### Document change log

<table>
<thead>
<tr>
<th>Revision</th>
<th>Edition date</th>
<th>Author</th>
<th>Modified sections / pages</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>V1.0</td>
<td>28/04/2016</td>
<td>VTT,</td>
<td>Submitted 28/04/2016</td>
<td>Rejected with comments</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fraunhofer, TRT-NL, CEA</td>
<td></td>
<td></td>
</tr>
<tr>
<td>V2.0</td>
<td>09/11/2016</td>
<td>VTT,</td>
<td>Submitted 20/11/2016</td>
<td>Comments taken into account</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Fraunhofer, TRT-NL, CEA</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Project co-funded by the European Commission within the Seventh Framework Programme (2007-2013)

**Dissemination level**

<table>
<thead>
<tr>
<th>PU</th>
<th>Public</th>
<th>✔</th>
</tr>
</thead>
<tbody>
<tr>
<td>PP</td>
<td>Restricted to other programme participants (including the Commission Services)</td>
<td></td>
</tr>
<tr>
<td>RE</td>
<td>Restricted to a group specified by the consortium (including the Commission Services)</td>
<td></td>
</tr>
<tr>
<td>CO</td>
<td>Confidential, only for members of the consortium (including the Commission Services)</td>
<td></td>
</tr>
</tbody>
</table>
# Internal Review

<table>
<thead>
<tr>
<th>Done by</th>
<th>Signature</th>
<th>Date</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eric Luijf</td>
<td>Eric Luijf</td>
<td>03/03/2016</td>
<td>Rejected</td>
</tr>
<tr>
<td>Eric Luijf</td>
<td>Eric Luijf</td>
<td>25/04/2016</td>
<td>Issues to correct</td>
</tr>
</tbody>
</table>

# Security Assessment

<table>
<thead>
<tr>
<th>Done by</th>
<th>Signature</th>
<th>Date</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eric Luijf</td>
<td>Eric Luijf</td>
<td>03/03/2016</td>
<td>No objections</td>
</tr>
<tr>
<td>Dominique Sérafin</td>
<td>Dominique Sérafin</td>
<td>17/11/2016</td>
<td>No security issue</td>
</tr>
</tbody>
</table>

# Ethical Assessment

<table>
<thead>
<tr>
<th>Done by</th>
<th>Signature</th>
<th>Date</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dominique Sérafin</td>
<td>Dominique Sérafin</td>
<td>17/11/2016</td>
<td>No ethical issue</td>
</tr>
</tbody>
</table>

# Quality Assessment

<table>
<thead>
<tr>
<th>Done by</th>
<th>Signature</th>
<th>Date</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yohan Barbarin</td>
<td>Yohan Barbarin</td>
<td>17/11/2016</td>
<td>Ready for submission</td>
</tr>
</tbody>
</table>

**DISCLAIMER**

“The contents of this document and the view expressed in the publication are the sole responsibility of the author and under no circumstances can be regarded as reflecting the position of the European Union.”
ACKNOWLEDGEMENTS

Writing, editing and revising of this document was made in close co-operation between VTT, TRT-NL, Fraunhofer and CEA. VTT had the main role in the writing process and the main responsibility in the development of the SOTM-methodology. TRT-NL was responsible for the application of Agent-based methodology, Fraunhofer took care of the human factors by adding the PSF-approach and CEA was responsible for the event trees & sequences assessment.
# LIST OF ABBREVIATIONS

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABS</td>
<td>Agent-based simulation</td>
</tr>
<tr>
<td>AD</td>
<td>Ambulance diversion</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>BE</td>
<td>Basic event</td>
</tr>
<tr>
<td>BFM-STPA</td>
<td>System-theoretic process analysis based on formalization model</td>
</tr>
<tr>
<td>BLEVE</td>
<td>Boiling liquid expanding vapour explosion</td>
</tr>
<tr>
<td>CI</td>
<td>Critical Infrastructure</td>
</tr>
<tr>
<td>CIP</td>
<td>Critical infrastructure protection</td>
</tr>
<tr>
<td>CPN</td>
<td>Coloured Petri net</td>
</tr>
<tr>
<td>DEIN</td>
<td>Dynamic expertise integration network</td>
</tr>
<tr>
<td>DES</td>
<td>Discrete event simulation</td>
</tr>
<tr>
<td>DPIF</td>
<td>Dynamic process integration framework</td>
</tr>
<tr>
<td>DSS</td>
<td>Decision support system</td>
</tr>
<tr>
<td>DT</td>
<td>Decision tree</td>
</tr>
<tr>
<td>EMS</td>
<td>Emergency medical service</td>
</tr>
<tr>
<td>EP</td>
<td>Error probability</td>
</tr>
<tr>
<td>ET</td>
<td>Event tree</td>
</tr>
<tr>
<td>FBIM</td>
<td>Fire brigade intervention model</td>
</tr>
<tr>
<td>FIERA</td>
<td>Fire evaluation and risk assessment system</td>
</tr>
<tr>
<td>GSPN</td>
<td>Generalized stochastic Petri net</td>
</tr>
<tr>
<td>HEP</td>
<td>Human error probability</td>
</tr>
<tr>
<td>HES-MAD</td>
<td>Hospital event simulation model: Arrivals to discharge</td>
</tr>
<tr>
<td>HMI</td>
<td>Human-machine interface</td>
</tr>
<tr>
<td>HRA</td>
<td>Human reliability analysis</td>
</tr>
<tr>
<td>ICU</td>
<td>Intensive care unit</td>
</tr>
<tr>
<td>ID</td>
<td>Infectious disease</td>
</tr>
<tr>
<td>IEC</td>
<td>International Electrotechnical Commission</td>
</tr>
<tr>
<td>iPDT</td>
<td>integrated PREDICT Decision support Tool</td>
</tr>
<tr>
<td>MAS</td>
<td>Multi-agent system</td>
</tr>
<tr>
<td>MS&amp;A</td>
<td>Modelling, Simulation &amp; Analysis</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>ND</td>
<td>Natural disaster</td>
</tr>
<tr>
<td>NPP</td>
<td>Nuclear power plant</td>
</tr>
<tr>
<td>PFS</td>
<td>Probabilistic fire simulator</td>
</tr>
<tr>
<td>PN</td>
<td>Petri net</td>
</tr>
<tr>
<td>PRA</td>
<td>Probabilistic risk assessment</td>
</tr>
<tr>
<td>PSF</td>
<td>Performance shaping factors</td>
</tr>
<tr>
<td>RT</td>
<td>Radiation therapy</td>
</tr>
<tr>
<td>SBR</td>
<td>Scenario based reasoning</td>
</tr>
<tr>
<td>SD</td>
<td>System dynamics</td>
</tr>
<tr>
<td>SDK</td>
<td>Software development kit</td>
</tr>
<tr>
<td>SOA</td>
<td>Service oriented architecture</td>
</tr>
<tr>
<td>SOTM</td>
<td>Stochastic operation time modelling</td>
</tr>
<tr>
<td>SPAR-H</td>
<td>Standardised plant analysis risk-human reliability analysis method</td>
</tr>
<tr>
<td>SPN</td>
<td>Stochastic Petri net</td>
</tr>
<tr>
<td>SRN</td>
<td>Stochastic reward net</td>
</tr>
<tr>
<td>SSM</td>
<td>Soft systems methodology</td>
</tr>
<tr>
<td>STEP</td>
<td>Sequentially timed event plotting</td>
</tr>
<tr>
<td>TCHPN</td>
<td>Timed coloured hybrid Petri net</td>
</tr>
<tr>
<td>TE</td>
<td>Top event</td>
</tr>
<tr>
<td>TIPN</td>
<td>Time interval Petri net</td>
</tr>
<tr>
<td>UAV</td>
<td>Unmanned aerial vehicle</td>
</tr>
<tr>
<td>UGV</td>
<td>Unmanned ground vehicle</td>
</tr>
<tr>
<td>XML</td>
<td>Extensible markup language</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

**EXECUTIVE SUMMARY** .............................................................................................................. 9

**1** INTRODUCTION ...................................................................................................................... 10

**2** STOCHASTIC OPERATION TIME MODEL (SOTM) ............................................................... 15
   - **2.1** Basic idea and purpose of the methodology ..................................................................... 15
   - **2.2** Mathematical description of the stochastic operation time model .............................. 17
   - **2.3** Modelling steps .............................................................................................................. 18
     - **2.3.1** Example case ........................................................................................................... 18
   - **2.4** Success-tree approach .................................................................................................. 22
   - **2.5** Fast construction of a model layout ............................................................................... 23
   - **2.6** Specification of model parameters ............................................................................... 24
   - **2.7** Applications to large scale events ............................................................................... 25

**3** SOTM IMPLEMENTATION OPTIONS .................................................................................... 26
   - **3.1** Motivation for a new method for the implementation .................................................. 26
   - **3.2** Petri nets ....................................................................................................................... 26
   - **3.3** Discrete event simulation ............................................................................................. 30
   - **3.4** Agent modelling ........................................................................................................... 34
   - **3.5** Conclusions .................................................................................................................. 37

**4** AGENT MODELLING ............................................................................................................. 37
   - **4.1** Dynamic Process Integration Framework (DPIF) ............................................................ 37
   - **4.2** Agent-based SOTM simulator in DPIF .......................................................................... 39

**5** HUMAN FACTORS ................................................................................................................... 42
   - **5.1** Performance Shaping Factors (PSF) .............................................................................. 42
     - **5.1.1** Example based on the HRA method SPAR-H ......................................................... 43
   - **5.2** Example of HEP/HR calculation .................................................................................. 45
   - **5.3** PSF correlations ............................................................................................................ 46
6  EVENT TREES (ET) & SEQUENCES ASSESSMENT ......................................................... 47
   6.1  EVENT TREES AND FAULT TREES ................................................................. 47
   6.2  BASIC FAILURE MODES MODELLING ........................................................... 47
   6.3  TOP EVENT MODELLING .................................................................................. 48
   6.4  EVENT TREE AND TOP EVENTS SEQUENCES .............................................. 48
   6.5  METHODS AND NUMERICAL TECHNIQUES .................................................. 49

7  CONCLUSIONS ...................................................................................................... 50

APPENDIX A: INTEGRATION OF DPIF INTO THE INTEGRATED PREDICT DECISION
SUPPORT TOOL (IPDT) .................................................................................................. 61
EXECUTIVE SUMMARY

This document describes the outcomes of the PREDICT Task 3.3 which focuses on the modelling of response actions and organizations. This is a part of the “incident evolution framework”, developed in WP3, which aims at developing a methodology for understanding the incident evolution based on different kinds of data, including results of computational modelling and simulations. Crisis management decisions require the estimation of the responding organization’s capability to perform the planned activity. Such a capability has two aspects: temporal response (when will the response take place) and reliability (will it take place, and how well).

Based on earlier studies and experiences about detailed event-based models of organization’s temporal response in the field of fire services, a new methodology called Stochastic Operation Time Modelling (SOTM) was recently developed under the Finnish national research programme for nuclear power plant safety to estimate the uncertainty related to human factors. This model has been further developed as part of PREDICT Task 3.3 to be used later by the PREDICT tool suite. SOTM is based on the assumption that human operations can be described as time delays and possible additional delays due to unexpected factors. This assumption is supported by empirical measurements of rescue services operation times. In this deliverable, a mathematical description of the model is presented and a generalized methodology for the construction of the model is developed. The methodology consists of modelling steps and timeline charts to visualize the action and communication processes. A set of related methodologies for the specification of the SOTM layout and parameters are presented.

