SuperMemo 8

[Update for SuperMemo 8, May 22, 1997]

SuperMemo allows you to **learn fast and forget about forgetting**!

All you have to know to start with SuperMemo is:

1. **Adding new information** with *Ctrl+A*
2. **Learning** with *Ctrl+L*

If you are using ready-made systems such as Cross Country, Video English or Deine Chancen, you will only need to select **Learn.** Everything else is taken care of.

However, you may be surprised to find that SuperMemo even makes it possible to create full-blown hypermedia systems that you can use to let others **learn fast and forget about forgetting**!

See Contents for more information about SuperMemo!

If you are familiar with SuperMemo 7 for Windows, you will find it useful to read remarks presented in blue italics (like this paragraph). These make reference to previous versions of SuperMemo software.

SuperMemo 8 Help Contents

Further reading

If you are familiar with SuperMemo 7 for Windows, you will find it useful to read remarks presented in blue italics (like this paragraph). These make reference to earlier versions of SuperMemo software.

Adding simple items

By pressing *Ctrl+A*, you can add a single piece of information to SuperMemo.
SuperMemo will create two editing fields: one for the question and one for the answer.
You can type in the question,
e.g. ***When was the first version of the SuperMemo program written?***
then click the answer field below and type the answer,
e.g. ***in 1987***.

It could not be any simpler!

After you add a couple of questions and answers, which in SuperMemo are called items, you can choose the button **Learn** and start the learning process.

Here are a couple of things you might want to know about adding new items:

* after you type in the answer, you can press *Esc* to see how the item would look like during repetitions
* if you would like to change the size of editing windows, change the font or add graphics to you items, read about advanced options for adding items
* if you want to add items to another collection (collections of items are called knowledge systems)choose **File : Open** (*Ctrl+O*) or **File : New** (*Ctrl+N*). Remember that the menu is invisible in hidden SuperMemo (press *Ctrl+Alt+F12* to make it visible)
* if SuperMemo is not hidden, you can notice the fact that apart from **Edit : Add a new item** which you activate each time you press *Ctrl+A*, you can also use **Edit : Append new items** that makes it possible to add a large number of items by just typing them in and pressing *Esc* continually (*like in earlier versions of SuperMemo*)

Learning

When you choose the button **Learn** or press *Ctrl+L* (**Learn : Outstanding material**), SuperMemo begins repetitions of the learning material.
Learning is based on repeating questions and answers selected by the program (these questions and answers are called items). The power of SuperMemo is in deciding when and which items should be repeated.
Each item is repeated along the following cycle:

1. the program displays the question
2. the user attempts to answer the question aloud or in his or her mind
3. after answering the question, users chooses **Show answer** at the bottom of the screen
4. the program displays the answer
5. the user compares his or her response with the correct answer and chooses a grade (e.g. **Bright, Good, Pass, Fail**, etc.)
6. the user chooses **Next repetition** to continue repetitions (or **Cancel grade** to change the grade, or **Stop** to stop repetitions)
7. if there is still some material to go through, the program goes back to Step 1 of this cycle

Note that you can use the mouse to go through this repetition cycle or use the keyboard instead. Indeed the only keys you have to use are: (1) *Space* or *Enter* (for all stages and for the grade **Pass**), and (2) grade keys (*5* for **Bright**, *4* for **Good**, *3* for **Pass**, *2* for **Fail**, etc.). You can use both the main keyboard on the numeric keypad.

It is worth to remember that at any time you can see the calendar of your repetitions by pressing Ctrl+B (**Tools : Workload**).

For more about learning see Learning topics and items

Introduction to SuperMemo 8

SuperMemo 7 for Windows, the flagship product of SuperMemo World, has become a major success worldwide. It has gained popularity through its ability to produce a documented increase in the students ability to retain once learned material in memory. SuperMemo 7 used question-answer pairs as the mainstay of its knowledge drilling approach. In SuperMemo 7, question-answer pairs could be illustrated with graphics and with sound. However, more and more users demanded a new approach targeted at knowledge structuring and hypermedia.

SuperMemo 8 for Windows is SuperMemo Worlds entry into the domain of hypermedia publishing that widens the scope of applications of earlier versions of SuperMemo.
Here is a short list of target application areas for SuperMemo 8:

* hypermedia publishing
* hypermedia presentations
* individual and corporate training
* classroom education
* student testing and automatic knowledge evaluation
* learning large bodies of facts and figures by means of repetition spacing (in the manner known from earlier versions of SuperMemo)

SuperMemo 8 is intended, indeed, only as a beginning of a long road of development proceeding along the lines established earlier in the work on developing SuperMemo 7 for Windows.
A major part of SuperMemo Worlds resources goes into market research and evaluation of user needs in reference to software development. This user-oriented approach has catapulted SuperMemo 7 into the position of certifiably most popular and most awarded Windows software ever born on Polish soil.
Our hope is that through user feedback and constant investment in development, the crude piece of software and a mere seed of a wider idea that we presented to you as SuperMemo 8.0 (initially codenamed ***Genius***) will in the future allow SuperMemo World pass the magic barrier of a million copies distributed worldwide, and join the list of classic software applications such as word processors, spreadsheets, and database systems.

Please do send your remarks and bug reports to SuperMemo World. The most useful and inspiring comments are rewarded with a free upgrade to the newest version of the program (Web: **www.SuperMemo.com;** e-mail: **info@SuperMemo.com**).

If you are familiar with SuperMemo 7 for Windows, you will find it useful to read remarks presented in blue italics (like this paragraph). These make reference to earlier versions of SuperMemo software.

Minimum requirements

The minimum hardware and software requirements for SuperMemo are: 486DX processor, 8MB RAM, SVGA (at least 800x600 with 256 colors), CD-ROM drive 2x (for multimedia knowledge systems), sound board (for multimedia knowledge systems) and Windows 3.11 or later. Additionally, you will need about 5MB of hard disk space to proceed with installation of knowledge systems such as Cross Country or Video English; however, you will need much more in case you want to create your own hypermedia knowledge systems.
Although the set of minimum requirements may seem daunting, you may try using SuperMemo on less powerful computers. Depending on the degree of compromise, you may wish to choose only a subset of SuperMemo functionality that is less hardware demanding. In particular, you would want to limit number of components placed in each knowledge element and reduce the use of large multimedia objects.

In case your computer does not meet the minimum requirements, you may opt for using SuperMemo 7.5 for Windows instead (available on CD-ROM with a number of multimedia databases for learning English, other languages, medicine, art, and many more).

If you would like to test SuperMemo before buying it, have a look at the good old SuperMemo 6 for DOS (freeware!).

Installation procedure

To install SuperMemo do the following:

1. Insert SuperMemo CD-ROM into your CD-ROM drive
2. In **Windows 3.11**: type <X>:\INSTALL at **File : Run** prompt in Program Manager
   In **Windows 95**: type <X>:\INSTALL at **Start : Run** on the Taskbar
   (in both cases, X: stands for your CD-ROM drive letter)
3. Chose the interface language that suits you most and click OK
4. Choose **Continue** in SuperMemo Setup dialog box
5. In **Windows 3.1**, if you want to install Video for Windows 1.1 choose *Yes* upon receiving *Install Video for Windows 1.1* query and follow the prompts of the installation procedure
6. To run SuperMemo:
   In **Windows 3.1**, double click the SuperMemo icon in the SuperMemo group in Program Manager
   In **Windows 95**, choose **Start : Programs : SuperMemo : SuperMemo** on the Taskbar

If your SuperMemo CD-ROM comes with commercial knowledge systems, you may notice that SuperMemo itself is hidden behind the displayed learning material. To activate full SuperMemo press *Ctrl+Alt+F12*.

If you use less than 800x600 resolution, you might want to maximize your element window area by doing what follows:

* click Hints on the main toolbar to turn hints off and eliminate the status bar
* make **Window : Toolbar** unchecked to eliminate the toolbar
* maximize the element window

Basic hints and tips for users of SuperMemo with Polish PWN Encyclopedia

If you are using SuperMemo with PWN Encyclopedia on CD-ROM, please skim the following paragraphs to become acquainted with the most essential actions in creating and memorizing a simple database based on the Encyclopedias material:

Fundamentals

* If you find an interesting entry in the Encyclopedia and you would like to remember some of the facts included in it, move the contents of the entry into your customized database in SuperMemo by choosing one of the options from **Edycja : Do SuperMemo** menu:
    1. **Do pytania** allows you to add the entry into the question part of the newly created item that will be added to a SuperMemo database. Before transferring the text, the Encyclopedia will ask you to select the part of the entry you would like to transfer. After that, you should choose the button **Przenies**. SuperMemo will move the selected text to the question part of the newly created item; however, before that happens, you are still given time to reedit your item and to add the answer part. To learn the principles of effectively formulating questions and answers see SuperMemo decalog. The most important thing to remember is to keep items as simple as possible and not to make then ambiguous. NB: SuperMemo will not allow of questions or answers longer than 250 characters. That shall limit your chances of getting over the top!
    2. **Do odpowiedzi** is analogous to **Do pytania**, but the text of the entry (or the selection) will be transferred to the answer part of the item.
    3. **Do obu** adds the text of the entry to both the question and the answers. Although you might think that this option is quite awkward (who needs the same texts in both the question and the answer), you may soon find it the most useful option of all. For example, if your selected text reads *Ludwik Lazar Zamenhof first presented Esperanto in 1887* you might want to add the following item to your database: Question: *Who first presented Esperanto in 1887?*, Answer: *Ludwik Zamenhof*. As you can see, by removing unnecessary texts, the original sentence was useful in editing both the question and the answer.
    4. **Haslo do pytania** may be useful in cases when the only thing you want to remember is the meaning of the entry in the Encyclopedia. In this case, the entry heading will form the question, and the entry contents will form the answer (it is recommended that in the answer you only keep that part of the entry that is most crucial to understanding its meaning). During repetitions (see later), you will grade yourself upon being able to recognize the entry and its semantics. Press *Ctrl+Q* to quickly execute this option.
    5. **Haslo do odpowiedzi** is analogous to **Haslo do pytania**. The only difference is that your question is now made of the definition of the entrys headword(s) and the answer is the headword itself. In other words, you will grade yourself upon being able to correctly recall the name of the entry defined in the question. Press *Ctrl+A* to quickly execute this option.
* Each time you add a new entry to SuperMemo, pressing **Add** completes the transfer of the item and reactivates the Encyclopedia. However, SuperMemo remains in memory and makes adding subsequent items much faster. To switch back and forth between the Encyclopedia and SuperMemo use *Alt+Tab*. Note that SuperMemo remains in memory even after the Encyclopedia is terminated.
* Do you know that SuperMemo is a technology that makes it possible to greatly increase the speed of learning? If not, see General principles of SuperMemo. SuperMemo uses the SuperMemo method.
* Note: except for **Edycja : Do SuperMemo : Nauka** considered in the next point, all

the options and shortcuts presented below refer to SuperMemo, and you have to switch to SuperMemo to be able to use them (you can switch to SuperMemo by using *Alt+Tab*).

* When you want to learn the material you have introduced to your SuperMemo database, choose the option **Edycja : Do SuperMemo : Nauka**, which allows you to launch SuperMemo and start repetitions with the material you have introduced to your database. During repetitions repeat the following steps: (1) choose **Pokaz odpowiedz** to view the answer, (2) choose the grade to determine how well your remembered the answer to the question, and (3) choose **Nastepna powtorka**. To find out more, read about the linear learning mode in How to learn with SuperMemo. The most important thing to remember here is: if you want to remember everything you have added to your database, you should use the option **Nauka** possibly **everyday!** You should continue repetitions until you see the message *Nie ma wiecej*.

Useful tips

* Note that the keyboard shortcut *Ctrl+L* can be used both in the Encyclopedia and in SuperMemo to start learning new material or repeating material that has already been memorized. It is equivalent to **Edycja : Do SuperMemo : Nauka** in the Encyclopedia, and to choosing the button **Learn** at the bottom of the element window in SuperMemo.
* To run the Encyclopedia, you need to insert the Encyclopedia CD-ROM into your CD-ROM drive. If you want to run SuperMemo without launching the Encyclopedia, place a SuperMemo icon in one of your Program Manager groups (**File : New : Program Item** in Windows 3.11) or create a SuperMemo shortcut (**New : Shortcut** on the desktops pop-up menu in Windows 95). After creating a SuperMemo icon, double click it, and choose *Ctrl+L* to start learning. This will make it possible to make your daily repetitions without inserting the Encyclopedias CD-ROM.
* If you want to learn more about SuperMemo, use the SuperMemo help file. To open help choose *F1* or *F2* in SuperMemo. Those two keys are used to access help in Polish or in English. If you find some clues unclear in Polish help, you might try to look them up in English help and vice versa (those two help files have been developed to a large degree independently)
* If you want to add new items to SuperMemo from sources other than the Encyclopedia, press *Ctrl+A* and type in the question and the answer (*Esc* can be used to switch from the question to the answer, and then to close the item editing areas).

Advanced tips

* You can delete items from SuperMemo by pressing *Del* (in case you added a Spell-Pad component to the item, or in case some of the components are in the editing mode, you will have to press *Shift+Ctrl+Del* because *Del* will be used is used in editing). *Del* works both in the Contents window and in the element window.
* To reedit once added items, choose *E* to put the question and the answer in the editing mode. Additionally you can use the keys *Q* and *A* to respectively edit only the question or only the answer. Press *Esc* when you are done with editing.
* If you want to create items containing images, sound, video, etc. see How to create knowledge systems and then Components available. To link multimedia files stored in the Encyclopedias DATA directory, choose *Ctrl+E* in the active component. For example, to link a video of Pope John Paul II do the following: (1) create a video component (see Components available), (2) choose *Ctrl+E* (remember that the video component has to be active, i.e. marked with a dark border; otherwise you might open a dialog linking files to other components), (3) select PAPIEZ.AVI in the Encyclopedias DATA directory (e.g. D:\DATA\PAPIEZ.AVI if your CD-ROM is D:). You will see the Pope in the video component window. Click the play button to get the message.
* During repetitions, you can always choose *Ctrl+R* to reset (i.e. postpone) currently displayed item. This will shift the item to the end of the queue of items that have not yet been memorized. Use *Ctrl+R* in the following circumstances: (1) when learning new material, if you do not yet want to memorize the item in question, and (2) during repetitions, if you decide that a troublesome item might be rememorized only after

memorizing all other items that have not yet entered the learning process

* During repetitions, you can choose *Ctrl+M* to recommit (i.e. memorize anew) an item. This is recommended if you reedited the items contents in a semantically significant way.

Using SuperMemo authoring

* If you are not a beginner to authoring software, you might wish to try out SuperMemo in its authoring mode that provides a number of advanced options such as creating new systems, using item filters, translating SuperMemo interface to other languages, analyzing the learning process, rescheduling outstanding repetitions, reviewing items at random, and more. Be warned, however, that the interface language of SuperMemo authoring tool is English, and some of the options might badly affect your learning process if used in a misguided way. To turn on SuperMemo authoring choose *Ctrl+Alt+F12*. To turn it off, choose *Ctrl+Alt+F12* again.
* For details on using the SuperMemo authoring tool see the manual in \SUPRMEMO\OPIS directory (you can read the manual with WordPad for Windows 95 or Word for Windows version 6 or later)
* In your Windows directory, you will find the file ENCY32.INI. You might use the entry SuperMemoDirectory in the section [Directories] to modify the SuperMemo directory or entry KNO_Name in the section [SuperMemo] to modify the default name of the database used by SuperMemo when called from the Encyclopedia

How to navigate in SuperMemo hyperspace

SuperMemo keeps knowledge as a collection of pages called here **knowledge elements**. Because of the specific needs of the process of learning, knowledge elements are usually composed of a few short sentences which makes them differ from pages of information stored, for example, in help systems, which often keep lengthy textual information that is used for reference, not for learning.
Each knowledge elements can keep information in the form of text, sound, images, video, animation, etc. Application of those different information media make SuperMemo a multimedia product.

Moving between knowledge elements in order to locate the one of interest is called **navigation**.
To make navigation easier, knowledge elements are organized into groups and subgroups that form the so-called **knowledge hierarchy**. As you run SuperMemo, you can click the Contents button to see the Contents window which presents the knowledge hierarchy in a graphic form.
In addition to knowledge hierarchy, particular knowledge elements in SuperMemo are often interconnected by means of so-called **hyperlinks**. A hyperlink is a connection between a part of a knowledge element, e.g. a sentence or a picture, and another knowledge element. You can click such a hyperlinked sentence or picture and move directly to the connected knowledge element.

Knowledge elements, organized into the knowledge hierarchy, and interconnected by means of hyperlinks are called here **knowledge systems**. Knowledge systems are used in SuperMemo to store knowledge that is designed specifically for the purpose of effective learning.

Now, having mastered the basic concepts related to knowledge systems in SuperMemo, you can start ... navigating.

Upon launching SuperMemo, two windows appear on SuperMemo desktop: (1) the element window with the current element of the knowledge system (in the front), and (2) the hierarchy window with the graphic presentation of the structure of the knowledge system. At the top of the element window, the navigation toolbar provides access to the following buttons:

- **Contents** - view the location of the currently displayed element in the knowledge hierarchy
- **Search** - search for elements containing a given word
- **<** - go to the previously viewed element
- **>** - go back to earlier visited places after choosing **<**
- **History** - see the list of the most recently viewed elements (you can call up any of the listed elements by double-clicking its title in the history window)
- **Next** - go to the next topic in the system (topics are elements that contain most important synthetic information of the knowledge system)
- **navigation group** - move to next or previous element in the hierarchy, move to elements parent, etc.
- **synchronize button** - determine whether changing the displayed element will be reflected by expanding the knowledge hierarchy in the hierarchy window to visualize the elements location in the system
- **refresh button** (eye icon) - refresh the element display
- **instant translation and transcription buttons** - determine whether mouse moves will result in translating underlying texts to the current translation language or current

phonetic transcription. Note that this button is hidden in knowledge systems that are not provided with translation or transcription registries (to create a translation registry use **File : Import : Translation**, in association with **File : Export : Translation**, if necessary)

bug button (only in debugging SuperMemo) - write a bug note in reference the currently processed element. These notes are stored in file REPORTS\BUGS.TXT in the knowledge system subdirectory. To activate debugging SuperMemo choose **Tools : Options : Data access : Debugging version**.

Apart from using SuperMemo navigation mechanisms, you can also use hyperlinks created by knowledge system author to move between different elements. The presence of hyperlinks is what makes the world of difference between multimedia and hypermedia.

Note that two kinds of knowledge elements are used in SuperMemo:

* **topics** - present synthetic presentation of the learned or presented material, and are mostly used in navigation or review
* **items** - formed as stimulus-response pairs, present topics decomposed along the principle of minimum information, and are mostly used in the learning process *(the same way as in SuperMemo 7)*

As topics are less suitable for repetition, they will be presented only once in the course of learning and dismissed (i.e. removed from the learning process for good). If you want a topic to enter the learning process along other items, choose **Commit** at the bottom of the element window.

General principles of the SuperMemo method

The main advantage you will find in SuperMemo that puts it apart from other hypermedia authoring tools is that it makes possible for you to use a speed-learning technique called SuperMemo. This technique was developed in Poland in the 1980s, and has become the main competitive advantage of software produced at SuperMemo World since 1991.
As you will have to wait a few weeks before you see the first effects of using the SuperMemo technology, please read the following paragraph to understand what SuperMemo is, and why it is worth a dose of your persistence.

Every student knows from his own experience that, after a period of learning, forgetting can ruin even the most meticulously woven structure of the learned knowledge. The obvious remedy against forgetting is repetition. However, repetition is time consuming. In the face of an increasing pressure to learn more and faster, most of the students on this planet waste a great deal of their life for learning things they are bound to forget. Ask a recent graduate about the proportion of the learned knowledge that he or she has retained in his or her brain after a 4-6 year course of studies. If the number produced is greater than 5%, it may only be attributed to the student's poor understanding of the devastating power of forgetting. Indeed, forgetting can ruin lives and careers, it inhibits progress of sciences, medicine, technology and all other branches of human activity where knowledge plays a part.
There is, however, a partial solution to the problem of forgetting: SuperMemo. The solution is partial, because it does not allow you to learn with no effort at all. However, it can be demonstrated in strictly scientific terms that SuperMemo helps you increase the speed of learning manifold. Indeed, it makes it possible to closely approach the maximum natural capability of the human brain to store and retain information. Over a lifetime, SuperMemo allows you to learn 10-50 times faster than by conventional methods, and to reach knowledge **retention** rates of 95% or more. How about gaining the level of knowledge of an experienced professor in just a few years?
SuperMemo minimizes the effects of forgetting and the overall time needed for learning. This is done by scheduling repetitions of knowledge items in carefully determined intervals of time called **optimal intervals**. These are calculated on the basis of two contradictory criteria:

* Intervals should be as long as possible to obtain the minimum frequency of repetitions, and to make the best use of the so-called **spacing effect**, which says that longer inter-repetition intervals, up to a certain limit, produce better memories
* Intervals should be short enough to ensure that the knowledge is still remembered.

In practice, these two criteria translate into the following one:
Intervals should be as long as it is necessary for a selected, small fraction of knowledge to be forgotten. This fraction, called the **forgetting index**, can vary from 3%, for slower and very exact learning, to 20% for blitz-learning characterized by lower knowledge retention. If the forgetting index drops below 3%, the knowledge acquisition rate becomes unacceptably low. On the other hand, the acquisition rate peaks at the forgetting index of about 20%. Above that value, both the retention and acquisition rate decrease. Note, that one can compare traditional learning, in which repetitions are arbitrarily set in time, to time-optimized learning with the forgetting index above 50%. Such learning is not only slow, but above all, the retention of knowledge may be unacceptably low, and in consequence, instead of remembering things which are important, the student remembers only things which are easy to remember.
As optimal intervals differ for particular facts or rules that are to be remembered, SuperMemo requires that the learned knowledge be split into smallest possible pieces called **items**. If items were to be repeated collectively, e.g., as chapters of a textbook in traditional learning, the overall optimal intervals would have to be as short as the optimal intervals for the most difficult subitem in the set. This would make learning even less effective than learning by means of classical methods. Splitting knowledge to small pieces makes it

possible for SuperMemo to determine an independent repetition spacing for each of the items.
Simplicity of items is perhaps even more important for cytophysiological reasons. Simple items make it possible for the right neuronal synapses to be fully stimulated at the right time. Complex items result in trains of diversified nervous impulses that result in unpredictably intricate molecular memory patterns, which make optimization of repetition spacing difficult or not possible at all.
Optimum repetition spacing and simplicity of items are a formula for speed and high retention. However, the speed of learning is not the most important element of education. It is the quality of knowledge and its representation that count most. SuperMemo allows you to learn very fast, but it is still your responsibility to select the learned material and represent it in the most suitable form.

Summary

1. The key to effective learning is in minimizing the number of repetitions necessary to maintain knowledge in the brain.
2. SuperMemo optimizes the process of learning by approximating **optimal intervals** that should separate repetitions of knowledge.
3. The knowledge memorized by means of SuperMemo must be split into smallest possible pieces called **items**.
4. Applying SuperMemo, one must not forget that the quality of learning will depend on selection of the learned material and the way it is split into items.

For some popular scientific insight into SuperMemo see:
Theoretical background of SuperMemo
Scientific conquest of the forgetful memory

If you are not scared by mathematics, you can find out more about SuperMemo optimization algorithm used in spacing repetitions. If you would like your application to use the SuperMemo method, read about SM8OPT.DLL.

SuperMemo method in SuperMemo 8

All elements that enter the learning process in SuperMemo are henceforth presented to the user in strictly determined intervals of time upon choosing the option **Learn : Outstanding material** (*Ctrl+L*). Those intervals are determined by the SuperMemo method. Repetition spacing (i.e. computing optimum inter-repetition intervals) has been and continues to be the mainstay of popularity of SuperMemo software due to its ability to eliminate the problem of forgetting in learning. If you continue working with SuperMemo regularly, you are guaranteed to achieve a desired level of knowledge retention that is determined by the parameter **Options : Learning : Forgetting index** (forgetting index is the default requested percentage proportion of knowledge elements that can be forgotten at the moment of repetition).

Important! Only one user can learn with a given knowledge system. For example, if two persons are to use Cross Country or Video English on one computer, each of them should have a separate copy of the system (about 3 MB).

New algorithms applied in SuperMemo are theoretically far more sound than the algorithms used in the latest releases of SuperMemo, and have been based on a large body of data collected while working with SuperMemo 6 and SuperMemo 7. The major improvements over SuperMemo are:

* mechanisms for faster determination of element difficulty
* establishing the concept of absolute item difficulty (in contrast to relative difficulty expressed by SuperMemo E-Factors)
* mechanisms for on-line determination of the optimum shape of the OF matrix optimum intervals used in repetition spacing

See also: New SuperMemo algorithm

How to learn with SuperMemo

You can choose two modes of learning:

1. **linear mode**, in which you choose to learn elements as they are stored in the knowledge system by pressing button **Learn** at the bottom of the element window *(this is exactly the mode you know from earlier versions of SuperMemo)*
2. **selective mode**, in which (1) you navigate towards the knowledge element of interest and (2) choose either button **Commit** that will enter the displayed element into the learning process or button **Review** that will enter the displayed topic with all its subordinate items into the learning process. Both buttons: **Commit** and **Review,** you will find at the bottom of the element window

In both selective and linear modes, the repetition proceeds along the following cycle *(same as in SuperMemo)*:

1. displaying question components of the elements
2. upon choosing **Show Answer**, displaying answer components of the element
3. displaying the grade panel *(note that unlike in SuperMemo, grade names are customizable with **Tools : Options : Language : Localization table**)*
4. choosing the grade on the part of the student
5. moving to the next element in the repetition cycle or returning to the navigating mode (if there are no more items to learn/review)

During repetitions in linear learning mode you can make a frequent use of the following commands:

Reset - reset the currently displayed item, i.e. remove it from the learning process and put it at the end of the pending queue to learn it again later. You should reset all items that are either difficult or less important. Reset will postpone the learning of the item until after you learn the rest of the material included in the knowledge system. Use *Ctrl+R* keyboard shortcut to reset items in the middle of the repetition cycle. If item is completely unimportant you should use **Dismiss** or **Delete**.

Dismiss - dismiss the currently displayed item, i.e. remove it from the learning process for good. You should dismiss all items that you will never want to learn again. If the item is not worth future reference, you can **Delete** it instead. Use *Ctrl+I* to dismiss (ignore) items in the middle of the repetition cycle.

Commit - reset the currently displayed item (like with **Reset**) and commit it again to the learning process. Use **Commit** to recommit items that have been reedited or gained new significance and you want to make sure you do not forget them before the lapse of the interval determined by the SuperMemo method. Use *Ctrl+M* to recommit (memorize) items in the middle of the repetition cycle. Note that unlike the button **Commit**, the command **Commit** on element pop-up menu makes it possible to determine the interval after which the first repetition will take place. Press *Enter* to schedule the item after the default interval proposed by the SuperMemo method.

During repetitions you may wish to change the wording of some items. You can do it in one of the following ways:

* press *Q* to start editing the first question component *(this will be most useful in question-answer items as used in SuperMemo)*
* press *A* to start editing the first answer component *(again you will find it most useful in question-answer applications)*
* click the selected component to start editing it (this will only be possible if you set **Options : Mouse : Edit texts : On click** to true). Use *Esc* to complete editing
* press *E* to enter editing mode in all text components. Introduce the changes and use *PgUp, PgDn,* or *Tab* to move between components. Use *Esc* to go back to presentation mode

If you use less than 800x600 resolution, you might want to maximize your element window area by doing what follows:

1. click Hints on the main toolbar to turn hints off and eliminate the status bar
2. make **Window : Toolbar** unchecked to eliminate the toolbar

maximize the element window

See also Learning topics and items

Learning topics and items

Knowledge elements in SuperMemo may have a form of:

- topic, which is a synthetic presentation of part of the learning material
- item, which is a particular test or exercise taking part in the learning process.

Topics, unlike items, are used to review the learning material, but do not take part in repetitions scheduled by SuperMemo (this comes from the so-called minimum information principle that is crucial for effectiveness of learning with the SuperMemo speed-learning technology used in SuperMemo).

Items present simple stimulus-response pairs used in the learning process conducted along the SuperMemo method. They are not very useful if you review the material; however, they are basic for the effective learning process.

When you choose the button **Learn**, you will be asked to practise all items that SuperMemo deems necessary for sustaining your memory traces. These will only be those items that must be repeated for you to remember. In other words, SuperMemo will try to minimize the number of repetitions to make sure you spend as little time for learning as it is only necessary to keep the material in memory.

If there are no items that require refreshing your memory, SuperMemo will ask you if you want to learn new elements: *Do you want to learn new material?* If you respond with *yes*, you will commit new items to your memory in the sequence they are stored in the so-called pending queue.

Once you commit new items, it will be SuperMemo that will determine the dates of subsequent repetitions, not you! That is one of the requirement of the SuperMemo method. For that reason, it is recommended that you choose the option **Learn : Outstanding material** or choose the button **Learn** as regularly as possible (preferably, once every day).

To commit items to memory you can choose one of the three methods:

1. choosing the button **Learn.** This will make you successively memorize all items of a given knowledge system in the logical sequence of the entire course.
2. choosing the button **Review** at the moment you find an interesting topic in the browsing process. If the topic you have on the screen is of special interest for you, by clicking **Review** you will memorize all items related to that topic.
3. choosing the button **Commit** at the moment you find an interesting item in the browsing process. Unlike **Review**, **Commit** will make you memorize only the currently displayed item.

Note that with **Commit** you can also memorize topics. However, you will probably find it quite annoying to watch whole video scenes during repetitions. After all, learning speed is the primary advantage of using SuperMemo, while video scenes include a great deal of silent material which you do not want to listen to again and again. This silent material is useful only when you review the entire scene with all its phrases placed in the proper situational context.

During repetitions, if you wish to review the parent topic of the repeated item, click the double up-arrow button on the navigation toolbar.

Depending on your grades in the learning process, particular items will be asked more or less frequently. This is why SuperMemo will make sure that **you do not waste time on reviewing material you already know**.

Important: On a given day, the learning process is complete only when the Outstanding parameter in the Statistics window displays 0+0. To bring up the statistics window, press *F5*. The window will be placed on the left side of the screen.

How to create knowledge systems

The best thing about creating knowledge systems in SuperMemo is that it is entirely interactive and there is no compilation stage. After you create your system, you only click the **Learn Only** button in **Options** (or **ReadOnly** if your system is designed for presentation rather than learning purposes), and follow that with **Hide SuperMemo** (this hides the **SuperMemo** shell behind your application). This way you can have simple presentations or courses developed literally in minutes, and later distributed on CD-ROM media.

To create your own knowledge systems, you have to activate full-access SuperMemo by making sure that SuperMemo shell is not hidden. Press *Ctrl+Alt+F12* if this is not the case. Press *Ctrl+Alt+F12* also if you are not sure in what state SuperMemo is. Finally, set **Tools : Options : Data Access : Access Mode** to **Full Access** (you can open the **Options** dialog by pressing *Ctrl+P*).

If you want to create simple question-answer items, read Adding items.

Otherwise, you can create more advanced knowledge systems by following the steps listed below:

1. create knowledge hierarchy in the Contents window. Use commands **Add, Sibling**, and **Insert** to expand the hierarchy. Drag and drop knowledge elements or use **Delete** to modify once created structure. See Hierarchy operations for more.
2. create individual knowledge elements: (1) Ctrl-click a selected element in the hierarchy to open the element window, (2) drag and drop element components from the editing toolbar in the element window (see Components available), and (3) change the properties of added components to create the final shape of the element (see Component pop-up menu).

To open the editing toolbar, click the **Edit** tab at the top of the element window.
To drag and drop components onto the element window click three spots in turn: (1) selected component on the editing toolbar, (2) top-left corner of the components requested location in the element window, and (3) bottom-right corner of the components location. To find out more about the types of components see Components available.
To fill out the content or to modify the properties of individual components, right-click the component and choose appropriate options from its pop-up menu.

Note, that all objects used in creating knowledge elements are stored in indexed object registries for reuse and search purposes. Creating registries is automatic. Each object maintains a list of its users. If the list of users is empty, the object is deleted with or without confirmation (images, sounds, or videos are examples or objects whose deleting must be confirmed, while texts or lexicon entries are deleted without confirmation).
To reuse registry objects, right-click the newly created component, choose **Link registry member** on the pop-up menu, and choose the linked object in the registry that appears on the screen. For more about registries see Using registries.

The best inspiration for creating ones own knowledge system might come from having a look at commercially released SuperMemo titles.
Note: having created a knowledge system, you will certainly be glad to know that to convert it into a commercial multimedia CD-ROM title is much easier than you might think. See Releasing ones own CD-ROM title.

