

# Designing Complex Technical Rescues with a Proprietary Application (Computer Program)

Csaba HAJDU,<sup>1</sup> Rajmund KUTI<sup>2</sup>

*Handling emergencies caused by different environmental and civilization catastrophes, [1] (e.g. extreme weather, flood, severe industrial accidents involving dangerous substances, possible terror activities) is an ever-increasing challenge for damage control rescue entities. Efficient executing of these tasks requires ever increasing preparedness of intervention units, constant technical equipment development, introduction of new technical rescue tactics and complex intervention plans.*

*Executing such a complex rescue plan is an extremely complicated task. At a given situation more units are to be coordinated in several disaster areas (sites). Several special technical rescue tasks have to be pre-planned to increase the efficiency of the interventions. The authors would like to help these planning efforts by an originally developed proprietary force-unit calculation application (software) program, described in this article. The possible applications, technical requirements, detailed usage steps are also described.*

**Keywords:** *complex technical rescue, disaster relief planning, force-unit calculation, originally developed proprietary application (software) program*

## Introduction

In case of simple accidents with a high number of occurrence technical relief operations usually are not problematic. It is easy to identify force-equipment to be used in such relief efforts based on similar past events. In more severe cases after evaluation, the Operations Control decides the forces and equipment to be alarmed. Their work is supported by different expert programs and databases. [2] If the alarm call is not received by Operations Control, but by the given facility or voluntary fire rescue then the Commanding Officer (CO) decides about the units and technical equipment to be mobilized. In this situation based on past experience the CO quickly evaluates whether the mobilized units and equipment is sufficient for the rescue effort. The decision is only verified on the actual disaster site, where the alarm level can be modified by the fire, rescue operation site commander. These decisions can be highly supported by an application which helps in allocation of the force-equipment. [3]

Such a decision supporting application is developed which helps the users especially in force-equipment allocation and alarm level determination.

---

<sup>1</sup> MSc student, Budapest University of Technology and Economics; e-mail: [h12csaba@t-online.hu](mailto:h12csaba@t-online.hu)

<sup>2</sup> Ph.D. (Military Technology), Associate Professor, Széchenyi István University, Department of Mechatronics and Machine Design; e-mail: [kuti.rajmund@sze.hu](mailto:kuti.rajmund@sze.hu)

## ***Planning Basics***

Planning technical rescue operations is not an easy task. It is expedient to plan operation steps in advance for better coordination, accuracy and saving time and life. In rescue operations units use special gear and technical equipment.

First, the list of available systematized equipment (gear) in Hungary has to be created in an equipment library. Special vehicles which carry rescue equipment, different special substances used in fire rescue operations are included in this library.

Manpower necessary for operation of the equipment which has to be taken into account during planning is defined in the professional guidelines for the equipment.

A User-Friendly User Interface is developed. During rescue operation time component is crucial, so simple handling, easy overview, speed, reliability are primary requirements. Experiences gained during past rescue operations are also taken into account.

During application planning tasks to be executed by fellow rescue organizations had to be also overviewed. It is known that gas body suits are very heavy to use as it is very hard to move in them and the respiratory equipment exhausts the rescue officer, so they have a very limited energy and time for actual work. Consequently, backup manpower has to be planned and provided by the application.

## **Overview of Computer Force-Equipment Calculation**

The hardware and configurational requirements of the application program are average. It can be run on every PC where the NET framework is installed (deployed).

The calculation is based on the available technical equipment and on required manpower defined in the operation's security manuals (directives). The application user has to enter only the rescue site and the required technical equipment quantity in input fields (left white), and the special substances if necessary. Based on the planned number of squads the application offers the alarm level qualified with BM regulation 39/2011 (XI. 5.) about "Common regulations about fire and rescue operations." If special substances are to be carried, the application offers special (emphasized) alarm execution (which is also displayed).

During rescue operations there can be additional tasks that cannot be foreseen. To address these situations additional preplanning is required, to avoid reclassifying and re-grouping of tasks. The application gives possibilities for these.

The layout of User Interface of Force-equipment calculation application program with extensions is given on the next diagram.

