

# knowledge representation

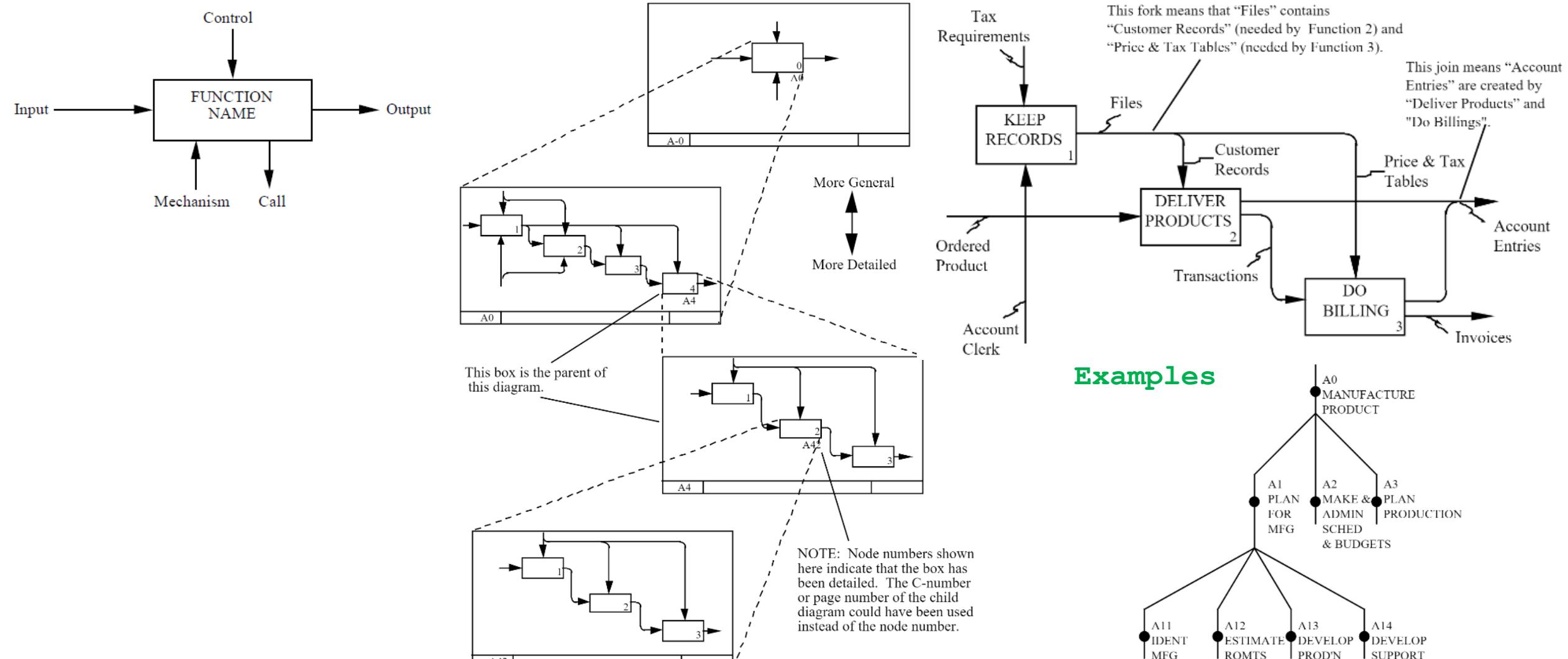
A	$\alpha$
B	$\beta$
$\Gamma$	$\gamma$
$\Delta$	$\delta$
E	$\varepsilon$
Z	$\zeta$
H	$\eta$
$\Theta$	$\theta$
I	$\iota$
K	$\kappa$
$\Lambda$	$\lambda$
M	$\mu$
N	$\nu$
$\Xi$	$\xi$
O	$\circ$
$\Pi$	$\pi$
P	$\rho$
$\Sigma$	$\sigma, \varsigma$
T	$\tau$
Y	$\upsilon$
$\Phi$	$\phi$
X	$\chi$
$\Psi$	$\psi$
$\Omega$	$\omega$



```
8 select 8
9 select 9
[Press Space for next page; Return to end.]
5 bash
SCREEN(1) SCREEN(1) SCREEN(1)
NAME
screen - screen manager with VT100/ANSI terminal
emulation
SYNOPSIS
screen [-options] [ cmd [ args ] ]
screen -r [[pid]][[host]]
screen -r sessionowner[[pid]][[host]]
DESCRIPTION
Screen is a full-screen window manager that multi-
plexes a physical terminal between several pro-
cesses (typically interactive shells). Each vir-
tual terminal can have its own window title and
background color. It can also be used to run mul-
tiple text programs to run at the same time, and
share a single interface productively. This enables
terminal sharing.
Manual page screen(1) line 1
2 man
nedozion * scrot gnuSCREEN.png
Welcome to GNU Screen - http://www.gnu.org/software/screen/
17:28 -!- Neo139 [*neo0] has joined #screen
17:28 -!- Topic for #screen: Welcome to GNU Screen -
http://www.gnu.org/software/screen/ |
http://savannah.gnu.org/projects/screen/ |
Mailing List Archive:
http://lists.gnu.org/archive/html/screen-users/ |
Wiki: http://aperiodic.net/screen/ | Manual:
http://www.gnu.org/software/screen/manual/ |
GSOC repository:
http://repo.or.cz/w/screen-lua.git?a=shortlog:h=refs/heads/
screen-scripting-soc
17:28 -!- Topic set by rudi_80 () (Set Aug 15 10:39:18 2009)
17:28 [Users #screen]
17:28 @ChenServ )| jake1 |
17:28 _n_ot_here )| Joyne |
17:28 esakura )| jdolson |
17:28 batrix )| jtrucks |
17:28 Beeny )| krisfremen |
17:28 blast_hardcheese )| levernu |
17:28 blueyed )| localghost |
17:28 Ceelum )| micosl |
17:28 caveet- )| minerale |
17:28 CIA-48 )| MissionCritical |
17:28 classin )| mmaticke |
17:28 contempt )| Neo139 |
17:28 Cybertinus )| nimred |
17:28 Deathvalley122 )| nuba |
17:28 der-onke1 )| OmikRoNIxZ |
17:28 diegoviola )| oskie |
17:28 dinwei )| Redo0 |
17:28
```

## Model.1: IDEF0 <[bh.github](#)> <[website](#)> <[how.to.doc](#)> <[how.to.video](#)> <[wikipedia](#)>

History: developed in 1970s by U.S. Air Force for improving manufacturing of fighting aircraft.  
Purpose: gather process inputs, outputs, resources, and constraints to methodically outline a system or specific process.



## Model .2: SWOT analysis <[bh.github](#)>

<[website](#)>

<[how.to.video](#)>

<[wikipedia](#)>

**History:** deep history and university dispute on ownership

**Purpose:** Apply this versatile model to personal situations and businesses to perform a high-level assessment of the value of performing system analysis work. It is also instrumental in helping to organize an unorganized group discussion quickly.



		<b>STRENGTHS</b>	<b>WEAKNESSES</b>
		Capabilities, resources or attributes that provide a competitive advantage	Capabilities, resources or attributes that need improvement
<b>INTERNAL</b>	Examples: production capacity, industry experience, financial resources, unique ingredients or packaging	Examples: lack of resources, limited experience, no marketing plan, no food safety or traceability program	
	<b>OPPORTUNITIES</b>		
<b>EXTERNAL</b>	Circumstances that if capitalized on could have a positive impact on the business	Circumstances that do or could have a negative impact on the business	<b>THREATS</b>
	Examples: favourable market trends, new technology, government policy changes, potential partnerships	Examples: regulatory changes; new trends, access to ingredients, exchange rate fluctuations	

### EXAMPLE SWOT ANALYSIS

<b>STRENGTHS</b>	<b>WEAKNESSES</b>
<b>Company:</b> <ul style="list-style-type: none"> <li>Food science expertise</li> <li>Sales have increased 57% in past six months</li> <li>Low rent due to location (Tofino)</li> <li>Utilizing low cost, approved, gluten-free, shared commercial kitchen*</li> </ul> <b>Product:</b> <ul style="list-style-type: none"> <li>Contains less sugar and more protein than competitive products</li> <li>Is on-trend (gluten-free, locally made)</li> <li>Is listed in local food stores</li> </ul>	<b>Company:</b> <ul style="list-style-type: none"> <li>Lack of financing</li> <li>Has no business or marketing plan</li> <li>High distribution cost due to location</li> <li>Utilizing shared commercial kitchen limits production capacity*</li> </ul> <b>Product:</b> <ul style="list-style-type: none"> <li>Higher priced than competitive products</li> <li>High ingredient costs</li> <li>Three month shelf life insufficient for retail sales</li> </ul>
<b>OPPORTUNITIES</b>	<b>THREATS</b>
<b>Company:</b> <ul style="list-style-type: none"> <li>High number of people with gluten sensitivities on Vancouver Island</li> <li>A higher volume in online purchases of natural health food products all over British Columbia</li> </ul>	<b>Company:</b> <ul style="list-style-type: none"> <li>CFIA's Food Labelling Modernization Initiative may impact the company</li> <li>Trend for gluten-free on downturn</li> <li>Crop failure has affected supply of the ancient grains</li> </ul>

\*Note that a strength can also be a weakness. In this example the cookies are produced in a low cost shared commercial kitchen, meeting food safety requirements and gluten-free status. While a strength for a newly established company, it is also a weakness in that production in a shared commercial kitchen limits production capacity.

