*

A Blind Judge at the Beauty Contest
- or -
The Page Rank Problem

ISC1057
Janet Peterson and John Burkardt
Computational Thinking
Fall Semester 2016

We have seen how an Internet search engine can find the web pages that match a user's search words. However, such a search can return hundreds of thousands of matches, with only a few being useful.

Once the page match procedure has been completed, the page ranking procedure must figure out which results are of such high quality that they should appear on the very first page of results, getting the user's attention.

This is another example of an impossible task, since the search engine can't actually understand the web pages.

Nonetheless, we will come up with a solution, and it will work well even if the pages are written in Polish, Esperanto, or the Martian language!

Newton's - Kepler's E-mail to Alfred Nobel Prize winner Physicists

There are no physicists after us only "University" Idiots like you

Space time - Relativistic - Quantum - Strings mechanics is not physics

Real time Universal Mechanics

Newton's - Kepler's equations solved wrong for 350 years

1001 new real time physics formulas

Changing physics and the History of physics

Annexing quantum mechanics to classical mechanics and deleting relativity and strings

It is the math formulas that matches a physics experiment results

Real time Newton's - Kepler's mechanics

Professor Joe Nahhas July 4th 1973

joenahhas1958@yahoo.com

Read my Lips: I Joe Nahhas (lucid) will end Alfred Nobel Mafia of Physics and physicists' stupidity.

There is one and only one Mechanics

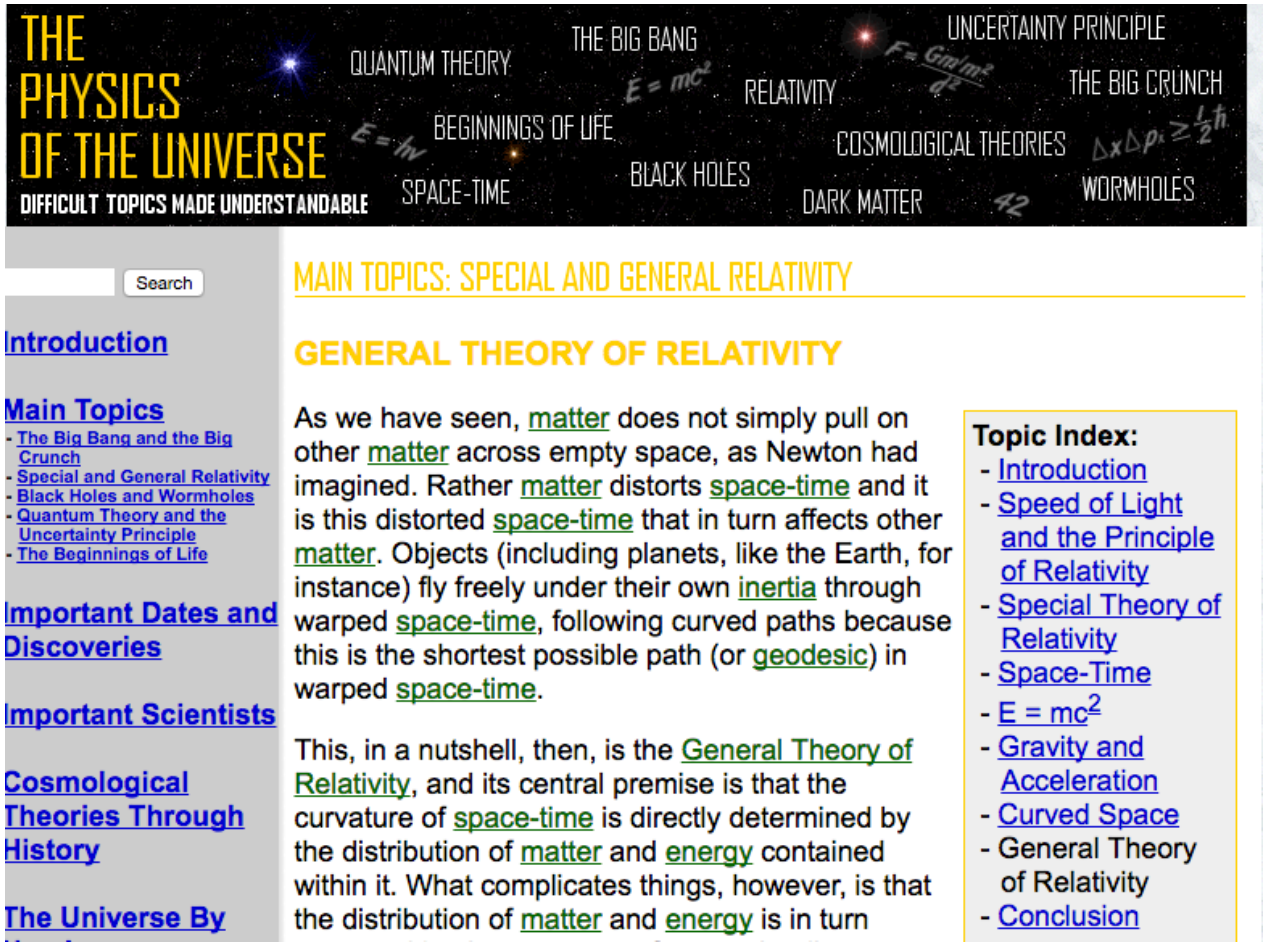
Real time universal mechanics

Real time mechanics is the natural law of past present and future of mechanics.

For 400 years Newton's - Kepler's were formulated and solved wrong.

The new real time solution of Newton's Kepler's equations deletes modern physics which is based on relativistic quantum string time travel mechanics and matches experiments with unprecedented accuracy to change physics and the history of physics in its entirety.

Modern Physics is based on relativistic quantum string time travel mechanics. Time is caveman and modern man scale of convenience and is not Alfred Nobel dimension for time travel regardless of what all Alfred Nobel time travel "Physicists" have to say about it. Time travel is not physics and accepting time travel as physics in classrooms and using it in scientific



Based only on the words in this page and the previous one, a search engine couldn't decide which one is more reasonable.

Using indexes of the World Wide Web, a search engine can rapidly identify all the web pages that match a given set of search words, and using the **Nearness Trick** and the **Metaword Trick**, it can even make some guesses about whether one page is more relevant than the other.

But *any one* can write a web page, and say anything that they like. This means that the Internet is full of "information" that is mis-informed, illogical, biased, fraudulent, deceptive, or ignorant. At a library, the librarian chooses which books to buy; on the web, things just appear with little authentication or checking.

