Miller, Diane M. (CDC/NIOSH/EID)

From: Fingerhut, Marilyn (CDC/NIOSH/OD) (CTR)
Sent: Thursday, June 12, 2008 1:42 PM
To: NIOSH Docket Office (CDC)
Cc: Fingerhut, Marilyn (CDC/NIOSH/OD) (CTR)
Subject: # 138 Comments for CB toolkit for nanoparticulate control
Attachments: toolkit_software_proposal.pdf

Dear Diane,
Please add these comments to the NIOSH Control Banding document.
Thanks
Marilyn

From: Andras Szucs [mailto:szucs@ilo.org]
Sent: Thu 6/12/2008 9:26 AM
To: henri.heussen; Rolf Packroff; Fingerhut, Marilyn (CDC/NIOSH/OD) (CTR); Niemeier, Richard W. (CDC/NIOSH/EID); Lentz, Thomas J. (CDC/NIOSH/EID); j.zalk@comcast.net; dnelson; alberto.camacho-henriquez; andrew.garrod; paul.evans; Pavan Baichoo; David Zalk; tomas.marques; jonathan.krueger; tom.grumbles; kortume; tempowskij; vickersc; WilburnS
Cc: Sameera Al-Tuwaijri; Gabor Sandi; Niall Mc Phillips - S+A; Emmert
Subject: Re: Nanotool: CB toolkit for nanoparticulate control

Dear All,

I was very pleased to read David's mail on the Nanoparticle Exposure Toolkit. It inspired me to make a project proposal for the development of a general Internet- and database-based Toolkit software (see attd).

Niall McPhillips, an Oracle database administrator and application developer, with whom I have worked together on several similar projects (currently on the new ICSC application package), liked the idea as well. He created a small Internet-based Oracle package for the Nanoparticle Exposure Toolkit (http://www.s-and-a.ch/dyn/toolkits/nanoparticles.showWorksheet). It is completely functional as a toolkit, but is also a simple, reduced, draft model developed according to the concepts of the proposal.

I would like to receive your views on these two contributions to our common plans to standardize the presentation and usage of toolkits. I'm pleased to inform you that I'll participate at the upcoming Congress in Seoul from 29 June to 2 July. I hope to meet there many of you and discuss this subject at the related Symposium and/or at an informal lunch meeting before that.

I look forward to hearing from you.
Best regards,

András Szücs
ILO/Safework
szucs@ilo.org

>>> David Zalk <zalk1@llnl.gov> 19.05.08 22:30 >>>
Dear all,

Enclosed please find a short synopsis of our Nanotool, a Control Banding toolkit to afford a qualitative risk assessment of work with nanoparticulate. It is designed to take into account the greatest limitation for traditional occupational hygiene applications with nanomaterials, the lack of toxicological data. In fact, it is designed so you can revisit the Nanotool outcomes with each and every new piece of toxicological and epidemiological information.

The Nanotool process is pre-loaded in an Excel spreadsheet that, with user input, can quickly and easily formulate the Risk Level outcome and control parameters per task. We are working to get the Nanotool released for continued application as well as our working draft of the manuscript that serves as a technical basis. We are hoping to make these available soon with limited distribution.

To date we are very pleased with its outcomes in testing and even more so with the uniformity of acceptance from multiple experts internationally. We are very happy to accept questions and comments as this process moves forward.

Please let us know if you or your colleagues would be interested in a copy when available.

All the best,
Dave

Software for toolkits - A project proposal

by

András Szücs, systems analyst, ILO/Safework

Niall McPhillips, Oracle DBA and programmer, Stephenson and Associates

Geneva, May 2008

Introduction

This proposal was influenced by the excellent structure of the Nanoparticle Exposure Toolkit, as presented in the article by Samuel. A. Paik, David Zalk and Paul Swuste. It also draws upon the accumulated experience of the ICCT, as well as on the authors' own ideas on how to develop a powerful software that would facilitate the creation, management and use of toolkits.

In the following we have drawn up a short, informal and non-comprehensive description of a proposed computer application package as if it already existed. We have made sure that what we describe can be programmed with current techniques, already used in existing applications, such as the new, sentence-based package developed for the International Chemical Safety Cards (ICSC) project.

Scenario 1 – Using a toolkit

You, as the end-user, click in your browser on the URL of the Toolkits Homepage in English. Following a short introduction, a list comes up, looking like this:

Toolkit 1	English, French, Spanish
Toolkit 2	English, Arabic, German, Hungarian, Russian
Nanoparticle	English, Spanish
Toolkit 4	English, Chinese, Japanese, Spanish
...	...

