Ripose

Anatomy of database design

A better way of designing
Databases
&
Integrating them with
Business knowledge

LESS IS MORE

© 2018 Copyright Ripose Pty Limited

Overview

Purpose
The purpose of these courses is to introduce a better way to use time to design and develop database designs from the plethora of business and technology objects.

In the immortal words of Warren Buffett “There is no way that I will be able to buy more time".

Time management is of the utmost importance and thus far, according to my research, just about all of the approaches that I have studied fail to deliver the results of manipulating the plethora of business and technology objects.

Another way to look at this approach is -=+ (less is More)

© 2018 Copyright Ripose Pty Limited

Strengths and Weaknesses

Let me start with the simple step of identifying the 3 main stream ('best practice') approaches and show why they fail to achieve the optimum database design in a timely and cost effective manner. I will then introduce a better way to plan.

Method	Strength	Weakness
Normalisation	None	1) No provision for extended normal forms 2) Need all i/o media
Object orientation	Taxonomy: Encapsulation Inheritance Polymorphism	1) No extended polymorphic constructs 2) Need all i/o media
Semantic modelling	Ontology	1) Brain storming objects 2) Relational joins 3) Need all i/o media
Knowledge modelling	Taxonomy Ontology Heuristics	None

© 2018 Copyright Ripose Pty Limited

Why normalisation fails

Normalisation fails due to following 3 factors:
1) It requires (according to Codd’s Law) that every attribute depends upon a ‘key’, the whole ‘key’ and nothing but the ‘key’.
The problem is that the ‘key’ is a fictitious artifact that is created in order to ensure every row in a database table is unique. As an example: A table called Person (which encapsulates all the known facts/attributes of a real ‘person’ - alive or deceased) has to be given a unique identifier (starting at 1 and going on to infinity). How can an ‘attribute’ (for eg a person_first_name) of a ‘person’ (‘Joe’) depend upon the key (eg.1010)? It is simply nonsensical and hence as the key is fictitious, Codd’s law is fallacious and normalisation is the wrong method of designing a database.

2) In 1974 Codd and Bryce only managed to discover their BCNF or the Bryce Codd normal form which was regarded as the 3.5 normal form. I can prove that there are 4 more normal forms, that if ignored, will perpetuate and exacerbate the problem of designing data bases using Codd’s law.

3) In order to identify every fact/attribute that every table in a database requires, it is necessary to locate every input and output document before beginning the arduous and erroneous exercise of ‘normalising’ the data.

© 2018 Copyright Ripose Pty Limited

Why OO fails

Object orientation fails due to following 2 factors:
1) OO depends on its strength of encapsulation (combining or grouping), inheritance (hierarchical dependency) and polymorphism (having or occurring in several distinct forms).

Encapsulation and inheritance are strengths as they allow lower level objects to be grouped together (or classified) to form a hierarchy and allow the lower level objects to ‘receive properties from a predecessor’ (like genes from parents).

The weakness with the polymorphic function is that it only allows for 1 type of separation (mutually exclusive – for eg. an animal could be a dog or a cat but not both) which will ultimately cause the design of the database to fail.

2) In order to identify every fact that every table in a database requires, it is necessary to locate every input and output document before beginning the arduous and erroneous exercise of using OO to design a database.

© 2018 Copyright Ripose Pty Limited

Ripose

Why Semantic modelling fails

Semantic ('Of or relating to meaning or the study of meaning') modelling fails due to following 3 factors:

1) Semantic modelling depends on its strength of using ontology (the relations of meaning or the study of meanings).

The weakness with this approach is that prior to beginning a semantic modelling session every term used by the business needs to be known in advance and often that requires lengthy and repetitive brainstorming sessions. If a single term is left undiscovered or duplicated this will ultimately cause the design of the database to fail.

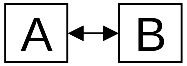
2) Semantic modelling often uses relational linking (or joining) of one ‘entity’ to another. The problem with this approach is that the number of possible joins is equal to the number of ‘objects’ squared. I will explain this in the next slide.

3) In order to identify every fact that every table in a database requires, it is necessary to locate every input and output document before beginning the arduous and erroneous exercise of using OO to design a database.

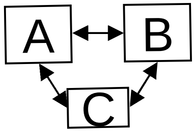
© 2018 Copyright Ripose Pty Limited

Relational joins

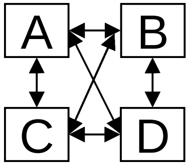
Relational joins (aka peer-to-peer) allows any object to be associated with any other object.
In the case of 2 objects this is easy as A relates to (-->) B & B-->A:



In the case of 3 objects this becomes a bit more difficult as A-->B, A-->C, B-->C, B-->A, C-->A & C-->B :



In the case of 4 objects even more so:



The reason why the number of permutation is equal to the number of ‘objects’ squared is because A can have a relationship with B and B can have a different relationship with A. In addition A can have a relationship with itself as can B. Hence 2 objects yields the possibility of 4 joins, 3 can have 9 joins, 4 can have 16, 5 yields 25, 6 yields 36 etc.

This makes semantic modelling unwieldy and prone to errors on a grand scale.

© 2018 Copyright Ripose Pty Limited

Knowledge modelling

Based on

- Taxonomy
  - Encapsulation
  - Extended polymorphism
  - Inheritance
  - Relationships
- Ontology
  - Synonyms – 'Two words that can be interchanged in a context'
  - Hyponyms - 'A word that is more specific than a given word'
- Heuristics/rule based
  - Navigational links
  - Business objective links

© 2018 Copyright Ripose Pty Limited

Knowledge modelling training course

This course is designed to teach how to identify the fundamental knowledge classes needed to support the resolution of problems.

Who should attend:

Any person interested in seeking knowledge and a way to categorise what they discover. They include IT and business managers, project managers, project leaders, business analysts, knowledge engineers, system analysts and programmers.

Content:

1. Knowledge conventions
2. Associations
3. The Generic Knowledge Model
4. Linking the knowledge base to the business objectives
5. Workshop
6. Expanding the knowledge base
7. Workshop

What you will learn:

1. What an Entity is
2. Why relationships are necessary
3. How to build an effective knowledge base
4. How to build flexible, stable, efficient and effective database structures
5. How to create quality documentation from the artifacts in the repository

© 2018 Copyright Ripose Pty Limited