If you want a taste of unusual information, search for: The Flat Earth Society, The Null Physics web page, Joe Nahhas's web site, Quantonics, aliens, the Einstein conspiracy, the Illuminati.

Now the search engine doesn't know physics, and it can't really read English (that is, it can't understand the meaning of English sentences), so if a user is searching for reliable physics information, how can the reliable, reputable, reasonable web pages be selected from the ocean of misinformed or irrelevant matter?

Again we seem to face an impossible problem, like a blind judge at a beauty contest, or a deaf person judging musical compositions, or an American teacher being asked to grade essays written in Korean.

Early versions of search engines actually tried to solve this problem by having people read and rate individual web pages; the ratings could be added to the web page index, as a general idea of the quality of the page. But such a rating system is very expensive, requires hiring experts in every possible field, and must be updated daily as web pages change and new ones are added.

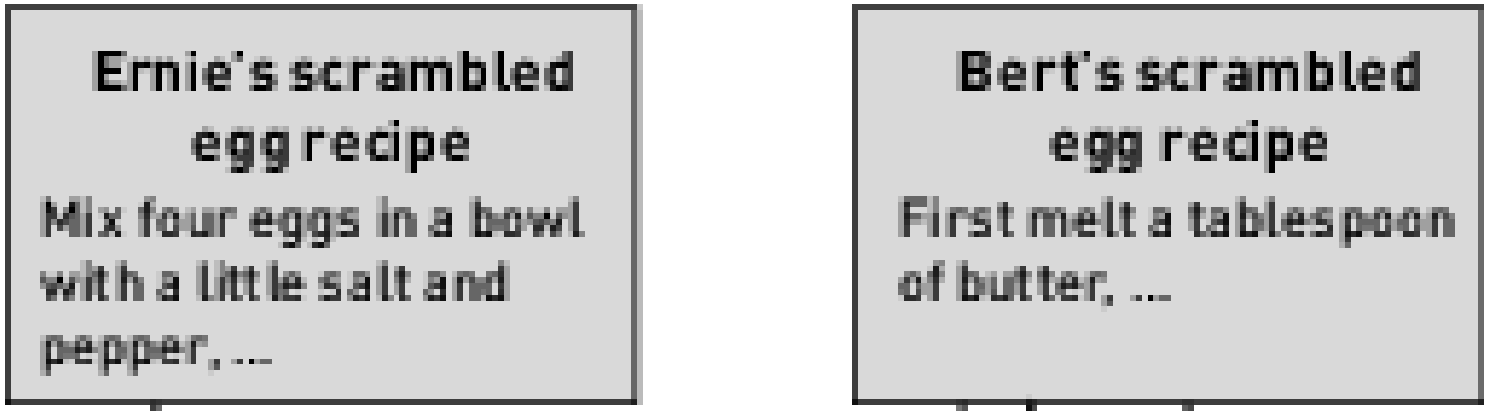
Since the search engine doesn't have intelligence, and can't understand web pages, the only solution must rely on taking advantage of someone else's intelligence.

The main key to the successful page ranking procedure can be called **The Hyperlink Trick**, because it has figured out a very clever way of analyzing the hyperlinks in web pages.

Most web pages include hyperlinks, allowing the author of a web page to point to other parts of the current web page, or, most interestingly, to other web pages that may be related, or helpful, or interesting to someone reading the current page.

Hyperlinks usually show up underlined or in a special color to catch the user's attention.

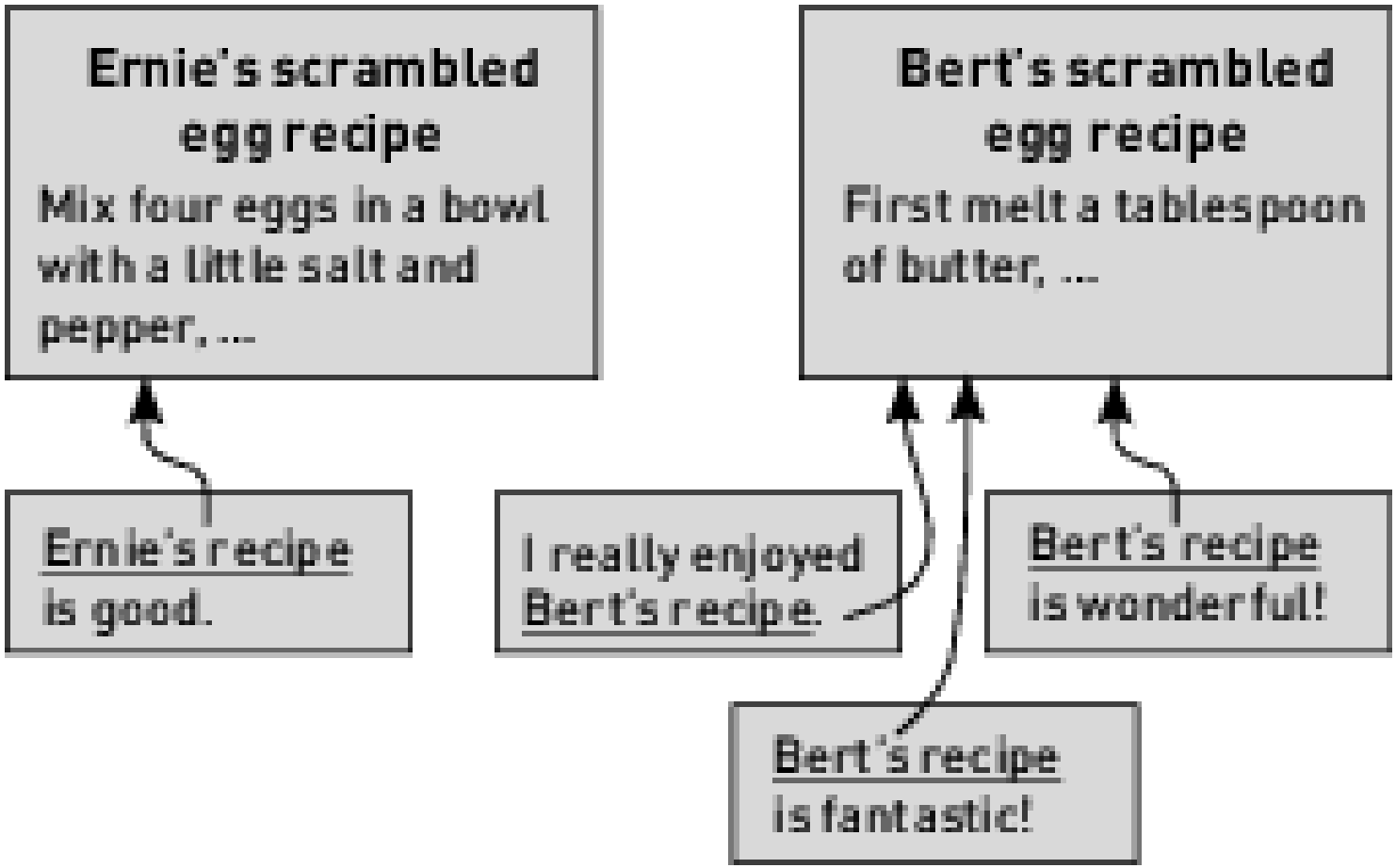
A hyperlink may offers a definition, an expanded discussion, or recommended further reading.