Adding customized items

To start adding items with customized fonts or colors, remember to activate full SuperMemo (e.g. by pressing *Ctrl+Alt+F12* and setting **Data Access** to **Full Access**).
To append items using the default classic template, just press *Ctrl+A* repeatedly. If you would like to use your own template, with your own colors, fonts and location of windows, do the following:

* define a template item that will contain the components in their requested locations (for example, you might create a text component for the question part of items, and a text component, marked as Answer on its pop-up menu, for the answer part).
* save the template with element pop-up menu command **Template : Save As Template**. Provide the template with a name, e.g. *Classic*.
* specify the default appending template with **Options : Appending : Default item template**. Use the template name provided in the previous step (e.g. *Classic*). Alternatively, you can use **Tools : Append nodes** and double click the **Default template** field to specify the template used with all append nodes (even if you define no append nodes, SuperMemo creates an append node called *Default* that you can use right after creating a new knowledge system).
* in the Contents window, choose the node to which you want to append all new items by default, and choose **Make hook** on the context menu. Alternatively, you can use **Tools : Append Nodes** and double click the **Hook Node** field to specify the title of the element that should become the hook of the particular append node.
* press *Ctrl+A* each time you want to add a new item to your knowledge system. All your items will be appended to the hook hierarchy node and will initially be created along the default template specified earlier.

Note that when you append new items with *Ctrl+A*, SuperMemo will at some point suggest that you change the default append node. See Append Node Change to learn more about this possibility.

Components available for creating knowledge elements

To be able to create new components in SuperMemo, you have to make sure that full SuperMemo is activated (press *Ctrl+Alt+F12* if that is not the case) and that **Tools : Options : Data Access** is set to **Full Access**.

To drag and drop components onto the element window click three spots in turn: (1) selected component on the editing toolbar (**Edit** tab at the top of the element window), (2) top-left corner of the components desired location in the element window, and (3) bottom-right corner of the components location.

The following are components available on the edit tab in the element window:

- text component
- Spell-Pad component
- image component
- sound component
- video component
- script component
- binary component
- circle
- rectangle
- rounded rectangle
- line

Each of the components can be placed in three of the following modes:

1. editing mode, in which parameters of the component can be changed (e.g. text typed in, size of the component changed, etc.)
2. dragging mode, in which the component can be dragged with a mouse to a new location (or resized)
3. presentation mode, in which the component is displayed read-only in the form as it will be seen by the student at repetitions

You can switch components between the three modes individually (by choosing an option on component pop-up menu activated with right-click), or all of them at once by clicking one of the mode buttons: editing, dragging and presentation mode buttons.

Text components are used to display text in topics or items. To create a text component (1) click on the text component button, (2) click the top-left corner of the created text component area, (3) click on the bottom-right corner of the text component area, (4) type in the text of the text component. To see the text as it will appear to the student, click the presentation mode button (pie-chart button presented rightmost above). You can change the properties of a text component by choosing options on the component pop-up menu. To open the menu, right-click the component. To see the function of particular pop-up menu options, drag the mouse over the option in question. See below for a general description of pop-up menu options for different components.

Spell-Pad component is used to test the students ability to spell (type-in) a given text. To create the Spell-Pad component, repeat the three-click actions as in creating the text component, and type in the text which the student should practise at repetitions. To test the Spell-Pad, click the presentation mode button (pie-chart button presented rightmost above). To change it properties, right-click the mouse to open the component pop-up menu. To provide several optional correct responses, separate them with slash characters. For example: *color/colour*.

Image component is used to present a graphic bitmap to the student. Top create the image component, repeat the three-click placement actions and import a desired bitmap file (right-click to get the pop-up menu, and choose **Import file**).

Sound component is used to illustrate topics and items with sound. To add sound to a newly created component, choose **Import file** from the component pop-up menu.

Video component is used to illustrate topics and items with video. To add sound to a newly created component, choose **Import file** from the component pop-up menu and import a

previously captured AVI files.

Script component is used to move, hide, and otherwise affect other components through simple script language programs. To learn how to write scripts see Writing scripts

Binary component is a user-written executable module (EXE or DLL). Writing EXE and DLL files requires some programming knowledge.

Shape components: circle, rectangle, rounded rectangle and line are useful in creating mind-maps and can form graphic background for the remaining components. Note that shape components in presentation mode do not have pop-up menus. To change shapes to editing mode click the editing mode button on the toolbar.

Hierarchy operations

The Contents window presents the tree structure of knowledge hierarchy in the currently used knowledge system. This structure can be edited and expanded by means of drag-and-drop operations as well as by means of options available from the toolbar on the **Edit** tab at the bottom of the Contents window (note that full SuperMemo has to be activated with **Data Access** set to **Full Access**):

View - makes it possible to edit the currently selected element by loading it into the element window

Add - adds a new child element to the set of children of the currently selected hierarchy element

Sibling - adds a new sibling element to the set of children including the currently selected hierarchy element

Insert - inserts a new element at the place of the currently selected hierarchy element

Delete - deletes the currently selected hierarchy element and all its children

The **Node** tab at the bottom of the hierarchy window provides the following operations on all node children, or optionally, on all node descendants:

Commit - memorize children of the node, i.e. enter them into the learning process.

Reset - remove children of the selected node from the learning process and add them to the pending queue of element awaiting memorization.

Dismiss - ignore children of the currently selected element in the learning process. This operation is useful in making sure that a collection of topics belonging to a given node does not take part in repetitions. All children of the dismissed topics, however, will by default remain in the pending queue of elements awaiting memorization.

Undismiss - override the Dismissed status of children of the currently selected element and add them to the pending queue, i.e. the queue of elements awaiting memorization.

Item - give all children of the selected element the item status. The item status is important when using **Review** in the element window. **Review** begins repetitions of all item children of the currently displayed element, and it skips all topic children.

Topic - give all children of the selected element the topic status. Note that this will make the option **Review** have no effect until some of the converted children receive the item status again (i.e. have their **Topic** pop-up menu item unchecked). Most importantly, topics do not take part in repetitions. They are only presented once before their associated items are memorized.

Make hook - make the currently selected element the hook of the current append node. Each time Ctrl+A is chosen, new item will be appended to the hook node as its child. This makes it easier to add a large number of simple items to the knowledge system without the need of specifying their position in the knowledge system.

Make root - make the currently selected element the root of the current append node. Each time a children limit is passed at the hook of the current append node, a new hook can be chosen automatically. The new hook will always be a descendant of the root node.

The **Tools** tab at the bottom of the hierarchy window provides the following operations:

Linearize - rebuild the pending queue using the sequence of items in the hierarchy. This will make sure that memorizing new items will proceed systematically through all branches of the hierarchy in sequence.

Search - search for a title among expanded nodes of the hierarchy.

Info - display basic data about the node.

The **Title** tab is used to edit element titles in the Contents window.

You can edit the structure of the knowledge hierarchy by dragging particular elements. The combo box present on the **Edit** tab at the bottom of the **Contents** window determines what exactly happens with elements dropped on the hierarchy:

* by default, element A dropped on element B replaces element B in the hierarchy (option **Replace the target** in the combo box)
* if **Add as last child** is set in the combo box, or when the *Shift* key is pressed during the drag-and-drop operation, element A dropped on element B is moved to become the last child of element B
* if **Add as last sibling** is set in the combo box, or when the *Alt* key is pressed during the drag-and-drop operation, element A dropped on element B is moved to become the last sibling of element B

Writing scripts

Program script components make it possible to use simple script language to manipulate knowledge element components (and not only). This makes it possible to create interactive presentations, animation, video subtitling, etc.

To create a script do the following:

1. create an knowledge element in the desired shape
2. add a program script component from the editing toolbar
3. right-click the component and choose **Script** on the pop-up menu
4. write the script and choose **OK**

While writing the script, use the script editors pop-up menu that makes it possible to insert full command syntax into the editor (right-click the editor to open the pop-up menu).

For example, to repeatedly play music from a sound component designated by number tag 3, you could write:

```
PLAY 3
GOTO 1
```

To view component number tags click **148** speed-button on the editing toolbar, or choose **Component Order** on the element pop-up menu. Note, that when you right-click within the element window, you will either open a component pop-up menu (if you click over a component) or element pop-up menu (when you click an empty area within the element).

The best way to learn more about scripts is to inspect script components within knowledge systems shipped with SuperMemo (e.g. Cross Country or Video English).

Note: apostrophe is reserved in scripts for delimiting string variables. If you need to include an apostrophe in a string, use a backquote instead. For example:

```
TEXTOUT 3 I`m delighted to have met Monica
```

Script commands used in SuperMemo

To insert a comment into the script, put it in a separate line and precede it with a semicolon

Syntax: ;<Comment>

To display a given registry image in an image component use DISPLAY. Its first argument is the number of the image component in the element. Its second argument is the name of the image registry member

Syntax: DISPLAY <Component> <Image Registry member>

To start a Drag&Match exercise, use DRAGMATCH. For multiple labels and multiple images use

Syntax: DRAGMATCH [<Label List>] [<Image list>]

For example: DRAGMATCH [1,3,4,5] [6,7,8,9]

For single image component with specified hit areas and multiple labels use

Syntax: DRAGMATCH <Component> [<Label List>] [<Rectangle List>]

For example: DRAGMATCH 1 [1,2] [(100,100,30,20),(150,200,50,20)]

Use **Edit Hit-Area** on the image pop-up menu to determine the positions of rectangles without worrying about coordinates.

To determine if labels erase background of the image on which they are dropped in Drag&Match exercises, use DRAGTEXTMODE

Syntax: DRAGTEXTMODE OPAQUE|TRANSPARENT

To change a font in a text, sound or Spell-Pad component, use FONT.

Syntax: FONT <Component> <Font registry member>

To execute a frame of an animation sequence, use FRAME. The <Component> argument specifies the image component within which the animation is executed (alternatively, BACKGROUND can be used to execute animation in element entire background). <Image Registry member> indicate the image that makes up the executed frame

Syntax: FRAME <Component>|BACKGROUND <Image Registry member>

To change the flow of script execution use GOTO with script label or script line as an argument:

Syntax: GOTO <Label>|<Line>

To hide a component use HIDE

Syntax: HIDE <Component>

To design Point&Click exercises use HITTEST. Use an image that displays the hit area as the first argument, and a list of points that specify the correct hit area.

Syntax: HITTEST <Component> [<list of points: (x1,y1)(x2,y2), etc.>]

For example: HITTEST 1 [(10,10),(100,10),(100,100),(10,100)]

Use **Edit Hit-Area** on the image pop-up menu to determine the position of the hit area without worrying about coordinates. The hit area may include no more than five points

To jump to a particular label in the script you need to define the label with LABEL.

Syntax: LABEL <Name>

To move a component to a new location use MOVE:

Syntax: MOVE <Component> <X> <Y>

To play a sound, video, script or execute a binary object, use PLAY:

Syntax: PLAY <Component> <Registry member>

To show a hidden component, use SHOW:

Syntax: SHOW <Component>

To play a sound or video without suspending the execution of the script use START.

Syntax: START <Component>

To stop the script use STOP.

Syntax: STOP

To display a particular text in a text component use TEXTOUT.

Syntax: TEXTOUT <Component> ''<Text>''

To change the display status of a component (from hide to show, or from show to hide), use TOGGLE.

Syntax: TOGGLE <Component>

To display another element of the knowledge system use TOPIC with the number of the element as the argument. Note that despite the commands name, you can also call up items with TOPIC

Syntax: TOPIC <Number>

To translate a text component to the currently selected translation language choose TRANSLATE (see **Tools : Options : Language** for interface and knowledge system translation options).

Syntax: TRANSLAT <Component>

To play user recorded sound in the sound component use UPLAY (to display the recorder panel in sound components check **Recorder** on sound component pop-up menu)

Syntax: UPLAY <Component>

To suspend the execution of a script for a number of milliseconds use WAIT.

Syntax: WAIT <Milliseconds>

To suspend the execution of a script until a given frame is displayed in a video sequence, or until a given position is reached in a sound track, use WAITFRAME with the frame or position as an argument.

Syntax: WAITFRAME <Component> <Frame>

To suspend the execution of a script until a grade is provided (e.g. in Drag&Match exercises or Point&Click exercises) use WAITGRADE.

Syntax: WAITGRADE

Changing the existing knowledge systems

Apart from creating new knowledge systems, you may opt for reediting existing ones (remember to activate full SuperMemo, e.g. by pressing *Ctrl+Alt+F12*, and to set **Data Access** to **Full Access**). For example, you can modify the **Cross Country** or **Video English** course released with SuperMemo. Before you begin editing, do the following:

* turn on the full access mode (*Ctrl+P* for **Tools : Options**, choose **Data access** tab and click **Full Access** radio button)
* show SuperMemo (also **Data access** tab, **Hide SuperMemo** check box). You can use *Ctrl+Alt+F12* to quickly turn SuperMemo on and off.

Now you can see SuperMemo and the knowledge system is in full access editing mode. The fastest way to modifying an element is to click the editing mode button on the Edit tab toolbar or choose Editing mode from the element pop-up menu (to open the pop-up menu, right-click the elements empty area; i.e. not over any component). In your modified course you can, for instance, do the following:

* create your own exercise items for topics that are particularly difficult. You can duplicate an existing exercise by (1) adding a new element in the hierarchy and dragging a template item via Element ClipBox (2) using **Save As Template** and **Load Template**, or (3) choosing **Duplicate** on the element pop-up menu
* change the parameters of components on the duplicated page to suit your needs
* change the texts and add your own comments (e.g. in your native language)
* translate the entire course to your native language (if the translation is not yet available from SuperMemo World). Use *Shift-Click* to display the translation window and modify the translation text.
  In case you translate one of commercial knowledge systems (e.g. Video English, Cross Country, etc.), contact SuperMemo World to discuss the possibility of including your translation in commercial upgrades offered to users of the system (the e-mail is: software@supermemo.com)
* change the location and sizes of particular components in the course (e.g. to adapt it to your lower resolution display), etc.

Using hints and ready-made knowledge systems to learn more about SuperMemo

Use button and menu hints to learn more about particular buttons, menu items and pop-up menu items in SuperMemo.

To view hints do the following:

1. make sure that the hints radio button on the main toolbar is checked and that SuperMemo is not hidden (*Ctrl+Alt+F12* toggles the hidden status of SuperMemo)
2. position the mouse pointer over the button in question, or over the selected menu item (in the latter case, the left mouse button must be down)

Use SuperMemo, Video English, Cross Country or other commercial knowledge system samples to learn more about creating knowledge elements in SuperMemo.

How to translate knowledge systems to other languages

SuperMemo provides you with an option of Shift-clicking individual texts in the knowledge system and displaying the translation of the clicked phrase to the elected language. Alternatively, by (1) depressing the translation button on the navigation toolbar in the element window or (2) choosing **Tools : Options : Mouse : Translation : On mouse move**, translations can be displayed on mouse move over text or sound components.
If you would like to translate SuperMemo knowledge systems to your native language, you can do it in hours by using one of the following algorithm:
To translate the system interactively:

1. Go from page to page and *Shift-click* individual texts
2. Provide new translation in the bottom part of the translation window

To translate the entire system without a need to open individual pages and revisiting once translated text components, you can translate the text registry ASCII file:

1. Choose **File : Export : Translation**.
2. Translate the generated text file to your native language by replacing question marks with the translation.
3. Choose **File : Import : Translation** to import the translation into the knowledge system.
4. In case you found any problems or mistakes, you can repeated the whole procedure again. **File : Export : Translation** retains the texts from your previous import

See also: Translating SuperMemo interface.

How to translate SuperMemo inferface to other languages

SuperMemo provides you with an option of translating its entire interface to another language. The entire translation should take you **no more than 5 minutes!** However, you can only translate the interface of hidden SuperMemo (i.e. the interface seen after you press *Ctrl+Alt+F12*).

To translate the interface of hidden SuperMemo to another language do as follows:

1. Choose **Tools : Options : Language : Localization language** and type in the name of the language (not longer than 8 characters and not equal to ENGLISH or NONE)
2. Press the **Localization table** button on the right
3. On the right side of the Localization table, type in the new name of buttons, menu items and messages
4. If you want to use a national font, choose it by clicking the font icon in the left bottom corner of the Localization table dialog
5. Choose **OK** in the Localization table dialog and your newly designed interface will show up on the screen

To help you translating the interface, here are some hints on the meaning of texts that are most often asked about:

Next is used to select the next topic that follows the currently displayed element
Navigate is used to group all buttons that are used in the navigation in the knowledge system (e.g. **Back, Next, History**, etc.)
Review is used to visit and memorize all descendants of a given element
Commit is used to memorize the currently displayed element (i.e. to introduce it to the process of repetitions)
Reset is used to covert a memorized element into a pending element (i.e. to remove it from the learning process)
Dismiss is used to ignore the displayed element in the learning process
Add is used to add a new element in the hierarchy as a child of the currently selected element
Insert is used to add a new element in the hierarchy in the place of the currently selected element
Sibling is used to add a new element in the hierarchy as the last sibling of the currently selected element
Final drill? is used to ask the user if he wants to proceed with reviewing items that scored grades less than **Good** or **4** in the current session

If you translate the SuperMemo interface to your native language, please e-mail your translation to software@SuperMemo.com (the translation is stored in the TRANSLAT subdirectory of SuperMemo with the extension TRA and the name of the language you have chosen). We would like to arm our CD-ROM releases with as many national interfaces as possible (the size of a single interface file is usually less than 1 KB).

See also: Translating knowledge systems

Main menu

The description of the main menu presented below refers to the authoring mode of SuperMemo. If your main menu contains only the **File** option, press *Ctrl+Alt+F12* first.
The following menu items can be found on the main SuperMemo menu bar:
File - open a submenu with various operations on system files
Edit - (note, that the submenu contents of this menu item depends on the currently active window).

- **Add new items** - add new item to the knowledge system. **Add new items** will add the new item to the currently active hierarchy node and load it with the default item template specified in **Options : Appending : Default Item Template**.

Search - open a submenu with access to various registries
Learn

- **Outstanding material** - proceed with repetitions of the outstanding material *(in the same manner as in SuperMemo)*. In the first stage, repetitions scheduled for the current day and all preceding days are executed. In the second stage, new elements are memorized (if available). In the third and final stage, final drill is executed in which all elements that scored a grade below *good* (i.e. 4) are repeated as long as it takes for all of them to score at least *good*.
- **New material** - proceed with memorizing new material (i.e. the second stage of **Learn : Outstanding material**).
- **Cut drills -** delete items scheduled for final drill (i.e. the last stage of **Learn : Outstanding material**).

View - open a submenu with various element browsers
Tools

- **Options ...** - open a tabbed dialog with options of SuperMemo
- **Workload ...** - display the number of items scheduled for repetition on particular days and months
- **Analysis ...** - display matrices and graphs illustrating the progress of the learning process
- **Mercy ...** - reschedule outstanding items in a selected period of time

Window - open a submenu with various windowing options
Help

- **Front page** - open the help file starting with the front page
- **Contents** - open the help file starting with the table of contents
- **Help in <language>** - open a help file in a given <language> (if available)
- **Tip of Day** - run Tip of Day with useful tips on using SuperMemo
- **System data** - display a dialog with the path and the total size of the knowledge system (only the size of the primary storage is considered)
- **Address** - address, e-mail address and web site address of SuperMemo World
- **About ...** - SuperMemo version, release date, and copyright note

File menu

The **File** submenu on the main menu bar provides the following options:

New ... - creates a new empty knowledge system. To create the system, type in its name (no more than 8 characters) and provide it with extension KNO.

Open ... - opens a new knowledge system for use. *Users of SuperMemo 7 often complain that the full name of the knowledge system has to be provided; including the extension. In SuperMemo 7, just the name would suffice. In SuperMemo 8, by providing the name only, the Open File dialog box changes the directory to systems own directory. Some users suggested adding an extension to system directory; however, this would make it impossible to store knowledge system on CD-ROM without modification (premastering does not accept directory extensions). The fastest way to open a file by typing its name is to follow it with a dot and any character. For example, to open knowledge system BIOLOGY type BIOLOGY.X or BIOLOGY.A.*

Copy ... - copy the currently opened knowledge system to a new location. To copy a system, type in the destination system name (provide it with extension KNO). If the destination system already exists, it will be deleted before proceeding with the copying procedure.

Semi-Copy - copy the currently opened knowledge system without its ELEMENTS subdirectory. This subdirectory contains multimedia objects such as video files, sound files, bitmaps, etc. In hypermedia knowledge systems, these files make up over 95% of the knowledge system size. **Semi-Copy** can be used for fast back-ups of large knowledge systems whose multimedia objects have already been backed up, or have not changed much since the last back-up.

Move ... - copy the currently opened knowledge system and deletes the original.

Delete ... - delete the currently opened knowledge system after confirmation.

Recover ... - recover from knowledge system integrity problems.

Reset - delete all repetition spacing data from the knowledge system. In other words, make it look as it has not been used in the learning process. **File : Reset** is useful if you would like to pass your knowledge system to another user, or even distribute it commercially. **File : Reset** will remove all data pertaining to your learning process, and leave all items in the pending (or dismissed) state.

Import : From SuperMemo... - import a database in SuperMemo format.
Note that you can edit import filters that will let you create a knowledge hierarchy on the basis of string inclusions in the old database. To create a filter, use filter editor the same way as you use hierarchy editor in the Contents window. To each node of the filter, attach the list of strings that will qualify imported items for inclusion in the selected category.
Please note that if your SuperMemo database uses fonts other than the default append font, you will have to select proper fonts manually before proceeding with import. For that purpose, append a parent import item with *Ctrl+A*. Set the font in question and answer fields by choosing **New font** or **Link registry font** on component pop-up menu, add image or sound components if you are importing an audiovisual database, and choose **File : Import : From SuperMemo**.

Import : Q-and-A text ... - import a standard Q&A file as produced by SuperMemo for DOS, Windows, Macintosh or Amiga

Import : Audiovisual files ... - import BMP, GIF or WAV files generated by conversion programs from SuperMemo databases for Macintosh or Amiga. File names are 1.BMP, 2.BMP, etc. and are imported to first encountered image or sound component in subsequent items starting with the import node.

Import : Translation... - import a file with the translation of the text registry to a selected language.

Export : Q-and-A text ... - exports SuperMemo Q&A files by using first question and first answer component of each element

Export : Translation ... - export translation file that, after being modified, can be imported back with **Import : Translation**

Export : Lexicon ... - exports the lexicon in the Q&A format for import to SuperMemo. For example, Cross Countrys. WORDS system have been generated using this option in conjunction with **Import : Q-and-A text and Import : Audiovisual files**

Exit - exit SuperMemo.

1-10 - make it possible to open up to ten recently opened knowledge systems.

File : Recover

Option **Recover** on the **File** submenu can be used to verify the integerity of files used by SuperMemo and to recover from possible problems if such were detected.
Integrity problems in knowledge systems may appear in case of hardware or system software problems. Bad clusters, lost sectors, data corruption, etc. can often be remedied by replacing affected files (e.g. corrupted AVI file, etc.). However, if the problem occurs in SuperMemo index files or files storing optimization data used by repetition spacing algorithm, only **File : Recover** can remedy the problem.

Important! Never run Recover before making a back-up copy of your knowledge system (e.g. with File : Semi-Copy)!

Before you begin recovery, choose appropriate recovery options in **Recovery Options** dialog box.
Recover does not recover from all sorts of errors. Here are some errors that are not remedied with **Recover:**

* damage to binary files such as BMP, WAV, AVI, EXE, etc. Replace these file from the original CD-ROM or from your back-up storage. If you do not have a backup of these files, you will have to remove them by using **Delete** in the relevant registry, or recreate them in any other way.
* damage to OF matrix, RF matrix, Cases matrix, etc. These matrices are stored in the file with the extension KNO. In case of errors indicating damage to these data structure, use **File : Reset : Optimization matrices**. You will lose some of data collected by SuperMemo about your performance with a given learning material, but you will be able resume your work without losing the learning process.
* damage to text files such as TRA, SCR, INI, etc. View these files in any text editor and change their contents. You can also edit legal TRA files with **Tools : Options : Language : Localization table**

See also: When **File : Recover** will not work

Note: If you encounter problems that could not be remedied with **File : Recover**, send a copy of your knowledge system to SuperMemo World for examination and recovery (CDR media recommended).

Recovery Options

After you choose **File : Recover** , before the system check-up and recovery begins, the **Recovery Options** dialog box appears.
The options are as follows:

* **Recover registries** - if you check this box all registries in the knowledge system will be rebuilt and compared with their old versions. All differences will be reported as errors. This process takes pretty long therefore you will want to go through it only if you experience some serious registry problems or you prepare your system for release and want to make sure that its integrity is complete.
* **Check filespace** - this option is active only if **Recover registries** is checked. If you check this box all files in ELEMENTS directory will be checked for their names, duplications, empty slots. etc. If you have made a mistake in backing up the ELEMENTS directory or you experimented with merging two versions of ELEMENTS from two different versions of the same knowledge system, chances are you have introduced some integrity problems in file naming.
* **Build lexicon** - this option is active only if **Recover registries** is checked. If you have done lots of lexicon editing and would like to revert to full lexicon, you can compile the lexicon anew by checking this box.
* **Verify translation** - this option is active only if **Recover registries** is checked. You can use it to make sure that your translation registry is properly sorted and that all text components in the system are translated or reported as having translation missing.

To perform a quick checkup of your learning process, do not check any boxes among **Recovery Options.** On the other hand, if you want to quickly perform the maximum checkup and recovery of the knowledge system, choose **Check all** and then **OK**.

Search

The **Search** submenu on the main menu bar provides the following options:

Find substring ... - search for a substring in texts used in the knowledge system

Find word ... - open lexicon registry with all words used in the system

Text ... - open text registry with all text uses in text, Spell-Pad and sound components.

Title ... - open title registry with all meaningful titles in the system.

Translation ... - open translation registry with all text translations (this option is disabled if no translation is chosen in **Tools : Options : Language : System translation**)

Pronunciation ... - open phonetic transcription registry with all phonetic transcription items available (this option is disabled if no transcription is chosen in **Tools : Options : Language : Phonetic transcription**)

Image ... - open image registry with all bitmaps used in the system

Sound ... - open sound registry with all WAV and MID files used in the system

Video ... - open video registry with all AVI files used in the system

Script ... - open script registry with all scripts used in the system

Program ... - open program registry with the list of all external EXE and DLL files executed by the system

Font ... - open font registry with all user-defined fonts used in the system

Template ... - open template registry with all templates defined and stored by the user

Go to element no ... - go to an element of a selected ordinal number

View

The **View** submenu on the main menu bar provides the following options:

All ... - open the browser with all elements stored in the system (including all deleted element slot that have not yet been filled by new material).

Pending ... - open the browser with all pending elements, i.e. elements that are waiting for memorization in the pending queue.

Memorized - open the browser with all memorized elements.

Dismissed - open the browser with all dismissed elements.

Topics - open the browser with all topic elements.

Items - open the browser with all item elements.

Outstanding ... - open the browser with all elements scheduled for repetition on a given day. These include current repetitions and repetitions from the recent period in which learning was not executed.

Drill ... - open the browser with all elements scheduled for final drill.

Window

The **Window** submenu on the main menu bar provides the following options:

Cascade - cascade all desktop windows
Tile Vertical - tile all desktop windows vertically
Tile Horizontal - tile all desktop windows horizontally
Default - arrange desktop windows in the default manner
Classic - arrange desktop windows in the classical manner *(as in SuperMemo)*
Toolbar - show or hide top toolbar
Next - activate the next available window on the desktop
Previous - activate the previous available window on the desktop
Arrange icons - arrange icons of minimized desktop windows

Options...

Options opens a tabbed dialog with options of SuperMemo. Here is a short list of the most important parameters you can set with Tools **: Options** (see mouse hints for more about parameters not listed below):

* **Learning : Classic layout** - *use classic SuperMemo window layout upon entering into the learning mode*
* **Learning : Forgetting index** - determine the proportion of elements that can be forgotten at repetitions (state it as percentage). Higher value of forgetting index will let you learn faster at the cost of knowledge retention
* **Data access : Access mode** - determines the scope of access given to the user of the system. In read-only mode, the user cannot modify the system. In the learn-only mode, the user can learn with the system but cannot modify its contents
* **Data access : Hide SuperMemo** - makes it possible to hide the SuperMemo shell
* **Data access : Secondary storage** - determines the location of multimedia files that have not been installed on your hard disk, and instead are accessed directly from CD-ROM
* **Appending : Default item template** - name of the item template that will be used when appending new item elements. You can save templates with **Save As Template** on the element pop-up menu. Templates are stored in template registry (see **Search : Template**) and make it easy to add a large number of knowledge elements with same component layout and properties (instead of using **Append new items** you can Ctrl-click nodes newly added to knowledge hierarchy and load templates into the knowledge element by means of **Load Template**)
* **Appending : Default append node** - hierarchy node to which new items are appended by default. By using *Ctrl+A* with default append node, you may add new items without being concerned with the structure of knowledge hierarchy *(in SuperMemo 7, there was no knowledge hierarchy and many users of SuperMemo 8 find hierarchy operations an unnecessary complication)*
* **Language : System translation** - determines translation registry used when Shift-clicking text components. You can use several translation registries with one knowledge system and decide upon the language used by choosing the appropriate file from the combo box
* **Language : Localization language** - determines translation file used to display the interface of hidden SuperMemo. Translation files are available in TRANSLAT directory and have the standard extension TRA. You can change the SuperMemo interface by choosing the button **Localization table**

Mercy

Mercy can be used to shift outstanding repetitions to later time, e.g. after a longer break in learning. It can also help you make your repetitions at an earlier date, e.g. before vacation.

Warning! Rescheduling items makes a departure from the optimum learning process and should be avoided whenever possible!!!

To quickly reschedule outstanding repetitions do the following

1. Choose **Tools : Mercy** (e.g. by pressing *Ctrl+H*).
2. Choose the maximum acceptable number of repetitions per day and type it in at **Number of items per day**. Alternatively, choose the period in which all outstanding repetitions should be done and type it in at **Rescheduling period**.
3. Choose **Update**, e.g. by pressing *Enter*. This will display the rescheduling parameters: number of items per day, length of the rescheduling period, the date on which last outstanding repetitions will be made, etc. (use hints to find out more about particular parameters).
4. If you are satisfied, choose **OK**. Otherwise type in new parameters or choose **Cancel** to quit the dialog box.

To make repetitions before a vacation period, you can shift later repetitions to an earlier period:

1. Choose **Tools : Mercy** (e.g. by pressing *Ctrl+H*).
2. Click the checkbox **Consider future repetitions.**
3. In the **Gathering period** editing field type in the number of days till the end of your vacation.
4. Press *Enter* and compare the date to the right of **Gathering period.**
5. In the **Rescheduling period** editing field type in the number of days till the end of your repetition period before the vacation (compare the date to the right of the edit field).
6. Press *Enter* and compare the date to the right of **Rescheduling period.**
7. If **Number of items per day** does not exceed a realistic value (in your own estimation), choose **OK**. Otherwise type in new parameters or choose **Cancel** to quit the dialog box.

See also: Short history of item rescheduling in SuperMemo

Short history of items rescheduling in SuperMemo

In June 1991, a journalist of Computer World, Andrzej Horodenski, noticed that for lazy users of SuperMemo an option called **Mercy** would be extremely useful. It would allow users to reschedule outstanding repetitions after a vacation period. Just one month later, SuperMemo World released a new version of SuperMemo 6 for DOS that included the suggested option. The first **Mercy** algorithm was based on a solid theoretical ground. The item sorting criterion was to minimize the drop of retention as a result of using **Mercy**. However, the choice of the sorting criterion was not very fortunate. It is easy to notice that an increase in intervals of short-interval items is more deterimental to retention than the same increase for long-interval items. Consequently, users abusing **Mercy** would pile up lots of hard-to-remember items that would recur again and again contributing to overall discouragement of the overwhelmed mind. It is difficult to estimate how many people got hooked on **Mercy** and dropped from among the users of SuperMemo.
A second option was then added to SuperMemo 6. It was called **Wipe** and it was supposed to remove from the learning process all short-interval items with a high degree of difficulty (expressed then by E-Factors). However, **Wipe** might have done more damage to SuperMemo than ill-conceived **Mercy**. Users would often pile up items with **Mercy** and then get them out of the learning process with **Wipe**. Soon they could see that no real progress in learning was taking place. As a result, they would drop out again with detriment to overall popularity of SuperMemo.

In 1994, a new **Mercy** algorithm was designed. The new sorting criterion: minimize the damage to the long-term learning process. The algorithm appeared extremely intricate but has changed **Mercy** beyond recognition. Indeed, it could be seen very soon that the option **Wipe** became entirely superfluous. New **Mercy** would be as abused by the users as the old one, but it would result in less damage and less discouragement. **Mercy** survived and was not removed from subsequent versions of SuperMemo. After all, we believe that we should give people a chance to decide what they need, even if they can occasionally harm themselves.
It became clear only much later that the second **Mercy** had a hidden snag. If abused frequently, it was able to repeatedly lenghten the first interval of newly memorized items (after all, they are supposed to be less important for the long-term learning process). This problem was compounded by the fact that all SuperMemo algorithms implemented in the years 1989-1996 were highly sensitive to delaying repetitions, esp. at the early stages of learning. Consequently, items that were dramatically postponed by **Mercy** and that would score well in repetitions would have reached disproportionately long intervals.

