

Rapid Information Processing Oriented Systems Environment

Rapid Information Processing Oriented Systems Environment (abbreviated to R.I.P.O.S.E) is a business development life cycle technique and knowledge management tool, designed to address the problem of cost and time over runs associated with developing and delivering business system projects.

R.I.P.O.S.E is based on a number of existing methodologies. These include [business systems planning](#) (BSP), [information engineering](#) (IE), [business process re-engineering](#) (BPR) and [software engineering](#). R.I.P.O.S.E has integrated their strengths and eliminated their weaknesses and overheads!

R.I.P.O.S.E has been developed to provide an automated solution to the three [Kantian](#) ideas:

- What one knows
- What one ought to do and
- What one can hope for

This document demonstrates how this can be achieved to assist in the development of [business information systems](#)

Aligning the R.I.P.O.S.E Technique to Kant's 3 volumes

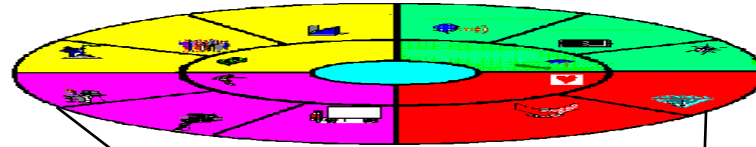
Kant's Vol I II & III			R.I.P.O.S.E technique			R.I.P.O.S.E architect	
Introduction							
Practice	Body	I Transcendental doctrine of elements Part I Transcendental aesthetic Section 1 - Space Section 2 - Time	What one can hope for	Physical	Program generators Web browsers Client server Terminal server	Body	Not applicable
	Mind	Part II Transcendental Logic Division 1 - Transcendental analytic Book I Book II Division 2 - Transcendental dialectic Book I Book II	What one ought to do	Logical	Data architecture Facts <hr style="border-top: 1px dashed black;"/> Data bases Process architecture Processes Applications	Mind	Grade 4 Grade 5 Grade 6 Grade 0
Theory	Soul	II Transcendental doctrine of method Chapter I - The discipline of pure reason Chapter II - The canon of pure reason Chapter III - The architectonic of pure reason Chapter IV - The history of pure reason	What one knows	Conceptual	Information architecture Objectives Goals Purpose Benefits/Missions Values/CSFs Issues/KPI	Spirit	Grade 1
				Knowledge	Soul	Grade 2	
				System architecture Strategies Tactics	Soul	Grade 3	

The R.I.P.O.S.E schematic – Part I

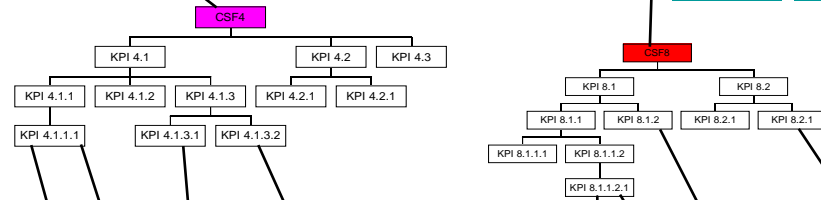
**IDEA
PROCESSOR**
(strategic planning)