A methodology for the estimation of the influence of human factors on individual error probabilities, using the concept of performance-shaping factors (PSF), is suggested. PSFs are capable of quantifying human reliability. PSFs identify a set of factors that are related to human performance: stress, action type, experience, time available, places where actions are taken, procedures, training, human machine interface (HMI), and teamwork. A PSF contains a set of multipliers that are used to compute Human Error Probability (HEP). Some other methodologies (Event Trees (ET) & Sequences Assessment) that can be used to support model specification are also presented.

There is a strong motivation for finding more universal methodologies for the implementation of the model and therefore Petri nets, discrete event simulation and agent modelling were investigated. Based on the applicability and availability of the modelling options in the PREDICT simulation tool development suite, agent modelling was chosen for further investigation. The Dynamic process integration framework (DPIF) seems to be well suited to implement an agent-based SOTM simulator to perform a stochastic analysis to obtain the total time delay of a response operation. Analysis clearly showed several benefits in setting up these types of simulations. Since the simulation has a modular setup, it can easily be changed by adding or removing different actors. Additionally, different time delay agents can be reused in other types of simulations, which will reduce the time to develop the stochastic simulation.
1 Introduction

“WP3 aims at developing a generic methodology for understanding the incident evolution and thus improving the capability to mitigate potential cascading effects. The overall objective of WP3 is to describe at technical and mathematical level, how the evolution of the incident can be observed, predicted and communicated. The incident in these contexts covers temporal, spatial and organizational aspects of the crisis. The outputs of WP3 will be used by the tool development work packages” (DoW p.14).

Task 3.1 created a methodology for the identification of the possible cascading effects as well as recognizing the dependencies between critical infrastructures (CIs). Also the means to quantify the likelihood of the cascading effects was developed to provide the decision support system (DSS) with the necessary information on the relevance. The main goal of Task 3.2 was to specify the physical and systemic models of threats related to the incidents, and to put predicted effects on the same timeline in order to create a common picture of the situation. As a result, the urgency of various branches of the incident could be determined, and the consequences of the decisions made concerning the allocation of resources could be demonstrated.

The main goal of Task 3.3 was to create “models for the organization’s response and communication” (DoW page 15) for the planning of the optimal use of available resources for the prevention of failures of critical infrastructures taking into account their interdependencies and for the mitigation of the consequences of the failures. In this task, a methodology was developed to estimate time dependent probability of the response actions to be in time for preventing or mitigating adverse effects of the accident.

The methodology to illustrate the “incident evolution on a timeline” was demonstrated in the application example of D3.2 for the flooding scenario (Figure 1. The relevant CIs in the accident scenario were the following:

- Roads A15, A27 and A2
- Railway (Betuwelijn)
- Drinking water supply plants
- Gas transport (compression) station and transport pipes
- Electrical supply to the area composed of substations and power line
- Telecommunication network
- An additional CI that reflects the crisis management (CM) capability in the cells of a hexagonal grid over the area studied
Figure 1 – Critical infrastructures on the hexagonal grid that is laid on the accident scenario map [PREDICT D3.2].

Table 1 shows the failure times of the CIs after taking into account the cascading effects based on the interdependencies between the CIs [PREDICT D3.2]. For instance, the electrical supply function needs both the substations and the power lines to be operational. When the first CI related to electrical supply is lost (i.e. power line in hex E2 at 30 hours), electrical supply in the whole area fails. This is reflected in the failure times of substations and power lines, all decreasing to 30 hours. Furthermore, since the telecommunication network is fully dependent on electrical supply, the telecommunication CIs are lost at the latest 30 hours after the initiating event. However, the telecommunication network is lost even earlier in certain hexes (D4, E1 and F3) due to the estimated vulnerability to water level.

In Task 3.3 we will develop a methodology to estimate the time that it takes to prevent this kind of failures or their consequences, to see if it is possible in time. If it appears that with a high probability the time needed to carry out the necessary measures will be longer than the time available, no resources should be used to these items. The methodology is applicable independently of the cause of the CI failure, i.e., whether it is due to a cascading effect or the inherent vulnerability of the CI to a threat. This document D3.3 (Methodologies for the operation time model lay-out and specification) focusses on **modelling of response actions and organizations**. Figure 2 shows schematically the evolution of cascading effects: the escalation of the situation, and the response to prevent or mitigate the consequences.
Table 1 – The failure times of CIs (in hours after the initiating event) after taking into account the cascading effects based on the interdependencies between the CIs [PREDICT D3.2].

<table>
<thead>
<tr>
<th>Hex ID:</th>
<th>Electrical power line</th>
<th>Electrical substation</th>
<th>Tele-comm. network</th>
<th>Road</th>
<th>Railway</th>
<th>Gas transport pipe</th>
<th>Gas transport station</th>
<th>Drinking water supply plant</th>
<th>Crisis manag. capability</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>63</td>
<td>-</td>
<td>16</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>B1</td>
<td>-</td>
<td>-</td>
<td>30</td>
<td>-</td>
<td>-</td>
<td>16</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>B2</td>
<td>-</td>
<td>30</td>
<td>30</td>
<td>55</td>
<td>1</td>
<td>16</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>C1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>C2</td>
<td>30</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>16</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>C3</td>
<td>-</td>
<td>-</td>
<td>30</td>
<td>52</td>
<td>1</td>
<td>16</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>D1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>16</td>
</tr>
<tr>
<td>D2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>16</td>
</tr>
<tr>
<td>D3</td>
<td>30</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>-</td>
<td>16</td>
<td>-</td>
<td>16</td>
</tr>
<tr>
<td>D4</td>
<td>-</td>
<td>-</td>
<td>15</td>
<td>16</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>16</td>
</tr>
<tr>
<td>E1</td>
<td>-</td>
<td>-</td>
<td>16</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>16</td>
</tr>
<tr>
<td>E2</td>
<td>30</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>E3</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>2</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td>F1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>20</td>
</tr>
<tr>
<td>F2</td>
<td>30</td>
<td>-</td>
<td>-</td>
<td>11</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>F3</td>
<td>-</td>
<td>-</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>1</td>
</tr>
<tr>
<td>G1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>288</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>G2</td>
<td>-</td>
<td>30</td>
<td>30</td>
<td>179</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>G3</td>
<td>-</td>
<td>-</td>
<td>30</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>11</td>
</tr>
<tr>
<td>H1</td>
<td>-</td>
<td>-</td>
<td>30</td>
<td>250</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>H2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>191</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>H3</td>
<td>-</td>
<td>-</td>
<td>30</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>I1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>250</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
<tr>
<td>I2</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>200</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>30</td>
</tr>
</tbody>
</table>
Making decisions on the actions of crisis management requires the estimation of the responding organization’s capability to perform the planned activity. This capability has two aspects: temporal response (when will the response take place) and reliability (will it take place and how well will it take place). There are some examples in the past, where detailed event-based models of organization’s temporal response have been developed. For the fire services, the Fire Brigade Intervention Model (FBIM) [Buckley et al. 2000] and the Fire Evaluation and Risk Assessment system (FIERA) [Benichou et al. 2002] were developed mainly for fire engineering design applications. The reliability of the human response, in turn, has been studied in the context of the Probabilistic Risk Assessment (PRA) of nuclear facilities, where the discipline has been called Human Reliability Analysis (HRA) [Kolaczkowski et al. 2005]. HRA models make use of so-called Performance Shaping Factors (PSF) for coupling the operating conditions to the probability of human errors.

A methodology called Stochastic Operation Time Modelling (SOTM) [Hostikka et al. 2012] has recently been developed under the Finnish national research programme for nuclear power plant safety. It can be used to build a model taking into account distributions of the time delays of successful unit operations, as well as probabilities and consequences of deviations from the optimal operation. The consequences are described as additional time delays. The model can be used to simulate the possible realizations of the human operations in Monte Carlo –manner, which is both probabilistic and quantitative method. The result is a distribution of time to reach the intended goal, which can be compared with the temporal progression of the crisis situation.

For the estimation of response actions in a multi-agency situation, SOTM has several benefits [Kling et al. 2013]:

- It provides a quantitative method to take into account the human factors and their influence on the development of the situation and the ability of response operations to get the situation under control,
- It is applicable to different crisis situations,
- It can provide a comprehensive model for interdependencies of various phenomena and activities of different actors in crisis situations,
It can reveal bottlenecks and contradictions in processes and organizations, thus eliminating confusion that is often related to complex situations, and

- It enables time-dependent simulation of complex processes including deviant behaviors.

Technically, SOTM is a flow chart-based Monte Carlo model describing human actions as time delays. In crisis situations, the response operations are typically running against the clock; time is a critical factor when trying to catch the advance of the cascading effects. SOTM includes approaches for identifying the critical steps and the resources needed to complete the missions and helps to assess the uncertainties due to human factors and complicated dependencies between various actors. The novelty of the methodology is related to the combination of temporal and reliability aspects: errors or failures in human operations are not considered as final stages but rather as additional time delays of a specific action.

The methodology will be further developed in Task 3.3 to take into account the uncertainty associated with human actions and the consequences of the unwanted behaviors which cause delays and risk of failure in the response operations. It will also be discussed how it can be applied in large scale events with possible cascading effects.

As required in the description of deliverables (PREDICT DoW - Workplan table - Page 16), D3.3 will present technical and mathematical descriptions of the model, and propose means to define the model structure for the scenarios and to collect the model inputs from the end-users. Moreover, D3.3 will cover other issues of the description of Task 3.3 “Models for the organization’s response and communication”.

The work has been divided into the following subtasks:

- A set of methodologies for the specification of model structure and parameters related to human operations were developed:
  - for the construction of a stochastic operation time model,
  - for the determination of a set of inter-connected actions that are necessary for a successful operation,
  - for the determination of time delays of human operations,
  - for the error consequence estimation,
  - for the estimation of the influence of human factors on individual error probabilities, using the concept of performance-shaping factors (PSF).

- Different options for the model implementation were investigated. Agent modelling was further investigated because of its applicability and availability for the PREDICT simulation tool development suite.

The modelling process as a whole is illustrated in Figure 3.
2 Stochastic Operation Time Model (SOTM)

2.1 Basic idea and purpose of the methodology

Stochastic Operation Time Modelling (SOTM) is a methodology for estimating the uncertainty related to human factors. In SOTM, human operations are described as time delays, representing the time it takes to perform a specified task. Time delays in this context are related to response actions. Response actions are related to possible failures that we are trying to prevent. When a crisis begins, we usually have an idea of how the situation will proceed, how the critical infrastructures will be threatened and what are the exact failures that will take place if we are not able to prevent it or mitigate its consequences.

In Task 3.2, a threat quantification modelling approach was developed to illustrate the progress, influencing factors and potential cascading effects of accident scenarios. The focus was in the revelation of crucial vulnerabilities and dependencies leading to cascading effects, not in the actual response actions to intervene in the crisis. In Task 3.3, we are looking at the progress of response actions and their possibilities to prevent and mitigate the consequences.

Usually the response organizations have a pre-designed plan of actions against various issues. When they get the information about the incident they know at once what they are supposed to do in the situation. In some situations that are often repeated, like fires, the procedures may be very precisely defined and there may be a lot of statistical information about the operation times, or even about the time lapse that an individual task may take. On the other hand, there are lot of situations where the
actions must be planned and decided in the situation itself. In large incidents with domino effects there are both kinds of features, and several response organizations need to co-operate, which causes even more variation in the way of action.

