

BENCHMARKING OMV's WORLDWIDE DRILLING OPERATIONS THROUGH
PLANNED VERSUS ACTUAL COMPARISON

by
Piotr Wilczek

AFFIDAVIT

I declare in lieu of oath, that I wrote this thesis and performed the associated research myself, using only literature cited in this volume.

Leoben, Austria

Date _____

Signed: _____
Piotr Wilczek

ABSTRACT

Setting new goals and targets for improvement is an important task for every growing and competing industry. Only the knowledge of weak points and critical parts can lead to targets that have a significant positive impact on the overall performance.

This task can become more challenging if applied to the petroleum drilling business. Unique, complicated and diversified projects are tough when it comes to analysis. Data availability is limited, quality often lacks the desired level, and standards for certain values are rare. Especially daily drilling reports can become a nightmare for analysis. Wrong codes, big time steps and a lack of definitions make automated analysis almost impossible, while manual analysis can become a lifetime objective.

In this thesis new approaches to overcome those issues are described. They include modifications of the currently used reporting structure by a change of code and a shift in responsibility. This thesis also includes review of the coding used in the reports and changes that could improve report efficiency through analysis software that automatically evaluates drilling projects and benchmarks plan versus actual reports.

If applied those concepts could change the way drilling projects are evaluated. The benchmarks could help find parts of the process that have potential for improvement. An analysis of both the plan and the report would also improve future plans and help in developing better estimates.

The benchmarks developed and applied on an example based on actual data lead to good results and achieve this thesis's goals. Simplified cases demonstrated further potential of the process.

ABSTRACT

Das Setzen neuer Ziele und Maßstäbe für Verbesserungen ist eine wichtige Aufgabe in jeder wachsenden und wettbewerbsfähigen Industrie. Nur das Wissen über Schwachstellen und entscheidende Punkte führt zu einem signifikanten positiven Einfluss auf die gesamte Leistung.

Diese Aufgabe kann in der Erdölindustrie zu einer echten Herausforderung wachsen. Einzigartige, komplizierte und unterschiedliche Projekte sind schwer zu analysieren. Die Verfügbarkeit an Daten ist begrenzt, ihre Qualität entspricht selten den Anforderungen und bestimmte Normen sind schwer zu finden. Vor allem Tagesberichte können bei einer Analyse zu einem Albtraum werden. Falsche Kodierung, große Zeitschritte und ein Mangel an festgesetzten Definitionen machen eine automatisierte Erfassung fast unmöglich, während manuelle Auswertungen eine Sisyphusarbeit darstellen.

In dieser Arbeit werden neue Ansätze für diese Problematik beschrieben. Sie beinhalten Änderungen der aktuellen Kodierung und einer Verschiebung der Verantwortungsbereiche. Diese Arbeit beinhaltet auch ein Screening der Kodizes, die momentan in Gebrauch sind, sowie daraus abgeleitete Verbesserungsansätze, die die Effizienz der Berichte über ein automatisiertes Programm, das geplante Zeit mit der tatsächlichen Arbeitszeit vergleicht, steigern könnten.

Die Anwendung dieser Konzepte könnte die Art der Auswertung von Bohrprojekten verändern. Die entwickelten Zahlen könnten Prozesse finden, die Verbesserungspotenzial bieten. Eine Evaluierung von geplanter und tatsächlicher Zeit würde ebenfalls bessere Pläne und Schätzungen ermöglichen.

Die ausgearbeiteten und auf einem tatsächlichen Projekt angewandten Benchmarks führten zu guten Ergebnissen und erreichten die gesetzten Ziele. Vereinfachte Beispiele zeigten weitere Anwendungsmöglichkeiten.

LIST OF CONTENTS

ABSTRACT	iii
LIST OF FIGURES	x
LIST OF TABLES.....	xii
LIST OF ACRONYMS.....	xiii
Operations.....	xiii
Actions	xiii
Descriptions.....	xiv
ACKNOWLEDGEMENTS	xvi
CHAPTER 1 INTRODUCTION	1
1.1. Benchmarking Drilling	1
1.2. Benchmarking for OMV	2
1.3. From a Plan to a Well	2
1.4. Associated Problems.....	3
1.5. Goals.....	4
CHAPTER 2 BREAKTHROUGH AND CONTROL	5
2.1. Benchmarking	5
2.2. Breakthrough.....	5
2.3. Control	6
2.4. The Optimum Level of Perfection.....	7
2.5. Management versus Operator Controllable Errors	9
2.6. The Feedback Loop	11
CHAPTER 3 CHANGING THE MORNING REPORT	15
3.1. The Coding System	16
3.2. Error Analysis of OMV's existing Coding System	17
3.2.1. Summary	18

3.2.2.	Conclusions	18
3.3.	Improving OMV's coding.....	19
3.3.1.	Requirements.....	19
3.3.2.	Operation Coding.....	19
3.3.3.	Action Coding.....	22
3.4.	The "Report versus Planned" Approach.....	23
3.4.1.	Introduction	23
3.4.2.	Real Case Example "Schönkirchen 436"	24
3.4.3.	Creating the Plan.....	24
3.4.4.	Adapting the Report.....	25
3.4.5.	Creating the Algorithms.....	25
3.4.6.	The Results.....	26
3.5.	Visualization	29
3.5.1.	Time versus Cost	29
3.5.2.	The Deviation Histogram	29
3.6.	Simplified Cases.....	31
3.6.1.	Case #1, Underestimating.....	31
3.6.2.	Case #2, Overestimating	33
3.6.3.	Case #3, Project on time with bad planning	34
3.6.4.	Case #4, the almost "Perfect" Project	36
CHAPTER 4	CONCLUSION.....	38
4.1.	Future Developments.....	39
REFERENCES CITED	40
SELECTED BIBLIOGRAPHY	41
APPENDIX A:	PROBLEMS WHILE EVALUATING DDR FILES	42
A.1.	Theoretical Issues	42

A.1.1.	Hidden actions (HA)	42
A.1.2.	Wrong code (WC).....	42
A.1.3.	Dual actions (DA).....	42
A.1.4.	Inconsistent Coding (IC)	43
A.1.5.	Encapsulation (CAP).....	43
A.2.	Error Evaluation	44
A.2.1.	Rig Service (RSE) related Problems	44
A.2.2.	Rig Repair (RRE) related Problems.....	46
A.2.3.	Health Safety and Environment (HSE) related Problems	48
A.2.4.	Work on Surface (WSF) related Problems	49
A.2.1.	Drilling (DRL) related Problems	54
A.2.2.	Milling Run (MIL) related Problems	59
A.2.3.	Coring (COR) related Problems	60
A.2.4.	Fishing (FIS) related Problems	60
A.2.1.	Conditioning Runs (CNR) related Problems	61
A.2.2.	Equipment runs (EQU) related Problems.....	62
A.2.3.	Perforation (PER) related Problems.....	62
A.2.4.	Completion (COM) related Problems	63
A.2.5.	Wait on Cement (WOC) related Problems	63
A.2.6.	Cementing (CEM) related Problems.....	64
A.2.7.	Casing Run (CAS) related Problems.....	65
A.2.8.	Logging Run (LOG) related Problems.....	65
APPENDIX B:	NEW SUGGESTED CODDING STRUCTURE.....	67
B.1.	Introduction.....	67
B.2.	Drilling Coding	68
B.3.	Casing Coding	70

B.4.	Rig Up/Down Type Run coding.....	72
B.5.	BHA Type coding.....	74
B.1.	Surface Work Coding	75
B.1.1.	Work on Surface	75
B.1.2.	BOP/HW/HP System	76
B.1.3.	Rig Service	76
B.2.	Intervention Coding.....	77
B.2.1.	Rig repair.....	77
B.2.2.	Well Control	78
B.2.3.	Waiting Time.....	78
APPENDIX C:	REPORT VERSUS ACTUAL EVALUATION CODE.....	79
C.1.	Introduction.....	79
C.2.	Requirements.....	79
C.2.1.	Coding Hierarchy.....	80
C.3.	Code Work Flow	81
C.4.	Results	82
C.5.	User Interface	83
C.6.	Report versus Actual Evaluation Code.....	84
C.6.1.	Introduction	84
C.6.2.	'Declaration of constant Column positions.....	84
C.6.3.	'Calculate Actual Running Time.....	85
C.6.4.	'Calculate Planned Running Time.....	85
C.6.5.	'Find Base Action.....	86
C.6.6.	'Find Base Operation.....	86
C.6.7.	'Find Continued Action	87
C.6.8.	'Find Intervention.....	87

C.6.9.	'Find Contineud Operations.....	88
C.6.10.	'Get Total Action Duration.....	89
C.6.11.	'Get Total Actual Operation Duration.....	90
C.6.12.	'Get Total Planned Operation Duration.....	90
C.6.13.	'Get Absolute Action Deviation.....	91
C.6.14.	'Get Lag Time.....	91
C.6.15.	'Get Relative Action Deviation.....	92
C.6.16.	'Get Absolute Operation Deviation.....	92
C.6.17.	'Get Relative Operation Deviation.....	93
C.6.18.	'Get Times without Intervention.....	93
C.6.19.	'Get Depth without Intervention.....	93
C.6.20.	'Get Times with Intervention.....	94
C.6.21.	'Get Depth with Intervention.....	94
C.6.22.	'Create Planned and Actual Operation Duration.....	94
C.6.23.	'Create All Action Histogram.....	98
C.6.24.	'Create Benchmarks.....	101
C.6.25.	'Create All Operation Histogram.....	103

LIST OF FIGURES

Fig. 1-1 From Plan to Well (present Situation)	3
Fig. 2-1 Control and Breakthrough cycles (Juran, 1995)	7
Fig. 2-2 Optimum quality costs (Juran and Gryna, 1980, p 27)	8
Fig. 2-3 From Plan to Well.....	10
Fig. 2-4 From Plan to Well, the Feedback Loop	11
Fig. 3-1 Example Flow Chart	21
Fig. 3-2 Histogram from Schönkirchen 467.....	28
Fig. 3-3 Deviation Histogram Ranges.....	30
Fig. 3-4 Case #1 TVD.....	32
Fig. 3-5 Case #1 Deviation Histogram.....	32
Fig. 3-6 Case #2 TVD.....	33
Fig. 3-7 Case #2 Deviation Histogram.....	34
Fig. 3-8 Case #3 TVD.....	35
Fig. 3-9 Case #3 Deviation Histogram.....	35
Fig. 3-10 Case #4 TVD	36
Fig. 3-11 Case #4 Deviation Histogram.....	37
Fig. B-1 Drilling related coding structure	68
Fig. B-2 Casing related coding structure.....	70
Fig. B-3 Rig up/down type operations coding structure	72
Fig. B-4 BHA type operations coding.....	74
Fig. B-5 Work on surface coding.....	75
Fig. B-6 BOP/WH/HP system coding.....	76
Fig. B-7 Rig service coding.....	76
Fig. B-8 Rig repair coding.....	77
Fig. B-9 Well control coding.....	78

Fig. B-10 Waiting time coding 78
Fig. C-1 VBA Code Flowchart..... 81

LIST OF TABLES

Table 3-1 Benchmarks for Schönkirchen 436	27
Table 3-2 Case #1 Input Data.....	31
Table 3-3 Case #1 Benchmarks	32
Table 3-4 Case #2 Input Data.....	33
Table 3-5 Case #2 Benchmarks	34
Table 3-6 Case #3 Input Data.....	34
Table 3-7 Case #3 Benchmarks	35
Table 3-8 Case #4 Input Data.....	36
Table 3-9 Case #4 Benchmarks	37
Table C-1 Data example.....	80
Table C-2 Summary	82
Table C-3 Benchmarks.....	82
Table C-4 Histogram Data	83
Table C-5 TvD Data.....	83

LIST OF ACRONYMS

Operations

- BOP BOP/WH/HP system
- CAS casing run
- CEM cementing run
- CNR conditioning run
- COM completion
- COR coring run
- DRL drilling run
- EQU equipment run
- EVR evaluation run
- FIS fishing run
- FTR formation treatment run
- HSE health safety environment
- LOG logging run
- MIL milling run
- OTH other
- PER perforation run
- RRE rig repair
- RSE rig service
- SUS suspension
- SUV survey
- TST testing
- WCO well control
- WOC wait on cement
- WOE wait on equipment
- WOO wait on order
- WOW wait on weather
- WSF work on surface

Actions

- BHA bottom hole assembly handling
- C/H circulate to clean hole
- C/M circulate to condition mud
- C/P circulate pill
- C/S circulate for sample
- C/T change tools
- CCA cut casing
- CEC cement casin
- CEP cement plug
- CIR circulate
- CRA change rams
- D/C drill cement
- OTH other
- P/H pipe handling
- P/J prepare job
- P/T pressure test
- PMP mud pumps
- POW power
- R/B rack back
- R/D rig down
- R/U rig up
- REA ream
- S/C slip and cut
- S/I shut in

- D/F drill formation
- DRI drill (kick, H2S)
- DRW drawworks
- E/C equipment check
- F/C flow check
- F/W flow well
- FIT formation integrity test
- HPS high pressure system
- INV investigation
- LOT leak off test
- LWD logging while drilling
- M/C mill casing
- M/O mill obstruction
- N/D nipple down
- N/U nipple up
- OPH opening hole
- S/M safety meeting
- S/O space out
- SCE solids control equipment
- T/S take survey
- TCL tubing conveyed logging run
- TDS top drive system
- TRA training
- TRI trip
- W/E work equipment
- W/L wireline logging
- W/P work pipe
- W/T wiper trip
- WAS wash
- WB wear bushing
- WHW well head work

Descriptions

- B/O break out
- B/U bottoms up
- BHA bottom hole assembly
- BU bottoms up
- CMT cement
- D/A drill ahead
- DC drill collar
- DHM down hole motor
- DP drill pipe
- F/ from
- FOP force of overpull
- FR flow rate
- GR gamma Ray
- N/D nipple down
- N/U nipple up
- P/U pick up
- POOH pull out of hole
- R/B rack Back
- RIH run in hole
- ROP rate of penetration
- RPM rotations per minute
- RT rotary table
- SPP stand Pipe Pressure
- SSD sliding side door
- SSP shut in surface pressure
- SST string stabilizer

- HWDP heavy weight drill pipe
- L/D lay down
- LOT leak off test
- M/U make up
- MM mud motor
- MW mud Weight
- MWD measurement while drilling
- T/ to
- TOF top of fish
- TOL top of Liner
- TRT trouble Time
- W/ with
- W/D wash down
- X/O cross over

ACKNOWLEDGEMENTS

I wish to express my gratitude for all the support I received while writing this thesis from my advisor Dr. Alfred Eustes of Colorado School of Mines and Dr. Gerhard Thonhauser of Montanuniversität Leoben and the entire Department of Petroleum Engineering of both schools.

Further gratitude is expressed for Herman Spörker, Stefan Knehs and Markus Doschek from OMV who provided me with the necessary input and suggestions throughout the work and giving me the opportunity to write this thesis as well as giving me the necessary financial support.

Special thanks are also due to Dr. Will Fleckenstein and Dr Jennifer Miskimins for serving on my thesis committee and to DDr. Ramona Graves for the support during the thesis defense.

CHAPTER 1 INTRODUCTION

Management responsible for continuously done processes, being it manufacturing of car components or processing of customer data in a call center, is interested in becoming better. Decreasing time, improving quality or reducing cost could be potential targets. To achieve those targets it is important to know what is going on. This knowledge is delivered through benchmarking, a continual process of monitoring and analysis.

1.1. Benchmarking Drilling

The benchmarking process for drilling operations faces a big challenge: Every drilling project is unique. Formation, target depth, drilling crew, equipment, section, weather situation and location have significant impact on the overall result. Thus, benchmarking has to take into account a lot of different parameters. A comparison of two different projects will only lead to useful results if they are similar enough.

Key performance Indicators

Benchmarks that take into account only a small portion of the drilling job, such as rate of penetration (ROP) or connection time, are a good way to overcome those issues (Valdez and Sager, 1995). However, those key performance indicators benchmark only parts of the process, parts that have the biggest impact on overall performance. By increasing their performance through benchmarking the overall performance is increased as well. However, they fail to describe the whole process and are only available and relevant if good monitoring and reporting take place.

Data quality is another issue: Morning drilling reports usually fail to provide the data that is of real interest for this kind of analysis (Thonhauser and Mathis, 2006)). Huge amounts of data that cannot be processed manually and high error rates making automated comparison inaccurate are big issues.

In the last years, new approaches have been developed that differed in the which data was collected. Reporting vs. planned (Thorogood et al, 2000) have been new approaches that followed a new paradigm. Instead of solely focusing on what was done, the report included information on what was supposed to be done. This increase in the spectrum of reported data could allow for new benchmarking methods, as well as an increase in data quality.

1.2. Benchmarking for OMV

OMV is Austria's biggest oil company operating in the upstream and downstream business with drilling sites worldwide. OMV approached the author of this thesis with the request to analyze the existing monitoring system and find ways how to improve it. This thesis is the result of that work; and while it is designed for OMV's worldwide operations, the processes are applicable to other operators.

1.3. From a Plan to a Well

The process of drilling a well (once a candidate has been selected and objectives defined) begins within OMV with the creation of an Intent to Drill (ITD). This legal document, for Austrian authorities, summarizes: the targets of the project, economics, certain technical aspects and a planned duration. If approved it serves as a plan for the next step: The Standing Drilling Instructions (SDI).

SDI's are simple (not standardized) documents created by the drilling management representative on site as a list of tasks for the contractor who is running the rig site. Since the contractor only does what is told on the SDI (S. Knehs, OMV, 2008, personal communication) everything (from trivial things like cleaning the rig, to advanced technical instructions for drilling the formation) that has to be done in order to achieve the defined targets from the ITD is explicitly stated in the SDI. It is a "To Do List" handed in daily and adapted for unexpected events or changes in plan (sidetracks, fishing operations, etc.).

For monitoring the contractor's operations, OMV uses a simple system: The Daily Drilling Report (DDR). DDRs are created for every workday; they include a chronological summary of all jobs performed on the rig, including parameters like time needed or depth. Those documents, all collected in one master file, are a summary of how the project progressed and what the results were.

The DDRs Master Files are a basis for an analysis and resources for new plans which close the cycle. This cycle is summarized in Fig. 1-1.

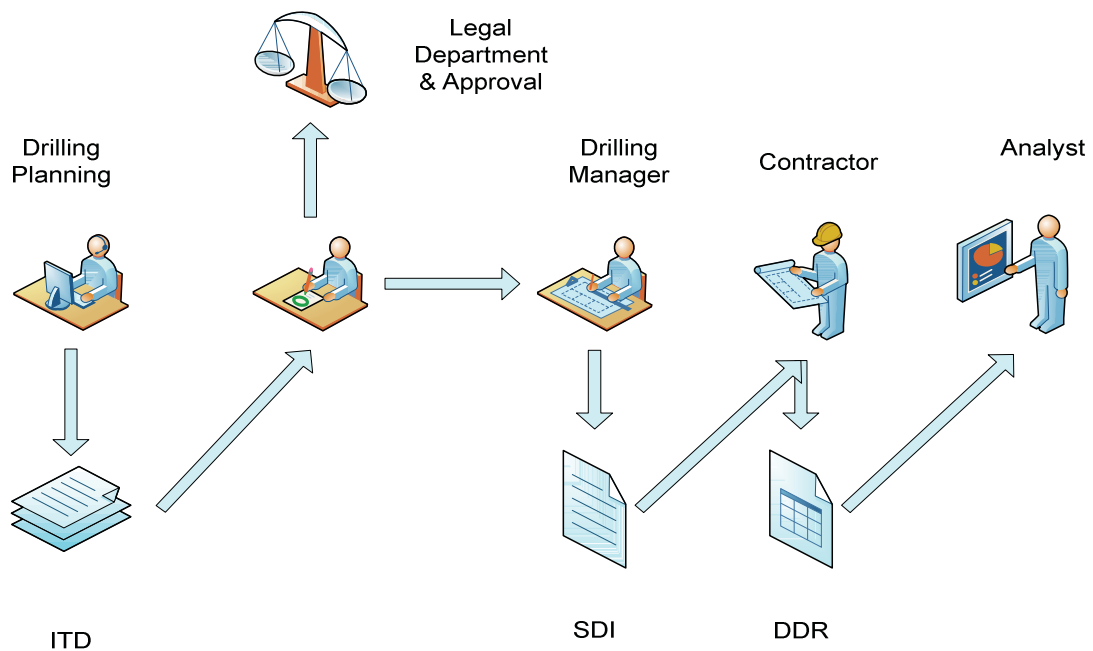


Fig. 1-1 From Plan to Well (present Situation)

1.4. Associated Problems

The biggest problem in this cycle is the DDR. It is inaccurate, inconsistent, and lacks the quality to create a meaningful Analysis (Details on quality problems in DDRs can be found in the chapter ‘ and in Appendix A).

A second issue is the redundancy in the documents. Both the SDIs and the DDRs describe the same thing, using the same terminology, but they are in different formats (SDIs are traditionally Word files, while DDRs are made through Excel) and they are created independently of each other. SDIs are describing the project before it is done, DDRs are a record of performed tasks.

Finally, another problem is the fact that the analyst doesn’t have access to the SDIs on a standard basis. The source of information is the DDR only, reducing this information to what was actually accomplished. The ITD is available but its use for planned vs. actual comparison is minimal. A comparison of the time versus depth curves is possible; however, due to lack of any detail, the only information one can get is that the well was not quite as it was planned.

1.5. Goals

The goals of the thesis are to address those issues by:

- Changing the DDRs: improving the coding structure with special focus on management techniques
- Changing the concept of SDIs: from instructions to an executable plan
- Benchmarking: creation of numbers that help to evaluate the drilling process

CHAPTER 2 BREAKTHROUGH AND CONTROL

This chapter introduces methods for quality control in other industries. Examined are how they work and how they could be applied to a whole different animal namely the Drilling Industry.

2.1. Benchmarking

Camp (1989) defines benchmarking as a:

Continuous process of measuring products, services, and practices against the company's toughest competitors or those companies renowned as industry leaders.

The goal of benchmarking is defined by Juran and Godfrey (1999) as:

The purpose of benchmarking is derived primarily from the need to establish credible goals and pursue continuous improvement. It is a direction-setting process, but more important, it is a means by which the practices needed to reach new goals are discovered and understood.

In subsequent chapters, “the standard” (A point that has to be reached like: Quota, Specification, Budget, Schedule...(Juran, 1995)) will be described. Benchmarking is a process that is about this standard. In benchmarking one tries to find a good standard, define it, and compare it, to other projects, to competitors and to other industries.

The standard is important because it is the pivot point for breakthrough and control and substantially a key in making management decisions. This makes the benchmarking process important since good decisions are a question of good standards.

2.2. Breakthrough

Breakthrough means change, a dynamic, decisive movement to new higher levels of performance. (Juran, 1995)

In the oilfield, breakthrough is usually described as “breaking the technical limit”. New technology, better techniques or changed procedures can result in a breakthrough. Everything that leads to a new level of performance can be described as breakthrough. Benchmarking itself is not a breakthrough tool, rather it helps find potential for breakthrough, but its main task is to achieve control.

2.3. Control

Control means staying on course, adherence to standard, prevention of change. Under complete control, nothing would change – we would be in a static, quiescent world. (Juran, 1995)

Although it seems bad, control is something that must be achieved on any project. It is crucial for a stable environment and for good planning as well as for reaching the targets.

At a drilling site example situations where control is needed might be:

- Operations are taking much longer than expected;
- Crews spend too much time non-productive time; or,
- Triplex pump problems are reported frequently because of low maintenance intervals.

These situations are changes, unwanted changes from what is defined as standard.

Methods to deal with those problems and preventing of change are subject to control (Juran and Godfrey 1999).

One must be aware of the fact that since control is the opposite of breakthrough; no progress will be made in a completely static environment. A process that is fully under control will reach its limits and won't go any further. In some cases; however, reaching the limits can mean a huge improvement.

Control and breakthrough usually form a cycle. A phase of control, where everything is set and brought to its limits is followed by a breakthrough leading to new targets and a new phase of control. This cycle is shown in Fig. 2-1.

