INTRODUCING EVENT-DRIVEN BUSINESS PROCESS MANAGEMENT TO INTEGRATED OPERATIONS: A CASE STUDY

Master thesis
60 credits

Thomas Bekkeheien
thombek@ifi.uio.no

01.05.2010
Abstract

Event-Driven Business Process Management (EDBPM) is an emerging discipline that combines modern Business Process Management (BPM) solutions with the techniques of Complex Event Processing (CEP), providing situational awareness to companies. The research conducted in this thesis investigates the suitability of EDBPM as a means to introduce event-driven behavior to CODIO, a research project included in Integrated Operations.

The thesis gives an introduction to the disciplines mentioned above, as well as some of the technologies and products used in the CODIO project. Furthermore, we have described and detailed a specific case based on event-driven BPM, elaborating one approach to detect and handle abnormal deviations occurring in directional drilling. To investigate the feasibility of this approach, a prototype has been developed based on the principals of Event-Driven Architecture (EDA). Six particular deviation scenarios have been identified and modeled within the prototype’s event processing engine. Finally, a series of experiments have been conducted in a simulated environment in order to demonstrate this first attempt to introduce event-driven behavior to the Integrated Operations.
Acknowledgements

This thesis is part of my Master’s degree, realized through the Department of Informatics at the University of Oslo. Several people have given me their help during the work on this thesis. First of all, I would like to thank Computas, and especially my supervisor Roar Fjellheim, for creating the problem description and giving me this exciting opportunity. Roar has continuously provided me with much appreciated feedback and advices throughout the work. Secondly, I want to thank my father, Egil Aanestad (Halliburton AS), for his valuable contribution of feedback and knowledge regarding drilling operations, and also for establishing contact with Halliburton. I would also like to thank Helge Rørvik at Halliburton for his thorough review and response to the deviation scenarios proposed in the thesis. Finally, I want to thank Alexey Martyushenko, Øystein Holo, and Miriam Castillo Amo in the CODIO research project for all their help and hints.

Oslo, 23.04.10

Thomas Bekkeheien
Contents

1 Introduction .................................................................................................................................. 1
  1.1 Motivation ............................................................................................................................... 1
  1.2 Problem statement .................................................................................................................. 1
  1.3 Project description .................................................................................................................. 2
  1.4 Research method .................................................................................................................... 2
  1.5 Thesis document structure ..................................................................................................... 3
2 Background .................................................................................................................................. 5
  2.1 Integrated Operations ............................................................................................................. 5
    2.1.1 IO Generations .................................................................................................................. 5
    2.1.2 Value potential .................................................................................................................. 6
    2.1.3 Benefits .......................................................................................................................... 7
    2.1.4 Risks and challenges ......................................................................................................... 8
  2.2 CODIO ................................................................................................................................... 8
    2.2.1 Work packages ................................................................................................................ 9
    2.2.2 Use of existing technology ............................................................................................. 10
  2.3 Directional drilling .................................................................................................................. 10
    2.3.1 Measurement While Drilling (MWD) .............................................................................. 11
    2.3.2 Geosteering ................................................................................................................... 12
  2.4 Related research ..................................................................................................................... 12
    2.4.1 EDBPM in financial services ......................................................................................... 12
    2.4.2 EDBPM in logistics ......................................................................................................... 12
3 Research ..................................................................................................................................... 13
  3.1 Research goal .......................................................................................................................... 13
  3.2 Research questions .................................................................................................................. 13
    3.2.1 What synergies can be found between BPM and CEP? ................................................ 13
    3.2.2 What is a viable architecture for an Event-Driven BPM prototype? ............................... 13
    3.2.3 How can an EDBPM prototype prove useful in context of the CODIO project? .......... 13
  3.3 Research scope ....................................................................................................................... 14
4 State of the art ............................................................................................................................ 15
  4.1 Categorization of technologies ............................................................................................... 16
  4.2 The technologies’ role in the CODIO project .......................................................................... 16
  4.3 Business Process Management (BPM) ................................................................................... 17
5.4 Techniques used in combination with deviation detection .............................................. 42
5.4.1 Analyze deviation outcome ....................................................................................... 42
5.4.2 Progress overview ..................................................................................................... 44
5.4.3 Event windows .......................................................................................................... 46
5.4.4 Change in deviation severity at various depths ......................................................... 47
5.5 Single events.................................................................................................................. 49
5.5.1 Relationships between the events ............................................................................. 51
5.6 Designing the deviation scenarios ................................................................................ 51
5.6.1 Scenario #1 – Northings and Eastings ..................................................................... 52
5.6.2 Scenario #2 – Azimuth ............................................................................................ 54
5.6.3 Scenario #3 – Dog Leg Section .............................................................................. 56
5.6.4 Scenario #4 – Inclination ......................................................................................... 57
5.6.5 Scenario # 5 – Total Vertical Depth ....................................................................... 58
5.6.6 Scenario #6 – Vertical Section .............................................................................. 59
5.7 Responsibility area of the deviation scenarios............................................................... 60
5.8 Triggered actions........................................................................................................... 60
6 Implementing the prototype.............................................................................................. 61
6.1 Prototype architecture ................................................................................................. 61
6.1.1 Event generator ........................................................................................................ 62
6.1.2 Event channel .......................................................................................................... 62
6.1.3 Event processing engine ......................................................................................... 62
6.1.4 Downstream event-driven activity ........................................................................... 62
6.2 Implementing the CEP Engine ..................................................................................... 63
6.2.1 Choosing the CEP Engine ....................................................................................... 63
6.2.2 CEP Engine Architecture ....................................................................................... 64
6.2.3 Register drilling plan .............................................................................................. 67
6.2.4 Register events ........................................................................................................ 68
6.2.5 Implementing the Event windowing technique ......................................................... 68
6.2.6 Implementing the complex event rules .................................................................... 70
6.2.7 Locating the correct plan entry .............................................................................. 71
6.2.8 Building the deviation report .................................................................................. 72
6.3 The K2 BlackPearl solution ......................................................................................... 73
6.3.1 The workflow ......................................................................................................... 73
6.3.2 Implementation of process logic ............................................................................... 74
Appendix A

8

7 Experiments and results .................................................................................................................. 79

7.1 Experiment setup .......................................................................................................................... 79

7.1.1 Approach .................................................................................................................................... 79

7.1.2 Success criteria ........................................................................................................................... 80

7.1.3 Presentation of the experiment .................................................................................................. 80

7.1.4 Experiment procedure ............................................................................................................... 80

7.2 Demonstration of the prototype .................................................................................................... 81

7.2.1 Presentation of rule criteria ......................................................................................................... 81

7.2.2 Presentation of experiment data ................................................................................................ 82

7.2.3 Presentation of experiment results .............................................................................................. 82

7.2.4 Explanation of results ................................................................................................................. 83

7.2.5 Starting the K2 process ............................................................................................................... 85

7.2.6 Validating the first deviation ....................................................................................................... 86

7.2.7 Validating the second deviation ................................................................................................. 88

7.3 Experiments ................................................................................................................................ 89

7.3.1 Scenario #1 - Northings and Eastings ....................................................................................... 89

7.3.2 Scenario #2 - Azimuth .............................................................................................................. 91

7.3.3 Scenario #3 - Dog Leg Section .................................................................................................. 92

7.3.4 Scenario #5 - Total Vertical Depth ............................................................................................ 94

7.3.5 Scenario #6 - Vertical Section ................................................................................................... 96

7.4 Discussion of results .................................................................................................................... 98

7.4.1 Summary of experiment results .............................................................................................. 98

7.4.2 Reliability of the experiment results ......................................................................................... 98

7.4.3 Benefits from introducing the prototype ................................................................................... 99

7.4.4 Addressing the research questions ............................................................................................ 100

8 Conclusion and further work .......................................................................................................... 103

8.1 Conclusion ................................................................................................................................. 103

8.2 Achievements ............................................................................................................................... 103

8.3 Subjects for further research ....................................................................................................... 104

8.3.1 Additional application areas within the CODIO project .............................................................. 104

8.3.2 Improvements of the prototype ............................................................................................... 104

Appendix A Detailed experiment data .................................................................................................. 109

A.1 Scenario #1 – Northings / Eastings .............................................................................................. 109
A.2  Scenario #2 – Azimuth ................................................................. 110
A.3  Scenario #3 – Dog Leg Section .................................................. 110
A.4  Scenario #4 – Inclination ............................................................ 111
A.5  Scenario #5 – Total Vertical Depth ............................................. 111
A.6  Scenario #6 – Vertical Section ...................................................... 112
Figure 1-1: Technology research principal pattern ................................................................. 2
Figure 2-1: Collaboration based on real-time data (Johnsen, 2007) ........................................ 6
Figure 2-2: Distributed conditions for calculated value potential ............................................ 7
Figure 2-3: Work packages in CODIO ..................................................................................... 9
Figure 2-4: Phases included in directional drilling ................................................................. 10
Figure 4-1: Technologies and products involved in the CODIO project ................................. 15
Figure 4-2: A BPMN process model ....................................................................................... 19
Figure 4-3: SharePoint Platform Services (Microsoft Corporation, 2009a) ............................. 24
Figure 4-4: The use of structure and knowledge work in BPM and Case Management ............ 26
Figure 4-5: The K2 Platform (K2, 2009) .................................................................................. 28
Figure 4-6: Example of various computer systems’ use of real-time data deadlines ............... 31
Figure 4-7: Single events forming a complex event ............................................................... 33
Figure 4-8: Example of event-driven behavior ....................................................................... 33
Figure 5-1: Deviation caused by erroneous inclination value ................................................. 35
Figure 5-2: The BPMN workflow ........................................................................................... 36
Figure 5-3: An excerpt from a real-time survey report ........................................................... 40
Figure 5-4: Parameters in the survey report ......................................................................... 41
Figure 5-5: The Azimuth parameter in the survey report ....................................................... 42
Figure 5-6: Deviation in Inclination, while directed towards and away from planned TVD .... 44
Figure 5-7: Calculation of well progression ........................................................................... 45
Figure 5-8: Inclination event window ..................................................................................... 46
Figure 5-9: Deviation in Azimuth at different values of measured depth ............................... 48
Figure 5-10: Deviations in the Northings / Eastings pair ....................................................... 52
Figure 5-11: Deviation in the Azimuth. Top-down 2D-view to the left, 3D-view to the right .... 54
Figure 5-12: Deviation in the DLS and TVD parameters ....................................................... 57
Figure 6-1: Prototype architecture, based on EDA ................................................................. 61
Figure 6-2: CEP Engine class diagram ................................................................................. 64
Figure 6-3: K2 workflow model of “Handle deviation in directional drilling” ....................... 73
Figure 6-4: K2 Events; Client Event – Server Event – Mail Event ........................................ 74
Figure 6-5: Part of execution logic - validate deviation results ................................................ 76
Figure 7-1: Changes in Inclination deviation, according to experiment results ....................... 84
Figure 7-2: Changes in TVD parameter, according to experiment results ............................. 85
Figure 7-3: Screenshot - Validating the Level 3 deviation ..................................................... 86
Figure 7-4: Screenshot - Inviting participants to collaboration session .................................... 87
Figure 7-5: Screenshot - Validating the Level 2 deviation ..................................................... 88
Figure 8-1: Prediction of optimal path based on preset targets ............................................. 107
List of Tables

Table 5-1: Demonstration of how to analyze deviation outcome in Inclination ........................................ 43
Table 5-2: Single events .................................................................................................................. 50
Table 5-3: Relationship between the identified single events .......................................................... 51
Table 5-4: Rule criteria for deviation in the Northings / Eastings pair ............................................. 53
Table 5-5: Rule criteria for deviation in Azimuth ............................................................................ 55
Table 5-6: Rule criteria for deviation in Dog Leg Section .............................................................. 56
Table 5-7: Rule criteria for deviation in Inclination ....................................................................... 58
Table 5-8: Rule criteria for deviation in Total Vertical Depth .......................................................... 58
Table 5-9: Rule criteria for deviation in Vertical Section ............................................................... 59
Table 5-10: Differences in the deviation scenarios’ responsibility area ........................................... 60
Table 6-1: Functions and parameters of the CepEngine class ......................................................... 66
Table 6-2: Examples of algorithm behavior when searching for the correct plan entry ................... 72
Table 7-1: Rule criteria for deviation in Inclination ....................................................................... 81
Table 7-2: Experiment input data - deviation in Inclination ............................................................ 82
Table 7-3: Experiment results - deviation in Inclination ................................................................. 83
Table 7-4: Rule criteria designed to detect a deviation in the Northings / Eastings pair ................. 89
Table 7-5: Experiment input data - deviation in the Northings / Eastings pair .............................. 90
Table 7-6: Experiment results - deviation in the Northings / Eastings pair .................................. 90
Table 7-7: Rule criteria designed to detect a deviation in Azimuth ................................................. 91
Table 7-8: Experiment input data - deviation in Azimuth ............................................................... 91
Table 7-9: Experiment results - deviation in Azimuth ................................................................... 92
Table 7-10: Rule criteria designed to detect a deviation in Dog Leg Section ................................. 92
Table 7-11: Experiment input data - deviation in Dog Leg Section ............................................... 93
Table 7-12: Experiment results - deviation in Dog Leg Section ................................................... 93
Table 7-13: Complex event rule designed to detect a deviation in Total Vertical Depth ............... 94
Table 7-14: Experiment input data - deviation in Total Vertical Depth ........................................ 95
Table 7-15: Experiment results - deviation in Total Vertical Depth ............................................... 95
Table 7-16: Rule criteria designed to detect a deviation in Vertical Section .................................. 96
Table 7-17: Experiment input data - deviation in Vertical Section ............................................... 96
Table 7-18: Experiment results - deviation in Vertical Section .................................................... 97
Table A-1: Experiment drilling plan - Northings / Eastings ........................................................... 109
Table A-2: Experiment survey report - Northings / Eastings ....................................................... 109
Table A-3: Experiment drilling plan - Azimuth .............................................................................. 110
Table A-4: Experiment survey report - Azimuth ............................................................................ 110
Table A-5: Experiment drilling plan - Dog Leg Section ............................................................... 110
Table A-6: Experiment survey report - Dog Leg Section ............................................................. 110
Table A-7: Experiment drilling plan - Inclination ......................................................................... 111
Table A-8: Experiment survey report - Inclination ....................................................................... 111
Table A-9: Experiment drilling plan - Total Vertical Depth .......................................................... 111
Table A-10: Experiment survey report - Total Vertical Depth ...................................................... 112
Table A-11: Experiment drilling plan - Vertical Section ............................................................... 112
Table A-12: Experiment survey report - Vertical Section ............................................................. 112
List of Code Snippets

Code Snippet 6-1: Inserting a plan entry into the drilling plan .......................................................... 67
Code Snippet 6-2: Calculating a parameter deviation. ........................................................................... 68
Code Snippet 6-3: Calculating changes in a parameter deviation. ......................................................... 68
Code Snippet 6-4: Calculating well progression. .................................................................................... 68
Code snippet 6-5: The Event Window struct.......................................................................................... 69
Code snippet 6-6: Identification of a suspicious survey reading............................................................. 69
Code snippet 6-7: Calculation of average values in the event window. ................................................. 70
Code snippet 6-8: Adding survey readings to the event window. ............................................................ 70
Code Snippet 6-9: Implementation of a complex event rule. ................................................................. 71
Code Snippet 6-10: Declaring current and previous drilling plan entries............................................ 71
Code Snippet 6-11: Aborting the search for a corresponding plan entry. ............................................. 71
Code Snippet 6-12: Algorithm to locate a corresponding drilling plan entry. .................................... 72
Code snippet 6-13: Creating a connection to the K2 process................................................................. 77
Code snippet 6-14: Initiating an instance of the K2 process................................................................. 77
1 Introduction

This chapter aims to motivate the reader through an introductory overview of the purpose and objectives of this thesis, including a short description of the original problem statement, a brief project description, the research method used, and the structure of the thesis.

1.1 Motivation
Drilling a well through the sea bed and several thousand feet into the earth, both vertically and horizontally, in order to finally reach a target destination, is a task filled with complexity. There are a lot of factors to consider in the process of directional drilling, and many threats to be aware of. Deviations from the original drilling plan might easily occur, bearing a varying degree of consequences. Currently, directional drillers stationed at the oil rig are responsible for manually analyzing incoming survey readings to properly assess whether a deviation has occurred or not. The survey readings consist of a range of parameters explaining the exact current position of the well. Correct analysis and understanding of the significance of these parameters, both separately and combined, is a necessary though complex process. Also, being able to figure out and initiate the appropriate follow-up actions after a deviation has occurred, based on the severity of the deviation, require much contextual knowledge.

An automatic service, based on event-driven behavior, could ultimately assist workers in respect to:

1. Detect occurring deviations, and
2. Suggest proper follow-up decisions based on the available information.

1.2 Problem statement
The following summarize the original problem statement:

The topic of the thesis concerns process-oriented systems handling real-time data and reacting to external events. The candidate shall briefly explain two relevant technology areas, Business Process Management and Complex Event Processing, and explore synergies between them. Architecture for event-driven BPM shall be proposed and demonstrated with a concrete example within context of the CODIO project (BPM for collaboration in oil and gas drilling operations, based on Microsoft technology).
1.3 Project description
This thesis is written as part of my Master of Science degree at the University of Oslo. The work performed in the thesis is included in the CODIO research project, a part of the oil and gas industry’s investment in Integrated Operations.

The thesis is realized through Computas AS, a Norwegian IT consultant company delivering IT-solutions and services to organizations both in public and private sectors. Computas has obtained profound experience from participating in a range of relevant research projects, addressing issues and investigating improvement possibilities throughout the oil and gas industry.

1.4 Research method
The research conducted in this thesis is based on a technology research method presented by Ketil Stølen and Ida Solheim at Sintef (Stølen, Solheim, 2007). The method includes principals from traditional research methods, although with a slightly different approach. While traditional research methods focus on questions meant to explore the real world, technology research investigates the possibility of producing a new or improved artifact to satisfy a need from the real world. As seen in Figure 1-1 below, the technology research method emphasizes research to be conducted according to the following principal pattern:

- The identification of a need, through problem analysis.
- The invention of a new or improved artifact, through innovation.
- Comparison of the new artifact against the identified need, through evaluation.

```
Figure 1-1: Technology research principal pattern
```

The research done in this thesis has been conducted through iterations of this pattern until a positive evaluation of the artifact was obtained, as advocated by the technology research method principals.
1.5 Thesis document structure

The thesis document is structured with inspiration from the technology research principals described above. The content of the thesis document is presented within three parent sections, as described in detail through the following:

- **Problem analysis** – contains three chapters providing background information, research goals and an introduction to relevant technologies and disciplines.
  - *Chapter 2: Background* – gives an introduction to the Integrated Operations, the CODIO research project, the method of directional drilling, and an insight in related research.
  - *Chapter 3: Research* – presents what we want to achieve through the research performed in this thesis, describing the overall research goal accompanied with research questions and a research scope.
  - *Chapter 4: State of the art* – aims to give the reader a basic, although sufficient understanding of some of the various existing technologies and products used in the CODIO project and this thesis.

- **Innovation** – contains two chapters describing our case scenario, including design and implementation of the prototype.
  - *Chapter 5: Case: Deviations in directional drilling* – elaborates the case scenario, comprising the scope of prototype functionality, the identified deviation scenarios, and the design of expected prototype behavior.
  - *Chapter 6: Implementing the prototype* – gives a thorough insight in how the prototype is implemented, in regards to architectural design and the technical aspects of the implementation.

- **Evaluation** – contains two chapters presenting experiments and discussion of results, as well as concluding statements.
  - *Chapter 7: Experiments* – gives an overview of experiments conducted from all deviation scenarios described in the case, along with a presentation and discussion of the experiment results. The chapter also demonstrates the prototype through a concrete deviation scenario retrieved from the case description.
  - *Chapter 8: Conclusion and further work* – concludes the work and summarizes achievements from research. This chapter also includes suggestions on further research and improvements of the prototype.
Problem Analysis
2 Background

This chapter gives an insight in background information relevant to the thesis. Topics include Integrated Operations, the CODIO research project, directional drilling, and related research.

2.1 Integrated Operations
The northern sea bed contains several known areas of interest in regards to recover oil and gas resources. Lots of oil rigs have been established within the most promising areas since the 1960’s, many of which are still currently performing drilling operations (Wikipedia, 2010b).

*Integrated Operations (IO)* is a collective term describing an initiative to introduce new working methods in the oil and gas industry, based on increased use of modern information- and communication technology (Seehusen, 2008). The main objectives of IO are to enhance oil production, reduce overall costs, and extend the lifetime of oil reservoirs at the Norwegian continental shelf (Gonzales et al., 2005).

The IO approach has been named differently by various actors throughout the industry, with terms like “eOperations”, “Smart Fields”, and “iField” to mention a few. “Integrated Operations” is the term provided and used by the Norwegian Oil Industry Association (Oljeindustriens Landsforening), and will also be the term used in this thesis.

2.1.1 IO Generations
The implementation of IO in the oil and gas industry is divided into two *generations*, called G1 and G2 (Oljeindustriens Landsforening, 2005).

G1 focus primarily on using information- and communication technology to integrate processes and workers onshore and offshore. To achieve this objective, the implemented technology has to ensure that onshore workers’ ability to support offshore operations is greatly improved. Central in G1 is the introduction of extensive real-time data transmissions between oil platforms offshore and operation centers onshore, as illustrated in Figure 2-1. The data sent back and forth can be shared and analyzed by workers in real-time, independent of geographic locations, making it possible for offshore personnel to collaborate with groups of experts onshore. For instance, rather than having geology experts stationed on each oil platform, a group of geology experts may assist several oil rigs simultaneously while stationed onshore. The real-time data can be presented in ways that support decision making, allowing groups of experts with backgrounds in various disciplines to make decisions in a quick and effective manner.
The second generation, G2, aims to help operators make more effective and extended use of vendors’ knowledge and services. The goal is to develop digital services, providing a large amount of the functionality required to operate a field remotely. Achieving this will largely affect production, oil recovery, costs, and safety. Implementation of G2 is expected to demand significant organizational changes, such as redefined relationships between operators and vendors, as well as development of new data standards.

2.1.2 Value potential
There are many motivating factors for implementing IO. However, the most dominant factor is undoubtedly the long-term economic gain. According to a report from OLF, the total investment in IO is estimated to sum up to 24 billion NOK (Oljeindustriens landsforening, 2007). By comparison, the expected value potential is approximately 295 billion NOK, which corresponds to a total value increase of 9.6% as compared to the situation without IO. As can be seen in Figure 2-2, the calculated value potential is based upon revenues from increased reservoir extractions and accelerated production, combined with reduced drilling and operational costs. In order to meet the conditions required to fulfill this value potential, a range of strategic improvements must be met:
• **Reduced Drilling costs – 7 % of calculated value potential:** Improved long-term planning of each well path, more accurate drilling, real-time optimization of drilling processes, fewer personnel being sent offshore, and constant involvement of service providers.

• **Reduced Operational costs – 15 %:** Condition-based maintenance, administrative activities onshore, remote operations, increased automation in surveillance, and new field support methods.

• **Increased reservoir extractions – 58 %:** Continuous collaboration between disciplines, consistent production- and reservoir data, real-time updates of reservoir models and data, and methods to locate the optimal placement of wells.

• **Accelerated production – 20 %:** Use of dynamic simulations when analyzing production and processes, tools for analyzing critical work processes, and operation centers to support work processes.

![Figure 2-2: Distributed conditions for calculated value potential](image)

2.1.3 Benefits

In addition to the major economic potential, implementing IO will provide the industry with a wide range of other expected benefits (Statoil, 2008), including:

• Improved conditions in regard to HMS.
• More efficient drilling operations.
• Better placement of wells.
• Increased control of production and reservoir.
• More efficient maintenance and monitoring of equipment.
• Enhanced utilization of resources.
• Increased uptime.
2.1.4 Risks and challenges

There are certainly some risks and challenges connected to the introduction of IO. With increased use of the internet, threats and challenges emerge that traditional administrative systems already have been struggling with for quite a while (Sintef, 2007). Trojans, viruses, hackers and other types of external attacks may prove to become an issue, even though a logical separation is implemented. These issues have never been a relevant problem in the oil and gas industry before. In order to successfully introduce increased use of IT, proper precautions should be considered in order to avoid such risks.

Another challenge is to make organizations adapt to changes in administration of various systems. Workers from the automation industry need to collaborate with workers from the IT industry, and a mutual understanding of each other’s working methods must be established. All participants involved in IO need to develop a culture of security, defining common attitude, knowledge and values.

2.2 CODIO

CODIO, or Collaborative Decision Making in Integrated Operations, is a research project included in the IO initiative. The project started in 2007, and is planned to last for a total of 3 years (Iris, n.d.). The CODIO project is funded by ConocoPhillips and the Norwegian Research Council’s Petromaks program, with participants including Computas (as project coordinator), Odfjell Drilling, Kongsberg Intellifield, International Research Institute of Stavanger (IRIS), Institute for Energy Technology (IFE), the IO-center at the Norwegian University of Science and Technology (NTNU), and the universities of Stavanger and Oslo.

The project focuses on developing a software product intended to assist workers in decision-making processes, based on cutting edge Bayesian Decision theory, combined with support for collaborative work processes (Fjellheim, Bratvold, Herbert, 2008) (Computas, n.d.a). The goal is to help workers make the right decisions when uncertainties occur in the drilling process, by providing updated recommendations on proper actions, depending on each particular situation (Computas, n.d.b). By introducing a good analysis implementation based on Bayesian Decision theory, it will become more trivial for experts to observe correlations between available data and probable outcomes, as well as allowing the system to recognize and recommend what steps should be taken to reach the expected optimal outcome. Recommendations provided by the system will change dynamically, according to ongoing analysis of freshly obtained data. To achieve all this, the data-feed from drilling operations need to be correctly filtered and prepared for presentation. Sufficient overview of the relevant data available in a critical situation makes it easier to identify the right decision. Tools implemented in CODIO will allow complete visualization of this information.

The CODIO product is gradually introduced to an Onshore Drilling Center (ODC) operated by ConocoPhillips Norway, supporting drilling operations at the Ekofisk field (Fjellheim et al., 2010).
2.2.1 Work packages

The CODIO project is divided into three work packages, WP0 – Project Management, WP1 – Collaboration Support, and WP2 – Decision Support (see Figure 2-3):

- WP0 concerns overall administration, project coordination, and management.
- Computas is responsible for developing and maintaining WP1. The purpose of WP1 is to implement a complete collaboration model for analyzing work processes.
- WP2 focus on providing support for decision making. Research activities in Bayesian Decision theory are conducted by IRIS in order to achieve viable and good decision making in drilling operations.

WP1 is using WP2, as illustrated in Figure 2-3. More specifically, WP1 aims to utilize the data continuously generated by WP2, by properly presenting updated decision recommendations in a user-friendly interface, allowing groups of experts to collaborate on discovering the optimal decision. The prototype developed in this thesis is planned to be incorporated as an extension to the WP1 product.

![Diagram showing the relationships between the work packages in CODIO](image)

Figure 2-3: Work packages in CODIO
2.2.2 Use of existing technology

The system developed in CODIO WP1 is built upon a variety of existing technologies and products. Many of the IT-systems used at the ODC facility at ConocoPhillips are based on the Microsoft platform (Fjellheim et al., 2010). K2 Blackpearl is chosen as the Business Process Management platform, based on K2’s compatibility towards relevant Microsoft technology. Also, K2 provides a case framework solution, supporting case management capabilities needed to address the knowledge intensive activities found in drilling operations. A comprehensive portal is created using Microsoft SharePoint, providing functionality like content management, document handling, and the overall graphical user interface. Microsoft Office Communicator is utilized through SharePoint web-parts.

