Prototyping

Prototyping Model

- Before starting actual development,
  - a working prototype of the system should first be built.
- A prototype is a toy implementation of a system:
  - limited functional capabilities,
  - low reliability,
  - inefficient performance.

What is a Prototype?

A prototype is a working model of an information system containing sufficient features to demonstrate the functionality of the system.

It can help one to determine the deficiencies which can be mitigated later in the system.

Reasons for developing a prototype >

- **For the customer:**
- Illustration of input data formats, messages, reports, or interactive dialogs.
- So, the possible gap between specification and actual system can be bridged

Many user requirements get properly defined and technical issues get resolved: these would have appeared later as change requests and resulted in incurring massive redesign costs.

Reasons for developing a prototype >

- **Development Productivity 1:**
-
- Examine technical issues associated with product development:
- Major design decisions depend on issues like:
  - response time of a hardware controller,
  - efficiency of a sorting algorithm, etc.

Reasons for developing a prototype >

- **Development Productivity 2:**
-
- Time wasted in building incorrect system is minimized
- design errors are minimized
- So,
- development work does not have to be redone

- Shows Operational Feasibility
  - serves to verify the operational feasibility of a designed Information system
  - users' reaction & response to the system can be evaluated for future improvements

Manik Chand Patnaik, RIT.

- Helps Redevelopment
  - A prototype helps in recognizing the needs and deficiencies at an early stage
  - such missing aspects can be systematically added to the system

Manik Chand Patnaik, RIT.

- User enthusiasm for prototypes
  - Instead of conceptualizing a final complete system, users may be curious to see some form of the system at an early stage of development
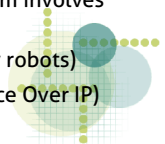
Manik Chand Patnaik, RIT.

- Building Candidates
  - Suitable when requirements are not known for new applications (automated help desk)
  - when requirements need evaluation
  - when the final resources, human effort and time required to implement a system involves high cost (Core banking)
  - high risk applications (operation by robots)
  - use of new technologies (VOIP: Voice Over IP)

Manik Chand Patnaik, RIT.

THE FINAL REASON

- The last reason for developing a prototype is:
  - it is impossible to "get it right" the first time,
  - we must plan to throw away the first product if we want to develop a good product.

Manik Chand Patnaik, RIT.

CHARACTERISTICS OF A PROTOTYPE

- a prototype is a live, working model
- helps in verifying the system features
- can be created in less time
- evolve through an iterative process
- inexpensive to build

Manik Chand Patnaik, RIT.

Building a Prototype >

- Start with approximate requirements.
- Carry out a quick design.
- Prototype model is built using several short-cuts:
  - Short-cuts might involve using inefficient, inaccurate, or dummy functions.
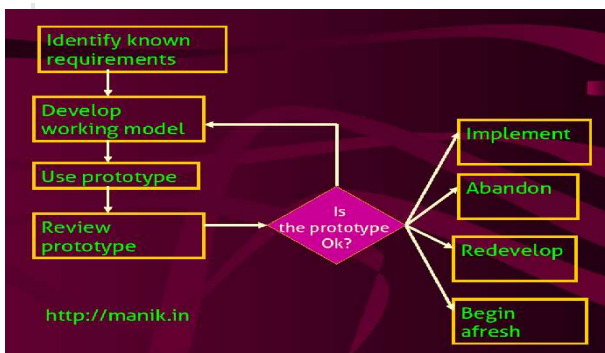 - A function may use a table look-up rather than performing the actual computations.

Manik Chand Patnaik, RIT.

Building a Prototype >

- The developed prototype is submitted to the customer for his evaluation:
  - Based on the user feedback, requirements are refined.
  - This cycle continues until the user approves the prototype.
- The actual system is developed using any waterfall based approach.

Manik Chand Patnaik, RIT.

Decision on a Prototype

Identify known requirements → Develop working model → Use prototype → Review prototype → Is the prototype Ok? → Implement / Abandon / Redevelop / Begin afresh

http://manik.in

Step-1 Requirement Gathering

- User and system analyst work together to identify the known requirements that must be met
- determine the purpose of the system
- determine the scope of the system's capabilities

Manik Chand Patnaik, RIT.

Step 2 Quick Design

- Develop a plan
  - fix a startup time & tentative completion time
  - describe documents to be produced & their layout
  - estimate the cost of prototyping
- components of a prototype to design
  - command language dialogue to facilitate interaction between user & system (user-system interface)
  - input screens & formats
  - processing modules
  - system output

Manik Chand Patnaik, RIT.

Step 3 Build

- Prototyping tools
  - 4GL
  - application generators
  - program generators
  - reusable code
  - application packages

Manik Chand Patnaik, RIT.

4ᵀᴴ Generation Language >

- 4GLs are helpful in developing application prototypes efficiently
- user has to specify "what is to be done" rather than "how to do it"- higher-level commands are replaced by lower-level instructions
- non-procedural languages

Manik Chand Patnaik, RIT.