These problems have finally been solved in December 1996 by implementing the following features in SuperMemo:

1. removing sensitivity to repetition delay of the Algorithm SM-8 by indexing optimization matrices with repetition categories rather than repetition numbers (for example, a item repeated twice would be treated as repeated 2.4 times in case of repetition delay).
2. implementing new multicriterial heuristic **Mercy** algorithm that combines its earlier emphasis on minimizing the damage to the long-term learning process with adding extra attention to items that have recently been introduced to the knowledge system.

In SuperMemo 8, **Mercy** is available from **Tools : Mercy**.

Main Toolbar

The main windows toolbar duplicates some of the most often accessed options on the main menu. See toolbar button hints to learn more about functions of particular buttons (pause a mouse pointer over the button in question).

A few buttons on the toolbar require some more comment.

* Element ClipBox can be used to copy elements. Click the rightmost button on the Edit tab in the element window to start dragging the current element. Drop the element on Element ClipBox. You will notice that the trash-bin fills out. You can then drag the element from the ClipBox to the desired destination. For instance, by dropping the element on a newly created element you will duplicate the element stored in the ClipBox. See Using ClipBoxes.
* Component ClipBox can be used to copy element components. Choose **Move to ClipBox** on the component pop-up menu to fill out the ClipBox., or *Ctrl-drag* the component to the ClipBox. Then you can drag the component to the desired destination. See Using ClipBoxes.
* Hyperlink ClipBox can be used to create hyperlinks. Hyperlink ClipBox is filled out the same was as Component ClipBox. By dropping the content of Hyperlink ClipBox on any element in the system, you will hyperlink the component stored in the ClipBox with the drop target element. This way, the student can access the element by clicking the hyperlinked component. See Using ClipBoxes.
* Timer is used to measure repetition time. However, by clicking the timer button, you can turn on and off the timer and do whatever time measurements you need. For example, you might measure video or music subtitling pauses, etc.
* Hints check-box turns on and off hints.
* Crossed out Play button can be used to stop scripts. In case you write a script that enters an endless loop, you can break the loop by clicking this button. You do not need to dislodge SuperMemo from memory.
* The red-cross Cancel button on the toolbar cancels the toolbar itself. You can turn the toolbar back on by choosing **Window : Toolbar** on the main menu.

Buttons at the bottom of the element window

The following buttons appear at the bottom of the element window (only in **Full Access** and **Learn-Only** modes; not in **Read-Only** mode):

- **Learn** - start the three-stage learning process: (1) repeating outstanding items, (2) learning new items, and (3) repeating again items that caused difficulties in stages (1) and (2).
- **Commit** - memorize the currently displayed item or topic (if not yet memorized).
- **Reset** - remove the currently displayed item or topic from the learning process (if already memorized).
- **Dismissed** - ignore the displayed item or topic in the learning process. Dismissed items are never presented to the student while memorizing new items; however, they can be memorized with **Commit.**
- **Review** - memorize (or review, if memorized) all child items of the currently displayed topic.

Element pop-up menu

If you right-click a topic or an item, you will bring up its pop-up menu (make sure that you do not right click over any of the components; this would result in opening component pop-up menu). When the pop-up menu is on the screen, you can press F1 to call up this page of help.

The pop-up menu contains the following options:

Reset - remove the currently displayed element from the learning process and place it at the end of the pending queue. If the element has not yet been memorized, it is shifted in the pending queue to its end. You can use shortcut *Ctrl+R* to reset elements during repetitions: to stop repeating the element (during repetitions of the outstanding materi) or to move it to the end of the pending queue (during final drill).

Commit - memorize the currently displayed element and commit it to the learning process. If the element has already been memorized, **Commit** will reset it and commit it again. **Commit** on the element pop-up menu, unlike **Commit** at the bottom of the element window, makes it possible to specify the length of the first inter-repetition interval in days. You can use shortcut *Ctrl+M* to recommit items during repetitions.

Dismiss - ignore the currently displayed element in the learning process. You can use *Ctrl+I* to dismiss items during repetitions.

Delete - delete the currently displayed element and all its children from the knowledge system. You can use *Del* to delete elements during repetitions.

Duplicate - make a copy of the currently displayed element in the same place of the hierarchy tree.

Move To ClipBox - move the current element to Element ClipBox. See Using ClipBoxes.

Template : Save As Template - save the current element as a template for creating other elements.

Template: Save To File - save the current element to a template file. Note, that no registry objects may be saved in the template file; therefore, only the general outline of components, colors, and parameters are saved. The template files have the extension TPL.

Template : Load Template - load a template from the template registry (previously created with **Save As Template**).

Template: Load From File - load a template from a file created earlier with **Save To File.**

Template : Save To File With Objects - same as **Save To File** but all registry objects are also exported to a separate directory

Template : Load From File With Objects - same as **Load From File** but all registry objects saved with **Save To File With Objects** are also imported

Selection : Link Registry member - makes it possible to link the same registry member to a number of components. All components must be selected, i.e. in editing mode

Selection : Link Registry Font - makes it possible to link the same registry font to a number of components. All components must be selected, i.e. in editing mode

Selection : Alignment and Size - make it possible to size and align all components that are set in the editing mode

Title - edit the elements title.

Color - change the color of the element.

AutoPlay - make the element autoplayable. When autoplayable element is called up, the first playable component is executed. The following are the playable components: sound, video, script and binary components.

Topic - mark the current element as a topic or item. If this menu item is checked, the element is interpreted as a topic. If it is not, it is interpreted as an item.

Multiple Choice Test - mark the current element as a multiple-choice test element. To

make the multiple-choice test functional, you should mark some of its components as wrong or correct answers (i.e. they should not display **Non-test** property on their pop-up menus).

Component Order - edit the order of components appearing in a given element. This option is useful in the following circumstances: (1) changing the order of components to adapt it to a script numbering, (2) making a component playable on entry by putting it in front of the order, and (3) reviewing component parameters (e.g. type, multiple-choice status, answer status, name, etc.).

Editing mode - put the element to editing mode.

Dragging mode - put the element to dragging mode.

Presentation mode - put the element to presentation mode.

Delete all - delete all components of the element.

Component pop-up menu

You can open component pop-up menu by right-clicking the selected component (e.g. text, image, etc.). On components pop-up menus you will find a relevant subset of the following options allowing you to change properties of the component in question. When the component pop-up menu is on the screen, you can press F1 to call up this page of help.

Editing Mode - set the component in the editing mode.
Dragging Mode - set the component in the dragging mode.
Presentation Mode - set the component in the presentation mode.
Delete - delete the component upon confirmation.
Move To ClipBox - move the component to the ClipBox (blue trash-bin on the main windows toolbar). See Using ClipBoxes.
Hyperlink - view or change the hyperlink connecting the component with a selected element. If you set a hyperlink different from zero, the student will be able to evoke the selected element by clicking the component. The easiest way to set a hyperlink is to chose **Move to ClipBox** option on the component pop-up menu, and then to drag the contents of the ClipBox on the element in question.
Answer - mark the component as answer (it will be hidden at question time during repetitions).
Display At ... - determine exactly when the component should be visible (e.g. only at answer time, only on first question, etc.).
Script - edit the script (only in script components).
Link registry member - link a new registry member (e.g. sound, image, video, etc.).
Import File - import a new object file (e.g. bitmap file, sound file, video file, etc.).
Link external file - links the component with a file that is not integrated with the knowledge system. This way you can, for example, link external AVI files from a multimedia encyclopedia; however, after removing the linked CD-ROM from the drive, the linked file will no longer be available in the knowledge system. To integrate files permanently, choose **Import File**
Paste - paste the object from the clipboard.
Play On Click - determine which playable component should be played upon clicking the given component.
Show Panel - determine if operation panel of sound or video components should be displayed.
Extract section - extract a section of a video file. Only the extracted section will be displayed to the student.
New Font - define a new font.
Link Registry Font - link a new font from the font registry.
Stretch Image - stretch bitmap image to fit the image component area.
Edit Hit-Area - makes it possible to edit polygons for HITTEST command and rectangles for DRAGMATCH command used in scripts
Remove Color - removes a given color (e.g. the background) from image components
Center Text - center the components text.
Recorder - use the sound component as a sound recorder for pronunciation drills (only in sound components).
Border - determine the border of the component.
Frames - determine the number of frames skipped in single step on clicking the step button in video operation panel.
Color - change the color of the component.
Line Thickness - change the line thickness in the line component.
Line Style - change the line style in the line component.
Brush Style - change the brush style in shape components.
Brush Pattern - change the image pattern in shape components.
Button text - change the text in execution buttons in script and binary components.

Non-test, Wrong, Correct - determine if the component takes part in multiple-choice test and if it makes up the correct or wrong answer.

Using registries

Registries in SuperMemo are universal object repositories. Each registry is composed of a collection of registry members sorted by their name. For example, image registry is a collection of image members, sorted by image names and associated with individual image objects stored in a given knowledge system. By using registries, SuperMemo minimizes the space used by multimedia files by making sure that each object is stored only once are reused by particular topics and items through access to particular registries. For example, if you import an image file into an image component of a given element (using **Import file**), you can later link another image component with the stored image object by providing its registry name (using **Link registry member**). By default, the name is the same as the original image filename, but you can change it at the moment of import.

The following are the types of registries used in SuperMemo:

- title registry stores the titles used by items and topics of your knowledge system
- text registry stores all text strings used in the knowledge system
- image registry stores all images used by the knowledge system (e.g. BMP, GIF, JPG, etc.)
- sound registry stores all sounds used by the knowledge system (e.g. WAV, MID, RMI, etc.)
- video registry stores AVI video files
- program registry stores external binary files executed by your knowledge system (DLL and EXE)
- script registry stores all scripts used in the knowledge system
- font registry stores all non-default fonts used in the knowledge system
- template registry stores all templates defined in the knowledge system. Unlike with other registries, you cannot link a template to an element or to a component. Instead, you load a template to a knowledge element and replace the content of a topic or item.

Additionally, a lexicon registry stores all words used in a knowledge system. The difference between the lexicon registry and the remaining registries is that individual words are not stored as links to a registry; therefore, deleting a word from the lexicon registry does not delete it from the knowledge system. Instead, this word becomes unavailable for **Search : Find word**.

Text translations are also stored as registries. These are equivalent to text registries and can be easily replaced by means of copying and renaming registry files (e.g. in the REGISTRY directory, COPY GERMAN.* TEXT.* will replace all text in a knowledge system with their German translations, on condition that the GERMAN registry has been created as a translation of the knowledge system, e.g. with ASCII export from the text registry).

Components which use particular objects store only registry references, not objects itself. For example, a sound component may store a reference to a sound registry for its sound file, and to text registry for its text. This approach makes sure that large objects (e.g. video files) can be reused without wasting disk space. Each time you import an new object with **Import file** on component pop-up menu, you add it to the appropriate registry. Similarly, each time you type a new text to text, spell-pad or sound components, you add it to the text registry. Each time you link a registry member with **Link registry member**, the reference count of this particular member is incremented. In other words, SuperMemo knows how many components in the knowledge system use that particular member. If you delete the last reference to a particular member, i.e. its reference count drops to zero, the corresponding registry object will be deleted with or without confirmation. For example, if the last component using a given image is deleted, you will be asked if you want that image to be removed from the image registry. However, if the last reference to a given text is deleted from the knowledge system, its corresponding registry member will be deleted without confirmation.

Using ClipBoxes

ClipBoxes are used as temporary caches for elements and components. They work not unlike the Windows clipboard. First you place an object in a ClipBox and then you can paste it as many times as you wish. When the program terminates, ClipBoxes are emptied. All ClipBoxes have a form of colorful garbage cans placed on the toolbar. When a ClipBox is filled up, the box will bulge with its contents. You can view the contents of the ClipBox by pausing a mouse over it for a short period of time (only if hints are on).
These are the kinds of ClipBoxes available on the toolbar:

* Element ClipBox
* Component ClipBox
* Hyperlink ClipBox

Element ClipBox is used to copy whole elements. To place an element in the ClipBox use one of the following methods:

* drag the element to the ClipBox. You have to locate empty space in the element to start dragging, otherwise the underlying component will be dragged or other mouse actions over component might be triggered
* if there is little empty space available in the element, drag its drag button placed on the Edit tab on the element window toolbar (yellow dragged form on the rights side of the button array)
* if the main toolbar is not visible, you can move the element to the ClipBox without dragging by simply choosing **Move to ClipBox** on element pop-up menu

To duplicate the contents of the element within another element, drag the contents of Element ClipBox on the destination element. This operation will have similar effect to **Load Template** or **Duplicate** (both available from element pop-up menu)**.** Note, however, that **Duplicate** is the only operation in this group that will increase the number of elements in the knowledge system.

Component ClipBox is used to copy components. To place a component in the ClipBox use one of the following methods:

* *Ctrl*+drag the component to the ClipBox (to *Ctrl*+drag means to drag with the *Ctrl* key pressed)
* if the main toolbar is not visible, you can move the component to the ClipBox without dragging by simply choosing **Move to ClipBox** on component pop-up menu

To copy a component, drag the contents of Component ClipBox onto the right spot in the destination element.

Hyperlink ClipBox can be used for creating hyperlinks. A hyperlink is a connection between a component and another element. By clicking a hyperlinked component, you move to the hyperlinked element. To create a hyperlink (1) place a hyperlinked component in the Hyperlink ClipBox as you would do it with Component ClipBox above and (2) create a hyperlink between a component stored in the Hyperlink ClipBox and a given element by dragging the contents of the Hyperlink ClipBox onto the linked element. Those two steps will open a way to accessing the linked element from the linking component by clicking.

Safety of knowledge systems developed with SuperMemo

To ensure the safety of knowledge systems developed with SuperMemo, stick to the following rules:

1. never modify nor move the knowledge system files by means of tools other than SuperMemo
2. keep frequent Semi-Copy backups of the knowledge system (every day is not too frequent)
3. back-up the entire knowledge system every month (e.g. on CDR media)
4. run **File : Recover** on the knowledge system every month (pass through registry recovery at least once a year)
5. contact SuperMemo World immediately in case of encountering errors that could not be remedied with **File : Recover** (best of all, send a CDR back-up of the entire system to SuperMemo World for free inspection and advice)
6. in case of system problems or other errors quit SuperMemo, quit Windows, back-up the system and run **File : Recover**
7. in case you suspect a bug in the program, send a report to SuperMemo World. The most useful reports are rewarded with a free upgrade to the newest version of SuperMemo or even free enrollment in beta-testing programs for future releases

Principles of effectively using SuperMemo

SuperMemo is based on the SuperMemo method that computes optimum inter-repetition intervals in the process of learning. You can use SuperMemo to navigate in hyperspace of the knowledge system and look up the most relevant topics of interest; however, the greatest advantage of SuperMemo over other hypermedia products is its ability to make you remember with little effort.
To make the best use of SuperMemo in learning, select button **Learn** at the bottom of the element window (the window which displays topics and items). **Learn** will guide you through a series of repetitions that will make sure that you spend minimum time to retain once learned material in your memory.

You must not forget, however, that the mere use of SuperMemo is not a formula for success. The mere personality of the student is likely to render the SuperMemo method applied in SuperMemo unusable in a substantial proportion of potential users. The main problem, as in many other human activities, is the perseverance. Only strongly motivated individuals are able to persist in repetitions, which, if not reasonably managed, may become unbearably monotonous. For others, the mental burden and short-term time expenditure seem to overshadow long-term benefits. However, there are yet more snags awaiting an unprepared student. The most important ones are discussed in the following sections:

Selection of material

As SuperMemo makes learning easy, you may be tempted to learn much more than you really need. Only in time will you discover the true extent of knowledge that pertains to your interest and the fact that although SuperMemo is extremely effective, it is not omnipotent and, in limited time, allows you to master only a limited amount of knowledge. Working immoderately, sooner or later, you will be overwhelmed by the amount of repetitions, and will discover the true capacity of SuperMemo (**throughput**), which is proportional to the daily allotment of time.

Working with SuperMemo imposes a strict limitation on the amount of new knowledge that can be learnt in unit time, and requires particularly accurate selection of material. In choosing facts and rules to remember, you should primarily use the criterion of applicability. A multitude of items may seem worth knowing, but because of the limited capacity of memory, only a small fraction of them may be mastered, and the choice should favor items of the highest applicability in day-to-day situations. You will not waste much time if you include items that you would remember anyway just because of their high daily applicability. Therefore, you better always err on the safe side, and include all important items you are not sure you would remember. Concerning the selection of the material to learn, another general principle is that you should first master the most fundamental aspects of the learned subject and get into details at later stages of work. Meticulous dismembering of textbooks page by page is bound to produce excess of details at the cost of principal knowledge. An important psychological aspect of the general-to-specialized approach is that you can soon discover great profits coming from the perfect knowledge of the most fundamental facts in your field and become yet more encouraged to expand your knowledge system into more detailed areas. With amazement, you will notice how new facts and rules slot in in what you have already learned.

Good foundations make it easier to slot in new knowledge structures, provide additional encouragement to work, and free you from the unhealthy impression that you are acquiring knowledge for the knowledge's sake.

Regularity

The most frequent reason for unsuccessful application of SuperMemo is lack of regularity on the part of the student. In fact, irregular repetitions fall out of the definition of the SuperMemo method. No wonder then that for those who lack regularity success is not forthcoming. Regular sessions with SuperMemo are among the most demanding factors and require a lot of will power and determination. Indeed, SuperMemo is not only a good therapy for poor memory, but it also helps to develop iron will. The importance of regularity arises from the fact that SuperMemo computes optimal intervals that should separate repetitions, and it does not allow the student to delay or advance repetitions. It should not be interpreted as a flaw of the method, but as a inherent property of memory which has fixed mechanisms regulating storage and erasure of memories. Every day of delay, not only increases the proportion of forgotten items, but also results in an inevitable accumulation of work that has to be caught up with in following days. This imposes additional stress on the student who, with each day passing by, is less inclined to return to repetitions. In SuperMemo, the best results are obtained when the student uses the method every day the same period of time. Working every second day, although less effective, is much better a solution than working every second week, etc. Systematic work with SuperMemo is a sine qua non of success, and those who do not intend to work regularly should not hope that SuperMemo will make up for their laziness. To paraphrase a well-known adage you might say that **SuperMemo is like sex: you cannot leave it to others**. Treat your repetitions like you treat your daily religious or physiological rituals.

Moderation

It is a frequent case that a fledgling SuperMemo student, amazed with results obtained by a friend who used the method, embarks on an exceedingly ambitious program of gaining new knowledge. The thing (s)he usually overestimates is his or her own learning capability as well as perseverance. SuperMemo does not guarantee that a student will be successful. It requires some skills and working strategy that develop over a longer period of time. The fact that a student can memorize a myriad of items in a short time does not necessarily mean that the same student will have enough perseverance to sustain repetitions of the newly memorized material. Nor does it indicate that the learned items are properly structured and will be maintained in memory as easily as they have been memorized. Two factors contribute to dangers of immoderation:

1. Learning speed (knowledge acquisition rate) decreases substantially during the first year of using SuperMemo and the decline is most visible in the first few weeks.
2. Learning new material is much more a straightforward process than maintaining the acquired knowledge in memory. Badly structured items reveal their vicious intractability only after several repetitions.

Before starting any extensive process of expanding his knowledge system, it is absolutely necessary that the student examine his ability to:

formulate simple, clear and univocal items (with hardly any exception, absolute beginners are bound to form a great deal of ill-structured items),

work continually for a longer time with SuperMemo (only the most persistent individuals can sustain memorathons lasting more than 1 hour).

Moderation, and cool judgment of one's needs and capabilities are, therefore, a very important prerequisite of a successful application of SuperMemo. The recommended time of working with SuperMemo is 5-20 minutes a day for beginners, and no more than 1-2 hours a day for advanced students.

Simplicity

It is a very sorry picture to see an enthusiastic SuperMemo beginner waste his time on learning monster items that not only do not want to stick to his or her memory, but are also absolutely redundant as far as the true student's needs are concerned. Imagine a student who wants to remember all European countries and tries to cram a long list of their names. The typical situation is that he remembers most of the names on the list, but usually fails to mention one or two. The result is that the item is considered forgotten again and again, and that the student cannot observe any learning progress. Using mnemonic techniques, the student could learn to produce the whole list in a determined order, but this is unlikely to represent knowledge which he really wants to possess. A vast majority of educated people is unable to list all the European countries, unless they use the trick of systematic scanning the European map in their imagination. Obviously, learning the list of names is not enough to know something about the geography of Europe, and the scanning trick cannot be done if the student knows the list and nothing more. A good SuperMemo student will not attempt storing the whole list in one item. He will use one of his tricks for knowledge structuring, and split the list into many items. Depending on what sort of knowledge might be useful, the student might use tricks such as:

grouping items by geographical locations of particular countries,

by their affiliation (e.g., EEC, Eastern Europe, Scandinavia, NATO),

by natural language family, etc.

These tricks will certainly introduce information redundancy, but a clever student will certainly use the redundancy to his or her additional profit. Monster items consume much more time and produce much less useful memories than the same items split into many subitems. To ensure full recallability, a multifaceted approach to a stored piece of knowledge should be taken, i.e., all possible combinations of questions concerning a given relation should be asked. For example, if somebody wishes to remember that the leading cause of death is heart disease, which accounts for almost 40% of deaths in western countries, the following question might appear insufficient:

What is the leading cause of death in western countries?

It does not let you remember the proportion of deaths caused by heart disease, nor does it ensure that the 40% statistics refers to western countries. Instead, a multifaceted set of questions could look as follows:

What is the leading cause of death in western countries?

In what group of countries is heart disease a leading cause of death?

What proportion of deaths in western countries is attributable to heart disease?

What disease is responsible for 40% of deaths in western countries?

Is heart disease a leading cause of death in western countries?

Such a collection of items, although redundant and time-consuming, will be learned faster and will leave a more stable trace in the student's memory. The simplicity and multifaceted approach in knowledge structuring are the most powerful tools in making SuperMemo repetitions truly effective in the sense of workload and quality of memories.

Univocality

The fact that your items are simple, doesn't yet guarantee that they are easy to remember. One of the powerful obstacles in creating easy knowledge systems is the problem of interference, or confusion, coming from the fact that two different items might have similar questions or similar answers. For example, imagine that you want to learn something about SuperMemo, and you create a special knowledge systems devoted to that purpose. Consider the two following items of your knowledge system concerning SuperMemo:

1. Q: *What is the name of items which resemble one another?*

   A: *ambiguous items*

2. Q: *How are similar-looking items called?*

   A: *equivocal items*

You need not much experience with SuperMemo to predict that the student who wants to master a knowledge system containing such two items will continually confuse *ambiguous items* with *equivocal items* until he discovers that he has two items whose questions are semantically identical but answers differ. The simple remedy to the problem is to formulate an item with two optional answers:

Q: *What is the name of items which are similar to one another?*

A: *a. ambiguous items OR*
b. equivocal items

In such a case, providing one of the answers is sufficient to classify the item as remembered (unless the user wishes to remember both terms and replaces OR with AND). The problem of interference is particularly visible in attempts to remember numbers. No simple trick of splitting items into subitems can suffice here. The ages long solution to the problem here is mnemonic techniques. One popular mnemonic technique is to represent numbers as pictures and attempt to remember scenes formed by those pictures instead of remembering numbers. For example, instead of having the item:

Q: *What is the value of the constant ?*
A: *3.14*

the student might memorize the universal list of 10 pictures corresponding to 10 digits e.g.:

1 - harpoon (because 1 looks like a harpoon),

2 - coin (because a coin has two sides),

3 - tripod (because a tripod has three legs),

4 - dog (because a dog has four legs), etc.

and formulate the mnemonic item in the following way:

A: *What is the value of the constant ?*

Q: *3.14 because: rhymes with fry and ... on a small tripod a big balloon is being fried (balloon represents the decimal point), suddenly somebody pierces the balloon with a harpoon, and from inside the bursting sphere a big dog jumps on the* student.

Although construction of the pictured scene may seem awkward and time consuming, it ultimately yields items that in most cases are much easier to remember than those composed of dry numbers. The net result is saved time.

Concentration

In the pursuit of more and more knowledge, the student may develop a tendency to give careless answers and hastily jump from question to question in order to reduce the repetition time. There are no dangers related to fast repetitions on condition that the speed is achieved by simplicity of items rather than by negligence in providing answers. It is often possible to automatically answer a question without understanding its important implications. Instead of being semantic, i.e., based on the meaning, the repetitions become syntactic or literal, based on the automatic rendition of the item's wording. Not only is the thus acquired knowledge of little value, but what is more dangerous, the student may become disillusioned with the SuperMemo method because of inadequate progress in the development of desired skills. To prevent such an outcome, the student must constantly control the learning process by asking the following questions:

Is the given item truly important for the skills I want to develop?

Does the ability to answer the question truly ensure that I remember exactly what I want to remember?

If I have any problems with remembering a given item, is it truly formulated in the simplest and most univocal way?

If I have problems with a simple and univocal item, what mnemonic technique could I apply to eliminate my problem?

Are the following elements of the learning process suitably chosen?

The extent of the subject I want to master in the context of my learning capability and availability of time.

The degree to which I want to get into details of knowledge.

The retention level I want to reach.

The amount of time a day I can afford to spend on SuperMemo.

All these questions must recur again and again, and the student must constantly maintain the highest level of alertness and concentration. The degree of improvement obtained by mastering concentration ability is extraordinarily high and worth while. Sloppy work will produce little results and cause disillusionment.

Awareness of the mental stages passed by a typical student

It may be quite useful to realize from the very beginning the stages passed by a typical student during the evolution of his understanding and attitude towards SuperMemo. The observations presented below were compiled from hundreds of letters and questionnaires sent to SuperMemo World, as well as from personal contacts of the author with dozens of students at both basic and advanced levels.

90% of students start their affair with SuperMemo because of good opinions they heard about the program from their colleagues or from press. In other words, the initial attitude is positive and full of expectation. After studying the basics of SuperMemo, reactions of students vary greatly, and are highly correlated with the general level of intelligence and education, especially with respect to exact sciences. A substantial group of students become discouraged because of mechanical and repetitive nature of SuperMemo workouts. In this group, individuals with no university education, or university education in humanities strongly prevail. On the other end of the spectrum are individuals with strong background in mathematics, physics or computing sciences, who seem to have fewer problems with grasping the basics of SuperMemo, and whose first contact with the method is likely to enhance their positive attitude. From this group, SuperMemo has recruited the greatest number of enthusiastic supporters. Only 75% of individuals continue their repetitions beyond one month! The drop-outs provide the following reasons for cessation of repetitions: (1) they do not believe in SuperMemo (10%, no university education or education in humanities), (2) they do not believe in their own learning powers (5%, mostly individuals over 45 years of age, and women - housewife type), (3) they do not have time for repetitions (25%, mostly private entrepreneurs, physicians, lawyers, etc.), and (4) they intend to return to repetitions in the future (60%, students). Only those students who pass the first month of repetitions can truly testify to the usefulness of the method. Among those the approval rate is 95%! Most of them consider SuperMemo an excellent aid in learning (50%) or a useful product (45%); only 5% of those polled did not have a specified opinion or rated SuperMemo as nothing special. SuperMemo World has not received a single questionnaire from a student who had tried out SuperMemo for at least one month, and was generally dissatisfied with the method or the program. However, the attrition rate after the first month remains still high. Having passed a month of repetitions, almost all students wish they had used SuperMemo on a permanent basis, but for hundreds of reasons they cannot cope with busy repetition schedules. The most widespread attitude is to hope to resume repetitions in some unspecified future. Here, the major drawback of SuperMemo becomes visible: it does not produce amazing results for free! It does produce amazing results, but only through hard and systematic work. In busy times of our dynamically growing civilization, we all too often stick to the principle that better an egg today than a hen tomorrow. 10 minutes spent on a business call, reading a scientific paper, or writing a report seem to bring immediate result. The same 10 minutes spent on reasonable repetitions with SuperMemo, can save several hours or even days in the lifetime, but the saving is hard to sense, even for an experienced student. Moreover, after a longer period of repetitions, the student may easily come to the false conclusion that the knowledge has already been mastered, and requires no further repetitions. Thus even the most enthusiastic students often drop weeks of repetitions; the more so, that abusing **Tools : Mercy** *(or in SuperMemo, Mercy and Wipe)* can easily produce the impression that all is OK with the learning process. For an average user, breaks in repetitions regularly make up 20% of the learning time.

You should not forget that you are not the only one who says *I will do it tomorrow, or next week,* or *I think I still remember it all right*. **Forgetting is like radiation: you cannot smell it, you cannot taste it, and when you finally notice its power, the damage is already irreversible.**

Pleasure

There is a sure way to tell if a given student will be successful in his work with SuperMemo or not. If he finds pleasure in long-lasting sessions with repetitory work, he is bound to do terrific job. Despite what meets the eye at first, SuperMemo repetitions do not have to be monotonous. Here is a couple of examples of extra activities that accompany repetitions and provide a splendid ground to make the work with SuperMemo interesting and challenging:

Modification of items that do not seem to comply with the principles of simplicity and univocality.

Removal of items that no longer seem worth remembering.

Cross-analysis of the knowledge system, in order to eliminate discrepancies or redundancy. Note that not all sorts of redundancy are undesirable. For example, having the same foreign word twice in a knowledge system introduces unnecessary noise in the optimization process; however, having the same word in different contexts might increase its semantic associativeness, and increase A-factors *(in SuperMemo, E-factors)* in all related items.

Adding items.

Analysis of the repeated items:

1. What does a given fact or rule imply? Does it agree with other facts I know?
2. In what situations can I use the knowledge I learn? What will the profit or satisfaction be?
3. In what situation did I find that I should remember the given item? Was I really able to make use of knowing it?

Inspecting statistical data used in monitoring the student's progress, etc.

If the repetition process appears to be tedious and monotonous, the student must seriously reconsider all the earlier mentioned prerequisites of success. Lack of enthusiasm is the first symptom of misguided application of the method. On the other hand, if reasonably and consequently applied, SuperMemo appears to be quite addictive, and you will find it absolutely indispensable for your personal and professional success.

Health

As in most human activities, general health helps to achieve better results in the process of learning. It increasingly becomes a common knowledge how to keep healthy and fit. It may still, however, be useful to list the most important principles that should be remembered by every student:

Sufficient amount of sleep. It is during sleep that the conversion of short-term to long-term memories takes place. Particularly, the REM phase of sleep is valuable, and each all-nighter reduces the proportion of REM sleep in the next night. Intense intellectual work may cause increased demand for sleep, and nothing should prevent the student from satisfying it. No one has ever benefited from trying to save time by sleeping less. By trying to emulate Edison, the student is bound to reduce his or her performance. Tiredness caused by lack of sleep is one of the main factors that can ruin a SuperMemo session. On the other hand, 40 minutes of SuperMemo a day are likely to increase your natural demand for sleep by up to one hour!

Physical exercise is absolutely necessary for every successful student. Without it, the respiratory and circulatory systems cannot adapt easily to increased demand for oxygen in intense intellectual work. An unfit student tends to be drowsy and cannot maintain the sufficient level of alertness in long sessions with SuperMemo. Pains of the back, eye strain, hemorrhoidal problems, gastrointestinal disorders, repetitive strain injury, and many other ailments of a hard-working student are greatly prevented by a high level of fitness in physically active individuals. Sports that result in a long-lasting increase in the heart rate and oxygen consumption, e.g., jogging, intense cycling, swimming, etc. are particularly recommended.

It is a common knowledge that **smoking and alcohol** are major preventable causes of a great spectrum of disorders that not only interfere with the process of learning, but are also dangerous to health and, in the long run, life. Alcoholism is notorious for its toxic effects to the brain, and is a major environmental cause of senile dementia (loss of memory in old age). It is also known to reduce the number of NMDA receptors in the brain (receptors involved in memory), and to reduce retention in recall memory tests. It affects both short and long-term memory. Smoking is most renowned for its contribution to heart disease and cancer; however, even for a student who is not much concerned with his or her own future, it is worth noting that smoking also causes hypertension, increased excitability of the nervous system, cerebral hypoxia (reduced oxygenation of the brain) and many other effects that are highly undesirable in learning.

Adequate nutrition is indispensable in successful learning. It is difficult to specify particular guidelines except for the fact that the foods consumed should comply with what is generally considered a healthy diet. For example, everybody knows from one's own experience that a satisfying lunch makes one more inclined to get down to learning, but on the other hand, research demonstrates that the level of mental alertness drops substantially after a meal. It is also known that a low-fat diet prevents arteriosclerosis and should be considered desirable. However, another research shows that reduced blood level of lipoproteins (a group of fat components) has a negative influence on the performance in intelligence tests. An average, healthy diet should satisfy all the nutritional needs of a student, so that no supplements shall be necessary. Regular administration of vitamins, particularly B and C, or mineral preparations, however, might be of preventive value.