As a simplified example of the Hyperlink Trick, suppose there are only 6 web pages on the Internet.

You search for a scrambled egg recipe and the search engine returns two results, Ernie's recipe and Bert's recipe.

Can the search engine choose which recipe to recommend more strongly?



The search engine sees that Ernie has only one positive review, while Bert has three, suggesting that Bert's web page is better.

Now a search engine can't read or understand the recommending pages.

But it can certainly count the number of links coming in to either of the two recipes.

The fact that Bert's page has more links is at least a suggestion that people (who can read and understand web pages!) found Bert's page more useful, or his recipe better tasting.

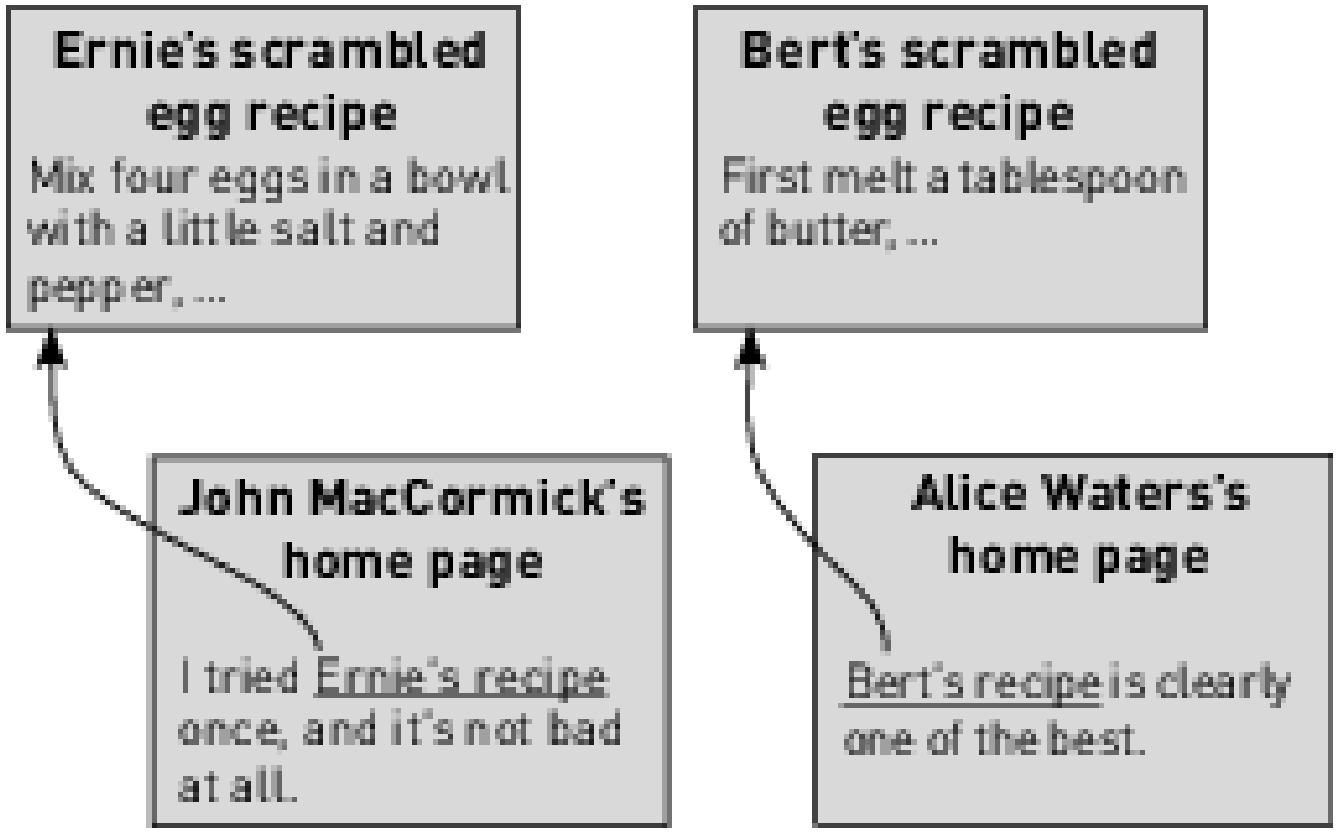
So in the absence of any better information, a search engine could take the number of incoming links to a web page as a rough indication of the value of that page.

It's easy to see some problems with such a simple ranking system.

For one thing, if all the web pages pointing to Bert's page said "This recipe is terrible!", the search engine would still give Bert's page the same ranking.

For another problem, if Ernie knew how the search engine works, he could quickly write 10 new web pages that praise his recipe and link to it.

Another issue arises because not all recommendations should be counted equally. High school students and film critics both make top ten lists of movies.



Let's go back to Bert and Ernie's recipes, assuming each now has a single recommendation.

John MacCormick is $\mathbf{not}$ a famous chef, but Alice Waters is.

