Antikythera Publications

DATABASE DESIGN NOTE SERIES

An In-Line Font for Simple ERD Creation

Prepared by: F. Oberle and S. L. Weiss

In order to facilitate the use of in-line Entity Relationship Diagrams in documents where numerous but fairly simple examples are required, and where it is inconvenient or time-prohibitive to create such diagrams with an external application and then import them into the documents, a free, sharable, special purpose True-Type Font was created to cover the most commonly encountered symbols.

18 June 2014

THIS PAGE IS INTENTIONALLY BLANK

Database Design Note Series – an In-Line ERD Font

Preface – Purpose

When designing a business database, any associations between Entities must be stated as True Normalized Propositions, and are thus formed from a Subject (a singular Common Noun), one of two forms of the verbs "to be" or "to have" ("is," "must be" or "must have" for a mandatory association, or "may be" or "may have" for an optional association), and an Object (another Common Noun). The verb+object combination forming the Proposition's Predicate states some fact about the Subject. It is often useful to show these Propositions in a graphic form – usually an Entity-Relationship-Diagram – using various "standard" diagramming conventions.

If you are not familiar with the use of formal Propositions, you are invited to consult some of the classics in the field of Predicate Logic, such as those by Lewis Carroll[1] or the book *Business Database Triage*.

In order to facilitate the use of in-line Entity-Relationship Diagrams (ERDs) in text documents or presentations where numerous but fairly simple examples are required, and where it is inconvenient or time-prohibitive to create such diagrams with an external application and then import them into the documents, a free, sharable, special purpose True-Type Font was created to cover the most commonly encountered symbols.[2]

With this font, generation of diagrams such as those shown below can be accomplished simply by selecting the font and typing them directly:

Slightly more interesting concepts, such as the two types of exclusivity arc shown next are also possible:
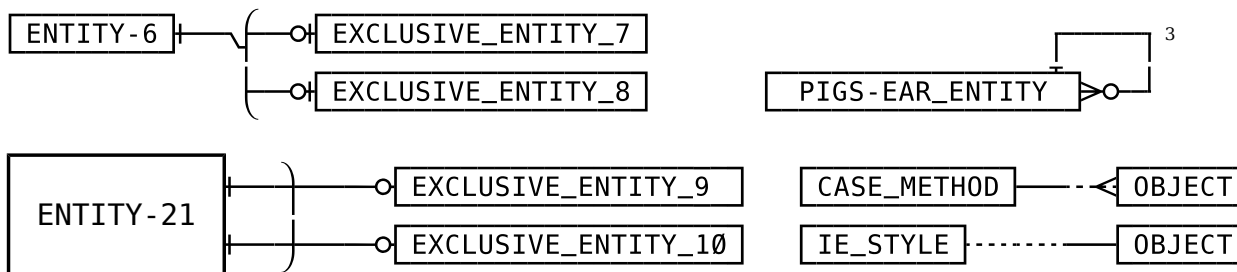
Of course, for use in a word processor or other software providing adjustable vertical spacing, multi-line diagrams such as the exclusive arcs and reflexive associations (the "pig's ear") shown above will benefit from setting any line-to-line or paragraph spacing to zero. In some cases, unfortunately, vertical gaps can't be entirely removed, but the font still communicates the sense of the propositions clearly.
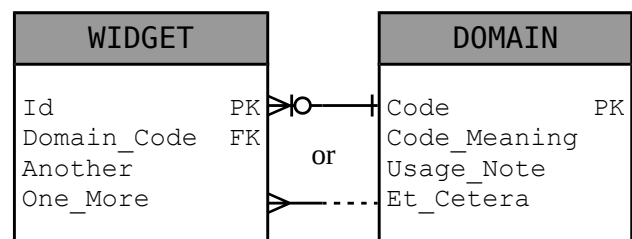
Same Relationship: Different Conventions
A Widget's Domain must be identified by 1 Domain Code
A Domain may be used on 1 or many Widgets
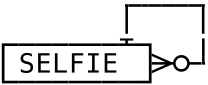
The ERD_A font provides the most commonly encountered symbols (some of which, unfortunately, are logically dubious) from several commonly used ERD conventions. It is entirely up to the user, however, to ensure that symbols from one convention are not inappropriately intermingled with those of another!