3 categories of 4GLs

- Non-procedural language
  - also called the front-end
  - SORT, SELECT, LOCATE etc.
  - usually embedded in query languages and report generators
- Query language
  - used to make retrieval requests on stored data
  - one can also enter data and modify a file/DB
  - e.g., INTELLECT, Query-By-Example, Quick Query, SQL
- Report generators
  - to extract data from files and DBs (can't modify)
  - gives control over the appearance and content of the output
  - reports can be customized by giving titles, column headings .. and specifying field names
  - e.g., GIS, NOMAD, Mark IV, Easytrieve Plus

Application Generators

- These are software programs that allow one to specify an application at a higher level which can be converted into source code.
- e.g., ADS, FOCUS, MANTIS, NATURAL, RAMIS, MAPPER, Application Factory

Manik Chand Patnaik, RIT.

Screen Generators

- It is an interactive facility for arranging items on the screen and performing automatic input validation and processing.
- Screen Designers

Manik Chand Patnaik, RIT.

Data Dictionary System

- It stores definitions and descriptions of the data used in an organization which can be used during prototyping for consistency.

Manik Chand Patnaik, RIT.

Reusable Code

- Modules of codes can be written & maintained in a library
- such pre-written, pre-tested modules can be reused with little or no modification
- helps in quick & inexpensive application development

Manik Chand Patnaik, RIT.

STEP 4 CUSTOMER EVALUATION

- The product is evaluated by the customer.
- Customer provides valuable feedback about the prototype.

REFINEMENT

- Requirements are refined by using inputs from the customer.

CYCLE OF FEEDBACK AND DESIGN

- What follows is a cycle of design, evaluation, feedback, refinement and again design... till the customer is satisfied with the results.
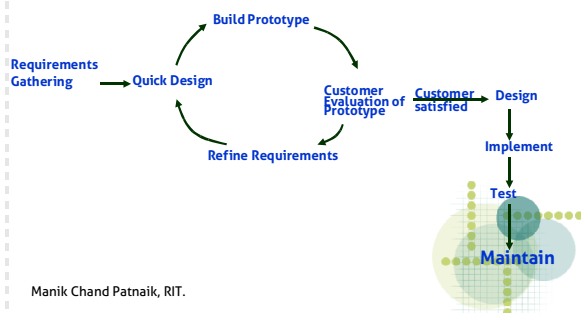
TRADITIONAL DEVELOPMENT

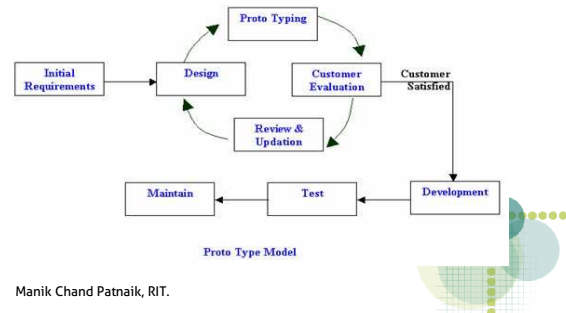- After the satisfaction of the customer, the prototype is redeveloped using traditional SDLC model

DIAGRAM OF PROTOTYPING PHASES >

Build Prototype

Requirements Gathering → Quick Design

Customer Evaluation of Prototype — Customer satisfied → Design

Refine Requirements

Implement

Test

Maintain

SAME DIAGRAM DRAWN DIFFERENTLY

Proto Typing

Initial Requirements → Design → Customer Evaluation — Customer Satisfied

Review & Updation

Maintain ← Test ← Development

Proto Type Model

PROTOTYPING STRATEGIES

- Prototype the screens
  - it enables one to evaluate the screen design to understand users' reaction to amount of information on the screen
  - display info. as per need (higher to more detail level)
- prototype the procedures
  - input, computation, info. retrieval and output activities
- prototype the mainline functions
  - build only important functions

Manik Chand Patnaik, RIT.

REQUIREMENT ANALYSIS?

- Requirements analysis and specification phase becomes redundant:
  - final working prototype (with all user feedbacks incorporated) serves as an animated requirements specification.

Manik Chand Patnaik, RIT.

REUSE IN PROTOTYPE?

- Design and code for the prototype is usually thrown away:
  - However, the experience gathered from developing the prototype helps a great deal while developing the actual product.
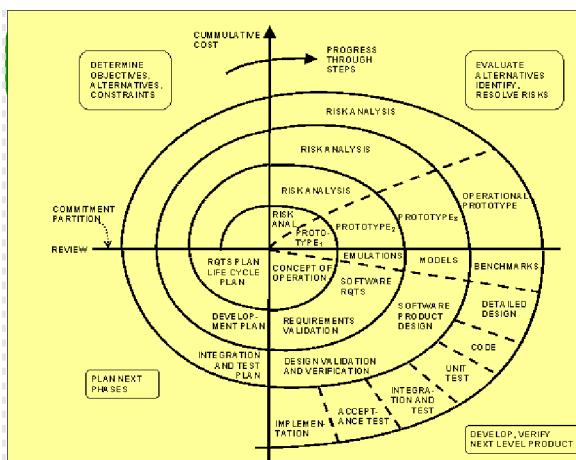
Manik Chand Patnaik, RIT.

SOME INFO

- Even though construction of a working prototype model involves additional cost --- overall development cost might be lower for:
  - systems with unclear user requirements,
  - systems with unresolved technical issues.

Manik Chand Patnaik, RIT.

SUITABILITY OF PROTOTYPING

- **Prototype model is suitable for projects not well understood and for**
- **Interactive Projects focusing on end-users (usability) in mind**

Manik Chand Patnaik, RIT.