

JOE CELKO'S
DATA,
MEASUREMENTS
AND STANDARDS
SQL

Joe Celko's Data, Measurements and Standards in SQL

Joe Celko



AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Morgan Kaufmann Publishers is an imprint of Elsevier



Morgan Kaufmann Publishers is an imprint of Elsevier.
30 Corporate Drive, Suite 400, Burlington, MA 01803, USA
This book is printed on acid-free paper.
© 2010 by Elsevier Inc. All rights reserved.

Designations used by companies to distinguish their products are often claimed as trademarks or registered trademarks. In all instances in which Morgan Kaufmann Publishers is aware of a claim, the product names appear in initial capital or all capital letters. All trademarks that appear or are otherwise referred to in this work belong to their respective owners. Neither Morgan Kaufmann Publishers nor the authors and other contributors of this work have any relationship or affiliation with such trademark owners nor do such trademark owners confirm, endorse or approve the contents of this work. Readers, however, should contact the appropriate companies for more information regarding trademarks and any related registrations.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopying, scanning, or otherwise—without prior written permission of the publisher.

Permissions may be sought directly from Elsevier's Science & Technology Rights Department in Oxford, UK: phone: (+44) 1865 843830, fax: (+44) 1865 853333, E-mail: permissions@elsevier.com. You may also complete your request online via the Elsevier homepage (<http://elsevier.com>), by selecting "Support & Contact" then "Copyright and Permission" and then "Obtaining Permissions."

Library of Congress Cataloging-in-Publication Data

Celko, Joe.

Joe Celko's data, measurements, and standards in SQL/Joe Celko.

p. cm.—(The Morgan Kaufmann series in data management systems)

ISBN 978-0-12-374722-8

1. SQL (Computer program language) 2. Database management. I. Title. II. Title: Data, measurements, and standards in SQL.

QA76.73.S67C4324 2010

005.74—dc22

2009030587

British Library Cataloguing-in-Publication Data

A catalogue record for this book is available from the British Library.

For information on all Morgan Kaufmann publications,
visit our Web site at www.mkp.com or www.elsevierdirect.com

Printed in the United States of America

09 10 11 12 13 14 15 16 5 4 3 2 1

Working together to grow
libraries in developing countries

www.elsevier.com | www.bookaid.org | www.sabre.org

ELSEVIER

BOOK AID
International

Sabre Foundation

Introduction

As I sit here typing this introduction, I am looking at a copy of *The North American Arithmetic, Part Third, for Advanced Scholars*, by Frederick Emerson. It was written in 1834.

It has chapters on the money, weights, and measures in use in Europe and America at that time. The English still had pence, shillings, and farthings. The countries and city-states on the continent had a local currency instead of a Euro. Troy, Avoirdupois, and Apothecary weights were mentioned as standards in Europe.

But after that, there were English ells, French ells, and Flemish ells for measuring cloth—all of them different. There were separate liquid measures for wine and beer. Cities had local measures, so “100 lbs” in Trieste was 123.6 pounds Avoirdupois and 74.77 pounds Avoirdupois in Rome. The good news was that the new metric system in France seemed to be catching on and becoming more popular. How, do you suppose?

All of these differences were handled by Gaugers in each city. These were usually government tax employees who kept a set of dip sticks (or gauges) to measure liquids in barrels, weights for balance scales, baskets for dry goods, and so forth. The term survives today in the oil refinery business for people who test oil in storage tanks. They also calibrated the local merchant’s tools to assure fair trade within their jurisdiction. The seal cast on the front of a wine carafe is a leftover of a Gauger’s official seal, which would have been stamped into a piece of lead.

Remember that all of this was a long time before computers and electronic data exchange. Think about the manual records needed. You can imagine what effect this had on commerce, since it made accurate data almost impossible to keep and exchange.

The bad news is that this “local measurement” mentality has not died in the 21st century. Ignorance of international and industry standards and a “cow-boy coder” mind-set led programmers to code first and think never. Instead of looking for an ISO or industry-specific standard for something, the “cow-boy coder” gallops in and writes a list or uses a proprietary auto-numbering feature in his current SQL product to give him an encoding for an attribute.

