# IBM Rational Software Development Konferenz

**The next generation is about to start…**

## Model Driven Product Engineering Process

### S Sivakumar, Astra Infotech Pvt. Ltd.

**Holiday Inn – München**
**4. - 5. September 2008**

# Agenda

- Introduction
- Product Engineering
  - Model Driven Development
  - The Process
  - The Transformations
- Demo and Case Study

# Introduction

- About Astra
  - The Software Engineering Company
  - IBM Business Partner (Advanced)
  - Services
    - Software Engineering Consulting & Training
    - IBM Rational tools implementation
    - Outsourced Software Testing
  - Documented case studies
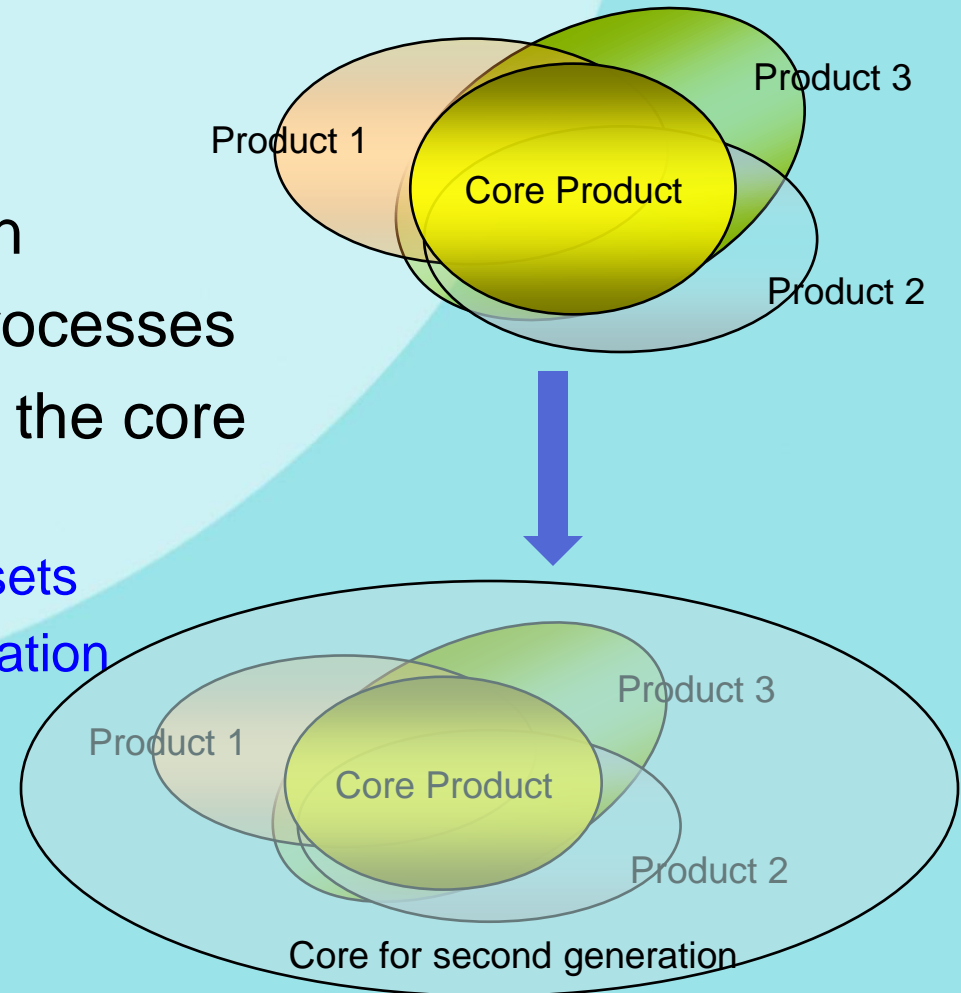    - http://www-306.ibm.com/software/in/rational/astra_casestudy