Tea and coffee are used by many people as stimulants, especially in case of sleep deficit. While having a positive effect on the alertness, high levels of caffeine and teophylline (active compounds of coffee and tea) may cause restlessness, problems with concentration and, worst of all, sleep disturbances. Both drinks, as well as all sorts of caffeine-containing colas, should be used with moderation.

In the state of rest, the brain uses only a small proportion of **oxygen** that is bound to hemoglobin in the lungs. However, in cases of intense intellectual work, it may use up 30 or more percent of total oxygen. As you will observe, SuperMemo is a particularly tiring technique, primarily because it forces your brain to work at its top gear (repetitions are scheduled at possibly the longest intervals). An average, untrained student will experience an overwhelming drowsiness as soon as after 20 minutes of continuous repetitions. Most of individuals find it impossible to work with SuperMemo for more that 1 hour, and each minute beyond that limit may bring a genuine mental torture. This entails the great importance of the availability of fresh air in the room when the learning takes place. Sufficient amount of sleep, fresh air and enthusiasm for work are the best allies of a student who wishes to spend long sessions with SuperMemo and to produce fabulous results.

SuperMemo decalog

1. Create your knowledge systems using only the most applicable knowledge. Don't get too far into details before forming a general picture of the field you want to master. **Too much pudding will choke the dog**.
2. Work regularly with the SuperMemo method the same amount of time every day. **No pain, no gain**.
3. Don't overwork. **More haste, less speed**.
4. Before embarking on intense expansion of your knowledge system, check if your ability to formulate easy-to-remember items is sufficient, and check what your mental powers are and how much time you can afford to sustain long-lasting repetition process. **Look before you leap**.
5. Make sure that your items are as simple as possible and that they ask about all important aspects of the learned fact or rule. **Small is beautiful**.
6. Save on future repetition time by using mnemonic techniques for the most intractable items. **A stitch in time saves nine**.
7. Concentrate and don't stop thinking. Repetitions should be conscious, not automatic. **All troubles come from ignorance**.
8. If you do not find working with SuperMemo pleasant, then stop Adding items and pay more attention to cleaning your knowledge system from useless or ill-structured garbage. **Nothings succeeds like success**.
9. Keep fit. Do not cut down on sleep, exercise regularly and give up all hazardous habits like smoking or drinking. **Healthy flesh nurtures healthy spirit**.
10. Read the above rules from time to time or formulate them as SuperMemo items that could be placed in one of your knowledge systems. **Learn fast and forget about forgetting** (says SuperMemo World). For example: Question: *What is the best method to use against intractable items?* Answer:

The best method to use against intractable items: use **Delete**!

Selected opinions of users and journalists about SuperMemo

Journalists and reviewers

"**IT REALLY WORKS**. After a month, half an hour a day, my head was packed with a whole spectrum of words and phrases, which I would never suspect myself of being capable to memorize." Andrzej Horodenski, Computer World, Poland, June 8, 1992, in the article "SuperMemo - a brain cell developer"

"For a few months, I have experimented with vocabulary of one of artificial languages (...) and the **success of my work [with SuperMemo] is evident**. Both the vocabulary and time used go beyond all standards (...). It really works!" Pawel Wimmer, Enter, Poland, Oct. 1992, in the article "Praising SuperMemo"

"**The authors do not exaggerate claiming that SuperMemo accelerates the speed of learning from 10 to 50 times** as compared to conventional methods" Maciej Br0mba Pietras, Bajtek, Poland, Jan 1993, in the article "SuperMemo. Learn fast and forget about forgetting"

"**After a month of learning [with Advanced English] (...) newly acquired knowledge started showing in communication with English speakers.** (...) SuperMemo (...) at worst is 90% effective [in knowledge retention]. It's hard to believe, but this is a fact" Tomas Baranek, Cena/Vykon, Czech Republic, Mar 1994, in the article "SuperMemo 7.0 - everybody can do it"

"**The use of the program [SuperMemo] is so simple that it seems impossible it could really work. You change your opinion after a month when you feel the results**" Wojciech Gryciuk, Computer World, Poland, Sep. 27, 1993, in the article "SuperMemo 7 for Windows"

"I installed the program on a computer in the editorial office, and, from time to time, rushed my brain cells under the skull. **The results, despite my irregularity, were sensational**", Adam Chabinski, Chip, Poland, July 1994, in the article "SuperMemo 7.2 for Windows"

"On the market we have a program that accelerates learning from 10 to 50 times. The authors do not lie. **SuperMemo is a true sensation** (...) After half-year work with the program I have no doubt: it really works" Piotr Plociennik, Gazeta Wyborcza, Poland, Oct. 18, 1993, in the article "Fast learning"

"The material which I had tried to master in pain for a longer time the traditional way, I learned thanks to SuperMemo (and a dose of persistence) in a much shorter period of time, and what is also important, with pleasure. One thing is sure, **I am a faithful advocate of SuperMemo (more and more), and can recommend it to everybody**." Dorota Partycka, Swiat PC-tow, Poland, July 1993, in the article "SuperMemo 6"

"... SuperMemo for Amiga is a very good piece of software worth recommending to everybody who wants to learn fast and effectively ... **The algorithm is based on one of the best methods of learning**" Marek Pampuch, Magazyn Amiga, Poland, Sep 1993, in the article "SuperMemo"

Newspaper notes

"**One of the sensations of CeBIT'92 was Polish educational software SuperMemo**, in which the excellent idea was backed up by firm skills in software engineering" Andrzej Horodenski, Poznanski Magazyn Targowy, Poland, May 1993, in the article "Europe ... not so much Eastern"

"**Among the winners [of the Gold Medal at Infosystem'94], probably the best known Polish computer export product - SuperMemo** (...) based on the modern method of learning (...). The authors of the program, which was created several years ago, continue improving the product", Gazeta Poznanska, May 20, 1993, in the article "Gold at Infosystem"

"A domain on which technology can have a significant impact is **education. It is one of the few fields, in which Polish programmers have contributed their share. I have in mind the authors of SuperMemo**" Janusz Kotarski, Computer World, Poland, Jan 25, 1993, in the article "Quo Vadis informatica?"

"The package of Poznan programmers **[SuperMemo] is one of our export specialties and is becoming a world success**" Pawel Wimmer, PC Kurier, Poland, Sep 30, 1993, in the article "SUPER SuperMemo"

"**SuperMemo (...) is a philosophy, maybe even a religion, in which the subject of worship is the unlimited knowledge.**" Jacek Grabowski, PC Kurier, Poland, Aug. 20, 1992, in an article "Forget about forgetting"

Jurors and scientists

"**I would like to congratulate you [Dr.Gorzelanczyk] and Piotr Wozniak on the success of your findings, which are now available as SuperMemo.**" Margaret Jean Intons-Peterson, Editor, Memory & Cognition, USA, Feb. 12, 1992, in a letter to Dr Gorzelanczyk, Medical Academy of Poznan

"**Congratulations! Your software is among the finalists in the SOFTWARE FOR EUROPE** Awards Competition. Our jury has decided that your software is among the finest programs in your home country." Ralph Peter Rauchfuss, IDG Communications Verlag AG, Germany, Nov 14, 1991, in a letter to SuperMemo World

Users in Poland

"In June I have noticed a review of a computer program SuperMemo. (...) I bought this program and used it to learn English. After a month I can say that **this program is excellent**, and can be used to learn any subject, not only foreign languages." Bogdan Kowalewski, Warszawa, Poland, Aug. 15, 1992, in a letter to the president of ESKK school of languages

"**This program is splendid**. Wholehearted congratulations. With impatience I wait for upgrades, esp. for Windows." Janusz Dyczek, Krakow, Poland, July 1, 1992, in a letter to SuperMemo World

"**I will use SuperMemo till the last day of my life**, and I will try to convince everybody that SuperMemo can change his life", Maciej Butrymowicz, Primo Loco, Poland, Nov 5, 1992, in a dedication on a book passed to the author of the method

"**Congratulations on creating the genius and sensational SuperMemo**. I am aware

that after many awards your masterpiece has collected, I can only join the common voice of praise and admiration" Ryszard Maczynski, Warszawa, 1994, in a letter to SuperMemo World

"For a few years now I have been using SuperMemo 6.2 (...). **Your project has changed my approach to learning. Thank you**" Marcin Skuratowicz, Elblag, Nov 1994, in a letter to authors of SuperMemo

"Thank you very much for SuperMemo. **This program not only makes learning easy; it gives me an extra incentive for work**", Maria Janiak, Aleksandrow, Poland, Oct. 30, 1992, in a letter to SuperMemo World

"**SuperMemo gives me more than all my courses of English taken together**", Artur Nowak, Bedzin, Poland, Dec. 3, 1993, in a letter to SuperMemo World

"**Piotr Wozniak hit the nail on the head with his approach to optimization of learning.** Developing his idea into an elegant piece of software deserves only praise", Adam Pawlowski, Krakow, Poland, Nov 9, 1992, in a letter to SuperMemo World

"I love this program! **This is my best program I've got! Thank you**", Piotr Wierzejewski, Poznan, Poland, Aug. 17, 1993, in an e-mail about SuperMemo 7 to SuperMemo World

"**Your software is swell!**", Piotr Kowal, Wesola, Poland, Dec. 4, 1993, in a letter to SuperMemo World

"**My interest in your program comes from the fact that my colleague (...) worked on examination material with SuperMemo. I was extremely surprised to discover that he passed the exam in English top in our school!**", Krzysztof Binkowski, Olsztyn, Poland, Aug. 6, 1994, in a letter to SuperMemo World

"**I have received my TOEFL score and it is 617!!! It wouldn't have happened without your brilliant idea.** Thanks a bunch", Przemyslaw Szaruga, Lodz, Poland, Feb 10, 1994, in a letter to SuperMemo World

"Cramming for exams, I dreamt about software that would test only the material I had problems with ... **SuperMemo made my dream come true**", Bogdan Kandybowicz, Gdansk-Oliwa, Poland, Sep 1, 1994, in a letter to SuperMemo World

"**Being 50 years old, for the first time in my life, I became truly systematic with my work** [repetitions in SuperMemo]", Stanislaw Jachimek, Ruda Slaska, Poland, Aug. 1993, in a questionnaire sent to SuperMemo World

"Reading press reviews of SuperMemo, one might think they are exaggerated. However, it is not so. **It really works.** At last, my computer is used to its best capability. Congratulations.", Pawel Lukasiewicz, Bydgoszcz, Poland, Oct. 1995, in a letter sent to SuperMemo World

"**The new version is marvelous**", Piotr Bajerski, Gliwice, Poland, Mar 1, 1995, in an e-mail to SuperMemo World in reference to SuperMemo 7.5 beta

"This year I passed my final exams. No exaggeration, I passed it thanks to you [SuperMemo World]. (...) **The system [SuperMemo] appeared to be a stroke of genius**. Thanks to SM I gained lots of time others wasted on unnecessary repetitions (...). On the exam, I wrote about the partition of Poland. This portion of material I learned in January [the exams took part in June] (...). Apart from SM I did not repeat anything. I got A and was released from further examinations. (...) I am not boasting. I just state the facts and the praise goes to the

method [SuperMemo] and to your program! ", Lukasz Lubbe, Gniezno, Poland, July 20, 1996, in a letter to SuperMemo World

SuperMemo is the best method to learn foreign languages and many other subjects, Maciej Filipczyk, Wojkowice, Poland, Oct 1, 1996, in a questionnaire sent to SuperMemo World

Users abroad

"I am very pleased with your product SuperMemo, which I have been using for some months now to learn Japanese. **It is indeed possible to learn up to 3000 words per month. Which I admit was very surprising**", Danny Browne, Future Fine Arts Ltd, Scotland, Aug. 23, 1994, in a fax to SuperMemo World

"**I found SuperMemo 7 a really impressive product** with the new audio-visual features", Carsten Haerle, Muenster, Germany, Aug. 1, 1993, in an e-mail to SuperMemo World

"**The idea is fabulous**", Winter Laite, Albuquerque, USA, Aug. 7, 1993, in an e-mail to SuperMemo World

"**Great product**", Hamish MacEwan, Information Technology & Systems, New Zealand, July 20, 1993, in an e-mail to SuperMemo World

Optimization algorithm used in spacing repetitions in SuperMemo

Dr P.A.Wozniak, Sep 10, 1995 (updated Dec 17, 1996)

Below you will find a general outline of the SuperMemo algorithm used in SuperMemo 8. This is the sixth major formulation of the algorithm denoted as Algorithm SM-8. Although the increase in complexity of Algorithm SM-8 as compared with its predecessor, Algorithm SM-6, is incomparably greater than the expected benefit for the user, there is a substantial theoretical evidence, and tentative practical evidence that the increase in the speed of learning resulting from the upgrade may fall into the range of 10-40%.

Scheduling repetitions in Algorithm SM-8 proceeds along the following lines:

1. Inter-repetition intervals are computed using the following formula:

   I(1)=OF[1,L+1]

   I(n)=I(n-1)*OF[n,AF]

   where:

   OF - matrix of optimal factors, which is modified in the course of repetitions

   OF[1,L+1] - value of the OF matrix entry taken from the first row and the L+1 column

   OF[n,AF] - value of the OF matrix entry that corresponds with the n-th repetition, and with item difficulty AF

   L - number of times a given item or topic has been forgotten

   AF - number that reflects absolute difficulty of a given item

   I(n) - n-th inter-repetition interval for a given item.
2. Because of possible delay in executing repetitions, matrix OF is not indexed with repetitions but with repetition categories. For example if the 5-th repetition is delayed, OF matrix is used to compute the repetition category, i.e. the theoretical value of the repetition number that corresponds with the interval used before the repetition. The repetition category may, for example, assume the value 5.3 and we will arrive at I(5)=I(4)*OF[5.3,AF] where OF[5.3,AF] has a intermediate value derived from OF[5,AF] and OF[6,AF]. *For the first time this modification introduced in December 1996 makes Algorithm SM-8 insensitive to delayed repetitions since introducing the matrix of optimal intervals in 1989.*
3. The matrix of optimal factors OF used in Point 1 has been derived from the mathematical model of forgetting developed by Dr P.A.Wozniak. Its initial setting corresponds with values found for a less-than-average student. During repetitions, upon collecting more and more data about the students memory, the matrix is gradually modified to make it approach closely the actual students memory properties
4. The absolute item difficulty factor (A-Factor), denoted as AF in Point 1, expresses the relative difficulty of an item (the higher it is, the easier the item). It is worth noting that AF=OF[2,AF]. In other words, AF denotes the optimum interval increase factor after the second repetition. This is also equivalent to the highest interval increase factor for a given item. *Unlike E-Factors in Algorithm SM-6 employed in SuperMemo versions 6.x through 7.x, A-Factors express absolute item difficulty and do not depend on the difficulty of other items in the same knowledge system.*
5. Optimum values of the entries of the OF matrix are derived through a sequence of approximation procedures from the RF matrix which is defined in the same way as the OF matrix (see Point 1), with the exception that its values are taken from the real learning process of the actual student. Initially, matrices OF and RF are identical; however, entries of the RF matrix are modified with each repetition, and a new value of the OF matrix is computed from the RF matrix by using approximation procedures. This effectively produces the OF matrix as a smoothed up form of the RF matrix. In simple terms, the RF matrix at any given moment corresponds to its best-fit value derived from the learning process; however, each entry is considered a best-fit entry on its own, i.e. in abstraction from the values of other RF entries. At the same time, the OF matrix is considered a best-fit as a whole. In other words, the RF matrix is computed entry by entry during

repetitions, while the OF matrix is a smoothed copy of the RF matrix.

6. Individual entries of the RF matrix are computed from forgetting curves approximated for each entry individually. Each forgetting curve corresponds with a different value of the repetition number and a different value of A-Factor (or memory lapses in the case of the first repetition). The value of the RF matrix entry corresponds to the moment in time where the forgetting curve passes the knowledge retention point derived from the requested forgetting index. For example, for the first repetition of a new item, if the forgetting index equals 10%, and after four days the knowledge retention indicated by the forgetting curve drops below 90% value, the value of RF[1,1] is taken as four. This means that all items entering the learning process will be repeated after four days (assuming that the matrices OF and RF do not differ at the first row of the first column). This satisfies the main premise of SuperMemo, that the repetition should take place at the moment when the forgetting probability equals 100% minus the forgetting index stated as percentage.
7. The OF matrix is derived from the RF matrix by: (1) fixed-point power approximation of the R-Factor decline along the RF matrix columns (the fixed point corresponds to second repetition at which the approximation curve passes through the A-Factor value), (2) for all columns, computing D-Factor which expresses the decay constant of the power approximation, (3) linear regression of D-Factor change across the RF matrix columns and (4) deriving the entire OF matrix from the slope and intercept of the straight line that makes up the best fit in the D-Factor graph. The exact formulas used in this final step go beyond the scope of this illustration.

Note that the first row of the OF matrix is computed in a different way. It corresponds to the best-fit exponential curve obtained from the first row of the RF matrix.

All the above steps are passed after each repetition. In other words, the theoretically optimum value of the OF matrix is updated as soon as new forgetting curve data is collected, i.e. at the moment, during the repetition, when the student, by providing a grade, states the correct recall or wrong recall (i.e. forgetting) (*in Algorithm SM-6, a separate procedure* ***Approximate*** *had to be used to find the best-fit OF matrix, and the OF matrix used at repetitions might differ substantially from its best-fit value*).

8. The initial value of A-Factor is derived from the first grade obtained by the item and the correlation graph of the first grade and A-Factor (G-AF graph). This graph is updated after each repetition in which a new A-Factor value is estimated and correlated with the items first grade.

Subsequent approximations of the real A-Factor value are done after each repetition by using grades, OF matrix, and a correlation graph that shows the correspondence of the grade with the average forgetting index (G-FI graph).

9. The G-FI graph is updated after each repetition by using the expected forgetting index and grade values. The expected forgetting index can easily be derived from the interval used between repetitions and the optimum interval computed from the OF matrix. The higher the value of the expected forgetting index, the lower the grade. Reversely, from the grade and the G-FI graph, we can compute the estimated forgetting index which roughly corresponds to the post-repetition estimation of the forgetting probability of the just-repeated item at the hypothetical pre-repetition stage. Because of the stochastic nature of forgetting and recall, the same item might or might not be recalled depending on the current overall cognitive status of the brain; even if the strength and retrievability of memories of all contributing synapses is/was identical! This way we can speak about the pre-repetition recall probability of an item that has just been recalled (or not). This probability is implicitly expressed by the estimated forgetting index. From (1) the estimated forgetting index, (2) length of the interval and (3) the OF matrix, we can easily compute the most accurate value of A-Factor. Note that A-Factor serves as an index to the OF matrix, while the estimated forgetting index allows one to find the column of the OF matrix for which the optimum interval corresponds with the actually used interval corrected for the deviation of the estimated forgetting index from the requested forgetting index.
10. The estimated forgetting index is computed from the G-FI graph by using normalized

grades that correspond to the expected forgetting index. This comes from the fact that the G-FI correlation refers to grades obtained at intervals corresponding with the requested forgetting index.

To sum it up. Repetitions result in computing a set of parameters characterizing the memory of the student: RF matrix, G-AF graph and G-FI graph. They are also used to compute A-Factors of individual items that characterize the difficulty of the learned material. The RF matrix is smoothed up to produce the OF matrix, which in turn is used in computing the optimum inter-repetition interval for items of different difficulty (A-Factor) and different number of repetitions (or memory lapses in the case of the first repetition). Initially, all students memory parameters are taken as for a less-than-average student, while all A-Factors are assumed to be equal.

Optimization solutions used in Algorithm SM-8 have been perfected over 8 years of using the SuperMemo method with computer-based algorithms. This makes sure that the convergence of the starting memory parameters with the actual parameters of the student proceeds in a very short time. Similarly, the introduction of A-Factors and the use of the G-AF graph greatly enhanced the speed of estimating individual item difficulty. The adopted solutions are the result of constant research into new algorithmic variants. The postulated employment of neural networks in repetition spacing is not likely to compete with the presented algebraic solution. Although it has been claimed that Algorithm SM-6 is not likely to ever be substantially improved (because of the substantial interference of daily casual involuntary repetitions with the highly tuned repetition spacing), the initial results obtained with Algorithm SM-8 are very encouraging and indicate that there is a detectable gain at the moment of introducing new material to memory, i.e. at the moment of the highest workload. After that, the performance of Algorithms SM-6 and SM-8 is comparable. The gain comes from faster convergence of memory parameters used by the program with actual memory parameters of the student. The increase in the speed of the convergence was achieved by employing actual approximation data obtained from students employing SuperMemo 6 and SuperMemo 7.
More research is needed before reliable comparisons of Algorithms SM-6 and SM-8 can be made.

If you would like your application to use the Algorithm SM-8, read about SM8OPT.DLL.

Analysis of the learning process. Graphic presentation of parameters used in Algorithm SM-8

The tabbed dialog available from **Tools : Analysis** in SuperMemo provides matrices and graphs that illustrate the current state of the learning process in the currently opened knowledge system. Some of these graph can be understood without understanding Algorithm SM-8; however, most of them require general understanding of how SuperMemo computes the optimum spacing of repetitions (see Optimization algorithm used in spacing repetitions in SuperMemo for a description of most general concepts related to Algorithm SM-8).

The following tabs and subtabs are available in the **Analysis** dialog:

Distributions

- **Interval distribution** - distribution of inter-repetition intervals in a given knowledge system
- **A-Factor distribution** - distribution of A-Factors in a given knowledge system (note, that the distribution itself is not used in Algorithm SM-8, and merely results from it)
- **Repetitions distribution** - distribution of the number of repetition in a given knowledge system (only memorized items are considered, i.e. there is no zero-repetitions category)
- **Lapses distribution** - distribution of the number of times particular items of the knowledge system have been forgotten (only memorized items are considered)

Curves - four hundred forgetting curves are independently plotted for the sake of computing the RF matrix. These correspond to twenty repetition number categories multiplied by twenty A-Factor categories (note that for the repetition, the columns of the RF matrix are indexed by the number of memory lapses rather than A-Factor). By choosing a proper combination of tab at the bottom of the graph, you can select a forgetting curve of interest. Horizontal axis represents time expressed as: (1) U-Factor, i.e. the ratio of subsequent inter-repetition intervals, or (2) days (only after the first repetition). Vertical axis represents knowledge retention in percent.

Blue circles represent repetitions (the greater the circle, the greater the number of repetitions). Red curve corresponds with the best-fit forgetting curve obtained by exponential regression.

Horizontal green line corresponds with the requested forgetting index, while the vertical green line shows the moment in time in which the approximated forgetting curve intersects with the requested forgetting index line. This moment in time determines the value of the relevant R-Factor. The values of O-Factor and R-Factor are displayed at the top of the graph. They are followed by the number of repetition cases used to plot the graph.

Note that at the beginning of the learning process, there is no repetition history and no repetition data that could be used to compute R-Factors. For that reason, the initial value of the RF matrix is taken from Wozniaks model of memory, and they correspond with the parameters of memory that characterize a less-than-average student (the model of average student is not used because the convergence from poorer student parameters upwards is faster than the convergence in the opposite direction).

Graphs

- **G-FI graph** - G-FI graph correlates the expected forgetting index with the grade obtained at repetitions. You can imagine that the forgetting curve graph might use average grade instead of retention on its vertical axis. If you correlated this grade with the forgetting index (which is 100% minus retention), you arrive at the G-FI graph.
- **G-AF graph** - G-AF graph correlates the first grade obtained by an item with the ultimate estimation of its A-Factor value.
- **DF-AF graph** - DF-AF graph shows decay constants of power approximation of R-Factors along columns of the RF matrix. The horizontal axis represents A-Factor, while the

vertical axis represents D-Factor (i.e. Decay Factor). D-Factor is a decay constant of power approximation of curves that can be inspected with the **Approximations** tab of the **Analysis** notebook dialog.

First interval graph - the length of the first interval after the first repetition depends on the number of times a given item has been forgotten. Note that the first repetition may also mean the first repetition after forgetting. In other words, a twice repeated item will have the repetition number equal to one after it has been forgotten (i.e. the repetition number will not equal three). The first interval graph shows exponential regression curve that approximates the length of the first interval for different numbers of memory lapses (including the zero-lapses category that corresponds with newly memorized items).

Matrices

O-Factor matrix - matrix of optimal factors indexed by the repetition number and A-Factor (only for the first repetition, A-Factor is replaced with memory lapses)

R-Factor matrix - matrix of retention factors

Cases matrix - matrix of repetition cases used to compute the corresponding entries of the RF matrix (double click an entry to view the relevant forgetting curve)

Optimal intervals - matrix of optimum intervals derived from the OF matrix

D-Factor vector - vector of D-Factor values for different A-Factor values (also repetition cases used in computing particular D-Factors)

3-D Graphs - 3-D graphs that visually illustrate the changes to OF, RF and Cases matrices

Approximations - twenty power approximation curves that show the decline of R-Factors along columns of the RF matrix. For each A-Factor, with increasing values of the repetition number, the value of R-Factor decreases (at least it theoretically it should decrease). Power regression is used to illustrate the degree of this decline that is best reflected by the decay constant called here D-Factor. By choosing the A-Factor tab at the bottom of the graph, you can view a corresponding R-Factor approximation curve. The horizontal axis represents the repetition number, while the vertical axis represents R-Factor. The value of D-Factor is shown at the top of the graph. The blue polyline shows R-Factors as derived from repetition data. The red curve shows the fixed-point power approximation of R-Factor (fixed-point approach is used because for the repetition number that is equal to two, R-Factor should equal A-Factor). The green curve shows the fixed-point power approximation of R-Factor taken from the OF matrix. This is equivalent to substituting the D-Factor obtained by fixed-point power approximation of R-Factors with D-Factor obtained from DF-AF linear regression.

Books and publications by the authors (in reference to SuperMemo):

Publications available on the Cross Country and Video English CD-ROMs are marked with (***). These cover most of the reference material relevant to the SuperMemo method. Publications available in this help file are provided with a hyperlink to the actual text. Publications available with SuperMemo shareware are marked with (*). These are available from shareware houses. SuperMemo World does not take part in distributing SuperMemo shareware!
For the remaining publications, see the appropriate source.

Important! Apart from commercial CD-ROMs, this help file, and materials available from SuperMemo Web site, SuperMemo World does not take part in disseminating reprints or soft copies of publications listed below.

Abramowicz, W., Wozniak , P.A., 1997, Enhancing hypertext documents by processing attributes and repetition spacing. *Systemy hipertekstu w ekonomii, Wydawnictwo Naukowe Akademii Ekonomicznej*, Poznan, Poland

Biedalak, K. 1991, (*) SuperMemo 5 User's Guide. SuperMemo World

Biedalak K., Wozniak, P.A, 1996, Build your Vocabulary with SuperMemo. *Spotlight*, Vol 10, p. 4

Gorzelañczyk, E.J., Wozniak, P.A. 1990, A model of the presynaptic terminal of the adrenergic synapse. *Proceedings of 10th International Biophysics Congress, Vancouver, Canada*, p. 404

Gorzelañczyk, E.J., Wozniak, P.A. 1991, Hypothetical model of processes involved in the release of norepinephrine in the adrenergic synapse. *BioSystems*, Vol. 25, pp. 275-281

Gorzelañczyk, E.J., Wozniak, P.A. 1994, A mathematical model of processes involved in long-term potentiation. *Abstracts of the 30-th Congress of Polish Biochemical Society,* Szczecin, Poland (in Polish)

Gorzelañczyk, E.J., Wozniak, P.A., Schneider Z., 1995, Looking for objects and parameters of the model of molecular processes underlying declarative memory. *Proceedings of Cybernetical Modeling of Biological Processes*, Cracow 1995 (in Polish)

Gorzelañczyk, E.J., Wozniak, P.A., Schneider Z., 1995, Theoretical basis of the SuperMemo algorithm. *Proceedings of the I-st International Congress on Education in Nursing*, Gniezno, 1995 (in Polish)

Gorzelañczyk, E.J., Wozniak, P.A., Schneider Z., 1995, A hypothetical model of molecular processes involved in formation of declarative memory. *Abstracts of the 31-th Congress of Polish Biochemical Society,* Warsaw, Poland (in Polish)

Gorzelañczyk, E.J., Wozniak, P.A., 1995, A molecular explanation of two independent components of long-term memory: hypothesis. *First Congress of the European Federation of Neurological Societies, Marseille, France*

Gorzelañczyk, E.J., Wozniak, P.A., Schneider Z., 1996, A proposition on the functions of objects in a synapse taking place in the process of forming long-term memory on the basis of experimentally determined variables of memory. *Abstracts of the 32-th Congress of Polish Biochemical Society,* Warsaw, Cracow (in Polish)

Gorzelañczyk, E.J., Wozniak, P.A., Schneider Z., 1996, Assisting declarative memory with the repetition spacing method. *Abstracts of the 27-th Congress of Polish Neurological Society,*

Warsaw, Katowice (in Polish)

Gorzelañczyk, E.J., Wozniak, P.A., 1996, A new approach to studying molecular memory based on understanding the optimum repetition spacing in learning. *Second Congress of the European Federation of Neurological Societies, Rome, Italy*

Wharton, Ch., Wozniak, P.A., 1994, SuperMemo. *CALICO Journal.* Vol. 12, No 1, p.66-70

Wozniak, P.A., 1990, (***) Optimization of learning. Masters Thesis. *Technical University of Poznañ*

Wozniak, P.A., 1991, (*) SuperMemo 6 for DOS. SuperMemo World

Wozniak, P.A., Biedalak, K., 1992, The SuperMemo method - optimization of learning. *Informatyka*, Vol. 10, p. 1 (in Polish)

Wozniak, P.A., Gorzelañczyk, E.J., 1992, Optimal scheduling of repetitions in paired-associate learning. *Acta Neurobiologiae Experimentalis*, Vol. 52, p. 189

Wozniak, P.A., 1993, A new method of fast learning and its software implementation. *Polish Technical Review*, Vol. 1, p. 8

Wozniak, P.A., 1993, SuperMemo as a new tool increasing productivity of a programmer. *Proceedings of the V-th International School of Polish Information Processing Society, Szczyrk 1993*

Wozniak, P.A., 1993, SuperMemo as a new tool increasing productivity of a programmer. *Informatyka,* Vol. 12, p. 1-11

Wozniak, P.A., 1994, Speed-learning with SuperMemo 7 for Windows. SuperMemo World

Wozniak, P.A., Gorzelañczyk, E.J., 1994, (***) Optimization of repetition spacing in the practice of learning. *Acta Neurobiologiae Experimentalis,* Vol. 54, p. 59-62

Wozniak, P.A., 1994, SuperMemo: Polish speed-learning technology. Multimedia release on CD-ROM. *Polish Technical Review*. 5-6, p. 30-31

Wozniak P.A, Gorzelañczyk, E.J., Murakowski J., 1995, (***) Two components of long-term memory. *Acta Neurobiologiae Experimentalis,* Vol. 55, p. 301-305

Wozniak, P.A., 1995, Selling and marketing Polish software applications. Case study: SuperMemo. *Proceedings of the V-th Forum of Information Technologies, Mragowo, 1995*

Wozniak, P.A., 1995, (***) Economics of learning. Doctoral Dissertation. *University of Economics in Wroclaw*

Wozniak, P.A., Abramowicz, W., 1997, The educational technology of repetition spacing in Information Systems Development, accepted for *Sixth International Conference on Information Systems Development, Boise, Idaho 1997*

see also Articles and reviews in popular press

Articles and reviews in popular press (in reference to SuperMemo):

Important! Apart from commercial CD-ROMs, this help file, and materials available from SuperMemo Web site, SuperMemo World does not take part in disseminating reprints or soft copies of publications listed below.

Baranek T., SuperMemo 7.0 - kdo chce, ten umi [SuperMemo - who wants, he can]. Cena/Vykon, Mar 1994, Czech Republic

Bergfors J., SuperMemo. Polskt program passar plugghast [Polish cramware]. *MikroDatorn*, Mar 1993, Sweden

Bergfors J. Plugga och lar, nu utan besvar [Learn fast and forget about forgetting]. *MikroDatorn*, Jun 1993, Sweden

Byte International, Speed-learning software, Dec 1993, p. IS-48, USA

Byte International, Learning Tool on CD, Jun 1995, p. IS-40 24, USA

Chabinski A., Repetitio est mater studiorum, SuperMemo version 7.2, *Chip*, Jul 1994, p.54-55, Poland

Chip, SuperMemo 6 a 7, Feb 1994, p. 173, Czech Republic

Chip, Nowe dziecko SuperMemo World [A new child of SuperMemo World], Dec 1995, Poland

Dramczyk B., SuperMemo version 2.0, Powtorka z powtarzania [Repetition on repetitions]. *Commodore&Amiga*, Sep 1994, p. 16-17, Poland

Dramczyk B., SuperMemo 3.0, jak skutecznie zapamietac [How to remember effectively]. *Bajtek*, May 1996, p. 17-19, Poland

Drazek T., Przedluzamy pamiec [Strengthening memory]. *PC World* Komputer, Jul-Aug 1994, p.95, Poland

Enter, Genius Piotra Wozniaka, Nov 1995, Poland

Grabowski J., Zapomniec o zapominaniu [Forget about forgetting]. *PC Kurier*, Aug 20, 1992, Poland

Gradisnik J., SuperMemo 7.2 - Repetitio mater studiorum est. *WIN.INI*, Vol 6, p.26-28, Slovenia

Gryciuk W., SuperMemo 7 for Windows, *Computerworld*, Sep 27, 1993, Poland

Gryciuk W., SuperMemo - zloty medal na Infosystemie [Gold medal for SuperMemo at Infosystem], *Computerworld,* May 30, 1994, p.6, Poland

Gryciuk W., SuperMemo. A remedy for the forgetful. *Warsaw Voice*, April 9, 1995, p. A5-A6, Poland.