Since all toolkits in the collection have an English version by default, this is a complete list of toolkits. In other language home pages only those toolkits are listed that have a working version in the language concerned.

Once you have clicked on a toolkit's name, the introductory screen for it is displayed. The language link brings the user directly to the screen with the calculation form.

Clicking on the English link of **Nanoparticle**, a form appears in which either the appropriate numeric value can be entered or an option can be chosen for each input variable needed for the calculation of the "Overall Risk Level" value.

A working version of the toolkit has been created and can be used here for demonstration purposes as well: http://www.s-and-a.ch/dyn/toolkits/nanoparticles.showWorksheet.

Each line in the form has a simple structure. It consists of a label, an input textbox for a numeric or text value, a pull-down selection box or a sequence of radio buttons or checkboxes. A short description of the field can also be added if necessary. In order to get a longer explanation or a hint, the label in front of an input field can be clicked.

At the bottom of the form there are some function buttons: Calculate, Clear, Save, List, List All and Return. Most of these work as they do in other form-based applications:

Calculate - The program calculates the result(s) from the data entered. If all requested data are not specified or there is a contradiction, the system will send an error message. Otherwise a summary page of the data and result(s) will appear in a separate window in PDF format, which can be saved or printed.

Clear - prepares the form for new data entry.

The next three functions require registration and log-in. Registration is free, it can be done for example by entering an e-mail address (which can then serve as a user ID) and a password. If the password is forgotten, there is no way to recover it, but on request the system can send a new password to the specified e-mail address.

Save - Registered users of the Toolkits site are entitled to save the data in the form under a unique name (typically the name of the substance). Saved data can be modified and re-saved only by the same user who saved them in the first place.

List - This function lists the entries created by the present user and already saved for the toolkit, in alphabetical order. Clicking on a name will load the data for modifications and/or recalculation. A type-ahead function can reduce the number of entries appearing to those beginning with the letter-sequence typed in.

List All – This function lists entries created by all users and saved for the toolkit, in alphabetical order, together with the e-mail address of the user who saved it. This concept enables users to benefit from the contribution of other toolkit users, allowing them to contact each other with ease.

Return - brings users back to the Toolkit Home page.

Scenario 2 – Definition of a toolkit

It can happen that someone wishes to create a brand new toolkit. First of all, this has to be carefully designed with all calculation details. Its specifications can be defined in a single document, as was done in the cited article on the Nanoparticle ExposureToolkit.

The specifications of the new toolkit can then be sent to the Toolkit Administrator (TKA), who formulates and fills the tables of form definitions and calculation instructions, constants, text for the labels, help files, etc. In theory, anyone can do this by oneself, but in practice the TKA probably has more experience in the matter, and has the database privileges to modify all tables.

If there are special calculation or presentation requirements in the toolkit design, the TKA contacts the programmer who extends the functionality of the package, or suggests a work-around solution.

Once the TKA has entered the data and the calculation flow into the database, the first version of the toolkit is ready for use. The TKA may grant users temporary rights to do the fine-tuning of the toolkit without a need, by the user, to understand the database structure completely. Constants, labels, calculation instructions, the content of help files, etc. can all be modified.. After a modification, results can be immediately verified by switching to the end-user interface.

The following virtual snapshot examples demonstrate how all this may work in practice. They are based on the Nanoparticle ExposureToolkit, but are not identical to the actual working model (which is in fact simpler).

Var	Name/label	Type	Source
V01	Surface reactivity	text	options
V02	Particle shape	text	options
V03	Particle diameter [nm]	number	textbox
V04	Solubility	text	options
...	...	...	...
V19	Severity Score	number	V01+V02+V03+...+V13
V20	Probability Score	number	V14+V15+...+V18
V21	Overall Risk Level	text	Nano_Risk (V19,V20)

Var	Option / range	Value	Display
...	...	...	...
V02	tubular or fibrous	10	
V02	anisotropic	5	
V02	compact or spherical	0	
V02	unknown	7.5	default
...	...	...	...
V03	1 - 10	10	
V03	11 - 40	5	
V03	41 - 100	0	
V03	unknown	7.5	default
...	...	...	...

Please note that in our example only the calculation of the "Overall Risk Level" requires a tailored programmed procedure (Nano_Risk), and even that could be done with a generalized built-in function.

At the point when the final touches on the tables of the new toolkit are finished, users need to do no extra work. The TKA, however, has to insert the new toolkit in the public list on the Toolkit homepage, revoke user rights to do further changes in the toolkit definition tables, notify the translators, etc.