# Introduction

- ## Why Astra?
  - – Small but, highly skilled, focused and experienced team
  - – More than 20 years of industry exposure with more than 100 years of collective experience
  - – Experience from variety of applications, organization and verticals
  - – Team scalability though various internal academic initiatives
  - – A well balanced, demonstrated software engineering ecosystem in place

# Agenda

- Introduction

- Product Engineering
    - Model Driven Development
    - The Process
    - The Transformations

- Demo and Case Study

# Software Product Development

- Core set of features
- Address a specific domain
- Automate the business processes
- Product lines built around the core
  - Represent customizations
  - Share a common set of assets
  - Core evolves to next generation



Product 1 · Product 3 · Core Product · Product 2

Product 1 · Product 3 · Core Product · Product 2 · Core for second generation

# Product Line Objectives

- Establish and maintain a market presence
- Respond to market quickly
- Large scale productivity improvement
- Systematic reuse of existing assets
- Reduced production cost

Need to follow engineering principles

# Product Line Imperatives

- Organizational structure and management
- Domain expertise
- Vision
- Development process based on sound software engineering
    - Process should be architecture centric
    - Development and management of core reusable assets
    - Close collaboration among teams

# Influence of Architecture
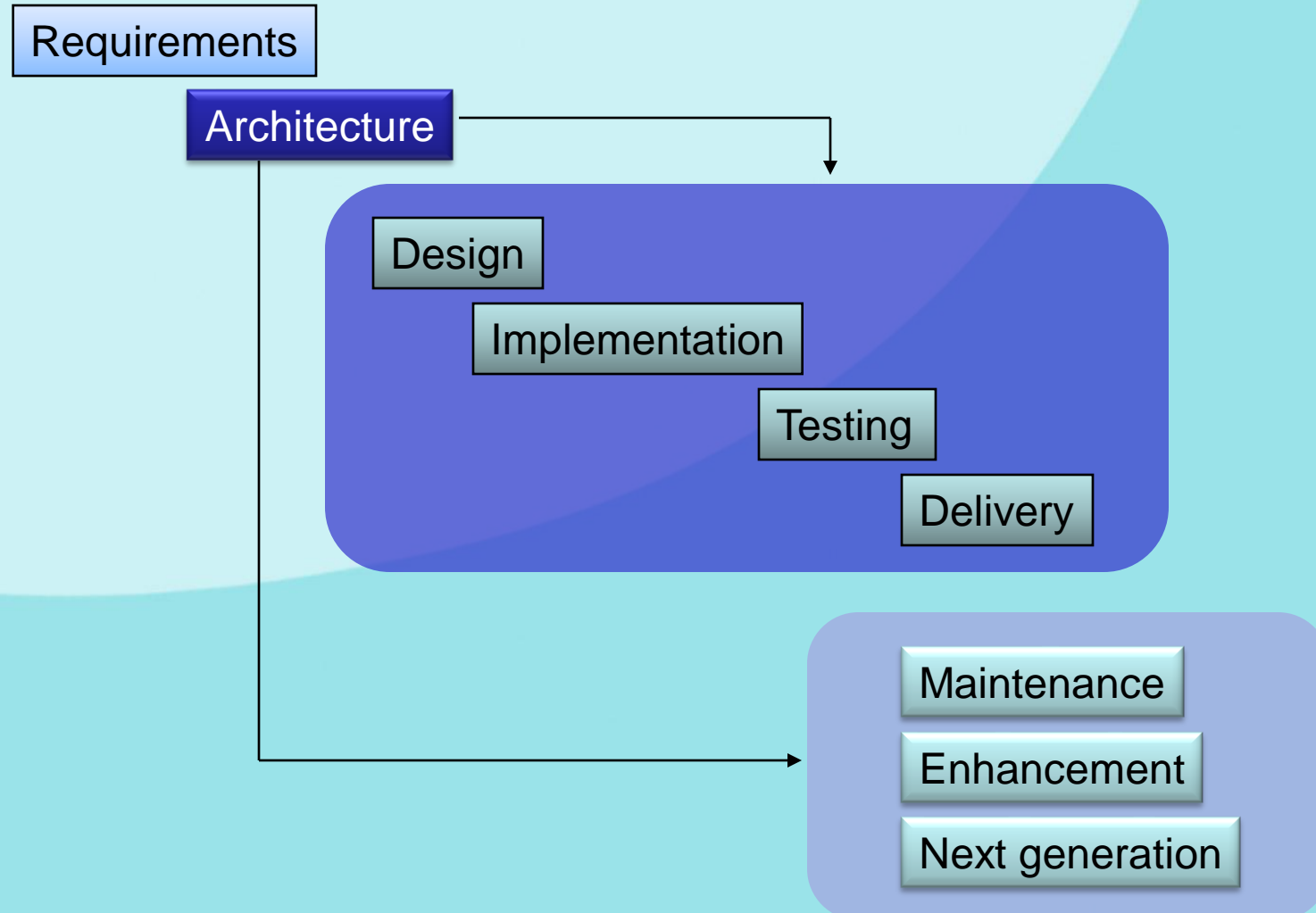
# Agenda

- Introduction
- Product Engineering
  - Model Driven Development
  - The Process
  - The Transformations
- Demo and Case Study

# Model Driven Development

- Models are the development artifacts
- Progress by developing a series of models
  - Provide a way to progress from one model to the other
- Provides inherent traceability and consistency
- Reduces effort
- Facilitates automation
- What models are required?
  - Depends upon the problem
    - Business – Use case – Analysis –
    - Architecture – Design
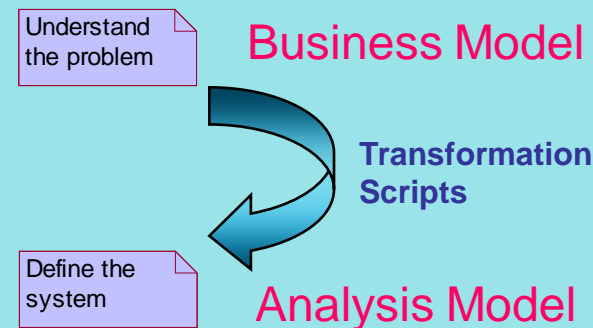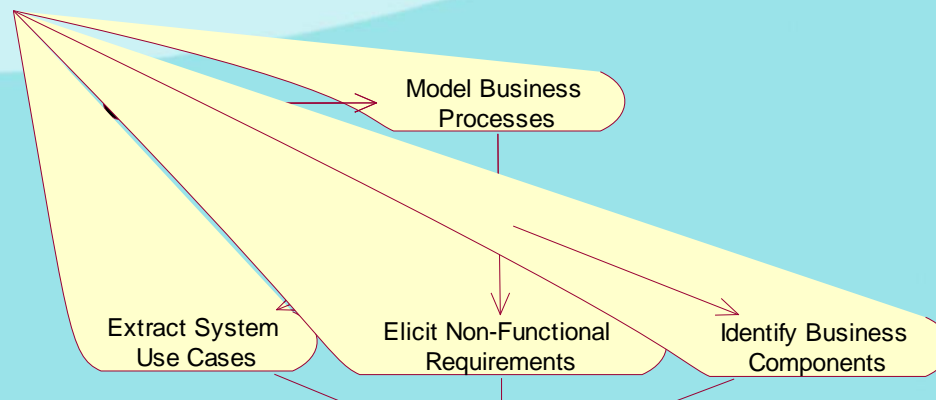  - Automation of model transformation
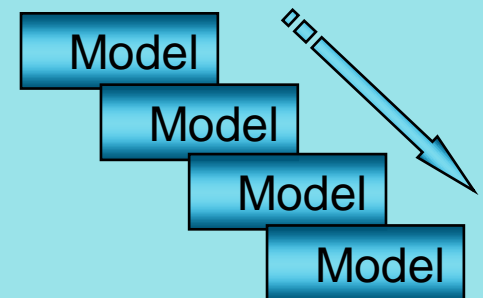
# Understand the Problem