Even if we would know at once what we should do it will take time before the planned operation has been performed. This is what we mean by time delay. The planned operation may consist of various steps, each having their own features and duration. Typically, when the members of the emergency response organisation have been informed about the situation (time delay 1), they gather to a pre-agreed place (time delay 2), travel together to the place of action (time delay 3), make there a situation assessment (time delay 4), and a plan about the actions (time delay 5), and finally carry out the measures required by the situation (time delay 6). The total time delay is then sum of the delays of the separate tasks.

It may happen that on their way to the place of action there is a traffic jam, which happens with a probability of 10% and causes an additional time delay of 10–20 minutes. If we would make a probabilistic Monte Carlo simulation about the situation, this would mean that in one of ten realizations we would add this additional time delay to the total time delay. We could describe the additional time delay by a random variable with a uniform distribution U (10, 20).

Co-operation with other responders may also cause additional time delays, which are dependent on the time delays of the other organizations. On the other hand, in some situations same measures may be performed by alternative actors having different time delays in their operations.

When trying to estimate the total time delay of an operation, we need to take into account:

1. All the time delays related to the separate tasks of the planned activities of the operation.
2. All the additional time delays related to unexpected factors, which may take place with a certain probability.
3. Possible co-operation or overlapping activities, which can either extend or shorten the operation time.

In this context, an operation means activities which aim at preventing a failure or mitigating its consequences. An operation time is the total time delay of an operation. An operation time becomes stochastic, when we use random variables and Monte Carlo simulation to produce a large amount of realizations of an operation time. When using this methodology we get a solution which is a probabilistic distribution of the total time delay of an operation.

To be able to carry out a Monte Carlo simulation about the situation we need a lot of data. Sometimes there is statistical data available, but sometimes we need deterministic models or expert estimation to get the parameter values for the simulation. For example, to be able to determine how long it takes for a group of responders to travel by a vehicle from one point to another, we can use a deterministic model “time = distance x speed”, where the speed is a random variable. To be able to determine the time delay of the situation assessment, we need an expert estimation.

The purpose of the methodology is to estimate operation times for the operations that are needed in a crisis situation to prevent the failures or mitigate their consequences. In large scale events that include cascading effects, there may be different kinds of failures or consequences to be prevented or mitigated. This means that there are also different kinds of operations, several groups of responders
etc. and also several operation time models needed to estimate the time delays. Operation time models may be separate or they may be connected to each other.

The generic methodology for understanding the incident evolution will then be the following:

- Development of the situation will be evaluated by different means ranging from expert judgement to complicated physical models.
- Cascading effects will be taken into account using the methodologies developed in Tasks 3.1 and 3.2.
- Different predictions are then put on the same time line using the methodology developed in Task 3.2.
- Models for the organization’s response and communication will be built based on the methodology developed in Task 3.3.
- Estimated operation times will be compared with the estimated failure times to see if there is enough time to carry out the operations.
- Based on advanced knowledge, the resources can be allocated to those targets where the efficiency is the best.

Modelling and simulation are complex issues best utilized when there is enough time available, that is, in the context of planning and training. Some features of the methodology may also be used real time, during the incident, but only among advanced users, that are familiar with the methodologies already in advance.

### 2.2 Mathematical description of the stochastic operation time model

In this paragraph, “the mathematical description of the Stochastic Operation Time Model” will be presented (PREDICT DoW - Workplan table - Page 16). The model includes “a methodology for the error consequence estimation” (PREDICT DoW - Workplan table - Page 15).

The methodology is based on the assumption that human operations can be described as time delays and possible additional delays. This assumption is supported by empirical measurements of rescue service operation times [Tillander 2004; Emergency services college 2012; Kokki 2014]. The results of the measurements indicate that the duration of most operations has a distribution with a long tail caused by relatively unlikely occurrences of deviations, such as a radio communication problem or a failure of a piece of equipment needed in the operation. These distributions are best modelled as combinations of two different mathematical distributions, one for the successful operation and another for the deviation.

When trying to prevent a specific failure of a CI or a CI element, the response procedure is divided to action steps. Each action step \( i \) is described as a time delay \( \Delta t_i \), which has a probability distribution. If there is statistical data available about the length of the time delay, it can be used to determine the distribution and its parameters. Sometimes there is no data available and the distribution needs to be determined by expert assessment. In such cases some simple distributions, for example uniform distributions \( U(\Delta t_{\text{min}}, \Delta t_{\text{max}}) \) or triangular distributions \( T(\Delta t_{\text{min}}, \Delta t_{\text{max}}, \Delta t_{\text{peak}}) \), can be used. In some other cases the time delays can be calculated deterministically, for example walking a certain distance.
at a certain speed, but even then some of the parameters may be random variables (e. g. walking speed). The occasional delays $j$ due to human and hardware errors or by other unfavorable circumstances are described by probabilities $p_j$ and additional delays $\Delta t_j$. During the Monte Carlo simulation, realizations are taken from the distributions. The total time delay for reaching node $x$ can be calculated according to a procedure specific formula:

$$
\Delta t_x = \sum_{i=1}^{n} \Delta t_i + \sum_{j=1}^{m} k_j \Delta t_j
$$

where $k_j = 1$, with a probability $p_j$,

$$
\text{and} \quad k_j = 0, \text{ with a probability } 1 - p_j
$$

In the equation, $n$ is the number of action steps, which have time delays $\Delta t_i$ and $m$ is the number additional time delays $\Delta t_j$, which happen with a probability $p_j$.

### 2.3 Modelling steps

*In this paragraph, “a generalized method for the construction of a stochastic operation time model” will be developed (PREDICT DoW - Workplan table - Page 15).*

For each of the possible failures, a model needs to be built to describe the needed response actions. The best experts to give the data for the modelling are the responders, who know their procedures and the related time delays. The main modelling steps are the following:

1. Define a threat → failure → consequence scenario for a specific CI element.
2. Define a scenario for the response actions to prevent the failure.
3. Determine the actors and their actions as well as the connections between them.
4. Analyze the response action steps and possible deviations from the planned procedure.
5. Describe the time delays (distributions) of the response action steps and additional delays (probabilities and distributions) of deviations.
6. Carry out a Monte Carlo analysis to get a probability distribution for the total response time.

### 2.3.1 Example case

When applied, for example, to the railway emergency scenario [PREDICT D7.1; Page 18] we get the following result:

1. A train loaded with hazardous chemicals and liquid gas derails close to the tunnel in Rheinartzkehl (Figure 4). Two tank cars loaded with liquid gas set fire. The other cars are leaking dangerous substance and there is a high risk that the fire will spread to the other wagons.

2. The goal of the rescue service operation is to provide sufficient cooling capacity in order to prevent a BLEVE (boiling liquid expanding vapor explosion) incident. A successful operation
requires that there are enough resources (staff, equipment, water) in use and that the resources are available fast enough. In this situation it is important to clarify what liquid is in question because it affects the selection of equipment and the course of action of the first responders. Before the operations can be started in the crisis area, the rail traffic has to be stopped and a so-called emergency grounding must be made to the traction power cable.

3. The fire which takes place on a railway concerns multiple actors, some of them immediately and the others at a later stage:

a. One way or another, the information of the situation is transmitted to the following actors: Traffic contractor, traffic control, power control center, emergency center, fire and rescue services, police, railway transport center, the traffic contractor’s clearing group, maintenance services, electrical engineering services, accident investigation board, and traffic safety office.

b. Those action steps and connections that form the critical path of the situation are the following:

   i. **Engine driver:**
      - detects the fire
      - makes the situation assessment,
      - communicates with traffic control and emergency center
      - takes care of the emergency grounding, if there is an equipment available
   
   ii. **Traffic control center:**
      - stops the rail traffic
      - communicates with power control and emergency center
      - takes care of the cargo clearance
   
   iii. **Power control center:**
      - switches off the voltage
      - communicates with electrical engineer
   
   iv. **Emergency center:**
      - communicates with fire service and traffic control
   
   v. **Fire service:**
      - departure, driving to the place, inquiry
      - ensures the traffic-stop, power-off and the type of the burning cargo
      - planning of rescue strategy
      - takes care of emergency grounding, if there is an equipment available
      - starts cooling/suppression
   
   vi. **Electrical engineer:**
      - grounding when not done by engine driver or fire service
   
   vii. Before the first responder operations at the accident location can be started, a so-called emergency grounding must be made to the driving cable. The
emergency grounding will be made by the engine driver, an electrical engineer or the fire service that has been trained to the task.

viii. Before performing the emergency grounding, the power control center of the railway will disconnect the traction power of all the rail tracks on both sides of the accident.

ix. An extended alarm will be given to get more first responder units to the place when it appears that one unit is not enough for the task (usually the extent of the situation is not immediately clear).

4. Each of the critical actors (engine driver, traffic control, power control, emergency center, fire service, electrical engineer) is asked to explain their action steps and related procedures in detail.

5. Each of the critical actors is asked to define the time delays of their actions as well as possible deviations and their likelihood. It may be possible to get part of the information from literature or statistics or arrange an exercise and/or measurements to get data for the modelling.


The action and communication processes can be visualized by a timeline chart (Figure 5), where time goes from left to right and information from top to down. An example concerning the railway emergency scenario is shown in Figure 6.
Figure 5. A generalized timeline chart for a situation where the actors are trying to prevent the consequences of a threat. Time goes from left to right and information from top to down.

Figure 6. An example of a timeline chart for a fire of a train loaded with hazardous chemicals and liquid gas shows the action and communication processes of the response organizations. The action procedures go from left to right and the communication from top to down. In a successful operation, the control is achieved before the BLEVE (boiling liquid expanding vapor explosion) [Hostikka et al.; 2012; Kling et al; 2012].

When the model structure has been created for “normal” situations, i.e. for situations where everything follows the planned procedures, it is time to look at the model again and try to figure out what could go wrong and what is the probability of the deviation. The estimation of the influence of human factors is explained in Chapter 4. In addition to human factors, other issues e.g. machines, weather, traffic, failures in other CI (cascading effects) etc. may interfere with the planned activities. This kind of unwanted behavior can be taken into account in the model using probabilities and additional delays as
explained in paragraph 2.1. The consequences are then extra delays in the response procedures, which may increase the risk of failure.

2.4 **Success-tree approach**

In this paragraph, “a set of interconnected actions that are necessary for a successful operation will be determined using a success-tree approach” (PREDICT DoW - Workplan table - Page 15).

Sometimes the situation can be quite complicated and have different actors trying to do the same thing, or some actions cannot be started before some other actions have been completed by other actors. This kind of situation can be visualized by a “success tree”, which is presented in Figure 7. An example concerning the railway scenario is shown in Figure 8. MIN operation in the figures means that it is enough that the required action is performed by one actor (for example detection or grounding), then the shortest time delay is realized. MAX operation means that several actions need to be finalized before the next procedure can be started (for example the cooling/suppression can be started first when the fire service has arrived, grounding has been made, and the nature of the fuel has been identified), then the longest time delay is realized.

**Figure 7. A generalized success-tree for a response operation. Time goes from top to down. One detection is enough for the situation to be detected, i.e. the minimum of the detection times determines the total detection time. Certain actions need to be carried out and certain states (for example certain amount of personnel and equipment need to be on the place) need to be received before an operation can start, i.e. the maximum of the time delays determines the total time delay.**
Figure 8. An example of a success-tree for a fire of a train loaded with hazardous chemicals [Hostikka et al.; 2012; Kling et al; 2012].

2.5 Fast construction of a model layout

In this paragraph, we develop a procedure “to enable a fast model layout during the incident response” (PREDICT DoW - Workplan table - Page 15).