Combining these technologies and products is beneficial in several ways. First of all, it makes it possible to display all content in one single user interface. Secondly, the system will be accessible from any computer with an internet connection. Finally, it will not be necessary to dedicate project resources to develop said functionality, as it is already available through using these established technologies and products.

Chapter 4 provides a further description of these mentioned products and technologies, as well as their role in the CODIO WP1 product.

2.3 Directional drilling

The approach of directional drilling differs from traditional vertical drilling, as the well at some point in the operation enters a phase where it is continuously directed horizontally towards the target zone (Wikipedia, 2010d). It is not uncommon to maintain this phase until the well is directed completely on the horizontal plane, leading to horizontal drilling (Norsk Oljemuseum, n.d.). Figure 2-4 illustrates the phases included in directional drilling.

![Figure 2-4: Phases included in directional drilling](image)
During directional drilling, the actual drill string has to remain stationary. Instead, a down-hole motor makes the drill bit at the end of the well rotate, powered by a drilling fluid called mud being pumped down the drill string. The direction and angle may be changed continuously during drilling, by using steerable motors controlled by a computer.

The practice of directional drilling started to develop at the end of the 1980’s (Norsk Oljemuseum, n.d.), bearing benefits that include:

- The introduction of the ability to reach reservoirs being difficult or impossible to access vertically.
- The possibility to reach a large area of the oil and gas reservoir by drilling multiple wells from one single platform installation.
- Some reservoirs have oil contained within thin layers. Directional drilling makes it possible to approach these layers at the most promising direction and angle, increasing the exposed section length of the reservoir and boosting the recovery rate of resources.

### 2.3.1 Measurement While Drilling (MWD)

A range of sensors equipped on a tool called Measurement While Drilling (MWD) are measuring the drilling situation during directional drilling. The MWD tool provides a variety of information (Wikipedia, 2010c), including:

- The current geology formation surrounding the well (like density and porosity).
- Overview of the drilling mechanics (like rotational speed of the drill-string and mud flow volume).
- The fluids that are contained inside the rock (water, oil or gas).
- The exact position of the well.

The measured data is transmitted to the surface by using mud pulse telemetry and electromagnetic telemetry, allowing consecutive measurements to be taken in real-time without disturbing the drilling operation.

Directional information obtained from MWD measurements, also called survey data, indicate if the current direction conforms to the predefined drilling plan. Deviations from the drilling plan may be corrected by applying different techniques, depending on the severity of the deviation. The techniques range from simply adjusting the rotation speed or drill string weight, to more complicated methods such as introducing a down-hole motor.

There is a continuous focus on developing new technology to better utilize the data retrieved from the MWD tools (Norsk Oljemuseum, n.d.).
2.3.2 Geosteering

The measured data retrieved from the MWD tool is often used by geologists in order to keep the well inside preferred geology formations, based on certain criteria (Helge Rørvik, Halliburton, personal communication 30.03.10). This practice is called geosteering. Geosteering is particularly relevant when drilling inside the reservoir, in order to successfully identify and enter specific areas with high exposure of resources (Egil Aanestad, Halliburton, personal communication 04.12.09).

Geosteering is based on the geological measurements obtained from the MWD tool. The purpose of geosteering is to reduce the risk of exposing water and gas to the producing well, and to maximize the economic production (Schlumberger, 2010).

2.4 Related research

Literature search during the work on this thesis has led to the discovery of some projects conducting research on how to implement EDBPM, as well as how the approach can further improve strategic gain in business process models. This section will briefly present two of the most relevant of these research projects.

2.4.1 EDBPM in financial services

Hans-Martin Brandl and David Guschakowski have designed a use-case that investigates how to apply event-driven BPM to an online-based credit application named easyCredit (Brandl, Guschakowski, 2007). The research is conducted in context with their Diploma Thesis at the University of Applied Sciences Regensburg in Germany.

The thesis analyze and evaluate various Complex Event Processing- (CEP) and Business Activity Monitoring (BAM) tools, as well as different Event Processing Languages (EPL), including “AptSoft”, “TIBCO”, “StreamBase”, “Apama”, and “Aleri” (Ammon et al., 2008). The purpose of the research is to introduce real-time monitoring of credit processes at a German company called TeamBank. More specifically, the use-case is focused on the potential of credit applications being cancelled, as well as effective monitoring of performance and status of components included in the business logic. The event-driven implementation is realized by using BPM and CEP in combination with a BAM tool, providing a detailed visual insight in the status of both single and complex events. Potential sales loss caused by cancelled credit applications will be detected and displayed in an interface managed by the BAM tool.

2.4.2 EDBPM in logistics

Florian Springer and Christoph Emmersberger have, in collaboration with the German company Deutsche Post AG, developed a use-case for introducing EDBPM in logistics (Springer, Emmersberger, 2008). The use-case has explored the possibilities of combining CEP and BAM with the latest Sopera Enterprise Service Bus (ESB) infrastructure technology, an approach to manage business and IT services as a distributed Service Oriented Architecture (SOA) (Ammon et al., 2008). The use-case proposed in the thesis implements a combination of advantages from SOA and CEP. Capabilities of Sopera are utilized in order to send information regarding process- and service status as event streams, received directly by the CEP and BAM components. Complex events generated by the CEP engine, based on predefined patterns, are displayed through the interface provided by the BAM component.
3 Research

3.1 Research goal
This thesis focuses on how to introduce Event-Driven Business Process Management to the Integrated Operations. The goal is to develop a prototype based on Event-Driven behavior using BPM combined with the techniques of CEP. The purpose of the prototype is to be capable to sense and react to real-time data in the context of drilling operations. The work conducted in the thesis will include a review of relevant topics in state of the art, the identification and complete definition of a viable case scenario related to the work done in the CODIO research project, the development of a functional Event-Driven BPM prototype, and experiments conducted to demonstrate the final prototype. The functionality implemented in the prototype will be designed to meet the requirements defined in the description of the case scenario.

3.2 Research questions
This section will discuss a couple of research questions, pinpointing the specific research topics of interest in context of this thesis. The research questions are essential in the process of having a structured focus on the research goal, as they represent the questions we want to answer through performing the research.

3.2.1 What synergies can be found between BPM and CEP?
We want to investigate how a combination of BPM and CEP can provide beneficial synergies, with the ultimate purpose to develop a functional Event-Driven BPM prototype. We will explore how these technologies together can contribute towards a successful implementation of event-driven behavior.

3.2.2 What is a viable architecture for an Event-Driven BPM prototype?
In order to successfully develop a usable Event-Driven BPM prototype, we need to conduct research on how to properly design a viable architecture. Relevant disciplines, technologies and products must be identified and understood in order to figure out how the prototype could be engineered.

3.2.3 How can an EDBPM prototype prove useful in context of the CODIO project?
It will be necessary to pinpoint an application area for the prototype in context of the CODIO project. The prototype is supposed to be utilized in combination with actual tasks included in ConocoPhillips’s drilling environment, providing real-time sensing capabilities that improve the accomplishment of said tasks. A relevant case scenario within the frames of the CODIO research project should be identified and described into detail. The prototype should be demonstrated with concrete examples according to this case scenario.
3.3 Research scope

A lot of research and work needs to be performed in order to fully realize the concept of event-driven behavior in a real world setting of directional drilling. Drilling operations are filled with complexity, making it important to investigate and analyze all aspects regarding the situation thoroughly. Also, event-driven BPM is a newly emerging discipline, and as far as we know, no similar or relevant research has been conducted within context of this application domain.

The work performed in this thesis starts from scratch, and should by that be considered a first attempt to find a suitable approach to introduce event-driven behavior to the process of directional drilling. Thus, it should be noted that the prototype developed during this research in no way should be considered a final working product, but rather as an attempt to create a suitable starting point for further research.

As a realistic scope to this thesis, the prototype will be developed with the objective to react and respond correctly according to concrete scenarios that is defined and described in detail through the case description in chapter 5. These scenarios describe the different deviation situations that might occur in directional drilling. Finally, the prototype will be experimented upon based on experiment data designed to trigger each particular deviation scenario.
4 State of the art

This chapter aims to give the reader a basic, although sufficient understanding of the various technologies, disciplines and products used in the CODIO WP1 project and this thesis. Most of the text compiled in the chapter is based on information found in scientific literature. The literature used is primarily retrieved from books, organizational websites, online scientific papers, and internet based encyclopedias.

As mentioned in section 2.2.2, the CODIO product is developed in conjunction with an assortment of existing products and technologies. A complete overview of products and technologies central in the WP1 part of the CODIO project and this thesis is shown in Figure 4-1 below. The figure also describes how all the technologies relate to each other, and what role they play in the CODIO framework.

![Figure 4-1: Technologies and products involved in the CODIO project.](image-url)
4.1 Categorization of technologies
As seen from the figure, the technologies are presented within three different boxes, distinguished by normal, dashed and bold surrounding lines. This categorization is intended to represent the following:

- **Normal** lines indicate that the respective technology is used both in the CODIO WP1 project and in this thesis. These technologies comprise the interaction between the prototype developed in this thesis, and the product engineered by the CODIO WP1 research team. More specifically, the prototype is developed with means to automatically initiate a BPM process created in the CODIO WP1 project, through the K2 BlackPearl platform.

- **Bold** lines represent the technologies and disciplines introduced to the CODIO WP1 project through this thesis. These elements, consisting of Complex Event Processing, Real-time data, and Event-Driven BPM, are exclusively used to develop the prototype functionality.

- **Dashed** lines distinguish technologies and products central in the CODIO WP1 project that have low relevance to this thesis. These technologies include Microsoft SharePoint, Microsoft Communicator, K2 CASE Framework, K2 Connect, and Case Management.

4.2 The technologies’ role in the CODIO project
The following summarizes the specific role of each technology and product included in the CODIO WP1 project:

- **K2 BlackPearl** is being used as the Business Process Management solution, which involves the handling of BPMN workflow modeling and execution of business process steps.

- The Case Management solution in CODIO is realized through K2’s Case Management Framework, a feature included in K2 BlackPearl.

- **Microsoft SharePoint** is used to manage content, handle documents, and enhance collaboration abilities, through a portal-based user interface. Sharepoint presents information and data specific to each unique oil and gas well, through a custom-made concept called well-sites. The portal capabilities are realized through a deep integration with K2 BlackPearl.

- **Complex Event Processing** combined with Business Process Management works as the foundation in the Event Driven Business Process Management prototype; the main subject of research inside the scope of this thesis.

- **Real-time survey data** from drilling wells is received and processed by the CEP implementation of the Event-Driven BPM Prototype. Selected data from this real-time feed is analyzed in the search for complex events.

- **K2 Connect** provides connectivity methods between K2 BlackPearl and external components such as SharePoint services, through its implementation of SmartObjects.

- **Microsoft Communicator** offers tools and functionality supporting enhanced collaboration between employees. These tools are implemented to give the workers a reliable overview of when co-workers are online and available. Workers may also invite colleagues into collaborated work sessions where the same information is displayed on all participants’ monitors.
4.3 Business Process Management (BPM)

A process defines the various steps that must be taken in order to reach a certain goal. Each of these steps often includes a set of activities and a description of which roles, artifacts and people are involved.

There exist a lot of different definitions trying to explain what processes are all about. Armistead et al. describes them as “... a generic factor in all organisations. They are the way things get done.” (Armistead et al., 1999, p. 10). BPM considers processes as a means of understanding and explaining business activities.

BPM can be viewed as “a systematic, structured approach to analyze, improve, control, and manage processes with the aim of improving the quality of products and services” (Elzinga et al. 1995, p. 1). It is considered an extension of traditional Work Flow Management Systems and approaches (Subramanian, 2005). A Work Flow Management System is “a system that defines, creates and manages the execution of workflows through the use of software, running on one or more workflow engines, which is able to interpret the process definition, interact with workflow participants and, where required, invoke the use of IT tools and applications” (cited in Van der Aalst et al., 2003, p. 4).

BPM was created based on a strong need to streamline internal processes between internal and external functions (Subramanian, 2005). The technology is being used to model, monitor, manage and manipulate processes quickly according to changes in strategy and market forces. One of the most central and important parts of BPM is called process modeling, which is a technique used to visualize process models in a diagram. Business Process Modeling Notation (BPMN) is a language engineered for the purpose of modeling processes in BPM.

4.3.1 Business Process Modeling Notation (BPMN)

Business Process Modeling Notation is considered a structured and consistent way of understanding, documenting and analyzing business processes and all involved resources (Recker et al., 2006). The notation was originally developed by the Business Process Management Initiative (BPMI), with the first version, BPMN 1.0, released in May 2004. Since then, BPMN has been continuously maintained by the Object Management Group (Wikipedia, 2009a). The Object Management Group has released all subsequent versions of BPMN, which includes the BPMN 1.1 and BPMN 1.2 specifications, as well as the current version, BPMN 2.0, that was released in August 2009 (Object Management Group, 2009).

BPMN, with its graphical notation for specifying and visualizing business processes in a Business Process Diagram (BPD), has become a standard for business process modeling. The notation has been specifically designed to coordinate the sequence of processes and messages that flow between different process participants in a related set of activities (Business Process Management Initiative, 2004). Using BPMN requires no license or fee, and some of the best BPMN tools are free of cost (Silver, 2009a).
4.3.1.1 Purpose of BPMN
From the beginning, BPMN was designed to be business friendly, as opposed to already existing modeling standards that were considered as too IT-centric. Thus, BPMN was created with elements and artifacts already familiar to business, like swim-lanes, boxes and diamonds connected by arrows. BPMN presents a standard notation that is well understandable by all participants involved in the business process modeling, making it a common language for describing process behavior that is shareable by business and IT.

4.3.1.2 Differences from traditional workflow modeling
While BPMN's similarity to workflow makes it familiar to business, its true advantages lie in features missing from traditional workflow modeling. For example, each element and artifact in the process model is precisely defined, and there are strict rules preventing incorrect modeling. This ensures unambiguous and understandable process models that are easy to interpret by others without further explanations by the modelers. Another feature missing from traditional workflow modeling is the support for event-triggered behavior. An event is something that happens while the process is running, like a customer calling to change an order, or when an item appears to be out of stock. A real business process is expected to handle such exceptions. BPMN lets the modelers design processes to manage all kinds of exceptions that may or may not occur while executing the process model.

4.3.1.3 Modeling shapes
There are only three primary modeling shapes in BPMN:
- The Activity shape, which is presented as a rounded rectangle,
- the Gateway shape, shown as a diamond, and
- the Event shape, distinguished by a circle.

In order to achieve sufficient expressiveness and precision, a great number of subtypes of each of the mentioned shapes are supported. The subtypes are distinguished by how the border style of the shape is drawn, where the shape is placed in the process model diagram, and what symbol is used inside the shape, chosen from a wide assortment of predefined symbols available.

Each shape in BPMN's toolbox has precise operational semantics, meaning that the execution of the process language is described directly within the shapes. This makes the process model standardized and subject to automated validation. Technical details are also specified, enabling the model to be executed as an automated workflow.

In order to fully understand how process modeling with BPMN work, we need to get more familiar with the various elements and artifacts BPMN has to offer. We will take a look into the BPMN toolbox to get a brief summary of how the most relevant artifacts and elements behave and function in the model. Figure 4-2 below illustrates a BPMN process containing commonly used workflow elements.
The workflow elements present in this model can be explained as follows:

- **Pool**: A pool is in *BPMN 1 versions* (BPMN 1.0, 1.1, and 1.2) specified as a container for a process. In BPMN 2.0 however, it is defined as a collaboration diagram, showing the message flow between the process and external participants. The pool shape is displayed as a rectangular box, either horizontal as shown in Figure 4-2, or vertical. The pool is technically representing a participant in collaboration with the process. In BPMN 1 versions, diagrams may contain a single process enclosed within a pool, while a diagram in BPMN 2.0 must contain either none or more than one pool. If a process is enclosed within a pool, it is called a *white-box pool*, meaning a pool where the content is visible. If the pool consists of a participant, like a role or a business entity, it is called a *black-box pool*. A black-box pool has invisible content.

- **Lane**: A lane is a subdivision of a process. In BPMN 1 versions, lanes were optional and frequently omitted, while in BPMN 2.0, at least one lane is always assumed to exist by default in the model, even if not drawn in the diagram. Like with pools, lanes are drawn as
rectangular boxes. In BPMN 2.0, lanes are used as a way to organize flow elements belonging to different categories.

- **Task:** A task is an *atomic activity*, meaning an activity with no internal subparts defined by the model. It is represented by a rounded rectangle, known as an activity shape. There are a number of task types defined by BPMN, with *user-* and *service* task being the two most commonly used. A user task is a human activity, while a service task is an activity that is performed automatically by the system. Tasks represent work performed, and not functions or states.

- **Subprocess:** A subprocess is a *compound activity*, meaning an activity that contains a flow of activities as subparts. These subparts represent a process at a lower level, hence the term “subprocess”. Subprocesses can be visualized in a number of ways in the diagram. A collapsed subprocess is drawn using a regular activity shape in combination with a “+” symbol displayed at the bottom center inside the shape. An expanded subprocess, also known as an inline subprocess, is represented by displaying the subprocess activity flow within an enlarged activity shape. Subprocesses can also be expanded into separate child-level diagrams, either within the same model or in an external model.

- **Event:** Events are represented by a circle shape. They indicate that something is happening during the course of a business process. BPMN provides three different types of events, *start*, *end*, and *intermediate*, and they are used based on when they affect the flow (White, 2004). Each of these events may have a symbol attached, making it possible to differentiate between various causes, called *triggers*, or impacts, called *results*, of the event being invoked (Silver, 2009a). There are a wide range of such symbols available in the BPMN toolbox. The most commonly used are called *Message*, *Timer*, and *Terminate*.
  
  - A *start event* is drawn as a thin circle, and signifies the start of a process or a subprocess. In a process, the symbol inside the start event represents the trigger, meaning the signal that creates a new instance of the process. If a start event is triggered by a white envelope symbol, which represents a Message start event, it means that the process is triggered by a signal outside the process, like a customer submitting an order. If the trigger is a clock symbol, known as a Timer start event, it means the process is running on some predetermined schedule. Subprocesses must be triggered with a *None* start event, which means that the trigger is unspecified.

  - An *end event* is drawn as a thick circle, indicating the end of a path in a process or a subprocess. It is often good practice to use more than one end event to identify distinct end states, in order to handle possible exceptions from the normal process flow. If a process or subprocess includes parallel paths, each path must reach an end event to complete the instance. The symbol drawn inside end events indicate which result signal is thrown when the end event is reached. A None end event means that no result signal is thrown. A Message end event (black envelope symbol) signifies that a message is sent, like a response message to a customer. A Terminate end event (bulls-eye symbol) is a special case. Reaching terminate immediately ends the process or subprocess on that particular process level, even if other parallel paths are still running.
Intermediate events are events that occur during a process or subprocess, meaning somewhere between the start and end of the respective process or subprocess. These events may either be interrupting, drawn with a double dashed border, or non-interrupting, drawn with a double solid border. The intermediate event is used to model many different types of process behavior, depending on the combination of the symbol inside, the event’s border style, and the placement in the diagram. While end events are only able to throw signals, and start events may only catch signals, intermediate events can do both.

- **Gateway:** A gateway, drawn as a diamond shape, represents a control point in the sequence flow. It is used for making decisions, as well as forking, merging and joining of paths, all determined by the symbol inside (White, 2004).
  - An exclusive gateway, also called a XOR gateway, with one sequence flow in and more than one sequence flows out, represents an exclusive decision (Silver, 2009a). Only one of the output sequence flows is to be followed, based on some condition. The XOR gateway may be drawn without any symbol, or optionally with a cross symbol inside.
  - A parallel gateway, having one sequence flow in and more than one out, signifies a parallel split, also known as an AND split. The purpose is to follow all outgoing sequence flows unconditionally in parallel. The AND split is drawn with a ‘+’ symbol inside the diamond shape.

- **Sequence flow:** A sequence flow is a solid line connecting activities, gateways, and events within a single pool. The sequence flow shows the sequential processing of the elements contained in the process model. When the element at the tail of the sequence flow is processed, the element at the arrowhead is enabled to start.

- **Message flow:** A message flow is represented by a dashed line with an open arrowhead. It is used to show the flow of messages between two process participants. It cannot be used to connect two elements within a single pool.

- **Text annotation:** The text annotation is an artifact used to link arbitrary text to objects located in the process diagram. It is represented by an enlarged left square bracket symbol.

### 4.4 Complex Event Processing (CEP)

In computing, an *event* can be defined as "anything that happens, or is contemplated as happening" (Luckham, Schulte 2008). It can be everything from the registering of a key stroke or a change of state of a value residing in a database, to a sensor reading or an airplane landing. Thus, all kinds of information systems, from web browsers to mobile operating systems, are in some way driven by events (Luckham, 2002). Network events are different from database events, which again are different from financial trading events. A *complex event* is an event that could happen only if a combination of other events happened.
CEP is a relatively new, emerging technology used to process, analyze and control multiple complex series of interrelated events from various modern distributed sources, in order to extract useful information from combinations of such events (Mendes et al., 2008). The technology makes it possible to achieve situational knowledge from distributed systems and data sources in real-time scenarios (Paschke, 2008).

CEP is based on a combination of old and new simple techniques (Luckham, 2002). Some of these techniques are well known in other types of computer applications, like for example rule based systems in intelligent programs. One of the new techniques CEP has to offer is functionality for tracking of event- and event relationship histories, in order to recognize the presence of complex events representing multiple simpler events flowing in the system.

The goal of CEP is to make it easy to keep track of the information flowing through all the layers of an IT infrastructure (Complex Event Processing, n.d.). Events containing this information may be discovered, understood, and acted upon in real-time. This may help IT professionals understand what is happening within the system, making it easy to quickly identify and solve problems, and to effectively utilize events for increased performance and security, avoiding threats and identifying opportunities. Users of high-end CEP-based systems are allowed to specify and modify what information is of interest, ranging from low level network processing data to high level enterprise management intelligence (Luckham, Frasca, 1998).

### 3.2.2 Application area

CEP is used for building and managing a lot of different types of information systems, including:

- Business Activity Monitoring
- Business Process Management
- Enterprise Application Integration
- Event-Driven Architectures
- Network and business level security
- Real time conformance to regulations and policies

The technology can be applied to a wide range of challenges in information systems, like automation of business processes, scheduling and control of processes, network monitoring and performance prediction, and also intrusion detection. Complex Event Processing can complement and contribute to other technologies, such as Service Oriented Architecture (SOA) and BPM.

### 4.4.1 Relationships between events

As mentioned above, CEP does not only operate on sets of events, but also on relationships between events (Luckham, Frasca, 1998). Events in a distributed system may have various relationships to each other. The relationships can be specified in event patterns in maps and filters, such as *time*, *causality*, and *abstraction*.

- **Time**, with respect to a system clock, means that event A happened *before* event B. (Luckham, Schulte, 2008).
- **Causality** describes a relation where one of the events caused the other one to happen, such as event A *caused* event B to happen. Events without a causality relationship are called *independent*.
- An event is an abstraction of a set of events if it represents that set of events.
4.4.2 CEP tiers

Applications based on CEP principals may be categorized into four tiers (Information Management, 2006). These tiers can be explained to the following:

- **Tier one**: Represents applications using simple complex event processing. Typically, only one event stream is used, and only one message from the stream is processed at the time. The event processing in tier one often involves the detection of a complex event pattern in order to notify a specific person or application. Custom coding of tier one applications may be a better choice than using existing CEP Engine software, as the implementation is usually easy and straightforward.

- **Tier two**: Includes more than one event stream, and in some cases also stored data. The event streams are more complex than in tier one, and may require quite sophisticated code to handle properly. Complex functions like union, merge, and join must normally be supported.

- **Tier three**: Involves complex pattern analysis across multiple event streams. Tier three requires the ability to identify complete sequences of event occurrences at a rapid pace. Multiple parallel threads need to be maintained, and there should be a good control of memory usage.

- **Tier four**: Needs to handle distributed event processing capabilities across multiple enterprise applications. Required functionality includes multi-threading, clustering and pipelining.

4.4.3 Our need for CEP capabilities

The capabilities offered by CEP will be a vital part of our approach in implementing an event-driven BPM prototype. Our solution requires the use of various CEP principles, such as event categorization based on values continuously received from real-time data input, and the ability to compare such events within a predefined set of complex event patterns.

The CEP functionality used in the prototype is categorized as *tier one*, since we are only going to perform simple event processing using one event stream. We will operate with a small amount of relevant events, all received simultaneously in batches from the MWD tool.

All events and complex event rule patterns designed in this thesis, as well as the relationship between the events, will be thoroughly discussed during the case description in chapter 5.

4.5 Microsoft SharePoint

Microsoft SharePoint, also known as *Microsoft SharePoint Products and Technologies*, is an integrated platform developed to optimize how people, content, processes, and business applications work together (Microsoft University, 2009). SharePoint holds a collection of products and software elements, including a platform for document management, modules for process management, functionality for web browser based collaboration, and various search modules (Wikipedia, 2009d). IT professionals and developers are provided with the tools and platform they need for server administration, application extensibility, and interoperability.
4.5.1 SharePoint Platform Services

The SharePoint platform provides a number of services helping to improve organizational effectiveness (see Figure 4-3). Content management and enterprise search, shared business processes, combined with extensive information sharing, are among the services that ensure a good business insight is maintained (Microsoft Corporation, 2009a). Here follows a brief summary of the SharePoint Platform Services:

- **Collaboration** - The collaboration service helps connecting and keeping teams productive. It provides access to information needed for users to make good decisions.
- **Portal** - The portal component includes functionality useful for designing, deploying and managing web-based portals.
- **Search** - The search service provides common search capabilities.
- **Content Management** - The content management component offers document management functionality, with features including versioning and workflow insight.
- **Business Process** - The business process service helps integrating business processes to external systems, through XML-based electronic forms.
- **Business Intelligence** - The business intelligence service helps employees share and control business information in order to make better decisions.

![SharePoint Platform Services](image)

Figure 4-3: SharePoint Platform Services (Microsoft Corporation, 2009a)
4.6 Microsoft Office Communicator

*Microsoft Office Communicator* is an application for communication and collaboration targeted towards corporate environments (Wikipedia, 2009e). The goal of the Office Communicator application is to enable teams to communicate and share information easily and efficiently without being limited by geographical distances.

The features in Communicator include instant messaging, traditional voice calls, desktop sharing, and video conferencing (Microsoft Corporation, 2009b). Additionally, the functionality integrates with other Microsoft Office products, like Excel and SharePoint, making users able to perform contextual collaboration.

4.7 Case Management

“*Case Management is the management of long-lived collaborative processes that coordinate knowledge, content, correspondence and resources to progress a case to achieve a particular goal; where the path of execution cannot be predetermined in advance of execution; where human judgment is required to determine how the end goal can be achieved; and where the state of a case can be altered by external out-of-band events*” (Business Process Trends, 2009, p. 2).

Business Process Trends has provided this definition, which is one of many attempts to come up with a solid definition for case management. With the definition being rather detailed, it includes many of the core characteristics of case management.

