Ripose Rapidly integrates patterns of strategic elements

Strategic planning to implemented solutions

Ripose Technique seven steps
White paper

This document is a white paper that describes the Ripose Technique seven steps.

Contents

Version 0.03/3
Date 24 November 2000

Prepared by Charles M Richter

Date printed 24 November 2000

This document was prepared using Word for Windows 97.
Printed in Australia.

Preface

Purpose

This document is a white paper that describes the Ripose Technique seven steps.

Intended audience

Document structure

Section 1 **Introduction** sets the scene for how the Ripose technique seven steps will ensure that an organisation will implement their systems successfully.

Section 2-8 **Steps 1–7** summarises the process for each step and provides estimates on the time that each step requires for completion.

1. Identify the people resources
2. Training the people
3. Build a conceptual architecture - vision
4. Build a logical architecture
5. Build several theoretical physical models
6. Promote the selected solution to management
7. Construct the physical architecture

Section 9 **Conclusion** provides a development vision.

Associated documents

Ripose technique grammar - fact sheet v0.02

Ripose technique - comparisons v0.04e

Ripose technique - patterns v0.03a

Introduction

Computerworld published three articles in the 16 September 1994 edition:

- How to be a strategic planner - Jennifer Roach
- Integration can still be a painful process - Rosemary Cafasso
- Software alone can't guarantee productivity - Peter De Jager

When person reads the articles above, it would appear that no logical connection existed between the three.

It is Ripose's assertion that *this is the major reason* very few organisations have successfully implemented any form of *strategically planned projects*. The fact that a number of very costly disasters have been documented, (ranging from $A2.5 million up to $US800 million) bears testimony to this statement.

The net effect is that whilst large organisations can continue to operate, despite writing off millions of dollars, smaller organisations go to the wall.

Each one of the above articles covers a specific spectrum of architecture, namely.

- Strategic planning with a conceptual architecture
- Integration with a logical architecture
- Software with a physical architecture

What can also be demonstrated is that the three architectures must be understood and used in an *exact* order if the seamless integration between strategic planning and physical systems is to be achieved.

The seven steps to achieve this seamless integration are:

1. Identify the people resources
2. Training the people
3. Build a conceptual architecture - vision
4. Build a logical architecture
5. Build several theoretical physical models
6. Promote the selected solution to management
7. Construct the physical architecture

1 – Identify people resources

Most of the people needed to build an integrated plan are already available in most organizations. What is required is to build a plan using each person's inimitable skill-set.

They are:

1. Senior managers
2. Middle managers
3. Operational managers
4. Workers
5. Project managers
6. Business analysts
7. System analysts
8. Data base designers
9. Programmers

The only person missing from this line up is the Information architect. This person should be a Business or Systems analyst and will have the overall responsibility for the quality of information placed into the knowledge repository.

The difference between large organisations and small businesses is the last four or five types of people. Hence a consulting organization whose consultants are trained in the seven step technique should be capable of providing these people.

2 – Training the people

Training will be required, however this should not take more than 15 days.

A minimum of eight days (with a maximum of 15 days) training in the art of:

- Strategic information planning
- Information architecture
- Systems architecture
- Data analysis
- Pseudo code design

Training to use the software tool provided with the Ripose technique is given during the courses.

3 – Build a conceptual architecture - The vision

This is achieved by undertaking the following tasks:

1. Identify the components of the conceptual architecture
2. Define and document the components of the conceptual architecture
3. Understand and agree with the components of the conceptual architecture
4. Have each senior manager rank the components of the conceptual architecture in terms of their - needs and wants
5. Develop a composite picture of all senior manager's ranking to identify the inherent weakness in the conceptual architecture
6. Have business analysts and middle management identify and build the indicators that senior management need to overcome the inherent weakness
7. Define and document these indicators
8. Understand and agree with these indicators
9. Have information architects, business analysts and business users, identify and build the conceptual knowledge base to support the agreed upon indicators
10. Have information architects and business analysts build the linkages between the components of the knowledge base
11. Identify the major business functions needed to support the conceptual knowledge base
12. Define and document these business functions
13. Understand and agree upon the business functions
14. Have system analysts identify and build the conceptual systems to support the business functions and the knowledge base
15. Define and document these conceptual systems
16. Understand and agree upon the priority of system implementation
17. Select the appropriate core and ancillary systems which the organisation requires
18. Reality check point.
    Build a proof of concept for each of the agreed upon systems. These should be easily understood by all levels of management