Objectives

Goals

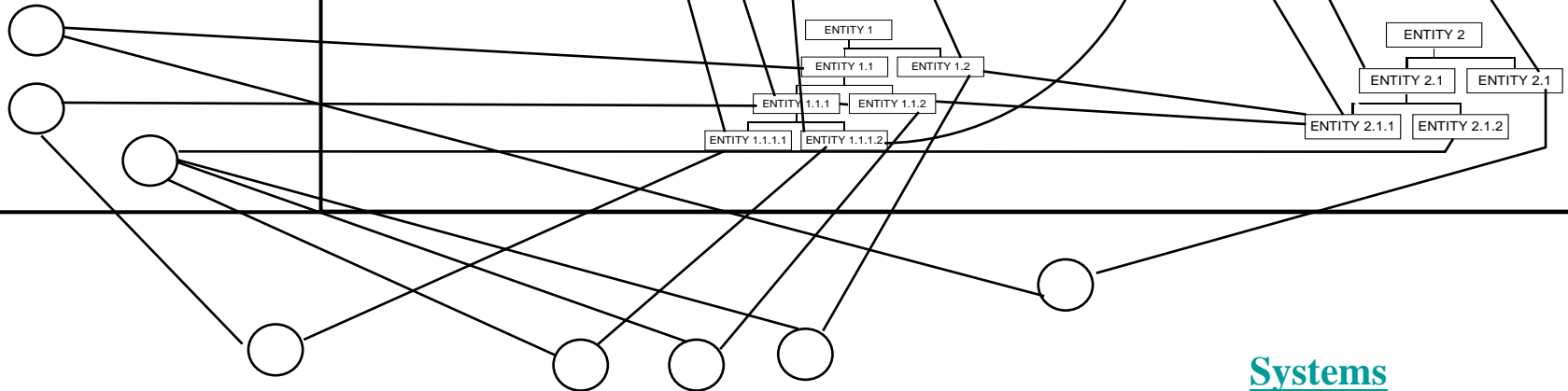


Issues/Measures



Actions

Knowledge

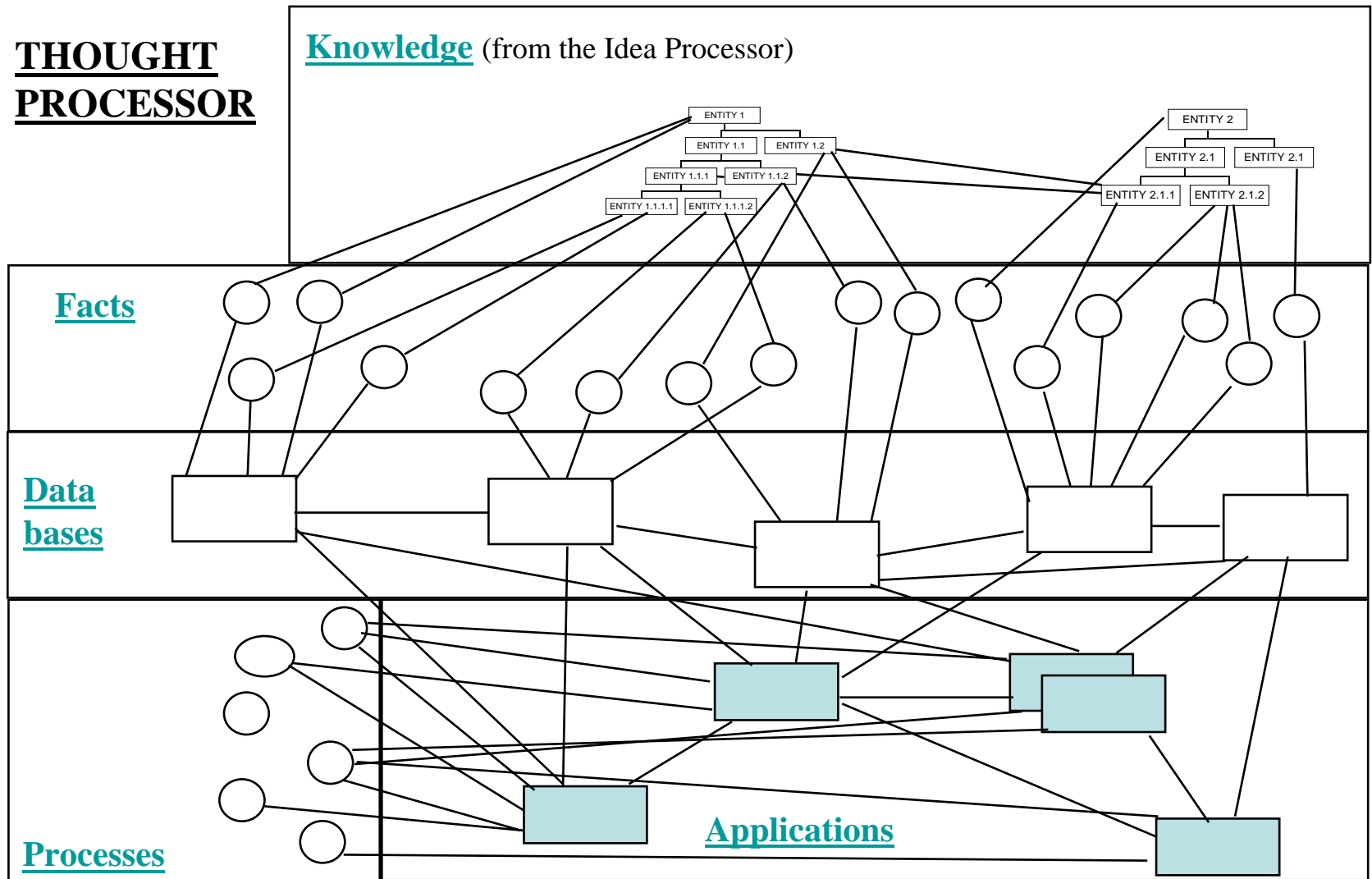


Systems

The R.I.P.O.S.E schematic – Part II

THOUGHT PROCESSOR

Knowledge (from the Idea Processor)



The genus of R.I.P.O.S.E

1989 – date: R.I.P.O.S.E was developed by [Charles M Richter](#). He achieved this by [transcending information engineering](#) into [information architecture](#), automating the [normalisation method](#) and transcending [computer languages](#) by developing a [pseudo code](#) engine. The next task is to create a series of [program generators](#) to translate the pseudo code into [executable code](#) – starting with [Smalltalk](#) and [Prolog](#).

July 1982 to January 1988: Charles was substantially involved with Information Engineering Pty Limited (IE). As Technical Director (1983 to 1987) he revamped the original ideas of [information engineering](#) (information and data analysis, data base design and process modelling) and designed and coded the IE software product (User:Data), which helped automate the IE methodology.

1978 to 1982: Charles was exposed to and used a number of software design methods to speed up the design of complex software programs. These included [Structured Analysis and Structured Programming](#) as well as the [Michael Jackson](#) and [Dijkstra](#) approach to developing programs from data structures.

1972 to 1978: Charles wrote programs in [COBOL](#) and [assembler](#). In 1976 he was taught to [normalise data](#).