## Model .2: SWOT analysis <[bh.github](#)><[website](#)><[how.to.doc](#)><[wikipedia](#)>

- Note: these models are for class quick reference use only.
- All students in IT.304 purchased and own "the decision book."

THE DECISION BOOK

### THE SWOT ANALYSIS

12

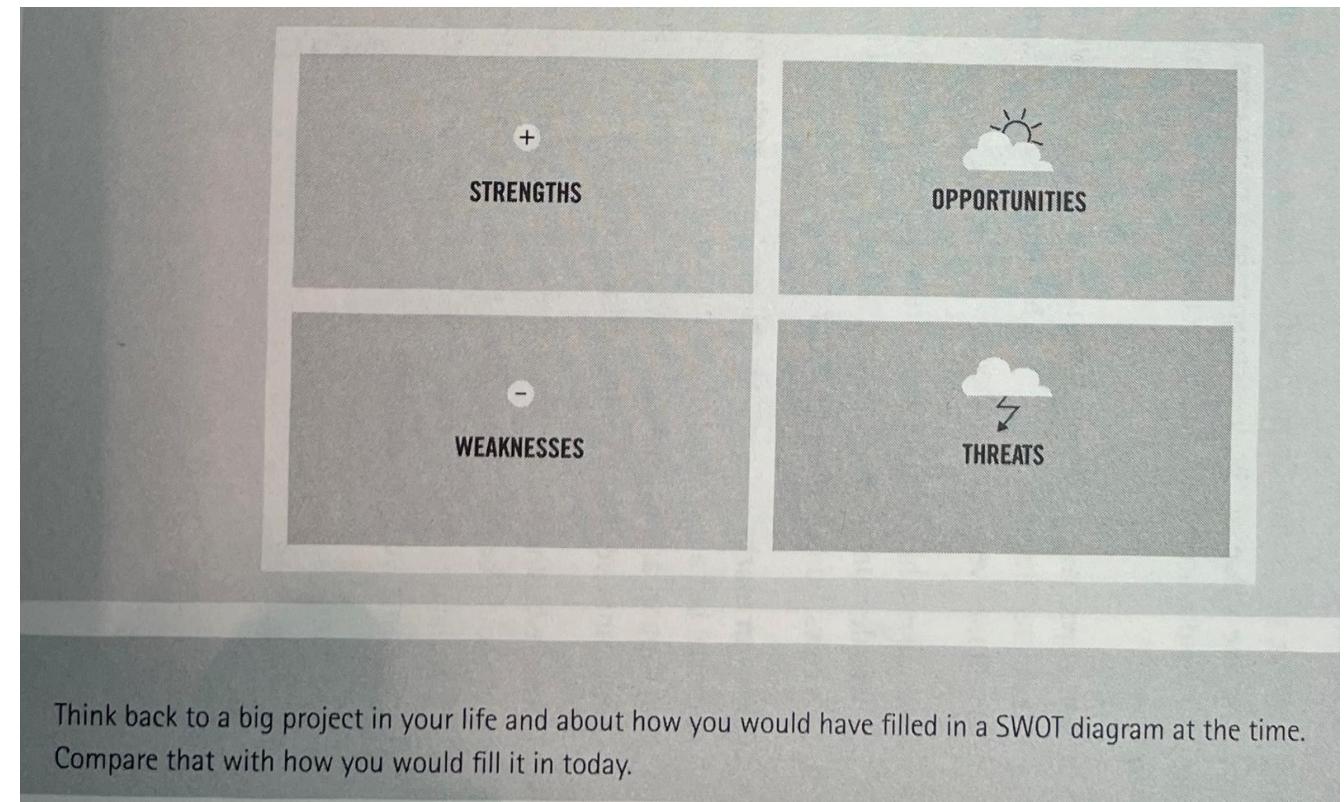
#### HOW TO FIND THE RIGHT SOLUTION

With SWOT analysis, you evaluate the Strengths, Weaknesses, Opportunities and Threats identified in a project. The technique is based on a Stanford University study from the 1960s which analyzed data from Fortune 500 companies. The study found a 35 percent discrepancy between the companies' objectives and what was actually implemented. The problem was not that the employees were incompetent but that the objectives were too ambiguous. Many employees didn't even know why they were doing what they were doing. SWOT was developed from the results of the study to help those involved in a project to gain a clearer understanding of it.

It is worth taking the time to think about each step of the SWOT analysis rather than just hastily fill it out. How can we emphasize our strengths and compensate for (or cover up) our weaknesses? How can we maximise opportunities? How can we protect ourselves against threats?

What is interesting about SWOT analysis is its versatility: it can be applied to business and personal decisions with equal success.

If you're not failing, you're not trying hard enough. *Gretchen Rubin*



- A) Retrieved from: Krogerus, M., Tschappeler, R., and Piennig, J. (2018). *The decision book: fifty models for strategic thinking*. ISBN-10: 0393652378, ISBN-13, 978-0393652376.
- [Amazon.com: The Decision Book: Fifty Models for Strategic Thinking: 9780393652376: Krogerus, Mikael, Tschäppeler, Roman, Piening, Jenny: Books](#)