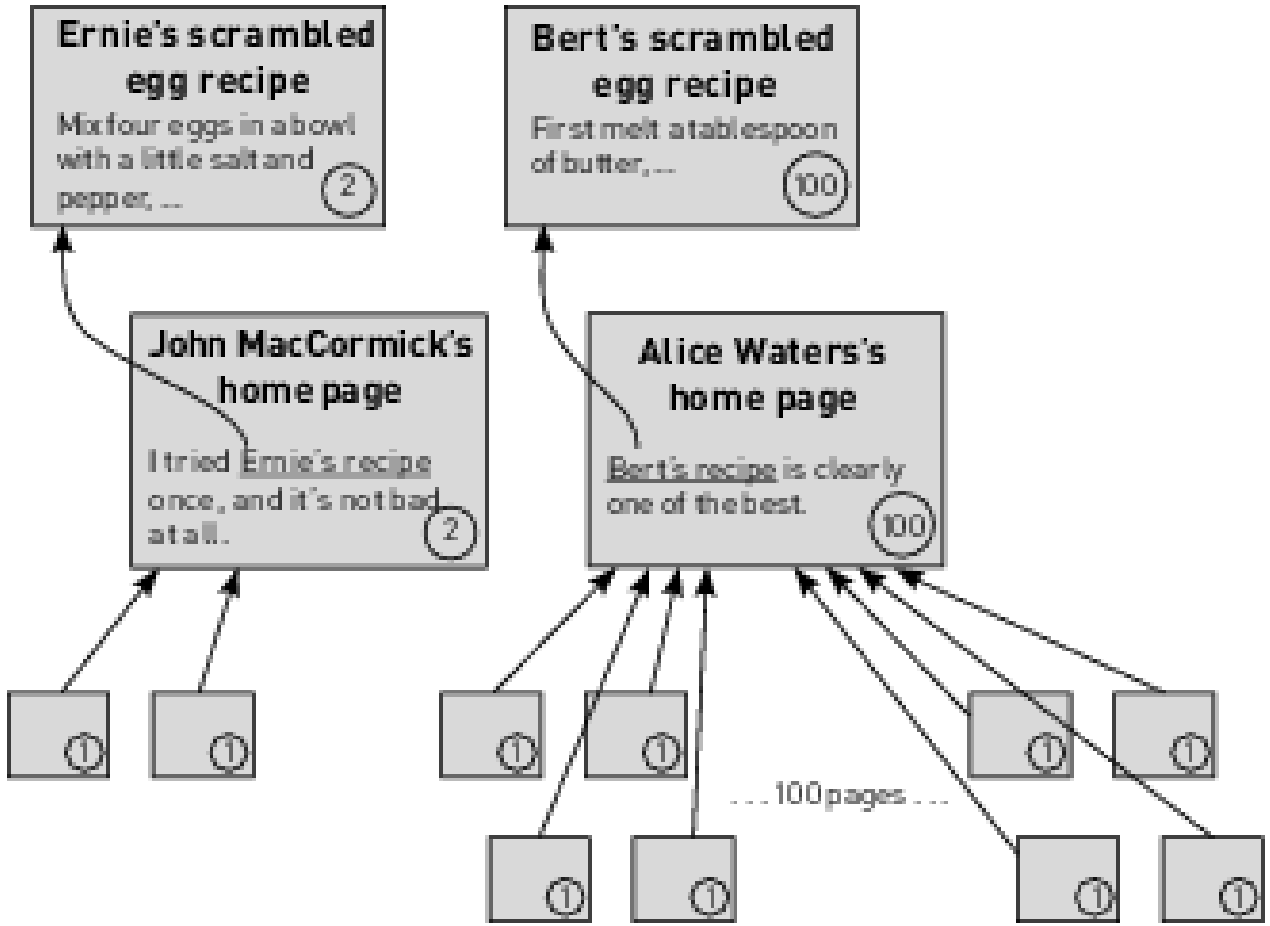
If we, being humans, know that John MacCormick is not a famous chef, but Alice Waters is, then we are likely to assume that it's safe to prefer Bert's recipe, because Alice Waters's recommendation has more authority.

We would like to modify our ranking procedure. Instead of only counting the number of hyperlinks to pages, we'd like to include somehow a measure of the authority of the web page that is making the recommendation.

If we can figure out how to do this, we will call this procedure **The Authority Trick**. Hyperlinks from pages with high "authority" will result in a higher ranking than links from pages with low authority.

But how can a computer determine authority?

Let's consider combining the Hyperlink Trick with the Authority Trick.



Trying to determine "Authority".

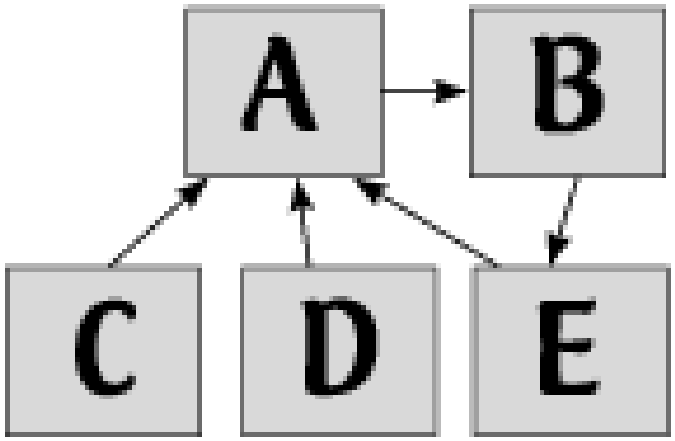
Let's start by assuming there are 102 web pages that point to John MacCormick or Alice Waters, and let's assign each of these web pages an authority of 1.

Now suppose John MacCormick has 2 hyperlinks pointing to his web page, while Alice Waters has 100.

Then we might give MacCormick an authority of 2, and Alice Waters an authority of 100, almost as though the lower level web pages were voting for them.

Then we might suppose that any recommendation (hyperlink) by Alice Waters should add 100 "authority points" to that web page, while a recommendation by John MacCormick would only be worth 2 points.

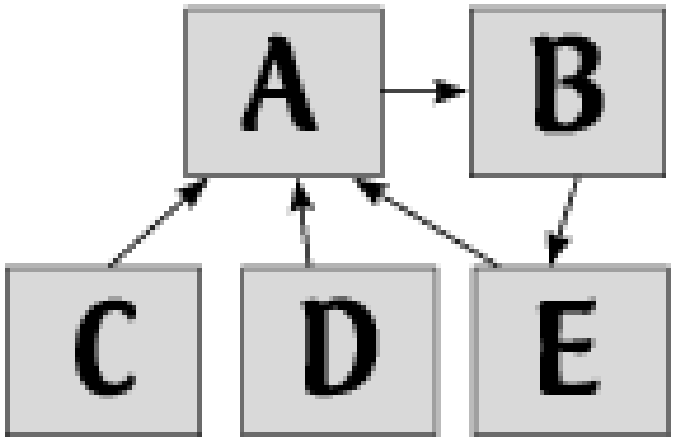
This means that Ernie's recipe has an authority score of 2, and Bert's 100.



The ideas of using hyperlinks and assigning authority are good ones. We might try to implement these ideas by starting every web page with one authority point. Then, each hyperlink in a page would cause that page's authority points to be added to the linked page's authority points.

Then we just have to do this for all web pages and we're done, right?