Hausner M., Neco super ... [A little bit super ...], *PC World*, Jul 1994, p.45-46, Czech Republic.

Horodenski A., Cud niepamieci naukowo okielznany [Scientific conquest of forgetful memory]. *Computerworld*, Feb 17, 1992, Poland

Horodenski A., SuperMemo - trener szarych komorek [SuperMemo - a brain cell developer]. *Computerworld*, Jun 8, 1992, Poland

Humlicek P., Repetitio mater studiorum est, *Computer*, Jan 1995, p.4-5, Czech Republic

Jablonski W., Pierwsze kroki Geniusza [The first steps of Genius], *Bajtek*, March 1996, Poland

Korec P., Brana nejen jazykum otverena [A gate for languages is open]. *PC World*, Jul 1994, p.44-45, Czech Republic

Kowalski J., SuperMemo (1). Zapomnij o zapominaniu [Forget about forgetting]. *Enter*, Oct 1994, p.88-90, Poland

Kowalski J., SuperMemo (2). Zapomnij o zapominaniu [Forget about forgetting]. *Enter*, Nov 1994, Poland

Kuvaja S., Panttayspaketti kielten opiskeluun. *Mikrobitti,* 6-7, 1996, Finland

Liszewski P., SuperMemo CD-ROM. *PC World Komputer*, Apr 1995

MicroPC, Memory devil with the help of a micro, Jun 1994, Finland

Mlodzki J., Od pomyslu do przemyslu. Rozmowa z panem Piotrem Wozniakiem, tworca metody SuperMemo [From an idea to an industry. Interview with Piotr Wozniak, the author of SuperMemo]*, Bajtek*, Jul 1994, p.18-19, Poland

Pampuch M., SuperMemo. *Amiga*, Sep 1993, Poland

Partycka D., SuperMemo 6. *Swiat PC-tow*, Jul 1993, Poland

Pietras Br0mba M., SuperMemo czyli ucz sie szybko i zapomnij o zapominaniu [Learn fast and forget about forgetting]. *Bajtek*, Jan 1993, Poland

Pietras Br0mba M., SuperMemo 7.2PL dla Windows, *Bajtek*, Sep 1994, p.16, Poland

PC Kurier, Genius 1.0 - A new product from SuperMemo World, Aug 17, 1995, Poland

PC Magazine Po Polsku, Genius - A new product from SuperMemo World, Sep 1995, Poland

Plociennik P., Szybka nauka [Fast learning]. *Gazeta Wyborcza*, Oct 18, 1993, Poland

Reisch E., Das Klassenziel Erreicht, SuperMemo fur Windows [Perfectness reached]. *Chip*, Feb 1995, p.82-84, Germany

Rekunen V., SuperMemo Europack Pro, *Tietokone*, *Maaliskuu* 1995, p. 79, Finland

Spruit K., SuperMemo for Windows. *Computer Totaal*, Mar 1995, p.119, Holland

Szewczyk T., Genius 1.0, *Computer World*, Sep 11, 1995, Poland

Tietokone, SuperMemo Europack Pro, Toukokuu 1995, p. 67, Finland

Topol P., Kilka slow o SuperMemo [A few words about SuperMemo]. *Edukacja Medialna*, Vol. 2, Dec 1996, p. 56-58, Poland

Wimmer P., Pochwala SuperMemo [Praising SuperMemo]. *Enter*, Oct 1992, Poland

Wimmer P., Super SuperMemo. *PC Kurier*, Sep 30, 1993, Poland

Wimmer P., SuperMemo kontra Sclerosis Mentis. *PC Kurier*, Dec 8, 1994, Poland

Wimmer P., Genius loci. *PC Kurier*, Nov 7, 1996, Poland

Zwierzchowski Z., Nastepca nazywa sie Genius [The heir is called Genius], *Rzeczpospolita*, Aug 24, 1995, Poland

Zwierzchowski Z., Slownictwo wyksztalconego Amerykanina [The vocabulary of an educated American], *Rzeczpospolita*, *Magazine*, Oct 1996, p. 32, Poland

see also Books and publications by the authors

60 most frequently asked questions and answers about SuperMemo

On all possible occasions we apologize for not being able to answer all individual questions about the SuperMemo method and software. However, all inquiries are scanned carefully, and, if general enough, find their place in the set presented below.

Introduction to SuperMemo

Differences between versions of SuperMemo

Using the SuperMemo method

Using SuperMemo

Theoretical aspects of SuperMemo: length of intervals

Theoretical aspects of SuperMemo: speed of learning

Theoretical aspects of SuperMemo: advanced questions

Other questions

Q: What should I do if SuperMemo displays strange error messages, aborts frequently, and **File : Recover** does not work?

A: Make sure that your Windows is stable and you have no hardware problems. If the messages are displayed by Windows rather than SuperMemo itself (e.g. modules such as Kernel, User, Ctl3D, etc.), the problem is probably not with SuperMemo. Some hardware problems are a frequent reason for such situations, esp. if similar events happen with other application. Check your memory, hard disk, system setup and the mainboard in the first order.

In case you are sure **File : Recover** is unable to recover your system for reasons related to SuperMemo, contact SuperMemo World at software@supermemo.com.

Q: I have heard that baroque music is the best for improving memory; particularly Bach and Vivaldi. Does it also work in association with SuperMemo?

A: The claim that any particular kind of music is best for memory is unlikely to be true. Indeed, music can have powerful impact on our emotions and, consequently, on remembering. However, a study that measures impact of a particular kind of music on recall in a group of people can be compared to trying to find the optimum size of a shoe for an average citizen. Depending on the musical education of an individual, the same kind of music may bring a variety of emotion from relaxation, through agitation to aggression. While Four Seasons may have a positive impact on the mind of majority of the population, the best bet would be that everyone should stick with the music he or she likes. Be it punk or funk. The truth is that all that is good for the psyche and health is good for remembering.

Q: How to complete missing translations reported by **File : Recover**?

A: If at recovery you get messages like *Missing translation for Member #186:* "*virtually every*" you can fill out the translation using the following procedure:

1. choose **Search : Text**
2. type in the name that follows the number of missing translation member (e.g. *virtually every*)
3. click the translation button on the registry toolbar (the one with seven stars)
4. type your translation on the right side of the window
5. press *Esc* (to quit the translation window)
6. press *Enter* (to show the list of elements that use the translated item)
7. press *Enter* again to go to the first element on the list (if there are no elements on the list, the item is not in use)
8. in the element window, check if the translation is in the right place (e.g. *Shift+click* the component in question)

Q: Can a computer-illiterate person use SuperMemo?
A: Yes. The entire knowledge needed to start is encompassed in a page long Minimum User's Guide. 5 simple operations needed to use SuperMemo can be learned in less than 10 minutes!

Q: Can I write my own SuperMemo knowledge system concerning subjects of my interest?
A: Yes.

Q: If I prepare a knowledge system for SuperMemo, can I sell it to SuperMemo World?
A: Yes. Your knowledge system may be included with one of SuperMemo Worlds commercial CD-ROMs. Please contact SuperMemo World for details.

Q: Can I buy SuperMemo COD from SuperMemo World?
A: Only in Poland, yes.

Q: After buying SuperMemo, what sort of discounts in buying upgrades can I expect?
A: SuperMemo World's policy is to provide as large upgrade discounts as it is only economically viable.

Q: If I don't have a computer, can I use the SuperMemo method?
A: Yes. A limited, paper-and-pencil variant of SuperMemo exists, and is shortly described in the pamphlet distributed by SuperMemo World (if included with your SuperMemo CD-ROM, see the document file SMPAP.DOC in DOCS subdirectory).

Q: Can I use SuperMemo in a computer lab?

A: Yes. However, if you are interested in using the SuperMemo method, the condition is that the access to computers in the laboratory be regular. Once you start working with SuperMemo, you must continue the process of repetitions. Otherwise, the acquired knowledge will gradually be forgotten (e.g. if you stop after having used SuperMemo for 5 years, you will forget 60% of the learned information in the first year!).

Q: What are the basic differences between SuperMemo 2, 5, 6, 7 and 8?

A: In short, all SuperMemos allow you to substantially increase the speed of learning. Figuratively, if traditional methods of learning are viewed as a pedestrian, SuperMemo 2 may be compared to a bicycle, and SuperMemo 7 to a car. SuperMemo 8 would stand here as a Mercedes. SuperMemos 2, 5 and 6 where written for DOS, while SuperMemo 7 is a Windows application. Last but not least, only SuperMemo 8 is hypermedia enabled. For more details on the development of SuperMemo, see Short history of SuperMemo.

Q: What is the size of SuperMemo 2, 5, 6, 7 and 8 in KB?
A: The sizes are as follows (not compressed):

SM2 77 KB + 48 KB of help
SM5 108 KB + 71 KB of help
SM6 156 KB + 154 KB of help
SM7 580 KB + 200 KB of help + 150 KB of DLLs
SM8 is 1300 KB + 150 KB of help

Note, that the size of the program cannot be used as a measure of its quality. A larger program may be larger because it has more functionality or simply because it was written in a wrong way.

Q: What is the size of SuperMemo together with a knowledge system?
A: Knowledge system may be as small as 100 KB or as big as 650 MB (or more). The size depends mostly on the number of multimedia objects included (esp. AVI files). For example, Cross Country and Video English nearly fill the capacity of a CD-ROM but less than 4 MB of it has to be installed on your hard disk.

Q: How to memorize an item after memorizing all other items?
A: Choose Reset on the element pop-up menu (*Ctrl*+*R*).

Q: I left for vacation, and my learning process is in mess. What should I do?
A: Use **Tools:Mercy**

Q: My computer was hit by a virus. SuperMemo reports integrity errors. What should I do?
A: Run **File : Recover**

Q: In your materials I found a contradiction. On one hand you claim that once learned knowledge is constantly maintained in the student's memory, on the other you say that after ceasing repetitions, I will gradually forget what I have learnt. Which is true?

A: Both facts are true. The term *maintained* is understood as *kept in memory by means of repetitions*, not as *remains in memory for ever*.

Q: In your help file you often mention that the user should periodically back up his/her knowledge systems. What does it mean and how to do it?

A: To back up a file means to copy it to an archive location in case the original file were to be lost or damaged. You can back up your files by means file management tools (e.g. Windows Explorer) or by using **File : Copy**. In case your system is very large, you may wish to back-up everything apart from multimedia files. For that purpose use **File : Semi-Copy**. It is recommended that you also back-up your knowledge systems on media other than your hard drive. If you systems are too large to go into a diskette, you might use any of tried-and-true archiving methods: CDR, magnetooptical disks, etc.

Q: Can SuperMemo work with other file formats, e.g., ASCII, DBase, Lotus, Word Perfect, Quattro Pro, ChiWriter, etc.?

A: No. However, if the files contain data in the form of questions and answers (e.g. word pairs), they can easily be converted to simple SuperMemo knowledge systems by means of **File : Import : Q&A Text**

Q: When I back up my systems, do I have to back up SM8.EXE as well?
A: No.

Q: How can Repetitions indicate 7 while Cases 31. Which is true?

A: Repetitions (left panel of Item data) says how many times a given item has been repeated. Cases (right panel of Item data after a repetition) indicates how many repetition cases, of various items, have been used to compute entries of OF and RF matrices. In other words, Repetitions concerns a single item, while Cases concerns many items with the same A-Factor and Repetitions values.

Q: What should I use **Duplicate** for? What for do I need the same item in the same knowledge system?

A: You can duplicate an item, if you want to add to the another item which is only slightly different. This way you can spare some time by reediting the old item instead of typing in the new one.

Q: I think that lower grades, e.g. 3, should produce shorter intervals in comparison to higher grades, e.g. 5. It is not always so in the SuperMemo method.

A: In SuperMemo, lower grades may produce longer intervals because of the two following reasons:

* Grade 3 may result in the enhancement of the so-called spacing effect, which may be less visible for Grade 5. The spacing effect says that longer intervals, and consequently greater recall efforts, produce more stable memory engrams. SuperMemo does not arbitrarily set the function of optimal intervals. It computes intervals which are most likely to result in the forgetting index defined in **Tools : Options : Learning : Forgetting index**; hence; the possibility of longer intervals for lower grades.
* all intervals are always slightly dispersed around the optimal value for the sake of more accurate plotting of forgetting curves

Q: I used SuperMemo 2 shareware, and was accustomed to repeating forgotten items on the next day. It is very irritating that in SuperMemo 8 I do not have this possibility.

A: SuperMemo will schedule forgotten items in intervals which are determined by the desired forgetting index. The greatest increase in the speed of learning in newer versions of SuperMemo as compared with SuperMemo 2 resulted from substantially increasing the length of the first interval! The student may be left with the feeling that he is likely to forget the item again if it is not repeated on the next day. Statistically, however, he will forget no more than the proportion defined by the forgetting index (specified in **Tools : Options : Learning : Forgetting index**). By reducing the forgetting index to less than 5%, the length of the first interval is likely to drop to 1-2 days in most cases. Moreover, if you are particular about repeating a given item on the next day, you can choose *Ctrl+M* to commit or recommit an item with a selected first interval.

Q: Why is the first interval after which the first repetition takes place not equal in all cases?
A: It is randomly modified to speed up computing its optimal value. Additionally, random dispersion of intervals around the optimum value prevents repetitions from being packed on a given day, while neighboring days have lots of room to accommodate new items.

Q: I have been using your program for one week. In my estimation, I can learn no more than 50-100% faster than without the program. Do you claim that my estimation is conservative?

A: No. The power of SuperMemo can only fully be observed after a longer period of time. The estimation of the 50-fold increase in the speed of learning concerns lifelong perspective. By using mnemonic techniques, you can easily beat SuperMemo in periods of up to one month. In other words, SuperMemo will not work for those you want to cram a great deal of data before an exam. On the other hand, it is of invaluable help for those who want to retain indispensable facts and figures for months and years.

Q: Why can I not see the correlation between intervals and the grades given in learning?

A: Your impression of no correlation between grades and intervals is quite common among those who begin their work with SuperMemo. It results from the fact that at memorizing new items, first intervals are randomly dispersed around a fixed value. This value comes from the model of an average student, and can be modified only after the repetitions have shed some light on if the value should be increased, decreased or kept at the same level. Consequently, it will often happen that a lower grade will produce a longer interval and vice versa. At memorizing new items, grades cannot be used to estimate item difficulty because the program has no way of knowing if good grades come from easiness of items or from the fact that a given group of items has just been input to the knowledge system (inputting items is a form of repetition). You will start noticing the correlation between grades and intervals in a week or two.

Q: (1) I want to spend my money on the MegaMemory program delivered on cassette tapes. How would you convince me, that I should rather spend money on buying SuperMemo (2) Is SuperMemo the same as SuperLearning developed by Dr Georgi Lozanov from Bulgaria? (3) Which techniques produce a greater increase in the speed of learning: SuperMemo or the mind maps proposed by Tony Buzan?

A: The mnemonic techniques (Tony Buzan, MegaMemory, etc.), SuperLearning (Dr Georgi Lozanov) and SuperMemo are all complementary, and make up three independent pillars of effective learning. All of them have to do with optimization:

* In SuperLearning the stress is on optimizing the cognitive environment which produces the best effects on the learning process.
* In mnemonic techniques, the representation of knowledge is optimized in order to produce the most durable memory engrams.
* In SuperMemo, the optimization concerns the spacing of repetitions.

To be a successful student, you have to apply mnemonic techniques while using SuperMemo in a optimum cognitive environment. The important advantage of SuperMemo is that many capable students develop quite good techniques as far as mnemonics is concerned. They can also reasonably master their own physiology to subject it to the learning effort. However, they have no way of optimizing the spacing of repetitions. Mostly due to the fact that the amount and complexity of computation needed to determine the spacing require the use of a computer, and the technology proprietary to SuperMemo World. You are advised to follow the teachings of Tony Buzan, Kevin Trudeau or Dr Georgi Lozanov; however, if you are a capable student, you can safely commit your foremost priority to learning how to optimize your learning schedule. As for now, SuperMemo is the world's best tool developed to assist you in this task.

Q: How do you respond to the accusation that the whole SuperMemo theory is contradicted by the claim stated in the MegaMemory program of the American Memory Institute, which says that forming indelible memories is possible if suitable representation of the learned knowledge is used?

A: AMI is a commercial, not scientific institution. The claim that their program produces indelible memories can only be understood as part of the marketing strategy used to promote their otherwise very interesting program. The concept of permastore has been proposed in a limited number of publications in the field of psychology; however, the data collected in the research on molecular aspects of memory clearly refutes such as possibility. In SuperMemo, items whose optimal intervals reach beyond the biologically determined lifespan can be considered part of the permanent memory store. The easiest way to disprove the AMI's claim is to use their program to memorize a large body of intractable knowledge with and without help of SuperMemo (e.g. 1000 phone numbers would do). The knowledge retention after a 6-month-long period will stand at 85-98% with SuperMemo (depending on the forgetting index), and 5-15% without it (depending on individual capabilities)!

Q: Do you not think that the minimum information principle stands in conflict with the ages old rule that the learned knowledge should be highly associative in nature?

A: No. The minimum information principles concerns the representation of knowledge in SuperMemo databases, not in the student's memory, and it does not prevent great advantages coming from proper structuring of the learned material. In the optimum situation, the student should first construct a cohesive model of the learned subject, and only then, apply SuperMemo to make sure that the learned knowledge is sustained in memory as a whole. The knowledge may be highly associative, but strictly targeted neural stimulation, achieved by means of granular representation of knowledge in SuperMemo, is necessary to effectively induce molecular processes responsible for memory formation. Indeed SuperMemo has been conceived in such a was so as to make it easier to formulate knowledge in a structured way (topics) and later learn it in a way typical for SuperMemo (items).

Q: On what basis do you ground your claim that SuperMemo increases the speed of learning from 10-50 times?

A: For knowledge retention of 95%, it can be computed that the number of repetitions in an average learning lifetime (i.e. about 55 years) is 50 times greater for equally spaced repetitions than for progressive repetitions (as used in SuperMemo). For repetitions with no regular spacing scheme, this number may even be greater. Moreover, the greater the required knowledge retention, the greater the increase in the knowledge acquisition rate (classical forms of learning almost never reach knowledge retention above 10%!!!). In practise, users of SuperMemo claim that it increases their speed of learning from 50% to 2000%. These values are, however, highly subjective, as they do not account for so-called intractable items, which are practically unmemorizable without SuperMemo. In other words, students tend to underestimate the fact that they reach knowledge retention from 90-99%, which would hardly be achievable using any other method.

Q: Can the SuperMemo method be used to forget things?

A: Forgetting is a molecular process that cannot easily be induced by natural methods. The more so, there are no sensitive methods to induce selective forgetting, though lesion to some parts of the cerebral cortex may produce roughly localized amnesia. However, there is a component of forgetting that may be influenced. This component is interference. Whenever we learn new things, they always interfere with previously learned material. The interference may enhance some of memories while obliterating others. This fact can be used to employ SuperMemo in forgetting, by formulating and memorizing a large number of contradictory items that strongly interfere with remembered facts that are to be forgotten. However, you should not expect the effectiveness of such a procedure to be anything but discouraging.

Q: Are SuperMemo 8 knowledge systems compatible with SuperMemo 7?

A: You can easily import SuperMemo 7 databases to SuperMemo 8. Some information about the learning process will be lost on the way because of different learning algorithms used by those two programs. Nevertheless, it shall not visibly influence the learning process. Import from SuperMemo 7, makes SuperMemo 8 open to thousands of databases developed for earlier versions of SuperMemo starting with SuperMemo 2 Public Domain software. Note, however, that upgrade programs distributed along SuperMemo 7 are needed to convert databases from formats earlier than SuperMemo 6 (UP_2_5.EXE and UP_5_6.EXE). Additionally, import of **Q&A text** files and **Audiovisual files** import, makes it possible to adapt SuperMemo databases developed for Mac and Amiga for use with SuperMemo 8.

Q: Why can I not see the effect of different grades during memorizing new items, and during the final drill? Are grades used there, and how?

A: At memorizing and at final drill, the interpretation of grades is limited to Pass (4 or more) and Fail (less than 4). None of the optimization parameters are affected. Here are the reasons:

* at memorizing new items, grades cannot be used to determine item difficulty, because the program has no way of knowing if a good grade resulted from easiness of the item or from the fact that the user has just input the given item to the knowledge system. Consequently, grades cannot affect optimization, and are only used to detect items that should be scheduled for final drill.
* Final drill is not used for optimization either. This is because of the fact that final drill relies on short-term memory, and is substantially more sensitive to inter-item interference as well as more dependent on the number of other items separating repetitions of the item in question. Final drill is used solely for the purpose of fixing the forgotten information in the user's brain. Grades are used only to eliminate from the final drill those items which score 4 or more.

Q: I have created my own question and answer files in ASCII. I would like to convert them into SuperMemo knowledge systems. What is the easiest way?

A: Manually or programmatically, convert the files to the TXT format acceptable by **File : Import : Q&A Text**. All question lines should start with Q: and all answer lines should start with A: . There should be no more than five lines, and the total length of text should not exceed 250 characters per element. For example:

Q: What is the capital of Zambia?
A: Lusaka

Q: What is the capital of Zaire?
A: Kinshasa

and so on.

Q: What sound boards is SuperMemo compatible with?
A: All Windows compatible sound boards.

Q: Can I use SuperMemo to improve intelligence?

A: Yes. Let us define intelligence as the capability to process information. Your brain is like a computer: the better software it runs, the better it performs. You can identify the elements of knowledge that, for example, will make you perform better in intelligence tests. Those elements, if representable in audiovisual or textual form, can be used to learn, and consequently, improve your intelligence. With SuperMemo programmability, you can even go beyond that and use all possible learning procedures that do not require specialized hardware. Thus, for example, by implementing a DLL for solving mathematical equations, you can practise problem solving tasks, and with a DLL for executing touch typing repetitions, you can even go beyond declarative learning, and practise procedural or motor tasks.

Q: Why are intervals in SuperMemo so long? In SuperMemo 2 they used to be much shorter.
A: They are as long as it takes to produce the desired proportion of items to be forgotten at repetitions. If intervals are irritatingly long, reduce the value of default forgetting index (**Options : Learning : Forgetting index**), or set a lower value of forgetting index for items belonging to a given hierarchy node in the Contents window.

Q: Why are some of the items repeated in intervals that increase by just one day between repetitions? I am certain that those intervals could be longer.

A: The intervals are as long as it takes to produce the desired proportion of items to be forgotten at repetitions. If repetitions are irritatingly frequent, increase the value of default forgetting index (**Tools : Options : Learning : Forgetting index**), or set a higher value of forgetting index for items belonging to a given hierarchy node in the Contents window.

Q: Why does SuperMemo ask only those items which are difficult?
A: That is exactly what SuperMemo is supposed to do. You do not want to learn things you already know, do you? The purpose of SuperMemo optimization is to compute the best intervals between repetitions. Difficult items must be repeated more often; hence the impression that only the hardest items are considered in the process.

Q: Why does not SuperMemo mix the order of items scheduled for repetitions? I have introduced some words in an alphabetical order and they are asked in the same order at repetitions. This is not optimum.

A: True. Please note, however, that items will be mixed as soon as their A-factors and intervals become different. The major advantage of not mixing items is the fact that at repetitions, short interval items are asked first. The student has a general feeling of how long intervals are assigned to items that are currently being asked. This makes it possible to sharpen the accuracy of grading for long-interval items whose inaccurate placement in the repetition schedule has more severe consequences for the learning process.

Q: When an infant burns its fingers, it is not likely to forget the experience. How does it relate to your claim that nothing learned once can be remembered for ever without repetition?

A: Some low-level memories may differ in nature from memories typically involved in high-level learning. This results from the existence of specialized nervous circuitry involved in storing inborn memories, which may combine with newly acquired experience. This way, a number of reflexes does not have to be ever learned, and remain imprinted in memory for lifetime. However, in the above case, one should not overlook the fact that repetition of an unpleasant experience such as burning one's fingers, does not have to be based on repeating the experience itself. The mere fact of visual contact with flames, may invoke unpleasant memories and serve as a repetition. This way, each subsequent contact with fire, hot plate, etc. will effectively function as a repetition, without a need to experience the pain.

Q: Tony Buzan claims that the first review (second repetition) of an item should take place in 24 hours. This is not so in SuperMemo. Why?

A: In SuperMemo, the length of the first interval is computed from the forgetting curve plotted in the course of repetitions to make sure that a defined proportion of items is remembered; usu. 80-97% (this proportion is programmed by means of setting the forgetting index). Depending on the forgetting index, the length of the first interval may range from 1 up to 20 days, and is not set arbitrarily. It is computed from the record of repetitions and determined by the desired forgetting index (requested forgetting index is the proportion of items that are not remembered at repetitions). While Buzans recommendation is valid in many cases, you should not forget that SuperMemo computes intervals with a high degree of accuracy that cannot be obtained by other means.

Q: Tony Buzan claims that 75% of information is lost if not reviewed in 24 hours. Does it not defeat the validity of SuperMemo in which the first interval is often longer than a week?

A: No. Buzan's claim does not seem valid in the light of results obtained with SuperMemo. In SuperMemo, if the student chooses the retention of 95%, the typical value of the first interval falls in the range 2-6 days depending on the students and the difficulty of the learned material. For retention 25%, the same interval might be as long as one month, though it cannot be verified experimentally with SuperMemo which limits the range of the forgetting index from 3-20%, which implies the overall retention in the range of 85-99%.

Q: Do you have SuperMemo for platforms other than Windows?

A: Currently, SuperMemo is available for DOS, Windows, Mac and Amiga. Versions for Psion and OS/2 are under development. Versions for Atari ST and Atari XL have been discontinued due to lack of interest among customers and are no longer available (please do not inquire).

To learn more about the Amiga version contact Twin Spark Soft, Os.Kolorowe 9/16, 31-939 Krakow, Poland, tel: (48) 12 444368, fax: (48) 12 477299.

To learn more about the Mac version contact AS Projekt, ul.Warynskiego 10/78, 85-320 Bydgoszcz, Poland, tel/fax: (48) 52 392759, e-mail: SOS@TELBANK.PL.

Q: Why is time updated only at repetition time?
A: The timer used in SuperMemo is used to estimate some parameters related to the learning process (Mean Time, Workload, etc.). Obviously, the time which the student spends on editing the knowledge system, or even eating his/her lunch, is of little relevance to the method per se. The time spent for the entire learning session, if at all needed, can easily be computed by using an external timer.

Q: I would like to keep the definition of backgammon in my SuperMemo database, but I cannot cut the Webster wording to suit the 250 character limit, and yet make the definition understandable: "a board game played with dice and counters in which each player tries to move his counters along the board and at the same time to block or capture his opponent's counters ..."

A: If you do not know what backgammon is, you should not attempt to store the rules of the game in just one definition! See *Principles of effective work with SuperMemo. Simplicity* to get some clues on how to formulate items. You might choose one or two features that define backgammon unambiguously, and keep the rest of details in separate items, for example: "What do players try to do with the opponent's counters in backgammon?", "What is the size of the board used in backgammon?", "How many counters does a player have in backgammon?", etc. If you know what backgammon is, and you just want to make sure that you do not forget the name, you might just ask "What is the name of the game I played with Robert in summer 1996?". However, if you are just a student of languages, it may appear sufficient to learn that backgammon is a kind of board game. In Advanced English you will find the following item:

Q: backgammon
A: kind of game.

Q: Your claim that SuperMemo allows one to learn with the speed that closely approaches the maximum speed with which memories can be formed does not sound convincing. What evidence do you have to substantiate this claim?

A: For a given collection of questions and answers SuperMemo computes a repetition schedule that cannot be much improved by algorithmic means. In this sense, it gives you almost the best possible repetition timing. Obviously, there is a great number of elements that can improve the learning process independently of SuperMemo. These are: (1) application of mnemonic techniques (the same item may be memorized in a number of ways, each of them providing different stability of memory traces), (2) repetition procedure and environment, (3) health and the status of the student's mind, (4) pharmacological factors, etc. Moreover, the same knowledge may be represented in a different way (different order of items, different wording, etc.) which may greatly affect the speed of learning. As far as SuperMemo optimization algorithms are concerned, their accuracy cannot be greatly improved because of the environmental noise that interferes with the learning process. Each item may be more or less consciously repeated in common life situations; the process being beyond the control of SuperMemo.

Q: What is SuperMemo World's e-mail?

A: **info@SuperMemo.com**

All e-mail messages are read carefully. Please understand, however, that we are unable to answer all e-mail inquiries.

Q: I neglected my repetitions for too long. I want to start learning again with the same knowledge system. What should I do?
A: Use **File : Reset**. Remember to set the correct date in your PC!

Q: When I use **Edit : Append new items**, how can I get to the previously input item?
A: Chose the button **<** on the element window navigation toolbar.

Q: In Advanced English, you use questions like *instance of reducing*, *instance of mocking*, etc. Do you not think that it directly suggests the answer (*reduction, mockery*, etc.)? Is that OK?

A: This is in accordance with the minimum information principle. The question assumes that the words *reduce* and *mock* are understood; therefore *reduction* and *mockery* are also understood. The only problem is the exact wording of the noun form. After all, in common situations, the simple forms of the verb or adjective are also known, and the problem is with forming the noun (e.g. *essential -> essentiality*, *proportional -> proportionality*, etc.)

Q: I had a break in learning, and now the retention in my knowledge system rapidly decreases. To avoid the drop, I made a copy of the knowledge system, reset it, and did repetitions in both the old and the reset knowledge system. Do you advise any better method?

A: Your action is an excellent example of what **not to do** with SuperMemo! A break in learning will always result in a drop in retention, and the only cure is to continue with repetitions as regularly as possible. By doing repetitions in two identical knowledge systems you will completely fool SuperMemo and your process may turn into mess. Delete the new knowledge system immediately, and work regularly with the old one! The only support for breaks in learning comes with **Tools : Mercy**. Remember, however, that it should be used with moderation.

Q: Why does the forgetting curve span only 20-days?

A: The 20-day limit pertains only to the first repetition. In most cases, the first interval does not reach beyond 20 days. When inspecting the curve, press the down-arrow key, and the curve for the second repetition will be presented. All graphs for repetition number greater than 1 use U-factor as the measure of time. If, for example, the first interval was 10 days, then the U-factor limit of 2 for the second repetition, corresponds to 2*10 days, i.e. 20 days (see the definition of the U-factor). Consequently, the graph will span the period of 20 days. For higher repetition numbers, U-factors and forgetting curve graphs will correspond with longer inter-repetition intervals.

Q: Why is the first interval not always of the same length?
A: Intervals are always slightly dispersed around the optimal value because of two reasons: (1) diversified retention measurements at repetitions increase the accuracy of forgetting curve sketching and accelerate the convergence of optimization parameters, (2) dispersion helps counteract workload lumpiness (large variations in the numbers of repetitions scheduled for particular days).

Q: Why do not questions in Advanced English suggest that there are two or three correct answers separated by a slash?

A: Because the purpose is to be able to recall at least one of these answers, not all of them. In real life situations, an idea springs up to your mind, and you must quickly put it to words. You do not need to remember all the relevant synonyms at the moment, though knowing and using them might add to the beauty of your statement. Trying to recall all the synonyms would greatly slow down your work with the knowledge system, though there is no counterindication against your proposed approach.

Q: I think that a parameter that would limit a number of repetitions per day would be of great help. Will you consider implementing it?

A: No. Consider a situation that the limit is 100, and you have already 100 items per day filled up for the next 365 days. Adding new items would be equivalent to running **Tools : Mercy** again and again; and **Mercy**, as you remember, if it is to be done optimally, takes quite a time to execute. Even if you conceive smart **Mercy** that circumvents the collect-sort-reschedule paradigm, and keeps schedule information in a quickly accessible format, each repetition at **Learn : New material** would involve an operation of rescheduling about 300 items (unless you want the first interval to be one year long).

Q: I have an exam for a driver's licence in 2 weeks. How can I best memorize the Traffic Regulations knowledge system? How can I increase the frequency of repetitions?

A: Although SuperMemo is not a cramming tool, and it would be much safer to start 2-3 months before the exam, the following shall work pretty well: (1) Set **Tools : Options : Learning : Forgetting index** at 3%. (2) When memorizing difficult items, choose *Ctrl+M* and provide the first interval value equal to one day. (3) Memorize the knowledge system in equal portions in the period spanning from today to 2-4 days before the exam. Use **Tools : Random review** intensely over the last 2-4 days.