Case Management is often being described as a style of document management, knowledge management, or customer relationship management (Silver, 2009b). Document content, collaborative decision-making and customer interactions are all important elements of case management, just as they are important in traditional BPM as well. As described back in section 4.3.1, a process instance in BPM runs through a sequence of steps in a diagram, starting at one start event, and ending at one or more end events. The characteristic difference of case management is that the progression of a case is *unstructured*, meaning that the flow logic cannot be defined in a diagram in advance, making it very difficult to automate case progression. Factors like human judgment, external events, and business rules are used to determine the flow logic in BPM, but this cannot be done in case management. Instead, in case management those factors determine at runtime which activities need to be initiated, and whether additional steps are required. As BPM relates to single processes, case management usually relates to *collections* of processes and isolated tasks. The number of processes in a case cannot be determined by predefined rules and templates.

4.7.1 The case folder

The central artifact in case management is the *case folder*. The main purpose of the case folder is to give an overall coordination of each case (Silver, 2009b). It is partially populated in advance by a case template, but with the flexibility to change at runtime as the case is progressing. The case folder contains a collection of all case data, usually including tasks, documents, rules, events, deadlines, and other elements related to or involved in the case. Even so, the case folder is not supposed to act as a representation of a set of isolated elements. Instead, it can be viewed as a *single element* that is progressing towards completion. The combined state of all tasks and documents in the case is what determine the state of the case as a whole. Thus, by simple inspection of the case folder, it should be possible to understand the status of the case as a whole.
4.7.2 Knowledge work

Case management has often proven to be intensely manual, paper-driven, and subject to delay and poor visibility (Business Process Trends, 2009). The reason for this is primarily the fact that case management is driven by knowledge work. Knowledge work is involved in the most important processes of an organization; they add the most value, as well as having the greatest impact on long-term success. Knowledge workers have acquired their knowledge through collaboration with more experienced colleagues, and from working on similar cases in the past. The knowledge of workers is often needed in important steps and decisions during the case process, involving ambiguous scenarios where judgment and creativity is required. This makes knowledge intensive processes quite difficult to analyze and structure.

Figure 4-4 illustrates the structural difference between BPM and Case Management, as well as in what degree these technologies require the use of knowledge work. BPM can be modeled at design time, making the flow logic structured and requiring little use of knowledge work. In case management on the other hand, the actual steps that need to be completed are determined on the fly. This makes case management highly unpredictable and unstructured, which significantly increases the requirement of knowledge work. As can be seen in Figure 4-4, case management is more structured and requires less knowledge work at design time, as compared to runtime. The reason for this is that case templates yield a certain benefit as a starting point for cases of particular types.

Figure 4-4: The use of structure and knowledge work in BPM and Case Management.
4.7.3 **Case-Oriented BPM**

Despite the fact that case management processes have an unstructured approach, it shares some of the issues that can be found in structured processes (Silver, 2009b), like:

- Long completion time.
- Inefficient use of resources.
- Not retained, or misplaced information.
- Lack of standardization across the organization.
- Hard to correlate with policies, regulations, and best practices.

These problems are all handled by traditional BPM suites. Case management processes need the same capabilities, but that would require another type of BPM platform. There have been several proposals of technologies that may support case management, including Customer Relationship Management (CRM) and Content Management Systems (CMS) (Business Process Trends, 2009). These technologies have however shown apparent shortcomings when trying to address key requirements of case management, like initiating new processes while a case is in progress.

**Case-Oriented Business Process Management** has been pointed out as the optimal approach to support knowledge workers in becoming more effective in their work. Case management based on BPM can combine knowledge and process, maintain unpredictability in case progression, and also organize several other technologies to support such knowledge intensive processes. Case-oriented BPM aims to overcome the limitations of the other proposed technologies, by providing support for:

- Visual models of business processes that can be managed and changed by people working on the case.
- Execution of complex processes that coordinate and interact with multiple systems.
- Collaboration between case workers in a team.
- Integration to content repositories.

4.7.4 **The benefits of effective case management**

The following gives a summary of some of the most important benefits of a solid and effective case management implementation:

- Easier management of paper-intensive tasks.
- Increased productivity and performance when working with knowledge centric processes.
- Case workers are able to monitor overall case status, including how many cases are in progress, average cost and time consumption for each case, as well as what stage each case is at.
- Extended consistency when dealing with predictable and repeatable case elements.
- Predictability of how many cases a given set of resources can handle.
- Important parts of knowledge work can be captured in models, making it possible to reuse that particular knowledge in similar cases in the future. This will lead to minimized knowledge loss when the staff changes, in addition to provide a strong foundation when training new staff members.
- Case workers are enabled to quickly adapt to changes by modifying their approach during execution of a case process, instead of getting locked into a structured and predefined path.
• Increased coordination between the various existing support systems used by the organization.

4.8 K2

K2 is a software platform intended for building process driven applications in order to improve business efficiency (K2, 2009). Various types of organizations use K2 to increase efficiency, save money, and to reduce risks. K2 introduces visual and familiar tools, thus making people with varying background and technical skill able to collaborate and create tailored process-driven applications for suitable corporate scenarios. These applications may be used to manage simple business processes, such as inventory tracking or document approval, or to perform more advanced tasks like orchestrating complex business process models. Smaller applications may also be used as building blocks when creating more comprehensive solutions.

The K2 platform is built on Microsoft technology, and the K2 tools are integrated with Microsoft software applications like Visio, Visual Studio, and SharePoint.

K2 introduces three different products; K2 Blackpoint, K2 Blackpearl, and K2 Connect, as shown in Figure 4-5.

![Figure 4-5: The K2 Platform (K2, 2009)](image)

CODIO’s case management solution is built using components provided by K2’s Case Management framework. This involves utilization of Microsoft SharePoint’s portal capabilities, combined with K2 Blackpearl for business process modeling and execution, and K2 Connect for connectivity.

4.8.1 Representation of BPMN models

K2 supports process design within visual design canvases inside Microsoft tools such as Visual Studio and Visio. The process flow designed in K2 is represented by activity boxes as process steps, each containing an arbitrary number of tasks. Pools and lanes are omitted from the presentation by default. Thus, the workflow presentation in K2 is notably different compared to traditional BPMN workflow models.
4.8.2 Benefits of using K2
K2 supports drag and drop functionality with wizard driven interfaces, providing straightforward implementation of flow logic, and eliminating the need for significant technical insight (SourceCode Technology Holdings, 2007). Microsoft .NET components that correspond to the elements designed in the canvas are created automatically. K2 provides several methods to configure and describe each process step sufficiently, making the process model subject for automatic execution.

4.8.3 K2 Blackpoint
K2 Blackpoint presents a Microsoft Office-styled designer called K2 Studio. K2 Studio is targeted towards non-developers and business users who want to create SharePoint workflows and process-driven applications without having to write any code. K2 Blackpoint is a subset of K2 Blackpearl, making it possible to optionally upgrade.

4.8.4 K2 Blackpearl
K2 Blackpearl offers features for building process-driven applications, and is considered K2’s flagship product. The product is built on the Microsoft Windows Workflow Foundation Framework, and integrates deeply with Microsoft Office Communication Server (Hill et al., 2009).

Some important functionality provided by K2 blackpearl is:

- Collaboration between users, regardless of technical skill or preference.
- Report building.
- Collection and presentation of data.
- Real time monitoring of events.
- Task management.
- Options for modifying processes, forms and reports.

4.8.5 K2 Connect
K2 Connect is an add-on for K2 Blackpearl. The product provides the tools needed to integrate K2 with Line-Of-Business (LOB) applications, which are among the critical applications that are vital to running an enterprise (Wikipedia, 2009). K2 Connect facilitates visual creation of reusable business entities called SmartObjects. SmartObjects are used in order to integrate and exchange information with applications and workflows.

4.8.6 Integration with Microsoft SharePoint
Content stored in the SharePoint environment is often used in connection with process-driven applications. Through its integration towards SharePoint, the K2 platform provides access to SharePoint functionality like:

1. Listing of items and documents.
2. Lists and libraries.
3. Sites and workspaces.
4. Users and groups.
5. Web content management.
6. Publishing of K2 data.
4.9 Real time data

Real-time data may be defined as “information that is available or delivered immediately after collection” (Wikipedia, 2009b). While traditional database management systems handle shared and persistent data, real-time systems are more concerned with data having a very short and meaningful lifespan (Graham, 1992).

The most important property of a system dealing with real-time data is predictability, meaning that the timing and functional behavior should be deterministic in order to satisfy system specifications (Stankovic, 1998). Real-time computing is not about building extremely rapid systems. It is rather about engineering systems fast enough to act on their environments in well specified ways.

Real-time systems relate to deadlines, a term used to describe time constraints in real-time data processing (Jensen, 2009). The deadlines define the time of when a system must be finished processing some real-time data. In order to engineer a system that meets its time constraints, it must be possible to analyze the system to determine its time consumption.

There exist two levels of deadlines in real-time computing; hard real-time deadlines, and soft real-time deadlines. The term “soft deadline” refers to the general case of a deadline, while the hard deadline is a special case. Not meeting a deadline may lead to consequences of varying degrees, depending on whether the deadline is categorized as hard or soft.

4.9.1 Hard real-time

Hard real-time data is characterized as data with critical demands of response time. If the data is not processed within a certain specified time window, it might lead to catastrophic failure with severe consequences (Williams, 1991, p. 110). In order to meet the strict deadlines in this environment, the system must guarantee to choose the appropriate actions combined with precise timing (Musliner et al., 1995). Research in real-time systems address this issue through methods that ensure the reaction rate of the system matches the rate of change in the environment. Hard real-time data systems are highly predictable, but they are not flexible enough to adapt to dynamic situations (Buttazzo et al., 2005).

4.9.2 Soft real-time

Soft real-time is, like hard real-time, depending on a certain response time (Wikipedia, 2009c). Failing to process data within the deadline will result in data loss, although some loss will be tolerated at the cost of quality of the service. The objective in soft real-time data processing is to obtain a low percentage of missed deadlines, leading to a low amount of lost data. Soft real-time systems are built to reduce resource consumption, tolerate overloads and adapt to system changes.

4.9.3 Categorization of computer systems within the real-time domain

Within the domain of real-time computing, computer systems can be categorized into three levels;

1. the ones who need to satisfy hard real-time deadlines,
2. the ones based on soft real-time data, and
3. the non real-time computer systems, which are systems that do not relate to real-time data deadlines.

This classification is illustrated with examples in Figure 4-6.
Figure 4-6: Example of various computer systems’ use of real-time data deadlines

As can be seen in the figure above, BPM is located within the domain of non real-time systems. Traditional BPM is not designed to respond to real-time data input within certain predefined deadlines. CEP on the other hand has to conform to time constraints, as its nature is built upon detecting and analyzing event streams. CEP can actually be placed within both the hard real-time and soft real-time domain, depending on the importance of satisfying the deadlines defined in the system.

In our example in Figure 4-6, CEP is represented as a soft real-time system, along with event-driven BPM. This is because CEP, as part of our event-driven BPM prototype created in context of the CODIO project, will be designed as a soft real-time system. Our CEP solution will analyze events received from real-time updates in directional drilling. Missing a few of these updates will not have catastrophic consequences, as updates can be obtained on request as soon as the prototype is ready for further processing.
4.10 Event-Driven BPM

Event-Driven BPM is an emerging discipline that introduces *event-driven behavior*, taking business processing one step further. The discipline is born from a combination of BPM and CEP.

As mentioned back in section 4.3, BPM is used to capture, model, and manage business processes in a process model diagram. CEP runs in parallel with BPM as a separate platform, continuously identifying, analyzing, and processing events. BPM and CEP correspond with each other through events both generated by the BPM workflow engine and the various IT services related to each business process step.

The purpose of Event-Driven BPM is to obtain situational knowledge from distributed systems in real-time or almost real-time scenarios (Ammon et al., 2008). Events are defined and selected in order to identify complex events, or *event patterns*, of interest. The event streams from all sources are continuously filtered and compared to a predefined set of *complex event rules*, holding certain criteria that specify what combinations of events are considered complex events. If all rule criteria of a complex event rule is satisfied, a *trigger* will be activated, resulting in an *action* being initiated.

4.10.1 An example of event-driven behavior

This section will present a brief example of the event-driven behavior concept. The example demonstrates one approach of utilizing event-driven behavior based on fictional data and objectives related to a simple drilling operation.

Imagine we have a BPM workflow describing a process regulating the overall drilling operation. This process keeps track of whether the well is actually being drilled, or if the operation is temporarily halted due to activities like casing or magnetic correlation. The current process step from this BPMN model can be continuously received by the event-driven system as an event, along with real-time data obtained from the MWD tool.

When the BPM process is initiated, it starts at an activity called “Drilling”. At this point, the MWD tool begins sending real-time data updates to the event-driven system. After drilling for a while, analyzes from the real-time data feed indicates an unexpected increase in rock density, which is considered an event of interest. The prototype compares the available events against the predefined complex event rules, and finds a match on a rule containing the following criteria (see Figure 4-7):

1. The BPM process has to be at the “Drilling activity”.
2. The rock density must have a value of at least 4.0 kg/dm3.
3. The movement speed must be more than 2 meters per second, or the rotational speed has to be less than 3 revolutions per seconds.
The events have met all the conditions of the rule, resulting in a trigger being invoked. In this case the trigger initiates an action to notify the Directional Driller, as shown in Figure 4-8. Once the Directional Driller gets informed of the complex event, he can investigate the situation and choose an appropriate follow-up decision.

Figure 4-7: Single events forming a complex event

Figure 4-8: Example of event-driven behavior
Innovation
5 Case: Deviations in directional drilling

This chapter will discuss the case description in detail. As mentioned earlier, the case should elaborate a possible addition to the functionality developed in the CODIO research project. A major focus in the CODIO project is to address communication and collaboration support in case of a deviation occurring during the drilling operation. Based on this, we have identified deviation handling in directional drilling as a suitable problem area. A deviation in directional drilling basically means that the actual drilling path at some point in the drilling process deviates from the original path described in the drilling plan. Such a deviation is illustrated in Figure 5-1, where we can see the actual path deviates from the planned path due to an erroneous inclination.

![Figure 5-1: Deviation caused by erroneous inclination value](image)

An uncaught deviation could lead to severe economical consequences, as the path could be led into a less optimal area, perhaps ending up interfering or colliding with other wells.

The focus of the CODIO WP1 project is to facilitate communication and collaboration between employees after a deviation in drilling has occurred, through extensive portal capabilities supported by Microsoft SharePoint. A BPMN workflow model has been designed and implemented in K2 BlackPearl, explaining a proposed deviation handling process applicable to the drilling environment of ConocoPhillips Norway.

The work performed in this thesis will be based on this BPMN model, but with a distinct focus on assisting employees by automatically detecting the actual deviation. In other words, the prototype developed in this thesis will ultimately be used to initiate the BPM workflow logic engineered in the CODIO project.
5.1 The BPMN Model
This section will discuss the BPMN model originally designed in the CODIO research project, and how the model is modified when introduced to the prototype developed in this thesis. Figure 5-2 illustrates the modified BPMN model.

Figure 5-2: The BPMN workflow
5.1.1 Roles
As we can see from the model, we have four roles that participate and interact with each other during the course of the deviation handling process. The roles are named the Drilling Supervisor, the Directional Driller, the Geologist, and the CODIO Process Engine.

- **The Drilling Supervisor** overlooks the entire drilling operation and participates in collaborative discussions. In this context, the role is defined as the highest organizational level in deviation handling, implying that members of this role need to have a certain amount of competence in drilling operations. Thus, the Drilling Supervisor has the final saying in challenging scenarios dominated with uncertainties and diverse opinions on how to proceed when dealing with a deviation. A proposed change in the drilling path must always be approved by the Drilling Supervisor in cases where a severe deviation has occurred.

- **The Directional Driller** is responsible for tasks involved in directional drilling. The role is set to evaluate, assess, and validate data whenever a possible deviation might have occurred. Directional Drillers are also the ones who determine and act on the best follow up once a deviation has occurred; either to arrange a collaboration session, or adjust the necessary drilling parameters. The Directional Driller also participates in collaboration sessions when critical deviations occur.

- **The Geologist** is a valuable asset in the process of determining the best course of action in a severe deviation. They have good insight in how the drilling path may be changed, as they are able to make well informed contributions based on available data. This competence also makes them suited to perform the task of creating a path change proposal when needed. The Geologists are responsible for concluding the collaborated evaluation, and they are responsible for initiating the proper follow up based on the collaborative conclusion.

- **The CODIO Process Engine** is the process engine solution engineered by the CODIO research group. The CODIO Process Engine is designed to perform automatic tasks such as to:
  - Create a case, or update an existing case, and to notify participants whenever a collaboration team has to be summoned.
  - Create new processes to initiate actions like ordering new simulations.
  - Send request for a new survey reading.
  - Subscribe and wait for the next survey to arrive.
  - Initiate another process with the purpose of changing the drilling plan.
5.1.2 Initiating the deviation handling process

In the original process model, only ‘some person monitoring real-time data’ could start the deviation handling process, based on manual investigation of survey data indicating that the well is not being drilled according to the plan. This activity is completed by employees who might know by heart what combinations of parameters from the survey data to look for when trying to identify possible deviations. To automate this activity, the same knowledge that already resides within the employees’ heads must be defined and modeled as complex event rules within the prototype. Relevant values must be understood, analyzed, and carefully compared to each other, in order to successfully identify all possible variations of a deviation scenario.

As a modification to the original BPMN model, we have introduced the EDBPM prototype as a source to initiate the deviation handling process. As seen in Figure 5-2, the EDBPM prototype receives real-time data updates from the survey tools as soon as they are available. The data is filtered and analyzed in the search for any occurring deviations. If no deviations are found, the prototype waits until the next survey update before further processing, enabling the search for deviations throughout the entire drilling operation. If a deviation is found however, the most appropriate follow-up is chosen based on the complex event rules defined in the prototype, and the deviation handling process is automatically initiated. Both information about the deviation and the selected follow up is sent to the directional driller, who is set to inspect and validate the results before proceeding. Proper design of the complex event rules will make the prototype able to provide viable and correct suggestions for proper follow-up decisions, effectively facilitating decision support.

As indicated by the model, ‘some person monitoring the survey data’ still has the possibility to initiate the deviation handling process, even though no deviations are detected by the prototype. Preserving the opportunity of human involvement in the search for deviations is essential to avoid any possible deviation scenarios to occur that the prototype for some reason is unable to detect.

5.1.3 Explanation of the deviation handling workflow

The BPMN model tells us exactly how the workflow is handled after a deviation has occurred. The workflow can be explained into the following:

When an instance of the deviation handling process is initiated, the directional driller has to inspect the available data in order to confirm and verify that a deviation in fact occurred. Also, the seriousness of the deviation has to be determined. The severity of the deviation is used to decide one out of three possible actions:

- **Insignificant deviation**: no action should be taken. The process ends.
- **Small deviation**: a small deviation is not of a critical nature. Some corrections of certain directional drilling parameters should be handled by the directional driller in order to solve the issue.
- **Critical deviation**: a final decision must be taken by competent personnel on how to proceed. A case will be established in the well-specific case folder, and geologists, drillers, and drilling supervisors will be invited to join a collaboration session.
In situations where a critical deviation has occurred, the established collaboration group has to discuss and evaluate the path, and make a decision on how to further proceed. The decision may be one of the following:

- Make new actions, which includes creating new processes
- Order a new survey report to get more updated information
- Wait for the next survey report before deciding
- Update the path.

If the decision is to update the path, it will be necessary to determine the extent of the path change. If the extent is comprehensive, the original drilling plan has to be modified. If the extent is of a smaller nature on the other hand, a change proposal must be submitted for approval. Once the change proposal has been approved by the drilling supervisor, the relevant drilling parameters are adjusted by the directional driller, and the path document is updated. If the path document were to remain unmodified after a change of path, the prototype would end up comparing all future survey readings against an invalid drilling path, resulting in a range of false deviation detections.

### 5.2 Categorizing deviations into levels

To reflect the three different outcomes when deciding a follow-up in case of a deviation, we will categorize the deviations into three different levels;

- **1 – Small deviations that require no action.** These deviations might not impose any direct issues in regards to reaching the final target, but might instead indicate that the well could be straying a bit off track.
- **2 – Deviations that require adjustment of the drilling parameters.** These deviations should be identified and taken care of as soon as possible, as they represent a medium threat that could easily escalate into critical deviations if ignored too long.
- **3 – Critical deviations requiring collaboration between competent staff members.** These deviations must be dealt with rapidly, as they could – in worst case scenarios – lead to consequences of huge economic losses.

#### 5.2.1 Search sequence to locate deviations in different levels

It will be pointless to search for and identify level 1 deviations, since the follow-up decision in such cases is to do nothing. Instead, we will only define complex event rules that capture the level 2 and 3 deviation scenarios. The prototype will start the search sequence by looking for level 3 complex events. If no level 3 deviation scenarios are found, it will begin searching for level 2 deviations. By performing the search sequence in this order, we ensure that the most critical deviations are found and reported first. Several deviations of the same level may be identified at the same time, in order to give the decision-makers complete information of the situation.

### 5.3 Real-time survey data

As mentioned back in section 2.3, modern oil and gas wells are equipped with a range of sensors and instruments, with the purpose to continuously measure and log ongoing drilling operations. The objective of this monitoring is to get a sufficient overview of every aspect of the operation, including updated information on the current drilling location, the geology situation, and other relevant situational data.
The real-time data of interest in this case is called *directional survey data*. This type of data is logged in a report periodically while the drilling operation is active, through MWD tools placed on the well. Raw data measured in directional survey data concerns directional parameters called *inclination* and *azimuth* (Egil Aanestad, Halliburton, personal communication 04.01.10). These parameters, combined with the *Measured Depth*, makes it possible to calculate the rest of the information needed to know the exact position of the well, including the location of the well based on both global UTM coordinates and local map coordinates.

Figure 5-3 below illustrates a part of a survey report retrieved from “Eldfisk”, an oil rig stationed at the northern sea bed.

<table>
<thead>
<tr>
<th>Station ID</th>
<th>MD (ft)</th>
<th>Incl (°)</th>
<th>Azim (°)</th>
<th>TVD (ft)</th>
<th>Vbee (°)</th>
<th>NF-S (ft)</th>
<th>Ef-W (ft)</th>
<th>DLS (ft)</th>
<th>Northing (m)</th>
<th>Easting (m)</th>
<th>Latitude</th>
<th>Longitude</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tie-In</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-63.46</td>
<td>-22.35</td>
<td>0.00</td>
<td>5164077.74</td>
<td>5164077.74</td>
<td>5164077.74</td>
<td>5164077.74</td>
</tr>
<tr>
<td>6.00</td>
<td>0.00</td>
<td>0.00</td>
<td>6.00</td>
<td>0.00</td>
<td>-63.46</td>
<td>-22.35</td>
<td>0.00</td>
<td>5164077.74</td>
<td>5164077.74</td>
<td>5164077.74</td>
<td>5164077.74</td>
<td></td>
</tr>
<tr>
<td>54.00</td>
<td>0.17</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-63.46</td>
<td>-22.35</td>
<td>0.11</td>
<td>5164077.76</td>
<td>5164077.76</td>
<td>5164077.76</td>
<td>5164077.76</td>
</tr>
<tr>
<td>256.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-63.54</td>
<td>-22.51</td>
<td>0.21</td>
<td>5164077.72</td>
<td>5164077.72</td>
<td>5164077.72</td>
<td>5164077.72</td>
</tr>
<tr>
<td>306.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-63.54</td>
<td>-22.51</td>
<td>0.16</td>
<td>5164077.70</td>
<td>5164077.70</td>
<td>5164077.70</td>
<td>5164077.70</td>
</tr>
<tr>
<td>556.00</td>
<td>0.17</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-63.66</td>
<td>-22.40</td>
<td>0.41</td>
<td>5164077.68</td>
<td>5164077.68</td>
<td>5164077.68</td>
<td>5164077.68</td>
</tr>
<tr>
<td>466.00</td>
<td>0.17</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-63.94</td>
<td>-22.48</td>
<td>0.57</td>
<td>5164077.66</td>
<td>5164077.66</td>
<td>5164077.66</td>
<td>5164077.66</td>
</tr>
<tr>
<td>506.00</td>
<td>0.67</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-64.04</td>
<td>-22.46</td>
<td>0.84</td>
<td>5164077.55</td>
<td>5164077.55</td>
<td>5164077.55</td>
<td>5164077.55</td>
</tr>
<tr>
<td>556.00</td>
<td>1.00</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-64.43</td>
<td>-23.53</td>
<td>0.84</td>
<td>5164077.45</td>
<td>5164077.45</td>
<td>5164077.45</td>
<td>5164077.45</td>
</tr>
<tr>
<td>600.00</td>
<td>0.51</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-64.61</td>
<td>-24.29</td>
<td>0.52</td>
<td>5164077.29</td>
<td>5164077.29</td>
<td>5164077.29</td>
<td>5164077.29</td>
</tr>
<tr>
<td>650.00</td>
<td>0.10</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-65.72</td>
<td>-24.71</td>
<td>0.20</td>
<td>5164077.16</td>
<td>5164077.16</td>
<td>5164077.16</td>
<td>5164077.16</td>
</tr>
<tr>
<td>700.00</td>
<td>0.75</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-66.29</td>
<td>-26.26</td>
<td>2.36</td>
<td>5164076.54</td>
<td>5164076.54</td>
<td>5164076.54</td>
<td>5164076.54</td>
</tr>
<tr>
<td>750.00</td>
<td>0.97</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-66.94</td>
<td>-27.36</td>
<td>2.36</td>
<td>5164076.50</td>
<td>5164076.50</td>
<td>5164076.50</td>
<td>5164076.50</td>
</tr>
<tr>
<td>800.00</td>
<td>0.40</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-67.46</td>
<td>-28.17</td>
<td>2.06</td>
<td>5164075.52</td>
<td>5164075.52</td>
<td>5164075.52</td>
<td>5164075.52</td>
</tr>
<tr>
<td>850.00</td>
<td>0.00</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-67.94</td>
<td>-28.62</td>
<td>1.05</td>
<td>5164074.59</td>
<td>5164074.59</td>
<td>5164074.59</td>
<td>5164074.59</td>
</tr>
<tr>
<td>900.00</td>
<td>0.25</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-68.57</td>
<td>-29.50</td>
<td>2.57</td>
<td>5164076.57</td>
<td>5164076.57</td>
<td>5164076.57</td>
<td>5164076.57</td>
</tr>
<tr>
<td>950.00</td>
<td>0.73</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-71.22</td>
<td>-31.24</td>
<td>2.17</td>
<td>5164074.67</td>
<td>5164074.67</td>
<td>5164074.67</td>
<td>5164074.67</td>
</tr>
<tr>
<td>1000.00</td>
<td>0.87</td>
<td>2.00</td>
<td>0.00</td>
<td>0.00</td>
<td>0.00</td>
<td>-73.25</td>
<td>-31.63</td>
<td>3.02</td>
<td>5164072.50</td>
<td>5164072.50</td>
<td>5164072.50</td>
<td>5164072.50</td>
</tr>
</tbody>
</table>

Figure 5-3: An excerpt from a real-time survey report

- **Station ID** – Shows if a special operation was conducted at the time of the specific survey reading entry. The first survey entry in Figure 5-3 reveals that an operation called “Tie-In” was completed. This means the well has been drawn back to the surface for casing, and that no survey updates should be taken until the depth has reached the tie-in. The Station-ID parameter will not be used when calculating deviations.

- **Measured Depth (MD)** – Keeps track of the total length of the borehole, measured in feet (see Figure 5-4). Measured depth will always be larger than vertical depth in cases with directional drilling, since the borehole is partially drilled horizontally (Wikipedia, 2009g). When calculating deviations, the Measured Depth parameter will be used as the point of reference when comparing the drilling plan with the survey readings.