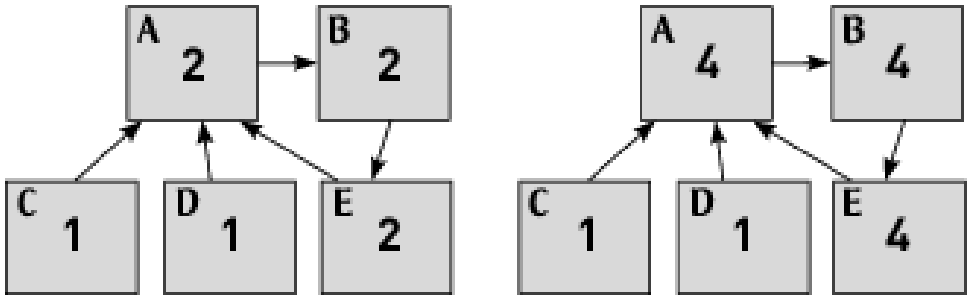
Unfortunately, this idea won't quite work. A problem arises if we encounter a cycle, a sequence of hyperlinks forming a loop.



In this situation, you can start on page A, jump to B, then E, and back to A.

The pages A, B, and E form a cycle.

This means that our method for assigning authority points will fail.



C and D have no links pointing to them, so they get a score of 1.
C and D point to A, so A gets a score of 2.
A points to B, and B points to E, so they get scores of 2 as well.

Are we done? No, A's score is out date now!

We update A to 4. But then we must update B and C...and A again. So we have to go around forever this way handling a cycle. And there must be an immense number of such cycles over the entire Web.

To save the **The Hyperlink Trick** and **The Authority Trick**, we need to add **The Random Surfer Trick**.

The random surfer trick imagines a person surfing the Internet.

A starting page is picked at random. If this page has any hyperlinks, the surfer picks one at random and moves to that new page. If that page has hyperlinks, another random choice leads to another page, and so on.

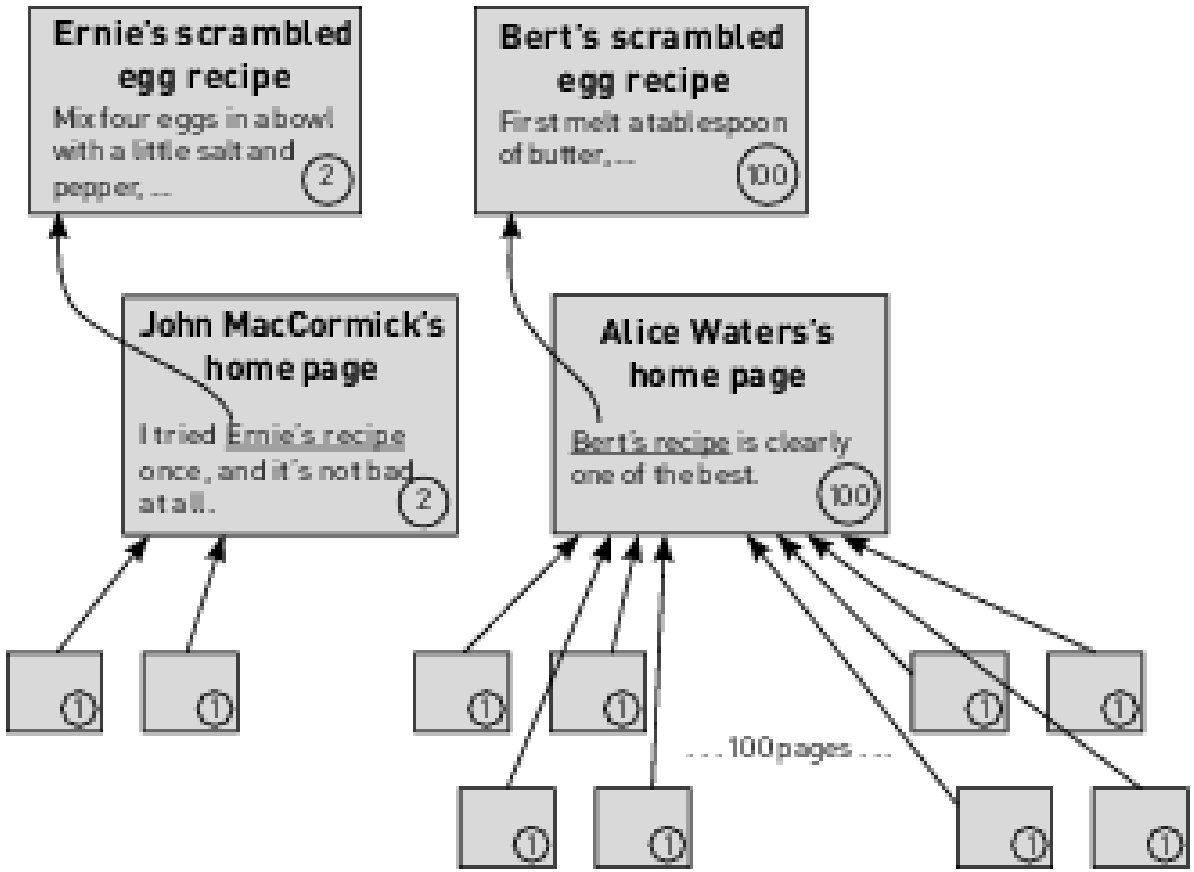
If a page has no links, a new page is chosen at random (a jump).

Even if the current page has links, the surfer is allowed occasionally to instead make a jump to a random page, as though the current topic was no longer interesting.

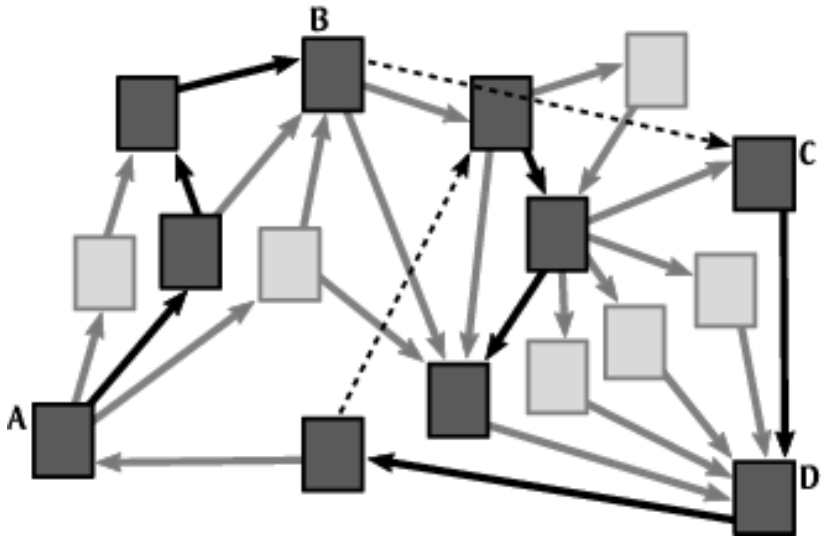
Randomly surfing the Internet seems an odd way of trying to understand the authority index we are seeking.

