A Motivating Example: data integration on the Web

- Our goal: for a given resource (person, an idea, an event or a product), we would like to know everything that has been said about it (on the Web, of course)
- We would like to accomplish this goal by collecting as much information as possible about this resource, we will then understand it by making queries against the collected information
  - ✓ Let us use myself as the resource
  - ✓ Let us assume we have already built a piece of software, which will walk around the Web and try to find everything about me on our behalf
  - ✓ this piece of software is almost like a crawler, but it is not exactly a crawler, so we call it a soft agent

A Motivating Example: data integration on the Web

- To get our agent started, we feed it with the URL of my personal homepage as the starting point of its journey on the Web: `http://www.liyangyu.com`
- our agent downloads this page, and tries to collect information from this page

if my Web page were a traditional Web document, our agent would not be able to collect anything that is much useful at all

- the only thing our agent is able to understand on this page would be those HTML language constructs, such as `<p>`, `<br>`, `<href>`, `<table>`, etc
- besides telling a Web browser about how to present my Web page, these HTML constructs do not convey any useful information about the underlying resource
- to our agent, my whole Web page would simply represent a string of characters that look no different from any other Web document

A Motivating Example: data integration on the Web

- however, let us assume my Web page is not a traditional Web document: besides the HTML constructs, it contains some "statements" that can be collected by our agent
- all these statements follow the same simple structure, and each one of them represents one aspect of the given resource

```
ns0:LiyangYu ns0:name      "Liyang Yu".
ns0:LiyangYu ns0:nickname  "LaoYu".
ns0:LiyangYu ns0:author    ns0:_x.
ns0:_x       ns0:ISBN      "978-1584889335".
ns0:_x       ns0:publisher http://www.crcpress.com.
```

- ✓ not to worry about the issues such as how these statements are added to my Web page, and how our agent collects them
- ✓ simply assume when our agent visits my Web page, it can easily discover these statements

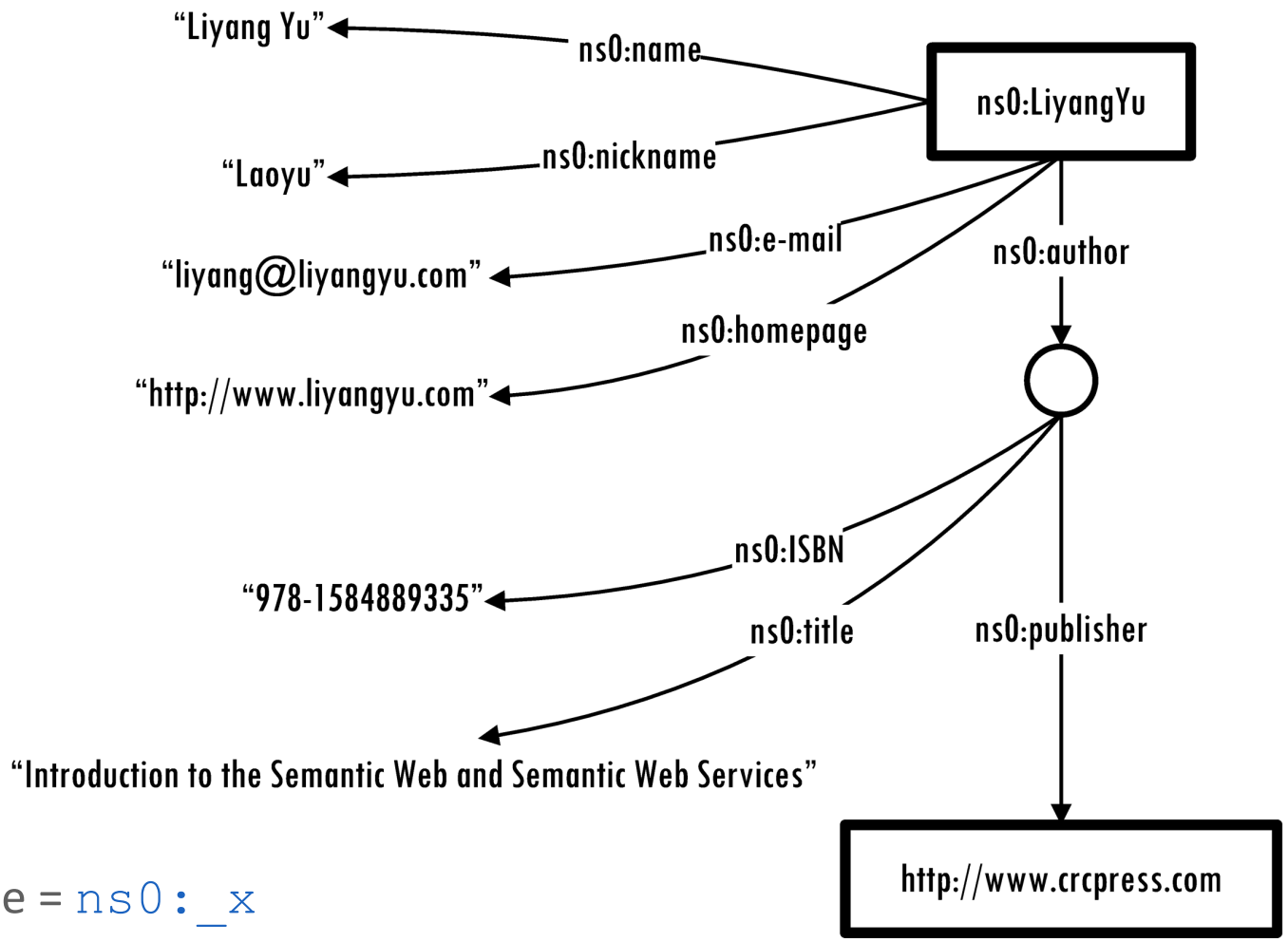
A Motivating Example: data integration on the Web

```
ns0:LiyangYu ns0:name       "Liyang Yu".
ns0:LiyangYu ns0:nickname   "LaoYu".
ns0:LiyangYu ns0:author     ns0:_x.
ns0:_x       ns0:ISBN       "978-1584889335".
ns0:_x       ns0:publisher  http://www.crcpress.com.
```