The rapid construction of the model is possible by two different methods 1) forward method and 2) backward method. The forward method is based on the timeline chart (Figure 5):

- Draw a separate timeline chart for each threat → failure → consequence scenario
- For each of the timeline charts, do the following steps:
  - Define how the threat will be detected and what kind of response operation is needed to prevent the failure and/or consequences.
  - Define the actors that are needed for the operation.
  - Draw the timelines of the threat and the actors.
  - Draw the arrows to show the information flow between the actors.

The backward method is based on the success tree (Figure 7):

- Draw a separate success tree for each threat → failure → consequence scenario
For each of the success trees, do the following steps:
- Define the successful response operation to prevent the failure + consequences
- Build the success tree starting from the bottom and proceeding upwards:
  - What is the state that has to be achieved to prevent the failure and/or consequences?
  - What are the earlier steps that have to be performed to reach this state?
  - Define the earlier states and steps to reach these states.
  - Define the MIN and MAX operations if needed.

Based on these two charts, the operation time model can be built.

### 2.6 Specification of model parameters

In this paragraph, “a set of methodologies for the specification of model parameters related to human operations” (PREDICT DoW - Workplan table - Page 15) will be presented.

There are several ways to specify the duration of specific actions and/or procedures:

1. **Measurements**
   - The duration of the response procedure steps can be measured during exercises or in experimental setups.

2. **Statistical data**
   - In some countries, the realization of response operations is carefully monitored and the information is collected in databases.

3. **Deterministic calculation**
   - The duration of certain operations can be deterministically calculated. For example, driving a certain distance by a car at a certain speed.

4. **Numerical simulation**
   - In some cases, it may be possible to use numerical simulation to determine the duration of certain operations, for example, the evacuation of a building.

5. **Expert assessment**
   - If none of the above mentioned methods is available or the assessment has to be carried out very quickly, the only method to be used is expert assessment. In this case, there are typically three ways of estimating the required time for the performance of individual measures:
     - Constant: « The action takes 10 minutes »
     - Uniform distribution: « The action takes 5 to 15 minutes »
     - Triangular distribution: « The action takes typically 10 minutes, but it may vary five minutes to one direction or another »

If there are several possibilities on how the situation may proceed, event trees can be used to define scenario chains and/or distributions of possible situations. As a result of an event tree, the probability of each possibility can be determined. The use of event trees is explained in paragraph 5.1.
2.7 Applications to large scale events

Although the presented examples are small, the size of the event does not limit the use of the methodology. In case of a large event the situation will be split into smaller pieces, as was shown in the example that was presented in Figure 1 and Table 1 shows the failure times of the CIs after taking into account the cascading effects based on the interdependencies between the CIs [PREDICT D3.2]. For instance, the electrical supply function needs both the substations and the power lines to be operational. When the first CI related to electrical supply is lost (i.e. power line in hex E2 at 30 hours), electrical supply in the whole area fails. This is reflected in the failure times of substations and power lines, all decreasing to 30 hours. Furthermore, since the telecommunication network is fully dependent on electrical supply, the telecommunication CIs are lost at the latest 30 hours after the initiating event. However, the telecommunication network is lost even earlier in certain hexes (D4, E1 and F3) due to the estimated vulnerability to water level.

In Task 3.3 we will develop a methodology to estimate the time that it takes to prevent this kind of failures or their consequences, to see if it is possible in time. If it appears that with a high probability the time needed to carry out the necessary measures will be longer than the time available, no resources should be used to these items. The methodology is applicable independently of the cause of the CI failure, i.e., whether it is due to a cascading effect or the inherent vulnerability of the CI to a threat. This document D3.3 (Methodologies for the operation time model lay-out and specification) focusses on modelling of response actions and organizations. Figure 2 shows schematically the evolution of cascading effects: the escalation of the situation, and the response to prevent or mitigate the consequences.

Table 1. With the methodology that was developed in Task 3.2 the failure times of CIs were shown in the same time line. The cascading effects based on the interdependencies between the CIs were already taken into account in the failure time estimates. A separate operation time model will be created for each of these failure time estimates to see if it is possible to prevent the failures or their consequences in time. No resources should be used for those measures, where the time needed to carry out the necessary operations will be longer than the time available. With a stochastic operation time model the response time estimates will be time distributions and we get a probability for response actions to be in time. In principle it is possible to calculate a probability of successful response for each of the time estimates in Table 1 shows the failure times of the CIs after taking into account the cascading effects based on the interdependencies between the CIs [PREDICT D3.2]. For instance, the electrical supply function needs both the substations and the power lines to be operational. When the first CI related to electrical supply is lost (i.e. power line in hex E2 at 30 hours), electrical supply in the whole area fails. This is reflected in the failure times of substations and power lines, all decreasing to 30 hours. Furthermore, since the telecommunication network is fully dependent on electrical supply, the telecommunication CIs are lost at the latest 30 hours after the initiating event. However, the telecommunication network is lost even earlier in certain hexes (D4, E1 and F3) due to the estimated vulnerability to water level.

In Task 3.3 we will develop a methodology to estimate the time that it takes to prevent this kind of failures or their consequences, to see if it is possible in time. If it appears that with a high probability the time needed to carry out the necessary measures will be longer than the time available, no
resources should be used to these items. The methodology is applicable independently of the cause of the CI failure, i.e., whether it is due to a cascading effect or the inherent vulnerability of the CI to a threat. This document D3.3 (Methodologies for the operation time model lay-out and specification) focusses on modelling of response actions and organizations. Figure 2 shows schematically the evolution of cascading effects: the escalation of the situation, and the response to prevent or mitigate the consequences.

Table 1.

The success tree approach was developed to enable fast model layout during the incident response. The success tree shows the critical path of the needed actions to prevent specific consequences of domino effects. Detailed modelling, which also produces probabilities as well as time distributions may need too much data to be practical at run-time during an incident. It can be used for training, planning and analyzing the incidents, thus providing results that are useful also during incidents.

3 SOTM implementation options

In this chapter “options for the model implementation will be investigated” (PREDICT DoW - Workplan table - Page 15).

3.1 Motivation for a new method for the implementation

By now SOTM has only been used for fire applications and the models have been created by using an Excel-based pre- and postprocessor PFS (Hostikka et al. 2003); the latest version can be uploaded here: https://github.com/atpaajan/pfs. PFS (Probabilistic Fire Simulator) includes a framework for Monte Carlo simulations and post-processing of the results. Previous experiences have shown that the model is capable of revealing the bottlenecks in an organization’s response and the most error-prone operations. It can also give quantitative information on the response. However, there is a strong motivation for finding more universal methodologies for the modelling. For this purpose, different implementation options (Petri nets, Discrete event simulation, Agent modelling) were investigated.

3.2 Petri nets

A Petri net is a mathematical representation of a system, originally developed by Petri [1962]. It is composed of a set of places P and a set of transitions T. A Petri net graph (Figure 9) is a representation of a Petri net as a bipartite directed multigraph. It is a directed multigraph, because it allows multiple directed arcs from one node of the graph to another. The nodes of the graph can be partitioned into two sets (places and transitions), and each arc is directed from an element of one set (place or transition) to an element of the other set (transition or place), thus it is a bipartite directed multigraph [Peterson 1981]. Usually the places are drawn as circles, and transitions as bars or boxes. Places represent actions or conditions and transitions represent events. Tokens (black dots) reside in places and represent the truth of the condition or the action associated with the corresponding place
If the number of tokens in a Petri net is large, numbers can be used instead of dots. A marking $\mu = (\mu_1, \mu_2, \ldots, \mu_n)$ is an assignment of tokens to the $n$ places of a Petri net.

The execution of a Petri net is controlled by the number and distribution of tokens, which reside in the places and control the execution of the transitions of the net. A transition fires by removing tokens from its input places and creating new tokens to its output places. A transition may fire if it is enabled, i.e. if the input places have at least as many tokens as there are arcs from the place to the transition. The state of a Petri net is defined by its marking.

A Petri net model can be used to represent and communicate the design of a concurrent system. The modelled system can then be analyzed to get important insights into the behavior of the modelled system. Petri nets may have different properties, e.g. [Peterson 1981]:

- **Safeness** – A Petri net is safe if all places in the net are safe, i.e. the number of tokens in the places never exceeds one.
- **Boundedness** – A place is $k$-bounded if the number of tokens in that place cannot exceed $k$; a Petri net is bounded if all places are bounded.
- **Conservation** – It is required that the total number of tokens in the net remains constant.
- **Deadlock** – A deadlock in a Petri net is a transition (or a set of transitions) which cannot fire.
- **Liveness** – A transition is live if it is not deadlocked.

Petri nets can be used as an auxiliary analysis tool for a system (Figure 10). In this approach conventional design techniques are used to specify a system, which is then modelled as Petri net and analyzed. The design is then modified and analyzed again until the analysis does not reveal any unacceptable problems. Another possibility is that the entire design and specification process is carried out in terms of Petri nets and when the design is error-free, the problem is transformed into an actual working system.
Petri nets were originally developed for computer hardware and software [Peterson 1981], but they can be directly applied to the modelling of a large number of other systems. PERT charts [Malcolm et al. 1959] have long been used in the planning and scheduling of large projects. A project consists of a large number of activities; some of them must be completed before other activities can start. In addition a time is associated with each activity indicating the amount of time it will take. A PERT chart can easily be converted to a Petri net, but a Petri net does not provide any timing information which is useful e.g. for determining minimum project completion time. Production systems, assembly lines as well as chemical systems can be modelled as Petri nets. Another suggested use for Petri nets has been the modelling and analysis of communication protocols [Merlin 1975].

Bulitko & Wilkins [2002] presented a formalism called Time Interval Petri Nets (TIPNs). TIPNs are effective for qualitative simulation of concurrent temporal processes as they "explicitly support temporal uncertainty in states and state changes, the context of processes simulated, modularity of the design, visual debugging and development, and adjustable degree of simulation fidelity". When the framework was applied to a real-time decision-making domain of ship damage control and a TIPN-equipped decision-making system was tested in a large exercise involving 160 simulated ship crisis scenarios, it showed a 318% improvement over Navy officers by saving 89 more ships.

Petri nets are widely used as a tool for analyzing safety and reliability of complex systems [Adamyan & He 2002]. Petri net modelling provides an ability to assess the quality and reliability impacts caused by combination of unplanned failures and their sequences. Combined failure modes can be analyzed to predict their potential severity and estimate their probability to see where to put effective means to prevent the impact of the failures. The Petri nets where random delays are exponentially distributed are referred as stochastic timed Petri nets (SPNs) [Zhou& Zurawski 1995; Mollow 1982 & 1985; Florin et al. 1991]. General software packages are available for solving SPN models [Chiola 1987; Ciardo 1989].

Vernez et al. [2003] discuss the uses and application perspectives of Petri nets in the fields of risk analysis and accident modelling. Dynamic systems modelling through Petri nets have encountered a large success and several developments have been made that are of utmost interest for safety related applications. These include:

- Coloured Petri nets (CPNs), where complex properties may be attributed to the tokens by colors
- Timed Petri nets (TPNs), where time logic is taken into account by the use of firing times or duration
- Stochastic Petri nets (SPNs), where random distribution functions are used to describe delays
- Generalized Stochastic Petri nets (GSPNs), which also include immediate transitions with negligible delay
- Stochastic Reward nets (SRNs), which include enabling functions, timed transition priorities, variable cardinality arcs, halting conditions and reward rates

According to Vernez et al. [2003] the possible translations of key concepts or functions used in safety sciences into the PN formalism suggest tremendous possibilities: Either qualitative aspects of accident mechanisms or quantitative data, such as time logic or reliability calculations, may be simulated.