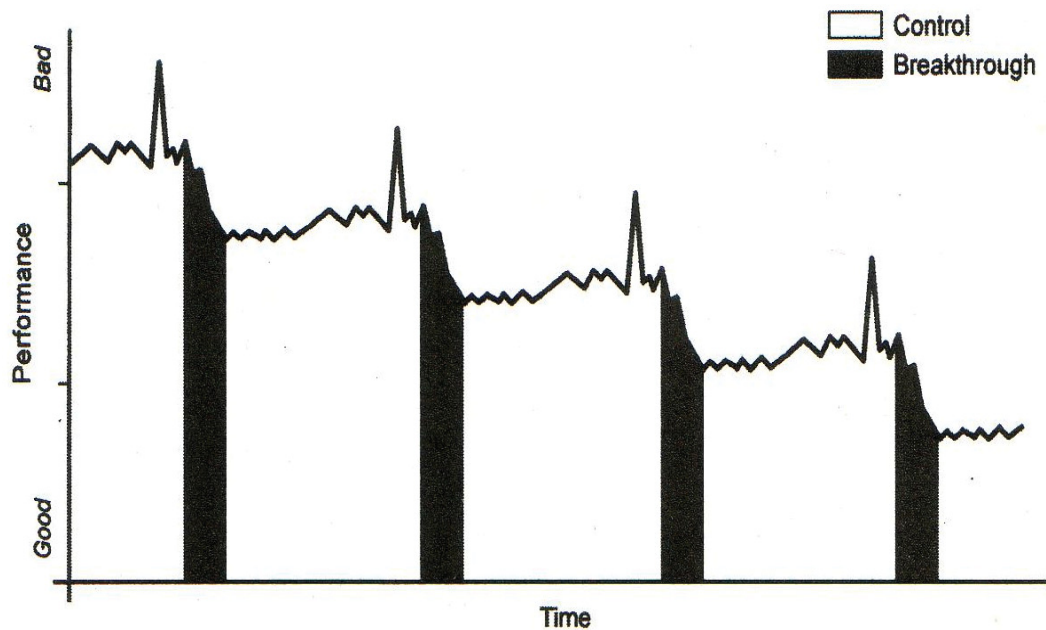


Fig. 2-1 Control and Breakthrough cycles (Juran, 1995)

The benchmarking approaches described in this thesis are all means of control. They are designed to find defective parts of the process. (The elimination of those defective parts is a different story which will not be part of the thesis. Good sources of information on this topic are books cited in the reference section, especially the J. M. Juran books.)

2.4. The Optimum Level of Perfection

This approach can be described by a simple example: a pump failure with no backup pump. There are two approaches to handle this problem:

- There is a chance that the pump will fail. One accepts this chance and handles the problem as it occurs. One repairs the pump when it fails and accepts the costs of repair or replacement and associated downtime.
- There is a chance that the pump will fail. One does not accept this chance and instead, installs means to control and avoid the failure. One does everything to prevent a pump failure of occurring by maintaining, inspecting and monitoring. However, this is costly.

Costs from the first approach are referred to as failure costs, whereas costs from the second approach are referred to as cost of appraisal and prevention (Juran and Gryna, 1980). The

sum of those results is the total cost needed for operating the equipment. In the optimum case, the machine is run at a chance of failure that is associated with the lowest total cost.

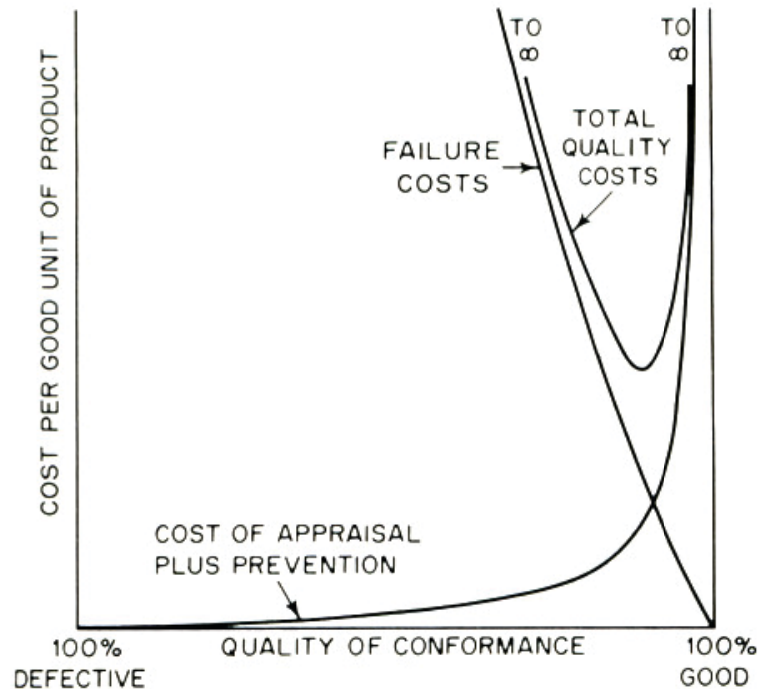


Fig. 2-2 Optimum quality costs (Juran and Gryna, 1980, p 27)

Fig. 2-2 demonstrates the concept. Since it originates from manufacturing, the terminology used might sound alien. The 100% defective area would represent a piece of machinery that doesn't work at all times, whereas 100% good is a machine that does its work perfectly all the time.

This approach can be used in the field in other instances as well. Everywhere one can choose between prevention or failure, this method can be applied to find the best answer. For example, stuck pipe incidents (washing or treating the incident) can be handled that way. But those solutions require data such as statistics about failure probability and time spent on maintenance.

This data could be inferred from DDR files. The requirements would be that there is a way to determine whether time was spent on prevention or failure. This is not possible yet, however if the coding is reshaped and defined accordingly, one could get this type of information.

Data that could be generated this way could be:

- Failure probability
- Cost/time of appraisal and prevention
- Cost/time of failure

Access to this data could allow optimization that might change the drilling process significantly and allow costs to be reduced dramatically.

2.5. Management versus Operator Controllable Errors

The bulk of defect (over 80%) are management controllable not operator controllable (Juran and Gryna, 1980)

It is assumed that this statement is also true for the oil industry. An operator, the person who is performing the actual work, in this case the contractor, can only be held responsible for errors if they match all of the following criteria:

They know what they are supposed to do, they know what they are actually doing [and] they can regulate what they are doing (Juran and Gryna 1980).

If the management fails to provide these conditions, the error is management controllable. This means that there is not a way to stop errors if contractors are not aware of the errors and cannot control them. And clearly, if they don't know what the errors are, they cannot control them. This also implies that there is need for a fast and reliable method for planned time versus actual time comparison. This is better known as "benchmarking".

The implication of this concept into the DDR system can be reached through the simplification of the reporting cycle. The problem of two differing documents describing the same process (DDR and SDI) can be solved by redesigning the system to one that can handle both the plan and the report.

Instead of creating just a SDI, the drilling manager would create an Excel file that would contain all the instructions, but put them in a format that would serve as a theme for the report. The actual operations would be reported on this same field. The people responsible, for the report (operator and contractor) would always be conscious of the plan. Any deviations would be immediately visible to the contractor allowing for better control of their own performance.

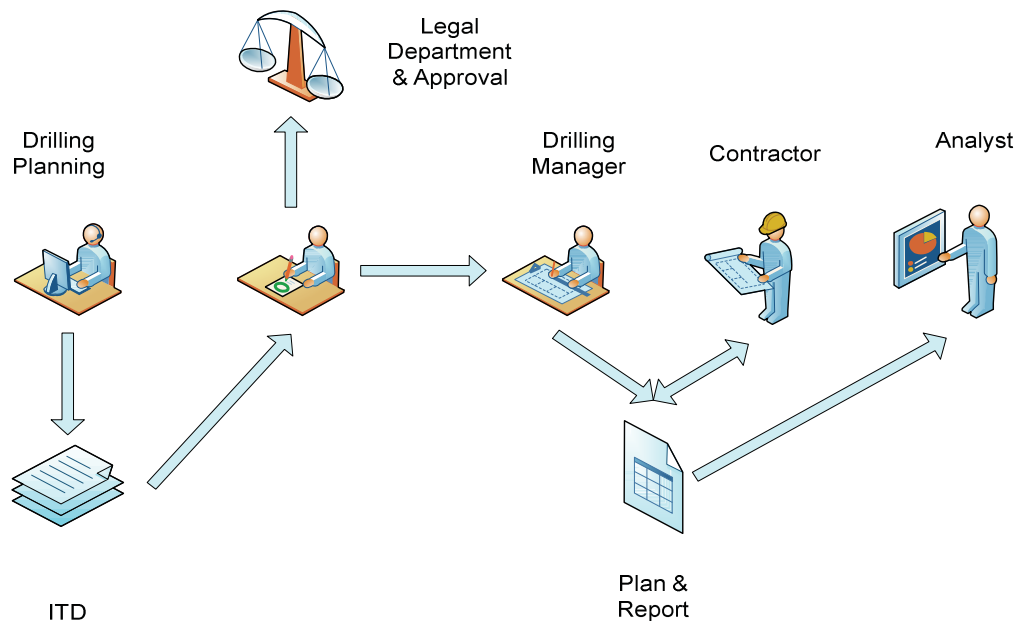


Fig. 2-3 From Plan to Well

Fig. 2-3 visualizes the suggested changes. Note that the analysis processors have access to both the plan and report instead of only the DDR. This gives them the opportunity to use the benchmarks that are introduced later on in this thesis by comparing planned versus actual as well as other approaches.

There are two types of errors: sporadic and chronic or in other words random and systematic. The first type is characterized by unplanned and/or sudden changes in the process and requires restoring the status quo prior to continuing the process such as: a leak in the pump liner requiring repair. The second type is characterized by an inefficient and/or off benchmark process that needs to be handled by a change of the status quo (Juran 1980) such as washing practices that are being performed more often than what would be needed.

The comparison of planned versus actual operations only works if one can compare actual with something that was planned. This isn't the case with random errors. Those errors or interventions have to be handled separately.

Systematic errors are different. A comparison of planned actions with actual operations can identify systematic errors to a certain degree. However, systematic errors have the tendency to be seen as part of the process and not errors. As soon as a systematic error has been accepted

by both the crew and the company man, those errors will remain invisible to the analyst as the analyst would not know that error is really not part of establishing an efficient process.

2.6. The Feedback Loop

The basic step for quality control is the establishment of a quality loop. The quality loop consists, according to Juran, of the following process:

A sensing device (sensor or receptor) for detecting or measuring operations. The resulting information is transmitted to a control center (collator) which compares what is happening to some concept of standard. The control center then issues orders to a motor device (effector) which takes action to bring results into line with standard. (Juran, 1995)

The implementation of this loop would look like in Fig. 2-4:

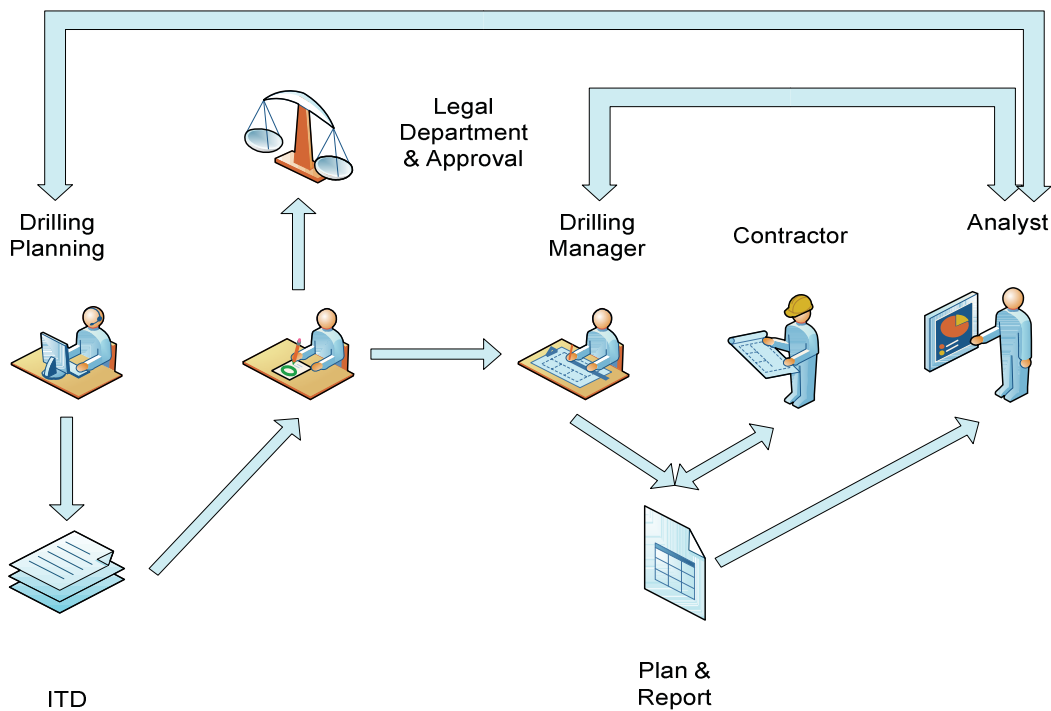


Fig. 2-4 From Plan to Well, the Feedback Loop

In Fig. 2-4 the analyst serves as the sensing device. They need the tools and information necessary to provide feedback. This feedback can contain whatever is needed by the drilling manager and/or the planning department. Those two are the control instances; they decide

whether there is a problem based on the analysis and take the necessary decisions to eliminate it. Those decisions are transmitted to the drilling manager who applies them to his work.

The changes from Fig. 2-3 to Fig. 2-4 are just the two new communication paths, it is important that whatever is determined in the analysis that the knowledge ends up with the people that are making the decisions.

A standard is a value that defines the target for an operation. Operations benchmarked below the standard should be considered problematic, and therefore be subject to in-depth analysis. However the task of setting a certain value standard is not an easy one.

According to Juran, 1995 Standards should be:

Attainable: Ordinary or “normal” individual, applying themselves with reasonable effort, should be able to meet the standards under the condition which are expected to prevail.

Economic: The cost of setting and administering standards should be low in relation to the activity covered to prevail.

Applicable: They should fit the conditions under which they are to be used. If these conditions vary, the standards should contain built-in flexibility to meet these variables.

Consistent: They should help to unify communication and operation throughout all function of the company. They should also be consistent in time, so that planning for tomorrow is done in the light of knowledge gained to date.

All-inclusive: They should cover all interrelated activities. Failing this, standards will be met at the expense of those activities for which standards have not been set.

Understandable: They should be expressed in simple, clear terms, which admit of no misinterpretation or vagueness. The instructions for use should be specific and complete.

Stable: They should have a long enough life to provide predictability and to amortize the effort of preparing them.

Maintainable: They should be so designed that elements can be added, changed, and brought up to date without redoing the entire structure.

Legitimate: They should be officially approved.

Equitable: The standards should be accepted, as a fair basis for comparison, by the people who have the job of meeting the standards.

An acceptable standard can be developed through various approaches. Comparison is one of them. From a pool of projects one selects the one that is running best and defines its performance as the new standard. Or one defines a standard that is not associated with a certain type project and measures it as seen in other industries (0).

Good standards are unfortunately rare when it comes to the drilling industry. Concepts like average rate of penetration, cost per measured depth meter and the technical limit are used, but compared to the list above, those aren't good standards. Rate of penetration as well as cost per measured depth meter fail to describe the whole drilling process and therefore are not all-inclusive, while the technical limit by definition is not attainable. The next section will review with this and use the technical limit as an example.

Technical Limit – Gambling for the Lottery Win

One of the requirements for a good standard is that it is attainable. A standard that defines 99% of the operations as candidates for improvement does not serve any good. Since the task of the standard is to identify the vital few, it has to be selected accordingly.

Technical limit (...) is a level of performance defined as the best possible for a given set of design parameters (Bond et al., 1998)

Technical limit describes an operation where everything has been run completely as planned; every single action has been performed as predicted. The analogy with sports events would be a world record in athletics (Bond et al 1998). The chances for this outcome are minimal. For example, if a drilling operation has 100 different actions and assume that each action has a 5% chance of failure (a 95% chance of success) the chances for reaching the technical limit are

$$(1 - 0.05)^{100} = 0.0059 \quad (3-3)$$

A chance of 0.6% of reaching the technical limit is far away from realistic. (Note that in reality, a drilling operation consists of far more than 100 actions lowering the overall odds of reaching the technical limit.

With this percentage in mind, a well planned on the technical limit will always be a failure if failure is defined as not reaching defined standards. This approach doesn't lead to anything. Although it is a good motivational standpoint to set high targets, but setting them out of reach won't do any good either.

In reality technical limits are usually reached not through perfection (control) but through applying new technology or new approaches that lead to an improvement in performance (breakthrough). In those cases however the technical limit changes as well, so technically it's not the technical limit anymore. That is the reason why the technical limit is not a good standard. Similar situations can be observed in sports. New levels for world records are usually set after new techniques or drugs have been developed (Berhelot et al 2008).

CHAPTER 3 CHANGING THE MORNING REPORT

Morning reports are one of the oldest methods to let the operator know what happened on the rig site. Hand written on a special form, the morning reports were the best way to monitor the progress of a drilling project. They were a basic summary of what happened, when it happened, what the change in drilled depth was and how much it cost. Out of this information, an interested engineer could calculate points necessary to draw time versus depth curves and use them for evaluation purposes.

The problems one faced back then are very similar to the ones an analyst faces nowadays. How accurate is the information that is reported? This question is important since bad data won't deliver any meaningful results. The principle used in statistics of "Garbage in, garbage out" is valid. The chances are that the person writing the reports omitted details because of unawareness, distress, deceit or simply because of laziness.

Handling this issue can become a hard task to the analyst since in most cases the only source of information she/he has are the morning reports. Cross checks with other sources of information like mud logs geo log graphs or drilling surveys might provide a crosscheck on the reality of the reports.

Another way to handle this problem is an approach where one tries to create a blame free environment and encourages the contractor to report what really happened without "bad" consequences to him. Good communication between operator and contractor as well as an understanding for the need of good reports that benefit both is a key for accurate reports one can rely on.

Those issues can be a reason why human coding is not the preferred method for reporting. Automated approaches are more reliable and more accurate however they require higher maintenance cost of the whole system. OMV approached the author of this thesis with the request of focusing on drilling reports only in order to get the most out of the existing system.

The morning report system that OMV is using nowadays includes much more information which is collected and processed electronically, what greatly improves possibilities for analysis. However the social problems just mentioned still can prevail if not handled properly.

3.1. The Coding System

It is of crucial importance for project estimation and evaluation to split the project into parts. A drilling project consists of many different operations: drilling, tripping, casing, cementing, rigging up and down, logging and so on. Each operation can be defined by different parameters. For example, the best way to describe a drilling phase is the rate of penetration (ROP). The importance of this one parameter is to define the time and cost of the operation. It is used as a benchmark for evaluation purposes.

The parameters usually only make sense if applied to the operation from which they originated. It is senseless to describe the rigging up process with the ROP. This implies that each operation will have its own parameters. In the optimum case those parameters are sufficient to completely define the operation. This means that the parameter will define what should be done, and will also allow for good estimates about time and cost. All operations needed for the process of drilling a well should then be the sum of those estimates for the whole project.

The existing coding system is a first step to achieve this task. OMV's coding system as it stands is basically a labeling system for different jobs. In the morning report, a task is described and then labeled. This allows the different jobs to be summed up based on their label and therefore be analyzed. For example, one can determine how much time was spent on drilling certain sections or how much time was spent on cementing or other operations. One can sum up all trouble times or make a productive versus non-productive time investigations. This analysis, however, lacks any significant depth since it does not compare those numbers with accepted standards and therefore is not really a mean for identifying problems or fields that need improvement.

There is no way to find out through this system that a driller could perform better if he would use a higher weight on bit (WOB). There is no granularity small enough to find out that Crew A does connections 50% faster than Crew B although they use the same equipment. Those issues, which have a big potential for breakthrough, cannot be addressed with the morning report system in a systematic and controlled way. Accidental discoveries can happen, but they do not happen through the system but through motivated crews that think about what they are doing.

Those issues are system specific. There is no way to handle them through morning reports, no matter how sophisticated the morning report is. Parameters such as the WOB are fluctuating variables that can not be expressed through one constant number. Morning reports will not be able to do that, but they can be used from a different point of view that focuses on time

analysis. However, those things cannot be done through the conventional approach such as productive and unproductive time comparisons.

If an analysis shows that a well had a productive to non productive ratio of 0.5, what does that really mean? It only means that the time spent making progress related to measured depth was only half the time needed for other operations. No other implications can be made. The statement: “Third of the time needed for drilling this well was unproductive.” is only true because of the technical definition of “unproductive”. This can have fatal motivational consequences if presented to the drilling crew implying that they’ve been unproductive or, as they might define it: lazy. The real question is whether this “unproductive” time was really needed? Why did it occur, was it planned or did something happen? Was there a way to avoid this time loss and how much would that cost? The traditional coding system and the corresponding analysis fail in answering these important questions.

OMV uses a two level set of codes for its coding system. It consists of an Action level and an Operation level. The hierarchy being one Operation consists of one or more Actions. Operations refer to big portions of the drilling process, such as drilling, casing, or logging. Usually a whole project only consists out of a few operational phases.

Actions consist of a wide spectrum of job descriptions. For example, they include things such as safety meetings, trips, repair on certain types of equipment, or making up a bottom hole assembly (BHA). The objective of the actions is to describe what has happened during an operation.

One data line (data set) in the morning report consists then of two codes: an operation and action code(s). The first one gives a general idea what type of work is done the second one to be more specific. The data set also includes other information. The most important information is the duration. A date, beginning and end time, a job description, the hole phase, and the depth reached at midnight are all additional information.

If done the right way, as described in the next chapter, those data sets can be used to derive useful information.

3.2. Error Analysis of OMV’s existing Coding System

The analysis of OMV’s coding system was based on the analysis of complete daily drilling reports (DDRs). The DDRs originated from 29 different drilling projects; 18 Austrian and

11 international. They summed up to a total 6607 codes which all have been taken into account during the coding analysis.

An exact description of the errors and problems as well as error examples encountered can be found in Appendix A of this thesis.

3.2.1. Summary

The problems with the existing coding can be reduced to four groups:

- Inclusive coding: One operation can be coded in two (sometimes more) different ways.
- Undefined coding: There is no clear definition when one action ends and the next begins.
- Redundant coding: Operations codes which are not necessary, since they can be addressed more simply.
- Granularity Issues: One code is used to cover up larger parts of the process that should be defined separately.

The last issue is addressed to a wide spectrum of errors. From short duration actions that are summed up as one to 13 hour operations that include more than what was coded.

All those issues make an analysis hard to impossible. A simple question like: “How much time did the rig spend running drill pipe in and out the hole?” is almost impossible to answer. A summation of all trip (TRI) actions will not give the desired result since in practice that time includes: circulation times, bottom hole assembly (BHA) makeup times, flow checks and wash and ream times. On the other hand it can lack those items since sometimes a whole 3000 m long trip is defined as wash because of a nasty 100 m thick section of formation.

3.2.2. Conclusions

Without clear definitions of the coding, an analysis will never provide the required results. It must be clear as to which code has been used to describe a certain work process. The same is valid for operation codes; but in addition, flow charts should be available describing required as well as optional action codes. A reevaluation based on the usage of the codes so far, should help dismiss unnecessary or redundant codes which will clarify and simplify the coding. However the questions asked during the evaluation must be kept in mind, since the main target is not to describe a drilling process but to benchmark it.

3.3. Improving OMV's coding

This chapter describes the process of changing OMV's coding. This change is done based on operation and on action coding level.

3.3.1. Requirements

The requirements for a modification of OMV's coding system are based on the problems reported in the previous section. The modification should be well defined, clear and simple. It should be capable of summing up specific work time easily and accurately and it should help creating a good visualization in form of a work breakdown structure (0). Besides the definition, the coding need rules that make sure the certain actions are used with proper operations, and that an operation consists of all the pertinent actions.

In addition, reporting based on planned procedures will require that the codes are defined in terms of the user. Coding describing unplanned processes, such as well control, will be reserved for the reporting instance while general planned processes will be mainly used by the planning authority.

The coding consists of two different coding types operation and action codes. The operation codes are a general description of a whole process like casing or fishing. Action codes describe the operation codes in more detail, like the making up of a bottom hole assembly for a drilling operation.

3.3.2. Operation Coding

Operation codes should sum up all actions necessary for a certain work process on the rig. A drilling operation on the rig consists of many small jobs such as assembling the BHA or tripping. Some of those jobs are crucial (there is no drilling operation without the bit drilling formation) and must be performed in a certain order (there is no drilling until the bit was tripped into hole). Besides, there are jobs that are optional (hole reaming and washing on the trip out) and jobs that are related to pressure integrity tests of the equipment.

When summing up operation codes of one type, one should see the entire picture of those processes clearly. All jobs that have been necessary should be included and therefore give a good understanding of how the operation is done.

The available coding would theoretically allow this kind of analysis; however, in reality, with the data that was available for the analysis in this thesis, it failed. The main problem was once again the lack of definitions. The procedure of slipping and cutting the drill string was reported correctly in the action code as slip and cut (S/C) however it was paired inconsistently with different operation codes. It would either be used as part of a run or as work on surface (WOS). Both approaches are possible but they have to be consistent, meaning that one uses S/C always as a WOS action or always as part of a certain run.

For the proposed new coding structure, the second option was chosen. S/C is always a job done on the surface. Pairing WOS – S/C contains redundant information because both codes have the same information content. The same approach was used with health safety and environment (HSE) and safety meetings (S/M), S/M is always part of a HSE concern so why code it twice?). However it is interesting for a benchmark to know how long it took to drill a certain hole section. The S/C procedure is a necessary part of this process so including it only makes sense. Once again, the same is valid for safety meetings.

The result of this approach was a simplification of the operation codes, the code HSE has become unnecessary on the operation level and WOS reduced in its range of definition.

There is a similar issue with waiting times, since it has been primarily paired with “other actions” (OTH) (actions that are not defined in the coding). It is not necessary for those codes to be in the operation level but should be action codes. This change would also allow the determination of operations that are more prone to waiting. In this case only wait on order (WOO) and wait on equipment (WOE) since one can assume that there is no correlation between wait on weather (WOW) and a specific operation.

All casing related operations for example casing run (CAS), cementing run (CEM), wait on cement (WOC) have been summed up in one CAS operation. This step was mainly based on the high error rate while coding the time that was spent between running casing and cementing. This modification allows the observation of the necessary procedure for running a casing string and allows the separation the cementing procedure through the cement casing (CEC) action.