- **Inclination (Incl)** – Measures the directional angle of the survey tool. 0° means the inclination is vertical, while 90° means horizontal inclination (Horizontal Mudlog, n.d.). As can be seen in the survey report in Figure 5-3, the current inclination values are noticeably low. This is because the part of the survey report that is shown in the figure is retrieved from the initial phase of drilling. The well is always drilled in an almost straight vertical direction during the first phase, thus resulting in a small well inclination. The value will increase substantially when the well is starting to lean more towards the horizontal direction.
- **Azimuth (Azim)** – Shows the horizontal angle of the borehole, measured clockwise from the center of the well at the oil rig. The azimuth is measured in degrees, using 0° as true north (See Figure 5-5). As seen in the survey report in Figure 5-3, the azimuth value is constantly bouncing back and forth. This is caused by an unstable direction during the initial phase. The azimuth value will stabilize as soon as the well gets more attached.

- **Total Vertical Depth (TVD)** – Keeps track of the total vertical distance from the surface to the end of the borehole in a straight perpendicular line (Wikipedia, 2009h).

- **Vertical Section (VSec)** – Shows the well’s horizontal progression towards the target. The VSec value is increased and decreased whenever the well is horizontally moving towards or away from its target, respectively.

- **N/-S (North/-South) and E/-W (East/-West)** – Also called Northing and Eastings, represents the coordinates of the well in context of a map. Positive numbers mean north/east, while negative numbers mean south/west.

- **Dog Leg Section (DLS)** – Indicates the difference in angle between every 100 feet. The angle is measured in degrees.

- **Grid Coordinates** – Provides the coordinates of the well, with the starting point of the well as origin. The coordinates are based upon the Northing / Eastings values, represented in a grid system. The grid used in this particular survey report is the UTM Zone 31 on ED50 Datum.

- **Geographic Coordinates** – Shows the geographical position of the well as global UTM coordinates. The grid coordinates and geographic coordinates are not used when calculating deviations, as the Northing / Eastings values will be used instead.

**Figure 5-4: Parameters in the survey report**
Typically, directional survey data is updated within intervals of 30 – 500 feet, depending on the directional steering activity (Wikipedia, 2010d). It might eventually prove beneficial to let the prototype continuously request new survey updates within a predefined time frame, allowing for a steady and rapid stream of input data. This would effectively increase the accuracy in deviation detection, as the prototype would receive larger amounts of data to investigate.

5.4 Techniques used in combination with deviation detection
This section will discuss techniques and approaches used to address the complexity of detecting deviations and determining their seriousness. The techniques are taken into account when identifying and designing the different deviation scenarios, including the establishment of viable complex event rules.

5.4.1 Analyze deviation outcome
When identifying a deviation, we need to be aware of the deviation outcome in order to recognize whether the deviation should be triggering any actions or not. Say for instance that the Inclination value appears to be too low (angled too much vertically), while at the same time the Total Vertical Depth is too high (drilled too far vertically). In this case it is fair to assume a real deviation in inclination is in progress, since the high inclination value will force the well to be drilled even further vertically. Thus, the deviation is considered to have a negative outcome, and the deviation handling process should definitely be triggered.
Similarly, if the survey report indicates that the inclination value is too high (angled too much horizontally), and at the same time the Total Vertical Depth is also too high (drilled too far vertically), it means the well is actually getting gradually back on track. The high inclination value makes the well drill less in the vertical direction than what is specified in the drilling plan, effectively reducing the deviation in the TVD parameter. This kind of deviation could very well be initiated on purpose by the directional driller, perhaps as a response from getting aware of the deviation; hence it should be considered a *positive* outcome, and not trigger any actions.

The method of analyzing the deviation status is demonstrated in more detail in Table 5-1 below.

<table>
<thead>
<tr>
<th>Deviation in Inclination</th>
<th>Deviation in TVD</th>
<th>Deviation status</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>-3°</td>
<td>6ft</td>
<td>Considered a deviation</td>
<td>Inclination is too low, distinguished by the subtraction symbol. Thus, the well is angled too much vertically. TVD is too high, meaning that the well has already been drilled too far vertically. Further drilling at this Inclination will lead to an even higher deviation in TVD.</td>
</tr>
<tr>
<td>3°</td>
<td>6ft</td>
<td>Not considered a deviation</td>
<td>Inclination is too high; the well is angled too much horizontally. TVD is also too high, thus the well has already been drilled too far vertically. Further drilling at this Inclination will lead to a decreased deviation in TVD.</td>
</tr>
<tr>
<td>-3°</td>
<td>-6ft</td>
<td>Not considered a deviation</td>
<td>Inclination is too low. The well is angled too much vertically. TVD is also too low, meaning that the well has not been drilled far enough vertically. Further drilling at this Inclination will lead to a higher TVD value, decreasing the deviation in TVD.</td>
</tr>
<tr>
<td>3°</td>
<td>-6ft</td>
<td>Considered a deviation</td>
<td>Inclination is too high, thus the well is angled too much horizontally. TVD is too low; the well has not been drilled far enough vertically. Further drilling at this Inclination will lead to an even higher deviation in the TVD parameter.</td>
</tr>
</tbody>
</table>

Table 5-1: Demonstration of how to analyze deviation outcome in Inclination

Figure 5-6 illustrates how a deviation in TVD affects the outcome of the deviation in inclination.
Also, we need to be aware of whether the actual deviation by itself is increasing or decreasing. If the deviation seems to be decreasing, we consider the outcome to be positive as it indicates the well is getting back on track.

In order to successfully implement the functionality needed to analyze the deviation outcome, the prototype must be designed to store both current and previous survey readings, making it possible to evaluate how the values are developing.

5.4.2 Progress overview
A factor playing a part in determining the seriousness of the deviation is how far the well currently appears to be from its target. This is important since a deviation will be remarkably more critical if the target location is close, than it would be if the target was far away. To be able to examine exactly how far the well has progressed, we need to calculate some sort of a progress event, displaying the total distance given in percent. Doing this would require us to use the values from the TVD and VSec parameters, where the current value is compared to their expected final value. Figure 5-7 illustrates the concept of a progress overview.
As we can see from the example in the figure, the planned TVD parameter at this MD is 600ft, while the final value should be 1000ft. At the same time, the actual VSec value is 450ft, while the final value is expected to be 1800ft. This gives us a total progress of 1050ft (600 + 450), and a final value of 2800ft (1000 + 1800). Thus, the overall progression is 37.5% (1050 / 2800).

The reason why we use the planned TVD in this calculation rather than the actual TVD, is because the calculation otherwise would become incorrect as soon as the actual TVD value started deviating. A calculation based on the actual TVD value would make the well progression increase even if the well is drilled too far in the vertical direction, which is obviously not correct. When we base our calculation on the planned TVD parameter, we get a more accurate representation of how far the well has actually progressed. The actual VSec parameter on the other hand will decrease when drilled in the wrong horizontal direction and similarly increase when a correct direction is maintained. Thus, the actual VSec parameter can be used to represent the exact progress in the horizontal plane.

The progress overview will be used as an indicator of how serious an occurred deviation is. A well that has been drilled further than 80% of the total distance towards the final target will be considered a level 3 deviation, since there will be far less space and time to adjust the direction once the well is closing in on its target. A well progression of less than 80%, on the other hand, will be treated as level 2, due to good opportunities to compensate for the occurred deviation.
5.4.3 Event windows

Despite all the advanced sensor technology present on modern drilling wells, the survey readings are not always completely reliable. (Egil Aanestad, Halliburton, personal communication 04.01.10). In some rare cases the tools are affected by magnetic disturbance, resulting in readings containing obviously invalid parameter values, like for instance the inclination value suddenly spiking to extreme and unlikely numbers. When such faults in the survey readings occur, we have to avoid triggering any complex event rules, even though the survey parameters seems to satisfy all of the rule criteria. To ensure this is avoided, we will implement a common technique called time windows (EsperTech, 2009) or event windows (Information Management, 2006), adding an additional level of complexity to our tier 1 CEP implementation. The event windows technique compares the current event against a specified number of the previous events of the same type. This comparison makes it possible to investigate the recent development of that event value, giving an indication on whether or not the current value is flawed. Imagine a scenario where the inclination deviation value from consecutive survey readings is monitored within a window of five events, as in the illustration in Figure 5-8 below.

![Inclination event window](image)

**Figure 5-8: Inclination event window**

As we can see from the figure, a new survey entry is received every five minutes. The inclination events inside the event window currently range from survey reading #2 to #6. Survey reading #1 has at this moment become obsolete, and is excluded from the window. The inclination values inside the event window varies between 1,5° and 2,0°, except in survey reading #6 where the value suddenly is reported as 6,5°. This leap in inclination seems highly unlikely, as it would imply an extreme change of direction. The implementation of the event window makes the prototype able to automatically interpret such survey readings as suspicious, and to wait for the next survey update to see how the inclination value evolves rather than to immediately report a deviation.

The illustration reveals that survey reading #7 contains an inclination value of 1,9°. This reading will be included in the event window as soon as it becomes available, which is expected to be at approximately 12:30. This newly obtained information will prove the value in survey reading #6 to be flawed, seeing that the inclination value in #7 is much more consistent with the other values inside the window. This applies to the opposite scenario as well; if survey reading #7 were to represent a value that resembles the value in #6, there would be a greater chance of survey reading #6 actually being correct.
5.4.4 Change in deviation severity at various depths

The severity of a deviation is highly depending on the location of the well in regards to the measured depth (Helge Rørvik, Halliburton, personal communication 30.03.10), of the following reasons:

- Several wells, depending on the location of the field, are drilled in close proximity to previously established wells. This makes well collisions an apparent potential issue that should be considered. To make sure collisions do not occur in these areas, the well has to be drilled with careful preciseness according to the plan, and with complete and continuous overview of any potential deviation scenario. Thus, the criteria determining whether any deviation should be detected in these areas have to be specifically sensitive. A deviation of perhaps 20 ft in the Northings / Eastings pair at a vertical depth of 8000 ft would be considered a great achievement, while the same deviation could become disastrous at 3000 ft because of the chance of colliding with surrounding wells.

- Another factor is the estimated size of the target. The target zone may differ a lot between wells, depending on the reservoir and the approaching direction. A large target zone would allow for more extensive deviations, while a smaller target would require more precise maneuvering in line with the drilling plan.

- If a deviation occurs early in the drilling process, the deviation could easily escalate to higher implications very quickly, as compared to a deviation occurring after drilling for several thousands of feet. This is illustrated in Figure 5-9 below, where deviations are experienced at different depths in the Azimuth parameter.

- Also, the geology formation gets a lot denser the further down the well is drilled, making it fairly more difficult to perform corrections at a higher depth. Thus, it would be of interest to search for deviations with a high sensitivity in the early phases of drilling. This implies a need for more strict and sensitive rule criteria at the start of the drilling operation, along with abilities to gradually change the criteria when needed later during the process.

These factors imply the requirement for different acceptance levels within deviations. As an approach to address this requirement, we introduce the concept of having multiple sets of rule criteria conditions within the complex event rules, where each set represents the acceptance level within a specific distance in measured depth.
In our implementation, we will design two sets of rule criteria conditions; one containing values representing the maximum allowed deviation before reaching a measured depth of 500 ft, and one containing slightly less sensitive values for deviations occurring after drilling 500 ft. These sets of rule criteria conditions will hereby be referred to as the condition sets inside their respective deviation scenario.

Naturally, it would be possible to incorporate even more sets of rule criteria conditions, to define acceptance levels conforming to the entire drilling operation of a well. However, this would make a lot more sense if the complex event rules were modeled *dynamically*, and with the ability to precisely configure the acceptance levels in advance according to the requirements of each individual well. These improvements in functionality are considered as interesting subjects for further research. They are discussed in more detail in section 8.3.2 as suggestions on how to improve the prototype.
5.5 Single events

Before we start defining the complex event rules representing the various deviation scenarios, we need to register each of the single events. Table 5-2 below shows a list of all single events relevant in the analysis for possible deviation scenarios in directional drilling. All of the events are based on real-time data input from MWD survey readings. The table has columns representing the following:

- A unique number of the event, used as identification to distinguish between the events.
- A short description of what the event represents.
- A listing of the data parameters used in the particular event.
- Different output values, describing what each event’s negative and positive value, or lack of a value, signifies.

<table>
<thead>
<tr>
<th>Event #</th>
<th>Description</th>
<th>Parameter(s)</th>
<th>Output value</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Deviation in the Northing value</td>
<td>Planned N/-S and actual N/-S</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>Deviation in the Eastings value</td>
<td>Planned E/-W and actual E/-W</td>
<td>No deviation</td>
</tr>
<tr>
<td>3</td>
<td>Total deviation in the Northing / Eastings pair</td>
<td>Event #1 and Event #2</td>
<td>No deviation</td>
</tr>
<tr>
<td>4</td>
<td>Deviation in Total Vertical Depth</td>
<td>Planned TVD and actual TVD</td>
<td>No deviation</td>
</tr>
<tr>
<td>5</td>
<td>Deviation in Vertical Section</td>
<td>Planned Vsec and actual Vsec</td>
<td>No deviation</td>
</tr>
<tr>
<td>6</td>
<td>Deviation in Dog Leg Section</td>
<td>Planned DLS and actual DLS</td>
<td>No deviation</td>
</tr>
<tr>
<td>7</td>
<td>Deviation in Inclination</td>
<td>Planned Incl and actual Incl</td>
<td>No deviation</td>
</tr>
<tr>
<td>8</td>
<td>Deviation in Azimuth</td>
<td>Planned Azim and actual Azim</td>
<td>No deviation</td>
</tr>
<tr>
<td>9</td>
<td>Change in Northing / Eastings deviation since the previous survey reading</td>
<td>Current Event #3 and Previous Event #3</td>
<td>No change</td>
</tr>
<tr>
<td>10</td>
<td>Change in TVD deviation since the previous survey reading</td>
<td>Current Event #4 and Previous Event #4</td>
<td>No change</td>
</tr>
<tr>
<td>Event</td>
<td>Description</td>
<td>Current Event</td>
<td>Previous Event</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td>---------------</td>
<td>----------------</td>
</tr>
<tr>
<td>11</td>
<td>Change in VSec deviation since the previous survey reading</td>
<td>Current Event #5 and Previous Event #5</td>
<td>No change</td>
</tr>
<tr>
<td>12</td>
<td>Change in DLS deviation since the previous survey reading</td>
<td>Current Event #6 and Previous Event #6</td>
<td>No change</td>
</tr>
<tr>
<td>13</td>
<td>Change in Inclination deviation since the previous survey reading</td>
<td>Current Event #7 and Previous Event #7</td>
<td>No change</td>
</tr>
<tr>
<td>14</td>
<td>Change in Azimuth deviation since the previous survey reading</td>
<td>Current Event #8 and Previous Event #8</td>
<td>No change</td>
</tr>
<tr>
<td>15</td>
<td>Current Measured Depth</td>
<td>MD</td>
<td>Not started drilling</td>
</tr>
<tr>
<td>16</td>
<td>Current well progression.</td>
<td>TVD + VSec</td>
<td>Not started drilling</td>
</tr>
<tr>
<td>17</td>
<td>Validity test of current survey data entry.</td>
<td>Up to 5x recent Azim + Incl values</td>
<td>True / False</td>
</tr>
</tbody>
</table>

Table 5-2: Single events

The content in the table can be explained into the following:

- **Events #1 - #8** represent an actual deviation in the corresponding input parameter. The parameters presented in these events are carefully selected from the survey readings, as they are vital in the process of determining if the well is deviating from its original path. The values are retrieved by comparing the planned parameter available in the drilling plan against the actual value stated in the current real-time survey reading.

- **Events #9 – #14** indicate a change of magnitude in an already existing deviation. The values represented by these events are based on a comparison of the current and previous deviation status retrieved from the event representing the corresponding parameter. These events will be used in combination with the *Analyze deviation outcome* technique we discussed back in section 5.4.1.

- **Event #15** contains the value in the Measured Depth parameter retrieved from the current survey reading. This value is used to identify the correct depth entry in the drilling plan, making us able to compare the current point in the progress against the plan.

- **Event #16** is designed to keep track of the current well progression, according to the technique discussed back in section 5.4.2.

- **Event #17** represents the validity of the survey reading, according to the *event windows* technique discussed back in 5.4.3. Invalid survey readings will be flagged as *false*. 


5.5.1 Relationships between the events

Table 5-3 below presents the event relationship types. As we discussed in section 4.4.1, events in CEP include relationships commonly categorized into the following:

- **Time** – Event A happened before B.
- **Causality** – Event A caused B to happen.
- **Independent** – Event A did not cause B to happen.
- **Abstraction** – The event represents a set of other events.

<table>
<thead>
<tr>
<th>Event A</th>
<th>Relationship description</th>
<th>Event B</th>
</tr>
</thead>
<tbody>
<tr>
<td>#3</td>
<td>Abstraction – Event #3 represents the combined values of events #1 and #2.</td>
<td>#1, #2</td>
</tr>
<tr>
<td>#3</td>
<td>Time and causality – Event #3 represents an already existing deviation, causing event #9 to happen.</td>
<td>#9</td>
</tr>
<tr>
<td>#4</td>
<td>Time and causality – Event #4 represents an already existing deviation, causing event #10 to happen.</td>
<td>#10</td>
</tr>
<tr>
<td>#5</td>
<td>Time and causality – Event #5 represents an already existing deviation, causing event #11 to happen.</td>
<td>#11</td>
</tr>
<tr>
<td>#6</td>
<td>Time and causality – Event #6 represents an already existing deviation, causing event #12 to happen.</td>
<td>#12</td>
</tr>
<tr>
<td>#7</td>
<td>Time and causality – Event #7 represents an already existing deviation, causing event #13 to happen.</td>
<td>#13</td>
</tr>
<tr>
<td>#8</td>
<td>Time and causality – Event #8 represents an already existing deviation, causing event #14 to happen.</td>
<td>#14</td>
</tr>
<tr>
<td>#15</td>
<td>No relationship – The current actual measured depth value is registered, which is not related to any of the other events.</td>
<td>-</td>
</tr>
<tr>
<td>#16</td>
<td>No relationship – The well progression is calculated based on values from the TVD and Vsec parameters, and is not related to any of the other events.</td>
<td>-</td>
</tr>
<tr>
<td>#17</td>
<td>Time and Abstraction – Validity of the survey reading is tested, based on the five most recent Azimuth and Inclination deviations.</td>
<td>#7, #8</td>
</tr>
</tbody>
</table>

Table 5-3: Relationship between the identified single events

5.6 Designing the deviation scenarios

We have now described all our single events. The next step is to identify and design the different deviation scenarios that might occur. To do this, we have to find what combinations of values inside the single events are required to satisfy a deviation. We will model various sets of value limits, indicating how low or high the different values are allowed to be. If all values are trespassed, it means all rule criteria are met, and we have a confirmed deviation.

We will model six different complex event rules in total, each reflecting its unique deviation scenario. The viability of the scenarios is verified through feedback from personnel with relevant knowledge working at Halliburton (Egil Aanestad, personal communication 04.12.09) (Helge Rørvik, personal communication 30.03.10).

The deviation scenarios will be presented one at the time through the following sections. Each scenario will include a description of situations where the deviation might occur, the consequence of the deviation in regards to the drilling direction, and a complex event rule designed to detect such a deviation.
5.6.1 Scenario #1 – Northings and Eastings

The first scenario is designed to detect a deviation that might occur in the combined Northings / Eastings pair. If these values are wrong, it indicates that the well has been misdirected on the horizontal plane, caused by a deviation over time in the Azimuth parameter. Given a correct vertical position, it means the well has an incorrect location in the X and Z axis in the 3D space.

Instead of only using the Northings and Eastings parameters separately in order to check for deviations, we have chosen to also combine the two to form a pair. This makes perfect sense, as they are originally used to represent coordinates on a map. The respective event representing the combined deviation value is calculated by using absolute values, enabling us to identify deviations that occur across these parameters. Figure 5-10 below illustrates four possible deviation situations in the Northings / Eastings pair.

![Diagram showing four possible deviation situations in the Northings / Eastings pair](image)

Figure 5-10: Deviations in the Northings / Eastings pair
Table 5-4 presents the complex event rule designed to detect a deviation in the Northings / Eastings pair. The table provides the following information:

- The number of the criterion.
- A logical operator stating how the criterion relates to the other rule criteria.
- The event parameter included in the criterion. The event parameters are retrieved directly from the single events that were listed in Table 5-2 above, distinguished by their identification number and description.
- The condition of the criterion, specifying the event parameter value required to meet the criterion. Notice that some of the conditions have a letter instead of a value, meaning that the value is currently unknown. This is due to the fact that each well requires different levels of deviation acceptance, as we discussed back in section 5.4.4. Hence, deviation detection must be based on well-specific value limits, and not static values as a standard for all wells.
- Additional notes connected to the specific criterion, providing details of the corresponding condition set and the specified deviation level.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#9 - Change in Northing/Easting deviation</td>
<td>&gt;= 0</td>
<td>First set starts</td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth</td>
<td>&lt; 500 ft</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#1 – Deviation in Northings</td>
<td>&gt;= X ft</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>OR</td>
<td>#2 – Deviation in Eastings</td>
<td>&gt;= X ft</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>OR</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&gt;= Y ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>6</td>
<td>OR</td>
<td>(#15 - Current Measured Depth</td>
<td>&gt;= 500 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#1 – Deviation in Northings</td>
<td>&gt;= Z ft</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>OR</td>
<td>#2 – Deviation in Eastings</td>
<td>&gt;= Z ft</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>OR</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&gt;= A ft</td>
<td>Second set ends</td>
</tr>
<tr>
<td>10</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>11</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>12</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 5-4: Rule criteria for deviation in the Northings / Eastings pair

We will use this first deviation scenario as an opportunity to thoroughly investigate the rule criteria. We will discuss what each criterion represent and how they are combined to successfully build the complex event rule. The following describes the purpose of each criterion:

1. The first criterion represents the change in deviation since the previous survey reading. The change indicates whether the deviation is increasing or decreasing. If the value from this event is higher than 0, we know the deviation has increased since the previous survey reading, meaning that the deviation is developing in a negative direction.

2. The second criterion starts with an opening parenthesis, signifying the first of our two condition sets. According to what we discussed in section 5.4.4, the first condition set applies to wells drilled less than 500 ft of measured depth, while the second set applies after reaching a measured depth of 500 ft or more.

3 – 4. These rule criteria represent the maximum allowed deviation applied to both the Northings and Eastings as separate parameters.
5. Holds the maximum deviation value allowed in the combined Northings / Eastings pair. Notice that this value differs from the deviation allowed in the separate parameters, signified by the Y symbol. This criterion ends the first condition set, indicated by the enclosed parenthesis.

6 – 9. Represents the second condition set. The criteria included here are the same as in the first condition set, although with different values.

10 -11. Criteria representing the current estimated well progression. As we can see, one of these criteria will always be true, as they include scenarios of both less and more than 80% of the total distance drilled. Thus, the well progression will not be used as a factor in determining whether a deviation has occurred, but rather as an indication of the seriousness of the deviation, used to suggest a proper follow-up decision.

12. This final criteria shows whether the survey data received can be considered reliable or not, as we discussed back in section 5.4.3.

5.6.2 Scenario #2 – Azimuth

An erroneous azimuth implies that the well is drilled in a wrong direction in the horizontal plane, as illustrated in Figure 5-11 below. The illustration to the left in the figure presents a top-down 2D view, where it is revealed that the azimuth leans too much towards north and east. The illustration to the right shows how this affects the well’s position in the X and Z axis through a 3D perspective, indicating that the well has been drilled too far on the Z axis.

Figure 5-11: Deviation in the Azimuth. Top-down 2D-view to the left, 3D-view to the right
Even though this deviation scenario (#2) has large similarities with the scenario described in context of a deviation in Northings / Eastings (#1), there are some essential differences between the two:

- An abrupt deviation may occur in the Azimuth without there being any indications of a deviation within the Northings / Eastings pair. Instead, an incorrect Azimuth will ultimately lead to a deviation in the Northings / Eastings pair after drilling for a while, if left unattended. This scenario (#2) is designed to detect exactly these situations.
- The Azimuth may contain a continuous and steady deviation that is too small to be detected by the complex event rule. This will eventually, after drilling for a certain period of time, build a considerable deviation in the Northings / Eastings pair, until it’s detected by the complex event rule designed in the previous deviation scenario (#1).

Table 5-5 represents the complex event rule designed to detect a deviation in Azimuth.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#14 - Change in Azimuth deviation</td>
<td>&gt;= 0</td>
<td>First set starts</td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth</td>
<td>&lt; 500 ft</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#8 - Deviation in Azimuth</td>
<td>&gt;= X ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>4</td>
<td>AND</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&lt; 3 ft</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>OR</td>
<td>(#15 - Current Measured Depth</td>
<td>&gt;= 500 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>6</td>
<td>AND</td>
<td>#8 - Deviation in Azimuth</td>
<td>&gt;= Y ft</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&lt; 3 ft</td>
<td>Second set ends</td>
</tr>
<tr>
<td>8</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>9</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>10</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 5-5: Rule criteria for deviation in Azimuth

As we can see from the definition of the complex event rule in Table 5-5 above, a deviation will only be found in situations where the Northings / Eastings value is not already deviating considerably. The reason for this is that if the Azimuth parameter is reported with a deviating value, while the Northings / Eastings values are almost correct, we know that the deviation has a good chance of escalating within the next survey reading, caused by the incorrect Azimuth. However, if the Northings / Eastings pair is also deviating by a considerable amount, the Azimuth could be set to deviate on purpose in order to later achieve Northings / Eastings values consistent with the drilling plan. Hence, no deviation should be detected.

Also notice that the first and second condition set in the Azimuth scenario by default signify a level 3- and 2 deviation, respectively. An incorrect value, either in the Azimuth or Inclination parameters during the early phase of a drilling process, may quickly evolve into a significant deviation, as we demonstrated back in Figure 5-9.
5.6.3 Scenario #3 – Dog Leg Section

A deviation in the Dog Leg Section means that the well has been drilled with a too high or too low directional curvature, either horizontally or vertically, during a limited distance. (Egil Aanestad, Halliburton, personal communication 04.12.09). If this is not adjusted, it could either lead to wrong values in Azimuth and Northings / Eastings, as we covered in the two previous scenarios, or it could result in an erroneous Inclination, which in turn will affect the vertical depth of the well.

Table 5-6 represents the complex event rule designed to detect a deviation in Dog Leg Section.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#12 - Change in DLS deviation</td>
<td>&gt;= 0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth</td>
<td>&lt; 500 ft</td>
<td>First set starts</td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#6 - Deviation in Dog Leg Section</td>
<td>&gt;= X ft</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt; 0 ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>5</td>
<td>OR</td>
<td>(#15 - Current Measured Depth</td>
<td>&gt;= 500 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>6</td>
<td>AND</td>
<td>#6 - Deviation in Dog Leg Section</td>
<td>&gt;= Y ft</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt; 0 ft</td>
<td>Second set ends</td>
</tr>
<tr>
<td>8</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>9</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>10</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 5-6: Rule criteria for deviation in Dog Leg Section

To accurately calculate whether there is an actual deviation in DLS, or if the parameter is intentionally set to deviate by the directional driller to compensate for a more optimal drilling path, we need to keep track of which way the TVD value is deviating as well, as shown in criteria 4 and 7 in the table. If the TVD value is deviating in the opposite direction of the DLS value, then it is fair to assume a real deviation is in progress. However, if the TVD value is deviating in the same direction, it means the well is about to catch up with the planned drilling path, which is a beneficial outcome. This is illustrated in Figure 5-12, where we can see the TVD value is supposed to be higher (drilled deeper) than it actually is.
5.6.4 **Scenario #4 – Inclination**

As mentioned earlier, the inclination parameter represents the overall directional angle of the entire well. If this value deviates from the planned path, it could result in the following:

- **Too high inclination** results in the horizontal length of the well getting too high, which gives too high value in the VSec parameter and too low TVD, as was illustrated in Figure 5-1 earlier in this chapter.
- **Too low inclination** leads to the exact opposite; the VSec gets too low value, while the TVD gets too high.