Wang et al. [2016] present an integrated hazard identification method based on a hierarchical coloured Petri net. The approach is called System-Theoretic Process Analysis Based on Formalization Model (BFM-STPA). The hierarchical control structure models of the socio-technical system are established through CPNs due to its strong description and execution ability. The hazards can be identified based on the CPN models following a guiding procedure and finally, an integrated hazard log can be derived for further hazard analysis and safety-guided design.

Zhou & Reniers [2016] present a simulation analysis for emergency response to multiple simultaneous large-scale fires. Their approach is called Timed Coloured Hybrid Petri Net (TCHPN). Petri-net can deal with time in several ways, including timed transition [Aybar & Iftar 2008], timed place [Mejia & Odrey 2005] and timed arc [Valero et al. 1999]; in this paper, timed transition is utilized. A token's color determines which transition is enabled by the token. Transitions represent actions in the emergency response and the delay time of a transition indicates the executing time of the corresponding emergency response action. Based on the TCHPN model, the fire fighter distribution strategies during the emergency response to multiple fires can be analyzed. Through the analysis it can be obtained which strategy is better under certain conditions. This can help improve the preparedness under different conditions.

Nývlt at al. [2015] have modelled and analyzed complex accident scenarios by Stochastic Petri Nets. In their approach the accidental scenario is described as a net of interconnected blocks, which represent parts of the scenario. The scenario is divided into parts, which are then modelled by Petri nets. Every block can be easily interconnected with other blocks by input/output variables to create complex ones. An application of the method is shown in two case studies: 1) Collision of an offshore installation with a vessel, and 2) A gas leak from the process plant of an offshore installation. The use of variables for firing of transitions and connection of different parts of a model makes it less complicated and readable also for non-experts. On the other hand the analyst has to be trained in the PNs to be able to construct the model.

Talebberrouane et al. [2016] have used generalized stochastic Petri nets (GSPN) for the availability analysis of safety critical systems. The GSPN is a Petri net with probabilistic analysis using Monte Carlo simulation. GSPN were used with predicates and assertions. The predicates or guards as defined by International Electrotechnical Commission [IEC 2010] are any formula which may be true
or false, validating transitions. Assertions are the mathematical variables that receive predefined changes as firing consequences. The behavior of these mathematical variables can be traced and used as outcomes of PN modelling. The GSPN formalism is applied to a flare system incident reported in The John Zink Hamworthy Combustion Handbook [Baukal 2012]. It is observed that GSPN provides a robust and reliable mechanism for accident scenario analysis. It provides additional information such as events’ frequencies at operating, failing modes and expected occurrence timings and durations resulting from different complex sequences. PN modelling provides complete dynamic system simulation; it also allows the modelling of a complex system in modular fashion into structurally separated sub-systems. Mathematical variables through guards and assignments ensure the relations between sub-systems. The modular approach enables more traceability while avoiding huge and complex models.

### 3.3 Discrete event simulation

Discrete Event Simulation (DES) is a dynamic simulation technique where changes in the system are represented over time. It creates a queue of events that affect the system states. DES has been applied successfully in a wide range of business and manufacturing applications. Typical DES software enables visual interactive simulation where changes in the system are animated and users can interact during the simulation [Alrabghi & Tiwari 2015].

According to Sachidananda et al [2016] the challenges in the model development of e. g. a manufacturing system include the following:

- Identify scope and assumptions
- Define simulation objectives
- Determine modelling entities, variables, attributes, resources
- Determine modelling constraints (process times, CIPs, changeovers, etc.)
- Collect simulation data
- Define key performance parameters
- Identify the elements that represent the system under study

The next step in model development is to identify the elements in the simulation engine that represent the system under study such as

- Entities (elements that flow through the system)
- Variables (elements that change over time)
- Attributes (the characteristics of elements)
- Resources (system assets, machines, conveyors, labor, buffers)

These are essential to build the simulation logic and to establish product and resource flows. The data input into the model development determines the accuracy of the simulation results.

DES models have also been developed for maritime transportation problems e. g. for:

- Balancing the allocation of manpower among shifts [Mattfeld and Kopfer 2003; Fischer and Gehring 2006],
• Assessing the ability of inland port facilities to cope with an increase in logistic volumes [Cortés et al 2007],
• Prediction and exploration of the terminal behavior under different operation decisions [Boschian et al. 2013; Bienwirth and Meisel 2010],
• Studying the impact of deepening the navigational efficiency of a river using some port performance measures [Almaz and Altiok 2011],
• Scheduling of the assignment of cars to parking rows [Cordeau et al. 2011],
• Estimation of the activity time for different handling equipment and different container types to choose the best one in container terminals [Cartenì and de Luca 2012],
• Evaluation of the impact of factors [Longo et al. 2013] such as
  o inter-arrival time between vessels
  o loading/unloading time
  o the total number of cars and trucks to be handled in the turn-around time
• Simulation of a port terminal in a three-dimensional virtual environment [Bruzzone and Longo 2013] and Longo et al. 2014,
• Investment planning in order to find the best cranes to cope with the demand of the next years [Lin et al. 2014],
• Assessing the daily operation decisions in a Ro-Ro terminal [Iannone et al. 2016].

Another typical application area for DES simulations is healthcare [Jun et al. 1999; Almagooshi 2015]. Since healthcare services are mostly dynamic and stochastic processes, discrete event simulation has been used to model and analyze healthcare processes. Caro et al. [2010] argued that discrete event simulation should be preferred to cohort Markov models for economic evaluations in healthcare.

DES simulation has been applied to healthcare problems e.g. for:

• Modeling and analyzing a physician clinic environment [Swisher et al. 2001] to provide high-quality, cost-effective medical care within a physician network setting.
• To improve radiation therapy planning processes [Werker 2009]. A simulation model of the radiation therapy (RT) planning process was constructed using the Arena simulation software, representing the complexities of the system. Several types of inputs feed into the model; these inputs come from historical data, a staff survey, and interviews with planners.
• For health policy design and decision making [Ramwadhdoebe et al 2009]. The steps in building a DES model was demonstrated using a real-world example, i.e., pediatric ultrasound screening for hip dysplasia. Questions such as referral schedule, number of ultrasound machines and type of screeners and how these entities interact were studied and a review of the statistical techniques appropriate to DES was provided. Discrete-event simulation appears to be a valuable tool in the policy maker’s armamentarium to analyze and understand complex healthcare systems and policy problems such as population screening.
• Utilizing simulation in the implementation of lean in healthcare [Robinson 2012]. This paper explores from a theoretical and an empirical perspective the potential complementary roles of DES and lean in healthcare. The aim is to increase the impact of both approaches in the improvement of healthcare systems.
• Evaluation of Medical Treatment Capability against Biochemical Terrorist Attacks [Wang et al. 2012]. A new method is proposed with two steps: 1) Building a model to calculate the number of victims arriving in hospital with Monte Carlo simulation and generating the victims-flow arriving in hospital. 2) Building another model to calculate the medical treatment capability based on the generated data from the first simulation.

• Improvement of a mental health clinic design [Kim et al. 2013]. DES simulation mirrored how the clinic currently operates. Hypothetical changes were made to the staffing to understand their effects on percentage of patients seen outside scheduled clinic hours and service completion time.

• Analysis of the diagnostic part of the stay of stroke patients in a stroke unit of a university hospital [Chemweno et al. 2014]. The patient flow is analyzed and the impact of potential changes in the capacity profile of test resources is investigated.

• Performance-driven design of an emergency department [Morgareidge 2014]. DES was used to optimize the care process and to design the space. DES was applied in three phases: master planning, process improvement and designing the new emergency department.

• Design and analysis of an outpatient orthopedic clinic performance [Baril et al. 2014]. The objective was studying the relationships and interactions between patient flows, resource capacity [number of consulting rooms and number of nurses] and appointment scheduling rules in order to improve an outpatient orthopedic clinic performance. Discrete event simulation was used to model outpatient flows.

• Incorporating the dynamics of epidemics in simulation models of healthcare systems [Nikakhtar & Hsiang 2014]. In this study, A discrete-event simulation model for a local community health clinic in Lubbock, Texas was developed with an additional level of uncertainty based on the dynamics of an epidemic. The susceptible-infected-recovery (SIR) process was developed for the model to generate epidemic patients for the model developed for the clinic.

• Evaluating the cost-effectiveness of the addition of omega-3 fatty acids to standard PN regimens in four European countries (Italy, France, Germany and the UK) from the healthcare provider perspective [Pradelli et al. 2014]. Using a discrete event simulation scheme, a patient-level simulation model was developed, based on outcomes from the Italian ICU (intensive care unit) patient population and published literature.

• A composite model for Chlamydia infection [Viana et al. 2014]. Two simulation approaches, discrete-event simulation (DES) and system dynamics (SD), are used together to address the sexually transmitted infection Chlamydia. The model shows how the prevalence of Chlamydia at a community level affects (and is affected by) operational level decisions made in the hospital outpatient department.

• Health economic evaluation of plasma oxyysterol screening in the diagnosis of Niemann–Pick Type C disease among intellectually disabled [Karnebeek et al. 2015]. Discrete event simulation was used to follow ID (infectious disease) patients through the diagnosis and treatment of NP-C, forecast the costs and effectiveness for a cohort of ID patients and compare the outcomes and costs in two different arms of the model: plasma oxyysterol screening and routine diagnosis procedure over 5 years of follow up.
Managing emergency department overcrowding via ambulance diversion [Lin et al. 2015]. A simulation model and a computer simulation program were developed. Three sets of simulations were executed to evaluate AD (ambulance diversion) initiating criteria, patient-blocking rules, and AD intervals. The crowdedness index, the patient waiting time for service, and the percentage of adverse patients were assessed to determine the effect of various AD policies.

Developing a multi-methodology framework to support facilitated simulation modelling in healthcare [Tako & Kotiadis 2015]. DES is combined with soft systems methodology (SSM) in order to incorporate stakeholder involvement in the study lifecycle. The framework consists of a number of prescribed activities and outputs as part of the stages involved in the simulation lifecycle, which include study initiation, finding out about the problem, defining a conceptual model, model coding, experimentation and implementation.

To study how a business game can be used jointly with discrete event simulation to test scenarios defined by team members during a Kaizen event [Baril et al 2016]. The aim was to allow a rapid and successful implementation of the solutions developed during the Kaizen. The approach was used to improve patients’ trajectory in an outpatient hematology–oncology clinic. Patient delays before receiving their treatment were reduced by 74 percent after 19 weeks.

Design, development and application of a hospital patient flow management support tool – Hospital Event Simulation Model: Arrivals to Discharge (HES-MAD) [Ben-Tovim et al. 2016]. HESMAD employs mathematical and statistical modelling techniques, as well as the concept of modular design, to construct functions and processes that are embedded in a discrete event simulation system.

Dynamic Decision Making in Biopharmaceutical Manufacturing [Sachidananda et al. 2016]. A simulation model for a subset of manufacturing activities was developed to improve current process in terms of throughout time reduction, better resource utilization, operating cost reduction, reduced bottlenecks etc.

Estimating the performance of emergency medical service (EMS) location models [Ünlüyurt & Tuncer 2016]. A simulation analysis was proposed to evaluate the performance of the location of EMS stations by estimating the “real” coverage of the population that takes into consideration the unavailability of the busy ambulances. The location of ambulances for each model can be found by using optimization tools and solving the mathematical models, and then simulating each setting for two different policies under the same parameters.

