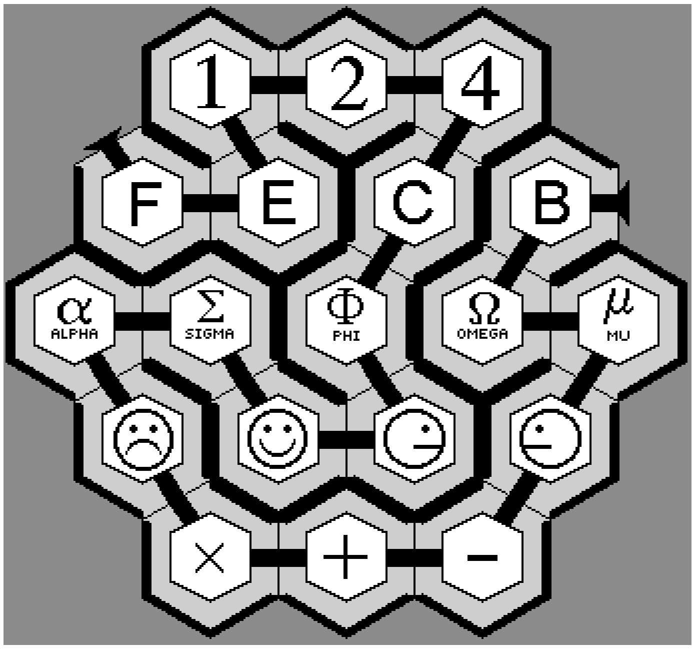
All About Clues

All images in this section of the help use the "printing" image set, to make the discussion a little simpler.

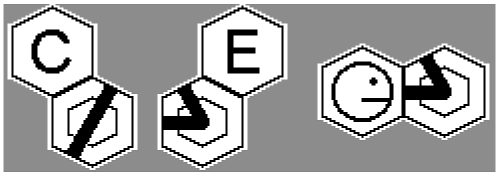
NOTE: If you have the "Clue explanations" option checked (enabled) under the View menu, a brief explanation of each clue will appear in the window beneath the puzzle as you point at clues with the cursor.   That is the easiest way to learn (and remember) what the clues mean.   However, a brief scan through this section may help you learn more quickly some of the deductions that can be made using the clues.

Here is an example medium-sized game board (completely solved) that will be used in explaining the rest of the clues:



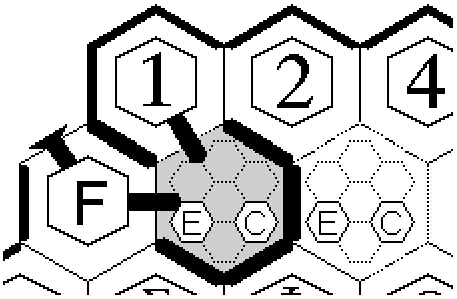
There are seven types of clues (ignoring *givens* on the game board), and they are arranged in four groups.   One of the types of clues, ***IS_PATH***, can be displayed in all four groups as space permits and requires.   All other clue types are restricted to appearing in only one or two of the clue groups.

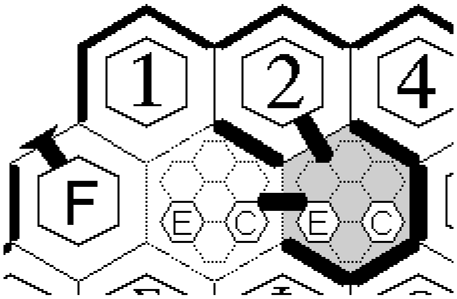
The ***IS_PATH*** clue type looks something like these (depending upon which clue group they're displayed in and what the actual path is that is being presented):



You will notice that in all three cases, the clue consists of an image (C, E, or RightFace in these cases) and a second hexagon with a line (or path) running through it. This is telling you that in whatever cell the image is located, that's the way the path runs through the cell. You can make a number of deductions using these ***IS_PATH*** clues, such as:

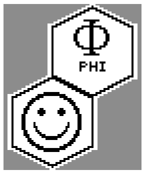
1) If the image is known, then you can mark both paths and all four walls in the cell containing the image.
2) If the image is a possibility in a cell, but the already known walls and/or paths in that cell don't match what's shown in the clue, then the image can be removed as a possibility from that cell.
3) If the image is a possibility in a cell, but placing it there would (using this clue) cause the path to do something illegal (like connecting the two ends of the path before all cells have been included, or creating a "loop" in the path, etc), then the image can be removed as a possibility there.
4) If the image is only possible in two adjacent cells, think about what would happen to all of the sides in the "neighborhood" in the two cases (if the image were placed in the one cell or the other). If there's a side that is forced to be a Wall or a Path in *both* cases, then it can be marked as such since it will be that no matter where the image is located. Did I lose you there? Well, look at the two following pictures. The letter E is possible in two locations next to each other (as is the letter C). Using the E IS_PATH clue shown above, if the E is in the left-most of the two cells, then the paths and walls will have to look like those shown in the left picture. If the E is in the right-most of the two cells, the paths and walls will have to look like those shown in the right picture. Notice that in BOTH cases, REGARDLESS of which cell the E is in, there is a WALL on the southwest side of the 2. Therefore, without even knowing where the E really is, we can go ahead and put a WALL on the southwest side of the 2.





There are 15 different possible paths through a cell, so there are 15 different ***IS_PATH*** clues. 11 out of those 15 enable you to deduce a wall or a path in this manner when the ***IS_PATH*** image can ONLY exist at two adjacent locations.

The ***SAME_RIGHT_DIAGONAL*** clues are displayed in the top-left clue group. They all look something like this:

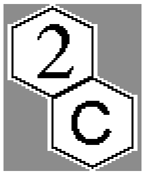


In the example above, 4, C, PHI, Smiley, and X are all in the same *right-leaning-diagonal*. 2, E, SIGMA, and SadFace are in the same *right-leaning-diagonal.* 1, F, and ALPHA are in the same *right-leaning-diagonal.* Get the idea?

Here are *some* of the deductions that you can make using right-leaning clues:

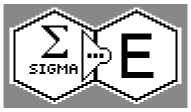
1) If one image is known and the other is not, you can deduce where the other one must be, as it's row is always known, and it must be in the same right-diagonal as the known image.
2) If one image is a possibility in a right-diagonal, but the other image is not, then the first image can be removed as a possibility from that right-diagonal.

A very similar clue type, ***SAME_LEFT_DIAGONAL***, is displayed in the top-right clue group. They all look something like this:



In the example, 2, C, OMEGA, and LeftFace are all in the same *left-leaning-diagonal.* 1, E, PHI, RightFace, and - (minus) are all in the same *left-leaning-diagonal.* You get the picture right? The same deductions that can be made with ***SAME_RIGHT_DIAGONAL*** clues can also be made with ***SAME_LEFT_DIAGONAL*** clues.

The ***IS_LEFT_OF*** clue type is almost always displayed in the lower-left clue group, although in *very* unusual circumstances it might appear in the lower-right clue group also. It looks something like:



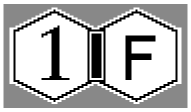
The ellipsis (three dots) in the right pointing arrow is super-imposed on the two images to indicate the relationship between the two images. In this case, it means that the

SIGMA is to the LEFT of the E. In the example at the top of this section, SIGMA is to the left of ALL of the images EXCEPT for 1, F, ALPHA, SadFace, and X. Note particularly that the 1 and the X are exactly above and below the SIGMA, and hence the SIGMA is NOT to the left of them (nor are they to the left of the SIGMA!)

