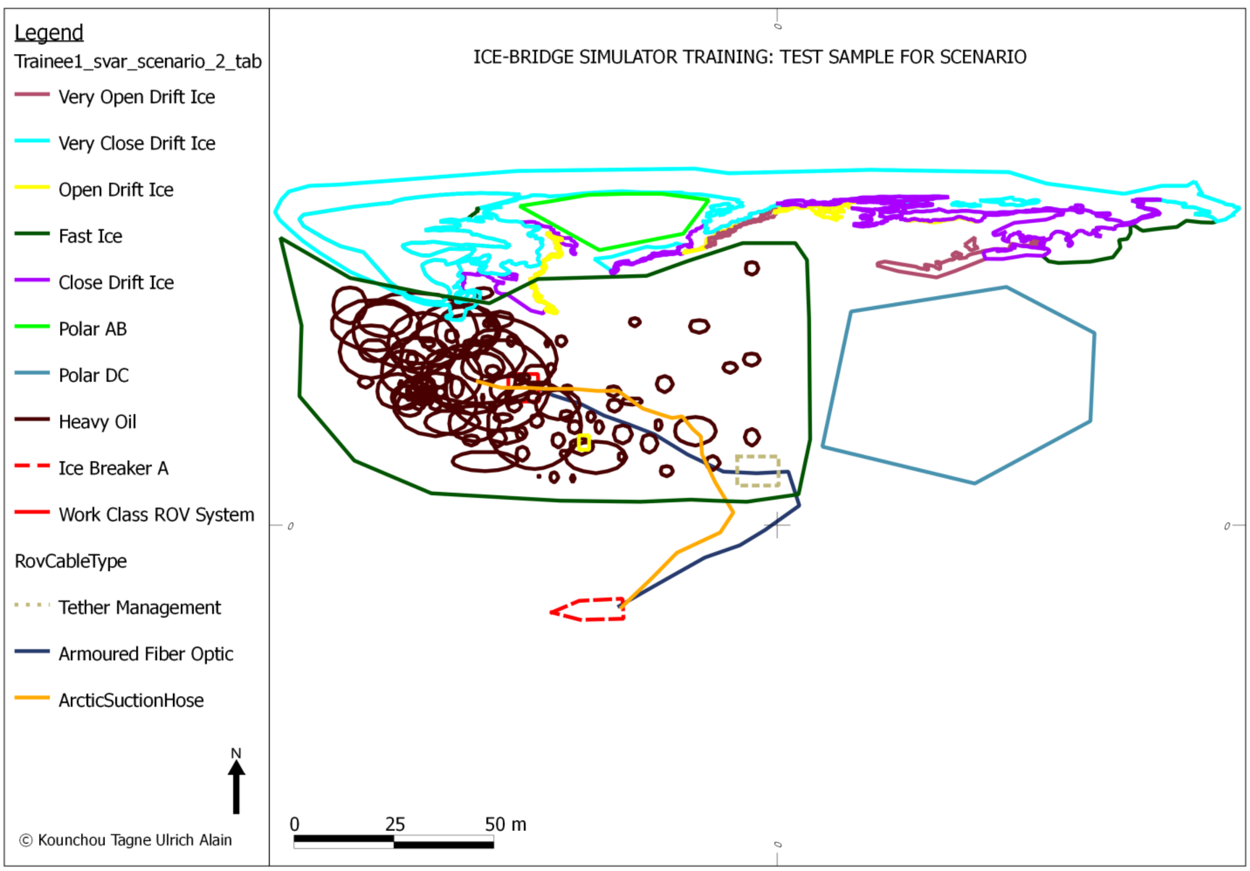


Figure 34: Plotting Result: Trainee submitting his Result - at End

e. TRAINER RETRIEVING TRAINEES’ DATA AND SUBMITTING THE CORRECTED EXAMINATION

As stated before, the Trainees do not have to send their results to the Trainer. All they need to do is to save their final work in their own computer database. Because the Master computer controls all slave computers, the Trainer will be able to retrieve the Trainees’ results.

NB: The Master is a central unit. It is therefore preferable that the Slaves have only read access and zero write access to it. On the other hand, the Master has both reads and writes access to all Slaves’ database servers.

Trainer Action: Retrieving and Storing the Trainee’s Data in a Master Computer Directory:

```
mysqldump -u Trainee1 -p IBSTraining Trainee1_Svar_Scenario_X_Tab --host=Slave_Host  >
Trainee1_Svar_Scenario_X_Tab.sql
```

Trainer Action: Loading the Trainee’s Data into Master Database:

```
mysql -u Master -p IBSTraining -h Slave_Host  < Trainee1_Svar_Scenario_X_Tab.sql
```

“ Trainee1_Svar_Scenario_X_Tab.sql ” is the file containing all the test answer data for the Trainee1

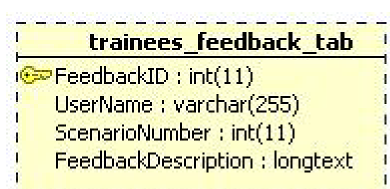
The Trainer can then analyse the submitted work, and make corrections or recommendations. After correction, the result is then dumped to the Trainee’s database in the following filename arrangement: ***Trainee1_Svar_Scenario_1_Corrected_Tab***.

f. TRAINEES SUBMITTING THEIR FEEDBACKS OR REQUEST

During the oil spill emergency response the Trainees may promptly request additional equipment or any information necessary to improve the cleanup operations. Also he / she can be asked to give an update status of the spill recovery or simply give a personal opinion (feedbacks) about the quality and the effectiveness of the overall spill response application.

Feedback Table:

```
CREATE TABLE Trainees_Feedback_Tab
(
FeedbackID INT NOT NULL AUTO_INCREMENT,
UserName VARCHAR(255) NOT NULL,
ScenarioNumber INT NOT NULL,
FeedbackDescription LONGTEXT NOT NULL,
PRIMARY KEY(FeedbackID)
) Engine = MyISAM;
```



To send their feedbacks they will only need to store their data in their own databases

Simplified Algorithms for Sending Scenarios during Training and Test

During the training scenarios are sent one after the other and each scenario have a limited duration, Fig. 35. The duration of a scenario may vary depending on its complexity.

During the test the Trainer sends each test scenario to Trainees. Once the Trainees have completed a scenario, the Trainer immediately deletes the scenario on the Slave side by running a truncation command – all table records on the Slave side are flushed (emptied). Alternatively, knowing the filename (example: *Trainee1_Svar_Scenario_1_Tab*) of all scenario tables, the Trainer can simply block the Trainees' tables from database entrance.

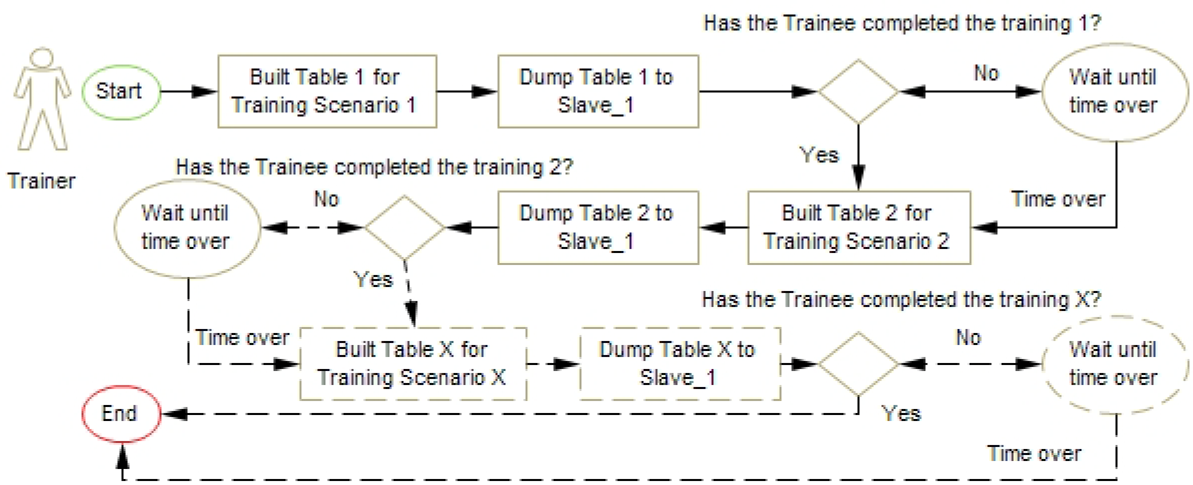


Figure 35: Sending Training scenarios

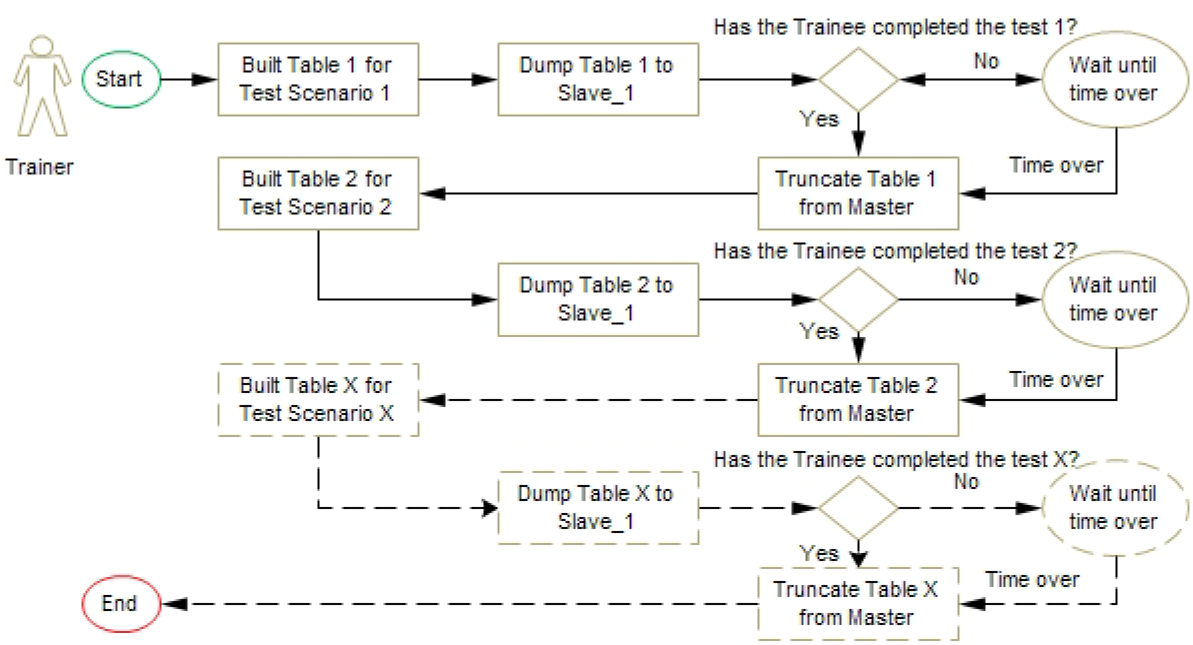


Figure 36: Sending Test Scenarios

Chapter 5

Evaluating the Validity of the Application

5.1 Evaluating the Validity of the Application

To evaluate the validity of the Ice-Bridge simulator training application, it is imperative to consider the user of the system and the system itself:

a) The system technical performance: Availability and Scalability
b) The system operational value: Usability and Achievement

a.1) APPLICATION TECHNICAL PERFORMANCE - AVAILABILITY:

The application that is being proposed requires high availability all the time and good scalability over time. Application's high availability simply means that when it is in use, it keeps on running smoothly all the time without (or with negligible) disruptions. Of course there is no complex system which does not experience downtime or failure at some point. Nevertheless, the application needs to automatically recover from failure mode without losing the data. Should a failure occurs, adequate procedures can be put in place to quickly return to its normal state.