---

1 Lewis Carroll; "Symbolic Logic" (1896). Available as "Mathematical Recreations of Lewis Carroll" which contains both "Symbolic Logic" (1896) and his "Game of Logic" (1887) in one volume: Dover paperback ISBN: 0-486-20492-8.
2 The ERD_A Font accompanying this Design Note is licensed under the SIL Open Font License, Version 1.1 , available with a FAQ at http://scripts.sil.org/OFL
3 Specific instructions for typing all the diagrams is given in "Using the Font" and "Getting Cute" below.

Using the Font

The keyboard mappings are shown in the table on page 6, so it is only necessary to install the font and then select it for use with the ERD Proposition you wish to type. There is a keyboard illustration on page 12 showing the positions of all the keys and the characters/glyphs they produce. Here are some examples of character usage:

Type this:	To get this:	Comments
`[employee]`	EMPLOYEE	The "[" and "]" characters create the left and right edges respectively of an Entity box. All characters used for anything that appears inside the box must be typed in lower case, although they are all displayed as upper case.
`"employee#`	EMPLOYEE	The double quotation mark and pound/hash symbols provide box edges with reduced internal spacing. To account for word processors and similar applications that "autocorrect" the quote character, the ERD_A font maps any "smart quote" characters to the narrow box edge as well.

`[:employee:]`
 	
`[+employee+]`
 	
`[=employee=]`
	EMPLOYEE
	EMPLOYEE
	EMPLOYEE
		Additional Internal Space within the Entity Box can be added using a variety of characters. See the section titled "Entity Name Space Characters" in the "Special Characters in Alternate Widths" chart on page 9 for a complete listing.

`E[glyphs-1]I`
 	
`S[glyphs_2]K`
	GLYPHS-1
	GLYPHS_2
		The most straightforward way of reproducing a Proposition graphically is to use the variety of "pre-built" connectors such as the capitals "E" "I" "S" and "K" used here. See the keyboard illustration on page 12 for these. The underscore and hyphen characters both have top and bottom edges and are intended for use in an Entity name.

`;;;<[glyphs]>;;;`
 	
`;;<[glyphs]>;;`
 	
`;<[glyphs]>;`
	GLYPHS
	GLYPHS
	GLYPHS
		To customize the appearance of the Propositions, there are characters defined for most atomic elements of the pre-built connectors. In this case, the discrete characters are used to duplicate the example above, and to produce the same Proposition with different length connecting lines.

`A[employee]L`
 	
`;;*.[employee].*.,;`
 	
`;,.*,[employee],*.,;`
 	
`;.*;[employee];*.;`
	EMPLOYEE
	EMPLOYEE
	EMPLOYEE
	EMPLOYEE
		The pre-built "A" and "L" glyphs, for example, have a 1 unit line between the edge contacting the entity box and the circle. Discrete connecting lines are available in lengths of 1, 2, and 4 units ("." "," and ";" respectively) if customization is desired. See the section titled "Horizontal Connector Characters" in the "Special Characters in Alternate Widths" chart on page 9 for a complete listing.

`"party,|;\{O"perso`
 	
`ZZ~~` (O"organ`
	PARTY — PERSO / ORGAN	A co-divisional exclusivity arc (sometimes incorrectly referred to as an "exclusive-or" association) is written on two lines using the complementary characters "{" and "(".

A combined cell containing "AC" with a border, followed in the next cell by:
`;|*};;,*|;[alt-1]`
`;|*);;,*|;[alt-2]`
	AC — ALT-1 / ALT-2	This shows the more commonly (but often incorrectly) used alternative exclusivity arc, in which there are exclusive references from a single Entity to either of two (but not both) alternatives targets.

Type this:	To get this:	Comments
`Z~~~~`';,;;;;'`  `[selfie&]K^`		Finally, reflexive (self) references (pig's ears) can be typed: the "&" character after the second letter "e" is a one unit table border with a raised connection. The " ' " and " ^ " characters provide vertical lines extending up (top row) and down (bottom row) from a horizontal connector line respectively.

Getting Cute

The ERD_A font is designed primarily for short, one-line expressions of the graphical representation of the Propositions and Predicates required for logical database design. Nonetheless, somewhat more elaborate diagrams may be created using tables in most word processors and spreadsheets, such as those in LibreOffice, Open Office, Microsoft Office, or similar. The last example on the previous page is displayed in more detail below showing how a simple table can be used – the first column is set to have centered text and a cell border width of 1.75 points, although this will vary depending on the size of the ERD_A font used and your personal taste. The second column has its internal margins set to 0, to insure that the ERD glyphs directly contact the "entity" box in the first column.