This deviation scenario has close resemblance to the scenario of a deviation in Dog Leg Section, with both parameters being used to measure vertical curvature in degrees. The difference is that the Dog Leg Section represents quite narrow points of distance of the well (approximately every 100 ft), while the Inclination parameter indicates the vertical angle throughout the whole well. This means that the Inclination may be deviating even though the DLS is currently correct. Similarly, the DLS may indicate a sudden and abrupt deviation occurring even though no deviations are registered in the Inclination parameter. Hence, both scenarios are needed.
Table 5-7 represents the complex event rule designed to detect a deviation in Inclination.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#13 - Change in Inclination deviation</td>
<td>&gt;= 0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth)</td>
<td>&lt; 500 ft</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#7 - Deviation in Inclination</td>
<td>&gt;= X ft</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt; 0 ft</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>OR</td>
<td>(#15 - Current Measured Depth)</td>
<td>&gt;= 500 ft</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>AND</td>
<td>#7 - Deviation in Inclination</td>
<td>&gt;= Y ft</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt; 0 ft</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 5-7: Rule criteria for deviation in Inclination

As seen in criteria 4 and 7, we take any occurring TVD deviations in consideration when figuring out if the inclination deviation has a negative or positive outcome, just as we did in the scenario of a deviation in Dog Leg Section (#3).

### 5.6.5 Scenario # 5 – Total Vertical Depth

If the actual vertical depth differs from the planned value, it means that the well has been drilled horizontally either too early or too late, due to an incorrect Inclination. If this is not corrected, the well might end up above or beneath the target.

Table 5-8 represents the complex event rule designed to detect a deviation in Total Vertical Depth.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#10 - Change in TVD deviation</td>
<td>&gt;= 0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth)</td>
<td>&lt; 500 ft</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt;= X ft</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>OR</td>
<td>(#15 - Current Measured Depth)</td>
<td>&gt;= 500 ft</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt;= Y ft</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 5-8: Rule criteria for deviation in Total Vertical Depth

When searching for deviations in the TVD parameter, we only need to look at the current deviation present in TVD. Once the well has exceeded its maximum allowed deviation in regards to the TVD parameter, we already know there has to be a deviation in Inclination directed with a negative outcome.
5.6.6 Scenario #6 – Vertical Section

Vertical Section represents how far the well has been drilled towards the target horizontally, much in the same way the TVD represents the total distance drilled vertically.

Taking this into account, a wrong value in the VSec parameter could mean the following:

- **Too high value** – There has been too much inclination, leading to a deviation caused by a low TVD value. This particular deviation is already modeled in the scenario of a deviation in Total Vertical Depth (#5).

- **Too low value** – May be caused by two factors:
  - Too low inclination, resulting in a high TVD.
  - Wrong Azimuth, leading to incorrect Northings / Eastings, which is already modeled in the scenario of a deviation in Northings / Eastings (#1).

This scenario will be designed to detect deviations occurring in the combined TVD and Northings / Eastings values, whenever the VSec parameter is too low. This allows for deviation detection across both the horizontal and vertical plane.

Table 5-9 represents the complex event rule designed to detect a deviation in Vertical Section.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#11 - Change in VSec deviation</td>
<td>&gt;= 0</td>
<td>First set starts</td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth</td>
<td>&lt; 500 ft</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&gt;= X ft</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>AND</td>
<td>#4 – Deviation in Total Vertical Depth</td>
<td>&gt;= Y ft</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>AND</td>
<td>#5 - Deviation in Vertical Section</td>
<td>&lt; 2 ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>6</td>
<td>OR</td>
<td>(#15 - Current Measured Depth</td>
<td>&gt;= 500 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&gt;= A ft</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>AND</td>
<td>#4 – Deviation in Total Vertical Depth</td>
<td>&gt;= B ft</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>AND</td>
<td>#5 - Deviation in Vertical Section</td>
<td>&lt; C ft</td>
<td>Second set ends</td>
</tr>
<tr>
<td>10</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>11</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>12</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 5-9: Rule criteria for deviation in Vertical Section

As we can see from criteria 3-6 and 7-9 in the table, a deviation is detected if both the TVD and Northings / Eastings pair are deviating, while the VSec parameter has a value that is too low.
5.7 Responsibility area of the deviation scenarios

Table 5-10 summarizes the differences in the various deviation scenarios’ responsibility area, describing what situations the respective scenario will detect.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Responsibility are</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1 – Northings and Eastings</td>
<td>Detects deviations in the horizontal plane that are built over a certain period of time, caused by a continuous, though small deviation in Azimuth.</td>
</tr>
<tr>
<td>#2 - Azimuth</td>
<td>Detects abrupt and severe deviations in the horizontal plane, before a significant deviation has occurred in the Northings / Eastings parameters.</td>
</tr>
<tr>
<td>#3 – Dog Leg Section</td>
<td>Detects abrupt deviations in vertical curvature, measured over a distance of approximately 100 ft. Used to detect deviations before a significant deviation has occurred in Inclination and Total Vertical Depth.</td>
</tr>
<tr>
<td>#4 - Inclination</td>
<td>Detects deviations in the total vertical curvature, measured from the start to end of the well. Used to detect deviations not found by the scenario of Dog Leg Section, before a significant deviation has occurred in Total Vertical Depth.</td>
</tr>
<tr>
<td>#5 – Total Vertical Depth</td>
<td>Detects deviations in the vertical plane that are built over a certain period of time, caused by a continuous, though small deviation in Inclination.</td>
</tr>
<tr>
<td>#6 – Vertical Section</td>
<td>Detects deviations across the horizontal / vertical plane, based on the combined deviations in Total Vertical Depth and the Northings / Eastings pair.</td>
</tr>
</tbody>
</table>

Table 5-10: Differences in the deviation scenarios’ responsibility area.

5.8 Triggered actions

As mentioned at the beginning of this chapter, a detected deviation from any of the complex event rules will trigger the deviation handling process that is implemented in K2 BlackPearl by the CODIO research group. The idea is to provide the end-user with an interface realized through Microsoft SharePoint, supporting detailed overview of well-specific information. The solution is built upon case management, adding any occurred deviations to the well’s case folder.
6 Implementing the prototype

This chapter will concentrate on giving the reader a detailed and profound insight in how the Event Driven BPM prototype is implemented, with focus on the architectural design, the technical aspects of the implementation, and how the relevant K2 workflow logic is developed.

We can divide the prototype into two separate components, the CEP Engine and the K2 BlackPearl solution.

- The implementation of the CEP Engine will to a large extent be described in combination with important code snippets, with the purpose to illustrate in detail how the CEP Engine is designed and built in order to achieve the required event processing goals.
- Implementation of the K2 BlackPearl solution is described with a distinct focus on the process workflow.

6.1 Prototype architecture

To be able to encapsulate event-driven behavior within our prototype, we have based our implementation on the concept of Event-Driven Architecture (EDA); a software architecture pattern designed to facilitate the production, detection, consumption of, and reaction to single events and patterns of events (Wikipedia, 2010a). An Event-Driven Architecture consist of four logical layers; the Event generator, Event channel, Event processing Engine, and Downstream event-driven behavior. Figure 6-1 below illustrates how the prototype developed in this thesis conforms to these layers.

Figure 6-1: Prototype architecture, based on EDA.
6.1.1 Event generator

The first logical layer is represented by the event generator, sensing predefined measurable facts, or a change in state, which are both considered and treated as events. The event generator proposed in our prototype implies the use of MWD sensors, providing real-time updates of survey data.

The final product is intended to base all calculations on events obtained from real survey data, consecutively received from the MWD tool used in the actual drilling operation. This has not been implemented in the prototype during the work on this thesis, due to the lack of accessibility to such tools. Instead, in order to achieve the ability to adequately perform experiments, we have arranged a simulation of constantly incoming survey readings, making the prototype read one survey entry at the time from an already completed survey report.

6.1.2 Event channel

The event channel provides the mechanism required to transfer the information about the detected events to the event engine. In a real-life execution of the prototype, the event channel would represent the appropriate connectivity methods to access the MWD tool’s mud pulse and electromagnetic telemetry.

6.1.3 Event processing engine

The third logical layer consists of the actual event processing engine, where events are analyzed and combined in order to be compared to event patterns of interest. The event processing engine triggers the proper reactions depending on the results from event pattern comparison.

The central logic behind the event processing engine implemented in our prototype is developed through C# code, as illustrated in our proposed architecture in Figure 6-1. A more detailed discussion of the architectural design behind the CEP Engine part of the prototype is included in section 6.2.2 later in this chapter.

6.1.4 Downstream event-driven activity

The final logical layer contains the downstream event-driven activity, which specifies what should happen if any event patterns of interest occurs. This specification includes a description of what actions should be triggered, what tasks should be performed, and what personnel should be involved in each task.

Our prototype is instructed to trigger the deviation handling process whenever an event pattern is recognized, providing information that grants the directional driller increased situational awareness regarding the detected problem. The activity flow in this business process, modeled and executed in K2 BlackPearl, provides a detailed insight in proper consecutive actions when dealing with a deviation.

As mentioned earlier, the K2 BlackPearl solution is being designed and implemented by the CODIO WP1 project team. Functional logic representing the business process steps included in the workflow is currently under development, including detailed configurations of the execution of each task involved in the process. Also, concepts such as role- and user specific work-lists, and comprehensive well-sites providing detailed insight in well-specific data and case history, is being designed and enabled within the K2 solution.
Unfortunately, we have not been able to incorporate our prototype into the CODIO K2 solution during the work on this thesis, due to limited time and resources available in the CODIO project. Instead, we have designed a lightweight K2 process on our own, in order to demonstrate the downstream event-driven activity of our prototype. As we can see from the proposed architecture in Figure 6-1, our K2 implementation consists of a K2 process solution (DeviationHandling.kprx) accompanied with two aspx .NET websites representing the most relevant business process step. Our K2 implementation is discussed and presented in more detail in section 6.3.

6.2 Implementing the CEP Engine

The CEP Engine is the prototype component responsible for performing automatic detection of deviations. It includes algorithms developed to examine incoming survey data, in order to check for complex event patterns. This section explains and illustrates how the CEP Engine is realized through programming code. Important code snippets will be presented and explained.

6.2.1 Choosing the CEP Engine

A certain variety of existing CEP software solutions is available for use, offering feature-rich event processing solutions both for commercial and non-commercial needs. The most interesting CEP solution in context of the prototype developed in this thesis is called Esper, which is an open source component based on the Java Programming Language (EsperTech, 2007). The purpose of Esper is to offer extensive and dynamic CEP functionality to satisfy typical needs in event processing tasks. Event processing logic is used by modeling event rules that analyze event streams in real-time. Esper’s capabilities are utilized through using a script language resembling common SQL syntax.

Even though the Esper engine seems to be a promising candidate for fulfilling the complex event processing required in the prototype, the decision has been made to engineer a tailored, custom made CEP Engine from scratch. There are several reasons behind this decision:

- First of all it would require a great amount of time and resources to learn how to effectively utilize the Esper engine, both in terms of correct syntax, proper use of the API, viable implementation of the interface towards K2 BlackPearl, and installation and configuration of the engine. These factors probably outweigh the benefits of using Esper, when compared to creating our own custom made CEP engine from scratch (Information Management, 2006).
- Secondly, the Esper engine comes with a wide range of CEP functionality that will not be used in the event processing tasks included in this prototype, which could result in an unnecessary complex and messy solution. By writing custom code, we can engineer a fully tailor-made CEP solution containing the exact functionality needed for our purpose.
- The event processing tasks in this prototype require a tier 1 CEP solution, supporting basic event processing functionality. This is fairly easy and straightforward to implement using custom code (Information Management, 2006).
- We will not operate with a large set of events. In fact, we are only going to process a handful of events received whenever the real-time survey tool has updates available, which usually happens within time-intervals of several minutes (Egil Aanestad, Halliburton, personal communication 04.12.09). This eliminates the need for advanced algorithms yielding rapid processing performance.
The CEP Engine is realized through writing custom code in the C# programming language inside the environment of Visual Studio 2005.

### 6.2.2 CEP Engine Architecture

The UML class diagram shown in Figure 6-2 below represents the complete architecture of the CEP Engine part of the prototype.

---

**Figure 6-2: CEP Engine class diagram**
As we can see from the class diagram, the CEP Engine is built using three custom developed classes; SurveyEntry, Events, and CepEngine.

### 6.2.2.1 The SurveyEntry class

The SurveyEntry class basically works as an object container for the purpose of storing updated data values received from each survey reading. Both the current and the previous collection of data from survey readings are stored within their respective instance of this class for further processing. The functions implemented in the SurveyEntry class are exclusively methods to get and set survey entry properties.

### 6.2.2.2 The Events class

The Events class operates as a container much like the SurveyEntry class, designed to store processed single events. Data retrieved from SurveyEntry objects is processed and refined into events of interest. The data members of this class represent each of the events that were defined back in Table 5-2, except the validity test of survey readings (Event #17). Also in this class, all the functions present are methods to get and set all the data members.

### 6.2.2.3 The CepEngine class

The main logic behind the CEP Engine is developed within the CepEngine class. This class contains a variety of functions, all contributing to achieve the desired event processing functionality. The functions and their parameters are explained in detail in Table 6-1 below.

<table>
<thead>
<tr>
<th>Function name</th>
<th>Parameter(s)</th>
<th>Purpose of the function</th>
</tr>
</thead>
<tbody>
<tr>
<td>CepEngine()</td>
<td>-</td>
<td>Class constructor, initiates the CEP Engine.</td>
</tr>
<tr>
<td>RegisterDrillingPlan()</td>
<td>-</td>
<td>Function designed to register the drilling plan, by storing each plan entry into an ArrayList object. Returns void.</td>
</tr>
<tr>
<td>SearchForCorrespondingPlanEntry()</td>
<td>double MD: Measured Depth value, received from current survey reading.</td>
<td>Function designed to locate a survey entry’s corresponding plan entry, if any. Returns the drilling plan entry as a SurveyEntry object.</td>
</tr>
<tr>
<td>SetTargetValues()</td>
<td>-</td>
<td>Function designed to identify and set the target values that are used when calculating well progression. Target values are retrieved from the drilling plan. Returns void.</td>
</tr>
<tr>
<td>SetEventWindowAverageValue()</td>
<td>EventWindow EW: The event window representing up to five consecutive values in Inclination or Azimuth.</td>
<td>Function designed to set the average deviation value, based on the five previous consecutive Inclination or Azimuth deviations. Returns the EventWindow after processing.</td>
</tr>
<tr>
<td>CheckForSuspiciousSurveyReading()</td>
<td>-</td>
<td>Function designed to investigate whether the deviation present in Inclination and Azimuth in the current survey reading exceeds the maximum allowed value when compared to the average deviation in these parameters. Used to filter out invalid survey readings. Returns a bool.</td>
</tr>
<tr>
<td>Function</td>
<td>Parameters</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------------------------------------------------</td>
<td>------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>UpdateEventWindows()</td>
<td>-</td>
<td>Function designed to update the event windows, adding the new values (if not a suspicious survey reading), and removing the oldest value (if the maximum number of window entries have been reached).</td>
</tr>
<tr>
<td>ResetEventWindows()</td>
<td>-</td>
<td>Function designed to reset the event windows. The function is used when consecutive survey readings are found suspicious (as this might indicate the suspicious survey readings are correct after all).</td>
</tr>
<tr>
<td>RegisterEvents()</td>
<td>-</td>
<td>Function designed to register all new events, based on the current survey reading. Returns void.</td>
</tr>
<tr>
<td>FindAzimuthDeviation()</td>
<td>double plan_azim: The Azimuth retrieved from the plan entry. double survey_azim: The Azimuth retrieved from the survey entry.</td>
<td>Function designed to calculate the actual deviation in azimuth. The azimuth parameter may only have a value that is higher than 0° and lower than 360°. Returns a double value.</td>
</tr>
<tr>
<td>RunCEP()</td>
<td>-</td>
<td>Function designed to run the sequential logic involved when searching for deviations. The sequential logic starts with reading a new survey entry. Returns void.</td>
</tr>
<tr>
<td>SearchForLevel2ComplexEvents()</td>
<td>-</td>
<td>Function designed to search for level 2 Complex Events. Returns the result as a bool value.</td>
</tr>
<tr>
<td>SearchForLevel3ComplexEvents()</td>
<td>-</td>
<td>Function designed to search for level 3 Complex Events. Returns the result as a bool value.</td>
</tr>
<tr>
<td>FinalizeOutputMessage()</td>
<td>int level: A number determining the level of the deviation, used to decide a proper follow-up.</td>
<td>Function designed to add general information to an output message, such as current Measured Depth and Well Progression. Returns void.</td>
</tr>
<tr>
<td>BuildDeviationMessage()</td>
<td>int Deviation: A number determining which deviation scenario occurred.</td>
<td>Function designed to insert useful information regarding the deviation that occurred. Returns void.</td>
</tr>
</tbody>
</table>

Table 6-1: Functions and parameters of the CepEngine class
6.2.3 Register drilling plan

Note that since we have been unable to establish a connection to an actual survey tool, we have to simulate incoming survey entries from a completed survey report. Thus, data from both the drilling plan and the survey entries can be obtained from regular text documents.

The first thing the CEP Engine should do is to register the drilling plan. The drilling plan is supposed to function as the frame of reference when calculating possible deviations in the drilling path, thus making the establishment of a viable plan a vital task before moving on to the actual calculations. To read from the drilling plan document, we create an object instance of the TextReader class embedded in C#. The TextReader class takes the actual drilling plan document as an input parameter.

To register the drilling plan properly, we need to create an object with the capability to store all the drilling parameters needed from the plan entry. Since the survey entries and drilling plan entries include identical parameters, we can simply use the SurveyEntry class for the purpose of storing each drilling plan entry. The plan entries are eventually inserted to the drilling plan, represented by a C# ArrayList object as a collection of plan entries (see Code Snippet 6-1 below).

```csharp
// A new new plan entry is created. // The plan parameters are retrieved from a table used in // connection with reading the plan entry. // In our case, InvariantCulture has to be declared, // as the drilling plan template is compiled using U.S. standards.
PlanInput = new SurveyEntry(
    // Northings
    Double.Parse(Parameters[6].ToString(), CultureInfo.InvariantCulture),
    // Eastings
    Double.Parse(Parameters[7].ToString(), CultureInfo.InvariantCulture),
    // Measured Depth
    Double.Parse(Parameters[1].ToString(), CultureInfo.InvariantCulture),
    // Total Vertical Depth
    Double.Parse(Parameters[5].ToString(), CultureInfo.InvariantCulture),
    // Vertical Section
    Double.Parse(Parameters[10].ToString(), CultureInfo.InvariantCulture),
    // Dog Leg Section
    // Inclination
    Double.Parse(Parameters[2].ToString(), CultureInfo.InvariantCulture),
    // Azimuth
    Double.Parse(Parameters[3].ToString(), CultureInfo.InvariantCulture)
);

// Adding the plan entry to the drilling plan.
DrillingPlan.Add(PlanInput);

// Done with this plan entry, read a new line.
PlanEntry = PlanReader.ReadLine();
```

**Code Snippet 6-1: Inserting a plan entry into the drilling plan**
6.2.4 Register events

The next step in the process is to register all the events we defined in Table 5-2 back in our case description in Chapter 5. As we remember from the table, Events #1 - #8 represent deviations in each respective drilling parameter of interest. These events, along with Event #15 – Current Measured Depth, are calculated by simply subtracting the current plan entry value from the value in the current survey reading, as demonstrated in Code Snippet 6-2 below.

```csharp
// Event #1 - Deviation in the Northing value
```

**Code Snippet 6-2: Calculating a parameter deviation.**

*Events #9 - #14* represent an eventual change in deviation since the previous survey reading, where positive and negative values mean that the deviation has increased or decreased, respectively. If previous events have been recorded, the change in deviation is calculated by subtracting the previous value from the current one while using absolute values, as demonstrated in Code Snippet 6-3 below.

```csharp
// Event #14 - Change in Azimuth Deviation.
CurrentEvents.Azim_Change = Math.Abs(CurrentEvents.Azimuth) - Math.Abs(PreviousEvents.Azimuth);
```

**Code Snippet 6-3: Calculating changes in a parameter deviation.**

As mentioned earlier in section 5.4.2, Event #16 – Current well progression, given in % is estimated through a calculation of the sum of the TVD and VSec parameters in the current survey reading, divided on the sum of the final target’s TVD and VSec parameters, as shown in Code Snippet 6-4 below.

```csharp
// Event #16 - Current well progression, given in %
CurrentEvents.Well_Progression = (CurrentPlanReading.Total_Vertical_Depth + CurrentSurveyReading.Vertical_Section) / (TVD_Target + VSec_Target);
```

**Code Snippet 6-4: Calculating well progression.**

The final event, Event #17 – Validity test of current survey data entry, will be calculated within the event window technique, as described in detail in section 6.2.5.

6.2.5 Implementing the Event windowing technique

Back in section 5.4.3, we discussed the benefits of implementing a technique called Event windows in order to discover and reject potential erroneous survey readings. We already know that if one or more of the values registered in the survey reading happens to be wrong, the flawed value needs to be present in one of the raw parameters measured by the MWD tool; either in Inclination or Azimuth.
We choose to implement Event Windows as a *struct*, a value type suitable for representing lightweight objects (MSDN, 2010). As we can see in Code snippet 6-5, the struct representing our EventWindow contains three data members:

- An `ArrayList` object representing the event values included in the window.
- An `int` keeping track of suspicious survey readings.
- A `Double` representing the average value of the window members.

```csharp
public struct EventWindow
{
    public ArrayList Window;
    public int Suspicious_Reading;
    public Double Average_Value;
}
EventWindow Incl_Window,
Azim_Window;
```

**Code snippet 6-5: The Event Window struct.**

By suspicious survey readings, we mean readings that have increased by a value exceeding a certain predefined maximum value, as illustrated with an example in Code snippet 6-6 below. The `Suspicious_Reading` variable will increase by 1 whenever a possibly flawed reading is found. If this variable has a value of 1, it means the current survey reading should be considered invalid. A further increase of this value may indicate that the survey reading was correct after all. If a complex event is triggered based on this survey reading, the sudden and unlikely increase in deviation will be commented and included in the deviation report.

```csharp
// Check if this is a suspicious survey reading, i.e. the reading value
// differs with more than a maximum value when subtracting the
// average value of the events inside the window.
if (Math.Abs((CurrentSurveyReading.Inclination
    - CurrentPlanEntry.Inclination) - Incl_Window.Average_Value) > 3)
{
    Incl_Window.Suspectious_Reading++;
    return true;
}
```

**Code snippet 6-6: Identification of a suspicious survey reading.**
Average values need to be calculated and updated according to consecutively incoming survey readings. The calculation is done by retrieving the value from each event residing inside the window, as shown in Code snippet 6-7.

```csharp
public void SetEventWindowAverageValue(EventWindow EW)
{
    // Check if the windows contain any values, and abort if not.
    if (EW.Window.Count > 1)
    {
        EW.Average_Value = 0;
    
        // Calculate the average value.
        for (int i = 0; i < EW.Window.Count; i++)
            EW.Average_Value += (double)EW.Window[i];

        EW.Average_Value = EW.Average_Value / EW.Window.Count;
    }
}
```

**Code snippet 6-7:** Calculation of average values in the event window.

Each survey reading not considered suspicious is added to the Event Window (see Code snippet 6-8). Whenever the window is filled, the oldest event present is removed in favor of the new event.

```csharp
// Check if there are readings present inside the window,
// and remove the oldest occurrence if the window is full (5 occurrences).
if (Incl_Window.Window.Count != 0 && Incl_Window.Window.Count > 4)
    Incl_Window.Window.RemoveAt(Incl_Window.Window.Count - 1);
Incl_Window.Window.Add(CurrentEvents.Inclination);
SetEventWindowAverageValue(Incl_Window);
```

**Code snippet 6-8:** Adding survey readings to the event window.

### 6.2.6 Implementing the complex event rules

The complex event rules are manually coded within C# if-conditions, where all event criteria of the particular rule (see section 5.6) are tested with logical operators. If a rule is triggered, a proper deviation report is built that will be delivered to the directional driller. A demonstration of how the CEP rules are implemented is shown in the example of *Deviation scenario #6 – Deviation in Vertical Section* in Code Snippet 6-9 below. The values inserted in the criteria correspond to an imaginary well with two different acceptance levels.
// Complex Event #6 - Deviation in Vertical Section, level 2.
  // Build the deviation report
  BuildDeviationMessage(4);
  FoundLevel2ComplexEvent = true;
}


6.2.7   Locating the correct plan entry
Before calculating the deviations, we need to make sure we are comparing the current survey reading entry to the correct plan entry. To guarantee this, we will use the Measured Depth parameter obtained from the current survey reading as a frame of reference. By doing so, we can navigate through the drilling plan until we find the most accurate match.

The first thing we need to do is to declare SurveyEntry objects for the current and the previous plan entry, as shown in Code Snippet 6-10. This makes us able to keep track of which of the plan entries is the most appropriate match compared to the current survey entry.

```csharp
SurveyEntry currentPlan = (SurveyEntry)DrillingPlan[0],
previousPlan;
```

Code Snippet 6-10: Declaring current and previous drilling plan entries.

Next, we need to check if there exists a plan entry for the current survey entry at this particular depth. If the MD parameter in the very first plan entry proves to be higher than the MD parameter in the current survey entry, we are forced to abort the search, since it will be impossible to find a proper frame of reference (see Code Snippet 6-11).

```csharp
if (currentPlan.Measured_Depth > SurveyMD) return null;
```

Code Snippet 6-11: Aborting the search for a corresponding plan entry.

On the other hand, if the MD parameter in the first plan entry appears to be lower than the MD parameter in the current survey entry, we may proceed by looping through to the end of the plan until we get a match. If the MD parameter in the plan entry suddenly gets higher than the MD parameter in the survey entry, we know we have found a match. However, if we end up traversing through the entire plan without locating a match, we have to abort since the well has been drilled too far, and no match can be found.