Programmers still spend time and effort avoiding the use of common units of measure and industry standards. It can actually become quite silly.

In a posting on the Microsoft SQL Server Programming newsgroup, an MVP (“Most Valuable Programmer”—an award given by Microsoft in their

product categories) posted a complaint that passing temporal data that will work across regional settings doesn't work in the ANSI Standard! His example was two values on a machine configured to Microsoft's U.K. formats instead of to ISO standards, thus:

```
CAST('2007-04-01' AS DATETIME) - ISO/ANSI SQL format
versus
CAST('2007-04-01T00:00:00' AS DATETIME) - set to UK format
```

To quote: "Here in the UK in cloudy Harpenden the default connection settings give these results ...":

```
'2007-01-04 00:00:00.000'
versus
'2007-04-01 00:00:00.000'
```

The use of a T is an alternative ANSI standard, which makes a timestamp into a single unbroken string (this can be useful in some systems). But the "T" is not part of SQL's subset of the ISO-8601 standard (this might be changed by the time you read this book). But the ordering of year-month-day is in all ISO-8601 standards—and it makes sense for sorting!

I have run into this problem with a system that collected data from the United States and the United Kingdom. You would be surprised how long you can screw up the database before you catch the error.

The poster ranted (imagine strong British accent): "At what point will this guy [Celko] stop telling us duff information and bashing people for not following best practice and standards when he himself knows there's a problem yet refuses to yield to the correct best practice?"

It was very hard to tell him that Harpenden, UK, is not the center of the world, whether it is cloudy or not! Even Greenwich Time is no longer the standard. The best practices are to store time in UTC and use the TIMEZONE options in SQL to get local lawful time in the applications. It is all there in the standards.

COMMON UNITS EVOLVE TO STANDARDS

There is a common pattern in moving to standards, which is easy to see in physical goods. Since beer is one of my favorite physical goods, consider this history.

Units of measurement become standardized and replace local measurements.

But we do not have standard sizes yet. For example, after a few hundred

years, we get rid of pints, firkins, and a host of local units with strange names to use liters for our beer. But every pub still has a different size glass.

Enterprises working in the same area agree on standard sizes for common use. For example, the pubs all get together in one town and agree that a glass of beer is 0.50 liters; in other town, they decide that a “fair pour” as it is called in the bar trade, should be 0.75 liters (nice place to visit).

National standards emerge, usually by law based on the need for taxation. Continuing my example, the government decided to pass a beer tax based on the glass, so we need a National Fair Pour Standard.

International standards emerge, usually by agreement based on the need for trade and data exchange.

Notice that the glassmaker has a strong influence with his offerings to the bar trade. One trade affects another. Let me give a specific example. Before the 1800s, machine screws were made by individual machine shops in the United States. The common units of measure were inches for the physical dimensions such as length, pitch, and diameter and degrees for the angle of the thread. But that did not mean that the screws were interchangeable or even used the same names for various sizes. It was very much a cottage industry.

Some national standards did eventually arise within industries inside the United States, but it was not until World War II that American, Canadian, and British manufacturers decided to standardize their inch-based sizes for the war effort. The Unified Thread Standard was adopted by the Screw Thread Standardization Committees of Canada, the United Kingdom, and the United States on November 18, 1949, in Washington, DC, with the hope that they would be adopted universally.

It did not work out that way. Europe and much of the rest of the world turned to the ISO metric screw thread system. Today, globalization has forced the United Kingdom and the United States to use this same system. The best example is that the ISO metric screw thread is now the standard for the automotive industry and that very few parts use any inch-based sizes, regardless of where they are made. The United States used to completely dominate the auto industry but lost out for various reasons. A major factor is that ISO machinery is engineered to the millimeter while the U.S. auto industry was working at 1/32 of an inch. An “ISO automobile” is simply more precise and accurate than an “Imperial automobile” because of the measurements used. This was not a problem with the Model T, but it is a disaster with a modern vehicle.

WHAT DOES THIS HAVE TO DO WITH DATABASES?

Today, the database designer is finding himself in the position of the Gauger from the Middle Ages and Renaissance, only his role is greatly expanded.