Flow Charts

The new coding structure has been summed up in flow charts that have been created for each operation. They include a flow of required actions (actions that always have to be in an operation), optional actions (actions that can be part of the process), related actions (actions that

are done during the operation but that are not related to it) and interventions (actions that have not been planned). An example flow chart is Fig. 3-1. Further flow charts can be found in Appendix B of the thesis.

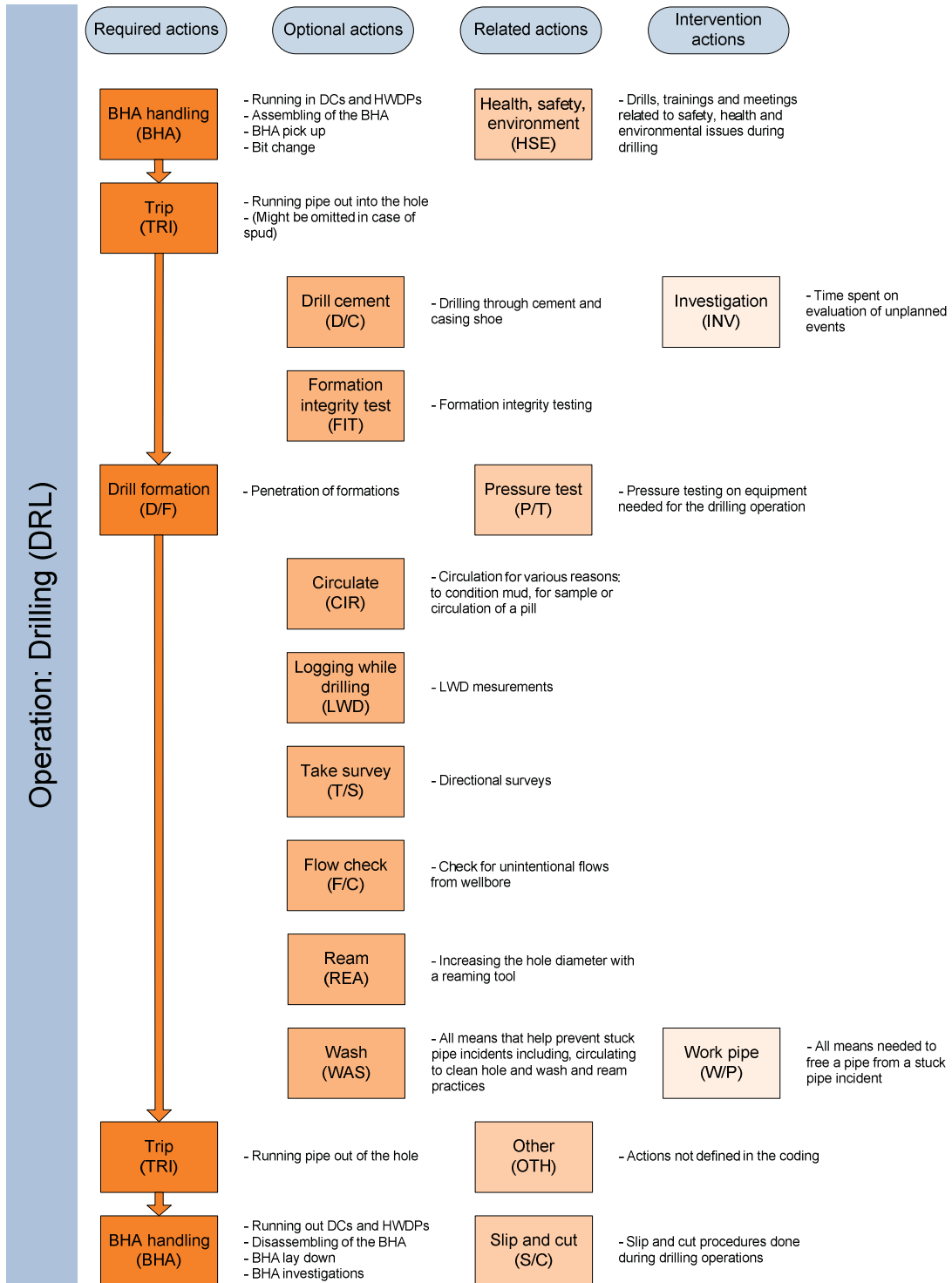


Fig. 3-1 Example Flow Chart

3.3.3. Action Coding

The most important part of the action coding is an accurate definition of the code. The definition makes certain that for each task, there is only one valid code. If there is more than one code that could be used. The report becomes inconsistent in its usage, especially if comparing reports written by different persons. That makes an analysis incorrect.

The definition of the individual action code is based on its pairing with an operation code. The definitions are part of the flowcharts and include usage examples.

The code OTH was not used on the flowcharts. This code is intended to be used for jobs that are unusual or atypical and therefore are not covered by the available codes. In reality, one can observe the trend of an overuse of OTH. Jobs which are rare but coded have the tendency to be covered up with OTH, which leads to inconsistency. An exclusion of OTH from the flowcharts could aid in solving this problem, leading the person planning and reporting to check the available codes first and then using OTH if they don't find one that would be appropriate. However, there should be an explanation of why the code was used.

Some action codes have been removed from the new coding structure. Those were actions that have not been used or used consistently incorrectly, or that have been defined by a different coding. Those coding include: Pipe handling (P/H) and rack back (R/B) that are part of a higher priority code, drill (DRI) and training (TRA) that have been proved not necessary and are now reduced to S/M, and all circulate -pill -for sample -to condition mud -to clean hole (C/*) codes that have been summed up with CIR. Exact error analysis with error examples as well as reasons for change can be found in Appendix A.

Required Actions

Required actions are actions that are required in a certain operation. As mentioned in the proceeding section a drilling operation makes sense if formation is drilled, a casing setting operation cannot be completed without running in the casing and a fishing run can't be successful without making up a fishing BHA. Those seem obvious and logical but a summation of the available data shows that although all those operations are performed they are not always reported in a coded form but become hidden in the description.

The enforcement of certain actions won't solve this problem since it only covers a small part of all codes; but, it is a first approach and will help to create much cleaner results and will

simplify evaluation. A possible solution will be discussed in the upcoming “Report versus Planned” section.

The required action usually consists of a beginning action that defines the beginning of a certain operation. This action typically is BHA for most runs or R/U for cementing and logging procedures. Prepare job (P/J) was excluded since it is used for jobs that can also be described by the two codes mentioned before. This beginning action is followed by a trip, then usually an operation defining action, meaning a job that is the purpose of the whole operation. This action can be wash (WAS) for a conditioning run or drill formation (D/F) for a drilling or coring run accordingly. Then the operation is finished as it started by a trip and a BHA or rig down (R/D).

Optional Actions

Optional actions are usually part of certain operations even though they are not necessarily required. They are directly related to a certain operation. Those actions include for example REA or LWD.

Related Actions

Those are actions which are done during a certain run but aren't directly related to it. HSE is a good example. The topic of the HSE time helps to increase safety for a certain operation but it also serves other following operation. Similar to HSE is slip and cut (S/C). This process might be necessary during a certain operation but it also can be useful in others.

3.4. The “Report versus Planned” Approach

This chapter describes the “Report versus Planned” approach. It also includes its implementation into OMV's existing system.

3.4.1. Introduction

The thought behind the Report versus Planned Approach is that of a basic control cycle (Juran and Godfey, 1999). The results of a work process are compared with other values which in this case the planned values. The resulting difference is an indicator of quality. In this case, the quality is not a measure of the quality of the work done at the rig, it is the quality of the plan. That is the ability of the planning engineer to correctly predict what will happen. This is of huge

significance. A good plan will significantly reduce delay time and will improve the allocation of resources to be more cost effective. It will help to predict feasible candidates for drilling jobs.

The comparison of the Report versus Planned can be performed on different aspects: cost, time, material, depth etc... In this thesis the focus is time since it is directly related to cost (day rate) and it is basically the only variable that is reported in a sufficient granularity and precision in the reports. In the comparison the whole project (e.g. drilling project of well A) is split into smaller parts. Those parts should be small enough to give a better insight but big enough that they can be measured (real time monitoring will deliver better granularity than morning reporting). Those parts are compared to the corresponding parts of the plan. The differences are then summarized as an average and/or plotted on a histogram.

The work processes required for this analysis can be minimal if all prerequisites are true. Those are:

- A plan that is comparable, one that has the same format as the report
- A valid report, a report that represents what happened
- An automated comparison process

Of course, the better the quality of the plan and report, the better is the quality of the results. The engineering axiom: “garbage in, garbage out” is valid.

3.4.2. Real Case Example “Schönkirchen 436”

Schönkirchen 436 was drilled in Austria between November 11th and December 5th 2007. The intended use was as gas storage well with a measured depth of 1490m and a true vertical depth of 1020m. The well has 3 Sections a 17 ½” conductor set at 450m, a 12 ½” intermediate set at 1390m and an 8 3/8” open hole section to TD.

The well was selected for the analysis because of its relatively good report quality. There were minor coding mistakes that were corrected easily with necessary care. And the well had an overall simplicity as well as only standard routines involved.

3.4.3. Creating the Plan

Since the plan available for the well was only a depth versus time curve and an Intent to Drill, a special plan for this analysis had to be created. This creation was based in the end on two things: the TVD curve and the report. The report was correlated with the TVD curve and then a plan was derived.

The plan was created by the author of this thesis based on a small amount of data, meaning that the results won't be a measure of the actual project. However, this data was sufficient to demonstrate the tool (Plan versus Report) and prove that it can deliver good results and add value.

The plan was created with the same overall format as the report. It consists of a classification of all work processes based on the coding structure. It includes duration and description. The description in this case is kept very simple but in reality it should include everything necessary for the contractor to drill the well and be able to completely replace the standing drilling instructions (SDI) that are in use now. Although there is an absolute start and ending time as well as a date, this information is not necessary for this example and can be ignored.

3.4.4. Adapting the Report

After the final creation of the plan, the report was modified. The first modification step was to correct the coding so that it would fit the plan. In future real life cases this wouldn't be necessary since the person reporting would have the code scheme which he could refer to without having to decide which code to use. All other parts of the code have been left untouched.

At this point the report and the plan are in the same form as they should be in real life after finishing the project.

3.4.5. Creating the Algorithms

After those first two steps, the plan and the report have been ready for automated analysis. Since evaluation of this type of data is a complicated process it was necessary to use Visual Basic for Applications (VBA) programming tool included in Microsoft Excel ©. The necessary algorithms are described and documented in detail in Appendix C of this thesis.

3.4.6. The Results

The data that is created during the analysis consist mainly of one thing: the difference between planned and reported time.

$$\Delta dur_{(abs)} = p - a \quad [h] \quad (3-1)$$

where

$$p \quad \text{planned time} \quad [h]$$

$$a \quad \text{actual time} \quad [h]$$

This absolute value tells how many hours the plan did not meet reality. Negative values mean that the planned time was shorter than the actual time and vice versa. Since those values are absolute, summing them up will only deliver one result, the deviation of the whole project. More meaningful results can be achieved with the relative deviation.

$$\Delta dur_{(rel)} = \frac{p-a}{p} \quad [%] \quad (3-2)$$

The relative value is independent of the duration and is a percent measure of how much the plan did not meet reality. A zero indicates no deviation while a +1, is equal to a hundred percent off area, meaning that the plan was scheduled twice as long as reality.

$$Avg. Deviation = \frac{\sum_{i=1}^n \Delta dur_{(rel)_i}}{n} \quad (3-3)$$

The average deviation sums up all those values and indicates how well the plan matched reality. But there are some issues with the average deviation one must to consider. The number will change depending on the level of detail one is looking at, action level, operation level or the whole operation. That is why it is not a good number to be used for future planning. But it can be used, if on the same level, to compare planning from different wells or different engineers. The number is an average that describes how strong the operations, or actions have been off.

A better benchmark is accuracy:

$$Accuracy = \frac{zeroCnt}{totalCnt} \quad (3-4)$$

where

zeroCnt Amount of data that equals zero

totalCnt Total amount of data

One can imagine the relative deviation data as distances from a target point. A zero indicates that the target was accurate. The accuracy describes how often the target was hit relative to the total amount of hits. Any hit not equal zero can be considered a miss. To measure the quality of the miss, the average miss parameter was created.

$$Avg. Miss = \frac{\sum_{i=1}^n |\Delta dur_{(rel \neq 0)_i}|}{n} \quad (3-5)$$

The average miss describes how far away the misses are from the target. The average miss does not differ between positive and negative values, and therefore can have a higher value than the average deviation where positive deviation can reduce the value in a negatively dominated data set. Examples of this situation are in the subsequent section “Simplified Cases”.

These numbers are for Schönkirchen 436:

Table 3-1 Benchmarks for Schönkirchen 436

	[%]		[%]
Total for:	All Actions	Total for:	All Operations
Avg Deviation:	-0.542	Avg Deviation:	-0.267
Accuracy:	0.336	Accuracy:	0.266
Avg Miss:	0.599	Avg Miss:	0.265

Although the numbers are generated from the same data there is a difference in the values between the action and the operation level. This difference can be explained by two facts: in the operation level times tend to become more global and though direct hits are improbable and small actions with high deviations and are summed up with bigger actions which lead to a reduction in the relative deviation.

For graphical visualization a deviation histograms has been created.

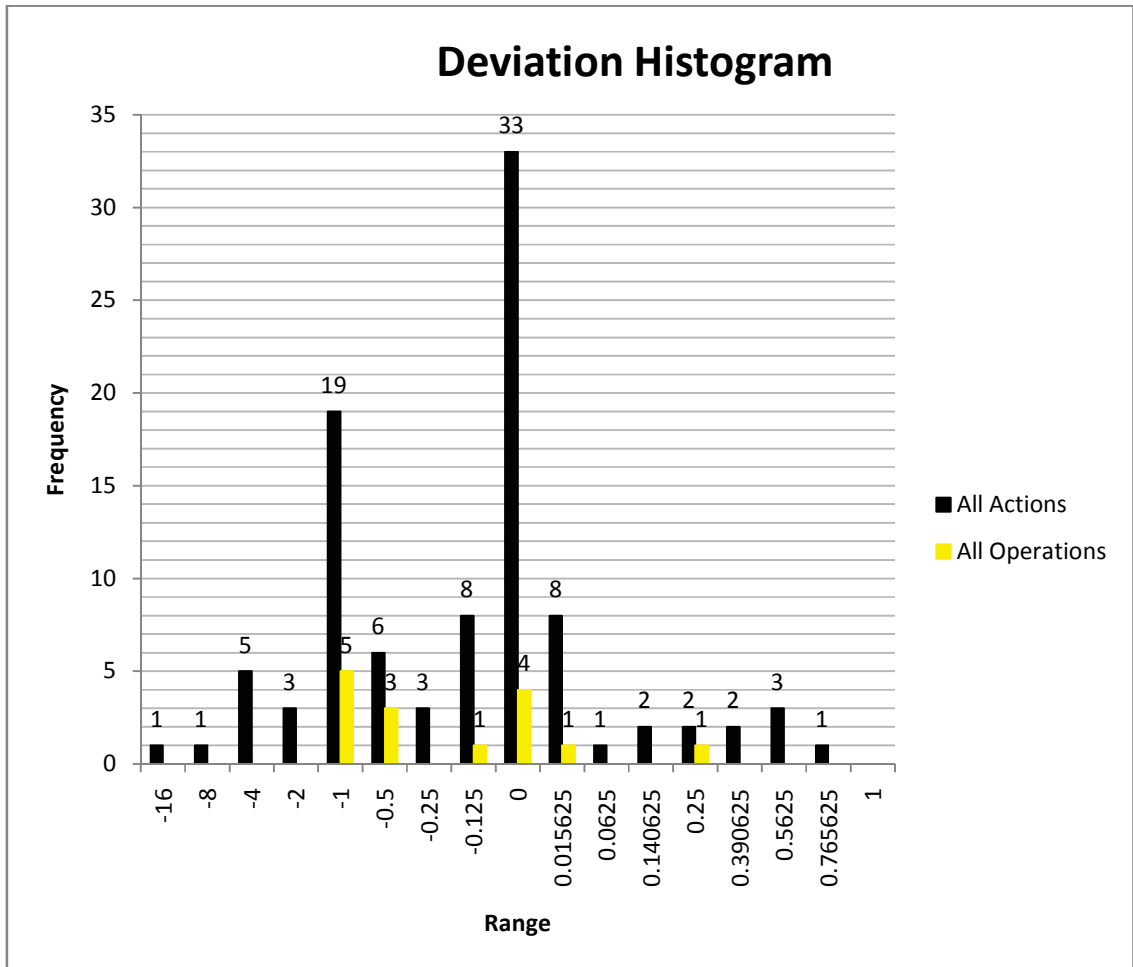


Fig. 3-2 Histogram from Schönkirchen 467

The ranges, plotted on the horizontal axis, start with the lowest number on the left. Any deviation that is lower is not listed. The lower number represents the lower limit, the higher number the upper limit of the range. In this case, the 1 above -16 represent a value that is bigger than -16 but smaller than -8. The 1 above -8 represents a single value bigger than -8 and smaller than -4 and so on. Numbers representing deviations above 0 are exactly zero.

The black bars represent the results based on an operation level of analysis while the gray bars are based on the action level. Reasons for this type of visualization are described in chapter 3.5.2 The Deviation Histogram.

3.5. Visualization

This chapter deals with the problem of visualizing the created data. Two used approaches are described.

3.5.1. Time versus Cost

When it comes to visualization the question that arises frequently is: “What are the variables that the chart should be based upon?” When it comes to drilling related chart there are two established practices: time versus depth and cost versus depth.

For the visualization of the results in this thesis the time versus depth method has been selected. The main reason is that reports as well as plans are based on this variable. However the benchmarks could also work with cost. A plan and report system that focuses on cost could be analyzed in the same way as one that is based on time. The results might provide accountants with valuable information.

The time versus depth charts in this thesis consist of three plotted times:

- Planned time
- Actual time
- Lag time

The first two are known in the oil industry and considered standard. [Reference] The third one, lag time, is plotted versus actual time and resembles the amount of hours the crew was off plan at a certain time.

3.5.2. The Deviation Histogram

The deviation Histogram can be a very good visualization tool when it comes to large data sets containing many deviations. In those cases, if the distribution is not a classical Gaussian distribution, the benchmarks themselves will give a very simplified picture. A more comprehensive insight can be achieved through the histogram.

The histogram plots the amount of events in a certain range. In this case the events are relative deviation data points, the ranges in which they occur can be chosen based on the data or on a special type of distribution comparison. For the work here the following ranges were chosen:

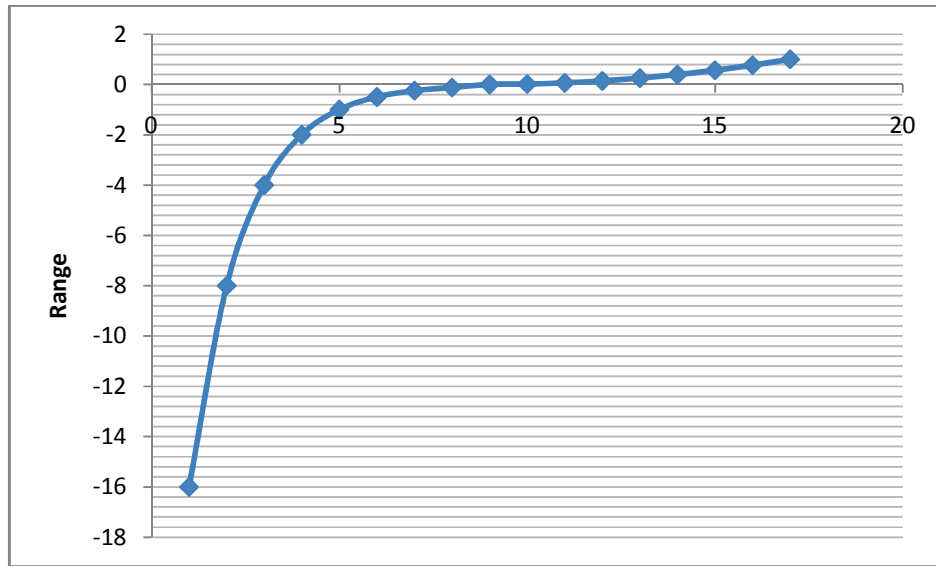


Fig. 3-3 Deviation Histogram Ranges

The ranges start with -16, meaning that deviations smaller than -16 are not plotted. Durations that are more than 16 times longer than what was planned are considered exceptions and not useful. From -16, the ranges become smaller by half until a defined number of ranges under zero are reached (in this case eight). Zero is a range itself, so all values at zero represent a direct hit (the target analogy seems to make good sense here).

The ranges above zero represent durations that have been shorter than scheduled. By definition they can't exceed one. So for defining the ranges in this area from 0 to 1 a different approach was selected. The reciprocal number of the desired steps (in this case eight again) is taken and increased by itself from step to step ($1/8, 2/8, 3/8\dots$) and multiplied with itself, leading to the same distribution scheme as in the negative range.

This approach gives a higher resolution around the target and lower resolutions further away. This accounts for the fact that there is a higher probability for a hit closer to the target. Fig. 3-3 visualizes this distribution. Note that the numbers on the horizontal axis only represent indices of the ranges. So in this case -16 is above 1, the next smaller range is above 2 and so on.

Further analysis of those histograms in the future which could, if the data quality is good, lead to much more sophisticated results. Certain histogram shapes and distributions could be used to identify certain types of errors and help answer the question: "What went wrong?" instead of "Did something go wrong?" (Thonhauser, 2008, personal communication).

3.6. Simplified Cases

After creating the benchmarks for “Schönkirchen 467” the question was: “How will the benchmark look like in other cases? Will those tools provide meaningful results with a project that had a different plan to report ratio?” Due to a lack of real life data that had not been processed, the author of this thesis decided to answer those questions with generic and simplified well plans.

Those plans are only based on operation coding; they have no intervention time, and are completely generic. The times selected for individual operation durations are fictional and don't necessarily represent a real operation. However they are sufficient to demonstrate how the benchmark numbers change for different situations.

3.6.1. Case #1, Underestimating

Table 3-2 Case #1 Input Data

	Planned			Actual		
Phase	Depth [m]	Time [h]	Time Cum [h]	Depth [m]	Time [h]	Time Cum [h]
WOS	0	24	24	0	28	28
DRL	1000	32	56	1000	36	64
CAS	1000	10	66	1000	16	80
BOP	1000	8	74	1000	16	96
DRL	2000	40	114	2000	50	146
LOG	2000	6	120	2000	14	160
CAS	2000	15	135	2000	20	180
COM	2000	10	145	2000	15	195
TST	2000	16	161	2000	24	219

In this case example, all phases of the well have been scheduled shorter than what they were reported.

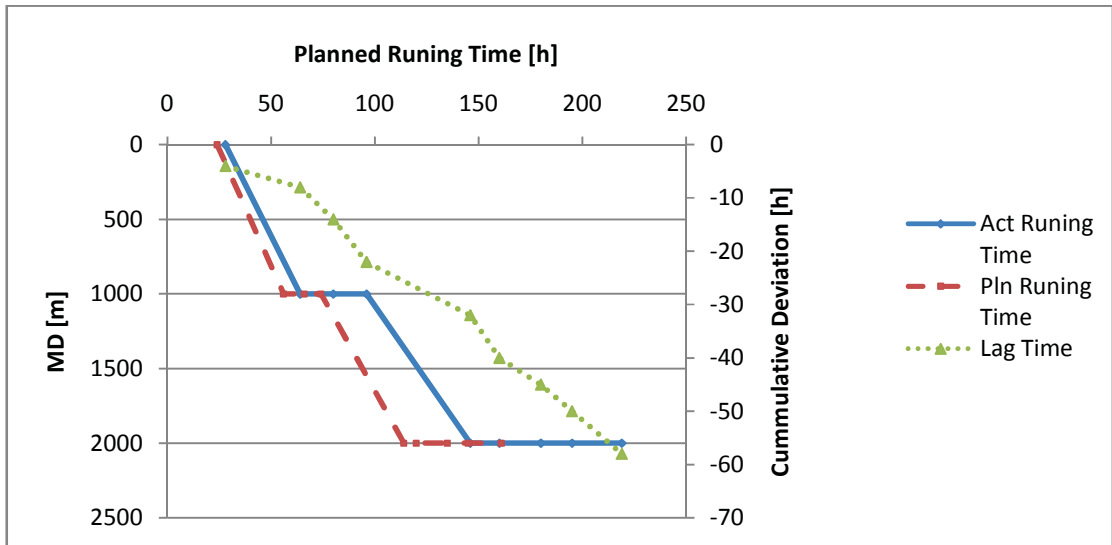


Fig. 3-4 Case #1 TVD

The TVD chart clearly indicates the deviations. Note the lag time curve that is introduced here. It resembles the lag time (the cumulative Δdur) as a function of project time. It indicates how many hours off plan (secondary, vertical axis) the project has been at a certain time (horizontal axis).

Table 3-3 Case #1 Benchmarks

Total for:	All Operations
Avg Deviation:	-0.534
Accuracy:	0
Avg Miss:	0.534

The benchmarks sum up the curves. The accuracy of zero indicates a very bad plan. A negative Avg Deviation indicates that the project took longer than what was planned. An Avg Miss that has the same absolute value as Avg Deviation what indicates that all operations have a negative deviation.