- `ns0:LiyangYu` represents a resource that is described by my Web page, in the case, this resource is me
- this is how to read the first statement: resource `ns0:LiyangYu` has a `ns0:name` property, whose value is `Liyang Yu`
- so, each statement adds one *property-value* pair to the resource that is being described
- also note that `ns0:author` property has used another resource as its value
- these statements make sense to our human eyes no matter how ugly they look
- the real interesting question is, how much does our agent understand these statements?

A Motivating Example: data integration on the Web

- not much at all
- but still, out agent can organize these statements into the following graph:



circle node = `ns0:_x`

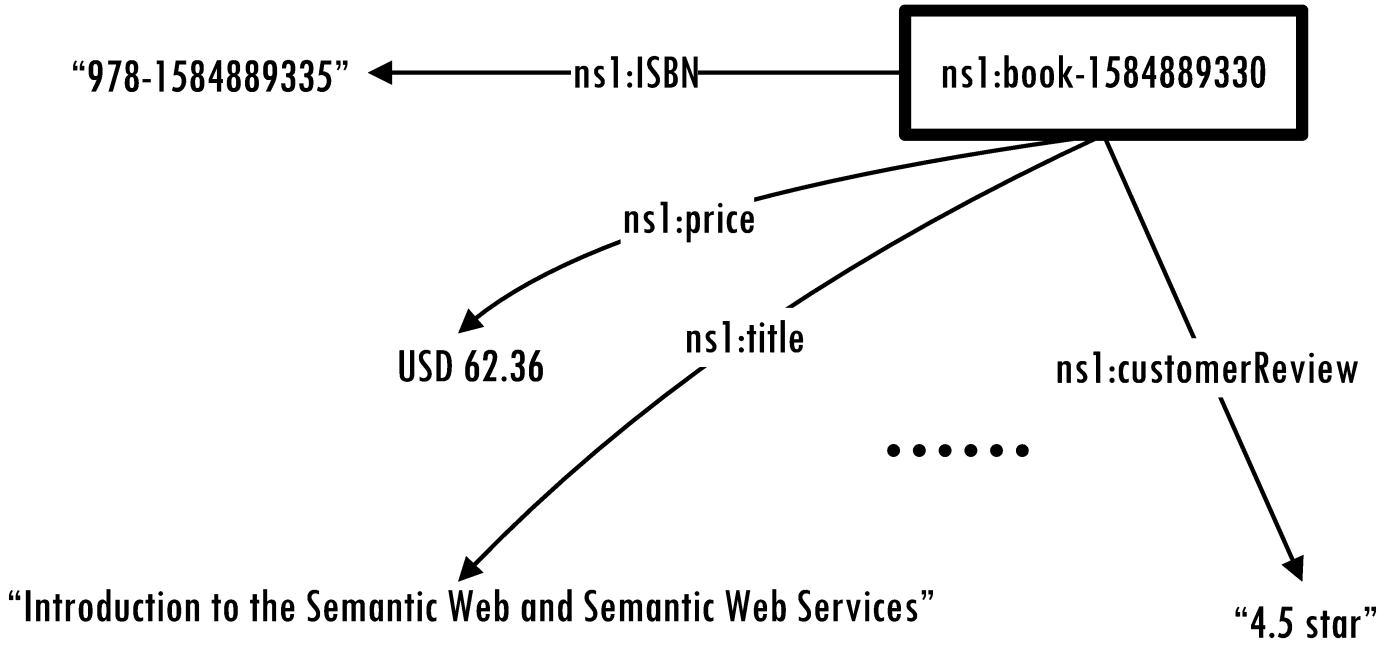
A Motivating Example: data integration on the Web