We have to share and move data, rather than physical goods, from one enterprise to another accurately. The enterprise is not in the next village but anywhere on Earth. We all have to “speak a common language” with the rest of the world.

We still deal with a lot of physical measurements, but we don’t use dip sticks to convert “London firkins” to “Harpenden fuddles” anymore. We need a bit more math and have rules about accuracy and computations.

The good news is that we have the Internet, ISO, the EU, ANSI, DIN, JIS, and scores of other organizations that do nothing but help us with standards. These days, many of those standards come with a description of how they are represented in a database. They come with Web sites so that we can download and update our systems for little or nothing. They are easy to find—God bless Google!

And if we don’t like them, these same organizations have mechanisms for submitting comments and proposals. In the old days, you had to appeal to the king to become the royal barrel maker, bottle blower, or whatever.

The purpose of this book is to turn a new database designer in a 21st-century Gauger. He is going to need some new skills, in particular:

- Knowledge of standards
- How to migrate as standards evolve
- How to create and use a data dictionary
- How to maintain data integrity
- How to maintain data quality

In *MAD* magazine #33, Donald E. Knuth wrote a parody of the metric system while in high school, long before he became one of the greatest computer scientists of all time. The title was “Potrzebie System of Weights and Measures,” and it was illustrated by Wally Wood (see next page for the image).

Potrzebie is a Polish noun (“potrzeba” translates as “a need”) that *MAD* magazine used as a nonsense word in their articles and cartoons.

While Dr. Knuth was being funny, the number of database designers who invent elaborate measurement and encoding schemes when standards exist is not so funny.

THE POTRZEBIE SYSTEM

OF WEIGHTS AND MEASURES

THE POTRZEBIE SYSTEM

This new system of measuring, which is destined to become the measuring system of the future, has decided improvements over the other systems now in use. It is based upon measurements taken 6-9-12 at the Physics Lab. of Milwaukee Lutheran High School, in Milwaukee, Wis., when the thickness of MAD Magazine #26 was determined to be 2.26334851-

7438173216473 mm. This length is the basis for the entire system, and is called one potrzebie of length.

The Potrzebie has also been standardized at 3515.-3502 wave lengths of the red line in the spectrum of cadmium. A partial table of the Potrzebie System, the measuring system of the future, is given below.

LENGTH

1 potrzebie = thickness of MAD #26
 .000001 p = 1 farshimmelt potrzebie (fp)
 1000 fp = 1 millipotrzebie (mp)
 10 mp = 1 centipotrzebie (cp)
 10 cp = 1 decipotrzebie (dp)

10 dp = 1 potrzebie (p)
 10 p = 1 dekapotrzebie (Dp)
 10 Dp = 1 hectopotrzebie (Hp)
 10 Hp = 1 kilopotrzebie (Kp)
 1000 Kp = 1 furshlugginer potrzebie (Fp)

VOLUME

1 cubic dekapotrzebie = 1 ngogn (n)
 .000001 n = 1 farshimmelt ngogn (fn)
 1000 fn = 1 millingogn (mn)
 10 mn = 1 centingogn (cn)
 10 cn = 1 decingogn (dn)