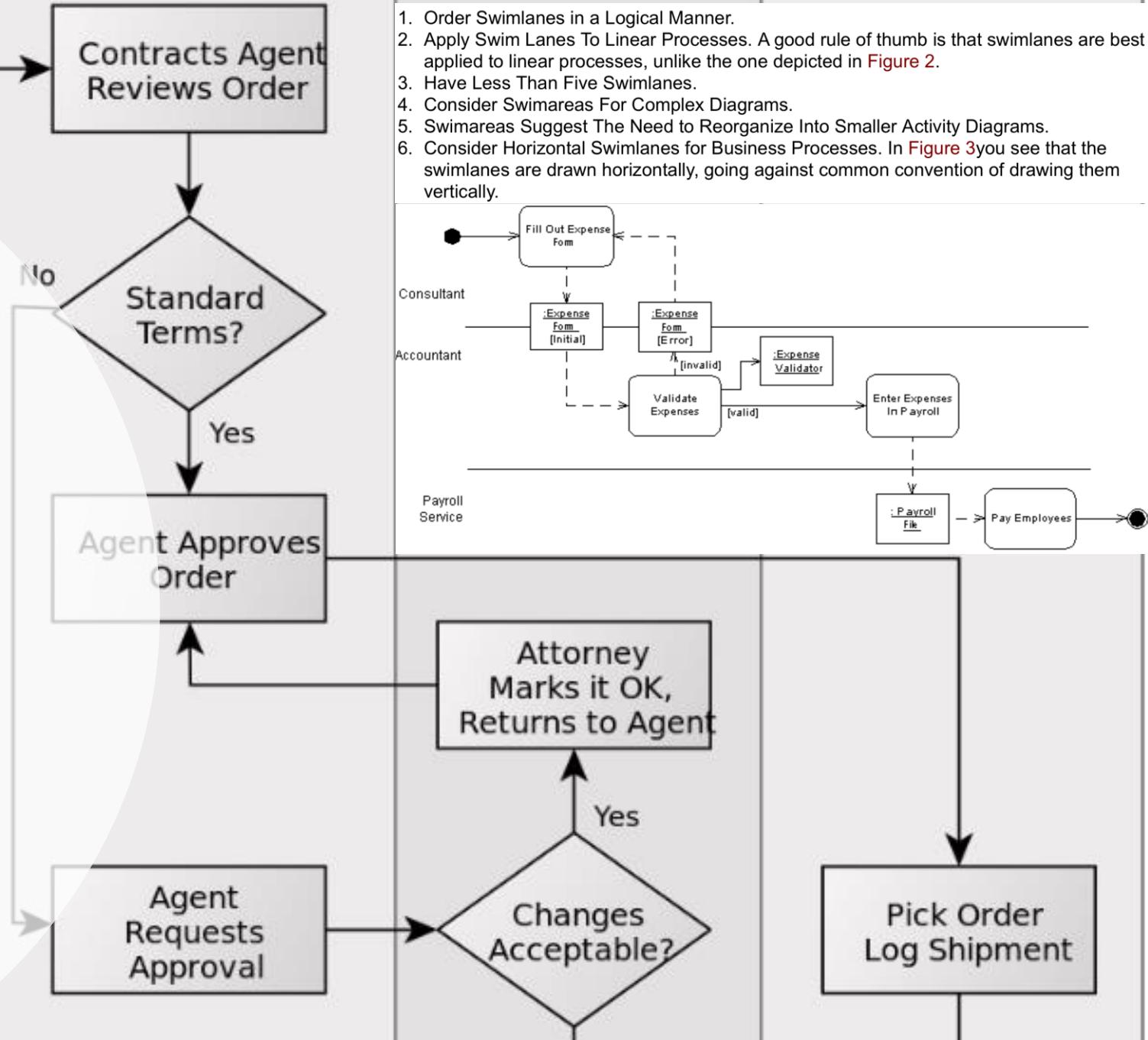
### Model.3: Swimlane diagram

[bh.github](https://bh.github.io)  
[website](#)  
[how.to.doc](https://how.to.doc)  
[how.to.in.visio](https://how.to.in.visio)  
[wikipedia](#)

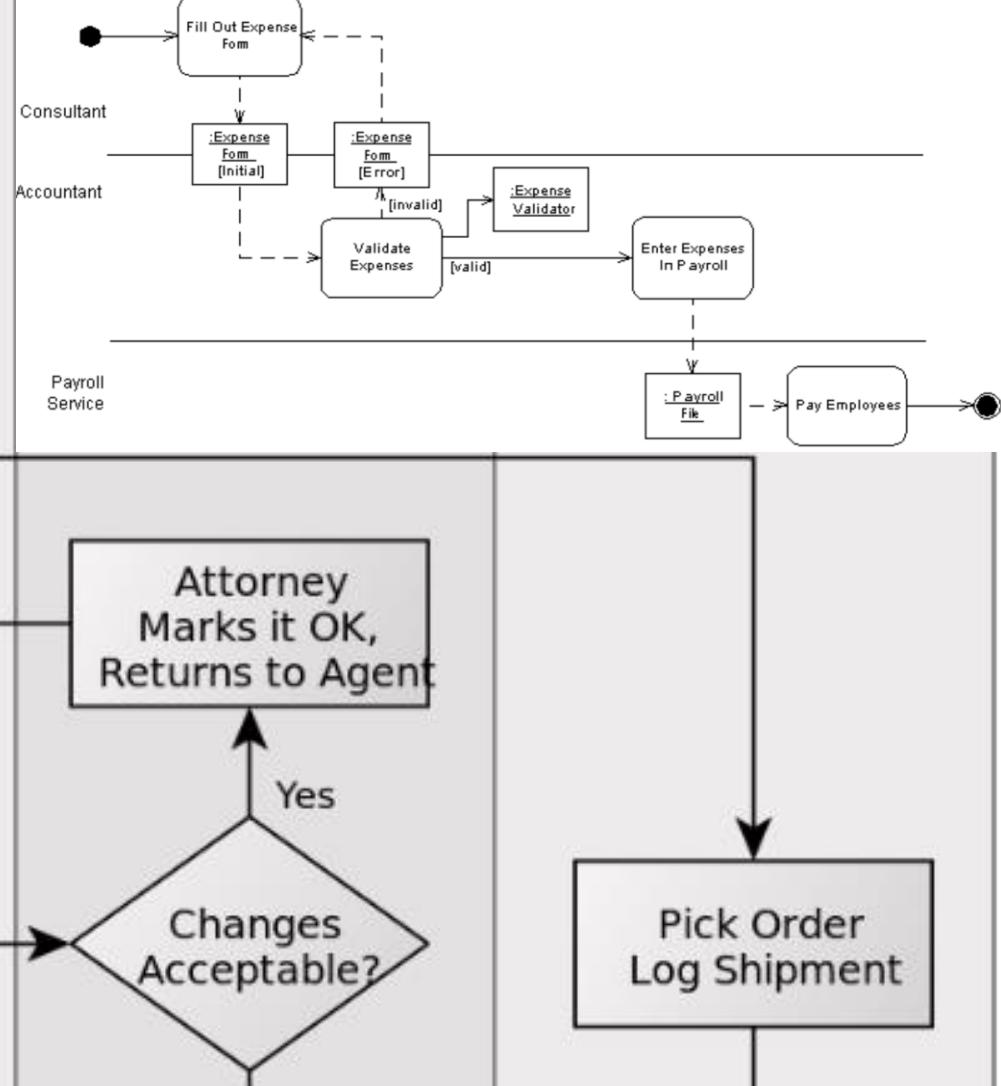
---

**History:** dating back to the 1940s, computerized in 1993 by Igraphx, and widely available via Microsoft Visio.

**Purpose:** use horizontal or vertical gradating color bars to demarcate business lines illustrating system inputs, activities, and decisions connected with arrows



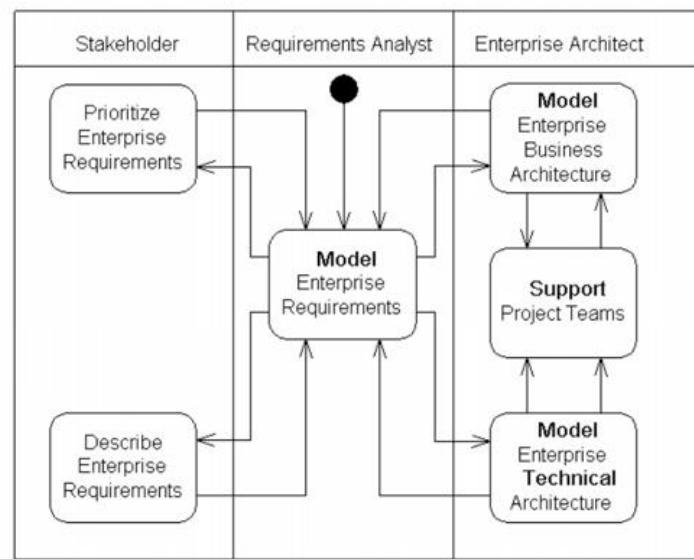
1. Order Swimlanes in a Logical Manner.
2. Apply Swim Lanes To Linear Processes. A good rule of thumb is that swimlanes are best applied to linear processes, unlike the one depicted in [Figure 2](#).
3. Have Less Than Five Swimlanes.
4. Consider Swimareas For Complex Diagrams.
5. Swimareas Suggest The Need to Reorganize Into Smaller Activity Diagrams.
6. Consider Horizontal Swimlanes for Business Processes. In [Figure 3](#) you see that the swimlanes are drawn horizontally, going against common convention of drawing them vertically.