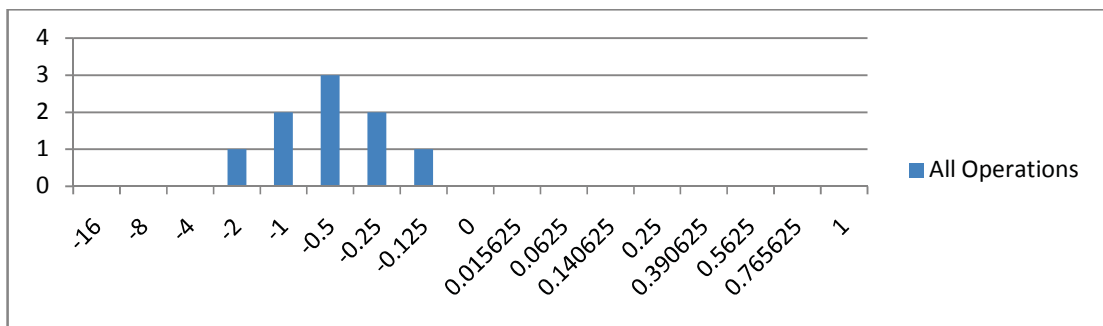


Fig. 3-5 Case #1 Deviation Histogram

The results are once again summarized in the histogram shown in Fig. 3-5 confirming the analysis developed from the benchmark numbers.

3.6.2. Case #2, Overestimating

Table 3-4 Case #2 Input Data

Phase	Planned			Actual		
	Depth [m]	Time [h]	Time Cum [h]	Depth [m]	Time [h]	Time Cum [h]
WOS	0	24	24	0	22	22
DRL	1000	32	56	1000	27	49
CAS	1000	10	66	1000	5	54
BOP	1000	8	74	1000	3	57
DRL	2000	40	114	2000	32	89
LOG	2000	6	120	2000	3	92
CAS	2000	15	135	2000	10	102
COM	2000	10	145	2000	7	109
TST	2000	16	161	2000	12	121

Case #2 is very similar to case #1 but this time all operations are assumed faster than what was planned.

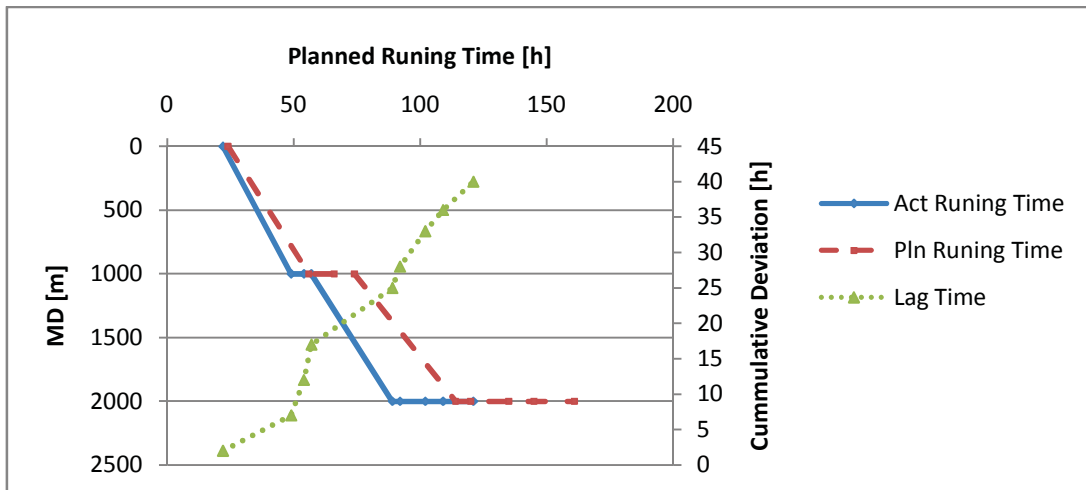


Fig. 3-6 Case #2 TVD

The faster progress is noticeable on the TVD curve. Note again the lag time curve. It is going upwards indicating a positive trend (actual shorter than planned).

Table 3-5 Case #2 Benchmarks

Total for:	All Operations
Avg Deviation:	0.333
Accuracy:	0
Avg Miss:	0.333

Once again the zero accuracy indicates a very bad plan (nothing was as it was supposed to be) but the positive Avg Deviation clearly shows that reality was shorter than planned.

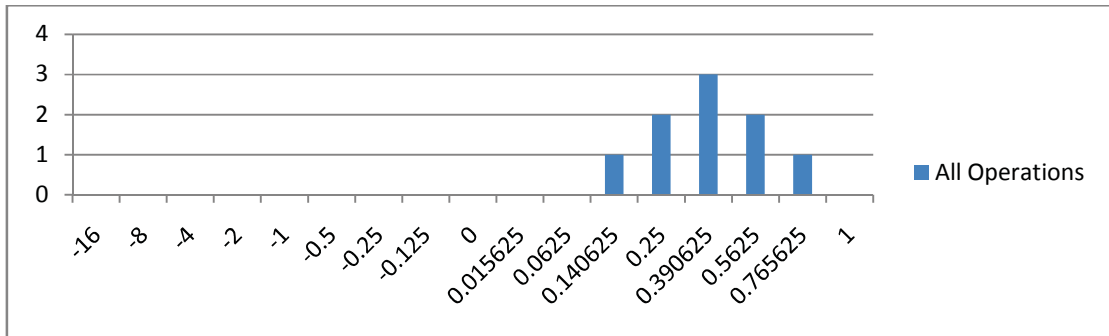


Fig. 3-7 Case #2 Deviation Histogram

The histogram in Fig. 3-7, similar as in Case #1, but this time the data points are in the positive region.

3.6.3. Case #3, Project on time with bad planning

Table 3-6 Case #3 Input Data

Phase	Planned			Actual		
	Depth [m]	Time [h]	Time Cum [h]	Depth [m]	Time [h]	Time Cum [h]
WOS	0	24	24	0	34	34
DRL	1000	32	56	1000	28	62
CAS	1000	10	66	1000	11	73
BOP	1000	8	74	1000	3	76
DRL	2000	40	114	2000	25	101
LOG	2000	6	120	2000	5	106
CAS	2000	15	135	2000	17	123
COM	2000	10	145	2000	18	141
TST	2000	16	161	2000	20	161

Case #3 is a more “Real life” example that demonstrates the value of the benchmarks. In this case, the entire project took the same time that it was scheduled. However all operations have different planned durations than reality.

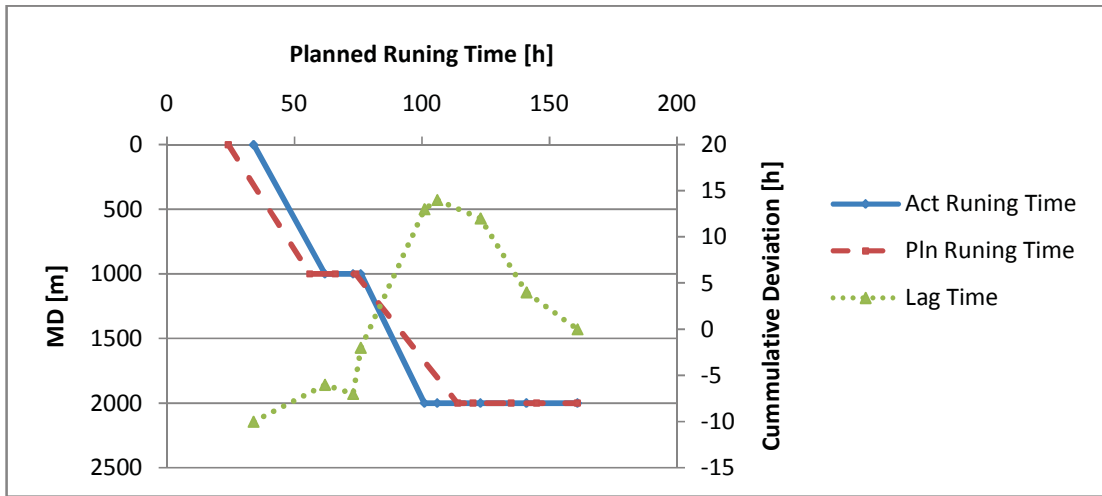


Fig. 3-8 Case #3 TVD

The TVD curve clearly indicates a common issue that could be based on motivation. If the motivation is only time, a crew that was fast with one operation might decide to take of the edge from the next one and work slower. This is indicated by the Lag Time curve, it goes up and down around the zero line. In the end, the process is on time, but the actual performance is poor.

Table 3-7 Case #3 Benchmarks

Total for:	All Operations
Avg Deviation:	-0.045
Accuracy:	0
Avg Miss:	0.332

The benchmarks are obvious, although the Avg Deviation is very low (-4%), the accuracy of zero and the Avg Miss at 33% speak a clear language: The plan was bad.

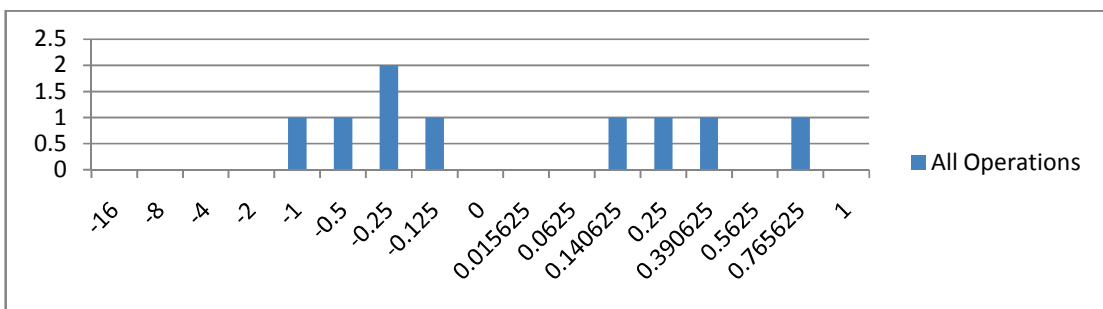


Fig. 3-9 Case #3 Deviation Histogram

The Deviation Histogram in Fig. 3-9 shows that the deviations have been positive and negative.

3.6.4. Case #4, the almost “Perfect” Project

Table 3-8 Case #4 Input Data

Phase	Planned			Actual		
	Depth [m]	Time [h]	Time Cum [h]	Depth [m]	Time [h]	Time Cum [h]
WOS	0	24	24	0	24	24
DRL	1000	32	56	1000	32	56
CAS	1000	10	66	1000	10	66
BOP	1000	8	74	1000	8	74
DRL	2000	40	114	2000	40	114
LOG	2000	6	120	2000	6	120
CAS	2000	15	135	2000	15	135
COM	2000	10	145	2000	10	145
TST	2000	16	161	2000	48	161

Case #4 summarizes an almost perfect project, the well was drilled exactly as planned but the testing took twice as long as expected.

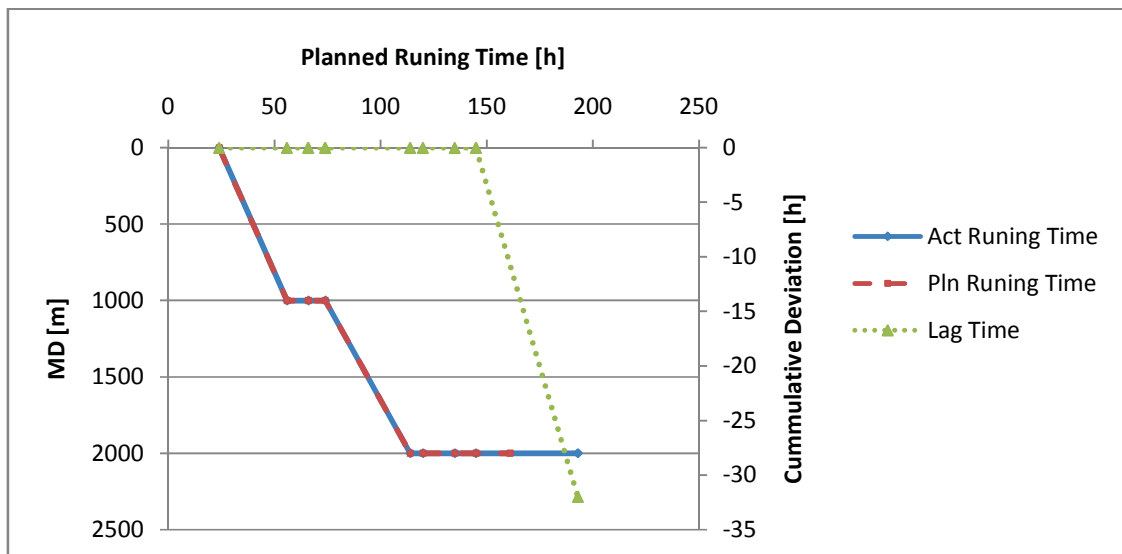


Fig. 3-10 Case #4 TVD

The TVD curves in Fig. 3-10 mach perfectly except for the last operation. The lag time is perfect during the most part of the project following the zero line and decreases during the underestimated length of the testing phase.

Table 3-9 Case #4 Benchmarks

Total for:	All Operations
Avg Deviation:	-0.222
Accuracy:	0.888
Avg Miss:	2

The benchmarks indicate this type of project by a high accuracy and a low deviation. The high Avg Miss shows that those few points that were off plan were off significantly.

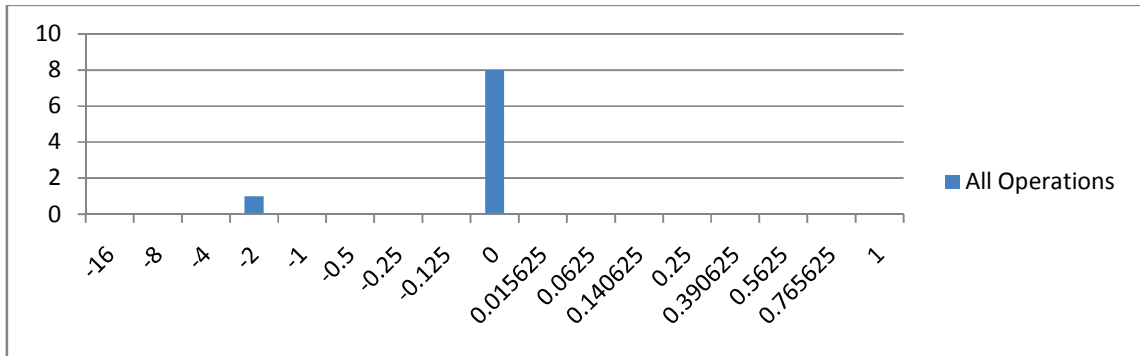


Fig. 3-11 Case #4 Deviation Histogram

The histogram in Fig. 3-11 is dominated by the huge peak in the center showing most of the plan and actual operations were predicted correctly and the small miss in the negative area.

CHAPTER 4 CONCLUSION

The three goals that were defined in the first chapter of this thesis: changing the daily drilling reports, changing the concept of standing drilling instructions and benchmarking have been addressed. However, only the third one, creation of numbers that help to evaluate the drilling process, could really be achieved. The necessary steps in changing the daily drilling report have been presented but only their application in reality will deliver the presented results. The same is valid for the concept of standing drilling instructions.

Based on the assumption made in the breakthrough and control chapter the developed benchmarks focus on the operating side of the drilling process which in this case is the person responsible for creating the plan. The numbers are a measure of the planner's ability to predict the future, in this case being able to tell how the contractor will work and how long it will take. This skill is essential, since it has an effect on cost and well okay.

The process of developing the benchmarks is very easy. However it requires good data. It has to include planned and the corresponding reported sets. It must be coded, but it does not matter which type of code is used. The codes only make a difference if one wants to select only certain data out of a larger data set. The granularity of the time steps involved isn't an issue for the generation of the number itself, however smaller time steps will provide better insight since smaller time steps can identify for example connection times. Note that the reported time steps only have to be as small as the planned steps to create the results. Efforts to reduce the reported time steps (through real time monitoring) only make sense if the planned time steps are corresponding small too. Automated well planning combined with real time monitoring could result in real breakthroughs in this area.

The numbers that are generated are universal; they can be derived from any plan and report and compared to other planners. It does not make any difference where the rig is, how big it is or who operates it. It doesn't even have to be a drilling project.

Benchmarking can take place during the project, since in the ideal case; the report is done in the same document as the plan. The first numbers can be generated right after the first report data is entered. So it can be a good way to monitor the project real time.

The deviation histogram, can give a good insight into the quality of the plan, and with further investigation and good data even reasons for deviations could be derived.

If used correctly, this benchmark system can add value to any operation analysis. It can improve performance by giving the planner an instant feedback and helping them to identify problems.

Depth deviations (For example, drilling operation that were planned to drill to 1000m but which actually drilled to 1200m) are not addressed by this approach yet. However, they can be implemented through transformation of depth into time. The necessary transformation formulas have to be developed.

Advantages

- Easy to use
- Universal results
- Focus on the planning instance
- Real-time benchmarking possible

Disadvantages

- Plan and report needed
- Focus on the planning instance
- Difficulties with depth deviations

4.1. Future Developments

Since all these values are generated from at least partial generic values, their applicability in the field is not proven yet. Any further development will therefore require testing on real data with real problems and real people. Only if the testing proves that all assumptions made here were true and the results are positive further development will make sense.

- Those future developments could be: Standards for those numbers. How good has the accuracy have to be to be considered good, tolerances for deviation calculation ions?
- Development of better numbers from the analysis.
- Increased detail of analysis by combination of automated planning and real time analysis.
- Tools to analyze the histogram, problem identification through distribution analysis and curve fitting.

Those developments could result in a higher predictability of a drilling project, its efficiency and a reduction in cost, thus in better control.

REFERENCES CITED

- G. Berhelot, V. Thiault, M. Tafflet, S. Escolano, N. El Helou, X. Jouven, O hermine, J-F. Toussaint, 2008, The Citius End: World Records Progression Announces the Completion of a Brief Ultra-Physiological Quest, PLoS ONE, February, Volume 3, Issue 2, e1552, 5 p.
- D. F. Bond, P. W. Scott, P. E. Page, T. M. Windham, 1998 , Applying Technical Limit Methodology for Step Change in Understanding and Performance, SPE Drilling & Completion, September, p 197 – 203
- R. C. Camp, 1989, Benchmarking: Milwaukee, ASQC Quality Press, 299 p.
- J. M. Juran, A. B. Godfrey, 1999, Juran's Quality Handbook 5th Edition: New York, McGraw-Hill, 1872 p.
- J. M. Juran, F. M. Gryna, Jr., 1980, Quality Planning and Analysis: New York, McGraw-Hill Book Company, 629 p.
- J. M. Juran, 1995, Managerial Breakthrough: New York, McGraw-Hill, Inc., 451 p.
- S. Knehs, OMV, 2008, personal communication
- P. F. Ostwald, 1984, Cost Estimating Second Edition: Englewood Cliffs, Prentice-Hall, 541 p.
- G. Thonhauser, W. Mathis, 2006, Automated Reporting Using Rig Sensor Data Enables Superior Drilling-Project Management, SPE, 103211, 11p.
- J. L. Thorogood, M. D. Jackson, O. H. Thorsen, 2000, Delivering World-Class Exploration Drilling – Integration of Design, Planning, and Execution, SPE, 61798, 8 p.
- H. Valdez, J. Sager, 2005, Benchmarking Drilling Performance: Achieving Excellence in MODU's Operating Practices for Deepwater Drilling, SPE/IADC, 92235, 10p.

SELECTED BIBLIOGRAPHY

Ralph L. Kliem, 2003, *The Project Manager's Emergency Kit*: Boca Raton, St. Lucie Press, 203 p.

Richard J. Schonberger, 1986, *World Class Manufacturing*: New York, The Free Press, 253 p.

Gary Hamel, C. K. Prahalad, 1994, *Competing For The Future*: Boston, Harvard Business School Press, 327 p.

APPENDIX A: PROBLEMS WHILE EVALUATING DDR FILES

A.1. Theoretical Issues

The acronyms used in this appendix are defined prior to their use. However the acronyms are also defined in the List of acronyms in the front section of this thesis.

A.1.1. Hidden actions (HA)

The most common error type has to do with multiple actions described as one action. This means two or more actions are reported in one set of data and coded with one code with one summed up time. The problem here is that actions are becoming invisible for the analyst. Only a check of the description code could possibly reveal that behind one code there is more than one action.

This issue is very common; however it could be handled with a combined plan and report. In this case the plan would already consist of all the necessary codes therefore hiding would not occur.

A.1.2. Wrong code (WC)

A simple but frequent error is the wrong code. In this case the code is used not as intended. Simple typos are sometime the reasons, but not exclusively, others could be: lack of training on the coding, unknown codes, misunderstanding of the codes. Those issues seem harmless but they make an analysis imprecise.

Once again, this error type could be handled by a combined plan and report. The person creating the plan would have a higher experience and training; thus, coding would be more accurate.

A.1.3. Dual actions (DA)

In a lot of cases, two jobs are performed at the same time. The question that arises is: which job should be used for the label. If the crew is performing maintenance operations while waiting on cement which code should be used: wait on cement (WOC) or rig service (RSE)?

Another example:

Example A-1

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
S 435	8 ½	6/21/2007	1916	13:00	13:30	0.5	RRE	POW	Circulate w/ 1000 l/min while problems w/generator

In Example A-1 two situations are described: circulation and generator problems. The labeling in this case seems like a good choice since the problems are probably more important to the analyst than the flow of mud.

Dual operations should be summed up based on the question: “What is the more important issue for the analyst?” The answer can be different from occasion to occasion so general rules would be difficult to establish.

A.1.4. Inconsistent Coding (IC)

Inconsistent coding is the use of one code set for different jobs or the description of one job through multiple code sets. This problem is caused mainly through a lack of definitions and therefore must be interpreted in the field. Those interpretations usually end up being inconsistent.

The problem caused is obvious, during analysis it is impossible to sum up specific jobs since the total time will include other operations or lack some of them. Another concern is the fact that sometimes it’s hard to discover IC issues. Only a detailed description analysis from multiple wells can reveal the problem.

The best way to solve IC is through established definitions of codes and training. However a switch to report on planned will also diminish this issue.

A.1.5. Encapsulation (CAP)

This means that one job can be theoretically described by two different codes. Good examples are “Work Pipe” (W/P), “Wiper Trip” (W/T), “Ream” (REA) and “Wash” (WAS).

Those codes are frequently used interchangeably for the same job descriptions. Jobs described as reaming are usually labeled REA but sometimes W/P and in some cases W/T or WAS. According to the authors understanding the codes should be defined the following way:

- REA: Back and forward underreaming with a reaming tool. Underreaming is the drilling work performed on preexisting drilled hole in order increase hole diameter.
- W/P: Operations needed to free a stuck pipe.
- WAS: Rotating and up - down movement of the drill assembly to eliminate/reduce restriction and clean hole.
- W/T: Should not be used as part of DRL at all. Since DRL is a run performed to drill formation, a trip that is done in order to “wipe” the hole should be reported as a “Conditioning Run” (CNR).

This definition would allow one code to be removed and the others being clearly encapsulated. The separation will allow distinguishing in the analysis between time that was spent on prevention (WAS) and time for failure correction (W/P).

A similar problem exists with all the circulation codes. “Circulate to clean hole” (C/H), “circulate to condition mud” (C/M), “circulate pill” (C/P), “circulate for sample” (C/S) and the general “circulate” (CIR). Although intended to be used only for specific purposes, in the field those codes highly interrelate with each other leading to an inconsistent report.

For example a job description like “circulate pill to condition mud” leaves open space for interpretation and errors. The question that arises is: Is the subdivision of CIR into the more specific parts really necessary?

An easier approach would be to define only one code that sums up all the circulation types. This code CIR would be in use every time the mud is being circulated without moving the drill string. This simplification only makes sense if more detailed analysis is not preformed.

A.2. Error Evaluation

This section covers various error evolutions. The error evaluations that follow consist of some general statistics as well as general information and examples. The examples are followed by one of the above defined error type codes in parenthesis and a description for example (HA) for hidden actions.

A.2.1. Rig Service (RSE) related Problems

From the 133 operations coded as RSE, 11 were found to be incorrect (8.2% error rate) all of them are listed below and are marked accordingly.

Example A-2

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Ma 626a	8 ½	7/24/2007	2025	11:15	15:30	4.3	RRE	DRW	Checked drawwork brake. Adjusted brake rim + break lever. Top drive service.

(HA) During the 4.3 hours rig repair (RRE) as well as rig service (RSE) operations were performed on the drawworks (DRW) and on the top drive system (TDS).

Example A-3

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Ma 626b	8 ½	7/27/2007	1930	12:15	12:45	0.5	RSE	OTH	Fix mast tilt

(WC) According to description the operation should be RRE.

Example A-4

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban	6	4/6/2007	2610	09:00	09:30	0.5	RSE	E/C	Service top drive, rep. one seal at top drive, changed oil & filter. Perform flow check (ok)

(WC) This is a TDS action and not E/C.

Example A-5

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 4	8 ½	3/20/2007	2087	22:00	22:30	0.5	RSE	OTH	Serviced and tested triplex pump – Accumulator unit.