The search sequence is further explained in Table 6-2 below. The code behind the algorithm used to search for the best plan entry match is shown in Code Snippet 6-12.
<table>
<thead>
<tr>
<th>Measured Depth at plan entry</th>
<th>Measured Depth at survey entry</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>First plan entry = MD of 400ft</td>
<td>Current survey entry = MD of 200ft</td>
<td>Current survey entry has a lower MD than the lowest (first) plan entry. No corresponding plan entry can be found, because the well has not drilled far enough. The algorithm aborts.</td>
</tr>
<tr>
<td>First plan entry = MD of 100ft</td>
<td>Current survey entry = MD of 200ft</td>
<td>A corresponding plan entry may potentially be found. The algorithm starts to loop through each plan entry to check for a proper match.</td>
</tr>
<tr>
<td>Current plan entry = MD of 2300ft</td>
<td>Current survey entry = MD of 2300ft</td>
<td>A corresponding plan entry has been found. The algorithm stops the search and returns the result.</td>
</tr>
<tr>
<td>Final plan entry = MD of 7000ft</td>
<td>Current survey entry = MD of 7100ft</td>
<td>Current survey entry contains a higher MD than the final plan entry. No corresponding plan entry can be found, since the well has drilled too far. The algorithm aborts.</td>
</tr>
</tbody>
</table>

Table 6-2: Examples of algorithm behavior when searching for the correct plan entry.

```csharp
// Loop through entire drilling plan to find the corresponding plan entry.
for (int i = 0; i < DrillingPlan.Count; i++)
{
    current = (SurveyEntry)DrillingPlan[i];
    // Keep looping until a plan entry with a higher MD value is found.
    if (currentPlan.Measured_Depth < SurveyMD)
        continue;
    previousPlan = (SurveyEntry)DrillingPlan[i - 1];
    // Select the most immediate value.
    if ((currentPlan.Measured_Depth - SurveyMD) <
        (SurveyMD - previousPlan.Measured_Depth))
        return currentPlan;
    else
        return previousPlan;
}
// Abort if no matching plan entries are found for this particular survey entry. (SurveyMD is too high).
return null;
```

Code Snippet 6-12: Algorithm to locate a corresponding drilling plan entry.

### 6.2.8 Building the deviation report

When a deviation is discovered, we have to build a report that sufficiently describes all the important details of the deviation scenario. This report will be available to the role responsible for handling deviations, and will act as a frame of reference when determining whether an actual deviation is found, and whether the suggested follow-up is optimal.

The deviation report will contain both contextual information regarding the specific deviation, and general details about the current drilling situation.
6.3 The K2 BlackPearl solution
Let’s look at the K2 BlackPearl solution component implemented in the prototype. As we discussed back in section 5.1, the prototype is designed to initiate a BPMN process called *Handle deviation in directional drilling* whenever a deviation is detected.

6.3.1 The workflow
The workflow model in Figure 6-3 below represents the BPMN process modeled in K2 BlackPearl.

![Figure 6-3: K2 workflow model of “Handle deviation in directional drilling”.

---
As we can see, the graphical representation of the workflow differs a lot from the traditional representation of a BPMN workflow.

6.3.1.1 *K2 Activities and Events*

In K2, business process steps, called *activities*, are represented as boxes, with the possibility to include *K2 events*. There may be multiple K2 events inside each process step, and there are a variety of different types. In our model, the following K2 events have been used:

- **Client Event** – The Client Event is used in cases where a human user physically interacts with the process instance. The Client Event is used on several occasions in our K2 model, like for instance when validating deviation results from the CEP Engine, or when organizing a collaboration session.
- **Server Event** – The Server Event is used to automatically execute code that resides on the K2 server component, without any kind of human interaction. Our K2 model makes use of the Server Event in tasks like automatically creating a case, or to update the well case history.
- **Mail Event** – Mail Events are being used to send automatic notifications by e-mail to roles and individuals. In our model, Mail Events are utilized to send notifications on requests for updated survey readings, and to invite participants to a collaboration session, among others tasks.

Figure 6-4 below gives an illustration of the icons associated with the mentioned events.

![Figure 6-4: K2 Events; Client Event – Server Event – Mail Event.](image)

6.3.1.2 *Representation of roles*

Another thing to notice in the K2 workflow model is that lanes representing the different roles are omitted. Instead, each role has been distinctly separated by having a unique color associated. In our model, the business process steps that belongs to the Drilling Supervisor has been assigned a blue color, the Directional Driller a red color, the Geologist a green color, and the CODIO Process Engine a yellow color. Process steps with a grey color, such as the start- and end event, are not associated with any role.

6.3.2 *Implementation of process logic*

The entire logic behind this workflow is currently under development by the CODIO research group. As mentioned earlier, we have not been able to incorporate the prototype functionality developed in this thesis with the product engineered by the CODIO team. Instead, we have created a lightweight K2 implementation to demonstrate the event-driven behavior. Only a few business process steps have been fully implemented with working logic. These process steps include the *Start Event*, *Validate results*, and *Organize collaboration*. 

74
6.3.2.1 The Start Event

As stated in section 5.1.2, the Start Event may be initiated either by the CEP Engine or manually by a person monitoring real-time data. Either way, as soon as the Start Event is triggered, the K2 process will automatically add a new worklist item to the Directional Driller’s Worklist inside K2 BlackPearl’s Workspace. The workspace is a browser-based interface that allows users to access and manage pending tasks, and view information about business processes (SourceCode Technology Holdings, n.d.)

Clicking on a worklist item will process the task. In our case, the processing of the worklist item will lead the Directional Driller to the next step in the process; Validate results.

It is also possible to further configure the start event to automatically send an e-mail to the desired roles, notifying them about the discovered deviation and their new pending worklist item.

6.3.2.2 Validate results and Organize collaboration

The Client Event in Validate results represents the Directional Driller’s task of investigating the deviation results, either retrieved from the CEP Engine or from ‘some person monitoring real-time data’. Verifications must be made on whether an actual deviation has occurred, and whether the selected follow-up is the most appropriate. The Organize collaboration activity represents the task of inviting participants to a collaboration session, in cases where a level 3 deviation is confirmed.

A simple Graphical User Interface (GUI) connected to these business process steps are designed and implemented through ASP.NET websites, as shown in the architecture back in Figure 6-1. The GUI comprise the elements required to perform the tasks contained in the Client Events, including presentation of relevant information, the ability to choose a follow-up decision, and an interface for inviting collaboration participants. The GUI is presented through the demonstration of the prototype in section 7.2.

Figure 6-5 below shows a screenshot of a wizard-based interface representing some of the execution logic configured in the Validate results process step. As the figure shows, there are three specified outcomes related to this business process step; organize collaboration, parameter adjustment, and no actions.
6.3.3 Starting the K2 process

When a deviation is detected, a new instance of the K2 process should be initiated. This section will explain how the DeviationHandling process is initiated through custom C# code in the CODIO WP1 project.

6.3.3.1 Establishing connection information

In order to successfully establish a viable connection to the K2 process, we need to create a connection string holding the required information about the connection. Code snippet 6-13 presents the code used to create this connection string. As seen in the code snippet, we make use of a K2 API called SourceCode.Hosting.Client.BaseAPI, which is the easiest method to create a K2 connection string (K2 Underground, 2008). Required connection details are inserted as connection properties.
// Connection details (Note that the actual connection details used in the
// implementation differs from the ones presented here)
string _serverName = "MyServer";
string _user = "MyUsername";
string _domain = "MyDomain";
string _password = "MyPassword";
string _host = "MyHost";
string _processName = "DeviationHandling\DeviationHandling";

// Setup a connection using K2 API
connectionString = new

// Set connection properties
connectionString.Authenticate = true;
connectionString.Host = _host; // Local host.
connectionString.Integrated = true;
connectionString.IsPrimaryLogin = true;
connectionString.Port = 5252; // Port used for workflow interaction.
connectionString.UserID = _user;
connectionString.WindowsDomain = _domain;
connectionString.Password = _password;
connectionString.SecurityLabelName = "K2"; // Default label

Code snippet 6-13: Creating a connection to the K2 process.

6.3.3.2 Initiating the process

With our connection string in place, we are ready to start a new instance of the K2 process. First, we
create a connection object, by using K2’s SourceCode.Workflow.Client API. This particular connection
object is required when establishing workflow interaction (K2 Underground, 2008). The connection
object is used to create and start the K2 process instance, as shown in Code snippet 6-14.

// Initiating a K2 Connection object

// Open connection to K2 server
connection.Open(_serverName, connectionString.ToString());

// Create process instance
ProcessInstance
processInstance = connection.CreateProcessInstance(_processName);

// Start the process
connection.StartProcessInstance(processInstance, false);

Code snippet 6-14: Initiating an instance of the K2 process.
Evaluation
7 Experiments and results

This chapter will give a descriptive overview of how experiments included in the thesis are designed and conducted. The results from the experiments will be reviewed and thoroughly explained.

7.1 Experiment setup
This section will discuss the setup of the experiment, in respect to the chosen approach, definition of our success criteria, how the experiment and results are presented, and the general procedure of the experiment.

7.1.1 Approach
The ideal approach when experimenting on the prototype would naturally be to directly deploy the prototype functionality to an ongoing real-life drilling operation. That way we could perform reliable examination of the prototype’s behavior in its natural environment. Every detail in the performance of the prototype could be carefully analyzed in order to identify both weaknesses and strengths of the product.

Unfortunately, we have not been able to implement such an approach in this thesis, due to the following reasons:

- The prototype is designed to provide extended functionality to the work done in the CODIO WP1 research project, making it unable to operate properly as a standalone product. In other words, the functionality provided by the prototype has to be incorporated into the final product of CODIO in order to actually yield any useful benefits to the end user. The incorporation has not been of any priority during this work, since the effort would consume too much of the CODIO project’s limited availability of time and resources.
- In order to successfully conduct experiments within the environment of an ongoing drilling operation, the prototype would have to be deployed to this environment in combination with the CODIO WP1 product. This is not possible, as the product engineered in the CODIO project remains unfinished when starting with these experiments.
- The approach would require access to the various API’s used by the relevant systems included in the drilling operation. It would for instance be necessary to get access to connectivity methods provided by the MWD survey tool. Such resources have been unavailable during the work with this thesis.

Instead, we will conduct experiments inside our own simulated environment. This approach is easy to implement, and far less resource demanding. We are able to customize the simulation environment to address each deviation scenario individually, enabling us to perform isolated experiments that examine the behavior of the prototype in every unique situation.
7.1.2 Success criteria

The simulation environment will be used to conduct experiments on each of the six deviation scenarios we described in section 5.6. The goal of the experiment is to confirm the prototype’s performance in regards to producing the expected results from the processed experiment data.

7.1.3 Presentation of the experiment

The presentation of each experiment is divided into the following four components;

- **A summary of the complex rule criteria** – A brief summary will start each experiment, presenting and explaining the complex event rule representing the respective deviation scenario. Each rule criterion condition will have experiment values assigned.

- **Presentation of experiment data** – The following section contains a table listing a presentation of the experiment data sent as input to the prototype. Only data specifically relevant to the deviation scenario will be listed in this presentation. Each row in the presentation represents values obtained from a new survey reading, accompanied with the corresponding values from the plan entry. An overview of the complete experiment data actually sent as input to the prototype is provided as attachments in Appendix A.

- **Presentation of experiment results** – This section will present a table illustrating the experiment results produced by the prototype. Each row in the table represents the result of its corresponding entry in the experiment data. Each cell in the table signifies the isolated result from each processed event, compared to its corresponding criterion.

- **Explanation of experiment results** - Each experiment will be completed by an explanation of the experiment results. This explanation will focus on one data entry at the time, and discuss how the results are interpreted in context of discovering a deviation.

7.1.4 Experiment procedure

We will start the experiments by demonstrating the prototype with a concrete example. This demonstration will include a thorough and detailed explanation of the prototype’s behavior in context of a deviation in inclination.

Once the prototype has been demonstrated, we will start conducting experiments on the remaining deviation scenarios. The experiments will be described in an informative, though briefer manner as compared to the detailed descriptions provided in the demonstration.
7.2 Demonstration of the prototype

The demonstration is based on experiments conducted within the scenario of a deviation in the Inclination parameter (Scenario #4).

7.2.1 Presentation of rule criteria

The Complex Event rule shown in Table 7-1 is designed to identify a deviation in Inclination.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#13 - Change in Inclination deviation</td>
<td>&gt;= 0</td>
<td>First set starts</td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth</td>
<td>&lt; 500 ft</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#7 - Deviation in Inclination</td>
<td>&gt;= 3°</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>4</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth)</td>
<td>&gt; 0 ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>5</td>
<td>OR</td>
<td>(#15 - Current Measured Depth</td>
<td>&gt;= 500 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>6</td>
<td>AND</td>
<td>#7 - Deviation in Inclination</td>
<td>&gt;= 5°</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth)</td>
<td>&gt; 0 ft</td>
<td>Second set ends</td>
</tr>
<tr>
<td>8</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>9</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>10</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 7-1: Rule criteria for deviation in Inclination

The rule criteria are the same as were modeled in the scenario of a deviation in Inclination back in section 5.6.4, though now with proper experiment values added. The rule criteria can be explained to the following:

1. An increase in the Inclination deviation since the previous survey reading.

2 – 4. First condition set, representing a level 3 deviation by default:
   - Measured Depth is less than 500 ft.
   - There is a deviation in inclination of 3° or more.
   - The Total Vertical Depth parameter is higher / lower than what it should be according to the plan, in the opposite direction of the deviation in Inclination. This criterion conforms to the Analyze deviation outcome technique we discussed back in section 5.4.1.

5 – 7. Second condition set, representing a level 2 deviation by default:
   - Measured Depth has passed 500 ft.
   - There is a deviation in inclination of 5° or more.
   - Total Vertical Depth is deviation in the opposite direction.

8. Well progression is less than 80 %, indicating a level 2 deviation.

9. Well progression is more than 80 %, indicating a level 3 deviation.

10. The survey reading is reported as valid.

Notice that the first condition set will be fulfilled by having a deviation in inclination of 3°, while the second set allows for a deviation of 5°. This is based on the need to define alternate acceptance levels at different depths, as we discussed back in section 5.4.4.
Note that complete benchmarks are prepared in advance in real life drilling operations, stating the maximum tolerated deviation at all depths (Helge Rørvik, Halliburton, personal communication 30.03.10). Our experiments make use of imaginary test values, which may not be directly applicable to any actual well in the real world. However, even though the values often vary from well to well, the rule scenarios themselves will always be relevant to a certain extent.

7.2.2 Presentation of experiment data

Table 7-2 below gives a presentation of the experiment data used to test the scenario of a deviation in Inclination. As mentioned earlier, we will only present the experiment parameters having a direct relevance to the specific scenario. In this case, the parameters of interest include Measured Depth, Inclination, and Total Vertical Depth. The complete experiment data can be found in Appendix A.

<table>
<thead>
<tr>
<th>#</th>
<th>Measured Depth</th>
<th>Planned values from drilling plan Inclination</th>
<th>Actual values from survey reading Inclination</th>
<th>Total Vertical Depth</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>495.0</td>
<td>19.47</td>
<td>16.61</td>
<td>424.29</td>
</tr>
<tr>
<td>2</td>
<td>496.0</td>
<td>19.75</td>
<td>16.63</td>
<td>424.42</td>
</tr>
<tr>
<td>3</td>
<td>497.0</td>
<td>20.53</td>
<td>17.45</td>
<td>424.77</td>
</tr>
<tr>
<td>4</td>
<td>498.0</td>
<td>20.97</td>
<td>25.34</td>
<td>424.94</td>
</tr>
<tr>
<td>5</td>
<td>499.0</td>
<td>21.25</td>
<td>17.87</td>
<td>425.15</td>
</tr>
<tr>
<td>6</td>
<td>500.0</td>
<td>22.41</td>
<td>17.94</td>
<td>425.31</td>
</tr>
<tr>
<td>7</td>
<td>501.0</td>
<td>23.16</td>
<td>18.11</td>
<td>425.49</td>
</tr>
</tbody>
</table>

Table 7-2: Experiment input data - deviation in Inclination

Each row in the experiment data represents a new collection of data sent as input to the prototype. A unique sequential numeric value is assigned to each row, making it possible to distinguish between the data entries.

7.2.3 Presentation of experiment results

This section will investigate the results from processing the experiment data. Table 7-3 presents the combined results gathered from the prototype. As explained in section 7.1.3, each cell in the table represents the value of the processed event. The processed events are associated with different fonts, signifying whether the corresponding rule criterion for that event was met or not. Events presented with a bold and italic font means that the condition in the criterion was met. If all rule criteria are fulfilled, we know a deviation has occurred.
### Rule criteria

<table>
<thead>
<tr>
<th>#</th>
<th>Current Measured Depth</th>
<th>Deviation in Inclination</th>
<th>Change in deviation</th>
<th>Deviation in TVD in the opposite direction?</th>
<th>Current well progression</th>
<th>Valid survey reading</th>
<th>Deviation status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>495.0 ft</td>
<td>- 2.86 °</td>
<td>0 °</td>
<td>No</td>
<td>79 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>2</td>
<td>496.0 ft</td>
<td>- 3.12 °</td>
<td>+ 0.26 °</td>
<td>No</td>
<td>79 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>3</td>
<td>497.0 ft</td>
<td>- 3.08 °</td>
<td>- 0.04 °</td>
<td>No</td>
<td>79 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>4</td>
<td>498.0 ft</td>
<td>4.37 °</td>
<td>- 7.45 °</td>
<td>No</td>
<td>79 %</td>
<td>False</td>
<td>No deviation</td>
</tr>
<tr>
<td>5</td>
<td>499.0 ft</td>
<td>- 3.38 °</td>
<td>+ 0.30 °</td>
<td>Yes</td>
<td>79 %</td>
<td>True</td>
<td>Level 3 deviation found</td>
</tr>
<tr>
<td>6</td>
<td>500.0 ft</td>
<td>- 4.47 °</td>
<td>+ 1.09 °</td>
<td>Yes</td>
<td>79 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>7</td>
<td>501.0 ft</td>
<td>- 5.05 °</td>
<td>+ 0.58 °</td>
<td>Yes</td>
<td>79 %</td>
<td>True</td>
<td>Level 2 deviation found</td>
</tr>
</tbody>
</table>

Table 7-3: Experiment results - deviation in Inclination

### 7.2.4 Explanation of results

This section will give a detailed explanation of each processed experiment input, by investigating the results one row at the time. The results can be explained to the following:

1. As we can see from Table 7-3, the experiment data started with a Measured Depth of 495.0 feet. According to the corresponding rule criterion, this means we should search for deviations within the first condition set. We can see that most of the rule criteria failed to meet the condition set, signified by the normal font in the results.

2. This time, the event representing the actual deviation in Inclination has exceeded its maximum value of 3 °, fulfilling its criterion. However, the Total Vertical Depth parameter is not deviating in the opposite direction of the inclination deviation. This effectively means that the well is currently directed too much vertically, signified by the negative deviation in the Inclination parameter, to compensate for the fact that the well has been drilled too much horizontally in total, represented by a negative deviation in the TVD parameter. Further drilling at this increased vertical direction will in a short amount of time decrease the deviation in the TVD parameter, eventually eliminating it, whereas the Inclination parameter could be adjusted to conform to its original direction described in the drilling plan.

3. This result reveals that the deviation has decreased since the previous reading. Even though the Inclination parameter still contains a deviation exceeding its limit according to the condition of the rule criterion, a decrease in deviation indicates that the deviation is becoming less severe, and could ultimately be eliminated. Also, the Total Vertical Depth parameter is still deviating in the same direction, further preventing the detection of a deviation.

4. The inclination parameter in this survey reading is reported with a decrease of 7.45°. This value stands out as clearly flawed, when compared to the previous survey readings included in the experiment data. The Event Window technique we implemented in the prototype detects this unlikely increase in inclination, and flags the survey reading as invalid. Thus, the current entry of experiment data is ignored and omitted from further calculation.
5. This result reveals that all criteria for a level 3 deviation have been fulfilled. The deviation has exceeded its maximum allowed value, and has increased since the previous survey reading. Also, the event representing a deviation in the TVD parameter is fulfilling its respective rule criterion, as the TVD parameter now deviates in the opposite direction of the Inclination deviation. Additionally, the survey reading is no longer flagged as invalid, caused by the fact that the current value in the Inclination parameter closely resembles the Inclination value in all the previous survey readings. A new instance of the deviation handling process in K2 is initiated, and relevant information about the deviation is compiled. The prototype continues to search for deviations, as further deviations might occur at a later point during the drilling operation.

6. The Measured Depth has reached 500 ft, meaning that we should start using the second condition set within our search pattern. The second condition set allows for a deviation in inclination by 5°. The current deviation is reported to be -4.47°, thus not fulfilling this set of conditions.

7. Yet again, all rule criteria are fulfilled. This time the severity of the deviation is classified as level 2, caused by the well surpassing 500 ft of measured depth, in addition to the current well progress not yet reaching 80% of the total distance. Another instance of our K2 process is initiated.

As we observed from the experiment results, the inclination parameter contained a continuous and fairly stable deviation varying between -2.86° and -5.05° throughout the whole experiment. Figure 7-1 below illustrates this change of deviation in more detail. Remember, the value in the inclination parameter at a Measured Depth of 498.0 ft was considered erroneous because of its low resemblance to the other values.

---

![Figure 7-1: Changes in Inclination deviation, according to experiment results.](image-url)
We learned from inspecting the experiment results that the first four entries in the experiment failed to meet all rule criteria, even though the deviation at several occasions exceeded its maximum allowed value of 3°. The reason we gave was that the TVD parameter was deviating in the same direction as the Inclination parameter, while the rule criterion requires a deviation in the opposing direction. Let’s take a look at Figure 7-2, to get a deeper understanding of how the deviation in inclination affected the deviation in TVD.

![Figure 7-2: Changes in TVD parameter, according to experiment results](image)

As we can see from the figure, the deviation in the TVD parameter is gradually decreasing throughout the first four input entries in the experiment data, continuously achieving a more correct value. However, the remaining entries reveals that the TVD parameter is starting to deviate in the opposite direction, all while still having a constant deviation in the Inclination parameter. This should be avoided, as the well at this point starts to constantly drill too far vertically, resulting in a negative outcome.

### 7.2.5 Starting the K2 process

Two deviations were found from processing the experimenting data; one level 2 and one level 3. Both of these findings created a unique process instance of the K2 process named “Handle deviation in directional drilling”. The directional driller gets notified of the situation through his K2 worklist.
7.2.6 Validating the first deviation

When visiting the K2 worklist, the directional driller finds that two task items has been assigned to him. When clicking on the first task, the K2 process moves to the process step called “Validate results”. A website providing a detailed description of the level 3 deviation appears, accompanied with relevant information regarding the status of the well situation, as illustrated in Figure 7-3 below. In order to be able to inspect the cause of the deviation in more detail, the directional driller is provided with buttons to open the files containing the drilling plan and survey report.

![Validate deviation results]

Validate deviation results

The following information were collected on the deviation:

A level 3 deviation was found.

Current general information about the situation:
- Measured Depth: 499.0 ft.
- Well Progression: 79 %.

Information specific to the deviation:
- Deviation in inclination: -3.38°.
- Increase in deviation: +0.3°.
- Deviation in TVD: +0.04 ft.

Suggested follow-up:
- Organize collaboration.

Select follow-up:
- View Drilling Plan
- View Survey Report
- Adjust drilling parameters
- Adjust drilling parameters
- Organize collaboration
- No action
- OK

Figure 7-3: Screenshot - Validating the Level 3 deviation

After validating that a real deviation actually has occurred, the directional driller may choose the appropriate follow-up option from a drop-down list available on the webpage. In this scenario, the option “Organize collaboration” is chosen, as recommended by the prototype. The “OK” button is clicked, and the process moves to the next process step; “Organize collaboration”. Another webpage appears, giving the directional driller the opportunity to invite participants to the collaboration session (See Figure 7-4).
Organize Collaboration

Invite collaboration participants:

- Drilling Supervisor
- Drilling Engineer
- Directional Driller
- Geologist

OK

Figure 7-4: Screenshot - Inviting participants to collaboration session

The appropriate participants are chosen, making a task item appear inside the personal K2 worklist of all participants involved.

Note that the user interface presented here differs from the actual interface developed in the CODIO WP1 product. In CODIO, the completion of business process steps will take place within a custom-made well-site portal, providing a range of additional functionality that is not included in this demonstration. The simple website interface implemented in our prototype should only be considered an attempt to illustrate how event-driven behavior eventually could be incorporated in the final CODIO product.
7.2.7 Validating the second deviation
Clicking on the second task in the K2 worklist will bring up information of the reported level 2 deviation, as illustrated in Figure 7-5.

![Validate deviation results]

The following information were collected on the deviation:
A level 2 deviation was found.
Current general information about the situation:
Measured Depth: 501.0 ft.
Well Progression: 79 %.

Information specific to the deviation:
Deviation in inclination: - 5.05°.
Increase in deviation: + 0.50°.
Deviation in TVD: + 0.47 ft.

Suggested follow-up:
Adjust drilling parameters.

![Select follow-up]

Figure 7-5: Screenshot - Validating the Level 2 deviation
Clicking on the “OK” button ends the process step. An action is triggered to automatically forward the information to the role responsible for adjusting the drilling parameters.
7.3 Experiments
In the demonstration of the prototype we conducted an experiment based on the scenario of a deviation in the Inclination parameter. This section will be oriented on experimenting on the remaining deviation scenarios. The experiments will be explained in a brief nature, with relevant information presented in a concise and structured order.

7.3.1 Scenario #1 - Northings and Eastings

7.3.1.1 Presentation of rule criteria
The Complex Event rule shown in Table 7-4 is designed to identify a deviation in the Northings / Eastings pair.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#9 - Change in Northing/Easting deviation</td>
<td>&gt;= 0</td>
<td>First set starts</td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth</td>
<td>&lt; 500 ft</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#1 – Deviation in Northings</td>
<td>&gt;= 6 ft</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>OR</td>
<td>#2 – Deviation in Eastings</td>
<td>&gt;= 6 ft</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>OR</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&gt;= 9 ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>6</td>
<td>OR</td>
<td>(#15 - Current Measured Depth</td>
<td>&gt;= 500 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#1 – Deviation in Northings</td>
<td>&gt;= 11 ft</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>OR</td>
<td>#2 – Deviation in Eastings</td>
<td>&gt;= 11 ft</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>OR</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&gt;= 15 ft</td>
<td>Second set ends</td>
</tr>
<tr>
<td>10</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>11</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>12</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 7-4: Rule criteria designed to detect a deviation in the Northings / Eastings pair

The rule criteria can be explained to the following:

1. An increase in the deviation in the Northings / Eastings pair since the previous survey reading.

2 – 5. First condition set:
- Measured Depth is less than 500 ft.
- There is a deviation in Northings of 6 ft or more.
- Or there is a deviation in Eastings of 6 ft or more.
- Or there is a deviation in the combined Northings / Eastings pair of 9 ft or more.

6 – 9. Second condition set:
- Measured Depth has surpassed 500 ft.
- There is a deviation in Northings of 11 ft or more.
- Or there is a deviation in Eastings of 11 ft or more.
- Or there is a deviation in the combined Northings / Eastings pair of 15 ft or more.

10. Well progression is less than 80 %, indicating a level 2 deviation.
11. Well progression is more than 80 %, indicating a level 3 deviation.
12. The survey reading is reported as valid.
7.3.1.2 Presentation of experiment data

Table 7-5 presents the experiment data.

