Hybrid Development Methodology for Client-Server Applications

Václav řepa

Prague University of Economics
Czech Republic
Hybrid Development Methodology for Client-Server Applications

Structure of presentation

- What is the PDIT methodology
- Underlying principles
- Main features of the PDIT
- Procedure of analysis in the PDIT
- Model of reality - central part of the analysis
PDIT - IS Development Methodology

- target environment for the IS:
  - relational database system
  - client-server DB technology

- development environment:
  - CASE tool supporting the Structured Analysis and Design
  - GUI oriented 4GL
Three Architectures

Model of reality

Technological model

Implementation model

Conceptual level

Technology level

Physical level

Design Techniques and Tools

Implementation Techniques and Tools
PDIT - IS Development Methodology
Main Features

- **Hybrid Methodology** (structured as well as OO)
  - OO nature of the CS technology
  - Structured nature of the Global Analysis and
  - Separation of the data and processes in the target environment

- **Prototyping** as the natural approach for the 4GL tools using

- **Conceptual origin of the Client-Server partitioning**
Transition From the Conceptual to the Technological Architecture

Conceptual Layer
- Object Model (Real World Structure)
- Interface Functions Structure
- Real World Structure (static view)
- Real World Behaviour (Object Life Histories and Objects Communication)
- Program Modules Structure
- Logical Database Structure
- Usability Requirements
  - * I/O functions
  - * Dialog functions
  - * Interface functions etc.
  - * Function hierarchy (collectivization)
  - * I/O functions
  - * Dialog functions
  - * Interface functions etc.
  - * Function hierarchy (collectivization)

Technology Layer
- Logical Server
- Logical Client
- Top-Down Function Structure
  - * user procedures
  - * presentation needs
- Transition Techniques (Design Rules)
  - * objects
  - * attributes
  - * life history
  - * hierarchy of objects (generalization)
  - * relationships among objects
    - * attributes
    - * mutual behaviour
- Technology Constraints and Requirements and Database Design Rules
  - * hierarchy of objects (generalization)
  - * relationships among objects
    - * attributes
    - * mutual behaviour
  - etc.

Vaclav Repa: Hybrid Development Methodology for Client-Server Applications BIT95
Relations between Conceptual Design and Client-Server Partitioning

Conceptual Model (Product of Analysis)

Data Model
Function Model

Technological Model (Product of Technical Design)

Output Function (reaction on the entity life state)

Processes of the Entity Life Model (storing the data about the events)

Output Function (reaction on the data request)

Client Procedures
Server Procedures

Data requests
Triggers
Events
Data
Events
Data requests
Global System Analysis
- Global Business Structure
- User Requirements
- Global Data Model

Detailed Analysis and Design Integration
- Detailed Data Model
- Events and Workflows
- Model of Reality
- Supporting and Out-of-Dialog I/O Processes
- GUI Prototype
- Increment Integration

Detailed System Analysis

Vaclav Repa: Hybrid Development Methodology for Client-Server Applications BIT95
Model of reality

- a set of Data Flow Diagrams (DFD) and
- an Entity Relationship Diagram (ERD)

**Entity Behaviour Model**
describes the behaviour of one object (entity or relationship) from the Data Model (ERD).

- **entity** (or relationship in the sense of associative entity) from the Data Model (data part of the object)
- **several essential functions** (actions of object state changes)
- **one control process** (STD) (object life history)
- **control flows** between the STD and essential functions (events and actions of the object).
  - **input** control flow (event stimulating the transition of the object from one state to the other one)
  - **output** control flow (action of the object as the reaction on the even)
- **data flows**
  - from the input/output interface to the essential function of the Entity Behaviour Model. (data part of the real-world event)
  - from (or to) the essential function of the other object (communication of the Entity Behaviour Models - transition of the information about real-world events between objects)
Mapping of the Conceptual Model to the Client-Server 3-tier Architecture

- Entity Relationships Diagram
- Model of the Entity Internal Behaviour
- Model of the Entities Communication
- I/O functions and User Procedures Models
- User Objects

Entities (static real-world structure)
Entities life cycles
Real-world behaviour
Application behaviour

Data Logic
Application Logic
Presentation Logic

Vaclav Repa: Hybrid Development Methodology for Client-Server Applications BIT95