(IC) The more specific action would be PMP.

Example A-6

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
KMPT1a	8 3/8	5/16/2007	3581	9:45	12:45	3	RSE	S/C	Slip and cut drilling line adjust ACS and COM Flow check = well not flowing
STR T6a	5 7/8	5/6/2007	5655.5	00:00	02:00	2	RSE	S/C	S/C. TDS service
SawanEast	8 ½	10/8/2007	3611	20:30	21:30	1	RSE	S/C	Slip and cut drilling line. Held H2S drill

(HA) The actions are hiding other actions.

Example A-7

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 4	6	4/9/2007	2835	12:30	13:00	0.5	RSE	TDS	Perform rig service, greased & re – tight wash pipe.
Haban 5	17 ½	5/17/2007	483	09:00	09:30	0.5	RSE	TDS	Safety meeting, pressure test cement line with 300 bar = o.k.
Habban 5b	8 ½	7/14/2007	2078	02:30	03:00	0.5	RSE	TDS	Perform rig service.
Habban 6	17 ½	9/10/2007	564	09:30	11:00	1.5	RSE	TDS	Perform rig service, changed fuse & starter (weak due to bouncing of rig during drilling/reaming)

(WC) The jobs are not part of TDS.

A.2.2. Rig Repair (RRE) related Problems

There are 75 operations coded as RRE. There were 11 of them wrong resulting in a 14,6% error rate.

Example A-8

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Ma 626 a	8 ½	7/24/2007	2025	11:15	15:30	4.3	RRE	DRW	Checked drawwork brake. Adjusted brake rim + brake lever. Top drive service

(WC) RSE coded as RRE.

Example A-9

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 4	6	4/15/2007	3239	16:30	17:30	1	RRE	E/C	Change controller at ELMAGCO brake

(WC) Abuse of E/C, should be DRW.

Example A-10

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 6	12 ¼	10/6/2007	1932	00:00	04:00	4	RRE	OTH	Washout on hammer union valve- stand pipe. POOH to 1463m 13 3/8" csg shoe. Repair on stand pipe. Meanwhile wiper trip to 1463m (13 3/8" csg shoe).

(HA) Hiding coding.

Example A-11

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 6	17 ½	9/26/2007	1413	13:30	18:00	4.5	RRE	R/D	Top drive failure: Rig down Columbia and CANRIG Top Drive with torque guide.

(WC) Although a rig down procedure was required the problem was TDS and not R/D.

Example A-12

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 6	17 ½	9/26/2007	1413	18:00	00:00	6.0	RRE	R/U	Assemble rig with Kelly, bushing, rathole & equipment. Static losses 5bbl/hr.
Habban 6	17 ½	9/27/2007	1461	00:00	01:30	1.5	RRE	R/U	Assemble rig with Kelly, bushing, rathole & equipment. Static losses 5bbl/hr.

(WC) Once again, the problem was some sort of repair (probably TDS) and not an R/U issue.

Example A-13

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
KMPT1a	12 ¼	5/1/2007	3190	21:30	00:00	2.5	RRE	REA	Ream and wash down from 3017m to 3090m, FR=2800 l/min, PP=275bar, RPM=90-120, Torque= 19-34kNm

(WC) Reaming is not part of repair (RRE).

Example A-14

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 4	6	4/6/2007	2610	23:00	00:00	1	RRE	TDS	RIH from 2091m to 2600m. Wash down to 2610m.

(WC) Both codes (operation and action) are out of place.

A.2.3. Health Safety and Environment (HSE) related Problems

The most common error with health safety and environment” (HSE) related codes is based on the weak/none definition of drills (DRI) and training (TRA). The result is an interchangeable use of those codes for the some jobs.

The possible solution for this problem is the elimination of one of the codes. Although DRI is more frequently used than TRA, there is a tendency of DRI being misused, as a drilling code. This would leave three actions for the operation HSE: TRA, S/M and INV. Whereas S/M would represent any means to educate the crew on safety matters in a theoretical manner, TRA would relate to all practical, hands-on, education.

The code INV has never been used with HSE and it is questionable whether it is necessary in this context.

Example A-15

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
KMPT1	8 3/8	19/03/2007	4161	17:00	20:30	3.5	HSE	BHA	Lay down schlumberger tools.

(WC) Wrong operation code

Example A-16

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Ma	12 ¼	7/11/2007	658	21:30	22:00	0.5	HSE	TRA	Fire Drill

(WC) This example shows the frequent mislabeling of drills as training. Actually all actions labeled TRA turn out to be drills. This implies a redundancy in the labels and confusion about their usage.

Example A-17

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
KMPT1	8 3/8	25/03/2007	1400	8:30	9:00	0.5	HSE	S/M	Safety meeting, pressure test cement line with 300 bar = o.k.
KMPT1	8 3/8	25/03/2007	1400	15:00	15:15	0.25	HSE	S/M	Safety meeting, pressure test cement line with 300 bar = o.k.

(HA) Have the operations been safety meetings or pressure tests or maybe both? The coding and the descriptions don't answer these questions.

A.2.4. Work on Surface (WSF) related Problems

WSF codes operations have a very high error rate, out of 156 codes 37 were incorrect (42.1% error). The main cause is that WSF is used for jobs that happen on the surface but are

actually related to a different kind of operation. BHA handling is work done on surface, for instance, but its purpose usually is to drill or ream. So the operation that should be used is a different one.

The problem is caused by a weak enforcement of pairing certain actions with certain operations. A flexible system might end up being confusing, which is the case here.

Example A-18

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
STR T 6a	12	3/15/2007	4737	00:00	00:05	5.0	WSF	P/T	Pressure test surface lines, choke manifold @ 500 bar, OK, L/D test equipment
STR T6a	8 3/8	3/28/2007	4919	05:30	08:15	2.8	WSF	OTH	P/T Select Tester @ 600 bar, tight. P/T Omni valve @ 600 bar, not tight

(WC, IC) The same pressure testing procedure is once referred to correctly as P/T and once misleading as OTH.

Example A-19

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
STR T6a	8 3/8	4/2/2007	4919	06:00	13:00	7	WSF	INV	POOH, L/D test string. Solids accumulated above test valve (probe to lab). Test valve never opened.

(HA) The investigation is probably only a very small part of the reported 7h.

Example A-20

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
HL	12 ¼	9/13/2007	421	09:30	10:00	0.5	WSF	OPH	Cleaned Rig Floor

(WC) This is improper labeling. The best fit would be OTH. It is probably a typo. Except for this case, the action OPH (open hole) is never used again and should be considered redundant.

Example A-21

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Obsu 3	12	10/22/2007	377	07:00	07:30	0.5	WSF	S/M	Safety Meeting
Schö 436	17 ½	11/19/2008	92	11:00	11:00	0	WSF	S/M	Safety Meeting

(IC, WC)The action S/M should always be part of HSE and not as in this case WSF.

Example A-22

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Spa 21	8 ½	8/23/2007	1951	10:30	21:00	10.05	WSF	L/D	L/D 5" DP, transfer 5"DP to Prottes. Fil up annular 9 5/8" x7" w/ 11101 mud

(HA) One action covers up a whole set of performed jobs. (Note that the action code L/D is not part of the coding system anymore, but is still in use.) WSF should not be used for any actions that have to do with tripping or drilling.

Example A-23

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
HL 74	8 ½	9/30/2007	1780	07:00	10:30	3.5	WSF	OTH	L/D 5" DP

(HA) Similar problem as above: Use of OTH in WSF in place of a drilling or a tripping action.

Example A-24

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Schö 434	8 ½	4/30/2007	494	21:00	00:00	3	WSF	P/H	RIH 6 ¾" DC + HWDP + Pick up 5" DP and Rack back DP + HWDP
Miano 07/ST	8 ½	7/16/2007	3429	11:00	12:00	1	WSF	P/H	POOH and L/D landing joint and Tbg. Hanger

(WC) WSF with P/H should only be used for jobs that are not directly associated with a run into the hole, like rearranging and pick up. In this case a drilling or run operation should be used in combination with BHA.

Example A-25

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
STR T6a	8 3/8	3/27/2007	4919	3:30	6:00	6.00	WSF	P/H	Dummy run to place the SSV correctly in the BOP

(WC) This is not a pipe handling action.

Example A-26

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
HL 74	12 ¼	9/13/2007	421	13:00	16:30	3.5	WSF	P/J	R/U Weatherford

(WC) A rig up R/U is not a prepare job P/J action. Clear definition is needed for what is what. Besides that the Weatherford unit is needed exclusively for a cementing job. CEM would be the right operation code.

Example A-27

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Tajjal-1	12 ¼	4/7/2007	3267	20:30	22:00	1.5	WSF	P/T	R/U CMT lines. Held PJSM, Dowell flush the surface line, P/T 500/4200 Psi, OK.

(HA) P/T was only a part of the Job. An R/U and an HSE Operations are hidden.

Example A-28

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 6	20	9/4/2007	230	19:00	22:30	3.5	WSF	R/D	Break out and lay down Kelly, problem breaking Kelly connections.

(WC) The action should be considered BHA and not R/D and part of a run operation and not WSF.

Example A-29

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 5	26	5/9/2007	80	07:30	08:30	1	WSF	R/U	Stop Hammer Drilling to extendet(sic!) the capacity of Air compressor. Pull Hammer bit to 70k
Habban 5	26	15/5/2007	230	15:30	00:00	1	WSF	R/U	Break out Kelly, L/D Kelly & rat hole, R/U torque guide & top drive. Install automatic V- door. Preform top job cement, pump 18bbl of cement in the annulus between 30" conductor and 20" csg

(WC, HA) Wrong usage of R/U is present in the first case, and hidden operations in the second.

Example A-30

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 5b	8 ½	7/26/2006	2388	02:00	03:30	1.5	WSF	TRI	RIH w/ 6 1/2" DC, 5"HWDP & 15td x 5" G-a05 DP

(WC) Both operation and action don't match the job description.

Example A-31

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Schö 436	12 ¼	11/20/2007	1390	09:00	10:45	1.75	WSF	W/E	R/U weatherfort(sic!) equipment
Schö 436	12 ¼	12/1/2007	1390	03:45	04:15	0.5	WSF	W/E	R/U cementing head
Spa 21	8 ½	8/14/2007	931	22:30	00:00	1.5	WSF	W/E	POOH to 912m. Repair washpipe.

(WC) These are examples of wrong labeling.

A.2.1. Drilling (DRL) related Problems

Wrong codes are relatively rare among DRL operations (163 out of 2097 Codes, 7.7%), probably because of the high frequency in usage and therefore a better understanding of what the codes mean. The problematic issues are usually related to weak encapsulation or collision.

Example A-32

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Bo 201	8 ½	10/31/2007	477	21:00	00:00	3	DRL	BHA	Pick up & make up 8 ½” BHA
Bo 201	8 ½	11/1/2007	760	00:00	04:30	4.5	DRL	BHA	M/U BHA & RIH to 440 m

(WC) The first code is correct; however the operation is reported after midnight as BHA although TRI is taking place.

Example A-33

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
S 433	8 ½	5/25/2007	549	11:45	15:00	3.25	DRL	BHA	Load sources. RIH HWDP, P/U jar & 19 joints 5” DP.

(WC) Running in DP should not be labeled as BHA.

Example A-34

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Ma 626	12 ¼	7/9/2007	658	00:00	03:00	3	DRL	C/H	POOH, clean DC, Stabilizer+ Rock Bit

(WC) Mislabeling, C/H instead of BHA

Example A-35

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 4	26	2/11/2007	83	12:00	13:00	1	DRL	C/H	Sweaped hole, bulled back to 17m, installed 26” string stabilizer
Habban 5b	6	8/4/2007	2700	02:00	04:00	2	DRL	C/H	Take MWD survey, Pump LV-HV pill, circulate btms up for sample
Habban 6	6	11/8/2007	2621	12:00	13:30	1.5	DRL	CIR	Circ b/u, take MWD survey, hold flow check (ok)

(HA) Those are typical hiding examples.

Most errors with DRL and C/H, C/M, C/S, C/P and CIR come from the fact that those actions are not distinguished clearly enough on the rig:

Example A-36

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
STR T6a	5 7/8	5/21/2007	6013	15:00	18:15	3	DRL	C/H	Pump 6 m ³ hi vis. Pill for hole cleaning & C/H. Backreaming not possible due to high Tq.

(CAP) Is this C/H or C/P?

Example A-37

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 5a	17 ½	6/5/2007	1145	16:00	17:00	1	DRL	C/M	Pump hi vis pill, circ. To condition mud.

(CAP) Is this C/M or C/P?

Example A-38

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 5b	12 ¼	7/7/2007	1430	21:00	00:00	3	DRL	C/T	Check trip to 1150m. Back reaming from 1411m-1327m, 1267m-1210m. Clear tight sports
Habban 5b	12 ¼	7/8/2007	1500	00:00	02:00	2	DRL	C/T	Check trip 1150m. Wash down from 1386m to 1430.

(WC) An example of misinterpretation of C/T as a check trip.

Example A-39

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
KMPT1a	12 ¼	4/27/2007	3018	9:30	10:15	0.45	DRL	D/F	Troubles with MWD & Gamma ray tool, loss puls(sic!) signal, try to restart signal from MWD & Gamma ray, worked with different pompe (sic!) pressure and RPM no results

(WC) A proper coding is hard to find in this case. INV could be used. D/F is not the best choice.

Example A-40

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
KMPT1a	12 ¼	4/30/2007	3190	00:00	4:30	4.5	DRL	D/F	Ream back to 2389, FR=2800l/min, SPP=270bar, RPM=70
Spa 21	6	8/29/2007	2049	00:00	02:00	2	DRL	D/F	POOH
SawanEast 1	17 ½	9/16/2007	1275	22:00	22:30	0.5	DRL	D/F	Observed dynamic losses 35 bph, no static losses.

(WC) First action should be REA, second TRI and third F/C.

Example A-41

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
HL 74	8 ½	9/22/2007	1560	00:00	00:15	0.25	DRL	DRI	Pit drill.
Habban 6	8 ½	10/24/2007	2109	07:30	00:00	16.5	DRL	DRI	Drill from 2070 t 2072m, slide & drill from 2072m to 2109m (61.5% Sliding)

(WC) First operation should be HSE, second is a typo. It should be D/F.

Example A-42

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
S 434a	8 ½	5/11/2007	1593	18:30	22:00	3.5	DRL	L/D	L/D BHA

(WC) L/D is not part of the coding anymore, however it is still in use. In most cases it can be replaced by BHA.

Example A-43

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 6	8 ½	10/30/2007	2318	15:00	16:30	1.5	DRL	OTH	M/U bit & scraper (hold u w/bit on wear bushing) pull wear bushing, dress of & re-install same
Habban 6	8 ½	10/15/2007	2047	10:00	11:30	1.5	DRL	OTH	Tag closing plug @1601m. Drill out closing plug, opening bomb & DV seat.

(WC) Action OTH is used instead of existing available codes.

Example A-44

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 4	6	4/4/2007	2530	09:00	10:30	1.5	DRL	P/H	POOH to csg shoe (slow to avoid swabbing)
Sawan-11	17 ½	11/6/2007	373	07:00	08:30	1.5	DRL	P/H	Break & L/D 9 ½” DC, stas & bit. Retrieve Totco, inclination @ 372m 0.1 degree

(CAP) BHA or TRI could be always used instead of P/H. Similar situation with P/J.

Example A-45

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 6	6	11/17/2007	2854	01:30	02:30	2.3	DRL	P/T	Perfrom formation injection test w/ SLB

(WC) P/T used instead of FIT.

Example A-46

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
KMPT1a	12 ¼	5/1/2007	3190	0:00	2:00	2.00	DRL	REA	RIH, HWDP and DP to 988m. Ream and wash down from 988m to 1034m,
STR T6a	5 7/8	5/9/2007	5682	15:00	17:30	2.5	DRL	REA	POOH dry to 5590m, overpull 20t, ream up to 5400m @ FR ...

(HA) Part of both actions is TRI, but it only shows up in the description.

Example A-47

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
SawanEast 1	8 ½	10/5/2007	3428	00:30	01:00	0.5	DRL	TDS	Service (sic!) TD

(WC) TDS service should not be part of a DRL operation.

Example A-48

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
HL 74	8 ½	9/17/2007	689	22:45	23:00	0.25	DRL	TRA	Pit drill
KMPT1a	12 ¼	4/12/2007	1602	18:00	18:15	0.25	DRL	TRA	POOH to 866m, pit drill, observe well via possum belly tank.
Habban 4	12 ¼	3/11/2007	29	2020	11:30	2.5	DRL	TRA	RIH Directional assembly with 8 1/4 " DCs and L/D

(WC) In the first case TRA was used correctly but with the wrong operation. The second case was labeled wrong.

Example A-49

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Bo 201	8 ½	11/7/2007	1756	16:30	18:30	2	DRL	TRI	Lay dwn (sic!) NMDC, MWD, Motor, Stab & pony. (18:00 Bit ART, End of Run)
S 435	8 ½	6/17/2007	1007	21:00	00:00	3	DRL	TRI	Ream back to 595m. O/p 10 to
Habban 4	6	4/22/2007	3285	21:00	22:00	1	DRL	TRI	Flow check well, pump out from 3285m to 3118m.

(HA) A typical mistake with DRL and TRI is that BHA or F/C or REA get covered up.

A.2.2. Milling Run (MIL) related Problems

Milling operations were used 12 times incorrectly out of a total of 80 (15% error). The main source of error: coding BHA as TRI. This is an encapsulation issue.

Example A-50

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
STR T6a	5 7/8	5/27/2007	6013	03:45	10:15	6.5	MIL	L/D	POOH to 4210m incl. L/D 5 1/2" DP.
STR T6a	5 7/8	5/28/2007	6013	00:00	00:30	0.5	MIL	L/D	Break out & L/D scraper & dressing mill assembly.

(WC) L/D is used instead of the correct TRI in the first case and BHA in the second.

Example A-51

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Miano-07ST	6	8/16/2007	3900	04:30	07:00	2.5	MIL	TRI	POOH F/181.72m T/surface, L/D scraper and polish mill.

(HA) A typical error, the BHA procedures are obscured by a TRI code.

A.2.3. Coring (COR) related Problems

13 out of 73 Codes are incorrect (17.8 error rate). A frequent problem during coring operations with the coding is that the core recover procedure and the L/D of the coring assembly is not referred to as BHA but is covered as TRI.

Example A-52

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
STR T6a	8 3/8	3/22/2007	4873,5	22:30	00:00	1.5	COR	OTH	Take out core 75cm. L/D coring equipment.
Tajjal-1	8 1/2	4/23/2007	3680	17:30	18:00	0.5	COR	OTH	Drop ball, circ, wait for pressure build up, pressure increase F/300 Psi, to 330 Psi

(WC) OTH is used twice instead of a better fitting code, like BHA and CIR.

Example A-53

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
STR T6a	8 3/8	3/22/2007	4873.5	08:45	15:00	6.25	COR	TRI	POOH to 2303m. Overpull to 15t at 4828m. Wash OOH with 20t Overpull from 4820m to 4810m.

(HA) The overpull and the resulting W/P action are all covered up 6,25 h of TRI.

A.2.4. Fishing (FIS) related Problems

FIS faces the same problems when it comes to coding like COR and MIL: The BHA time is frequently combined with the trip, and there is no clarity when it comes to label the fishing job itself. It is referred to as W/P, W/E, and OTH. In the ideal case a FIS procedure would look like this: BHA – TRI – (CIR) - W/P – TRI – BHA. This is unfortunately not the case in the available codes.

Example A-54

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Schö 436	17 ½	11/22/2007	460	12:15	15:45	6.25	FIS	TRI	Make up fishing tool % RIH to 460 m.
Miano-11	6	6/22/2007	3330	20:00	22:30	2.5	FIS	L/D	Break & L/D fishing assy, and fish (4 1/8" DC).

Hidden BHA actions

A.2.1. Conditioning Runs (CNR) related Problems

In this case, 68 out of 249 CNR codes are incorrect (27.3% error rate). A big problem with CNR runs is that they are seldom defined as a whole but they start as DRI operations and over time are converted to CNR. The result is an incomplete coding for both CNR and DRI. An approach to solve this problem would be to define a priority in coding. A run designed primarily to drill should be consistently coded as DRI even when there is conditioning action involved.

Other frequent errors are once again the missing BHA definitions at the beginning and end of the run.

Example A-55

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Schö 436	12 ¼	11/29/2007	1390	02:00	03:00	1	DRL	D/F	Drill formation to 1390m
Schö 436	12 ¼	11/29/2007	1390	03:00	04:00	1	DRL	C/S	Circulate for sample.
Schö 436	12 ¼	11/29/2007	1390	04:00	11:30	7.5	CNR	REA	Backreaming to 800m

(WC) The DRL runs suddenly turns into CNR.

Example A-56

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
STR T6a	5 7/8	5/24/2007	6013	03:45	10:30	6.8	CNR	TRI	P/U & M/U 9 5/8" scraper, RIH to 1370m.

(HC) A covered BHA procedure.

A.2.2. Equipment runs (EQU) related Problems

In this case, 15 out of 57 EQU codes are incorrect (26.3% error rate). The EQU runs are unfortunately rare and inconsistent. This means that there are multiple options for describing the same task. Setting a packer is labeled as W/E or OTH. Similar situation exists with P/J and R/U. Once again a clear definition of the codes is needed as well as a guide with action codes can and can not be used with EQU.

Example A-57

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Miano-07ST	6	8/21/2007	3900	19:00	21:00	2	EQU	TRI	P/U and RIH dummy 2 7/8" guns (12m).
Miano-07ST	6	8/21/2007	3900	21:00	21:30	0.5	EQU	TRI	Perforation assy easily passed through 4.5 liner top, Tag L/C @ 3872m.
Miano-07ST	6	8/21/2007	3900	21:30	23:00	1.5	EQU	TRI	POOH T/surface. R/D Wireline.

(WC) Perforation operation coded as EQU.

Example A-58

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Miano-07ST	8 ½	7/15/2007	3429	16:00	17:00	1	EQU	P/J	R/U Slickline lubricator.

(CAP) Collision between P/J and R/U.

A.2.3. Perforation (PER) related Problems

In this case, 6 out of 30 PER codes are incorrect (20% error rate). The problems are very similar to EQU; no unified coding for same procedures and short operations hidden in long ones.

Example A-59

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Swann-11	6	12/14/2007	3415	03:30	04:30	1	PER	W/E	POOH W/L after Run #1 Perforation guns. Physically saw guns fired.

(WC) TRI would be more accurate.

A.2.4. Completion (COM) related Problems

28 out of 154 COM codes are incorrect (18.1% error rate). COM is very badly defined. When going through the codes one has the feeling that most coding is just random. There are unfortunately no standards on what kind of operations should be addressed as COM. One can find packer setting procedures (which are also found in EQU), scraper runs (part of CNR) and operations on wireline and on tubing. Same is true for actions in COM, they range from TRI, RUN, OTH, WRL for the same type of operations.

Example A-60

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 4	6	04/29/2007	3285	02:00	16:30	0.5	COM	RUN	Run 2 7/8" Tenaris completion tbg. Pressure test 2 7/8" Completion string all 500m to 3500psi w/SLB
Habban 6	6	12/092007	3154	03:00	07:00	4.0	COM	TRI	RIH with Scraper fr. 1216m to 3100m.

(IC) Different coding for running in hole.

A.2.5. Wait on Cement (WOC) related Problems

In this case, 18 out of 100 WOC codes are incorrect (18% error rate). There are questions associated with WOC that are not easy to answer. Since this time is used for operations that are usually not directly related to the cementing itself but to operations such as, pipe handling or all kinds of surface service work, a wide variety of codes can be used. A TDS service performed while WOC can be addressed to as RSE – TDS or WOC – TDS or as it is done frequently WOC – OTH. The last option is lacking information. The first two allow it; however, one has to be aware of the fact that two coding sets are available. It is not really necessary to know that the service took place during a WOC. Coding TDS as part of RSE on the other hand allows a summation of the overall time spent on rig service. That's why the first option should be preferred.

Example A-61

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Bo 201	12 ¼	10/30/2007	477	13:15	17:00	3.75	WOC	OTH	Rig down Schlumberger cementing equipment clean cellar & equipment.

(DA) The primary action is rigging down. Cleaning, correctly reported as OTH, is of minor importance.

Example A-62

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Habban 5b	12 ¼	07/01/2007	1145	10:00	14:30	4.5	WOC	OTH	WOC. Tag fluid level @27m. Fill u hole with 16bbl-well static.

(WC) CIR would be the more accurate code.

A.2.6. Cementing (CEM) related Problems

In this case, 47 out of 585 codes are wrong (8% error). A lot of operations end up being coded as CEM although they aren't. Washing and scraper runs that are performed ahead of the cementing, bridge plug runs, and tripping that is not directly associated with cementing. Since the CEM procedures are performed in multiple ways, it is difficult to define an order of codes that would universally describe the work process best. That's why, exact definitions are needed. What is CEM? When does it start and end? And what are the actions involved?

Example A-63

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
KMPT1	8 3/8	03/22/2007	2900	00:15	06:00	5.75	CEM	L/D	POOH and lay down 45 x 5" DP (MD=2500m) for black light inspection, use rig tong for break out, break out torque = 40 - 75kNM, observe borehole via possum belly tank.