The freely available font DejaVu Sans Mono is used in the cell with the border to exactly match the characters in the ERD_A font, although any similar font (such as one from the commonly available Arial family) will work just as well in most circumstances. The associations and two "entities" (11 and 12) on the right are written using the ERD_A font – in this example at a size of 20 points – but this again is a matter of choice.
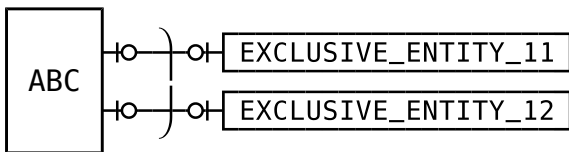
Getting Cuter:

See page 9 for instructions on vertical examples of using the font.

The more detailed diagram on the right is an example of a 3 column by 2 row table, with the borders removed from the second column; the second column's internal margins are set to 0.0" to permit the relationship symbols to abut the first and third column edges.
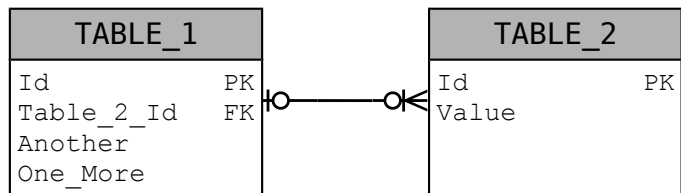
The center column's width depends, of course, on the size of the ERD_A font chosen for the symbols.
In this example, the second row of the center column contains the characters OGT (the "G" provides additional length). By using the pre-built relationship symbols (all of which are the same width: 15 units), you can establish a consistent width when using identical tables. This permits examples to be created quite quickly without requiring the use of a graphic modeling tool.

Shading was added to the top row of the first and third columns for effect.

Note that if this font is used in any editable document (such as a *.doc or *.odt) intended to be shared with others, each user must have the font installed. If generating pdf files for common distribution, it is recommended that the font be embedded in the pdf to preclude the need for the ERD_A font to be present. How this is accomplished depends on the application in use.

Keyboard Mappings

The characters are mapped to the lower USASCII positions for two reasons: 1) There is no Unicode area defined for such a character set and 2) it is simpler to use in practice, since having the characters in those positions doesn't require the use of any input methods – just a basic keyboard. A template for the keyboard is shown on page 12.

dec	α	ERD	x	Unit Width	Description	ERD_A-TYPEFACE-CHARACTERS	Comments
32			1	128	Space character (not part of Entity Name)		[4] See #88, 90, 96, 126
33	!	│	6	768	Used for connecting upper and lower multi-line Exclusivity Arcs		See #40, 123
34	"	[	3	384	Entity-Box Alternate (narrow) Left Border		[5] Also see #91
35	#	]	3	384	Entity-Box Alternate (narrow) Right Border		See #93
36	$	╲	6	768	Horizontal Exclusivity Arc Angled Transition (Top)		See #37, 47, 92
37	%	╱	6	768	Horizontal Exclusivity Arc Angled Transition (Bottom)		See #36, 47, 92
38	&	⊼	5	640	1 unit horizontal line with upper connector for reflexive arc		aka: pig's ear
39	'	╷	1	128	Character Space with lower connector for pig's ear		[5] See #94
40	(	├	6	768	Horizontal Exclusivity Arc Connector (used with "{" and "!")		See #123 for Top
41	)	┤	6	768	Horizontal Inclusion Arc Connector (used with "}" and "?")		See #125 for Top
42	*	⦾	6	768	Connector Symbol "0" or "optional"		See @ (64)
43	+	┊	2	256	2 unit Space character (as part of an Entity Name)		[6] See #58, 61
44	,	-	2	256	2 unit horizontal connection line		[7] See #46, 59, 71, 72
45	-	⁝	5	640	Entity-Name Hyphen character		
46	.	·	1	128	1 unit horizontal connection line		[7] See #44, 59, 71, 72
47	/	╱	6	768	Horizontal Exclusivity Arc Rounded Transition (Bottom)		See #36, 37, 92
48	0	0-9	5	640	Entity-Name Characters 0 through 9		[8]
58	:	⁚	1	128	1 unit Space character (as part of an Entity Name)		[9] See #43, 61
59	;	–	4	512	4 unit horizontal connection line		[7] See #44, 46

---

4   The space character is for use outside of an Entity box formed with letters set between the [and] characters.

5   The ERD characters generated by the dual and single quotation mark characters ['] and ["] are also generated by the "smart" quote characters ['], ['], ["] and ["], (u+2018, u+2019, u+201c and u+201d respectively) for convenience when using word processors that automatically substitute these.