- our agent declares its success on my personal Web site and moves on to the next one
- Let us say the next Web site our agent hits is `www.amazon.com`
- similarly, if Amazon were still the Amazon today, our agent could not do much either
- again, assume Amazon is a new Amazon already. Our agent can therefore collect lots of statements, which follow the same format as shown before

```
ns1:book-1584889330 ns1:ISBN           "978-1584889335".
ns1:book-1584889330 ns1:price          USD62.36.
ns1:book-1584889330 ns1:customerReview "4.5 star".
```

A Motivating Example: data integration on the Web

```
ns1:book-1584889330 ns1:ISBN           "978-1584889335".
ns1:book-1584889330 ns1:price          USD62.36.
ns1:book-1584889330 ns1:customerReview "4.5 star".
```



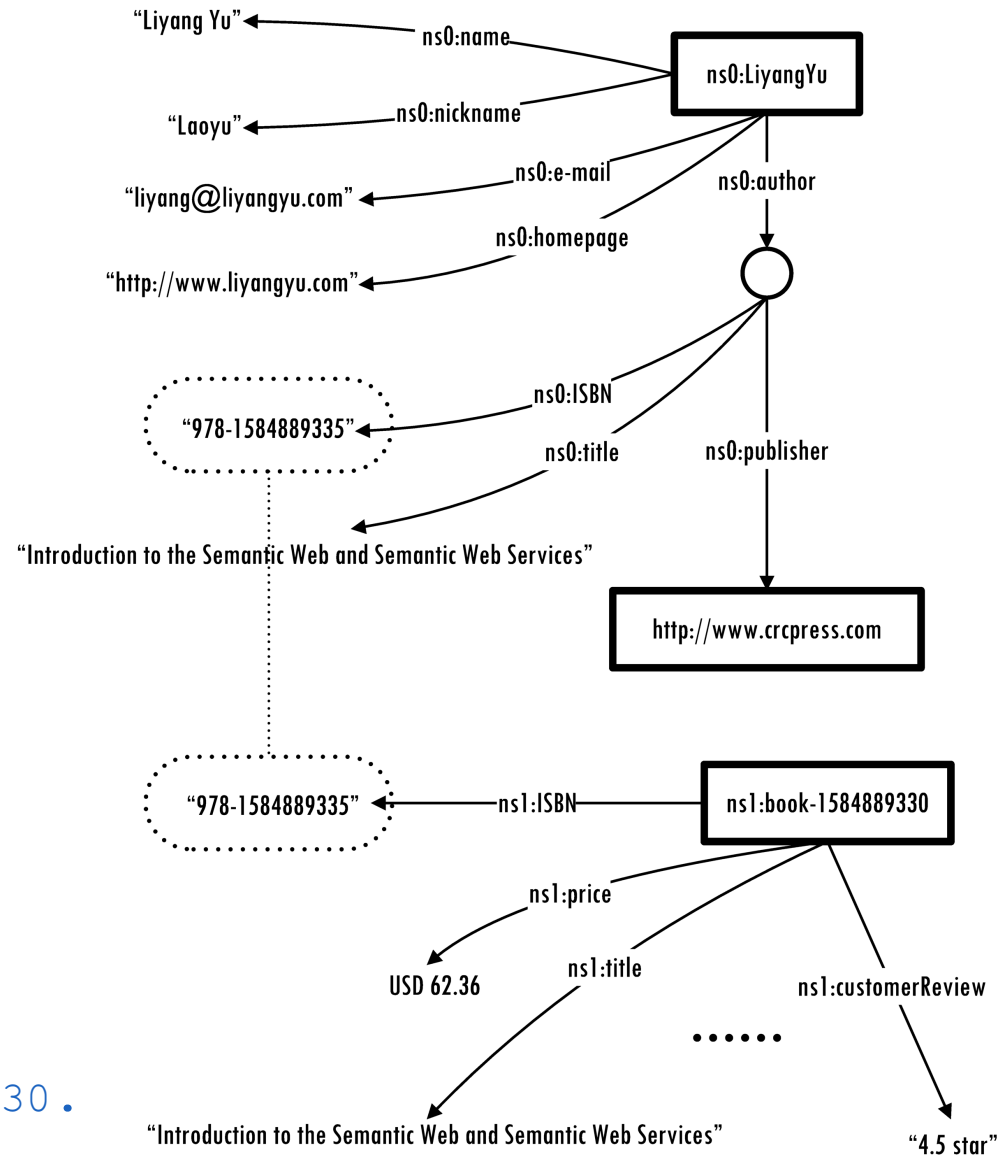
A Motivating Example: data integration on the Web

```
ns1:book-1584889330 ns1:ISBN           "978-1584889335".
ns1:book-1584889330 ns1:price          USD62.36.
ns1:book-1584889330 ns1:customerReview "4.5 star".
```