(WC) Should not be coded as CEM.

A.2.7. Casing Run (CAS) related Problems

There are 103 out of 416 CAS codes wrong (24.7% error rate). The majority of errors with CAS consist of including CEM action into CAS. Since the process is very much interrelated and a boundary is not defined, those errors are understandable but they make a meaningful analysis very difficult.

Example A-64

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
Schö 434	12 1/4	04/29/2007	494	09:15	10:45	1.5	CAS	CEC	Cement Casing with 39m ³ Lead Cement & Tail Cement. ...

(WC) The cementing should be coded as CEM not as CAS.

A.2.8. Logging Run (LOG) related Problems

43 out of 395 LOG codes are incorrect (10,8% error). The codes available to describe logging operations are LOG for the Operation and W/L for wire line logging as well as LWD for logging while drilling and TCL Tubing conveyed logging run. LWD however seems to be more useful with DRL as Operation than LOG. W/L relates to wire line and excludes logging performed on drill pipe without the intent to drill formation.

Example A-65

Well	Section	Date	Depth at Mid	From	To hr	hr	Op	Act	Description
STR T6a	12	03/11/2007	4737	02:30	06:15	3.8	LOG	TRT	Logging run #2.: HALS-TLD-SP-GR-LEHT. Problems with cable-tool connection
Habban 4	8 ½	03/19/.2007	2087	00:30	05:00	4.5	LOG	RUN	1st Logging run: DIP-ORIT-GR

(IC) Different coding used for the same process.

APPENDIX B: NEW SUGGESTED CODDING STRUCTURE

B.1. Introduction

This appendix is designed to serve as an example of a possible coding structure. It is based on OMVs coding but modified in order to overcome some of the issues mentioned in appendix A. Most changes were simplifications of the code by the reduction of available action and operation codes.

The coding presented here is structured in flow charts. They present an overview of the available coding, definitions and the required work flow of actions that guarantees that each operation consists of its crucial actions. For example that a drilling run is coded with the necessary bottom hole assembly phase.

Categories for the action coding have been already introduced in the main section of the thesis and include:

- Required actions
- Optional actions
- Related actions
- Intervention actions

All actions are sorted under the corresponding category. Exceptions are unstructured operations that usually consist only of few actions that don't require a certain work flow.

Note that only the required actions are in a certain order, the other types of actions are free to use throughout the operation (technical considerations being the limiting factor).

Definitions are not ultimate. They are important but they can be changed and adapted for different uses or special needs in the field. The changing however should be done by one authority responsible for the coding.

B.2. Drilling Coding

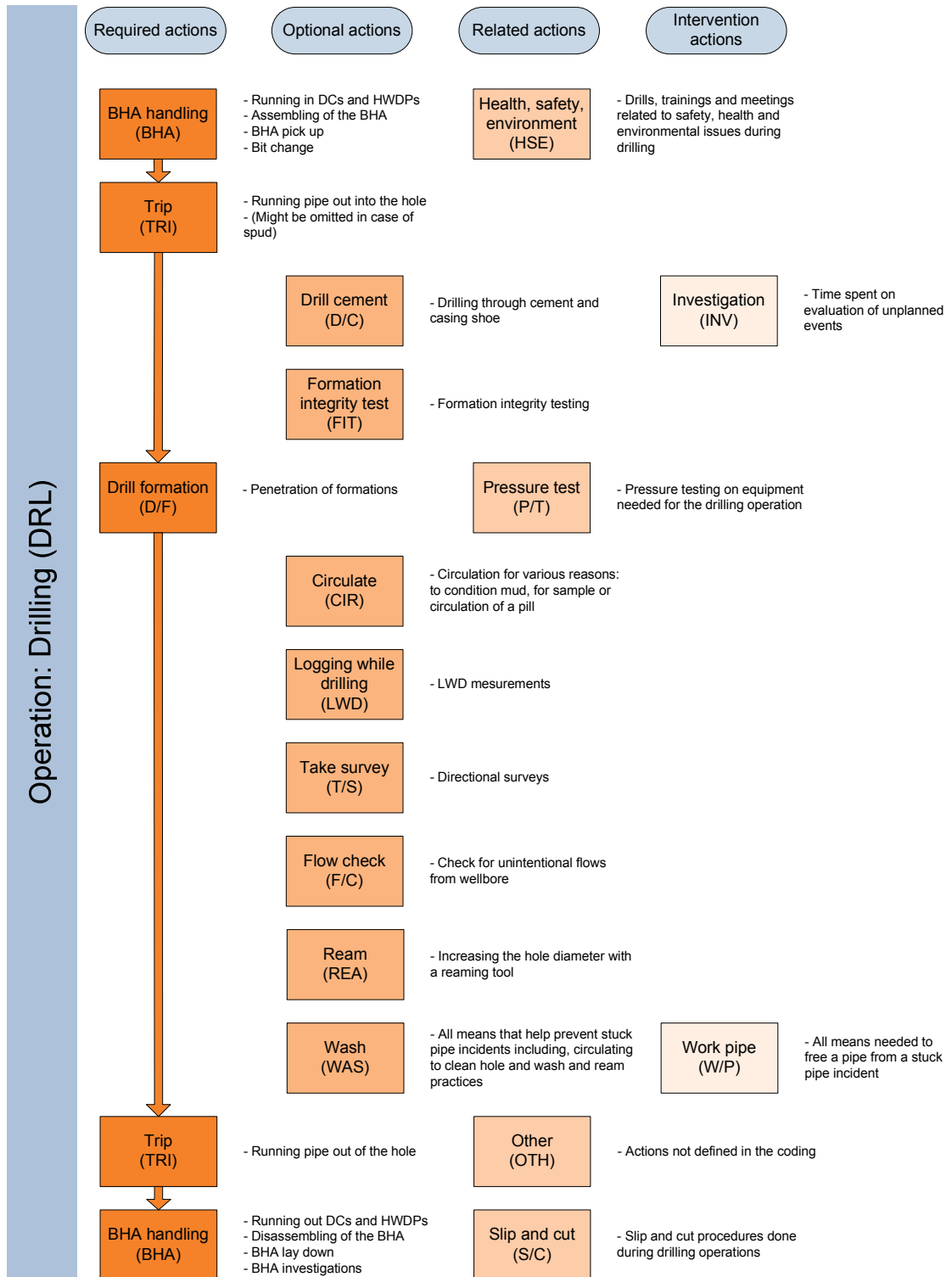


Fig. B-1 Drilling related coding structure

The most frequently used operation coding is drilling (DRL). Fig. B-1 sums up all actions to describe DRL operations. WAS and W/P are defined in a way that an optimum level of perfection (0) (see main section of the thesis for more detail) analysis can be performed.

The required actions include the opening and ending BHA action, a drill formation (D/F) action and the tripping (tri) action. Although TRI is used once in the scheme it can and should be used more than once to best resemble reality for every operation.

B.3. Casing Coding

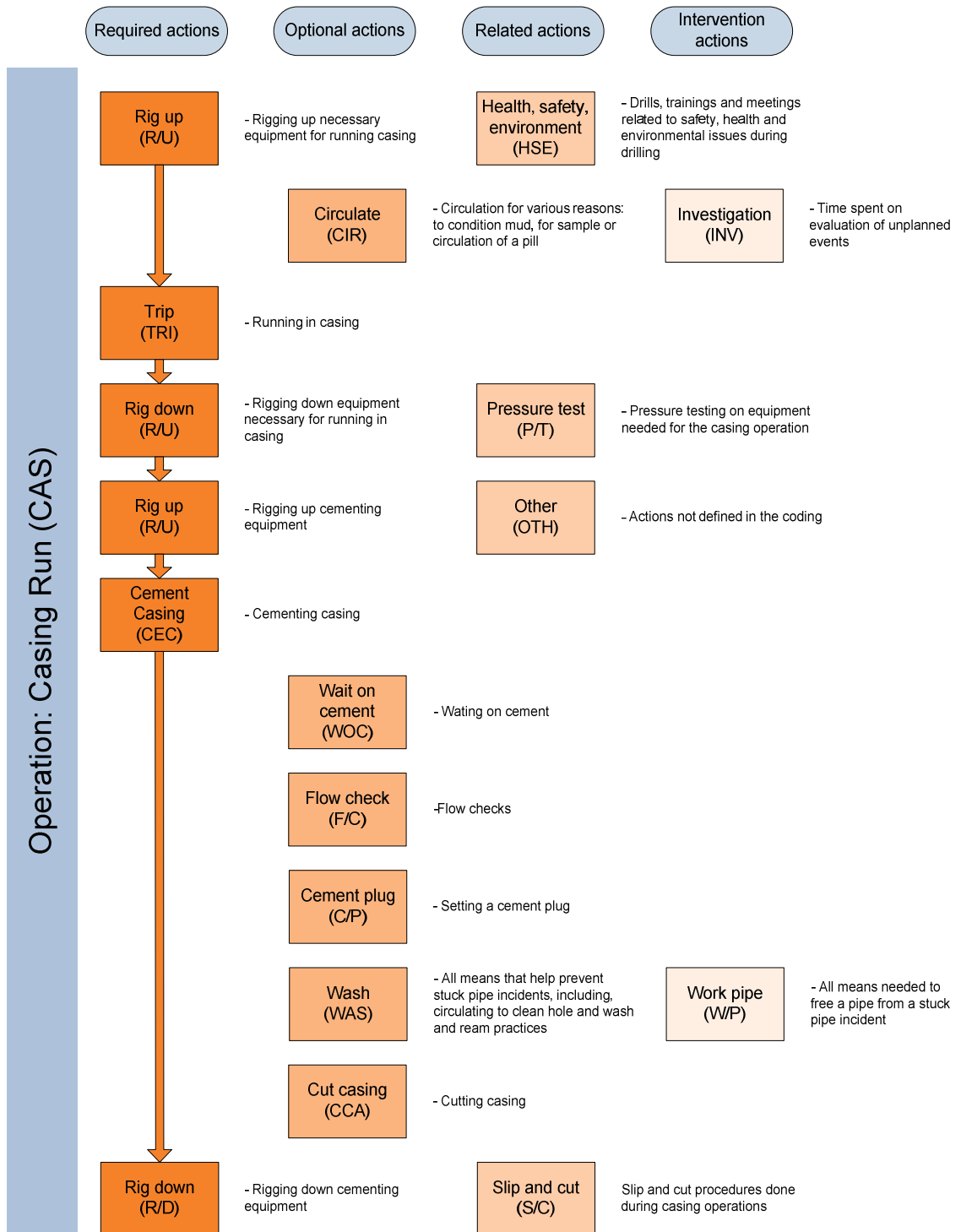


Fig. B-2 Casing related coding structure

The suggested operation code (CAS) is different from the one in OMV's old coding. It includes the casing running process the cementing process and the cement waiting time. This change was made because of the undefined border line between casing and cementing leading to wrongly paired action and operation codes.

B.4. Rig Up/Down Type Run coding

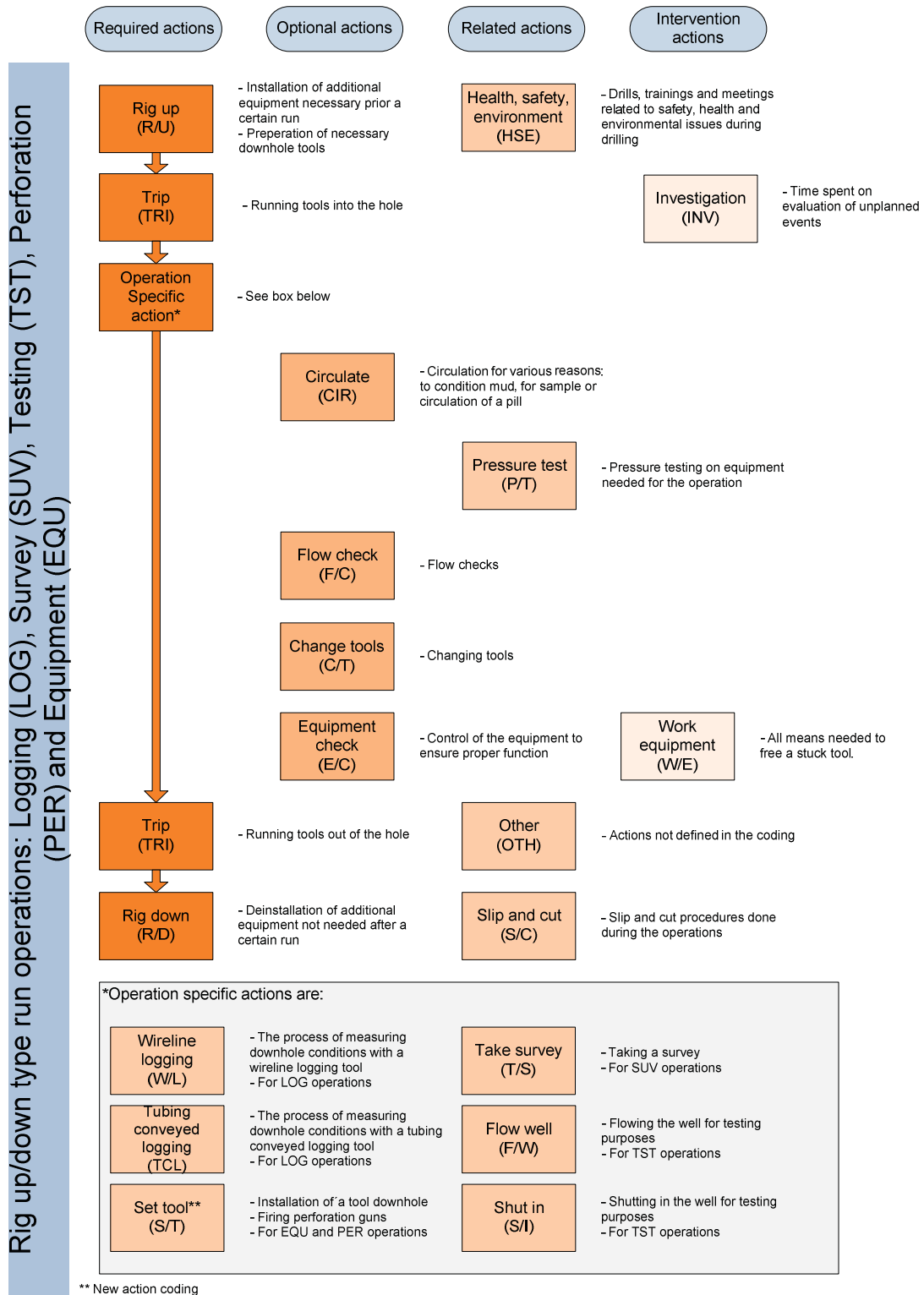


Fig. B-3 Rig up/down type operations coding structure

Rig up/down type run operations are all runs that start with a rig up action and end with a rig down action. They include logging runs (LOG), equipment runs (EQU), perforation runs (PER), survey runs (SUV) and testing runs (TST).

The required actions are rigging up and down, tripping and the performance of the operations specific action. Those specific actions are summed up in the lower part of the flow chart in Fig. B-3.

B.5. BHA Type coding

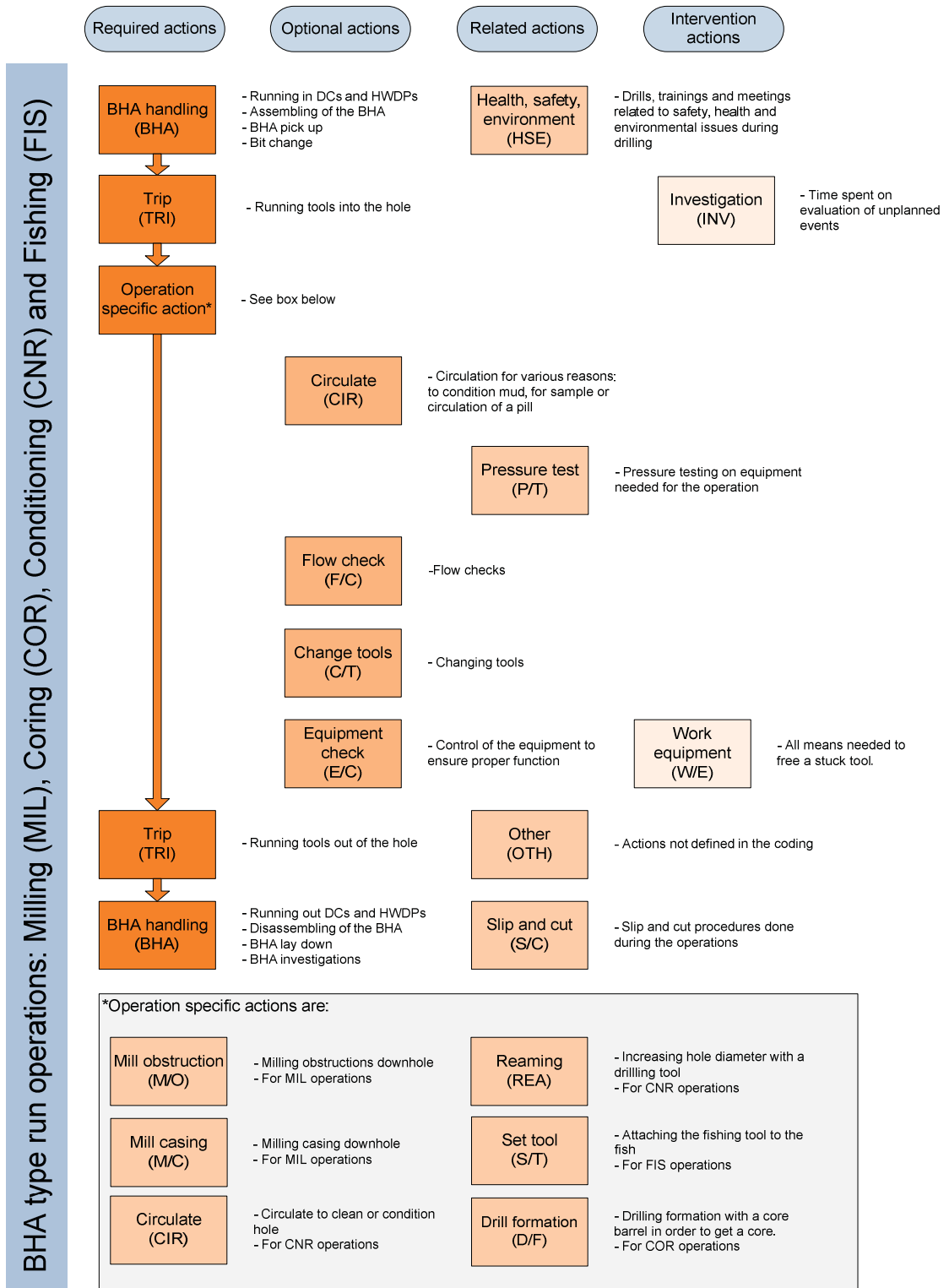


Fig. B-4 BHA type operations coding

BHA type run operations are runs that start with the assembly of a new bottom hole assembly (BHA). They include milling runs (MIL), conditioning runs (CNR), reaming runs (REA), fishing runs (FIS) and coring runs (COR). Fig. B-4 sums up the actions for those operations.

B.1. Surface Work Coding

B.1.1. Work on Surface

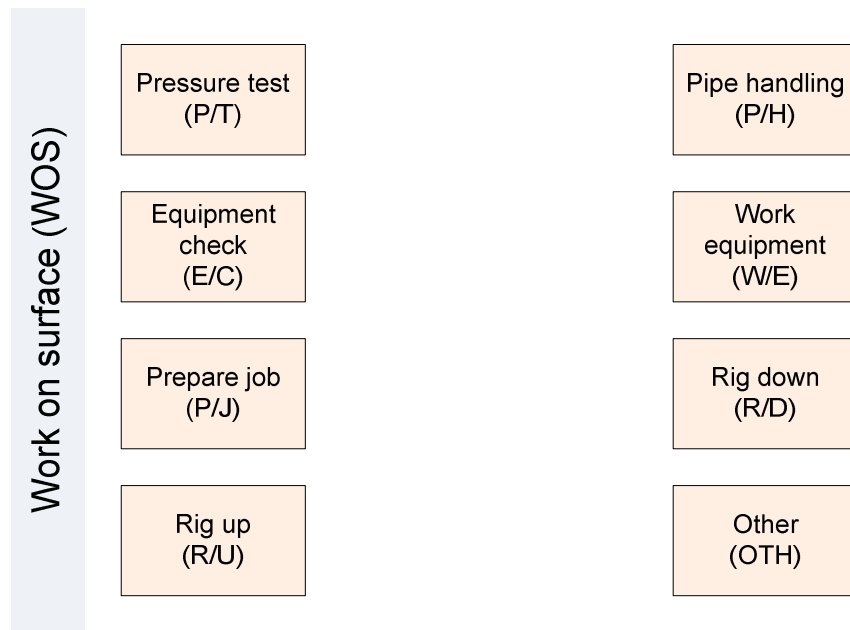


Fig. B-5 Work on surface coding

Work on surface codes describe jobs that are done while there is no downhole operation going on.

B.1.2. BOP/HW/HP System

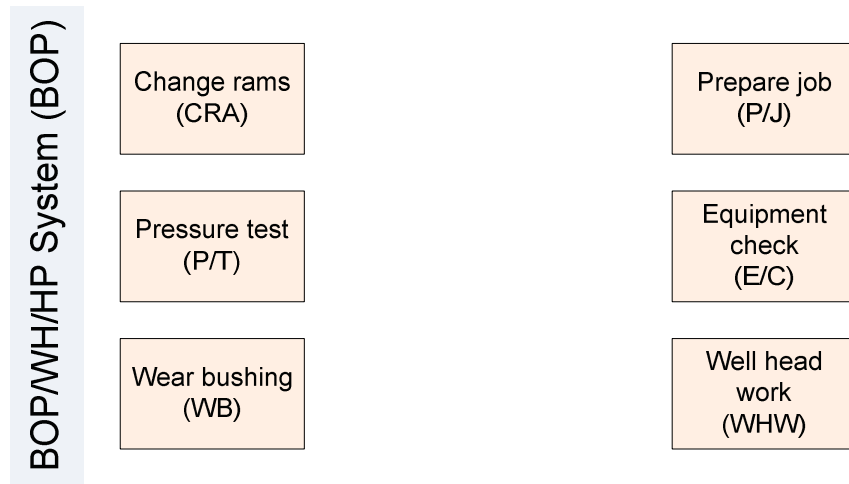


Fig. B-6 BOP/WH/HP system coding

BOP type operations should consist of the actions summed up in Fig. B-6.

B.1.3. Rig Service

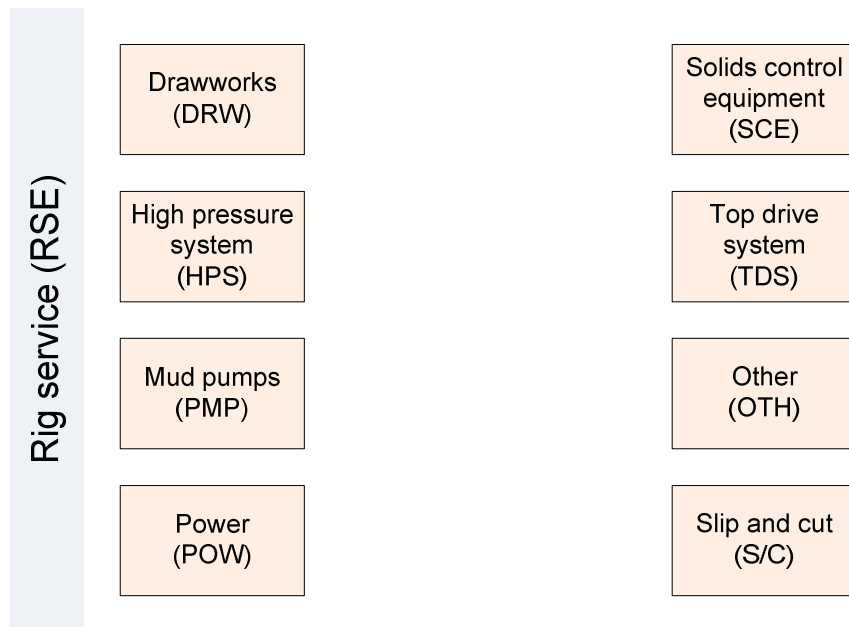


Fig. B-7 Rig service coding

Rig service operations are maintenance jobs on certain pieces of equipment. Time spent on (RSE) is considered prevention time and should be analyzed accordingly.

B.2. Intervention Coding

The following coding is by definition Intervention coding since it is not planned for.