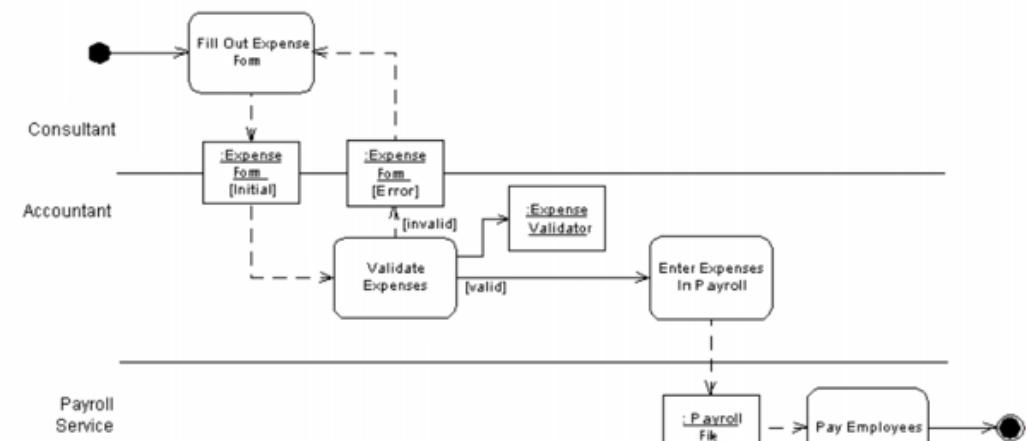
## 6. Swimlane Guidelines

A swimlane is a way to group activities performed by the same actor on an activity diagram or to group activities in a single thread. [Figure 2](#) includes three swimlanes, one for each actor.

**Figure 2. A UML activity diagram for the enterprise architectural modeling (simplified).**



**Figure 3. Submitting expenses.**



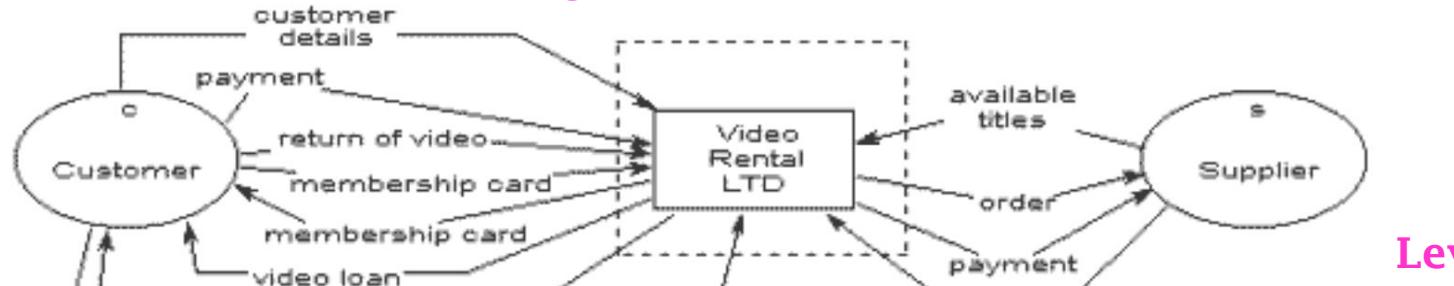
1. Order Swimlanes in a Logical Manner.
2. Apply Swim Lanes To Linear Processes. A good rule of thumb is that swimlanes are best applied to linear processes, unlike the one depicted in [Figure 2](#).
3. Have Less Than Five Swimlanes.
4. Consider Swimareas For Complex Diagrams.
5. Swimareas Suggest The Need to Reorganize Into Smaller Activity Diagrams.
6. Consider Horizontal Swimlanes for Business Processes. In [Figure 3](#) you see that the swimlanes are drawn horizontally, going against common convention of drawing them vertically.

## 7 Action-Object Guidelines

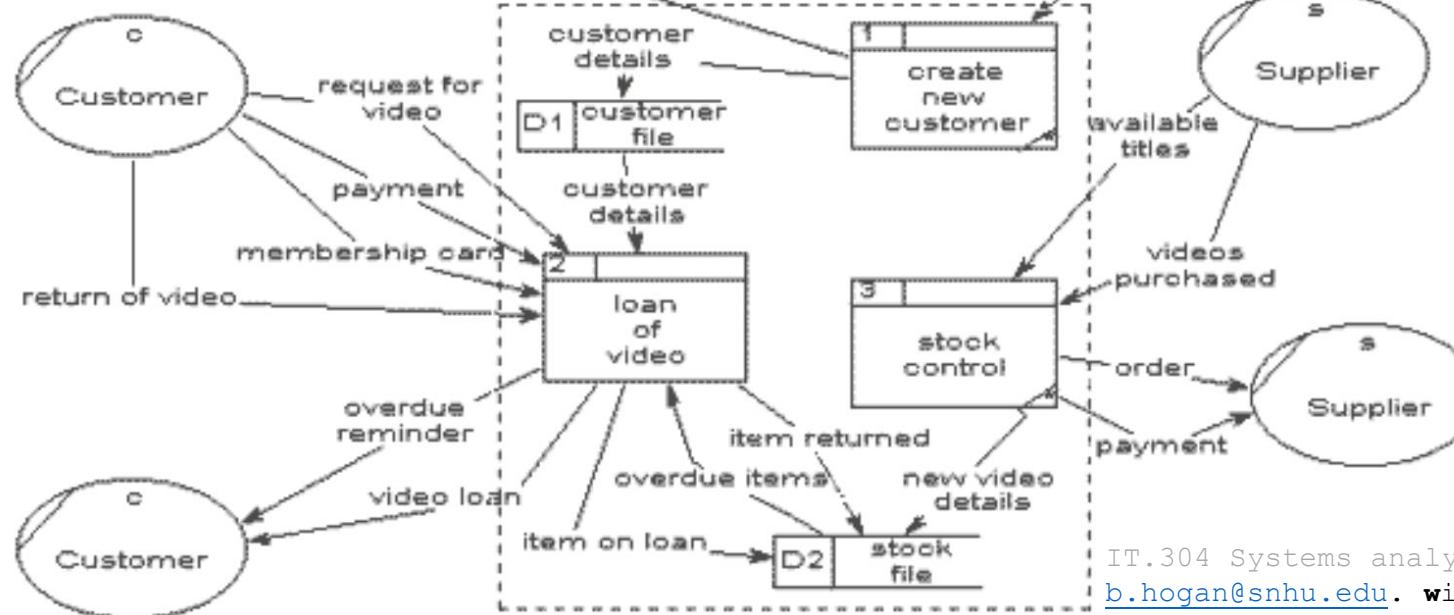
Activities act on objects, In the strict object-oriented sense of the term an action object is a system object, a software construct. In the looser, and much more useful for business application modeling, sense of the term an action object is any sort of item. For example in [Figure 3](#) the *ExpenseForm* action object is likely a paper form.

**Model .4: Data Flow Diagraming** <[sparx-models](#)> <[website](#)> <[how.to.doc\[VG\]](#)> <[how.to.video](#)> <[wikipedia](#)>  
 Purpose: process of representing simplified data transactions to help process and stakeholder owners agree on scope and boundaries of a systems analysis and design reengineering effort. Level 0 is the highlevel context. Key tasks are detailed in level 1 indicating storage medium and transactions. Level 2 specifies transactions and their storage.