Deductions that can be made from the ***IS_LEFT_OF*** clue type include:

1) If there are any cells containing the second image that are further left than (or directly above or below) the left-most possibility of the first image, then those images can be removed as possibilities.
2) If there are any cells containing the first image that are further right than (or directly above or below) the right-most possiblity of the second image, then those images can be removed as possibilities.

The ***NEXT_TO_WITH_WALL_BETWEEN*** clue is displayed in both the lower-right and lower-left clue groups. It looks something like:

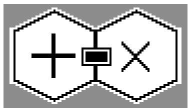


This clue tells us that the two images are directly adjacent to each other. Notice the vertical "wall" between the two images which indicates that the common side between the two cells containing these two images is a WALL, not a PATH. In the example, 1 is next to 2, E, and F. F is next to 1, E, SIGMA, and ALPHA. In case this still isn't quite clear to you, a further example would be that PHI is next to E, C, OMEGA, RightFace, Smiley, and SIGMA. Of course, since an image has to be in a particular row, if you know where one of the images is, then the other has to be in one of at most two locations.

The ***NEXT_TO_WITH_PATH_BETWEEN*** clue is just like the ***NEXT_TO_WITH_WALL_BETWEEN*** clue except that one is vanilla and the other is chocolate.

(Just checking to see if you're really reading or not... :-)

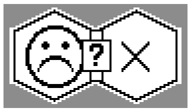
They're just the same except, of course, that one specifies a WALL is between the adjacent images and the other specifies that a PATH is between them. The ***NEXT_TO_WITH_PATH_BETWEEN*** looks something like:



When making deductions with these two clues, you have to keep in mind the path/wall between them as well as the fact that they must be adjacent to each other. Some deductions that can be made with these two clue types are:

1) If one image is known, the other can only be in one of the two spaces adjacent to the one that is known. So, all possibilities of the other one can be removed from any cells that are not adjacent to the known one.
2) If one image is possible in a certain cell, but both sides are wrong between it and the adjacent cells containing the other image (there's a PATH between the two images, but both sides are WALLS, or vice-versa), then the one image can't be in that cell and can be removed as a possibility.
3) If one image is known and the other image is only possible in one of the adjacent cells, then the other image is also known.
4) If both images are known, then the state of the side between them (WALL or PATH) is also known.

The final clue type is the ***NEXT_TO*** clue. It tells us that the two images are immediately adjacent to each other, but it tells us NOTHING about whether there is a WALL or a PATH between them. It looks something like:



The deductions that can be made using this clue type are:

1) If one image is known, the other can only be in one of the two spaces adjacent to the one that is known. So, possibilities of the other one can be removed from any cells that are not adjacent to the known one.
2) If one image is known and the other image is only possible in one of the adjacent cells, then the other image is also known.
3) If one image is possible in a cell, but the other image is not possible in either of the adjacent cells, then the one image is not possible there either, and it can be removed as a possibility from that cell.

__NOTE__: There are some fairly complex deductions that can be made in certain situations. This help file can't possibly list every possible deduction that can be made under all circumstances, so use that gray matter resting between your ears and see what you can come up with! As an example of a slightly more arcane deduction: suppose that you have one of the three "next-to" types of clues, and both images are from the same row, such as C and F. Now, let's further suppose that C and F are ONLY possible in 2nd, 3rd, and 4th cells of their row (it can be ANY three cells that are "sequential", or all together in the row). If there are ANY other possibilities in the middle cell of the three (the 3rd cell in our example) besides C and F, then those other possiblities can ALL be removed from the middle cell. Why? Because no matter WHERE the C or F is really located, EITHER the C or the F MUST be in the middle cell (since they're NEXT TO each other). Therefore, nothing else EXCEPT for C or F can be in the middle cell. This is a tough deduction to spot until you get use to looking for it. The thing to do is to always check your clues right at the start of the puzzle for any "next to" type clues where the two images are from the same row. If you have any, then tie a string around your finger so that you remember to keep watching the possiblities for those two images, and if they ever get whittled down to just three adjoining cells, apply this deduction!

Another similar one that can be hard to spot: if there are two cells in the same row that have the SAME two images as the only possibilities, then those two images CAN'T be in any OTHER cells in the row, because one of them has to be in one of the two cells, and that leaves the other one as the only possibility in the second cell. So, neither of them can be in any of the other cells in the row. The same thing can be applied to the same three images in three cells, but that happens only very rarely.

Using Clues Together

The computer never uses two or more clues together to make deductions when generating the puzzles or finding a hint. It always examines the clues one at a time. However, that doesn't keep you from using the clues together, in order to make deductions that the game would never make.

For example, if you had one clue that told you that the SIGMA was next to F and another clue that told you that SIGMA was next to E, you could combine these clues and deduce that SIGMA could NOT be on either end of the its row (because then it would only be able to have one of the letters next to it). You could also check to see if SIGMA was marked as a possibility at a cell where E and/or F were only possible in ONE of the adjacent cells. If you found any like that, then SIGMA could be removed as a possiblity from that cell.

There are quite a number of deductions of this sort that can be made by combining two or more clues. But be careful: it can get pretty subtle, and mistakes are easy to make. :-)

The OTHER CLUES Bin

Think of the clues that are displayed on the screen as being stored in a "bin" (or box). There is a second bin (called the Other Clues bin) which is never displayed, but which can act as a place to store clues that you think you're done with. As you play the game, as the puzzle gets solved, some clues will no longer be of any use (all of their information will have already been "placed" in the solution). These clues are just in the way, and you can remove them from visibility by sending them to the Other Clues bin. You do this by pointing the mouse cursor at the desired (or UNdesired) clue, and clicking the right mouse button. The clue will disappear!

However, occasionally, even the greatest of intellects will make a mistake (I know, I know, it's hard to believe!) and a clue may get sent to the Other Clues bin before all of its information has really been used. You can swap the clues between the two bins (send all of the current clues to the Other Clues bin and bring all of the clues from the Other Clues bin to the current clues bin). You do this using the *View-Other Clues* menu item or by clicking on the 🔍 Other Clues button on the toolbar. At this point, you can right-click on clues to send them back to the OTHER group (which is now residing in the Other Clues bin). Subsequent clicks on the Other Clues button will swap the clues back again.

The game always starts out with ALL of the clues in the current clue bin (showing on the display). The Other Clues bin will be empty unless YOU send some clues there. So don't go looking in the Other Clues bin for help if you get stuck on a puzzle unless YOU sent some clues there!

NOTE: When you ask the game for a Hint, it ONLY uses the clues that are visible to YOU, so it ignores the ones that are in the Other Clues bin. It is possible (by sending clues to the Other Clues bin before they've been completely "used up") for you to get the clues split between the two bins in such a way that the computer can't find a Hint, even when you swap the clue bins. It will give you a message, "No hint available. Try other clue bin." or some such nonsense. So, you'll swap to other clues, hit Hint again, and it will tell you the same thing. If this happens, it means that the game is at the point at which a *What If* deduction needs to be made, but you've got the clues split between the bins in such a way that the game can't make the *What If* deduction that it needs to make. The thing to do in this situation is to use right-click to move ALL of the clues to the Other Clues bin, then swap the bins so that ALL of the clues are visible. Then ask for another Hint.

Best Scores & Game Logging

Best Scores

There are two kinds of people in the world: those who think there are two kinds of people in the world, and those who don't.

No, wait a minute...wrong speech.