6   See the table titled "Special Characters in Alternate Sizes: Entity Name Space Characters" on page 9.

7   See the table titled "Special Characters in Alternate Sizes: Horizontal Connector Line Characters" on page 9.

8   Numeric characters from 0 to 9 are based on the DejaVu Sans Mono font, and are for use within Entity boxes.

dec	α	ERD	x	Unit Width	Description	ERD_A-TYPEFACE-CHARACTERS ▷——	Comments
60	<	≮	6	768	Horizontal Connector Symbol for "to Many" cardinality		See #62, 94
61	=	‾_	5	640	5 unit Space character (as part of an Entity Name)		See #43, 58
62	>	≯	6	768	Horizontal Connector Symbol for "Many to" cardinality		See #61, 94
63	?	│	6	768	Used for connecting upper and lower multi-line Inclusion Arcs		See #41, 125
64	@	●	6	768	Connector Symbol for "1" (used on Bachman notation only)		See * (42)
65	A	—○	15	1920	(0,1)		See L (76)
66	B	—●	6	768	Connector Symbol (used on Bachman notation only)		See N (78)
67	C	▶	6	768	L:(1,n) (used on Bachman notation only)		See X (88)
68	D	—‖	5	640	L:(1!); more explicit than Q (81)		See J (74)
69	E	—<	15	1920	L:(1,n)		See I (73)
70	F	‾F	15	1920	Entity-Name Uppercase Character F		[9]
71	G	——	15	1920	LR:(1)		See 44, 46, 59, 72
72	H	·····	15	1920	LR:(0,1)		See 44, 46, 59, 71
73	I	>—	15	1920	R:(1,n)		See E (69)
74	J	‖—	15	1920	R:(1!); more explicit than P (80)		See D (68)
75	K	>○—	15	1920	R:(0,1,n)		See S (83)
76	L	○—	5	640	R:(0,1)		See A (65)
77	M	>··	15	1920	R:(0,n)		See C (67)
78	N	●—	15	1920	R:(1) (used on Bachman notation only)		See B (66)
79	O	⊦○—	15	1920	R:(0,1)		See W (87)
80	P	⊢—	15	1920	R:(1)		See Q (81)
81	Q	—⊣	15	1920	L:(1)		See P (80)
82	R	—‖<	15	1920	L:(1,n)		See U (85)
83	S	○<	15	1920	L:(0,n)		See K (75)

---

9 Upper Case Alphabetic characters from A to Z that are not used for "pre-built" optionality/cardinality glyphs are based on the DejaVu Sans Mono font, and are for use within Entity boxes.

dec	α	ERD	x	Unit Width	Description	ERD_A-TYPEFACE-CHARACTERS ▷⊢—— Comments
84	T	⊸◁	15	1920	L:(0,1,n) generally a more explicit version of S	See Y (89)
85	U	⊁⊢	15	1920	R:(1,n)	See R (82)
86	V	··⊰	15	1920	L:(1)	See N (78)
87	W	�leftarrow○⊦	15	1920	L:(0,1)	See O (79)
88	X	◀	5	640	R:(1,n) (used on Bachman notation only)	See C (67)
89	Y	⊁○-	15	1920	R:(0,1,n) generally a more explicit version of K	See T (84)
90	Z		15	1920	15 unit Space character (not part of an Entity Name)	See 32, 88, 90, 96, 126
91	[	⌈	5	640	Entity-Box Left Border	See #34
92	\	⤸	6	768	Horizontal Exclusivity Rounded Transition (for Top Row)	See #36, 37, 47
93	]	⌉	5	640	Entity-Box Right Border	See #35
94	^	⏐	6	768	Character Space with upper connector for pig's ear	[10]
95	_	=	5	640	Entity-Name Character for Underscore	[11]
96	`		2	256	2 unit Space character (not part of an Entity Name)	[8] See #32, 88, 90, 126
97	a	A̅	5	640	Entity-Name Lowercase Character A (printed/displayed as Caps)	[9, 12]
98-122	b	B̅-̅Z̅	5	640	Entity-Name Lowercase Characters B through Z (as Caps)	
123	{	⊢	6	768	Horizontal Exclusivity Connector (used with "(" and "!")	See #40 for Bottom
124			⊦	1	128	Connector Symbol for a cardinality of "1"	
125	}	⊣	6	768	Horizontal Inclusion Connector (used with ")" and "?")	See #41 for Bottom
126	~		5	640	5 unit Space character (not part of an Entity Name)	See #32, 88, 90, 96
u2018 u2019	' '	⏐	1	128	"Smart" single quotation marks are mapped to normal ones [']	[7] See #39
u201c u201d	" "	⌈	1	128	"Smart" double quotation marks are mapped to normal ones ["]	[7] See #34