Using the Ripose technique and software, the following can be achieved

- The latest definition of the components of the vision are immediately available to every manager
- Tasks one through eight should be completed in less than 3 days. However, senior management must be prepared to participate
- Tasks nine and ten should be completed in not more than 4 weeks, however, middle management must be prepared to participate. Sessions with middle management should not take more than 1 to 2 hours
- Tasks 11 through 17 should be completed in 1 week
- Task 18 should take about 1 week per system (depending on the complexity)

4 – Build a logical architecture

This is achieved by carrying out the following tasks:

- Have data analysts identify the existing facts for each component of the required systems
- Define and document these facts
- Understand and agree upon these definitions
- Develop the current model of the business' data architecture

Using our method and software, the following is achievable:

- The latest definition of the components of the facts and data model are immediately available to every developer, wherever they may be
- These tasks should not take more than three weeks

5 – Design a theoretical technical solution for the business

This is achieved by carrying out the following tasks:

1. Extend the facts base
2. Develop a number of futuristic data models based on the conceptual knowledge base
3. Identify the processes for each model
4. Develop high level applications in Pseudo Code for each model
5. Reality check point. The end results of Tasks 1 through 4 should provide sufficient proof (of logical) that the construction phase of the system development life cycle will not lead to yet another failed system
6. Repeat tasks two through five should the selected logical data model fail to live up to expectations
7. Develop a conversion strategy for the most appropriate model
8. Carry out a cost benefit analysis to decide on a "build or buy" strategy - request for information etc.

Using our method and software, the following is achievable:

- The latest definition of the components of the facts and data model are immediately available to every developer, wherever they may be
- SQL Data base create statements are provided. Using our generic SQL data base domain parameters, a quick start SQL data base definitions can be achieved regardless of target data base
- Structured walk throughs can be conducted using the Pseudo Code Designer, prior to committing the design to a proprietary programming language
- These tasks should not take more than 8 weeks (depending on the complexity of the design)

6 – Sell the solution

Using the tasks described above, you should have more than sufficient evidence to:

- Convince your sponsors to go ahead with the project
- Continue to live with what they currently have and maintain the systems to the best of IT's ability
- Select a package that will best fit the selected strategic requirements and of one is not available, either outsource the design or build an "in-house" solution

7 – Construct the systems - physical implementation

This is achieved by carrying out the following tasks:

- Build the data base structure
- Build and test the conversion suite of programs
- Populate the new data base
- Build the top level menus for the business functions and systems in the same order as discovered in tasks 11 and 14 in step 3
- Build the applications based on the data base implementation plan that was developed during task 16 in step 3 and depending on the data base design selected in task 2 of step 5

It is now up to the individual organisation to use CASE or Object Oriented techniques to complete the job.

Conclusion

Our technique (referred to in previous sections) is called Ripose (Rapid Information Processing Orientated Systems Environment).

In short Ripose has been developed to harness, manage; and focus the power of the human imagination!

Identification

General

Title	White paper
Subject	Ripose Technique seven steps
Author	Peter H. Eburne
Version/revision	v0.03/22, 5 January 2000
Abstract	This document is a white paper that describes the Ripose Technique seven steps.
Keywords	White paper, Ripose technique, Seven steps
File name	E:\RIPOSE\RESEARCH\Ripose technique seven steps - White paper v0.02.doc

Creation details

Creation date	4 January 2000, 14:43

Modifications details

Change date	5 January 2000, 21:10
Changed by	Peter H. Eburne
Last printed	24 November 2000, 07:27

Revision history

Version	Date	Change description
v0.02	5 January 2000	Initial release
v0.03	24 Nov 2000	Table of contents moved to first page