- obvious, `ns0:_x`, as a resource, represents the same item denoted by resource named `ns1:book-1584889330`
- once we made this connection, we start to see other facts easily. For example, a person whose homepage is `www.liyangyu.com` has a book published, its current price : US$ 62.36 on Amazon
- this fact is not explicitly stated on either one of the Web sites, our human minds have integrated the information from both sites to reach this conclusion

A Motivating Example: data integration on the Web

- our agent sees the ISBN number, 978-1584889335, showing up in both graphs, it will therefore make a connect between these two appearances
- two graphs are now "glued" together by overlapping `ns0:_x` node with `ns1:book-1584889330`




```
ns0:_x sameAs ns1:book-1584889330.
```

A Motivating Example: data integration on the Web

- now our agent hits `www.linkedIn.com`

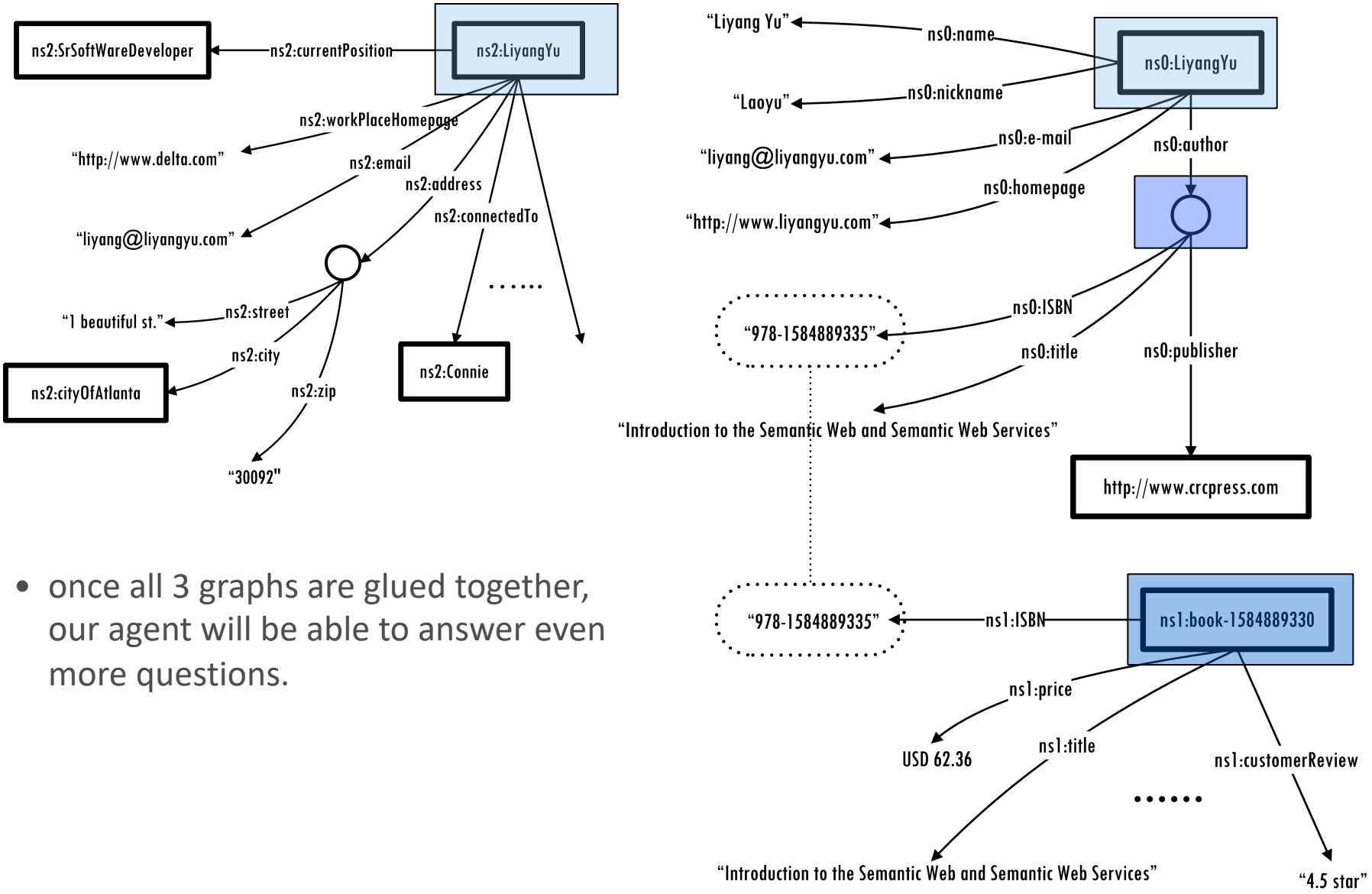
```
ns2:LiyangYu ns2:email               "liyang@liyangyu.com".
ns2:LiyangYu ns2:workPlaceHomepage   "http://www.delta.com".
ns2:LiyangYu ns2:connectedTo         ns2:Connie.
```