B.2.1. Rig repair

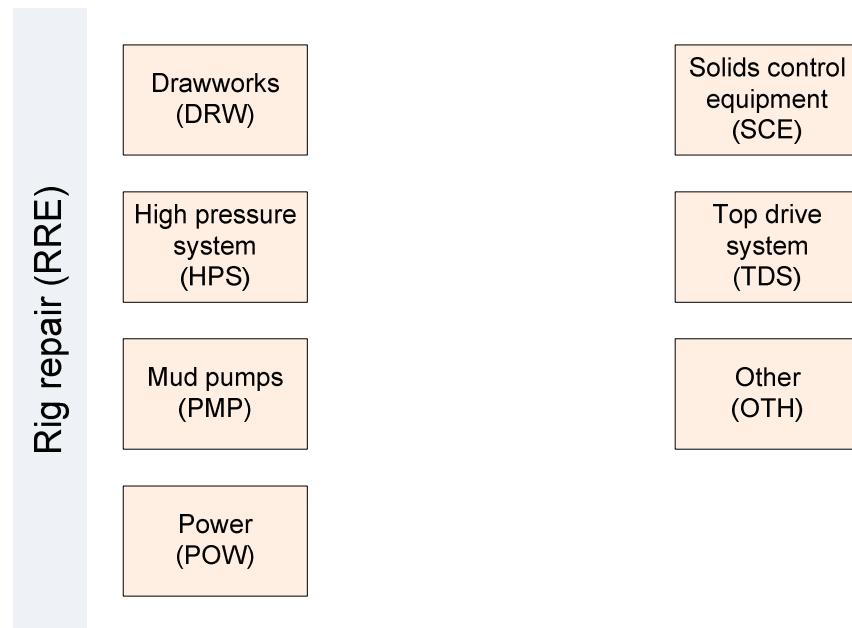


Fig. B-8 Rig repair coding

Rig repair actions are those that are need for repairing broken machinery and restoring the status quo. Replacing of broken equipment parts should be coded as RRE as well. Together with RSE this coding is the basis for a optimum level of perfection analysis.

B.2.2. Well Control

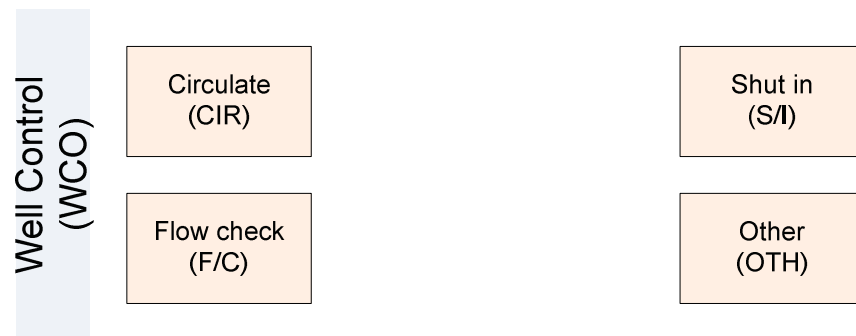


Fig. B-9 Well control coding

Fig. B-9 sums up coding that is needed to report well control situations.

B.2.3. Waiting Time

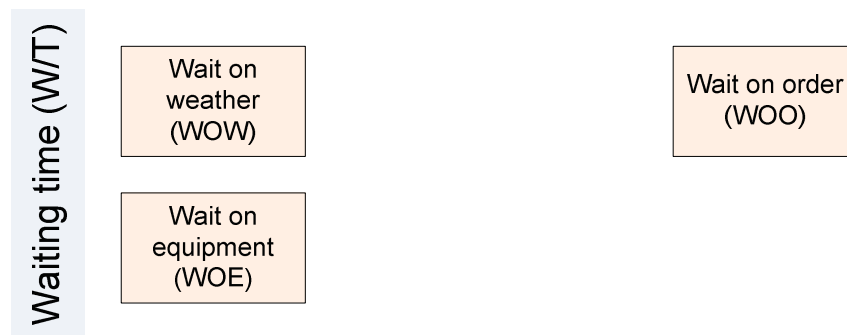


Fig. B-10 Waiting time coding

Fig. B-10 sums up actions needed to report waiting time.

APPENDIX C: REPORT VERSUS ACTUAL EVALUATION CODE

C.1. Introduction

This chapter is designed to give a basic overview of the Report versus Actual Evaluation code. It will briefly discuss the format requirements for the data, the use of the code and the generation of final results.

C.2. Requirements

The code¹ requires input data sets formatted in Excel. It was designed for Microsoft Excel 2007©. At the present time there is no plan to export to previous versions. The dataset, need for analysis, has to consist of the following information:

- Planned and actual durations
- Planned and actual coding
 - The algorithm can handle a two level hierarchic coding
 - One level works fine as well
- Depth

Every planned and actual dataset must be in one row. Actual actions that have no corresponding planned equivalent will be identified as interventions and will not be analyzed. Exceptions are continued actions which are present on the actual side with no planned equivalent within the same line but which do have a corresponding planned base action.

¹ Some terminology issues: „code“ or „the code“ in this thesis always refers to the VBA code presented in this Appendix. „Coding“ will always refer to OMV’s operation and action coding system.

Table C-1 Data example

Planned	Actual				
Time [h]	Code	Description	Time [h]	Code	Description
8	RL	Drill f/1000 t/1500	3	DRL	Drilled f/1000 t/1200
			1	HPS	Pump failure
			2	DRL	Drilled f/1200 t/1300
			3	DRL	Drilled f/1300 t/1500
2	RI	Trip out	2	TRI	Tripped out

Table C-1 presents a simple data set. The first line is identified by the VBA code as a base action. Since empty lines follow it, that means that it could be interrupted by unplanned events or it may be reported in more than one line. Line 2 is identified as an intervention since it has no corresponding planned code. Line 3 is identified as a continued action of the first line. It does not have a corresponding planned entry in the same line but it does match the base action coding. The same is valid for line 4. Its time will be added to the total drill time. Line 5 starts a new action.

In this example, the code will compare the eight hours of planned drilling with eight hours of actual drilling. The actual drilling consists of three hours from line one, two hours from line 3 and three hours from line 4. Since the pump failure is treated as an intervention, it will not be considered.

Table 3-1 is also a good example showing what the data set should look like. Note that depth is not included here since it is only needed for generation of TVD curves. The description is completely irrelevant for the code itself but might be a good source for additional information and double check of the coding.

C.2.1. Coding Hierarchy

The code can handle a two level coding hierarchy. That means that the coding can consist of higher level (operation) codes like drilling or cementing and lower level (action) codes that describe the upper level in more detail. This system corresponds to a 2 level work breakdowns structure (0).

In a situation where one only uses one level of coding, the code is set in a way where both levels refer to the same column in the spreadsheet. (See chapter “User Interface” for more details on setting columns)

C.3. Code Work Flow

The code consists of a number of algorithms (identification, sorting and calculation) that do all the work in order to get the benchmarks and the histograms.

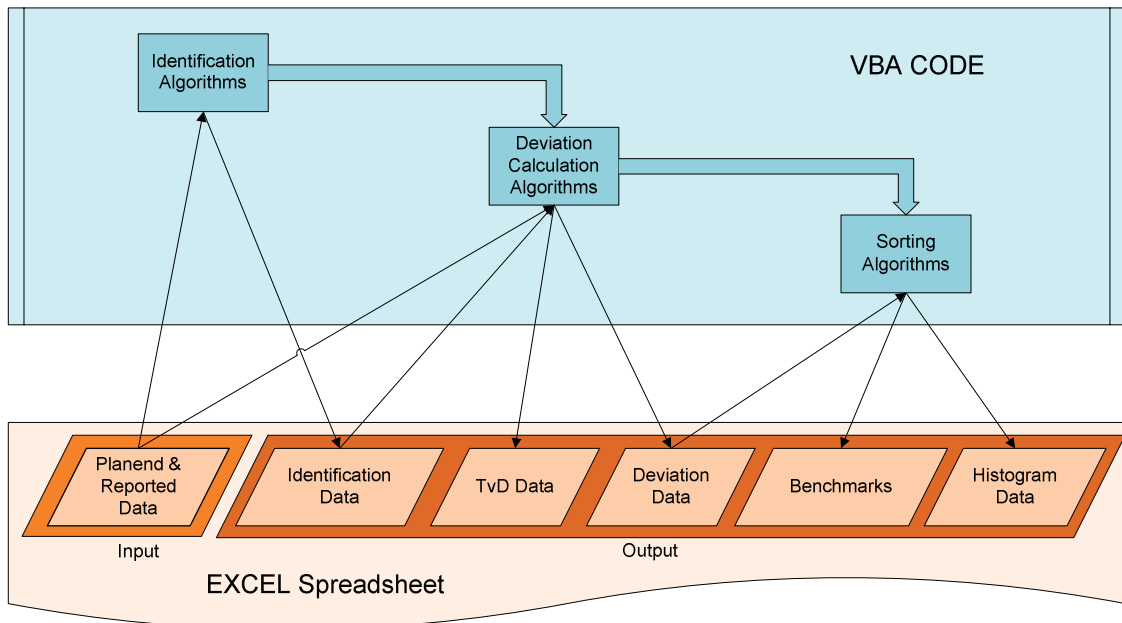


Fig. C-1 VBA Code Flowchart

Fig. C-1 demonstrates the workflow of the code. The first step is done by the identification algorithms. Those algorithms are named beginning with the word “Find”. The task here is to identify the coding correctly and therefore make the deviation calculation possible.

The deviation calculation algorithms calculates, based on the planned and reported data and the identification data the deviation and the TVD data. All this data is plotted in the corresponding cells in the spreadsheet. Calculation algorithms start with the word “Get”.

Out of this deviation data, the sorting algorithms (all named with the word “Create” in the beginning) create the benchmarks and the histogram data.

C.4. Results

The code delivers multiple types of numeric results.

Table C-2 Summary

Codes	Time Planned	Time Spent
WSF	3.50	5.5
DRL	163.75	222.5
CNR	14.25	14.25
LOG	5.50	4.75
CAS	46.25	56.25
BOP	26.00	41.5
Interventions	0.00	63.25
Sum	259.25	408

Table C-3 Benchmarks

Type	Value
Total for:	All Actions
Avg Deviation:	-0.54247092
Accuracy:	0.33673469
Avg Miss:	0.91974882
Total for:	WSF
AD	-0.44444444
Acc	0.33333333
AM	0.66666667

Table C-2 displays the time planned and time spent for all operations. Interventions are displayed as well as a sum.

Table C-3 is an excerpt of the available benchmark. The code generates benchmarks for the whole process based on the action level and the operation level. It also provides benchmarks for individual operations based on the action level, as shown in Table C-3 for work on surface (WSF). The benchmarks are displayed in the first two free columns of the spreadsheet.

The next results are of Diagram/Histogram type.

Table C-4 Histogram Data

Range	Count
-16	1
-8	1
-4	5
-2	3
-1	19
-0.5	6
-0.25	3
-0.125	8
0	33
0.015625	8
0.0625	1
0.140625	2
0.25	2
0.390625	2
0.5625	3
0.765625	1
1	

Table C-5 TvD Data

Time w/o Intervention	Depth w/o Intervention	Time w/ Intervention	Depth w/ Intervention
2.5	92		
5	92		
11	92		
11.25	92		
24	92		
40.25	366		
		41	366
48	366		
56	366		
57	366		
64.5	366		
65.5	366		
67	366		
67.75	366		
69.5	366		
70.25	366		
72	366		
72.75	460		
		73.5	460
76	460		
80.5	460		
81	460		
83	460		

Table C-4 shows the generated histogram data. This data has to be processed manually into a diagram (by the use of Microsoft Excel’s © “Insert Diagram” function) like the one presented in the thesis. The code will generate histogram data for all actions, all operations, and for individual operations based on the actions. It is the user’s choice which one to plot.

Table C-5 is an excerpt of the TVD Data. This data can be used to create a time versus depth curve of the actual drilling process, where interventions are plotted separately. Once again this data, if a diagram is needed, is plotted manually by the user.

C.5. User Interface

The only point where the user directly interacts with the code is in the first part, the constant declaration. In this part the user has to “tell” the code where the necessary data is. This is done by declaring the column positions of the necessary data. Note that VBA does not use letters

but numbers to distinguish between columns. So column “A” is column “1”, “B” is “2” and so on. The user can switch Excel to this numeric labeling through “Excel-Options”.

Another point where the user might want to interact is the histogram constants declaration. These two numbers define the amount of resolution steps and the negative borderline. A change of the resolution distribution requires a change in the code (Algorithm: 'Create All Action Histogram).

C.6. Report versus Actual Evaluation Code

C.6.1. Introduction

The following documents consist of the complete code used in the Report versus Actual Evaluation as well as explanations of the code. Basic explanations are in italic and part of the code (as notes). Beginnings of individual procedures are marked with two “---“lines intersected by the name of the procedure. Those names are formatted as chapters and are listed in the List of Contents above.

Further explanations wherever available are marked with footnotes. The code from the word “Sub” to the words “End Sub”, can be copied and pasted directly to VBA in an Excel file and executed (assuming the constants are declared correctly).

```
Sub PerformPlannedVsActaulAnalysis()
```

```
'-----
```

C.6.2. 'Declaration of constant Column positions

```
'-----
```

'All constants declared here are dependent on the actual style of the data that has to be analyzed. They define the column position (CP) where the algorithm gets its data from.

```
Const plannedDurationColumn As Integer = 8 'CP with planned duration  
Const plannedOpsColumn As Integer = 9 'CP with planned operation coding  
Const plannedActColumn As Integer = 10 'CP with planned action coding  
Const dateColumn As Integer = 12 'CP with date2  
Const actualRunningDepthColumn As Integer = 14 'CP with running depth  
Const actualDurationColumn As Integer = 17 'CP with actual duration  
Const actualOpsColumn As Integer = 18 'CP with actual operation coding
```

² „dateColumn“ doesn't necessarily have to be a date. The algorithm only needs a column that is continuously filled with some sort of information (doesn't matter what as long it's not empty) from the beginning to the end of the analyzed data.

```

Const actualActColumn As Integer = 19 'CP with actual action coding
Const cfirstFreeColumn As Integer = 21 'CP with the first empty column3
Const firstLineWithCode As Integer = 4 ' Number of the first line with code

```

'Histogram constants

```

Const borderline = 16 'Maximum negative relative deviation value
Const resolutionSteps = 8 'Amount of Steps, from zero into one directon

```

C.6.3. 'Calculate Actual Running Time

'This subroutine calculates a cumulative running time based on the actual durations for TvD curves.

```

firstFreeColumn = cfirstFreeColumn
zeile = firstLineWithCode
actualRunningTime = 0
actualRunningTimeColumn = firstFreeColumn + 6
firstFreeColumn = firstFreeColumn + 7
Cells(zeile - 1, actualRunningTimeColumn) = "Act Runing Time"
Do While Cells(zeile, dateColumn).Value <> ""
    actualRunningTime = Cells(zeile, actualDurationColumn).Value + actualRunningTime
    Cells(zeile, actualRunningTimeColumn).Value = actualRunningTime
    zeile = zeile + 1
Loop

```

C.6.4. 'Calculate Planned Running Time

'This subroutine calculates a cumulative running time based on planned durations.

```

zeile = firstLineWithCode
plannedRunningTime = 0
plannedRunningTimeColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, plannedRunningTimeColumn) = "Pln Runing Time"
Do While Cells(zeile, dateColumn).Value <> ""
    plannedRunningTime = Cells(zeile, plannedDurationColumn).Value + plannedRunningTime
    Cells(zeile, plannedRunningTimeColumn).Value = plannedRunningTime
    zeile = zeile + 1

```

³ This is the position were the algorithm will start writing its results

Loop

C.6.5. 'Find Base Action

'This subroutine finds base actions⁴ and marks them with a "1" in the designated column

```
zeile = firstLineWithCode
baseActColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, baseActColumn) = "Base Act ID"
Do While Cells(zeile, dateColumn).Value <> ""
    If Cells(zeile + 1, plannedOpsColumn).Value = "" And Cells(zeile,
plannedOpsColumn).Value <> "" Then Cells(zeile, baseActColumn).Value = 1 Else Cells(zeile,
baseActColumn).Value = 0
    zeile = zeile + 1
Loop
```

C.6.6. 'Find Base Operation

'This subroutine identifies a base operations⁵ and marks them with a "1" in the designated column

```
zeile = firstLineWithCode
baseOprColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, baseOprColumn) = "Base Opr ID"
lastOpr = "xxx"
Do While Cells(zeile, dateColumn).Value <> ""
    If Cells(zeile, plannedOpsColumn).Value <> lastOpr And Cells(zeile,
plannedOpsColumn).Value <> "" Then
        Cells(zeile, baseOprColumn).Value = 1
        lastOpr = Cells(zeile, plannedOpsColumn).Value
    Else
        Cells(zeile, baseOprColumn).Value = 0
    End If
    zeile = zeile + 1
```

⁴ A base action is a planned action that was planned in one line, but reported in more than one. Reasons could be a midnight split or interventions during the action.

⁵ A base operation marks the line where an operation starts and the one before ends.

Loop

C.6.7. 'Find Continued Action

'This subroutine Identifies continued actions⁶ and marks them with a 1

```
zeile = firstLineWithCode
counter = 0
contActColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, contActColumn) = "Cont Act ID"
Do While Cells(zeile, dateColumn) <> ""
  counter = 0
  If Cells(zeile, plannedOpsColumn) = "" Then
    Do While Cells(zeile + counter, plannedOpsColumn) = ""
      counter = counter + 1
      If counter > 50 Then Exit Do
    Loop
    For x = 0 To counter
      If Cells(zeile - 1, plannedOpsColumn).Value = Cells(zeile + x, actualOpsColumn).Value
Then
      If Cells(zeile - 1, plannedActColumn).Value = Cells(zeile + x,
actualActColumn).Value Then Cells(zeile + x, contActColumn).Value = 1
      End If
    Next x
  End If
  zeile = zeile + counter
  zeile = zeile + 1
Loop
zeile = firstLineWithCode
'Fills the gaps with zeros
Do While Cells(zeile, dateColumn) <> ""
  If Cells(zeile, contActColumn).Value <> 1 Then Cells(zeile, contActColumn).Value = 0
  zeile = zeile + 1
Loop
```

C.6.8. 'Find Intervention

⁶ Continued actions are those planned actions that are parts of a base action with no corresponding entry in the plan.

'This routine identifies Interventions⁷ in the corresponding column with a 1

```
zeile = firstLineWithCode
interventionIdentificationColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, interventionIdentificationColumn) = "Intervention ID"
counter = 0
Do While Cells(zeile, dateColumn) <> ""
    counter = 0
    If Cells(zeile, plannedOpsColumn) = "" Then
        Do While Cells(zeile + counter, plannedOpsColumn) = ""
            counter = counter + 1
            If counter > 50 Then Exit Do
        Loop
        For x = 0 To counter - 1
            If Cells(zeile - 1, plannedOpsColumn).Value <> Cells(zeile + x,
actualOpsColumn).Value Or Cells(zeile - 1, plannedActColumn).Value <> Cells(zeile + x,
actualActColumn).Value Then
                Cells(zeile + x, interventionIdentificationColumn).Value = 1
                Cells(zeile + x, interventionIdentificationColumn).Interior.Color = RGB(255, 0, 0)
            End If
        Next x
    End If
    zeile = zeile + counter
    zeile = zeile + 1
Loop
zeile = firstLineWithCode
'Fills the gaps with zeros
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, interventionIdentificationColumn).Value <> 1 Then Cells(zeile,
interventionIdentificationColumn).Value = 0
    zeile = zeile + 1
Loop
```

C.6.9. 'Find Continued Operations

'This subroutine identifies continued operations⁸ and marks them with a 1.

```
zeile = firstLineWithCode
contOprColumn = firstFreeColumn
```

⁷ Interventions are unplanned actions (per Def.).

⁸ Continued operations are operations that follow a base operation and are of the same kind.


```

firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, contOprColumn) = "Cont Opr ID"
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, baseOprColumn) <> 1 And Cells(zeile, interventionIdentificationColumn) <> 1
Then Cells(zeile, contOprColumn).Value = 1 Else Cells(zeile, contOprColumn).Value = 0
        zeile = zeile + 1
    Loop

```

C.6.10.'Get Total Action Duration

'This function calculates the total duration of an action if the action consists of a base action and continued actions

```

zeile = firstLineWithCode
totalActDurationColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, totalActDurationColumn) = "Total Action Duration"
totalActDuration = 0
limit = 50
counter = 0
Do While Cells(zeile, dateColumn) <> ""
    counter = 0
    totalActDuration = 0
    If Cells(zeile, plannedOpsColumn) = "" Then
        Do While Cells(zeile + counter, plannedOpsColumn) = ""
            counter = counter + 1
            If counter > limit Then Exit Do
        Loop
        For x = 0 To counter - 1
            If Cells(zeile - 1, plannedOpsColumn).Value = Cells(zeile + x, actualOpsColumn).Value
Then
                If Cells(zeile - 1, plannedActColumn).Value = Cells(zeile + x,
actualActColumn).Value Then
                    totalActDuration = totalActDuration + Cells(zeile + x, actualDurationColumn).Value
                End If
            End If
        Next x
        totalActDuration = totalActDuration + Cells(zeile - 1, actualDurationColumn).Value
        Cells(zeile - 1, totalActDurationColumn).Value = totalActDuration
    Else
        Cells(zeile, totalActDurationColumn).Value = Cells(zeile, actualDurationColumn).Value
    End If
    If counter = 0 Then zeile = zeile + 1 Else zeile = zeile + counter

```

Loop

C.6.11.'Get Total Actual Operation Duration

'This function calculates the time needed for a total Operation. The total Operation is consists of the base action + continued operation time

```
zeile = firstLineWithCode
totalOprDurationColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, totalOprDurationColumn) = "Total Actual Operation Duration"
totalOprDuration = 0
limit = 200
Do While Cells(zeile, dateColumn) <> ""
    counter = 1
    totalOprDuration = 0
    If Cells(zeile, baseOprColumn) = 1 Then
        totalOprDuration = totalOprDuration + Cells(zeile, actualDurationColumn).Value
        Do While Cells(zeile + counter, baseOprColumn).Value = 0
            If Cells(zeile + counter, contOprColumn) = 1 Then totalOprDuration = totalOprDuration
+ Cells(zeile + counter, actualDurationColumn)
            counter = counter + 1
            If counter > limit Then Exit Do
        Loop
        Cells(zeile, totalOprDurationColumn).Value = totalOprDuration
    Else
        Cells(zeile, totalOprDurationColumn).Value = 0
    End If
    zeile = zeile + 1
Loop
```

C.6.12.'Get Total Planned Operation Duration

'This function calculates the time planned for a total Operation. The total operation consist of the base operation + continued operation planned time.

```
zeile = firstLineWithCode
plannedOprDurationColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, plannedOprDurationColumn) = "Total Planned Operation Duration"
totalOprDuration = 0
```

```

limit = 200
Do While Cells(zeile, dateColumn) <> ""
    counter = 1
    plannedOprDuration = 0
    If Cells(zeile, baseOprColumn) = 1 Then
        plannedOprDuration = plannedOprDuration + Cells(zeile, plannedDurationColumn).Value
        Do While Cells(zeile + counter, baseOprColumn).Value = 0
            If Cells(zeile + counter, contOprColumn) = 1 Then plannedOprDuration =
plannedOprDuration + Cells(zeile + counter, plannedDurationColumn)
            counter = counter + 1
            If counter > limit Then Exit Do
        Loop
        Cells(zeile, plannedOprDurationColumn).Value = plannedOprDuration
    Else
        Cells(zeile, plannedOprDurationColumn).Value = 0
    End If
    zeile = zeile + 1
Loop
'-----

```

C.6.13.'Get Absolute Action Deviation

```

'-----
'This function calculates the absolute deviation between planned and reported actions

zeile = firstLineWithCode
absoluteDeviationColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, absoluteDeviationColumn) = "Absolute Action Deviation"
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, plannedOpsColumn) <> "" Then Cells(zeile, absoluteDeviationColumn).Value
= Cells(zeile, plannedDurationColumn).Value - Cells(zeile, totalActDurationColumn).Value Else
Cells(zeile, absoluteDeviationColumn).Value = ""
    zeile = zeile + 1
Loop
'-----

```

C.6.14.'Get Lag Time

```

'-----
'This function calculates the lag time9.

zeile = firstLineWithCode

```

⁹ Lag Time is a cumulative sum of all absolute deviations

```

lagTimeColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, lagTimeColumn) = "Lag Time"
Dim lagTime As Double
Do While Cells(zeile, dateColumn) <> ""
    lagTime = lagTime + Cells(zeile, absoluteDeviationColumn)
    Cells(zeile, lagTimeColumn) = lagTime
    zeile = zeile + 1
Loop

```

C.6.15.'Get Relative Action Deviation

'This function calculates the relative deviation between a planned and an actual action.

```

zeile = firstLineWithCode
relativeDeviationColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, relativeDeviationColumn) = "Relative Action Deviation"
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, plannedOpsColumn) <> "" Then Cells(zeile, relativeDeviationColumn).Value =
(Cells(zeile, plannedDurationColumn).Value - Cells(zeile, totalActDurationColumn).Value) /
Cells(zeile, plannedDurationColumn).Value Else Cells(zeile, relativeDeviationColumn).Value = ""
    zeile = zeile + 1
Loop

```

C.6.16.'Get Absolute Operation Deviation

'This function calculates the absolute deviation between planned and actual operation duration.

```

zeile = firstLineWithCode
absoluteOprDeviationColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, absoluteOprDeviationColumn) = "Absolute Operation Deviation"
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, baseOprColumn) = 1 Then Cells(zeile, absoluteOprDeviationColumn).Value =
Cells(zeile, plannedOprDurationColumn).Value - Cells(zeile, totalOprDurationColumn).Value Else
Cells(zeile, absoluteOprDeviationColumn).Value = ""
    zeile = zeile + 1
Loop