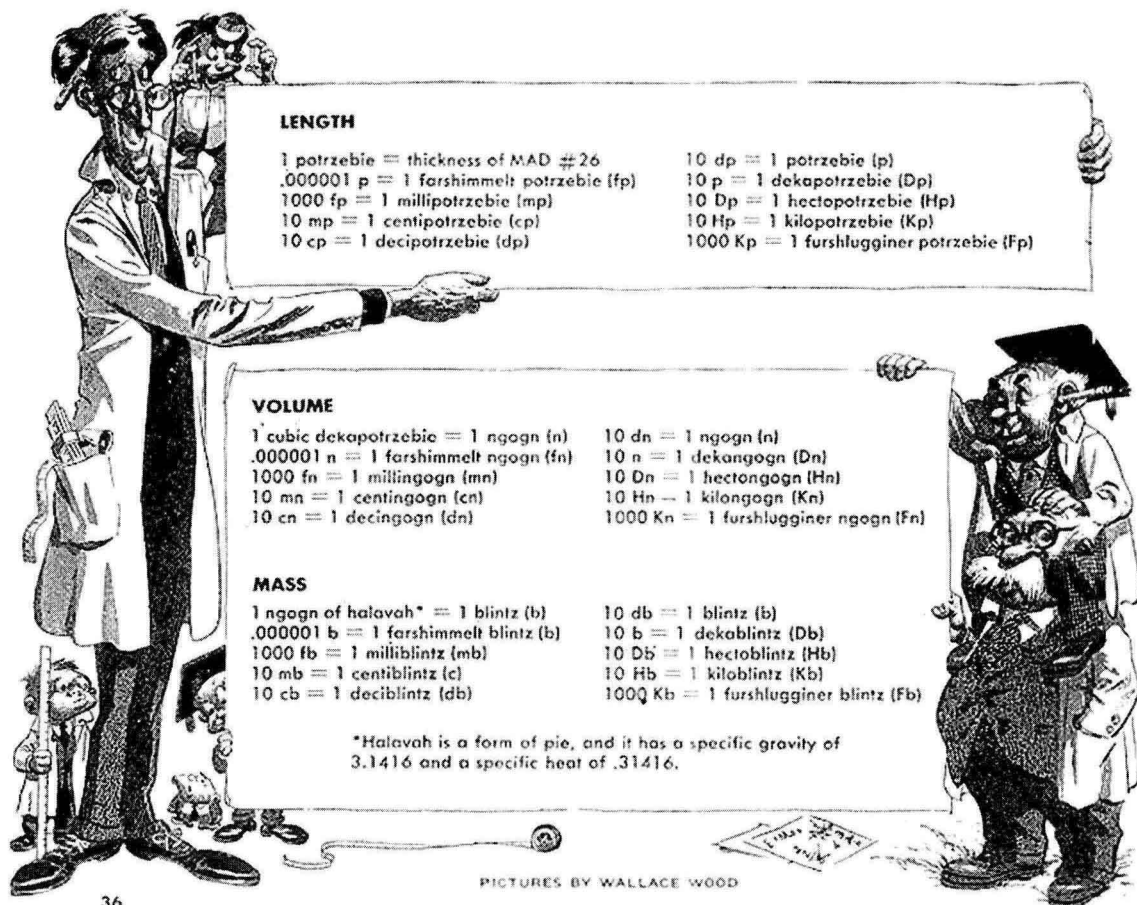
10 dn = 1 ngogn (n)
 10 n = 1 dekanoggn (Dn)
 10 Dn = 1 hectogogn (Hn)
 10 Hn = 1 kilogogn (Kn)
 1000 Kn = 1 furshlugginer ngogn (Fn)

MASS

1 ngogn of halavah* = 1 blintz (b)
 .000001 b = 1 farshimmelt blintz (fb)
 1000 fb = 1 milliblintz (mb)
 10 mb = 1 centiblintz (cb)
 10 cb = 1 deciblintz (db)

10 db = 1 blintz (b)
 10 b = 1 dekablintz (Db)
 10 Db = 1 hectoblintz (Hb)
 10 Hb = 1 kiloblintz (Kb)
 1000 Kb = 1 furshlugginer blintz (Fb)

*Halavah is a form of pie, and it has a specific gravity of 3.1416 and a specific heat of .31416.



PICTURES BY WALLACE WOOD

Let's get started. The first part of this book is the preliminaries and basics of measurements, data, and standards. The second part is the particulars for standards that should be useful for a working SQL programmer.

Please send any corrections, additions, suggestions, improvements, or alternative solutions to me or to the publisher.

