Fort Lauderdale, Florida, USA
inter•noise 99
1999 December 06-08

A WEAPON DATABASE FOR THE PREDICTION OF SHOOTING NOISE

Karl-Wilhelm Hirsch, Mattias Trimpop

Institut fuer Laermschutz, IfL, Arnheimer Strasse 107, D-40489 Duesseldorf, Germany

INTRODUCTION

Since the early ’80s, the Institut fuer Laermschutz (IfL) has been involved in predicting shooting noise in the vicinity of civil and military training areas. In the beginning, much work was done to describe sound propagation of high impulsive low frequency blast noise over large distances because it was assumed that this topic was the most important. Indeed, it is one of the three most important problems with the prediction of shooting noise. However, there are two more fields of similar importance i.e. the recording of activities on, in particular, military training installations and, primarily, the acoustical emission data for the numerous weapons and ammunition used at these installations. This paper deals with the indication of the blasts sources, the storage of their acoustical properties and the estimation of these data should there be missing values for muzzle blast, explosion of charges and the projectile sonic boom, where relevant.

DATA EXCHANGE FORMAT

It is a challenge to acquire source data of weapon blasts with respect to both measuring technique and costs. In 1996, the ‘Ad hoc Working Group On Low Frequency Impulse Noise’ published a test plan for measuring blast data from large weapons, /1/. In order to facilitate the sharing of data amongst its members, the ‘Ad hoc Working Group’ then decided to set a clearly defined exchange data format.

The format is based on a clear and unique identification of the blast source. Military codes, national or NATO codes, do not cover all blast source fired on a military installation. Therefore, a new source code had to be developed. Based on this unique source code identification, the exchange format defines a list of mandatory and an expandable list of optional attributes (properties) of blast sources. In addition, the format prescribes two mandatory SQL queries that a database or program that is part of the data pool must be able to answer. The first SQL retrieves the available fields including their declaration and explanation available in the queried database. The second query gives the source data with respect to the unique source code.

THE DATABASE

The IfL has been tasked to propose and maintain this unique source code and to develop a database to demonstrate the benefits of this new exchange format. The result is a relational database structure called WAF2000 and the program BLASTER, which is now a constituent application of IfL’s WinLarm suite, a software used in Germany to predict shooting noise in the vicinity of military installations.

With respect to the design rules of relational database, the data analysis yields a solution with two collections of tables to hold the essential information. The tables ‘WeaponSystem’, ‘Weapon‘, ‘Propellant‘, ‘Projectile‘ and ‘IgnitionDevice‘, the so-called data collection, holds the information corresponding to their names. The table collection ‘WeaponRelWeaponSystem‘, ‘AmmunitionRelWeapon’ and ‘Ammunition’, the so called relation collection, record the information about the relations between tables of the data collection. This structure takes care of the requirements that a weapon system can have more than one weapon, that the same type of weapon can be mounted to more than one system, that there can be a lot of ammunition for one weapon, that the same ammunition can be used for more than one weapon and that ammunition is compiled by combinations of propellant and ignition device and the projectile and its ignition device.

Each entity of each table of the data collection is identified by a 4 byte primary key field called the table’s code, referred to as ‘weapon code’ for the table Weapon and so on. The so-called source code is then compiled as a 24 bytes long string sequentially containing the codes of the data collection tables. By definition this code is unique for any combination of the entities of these tables. (The IgnitionDevice code occurs twice in the source code to indicate the ignition firstly of the propellant and secondly for the projectile.) In this view, the relation collection only determines whether the code exists or not.

There is a third collection of tables holding names, descriptions and remarks and national codes for each entity of each table if necessary. This structure strongly supports the multi-lingual usage of the database, because in Germany, for instance, military from nearly all NATO-members practice. This forces the database to take care of national naming convention and codes in order to provide clear access for everyone. However, it is important to stress that this collection is an user-friendly add-on never meant to identify an entity in any table. Programs are included to link user-friendly names and national codes, or what so ever, to the unique codes though the database stores all the text relative to the codes.

To fulfill the German military requirements, their version of this database currently provides more than 1,000 source codes not just for noise prediction purposes. Military management software, providing tools to plan safe practicing on training areas, uses WAF2000 and an ActiveX-version of BLASTER to access the database to uniquely identify the planned missions. WinLarm and its application COUNTER also use the database to collect and record the activities on 4 large training areas in Germany. In the near future, the database will also hold small arms data including German and English names and descriptions.

ACOUSTICAL DATA

The program BLASTER not only manages this database. It also manages the acoustical data for the blast on the basis of the source code. It appends four more bytes to the code to generate a so-called blast code unique for an acoustical event. The bytes identify the kind of blast (muzzle blast, explosion, sonic boom etc.) the acoustical measure (CSEL, AFmax etc.) the propagation model (i.e. German TA Schießlärm) and the data origin. This blast code links to the WinLarm specific database structure to retrieve, to present and to edit the coefficients of the propagation model and the directivity pattern for the specified blast and acoustical weighting with respect to the data origin. The blueprint of this database structure is discussed in /2/.