<table>
<thead>
<tr>
<th>#</th>
<th>Measured Depth</th>
<th>Northings</th>
<th>Eastings</th>
<th>Northings</th>
<th>Eastings</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>396.0</td>
<td>40.52</td>
<td>24.49</td>
<td>40.52</td>
<td>24.49</td>
</tr>
<tr>
<td>2</td>
<td>437.0</td>
<td>40.52</td>
<td>24.49</td>
<td>43.52</td>
<td>26.49</td>
</tr>
<tr>
<td>3</td>
<td>468.0</td>
<td>40.52</td>
<td>24.49</td>
<td>45.52</td>
<td>25.49</td>
</tr>
<tr>
<td>4</td>
<td>499.0</td>
<td>40.52</td>
<td>24.49</td>
<td>48.52</td>
<td>24.49</td>
</tr>
<tr>
<td>5</td>
<td>530.0</td>
<td>40.52</td>
<td>24.49</td>
<td>48.52</td>
<td>28.49</td>
</tr>
<tr>
<td>6</td>
<td>561.0</td>
<td>40.52</td>
<td>24.49</td>
<td>48.52</td>
<td>31.49</td>
</tr>
</tbody>
</table>

Table 7-5: Experiment input data - deviation in the Northings / Eastings pair

7.3.1.3 Presentation of experiment results

Table 7-6 presents the results produced by the prototype.

<table>
<thead>
<tr>
<th>#</th>
<th>Current MD</th>
<th>Deviation in Eastings</th>
<th>Deviation in Northings</th>
<th>Total deviation in Northings / Eastings pair</th>
<th>Change in total deviation</th>
<th>Current well progress</th>
<th>Valid survey reading</th>
<th>Deviation status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>396.0 ft</td>
<td>0 ft</td>
<td>0 ft</td>
<td>0 ft</td>
<td>0 ft</td>
<td>58 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>2</td>
<td>437.0 ft</td>
<td>3 ft</td>
<td>2 ft</td>
<td>5 ft</td>
<td>+ 5 ft</td>
<td>64 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>3</td>
<td>468.0 ft</td>
<td>5 ft</td>
<td>1 ft</td>
<td>6 ft</td>
<td>+ 1 ft</td>
<td>69 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>4</td>
<td>499.0 ft</td>
<td>8 ft</td>
<td>0 ft</td>
<td>8 ft</td>
<td>+ 2 ft</td>
<td>73 %</td>
<td>True</td>
<td>Level 2 deviation found</td>
</tr>
<tr>
<td>5</td>
<td>530.0 ft</td>
<td>8 ft</td>
<td>4 ft</td>
<td>12 ft</td>
<td>+ 3 ft</td>
<td>76 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>6</td>
<td>561.0 ft</td>
<td>8 ft</td>
<td>7 ft</td>
<td>15 ft</td>
<td>+ 4 ft</td>
<td>80 %</td>
<td>True</td>
<td>Level 3 deviation found</td>
</tr>
</tbody>
</table>

Table 7-6: Experiment results - deviation in the Northings / Eastings pair

7.3.1.4 Explanation of results

1. Measured Depth is below 500 ft; first set of conditions is activated. No deviations are detected.
2. There are deviations, but none are significant enough to fulfill the rule criteria.
3. The deviations are still too small to be detected.
4. Deviation in the Northings has increased to 8 ft, exceeding its maximum allowed value. A level 2 deviation is detected.
5. Measured Depth has passed 500 ft; second set of conditions is activated. The Northings parameter still has a deviation of 8 ft, but the new condition set require a deviation of 11 ft. No deviations are found.
6. A level 3 deviation has been found in the combined Northings / Eastings pair.
7.3.2 Scenario #2 - Azimuth

7.3.2.1 Presentation of rule criteria

The complex event rule shown in Table 7-7 is designed to identify a deviation in the Azimuth.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#14 - Change in Azimuth deviation</td>
<td>&gt;= 0</td>
<td>First set starts</td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth</td>
<td>&lt; 500 ft</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#8 - Deviation in Azimuth</td>
<td>&gt;= 3° ft</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>4</td>
<td>AND</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&lt; 3 ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>5</td>
<td>OR</td>
<td>(#15 - Current Measured Depth</td>
<td>&gt;= 500 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>6</td>
<td>AND</td>
<td>#8 - Deviation in Azimuth</td>
<td>&gt;= 5° ft</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&lt; 3 ft</td>
<td>Second set ends</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>9</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>10</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 7-7: Rule criteria designed to detect a deviation in Azimuth

The rule criteria can be explained to the following:

1. An increase in the deviation in Azimuth since the previous survey reading.

2 – 4. First condition set, representing a level 3 deviation by default:
   • Measured Depth has passed 500 ft.
   • There is a deviation in Azimuth of 3° or more.
   • A deviation of less than 3 ft is present in the Northings / Eastings pair.

5 – 7. Second condition set, representing a level 2 deviation by default:
   • Measured Depth has passed 500 ft.
   • There is a deviation in Azimuth of 5° or more.
   • A deviation of less than 3 ft is present in the Northings / Eastings pair.

8. Well progression is less than 80 %, indicating a level 2 deviation.

9. Well progression is more than 80 %, indicating a level 3 deviation.

10. The survey reading is reported as valid.

7.3.2.2 Presentation of experiment data

Table 7-8 presents the experiment data.

<table>
<thead>
<tr>
<th>#</th>
<th>Measured Depth</th>
<th>Planned values from drilling plan</th>
<th>Actual values from survey reading</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Azimuth</td>
<td>Northings</td>
<td>Eastings</td>
</tr>
<tr>
<td>1</td>
<td>479.0</td>
<td>358.27</td>
<td>40.52</td>
</tr>
<tr>
<td>2</td>
<td>500.0</td>
<td>358.27</td>
<td>40.52</td>
</tr>
<tr>
<td>3</td>
<td>521.0</td>
<td>358.27</td>
<td>40.75</td>
</tr>
</tbody>
</table>

Table 7-8: Experiment input data - deviation in Azimuth
7.3.2.3  Presentation of experiment results

Table 7-9 presents the results produced by the prototype.

<table>
<thead>
<tr>
<th>#</th>
<th>Current Measured Depth</th>
<th>Deviation in Azimuth</th>
<th>Change in deviation</th>
<th>Deviation in the Northings/Eastings pair</th>
<th>Current well progression</th>
<th>Valid survey reading</th>
<th>Deviation status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>479.0 ft</td>
<td>4.30 °</td>
<td>0 °</td>
<td>3.22 ft</td>
<td>75 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>2</td>
<td>500.0 ft</td>
<td>5.26 °</td>
<td>+0.96 °</td>
<td>2.8 ft</td>
<td>77 %</td>
<td>True</td>
<td>Level 2 deviation found</td>
</tr>
<tr>
<td>3</td>
<td>521.0 ft</td>
<td>5.69 °</td>
<td>+0.43 °</td>
<td>4.1 ft</td>
<td>80 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
</tbody>
</table>

Table 7-9: Experiment results - deviation in Azimuth

7.3.2.4  Explanation of results

1. Measured Depth is below 500 ft; first set of conditions is activated. Azimuth deviation is 4.3°, exceeding its maximum allowed value of 3°. However, there is a deviation of more than 3ft present in the Northings/Eastings pair, preventing any deviation to be detected. See section 5.6.2 for a detailed explanation of this criterion.

2. Measured Depth has passed 500 ft; second set of conditions is activated. The new condition allows for a deviation of 5°, although this is exceeded. Also, the deviation in the Northings / Eastings pair is less than 3ft. All complex rule criteria are fulfilled; a level 2 deviation is discovered.

3. The Azimuth deviation is still fulfilling its criterion. However, the Northings / Eastings pair is deviating more than 3ft. No deviation detected.

7.3.3  Scenario #3 - Dog Leg Section

7.3.3.1  Presentation of rule criteria

The Complex Event rule shown in Table 7-10 is designed to identify a deviation in Dog Leg Section.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#12 - Change in DLS deviation</td>
<td>&gt;= 0</td>
<td>First set starts</td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth</td>
<td>&lt; 500 ft</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#6 - Deviation in Dog Leg Section</td>
<td>&gt;= 4 °</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt; 0 ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>5</td>
<td>OR</td>
<td>(#15 - Current Measured Depth</td>
<td>&gt;= 500 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>6</td>
<td>AND</td>
<td>#6 - Deviation in Dog Leg Section</td>
<td>&gt;= 7 °</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt; 0 ft</td>
<td>Second set ends</td>
</tr>
<tr>
<td>8</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>9</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>10</td>
<td>AND</td>
<td>#17 - Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 7-10: Rule criteria designed to detect a deviation in Dog Leg Section
The rule criteria can be explained to the following:

1. An increase in the deviation in DLS since the previous survey reading.

2 – 4. First condition set:
   - Measured Depth has passed 500 ft.
   - There is a deviation in DLS of 4° or more.
   - Total Vertical Depth is deviating in the opposite direction.

5 – 7. Second condition set:
   - Measured Depth has passed 500 ft.
   - There is a deviation in inclination of 7° or more.
   - Total Vertical Depth is deviating in the opposite direction.

8. Well progression is less than 80 %, indicating a level 2 deviation.
9. Well progression is more than 80 %, indicating a level 3 deviation.
10. The survey reading is reported as valid.

### 7.3.3.2 Presentation of experiment data

Table 7-11 presents the experiment data.

<table>
<thead>
<tr>
<th>#</th>
<th>Measured Depth</th>
<th>Dog Leg Section</th>
<th>Total Vertical Depth</th>
<th>Dog Leg Section</th>
<th>Total Vertical Depth</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>477.0 ft</td>
<td>0.34</td>
<td>404.49</td>
<td>4.44</td>
<td>404.59</td>
</tr>
<tr>
<td>2</td>
<td>488.0 ft</td>
<td>0.34</td>
<td>413.49</td>
<td>4.74</td>
<td>413.49</td>
</tr>
<tr>
<td>3</td>
<td>499.0 ft</td>
<td>0.34</td>
<td>421.49</td>
<td>6.04</td>
<td>421.39</td>
</tr>
<tr>
<td>4</td>
<td>510.0 ft</td>
<td>0.34</td>
<td>435.49</td>
<td>7.24</td>
<td>435.29</td>
</tr>
<tr>
<td>5</td>
<td>521.0 ft</td>
<td>0.34</td>
<td>446.49</td>
<td>7.44</td>
<td>446.19</td>
</tr>
</tbody>
</table>

Table 7-11: Experiment input data - deviation in Dog Leg Section

### 7.3.3.3 Presentation of experiment results

Table 7-12 presents the results produced by the prototype.

<table>
<thead>
<tr>
<th>#</th>
<th>Current Measured Depth</th>
<th>Deviation in Dog Leg Section</th>
<th>Change in deviation</th>
<th>Deviation in TVD in the opposite direction?</th>
<th>Current well progression</th>
<th>Valid survey reading</th>
<th>Deviation status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>477.0 ft</td>
<td>4.1 °</td>
<td>0 °</td>
<td>No</td>
<td>72 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>2</td>
<td>488.0 ft</td>
<td>4.4 °</td>
<td>+ 0.3 °</td>
<td>No</td>
<td>74 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>3</td>
<td>499.0 ft</td>
<td>5.7 °</td>
<td>+ 1.3 °</td>
<td>Yes</td>
<td>76 %</td>
<td>True</td>
<td>Level 2 deviation found</td>
</tr>
<tr>
<td>4</td>
<td>510.0 ft</td>
<td>6.9 °</td>
<td>+ 1.2 °</td>
<td>Yes</td>
<td>78 %</td>
<td>True</td>
<td>Level 3 deviation found</td>
</tr>
<tr>
<td>5</td>
<td>521.0 ft</td>
<td>7.1 °</td>
<td>+ 0.2 °</td>
<td>Yes</td>
<td>80 %</td>
<td>True</td>
<td>Level 3 deviation found</td>
</tr>
</tbody>
</table>

Table 7-12: Experiment results - deviation in Dog Leg Section
7.3.3.4 Explanation of results

1. Measured Depth is below 500 ft; first set of conditions is activated. The deviation in DLS is significant enough to be detected, but there has not been reported any increase in deviation.

2. Deviation in DLS has increased to 4.4°. However, the TVD parameter is not deviating in the opposite direction.

3. Deviation in DLS is 5.7°, fulfilling its criterion. Also, TVD is deviating in the opposite direction. A level 2 deviation is found.

4. Measured Depth has passed 500 ft; second set of conditions is activated. The deviation in DLS does not conform to the new conditions.

5. All rule criteria are fulfilled; a level 3 deviation is detected.

7.3.4 Scenario #5 - Total Vertical Depth

7.3.4.1 Presentation of rule criteria

The Complex Event rule shown in Table 7-13 is designed to identify a deviation in Total Vertical Depth.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#10 - Change in TVD deviation</td>
<td>&gt;= 0</td>
<td>First set starts</td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth</td>
<td>&lt; 500 ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt;= 10 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>4</td>
<td>OR</td>
<td>(#15 - Current Measured Depth</td>
<td>&gt;= 500 ft</td>
<td>Second set ends</td>
</tr>
<tr>
<td>5</td>
<td>AND</td>
<td>#4 - Deviation in Total Vertical Depth</td>
<td>&gt;= 15 ft</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>7</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>8</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 7-13: Complex event rule designed to detect a deviation in Total Vertical Depth

According to the table, the rule criteria can be explained to the following:

1. An increase in the deviation in TVD since the previous survey reading.

2 – 3. First condition set:
   - Measured Depth is less than 500 ft.
   - There is a deviation in TVD of 10 ft or more.

4 – 5. Second condition set:
   - Measured Depth has passed 500 ft.
   - There is a deviation in TVD of 15 ft or more.

6. Well progression is less than 80 %, indicating a level 2 deviation.

7. Well progression is more than 80 %, indicating a level 3 deviation.

8. The survey reading is reported as valid.
7.3.4.2 Presentation of experiment data

Table 7-14 presents the experiment data.

<table>
<thead>
<tr>
<th>#</th>
<th>Measured Depth</th>
<th>Total Vertical Depth</th>
<th>Actual values from survey reading</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>457.0</td>
<td>365.0</td>
<td>374.5</td>
</tr>
<tr>
<td>2</td>
<td>478.0</td>
<td>381.0</td>
<td>391.5</td>
</tr>
<tr>
<td>3</td>
<td>499.0</td>
<td>402.0</td>
<td>412.2</td>
</tr>
<tr>
<td>4</td>
<td>520.0</td>
<td>418.0</td>
<td>430.8</td>
</tr>
<tr>
<td>5</td>
<td>541.0</td>
<td>430.0</td>
<td>445.1</td>
</tr>
</tbody>
</table>

Table 7-14: Experiment input data - deviation in Total Vertical Depth

7.3.4.3 Presentation of experiment results

Table 7-15 presents the results produced by the prototype.

<table>
<thead>
<tr>
<th>#</th>
<th>Current Measured Depth</th>
<th>Deviation in Total Vertical Depth</th>
<th>Change in deviation since the previous survey reading</th>
<th>Current well progression</th>
<th>Valid survey reading</th>
<th>Deviation status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>457.0 ft</td>
<td>9.5 ft</td>
<td>0 ft</td>
<td>68 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>2</td>
<td>478.0 ft</td>
<td>10.5 ft</td>
<td>+ 1 ft</td>
<td>71 %</td>
<td>True</td>
<td>Level 2 deviation found</td>
</tr>
<tr>
<td>3</td>
<td>499.0 ft</td>
<td>10.2 ft</td>
<td>- 0.3 ft</td>
<td>75 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>4</td>
<td>520.0 ft</td>
<td>12.8 ft</td>
<td>+ 2.6 ft</td>
<td>78 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>5</td>
<td>541.0 ft</td>
<td>15.1 ft</td>
<td>+ 2.3 ft</td>
<td>80 %</td>
<td>True</td>
<td>Level 3 deviation found</td>
</tr>
</tbody>
</table>

Table 7-15: Experiment results - deviation in Total Vertical Depth

7.3.4.4 Explanation of results

1. Measured Depth is below 500 ft; first set of conditions is activated. The deviation in TVD is not significant enough to be detected.
2. Deviation in TVD has increased to 10.5 ft, exceeding its maximum allowed value. A level 2 deviation is detected.
3. There has been a decrease in deviation since the previous survey reading, preventing the detection of a deviation.
4. Measured Depth has passed 500 ft; second set of conditions is activated. The deviation in TVD does not conform to the conditions.
5. All criteria has been fulfilled, a level 3 deviation has been detected.
7.3.5 Scenario #6 - Vertical Section

7.3.5.1 Presentation of rule criteria

The Complex Event rule shown in Table 7-16 is designed to identify a deviation in Vertical Section.

<table>
<thead>
<tr>
<th>#</th>
<th>Operator</th>
<th>Event parameter</th>
<th>Condition</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IF</td>
<td>#11 - Change in VSec deviation</td>
<td>&gt;= 0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>AND</td>
<td>(#15 - Current Measured Depth)</td>
<td>&lt; 500 ft</td>
<td>First set starts</td>
</tr>
<tr>
<td>3</td>
<td>AND</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&gt;= 3 ft</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>AND</td>
<td>#4 – Deviation in Total Vertical Depth</td>
<td>&gt;= 3 ft</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>AND</td>
<td>#5 - Deviation in Vertical Section</td>
<td>&lt;= -9 ft</td>
<td>First set ends</td>
</tr>
<tr>
<td>6</td>
<td>OR</td>
<td>(#15 - Current Measured Depth)</td>
<td>&gt;= 500 ft</td>
<td>Second set starts</td>
</tr>
<tr>
<td>7</td>
<td>AND</td>
<td>#3 - Total deviation in the Northing/Easting pair</td>
<td>&gt;= 5 ft</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>AND</td>
<td>#4 – Deviation in Total Vertical Depth</td>
<td>&gt;= 5 ft</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>AND</td>
<td>#5 - Deviation in Vertical Section</td>
<td>&lt;= -15 ft</td>
<td>Second set ends</td>
</tr>
<tr>
<td>10</td>
<td>AND</td>
<td>#16 - Current well progression given in %</td>
<td>&lt; 80 %</td>
<td>Level 2 deviation</td>
</tr>
<tr>
<td>11</td>
<td>OR</td>
<td>#16 - Current well progression given in %</td>
<td>&gt;= 80 %</td>
<td>Level 3 deviation</td>
</tr>
<tr>
<td>12</td>
<td>AND</td>
<td>#17 – Validity of survey reading</td>
<td>= True</td>
<td></td>
</tr>
</tbody>
</table>

Table 7-16: Rule criteria designed to detect a deviation in Vertical Section

The rule criteria can be explained to the following:

1. An increase in the deviation in VSec since the previous survey reading.

2 – 5. First condition set:
   - Measured Depth has passed 500 ft.
   - There is a deviation in the Northings / Eastings pair of 3 ft or more.
   - There is a deviation in TVD of 3 ft or more.
   - There is a negative deviation in VSec of -9 ft or more.

6 – 9. Second condition set:
   - Measured Depth has passed 500 ft.
   - There is a deviation in the Northings / Eastings pair of 5 ft or more.
   - There is a deviation in TVD of 5 ft or more.
   - There is a negative deviation in VSec of -15 ft or more.

10. Well progression is less than 80 %, indicating a level 2 deviation.

11. Well progression is more than 80 %, indicating a level 3 deviation.

12. The survey reading is reported as valid.

7.3.5.2 Presentation of experiment data

Table 7-17 presents the experiment data.

| #  | MD  | VSec | Northing | Easting | TVD  | VSec | Northing | Easting | TVD  |
|----|-----|------|----------|---------|------|------|----------|---------|------|------|
| 1  | 397.0 | 16.76 | 30.52    | 29.49   | 444.70 | 8.96 | 29.80    | 29.49   | 452.70 |
| 2  | 448.0 | 17.76 | 40.52    | 34.49   | 465.70 | 7.97 | 38.78    | 33.49   | 474.20 |
| 3  | 499.0 | 18.76 | 50.52    | 39.49   | 486.70 | 7.16 | 48.36    | 38.34   | 494.90 |
| 4  | 550.0 | 19.76 | 60.52    | 44.49   | 507.70 | 6.46 | 57.87    | 42.84   | 516.0 |
| 5  | 601.0 | 20.76 | 70.52    | 49.49   | 528.70 | 5.56 | 66.89    | 47.32   | 536.1 |

Table 7-17: Experiment input data - deviation in Vertical Section
7.3.5.3 Presentation of experiment results

Table 7-18 presents the results produced by the prototype.

<table>
<thead>
<tr>
<th>#</th>
<th>Current MD</th>
<th>Deviation in VSec</th>
<th>Change in deviation</th>
<th>Deviation in Northings/Eastings pair</th>
<th>Deviation in TVD</th>
<th>Current well progression</th>
<th>Valid survey reading</th>
<th>Deviation status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>497.0 ft</td>
<td>-7.8 ft</td>
<td>0 ft</td>
<td>-0.72 ft</td>
<td>8 ft</td>
<td>68 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>2</td>
<td>498.0 ft</td>
<td>-9.79 ft</td>
<td>+1.99 ft</td>
<td>-2.74 ft</td>
<td>8.5 ft</td>
<td>71 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>3</td>
<td>499.0 ft</td>
<td>-11.6 ft</td>
<td>+1.81 ft</td>
<td>-3.31 ft</td>
<td>8.2 ft</td>
<td>74 %</td>
<td>True</td>
<td>Level 2 deviation found</td>
</tr>
<tr>
<td>4</td>
<td>500.0 ft</td>
<td>-13.3 ft</td>
<td>+1.7 ft</td>
<td>-4.3 ft</td>
<td>8.3 ft</td>
<td>77 %</td>
<td>True</td>
<td>No deviation</td>
</tr>
<tr>
<td>5</td>
<td>501.0 ft</td>
<td>-15.2 ft</td>
<td>+1.9 ft</td>
<td>-5.8 ft</td>
<td>7.4 ft</td>
<td>80 %</td>
<td>True</td>
<td>Level 3 deviation found</td>
</tr>
</tbody>
</table>

Table 7-18: Experiment results - deviation in Vertical Section

7.3.5.4 Explanation of results

1. Measured Depth is below 500 ft; first set of conditions is activated. The deviation in VSec is not significant enough to be detected, and there has not been reported any increase in deviation. Also, the deviation in the Northings/Easting pair is not fulfilling the rule criterion. No deviation detected.

2. Deviation in VSec has increased to 9.2 ft, meeting its criterion. However, the deviation in the Northings/Easting pair is still not significant enough.

3. All rule criteria are met; a level 2 deviation is detected.

4. Measured Depth has passed 500 ft; second set of conditions is activated. The deviations in the VSec parameter and Northings/Easting pair do not conform to the new conditions.

5. There has been an increase in deviation, making all criteria fulfilled. A level 3 deviation is detected.
7.4 Discussion of results
This section will summarize and discuss the research results.

7.4.1 Summary of experiment results
The results from the experiments can be summarized into the following:

- The demonstration and experiment results prove the CEP Engine successful in respect to deliver consistent and expected output from experiment data, making the results compliant to the experiment success criteria.
- The actual behavior of the prototype during the entire experiment correctly matches what was expected. Examples of this include:
  - Processing of events based on values retrieved from the drilling plan and survey report. (Demonstrated in all deviation scenarios).
  - Flagging of invalid survey entries according to the Event Windows technique described in section 5.4.3. (Demonstrated in all deviation scenarios).
  - Functional application of the Analyze deviation status technique described in section 5.4.1 (Demonstrated in Table 7-3 and Table 7-12).
  - Calculation and application of the well progression technique when suggesting a proper follow-up. (Demonstrated in all deviation scenarios).
  - Providing availability of information regarding an occurred deviation, realized through ASP.NET websites. (Demonstrated in Figure 7-5 and Figure 7-3).
- Basic principles of CEP have successfully been combined with BPM, utilizing event-driven behavior. Communication between the CEP Engine and the BPMN model has been enabled through K2 API’s.

7.4.2 Reliability of the experiment results
We acknowledge that the results obtained from conducting experiments perfectly match the expected outcome. However, we only know for sure that the prototype performs as intended on exactly these experiments. Further testing may very well prove the prototype to behave in an unexpected way when facing other challenges, especially when applied to its natural environment. We will discuss a couple of known factors that may compromise the validity of the prototype’s current capabilities:

- We have not been able to experiment on the prototype in its natural environment. Thus, we have not had the opportunity to examine the performance of the prototype’s functionality in a realistic setting, to analyze its true strengths and weaknesses inside the intended application area.
- We have been unable to test the prototype using a combination of a drilling plan and survey report containing real-life deviations. Our experiments have been realized using custom made drilling plans and survey reports, based on fictional, though informed assumptions of real-life deviations, each tailored to suit our simulation environment in order to test every aspect of our deviation scenarios. The experiment input is designed with rapidly changing survey parameters, in order to efficiently test each deviation scenario. This will however not be the case in real-life drilling operations, as deviations are often built over a longer period of time, and with parameters correlating naturally according to each other.
We have not had the opportunity to test the prototype as a part of the full product created in the CODIO project. Therefore we have not been able to examine whether the prototype interacts properly with the rest of the functionality that is supposed to be embedded in the final CODIO product.

There could be some deviation scenarios that are not specified in any of the complex event rules. If any such scenario should exist, the prototype would be unable to detect its eventual occurrence. However, experts on deviation handling from Halliburton has contributed with valuable feedback in the process of designing the complex event rules, strengthening the credibility of the prototype being able to discover all possible deviation variations.

These validity threats should be considered and elaborated in further development of the prototype.

Note that the intention of the experiments conducted in this thesis were to demonstrate that the prototype acts as expected inside our deterministic simulated environment, and not to state that the current prototype functionality applies directly to the real world.

7.4.3 Benefits from introducing the prototype

When incorporated into the product developed by the CODIO team, a final prototype will provide solid capabilities when it comes to identifying and handling deviations occurring within a drilling operation. The functionality implemented in the prototype will effectively ease the directional driller’s process in analyzing and monitoring continuously new incoming survey reports, as well as the identification of a proper follow-up decision. The prototype will provide valuable additions to the functionality developed in the CODIO product, adding another level of decision support and automation of tasks in directional drilling. With that being said, the prototype is intended to work as a tool that utilizes the efficiency of tasks involved in detection and handling of deviations, and not to replace human competence.

Considering the technological advance experienced in the oil and gas industry during the past few years, the question emerges why the need for deviation detection in directional drilling has not been addressed already, through implementation of the functionality as an operative service throughout the drilling industry. According to Helge Rørvik at Halliburton, the answer is twofold (personal communication, 30.03.10):

- The industry has maintained a rather conservative attitude towards investments in technology.
- Automatic deviation detection is commonly considered a wanted functionality rather than a needed functionality, which means that the overall cost of producing the functionality outweighs the expected benefits.

With that being said, a full scale event-driven solution handling dynamic and configurable deviation scenarios, including the detection of possible collision scenarios in areas crowded with drilling activities, will greatly reduce the costs and increase the performance in directional drilling (Helge Rørvik, Halliburton, personal communication 30.03.10). Section 8.3.2 elaborates the possibility of extending the functionality implemented in the prototype to support dynamic and configurable deviation scenarios that includes collision detection.
7.4.4 Addressing the research questions

This section will reflect and summarize the research results in context of addressing the research questions we defined back in section 3.2.

7.4.4.1 What synergies can be found between BPM and CEP?

Section 4.10 discusses how the capabilities of BPM and CEP can be combined in order to achieve event-driven behavior in computer systems. BPM is used to manage workflow models containing various business process steps and a description of the entire activity flow in a process, while CEP is used to identify and process events and complex events. By combining these technologies, it is possible to obtain situational knowledge that is discovered, understood, and acted upon in real-time. Our implementation of the prototype utilizes these synergies, introducing event-driven behavior to the process of handling deviations in directional drilling.

In our approach, we only use the prototype to initiate a BPM process whenever some event conditions external to the process are fulfilled, only providing input to the process. However, it is also possible to implement event-driven behavior inside a running BPMN workflow process, letting the prototype invoke specific event processing based from input received from the BPMN process (EDBPM, 2009). Achieving this would require the CEP implementation to monitor one or several BPMN process workflows, and act according to events defined from the current process steps within these models.