Q: How can I undo a wrong grade?
A: Choose **Cancel grade** instead of **Next repetition**

Q: I am writing my Master's Thesis in mathematics on algorithms used in SuperMemo. However, me and my supervisor cannot figure out why does not SuperMemo allow intervals to decrease, even if it might seem optimal?

A: An item which cannot sustain increasing intervals in repetitions must be considered ill-defined and reformulated, or even removed from the learning process altogether. One of the earlier versions of SuperMemo adopted the decreasing interval approach with disastrous effects. The solution was promptly discarded. Decreasing intervals is like letting items linger indefinitely in a semi-forgotten state. The present solution puts all items to a true test of well-structuredness, and those items that cannot pass it are forgotten again and again. Forgetting occurs up to the point when the impatient student looks for a remedy to the problem (restructuring, mnemonic techniques or ... deleting the item).

Q: Why do you not differentiate the value of the forgetting index for items with different repetition numbers? This would make it possible to increase the retention for items with long intervals at the cost of short interval items. The overall retention could remain the same.

A: The proposition is very interesting. There are, however, some reasons against the proposed solution:

* SuperMemo would not live up to the promise of assuring the selected level of knowledge retention. Especially at the beginning of the learning process, retention might be much lower than desired.
* For high repetition numbers, optimal factors tend to be very low. Reducing the forgetting index might result in annoyingly small increase of the inter-repetition interval, and the spacing effect might reap its toll.
* Increased forgetting index for low repetition numbers might make it difficult for a large number of items to lock in in the process. This would result in the excessive number of lapses, and an overall drop in the average A-factor.

Q: Where can I get old shareware versions of SuperMemo 1 through 4?
A: You can get SuperMemo 2 from shareware houses, BBS'es, CD-ROMs, etc. throughout the world. For example, it is available from Simtel as well as from Simtel MS-DOS Archive on CD-ROM (directory EDUCATIN, file SM2.ZIP). SuperMemo 1, 3, and 4 have never been released to the public in any form.

Q: How can I get SuperMemo 6 Public Domain?

A: **(1)** The simplest way is as follows:

* on World Wide Web go to **www.shareware.com**
* type in *SuperMemo* in the Quick Search box
* click SUPRMEM6.ZIP (later you can also choose SM6TOOLS.ZIP)
* choose from one of 30 locations round the world from which you can download SUPERMEMO 6 (choose the closest and most reliable link; link ratings are visible as small squares to the left of the site)

(2) You can also find SuperMemo 6 in Simtel MS-DOS Archive on CD-ROM (directory EDUCATIN, file SUPERMEM.ZIP).

(3) You can ftp SuperMemo 6 from OAK.Oakland.Edu (log-on as anonymous). SuperMemo is in the file /pub/msdos/education/supermem.zip. Remember to set ftp to binary mode before downloading supermem.zip.

(4) In Poland, you can order it by mail from Zbych Shareware Company or Bajtek's PC Shareware Disk 19 (tel. 02 6211205).

Q: I want to learn the English word *than*. My item is illustrated with an example: *he is higher than you*. If I remember *than*, but I forget the example, I respond with grade Good(4), and write down the example on paper. Later I drill the examples on paper. Can I simplify the process in SuperMemo?

A: Yes. You can keep the example with *than*, but it should not affect the grade, and consequently, drilling. If you want to remember the example, you must keep it in a separate item (use *Ctrl*+D or **Duplicate** on element pop-up menu). Delete *than* from the duplicate, and choose **Commit** or *Ctrl+M*. Text-book application of SuperMemo is certainly paper-free. Follow the rules, and paper shall not be needed.

Q: How can I drill all memorized items before an exam?
A: Use **Tools : Random review**

Q: How can I review difficult items in my knowledge system?
A: Use **View : Leeches**

Q: How can I make my repetitions ahead of time?
A: Use **Tools : Mercy** with **Consider future repetitions** check-box checked. Set **Gathering period** up to the last date you want to collect repetitions from

Q: Why are sounds in Advanced English once associated with questions and once with answers?

A: Sounds are, if possible, linked to the subject of learning, which may be an active recall of the answer, or passive recognition of the question. If you do not like a particular association, you can either delete it, or replace with another one (use **Link registry member** on the sound component pop-up menu).

Q: How can I review items scheduled for final drill?
A: Choose **View : Drill**.

Q: How can I easily review items that I have already been memorized?
A: Choose **View : Memorized**

Q: Is it possible to review all particularly difficult items?
A: Yes. Use **View : Leeches** and set the maximum A-factor to the desired value (e.g. 2.0)

Q: How can I list items with short intervals?
A: Choose **View : Filter** and specify the maximum length of the interval

Q: What is the meaning of vertical and horizontal axes in **Tools : Analysis : Curves**?
A: Vertical axis corresponds to retention expressed in percent. Horizontal axis corresponds to time expressed as U-factor.

Q: Why doesn't Spell-Pad provide grades from Null to Bright; just pass or fail?
A: It is all-or-none comparison. For more sophisticated comparisons, you must input grades manually.

Q: Can I release my own knowledge system on CD-ROM or I need to obtain a licence from SuperMemo World?

A: You can release your own title without a permission from SuperMemo World; however, you might consider adding SuperMemo 8 to your CD-ROM at the most affordable price. See Releasing ones own multimedia CD-ROM.

Commercial knowledge systems released with SuperMemo

The following commercial knowledge systems were available from SuperMemo World at the moment of compiling this help file (prices around $50 per CD-ROM; contact SuperMemo World for details):

* **Cross Country** - a vigorous course of American English for beginners and intermediate-level students. It comprises 2000 topics and exercises full of animation, video, games, songs, and more. Cross Country is provided with a database of the entire vocabulary used in the course. This database can be used to memorize the vocabulary by means of the SuperMemo method. Cross Country has been licensed from Phoenix ELT and Macmillan Publishers, and is based on an excellent 1994 book by Jane Revell.
* **Video English** - a course of British English for intermediate and advanced-level students. It comprises 1000 scenes and exercises entirely based on video sequences compiled by The British Council. Video English makes it possible to get in touch with life in London in its most natural setting.

Apart from being published by SuperMemo World, SuperMemo 8 has also been released as a supporting product by other CD-ROM publishers (e.g. as a speed-learning tool for Polish PWN Encyclopedia, as a knowledge system demonstration on Chip cover CD-ROMs, etc.).
All authors interested in releasing their own hypermedia titles with SuperMemo may contact SuperMemo World and will meet with full technical support.

If you would like to test SuperMemo before buying it, have a look at the silly, old, good SuperMemo 6 for DOS (freeware!).

See also How to release ones own multimedia CD-ROM

How to release ones own CD-ROM with SuperMemo

If you have long thought about releasing your own multimedia CD-ROM, SuperMemo provides you with a unique opportunity to do that at the most affordable price. The main reasons why you might want to choose SuperMemo over other authoring tools are:

* if you want to release an educational title, the success of your effort may greatly be enhanced by the immensely popular SuperMemo technology supported by SuperMemo. SuperMemo makes it possible to greatly increase the effectiveness of learning and has long been appreciated worldwide. SuperMemo is the only authoring tool that uses the SuperMemo method
* at the current licence fee for title releases of up to 1000 CDs, SuperMemo is probably the cheapest fully-fledged educational authoring tool available

Your algorithm for releasing your own title is very simple, and the most difficult part indeed is up to you. Here is what you should do to see your own CD-ROM title on the shelves of software stores:

1. Compile the material you want to release on your multimedia CD-ROM (i.e. pictures, sounds, texts, etc.)
2. Contact SuperMemo World in order to obtain a fully-featured version of SuperMemo 8 to start your work
3. Create a SuperMemo knowledge system based on your material (this step seems to require most effort and skill; however, it is entirely dependent on you, and it concerns the material you are probably very familiar with)
4. Pay the licence fee to SuperMemo World in order to be legally authorized to mass-produce SuperMemo CD-ROMs
5. Place your knowledge system in the place of a SuperMemo CD-ROM image of another SuperMemo title (this is equivalent to copying a single directory!)
6. Produce a CDR disk with your title. A CDR is a writable variety of compact disk. If you do not have a CDR recorder, many third parties provide recording a CDR for as little as $30.
7. Send your CDR disk to a compact disk manufacturer for replication. The cost of replication usually does not surpass $1 per single CD-ROM disk. Contact SuperMemo World if you would like to get help at this step.

At this point, your own CD-ROM title is ready for sales. SuperMemo World will gladly take part in distributing your title through its own channels.

Note for **software developers:** If your software has anything to do with information that may provide a learning material (encyclopedias, dictionaries, factual databases, etc.), and you would like to provide your software with an interface to SuperMemo, contact SuperMemo World.

SuperMemo: Forget about forgetting

J.Kowalski, November 1994, *Enter*, Poland
(translated and updated, September 1996)

Increase the speed of learning 50-fold! This is how SuperMemo World advertises its product: SuperMemo. Is it yet another case of product hype or SuperMemo is indeed a case worth serious consideration?

The author of SuperMemo, Piotr Wozniak, PhD, graduate of Adam Mickiewicz University (molecular biology) and Technical University of Poznañ (computer science), claims that there are very few secrets about why SuperMemo is so effective. When we met in November 1993, Wozniak tried to dispel all my doubts about SuperMemo. It really works, that's a fact, and nobody has to convince me about this. However, is it really the ultimate? Is it really science, or is it just a product born out of the promotional effort of the SuperMemo World's marketing team.

See also: Scientific conquest of the forgetful memory

Evolution and memory

(from Forget about forgetting)

The most interesting and common-sense evidence speaking for SuperMemo is the evolution. Wozniak's approach to SuperMemo is deeply evolutionary: "*It's enough to have some basic understanding of the mechanisms of evolution to understand why memory works the way it does, and why SuperMemo is the ultimate solution for fast learning*".

The nervous system has been the evolutionary invention that introduced central control in, otherwise, homeostatically controlled organisms. It was like introducing a global communist government in a conglomerate of countries and federations, i.e. cells and organs, governed by purely free market of enzymes, free-flowing metabolites and hormones. In its introduction of communism, the evolution was smarter than humans in this sense that the communist practices were introduced not outright, but in proportion to available means. The developing nervous system, in the successive stages of the evolution, took over an increasing control over the organism in tandem with the increasing complexity of its neural structure. The culmination of this process was the human brain. The ultimate creator of our civilization.

Not only for the successful introduction of neural communism, the evolution can be viewed as an incredibly smart designer which would not miss an opportunity for improvement. Though its mechanisms are dead slow and purely random, what could not be accomplished by guided progress has been accomplished by the power of 4 billion years that have passed since the appearance of the first living cell. The infallibility of the evolution in the range of what can be accomplished by living matter based on DNA and proteins can be a very useful guidance in understanding neurophysiology and human psychology; including the mechanisms of memory. Optimization of the mechanisms involved in memory has been based on fine-tuning the regulatory properties of metabolical, and to a degree, electrochemical processes occurring in the synapse. It did not need the involvement of new organs, nor even cells. Therefore unlike the invention of flight, which nota bene has been worked out by the evolution more than once, optimization of memory can be compared to tuning up a radio receiver as opposed to constructing it from scratch (as in the case of developing wings and the ability to fly).

It is not difficult to notice that the memory mechanisms have been very much conserved in the course of philogenesis. After all, famous American neuroscientist Dr Eric Kandel has spent a few decades studying the primitive nervous system of a mollusc *Aplysia caliphornica* (just a couple of nerve cells making up the entire system). Kandel's involvement with Aplysia did not prevent him from drawing far-reaching conclusions concerning the mechanisms of memory in humans. Similarly, one of the most important discoveries in the molecular research on memory in the last decade, involvement of the membrane protein kinase C in conditioning, has first been spotted in a marine snail *Hermissenda*.

In other words, evolution did not take long to figure out the optimum properties of memory, which, according to Wozniak, are as widespread in the nervous system as the citric acid cycle in the organism.

In the next sections, we will have a look at optimum properties of memory and their relationship with SuperMemo. If indeed evolution is infallible, SuperMemo makes the best use of optimum memory properties.

Evolutionary optimization of forgetting

(from Forget about forgetting)

Let us have a more detailed look why, according to Wozniak, memory and forgetting work the way that makes SuperMemo possible.

To think about the brain as about a computer is a very useful metaphor. Everyone who has some basic understanding of computation will know that no computer can solve problems without memory. Memory is needed to keep the record of the computation; however, it can also be used to keep a modifiable program. After all the power of computers rests in their programmability. Human beings, more or less consciously, program their brains using the so-called long-term memory, i.e. memory which lasts for months and years. However, they can also use short-term memory, different in its physiological nature, to keep the record of the computation, or thinking, which leads to the solution, response, reflex, etc. Short-term memory, apart from its short-term functions, also serves as the framework for establishing long-term memories.

One of the first questions the user of a PC asks is how much RAM does a computer have? The same question was asked by the evolution in reference to the brain. Human RAM is enormous in its capacity. Some researchers estimate its size to several gigabits (Wozniak, using a simple mathematical models of learning, has also arrived to the hypothetical life-time limitation on the learning capacity of the brain; see later). However, memory is not unlimited, and a living organism cannot attempt storing all incoming information. A very substantial selection has to be made if the storage capacity is not to overflow in a life-time. It appears that the solution is forgetting. Let the brain filter the incoming messages and store as much as it is only possible in the long-term memory. Then let forgetting do the rest of the job by eliminating pieces of information in the order of least relevance.

An important question that had to be answered by the evolution was in what order should pieces of information be forgotten so that to maximize the survival rate. It is obvious, at least for those who understand the concept of probability in an incompletely specified event space, that encountering an average event increases the probability of the same event being encountered again. For example, if you do not know Mr X and you meet him on the street today, the probability that you meet him again tomorrow must be consider greater than from before the first meeting. Naturally, if you meet him again, you have yet more reasons to believe in more meetings in the future. In other words, successive repetitions should have an increasing stimulatory effect on memory. Unfortunately, evolution proceeded mostly in the absence of volitional aspects of the human brain; hence, we do not have the capability to forget at will. We cannot decide to free memory by forgetting Mr X on hearing the news that he has died or moved away to the antipodes.

Spacing effect

(from Forget about forgetting)

The little problem remains of how the brain can prevent events that are not likely to be encountered in the future from being permanently transferred to memory as a result of a great number of repetitions? The answer was found in applying the so-called spacing effect, which says that the longer the interval between repetitions, the better the memory effect. This way a large number of repetitions in short intervals has very little impact on memory. Simply speaking, memory uses the spacing effect and the principle of increasing intervals to most effectively fix relevant information in the brain. Upon encountering an event it is temporarily transferred to long-term memory and forgotten in the matter of a few days. However, if the event is reencountered, the memory assumes increased probability of the event in the future and increases the retention period. Initially, in the retention period, memory is not sensitive to more encounters of the same event. Only at later stages does memory become sensitive again and a new encounter will act as a repetition that will increase the retention period and make memory temporarily insensitive to further encounters. If anybody doubts the importance of the spacing effect, Wozniak proposed to consider the following example:

Could the reader provide the name of the infamous lady that alleged having slept with the majority of Polish parliamentarians? If the reaction is: *Sure, yes, wait a second, I am sure I remember it but ... aha!* then this can be taken as an example of spacing effect. Despite the fact that the lady has dominated Polish political life for a short period of time, many of us might find it hard to recall her name. The reason is simple, hundreds of repetitions concerning the name of the lady were cramped in a very short period of time. Because of the spacing effect, memory reacted to the phenomenon more like to a single repetition rather than a volley of memory stimulations. The biological value of such a property of the brain may be explained by the fact that events occurring densely in a short period of time may be unworthy of the precious memory storage. Otherwise, a great number of repetitions in a week could leave a useless memory trace for lifetime. Do we really need to remember the name of the promiscuous lady? We dont ... unless we are members of Polish parliament at breeding age, naturally.

Using again the computer metaphor, the problem of choosing the least relevant pieces of knowledge in the process of forgetting is analogous to the problem of paging in virtual memory. In paging, the question is which memory blocks should be discarded to maximize the probability that the next memory reference will concern a block that is already placed in memory.

Unlike in operating systems, the LRU algorithm (Least Recently Used) would not work fine for human memory. If LRU were used, first to forget would come the rusty primitives mastered in the primary school. It would be enough to use a calculator for a few months to have all the multiplication table discarded in priority behind the morning breakfast. The grandmother that has passed away a decade ago would serve as another early victim. Definitely, LRU would deprive the brain of flexibility and us ... of humanity.

Human memory vs virtual memory in operating systems

(from Forget about forgetting)

The question arises immediately: If the biological optimization of the memory storage is as efficient as in case of humans, why do not developers of operating systems assign memory attributes to blocks of memory, and use increasing-intervals combined with the spacing effect in developing, say, the next version of Windows. The key to the answer is in one major difference between the brain and the operating system: memory blocks can be reloaded from the disk in a wink which is not true with forgotten memories. You will not see a student at an exam say to the examiner: *Wait a second, I have just forgotten it, and must reload it from my slow external storage*. Obviously, a crib, or any kind of external reference can serve as a smart crutch for those who do not wish to burden their mind with the effort of remembering. Sadly, in the dog-eat-dog pace of our civilization, the LRU approach becomes more and more often applied in humans. Cribs, help systems and encyclopedias play a greater role than the memory training. The poor record of American graduates in verbal, analytical and logical tests as compared with Chinese, Koreans, or even students coming from Eastern Europe is a sad side effect of a dynamic capitalist economy promoting the shallow LRU education and a race to early accomplishment at any price. Does this LRU trend bode ill for SuperMemo? No, says Wozniak, individuals and governments has long realized the importance of education targeted on areas lifetime-applicable in the activities of the modern man. The pressure of the urgent is considered a negative factor not only in education. Even in business! Get into the office of a modern businessman, arguably the primary candidate for stress-related heart disorders (consequence of LRU thinking and prioritizing), and increasingly often you will find in broad display famous maxims targeted on fighting urgency. To ground the belief in the new trends even deeper, it is worth to note that businesspeople are indeed one of the major customer groups of SuperMemo World.

Retrievability and stability of memory

(from Forget about forgetting)

We got to the point where the evolutionary interpretation of memory indicates that it works using the principles of increasing intervals and the spacing effect.
Is there any prove that this action of memory apart from the evolutionary speculation?
In his Doctoral Dissertation, Wozniak discussed widely molecular aspects of memory and has presented a hypothetical model of changes occurring in the synapse in the process of learning. The novel element presented in the thesis was the distinction between the stability and retrievability of memory traces. This could not be used to support the validity of SuperMemo because of the simple fact that it was SuperMemo itself that laid the groundwork for the hypothesis. However, an increasing molecular evidence seems to coincide with the stability-retrievability model providing, at the same time, support for the correctness of assumptions leading to SuperMemo. In plain terms, retrievability is a property of memory which determines the level of efficiency with which synapses can fire in response to the stimulus, and thus elicit the learned action. The lower the retrievability the less you are likely to recall the correct response to a question. On the other hand, stability reflects the history of earlier repetitions, and determines the extent of time in which memory traces can be sustained. The higher the stability of memory, the longer it will take for the retrievability to drop to the zero level, i.e. to the level where memories are permanently lost.
According to Wozniak, when we learn something for the first time we experience a slight increase in the stability and retrievability in synapses involved in coding the particular stimulus-response association. In time, retrievability declines rapidly; the phenomenon equivalent to forgetting. At the same time, the stability of memory remains at the approximately same level. However, if we repeat the association before retrievability drops to zero, retrievability regains its initial value, while stability increases to a new level, substantially higher than at primary learning. Before the next repetition takes place, due to increased stability, retrievability decreases at a slower pace, and the inter-repetition interval might be much longer before forgetting takes place. Two other important properties of memory should also be noted: (1) repetitions have no power to increase the stability at times when retrievability is high (spacing effect), (2) upon forgetting, stability declines rapidly.

Molecular basis of memory

(from Forget about forgetting)

As mentioned earlier, the molecular mechanisms thought of underlying the memory have not been used as the basis to develop SuperMemo. Though the cross-inspiration was mutual, it is rather the retrievability-stability model which is likely to contribute more to understanding the molecular aspect of memory than vice versa. The correlates between the model and the findings on molecular memory might not be striking at first. After all most of research on memory consequently turns around the concept of the retrievability of a synaptic connection. The concept of stability is absolutely new and no mention of similar phenomena can be found in widely published research. However, both short-term memory, as well as the components of long-term memory: retrievability and stability, fit nicely into the presently investigated models of memory and learning.

Internals of SuperMemo

(from Forget about forgetting)

We have already seen that evolution speaks for SuperMemo, findings in the field of psychology coincide with the method, and that facts of molecular biology and conclusions coming from Wozniak's model seem to go hand in hand. Here is the time to see how the described mechanisms have been put to work in the program itself.
In the course of repetitions, SuperMemo plots the forgetting curve for the student and schedules the repetition at the moment where the retention, i.e. proportion of remembered knowledge, drops to a previously defined level. In other words, SuperMemo checks how much you remember after a week and if you remember less than desired it asks you to make repetitions in intervals less than one week long. Otherwise, it checks the retention after a longer period and increases the intervals accordingly. A little kink to this simple picture comes from the fact that items of different difficulty have to be repeated at different intervals, and that the intervals increase as the learning process proceeds. Moreover, the optimum inter-repetition intervals have to be known for an average individual, and these must be used before the program can collect data about the real student. There must be obviously the whole mathematical apparatus involved to put the whole machinery at work. All in all, Wozniak says that there have been at least 30 days in his life when he had an impression that the algorithms used in SuperMemo have significantly been upgraded. Each of the cases seemed to be a major breakthrough. The whole development process was just a long succession of trials and errors, testing, improving, implementing new ideas, etc. Unfortunately, those good days are over. There have not been any breakthrough improvement to the algorithm since 1991. Some comfort may come from the fact that since then the software started developing rapidly providing the user with new options and solutions. Can SuperMemo then be yet better, faster, more effective? Wozniak is pessimistic. Any further fine-tuning of the algorithms, applying artificial intelligence or neural networks would be drown in the noise of interference. After all, we do not learn in isolation from the world. When the program schedules the next repetition in 365 days, and the fact is recalled by chance at an earlier time, SuperMemo has no way of knowing about the accidental recollection and will execute the repetition at the previously planned moment. This is not optimal, but it cannot be remedied by improving the algorithm. Improving SuperMemo now is like fine tuning a radio receiver in a noisy car assembly hall. The guys at SuperMemo World are now less focused on science. In their view, after the scientific invention, the time has come for the social invention of SuperMemo.

In 1995, one year after writing this article, a new SuperMemo algorithm has been developed that promises to still increase the speed of learning, esp. in the very first weeks of repetitions. For more see Optimization algorithm used in spacing repetitions in SuperMemo

Theory and practice

(from Forget about forgetting)

Using a simple mathematical model, according to Wozniak, one can easily predict how the learning process will look like in the long perspective. One of the most striking observations is, that apart from the initial period, the speed of learning does not decrease substantially in time (one would rather expect a rapid decline of the knowledge acquisition rate because of the accumulation of outstanding repetitions). Another interesting fact is that even with SuperMemo, one is not likely to master more than a several million facts and figures corresponding to SuperMemo items in lifetime. The average learning speed of an average student amounts to about 300 items/year/min (i.e. the student can memorize 300 items per year if he or she works one minute per day).

This theoretically predicted speed of learning has been confirmed by Wozniak and Gorzelañczyk more than once in small groups of subjects. A recent poll conducted by SuperMemo World among all registered users in Poland, indicates that the average speed of learning reported by registered users of SuperMemo is also close to 300 items/year/min, though individual differences have been more than substantial (from 50 to 3000 items/year/min), and extreme values had to be rejected for a more reliable picture.

Simulation experiments based on Wozniak's model of learning show that a student who stops repetitions after a 5-year-long work with SuperMemo is likely to forget 60% of the learned material in the first year after the cessation! Though for shorter periods of time, this staggering figure has been confirmed in practice.

At this point one might be disappointed with the volatility of knowledge gained with SuperMemo, but the above figures also confirm once again that learning without SuperMemo is no learning at all.

It might work but it cannot be that good

(from Forget about forgetting)

If one is convinced of the validity what has been said about SuperMemo until now, will he or she be already convinced that the program is a perfect cure for the ailing memory?
Can it really capitalize on the properties of the nervous system and let learning proceed a dozen times faster than in standard circumstances? After all there have been generations of students trying to figure out better methods of learning, and a breakthrough comparable with what SuperMemo claims to be seems to be highly unlikely even to quite an open-minded observer.
Wozniak discounts the low-probability argument as the viable source of scepticism, and says that he has personally traced down evidence that SuperMemo-like approaches to learning have already been tried before with lesser or greater degree of success. Moreover, it is worth noticing that SuperMemo might not see the light were it not implemented as a computer program which can easily be transferred between individuals. In other words, it could have fallen into oblivion as the previous attempts to put order in the process of learning. One must remember that the skeletal algorithm of SuperMemo has been formulated in 1985, and only 1987 saw its very slow expansion in selected scientific circles in Poznañ. Another turning-point to be kept in view is that SuperMemo World would not have been formed in 1991 were it not for the inspiring meeting of minds between Wozniak and his colleague from the university, Krzysztof Biedalak, presently SuperMemo World's Vice-President. Both top-students at the university, Wozniak intended to study neuroscience in the US, Biedalak wanted to do the same in the field of artificial intelligence. Only by coincidence, they were both thrown into the world of entrepreneurial science.
All this shows that despite the fact that the principles of SuperMemo are extremely simple and might have been invented several dozen times independently in several dozen countries of the planet, SuperMemo is not just a run-of-the-mill. The distinctive merit of SuperMemo World was to put the idea in practice, invest a great deal of man-hours in development of software, and focus on marketing the idea to the potential customer. Otherwise, SuperMemo would remain for ever limited to the small circle of its early enthusiasts.

How was SuperMemo developed?

(from Forget about forgetting)

Perhaps, while in the context of fulfilled-vs-unfulfilled inventions, it is interesting to take a short look at the entire story of SuperMemo from its very beginning.
It was 1982, when 20-year-old student of molecular biology at Adam Mickiewicz University of Poznañ, Piotr Wozniak, became quite frustrated with his inability to retain newly learned knowledge in his brain. This referred to the vast material of biochemistry, physiology, chemistry, and English, which one should master wishing to embark on a successful career in molecular biology. One of the major incentives to tackle the problem of forgetting in a more systematic way was a simple calculation made by Wozniak which showed him that by continuing his work on mastering English using his standard methods, he would need 120 years to acquire all the important vocabulary. This not only prompted Wozniak to work on methods of learning, but also, turned him into a determined advocate of the idea of one language for all people (bearing in mind the time and money spent by the mankind on translation and learning languages). Initially, Wozniak kept increasing piles of notes with facts and figures he would like to remember. It did not take long to discover that forgetting requires frequent repetitions and a systematic approach is needed to manage all the newly collected and memorized knowledge. Using a obvious intuition, Wozniak attempted to measure the retention of knowledge after different inter-repetition intervals, and in 1985 formulated the first outline of SuperMemo, which did not yet require a computer. By 1987, Wozniak, then a sophomore of computer science, was quite amazed with the effectiveness of his method and decided to implement it as a simple computer program. Effectiveness of the program appeared to go far beyond what he had expected. This triggered an exciting scientific exchange between Wozniak and his colleagues at Technical University and Adam Mickiewicz University. A dozen of students at his department took on the role of guinea pigs and memorized thousands of items providing a constant flow of data and critical feedback. Dr Gorzelañczyk from Medical Academy was helpful in formulating the molecular model of memory formation and modelling the phenomena occurring in the synapse. Dr Makalowski from the Department of Biopolymer Biochemistry contributed to the analysis of evolutionary aspects of optimization of memory (NB: he was also the one who suggested registering SuperMemo for Software for Europe). Janusz Murakowski, MSc in physics, currently enrolled in a doctoral program at the University of Delaware, helped Wozniak solving mathematical problems related to the model of intermittent learning and simulation of ionic currents during the transmission of action potential in nerve cells. A dozen of forthcoming academic teachers, with Prof. Zbigniew Kierzkowski in forefront, helped Wozniak tailoring his program of study to one goal: combining all aspects of SuperMemo in one cohesive theory that would encompass molecular, evolutionary, behavioral, psychological, and even societal aspects of SuperMemo. Wozniak who claims to have discovered at least several important and never-published properties of memory, intended to solidify his theories by getting a PhD in neuroscience in the US. Many hours of discussions with Krzysztof Biedalak, MSc in computer science, made them both choose another way: try to fulfil the vision of getting with SuperMemo to students around the world.

Epilog

(from Forget about forgetting)

When I asked Wozniak if his models like retrievability-stability model, model of optimum repetition spacing, etc. have been confirmed by other researchers in the field of memory and learning, I did not get an unambiguously affirmative answer. After all, says Wozniak, the outline of his methodology employed in SuperMemo has only been published in a worldwide scientific journal only in 1992 (Acta Neurobiologiae Experimentalis), and all his basic findings build up on the model of optimum repetition spacing. Why did he wait so long with publishing the theory in a respectable journal? According to Wozniak, as early as in 1990, he first attempted to publish the results of his early experiments on repetition spacing in several journals, including the most renowned Memory and Cognition. However, the editors, though expressly praising the novel approach to studying memory, constantly tossed him from journal to journal claiming that his paper did not comply exactly with their target profile. Those involved in psychology complained about the intrusion of convoluted computer algorithms, while those closer to mathematics did not want to see their journals review recent literature on spacing effect in memory. Both these components, and more, however, are central to SuperMemo. A great deal of scepticism has also been generated by the regularity of the findings. Wozniak says: *the experimental data looked too good to be true; more like they were cooked for the paper*.

All in all, one can either trust Wozniak and try SuperMemo, or wait months or years before its true scientific recognition. In the meantime, the marketing team of SuperMemo World is beaming with optimism: *it's enough to ask users of SuperMemo, forty thousands of them in Poland only, how the method fares in their educational pursuits. The general opinion is more than enthusiastic. SuperMemo simply works and we do not need to prove it to our customers*.

In questionnaires received at SuperMemo World, when asked what they like most in the program, users of SuperMemo overwhelmingly indicate its effectiveness. The software may be OK, but what really counts is results in learning. How about dislikes? Users are not pleased with this or that, most often with the fact that, even in Poland, SuperMemo is always released first in English. But there is no particular turn-off that takes precedence. Definitely, nobody questions the fact that with SuperMemo one can learn faster and never to worry about forgetting.

Taking this rosy picture into heart, one might wonder why has SuperMemo not yet sold in millions of copies worldwide. Marczello Georgiew, Marketing Director at SuperMemo World proposed to recall the problems Graham Bell experienced when trying to introduce his *funny machine for talking over a wire* or how pessimistic the predictions of industry futurologists were about the expansion of the *air-polluting mechanical horse*. Then he adds confidently: *It took Wozniak 10 years to turn necessity into invention, give us half this time, and we will turn his invention into global necessity*.

Implementing the SuperMemo method with SM8OPT.DLL

Piotr Wozniak, Dec. 17, 1996

SM8OPT.DLL is a library designed for software developers intending to implement the SuperMemo method in their software. Licensing conditions for using SM8OPT.DLL will differ depending on the target application (contact SuperMemo World for more details). In particular, no license fee is required for projects whose sole purpose is research on repetition spacing or on the Algorithm-SM8 used by SM8OPT.DLL. Nevertheless, written permission from SuperMemo World must be obtained beforehand in all cases.

SM8OPT.DLL can be called from any application as any standard Windows DLL library. The job of the developer is to:

* store data pertaining to the learning process (a record less than 20KB in size, later called OptimizationRecord)
* store data pertaining to the individual items of knowledge (a record of about 20 bytes per item, later called ItemOptimizationData)
* execute repetitions on days determined by the date stored at ItemOptimizationData.NextRepetition
* call SM8OPT.DLL at the moment of repetition to obtain new values of OptimizationRecord and ItemOptimizationData

Here is the detailed algorithm for using SM8OPT.DLL:

1. Call GetOptimizationRecordSize to obtain the size of OptimizationRecord.
2. Allocate a memory block of the size determined in Step 1. This block will be used to keep OptimizationRecord throughout the learning session.
3. In the first session, call CreateNewOptimizationRecord(OptRec) where OptRec is a pointer to the memory block allocated in Step 2. This will initialize the value OptimizationRecord. In all following sessions load OptimizationRecord from a file created in Step 12.
4. After creating or loading OptimizationRecord in Step 3, call InitializeOptimizationRecord(OptRec) that will create all temporary variables used by the repetition spacing algorithm.
5. By analogy, for each item call GetItemOptimizationDataSize, allocate memory for ItemOptimizationData, and call CreateNewItemOptimizationData(IOD) where IOD is a pointer to the allocated memory block. Alternatively reload ItemOptimizationData from a file.
6. Having OptimizationRecord and ItemOptimizationData available in memory (pointers OptRec and IOD), repeat the procedure of executing individual repetitions presented in Steps 7 through 12.
7. Using ItemOptimizationData.NextRepetition field (IOD^.NextRepetition) determine the date of the next repetition for each individual item (see later for more on interpretation of NextRepetition value).
8. On the date of repetition call GetRepetitionData(IOD) with IOD pointing to ItemOptimizationData. This will set the new value of ItemOptimizationData with the field NextRepetition pointing to the day on which the next repetition will take place (see later for more on interpretation of NextRepetition value).
9. Before calling GetRepetitionData, set two fields of ItemOptimizationData: Grade - the grade provided in the course of the repetition, and Today - current date relative to the first day of the learning process (see later for more).
10. GetRepetitionData can be called several times in succession (e.g. for grade-undo purposes).
11. Call CommitOptimizationRecord (no parameters) after the final call to GetRepetitionData and after the confirming the validity of the repetition (e.g. by choosing by the user the Next Repetition button). CommitOptimizationRecord updates

all temporary variables of the repetition spacing algorithm and updates OptimizationRecord (pointer to this record was provided in Step 4 with InitializeOptimizationRecord and does not have to be specified again).