There are two kinds of people who play games, those who have to have a score to try to beat, and those who just play for the sake of playing. The "Best scores" table is maintained for the the first group. It keeps track of up to 25 scores for each of Small, Medium, and Large puzzle sizes. It ranks these scores (least significant items first) according to:

How much TIME it took to solve the puzzle.
Whether AUTO-COMPLETE IMAGES was used.
Whether AUTO-COMPLETE SIDES was used.
How many HINTS were used.
What HANDICAP was used.

Notice that the TIME is the least important factor. This is because all of the other factors can GREATLY affect the amount of time required to solve a puzzle. Therefore, a low time only counts as better if the other factors were also at a "low level."

In the "Best scores" display, the above factors are displayed left-to-right in increasing order of significance, least significant on the left. As many of the scores as possible are displayed, given the available room (height) in the game window. Rarely will all 25 scores be visible for each puzzle size, and there is no way to scroll them. The font used in the Best scores display is controlled by the size of the image set being used (for a number of reasons, the font is kept "proportional" in size to the images being used). Therefore, you'll be able to display the most scores by maximizing the window size, and then selecting a 640 resolution image set.

The Best scores table can be viewed using the *View-Best Scores* menu item. The table can be cleared of scores using the *View-Clear Best Scores* menu item.

Game Logging

If you'd rather view the scores in a different fashion from how they're presented in the Best Scores display (or want more complex statistics), you can do that outside of the game using a spreadsheet program or some other data analysis tool. As you play the game, every puzzle that you solve is logged (recorded) in an ASCII file called HOTELLOG.CSV. This file is a standard "comma-delimited values" formatted file, with each line containing information about one solved puzzle. If you solve the same puzzle multiple times, there will be one line in the file for EACH time you solve the puzzle. In other words, the file contains a complete history of EVERY puzzle solved. (You can simply delete the file at any time to restart your statistics from scratch. The program will recreate the file and start logging into it again with the next puzzle you solve.)

The contents of each line of the HOTELLOG.CSV file are (in order):

- The date and time-of-day when the puzzle was solved.
- S, M, or L (the size of the puzzle).
- The puzzle number.
- The hours, minutes, and seconds required to solve the puzzle.
- The handicap used.
- The number of hints used.
- Whether Auto-complete sides was used (0 for no, 1 for yes).
- Whether Auto-complete images was used (0 for no, 1 for yes).
- The player's name who solved it.

If you load this file into a spreadsheet program, then you can sort and analyze it till you're blue in the face...er...sick and tired of it...er..ummm...till your heart's content. Yeah, that's it.

Contents

Welcome to Honeycomb Hotel

The licensed version of this game costs $19.95 + $2 shipping (within North America, $4 shipping elsewhere). It contains over 65,000 each of small, medium, and large puzzles.

Contact information for orders and support:

Everett Kaser Software
Honeycomb Hotel
PO Box 403
Albany, OR 97321-0117

Phone:
(541) 928-5259

email:
everett@kaser.com

Web page:
http://www.kaser.com

Credits

I would like to thank everyone who helped during the development of this game, testing and making suggestions. A lot of folks sent comments and suggestions at one time or another. The following people were especially helpful with comments, bug reports, and suggestions over most of the development cycle, and they deserve special recognition! Thank you, each and every one!!! (Arranged in reverse-alphabetical order by first name, just to be contrary to what is normally done to us by large, faceless organizations and public schools... :-)

Yaacov Yoseph Weiss
Ulrich Meierfrankenfeld
Sunny Kiser
Steve Blendermann
Sandra Whiting
Richard Schiveley
Reginald User
Philip Bartol
Phil Conrad
Paul Baum
Natalie Ford
Margaret Greer
Leah Forman
Kris Pixton
Kirsty Darbyshire
Joel P Bion
Diane Graff
Deborah Lipp
Clint Chaplin
Carol Gortat
Brandon Freels
Bill Landau
Barbara De Roes
Anne-Marie Auriault
Anne Lombardo

Everything that you like about Honeycomb Hotel is due to my diligence and creativity. Anything that you dislike or have problems with is solely due to the interference and knuckle-headedness of the folks above. Woof-woof. :-)

How To Play The Game

"Exactly, Watson. Here is the fruit of my leisured ease, the magnum opus of my latter years!" He picked up the volume from the table and read out the whole title, Practical Handbook of Bee Culture, with some Observations upon the Segregation of the Queen. "Alone I did it. Behold the fruit of pensive nights and laborious days, when I watched the little working gangs as once I watched the criminal world of London."

A. Conan Doyle: His Last Bow

Harold the Honey Bee is up at dawn every morning, rousting the other bees from their comfortable quarters at the Honeycomb Hotel. Harold tap-dances from room to room, visiting every room but once (time is critical in a bee's life, you know). Bees (being different in many interesting ways from you and me) use various strange symbols upon the doors of their hotel rooms, with each floor having a different theme.

Honeycomb Hotel is a game of logic and deduction. You must determine which symbols are on which doors of the Honeycomb Hotel, and the exact path that Harold takes in his task of awakening the workers. Using the provided clues, you deduce where symbols can and can't be on the doors of the hotel, and which way Harold did (or didn't) go, until all is revealed.

The Basic Rules and Ideas

There are three sizes of puzzles, selected by three buttons on the toolbar (or using the Options menu). Small-sized puzzles have 3 horizontal rows of hexagons, medium-sized puzzles have 5 horizontal rows of hexagons, and large-sized have 7 rows. In all cases, each row contains images of the same *type* (ie, letters, insects, animals, etc). Initially, all of the images that can possibly be located within a particular cell are displayed there as smaller images. As you deduce that an image can't possibly be located within a particular cell, you remove that image from that cell by right-clicking upon it. When you deduce that an image MUST be the one that is located within that cell, then you enlarge it by left-clicking upon it.

At the same time that you're trying to deduce the locations of all of the images, you are also trying to determine the path that runs through ALL of the cells. You're shown the two ends of the path at the beginning of the puzzle (the two ends are always around the perimeter of the honeycomb). The path *always* passes through each cell *once and only once*, and the path *never* crosses itself. This means that for each cell, four sides will be ***WALLS*** and two sides will be ***PATHS.*** Once you've located four sides of a cell that have to be walls, then both of the remaining sides *must* be paths. Once you've located two sides of a cell that have to be paths, then all of the remaining sides *must* be walls. You mark a **PATH** by clicking the left mouse button, and you mark a **WALL** by clicking the right mouse button. A **PATH** may be removed (unmarked) by clicking on it again with the left mouse button, while a **WALL** may be removed by clicking on it again with the right mouse button.

When you've successfully (correctly) located all of the images within the cells AND have successfully (correctly) deduced the complete path (including marking all walls), then you've won the game (solved the puzzle).

What's Where On The Screen

The game window contains two primary things: the clues and the "game board". The game board is in the center, with the clues arranged around it. All of the clues are comprised of two images, and sometimes there is a smaller third image superimposed between the two primary images. To learn everything you could possibly want to know (and probably more) about clues, read All About Clues.

The game board shows all of the cells in the puzzle, along with all of the images that are possible in each cell. The possible images are shown as smaller images. When a cell is *known* to contain a particular image, that image is displayed alone in the cell and larger (the size of the clue images). Some puzzles begin with one or more cells marked with their correct image. These are *givens*, and make the puzzle a little easier. As you get to higher puzzle numbers, the number of *givens* decreases. The number of *givens* will be higher on large puzzles than on medium puzzles, and higher on medium puzzles than on small puzzles. The decrease in the number of *givens* occurs at puzzles 15000, 30000, and 45000.