A simple example of this is the scenario we used to demonstrate event-driven behavior back in section 4.10.1. In this scenario, the CEP implementation uses BPM as one of the event streams. The CEP engine is set to monitor the BPMN process flow during the drilling operation, keeping track on whether the well is being drilled or whether activities like casing and magnetic correlations are in progress. The business process step is received by the CEP engine as an event, where it is compared to the conditions of the complex events.

7.4.4.2 What is a viable architecture for an Event-Driven BPM prototype?

Section 6.1 describes the architecture proposed as a basis for our prototype implementation. The architecture is designed according to the four logical layers involved in the concept of an Event-Driven Architecture.

Our architecture emphasizes the use of K2 BlackPearl as the Business Process Management System handling the BPMN process, while our own custom-made CEP Engine is set to manage processing of events from incoming real-time data, facilitating the required performance in complex event processing.
There are plenty of other possible architectural solutions that address the implementation of an Event-Driven BPM prototype. The architecture may differ in a varying degree, depending on the needs of the final product and the requirements from the application area. The most important architectural differences will comprise the following factors:

- **The chosen approach to implement CEP capabilities**: As we discussed back in 4.4.2, event processing software may be divided into four tiers. Each of these tiers represents a level of complexity in required CEP capabilities. In our implementation, we have chosen to write custom code that satisfies our need for tier 1 CEP functionality. However, a more complex EDBPM solution might need more extensive complex event processing, introducing a higher level of complexity. Existing professional CEP solutions, both as engines and frameworks, offer CEP capabilities supporting all tiers of event processing.

- **The event streams used**: Event streams may be received from various types of sources. Our implementation makes use of directional data measured from MWD tools at the drilling well, available in real-time. Other possible event stream sources include BPMN processes, databases, timers, RFID, Enterprise Application Integration (EAI), and other IT-systems and services related to the environment of the particular application area.

- **The handling of real-time event streams**: Event streams may vary significantly in regards to hard and soft real-time deadlines, a concept we discussed back in 4.9. The survey data processed in our prototype is based on soft deadlines, with no requirements of rapid processing. Complex EDBPM systems may include multiple large and constantly active event streams, often with the need to process certain events before receiving other events, making quick processing and priority handling a matter of importance.

- **The BPM solution**: Currently, there exists a diversity of systems realizing BPM solutions. K2 BlackPearl is used as the BPM platform in the CODIO project, providing Case Management capabilities and a deep integration to Microsoft technology like SharePoint and Communicator. Alternate BPM solutions may include other features and support integration towards different products and platforms, making it beneficial to find the optimal solution conforming to the specific application environment.

Note that the architecture proposed in combination with our prototype should be considered a first approach to introduce event-driven behavior to the Integrated Operations, and will probably become subject of changes and extensions during further development of the prototype.

### 7.4.4.3 How can an EDBPM prototype prove useful in context of the CODIO project?

In chapter 5 we designed a case based on a realistic scenario regarding the detection and handling of deviations in drilling operations. The case models and proposes an extent in the functionality of the product engineered in the CODIO project. The relevant work conducted in the CODIO research project is based on a need to create better routines in regards to the communication and collaboration included when dealing with deviations in drilling operations.
The case proposed in this thesis elaborates the usefulness in automatic detection of deviations occurring in directional drilling. The need is oriented both on the actual detection of a deviation, and the handling of the immediate process once the deviation is known. Our prototype automatically initiates the BPMN process based on findings from real-time data analysis. This makes the prototype able to discover direct threats to the drilling process, perhaps resulting in severe consequences if uncaught.

There may be other application areas for event-driven behavior within the context of the CODIO project. It may be interesting to do further research on this topic, to investigate whether additional needs in the CODIO project could be addressed using event-driven behavior. Section 8.3.1 discusses this in more detail.
8 Conclusion and further work

8.1 Conclusion
The thesis presents an approach to identify and handle possible deviation scenarios in directional drilling, using the concepts of EDBPM. We have identified and described a case elaborating the possibility and the potential usefulness of adding event-driven behavior to the product engineered by the CODIO research group. A software prototype has been developed with the purpose of assisting workers in the detection and immediate management of occurring deviations. We have through experiments successfully demonstrated event-driven behavior in the process of identifying and handling deviations inside our simulated environment. Based on this, we can conclude that this first step towards EDBPM within Integrated Operations has proven to be successful, although a lot of work remains until the ultimate vision of automatic deviation detection is realized. The research conducted in this thesis has given us great knowledge and insight in the problem area, and we consider our approach an important step towards a final and operative product.

8.2 Achievements
We have accomplished several achievements throughout our research. The most important achievements can be summarized into the following:

- Explored and discussed synergies between BPM and CEP, and explained the concept of EDBPM.
- Identified a suitable application area for event-driven behavior within the CODIO project. The event-driven behavior has been demonstrated through a concrete example in context of the CODIO project.
- Proposed and developed an event-driven prototype that conforms to the concepts of event-driven architecture. Event processing logic is successfully implemented within our tailor-made CEP solution, based on tier 1 CEP principals.
- Engineered workflow logic within K2 BlackPearl, in order to demonstrate the prototype’s expected behavior when integrated with the product developed in the CODIO project.
- Described and demonstrated an approach to automatically detect deviations occurring in directional drilling, as well as suggesting a proper follow-up action.
- Obtained sufficient insight in directional drilling, including an adequate understanding of what the various directional survey parameters represent. Furthermore, we have acquired the ability to interpret the significance of parameter combinations, in order to identify deviations occurring in the drilling space.
- We have provided detailed elaboration of six proposed deviation scenarios, each representing a unique deviation situation. We have also created sets of complex event rules reflecting these deviation scenarios.
- Identified important techniques to address the detection of invalid survey readings, the analysis of deviation outcomes, calculation of well progression, and the need for acceptance levels at alternate depths.
Successfully demonstrated all aspects of the prototype’s current functionality and behavior through experiments conducted in our simulated environment. The experiment results match the expected prototype output.

8.3 Subjects for further research
This section will propose and discuss interesting subjects for further research.

8.3.1 Additional application areas within the CODIO project
Our approach to introduce event-driven behavior within the CODIO project has been oriented on detecting deviations in directional drilling. However, there may be several additional application areas within the drilling domain where event-driven behavior could yield beneficial functionality. EDBPM is quite adaptive, with the ability to identify and monitor a large variety of event types within alternate event stream sources. Thus, EDBPM can be utilized to detect opportunities and avoid threats within a range of different problem areas.

Another application area was briefly discussed during the early phase of this work, as an alternative to the case described in this thesis. The application area in question was about development of a prototype with the capability to track overall performance of a drilling well, based on event-driven behavior. The well performance could comprise events including:

- BPM process monitoring.
- Total cost compared to budget.
- Distance drilled compared to final target.
- The amount of time used compared to time estimates made in advance.
- Deviations in technical aspects of drilling, such as the mud volume pumped down the well, or the torque and weight on the drill bit.

The information gained from tracking well performance should be presented in a suitable format at CODIO’s well-site portal. Such information could be used to provide insight in estimated future situations, explaining what will happen if the drilling operation continues with its current properties. Ultimately, this insight could be considered as valuable arguments within decision management.

In order to support such functionality, the prototype would require analysis of events from multiple sources, at least including relevant BPMN processes, cost calculator tools and extended use of the MWD tools. The prototype would also have to include complete integration capabilities towards the SharePoint well sites. A user-friendly interface could also be implemented, providing input of well-specific data such as estimated time and cost for well completion.

8.3.2 Improvements of the prototype
We have demonstrated some important concepts throughout this first attempt to address automatic deviation detection in directional drilling. We have also briefly discussed several interesting topics in regards to further improving the prototype. Due to limited time and resources available, the functionality presented within these suggested topics has not been included in the scope of this thesis. This section will discuss these improvements in more detail, explaining how they might provide functional advantages over the current solution.
8.3.2.1 Dynamic complex event rules

One of the most immediate and imperative improvement of the prototype is to investigate the possibility of introducing dynamic complex event rules. By dynamic rules, we mean rules having criteria conditions that collaborate with each other. Efficient collaboration between the rule criteria could be realized through condition values that dynamically scale and correlate to each other in order to conform to all situations.

The prototype’s current functionality presents standardized and static condition values, applied to all wells and throughout the entire drilling operation. As we already discussed back in section 5.4.4, this does not entirely reflect the real world, as each well typically differ in acceptance level in deviations due to the following reasons (Helge Rørvik, Halliburton, personal communication 30.03.10):

- The geology is a lot denser at lower depths, making it easier and more beneficial to perform corrections early in the process.
- Several wells are drilled in close proximity to other wells, making particular depths subject to collision issues.
- The estimated size of the target varies from each well, affecting the maximum allowed deviation.
- A deviation early in the drilling process could easily escalate to a more significant deviation after a short period of drilling, as opposed to deviations occurring at a higher depth.

We have already introduced a technique attempting to cope with this issue, by letting the rules have several condition sets, each conforming to a specific depth. By also introducing dynamic complex event rules, this issue could be addressed more efficiently as the rule criteria could be designed to adapt to each unique situation. Criteria conditions could be modeled based on key information compliant to each specific well, scaling to all particular depths and supporting a smooth transition between the different condition sets. Rule criteria having the ability to alter condition values dynamically would also allow for more complex configurations in regards to categorizing deviations into different severity levels, providing increased accuracy in suggested follow-up decisions.

8.3.2.2 User-based manipulation through a graphical user interface

It could become beneficial to extend the prototype’s capabilities to also include a user interface supporting user-based rule criteria manipulation. When introducing arbitrary numbers of acceptance levels, as well as dynamic event rules that corresponds to these levels, the need to configure well-specific deviation scenarios in a simple and effective manner emerges. Key information regarding all relevant aspects of a well needs to be made available to the prototype, in order to be able to calculate and figure out the optimal combinations of rule condition values. By providing a user-based interface directly connected to the prototype, the user could design custom-made condition sets containing values that are completely compliant with each unique well at all depths.
8.3.2.3 Extended functionality to include collision detection

Further along the way, the prototype could be extended to include functionality supporting collision detection. As mentioned above in section 8.3.2.1, some wells are drilled within close proximity to previously drilled wells, introducing the need to avoid well collisions. Additional complex event rules, defining the maximum allowed distance to surrounding wells, could eventually be implemented in the prototype. These anti-collision rules could be designed with extremely tight deviation tolerance, only activated in areas where collisions might occur. The events behind these criteria could be engineered from separation factors and the distance between the wells, which are two important factors currently used to manually determine anti-collision status in drilling operations (Helge Rørvik, Halliburton, personal communication 30.03.10).

8.3.2.4 Including deviation detection in geosteering

As mentioned earlier in section 2.3.1, measured geological data retrieved from the MWD tool is often used by geologists to ensure the well stays within desired geology formations. The approach of geosteering involves drilling in directions determined by predefined geological criteria, based on formation properties like density, porosity, the type of the rock surrounding the well, and the fluids contained in the rock. These geology criteria could be captured within complex event rules, making the prototype able to also detect deviations in geosteering. In order to achieve this, research should be conducted with focus on analyzing the actual parameters included within the geological information received by the MWD tool. Eventually, after identifying possible deviation scenarios related to geosteering, complex event rules conforming to each respective deviation scenario could be designed and implemented in the prototype. In other words, deviation detection in geosteering may be achieved using the same approach that we used to detect deviations in directional drilling during this thesis.

8.3.2.5 Prediction of optimal path according to preset targets

In current directional drilling, preset targets are described and implemented in the drilling plan to define checkpoints in the drilling path (Helge Rørvik, Halliburton, personal communication 30.03.10).

It could become beneficial to further develop the prototype to be aware of these preset targets, in order to achieve the ability to predict the optimal path. In effect, this would mean that a deviation occurring in directional drilling remains undetected as long as the deviation seems to have a positive long-term outcome. Figure 8-1 illustrates a scenario explaining this concept.
Figure 8-1: Prediction of optimal path based on preset targets

As we can see from the figure, the well is planned to be drilled directly north between the starting point and the first preset target. After hitting the first preset target, the planned path is directed towards west in order to reach the second preset target.

Knowing this in advance, we recognize that a deviation towards west between the starting point and the first preset target (represented by the red line in the illustration) should be avoided, as the deviation has a negative outcome. If this particular deviation occurs, the well has to be corrected towards east in order to reach the first preset target. Furthermore, the well has to be directed towards west again to meet the second preset target, implying several abrupt dog leg sections.
On the other hand, if the well is deviating towards east between the starting point and the first preset target, we know the deviation has a positive long-term outcome (represented by the green line in the illustration). This deviation implies a smooth transition through the first preset target, allowing for a consistent and low inclination in dog leg sections.

8.3.2.6 Re-evaluating the CEP implementation

When investigating the viability of these proposed improvements, there should also be a focus on re-evaluating the approach of the technical CEP implementation. The prototype improvements suggested require a CEP solution capable of handling a higher level of complexity. Thus, it might eventually prove more beneficial to migrate to an existing CEP software solution, rather than to keep expanding the current custom-made solution.
Appendix A  Detailed experiment data

This appendix gives a complete overview of the experiment data used in this thesis. Both drilling plans and survey reports are custom made, based on templates received from ConocoPhillips. The values inserted are designed with the sole intention of triggering the corresponding deviation scenario. Hence, the values do not represent an attempt to simulate a complete, real drilling operation, where the values would correlate and be affected by each other.

A.1  Scenario #1 – Northings / Eastings
Table A-1 presents the complete drilling plan used to experiment on deviations in Northings / Eastings.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northings</th>
<th>Eastings</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>396.0</td>
<td>19.47</td>
<td>303.080</td>
<td>324.29</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>437.0</td>
<td>19.75</td>
<td>303.080</td>
<td>354.42</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>468.0</td>
<td>20.53</td>
<td>303.080</td>
<td>384.77</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>499.0</td>
<td>20.97</td>
<td>303.080</td>
<td>404.94</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>530.0</td>
<td>21.25</td>
<td>303.080</td>
<td>425.15</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>561.0</td>
<td>22.41</td>
<td>303.080</td>
<td>445.31</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>721.0</td>
<td>25.87</td>
<td>303.080</td>
<td>557.0</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Table A-2: Experiment survey report - Northings / Eastings

Table A-2 presents the complete survey report used to experiment on deviations in Northings / Eastings.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northings</th>
<th>Eastings</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>396.0</td>
<td>19.47</td>
<td>303.080</td>
<td>324.29</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>437.0</td>
<td>19.75</td>
<td>303.080</td>
<td>354.42</td>
<td>43.52</td>
<td>26.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>468.0</td>
<td>20.53</td>
<td>303.080</td>
<td>384.77</td>
<td>45.52</td>
<td>25.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>499.0</td>
<td>20.97</td>
<td>303.080</td>
<td>404.94</td>
<td>48.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>530.0</td>
<td>21.25</td>
<td>303.080</td>
<td>425.15</td>
<td>48.52</td>
<td>28.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>561.0</td>
<td>22.41</td>
<td>303.080</td>
<td>445.31</td>
<td>48.52</td>
<td>31.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
</tbody>
</table>
### A.2 Scenario #2 – Azimuth

Table A-3 presents the drilling plan used to experiment on the Azimuth parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Easting</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>479.00</td>
<td>1.007</td>
<td>358.270</td>
<td>445.00</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>485.00</td>
<td>1.008</td>
<td>358.270</td>
<td>450.00</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>492.00</td>
<td>1.009</td>
<td>358.270</td>
<td>455.00</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>500.00</td>
<td>1.010</td>
<td>358.270</td>
<td>460.00</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>506.00</td>
<td>1.011</td>
<td>358.270</td>
<td>465.00</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>517.00</td>
<td>1.013</td>
<td>358.270</td>
<td>470.00</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>521.00</td>
<td>1.014</td>
<td>358.270</td>
<td>475.00</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>651.00</td>
<td>4.392</td>
<td>358.270</td>
<td>594.00</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Table A-3: Experiment drilling plan - Azimuth

Table A-4 presents the survey report used to experiment on the Azimuth parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Easting</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>479.00</td>
<td>1.007</td>
<td>2.570</td>
<td>445.00</td>
<td>39.38</td>
<td>26.57</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>500.00</td>
<td>1.010</td>
<td>3.530</td>
<td>460.00</td>
<td>40.52</td>
<td>27.29</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>521.00</td>
<td>1.014</td>
<td>3.960</td>
<td>475.00</td>
<td>41.67</td>
<td>27.72</td>
<td>0.00</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Table A-4: Experiment survey report - Azimuth

### A.3 Scenario #3 – Dog Leg Section

Table A-5 presents the drilling plan used to experiment on the Dog Leg Section parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Easting</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>477.0</td>
<td>0.007</td>
<td>303.080</td>
<td>404.49</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>488.0</td>
<td>0.010</td>
<td>303.080</td>
<td>413.49</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>499.0</td>
<td>0.014</td>
<td>303.080</td>
<td>421.49</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>510.0</td>
<td>0.017</td>
<td>303.080</td>
<td>435.49</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>521.0</td>
<td>0.020</td>
<td>303.080</td>
<td>446.49</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>676.0</td>
<td>0.084</td>
<td>303.080</td>
<td>558.00</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Table A-5: Experiment drilling plan - Dog Leg Section

Table A-6 presents the survey report used to experiment on the Dog Leg Section parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Easting</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>477.0</td>
<td>0.007</td>
<td>303.080</td>
<td>404.59</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>4.44</td>
</tr>
<tr>
<td>488.0</td>
<td>0.010</td>
<td>303.080</td>
<td>413.49</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>4.74</td>
</tr>
<tr>
<td>499.0</td>
<td>0.014</td>
<td>303.080</td>
<td>421.39</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>6.04</td>
</tr>
<tr>
<td>510.0</td>
<td>0.017</td>
<td>303.080</td>
<td>435.29</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>7.24</td>
</tr>
<tr>
<td>521.0</td>
<td>0.020</td>
<td>303.080</td>
<td>446.19</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>7.44</td>
</tr>
</tbody>
</table>

Table A-6: Experiment survey report - Dog Leg Section
A.4 Scenario #4 – Inclination
Table A-7 presents the drilling plan used to experiment on the Inclination parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Eastings</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>495.00</td>
<td>19.47</td>
<td>303.080</td>
<td>424.29</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>496.00</td>
<td>19.75</td>
<td>303.080</td>
<td>424.42</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>497.00</td>
<td>20.53</td>
<td>303.080</td>
<td>424.77</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>498.00</td>
<td>20.97</td>
<td>303.080</td>
<td>424.94</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>499.00</td>
<td>21.25</td>
<td>303.080</td>
<td>425.15</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>500.00</td>
<td>22.41</td>
<td>303.080</td>
<td>425.31</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>501.00</td>
<td>23.16</td>
<td>303.080</td>
<td>425.49</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>653.00</td>
<td>27.33</td>
<td>303.080</td>
<td>538.60</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Table A-7: Experiment drilling plan – Inclination

Table A-8 presents the survey report used to experiment on the Inclination parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Eastings</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>495.00</td>
<td>16.61</td>
<td>303.080</td>
<td>424.13</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>496.00</td>
<td>16.63</td>
<td>303.080</td>
<td>424.27</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>497.00</td>
<td>17.45</td>
<td>303.080</td>
<td>424.56</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>498.00</td>
<td>25.34</td>
<td>303.080</td>
<td>424.87</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>499.00</td>
<td>17.87</td>
<td>303.080</td>
<td>425.19</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>500.00</td>
<td>17.94</td>
<td>303.080</td>
<td>425.54</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>501.00</td>
<td>18.11</td>
<td>303.080</td>
<td>425.96</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Table A-8: Experiment survey report - Inclination

A.5 Scenario #5 – Total Vertical Depth
Table A-9 presents the drilling plan used to experiment on the Total Vertical Depth parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Eastings</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>457.0</td>
<td>19.47</td>
<td>303.080</td>
<td>365.0</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>478.0</td>
<td>19.75</td>
<td>303.080</td>
<td>381.0</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>499.0</td>
<td>20.53</td>
<td>303.080</td>
<td>402.0</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>520.0</td>
<td>20.97</td>
<td>303.080</td>
<td>418.0</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>541.0</td>
<td>21.25</td>
<td>303.080</td>
<td>430.0</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>668.0</td>
<td>28.39</td>
<td>303.080</td>
<td>538.0</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
<td>0.34</td>
</tr>
</tbody>
</table>

Table A-9: Experiment drilling plan - Total Vertical Depth
Table A-10 presents the survey report used to experiment on the Total Vertical Depth parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Eastings</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>457.0</td>
<td>18.47</td>
<td>303.080</td>
<td></td>
<td>374.5</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
</tr>
<tr>
<td>478.0</td>
<td>18.75</td>
<td>303.080</td>
<td></td>
<td>391.5</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
</tr>
<tr>
<td>499.0</td>
<td>19.53</td>
<td>303.080</td>
<td></td>
<td>412.2</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
</tr>
<tr>
<td>520.0</td>
<td>19.97</td>
<td>303.080</td>
<td></td>
<td>430.8</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
</tr>
<tr>
<td>541.0</td>
<td>20.25</td>
<td>303.080</td>
<td></td>
<td>445.1</td>
<td>40.52</td>
<td>24.49</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Table A-10: Experiment survey report - Total Vertical Depth

A.6 Scenario #6 - Vertical Section

Table A-11 presents the drilling plan used to experiment on the Vertical Section parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Eastings</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>397.0</td>
<td>19.47</td>
<td>303.080</td>
<td></td>
<td>444.70</td>
<td>30.52</td>
<td>29.49</td>
<td>16.76</td>
</tr>
<tr>
<td>448.0</td>
<td>19.75</td>
<td>303.080</td>
<td></td>
<td>465.70</td>
<td>40.52</td>
<td>34.49</td>
<td>17.76</td>
</tr>
<tr>
<td>499.0</td>
<td>20.53</td>
<td>303.080</td>
<td></td>
<td>486.70</td>
<td>50.52</td>
<td>39.49</td>
<td>18.76</td>
</tr>
<tr>
<td>530.0</td>
<td>20.97</td>
<td>303.080</td>
<td></td>
<td>507.70</td>
<td>60.52</td>
<td>44.49</td>
<td>19.76</td>
</tr>
<tr>
<td>561.0</td>
<td>21.25</td>
<td>303.080</td>
<td></td>
<td>528.70</td>
<td>70.52</td>
<td>49.49</td>
<td>20.76</td>
</tr>
<tr>
<td>715.0</td>
<td>25.83</td>
<td>303.080</td>
<td></td>
<td>625.07</td>
<td>120.52</td>
<td>74.49</td>
<td>42.76</td>
</tr>
</tbody>
</table>

Table A-11: Experiment drilling plan - Vertical Section

Table A-12 presents the survey report used to experiment on the Vertical Section parameter.

<table>
<thead>
<tr>
<th>Measured Depth</th>
<th>Incl.</th>
<th>Azim.</th>
<th>Total Vertical Depth</th>
<th>Northing</th>
<th>Eastings</th>
<th>Vertical Section</th>
<th>Dog Leg Section</th>
</tr>
</thead>
<tbody>
<tr>
<td>397.0</td>
<td>18.47</td>
<td>303.080</td>
<td></td>
<td>452.70</td>
<td>29.80</td>
<td>29.49</td>
<td>8.96</td>
</tr>
<tr>
<td>448.0</td>
<td>18.75</td>
<td>303.080</td>
<td></td>
<td>474.20</td>
<td>38.78</td>
<td>33.49</td>
<td>7.97</td>
</tr>
<tr>
<td>499.0</td>
<td>19.53</td>
<td>303.080</td>
<td></td>
<td>494.90</td>
<td>48.36</td>
<td>38.34</td>
<td>7.16</td>
</tr>
<tr>
<td>530.0</td>
<td>19.97</td>
<td>303.080</td>
<td></td>
<td>516.0</td>
<td>57.87</td>
<td>42.84</td>
<td>6.46</td>
</tr>
<tr>
<td>561.0</td>
<td>20.25</td>
<td>303.080</td>
<td></td>
<td>536.1</td>
<td>66.89</td>
<td>47.32</td>
<td>5.56</td>
</tr>
</tbody>
</table>

Table A-12: Experiment survey report - Vertical Section
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Azim</td>
<td>Azimuth</td>
</tr>
<tr>
<td>BAM</td>
<td>Business Activity Monitoring</td>
</tr>
<tr>
<td>BPD</td>
<td>Business Process Diagram</td>
</tr>
<tr>
<td>BPM</td>
<td>Business Process Management</td>
</tr>
<tr>
<td>BPMI</td>
<td>Business Process Management Initiative</td>
</tr>
<tr>
<td>BPMN</td>
<td>Business Process Modeling Notation</td>
</tr>
<tr>
<td>CEP</td>
<td>Complex Event Processing</td>
</tr>
<tr>
<td>CMS</td>
<td>Content Management Systems</td>
</tr>
<tr>
<td>CODIO</td>
<td>Collaborative Decision Making in Integrated Operations</td>
</tr>
<tr>
<td>CRM</td>
<td>Customer Relationship Management</td>
</tr>
<tr>
<td>DLS</td>
<td>Dog Leg Section</td>
</tr>
<tr>
<td>E/-W</td>
<td>East / -West</td>
</tr>
<tr>
<td>EAI</td>
<td>Enterprise Application Integration</td>
</tr>
<tr>
<td>EDA</td>
<td>Event-Driven Architecture</td>
</tr>
<tr>
<td>EDBPM</td>
<td>Event-Driven Business Process Management</td>
</tr>
<tr>
<td>EPL</td>
<td>Event Processing Languages</td>
</tr>
<tr>
<td>ESB</td>
<td>Enterprise Service Bus</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>IFE</td>
<td>Institute for Energy Technology</td>
</tr>
<tr>
<td>Incl</td>
<td>Inclination</td>
</tr>
<tr>
<td>IO</td>
<td>Integrated Operations</td>
</tr>
<tr>
<td>IRIS</td>
<td>International Research Institute of Stavanger</td>
</tr>
<tr>
<td>LOB</td>
<td>Line-Of-Business</td>
</tr>
<tr>
<td>MD</td>
<td>Measured Depth</td>
</tr>
<tr>
<td>MWD</td>
<td>Measurement While Drilling</td>
</tr>
<tr>
<td>N/-S</td>
<td>North / -South</td>
</tr>
<tr>
<td>NTNU</td>
<td>Norwegian University of Science and Technology</td>
</tr>
<tr>
<td>ODC</td>
<td>Onshore Drilling Center</td>
</tr>
<tr>
<td>RFID</td>
<td>Radio-frequency identification</td>
</tr>
<tr>
<td>SOA</td>
<td>Service Oriented Architecture</td>
</tr>
<tr>
<td>TVD</td>
<td>Total Vertical Depth</td>
</tr>
<tr>
<td>VSec</td>
<td>Vertical Section</td>
</tr>
</tbody>
</table>
Bibliography


Norsk Oljemuseum. n.d., “Retningsborer og MWD”. [online] Available at: http://www.kulturminne-ekofisk.no/modules/module_123/templates/ekofisk_publisher_template_category_2.asp?strParams=8%233%23%23367&iCategoryId=60&iInfold=0&iContentMenuRootId=&strMenuRootName=&iSelectedMenuItemId=1087&iMin=231&iMax=232 [retrieved: 21.04.10]


Wikipedia. 2009g, ”Measured Depth”. [online] Available at: http://en.wikipedia.org/wiki/Measured_depth [retrieved: 01.02.10]