---

10 This symbol is generally only useful when creating an entity with a word processor table as shown with Entity_9 above.
11 The underline character is for use within an Entity box and therefore the underscore is higher than usual.
12 For convenience while typing, the lower case characters from "a" to "z" all produce upper case characters.

Special Characters in Alternate Widths

Relative Unit Width:	1	2	4	5	15
Normal Space Characters:	{space-bar} (32)	[`] key; (96)		[~] key; (126)	[Z] key; (90)

	1	2	4	5	15
Entity Name Space Characters:	[:] key; (58)	[+] key; (43)		[=] key; (61)	
As displayed using the ERD_A font:	. .	- -		—— ——	

*(width of both G and H = 5)

	1	2	4	5	15
Horizontal Connector Characters:	[.] key; (46)	[,] key; (44)	[;] key; (59)	[G] key; (71)	[H] key*; (72)
As displayed using the ERD_A font:	.	-	–	——	· · · · ·

Vertical Connector Characters	['], ['], ['] keys; (39)		[!] key; (33)		[^] key; (94)	
As displayed using the ERD_A font:	│		│		│	

(left margin vertical text) VERTICAL_ENTITY

(left margin vertical text) This column's text is set to 90°

Pre-assembled Connectors

(left margin vertical text) Remember – it is up to the user to insure that glyphs from different modeling conventions are not intermingled.

| To create these combinations ...	… type these keys	… or type the following characters to customize	
`H-OUT` ·····——`G-IN`	`[h-out]H~G[g-in]`	`H = .`,`,`,`,`.`	`G = .,;;;`
`G-OUT` ——·····`H-IN`	`[g-out]G~H[h-in]`	`G = ;;;,.`	`H = .`,`,`,`,`.`
`P-OUT` ┠——┨`Q-IN`	`[p-out]P~Q[q-in]`	`P = ,|;;`	`Q = ;;|,`
`J-OUT` ╫——╫`D-IN`	`[j-out]J~D[d-in]`	`J = ,|.|.,;`	`D = ;,.|.|,`
`L-OUT` ○——○`A-IN`	`[l-out]L~A[a-in]`	`L = ,*;`	`A = ;*,`
`N-OUT` ●——●`B-IN`	`[n-out]N~B[v-in]`	`N = ,@;`	`V = ;@,`
`O-OUT` ⊶——⊷`W-IN`	`[o-out]O~W[w-in]`	`O = ,|.*,`	`W = ,.*|;`
`M-OUT` ▷·· ··◁`V-IN`	`[m-out]M~V[c-in]`	`M = >`,`,`	`C = . ,  ,<`
`I-OUT` ▷——◁`E-IN`	`[i-out]I~E[e-in]`	`I = >;;`	`E = ;;<`
`U-OUT` ▷┥——┝◁`R-IN`	`[u-out]U~R[r-in]`	`U = >|;`	`R = ;|<`
`K-OUT` ▷○——○◁`S-IN`	`[n-out]N~V[v-in]`	`N = >*;`	`S = ;*<`
`Y-OUT` ▷○——○◁`T-IN`	`[y-out]Y~T[t-in]`	`Y = >|*;`	`T = ;*|<`

Other Useful Characters

Some notations, such as the Barker/Oracle notation, prefix column/attribute names with symbols such as [#] (Unique ID), [*] (Mandatory), and [○](Optional) to provide further information. Since these and similar entries would not be made with the ERD_A font, the meaning and use of such symbols are not within the scope of this document, nor is the method for entering them when they don't fall in the set of characters available on a standard keyboard – the ○ character, for example..