In addition to the *givens* that occur naturally in the puzzles, the Handicap setting (under the Options menu) can cause some additional cells to be *given.* The game starts with default Handicap settings which you will probably want to decrease as you get better at the game.

WHAT-IF...

All of the first 15000 puzzles can be solved purely through repeated application of the the clues to the puzzle, using reasonably simple, straight-forward logic. However, starting at puzzle 15000, the game sometimes uses more complex "what-if" type of logic in determining where the PATH must run. Starting at puzzle 30000, the program uses this "what-if" logic also to sometimes determine the locations of IMAGES. Starting at puzzle 45000, the program gets really nasty and uses a great deal of "what-if" logic in constructing the puzzles. To learn more about "what-if" logic and how to use it in solving puzzles, read What Is "WHAT IF" and How Is It Used?.

Image Sets and Creating Your Own

The Honeycomb Hotel program can use alternate image sets (which are selected with the *Options-Images...* menu item, and in the *File-Print Options...* menu item). The game includes four "default" image files designed for use at four different screen resolutions: HTL640, HTL800, HTL1024, and HTL1280. These all contain the same images, just at different resolutions (number of pixels, sizes). You can use a graphics editor of your choice to create variations on these files or totally new image sets. Several alternate image sets are also included with the licensed version of the game.

To modify an existing image set or to create an entirely new one, first make a copy of the image set you wish to modify (or one that is the same resolution as the new one you wish to create) under a different file name. Then, use the graphics editor of your choice to edit that copy of the images. You can place whatever graphics you want in the image file, and they will be used by the program (when you select that new image file as your image set). However, the program expects (and requires) that the graphics be located within the file in exact places. That's why you should start with a copy of an existing image set, and just make modifications to it.

NOTE: The program ONLY looks in the directory in which it is installed for image sets (*.BMP files). If you create your own file, you must place it in the HOTEL directory (wherever you installed the game) in order for it to be visible in the *Options-Images...* or *File-Print Options...* dialogs.

Not all of the graphics file is used by the program. Those areas not used are colored solid blue in the four default sets. You can put whatever you want in those areas, as the program makes absolutely no use of them. The rest of the graphics file is divided into these areas:

Small images: these are the top-left corner, and are seven rows of the small images that are used as the "possibilities" on the game board.

Large images: these are the top 2/3 of the bitmap, in the middle. They are the large images used to draw the clues and the "known" images on the game board. A special case is the center image on the bottom row (the seasons row). That center image is the image used in "clue places" where there is no clue. In other words, it's the image used for "empty clues."

Path images: these are just to the right of the Large images, and are the "path images" used in "IS_PATH" types of clues.

Clue types: in the top-right corner of the bitmap are four small graphics, used in the drawing of clues.

Game board backgrounds: on the right side, below the "Clue types" images, are the backgrounds used to draw the game board itself. From the top, they are:

Not completed with possibilities.
Walls completed with possibilities.
Known but walls not completed.
Known and walls completed.
What If image is in this cell.

Win display images: the eight images (bees by default) below the Large images are used by the win display. They are the same size as the Large images.

Clue backgrounds: below the Win images are the two possible "clue backgrounds", white and red by default. The only really important part of these two images is the outline, that portion outside of the black line, which is larger than the Large images, as that is the only part that ever shows.

Walls and paths: on the left side beneath the Small images are all of the various walls and paths. There are two major sets, the top being the normal ones, the bottom (blue) set being used when a wall or path is the subject of a *What If*. The left-most column is used when the path is in a cell where the image is KNOWN. The middle column are the six possible walls. The right column is used when the path is in a cell where the image is NOT known (possibilities). In the top-right corner are the six different possible "path ends."

The color palette in the bottom-right corner (by the blue game board background) is there only for your use and reference, and is not used by the game.

It is HIGHLY recommended that in making modifications, you only change the "internals" of the images, and leave the outlines "as-is." Otherwise, things may not line up properly when drawn on the screen, looking less than desirable.

If you look at the small and large images, you will notice first that the rows are "upside down" from how they're actually used on the screen. The top row (insects in the default images) actually appear on the bottom row of the game board, and the bottom row (seasons) appears on the top row of the game board. On small- and medium-sized puzzles, the game uses the images starting from the top of the bitmap (the insects), so the lower rows in the bitmap (flowers and seasons for medium puzzles, also hives and produce for small puzzles) don't get used.

The second thing that you'll notice is that some of the images are duplicated. This is done to simplify the programming of the game. If you look closely at the way the small images are arranged in the cells on the game board, you will notice that different locations are used for small, medium, and large puzzles (in order to keep them arranged "symmetrically", depending upon whether there are 3, 4, 5, 6, or 7 possibilities or cells in the row). That's the driving reason for this duplication of images in the bitmaps. You don't need to understand a whole lot about the details of WHY, so long as you know that you HAVE to duplicate the images that you change, just as they are in the default image sets. So, there are double images of Grasshopper, Bee, Ant, H, Y, Dog, and Yellow Hive. If you modify any of those, then you'll need top copy/paste that modification into the other location as well. This is true for both the Large and Small images.

Last (but certainly not least!) is the issue of "transparency color". In the default image sets, the "transparency color" in the bitmaps is a very light yellow (the bottom-right-most color in the palette in the bitmaps). This color can actually be ANY color you want, BUT...whatever color you choose, you can NOT use that color as part of any of the images. When computers draw images, they always draw rectangular images. If you want to draw a non-rectangular image (hexagonal, for example), then you have to do so by actually drawing a rectangular area and using a series of masking operations (logical ANDing and ORing) to combine the image you're trying to draw with the image on the screen. In doing so, there has to be some way of knowing what to keep and what to throw away. This is traditionally done with a "transparency color", a color that is thrown away, leaving the original display contents there. Honeycomb Hotel, when it reads in the bitmap graphics, looks at the pixel in the top-left corner of the bitmap (up by the small grasshopper). Whatever color that pixel is, that's the color the program uses as the transparency color. So, it's very important that you maintain this color wherever it is in the bitmap. If you DO change the outlines of any of the objects (such as the Win display bees, or the "Clue Type" images), it is VERY important that you use this "transparency color" to fill in the areas that you don't want drawn. Otherwise you'll end up with "garbage" showing up on the display when you use the image set.

If you do create your own image set and would like to share it with others, please send a copy to me at:
everett@kaser.com
and I'll make it available on my web site.

Keyboard and Mouse

The Mouse

Most of the game is played with the mouse, although a few features are accessed with the keyboard.

With *View-Clue Explanations* turned on (checked, which is the default), as the mouse is passed over various parts of the game window, an explanation of what is behind the mouse cursor will be displayed in the message line at the bottom of the game window. Point at a clue and a one line description of what the clue means will be displayed. Point at things on the game board, and they will be described, as well as which buttons to press on the mouse to accomplish certain actions.

The left mouse button is used to enlarge (make KNOWN) images on the game board, and to place/remove PATHS. The right mouse button is used to remove small images (possibilities) from the game board, to place/remove WALLS, and to send clues to the Other Clues bin. See All About Clues for more on the Other Clues bin. If you left-click on a clue, it is marked with the "bookmark", a red outline. This has no effect upon the game, but is merely a tool for you to keep track of which clue you're working on.