- Model the business processes
  - Provides the context for the system
  - Helps understand the problem
  - Provides the basis for model transformation
  - Recommended starting point

# System Requirements

- Functional - Modeled as system use cases
  - Starting point for small applications
  - Extracted from business processes
    - System use cases
    - Business components
      - Architectural elements

- Non-functional - Documented

**Transformation Scripts**

Model Business Processes

Extract System Use Cases

Elicit Non-Functional Requirements

Identify Business Components

Understand the problem

Define the system

**Business Model**

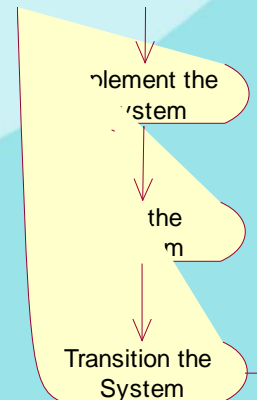**Transformation Scripts**

**Analysis Model**

# System Structure

- System Architecture
  - Multi-tier structure proposed
  - Subsystems from
    - Use cases and business components
    - Architectural mechanisms
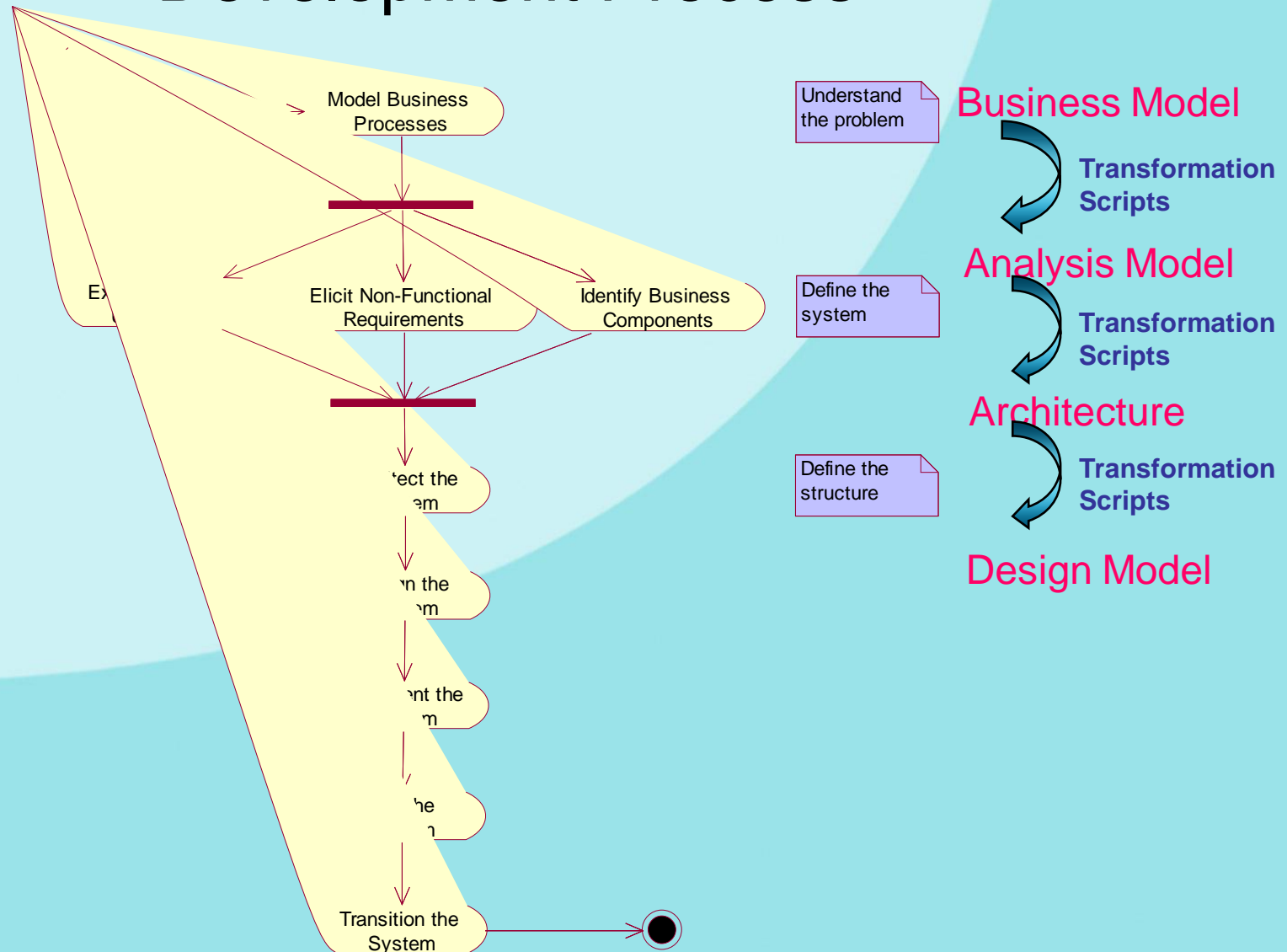      - Balance among non-functional requirements