Other areas where DES has been applied are e. g.:

- Nuclear waste management [Garcia 2000]
- Software project management [Kouskouras & Georgiou 2007]
- Supply chain management [Liston et al. 2007]
- Assessing sign occlusion in buildings [Nassar & Al-Kaisy 2008]
- Power system networks [Tavakoli et al. 2008]
- Military deployment [Yıldırım et al. 2009]
- Bidder inquiry [East et al. 2009]
- Road construction operations [González & Echaveguren 2012]
- Harvest management [van’t Ooster et al. 2013]
• Cloud management [Amoretti et al. 2013; Bohez et al. 2015]
• Transportation and traffic management [Cha & Mun 2014; Gunavan 2014; Mihaita et al. 2014; Xu et al. 2016]
• Social service systems [Harpring et al. 2014]
• Virtual ergonomics [Perez et al. 2014]
• Logistics and maintenance scheduling [Cui et al. 2015]
• Buildings energy simulation [Frances et al. 2015]
• Drying models [Windisch et al. 2015]
• Maintenance systems [Alrabghi & Tiwari 2016]
• Operating Theater Layout Problem [Chraibi et al. 2016]
• Construction site management [ElNimr et al. 2016]
• Energy consumption modelling [Kouki et al. 2017]

No application example was found, where discrete event simulations would have been used for crisis management.

Shen [2000] proposes to facilitate discrete event simulation as a service on the Internet/Web. The proposed simulation facility enables portability and interoperability of simulation applications which may be developed in heterogeneous languages and environments and communicate over the Internet. Moreover, by integrating with Java, the approach also enables simulation applications to be developed in Java and executed on the Web.

Gyimesi [2008] investigates the usefulness of the eXtensible Markup Language (XML) in the simulation domain and presented a framework for discrete event simulation (DES) that uses Web Service technology. This work showed the use of XML can increase the benefit of DES.

Budgaga et al [2016] have developed an approach to enable real-time exploration of discrete event simulations with a framework that distributes the workloads across a range of commodity hardware, including public and private cloud resources. Once the models have been created, their performance is evaluated to improve the prediction accuracy by employing dimensionality reduction techniques and ensemble methods. To make these models highly accessible, a user-friendly interface is provided that allows modellers and epidemiologists to modify simulation parameters and see projected outcomes in real time. The corresponding outputs from this process are analyzed and used by a framework to produce models that accurately forecast simulation outcomes in real time, providing interactive feedback and facilitating exploratory research.

3.4 Agent modelling

In an agent model an “agent”, or a component, is a software module that performs a defined task. Components interact with each other through various communication mechanisms. The whole simulation system can be achieved by integrating these agent components into a Multi-Agent System (MAS). Yujun et al. [2012] proposed an agent-based simulation framework for designing and evaluating the performance of a safety critical system. In an application to a railway case the agent-based system shows very similar behavior and performance with the real system. The experiment
shows that agent-based simulation can express the behavior and performance of the system as a whole, especially, safety attributes can be displayed.

The organizational structure and the policies are important elements that have to be taken into account to simulate a real emergency activity. To facilitate the design of these simulations, an agent-based methodological framework for complex systems is proposed by Mustapha et al. [2013]. The main contribution of the framework is that it reflects the organizational structure and policies within the simulation, with a truly dynamic dimension.

Multi-agent systems are among the methods used for modelling and simulating natural disaster (ND) emergencies, because the agents have a degree of autonomy and they are able to interact with their environment. In this context there are several areas where MAS can be applied; they can act as a modelling paradigm or as a solution for software implementation. Modellers can also use MAS to create computer representations of dynamic events such as ND emergency. Therefore, the application of MAS in this area could help managers to experiment all possible scenarios of a disaster and assist them in making decisions [Mustapha et al. 2013].

Multi-agent systems include a group of robots which can collaborate together to accomplish complicated tasks that cannot be done by a single agent. The applications of multi-agent systems are increasingly wide in many aspects of civilian domains, from rescue, surveillance to discovery. In the paper of Rahimi et al. [2014] a time-varying formation of multi-agent systems is studied. For a special application of rescue and surveillance, a set of agents, consisting of unmanned aerial vehicles (UAVs) and unmanned ground vehicles (UGVs) are considered. By considering coupling between the agents in a neighborhood and a synchronization method, a proper controller is proposed for each agent who provides acceptable performance. This algorithm is evaluated by simulation and the results approve accepted performance of the proposed approach.

A multi-agent system is a powerful modelling technique for simulating individual interactions in a dynamic system and is distinctive in its ability to simulate situations with unpredictable behavior [Lampert 2002]. The current technological developments allow envision of systems approach that includes modelling of all aspects of the disaster event, its impact on the resources, population and the required response by the involved agencies. Agent-Based Simulation (ABS) allows a modeller to manipulate different levels of representations, such as individuals and groups of individuals. Agent-based modelling allows capturing the dynamic nature of the ND and facilitates the study of numerous resource coordination associated with the interaction of multiple teams [Monteiro et al. 2008]. Hence, multi-agent technology is widely used in various disaster management and community awareness issues.

The management of crisis situations has been a challenging problem from different points of views, such as communication efficiency and avoiding casualties. García-Magariño & Gutiérrez [2013] present an approach that includes an interaction organization pattern for Multi-agent Systems in crisis management, abstracted from several existing case studies in which the agents follow a sequence of interactions and the organization must optimize the use of human resources. The pattern considers an emergent organization of peers that adopt different roles according to the circumstances. The key features of the organization are its robustness, scalability (in terms of both agents and roles), flexibility to deal with a changing environment, and the efficient use of resources. Another key aspect is the
application of metrics for validating and improving the MAS in terms of response time. The MAS has been tested with 600 agents representing 200 citizens, showing its performance.

In the scenarios of crisis response, there is usually a catastrophe where several groups of agents with specific skills must be coordinated to assist the victims. These assistant agents collect information of the environment while working, and autonomously solve emerging issues. This kind of problem has several features that make MASs a proper tool for the analysis. It has a clear identification of agents, roles and groups and the agents have to interact with a customizable and changing environment. The solution of the problem requires collaboration between the involved agents. The scenarios offer a clear statement of the goals to satisfy both at the individual and group levels. This relevance has led to the development of common scenario implementations, like RoboCup Rescue [Kitano & Tadokoro 2001] and Drill-Sim [Balasubramanian et al. 2006].

General lessons can be extracted from the implementations made for the crisis response scenarios, providing useful insights for the design of the communication of systems with similar requirements and expected benefits. There are several groups of agents with different capabilities and goals, mainly citizens and specialists, like medical doctors and firefighters. Each of these groups shares some common goals, although individual persons have specific goals and behave autonomously in the field responding to their specific situations. The use of resources (like communications, movements, or medical supplies) must be planned and optimized as much as possible given their scarcity. The overall success of the system is regarded as the fast communication to reduce the casualties and, if possible, the damaged infrastructures and properties. In the paper of García-Magariño & Gutiérrez [2013] a MAS is presented to achieve the proper communication in the aforementioned scenario.

There are many studies and developments regarding the simulation of crisis scenarios, where MAS are used for different purposes e.g.

- use of a robot in the rescue task [Davids 2002]
- interactions among leaders and evacuees [Murakami et al. 2002 ; Nakajima et al. 2006],
- contingency management [Sheremetov et al. 2004],
- preventing crisis situations in city traffics and reacting against them [Kozlak et al. 2008],
- preventing firms from bankruptcy by means of contagious effects of other firms [Zhang et al. 2012]
- crisis response [Balasubramanian et al. 2006; García-Magariño & Gutiérrez 2013]
- mobile medical emergencies [Centeno et al. 2009 ; Cáceres et al. 2005 & 2006],
- emergency medical transportation [López et al. 2008]
- establishing a wireless sensor network in crisis situations [Sardouk et al. 2013]
- anticipating crisis situations in the context of military planning [Kaddoussi & al. 2011]
- emergency situations within a building, with a need for rapid evacuation [Dimakis et al. 2010]

On the whole, it can be concluded MASs are an appropriate technology to cope with crisis management situations, regardless their type [Garcia-Magariño & Gutiérrez 2013].

Agent-based simulation has spread out into many areas, including sociology, biology, economics, physics, chemistry, ecology, industrial applications and ND. For most of these areas, the ABS is
gradually replacing the micro-simulation techniques and object-oriented simulation. This is due to the ability of the ABS to capture different dynamic models [Russel & Norvig 2003].

3.5 Conclusions

Based on the applicability and availability of the modelling options in the PREDICT simulation tool development suite (Table 2), agent modelling was chosen for further investigation. A trial case of employing agent-based simulation to develop a stochastic operation time model of a realistic firefighting scenario has earlier been reported by Paajanen & Kling [2014]. The results were promising for a more generic agent-based SOTM simulator. In the next chapter an agent-based SOTM simulator for a Dynamic Process Integration Framework (DPIF) will be developed.

Table 2 – Implementation options.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Petri nets</th>
<th>Discrete event simulation</th>
<th>Agent modelling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommended by WP2</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>Covered by PREDICT consortium, in terms of expertise and relevant assets</td>
<td>NO</td>
<td>YES</td>
<td>YES</td>
</tr>
<tr>
<td>Applied to crisis management problems</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>Suitable for complex systems</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
</tr>
</tbody>
</table>

4 Agent modelling

In this chapter, the agent modelling option for the model implementation will be further investigated. “Applicability and availability of the modelling option in the simulation tool development suite” (PREDICT DoW - Workplan table - Page 15) will be discussed.

4.1 Dynamic Process Integration Framework (DPIF)

The Dynamic Process Integration Framework (DPIF) [Pavlin et al. 2010] is an information management system based on a service oriented approach which supports efficient creation of distributed systems for collaborative reasoning. In such environments, information management and information filtering are achieved by cascading interoperable services. The right sources are dynamically connected via service discovery and filtering/negotiation mechanisms. The cascaded services allow data-push/data-pull between compatible sources and consumers and querying on demand.
DPIF allows creation of platforms that support fast integration of services and information management between those services. Such a platform is a combination of DPIF components, arbitrary communication middleware and directory services (see Figure 11).

In DPIF, each service is represented by a proxy (also agent), a process that (i) translates between an arbitrarily complex service and the communication middleware, and (ii) allows the creation of information flows between compatible services.

DPIF proxies translate between the application layer, where information requests are based on the information type and the context, and the middleware layer, where the data is accessed by using the right API (application programming interface) calls and arguments; i.e. DPIF translates between the domain concepts used in the information management process and the middleware calls and parameters used for the physical connection between services. More details about DPIF can be found in [PREDICT D4.1 2015] and [Pavlin et al. 2010].

DPIF supports the implementation of distributed analysis processes by using the DPIF Software Development Kit (SDK) (see Figure 11). With the help of the development library, user specific logic can be implemented in each DPIF proxy (called agent hereafter). This functionality allows the implementation of distributed analysis processes over several agents through dedicated workflows.

The creation of workflows between several agents is driven by the agent’s service descriptions, i.e. each agent specifies which type of service it provides and needs. Through these service descriptions, different agents can establish a workflow between each other and share information, or, in other words, these agents can collaboratively perform certain analysis functions.
4.2 **Agent-based SOTM simulator in DPIF**

As it turns out, the DPIF framework seems well suited to implement an agent-based SOTM simulator to perform a stochastic analysis to obtain the total time delay of a response operation. In order to illustrate this, we use the Nuclear Power Plant (NPP) fire brigade response scenario example described in [Kling et al 2013] and [Paajanen & Kling 2014]. The timeline with the actors, connections and time delays of this example is shown in Figure 12. In this example, four agents are used, namely an agent for the control room, the control room employee, the guard center, and the fire brigade.