Morgan Kaufmann Publishers
30 Corporate Dr #400
Burlington, MA 01803

Contents

Introduction.....	xiii
PART 1 HISTORY, STANDARDS, AND DESIGNING DATA	1
CHAPTER 1 Scales and Measurements.....	3
1.1. Measurement Theory.....	4
1.1.1. Range, Granularity and Your Instruments	6
1.1.2. Range	8
1.1.3. Granularity, Accuracy and Precision	8
1.2. Defining a Measurement.....	10
1.3. Tolerance.....	11
1.3.1. Scale Conversion Errors	12
1.4. Validation.....	12
1.5. Verification.....	13
1.5.1. Erroneous Values.....	13
1.5.2. Phony Values	14
1.5.3. Degree of Trust Versus Risk of Error.....	14
CHAPTER 2 Validation.....	17
2.1. Look-Up Tables.....	18
2.1.1. Auxiliary Tables for Noncomputed Data	20
2.2. Check Digits.....	21
2.2.1. Error Detection Versus Error Correction	22
2.2.2. Check Digit Algorithms	22
2.3. Declarations, not Functions, not Procedures	30
2.4. Patterns and Regular Expressions.....	34
2.4.1. Tricks with Patterns.....	35
2.4.2. Results with NULL Values and Empty Strings.....	37
2.4.3. Like is not Equality.....	37
2.4.4. Avoiding the Like Predicate with a Join	38
2.4.5. CASE Expressions and Like Predicates.....	40
2.4.6. Similar to Predicates	41
2.4.7. Tricks With Strings.....	42
2.4.8. Regular Expression Web Sites.....	45
2.5. Nondatabase Validation.....	45
CHAPTER 3 Data Encoding Schemes.....	47
3.1. Bad Encoding Schemes	48
3.2. Encoding Scheme Types	51
3.2.1. Enumeration Encoding	51

3.2.2. Measurement Encoding.....	52
3.2.3. Abbreviation Encoding	53
3.2.4. Algorithmic Encoding	53
3.2.5. Hierarchical Encoding Schemes	55
3.2.6. Vector Encoding.....	56
3.2.7. Concatenation Encoding.....	57
3.3. Atomic Versus Scalar	58
3.4. Transition States	59
3.4.1. State Transitions.....	59
3.4.2. State Transition DDL	60
3.4.3. State Transition Tables.....	61
3.4.4. Automatic State Transition Tables	63
3.5. General Guidelines for Designing Encoding Schemes	63
3.5.1. Use Existing Encoding Standards	63
3.5.2. Allow for Expansion	64
3.5.3. Use Explicit Missing Values to Avoid NULLs.....	64
3.5.4. Translate Codes for the End User	65
3.5.5. One True Look-up Table	66
3.6. Keep the Codes in the Database.....	69
3.7. Multiple Character Sets	70
CHAPTER 4 Scales	71
4.1. Bit Flags are not Scales.....	72
4.1.1. BITs.....	73
4.1.2. BITs and Booleans.....	74
4.1.3. BIT Vectors.....	75
4.1.4. Replacing BITs	76
4.2. Dimensionless Measurements.....	78
4.3. Types of Scales.....	78
4.3.1. Nominal Scales.....	79
4.3.2. Categorical Scales	79
4.3.3. Absolute Scales.....	81
4.3.4. Ordinal Scales.....	82
4.3.5. Rank Scales	83
4.3.6. Interval Scales	84
4.3.7. Ratio Scales.....	85
4.4. Using Scales	85
4.5. Scale Conversion.....	86
4.6. Derived Units.....	88
4.7. Punctuation and Standard Units	90
4.8. General Guidelines for Using Scales in a Database.....	92

CHAPTER 5 Data with Ignorance.....	95
5.1. Get it Right	95
5.2. Replace Bad Values with a General Dummy Value	96
5.3. Replace Bad Values with a Statistical Dummy Value	96
5.4. Replace Bad Values to Complete a Statistical Distribution ...	97
5.5. Replace Bad Values with Statistical Profiling.....	98
CHAPTER 6 Keys.....	99
6.1. Uniqueness	99
6.2. A Key Cannot Be NULL.....	99
6.3. Invariant Or Universally Controlled Values	100
6.4. Surrogate And Physical Locators Keys	101
6.4.1. Physical Locators for Performance	102
6.4.2. Physical Locators for Lack of a Proper Key	103
6.4.3. Trusted Sources.....	104
PART 2 A SAMPLING OF STANDARDS	105
CHAPTER 7 Dates	107
7.1. ISO-8601 standard	107
7.1.1. Year Field	108
7.1.2. Month Field.....	109
7.1.3. Week Field	109
7.1.4. Day Field	110
7.1.5. Time Field.....	111
7.1.6. Time Zones and Daylight Saving Time	112
7.2. Putting it all Together	113
7.3. Durations and Intervals	113
7.3.1. Time Intervals	114
CHAPTER 8 Sex Codes.....	115
8.1. Sex Codes.....	115
8.2. Other Sources.....	117
CHAPTER 9 Ethnicity and Race Codes	119
9.1. Race Versus Ethnicity.....	119
9.2. U.K. Ethnic Groups.....	120
CHAPTER 10 ISO-3166 and Other Country Codes.....	123
10.1. ISO 3166-1	123
10.2. ISO 3166-2	123
10.3. ISO 3166-3	124