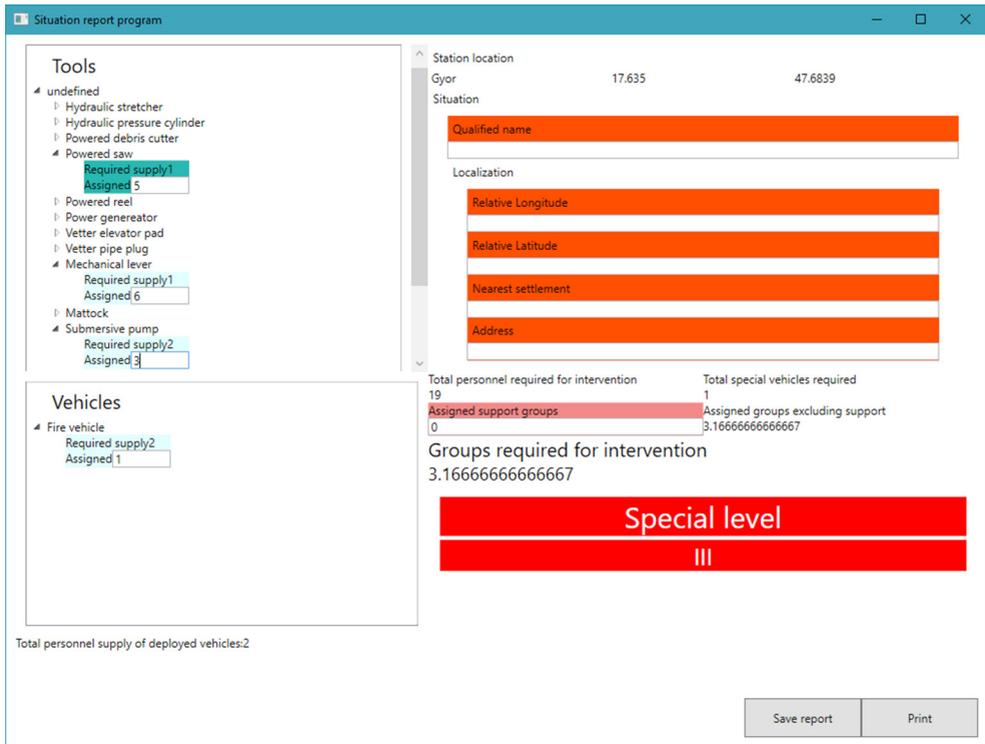


Figure 1. *Target program operation example.* [Edited by the authors.]

Filing out the input fields is easy as it is only possible to input data in the white fields. Other fields are write protected, to avoid overwriting of mathematical formulae.

Where numbers are displayed in red, the computer fills them automatically.

The lowest red bar is the calculated alarm level by the application program based on input data.

### ***Structure of the Recommender System***

The Recommender System process has two components. The process and the components are identified on Figure 2. With the target program an alarm level can be set for an incoming rescue operation. The second component (*Inventory description editor*) generates the list of equipment according to regulations with the help of an internal *generator component* interpreting the abstract description. This list is used by the *target user program*. During the execution of the program the list of the available force-equipment must not be changed.

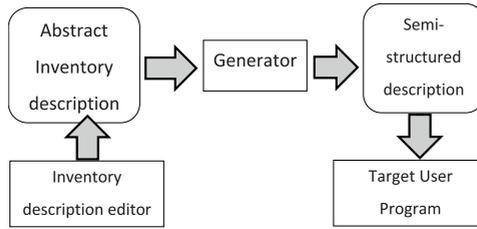


Figure 2. Description of the configuration generation process. [Edited by the authors.]

The list is produced by an Eclipse-based editor. A force-equipment can be modelled as a labelled weighted element, where the weight corresponds with the personnel needed to operate the equipment. Breaking down on the context of the problem vehicles are differentiated from other equipment. The model is based on Eclipse-Ecore (Eclipse Modelling Framework – EMF); [5] there is a possibility of creating a textual description language. The Xtext [6] [7] framework provides an efficient way to produce a text editor fitting the problem-specific language that is derived from the meta-model of the inventory description. The text language should be unambiguous and transparent like natural languages in order to be easy to learn even without programming skills and experience. This editor generates a semi structural file (in XML format) after every change.

The target program reads equipment (device) inventory generated description. The user has limited possibilities to change certain fields. The number of deployable force-equipment and the description (unique name, alarm site, etc.) of the alarm can be changed only. The input data should be strictly controlled by the program. Exemptions and bad data should be treated also (e.g. negative or fractional number of deployable units etc.). After data input the report can be saved in a text format or printed out on the default printer.

The usage can be demonstrated through a simple example; an alarm about a vehicle accident carrying gaseous explosive material comes in. The necessary assets are detached from the equipment library for emergency handling. This is increased by the weighted manpower necessary for the unit. If a vehicle (e.g. technical rescue, special substance) is necessary, the alarm level automatically escalates to extensive (critical) and the number of manpower needed to operate the vehicle is added to the result. According to regulations, the number of squads (per 6 person) can be determined. The program recognizes complete and half squads. Further squads can be assigned for tasks not foreseen.