The time delay from the ignition to the suppression of a fire can be described by a mathematical formula with the following time delays: $t_{\text{OPER}}$ is the total operation time from the ignition to the suppression; $t_{\text{DET}}$ is the time delay from the ignition to the detection; $t_1$ is the time delay from the detection to the moment when the guard center makes the alarm; $t_{2,1}$ is the time delay from the detection to the moment when the control room starts co-operation with the fire brigade; $t_{2,2}$ is the time needed for cooperation between fire brigade and the control room personnel; $t_{FB,1}$ is the time for the fire brigade to get to the building entrance; $t_{FB,2}$ is the time delay of the co-operation with the control room personnel; and finally $t_{FB,3}$ is the time required for locating and extinguishing the fire.

\[ t_{\text{OPER}} = t_{\text{DET}} + \max \left[ t_1 + t_{FB,1}, t_{2,1} \right] + t_{2,2} + t_{FB,3} \]

*Figure 12 – An example of modelling the response in a fire scenario of a Nuclear Power Plant (NPP) [Paajanen & Kling 2014].*
Since DPIF is a Service Oriented Architecture (SOA), the information that agents can provide and need must be described in terms of services. This means that for each of the four agents the provided and required services need to be specified. In Table 3, a list of all the services of the four agents is given.

Table 3 – The provided and required services of the four agents from the NPP scenario.

<table>
<thead>
<tr>
<th>Agent</th>
<th>Provided Service</th>
<th>Required Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control Room</td>
<td>FireDetectionService</td>
<td>n/a</td>
</tr>
<tr>
<td>Guard Centre</td>
<td>FireAlarmService</td>
<td>FireDetectionService</td>
</tr>
<tr>
<td>Control Room Employee</td>
<td>CooperationInfoService</td>
<td>FireDetectionService</td>
</tr>
<tr>
<td></td>
<td>VoltageCutOffNotificationService (Dependency)</td>
<td>CooperationInfoService</td>
</tr>
<tr>
<td>Fire Brigade</td>
<td>CooperationInfoService</td>
<td>FireAlarmService</td>
</tr>
<tr>
<td></td>
<td>SuppressionNotificationService</td>
<td>VoltageCutOffNotificationService (Dependency in Figure 13)</td>
</tr>
</tbody>
</table>

In Figure 13, the different DPIF agents are shown together with their interactions based on the services they provide and need. The initial message that is produced comes from the control room agent about the fire detection. This detection signal is relevant for both the control room employee agent and the guard center agent, so these agents are subscribed to ‘FireDetectionService’ (required service). Despite the FireDetectionService message being a notification that a fire is detected, it should also contain the value for the detection time delay $\Delta t_{DET}$ that can later be used for the computation of the total time delay.

After the Guard Centre agent received the fire detection message, it triggers the fire alarm by sending a FireAlarmService message to the Fire Brigade agent that is subscribed to this service. The FireAlarmService message will contain the sum of $\Delta t_{DET} + \Delta t_1$. Subsequently, the Fire Brigade agent sends a CooperationInfoService message to the Control Room Employee agent that includes the sum $\Delta t_{DET} + \Delta t_1 + \Delta t_{FB,1}$.

The Control Room Employee agent, next to the Fire Brigade agent, also received the FireDetectionService message together with the value $\Delta t_{DET}$. Next, this agent sends a CooperationInfoService message to the Fire Brigade agent together with the sum $\Delta t_{DET} + \Delta t_{2,1}$. Next, the Control Room Employee agent sends a VoltageCutOffNotificationService to the Fire Brigade agent together with the value for $\Delta t_{DET} + \Delta t_{2,1} + \Delta t_{2,2}$. The Fire Brigade agent compares this value with $\Delta t_{DET} + \Delta t_1 + \Delta t_{FB,1} + \Delta t_{FB,2}$ and determines which value is the maximum and adds the value $\Delta t_{FB,3}$ to get $\Delta t_{OPER}$. 
Note that,

\[ \Delta t_{OPER} = \max(\Delta t_{DET} + \Delta t_{2,1} + \Delta t_{2,2}, \Delta t_{DET} + \Delta t_{1} + \Delta t_{FB,1} + \Delta t_{FB,2}) + \Delta t_{FB,3} \]

\[ = \Delta t_{DET} + \max(\Delta t_{1} + \Delta t_{FB,1}, \Delta t_{2,1}) + \Delta t_{2,2} + \Delta t_{FB,3} \]

Here we assume that the CooperationInfoService messages result in cooperation that ends for both parties at the same time, which means that \( \Delta_{FB,2} = \Delta t_{2,2} \). Note that the delivery time of the CooperationInfoService for the Control Room Employee agent and the Fire Brigade agent might in practice be slightly different and could jeopardize the equality of the above equation. In order to ensure \( \Delta_{FB,2} = \Delta t_{2,2} \), synchronization between these two agents is required to ensure that they both use the same end of collaboration time.

**Figure 13 – Interaction between the different agents showing the provided and required services.**

This example illustrates that through DPIF a stochastic total time delay simulation can be implemented using DPIF agents (DPIF proxies). This clearly provides several benefits in setting up these types of simulations. Since the simulation has a modular setup, it can easily be changed by adding or removing different actors. This facilitates the computation of the total time delays for different configurations of actors involved in the firefighting scenario. Additionally, different time delay agents can be reused in other types of simulations reducing the time to develop the stochastic simulation.
5 Human factors

In this chapter, “a methodology for the estimation of the influence of human factors on individual error probabilities, using the concept of performance-shaping factors” (PREDICT DoW - Workplan table - Page 15) will be developed.

5.1 Performance Shaping Factors (PSF)

This section describes a methodology for the estimation of the influence of human factors on individual error probabilities by using the concept of performance shaping factors (PSF). Human reliability analysis (HRA) estimates “the safety and risk significance” of human tasks [Blackman et al. 2008]. It provides methods and a technology to minimize and hopefully eliminate human error in production environments and in particular to help operators make the right decision at the right time to ensure safe operation. The “Human Element” is one of the most important elements in most production environments since 60-90% of the accidents in manufacturing operations can be referred to human errors [Benichou et al. 2002]. Whenever the “human element” fails, it leads to incidents and accidents or crisis situations. PSFs are capable of quantifying human reliability. PSFs identify a set of factors that are related to human performance based on questionnaire surveys and experiences, and can be categorized into nine sections (see Table 4).

Each HRA method can have a multitude amount of PSFs, which can be classified into the nine categories outlined in Table 4. Moreover, HRA methods can be separated into first and second generation HRA methods with the difference that the second-generation methods include the cognitive aspect in the PSFs [Lee et al. 2011].

A PSF contain a set of multipliers (M) of which one multiplier is chosen that best reflects the human’s performance. When the PSF represents a good result, then the multiplier corresponds to a level less than 1. However, when the operator’s performance was poor, then the multiplier correspond to a level higher than 1 [Boring 2010]. The poorer the human’s performance, the higher is the multiplier M. This constant is used to compute Human Error Probability (HEP).

Each type of task contains a nominal HEP that has to be determined beforehand. Multiplying the corresponding PSF multiplier by the nominal HEP yields the overall HEP [Boring 2010].

\[ HEP_{overall} = HEP_{nominal} \times M. \]
Table 4 – Performance Shaping Factor categories [Lee et al. 2011].

<table>
<thead>
<tr>
<th>Performance Shaping Factor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stress</td>
<td>Environmental factor that affects human performance.</td>
</tr>
<tr>
<td>Action type</td>
<td>Action type describes the type of task the operator is asked to execute. It includes also the complexity of the task.</td>
</tr>
<tr>
<td>Experience</td>
<td>Level of operator’s experience.</td>
</tr>
<tr>
<td>Time available</td>
<td>Available Time for fulfilling the task.</td>
</tr>
<tr>
<td>Places where actions are taken</td>
<td>(Self-explanatory)</td>
</tr>
<tr>
<td>Procedures</td>
<td>The procedures that are needed to complete a task.</td>
</tr>
<tr>
<td>Training</td>
<td>Include all activities that are related to training processes.</td>
</tr>
<tr>
<td>HMI</td>
<td>Human machine interaction.</td>
</tr>
<tr>
<td>Teamwork</td>
<td>Communication with others and inter-organizational work.</td>
</tr>
</tbody>
</table>

5.1.1 Example based on the HRA method SPAR-H

The standardized plant analysis risk-human reliability analysis method (SPAR-H) incorporates nine PSFs to identify and quantify human performance and was dedicated for plants [Blackman et al. 2008]. It distinguishes between two types of operator tasks:

- diagnosis-oriented tasks
- action-oriented tasks

Diagnosis-oriented tasks include “planning and prioritizing activities, determining appropriate courses of action and using knowledge and experience to understand conditions” [Blackman et al. 2008]. The nominal HEP for diagnosis-oriented tasks are equal to 0.01. Action-oriented tasks include all activities that require processes such as conducting training or taking decisions. The nominal HEP for action-oriented tasks are equal to 0.001 [Blackman et al. 2008]. This human error rate is used to quantify human performance and will be illustrated based on an example later. “A PSF is an aspect of the human’s individual characteristics, environment, organisation, or task that specifically decrements or improves human performance, thus respectively increasing or decreasing the likelihood of human error” [Blackman et al. 2008]. PSFs contain different levels that are rated with a multiplier (M). This M
is multiplied by the error probability to get the HEP for the specific factor. In the following, outlines of eight example PSFs that were developed for the SPAR-H are presented.

5.1.1.1 Available Time

This factor determines the time the operator has in order to fulfil the tasks or diagnoses. The amount of time can affect the human error probability. For instance, when an operator cannot complete a diagnosis task within the given amount of time, then the human error probability (HEP) is equal to 1. However, if the operator acquires, for instance, 2/3 of the time, then the factor M is equal to 10.

5.1.1.2 Stress and Stressors

SPAR-H considers stress as a disadvantageous condition that impairs the operator’s human performance. Stressors are referred to environmental factors that cause stress. The extent of stress and stressors can be categorized into three levels [Blackman et al. 2008]:

- Extreme: Potential threats or persistent stress for a long period of time that can lead to catastrophic consequences (M = 5).
- High: A high level of stress impairs the operator’s performance, for instance, due to environmental factors such as noise (M = 2).
- Normal: This level of stress is considered helpful in order to achieve a high performance (M = 1).

5.1.1.3 Complexity

This factor describes the difficulty of the operator’s task and includes also the environment in which the task is performed. The higher the complexity, the higher is the probability that the operator makes an error. For instance, the complexity of diagnosis-oriented tasks can be categorized into four levels:

- Highly complex: The task is very difficult and require high expertise of the operator (M = 5).
- Moderately complex: The task is less difficult but still requires a certain level of expertise in order to fulfil the task (M = 2).
- Normal: The task is not difficult (M = 1).
- Obvious diagnosis: The task is obvious (M = 0.1).

5.1.1.4 Experience and Training

This factor considers the experience and training of the operator. SPAR-H distinguishes between three levels:

- Low: An operator with low experience and training (< 6 month) has a high probability to make mistakes (M = 10).
- Normal: An operator with decent amount of knowledge and experience (> 6 month) is able to act adequately in “abnormal situations” (M = 1).
- High: An operator provides “extensive knowledge” and training and is prepared for high threatening scenarios (M = 0.1).
5.1.1.5 Procedures

This factor includes procedures (if they exist) for certain tasks and activities in order to enhance the human performance. According to SPAR-H, a possible categorization can be:

- Not available: The procedure to handle a certain task or activity does not exist (M = 50).
- Incomplete: The procedure is not complete. Important information is missing (M = 20).
- Available, but poor: The procedure is complex and not easy to use (M = 5).
- Nominal: Procedure exists and supports the operator to fulfill tasks (M = 1).