Basically, the IBST application might experience Master failure - loosing connection with Slaves; Slaves failure - loosing connection with the Master; The binary and relay log files might be corrupt; the Slave might read on a wrong position after crash or downtime recovery; data drift between the Master and Slave due to long time setup and write access from Slave server. When these types of failures scenarios occur the application becomes poorly manageable and the data received by different Trainees become inconsistent.

From Fig. 15, it was shown that one of the main tasks of the system administrator is to maintain the system so that the training program does not experience any malfunctions. To secure the application, regular scheduled backup of the master server is taken. The backup is done by creating a consistent snapshot using *mysqldump* command.

When the first data (first training scenario) are distributed to slaves, the administrator can then take a snapshot without blocking the Trainees from running their scenarios. Taking a snapshot from the Master server includes flushing all tables with read lock in order to avoid any concurrent update during the operation.

```
-- MySQL dump 10.10
--
-- Host: 78.91.65.28    Database: IBSTraining
-- ------------------------------------------------------
-- Server version     5.0.18-nt-log

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Position to start replication or point-in-time recovery from
--

CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin.000015', MASTER_LOG_POS=98;
```

Figure 37: Extract from the Master Database *MYSQLDUMP* File

The output of the Master "filename" and its "position" in the "Show Master Status" as shown in paragraph 4.1 must match with the point-in-time Master log file and position, Fig. 37. When the snapshot has been successfully taken, all tables must be released from read lock.

Another possibility to deal with a Master failure is to simply promote another server in standby as a new Master and redirect all Slaves to it. The good news is that, the Slaves will not check the database compatibility on the new Master in order to take on. They will simply start reading and executing the on the new Master's binary log filename and position.

The immediate consequence of Slaves connected to a failed Master is that all data from the Slaves will be stalled. Some Slaves' I/O thread queries may not be executed at all, especially those awaiting the Master.

In case of Slave failure, the proper procedure to follow is to check its status along with the Master status. This is often followed by resetting the Slave; Stopping the Slave; re-fetching the new Master log filename and position; and restarting the Slave. If this procedure is good enough to keep the failed Slave up and running again, this Slave will immediately catch-up with the new Master updated status.

From the training point of view, if for example the Master status changes in the time of transition from scenario 1 to scenario 2 and the Slave failed during the scenario 1, this will simply mean that the Trainee on the failed Slave server will miss his scenario 1 and catch-up with scenario 2 at recovery time. To avoid this, the Slave has to be setup from the Master's binary log coordinates (filename and position) corresponding to scenario 1.

Usually, problems are related to Master / Slave synchronization issues. The best running state of the application is when the data on the Slaves side are consistent with those of the Master. Troubleshooting inconsistencies is often base on finding a correct Master's coordinates input for the Slaves.

Another issue as mentioned previously is the data drift - poor synchronization between the Master and the Slave. This drift often occur when the application has been running for quite long time and especially when the Slaves have been granted too many write privileges on their own server. The drift can be checked by running the following command from the Master server:

```
mk-table-checksum –replicate=IBSTraining.checksum Master_IP
```

The command will simply provide information on which tables that are out of sync between the Master and the Slave database servers. If a data drift is found, it will indicate the number of fewer tables that are out of sync and on which Slaves it does occur. To solve this data drift issues, one should plan to run the following command on the Master and have all tables or rows be sync with all Slaves:

```
mk-table-sync --execute --replicate IBSTraining.checksum Master_IP
```

a.2) APPLICATION TECHNICAL PERFORMANCE - SCALABILITY:

The scalability issue in this application lies mainly on the ability to cope with a number of slave servers at the same time. Bridge simulators are generally used with a limited number of trainees. In other word, training for example 60 trainees simultaneously may not be an issue at all, provided that the computer processor is modern. The MySQL server database itself is very fast, reliable and can handle billions of data.

However, due to economical issues some servers may be primarily configured to handle a limited amount of load. As the demand grows up the need for scaling the servers become a reality.

There are two alternative to achieve scalability: the ***scaling up*** - which means adjusting, tuning or reconfiguring the existing servers. Scaling up the servers is particularly achieved by setting new parameters on the servers or by reviewing the topology of the servers in interaction. This means analyzing the data distribution algorithm. Meanwhile, scaling up might also means changing the hardware components of the servers.

The ***scaling out*** – means putting additional servers in the system to compensate the load. This alternative solution is a fast, easiest, and cost effective. Example: relay server, Fig. 29.

b.1) APPLICATION OPERATIONAL VALUE - USABILITY:

Usability is much about the application "easy to use". One of the advantages in using the application is the ability to manage databases and data remotely including: storing, retrieving, plotting and analysis with GIS tools, deleting, updating, transaction, table locking, etc. When the Trainees are connected with their server, plotting the oil spill map or other data only takes few seconds. On the map, all objects' attributes are well organized and easily accessible. They can be displayed in a tabular format, for example Fig. 31. The color and ***symbology*** (symbols with piece of information) on each vector layer can be carefully chosen and classified in order to increase the readability, as shown in Fig. 4.

Most of the errors that are displayed from the application are informative, meaning that the error feedbacks usually guide the user to try new solutions.

One of the drawbacks of this application is the impossibility for a Trainee to communicate with another Trainee. But this might be convenient while taking the test scenarios. Oil spill response often requires coordinate efforts among response personnel. However, it is possible to revise the replication topology and eliminate this drawback. This can be achieved by setting up a Master to Master communication topology. In that sense, the Trainees will have the same database privileges as the Trainer. If the database security is not of concern, this kind of topology could be promising in real oil spill emergency response.

This application does not require any user to be a computer expert to in order to follow the training course. The application is easy to use and the trainees will have to focus more on analysis of plotted data and find the best options for response.

The added value is clarified on the application's capability to be operated remotely. The application has high re-usability as new functionalities can be added without changing its main structure.

b.2) APPLICATION OPERATIONAL VALUE - ACHIEVEMENT:

The achievement is based on gaining knowledge and experience in dealing with the oil spill while using the application. To focus on the achievement, it is assumed that the application meets high availability, scalability, and great usability.

Visualization is so crucial that it may be considered as a prerequisite to any oil spill response. One can simply not perform any spill recovery without detecting or seeing the actual spill.

"T*he current trend for GIS is that accurate mapping and data analysis are completed while in the field*", [18].

However, it seems reasonable to consider both visualization and information. Information is based on knowledge of data and visualization is based on picturing the data in the mind. Both approaches are complementary and valuable

Most of the time, oil spill emergency team rush at the spill site and rely on visualization only; they are not well aware or pay less attention on the characteristics of the objects they are relying on or combating. Their only focus is on the spill removal based on visual means. They simply neglect the characteristics of objects present in the system. This attitude can have serious impacts on the outcome of the spill response. One of the consequences is a scenario in which the spill intervention induces more damage to the environment than the spill itself. This situation often occurs with the lack of information related to the spill site. For example, the spill may be mistakenly diverted to a risky (low, medium, or high) area instead of a predefined sacrificial zone (area with zero environmental sensibility indexes with regards to the marine life and human interests).

The lack of information or a poor information sharing can lead to terrible decisions during spill response. Since there are different categories of oil – for example "Light oil" and "Heavy oil", these two types of oil have different weathering and disintegration process and rate in sea water. For Light oil, just monitoring its fast disintegration might be sometimes the best response alternative, as long as the spill does not threaten the marine and human resources. This is also where the NEBA (Net Environmental Benefit Analysis) comes into place as it is assessed whether action or no action needs to be taken with regards to consequences – damage, cost and benefit.

There is no clarity on the amount of spill that really necessitates an emergency response. Some spill might be very large, small, or very small. In any of these cases the SAR radar might detect the spill and trigger the alarm. These types of decisions are made by decision makers.

Some of the main drawbacks of this ice-bridge simulator application are the non-real time dynamic simulation of layers such as Oil, Ice, Water current, Ship motion, and equipment. The Trainees have to position their equipment by dragging them in proper locations during spill response.

The application has six different testing scenarios, (App. B) and the Trainees can apply several recovery methods in each scenario. The question which arises is whether there is an established regulatory framework or standard procedure in selecting a specific spill response technique.

The best value of the application is the gain of knowledge on existing recovery methods and how they can be correlated with real-time information. The process of achieving this knowledge is invaluable.

The new approach is that oil spill emergency personnel truly need to consider the possibility of having immediate access of all relevant objects' information on the spill site as well as benefiting real time information data sharing with onshore base station.

Chapter 6

Concluding Remarks

6.1 Conclusions

The objective of this thesis was to build an application for oil spill response model in ice-bridge simulator training and demonstrate its added value.

After a literature search and analysis report, it appeared that such application were non-existent. It was found that technologies such as: spill tracking and trajectory system; response options – mechanical and non-mechanical recovery systems, were the major focus of many scientists and oil spill response agencies. Such technologies although very useful, they cannot guaranty satisfaction unless they are correlated with emergency personnel skill and real-time object information and data sharing with onshore base operator. During spill response, most on-scene emergency personnel do not have knowledge on the critical objects (spill, ice, sensitive area, sacrificial zone, etc) they are surrounded with. The lack of object information and sharing is one of the major sources for recovery delay or failure. It can induce more damage to the environment.

This application simulates the training scenario and brings new ideas on how to handle oil spill emergency crisis at sea. It demonstrates the possibility to extract important object information from the spill side while performing the spill recovery. The application also simulates real-time communication and object information sharing between the trainees and the Trainer by dynamically using database and data replication techniques. It was found that this application can be applied on a mobile base and all the trainees do not have to be in the same simulator centre to attend the training course at the same time.

For validity purposes it was found that the application is very fast and has high availability and scalability. This application is easy to use and new features can be added without changing its overall structure.