The Keyboard

Honeycomb Hotel is mostly played through manipulation of the mouse. However, there are some keystrokes that can be of use:

+ moves to the next higher puzzle number
- moves to the next lower puzzle number
w turns on and off the "what-if" counter. See What Is "WHAT IF"...
n turns on and off "what-if" needed notification. See What Is "WHAT IF"...
h requests a Hint (same as clicking on the toolbar Hint button).
i one-time auto-complete of IMAGES (when option is turned off).
s one-time auto-complete of SIDES (when option is turned off).
a one-time auto-complete of both IMAGES and SIDES (when options are turned off)
ctl-Z invokes the Undo feature (same as UNDO button on toolbar).
ctl-P invokes the *File-Print Puzzle* feature.
k keep the current "what-if" deductions, but removes the game from the "what-if" state. This is mostly used when you:
1) press the "what-if" button on the toolbar.
2) make a series of deductions, reaching an inconclusive "dead end."
3) save that state in one of the 10 "mark" buffers (see below).
4) release the what-if button (returning to the previous normal game state).
5) try other what-ifs, and realize that the first what-if you tried (in steps 1-2) was correct.
6) restore that previously "marked" what-if state (see below) from step 3.
7) press 'k' to KEEP that series of deductions, since you now know them to be

correct.

0-9 Save (in RAM) the current state of the game to one of 10 "mark" buffers. These states are NOT saved, when you do a *File-Save Game*, they are temporary "working" buffers only.

shift 0-9 Restore game to state that was saved in one of 10 "mark" buffers. (You must use the 0-9 keys above the normal alphabet keys, NOT the number pad.)

For more details on K, 0-9, and shift 0-9, see What Is "WHAT IF"...

Printing Puzzles

Honeycomb Hotel has been designed primarily to be played on a computer screen. However, thought was also put into playing the games on paper using two colored pencils (or just one if you're desperate!)

The *File* menu has *Print puzzle*, *Print answer*, and *Print options* entries. These should be fairly self-explanatory, but I'll give you a few more details.

To make the puzzles as easy as possible to solve on paper, an alternate image set is used (by default) which consists of letters, numbers, and other simple symbols, and it is all black and white. This makes it far easier to "cross things out" with your pencil and to SEE what you've crossed out and what you haven't. You can select the image set YOU wish to be used during the printing process by clicking on *File-Print Options* and selecting the image set of your choice from the list.

The puzzles can be printed in Portrait or Landscape mode (use the SETUP button in the print dialog to select your preference). When being printed, the puzzle is always stretched PROPORTIONALLY to fit the available space. Because the puzzles are much wider than they are tall, you will get the largest printout in Landscape mode. However, printed puzzles in Portrait mode are quite acceptable (as long as you have young eyes... :-). The puzzles are stretched as far as they can be, given the limits of the printable area on your printer. If you don't want the puzzle stretched quite so far, you can control this by setting margins in the *File-Print Options* dialog. Enter values into these fields (decimal fractional values are accepted). These measurements are in inches (sorry everyone else, someday the US will maybe join the rest of the world in using the metric system!)

To solve the puzzles on paper, I recommend hunting down one (or more :-) of those colored pencils that is RED on one end and BLUE on the other. That way you can just flip the pencil over in your fingers, using RED to cross out impossible images and to mark WALLS, and using BLUE to mark KNOWN images and to mark PATHS. It works quite well!

There's one other option in the *File-Print Options* dialog, a check box which causes a box to be drawn around the puzzle when it's printed. You may like that, you may not. That's why it's an option.

The answer (solution) to the puzzle is, of course, printed using the *File-PrintAnswer* menu item.

Remember, all of the puzzles in this game are copyrighted, and are for your own use only. You may not publish or redistribute these puzzles without written permission from the copyright holder. (But you wouldn't even THINK about doing that, would you??? :-)

Quick Start

This section is just a quick overview of the game. For further details, read:

How To Play The Game

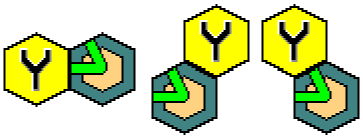
All About Clues

For help on many other subjects, select from the list available in the Contents.

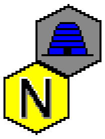
Honeycomb Hotel is a game of logic puzzles. When first learning the game, you may wish to start on the Small puzzles. There are Small, Medium, and Large sized puzzles, selected by clicking on one of the buttons on the toolbar.

The goal is to figure out which images are located where in the cells of the game board, and to figure out how the path runs through the honeycomb. Each row of the honeycomb contains images of the same type (seasons, flowers, insects, etc). The path has two ends (always at the edge of the honeycomb). It runs from one end, through EVERY cell of the honeycomb ONCE, never crosses itself, and finishes at the other end. This means that the six sides of each cell of the honeycomb will consist of exactly 2 paths and 4 walls.

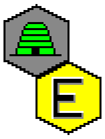
The game board (in the center of the display) is surrounded by the clues. There are seven different types of clues:



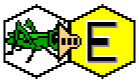
These are all the same clue: wherever Y is, the path is as shown.



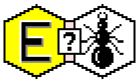
This means that the Blue Hive and N are in the same right-leaning diagonal.



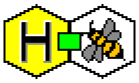
This means that the Green Hive and E are in the same left-leaning diagonal.



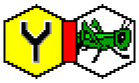
This means that the Grasshopper is further LEFT than the E.



This means that E and the Ant are in adjacent cells (the path or wall between them is unknown, not specified).



This means that H and the Bee are in adjacent cells and have a PATH between them.



This means that the Y and Grasshoper are in adjacent cells and have a WALL between them.

NOTE: None of the last three clues specify ANYTHING about the relative positions of the two images (left-to-right). For example, the Y could be on the left side of the Grasshopper OR on the right.

GAME CONTROLS

Use the LEFT mouse button to:

1) Enlarge (mark as KNOWN) images on the game board when you KNOW where they are located.
2) Place a PATH between cells on the game board when you KNOW a path is located there.
3) Remove a PATH that was previously placed on the game board.
4) Place a "bookmark" (red outline) around a clue, to remember where you were working.

Use the RIGHT mouse button to:

1) Remove small images from the game board when you KNOW that they CAN'T be located in a cell.
2) Place a WALL between cells on the game board when you KNOW a wall is located there.
3) Remove a WALL that was previously place on the game board.
4) Send clues to the Other Clues bin when you're done with them, when you've "used up" all of the information they contain. BEWARE OF SENDING CLUES TO THE OTHER CLUES BIN *TOO SOON!* See All About Clues - The Other Clues Bin.

Placing the mouse cursor over a clue will cause a one-line explanation of that clue to appear in the message box across the bottom of the window. Placing the mouse cursor over the game board reminds you of what the mouse buttons do at that point.

The Menu, The Toolbar, & Game Options

Honeycomb Hotel has quite a few options available in order to make the game as customizable as possible, to best fit your tastes. These options are controlled by menu items, buttons on the toolbar, and a few keystrokes.

Most of the options are available through the menu system and the toolbar. The only feature that is ONLY available on the toolbar is changing the puzzle number (by more than plus or minus 1). To select a specific puzzle, first select the puzzle size (Small, Medium, or Large), then place the mouse cursor in the puzzle number window on the toolbar between the NextPuzzle and PreviousPuzzle buttons (◀ 45000 ▶) and click the left mouse button. This is a normal Windows "edit box", and you can change the number to any value between 0 and 65535, then press the ENTER key. The game will immediately jump to the that puzzle number. Some of the more advanced puzzles may take a second to generate, and the mouse cursor will change to an hourglass during that time.

The other options available are:

FILE

Save game - saves the current state of the game to a disk file. This file is NOT specific to a player. If Player#1 saves game #15, then Player#2 saves game #15, Player#2's saved game will overwrite Player#1's saved game. Player beware. The saved games are stored in files whose names are in the form "Xnnnnn.SAV", where X is one of 'S', 'M', or 'L' depending upon whether the game is a Small, Medium, or Large game. The "nnnnn" is the puzzle number. The file extension is always .SAV.

