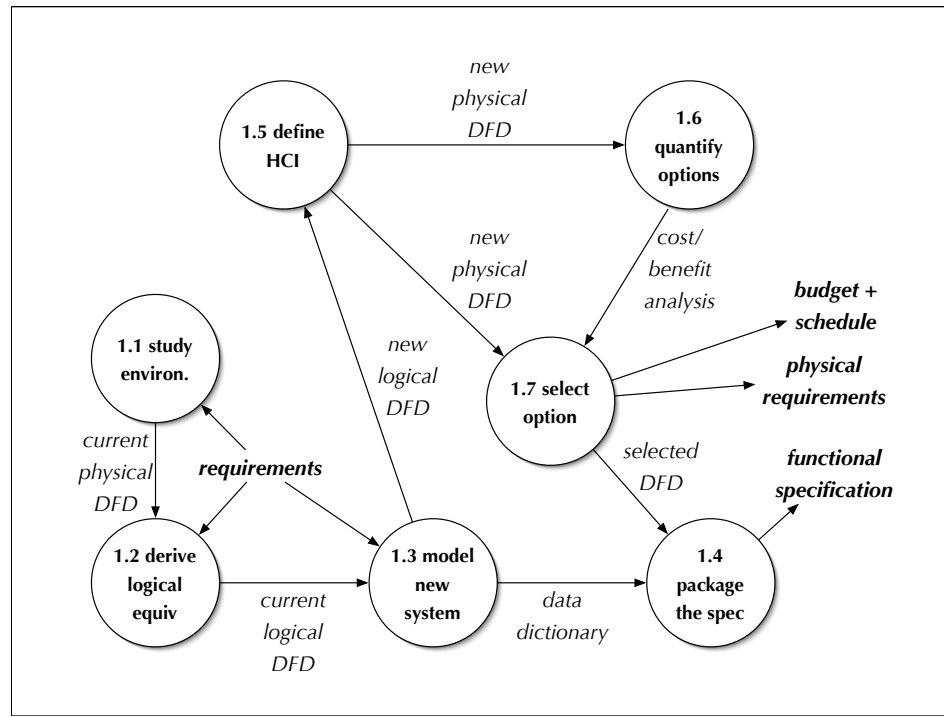


This is Diagram 0 of a Data Flow Diagram.



This is Diagram 1 for the same Data Flow Diagram.

```
data-dictionary := {data-entry}
```

```
data-entry := field-entry1  
            + field-entry2  
            +
```

These are Data Dictionary entries for the same DFD.

1.6 Quantify options

For each option given

- a. enumerate hardware requirements
- b. enumerate software requirements
- c. assign a cost value
to each requirement
- d. assign a benefit value
to each requirement
- e. report the net benefit
of the option

This is a Structured English description of bubble 1.6 in Diagram 1 of the DFD.

These are the
products
of analysis.

These are **tools** for documenting the analysis,
for sharing information,
for communicating on team and with client.

Analysis is the **study**
of a problem prior to
taking an action.

Where do the products come from?

How do we **do** analysis?

Read a book.

There are chapters of SE textbooks on analysis.
There are even whole books on “the requirements process”.

Example: **Mastering the Requirements Process**, by Robertson and Robertson.
It is a favorite of many SE profs, including ours.
So I read it.

It really is a good book. But...

- There is lot of material about products (use cases) and ideas (functional requirements)
- The advice for doing analysis reminds me of books on how to write better:

“Write clearly.”

Well, duh.

"Clear writing" is a goal, not an action.

Much advice on how to do analysis feels lot like this.

It is useful to get you thinking about what is important,
about what you need to do.

Recognizing that you
have a problem
is the first step
toward solving it.

Like the early steps in AA-like 12-step programs:

“1. We admitted we were powerless over alcohol --
that our lives had become unmanageable.”

“4. We have made a searching and fearless moral inventory of ourselves.”

That really is useful -- **to put you in the context to learn.**

Now you have to learn.



How do you learn?

Just do it.

What are the skills?