CHAPTER 11	Language Codes.....	125
CHAPTER 12	Currency Codes	127
	12.1. Noncurrency Units	128
CHAPTER 13	National Identification Numbers.....	129
	13.1. Social Security Numbers	130
	13.1.1. SSN Area Numbers.....	130
	13.1.2. SSN Group Numbers	134
	13.1.3. SSN Serial Numbers.....	135
	13.1.4. SSN Validation	135
	13.2. Social Insurance Number.....	136
	13.2.1. SIN First Digit	136
	13.2.2. SIN Validation	137
	13.3. Swedish Personal Identity Number	137
	13.3.1. Personnummer Format	137
	13.4. Eu Biometric Passports.....	138
	13.4.1. Fingerprint Classification Systems.....	139
CHAPTER 14	Occupations	141
	14.1. National Occupational Classification (NOC)	142
CHAPTER 15	Colors	145
	15.1. International Color Consortium.....	146
CHAPTER 16	Telephone Numbers	147
	16.1. The International Telephone Number Components ...	147
	16.2. Subscriber Local Exchange or Prefix	149
CHAPTER 17	E-Mail Addresses	151
CHAPTER 18	Universal Postal Union.....	153
	18.1. ZIP Code	154
	18.1.1. Basic ZIP Code	154
	18.1.2. ZIP + 4	154
	18.1.3. Postal Zone Charts.....	155
	18.1.4. ZIP Code Validation	156
	18.2. Canadian Postal Codes.....	156
	18.3. Postcodes in the United Kingdom.....	158
	18.3.1. Postcode Formats	158
	18.3.2. Greater London Postcodes.....	159
	18.4. Case Expression for Many International Postal Codes ..	160
CHAPTER 19	Hierarchical Triangular Mesh	163

CHAPTER 20	Shoe Sizes	167
CHAPTER 21	International Clothing Sizes	169
CHAPTER 22	ICD Codes	173
	22.1. Local Versions	173
	22.2. Mental and Behavioral Disorders	174
CHAPTER 23	Vehicle Identification Number (VIN)	175
	23.1. Vin Format	175
CHAPTER 24	Freight Containers	179
	24.1. Freight Container Codes	179
	24.1.1. Map Letters to Numeric Values	180
	24.1.2. Assign Weights to Each Position	180
	24.1.3. Compute the Check Digit from the Weighted Sum	180
	24.2. Size and Type Codes	180
	24.3. Related ISO Standards	180
CHAPTER 25	Credit Card Numbers	183
	25.1. Card Issuers	183
	25.2. Account Number	184
	25.3. Other Numbers	184
	25.4. Personal Identification Numbers	185
	25.5. PCI DSS and Related Standards	185
	25.6. Tools and More Information	186
CHAPTER 26	SWIFT And Related Banking Standards	187
	26.1. BIC Codes	188
	26.2. International Bank Account Number (IBAN)	189
	26.2.1. IBAN Check Digits	190
CHAPTER 27	Data Universal Numbering System	193
CHAPTER 28	Global Trade Item Number	195
	28.1. GTIN Family	196
	28.1.1. Prefixes for UPC	196
	28.2. ISBN	197
CHAPTER 29	Digital Object Identifier (DOI)	199
	29.1. DOI Syntax	199
	29.2. ISBN-A	201
CHAPTER 30	Audiovisual Media	203
	30.1. Format	204
	30.1.1. Root	204