The following short list, however, provides the Unicode mappings for some of these that are useful in creating text-only Entity-Relationship Diagrams:

|		
○	u+25cb	The notation for an optional attribute as mentioned above.
●	u+25cf	Compare to the [@] character in the ERD_A font.
▶	u+25ba	Compare to the [C] character in the ERD_A font.
◀	u+25c4	Compare to the [X] character in the ERD_A font.
◊	u+25ca	Seldom used character indicating non-transferability.

Font Size Samples

Because the ERD_A font is as non-sophisticated as it is possible to be, it contains no optimizations for display at different sizes, nor for kerning or other niceties. The time it would require to do so seemed unjustified for its limited purpose and the fact that it is not intended for text that is to be "read." Nonetheless, it seems satisfactory for its intended use – as illustrated by the samples below – and is therefore made available in hopes that others may find it useful.
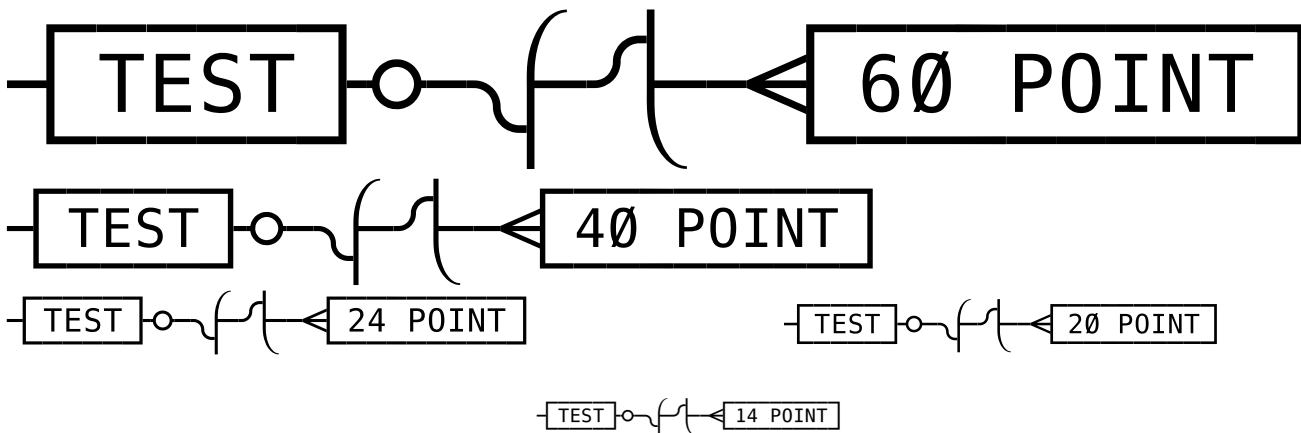
Installing the font in Linux/Unix Systems

The ERD_A font has been successfully tested on various Linux and Windows operating systems.

For Linux/Unix systems that have no automated font installation routines, copy the ERD_A.ttf font file into the hidden subdirectory $HOME/.fonts; if it is necessary to update the font cache on some systems to make the font appear, issue the command fc-cache -fv from a terminal (Ctrl+Alt+t) as a super user. Any font cache is automatically recreated on a new session.

Antikythera Publications
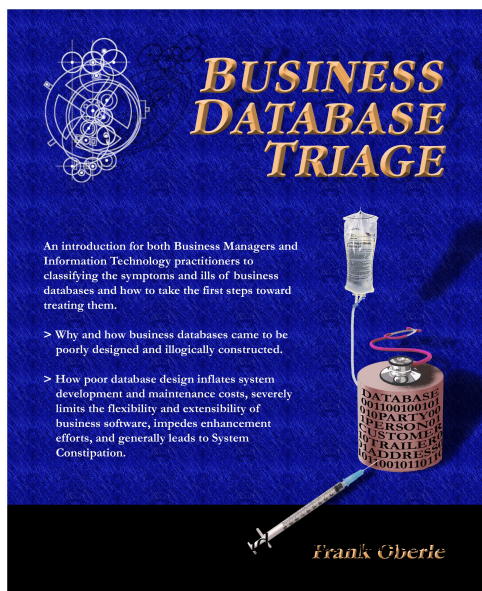
In addition to an ongoing series of Database Design Notes, Antikythera Publications recently released the book "*Business Database Triage*" (ISBN-10: 0615916937) that demonstrates how commonly encountered business database designs often cause significant, although largely un-recognized, difficulties with the development and maintenance of application software. Examples in the book illustrate how some typical database designs impede the ability of software developers to respond to new business opportunities – a key requirement of most businesses.