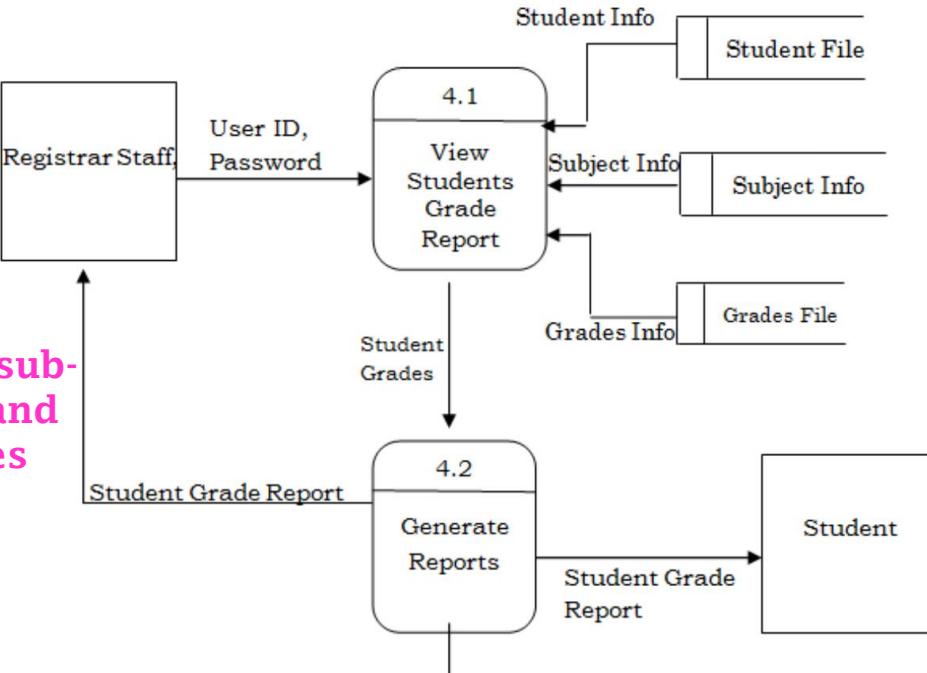
## Level 0 - DFD - Context Diagram



**Level 1  
DFD -  
Details + 1**



**Level 2  
DFD - main sub-  
processes and  
data stores**

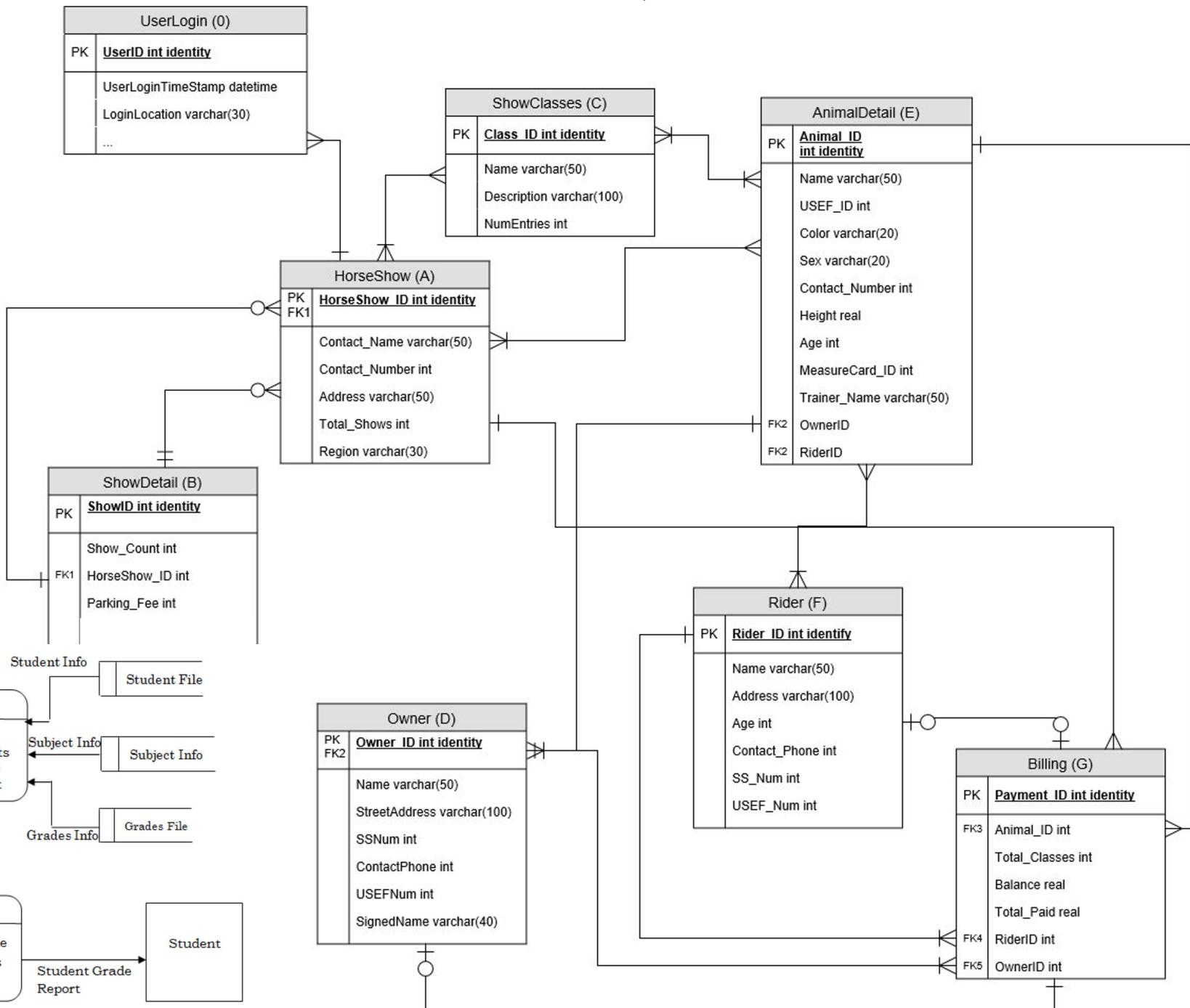


- **Context diagrams** — context diagram DFDs are diagrams that present an overview of the system and its interaction with the rest of the “world”.
- **Level 1 data-flow diagrams** — present a more detailed view of the system than context diagrams, by showing the main sub-processes and stores of data that make up the system as a whole.
- **Level 2 (and lower) data-flow diagrams** — a major advantage of the data-flow modelling technique is that, through a technique called “levelling”, the detailed complexity of real world systems can be managed and modeled in a hierarchy of abstractions. Certain elements of any dataflow diagram can be decomposed (“exploded”) into a more detailed model a level lower in the hierarchy.