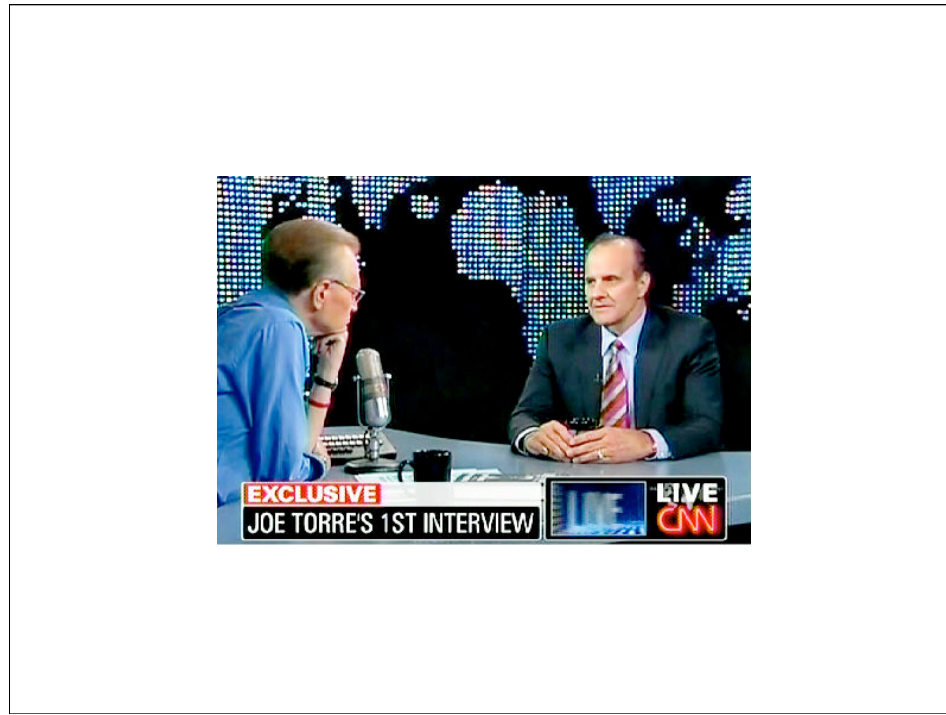
There are two basic skills. I mentioned them last time.

**You can observe a lot
just by watching.**



Observation

You **watch people work** and document what you see.



Interviewing.

You **talk to people about their area and their work**, and document what you hear.



Larry is known for not being all that well informed.
In that regard, you may want emulate Letterman or Leno or ...

But acting as if you don't know much can be helpful...

Be informed.

Study the domain, the company, anything you can to learn how things work **in general**.

Ask open questions.

Don't lead the respondent. Make them do the work.
It is the the best chance to hear truth and not a polite echo.

Ask again.

And again.

And again.

Ask the same question many times.

- of the one person, in one setting and across multiple settings
- of as many people as you can.

You will get different details, more details, different perspectives.

All of those give you more raw material, in order to assemble a more complete picture.

Doubt what you know.

You probably don't (know).

Doubt what you think you know.

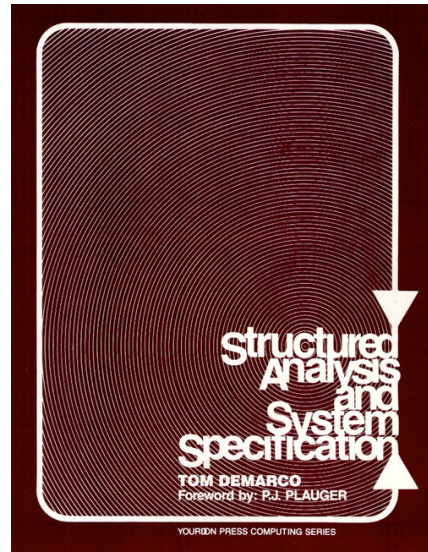
State assumptions. Invert them.

Violate the assumptions to see what people say.

Our friends in the humanities talk about deconstruction.

That can be a surprisingly useful skill in systems analysis, too.

Tom DeMarco



The source of the tools we have talked about this week.
The source of your reading assignment and homework.