Analysis Model

Transformation
Scripts

Architecture

Transformation
Scripts

Design Model

ect the
m

Design the
System
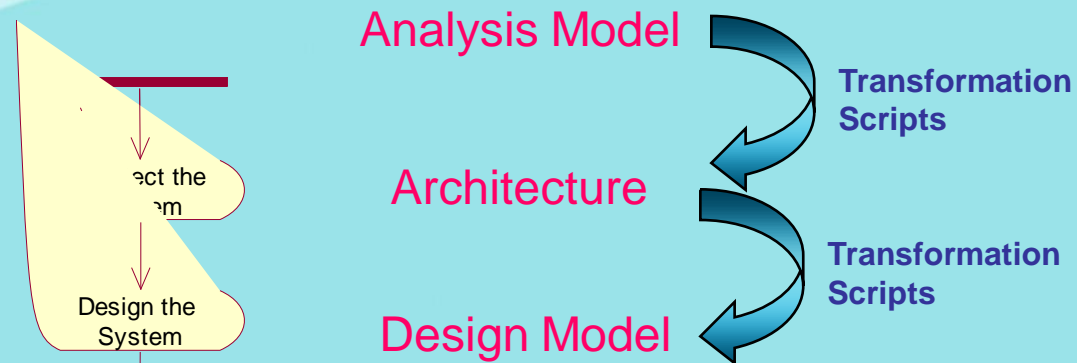
# System Realization

- Design model
  - Forward engineered
- Implementation
- Testing
- Delivery

plement the
stem

the
m

Transition the
System

# Development Process

# Traceability

*Engineering "Software Engineering" – The Astra Way !*
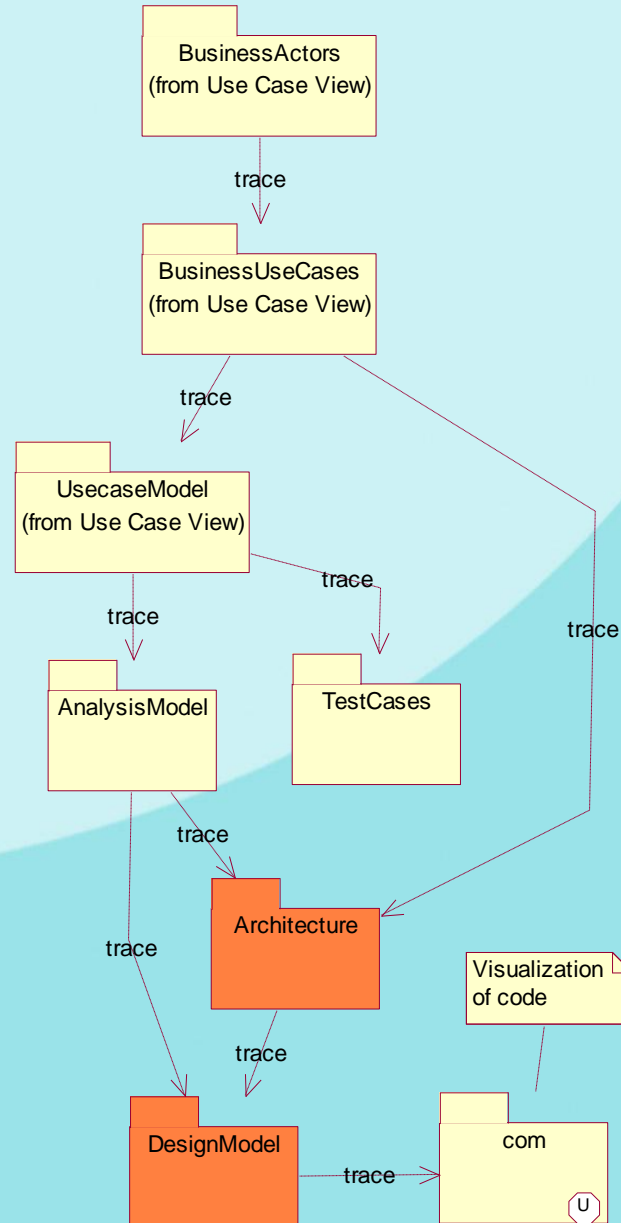
# Tools

- Modeling
  - IBM Rational Software Architect
    - Facilitates model driven development
    - Transformation scripts written using the APIs
      - Developed as a plug-in

- Requirements Management
  - IBM Rational RequisitePro

- Change and Test Management
  - IBM Rational ClearQuest Test Manager

# Benefits

- Less effort and time for
  - Change implementation
  - Product customization and deployment
  - Development of different versions
  - Induction of new members into the team
- Architectural consistency
  - Harvesting reusable assets made easier
  - Same look, feel and behavior for all products
  - Automation enabled
    - Easier metrics and reports collection
    - Integrations across tools

# Agenda

- Introduction
- Product Engineering
  - Model Driven Development
  - The Process
  - The Transformations
- Demo and Case Study