Acoustical data in this database are often linked with a full 24 byte source code because the weapon system and its construction will influence the acoustical behavior of a the weapon and can change the directivity pattern for instance. Alternatively, the same ammunition fired with different weapons – with or without a muzzle brake – has different directivity and/or acoustical source strength. However, shells fired by howitzers will yield the same acoustical output irrespective of the weapon system or weapon or propellant used. In this case, only the projectile code and the ignition device code are relevant to determine the acoustics.

Currently, real measured data are only available for some important source codes and these are stored in the database, but there will never be a complete set of acoustical data assigned to each source code. Therefore, there is need for a strategy on how to retrieve acoustical data even if only poor information is stored in the database.

THE 'FRIENDLY SYSTEM'

This strategy and the corresponding rules of replacement to find the best estimation for acoustical data is called the friendly system. To support such a friendly system the database holds special codes in each table that can be used as references if no specific data are available. For example, the code '~ ~ ~ ~' means in all tables that this part is not specified or 'unknown', '~NON' means that this part does not exist. If the projectile ignition device code in the source code reads '~NON' there will be no explosion at the target from this source. If the propellant is '~NON' it may be a hand grenade without any muzzle blast. Using these special entry for the codes yields various source codes helpful to define a friendly system within the accessing program.

The program BLASTER provides such a friendly system controlled by flags. The strategy in BLASTER depends on the type of blast, of course. For instance, for an anti-tank projectile it is not important which weapon mounted on which system using which propellant brought the projectile to the target. The acoustical data depend on the projectile and ignition code only. Therefore, if data are not available for the full code all information about the system, the weapon and the propellant is ignored. If there are still no data the ignition code is set to '~ ~ ~ ~'.

If there are still no data for the projectile a second strategy is applied. The propellant and projectile tables provide attributes called 'acoustical class'. This class correspond to demolitions of equivalent TNT and therefore correspond to a certain level estimation. The database knows 26 predefined classes logarithmically scaled from micrograms (class A) to Megatons (class Z). If this attribute is correctly set the estimation will be not far away from reality. However, this entry is constraint to be set at least to 'Z' as default. Therefore, even if this attribute is not set, there will be an answer, but it will yield a very high level. Normally this is an data input error because acoustical classes should be set to reasonable letters. But this answer makes clear that if the friendly systems can rely on less and less information the levels should increase step by step finally ending with nonsense. In BLASTER however, the flags can stop the friendly system from using class 'Z' and send an warning message instead.

The friendly system is much more complex for muzzle blast because unknown full code data can be searched by replacing either propellant or weapon or weapon system code or combinations of two of them. However there will always be a reasonable answer if the friendly system is switched on; if there is no answer, then the database is not correctly filled with data for the reference codes.

Missing acoustical data is one reason for the use of such a sophisticated friendly system, incomplete or not fully specified activity data from the installation is the next reason. For planning purposes for example, such data may read like ‘2,000 rounds with Leopard 2 tank firing on range R12’ or ‘50,000 rounds with 20 mm gun firing anti-aircraft missions at range R23’. The first line denotes the weapon system only, the second line determines the weapon. In both cases the friendly system must yield reliable data. The system will indeed find reference data for the 20 mm gun and the 'Leopard' in the German version of the database. If not, in case of the 20 mm gun it will search for the acoustical class data and be successful. In case of the 'Leopard 2' there is no acoustical class assign to the weapon system. However, a weapon system offers a further attribute: the weapon system group which reads like ‘battle tank’, ‘howitzer’, ‘bomb’ and so on. This group again has an acoustical class which should point to the loudest member of this group.

SUMMARY

The relational database structure WAF2000 provides an unique key to clearly define sources of blasts from military and civil weapons. The structure offers a multi-lingual management of names and descriptions for each entity of each table. This structure is public and available from the Institut fuer Laermschutz.

The German military instance of the weapon database using the blueprint of WAF2000 currently provides more than 1,000 source codes for weapon blasts. For scientific purposes in the field of shooting noise prediction, the German MOD co-operates with scientists from NATO members within the Ad hoc Working Group.

The program BLASTER retrieves a best estimation of acoustical data for every blast. A so-called friendly system supports the estimation of acoustical data with respect to source strength and directivity pattern. The program also allows to input, edit and present acoustical data for sound propagation and directivity of the blast.

ACKNOWLEDGEMENTS

The German Ministry of Defense supports this work.

REFERENCES

/1/ „Standard Method to Measure the Sound Exposure Emissions and Immissions from Large Weapons“, Fortschr. Ber. VDI Reihe 8 Nr. 559, E. Buchta (Editor), in English and in German; Duesseldorf: VDI Verlag 1996

/2/ M. Trimpop, K.-W. Hirsch, „On The Advantage Of Relational Data Structures And Client/Server Applications For Shooting Noise Data Using ODBC, SQL and OLE Standards, Proceedings Internoise’96, Liverpool, UK, 1996, 545-548.