However, if we do this experiment many, many times, then you should be able to see that a web page that is pointed to by many links will be more likely to be visited often by the random surfer.

So the random surfer trick estimates the authority index by wandering through the Internet, and noticing which pages it visits most often.



In the simple world of Bert and Ernie's recipe pages, the random surfer would be much more likely to start in the 100 pages that point to Alice Waters, and hence to end up at Bert's recipe.



This sample of random surfing starts at page A, then moves to another page following a randomly selected link (darker arrow). Three such steps reach page B.

From page B, the surfer jumps (dashed line) to page C, then links to page D, then another page, then another random jump.

From there, the surfer takes two linking steps and stops.

It turns out that if you let the random surfer wander around the web like this, then you have solved the authority index problem.

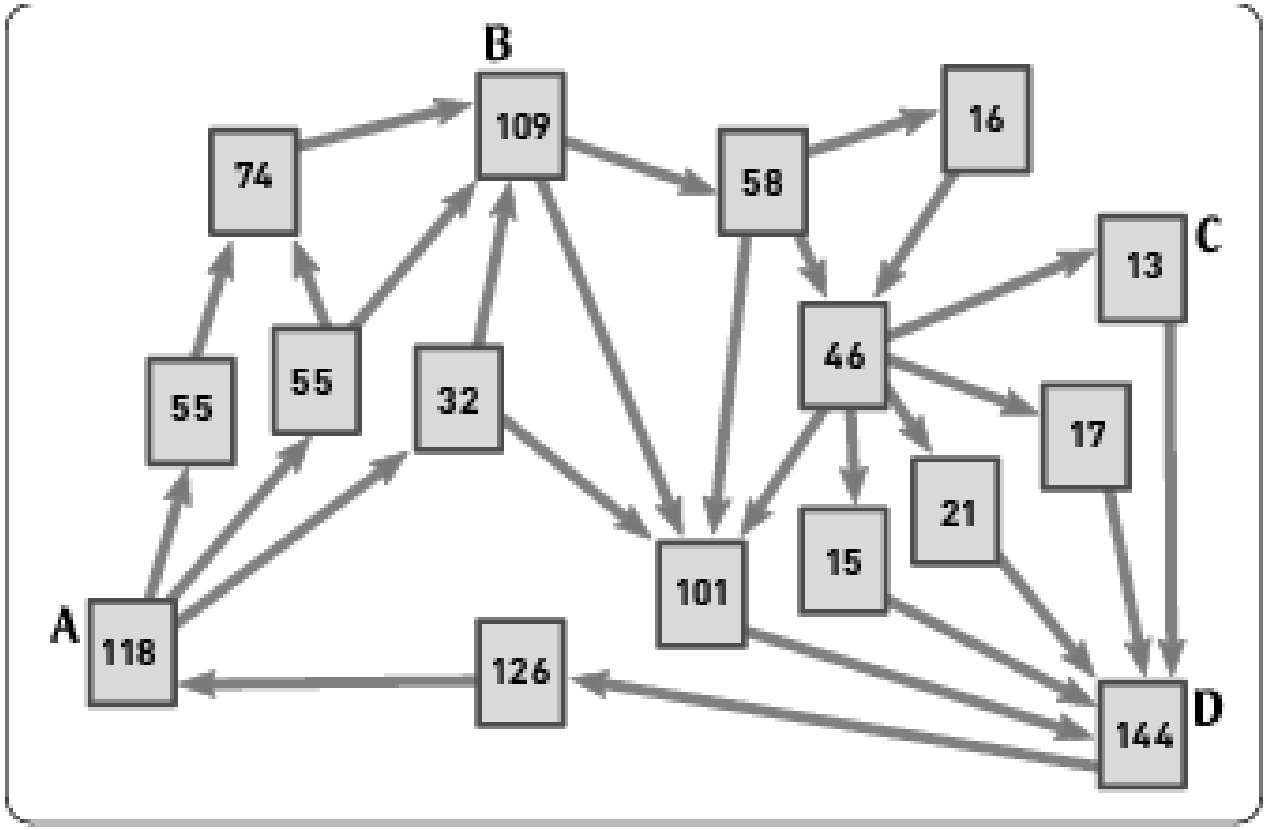
This is because, in a natural way, the importance or authority of a web page is related to the number of times the random surfer visited that web page.

More precisely, if we make the authority index a percentage, then the authority of a web page is the percentage of visits that were made to that page.

Of course, we don't want an actual person to have to browse the billions of web pages. But this is another job that is perfect for a computer.