12. Save ItemOptimizationData to a file (the new value of all fields was determined in Step 8 by calling GetRepetitionData).
13. Upon completing the series of repetitions, save the OptimizationRecord to a file. Call FreeOptimizationRecord and deallocate the memory blocks allocated in Step 2 and Step 5.

Here is the definition of ItemOptimizationData record:

```
TItemOptimizationData=record
        Today:word; {relative to the first day of the learning process. On the first day, Today=1}
        Grade:byte; {grade from 0 to 5, where 5 is the best grade}
        FirstGrade:byte; {first grade ever obtained by the item during its first repetition}
        LastRepetition,NextRepetition:integer; {relative to the first day of the process}.
        OldInterval,UsedInterval,OptimumInterval,NewInterval:integer;
        Repetitions,NewRepetitions:byte;
        Lapses,NewLapses:byte;
        RequestedFI:byte;
        Ordinal:real;
        AFactor,NewAFactor:real;
        UFactor,NewUFactor:real;
        OldRF,NewRF,OldOF,NewOF:real;
        Cases:word;
        EstimatedFI,ExpectedFI:real;
        NormalizedGrade,NGMin,NGMax:real;
        RepetitionsCategory:real;
end;
```

The only fields that are indeed needed by the developer are: Today, Grade and NextRepetition. Other fields can be used for diagnostic or information purposes.
Today and NextRepetition are counted relative to the first day of the process. It is on the first day of learning, Today=1, and the repetition takes place on the date on which NextRepetition=Today.
Note that LastRepetition and NextRepetition values may be negative in cases where items have been transferred from another system that was created before the initialization of the system in use.

As some of the fields of ItemOptimizationData are computed anew, only a subset of TItemOptimizationData has to be stored in a file. Developers that would want to save disk space may note that only the following fields have to be stored: **FirstGrade, LastRepetition, OldInterval, Repetitions, Lapses, RequestedFI, Ordinal, AFactor, and UFactor**.
Most notably, NextRepetition does not have to be stored as it can be derived from NextRepetition=LastRepetition+OldInterval.
After a repetition, new values of those fields have to read from: NewInterval, NewRepetitions, NewLapses, NewAFactor and NewUFactor. The new value of LastRepetition equals Today.
The definition of TOptimizationRecord is not presented here as it is quite meaningless without understanding the details of the repetition spacing algorithm denoted Algorithm-SM8.

The use of GetOptimizationRecordSize and GetItemOptimizationDataSize is obligatory as in future releases of SM8OPT.DLL, the size of the optimization data may change. Obviously,

there will be no change to the present structure apart from adding new fields taking part in enhanced optimization. For example, the RepetitionCategory has been introduced in December 1996 in a new release of Algorithm SM8 in which interval categories instead of repetition numbers are used to index the matrix of optimal factors. This prevents fooling the algorithm in cases the actual repetition takes place long after the scheduled optimum date.
In addition, SM8OPT.DLL exports the following function that may be of use for developers interested in implementing repetition rescheduling mechanisms:

function ComputeMercyOrdinal(IOD:PItemOptimizationData;ScheduleLength:word):longint;

This function produces and ordinal number from ItemOptimizationData and the proposed length of the rescheduling period. Such ordinals can be used in sorting items for rescheduling. The ordinals range from 0 to 10000 and items with low ordinals should be scheduled first.
SM8OPT.DLL is also available as a Delphi DCU unit file for incorporation in Delphi executables.

Here is the list of all procedures published in SM8OPT.DLL:

procedure CreateNewOptimizationRecord(OptimizationRecord:POptimizationRecord)
function GetOptimizationRecordSize:word
procedure InitializeOptimizationRecord(OptimizationRecord:POptimizationRecord)
procedure FreeOptimizationRecord
function GetItemOptimizationDataSize:word
procedure CreateNewItemOptimizationData(IOD:PItemOptimizationData)
procedure GetRepetitionData(IOD:PItemOptimizationData)
procedure CommitOptimizationRecord
function ComputeMercyOrdinal(IOD:PItemOptimizationData; ScheduleLength:word; var MercyFactors:TMercyFactors):longint

SM8OPT.DLL is available for download from http://www.SuperMemo.com.

Append Node Change

To make sure that the performance of the program does not deteriorate as a result of adding an excessive number of children to a single hierarchy node, SuperMemo provides a user-defined parameter **Tools : Options : Appending : Children limit per hierarchy node**. By setting this value, you can make sure that SuperMemo will warn you each time you exceed the determined limit (you can set it anywhere between 1 and 9999). It seems sensible to never add more than a pageful of items to a node for easy review in the Contents window. Therefore you are most likely to set the children limit to 10-60.

When you choose one of the appending options (e.g. **Edit : Add a new item** or **Edit : Append new items**, etc.), SuperMemo will check the number of children at the current append node hook and display **Append Node Change** dialog box to let you choose the new append node hook or let SuperMemo determine the new node automatically.

In **Append Node Change** dialog you can choose one of the following:

* **OK** - to let SuperMemo choose the new hook node automatically. The new node hook will naturally be a descendant of the append node root as determined by **Tools : Append Nodes**
* **Change limit** - to change the children limit per hierarchy node
* **Select node** - to manually choose the new hook node in the Contents window (by choosing **Make hook** on the context menu or pressing *Ctrl+D*)

If you always want SuperMemo to choose the new node hook automatically, you can tick the check-box **Do not show this dialog box**.

If you opt for automatic selection of the new hook of the current append node, you should know that you can choose between two modes of selecting the new hook:

* selecting the new hook at the sparsiest location among the descendants of the current append nodes root (**Tools : Options : Appending : Automatic append node change : To sparsiest branches**)
* selecting the new hook at the last branch of the current append nodes root that does not exceed the children limit (**Tools : Options : Appending : Automatic append node change : To terminal branches**)

Build your vocabulary with SuperMemo

Krzysztof Biedalak, Piotr Wozniak, SuperMemo World

Spotlight, October, 1996

Have you ever heard of SuperMemo? If not ... youve got a serious gap in your knowledge of tools for learning. SuperMemo is a knowledge machine that lets your remember things almost without fail.

As an avid reader of Spotlight, you are probably eager to get a better command of English. Thats where SuperMemo comes in.

Heres how it works. Imagine you have heard a new word - for example: morganatic. If you want to make sure you remember morganatic youll probably have to repeat it a few times. Nothing gets to your memory and stays there for ever without practise. If you do not use the word, or do not repeat it in some way ... it is gone! The question is how often the repetitions should take place. You probably would answer: if I do not refresh it in a week, the word is no longer there. You are right. Most of us, need a repetition within a week to stand a 90% chance to recall a word or phrase. But is a single repetition all that you need? Nope. So how about the second repetition? Should it come again in a week or can you afford a month-long break? There is no simple answer to the question.

With a personal computer and SuperMemo, you do not have to think about anything but the material you want to learn. You put full focus on the knowledge and zero focus on your private memory management.

It is all as simple as this:

(1) turn on the computer, (2) click the SuperMemo icon on the screen, and (3) press the Learn button. Now just answer the questions on your screen and grade yourself (by pressing buttons Pass, Fail, etc.). The grading lets the program know how often to bring particular words back to your memory and tracks the record of how well youve remembered them. In a month or two, SuperMemo will understand your memory better than you yourself do. This will let it schedule repetitions for you with accuracy that no other force on earth can beat. If you grasp the general idea, you will soon be banging grades and repeating in mind thousands of newly memorized words and phrases.

All you have to do is to ask your computer-literate friends how to obtain a copy of the program, and how to set your first steps. All you need to know about SuperMemo can be learned within an hour!!!

We are keen users of SuperMemo ourselves. In 1991 we set up a company called SuperMemo World. The company devotes all its resources to popularizing SuperMemo and developing new versions of the program. Believe it or not, by the end of 1996 we will have distributed 100,000 copies of SuperMemo and the number of shareware copies in use may be several times that! The method that was invented in 1985 and implemented as a computer program in 1987, has stayed virtually unknown until 1992 when the program received a series of awards at some prestigious software competitions in Poland and abroad.

Do you want to get great scores on FCE, TOEFL and GRE tests? Lots of people use SuperMemo for the purpose! We can point to people who have memorized 30,000 words and phrases in a single year! Naturally, those are the sturdiest and most dogged guys around. But the figures produced even by ordinary lazybones are still stunning.

There are hundreds of little and big databases available for use with SuperMemo (a database is a collection of questions and answers used in the process of learning). Recently, you can also enhance the learning process by sound, live video, animation and much more. The newest versions of SuperMemo provides you not only with the SuperMemo method, but also with the whole package of modern software tools that you will find useful in the process of learning.

OK, folks. Time to get down to serious business with English. Get yourself a copy of SuperMemo on your desk and ... Right on!

Short history of SuperMemo

1982 - Application of the recall principle

1984 - Application of the minimum information principle

1985 - First ever application of the optimum spacing of repetitions

1987 - First computer implementation. SuperMemo 1.0 introduces the concept of E-factor

1988 - SuperMemo 2 and SuperMemo 3

1989 - SuperMemo 4 - optimization based on the matrix of optimal intervals
SuperMemo 5 - optimization based on the matrix of optimal factors

1990 - *Optimization of learning* with its enthusiastic predictions about the future of SuperMemo provides an inspiration and incentive to commercialize the method. It also includes a comprehensive theoretical analysis of the learning process conducted with SuperMemo and presents the model of intermittent learning

1991

July 5, SuperMemo World is formed by recent graduates of computer science: Piotr Wozniak and Krzysztof Biedalak. The real capital: one 286 personal computer.
SuperMemo 6 - optimization based on approximation of forgetting curves
November, SuperMemo 6 becomes a finalist of the Software for Europe 1992 Award Competition

1992

March, SuperMemo World displays its products at CeBIT '92
September, SuperMemo 6 receives the first degree award at Softarg '92, Katowice, Poland
September, SuperMemo 6 receives a special award of the Polish Computer Science Society
December, SuperMemo 6 is awarded in the Product of the Year 1992 competition organized by PC Kurier, Poland

1993

January, releasing SuperMemo 7.0 for Windows, one of the first Polish Windows software products
March, releasing SuperMemo 5 as shareware
March, SuperMemo World displays its products at CeBIT '93
May, releasing AudioVisual SuperMemo 7.2 for Windows
August, releasing SuperMemo for Amiga
September, SuperMemo tops the chart of best Polish software applications for Amiga (Amiga Magazine)
September, SuperMemo 7 is awarded at Softarg
December, SuperMemo becomes the first ever Polish software reviewed in Byte International
December, SuperMemo 7 is awarded in Product of the Year 1993

1994

January, SuperMemo is voted 4-th most popular Polish software by readers of Polish PC World (behind anti-virus and word- processing products)
March, SuperMemo for Amiga is voted the Product of the Year 1993 among Polish software for Amiga by readers of Amiga Magazine
March, SuperMemo World displays SuperMemo 7.3 at CeBIT '94
March, SuperMemo is awarded for the best value/price ratio by the CENA/VYKON journal, Czech Republic
April, SuperMemo USA, Inc. is formed
May, SuperMemo 7 is awarded a prestigious Gold Medal at Infosystem Computer Fair in Poznan

July, 10,000-th copy of SuperMemo has been sold
August, SuperMemo USA, Inc. publishes SuperMemo 7.4 on CD-ROM, including the audio release of Advanced English best-seller. This title becomes the first Polish Windows software title on CD-ROM ever
September, SuperMemo World is granted the special award of the fair's president at Softarg and receives a free display at Softarg 1995
September, SuperMemo World signs its first OEM bundling deal with JTT, second largest computer-maker in Poland
October, SuperMemo World publishes the European release of SuperMemo on CD-ROM
October, SuperMemo 7.2 tops the ranking of best-selling software sold on illegal software market, ahead of Microsoft Word 6.0 and Microsoft Works 3.0 (PC World Komputer, Vol. 11, 1994)
November, *Economics of learning* lays the theoretical foundation for SuperMemo 8
December, SuperMemo for Windows receives the Reader's Choice distinction from PC World Komputer (Vol. 12, 1994) by being voted second most popular Polish software in Poland and the most popular Polish software for Windows in Poland

1995

January, Borlands Delphi is chosen over Borland C++ and Visual C++ as the development language for SuperMemo 8.0
January, SuperMemo 6 is released as freeware
February, SuperMemo receives a positive review in the most prestigious European journal: German Chip
March, releasing SuperMemo for MacIntosh
March, SuperMemo Worlds displays its Multilingual SuperMemo 7.5 CD-ROM at CeBIT'95
April, SuperMemo World signs a comprehensive OEM deal with Optimus, the greatest computer-maker in Poland
May, 30,000-th copy of SuperMemo has been sold
September, five hundred beta-testers get their hands on a Pre-Release CD-ROM with the first hypermedia SuperMemo codenamed Genius - the precursor of SuperMemo 8.0.
November, SuperMemo World is the first Polish company ever to display its products at Comdex in Las Vegas

1996

January, SuperMemo is voted the best Polish application by readers of PC World, Komputer
January, first SuperMemo hypermedia CD-ROM title is released: Video English (with Genius 1.0, early precursor of SuperMemo 8.0)
June, SuperMemo Corporation is set up in the US with a goal of making it the first company of Polish origin on NASDAQ
September, Cross Country hypermedia best-seller CD-ROM is released (with Genius 2.0, precursor of SuperMemo 8.0)
Septemer, precursor of SuperMemo 8, Genius 2.0, receives its first award as the hypermedia authoring tool at Softarg, Poland
October, 60,000 readers of CHIP in Poland receive a CD-ROM with the full version of the hypermedia authoring SuperMemo (Genius 2.0) for promotional purposes
November, first Polish multimedia encyclopedia CD-ROM is released together with hypermedia authoring SuperMemo (Genius 2.0) as an integrated tool for learning selected encyclopedic material

1997

January, Polish PWN Encyclopedia with Genius 2.0 (precursor of SuperMemo 8.0) is voted the most popular Polish software product by readers of PC Kurier
January, SuperMemo World, Polish vetaran of CeBIT and the first Polish company present at Comdex makes a strategic conversion towards Internet away from major software fairs

January, SuperMemo Web site is inaugurated (**www.SuperMemo.com**)
February, SuperMemo 8.0 enters beta-testing.
March, scheduled date for SuperMemo 8.0 release

SuperMemo 1.0 (December 1987)

(from Short history of SuperMemo)

SuperMemo 1.0 was a simple question-and-answer program with options for adding new items and learning them in the user determined sequence. It was written in Borlands Turbo Pascal 3.0 by Piotr Wozniak, a junior student of computer science at the Technical University of Poznan.

The most important novelties:

1. First software implementation
2. Applying the optimization to individual items grouped by E-Factor category

SuperMemo 2.0 (July 1988)

(from Short history of SuperMemo)

SuperMemo 2.0 had the look of SuperMemo 1.0 and has mostly enhanced capabilities of the software without adding much to the optimization method.
The program was recompiled with Turbo Pascal 4.0.

In June 1991, SuperMemo 2 was released as public domain software and circled the globe over academic Internet.

SuperMemo 3.0 (October 1988)

(from Short history of SuperMemo)

The whole program was written again from scratch to apply the then popular standards of windows, pull-down menus and other elements of user-friendly interface.

Most interesting enhancements:

1. Database were given separate names (previously each database, made up of 6 files had to be kept in a separate directory!).
2. Item editor activated during repetitions (earlier, learning had to be interrupted to edit items!).
3. Burden parameter became one of the best indicators of user progress.
4. An option for changing system date appeared very useful.

SuperMemo 4.0 (February 1989)

(from Short history of SuperMemo)

SuperMemo 4.0 was only a minor improvement over SuperMemo 3.0 in terms of its functionality, but it has, for the first time, used a flexible function of optimal intervals that would adapt to the users performance.

New features over the earlier versions:

1. Improved optimization method: function of optimal intervals is modified to adjust it to a given learner and given knowledge
2. Option for displaying optimal intervals
3. Option for displaying the number of repetition cases used in computing optimal intervals
4. Option for displaying distribution of intervals
5. Fast search (1000 items per second: 8086, RAM-disk)

SuperMemo 5.0 (October 1989)

(from Short history of SuperMemo)

1989 has seen two turning points in the development of the SuperMemo method. The old repetition spacing algorithm from the earliest versions of SuperMemo gave way to the function of optimal intervals based on the matrix of optimal intervals (SuperMemo 4.0) that was promptly replaced with the matrix of optimal factors (SuperMemo 5.0). In terms of functionality, SuperMemo 5.0 did not go far beyond SuperMemo 3.0 developed just a year earlier. However, its new approach to computing the optimum repetition spacing has provided a foundation for Algorithm SM-6 (1991) and yet sturdier Algorithm SM-8 (1995). In March 1993, SuperMemo 5 has entered public domain and has sinced gained substantial popularity among freeware users around the globe.

New features over the earlier versions:

1. A new, advanced optimization technique based on computing optimal factors rather than optimal intervals
2. Option for displaying optimal factors
3. Option for displaying the approximate values of optimal intervals
4. Option for printing items in the browsing mode
5. Reset operation in the browsing mode (to reset memorized items)
6. Date of the next repetition displayed in the item window

SuperMemo 6.0 (May 1991)

(from Short history of SuperMemo)

SuperMemo 6.0 for DOS was written from scratch in Turbo Pascal 6. Apart from introducing major changes to the repetition spacing algorithm, it was the first SuperMemo designed with a view to commercial distribution. It was also the first commercial SuperMemo that has sold over 10,000 copies. The idea of plotting forgetting curves for all optimal factors has added greatly to theoretical validity of the SuperMemo algorithm that could now be verifiably shown to produce the optimum repetition spacing for a given level of knowledge retention. Future improvements to the algorithm could therefore only accelerate the covergence of the starting memory model to the optimum model.
SuperMemo 6, as the finalist of Software for Europe Competition 1992, has brought SuperMemo to public attention allowing it to break the image of an intricate software product for a hermetic group of technically-minded people. Instead SuperMemo 6 became one of the best-selling pieces of software in Poland in the early 1990s.
In January 1995, SuperMemo 6 entered public domain and is since available from multiple shareware sites and shareware CD-ROMs around the world.

New features over the earlier versions:

1. New optimization technique based on sketching forgetting curves for particular difficulty categories and repetition numbers.
2. User-defined forgetting index is used to control the speed of learning!
3. Daily burden option with the possibility of editing items scheduled for a given day
4. Monthly and annual burden
5. New process parameters: Average repetition, Average lapses, Average interval, Average E-factor, Requested and measured forgetting index, Retention, Mean repetition time, Workload, etc.
6. Extending the item editor by commands such as Paste, Cut, Delete Word, Delete-EOL, Copy Block, Move Block, Delete Block, Word Right/Left, Replace String, Swap Lines, etc.
7. Option Find extended by Replace
8. Option Duplicate Item in the browsing mode
9. Maximum number of items in a database increased from 10,000 to 60,000
10. Option Schedule item in the browsing mode (used to memorize selected intact items)
11. Option Mercy used to evenly distribute outstanding items (e.g. after a vacation break)
12. Option Wipe is used to reduce the number of repetitions by elongating intervals of short-interval items)
13. Option Reset-Wipe is used to reduce the number of repetitions by resetting short-interval items)
14. REPSTR.EXE utility for font and substring conversion
15. Possibility of redefining keyboard codes with KBD files
16. Font Designer shareware included in the package

SuperMemo 7.0 for Windows (January 1993)

(from Short history of SuperMemo)

SuperMemo 7.0 was one of the few Windows applications available in Poland at its time (early 1993). Although initially, it found it difficult to compete with SuperMemo 6, it has ultimately became the most popular SuperMemo ever (over 30,000 copies sold by the end of 1996).
In 1993, it became one of the first Polish applications using sound and bitmapped graphics. In mid-1994, it was also the first Polish Windows software released on CD-ROM.
The number of awards given to SuperMemo 7.0 is too long to be listed here. In 1994 and 1995, SuperMemo 7 was voted the most popular Polish software by readers of PC World Komputer. It was also one of the most eagerly copied pieces of software on illegal software market ahead of the entire Microsoft suite.

New features over the earlier versions:

1. The whole program was written anew from scratch in Borland Pascal 7.0 for Windows using Object Windows Library application framework
2. New database file managing options: Delete database, Rename database, Copy database, Move database, Cut drills (removing items scheduled for Final Drill)
3. Possibility of using several database editing windows at the same time; plus independent Learn, Search and Random Test windows with all operations of the editing mode available. This allows, for example, to browse, search, edit or expand the database during learning
4. Possibility of opening a number of independent Append windows at the same time (e.g. to add or duplicate items at Learn or at Search)
5. Possibility of transferring items and text blocks between all sorts of item windows (via Clipboard)
6. Possibility of duplicating items in the Append window
7. Forgetting curves available from all options displaying optimization matrices (R-Factors, O-Factors, optimal intervals, and Cases)
8. Possibility of working with a few databases at the same time using separate instances of the program

In the following 28 months, the following major versions of SuperMemo 7 have been released: AudioVisual SuperMemo 7.2, SuperMemo 7.3, SuperMemo 7.4 for CD-ROM applications, and SuperMemo 7.5 Programmable that terminated SuperMemo 7 line.

SuperMemo 7.2 (June 1993)

(from Short history of SuperMemo)

SuperMemo 7.2 was commercially the most successful step in the development of SuperMemo 7 line. By adding the possibility of attaching sound and graphics to item, SuperMemo 7.2 has become one of the first Polish multimedia applications.

New features over the earlier versions:

1. Adding Image menu in editing windows (creating up to 60,000 graphic images associated with items)
2. Adding Audio menu in editing windows (recording and playing up to 60,000 sound files associated with items)
3. Configuration AutoSave
4. Remembering location and sizes of working windows in the program configuration
5. Parameter Future Reps estimating the number of repetitions of a given item in the nearest 30 years
6. Possibility of using any of the Windows fonts in any color in SuperMemo items (option Font)
7. Adding case insensitive item search
8. Adding an option for defining text, sound and graphic editors in Options
9. Adding a configuration file for each database with parameters such as desktop, font, color, search string, etc.
10. Enhancing SuperMemo Toolkit by operations on bitmap and sound files

SuperMemo 7.3 (February 1994)

(from Short history of SuperMemo)

SuperMemo 7.3 was probably the greatest functional leap in the development of SuperMemo 7 line, however, it went nearly unnoticed by users of SuperMemo and was soon swept into oblivion by CD-ROM releases of SuperMemo 7.4.

New features over the earlier versions:

1. Adding full-screen item browsers (e.g. for all items, for intact items, for memorized items, for outstanding items, for intractable items, etc.)
2. Possibility of searching for all items containing a given substring, and reviewing these items in the search browser (the search browser sequence is preserved between sessions)
3. Possibility of changing the order of items in the database (by means of the general browser)
4. Possibility of editing the intact queue (by means of the intact browser)
5. Option Item:Transfer in editing windows (used to quickly transfer the current item to another database)
6. RESCUE.EXE integrated with the program as File:Recover
7. Introducing the concept of a leech (item which slows down the learning process). Possibility of extracting leeches using the criteria of E-factor, number of lapses, and the interval. Possibility of browsing leeches and randomly reviewing leeches.
8. Possibility of using filter files with database-to-text conversion tool TO_TEXT.EXE that allows defining (1) items that should be included/excluded in/from the output, (2) strings that should be replaced, (3) comment that should be cut, (4) ordinal number delimiters, (5) definitions of ordinals used in sorting databases, etc.
9. Possibility of quickly opening one of the 5 recently used databases from the pick list on the File menu
10. Automatically opening the last used database at running SuperMemo
11. Resizable question-and-answer field in editing windows
12. Possibility of fast switching between appending and editing modes (e.g. to edit recently appended items)
13. Possibility of independently passing one of the three stages of learning: outstanding, memorizing and final drill
14. Random test on item subsets: intact items, memorized items, intractable items, outstanding items, etc.
15. Possibility of alternative viewing items scheduled for a particular day in the item browser (in Analysis : Burden)
16. Possibility of viewing items scheduled for a particular month by clicking entries in the Monthly Burden
17. Possibility of arbitrarily choosing the first interval when memorizing an item
18. True Type support in printing
19. 256-color bitmap support

SuperMemo 7.4 (September 1994)

(from Short history of SuperMemo)

SuperMemo 7.4 was the first version of SuperMemo released on CD-ROM. Boosted by the best-selling audiodatabase Advanced English, it sold 5000 copies in the late 1994 alone.

New features over the earlier versions:

1. Integrated Toolkit (including tools such as: Recover, Reset, Transfer, Sort, Export text, Import text, Export SM6, Import audiovisual files, Cross-section, SetUp, etc.)
2. Child browsers (e.g. Child:Search makes it possible to selected from the browser all items containing a given substring; Child:Leech extracts all difficult items, etc.)
3. Lexicon and Audio Lexicon make it possible to instantly access words and audio phrases stored in the database (the lexicon may be built for a custom-made database with Build Lexicon using a user-defined filter file)
4. Mercy into the future (e.g. in order to pass all repetitions scheduled for a vacation period before the vacation)
5. Possibility of spelling drills with the Spell-Pad
6. Fuzzy search (e.g. in order to find 'bureaucracy' without remembering the exact spelling)
7. Multiple user menu that makes it possible to define a number of users with separate database directories and database password protection
8. Possibility of using different fonts in questions and in answers (e.g. Latin alphabet in questions and Cyrillic in answers)
9. Four modes of accessing audiovisual files (e.g. in order to repeat audiovisual items whose bitmaps and sound files come from a database stored on CD-ROM, in a colleague's directory, etc.). Possibility of editing audiovisual links in a database with BMP and WAV files stored on CD-ROM.
10. Smart Reset makes it possible to selectively reset the elements of the learning process (e.g. forgetting index, optimization matrices, entire database, etc.)
11. Synchronous Browser's Edit window in browsers makes it easier to browse multiline items, and items illustrated with bitmaps
12. Possibility of defining user's keyboard (by defining KBD files that will assign selected character codes to Alt-key combinations in the item editor)
13. Possibility of comparing one's vocal responses with original audio records in the database
14. 3-D graphs with optimization matrices

SuperMemo 7.5 Programmable (April 1995)

(from Short history of SuperMemo)

SuperMemo 7.5 was the last step in the development of SuperMemo 7 line and is now available with all SuperMemo 7 products. The greatly hyped programmability of SuperMemo 7.5 has not caught wider attention and did not bring the expected rush towards developing new-class databases (this was probably due to intricate DLL interface and little interest in non-standard SuperMemo applications).
With the fullblown switch towards hypermedia, SuperMemo World terminated the development of SuperMemo 7 in early 1995. This move has greatly disappointed many faithful users of SuperMemo who would then have to wait nearly two years for hypermedial SuperMemo 8. However, this allowed SuperMemo World to dedicate all resources to the new development line, and hopefully will soon reward the wait of diehard users of SuperMemo 7.

New features over the earlier versions:

1. Database programmability, i.e. the ability to execute DLL database modules (to open the way to using SuperMemo in procedural learning, problem solving, playing musical instruments, creativity training, etc.)
2. Adding Touch-typing programmed database to every package
3. Tools menu in the browser makes it possible to perform database operations on the contents of the browser
4. Operations Print All, Delete All, Memorize All and Reset All in the browser
5. Subset operations on the contents of the browser (Open, Save, Add, Subtract, and Intersect)
6. Adding Phonetic Lexicon to Advanced English distributed on CD-ROM. Phonetic Lexicon makes it possible to search for English words by using their phonetic transcription
7. Option of selecting individual items in the browser, and creating a child browser with selected items
8. Option Previous in editing windows provides access to up to fifty recently reviewed items
9. Adding final drill browser and history browser (for up to fifty recently used items)
10. Option Item : Drill in editing windows makes it possible to add an item to the final drill queue
11. Option Move : Randomize in the browser makes it possible to scramble the browser's content (e.g. to randomly reshuffle the final drill queue, etc.)
12. Options Edit : Replace and Search : Replace all in the browser
13. Option Browser : Random test makes it possible to run a random test on items kept in the browser
14. Sound importing options in the browser
15. Show answer, Next and Cancel buttons in learning windows
16. Two menu variants: (1) simple (default) and (2) professional.

SuperMemo 8 Pre-Release 2 (codenamed *Genius* 1.0) (January 1996)

(from Short history of SuperMemo : section Developing SuperMemo 8)

New features over the earlier versions:

1. The whole program was written anew from scratch in Borland Delphi using Delphi's Visual Component Library, Plug-In components from Access Softek, and speech compression technology from Vimas International
2. Greatly improved repetition spacing algorithm with: (1) possibility of using different forgetting indices for different items, (2) replacing E-factors with A-factors (absolute E-Factors) that define absolute difficulty of items (E-factors defined relative difficulty of items within a given database), (3) on the fly approximation of the matrix of optimal factors (no need for smoothing, no need for periodic approximation of the matrix with Miscellaneous : Approximation), (4) faster approximation of A-factors by keeping the record of A-factor vs Grade and Grade vs Forgetting index relationships
3. Hierarchical knowledge structure
4. True hypermedia: combination of text, graphics, sound, video, animation, hypertext, executables, scripting, and many more
5. Mnemonic hypertext connects mnemonic topics via a network of hyperlinks
6. Navigation tools modelled on standard multimedia viewing, presentation and help systems
7. Hierchical structure and knowledge system hyperlinks can be rebuilt on the fly using drag-and-drop tools (no recompilation necessary)
8. Programmability via DLL components and interpreted program scripts using commands such as SHOW, PLAY, WAIT, etc.
9. Dynamically updated database lexicon used in word search
10. Multimedia object registries for storing images, sound, video, etc. Supporting formats: BMP, GIF, WAV, and AVI.
11. Multiple choice test exercises
12. Template registries make it possible to quickly create items and topics designed according to a selected style
13. Exchanging topics and topic components via Topic ClipBox and Component ClipBox
14. On-line hints make it easy to understand the function of particular menu items and speed-buttons
15. Number of items virtually limited by the memory of the computer (theoretically: up to 10 million items with 8 million hypermedia objects)

See more about the development of SuperMemo 8 in: SuperMemo 8 Pre-Release 3 and SuperMemo 8.

SuperMemo 8 Pre-Release 3 (codenamed *Genius* 2.0) (September 1996)

(from Short history of SuperMemo : section Developing SuperMemo 8)

SuperMemo 8 Pre-Release 3 added on the functionality of Pre-Release 2 and provided the following enhancements:

New features over the earlier versions:

1. Possibility to hide the SuperMemo shell behind the knowledge system
2. ReadOnly and LearnOnly modes
3. Interactive translation of knowledge system to other languages
4. Simple translation of the program's interface to other languages
5. New script commands: HIDEPROGRAM, SHOWPROGRAM, FONT, WAITGRADE, DRAGMATCH, HITTEST, etc.
6. Support for new file formats: JPEG, MIDI and RIFF
7. Possibility of using knowledge systems directly from CD-ROM
8. Component ClipBox and Hyperlink ClipBox
9. Operations on a selected group of components: alignment, sizing, linking registry items, setting font, etc.
10. Default appending mechanisms (Default Append Node, Default Append Template, etc.)
11. SM7-CDROM file access mode makes it possible to use the program in association with audiovisual databases released on SuperMemo 7 CD-ROMs (Advanced English, Multilingual, MegaMix, etc.)
12. Full record of repetitions (optional)
13. Saving templates with all associated registry objects
14. Customized sound feedback at grading
15. Topics are kept in the pending queue, presented before introducing their child items and dismissed after the first presentation
16. Linear export from the lexicon makes it possible to easily create lexical databases associated with language courses These databases can also be imported back to SuperMemo 7
17. Possibility of changing the forgetting index of the entire hierarchy node
18. Possibility of resetting, committing, dismissing or undismissing nodes in the hierarchy

Pre-Release 3 will be followed in March 1997 by SuperMemo 8.