5.1.1.6 Ergonomics and Human Machine Interaction

This factor considers the ergonomics and the human-machine interaction and deals with the interaction of human with any kind of systems such as software or other information systems. Possible levels are: missing/misleading (M = 50), poor (M = 20), nominal (M = 1) and good (M = 0.5).

5.1.1.7 Fitness of Duty

This factor includes the physical and mental fitness of the operator. Influences like “fatigue, sickness, drug use” can affect the human performance significantly [Blackman et al. 2008]. Possible levels are: unfit (Probability of failure = 1), degraded fitness (M = 5), and nominal (M = 1).

5.1.1.8 Work Processes

This factor incorporates aspects of doing work such as “inter-organizational, safety culture, work planning, communication, and management support and policies” [Blackman et al. 2008]. Possible levels are: poor (M = 2), nominal (M = 1) and good (M = 0.8).

5.2 Example of HEP/HR calculation

For an action-oriented task, an operator has (EP=Error Probability):

- Nominal time ($EP = 0.001 \times 1$)
- Extreme stress ($EP = 0.001 \times 5$)
- Highly complex task ($EP = 0.001 \times 5$)
- High experience and sufficient training ($EP = 0.001 \times 0.1$)
- Other performance shaping factors as explained in the previous section The HEP (Human Error Probability) would be:

$$HEP = f(EP_i)$$

with HR (Human Reliability)

$$HR = 1 - HEP = 1 - f(EP_i).$$
5.3 **PSF Correlations**

The number of PSFs for a HRA method is not necessarily restricted. However, the number of PSFs does not correspond to the accuracy of the overall HEP. Methods that require a huge number of PSFs tend to have non orthogonal PSF, which may lead to a significant correlation between them [Boring 2010]. Spearman measured the correlations of Action and Diagnosis PSFs of the HRA method SPAR-H. Table 5 illustrated the results of diagnosis PSFs. For instance, the PSF Experience and Work Processes shows a significant correlation with the value 0.55 highlighted with green.

**Table 5 – Correlation of Diagnosis PSFs of SPAR-H [Boring 2010] where significant correlations are marked with orange, and the highest correlation with green.**

<table>
<thead>
<tr>
<th>Diagnosis</th>
<th>Available Time</th>
<th>Stress/Stressors</th>
<th>Complexity</th>
<th>Experience/Training</th>
<th>Procedures</th>
<th>Ergonomics/HMI</th>
<th>Fitness of Duty</th>
<th>Work Processes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Available Time</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stress/Stressors</td>
<td>0.67</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Complexity</td>
<td>-0.2</td>
<td>0.15</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Experience/Training</td>
<td>-0.3</td>
<td>0.06</td>
<td>0.21</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Procedures</td>
<td>-0.7</td>
<td>0.01</td>
<td>0.25</td>
<td>0.28</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ergonomics/HMI</td>
<td>0.01</td>
<td>0.06</td>
<td>-0.05</td>
<td>0.20</td>
<td>0.09</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fitness of Duty</td>
<td>-0.03</td>
<td>0.03</td>
<td>-0.03</td>
<td>0.18</td>
<td>0.09</td>
<td>0.44</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Work Processes</td>
<td>-0.06</td>
<td>0.00</td>
<td>0.24</td>
<td>0.55</td>
<td>0.36</td>
<td>0.15</td>
<td>0.10</td>
<td></td>
</tr>
</tbody>
</table>
6  **Event Trees (ET) & Sequences Assessment**

*In this chapter, other methodologies are presented that can be used to “support the model specification” (PREDICT DoW - Workplan table - Page 15).*

6.1  **Event Trees and Fault Trees**

Event Trees and Fault Trees (Figure 14) are the most common methods used in practical applications of quantitative risk assessment [Rausand 2011, Rausand & Høyland 2004, Signoret et al. 2013]. They have proven to be useful for modelling accident scenarios taking into account both causal and event sequence modelling. Both methods are usually used to model static systems or sequences. There is, however, also a less frequently used time-dependent event-tree method, which has been applied to the fire resistance of single-storey industrial buildings [Hietaniemi et al. 2002, Korhonen et al. 2003].

![Figure 14](image)

**Figure 14** - A typical event tree with four end states (ES1, ES2, ES3, ES4) and two pivotal events, which are modelled by fault trees (FT1, FT2) [Nývlt et al. 2015].

6.2  **Basic failure modes modelling**

The following hypotheses are considering in modelling such basic events (BE) and fault trees illustrated in Figure 14:

- Basic failure dependence: basic failure events are often independent and constant with time. In case of common mode failures, a beta-factor linear model will be used. Failure occurrence rates are generally considered time independent (Poisson Stochastic Process). The occurrence rate $\lambda_i$ of a basic failure mode ($i$) will be generally described as:

$$
\lambda_i = \lambda_i^0 + \sum_{j \neq i} \beta_j \lambda_j^0
$$
\( \lambda_i^0 \): the independent occurrence rate of the basic failure mode \( i \).

\( \beta_{ij} \): the beta-factor describes the dependence of failure mode \( i \) on failure mode \( j \).

- **Time-dependence:** If time-dependence should be considered, we will use a linear approximation:
  
  The occurrence rate \( \lambda_i(t) \) of a basic failure mode \( (i) \) will be generally described as:
  
  \[
  \lambda_i(t) = (\lambda_i^0 + \alpha_i t) + \sum_{j \neq i}^n \beta_{ij} (\lambda_j^0 + \alpha_j t)
  \]

- **General and tabulated time-dependence:** PREDICT predictive models would equally accept general time-dependent failure rates given in analytical form or in tabulated form.

### 6.3 Top event modelling

Top events are calculated given the full description of the basic failure modes (failure rates, initial conditions, common modes, etc.). For each identified top event (TE), PREDICT’s predictive models will determine:

- The occurrence probability function: TE occurrence probability function is a time dependent function.
- Mean time to occur of each TE, if required by the user.
- The equivalent occurrence rate (if required): the TE has generally a time dependent occurrence rate. It can optionally be determined if it is demanded by the user. However, the TE may have a constant equivalent failure rate if it is described by an OR logical gate including time-independent basic failure modes.

### 6.4 Event Tree and top events sequences

Figure 14 shows that an event tree with \( n \) independent pivotal events \( (E_i, i = 1, 2, ..., n) \) results in \( 2^n \) sequences (cascading of events). The PREDICT predictive tool would determine for any pre-defined set of sequences the following:

- The occurrence probability of the required sequence(s).
- The equivalent occurrence rate, if required by the user.
- Mean time to occur of each sequence, if required by the user.
6.5 **Methods and numerical techniques**

As mentioned above, PREDICT tool executes the MS&A (Modeling Simulation & Analysis) process at three levels:

- Basic failure modes occurrence level (Level 1),
- Top events level (Level 2),
- Event tree and sequences level (Level 3).

Methods and numerical techniques used will differ in nature and in complexity from one level to the other (Table 6).

**Table 6 – Methods and numerical techniques used will differ in nature and in complexity from one level to the other.**

<table>
<thead>
<tr>
<th>Methods &amp; Numerical techniques</th>
<th>Level 1</th>
<th>Level 2</th>
<th>Level 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fault trees</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Event trees</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Graph of states</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use of constant failure occurrence rates and linearization (with/without common modes)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Use of time dependent equivalent occurrence rate (analytical form/tabulated)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Analytical solution (if occurrence rates are constant at the corresponding level)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Numerical technique – Markov/semi-Markov processes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Numerical technique – Monte Carlo Simulation (biased/unbiased), in all configurations, especially with tabulated occurrence rates</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Uncertainty propagation using Monte Carlo Simulation</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
7 Conclusions

This document described the outcomes of the PREDICT Task 3.3 which focuses on the modelling of response actions and organizations. Stochastic Operation Time Modelling (SOTM) has been further developed as part of PREDICT Task 3.3 to be used later by PREDICT tool suite. The methodology consists of modelling steps for the stochastic estimation of operation times, and timeline charts to visualize the action and communication processes of the response procedures. A set of related methodologies for the specification of the SOTM layout and parameters are presented.

A methodology for the estimation of the influence of human factors on individual error probabilities, using the concept of performance-shaping factors (PSF) is suggested. PSFs are capable of quantifying human reliability by using set of factors that are related to human performance: stress, action type, experience, time available, places where actions are taken, procedures, training, human machine interface (HMI), and teamwork.

Other methodologies are presented that can be used to support the model specification. Event Trees and Fault Trees are the most common methods used in practical applications of quantitative risk assessment. PREDICT tool will execute the MS&A process at three levels: Basic failure modes occurrence level (Level 1), top events level (Level 2), event tree and sequences level (Level 3). Methods and numerical techniques used will differ in nature and in complexity from one level to the other.

Petri nets, discrete event simulation and agent modelling were investigated as possible implementation options for SOTM simulations. Based on the applicability and availability in the PREDICT simulation tool development suite, agent modelling was chosen for further investigation. The Dynamic process integration framework (DPIF) seems to be well suited to implement an agent-based SOTM simulator to perform a stochastic analysis to obtain the total time delay of a response operation. Analysis clearly showed several benefits in setting up these types of simulations. Since the simulation has a modular setup, it can easily be changed by adding or removing different actors. Additionally, different time delay agents can be reused in other types of simulations, which will reduce the time to develop the stochastic simulation.
REFERENCES


PREDICT D3.2 Methods of threat quantification (2015).

PREDICT D4.1 System design document (2015).

PREDICT D7.1 End-users requirements (2015).


APPENDIX A: Integration of DPIF into the integrated PREDICT decision support tool (iPDT)

DPIF is an interoperability and information management platform that allows easy integration between producers and consumers of data/information. For the 1st integration cluster, i.e. the integration of the tools PROCeed, MYRIAD and SBR it was decided to use RESTful web services directly between the different tools without using information management intermediary that determines between who and under what conditions certain data must be shared. For this integration cluster that makes a lot of sense since the producers and consumers of information are always configured in the same way, i.e. the producer-consumer pairs remain the same throughout the lifetime of the system.

For the full iPDT integration (3rd integration cluster) the 1st and 2nd (DEIN/DPIF) are integrated. DEIN is a communication system that is based on DPIF. It allows easy sharing of information between human experts and is in that sense a sort of communication tool. Contrary to other communication systems DEIN is service-based, that means that communication channels between producers and consumers of information are established in an ad-hoc manner based on the information needs and the availability of certain information. Different from a chat communication system where you contact a certain person, in DEIN you provide data/information for a certain service to which consumers are subscribed.

DPIF can be used to make the services that the tools PROCeed, MYRIAD and SBR provide interoperable with the services that the DEIN experts require creating a sociotechnical system. For example, SBR could provide the service “LikelihoodFailureCIIService” that provides a likelihood computation of the failure of a certain CI at a given time. This information could be very helpful to operators of other dependent CIs. In such cases the CI operators can easily get access to these services as long as he/she has an Internet connection (such as service could be compared to a cloud service, i.e. Software as a Service (SaaS) cloud service model). Note that the information needs of the CI operator are probably very different in case of a different crisis. This means that it is likely that the CI operator uses different services in different crises situations. Also the conditions under which the operator will receive this information can be set (n case of alerting for example). He/she might only want to get this information if the likelihood is above a certain threshold. DPIF provides the means to configure such a logistic information management layer to guarantee that information is shared efficiently and effectively.
PREDICT PROJECT PARTNERS

- CEA
- SYKE
- Veiligheidsregio ZHZ
- TNO innovation for life
- CEIS
- ITT e-technologies & business
- Fraunhofer IAIS
- UIC
- VTT