In summary the complete deployable weighted manpower for operation of vehicles and equipment can be produced. (Figure 3)

$$\begin{aligned}
 |personnel| = & \sum_{i=0}^{|vehicles|} capacity(vehicle[i]) \times count(vehicle[i]) \\
 & + \sum_i^{|tools|} capacity(tool[i]) \times count(tool[i])
 \end{aligned}$$

Figure 3. Final manpower equation. [Edited by the authors.]

The structure of the developed meta-model is depicted on the following UML Class diagram (Figure 4), the basis of the textual tool describing an inventory set. Notice that the inventory contains two distinctive categories of inventories: vehicle and tool inventories. Both inventories are filled with corresponding tools and vehicles, both elements derived from an abstract inventory element type. Also, the inventory contains information about the location of the emergency station (name, latitude and longitude).

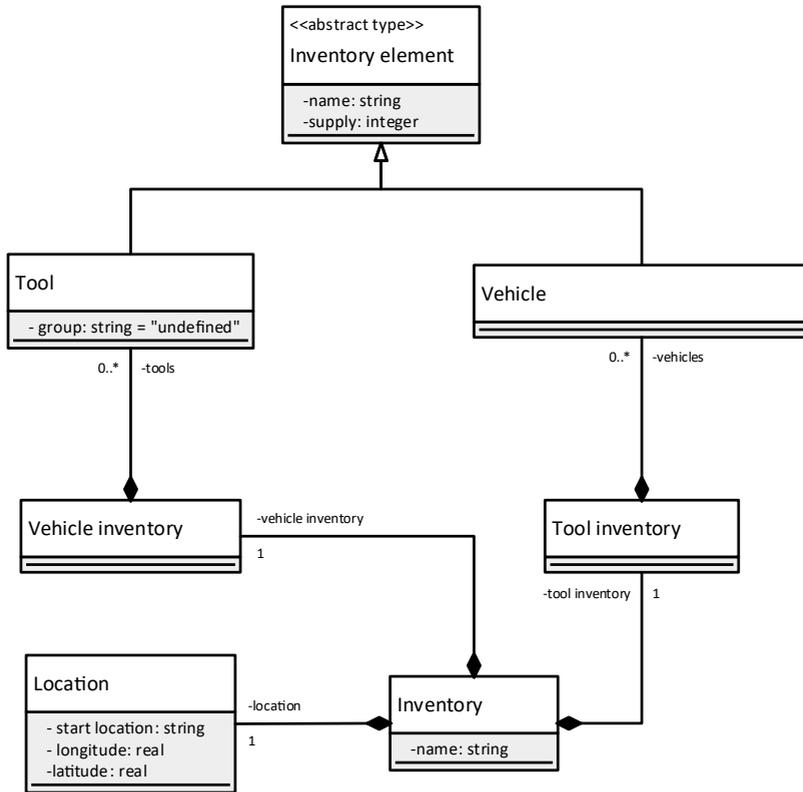


Figure 4. Structure of inventory meta-model. [Edited by the authors.]

The equipment library can be abstractly modelled in context of the problem. In the library vehicles are differentiated from other equipment because deploying even one vehicle leads to another level of alarm. In the problem context vehicles and other equipment can be characterized with common attributes (name, manpower required for the operation etc.). Common devices can be rated in groups for an easier overview. For program operation and ordering of an alarm it is necessary to note from which station the alarm was originated. It is feasible to store these data in the configuration of the Target Application (e.g. station name, city, geographical coordinates etc.).

## ***Description Language***

The description language follows the syntax of ALGOL/Pascal that fits into the meta-model of the equipment library. Equipment library is described in a structural form, so first the universal description of the equipment library should be given (unique name for the equipment library). After that the vehicles and further equipment can be characterized. The attributes of certain equipment are given divided by line space. All characteristics should be given—excluding groups of equipment. (Figure 5)

```

inventory DefaultInventory
  toolinventory
    tools
      tool Squirt
        supply: 2
        group: Liquid actuator
      end tool
      tool Precision drill
        supply: 1
        group: General tool
      end tool
      tool Powered hammer
        supply: 2
      end tool
      tool Powered debris cutter
        supply: 3
        group: Powered tool
      end tool
    ...
      tool Powered mattock
        supply: 1
        group: Technical rescue tool
      end tool
      tool Powered reel
        supply: 2
        group: Technical rescue tool
      end tool
    end tools
  end toolinventory
  vehicleinventory
    vehicles
      vehicle Fire vehicle
        supply: 2
      end vehicle
      vehicle Daru
        supply: 2
      end vehicle
      vehicle Chemical container
        supply: 2
      end vehicle
      vehicle Technical Rescue Vehicle
        supply: 2
      end vehicle
    end vehicles
  end vehicleinventory
  station
    location Gyor
      latitude 47.6839
      longitude 17.635
    end location
  end station
end inventory

```

Figure 5. *Equipment library description example.* [Edited by the authors.]