# Model.5 Data Flow Diagram combined with data table normalization:<bh.github>

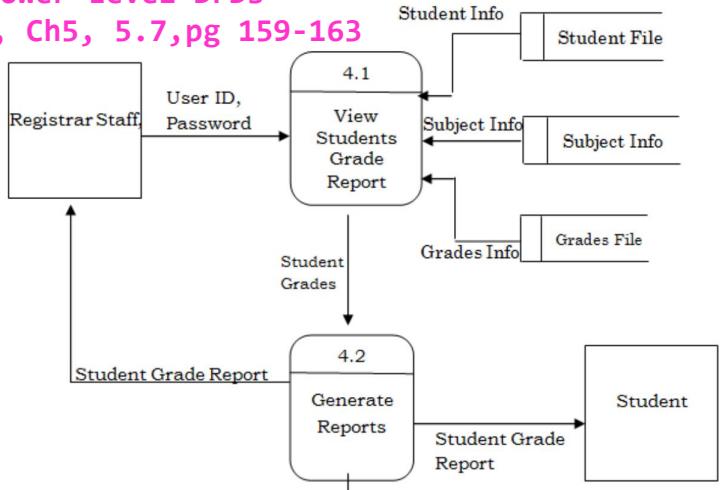
History: abc

Purpose: abc



Level 2-Data Flow Diagram  
Lower Level DFDs

Tilley, Ch5, 5.7, pg 159-163

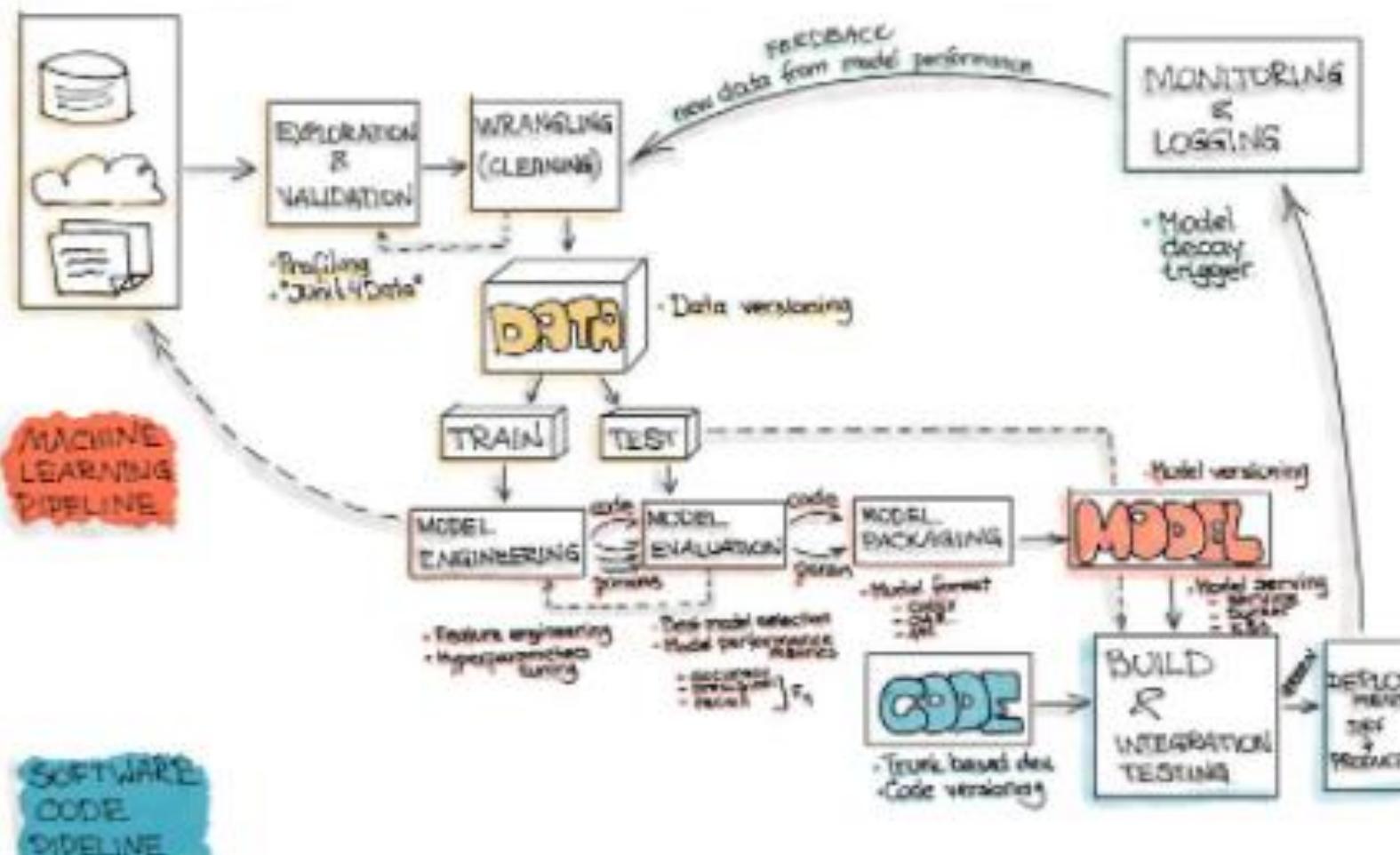


relatio  
nal  
databas

## Model.6: Machine Learning <website> <how.to.doc> <josh.Gordon.how.to.video> <wikipedia>

Purpose: data pipes execute in parallel or time-sliced fashion providing new clickbait or similar transaction behavior for an established ML model. New data kicks off further analysis as captured in the graphic. From a high level, if more people like you find a new Amazon "whizzy" somewhat of a "tizzy," then ML should indicate your profile as a potential buyer to provide you a link on your search screen. Welcome to modern business 101.

### MACHINE LEARNING ENGINEERING



#### An Overview of the End-to-End Machine Learning Workflow

Generally, the goal of a machine learning project is to build a statistical model by using collected data and applying machine learning algorithms to them. Therefore, every ML-based software includes three main artifacts: Data, ML Model, and Code.

Corresponding to these artifacts, the typical machine learning workflow consists of three main phases:

1. Data Engineering: data acquisition & data preparation,
2. ML Model Engineering: ML model training & serving, and
3. Code Engineering :integrating ML model into the final product.

The Data Engineering pipeline includes a sequence of operations:

1. Data Ingestion
2. Exploration and Validation
3. Data Wrangling
4. Data Labeling
5. Data Splitting

Model Engineering  
The Model Engineering pipeline includes a number of operations that lead to a final model:

1. Model Training
2. Model Evaluation
3. Model Testing
4. Model Packaging

Model Deployment  
The final stage of the ML workflow is the integration of the previously engineered ML model into existing software. This stage includes the following operations:

1. Model Serving
2. Model Performance Monitoring
3. Model Performance Logging

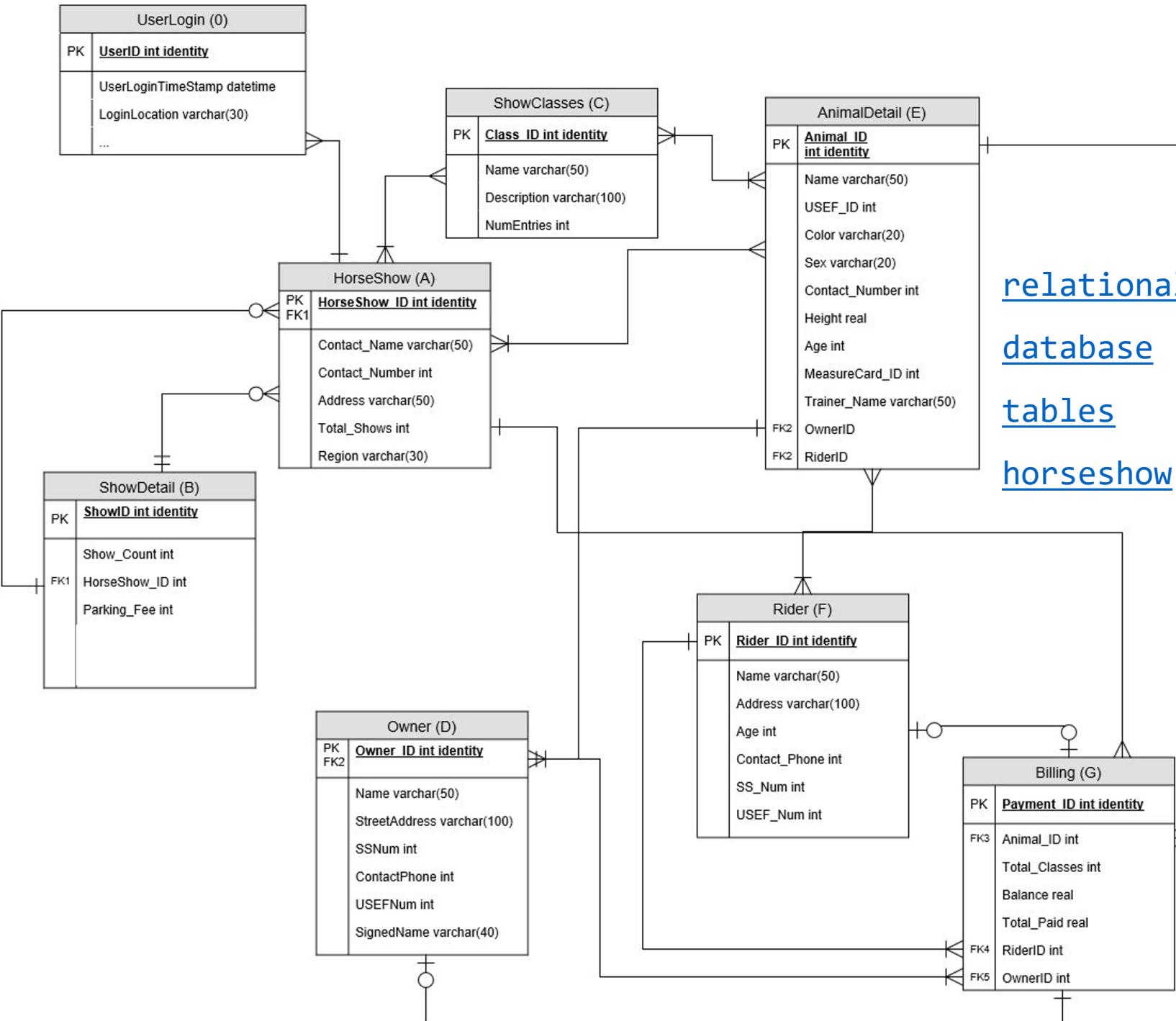
from abhishek\_prasad but the linked in post page was torn  
IT.304 Systems analysis, design, and implementation planning. snhu.edu  
[b.hogan@snu.edu](mailto:b.hogan@snu.edu) [bh.github](https://github.com/bhogan) Note: Wikipedia is an information only reference. It is not an academic reference.

## Model.7 Database Normalization:

<bh.github> <website> <how.to.doc>  
 <how.to.video> <wikipedia>

History: Kimball, Inmon

Purpose: decompose a data structure  
 into natural structures, ie no  
 redundant data.