# Plug-in Screen Shot
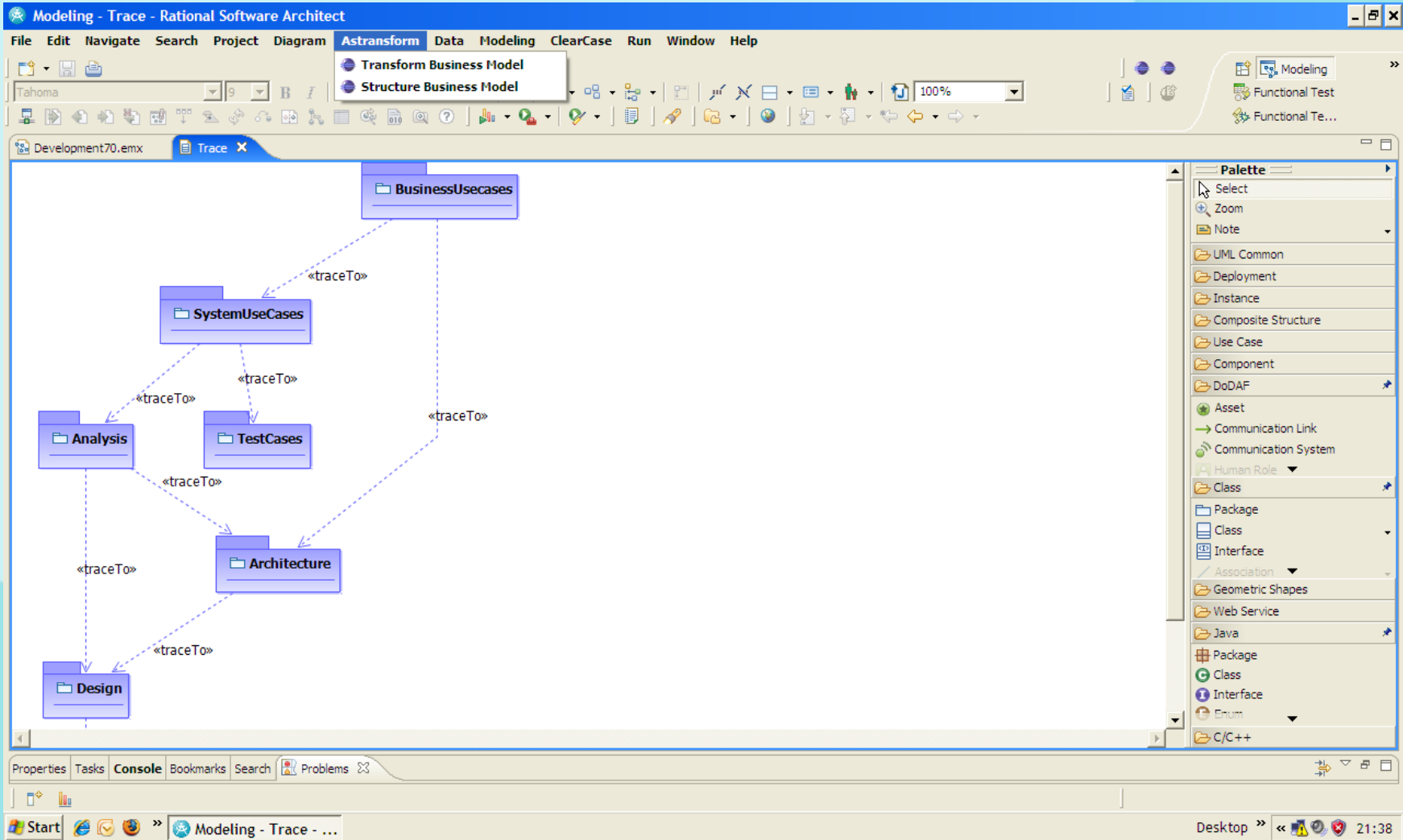
# Case Study

- Customer developing products for billing for different domains

- Knowledge existed in the minds of experts
  - Experts guided the team to develop the code
  - Resulted in large customization and change management efforts
  - The management missed the 'big picture' of development
  - Educating the new inductees became costlier

# Case Study

- Process adapted
  - Business process model documented by extracting information from domain experts
    - Enabled the 'big picture' to the management
    - Enabled the team members in understanding the context
  - Transformations applied to arrive at system use cases
    - Resulted in system requirements model
    - Enabled easier requirements management
  - Transformation provided the basis for architecture and design
    - Business components identified
    - Logical design guided by architecture
    - Existing code base re-factored as per architecture

# Case Study

- Benefits
  - Business Processes documented
    - Team enabled with proper context
  - Easier Customization and change management
    - Initial studies indicate 30-60% savings in effort
  - Enabling the new team members
    - New members became productive in less time
      - Maximum savings attained here was 50-70% on time to learn the requirements and structure of the products
  - Enabling the development managers
    - Objective prediction and metrics collection

# References

- Patterns:Model Driven Development using IBM Rational Software Architect
  - IBM Redbook – sg247105 – 2005
- Use Models, Don't Worship Them
  - James Rumbaugh, Rational User Conference 2003
- Modeling for enterprise initiatives with IBM Rational Unified Process
  - Peter Eeles and Maria Ericsson, Rational Edge, Oct 2003.
- Rational Unified Process 2003—the Business Modeling Workflow

A

Great Idea

Effective Association

Successful Implementation!

# Thank You!

Dr S Sivakumar
Astra Infotech Pvt Ltd, Hannover

ssk@astrainfotech.com