Load game - loads a previously saved game. Loading the game does not remove the saved game from disk. However, there is an option under the Game menu that (by default) causes a saved game file to be automatically deleted when that game is won.

Print puzzle - prints the current puzzle in its current state. The images used to print the puzzle may not (and by default won't) match those used on the screen. By default the game uses an alternate image set (PRT1280) to print puzzles. The reason is that the most likely reason for printing a puzzle is to play it on paper with a pencil (when you're away from your computer). To do so, it is far easier if there is as much "white space" in the printed puzzle as possible, making it easier to cross out images and mark in walls and paths. The colored images (or gray scale on a black and white printer) make this more difficult. However, if you DO want the same images used on the printout as on the screen, the image set used to print the puzzle can be selected in the *File-Print Options* dialog. The puzzles can be printed in either portrait or landscape mode (using the SETUP button in the Print dialog), and you can control the margins and whether an outlining box is drawn by selecting those options in the *File-Print Options* dialog. The puzzle is automatically "stretched" to fill the available space on the paper within the specified margins. See Printing Puzzles for more information.

Print answer - prints the answer (solution) to the current puzzle.

Print options - in this dialog you can select the image set to be used in printing the puzzles, the margins within which the puzzle will be printed, and whether or not a box will be drawn around the printed puzzle.

Print order form - prints an order form for Honeycomb Hotel and other software available from Everett Kaser Software.

Make shareware copy - makes a copy of the game on diskette that you can give to friends. Remember, ONLY the shareware version may be given to others, the licensed version is copyrighted material and may not be redistributed. (Thanks for helping me to earn a living! :-)

Exit - quits the game (but why would you???)

GAME

Restart - starts the game over. Same as the ⏮ button on the toolbar.

Next puzzle - moves the game to the next higher puzzle number. Same as the ▶ button on the toolbar, and also activated by the '+' key.

Previous puzzle - moves the game to the next lower puzzle number. Same as the ◀ button on the toolbar, and also activated by the '-' key.

Hint - causes the program to look for a possible deduction and explain it to you. Same as the ? button on the toolbar. Also can be activated by pressing the 'H' key. Each hint adds 1 to the Hint Count, which lowers the possible placement of this game on the "best scores" list.

Undo - causes the program to undo the previous move on the game board, or movement of clues to the Other Clues bin. Same as the ↶ button on the toolbar. Also can be activated by pressing CTL-Z.

What if - causes the program to enter *What If* mode. Same as the ? button on the toolbar. See What Is WHAT IF and How Is It Used.

Pause game - pauses the game (halts the timer and hides the game display). Same as the P button on the toolbar. Click again on the pause button to resume playing the game.

Auto-delete saved games - when enabled (checked), saved games are automatically deleted when that game number is won. See *File-Save Game* above.

Bypass title screen - when checked, causes the title/copyright screen to not be displayed when the program is started. This option is disabled in the shareware version of the game.

Default window positions - causes the game window to be reset to its default position and size (for your computer's screen resolution, NOT for the size of the image set).

VIEW

Other clues - swaps the clues in the current (visible) clue bin with the clues in the Other Clues bin. Same as the 🔍 button on the toolbar. See All About Clues - The Other Clues Bin for more information.

Best scores - causes the "best scores" table to be displayed. The game is resumed with any click of the mouse or key press. See Best Scores.

Clear best scores - clears the Best Scores display of entries.

Timer - The elapsed time since the start of the puzzle is normally displayed on the toolbar to the right of the buttons. This display can be supressed by disabling (unchecking) this menu item.

Clue explanations - By default, as you pass the mouse cursor over clues in the game window and over the playing board, a one line explanation of that clue or game board area is shown across the bottom of the window. This explanation can be suppressed (which you will probably want to do once you get familiar with the game and the clues) by disabling (unchecking) this menu item.

Toolbar tips - By default, if you place the mouse cursor on a toolbar button (but don't click a button) and wait a second, a small "pop-up" tip will appear telling you what that button does. These can be annoying after a while, so this menu item enables you to disable them (or disables you to enable them...or something like that).

OPTIONS

Select player - brings up a dialog which lists all players, and allows you to select which one you want. A separate list of configurations and options are maintained by the program for each player. Selecting a different player automatically loads that player's options.

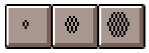
Add new player - Add a new player to the game. (At least, I THINK that's what it does...)

Rename current player - Let's you change the current player's name while maintaining all of that player's options and settings. If you're tired of being Bob, you can now be Dandelion or Grapenuts or...I'll stop now.

Delete current player - removes the current player from the player list and deletes that players setup and configuration. So, to delete a player, FIRST you must select that player, then use this menu item.

Small (3-wide) puzzles

Medium (5-wide) puzzles

Large (7-wide) puzzles - These three select the three different sized puzzle sets. Same as the  buttons on the toolbar. A separate Handicap setting and a separate "current puzzle number" are maintained for each of the three puzzle sizes, for each of the players.

Images - brings up the image set selection dialog, from which you can select which image set you wish to use while playing the game. The images can NOT be "stretched" within the game window. They are displayed at their natural size. The HTL640 image set will be full-screen when your display is set to 640 resolution, but will be smaller and smaller on the screen the higher your screen resolution. It is recommended that you use the image set that is designed for your screen resolution (the game picks the best, largest one by default when you first run the game). If you choose to switch to a smaller one later on, that's entirely between you and your mouse. I don't want to know. A few alternate image sets are provided with the game, some of which exist in multiple resolutions, some only in one resolution. Your mileage may vary. By default the images that are used when printing puzzles (*File-Print*

Puzzle) are different from the ones used on the screen, and are selected in the *File-Print Options* dialog.

Background colors - The game window background color can be changed in this dialog. There are two colors used by the game, the "normal" background color, and the *What If* background color. By default, these are a dark greenish-gray, and blue. The background turns blue when you enter *What If* mode to help you remember that you're in that mode. When changing colors in this dialog, it is recommended that you first find a preferred *What If* color, by using the left-mouse button (ie, changing the "normal" background color), as that background color is updated on-the-fly as you change them in the dialog. That way, you can SEE what it's going to look like in the window behind the dialog. Once you've settled on your preferred *What If* background color, then RIGHT click on that color to set it properly, THEN go back to left-clicking to select the "normal" background color. Yes, it's a hokey interface. But, it works.

Handicap - This is where you specify how many additional "given" images you want at the beginning of each puzzle. The game always generates the puzzles the same way, regardless of Handicap, including the clues and how many "given" locations there are. Then, when it's all done generating the puzzle, it gives you an ADDITIONAL number of "given" locations controlled by the Handicap setting. A Handicap of 0 gives you NO additional "given" images. A Handicap of 2 will give you 2 additional "given" images. Get the idea? I knew you would! The game maintains a different Handicap setting for each of the three puzzle sizes for each player. They default to 1, 2, and 3 for the Small, Medium, and Large puzzles. As you get better at the game, you should decrease the Handicap numbers in this dialog.

Music - the program uses MIDI files to play music in the background, either during the game and/or when you win a puzzle. A number of MIDI files are shipped with the licensed version of the game, although by default only one is selected for the background music and one for the "win" music. You can change the selections, or add multiple music files to the "play list" in this dialog. You can also add your own MIDI files to the game by simply copying them into the HOTEL directory (wherever you installed the game). They will then be available in the MUSIC dialog for adding to the playlists. You can also totally disable background music without modifying the playlists by simply unchecking the appropriate box in this dialog.