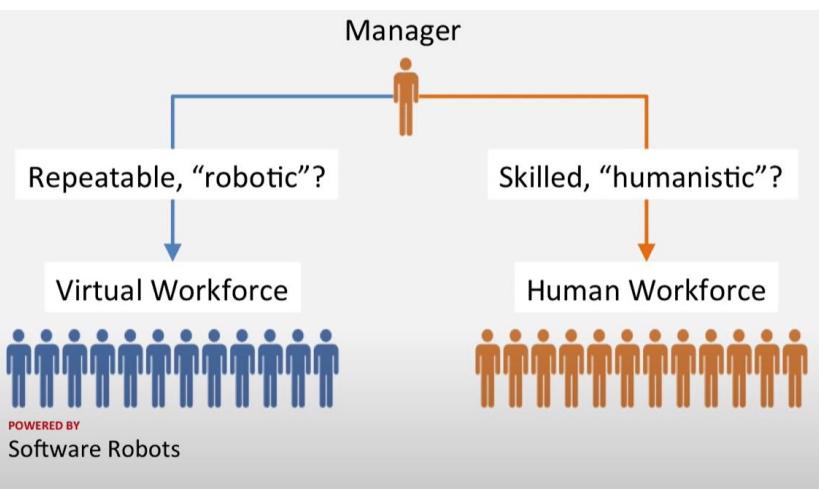
[relational  
database  
tables  
horseshow](#)

**Model.9: Robotic Process Automation (RPA)** <[bh.github](#)> <[what.is](#)> <[how.to](#)> <[IBM.vid.2022](#)> <[wikipedia](#)>

**History:** Recent form of business process automation using software robots (bots) and artificial intelligent AI-digital workers to perform tasks user do by watching users perform the tasks in the GUI and then replicate it.

**Purpose:** Use of IT widgets to read emails, forms, to replicate human behavior and eliminate the “digital tape and glue” of cross updating amongst mix-matched current and legacy system applications. AI and machine learning can also help figure out and classify what the newly received information is and suggest where it should go instead of a human orchestrating it because systems have been codified, labeled, and connected to AI-digital workers.

## What are people doing in Reality?



https://www.ibm.com/docs/en/cloud-paks/cp-biz-automation

IBM Documentation Search in IBM Cloud Pak for Business Automation

IBM Cloud Paks / IBM Cloud Pak for Business Automation /

IBM Cloud Pak for Business Automation documentation

Change product →

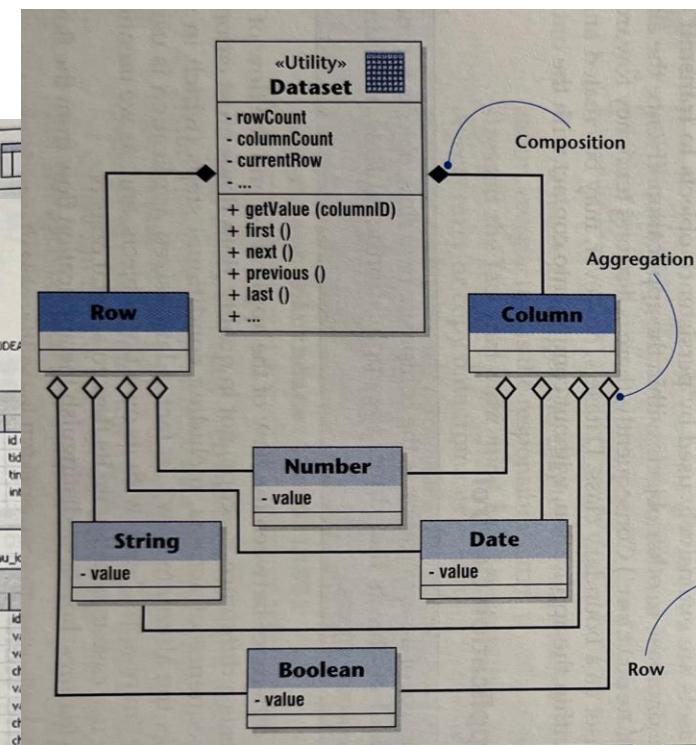
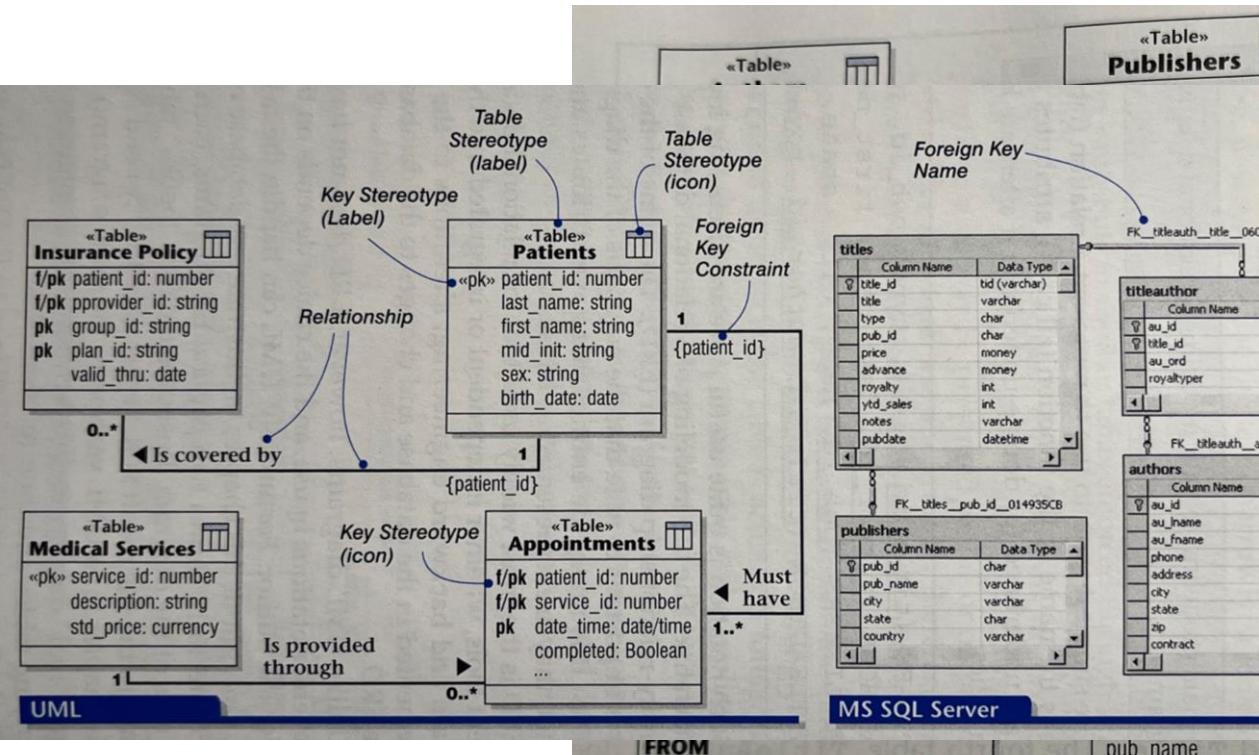
IBM Cloud Pak for Business Automation 22.0.1	IBM Cloud Pak for Business Automation 21.0.3	IBM Cloud Pak for Business Automation 21.0.1 & 21.0.2	IBM Cloud Pak for Automation 20.0.x
Version 22.0.1	Version 21.0.3	Version 21.0.x	Version 20.0.x
IBM Cloud Pak for Automation 19.0.x	IBM Digital Business Automation for Multicloud 18.0.x	Version 18.0.x	
Version 19.0.x			

# Model.8: Entity Relationship Diagram (ERD)

[bh.github](http://bh.github) [what.is](http://what.is) [how.to](http://how.to) [IBM.vid.2022](http://IBM.vid.2022) [wikipedia](http://wikipedia)

History: Rece

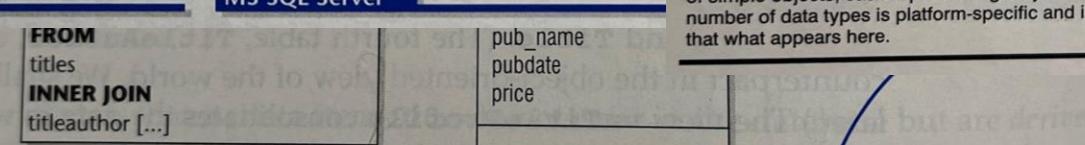
Purpose: U



A table illustrating the **«Utility» Dataset** structure:

Column	ID	First Name	Last Name	Birth Date
01	05630102	Albert	Perez	05/22/1975
02	76530991	Arthur	Perkins	04/15/1982
03	33201985	Benjamin	Perl	11/12/1968
04	97654300	Bernard	Perl	10/15/1989
05	85087365	Christopher	Perlman	12/21/1980
06	76560999	Fey	Perlman	11/16/1979
07	20056789	James	Perlmutter	02/21/1979
08	33379087	Jeanne	Perrot	04/01/1992
09	49087543	John	Perrot	05/12/1983
10	79988832	Kim	Perry	08/17/1986
11	19800874	Mathilda	Perry	02/02/2002
12	55666987	Mark	Perry	09/15/1977
13	98546773	Mary	Perry	06/04/1976
14	65734297	Matthew	Person	11/01/1986
15	44598732	Norma	Person	07/07/1988
16	54398002	Sara	Person	06/08/1982

Each column in the dataset represents one attribute shared by all items in the dataset. (Note that what is shared is the **attribute**, not the **value** of the attribute.) A row represents the collection of all attributes for one distinct item. Each cell is an object that exposes the value of one attribute for the item.