```

C.6.17.'Get Relative Operation Deviation

'This function calculates the relative deviation between planned and actual operation duration.'

```
zeile = firstLineWithCode
relativeOprDeviationColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, relativeOprDeviationColumn) = "Relative Operation Deviation"
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, baseOprColumn) = 1 Then Cells(zeile, relativeOprDeviationColumn).Value =
(Cells(zeile, plannedOprDurationColumn).Value - Cells(zeile, totalOprDurationColumn).Value) /
Cells(zeile, plannedOprDurationColumn).Value Else Cells(zeile,
relativeOprDeviationColumn).Value = ""
    zeile = zeile + 1
Loop
```

C.6.18.'Get Times without Intervention

'This function calculates time points for the TvD diagram.'

```
zeile = firstLineWithCode
timesWithoutInterventionColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, timesWithoutInterventionColumn) = "Time w/o Intervention"
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, interventionIdentificationColumn).Value = 0 Then Cells(zeile,
timesWithoutInterventionColumn).Value = Cells(zeile, actualRunningTimeColumn).Value Else
Cells(zeile, timesWithoutInterventionColumn) = ""
    zeile = zeile + 1
Loop
```

C.6.19.'Get Depth without Intervention

'This function calculates depth points for the TvD Diagram'

```
zeile = firstLineWithCode
depthWithoutInterventionColumn = firstFreeColumn
firstFreeColumn = firstFreeColumn + 1
Cells(zeile - 1, depthWithoutInterventionColumn) = "Depth w/o Intervention"
Do While Cells(zeile, dateColumn) <> ""
```

```

    If Cells(zeile, interventionIdentificationColumn).Value = 0 Then Cells(zeile,
depthWithoutInterventionColumn).Value = Cells(zeile, actualRunningDepthColumn).Value Else
Cells(zeile, depthWithoutInterventionColumn) = ""

```

```

    zeile = zeile + 1

```

```

Loop

```

C.6.20.'Get Times with Intervention

'This function calculates time points for the TvD diagram.

```

zeile = firstLineWithCode

```

```

timesWithInterventionColumn = firstFreeColumn

```

```

firstFreeColumn = firstFreeColumn + 1

```

```

Cells(zeile - 1, timesWithInterventionColumn) = "Time w/ Intervention"

```

```

Do While Cells(zeile, dateColumn) <> ""

```

```

    If Cells(zeile, interventionIdentificationColumn).Value = 1 Then Cells(zeile,
timesWithInterventionColumn).Value = Cells(zeile, actualRunningTimeColumn).Value Else
Cells(zeile, timesWithInterventionColumn) = ""

```

```

    zeile = zeile + 1

```

```

Loop

```

C.6.21.'Get Depth with Intervention

'This function calculates depth point for the TvD diagram

```

zeile = firstLineWithCode

```

```

depthWithInterventionColumn = firstFreeColumn

```

```

firstFreeColumn = firstFreeColumn + 1

```

```

Cells(zeile - 1, depthWithInterventionColumn) = "Depth w/ Intervention"

```

```

Do While Cells(zeile, dateColumn) <> ""

```

```

    If Cells(zeile, interventionIdentificationColumn).Value = 1 Then Cells(zeile,
depthWithInterventionColumn).Value = Cells(zeile, actualRunningDepthColumn).Value Else
Cells(zeile, depthWithInterventionColumn) = ""

```

```

    zeile = zeile + 1

```

```

Loop

```

C.6.22.'Create Planned and Actual Operation Duration

'This function sums up planned and actual times based on the type of operation as well as overall. The results are written summed up in a table.

```

Dim usedOperationsLA(0 To 20)
Dim usedInterventionsLA(0 To 20)
zeile = firstLineWithCode
opsNumber = 0
'Array with all used operation codes from the planned operation is created
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, interventionIdentificationColumn).Value = 0 Then
        For x = LBound(usedOperationsLA) To UBound(usedOperationsLA)
            If usedOperationsLA(x) = Cells(zeile, plannedOpsColumn) Then Exit For Else
                If usedOperationsLA(x) = "" Then
                    usedOperationsLA(x) = Cells(zeile, plannedOpsColumn)
                    opsNumber = opsNumber + 1
                Exit For
            End If
        Next x
    End If
    zeile = zeile + 1
Loop
zeile = firstLineWithCode
'Operations that are have actually been used are added if not present yet
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, interventionIdentificationColumn).Value = 0 Then
        For x = LBound(usedOperationsLA) To UBound(usedOperationsLA)
            If usedOperationsLA(x) = Cells(zeile, actualOpsColumn) Then Exit For Else
                If usedOperationsLA(x) = "" Then
                    usedOperationsLA(x) = Cells(zeile, actualOpsColumn)
                    opsNumber = opsNumber + 1
                Exit For
            End If
        Next x
    End If
    zeile = zeile + 1
Loop
zeile = firstLineWithCode
'Array with alle used Intervention operation codes is created
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, interventionIdentificationColumn).Value = 1 Then
        For x = LBound(usedInterventionsLA) To UBound(usedInterventionsLA)
            If usedInterventionsLA(x) = Cells(zeile, actualOpsColumn) Then Exit For Else
                If usedInterventionsLA(x) = "" Then
                    usedInterventionsLA(x) = Cells(zeile, actualOpsColumn)
                Exit For
            End If
        Next x
    End If

```

```

    End If
    zeile = zeile + 1
Loop
'Reshaping of the operation arrays'
x = LBound(usedOperationsLA)
Do While usedOperationsLA(x) <> ""
    x = x + 1
Loop
ReDim usedOperations(x)
ReDim timePlanned(x)
ReDim timeSpent(x)
x = LBound(usedOperationsLA)
Do While usedOperationsLA(x) <> ""
    usedOperations(x) = usedOperationsLA(x)
    x = x + 1
Loop

x = LBound(usedInterventionsLA)
Do While usedInterventionsLA(x) <> ""
    x = x + 1
Loop
ReDim usedInterventions(x)
ReDim interventionTime(x)
x = LBound(usedInterventionsLA)
Do While usedInterventionsLA(x) <> ""
    usedInterventions(x) = usedInterventionsLA(x)
    x = x + 1
Loop
'Cumulative time of each used operation is calculated'
For x = LBound(usedOperations) To UBound(usedOperations)
    zeile = firstLineWithCode
    Do While Cells(zeile, dateColumn) <> ""
        If Cells(zeile, interventionIdentificationColumn).Value = 0 Then
            If usedOperations(x) = Cells(zeile, plannedOpsColumn) Then timePlanned(x) =
timePlanned(x) + Cells(zeile, plannedDurationColumn).Value
        End If
        zeile = zeile + 1
    Loop
Next
For x = LBound(usedOperations) To UBound(usedOperations)
    zeile = firstLineWithCode
    Do While Cells(zeile, dateColumn) <> ""
        If Cells(zeile, interventionIdentificationColumn).Value = 0 Then
            If usedOperations(x) = Cells(zeile, actualOpsColumn) Then timeSpent(x) =
timeSpent(x) + Cells(zeile, actualDurationColumn).Value
        End If
        zeile = zeile + 1
    Loop
Next

```



```

        End If
        zeile = zeile + 1
    Loop
Next
For y = LBound(usedInterventions) To UBound(usedInterventions)
    zeile = firstLineWithCode
    Do While Cells(zeile, dateColumn) <> ""
        If Cells(zeile, interventionIdentificationColumn).Value = 1 Then
            If usedInterventions(y) = Cells(zeile, actualOpsColumn) Then
                interventionTime(y) = interventionTime(y) + Cells(zeile, actualDurationColumn).Value
            End If
        End If
        End If
        zeile = zeile + 1
    Loop
Next
y = LBound(usedOperations)
'Tabellaric Display of Results
totalTimePlanned = 0
totalTimeSpent = 0
sumCodesColumn = cfirstFreeColumn
sumTimePlannedColumn = cfirstFreeColumn + 1
sumTimeActualColumn = cfirstFreeColumn + 2
Cells(firstLineWithCode - 1, sumCodesColumn) = "Codes"
Cells(firstLineWithCode - 1, sumTimePlannedColumn) = "Time Planned"
Cells(firstLineWithCode - 1, sumTimeActualColumn) = "Time Spent"
Do While usedOperations(y) <> ""
    Cells(firstLineWithCode + y, sumCodesColumn).Value = usedOperations(y)
    Cells(firstLineWithCode + y, sumTimePlannedColumn).Value = timePlanned(y)
    totalTimePlanned = totalTimePlanned + timePlanned(y)
    Cells(firstLineWithCode + y, sumTimeActualColumn).Value = timeSpent(y)
    totalTimeSpent = totalTimeSpent + timeSpent(y)
    y = y + 1
Loop
Cells(firstLineWithCode + y, sumCodesColumn) = "Interventions"
Cells(firstLineWithCode + y, sumTimePlannedColumn).Value = 0
Cells(firstLineWithCode + 1 + y, sumCodesColumn) = "Sum"
Cells(firstLineWithCode + 1 + y, sumTimePlannedColumn).Value = totalTimePlanned
t = LBound(usedInterventions)
totalInterventionTime = 0
Do While usedInterventions(t) <> ""
    Cells(firstLineWithCode + t + y + 4, sumCodesColumn).Value = usedInterventions(t)
    Cells(firstLineWithCode + t + y + 4, sumTimeActualColumn).Value = interventionTime(t)
    totalInterventionTime = totalInterventionTime + interventionTime(t)
    t = t + 1

```

```

Loop
Cells(firstLineWithCode + t + y + 4, sumCodesColumn) = "Sum"
Cells(firstLineWithCode + t + y + 4, sumTimeActualColumn).Value = totalInterventionTime
Cells(firstLineWithCode + y, sumTimeActualColumn).Value = totalInterventionTime
Cells(firstLineWithCode + 1 + y, sumTimeActualColumn).Value = totalTimeSpent +
totalInterventionTime

```

C.6.23.'Create All Action Histogram

'This Function collects the necessary data and creates a histogram at the action level.

```

zeile = firstLineWithCode
v = 0
x = 0
'Counts necessary size of Array
Do While Cells(zeile, dateColumn) <> "" 'Defines the size of histogramData Array needed
    If Cells(zeile, relativeDeviationColumn).Value <> "" Then x = x + 1
    zeile = zeile + 1
Loop
x = x - 1
zeile = firstLineWithCode
ReDim histogramData(x)
x = LBound(histogramData)
'Collects all data into histogramData array
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, relativeDeviationColumn).Value <> "" Then
        histogramData(x) = Cells(zeile, relativeDeviationColumn).Value
        x = x + 1
    End If
    zeile = zeile + 1
Loop
x = 0
'Visualisation of histogramData
For x = LBound(histogramData) To UBound(histogramData)
    Cells(firstLineWithCode + x, firstFreeColumn).Value = histogramData(x)
Next x
firstFreeColumn = firstFreeColumn + 1
t = 1
Do While t <> 0 'Sorts the histogramData from low to high
    t = 0
    For x = LBound(histogramData) To UBound(histogramData) - 1
        If histogramData(x) > histogramData(x + 1) Then
            t = t + 1

```

```

        v = histogramData(x)
        histogramData(x) = histogramData(x + 1)
        histogramData(x + 1) = v
    End If
Next x
Loop
x = 0
'Visualisation of sorted histogramData
For x = LBound(histogramData) To UBound(histogramData)
    Cells(firstLineWithCode + x, firstFreeColumn).Value = histogramData(x)
Next x
firstFreeColumn = firstFreeColumn + 1
'Definition of histogram arrays
ReDim histogramArrays(2 * resolutionSteps)
For x = LBound(histogramArrays) To UBound(histogramArrays)
    If x < resolutionSteps Then
        histogramArrays(x) = -borderline
        borderline = borderline / 2
    End If
    If x = resolutionSteps Then
        histogramArrays(x) = 0
        borderline = 1 / resolutionSteps
    End If
    If x > resolutionSteps Then
        histogramArrays(x) = borderline ^ 2
        borderline = borderline + 1 / resolutionSteps
    End If
Next
'Visualisation of histogramArray()
For x = LBound(histogramArrays) To UBound(histogramArrays)
    Cells(firstLineWithCode + x, firstFreeColumn).Value = histogramArrays(x)
Next x
firstFreeColumn = firstFreeColumn + 1
'Data count based on histogram Arrays
ReDim histogramDataCnt(2 * resolutionSteps)
outOfRangeCnt = 0
zeroCnt = 0
Cnt = 0
x = LBound(histogramArrays)
lower = histogramArrays(x)
x = UBound(histogramArrays)
upper = histogramArrays(x)
For x = LBound(histogramData) To UBound(histogramData)
    Cnt = Cnt + 1

```

```

    If histogramData(x) < lower Then outOfRangeCnt = outOfRangeCnt + 1
    If histogramData(x) > upper Then outOfRangeCnt = outOfRangeCnt + 1
    If histogramData(x) = 0 Then zeroCnt = zeroCnt + 1
    If histogramData(x) < 0 Then
        For y = LBound(histogramArrays) To resolutionSteps
            If histogramData(x) < histogramArrays(y + 1) And histogramData(x) >=
histogramArrays(y) Then histogramDataCnt(y) = histogramDataCnt(y) + 1
        Next
    Else
        If histogramData(x) > 0 Then
            For y = resolutionSteps + 1 To UBound(histogramArrays)
                If histogramData(x) < histogramArrays(y) And histogramData(x) >=
histogramArrays(y - 1) Then histogramDataCnt(y) = histogramDataCnt(y) + 1
            Next
        End If
    End If
Next x
histogramDataCnt(resolutionSteps) = zeroCnt
totalCnt = 0
For x = LBound(histogramDataCnt) To UBound(histogramDataCnt)
    Cells(firstLineWithCode + x, firstFreeColumn).Value = histogramDataCnt(x)
    totalCnt = totalCnt + histogramDataCnt(x)
Next x
firstFreeColumn = firstFreeColumn + 1
'Average deviation Calculation +Display
zeile = firstLineWithCode
avrDev = 0
y = 0
Dim avrMiss As Double
avrMiss = 0
For y = LBound(histogramData) To UBound(histogramData)
    avrDev = avrDev + histogramData(y)
    avrMiss = avrMiss + Abs(histogramData(y))
    zeile = zeile + 1
Next
If avrMiss <> 0 Then avrMiss = avrMiss / (totalCnt - zeroCnt)
avrDev = avrDev / totalCnt
benchDescriptionColumn = cfirstFreeColumn + 3
benchValueColumn = cfirstFreeColumn + 4
Cells(firstLineWithCode - 1, benchDescriptionColumn) = "Type"
Cells(firstLineWithCode - 1, benchValueColumn) = "Value"
Cells(firstLineWithCode, benchDescriptionColumn) = "Total for:"
Cells(firstLineWithCode, benchValueColumn) = "All Actions"
Cells(firstLineWithCode + 1, benchDescriptionColumn) = "Avr Deviation:"
Cells(firstLineWithCode + 1, benchValueColumn).Value = avrDev

```

'Accuracy Calculation + Display

```
accuracy = zeroCnt / totalCnt
Cells(firstLineWithCode + 2, benchDescriptionColumn) = "Accuracy:"
Cells(firstLineWithCode + 2, benchValueColumn).Value = accuracy
Cells(firstLineWithCode + 3, benchDescriptionColumn) = "Avr Miss:"
Cells(firstLineWithCode + 3, benchValueColumn).Value = avrMiss
```

C.6.24. 'Create Benchmarks

'This Function collects the necessary data and creates a histogram at the operation level.

```
r = 4 'Column factor
l = 5 'Line factor
For x = LBound(usedOperations) To UBound(usedOperations) - 1
    u = 0
    zeile = firstLineWithCode
    Do While Cells(zeile, dateColumn) <> "" 'Defines the size of specificHistogramData Array needed
        If Cells(zeile, relativeDeviationColumn).Value <> "" And Cells(zeile, plannedOpsColumn) = usedOperations(x) Then u = u + 1
        zeile = zeile + 1
    Loop
    zeile = firstLineWithCode
    ReDim specificHistogramData(u)
    u = LBound(specificHistogramData)
    Do While Cells(zeile, dateColumn) <> "" 'Collects all data into specificHistogramData array
        If Cells(zeile, relativeDeviationColumn).Value <> "" And Cells(zeile, plannedOpsColumn) = usedOperations(x) Then
            specificHistogramData(u) = Cells(zeile, relativeDeviationColumn).Value
            u = u + 1
        End If
        zeile = zeile + 1
    Loop
    Cells(firstLineWithCode + 5 + x * l, benchDescriptionColumn) = "Total for:"
    Cells(firstLineWithCode + 5 + x * l, benchValueColumn) = usedOperations(x)
    Cells(firstLineWithCode - 1, firstFreeColumn + x * r) = usedOperations(x)
    For u = LBound(specificHistogramData) To UBound(specificHistogramData)
        Cells(firstLineWithCode + u, firstFreeColumn + x * r).Value = specificHistogramData(u)
    Next u
    t = 1
    Do While t <> 0 'Sorts the histogramData from low to high
```

```

t = 0
For u = LBound(specificHistogramData) To UBound(specificHistogramData) - 1
    If specificHistogramData(u) > specificHistogramData(u + 1) Then
        t = t + 1
        v = specificHistogramData(u)
        specificHistogramData(u) = specificHistogramData(u + 1)
        specificHistogramData(u + 1) = v
    End If
Next u
Loop
avrDev = 0
avrMiss = 0
For u = LBound(specificHistogramData) To UBound(specificHistogramData)
    Cells(firstLineWithCode + u, firstFreeColumn + 1 + x * r).Value =
specificHistogramData(u)
    avrDev = avrDev + specificHistogramData(u)
    g = specificHistogramData(u)
    avrMiss = avrMiss + Abs(g)
Next u
For u = LBound(histogramArrays) To UBound(histogramArrays)
    Cells(firstLineWithCode + u, firstFreeColumn + 2 + x * r).Value = histogramArrays(u)
Next u
Cells(firstLineWithCode + 6 + x * l, benchDescriptionColumn) = "AD"
Cells(firstLineWithCode + 7 + x * l, benchDescriptionColumn) = "Acc"
Cells(firstLineWithCode + 8 + x * l, benchDescriptionColumn) = "AM"
'Counts the Values for the Histogram
ReDim histogramDataCnt(2 * resolutionSteps)
outOfRangeCnt = 0
zeroCnt = 0
Cnt = 0
u = LBound(histogramArrays)
lower = histogramArrays(u)
u = UBound(histogramArrays)
upper = histogramArrays(u)
For u = LBound(specificHistogramData) To UBound(specificHistogramData)
    Cnt = Cnt + 1
    If specificHistogramData(u) < lower Then outOfRangeCnt = outOfRangeCnt + 1
    If specificHistogramData(u) > upper Then outOfRangeCnt = outOfRangeCnt + 1
    If specificHistogramData(u) = 0 Then zeroCnt = zeroCnt + 1
    If specificHistogramData(u) < 0 Then
        For y = LBound(histogramArrays) To resolutionSteps
            If specificHistogramData(u) < histogramArrays(y + 1) And specificHistogramData(u)
>= histogramArrays(y) Then histogramDataCnt(y) = histogramDataCnt(y) + 1
        Next
    Else

```

```

        If specificHistogramData(u) > 0 Then
            For y = resolutionSteps + 1 To UBound(histogramArrays)
                If specificHistogramData(u) < histogramArrays(y) And specificHistogramData(u) >=
                    histogramArrays(y - 1) Then histogramDataCnt(y) = histogramDataCnt(y) + 1
            Next
        End If
    End If
Next u
histogramDataCnt(resolutionSteps) = zeroCnt - 1
totalCnt = 0
For u = LBound(histogramDataCnt) To UBound(histogramDataCnt)
    Cells(firstLineWithCode + u, firstFreeColumn + 3 + x * r).Value = histogramDataCnt(u)
    totalCnt = totalCnt + histogramDataCnt(u)
Next u
accuracy = (zeroCnt - 1) / totalCnt
avrDev = avrDev / totalCnt
If avrMiss <> 0 Then avrMiss = avrMiss / (totalCnt - (zeroCnt - 1))
Cells(firstLineWithCode + 6 + x * l, benchValueColumn).Value = avrDev
Cells(firstLineWithCode + 7 + x * l, benchValueColumn).Value = accuracy
Cells(firstLineWithCode + 8 + x * l, benchValueColumn).Value = avrMiss
Next x
h = x
firstFreeColumn = firstFreeColumn + x * r
'-----

```

C.6.25. 'Create All Operation Histogram

```

'-----
'This Function harvests, sorts, writes the data for an histogram at the operation level.

zeile = firstLineWithCode
v = 0
x = 0
'Counts necessary size of Array
Do While Cells(zeile, dateColumn) <> "" 'Defines the size of histogramData Array needed
    If Cells(zeile, relativeOprDeviationColumn).Value <> "" Then x = x + 1
    zeile = zeile + 1
Loop
x = x - 1
zeile = firstLineWithCode
ReDim histogramOprData(x)
x = LBound(histogramOprData)
'Collects all data into histogramData array
Do While Cells(zeile, dateColumn) <> ""
    If Cells(zeile, relativeOprDeviationColumn).Value <> "" Then

```

```

        histogramOprData(x) = Cells(zeile, relativeOprDeviationColumn).Value
        x = x + 1
    End If
    zeile = zeile + 1
Loop
x = 0
'Visualisation of histogramOprData
For x = LBound(histogramOprData) To UBound(histogramOprData)
    Cells(firstLineWithCode + x, firstFreeColumn).Value = histogramOprData(x)
Next x
t = 1
Do While t <> 0 'Sorts the histogramOprData from low to high
    t = 0
    For x = LBound(histogramOprData) To UBound(histogramOprData) - 1
        If histogramOprData(x) > histogramOprData(x + 1) Then
            t = t + 1
            v = histogramOprData(x)
            histogramOprData(x) = histogramOprData(x + 1)
            histogramOprData(x + 1) = v
        End If
    Next x
Loop
x = 0
'Visualisation of sorted histogramData
For x = LBound(histogramOprData) To UBound(histogramOprData)
    Cells(firstLineWithCode + x, firstFreeColumn + 1).Value = histogramOprData(x)
Next x
'Visualisation of histogramArray()
For x = LBound(histogramArrays) To UBound(histogramArrays)
    Cells(firstLineWithCode + x, firstFreeColumn + 2).Value = histogramArrays(x)
Next x
'Data count based on histogram Arrays
ReDim histogramOprDataCnt(2 * resolutionSteps)
outOfRangeCnt = 0
zeroCnt = 0
Cnt = 0
x = LBound(histogramArrays)
lower = histogramArrays(x)
x = UBound(histogramArrays)
upper = histogramArrays(x)
For x = LBound(histogramOprData) To UBound(histogramOprData)
    Cnt = Cnt + 1
    If histogramOprData(x) < lower Then outOfRangeCnt = outOfRangeCnt + 1
    If histogramOprData(x) > upper Then outOfRangeCnt = outOfRangeCnt + 1

```



```

    If histogramOprData(x) = 0 Then zeroCnt = zeroCnt + 1
    If histogramOprData(x) < 0 Then
        For y = LBound(histogramArrays) To resolutionSteps
            If histogramOprData(x) < histogramArrays(y + 1) And histogramOprData(x) >=
histogramArrays(y) Then histogramOprDataCnt(y) = histogramOprDataCnt(y) + 1
        Next
    Else
        If histogramOprData(x) > 0 Then
            For y = resolutionSteps + 1 To UBound(histogramArrays)
                If histogramOprData(x) < histogramArrays(y) And histogramOprData(x) >=
histogramArrays(y - 1) Then histogramOprDataCnt(y) = histogramOprDataCnt(y) + 1
            Next
        End If
    End If
Next x
histogramOprDataCnt(resolutionSteps) = zeroCnt
totalCnt = 0
For x = LBound(histogramOprDataCnt) To UBound(histogramOprDataCnt)
    Cells(firstLineWithCode + x, firstFreeColumn + 3).Value = histogramOprDataCnt(x)
    totalCnt = totalCnt + histogramOprDataCnt(x)
Next x
'Avrage deviation Calculation + Display
zeile = firstLineWithCode
avrDev = 0
y = 0
avrMiss = 0
For y = LBound(histogramOprData) To UBound(histogramOprData)
    avrDev = avrDev + histogramOprData(y)
    avrMiss = avrMiss + Abs(histogramOprData(y))
    zeile = zeile + 1
Next
If avrMiss <> 0 Then avrMiss = avrMiss / (totalCnt - zeroCnt)
avrDev = avrDev / totalCnt
Cells(firstLineWithCode + 5 + h * l, benchDescriptionColumn) = "Total for:"
Cells(firstLineWithCode + 5 + h * l, benchValueColumn) = "All Operations"
Cells(firstLineWithCode + 6 + h * l, benchDescriptionColumn) = "Avr Deviation:"
Cells(firstLineWithCode + 6 + h * l, benchValueColumn).Value = avrDev
'Accuracy Calculation + Display
accuracy = zeroCnt / totalCnt
Cells(firstLineWithCode + 7 + h * l, benchDescriptionColumn) = "Accuracy:"
Cells(firstLineWithCode + 7 + h * l, benchValueColumn).Value = accuracy
Cells(firstLineWithCode + 8 + h * l, benchDescriptionColumn) = "Avr Miss:"
Cells(firstLineWithCode + 8 + h * l, benchValueColumn).Value = avrMiss
End Sub

```