- for human readers, we know `ns0:LiyangYu` and `ns2:LiyangYu` represent the same resource
- for our agent, just by comparing the two identifiers does not ensure the fact that these two resources are the same

if the `e-mail` property of resource A has the same value as the `e-mail` property of resource B, then resource A and B are the same resource.

```
ns0:LiyangYu sameAs ns2:LiyangYu.
```

A Motivating Example: data integration on the Web



- once all 3 graphs are glued together, our agent will be able to answer even more questions.

A Motivating Example: data integration on the Web

- what is Laoyu's workplace homepage?
- how much does it cost to buy Laoyu's book?
- which city does Liyang live in?
- to answer these questions, the integrated graph is necessary

```
ns0:LiyangYu ns0:nickname          "LaoYu".
ns0:LiyangYu sameAs                ns2:LiyangYu.
ns2:LiyangYu ns2:workPlaceHomepage "http://www.delta.com".
```

these statements are all contained by the integrated graph, and answers are found by pattern-matching

A Motivating Example: data integration on the Web

- each statement represents a piece of knowledge. There must be a way (a model) to represent knowledge on the Web
- this model must be easily and readily processed (understood) by machines
- this model must be accepted as a standard by all the Web sites
- there must be a way to create these statements on each Web site (manually added or automatically generated)
- the statements cannot be completely arbitrary. They should be created by using some common terms and relationships
- there must be a way to define these common terms and relationships
- perhaps more to be included...

A Motivating Example: data integration on the Web

- so, it is quite complex
- yet the benefit is obvious: large-scale data integration is easier, new applications will be simply limited by our imaginations...

so now, what is the Semantic Web?

- all the technologies and standards that will make data integration on the Web possible

A More General View: what is the Semantic Web?

- Tim Berners-Lee, *a.k.a*, Tim BL
- inventor of the web
- was knighted by Queen Elizabeth II for his pioneering work (2004)

- Web and Web-based applications have changed our way of living
- *if you are not on the Web, you don't exist*
- however, Web content is mainly consumed by human eyes

it is slow, tedious, not scalable, can be wrong…

A More General View: what is the Semantic Web?

- Tim Berners-Lee, *a.k.a*, Tim BL
- inventor of the web
- was knighted by Queen Elizabeth II for his pioneering work (2004)

- Web and Web-based applications have changed our way of living
- *if you are not on the Web, you don't exist*
- however, Web content is mainly consumed by human eyes

what if machine can consume the content?

A More General View: what is the Semantic Web?

Tim BL suggests:

- add some meanings (semantics) to those existing Web documents, so machines can understand them
- the resulting Web is called the *Semantic Web*

Tim BL formally introduced the idea to the world by publishing a paper titled "**the Semantic Web**" on *Scientific American, May of 2001*

A More General View: what is the Semantic Web?

For different people, the Semantic Web means different things:

- can improve search by adding structured data to Web contents
- a way to integrate many different pieces of data
- a system that manipulates and analyzes knowledge (via big ontologies, vocabularies)
- ……

and a mixture of all the above…

A More General View: what is the Semantic Web?

But let us start from the most basic:

Tim BL: add meanings to the current Web

- **code the meaning**
- **add it to the current Web**

A More General View: what is the Semantic Web?

But let us start from the most basic:

Tim BL: add meanings to the current Web

- **code the meaning**
- **add it to the current Web**
- **meaning = structured data**