The equipment library was uploaded with data concerning systematized equipment associated with the number of required operating personnel. The equipment library can be extended after procurement and systematization of a new equipment.

The handling of the Application is easy, UI is user friendly, different deployment scenarios can be evaluated, tested in short time. The result of the tests can be printed out. Further steps can be decided after result comparison. The data is stored in a structural form after the program is run, allowing further data analysis and evaluation.

## Conclusion

After testing the application, the conclusion is reached that it can be of valuable help in supporting decisions about alarm level determination.

Handling is easy, results are quickly displayed, results can be printed out if required. Databases (libraries) are extendable, so newly developed equipment can be included. This is crucial for the up to date operation.

It is also proved that the application is capable of analysis of past technical rescue operations. The result of this analysis can be used for case studies.

## Summary

The developed application program supports decision making of pre-planned tasks and personnel assignment for rescue operations therefore these operations are quicker and easier executed.

The application program can be used for situation training planning, stored data is saved, can be printed out data and the necessary force-equipment is perspicuously laid out. The validity of the proposed alarm level, the availability of necessary equipment can be evaluated during drills, so gained experience can be used during live rescue operations.

Complex, combined technical rescue operations are becoming efficient if the parts of the system are chosen according to the task goal. To achieve that a complex technical rescue operation reaches suitable results, all the following factors are necessary:

- suitable equipment,
- appropriate staff with protecting gear,
- trained and prepared personnel,
- appropriate tactics.

Decision supporting computer applications and experience gained on pre-planned technical rescue drills (exercises) support the elaboration of appropriate rescue tactics.

Increased attention should be focused on possibilities offered by Information Technology (IT) also. The main goal is however that the rescue operation should be started in the shortest time frame from the event with the greatest efficiency and optimal usage of available resources (equipment), so the damage handling (liquidation) can be done more quickly.

## References

- [1] HALÁSZ L., FÖLDI L., PADÁNYI J.: Climate Change and CBRN Defense. *Hadmérnök*, 7 3 (2012), 42–49. [http://hadmernok.hu/2012\\_3\\_halasz\\_padanyi\\_foldi.pdf](http://hadmernok.hu/2012_3_halasz_padanyi_foldi.pdf) (Downloaded: 15.08.2017)
- [2] MOLNÁR R.: A tűzoltói beavatkozások hatékonyságát növelő fejlesztések az egységes katasztrófavédelmi rendszerben. *Műszaki Katonai Közlöny*, XXVII 3 (2017), 131–145. [http://hhk.uni-nke.hu/downloads/kiadvanyok/mkk.uni-nke.hu/PDF\\_2017\\_3sz/2017\\_3sz.pdf](http://hhk.uni-nke.hu/downloads/kiadvanyok/mkk.uni-nke.hu/PDF_2017_3sz/2017_3sz.pdf) (Downloaded: 20.09.2017)
- [3] KUTI R.: Komplex műszaki mentések tervezésének lehetőségei. *Védelem Online. Tűz-és Katasztrófavédelmi Szakkönyvtár*, 233 (2010), 1–7. [www.vedelem.hu/letoltes/anyagok/233-komplex-muszaki-mentesek-tervezesenek-lehetosegei.pdf](http://www.vedelem.hu/letoltes/anyagok/233-komplex-muszaki-mentesek-tervezesenek-lehetosegei.pdf) (Downloaded: 20.09.2017)
- [4] PADÁNYI J., FÖLDI L.: Tasks and Experiences of the Hungarian Defence Forces in Crisis Management. *Bilten Slovenske Vojske*, 17 1 (2017), 29–46.
- [5] *Eclipse Modeling Framework (EMF)*. [www.eclipse.org/modeling/emf/](http://www.eclipse.org/modeling/emf/) (Downloaded: 20.09.2017)
- [6] BETTINI, L.: *Implementing Domain-Specific Languages with Xtext and Xtend*. Birmingham: Packt Publishing, 2013.
- [7] *XTEXT homepage*. [www.eclipse.org/Xtext/](http://www.eclipse.org/Xtext/) (Downloaded: 20.09.2017)