	30.1.2. Episode or Part	204
	30.1.3. Version	204
	30.2. Cataloging AV Materials	205
CHAPTER 31	Isin And Related Securities Identifiers	207
	31.1. Cusip	208
	31.2. Sedol and Other Countries	209
	31.3. Classification of Financial Instruments	209
CHAPTER 32	Temperature Scales.....	211
	32.1. Celsius Scale	212
	32.2. Fahrenheit Scale	213
	32.3. Kelvin Scale.....	213
	32.4. Other Temperature Scales	214
	32.4.1. Rankine Scale.....	214
	32.4.2. Delisle Scale	214
	32.4.3. Newton Scale	215
	32.4.4. Réaumur Scale	215
	32.4.5. Rømer Scale	215
CHAPTER 33	National Animal Identification System (NAIS)	217
	33.1. Premises Identification Number (PIN)	217
	33.2. Animal Identification Number (AIN and GIN)	218
	33.3. Animal Tracking and other Considerations	221
CHAPTER 34	Iso 216 Paper Sizes ("A," "B," And "C" Series).....	223
	34.1. "B" Series	224
	34.2. "C" Series	225
	34.3. Tolerances.....	225
	34.4. Non-ISO Paper Sizes	225
CHAPTER 35	Compass Points.....	227
	35.1. Traditional Compass Points	227
	35.2. Other Compass Point Systems.....	228
CHAPTER 36	Unicode.....	231
	36.1. Types of Written Languages	232
	36.2. Practical and Political Problems.....	232
	36.3. Normalization.....	233
CHAPTER 37	Driver's Licenses.....	235
	37.1. ID-1 Cards	236
	37.2. U.S. Driver's Licenses	236
	37.3. Enhanced Driver's License (EDL)	237

CHAPTER 38	Currency Units and Near Money.....	239
	38.1. Stock Exchanges and Nondecimal Units	240
	38.2. Decimalization in the United Kingdom.....	240
	38.3. Physical Currency Choices.....	241
	38.4. Coupons	243
	38.4.1. Types of Coupons	244
	38.4.2. Trade Groups and References.....	245
CHAPTER 39	Recipes and Food Preparation	247
	39.1. Weight Versus Volume	247
	39.2. Scaling a Recipe	248
CHAPTER 40	Portable Document Format (PDF).....	249
	40.1. ISO 32000.....	249
	40.2. Things that Paper Cannot do.....	251
CHAPTER 41	Temporal Data	253
	41.1. Notes On Calendar Standards	253
	41.1.1. The Julian and Gregorian Calendars.....	254
	41.1.2. Computerizing Calendars	256
	41.1.3. Leap Seconds.....	257
	41.1.4. Calendar Tables	258
	41.2. SQL Temporal Data Types.....	261
	41.2.1. Tips for Handling Dates, Timestamps and Times	262
	41.2.2. Date Format Standards.....	262
	41.2.3. Handling Timestamps.....	263
	41.2.4. Handling Times	265
	41.3. Expressions with Date Arithmetic	267
	41.4. The Nature of Temporal Data Models	268
	41.5. References on Temporal Data and SQL.....	269
CHAPTER 42	Additive Congruential Generators	271
CHAPTER 43	Traditional and Metric Typographic Units.....	275
	43.1. DIN 16507-2 Font Sizes	277
	Index.....	281

History, Standards, and Designing Data

The first of this book discusses the principles of designing data encoding schemes and some of the history of standardization. This second part will give examples of actual standards used in a variety of industries.

In the first six chapters, I look at the foundations from the view point of a database designer who needs some understanding of the how and why.

I find it odd that database designers are very physical about their data and do not work with many abstractions. They were never taught the theory of scales and measurements. They have only a minimal knowledge of validation, verification, and risk of error as a part of the data.

Check digit algorithms are taught as single “programming tricks” in undergraduate computer science classes rather than a mathematical discipline.

To the best of my knowledge, I am the only person who teaches Data Encoding Schemes in an orderly fashion.

The failure of cowboy coders to use standards leads to problems. The homegrown encoding schemes have to be