Later on, if further changes have to be made in the definition tables, the fine-tuning procedure can be repeated. During the modification period end-users do not have access to the toolkit.

Scenario 3. – Translation of a toolkit into other languages

In order to translate a Toolkit into another language, permission must be granted by the TKA. This permission will depend on evidence that the person(s) doing the translation is/are capable of the task – after all, there may not be any speakers of the language among the managers of the Toolkit site.

The permission will include the provision of access rights to the tables. Translators, like authors, will use forms accessible through the browser. These forms are very simple: each row has the English version of the toolkit text and an input box for the translated text. Numeric values, calculation algorithms, form definitions, etc. will, of course, need no translation. The interface design will include a separately openable window in which the English version will appear in the original context.

Like authors, the translator can also check the translated version while working on it. The texts not yet translated will appear in the English original, with a different colour indicating, that they have not been translated yet.

The following screen-shot demonstrates how a translation window will look in practice:

Var	Name/label in English	Target language
V01	Surface reactivity	
V02	Particle shape	
V03	Particle diameter [nm]	
V04	Solubility	
...	...	...
V19	Severity Score	
V20	Probability Score	
V21	Overall Risk Level	

Var	Option / range in English	Target language
...	...	...
V02	tubular or fibrous	
V02	anisotropic	
V02	compact or spherical	
V02	unknown	
...	...	...
V03	1 - 10	
V03	11 - 40	
V03	41 - 100	
V03	unknown	
...	...	...

After the translation work is finished, the TKA will revoke you're the translator's access rights to the database tables, and will insert the new language version in the public list on the homepage.

Should the English version be changed subsequently, the TKA will notify potential translators, who can then request modification rights in order to translate the modified texts. The interface for such modifications will be identical to the normal translation interface.

Behind the scenes – the technical background

The application is Internet-based both for authors and the end-users of the toolkits.

The package is based on Oracle, a robust and widely used database management system. The programs were developed in PL/SQL, a native, high-level, structured programming language of the Oracle system, fully adapted to the needs of Internet-based applications. In addition to the usual operations of a relational database, mathematical calculations can be programmed in it without difficulty. Javascript technologies such as AJAX will also be used for user interface functions.

The end-user interface is multilingual and uses Unicode encoding, enabling the use of all languages, whatever writing system they use. The author interface is in English only. In the database environment

the creation and adjustment of other language interfaces is automatic. Entering the translations of screens, messages, help files, etc. into the system is a fast, simple and easy-to-administer exercise.

Although all toolkits are different from each other, most of their particularities can be easily specified through the use of relational database tables and related generalized subroutines. As a consequence, there is no need for additional programming in the case of the majority of toolkits: the algorithms, like simple data, can be easily entered into the database. The Nanoparticle Exposure Toolkit is a perfect example of this principle. While testing a new toolkit, the tuning of parameter values for calculations is also a simple matter.

More complex toolkits can also be developed with the software. In order to do this, the capability of the package may have to be extended by new functions in some cases. These functions, however, can be programmed in a generalized way, and once they are developed for a toolkit, can be applied in other toolkits as well. The case of ICCT can be used as an example. In determining the hazard class of a substance, the assigned risk phrases can be applied as a shortcut. For substances with an existing ICSC card, this information can be automatically obtained from a public collection or database through an appropriately designed procedure. For this reason, this software can be considered to be open-ended.

All toolkits designed this way will have a uniform look-and-feel: identical styles and dialogues will be applied. This will be of great help to end-users during their orientation.

The layout of the forms is very simple. As a result, it will be easy to generate the forms automatically from the underlying database tables. The definition of some elements in the tables may need some HTML knowledge, but only if the presentation goes beyond normal complexity.

Thanks to the underlying database environment, there is an obvious possibility of storing results of calculations in any of the toolkits. Data on any substance can be recorded, changed, re-evaluated and displayed. Every user can display calculations stored by other users, but the modification and delete functions will be reserved only for his/her own entries. The free registration model of Internet sites is well adapted to this system.

The security mechanism of the database is based on a simple model. Authors and translators will have password-protected, form-based temporary access to the definition tables, while end-users will have similar access to the pool of concrete calculation results only. Coordination and management functions will be reserved to the Toolkit Administrator (TKA).

End-user actions on toolkits will be logged in order to be able to generate simple user statistics.

Summary

The outlines of a modern, centralized toolkit creation and maintenance system have been drawn up in this proposal. Without going into technical details, we wanted to prove that such a package can be developed and maintained. We tried to show briefly the working of the package from different aspects. Last but not least, we have created a working model based on an existing toolkit specification. The floor is open for discussions and further ideas as well as for the development itself.