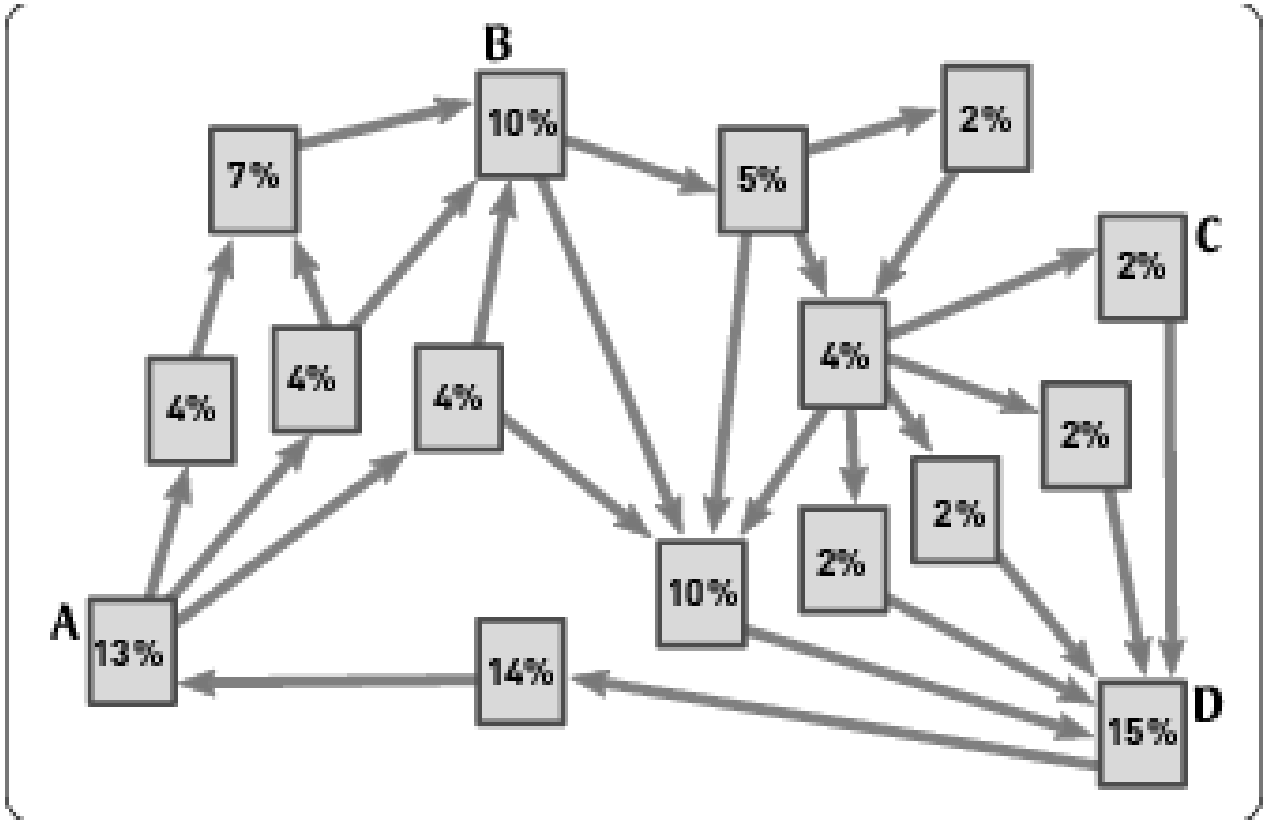
The program might look something like this:

```
start on a random page

repeat 1000 times:

  "remember" that you visited this page.

  if no hyperlinks on this page
    jump to a new random page;
  or if "bored":
    jump to a new random page;
  otherwise:
    choose a random hyperlink and move to that page.
```



In this example, the Internet only has 16 web pages.

We have recorded the number of times each page was viewed by the random surfer over 1000 steps.



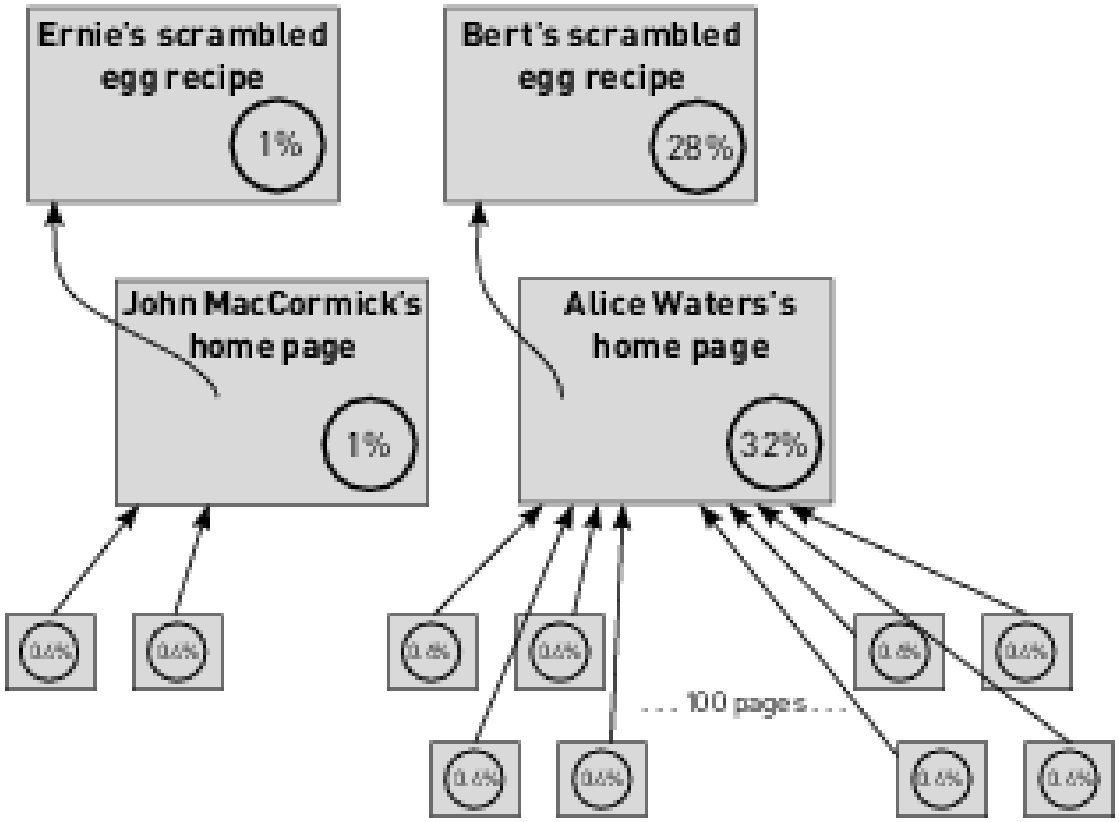
After 1,000,000 steps, we estimate authority as a percentage.

Authority = 100 * number of visits to this page / number of steps.