The main drawbacks of this application is the absence of real-time simulation of environmental data such as wave currents, ice dynamic, oil weathering, and wind. The current application does not support communication between trainees. All communication is between the trainer and the trainees. This setting can be acceptable during the trainees' test- examination, but problematic otherwise. Because the oil spill response necessitates emergency personnel's joint effort.

However, is possible to redesign this application to resolve these issues.

6.3 Proposals for Further Work

The proposed application can be enhanced by integrating real-time simulation of in-situ systems such as oil weathering, and trajectory, ice motion dynamic, and wave current, and wind.

The application can be extended to include new features such as sharing the information among trainees and establish joint training simulation efforts.

References

[1] Ian G. Cumming and John R. Bennett: Digital Processing of SEASAT SAR Data. In Record of the IEEE 1979 International Conference on Acoustics, Speech and Signal Processing , Washington, B.C.

[2] J. C. Curlander, R. N. McDonough: Synthetic Aperture Radar- Systems and Signal Processing, John Wiley and Sons, 1991.

[3] Soumekh M: Synthetic Aperture Radar Signal Processing, John Wiley & Sons, New York, 1999.

[4] Peter Buxa, LeRoy Gorham, Mathew Lukacs, David Caliga: Mapping of a 2D SAR Backprojection Algorithm to an SRC Reconfigurable Computing MAP Processor

[5] Andrei Yu. Ivanov, Victoria V. Zatyagalova: A GIS Approach to Mapping Oil Spill in the Marine Environment

[6] Alistair Cockburn: Writing Effective Use Cases, Addison-Wesley 2000

[7] Cumming I. and Wong F: Digital Processing of Synthetic Aperture Radar Data, MA, 2005.

[8] Hee-Sub Shin and Jong-Tae Lim: Omega-k Algorithm for Airborne Spatial Invariant Bistatic Spotlight SAR Imaging, IEEE, 2009

[9] Pascal Roques: UML - Modéliser un Site E-commerce Cahier du Programmeur, Eyrolles, 2002

[10] Doug Rosenberg and Matt Stephens: Use Case Driven Object Modeling with UML, Apress 2007

[11] Doug Rosenberg, Kendall Scott: Applying Use Case Driven Object Modeling with UML, 2001

[12] Iphigenia Keramitsoglo, Constantinos Cartalis, Chris T. Kiranoudis: Automatic identification of Oil Spills on Satellite Images, , 2004

[13] Kounchou Tagne Ulrich Alain: Deepwater Blowout- Operational Aspects of Oil Spill Response in Arctic Ice-covered Waters, 2010

[14] http://dev.mysql.com/tech-resources/articles/4.1/gis-with-mysql.html

[15] Alias Abdul-Rahman, Morakot Pilouk: Spatial Data Modelling for 3D GIS, Springer 2008

[16] B, Schwartz, P. Zaitsev, V. Tkachenko, Jeremy D. Zawodny, A. Lentz, Derek J. Balling: High Performance MySQL, 2008

[17] Ryan Bannon, Alvin Chin, Faryaaz Kassam, Andrew Roszko: MySQL Conceptual Architecture, 2002

[18] http://en.wikipedia.org/wiki/Geographic_information_system

[19] Poojitha D. Yapa, Lalith K. Dasanayaka, Hung Tao Shen, Hayley H. Shen: Modeling Oil Transport and Spreading in Icy Waters

[20] Jong-Sen Lee and Eric Pottier: Polarmetric Radar Imaging - from Basic to Applications, CRC Press 2009

[21] Mark Reed, Henrik Rye, Ismail Durgut, Øistein Johansen, May Kristin Ditlevsen, Ben Hetland, Marinela Gerea, Boye Høverstad, Kjell Skognes, Ole Morten Aamo, Narve Ekrol, Keith Downing: Oil Spill Contingency and Response (OSCAR) Model System, 2010

[22] http://response.restoration.noaa.gov/gnome

[23] http://www.asascience.com/software/oilmap/

[24] Paul M. Mather: Computer Processing of Remotely-Sensed Images - An Introduction, Wiley 2004

[25] http://www.gpsbabel.org/

[26] Antonio Martinez, Victoriano Moreno: An Oil Spill Monitoring System Based on SAR Images, 1996

[27] Iphigenia Keramitsoglou, Constantinos Cartalis, Chris T. Kiranoudis: Automatic Identification of Oil Spills on Satellite Images, 2006

[28] Lena Chang, Z.S. Tang, S.H. Chang, Yang-Lang Chang: A Region-based GLRT Detection of Oil Spills in SAR Images

Appendix A

Plotting Figures in Training Scenarios

Training Scenario 1:

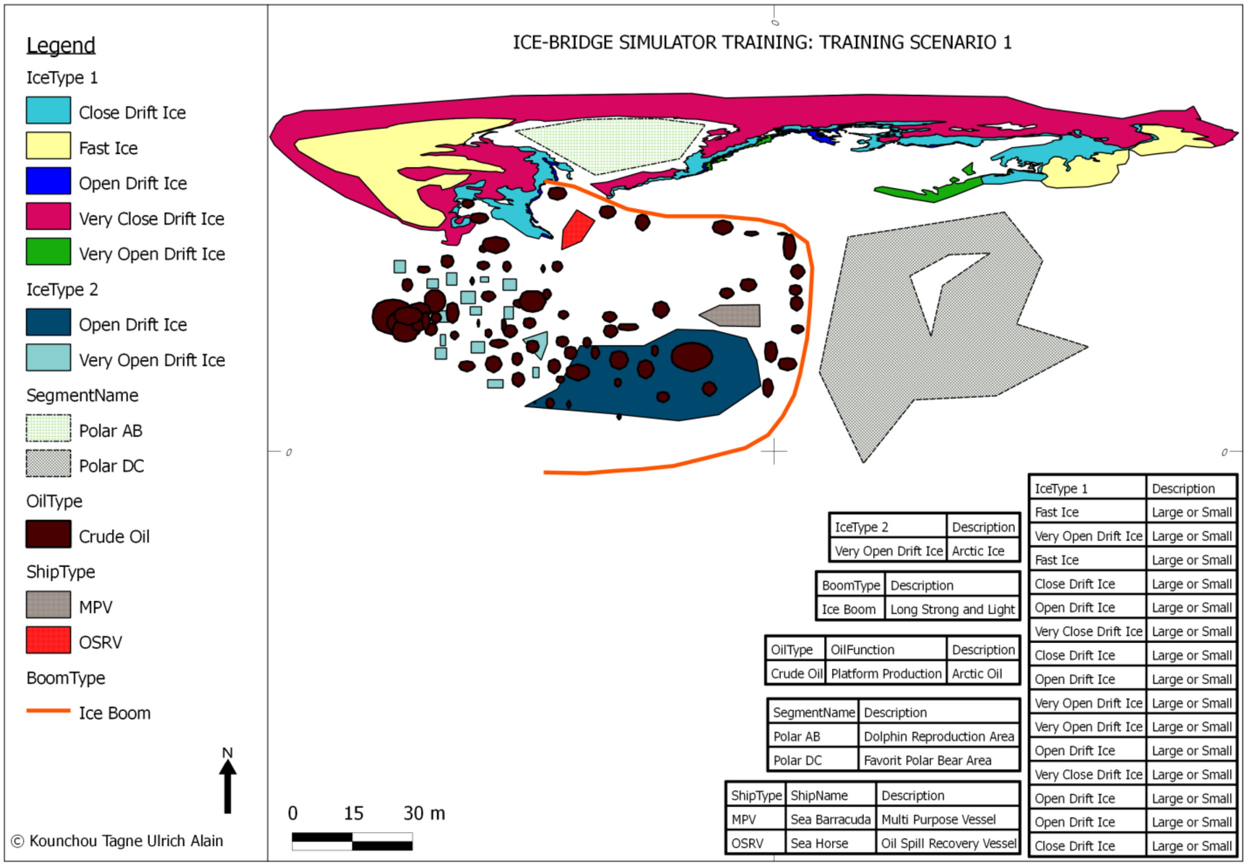


Figure 38: Training Scenario 1

Training Scenario 2:

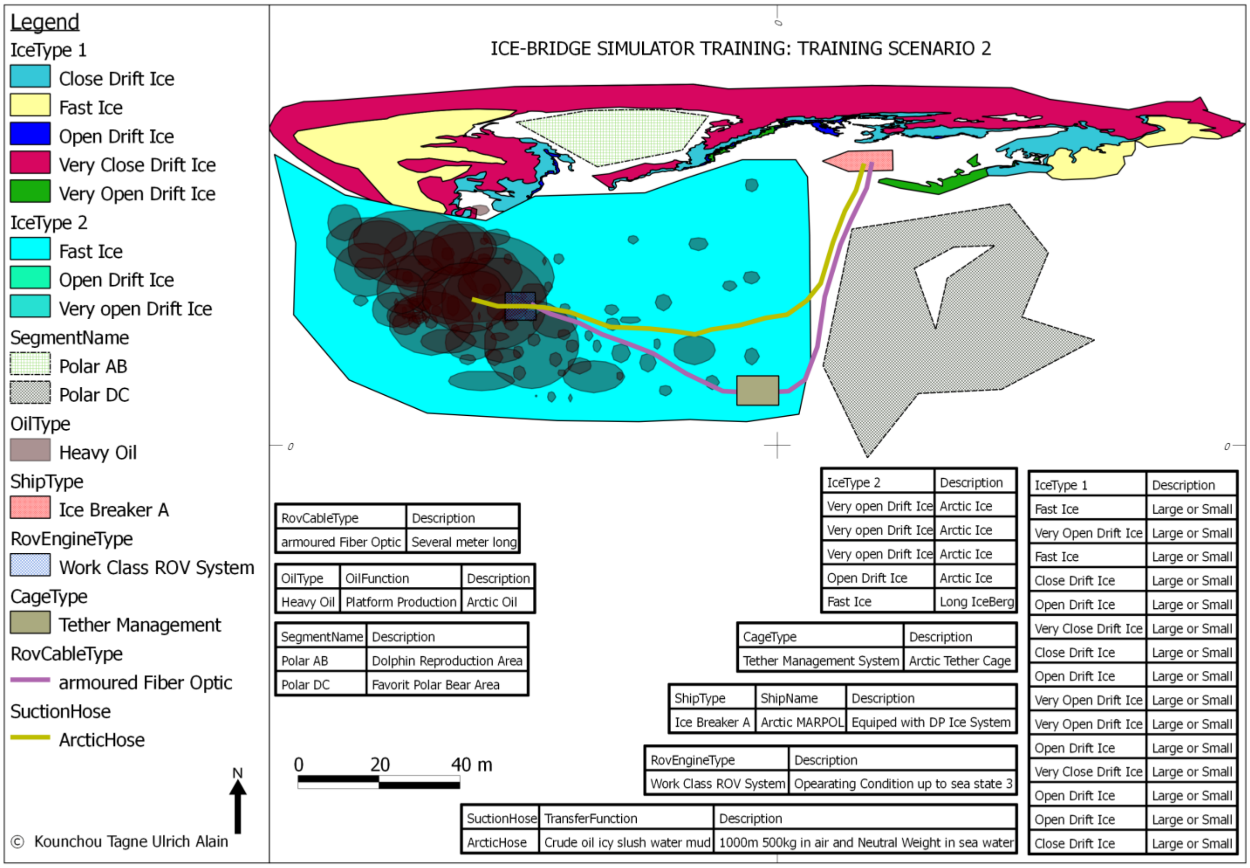


Figure 39: Training Scenario 2

Training Scenario 3:

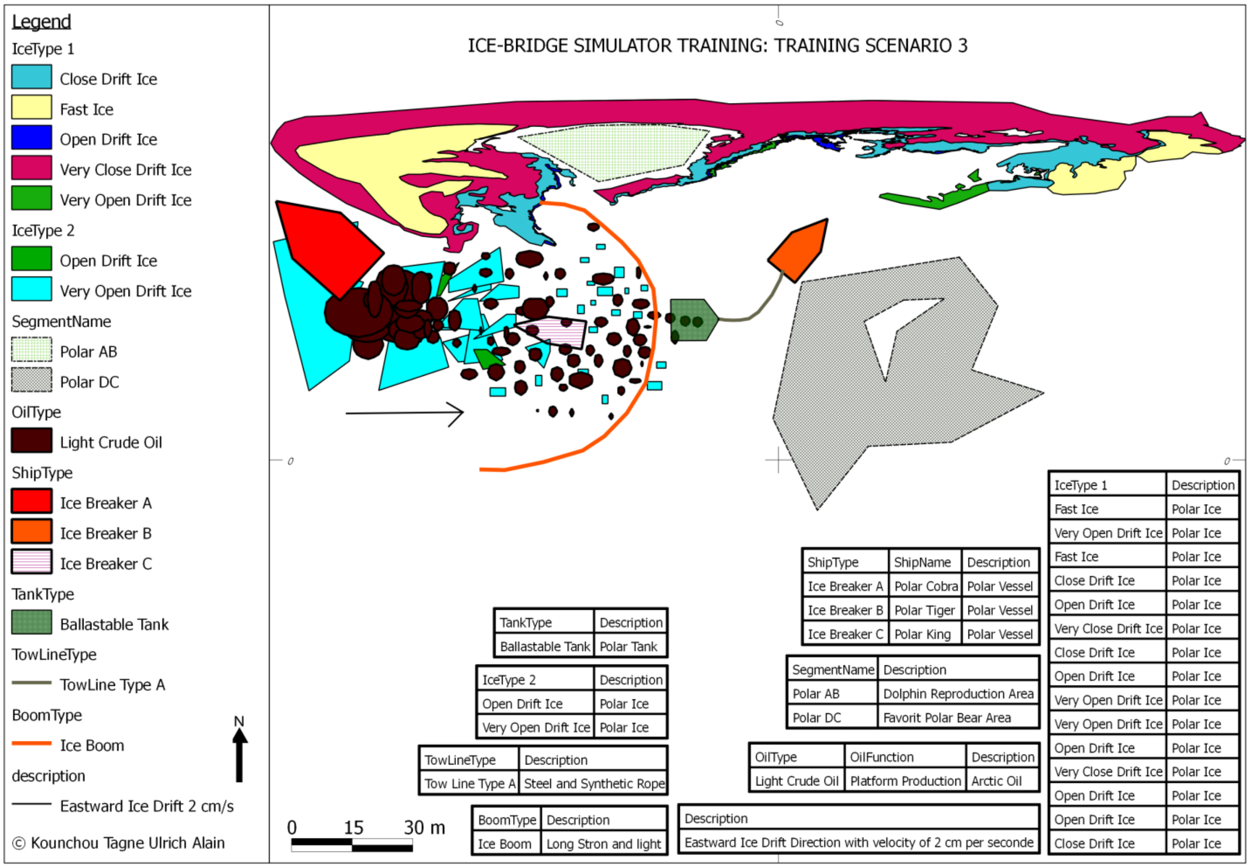


Figure 40: Training Scenario 3

<u>Training Scenario 4</u>:

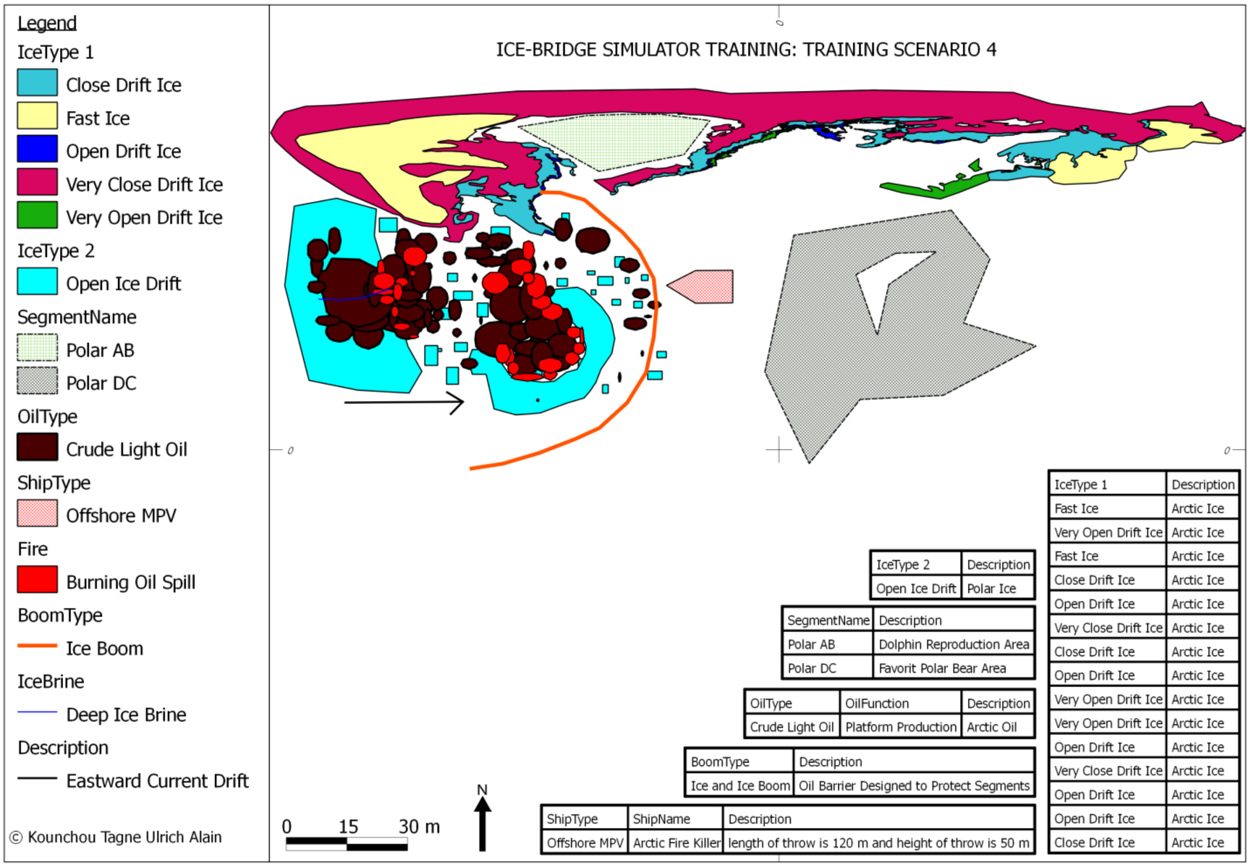


Figure 41: Training Scenario 4

Training Scenario 5:

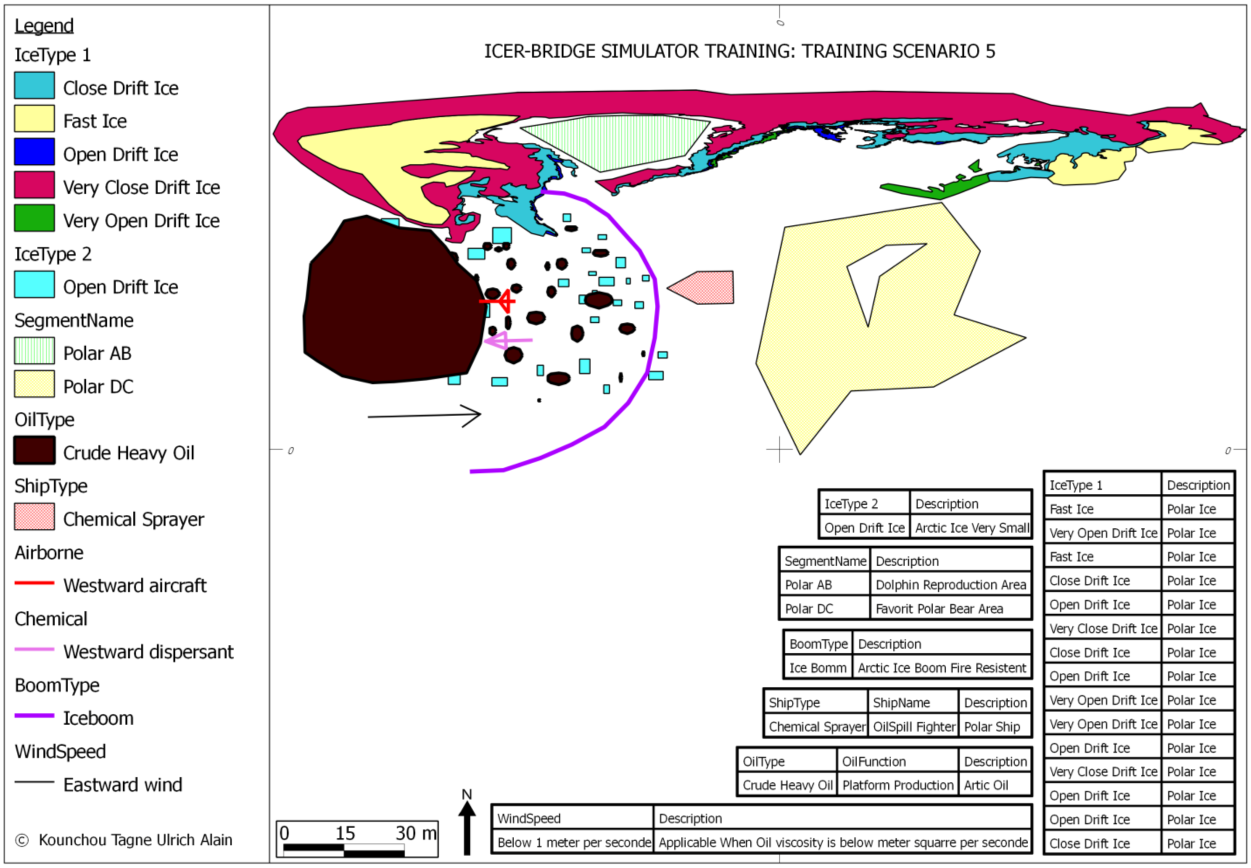


Figure 42: Training Scenario 5

Training Scenario 6:

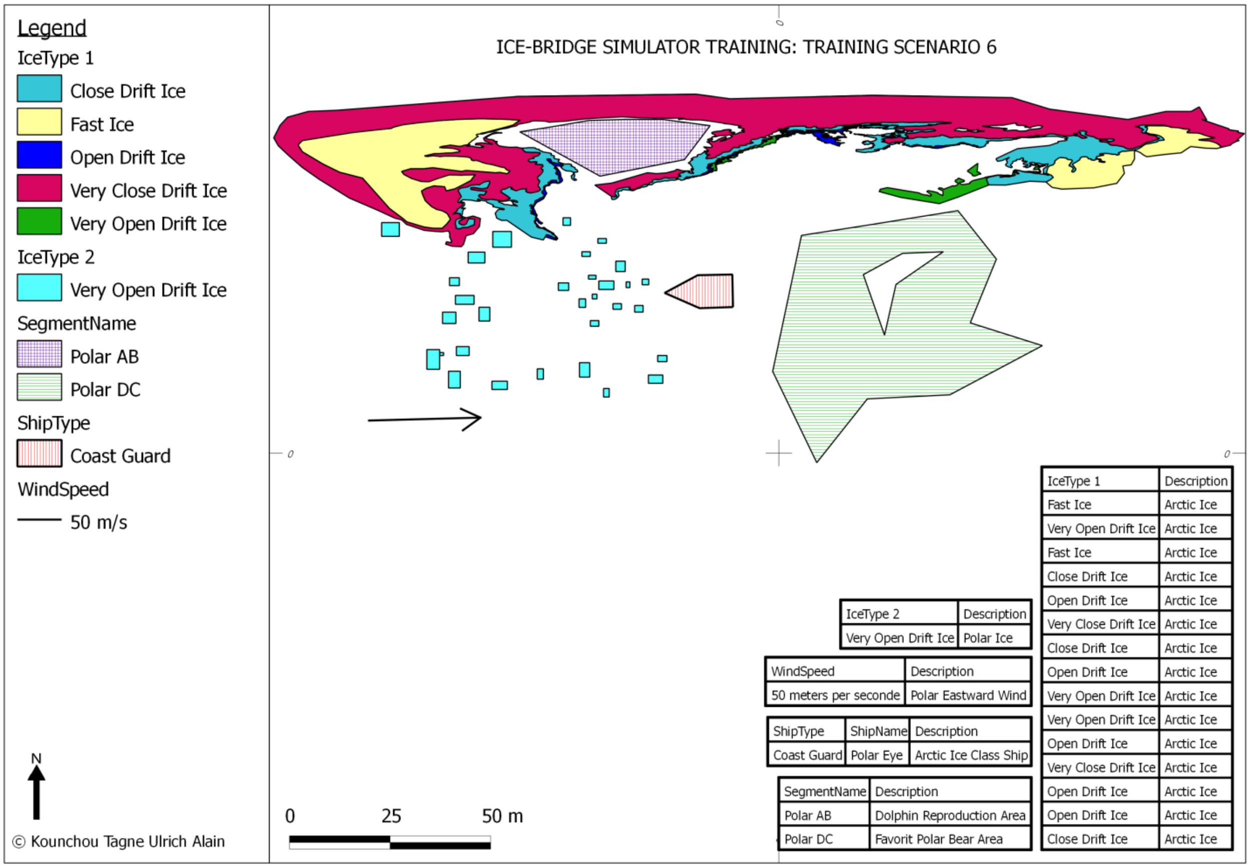


Figure 43: Training Scenario 6

Appendix B

Plotting Figures in Testing Scenarios

Test Scenario 1:

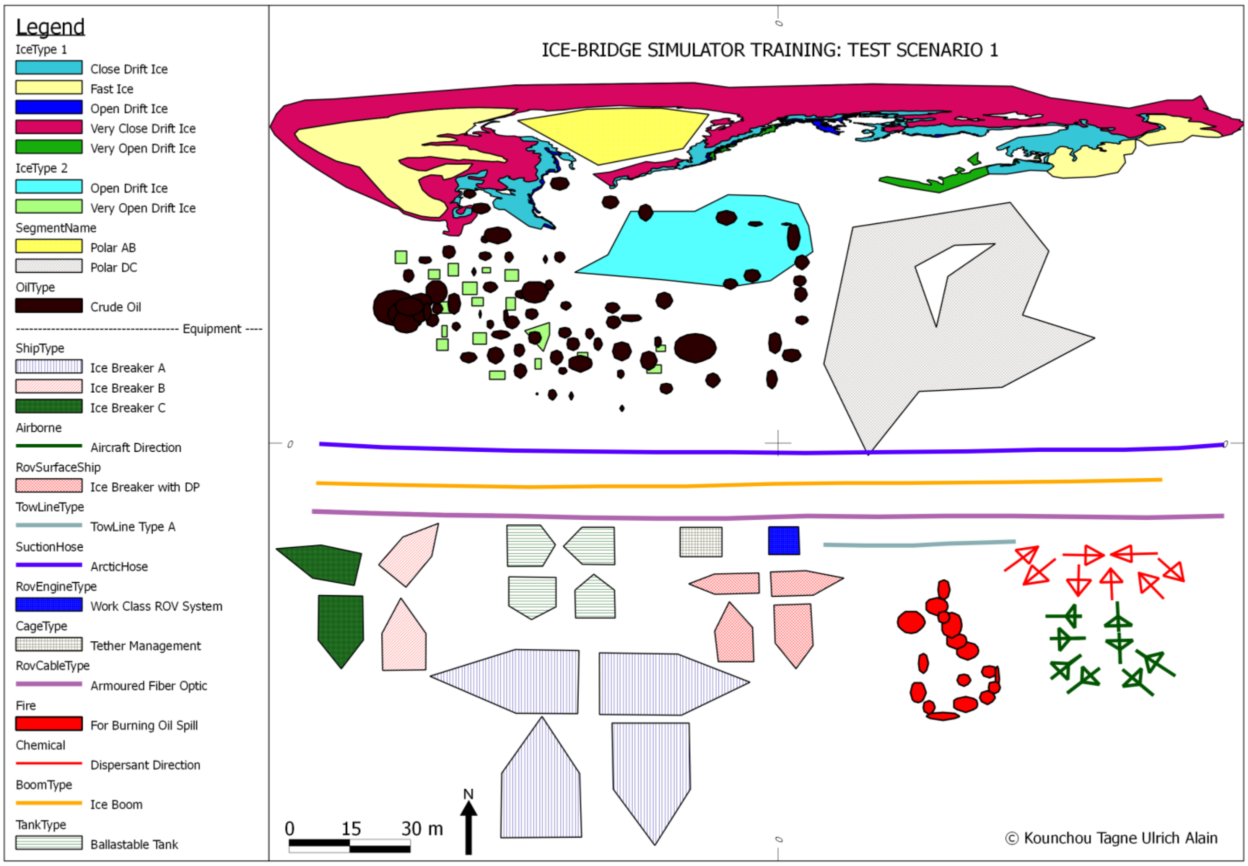


Figure 44: Test Scenario 1

Test Scenario 2:

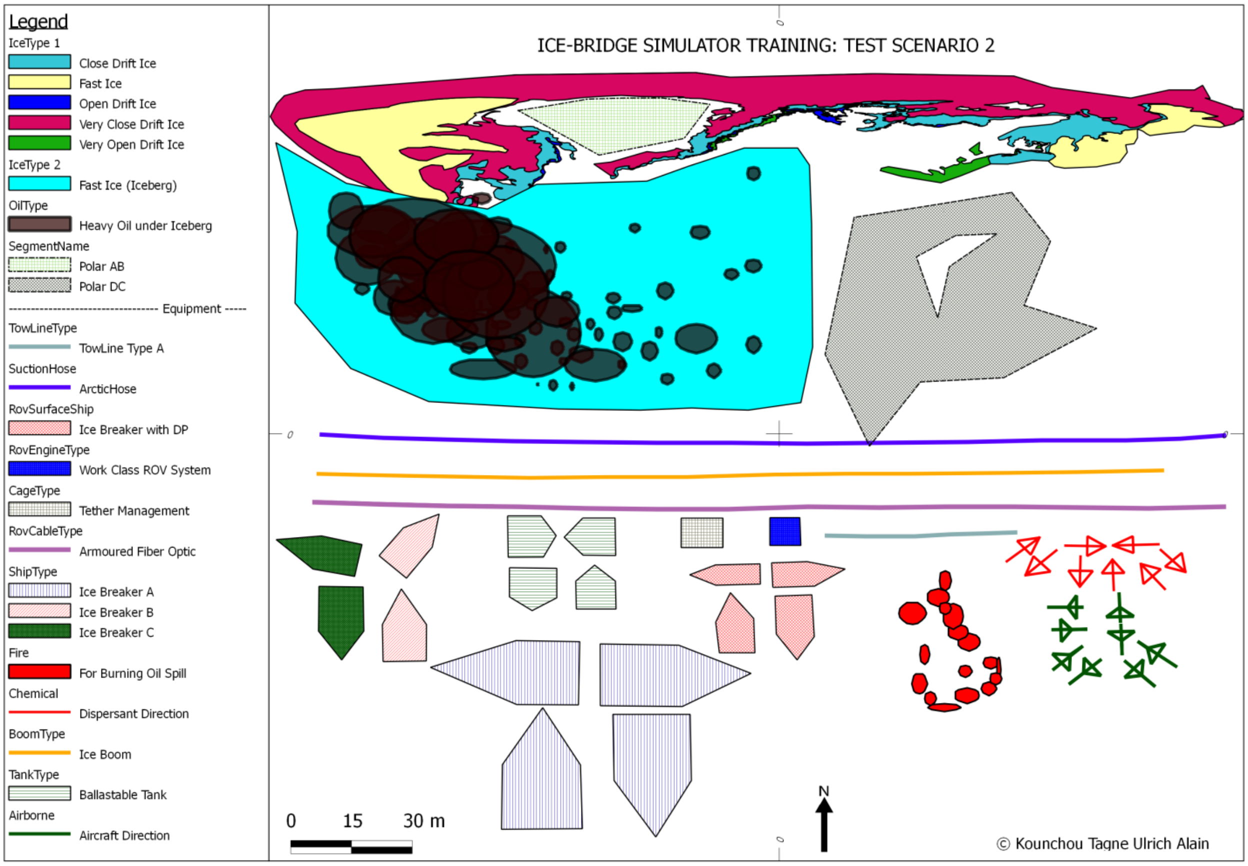


Figure 45: Test Scenario 2

Test Scenario 3:

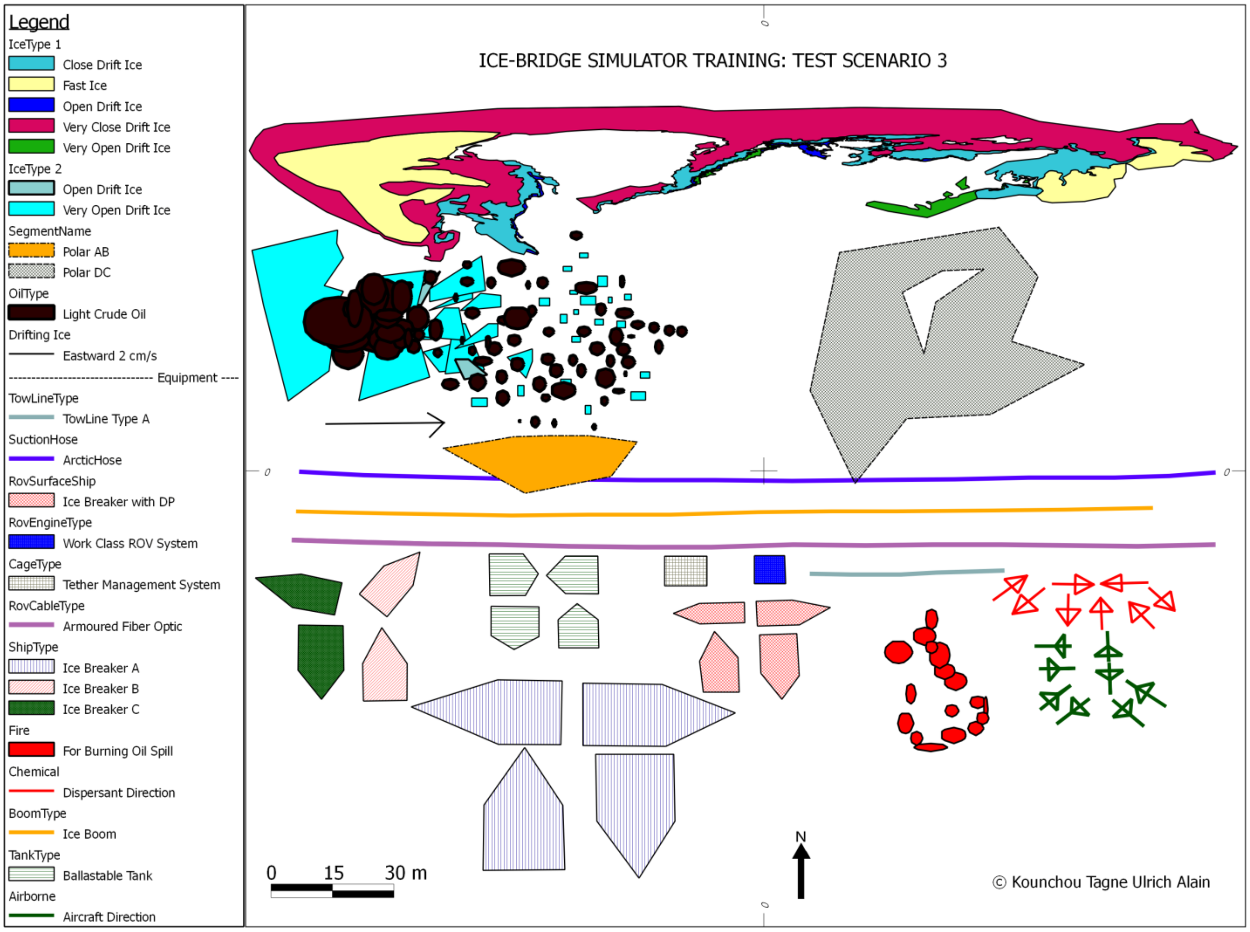


Figure 46: Test Scenario 3

Test Scenario 4:

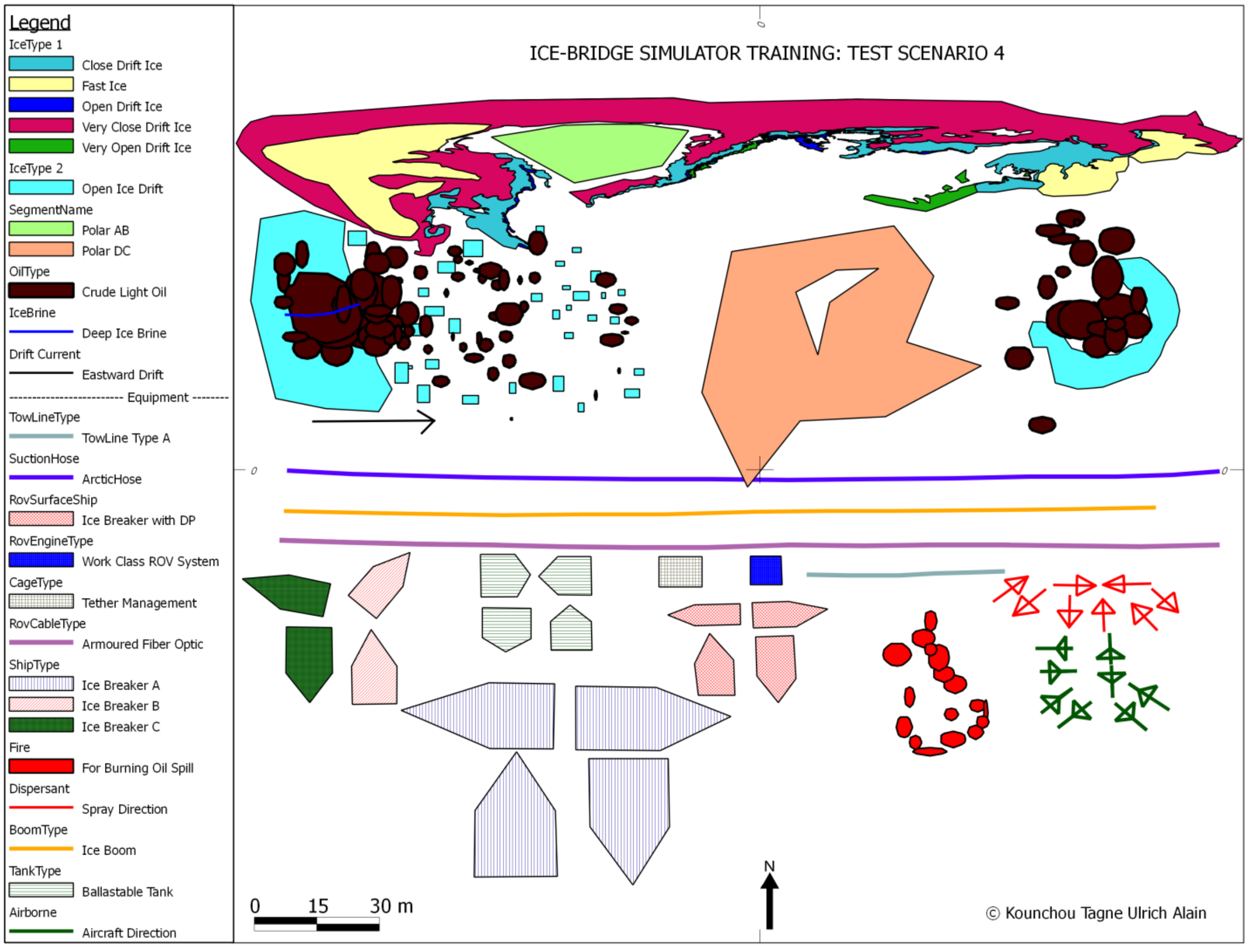


Figure 47: Test Scenario 4

Test Scenario 5:

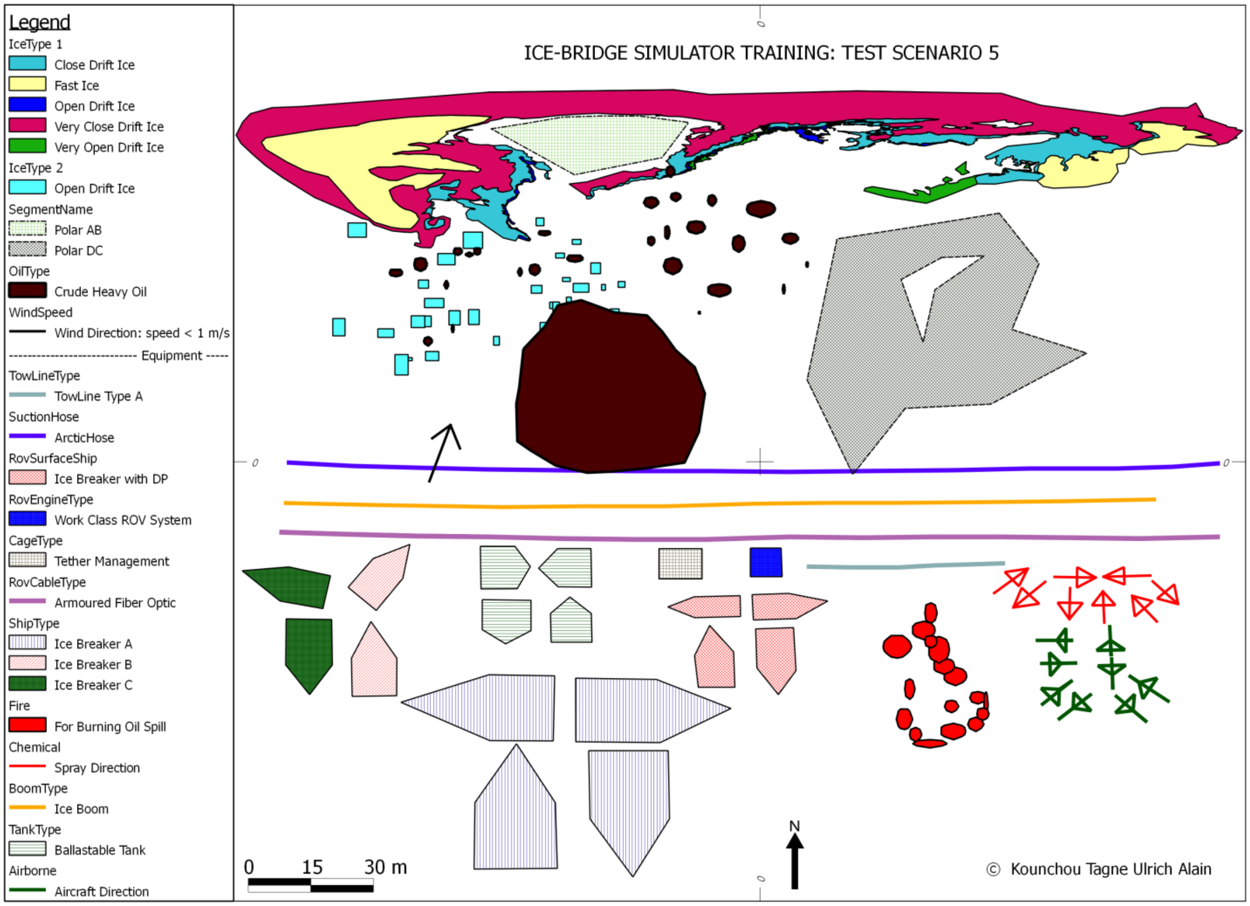


Figure 48: Test Scenario 5

Test Scenario 6:

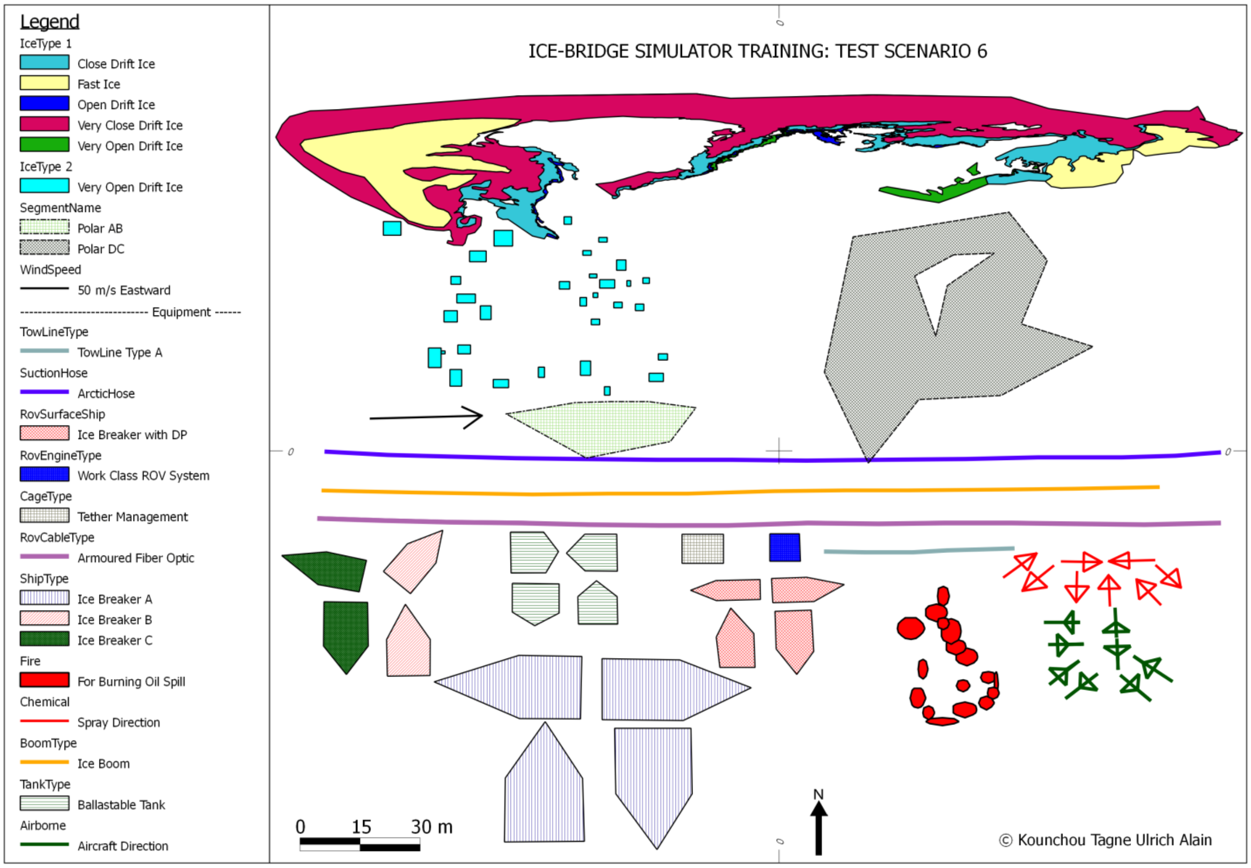


Figure 49: Test Scenario 6

Appendix C

Servers Setting for the Training and Testing

GENERAL CONSIDERATIONS

Database Server used:	=	MySQL version 5.xx
IBSTraining	=	Database name for Ice-bridge simulator training.
IBSTesting	=	Database name for Ice-bridge simulator testing.
Master	=	Username of the Master
Trainee1	=	Username of a trainee
Master_IP	=	Master IP or host address
Slave_IP	=	Slave IP or host address
Master_password	=	Password of the Master User
Slave_password	=	Password of Trainee1
1_Train_All_SQL_Tables.sql	=	File containing all data for the training scenarios.
1_Test_All_SQL_Tables.sql	=	File containing all data for the testing scenarios.
Trainee1_Svar_Scenario_1_Tab.shp	=	Shapefile generated by the Trainee for answering scenario 1

MASTER SERVER – TASKS

/* 1 SHUTING DOWN THE MySQL SERVER */

In windows 7: Start + Control Panel + System and Security + Administrative Tools + Services + Find "MySQL" in the list and click "Stop".

/* 2 CREATING FOLDER DIRECTORIES */

```
C:\Program Files\MySQL\MySQL Server 5.0\bin\dump

C:\Program Files\MySQL\MySQL Server 5.0\data\ulratagnmasterlog
```

/* 3 CONFIGURING THE MASTER " my.ini " FILE */

```
[mysqld]

# changes made to do master

server-id = 1
binlog-do-db=IBSTraining

log-bin = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-bin
log-error = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql.err
log-bin-index = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-log-bin.index

relay-log = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-relay-bin
relay-log-index = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-relay-bin.index
relay-log-info-file = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-relay-log.info

master-info-file = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-master.info

# end master
```

/* 4 STARTING THE MASTER MySQL SERVER */

In windows 7: Start + Control Panel + System and Security + Administrative Tools + Services + Find "MySQL" in the list and click "Start".

/* 5 SETTING THE STORAGE ENGINE SESSION */

```
master > SET storage_engine=MYISAM;
```

/* 6 CREATING DATABASE */

```
master > Create database IBSTraining;
master > Use IBSTraining;
```

/* 7 CREATING ALL THE TRAINING TABLES AND LOAD THE DATA INTO THEM: */

Retrieve and run the "complete scenario 1" from the " 1_Train_All_SQL_Tables.sql " file in a SQL Query Browser

/* 8 GRANTING REPLICATION PRIVILEGES TO SLAVE (Trainee1) WITH IP */

```
master > Grant replication slave, replication Client, reload, super, select on *.* to 'Trainee1'@'Slave_IP' identified by 'Slave_Password';

master > Grant all on IBSTraining.* to 'Master'@'Master_IP'  identified by 'Master_Password';

master > Grant Select, Reload, Super on *.* to 'Master'@'Master_IP' identified by 'Master_Password';

master > grant drop on *.* to 'Trainee1'@'Slave_IP' identified by 'Slave_Password';

master > grant drop on *.* to 'Trainee1'@'Master_IP' identified by 'Slave_Password ';

master > flush privileges;
```

/* 9 LOCKING THE TABLES FOR READING AND SHOWING MASTER STATUS AND READING THE BINARY FILE NAME AND IT POSITION NUMBER (to be remembered): */

```
master  > flush tables with read lock;

master > show master status;
master > show processlist\G;
```

/* FROM DOS COMMAND PROMPT */

/* 10 NAVIGATING TO DUMP DIRECTORY */

```
C:\Users\Ulrich Tagne>cd C:\Program Files\MySQL\MySQL Server 5.0\bin\dump
```

/* 11 PRODUCING A DATABASE DUMP FILE (IBSTrainingdump.sql) IN DUMP FOLDER */

```
mysqldump -q -u Master -p --opt IBSTraining --single-transaction --master-data=1 --host=Master_IP
>IBSTrainingdump.sql
```

/* GO AND CONFIGURE THE SLAVE COMPUTER */

/* 22 SENDING THE MASTER DUMP FILE TO SLAVE COMPUTER FROM DOS IN MASTER*/

```
C:\Program Files\MySQL\MySQL Server 5.0\bin\dump>IBSTrainingdump -q -u Master --
password=Master_password --opt IBSTraining --single-transaction --master-data=1 --host=Master_IP | mysql -u
Trainee1 -p IBSTraining -h Slave_IP < IBSTrainingdump.sql
Enter password: ******** (Enter the Slave_password)
```

/* GO AND START THE SLAVE ON THE SLAVE COMPUTER */

ADDITIONAL TASKS ON MASTER:

/* Master retrieving **All data** from Slave and saving it in the Master dump directory */

```
mysqldump  --add-drop-table -q -u Trainee1 -p --opt IBSTraining --single-transaction --host= Slave_IP >
Trainee1_Svar_Scenario_1_Tab.sql
Enter password: ********(Enter the Slave password)
```

/* Master retrieving **only svar data** from Slave and saving it in the Master dump directory */

```
Mysqldump  -u Trainee1 -p IBSTraining Trainee1_Svar_Scenario_1_Tab --host=Slave_IP >
Trainee1_Svar_Scenario_1_Tab.sql
Enter password: ********(Enter the Slave password)
```

/* Loading the dumped file into Master Database (IBSTraining) from DOS*/

```
mysql -u Master -p IBSTraining -h Master_IP  < Trainee1_Svar_Scenario_1_Tab.sql
```

/* Correcting the Trainee Svar + dumping the corrected file + sending the dumped file to slave */

The corrected file name is: **Trainee1_Svar_Scenario_1_Corrected_Tab**

Make a dump of "*Trainee1_Svar_Scenario_1_Corrected_Tab* " on Master:

```
mysqldump -q -u Master -p --opt IBSTraining Trainee1_Svar_Scenario_1_Corrected_Tab --single-transaction --
master-data=1 --host=Master_IP > Trainee1_Svar_Scenario_1_Corrected_Tab.sql
```

Send the dumped file to Slave:

```
C:\Program Files\MySQL\MySQL Server 5.0\bin\dump> Trainee1_Svar_Scenario_1_Corrected_Tab -q -u Master --
password=Master_password --opt IBSTraining --single-transaction --master-data=1 --host=Master_IP | mysql -u
Trainee1 -p IBSTraining -h Slave_IP < Trainee1_Svar_Scenario_1_Corrected_Tab.sql
Enter password: ******** (Enter the Slave_password)
```

SLAVE SERVER - TASKS

/* 12 SHUTING DOWN THE MySQL SERVER */

In windows 7: Start + Control Panel + System and Security + Administrative Tools + Services + Find "MySQL" in the list and click "Stop".