Sound effects - this enables (when checked) sound effects. Uncheck it for silence.

Require images & sides for highlighting - As you play the game, the background of a cell will change color when all six sides have been marked as a wall or a path. This does not indicate CORRECTNESS, just that you're "sort of done" with that cell. Some folks are only interested in knowing which cells have all of the sides completed, other folks want the IMAGE to be known also (so that the cell is COMPLETELY done). When this menu item is checked, the IMAGE must be known as well as the sides before the cell will be highlighted.

Notify on wrong deductions - By default, the program watches every move you make and will laugh at you when you make a mistake. When this happens, it actually won't LET you make a mistake, and will add 1 to the Hint count (since it's just given you a BIG hint). If you like to fly without wires, fall without a net, uncheck this menu item, and the program will no longer laugh at you.

Auto-complete images - When you've removed all but one image as a possibility from a cell, it's obvious that the remaining image MUST be there. With this menu item checked, the program will automatically make KNOWN images in that situation. It will also take care of instances where an image is only possible in one cell (even though there are still other possibilities marked in that cell). This makes the game more fun for some folks, and it's considered cheating by others. There's just too many kinds of people in this world! Using this option will affect your puzzle solving "standing" in the Best Scores list.

Auto-complete sides - Similar to Auto-complete images, only for sides. When a cell has 4 sides marked as Walls, the remaining 2 sides HAVE to be Paths. When a cell has two sides marked as Paths, the remaining 4 sides HAVE to be Walls. With this menu item checked, the program will automatically fill those in for you. This can sometimes result in a REALLY extended chain reaction of side completions! Using this option will affect your puzzle solving "standing" in the Best Scores list.

Prompt before new game - Some folks like to just whizzzzz from one game to another, playing endlessly through the night (or their day job...). Other folks like to not be tempted to "chain play". With this menu item checked, a dialog will come up at the end of the a puzzle (after the win display) asking if you want to play another game or not. Selecting "Yes" will take you immediately to the next game. Selecting "No" will leave you in the current puzzle in it's solved state (in case you want to look at it some more). Selecting "Exit game" will cause you to grow (more) hair in your ears.

Disable win display - Some folks don't need any kind of celebration of their victory, others just don't LIKE bees. Checking this menu item will cause the "win display" to not run when you solve a puzzle.

There are a couple of other options that are controlled only with keyboard keystrokes. To read about those, check out:

Keyboard and Mouse
Saving, Restoring, and Keeping *What If* Situations
Knowing What The Computer Knows: W and N keys

Uninstalling the Game

Why on earth would you want to do that???   Sigh...

Honeycomb Hotel installs ALL of its files in the HOTEL directory (or wherever you instructed the SETUP program to install it).   No other files in the system are modified (except for the results of adding the ICON for running the game).   So, to uninstall the game, merely delete the entire HOTEL directory, then delete the icon for the game.

Ta-da.   It's gone.

What Is "WHAT IF" and How Is It Used?

NOTE: *What-if* deductions are only necessary on higher numbered puzzles:
Puzzles 0-14999 require NO *what-if* deductions to solve them.
Puzzles 15000-29999 MAY require occasional PATH *what-if* deductions.
Puzzles 30000-44999 frequently require one or more *what-if* deductions (either image or path).
Puzzles 45000-65535 almost always require *what-if* deductions to solve them.

Most logic puzzles (the kind that you find in a magazine) that aren't "beginner" level puzzles require some advanced deductions where you have to SUPPOSE that something is true, then see if that causes any contradictions with the specified information. In Honeycomb Hotel, those are called *What-If* deductions. "*What if* this image were here?" or "*What if* this side were a WALL?" or "*What if* this side were a PATH?"

The goal of a *What If* supposition is NOT to guess something that is CORRECT, but rather to test the available possibilities and eliminate one or more by proving that they can NOT be correct. This seems a little backwards at first blush (something I rarely do...), but stick with me and I'll try to make it a little more clear.

When you make a *What If* supposition, followed by a string of deductions that result from having made that supposition, there are several things that can happen:

1) You solve the puzzle and the program tells you that you've won (bully for you).
2) You reach a point at which you can't make any more deductions (you get stuck).
3) You reach a point at which it becomes obvious that something is wrong (all right!)

In case #1 you're relying upon the computer to tell you that you've won, you haven't actually *proven* that the solution you've come up with is the *one and **only*** solution to the puzzle. Now, the program was written to generate puzzles that only have one solution, but do YOU trust computers??? It's fun to be right, but it's even more fun to *prove* you're right.

In case #2 you're...well...you're stuck. Let's leave it at that for the moment, and we'll come back to #2 in a minute.

In case #3 you've encountered a *contradiction*, such as an image that's deduced to be somewhere but which doesn't match up with what's specified by a clue (such as it's not in the same diagonal as another image, or there's the wrong wall/path between it and another image). Also, sometimes you'll end up removing ALL possibilities from a cell, which can't be since we KNOW that the game places an image in each cell. Other times, you'll see that the path has to form a loop or excludes some cells (prematurely connecting the ends of the path before it's passed through all cells) if the assumption were true. Any one of these contradictions immediately tells you that your assumption is WRONG, that it CAN'T be true. That means that you can either remove that "supposed" image from that location as a possibility, or if you were assuming that a side was a WALL then you know that it has to be a PATH (and vice versa).

The problem with *What If* deductions, though, is trying to keep track of all of the deductions that you make AFTER the initial supposition, because they all have to be UNDONE (since they were all made based upon a wrong assumption, or an assumption which, as in case #2, didn't tell you anything about whether it was right or wrong). The program comes to your rescue with a button on the toolbar called the

.....drum roll, please....

? "What If" button. (Imagine that!)

The "What If" button saves the current state of the game. Then, the very next thing you do must be to "enlarge" (make known) an image or mark a side as either a WALL or a PATH. That action is the *What If* assumption, and will be marked in BLUE. Then, you're free to make all further deductions that you can (until you run out of deductions or hit a contradiction). When you're done with the *What If* (for whatever reason), you click once more on the "What If" button (which stayed depressed after the first click) and that causes the program to restore the game to how it was when the "What If" button was originally pressed. HOWEVER (and this is IMPORTANT), before pressing the button the second time to exit the "what if" mode, if you've encountered a contradiction be SURE to note what your original assumption was (look for the "blue" color), as that will disappear when you click the button, and you'll need to remember that so that you can take the appropriate action!

Anytime the program is in this *What If* mode (the "What If" button is depressed) the background color of the game window changes. The default color is blue, but this can be changed in the *Options-Background Colors* dialog.

It's important to note that, while in this *What If* mode, the program does NO checking for the correctness of your deductions, EVEN IF YOU HAVE THE *"Notify On Wrong Deductions"* OPTION TURNED ON!!! This is because the entire purpose of *What If* is to make wrong deductions, so that you can see the contradiction, so that you can eliminate that original supposition.

It's also important to note that ONLY the original supposition can be proved wrong. The additional deductions that you make based upon that wrong supposition may or may not be correct. Frequently a cell image or a side will have the same value in several different *What If* cases. That doesn't mean that the additional cell or side is right or wrong. ONLY the original assumption can be proved wrong.