New features in SuperMemo 8 (scheduled for March 1997)

(from Short history of SuperMemo : section Developing SuperMemo 8)

SuperMemo 8 combines all new features of SuperMemo 8 pre-release versions: Pre-Release 2 and Pre-Release 3. It also adds the following functionality:

New features over the earlier versions:

1. Indexing optimization matrices with repetition categories rather than with repetition numbers. This eliminates the problem of using exceedingly long intervals in cases of long breaks in learning. This problems has haunted SuperMemo since the introduction of the matrix of optimal factors in 1989
2. Fast substring search in the system and in registries. Registry substring search generates registry subsets
3. Transferring hierarchy nodes between different knowledge systems
4. Exporting and importing hierarchy nodes as text
5. Possibility of keeping a collection of append nodes with different append templates
6. Operations Reset, Commit and Dismiss are available in the course of repetitions
7. Possibility of duplicating elements in a single operation
8. Automatic change of the append node on surpassing the limit of children
9. Introducing editable translation and phonetic transcription registries
10. Making phonetic transcription available in the lexicon registry
11. Making translation available in the text registry
12. Random review in browsers, registries and in the element window
13. Two-way history list with Back and Forward buttons
14. Introducing child browsers, browser subsets and browser selections
15. Possibility of working with other hypermedia tools like dictionaries, encyclopedias, etc. via a standard communication interface (first implemented in Polish PWN Encyclopedia developed by Vulcan, Poland)
16. Reset moves pending items to the end of the pending queue
17. Adding substring searching options to Contents, Registries and Browsers
18. Search and Replace in registries
19. Instant translation and instant phonetic transcription on mouse move
20. Providing children and descendant count in the hierarchy
21. Optional synchronization of Contents selection with the contents of the element window and/or Item Data window
22. New statistics parameters: Burden, Average Repetitions, Average Lapses, Average Interval, Average A-Factor, Average Last Repetition Date, Average Next Repetition Date
23. Using custom editors for editing sound and images
24. Automatic student testing
25. Outdated final drill queue is deleted upon confirmation
26. Element filter for selecting items or topics with specified properties such as number of repetitions, length of interval, etc.
27. Activating the debugging version of SuperMemo in Options
28. Option for testing repetition cycle at design time
29. Continuous item appending (like in earlier versions of SuperMemo)
30. Extract section in sound components
31. Making all pop-up menus available from toolbars in Contents, Browsers, Registries and in the element window (earlier available only with right-click)
32. Circular setting individual components in editing mode with Tab
33. Adding hierarchy toolbar
34. External multimedia files can be associated with components (e.g. to compile knowledge systems based on multimedia CD-ROMs)
35. Possibility of resetting optimization matrices and the measurements of the actual

forgetting index
36. Option for counting items and topics in particular nodes of the hierarchy
37. Using universal file browser with multiple-format preview
38. Releasing SM8OPT.DLL with all optimization procedures of Algorithm SM-8 used in SuperMemo
39. Context-sensitive help

Developing SuperMemo 8 (scheduled for March 1997)

(from Short history of SuperMemo)

In late 1994, SuperMemo World started work on SuperMemo 8. The program was supposed to combine all suggestions of users of SuperMemo, journalists, scientists and merge it in one coherent entity based on theoretical grounds delineated in the doctoral dissertation *Economics of learning* by the author of the SuperMemo method, Dr Piotr Wozniak.
The projects was planned to be completed within three years.
As early as in September 1995, SuperMemo World released SuperMemo 8 Pre-Release 1 CD-ROM codenamed Genius Pre-Release. It included concepts that were entirely new to SuperMemo: knowledge hierarchy, hypermedia, mind maps, and more. This product was presented to the hands of 500 beta-testers who have collectively shaped the development of SuperMemo 8 pre-release line over the next 16 months.
In January 1996, the SuperMemo 8 Pre-Release 2, codenamed Genius 1.0 was released as the hypermedia shell for the Video English course of English. This title sold 2000 CD-ROMs in the first month and provided further feedback from users of earlier versions of SuperMemo, as well as from people who had never heard the name SuperMemo before.
In September 1996, the SuperMemo 8 Pre-Released 3, codenamed Genius 2.0 was released as the shell for the Cross Country course of English.
The three Pre-Releases have totalled about 10,000 users providing a solid basis for making sure that SuperMemo 8.0 does not miss functionality and simplicity to which users of SuperMemo 7 have been accustomed. And all that along the new hypermedia capabilities, CD-ROM title authoring, courseware authoring, automatic student testing, and more.

Here again are the main steps in the development of SuperMemo 8.0 in the years 1995-1997:

* SuperMemo 8 Pre-Release 1, September 1995. Released for beta-testing purposes only. Included: hypermedia capabilities, knowledge hierarchy, scripting, programmability, new repetition spacing algorithm, and more.
* SuperMemo 8 Pre-Release 2, January 1996. Released as a shell for Video English. Functionally comparable with Pre-Release 1.
* SuperMemo 8 Pre-Release 3, September 1996. Released as a shell for Cross Country.
* SuperMemo 8, scheduled for March 1997

Grades used in learning

After each repetition, SuperMemo asks the student to produce a grade that is supposed to reflect the accuracy of the provided response.

The grade point scale is defined as follows:
Bright (5), excellent response

Good (4), correct response provided after some hesitation

Pass (3), answer recalled with difficulty; perhaps, slightly incorrect

Fail (2), wrong response that makes you say *I knew it!*

Bad (1), wrong response; the correct answer seems to be familiar

Null (0), complete blackout; you do not even recall ever knowing the answer

Note, that in SuperMemo you are unlikely to need grades **Null (0)** or **Bad (1)**.

Having responded to the question, you can specify a grade by choosing one of the actions:
* click the corresponding grade button at the bottom of the element window

* press the corresponding numeric key

Scientific conquest of the forgetful memory

SuperMemo: a finalist of Software for Europe 1992
Andrzej Horodenski, Computer World, Feb. 17, 1992
(translated and updated January 1997)
(***) marks annotation of the translator

Laziness is the mother of progress. How to learn in order to get good results and not to overwork at the same time?
This question pervaded the mind of Dr Piotr Wozniak, from Adam Mickiewicz University of Poznan, when in 1982, as a student of molecular biology, he commenced his research on mechanisms of memory and forgetting.
The outcome of his least of all lazy work, lasting three years, was a phenomenological model of human memory, which recovered statistical properties of the process of memory formation and forgetting.
Conclusions coming from the model provided guidelines making possible to define optimum repetition scheduling that could be used to minimize learning time.
Consequently, a set of learning principles was developed and named by their author as the SuperMemo method. By sticking to the principles of SuperMemo one can see an up to 50-fold increase in the speed of learning as compared with traditional methods.

Repetitio mater studiorum est - repeating is the basis of learning. True, says Piotr Wozniak joining the over 2000-year-old discussion on the learning methodology. It is not, however, unimportant in what intervals repetitions take place. The effect of memory formation is the greatest if the inter-repetition interval is as long as it is only possible. In SuperMemo, the repetition occurs at the moment when a defined proportion of the material is likely to be forgotten (usually 10%).
The success in learning is determined by proper choice of intervals between repetitions. That is the basic thesis of SuperMemo.

Optimal intervals between repetitions are computed in order to satisfy two opposing criteria:
* to maximize retention, i.e., the proportion of remembered knowledge
* to minimize the learning time, i.e., the number of repetitions

In other words, intervals between repetitions should be long for the sake of minimizing the learning time, and short enough to make sure that the learned knowledge is not forgotten.

As it can be demonstrated, the optimized learning process may closely approach the maximum speed with which human brain can form memory traces.
This can be done with the retention level determined arbitrarily by the student (usually between 90% and 99%). Thereby, using SuperMemo, the student may learn as fast as his brain makes it possible. Moreover, he or she can choose between fast and superficial as opposed to slow and in-depth learning.

To compute optimum intervals between repetitions, SuperMemo uses a sophisticated algorithm. Intervals are different for different students, for different repetition numbers, and for pieces of knowledge with different difficulty. In other words, intervals are adapted to individual learning capability of the student and the character of the mastered material.

The SuperMemo method may be used in all forms of learning from touch-typing, through playing musical instruments, through textbook learning, to performing well in intelligence tests.
Knowledge or skills that are to be mastered should be formulated in the impulse-response form, i.e., usually as pairs of questions and answers.
For example, when learning Esperanto:

Question: What suffix denotes a noun?
Answer: -O

Question: school
Answer: lernejo

or when learning biology:

Question: What does DNA stand for?
Answer: deoxyribonucleic acid

Pieces of knowledge, called items, should be well structured, i.e., they should comply with knowledge structuring principles formulated by the SuperMemo method (e.g., minimum information principle, univocality principle, etc.).
Partition of knowledge into small pieces makes it possible to determine the difficulty of individual questions. Difficulty, expressed as the so-called A-factor, is determined on the basis of the history of repetitions of the item in question. A-factors are one of the basic elements used to determine the date of the next repetition.

An interesting and important characteristics of SuperMemo is the fact that the learning speed is practically independent of the size of the learned material. This implies, for instance, that the learning time for 100,000 items will be, more or less, 100 times longer than in the case of memorizing 1000 items.
In the lifetime perspective, in to neglect the effects of aging, the student will not observe any substantial decrease of his learning power despite accumulation of repetitions from the previously learned material.
In a well-structured SuperMemo database, time necessary to learn and retain a typical item is about one minute per lifetime. For retention of 95%, majority of items will be repeated no more than 10 times in the whole life.
Usually, the first repetition takes place in 1-10 days; however, as soon as after 5-6 repetitions, intervals reach the order of months and years.
By means of repetition spacing, SuperMemo virtually eliminates the problem of forgetting. Although incidental lapses of memory form an inherent part of the method, all forgotten items are put back into the learning process, and the overall knowledge retention may be kept as high as 99%.
It goes without saying that SuperMemo is not inherently a computer product. However, because of the bulk of computation that has to be done after each repetition, the idea of software implementation inevitably springs to mind.

First computer implementation of the method was developed in 1987. In the years 1988-1990, the optimization algorithms of SuperMemo were improved, making it possible to increase the speed of learning yet about twice.
In July 1991, SuperMemo World company was set up with the intention of making the SuperMemo method and software world-wide popular.
The present owners of the company are: the author of the method, biologist and computer scientists, Dr Piotr A. Wozniak, computer scientists Krzysztof Biedalak and Tomasz Kuehn, a physicist, Janusz Murakowski, and a mathematician, Marczello Georgiew.
At present, SuperMemo World distributes SuperMemo version 2 as shareware, and versions 5 and 6 as commercial products (***)(at the moment of translating the text, SuperMemo World had SuperMemo 6 released as shareware and focused on distribution of low-end SuperMemo 7 for Windows and a hypermedial authoring tool: SuperMemo 8 for Windows).

The latest, sixth implementation of the SuperMemo program, SuperMemo 6 Professional has, for several months, been beta-tested in Poznan scientific circles. Its first success SuperMemo 6 had scored long before its first commercial copy was sold. In November 1991, the program

was chosen a Polish finalist of the **Software for Europe** 1992 Award Competition.

In present implementations, knowledge is represented exclusively in the textual form; however, this is absolutely sufficient to obtain spectacular results in many learning domains (***)(SuperMemo 8 is a hypermedia product).
Practically, any information that can be represented in the question-answer form may be used in learning aided by SuperMemo. The learning speed is several times greater than when using any other classical memorization technique.

Work with SuperMemo is straightforward. The student may use existing, ready-made collections of items, called databases, or he may develop his own database according to his own conception. All operation of determining dates and order of repetitions are done automatically so that the student may focus his attention on questions asked by the computer, and on assessing the quality of his responses in 0-5 grade point scale. Grades produced by the student are used in repetition spacing. The grade scale is clearly defined and, after all, indulgent self-assessment for the sake of self-satisfaction makes no sense at all.

Inquisitive individuals may use the program to trace statistical parameters of their progress. Statistical menu options used in supervising the learning process include among others:

* forgetting curve graphs characteristic for a given student and given knowledge
* distribution of repetitions in time (i.e., number of repetitions assigned to particular days, months and years)
* distribution of items with respect to their difficulty (***)(A-factor distribution)
* approximation of optimum intervals which depend on the student, item difficulty, and repetition number
* parameters of the learning process (total number of items in the database, number of memorized items, number of items scheduled for repetition on the present day, average difficulty of items, average interval between repetitions, average number of repetitions per item, average response time, average daily working time, average daily number of repetitions, total response time in the current learning session, proportion of remembered items, etc.)

Until now, the most popular and spectacular application of SuperMemo appears to be learning English vocabulary. SuperMemo World developed a database containing 3000 basic words and phrases of contemporary English (***)(Basic English available from SuperMemo World). The selection of material was based on the word frequency analysis of standard texts. This vocabulary is considered sufficient to use monolingual dictionaries of English. An average student, working 30 minutes a day, will need about 30 days to master the 3000-item database.
An important difference between SuperMemo and other methods of speed-learning, e.g., based on mnemonic techniques, is that the once acquired knowledge is constantly maintained in the student's memory.
For a knowledge hungry student with some background in grammar, the 3000 item database may be mastered in just 3-6 days, thus allowing to develop a basic reading capability in less than a week!

SuperMemo World established the SuperMemo Database Bank, a collection of databases created by users of the program. Most of the existing databases is distributed free (only costs of postage and packaging are
charged). Majority of databases was created for the purpose of learning English. Other domains include foreign languages, medicine, molecular biology, physics, mathematics, general knowledge, computer
science, etc.

See also: Theoretical background of SuperMemo

PS: If you would like to test SuperMemo before buying it, have a look at the silly, old, good SuperMemo 6 for DOS (freeware!).

Creating multiple choice tests, drag&match tests, etc.

Apart from standard question-answer items, you can also create a variety of other tests such as:

* multiple choice tests
* spelling tests
* pronunciation tests
* drag&match tests
* point&click tests, and more

Multiple choice test

To create a multiple choice test do as follows:

1. Create text components: question, correct answer and a couple of wrong answers
2. On the pop-up menu of the correct answer component choose **Non-test** and check the submenu item **Correct**
3. On the pop-up menu of wrong answer components choose **Non-test : Wrong**
4. Test the item by trying to click particular components. The correct response should be highlighted green, and you should get the correct or wrong response feedback (e.g. with custom sounds, etc.)

Spelling test

To create a spelling test, simply place a Spell component in the element window and fill it out with the correct spelling (or with spelling variants separated by a slash). Press *Esc* to move to the presentation mode, and test the spelling by typing in the text in the spelling field and pressing *Enter*. Remember that SuperMemo tries to match the first string among correct answers separated by the slash; therefore you should remember to keep the appropriate sequence of correct answers.

Pronunciation test

To create a pronunciation test, place a Sound component in the element window and check the **Recorder** option on the component pop-up menu. Link the correct pronunciation WAV file by using **Import file** or **Link registry member** on the same pop-up menu. Your pronunciation test is ready. The user can now record himself by pressing the microphone button and play his own or the correct pronunciation by using blue and green play buttons respectively. Finally, he can erase his recording by pressing the crossed-out button. All user recording are kept in TEMP subdirectory of the knowledge system directory and can only be deleted with the recorder crossed-out button. To clean all recordings, go to the TEMP directory and delete all WAV files.

Creating Drag&Match and Point&Click exercises is more difficult and requires using scripts and the hit-area editor.

Point&Click test

An element used for the Point&Click test must contain an image component. The user is supposed to click the right place in the picture. This test may be used, for example, to click the described organ on an anatomical cross-section image. Another example is pointing to the recognized picture in a word recognition test. Here, the question can be a recording of the recognized word, and the test will require clicking the pictorial representation of the word.
To create Point&Click test, place a script component in the element (make sure it is the first playable component, e.g. by placing the script before placing all other components).
To make sure that the script is run as you enter the element, check **Autoplay** on element pop-up . The script must contain the HITTEST command that will evoke Point&Click test, and

all other components of the test if necessary (e.g. playing the question).
In the example below, the script plays the component with the question (component no 3) and calls the HITTEST command to execute the test. HITTEST says that the image component used in the test has the number 4, and that the coordinates of the correct selection area are [(387,10)(444,237)(546,229)(536,4)]
Note that these coordinates can be determined by means of **Edit Hit-Area** on the image component pop-up menu (see below).
It is worth to remember that the numbers of components on a page can be displayed with the 148 button on the **Edit** tab in the element window. Alternatively, **Component order** on element pop-up menu can be chosen for the same purpose.
Here is the script:

```
START 3
HITTEST 4 [(387,10)(444,237)(546,229)(536,4)]
WAITGRADE
GOTO 2
```

Commands WAITGRADE and GOTO 2 make sure that after the test (i.e. after the grade is given to the user), the script goes back to line 2 and makes it possible to resume the Point&Click test.

Single Drag&Match test

This is a test for matching text components with hot spots in a single image components. The users is supposed to drag the text components and drop then at appropriate places in the image component. Single Drag&Match is implemented with a DRAGMATCH script command.
For example:
Let us imagine an exercise in which the users is to drag text components numbered 5, 6, 7 and 8 to appropriate hot spots on the image component number 2. Let us now assume that the hotspot for dragging the text number 5 is delimited by a rectangle having the following coordinates: (123,315,264,357). By analogy, components 6, 7 and 8 have also their own hot spot rectangle coordinates. The following script might do what is needed in this exercise:

```
DRAGTEXTMODE TRANSPARENT
SHOW 5
SHOW 6
SHOW 7
SHOW 8
DRAGMATCH 2 [5,6,7,8] [(123,315,264,357)(34,37,64,77)(43,37,58,177)
(21,25,53,37)]
WAITGRADE
GOTO 2
```

In the above script, DRAGTEXTMODE makes sure that texts are transparently displayed on the image upon dropping. The SHOW commands will redisplay all text components after executing GOTO 2 (dragging will make the components invisible). DRAGMATCH will define the appropriate relationships between text components and hot spot rectangles in the image component number 2. WAITGRADE will suspend the execution of the script until all texts are dragged into appropriate places or until the first mistake is made by the user. Finally, GOTO 2 will allow repeated execution of the exercise until satisfactory performance.
To simplify determining the correct rectangle coordinates, you can use the command **Edit Hit-Area** on the image component pop-up menu. This command is described below.

Multiple Drag&Match test

This is a test for matching text components with a number of image components. The users is supposed to drag the text components to appropriate image components. Multiple Drag&Match is also implemented with a DRAGMATCH script command. This time its syntax differs slightly. There is no image argument. Instead, a list of images is used to match a list of text components.
For example:
Let us imagine an exercise in which the user is supposed to drag text components numbered 2, 3, 4 and 5 to image components numbered 6, 8, 7, and 9. Those two lists of components will form the two arguments of the multiple DRAGMATCH script command. Again DRAGTEXTMODE determines if the texts erase the background or are displayed transparently:

```
DRAGTEXTMODE OPAQUE
SHOW 2
SHOW 3
SHOW 4
SHOW 5
DRAGMATCH [2,3,4,5] [6,8,7,9]
WAITGRADE
GOTO 2
```

The behavior of the above script is analogous to the behavior of the script described in the previous section devoted to single DRAGMATCH command.

Hit Area Editor

Hit area editor is used to easily determine the coordinates of image hot spots for HITTEST script command and for single DRAGMATCH script command.
To use Hit Area Editor follow the steps listed below:

1. Close the Script Editor if you are in the middle of writing a script
2. Right click the image whose hot spots you want to edit
3. Chose **Edit Hit Area** on the pop-up menu
4. At the bottom of the Hit Area Editor choose the command for which editing is to be used: **HitTest** or **DragMatch**
5. For DragMatch exercise, click in turn top-left and bottom-right corners of rectangles describing the hot spots (you can create up to ten rectangles)
6. For HitTest exercise, click in turn all corners of the hot spot area (the area cannot be more than pentagonal, i.e. you should click no more than five points)
7. If you make a mistake, you can clean the editing area by choosing **Clear**
8. After you are done with editing, click **Copy** to move the edited coordinates to clipboard
9. Now you can close Hit Area Editor and return to Script Editor
10. Place the cursor in the appropriate place of the DRAGMATCH or HITTEST command and paste the coordinates from the clipboard (e.g. with *Shift+Ins* or *Ctrl+V*)

Writing EXE and DLL files for SuperMemo 8

Piotr Metzler, Piotr Wozniak, February 1996

To add an EXE component to an element in SuperMemo 8, do the following:

1. Create a programmed component by dragging it from the edit tab at the top of the element window (a button **Execute** should appear on the screen)
2. From the created component pop-up menu choose **Import file**
3. In the file browser dialog, choose the EXE file to import and click **OK**

The above sequence should make sure that you can execute the EXE file by pressing the **Execute** button. You can change the name of the button by choosing **Button text** on the components pop-up menu. For example by typing *&Calculator* and importing CALC.EXE from Windows, you can allow the user to execute the Windows calculator by choosing the **Calculator** button. Note that in this, a separate copy of the EXE file will be created within the knowledge system and deleting the calculator in Windows will not affect the behavior of the system.

To add a DLL component to an element in SuperMemo 8, act as with EXE components. Note that the DLL must be written specifically for SuperMemo 8, i.e. it cannot be just any DLL. The only difference here is that DLLs cannot be executed independently. Instead, SuperMemo 8 will call specific procedures coded in the DLL such as: *Repetition, Edit,* or *Install.*

To write a DLL that will work with SuperMemo 8, you should implement one or more of the following procedures (all procedures are optional!):

- *Repetition* - called at the moment of executing the programmed component (e.g. by choosing the **Execute** button or calling PLAY in a script). If this procedure is not defined, no action is taken and no error is reported.
- *Edit* - called only at the moment of choosing **Editing mode** on the programmed component pop-up menu and used, for example, to edit parameter files associated with the DLL. If this procedure is not defined, no action is taken and no error is reported. Note that the DLL can create only one parameter file within the knowledge system without calling allocation and registration procedures. This file has the same path and name as the DLL file; only its extension is PAR. All other files created by the DLL within the knowledge system should be properly registered with the system by means of procedures available from KSI.DLL (Knowledge System Interface) described later.
- *Install* - called only at the moment of importing the DLL file into the knowledge system. If this procedure is not defined, no action is taken and no error is reported. This is the best procedure to place a code creating and initializing the parameters file associated with the DLL.

All the above procedures use only one parameter: pointer to a record called *RepDat*. This record can be used to pass parameters to and from the DLL. The definition of *RepDat* is as follows:

```
TRepData=record
        MainWindow:HWnd; (2 bytes)
        TopicWindow:HWnd; (2 bytes)
        CompWindow:HWnd; (2 bytes)
        TopicRect:TRect; (4 bytes)
        ItemNo:integer;   (4 bytes)
        grade:byte; (1 bytes)
        reserved:array[0..1000] of byte;
        end;
```

The interpretation of the fields is as follows:

- MainWindow - handle of the main window in SuperMemo
- TopicWindow - handle of the element window in SuperMemo
- CompWindow - handle of the component window (the one hosting the **Execute** button)
- TopicRect - rectangle coordinates of the element window in SuperMemo
- ItemNo - number of the element hosting the programmed component
- grade - grade given or obtained in the course of a repetition (this is an output parameter used by the *Repetition* procedure)

The specification of TRepData is open so that new fields can be added in the future as needed. In particular all learning parameters and metrics can be obtained from TRepData (see updates to this document in the future). The open specification also means that you, DLL developer, can also postulate new fields that are needed in your implementation. Those fields can be filled out in future specifications.
Most of the fields of TRepData are set by SuperMemo upon executing relevant DLL procedures. The author of the DLL does not have to make use of those fields, nor does he have to set output fields! No errors will result.

Here are some useful procedures available from KSI.DLL that can be used in communicating with SuperMemo 8 while executing a DLL associated with a programmed component:

procedure PlayComponent(ItemNo:longint;CompNo:byte); export;
Used to play or execute the component associated with the element *ItemNo* and component *CompNo* (both can be viewed with the button displaying component number tags on the Edit tab in the element window). The component can be sound, video, script or a programmed component.

procedure GetNewFilename(filename:PChar); export;
Used to obtain new legal *filename* within the knowledge system (without extension!)

procedure RegisterSound(filename,itemname:PChar;var Posit:longint); export;
Used to register a wave or midi file used by the DLL. *Filename* is the filename obtained from *GetNewFilename* with a proper extension added. *Itemname* is any name under which the registered sound will appear in the sound registry. *Posit* is the newly allocated registry position for the registered sound file. Use *Posit* with *GetSoundFilenameFromPosition* if you want to make sure you use the updated sound file registered at that position (the user might wish to change the sound registered by your DLL). The *var* parameters indicates that *Posit* is an output parameter here

procedure GetSoundFilenameFromPosition(Posit:longint;filename:PChar); export;
Used to obtain the name of the file registered under a given position in the sound registry. *Posit* is the position. *Filename* is the returned filename

procedure RegisterImage(filename,itemname:PChar;var Posit:longint); export;
Analogous to *RegisterSound.*

procedure GetImageFilenameFromPosition(Posit:longint;filename:PChar); export;
Analogous to *RegisterImage.*

For example, to add a new sound file to the knowledge system do the following:

1. Call *GetNewFilename* from KSI.DLL to obtain a new legal filename within the knowledge system (the filename does not have an extension).
2. Copy the sound file to the filename obtained in the previous step (do not forget to add an appropriate extension)
3. Register the sound file by calling *RegisterSound*. Provide the new sound file name used in the previous step as *filename*. Use any name as *itemname*, e.g. My sound (this name will later appear in the sound registry).

To write the simplest DLL for SuperMemo 8 that displays a Hello

world dialog, do as follows:

1. Implement and publish the *Repetition(RepDat:pointer)* procedure in your DLL.
2. Display Hello world dialog within the *Repetition* procedure.
3. Ignore *RepDat* parameters.
4. Ignore *Install* and *Edit* procedures, i.e. do not bother writing anything.
5. Compile your DLL and import it to any knowledge system as described earlier.
6. Click the **Execute** button. Hello world should appear on the screen.

Important! For the newest update to the KSI.DLL and this specification, go to the SuperMemo Web Site (http://www.supermemo.com/english/dlls.htm).
If you have any questions about implementing DLL files for SuperMemo 8 write to software@supermemo.com

acquisition rate - speed of learning, usually expressed in items memorized per year per minute. For example, if 20 minutes a day yield 10,000 items a year, the acquisition rate is 500 items/year/min. In SuperMemo, the acquisition rate may vary from 100-1000 depending on the difficulty of the material, forgetting index and the stage of the process (acquisition rate may substantially decrease in the period of the first year; later it stabilizes asymptotically)

burden - (in the Statistics window) number which estimates the average number of items that have to be repeated daily:

burden = 1/I(1)+1/I(2)+ ... + 1/I(n)

where:

I(n) - current interval of the n-th item

ClipBox - a temporary cache for elements or components. It works in a similar way like the Windows clipboard. First you place an object in a ClipBox and then you can fetch it from the ClipBox as many times as you wish. Quitting the program empties all ClipBoxes. There are three kinds of ClipBoxes placed on the toolbar: Element ClipBox (used in copying elements), Component ClipBox (used in copying components) and Hyperlink ClipBox (used in creating hyperlinks). See Using ClipBoxes.

component - single object placed within the knowledge element display area. A component may be of the following types: text, Spell-Pad, sound, image, video, program script, external binary object (DLL or EXE) or shape (rectangle, ellipse, line, etc.).

database - in earlier versions of SuperMemo, a database was a collection of question-answer pairs used in learning. In SuperMemo 8, a term *knowledge system* is used instead of the term *database.* This is to amplify the fact that in SuperMemo 8, (1) knowledge elements may have multimedia character, (2) knowledge is structured by means of knowledge hierarchy, and (3) individual knowledge elements are connected by means of hyperlinks.

element - single page-like part of the knowledge system associated with a single knowledge hierarchy node. An element can have the form of a topic presenting synthetic knowledge or an item that usually has the stimulus-response character and takes part in repetitions (most often, items are formulated as questions and answers)

element window - window in which the content of knowledge elements is displayed. Its toolbar contains buttons **Contents**, **Search**, **<, >**, **Next** and **History**.

final drill - last stage of the learning process passed on a given day, in which only elements that scored a grade less than **Good** (4) are repeated as long as they continue scoring less than **Good**

forgetting index - proportion of items that are not remembered at repetitions. Forgetting index can be programmed to fall between 3% and 20%. This way, the speed vs. accuracy trade-off in learning can be controlled by the student

hierarchy - the tree structure in which particular elements of the knowledge system are organized. The knowledge hierarchy is presented in the Contents window. The root element (the first element) in the hierarchy is called the master node. Particular nodes of the tree can hold several thousand children each, but for performance reasons it is recommended not to keep more than a hundred children elements in a node

hierarchy node - one element in the knowledge hierarchy tree presented in the Contents window and corresponding to a single topic or item of the knowledge system. Adding new nodes is accomplished by **Add**, **Sibling** and **Insert** in the hierarchy, or by **Edit : Add a new item** on the main menu. In the latter case, not only a node is added but also the elements contents is filled out with the default append template

hyperlink - connection established between a component and an element different than the element owning the component. Hyperlinks make it possible to navigate in knowledge hyperspace by clicking individual components rather than using navigation toolbar commands. You can set a hyperlink by using **Hyperlink** on component pop-up menu. You can also Ctrl+drag a component to Hyperlink ClipBox and drop it on the hyperlinked element.

item - knowledge element that has a stimulus-response structure and is used in repetitions (most often, the stimulus-response pair is just a question-answer pair). Usually, items are associated with their owning topic. The simplicity and univocality of items is one of the fundamental preconditions for success with SuperMemo

knowledge system - collection of pages called knowledge elements organized into knowledge hierarchy and connected by hyperlinks. A knowledge systems of a given name is stored in a directory that bears the same name, and all its important statistical and learning data are stored in a file with the extension KNO

mnemonic hyperspace - extension of Tony Buzans concept of mind maps by application of hyperlinks between the mind map components and mind map editability. Mind maps are considered an excellent form of representing knowledge for the purpose of learning. SuperMemo makes it possible to create simple mind maps that contain multimedia objects

outstanding element - element that is awaiting repetition that satisfies the criteria of optimum interval. Each element in the learning process has its next repetition date determined by the repetition spacing algorithm. On the given date and later on, the element is considered outstanding

pending queue - queue of elements that are awaiting memorization. Pending queue is modified upon using learning options, **Commit** (removing the current element from the pending queue), and **Reset** (adding a memorized element back to the end of the pending queue). Pending queue determines the order of learned elements in the linear learning mode

programmed component - a component that can execute a binary file written in any language (DLL or EXE). Programmed components make it possible to extend the range of applications of SuperMemo into any imaginable area

repetition spacing - technique of optimizing the learning process by computing optimum intervals that should separate repetitions of individual pieces of knowledge. SuperMemo pioneered application of repetition spacing in learning (first implemented as software in 1987) and has gained world-wide acclaim through its impact on increasing the productivity of learning. SuperMemo 8, the hypermedia-enabled version of SuperMemo, uses the most recent unpublished repetition spacing algorithm

registry - a sorted collection of named objects that are used in creating knowledge elements in SuperMemo. Objects stored in registries may have the form of sound, video, image, executable program, DLL, font, text, etc. You can link particular objects with components in knowledge elements by means of **Link registry member** on component pop-up menu. Upon choosing the appropriate object name in the registry, choose **Link** and the object will appear within the selected component

SuperMemo - speed-learning technology developed by Polish biologist Dr P.A.Wozniak, and promoted worldwide by SuperMemo World. SuperMemo makes it possible to minimize time necessary to retain once learned material in the students memory. SuperMemo uses the model of user memory to compute the optimum spacing of repetitions. SuperMemo 8 uses the newest SuperMemo algorithm called Algorithm SM-8.

SuperMemo 8 for Windows - educational hypermedia authoring tool based on the SuperMemo method. All knowledge systems developed with SuperMemo 8 are shipped with the program itself. You can always activate the full SuperMemo 8 environment in knowledge systems in which it is hidden. For that purpose, press *Ctrl+Alt+F12.*
(note that before releasing its first authoring version, SuperMemo 8 was codenamed ***Genius***).

SuperMemo World - software company founded in July 1991 by a computer scientist K. Biedalak (MSc) and the author of the SuperMemo method P.A. Wozniak (Ph.D.). SuperMemo World devotes all its resources to the promotion of the SuperMemo method of fast learning. Its product SuperMemo is one of the most successful Windows applications developed in Poland. SuperMemo software line has collected innumerous awards at software competitions in Poland and abroad and has twice topped the ranking of most popular Polish software by PC World Komputer (1994 and 1995).

SuperMemo World
ul.R.Maya 1, 61-371 Poznan
Poland
tel.: 48 61 764066
tel/fax: 48 61 764073
e-mail: **info@SuperMemo.com**
Web**: www.SuperMemo.com**

topic - knowledge element that presents the synthetic review of a selected piece of knowledge. Optimally, topics should be used only in selective learning mode for knowledge presentation and should be dismissed in the learning process (that is the default behavior of SuperMemo). All children of the topic are considered elements that decompose the synthetic knowledge of the topic into stimulus-response pairs used in the learning process *(in the manner known from SuperMemo)*. In the automatic learning mode (evoked with *Ctrl+L*), topics are presented before memorizing relevant items and are dismissed immediately upon the first presentation.

Love Molie forever, Jan 17, 1997