/* 13 CREATING FOLDER DIRECTORIES */

C:\Program Files\MySQL\MySQL Server 5.0\bin\dump

C:\Program Files\MySQL\MySQL Server 5.0\data\ulratagnmasterlog

/* 14 CONFIGURING THE SLAVE " my.ini " FILE */

```
[mysqld]

# changes made to do slave

server-id = 2
replicate-do-db=IBSTraining

log-bin = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-bin
log-error = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql.err
log-bin-index = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-log-bin.index

relay-log = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-relay-bin
relay-log-index = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-relay-bin.index
relay-log-info-file = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-relay-log.info

master-info-file = C:/Program Files/MySQL/MySQL Server 5.0/data/ulratagnmasterlog/mysql-master.info

# end slave setup
```

/* 15 STARTING THE SLAVE MySQL SERVER */

In windows 7: Start + Control Panel + System and Security + Administrative Tools + Services + Find "MySQL" in the list and click "Start".

/* 16 CREATING DATABASE */

```
slave > Create database IBSTraining;
slave > use IBSTraining;
```

/* 17 GRANTING REPLICATION PRIVILEGES TO SLAVE (Trainee1) WITH IP */

```
slave > Grant replication slave, replication Client, reload, super, select on *.* to 'Trainee1'@'Master_IP' identified by 'Slave_Password';
```

```
slave > grant all on IBSTraining.* to 'Trainee1'@'Slave_IP' identified by ' Slave_Password ';

slave > grant drop, create, update on IBSTraining.* to 'Trainee1'@'Slave_IP' identified by 'Slave_Password ';

slave > grant select, reload, super on *.* to 'Trainee1'@'Slave_IP' identified by 'Slave_Password ';
```

```
slave > grant drop, create, update on *.* to 'Trainee1'@'Slave_IP' identified by 'Slave_Password ';

slave > flush privileges;
```

*/ 18 CONNECTING THE SLAVE WITH THE MASTER */

```
slave> CHANGE MASTER TO MASTER_HOST='Master_IP';

slave> CHANGE MASTER TO MASTER_USER='Trainee1';

slave> CHANGE MASTER TO MASTER_PASSWORD='Slave_Password';

slave> CHANGE MASTER TO MASTER_LOG_FILE='put the binary file name ';

slave> CHANGE MASTER TO MASTER_LOG_POS= put the position of the binary file name;
```

*/ 19 STARTING SLAVE */

```
slave > start slave;
```

*/ 20 CHEKING THE STATUS OF THE SLAVE AND ITS I/O AND SQL THREAD */

```
slave > show slave status\G
```

```
slave > SHOW PROCESSLIST\G;
```

/* 21 STOPING SLAVE */

```
slave > stop slave;
```

/* GO AND CONTINUE WITH THE MASTER COMPUTER */

/* 23 STARTING THE SLAVE */

```
slave > start slave;
```

NB: At this point, the Slave has retrieved all IBSTrainingdump data.

/* 24 CHECKING THE POPULATED DATABASE ON THE SLAVE */

```
slave > Show Tables;
```

ADDITIONAL TASKS ON SLAVE:

/ * LOADING TRAINEES ANSWER INTO THEIR OWN DATABASE USING FWTOOLS SHELL OR EXPORT2MySQL */

/* EXAMPLE: FOR SNENARIO 1 WITH Trainee1 */

The directory where the trainee save his / her work: C:/IBSTResponse /

The Trainee file containing answer: Trainee1_Svar_Scenario_1_Tab.shp

The filename to be loaded into database: Trainee1_Svar_Scenario_1_Tab

```
ogr2ogr -f "MySQL"
MySQL:"IBSTraining,user=Trainee1,host=Slave_IP,password=Slave_Password,port=3306" -nln
"Trainee1_Svar_Scenario_1_Tab" -a_srs "EPSG:4326" -lco engine=MYISAM C:/IBSTResponse
/Trainee1_Svar_Scenario_1_Tab.shp
```

/* LOADING TRAINEE FEEDBACK INTO THE IBSTraining DATABASE (Slave) */

EXAMPLE OF INSERTING FORMAT:

```
INSERT INTO Trainees_Feedback_Tab (UserName, ScenarioNumber, FeedbackDescription)VALUES
(
'Trainee1',
'1',
'
My long text feedback

'
);
```

Appendix D

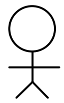
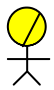
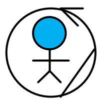
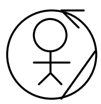
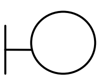
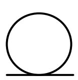
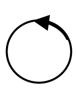
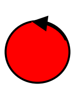
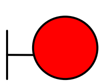
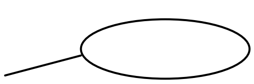
Symbols and Definitions

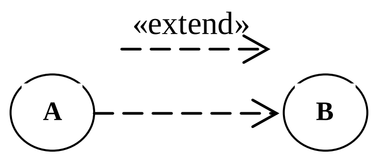
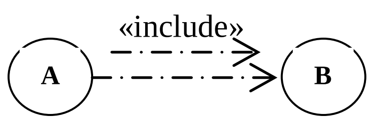
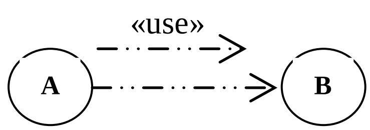
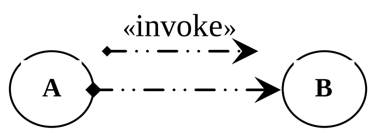
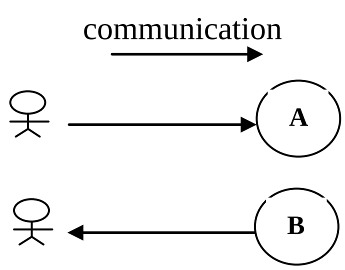
Table 7: ASSOCIATION BETWEEN CLASS OBJECTS

Type of associations	Symbol	Meaning
Bidirectional		Both class A and B are aware of each other
Association class		When two instances of classes A and B are associated, there will be another instance of class C providing additional information to them
Dependency		Class A depends on class B
Aggregation		Class B is a weak part of class A. B will survive if A is deleted
Composition		Class B is a strong part of class A. B will not survive if A is deleted
Generalization		A is a super-class (parent) while B and C are sub-classes (child). So in addition to their own specific attributes, B and C have same attributes in A

Table 8: CLASS MULTIPLICITY

Type of Multiplicities	Meaning
0..*	Zero or many
1	Only one
*	Many
1..*	One or many

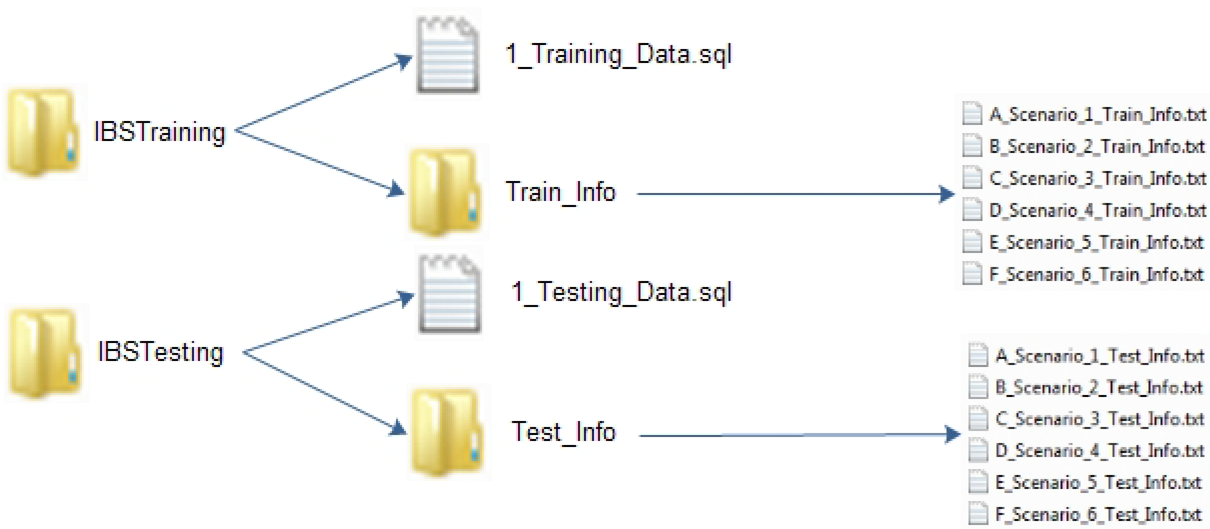
	Actors representing: Trainer, Trainees, Administrator, and Stakeholder
	Trainee in Ice-bridge Simulator
	Trainer in Ice-bridge Simulator
	Administrator in Ice-bridge Simulator
	Stakeholder – a service company delivering environmental data
	Boundary objects: these are screens used by actors to communicate with the system.
	Entity objects: these are entity objects from the conceptual domain model.
	Controllers: These are logical software functions controlling the interaction between the boundary and the entity objects.
	Controller in failure mode, respectively
	Boundary object in failure mode, respectively
	A use case showing the user's operation in the system. it uses a verbal language.

	There is a very weak relationship between the two use cases. The extending use case starts the operation and the base use case may accomplish it. Read the association from the end of arrowhead. *Ex: to achieve B, A may be is called.*
	There is a very strong relationship between the two use cases. It explains how a use can be achieved. *Ex: achieving A also implies doing B*
	There is a strong relationship between the two use cases. It explains how a use case relays on another use case. *Ex: to achieve A, B is used.*
	There is a weak relationship between the two use cases. It explains how use requests help from another use case. *Ex: to achieve A, B can give help, but without certitude.*
	There is a communication link between one actor and a use case. Is explains where the operation goes. *Ex: the actor is providing assistance to A* *Ex: the use case B need assistance from the actor.*

Appendix E

Data for Running the Training and Testing Scenarios

<u>Enclosed on the CD:</u>



The file *1_Training_Data.sql* includes data to plot the training scenario.

The file *1_Testing_Data.sql* includes data to plot the testing scenario.

The file *A_Scenario_1_Train_Info.txt* includes additional information attached to each scenario.

For example refer to Fig. 32.

All other file have same format but different contents.