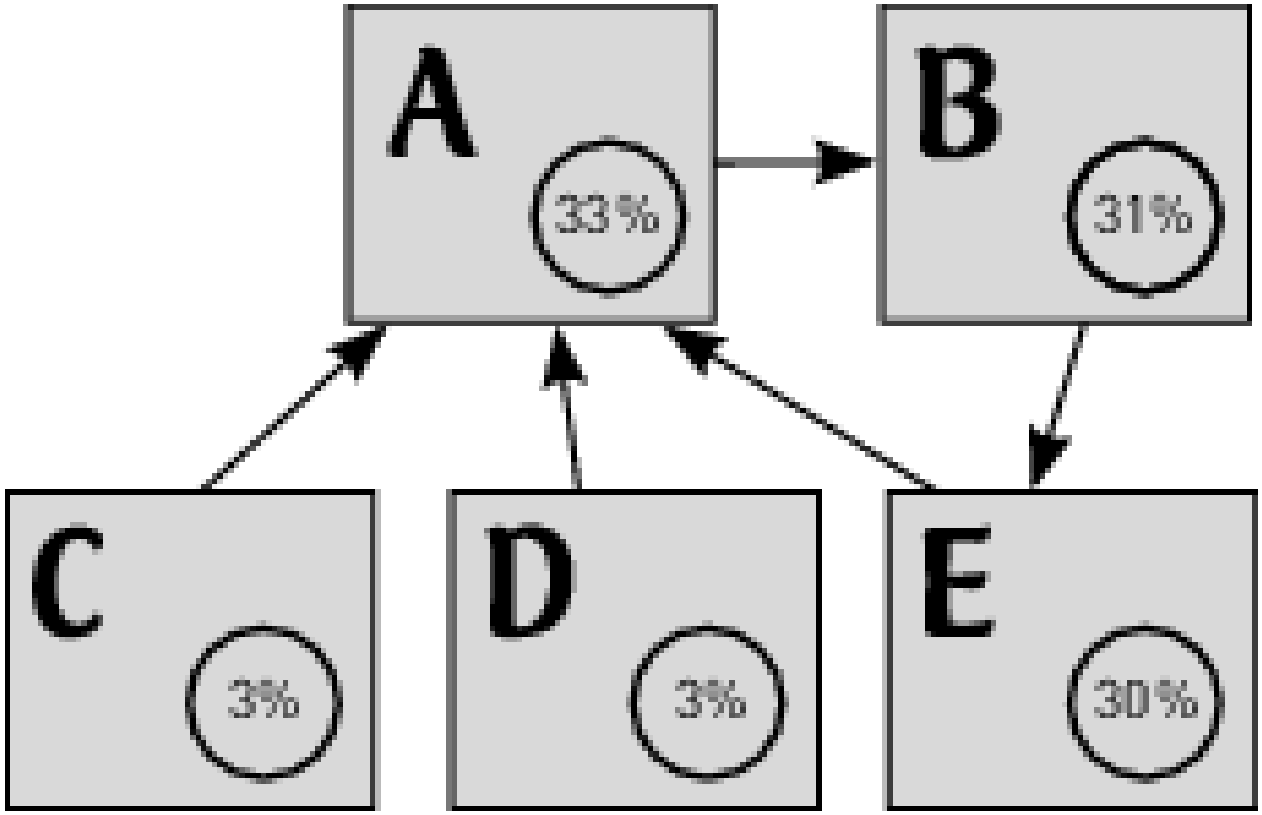
The authority index produced by random surfing turns out to incorporate both of the tricks we came up with earlier:

- *the hyperlink trick* suggested that a page with many incoming links should receive a high ranking. But the more incoming links a page has, the more likely the random surfer will visit it;
- *the authority trick*: an incoming link from a highly authoritative page should improve a page's rank more than a link from a less authoritate page. But a popular page will be visited more often than an unpopular one, and so there will be more opportunities for the surfer to arrive from the popular page than the less popular one.

In our previous example, notice that web pages A and C both have only one incoming link. However, A's link comes from a very popular page, whereas C's link is much less popular. The result is that A has an authority index of 13%, while C's index is only 2%.



Consider what happens if we use the random surfer trick to compare the authority of Bert and Ernie's recipes. Even though both recipes only have one incoming link, Bert's is rated much higher.



Earlier, we had a problem if there was a cycle among a set of web pages. However, the random surfer gets "bored", so it will never get stuck repeating the same sequence of pages forever. The authority figures for this example can now easily be worked out.

A good search engine needs two things:

- a page match finds pages that match the user's search words;
- a page rank method sorts matching pages so the most authoritative are listed first.

The hyperlink trick (which connects one page to others) and the authority trick (which suggests that a page with many incoming links is important) allow a computer to guess which pages are important, even though the computer has no idea of what the pages mean.

The random surfer trick makes this computation possible to carry out even when pages have a cycle.

It only took half a second to carry out 1,000,000 steps of the random surfer procedure in our earlier example using 16 pages.

Since the World Wide Web has 4 billion pages, it will obviously take much longer to compute a complete authority index list.

However, if several computers carry out a separate random surfer analysis, the results can be combined. Since Google has about two million computer processors available, the task suddenly becomes much more doable.

Moreover, web pages don't change very fast, so results can be computed every week or so.

So a good search engine will have available an up-to-date ranking of all web pages, **before** a user has made any search requests.

Thus, a good search engine can respond in seconds to a user request, *without having to access the web at all* and *without understanding the meaning of the search words or the matching web pages.*

Preprocessing created the word/metaword index for all pages.

Preprocessing created the authority index for all pages.

In response to a user request, the search engine finds all the matches in the word/metaword index, then sorts them by the authority index and presents the list.

Only when the user clicks on a particular result in the list do we actually have to be able to access the full World Wide Web.

Early search engines like Infoseek, Lycos, AltaVista, relied on indexes and authority ratings created by humans, but this only allowed a small set of web pages to be indexed and the information was constantly getting out of date.

In 1998, Larry Page and Sergey Brin announced their PageRank algorithm, which was built into the Google search engine. Search results were noticeably better and faster, and the "first page" results often exactly what users were seeking, so that Google soon became the dominant search engine.

Google and its competitors have continued to improve their search engines, and the authority index is hence only one part of the procedure by which pages are ranked.

A web page may get an improved rank because it is relatively new, or the search word appears many times, or it uses metawords that indicate the structure and meaning of the web page.

Initially, the World Wide Web was simply a way for scientific researchers to communicate.

Over time, an increasing portion of the web involved commercial services, and advertising. A company would be willing to pay a programmer to create a web page advertising its products, and the programmer would be paid every time a user clicked on the web page. Suddenly, making a web page have more authority became a matter of big money.

Programmers developed techniques of Search Engine Optimization, or **SEO**, to increase the rank of their web pages so that they would show up on page 1 of search results.

One way to do this involves creating thousands of dummy web pages that all hyperlinked to the ad web page (tricking the Hyperlink Trick!). Another way was to include thousands of likely keywords in the web page title, (tricking the MetaWord Trick!).

Thus there is a kind of evolutionary battle between search engines (looking for useful information to keep their users happy) and search engine optimizers (looking for users and ratings to keep their advertisers happy).

The success of the page ranking procedure is amazing.

It's another example in which a computer seems to be able to do something that only could be done with human intelligence.

In fact, once again, we have seen that the computer is not using human intelligence; it is not reading web pages, it is not understanding them.

So the amazing thing about the page rank procedure is that it shows that, if you're willing to accept imperfect results, human intelligence is not necessary.