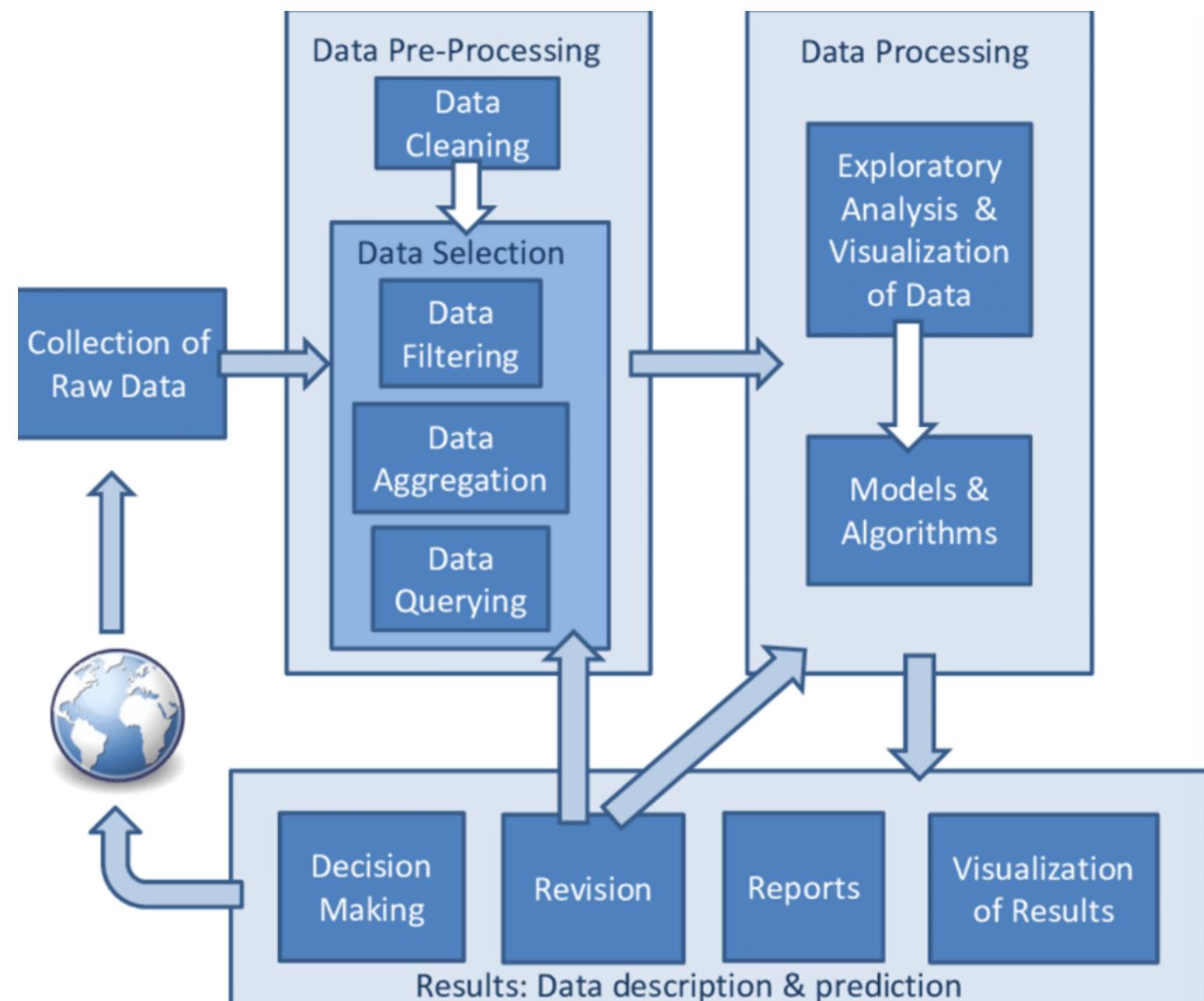


A table showing the results of the SQL query:

	title	au_lname	au_fname	pub_name	pubdate	price
1	But Is It User Friendly?	Carson	Cheryl	Algodata Infosystems	1991-06-...	22.9500
2	Computer Phobic AND Non-Ph...	MacFeather	Stearns	Binner & Hardley	1991-10-...	21.5900
3	Computer Phobic AND Non-Ph...	Karsen	Livia	Binnet & Hardley	1991-10-...	21.5900
4	Cooking with Computers: Su...	O'Leary	Michael	Algodata Infosystems	1991-06-...	11.9500
5	Cooking with Computers: Su...	MacFeather	Stearns	Algodata Infosystems	1991-06-...	11.9500
6	Emotional Security: A New ...	Locksley	Charlene	New Moon Books	1991-06-...	7.9900
7	Fifty Years in Buckingham ...	Blotchet-Halls	Reginald	Binnet & Hardley	1991-06-...	11.9500
8	Is Anger the Enemy?	Ringer	Anne	New Moon Books	1991-06-...	10.9500

History: abc

Purpose: abc



History: abc

Purpose: abc



## Machine Learning Recipes with Josh Gordon

10 videos • 1,153,971 views • Last updated on Apr 8, 2022



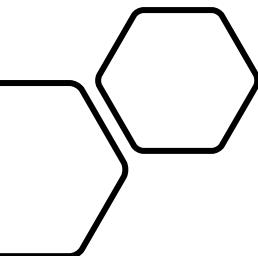
Josh Gordon is cooking up some Machine Learning models from scratch. Learn how to use open source libraries to create ML recipes that can be adapted to a wide range of circumstance. Let's get cooking!



Google Developers

SUBSCRIBE

- |    |  |  |
|----|--|--|
| 1  | (ML) Hello World 6:53                            | Hello World - Machine Learning Recipes #1<br>Google Developers   |
| 2  | (ML) Decision Tree 6:42                          | Visualizing a Decision Tree - Machine Learning Recipes #2<br>Google Developers                         |
| 3  | (ML) What makes a good feature? 5:41             | What Makes a Good Feature? - Machine Learning Recipes #3<br>Google Developers                          |
| 4  | (ML) Let's Write a Pipeline 7:54                 | Let's Write a Pipeline - Machine Learning Recipes #4<br>Google Developers                              |
| 5  | (ML) writing a classifier 8:44                   | Writing Our First Classifier - Machine Learning Recipes #5<br>Google Developers                        |
| 6  | (ML) Train Your Own Image Classifier 7:07        | Train an Image Classifier with TensorFlow for Poets - Machine Learning Recipes #6<br>Google Developers |
| 7  | (ML) Classifying handwritten digits 7:01         | Classifying Handwritten Digits with TF.Learn - Machine Learning Recipes #7<br>Google Developers        |
| 8  | (ML) Let's Write a Decision Tree Classifier 9:53 | Let's Write a Decision Tree Classifier from Scratch - Machine Learning Recipes #8<br>Google Developers |
| 9  | (ML) Feature Engineering with TensorFlow 7:38    | Intro to Feature Engineering with TensorFlow - Machine Learning Recipes #9<br>Google Developers        |
| 10 | (ML) Getting Started with Weka 9:24              | Getting Started with Weka - Machine Learning Recipes #10<br>Google Developers                          |



[https://www.youtube.com/playlist?list=PLOU2XLYxmsIlliuiBfYad6rFYQU\\_jL2ryal](https://www.youtube.com/playlist?list=PLOU2XLYxmsIlliuiBfYad6rFYQU_jL2ryal)