However, other cells or sides CAN be proven RIGHT by a series of *What If*s. For example, let's say there are only two images possible in a cell. You first try a *What If* on one of them, but it doesn't pan out (doesn't solve the puzzle and doesn't reach a contradiction). So, you try the other one, but it doesn't pan out either. However, you notice that in BOTH CASES (regardless of which of the two possible images you choose) another image ALWAYS ends up being known in the same cell, or a particular side ALWAYS turns out to be a WALL (or a PATH). Then, since you've tried all possible cases of images in that one cell, and in all cases this other "thing" turned out the same, then you KNOW that that "thing" must be true. (This is a tough one to explain!)

By the way, the above process (trying all cases and noting something that is the same in ALL cases) is something that YOU can do and make a valid deduction from (even if all of the assumptions individually had inconclusive results), but the program does NOT make use of this kind of logic when generating puzzles.

NOTE: When you click on the "What If" button to try a *What If* deduction, the game first checks to make sure that there are no errors in the current state of the game board, and will notify you if there are EVEN IF YOU HAVE "Notify on wrong deduction" TURNED OFF! This is because things (like the Hint system, etc) can get REALLY confused if you enter the *What If* mode with errors, and then ask for a Hint.

Now, back to #2 above (remember, I TOLD you we'd get back to it!) When you try a *What If* and it proves inconclusive because you reach a point at which you can make no further deductions, the puzzle has not been solved, and you've encountered no contradictions, then this *What If* must be abandoned. Conceivably you could make futher *What If*s at that point, but it quickly becomes very complex to do so, and is beyond the scope of this program. The program does not support "nested" or "multi-level" *What If*s, so abandon it we must.

The fact that this *What If* didn't "work out" (was inconclusive) tells us NOTHING. It does NOT mean that the *What If* was WRONG, it doesn't mean that the *What If* was RIGHT, and it doesn't mean that some other possibility is right or wrong. It tells us absolutely NOTHING except that we made an unfortunate choice for a *What If* at this point in the puzzle solving. However, if you made a large number of deductions based upon the *What If*, you may wish to save the state of the game before ending the "What If" mode in case later deductions prove that this *What If* really was the correct situation. See the Saving, Restoring, and Keeping What If Situations section below for more details.

Ranges Of Difficulty In The Puzzles

The NOTE at the top of this section describes the ranges of puzzle numbers, and what kinds of deductions MIGHT be necessary to solve those puzzles. This applies primarily to the Large-sized puzzles, as the small and medium puzzles rarely use *What If* deductions, except for the highest range (45000-65535) of Medium-sized puzzles.

0-14999 require no *What If* deductions to solve. They can all be solved using fairly straight-forward deductions, repeatedly applying the clues one at a time to the puzzle.

15000-29999 require occasional *What If* deductions for solving the path (but NOT for placing images).

30000-44999 frequently require *What If* deductions on both images and the path. However, the computer (in generating this range of puzzles) limits itself to checking for *What If* deductions in these cases:

1) If there are only two images possible in a cell, it will try a *What If* on both of those images in that cell.
2) If a specific image is only possible in two cells, it will try a *What If* on that image in both cells where it is possible.
3) If a cell has exactly one side known to be a PATH and no more than one other side known to be a WALL, then it will try a *What If* on each unknown side of that cell as a PATH. This is because with that one side *What If*, it also gets to deduce that the other 3 or 4 unknown sides are WALLs. This sudden appearance of lots of paths/walls means there's a good chance of further deductions.
4) If a cell has exactly three sides known to be WALLs and 0 or 1 of the other sides are known to be PATHs, then it will try a *What If* on each unknown side of that cell as a WALL. Again, this will "complete" ALL of the sides of that cell, making further deductions much more likely.

45000-65535 almost always require both image and path *What If* deductions. In this range, the computer (in generating the puzzles) will do image *What If*s in cells where there are 2 or 3 images remaining, and on images which appear in no more than 3 different cells. It will try doing PATH *What If*s almost anywhere.

Saving, Restoring, and Keeping *What If* Situations

Sometimes you will have two images as possibilities in a cell. You choose one to try a *What If* deduction upon. You make 20 or 30 further deductions, and eventually encounter a dead end, without reaching any conclusion or contradiction. So, you abort that *What If* and then try the other one. This second one reaches a contradiction, so you know that the FIRST *What If* assumption you made was actually the correct one. But now, you have to redo those 20 or 30 (or MORE) further deductions that you already did the first time around. What a pain in the aardvark! Well, there's a way to avoid the pain. :-)

When you reach that first dead end, inconclusive *What If*, you can save that current state (before undoing the *What If*) by pressing a number key from 0 to 9. This will save the current state of the game in an internal RAM buffer. Then, terminate the *What If*, try the other one. If it reaches a contradiction so that you know it's wrong, then you can hold down the SHIFT key and press the same number (from 0 to 9), and it will RESTORE the game to that previous state. It's kind of like a quick and dirty *File-Save Game* and *File-Load Game*. The problem now, though, is that you're still (or back) in the original *What If* mode, and if you click the "What If" button you will still lose all of those 20 or 30 further deductions. So, at this point you can press the K key (for "Keep"). It will leave all of the deductions in place and merely remove the game from the *What If* mode. If you have "Notify on wrong deduction" turned on, it will check for errors first and notify you if you're wrong, otherwise no check will be made.

There are 10 of these internal buffers, so you can save up to 10 distinct situations in the game. The first time you press one of these keys to save the game, you will simply hear a sound (like something heavy falling) and a message will appear in the game's message window below the game board. However, on subsequent saves to the same buffer, you will be prompted for confirmation so as to prevent you from accidentally overwriting a buffer's contents when you really wanted to press the SHIFT key first.

NOTE: These save/restore buffers are NOT saved to disk by a *File-Save Game*. They are very temporary and are intended only as a short-term "scratch" storage.

Knowing What The Computer Knows: W and N keys

There is a special advanced feature that can be used when you get more comfortable with the game and the idea of *What If* deductions. Some folks like to know how MANY *What If* deductions are needed to solve a given puzzle. Because the game is not capable of making some of the deductions that a human can make, this is a difficult (virtually impossible) question to answer. However, a question that CAN be answered is, "How many *What If* deductions did the COMPUTER have to make to solve this puzzle?" To find out this information, set the game to a puzzle number greater than 14999, and then press the W key. This will toggle (turn on or turn off) a special display that shows up on the left side of the game window that contains two numbers separated by a colon (like 3:5). The first number (3 in this example) denotes the number of IMAGE *What If*s the program had to make to solve the puzzle. The second number (5 in this example) denotes the number of PATH *What If*s the program had to make.

Now, take these numbers with a large grain of salt (about the size of a small boulder). The computer does not make very good guesses at where to start trying *What If*s, it just starts at one end of the puzzle (the bottom) and works its way to the other end, trying everything that fits within its limits to try. Sometimes the best *What If* to try is near the top of the puzzle, and the computer will find 3 or 4 other minor ones lower down that don't accomplish much. You (the human in this discussion) might spot that one at the top right off the bat, and thus only need one *What If* where the computer needed half a dozen. Of course, the reverse can also happen, but more often than not you should be able to solve the puzzles using fewer *What If*s than the computer did.

None the less, this counter display WILL give you SOME indication of the difficulty of a puzzle. Just be sure to swallow that boulder of salt, okay?

Once this *What If* display is turned on, you can press the N key to toggle "Notification when a *What If* might be needed." When this feature is turned on, the game will continuously monitor the state of the game. When it detects that IT would have to make a *What If* deduction in order to proceed further, it will turn the background of the *What If* display box BLUE (from it's normal black). This indicates that there are no more straight-forward deductions to be made using a single clue at a time (that is, that the PROGRAM knows about!)

These two features can ONLY be toggled on and off using the W and K keys. They are NOT available in the menu system.