A number of examples of solutions to curing business system constipation are presented. Urban legends, such as the so-called object-relational impedance mismatch, are debunked – shown to be based mostly on illogical database (and sometimes object) designs.

"*Business Database Triage*" is available through major book retailers in most countries, or from the following on-line vendors, each of which has a full description of the book on their site:

CreateSpace: https://www.createspace.com/4513537

Amazon:
www.amazon.com/Business-Database-Triage-Frank-Oberle/dp/0615916937
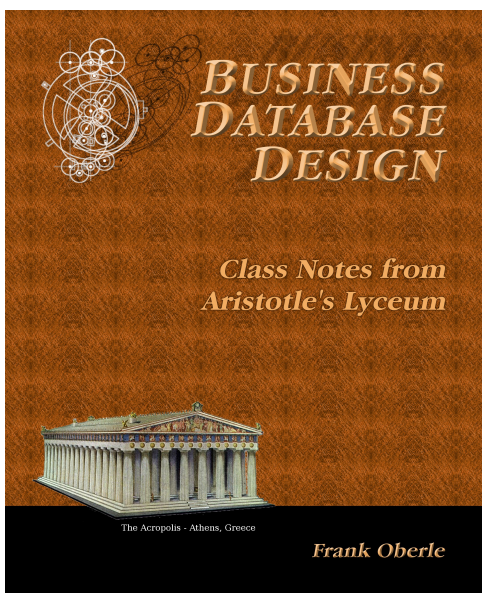
A follow-up book, "*Business Database Design – Class Notes from Aristotle's Lyceum*" is due to be available in the early part of 2015.

"*Business Database Design*" leads the reader through the logical design and analysis techniques of data organization in more detail than the earlier work – which concentrated more on understanding and identifying problems caused by illogical database design rather than their solutions.

These logical approaches to data organization, espoused by Aristotle and an "A-List" of his successors, have formed the basis for scientific discovery over more than 2,400 years, and directly led to the technology we deal with today, notably including both relational and object theory.

"*Business Database Triage*" explained the reasons why these principles were virtually impossible to apply during the early years of our transition to the use of computers in business, but since the technology is now sufficiently mature that such compromises can no longer be justified, the time has come to relearn logical data organization techniques and apply them to our businesses.

To download the ERD_A TrueType Font used in this document, along with a tutorial and keyboard map, visit www.AntikytheraPubs.com, where a variety of other Database Design Notes are also available for download.

Keyboard Layout for ERD_A Font

The [:], [+] and [=] keys produce 1, 2 and 5 unit spaces used in Entity Names.
The space bar produces a normal (1 unit) space.
The [`], [~] and [Z] produce 2, 5 and 15 unit spaces respectively.

To create these combinations ...	… type these keys
H-OUT ···· ── G-IN	[h-out]H~G[g-in]
G-OUT ── ···· H-IN	[g-out]G~H[h-in]
P-OUT ── ── Q-IN	[p-out]P~Q[q-in]
J-OUT ── ── D-IN	[j-out]J~D[d-in]
L-OUT ── ── A-IN	[l-out]L~A[a-in]
N-OUT ── ── B-IN	[n-out]N~B[b-in]

NB: The "~" character in these examples generates 5 spaces

To create these combinations ...	… type these keys
O-OUT ── ── W-IN	[o-out]O~W[w-in]
M-OUT ── ── V-IN	[m-out]M~V[c-in]
I-OUT ── ── E-IN	[i-out]I~E[e-in]
U-OUT ── ── R-IN	[u-out]U~R[r-in]
K-OUT ── ── S-IN	[k-out]K~S[s-in]
Y-OUT ── ── T-IN	[y-out]Y~T[t-in]