



Effective Test Automation of SAP Implementations

Vipin Kumar

Managing Director & Software Engineering Evangelist
Astra Infotech Pvt Ltd

vk@astrainfotech.com

Rational. software

QM15

Agenda

- Introduction
- The Project
- The Solution
 - ▶ Highlights
 - ▶ The Process
 - ▶ The Architecture
 - ▶ Challenges and Solutions (Illustrative)
 - ▶ Benefit

Agenda

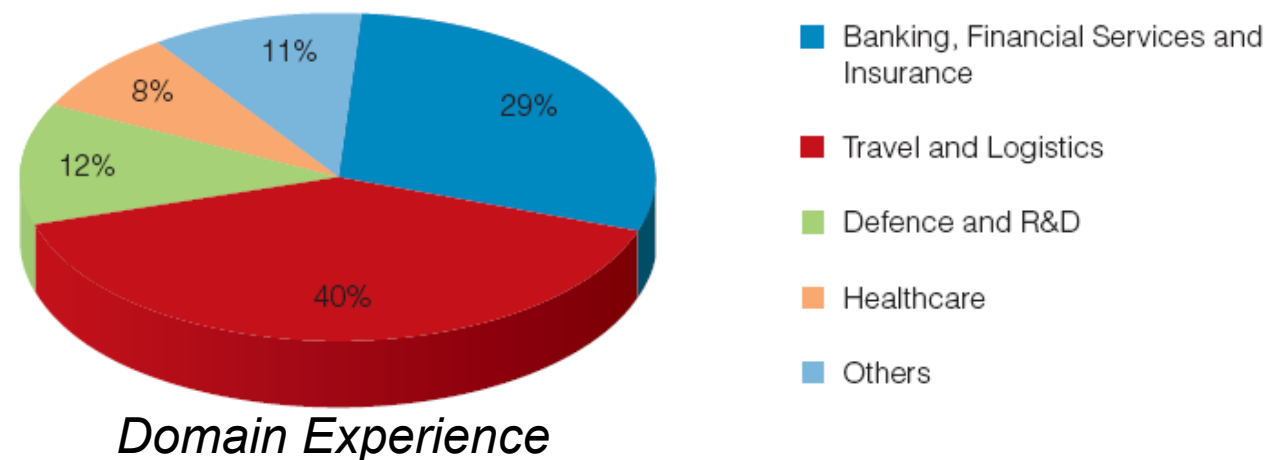
- **Introduction**
- The Project
- The Solution
 - ▶ Highlights
 - ▶ The Process
 - ▶ The Architecture
 - ▶ Challenges and Solutions (Illustrative)
 - ▶ Benefit

Introduction

- About Astra Infotech Private Limited
 - ▶ The Software Engineering Company
 - ▶ IBM Business Partner (Advanced)
 - ▶ Services
 - Software Engineering Consulting & Training
 - IBM Rational tools implementation
 - “Ready for Rational” plug-ins [Astra Model Creator](#) and [Astra Test Automat](#)
 - Outsourced software testing
 - Software engineering automation & tools integration
 - ▶ Documented case studies
 - http://www-306.ibm.com/software/in/rational/astra_casestudy
 - http://www-01.ibm.com/software/success/cssdb.nsf/CS/STRD-7M6NRE?OpenDocument&Site=default&cty=en_us

Introduction

- Why Astra Infotech?
 - ▶ 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 through various internal academic initiatives
 - ▶ A well balanced, demonstrated software engineering ecosystem in place

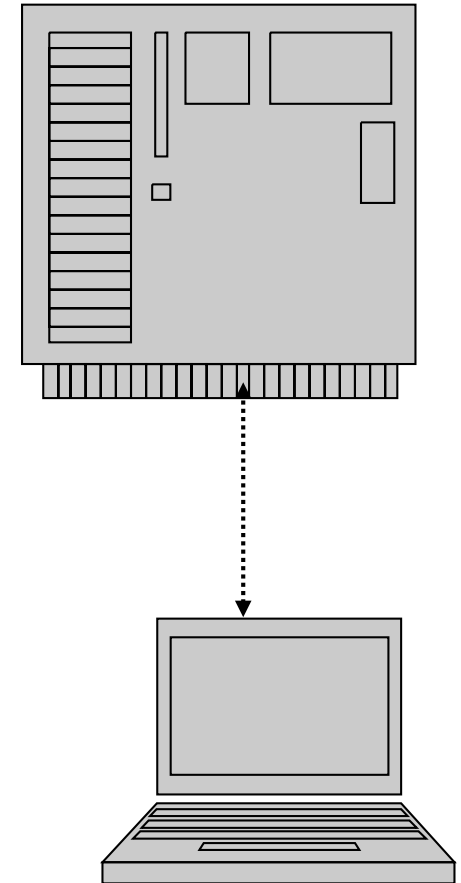


Agenda

- Introduction
- **The Project**
- The Solution
 - ▶ Highlights
 - ▶ The Process
 - ▶ The Architecture
 - ▶ Challenges and Solutions (Illustrative)
 - ▶ Benefit

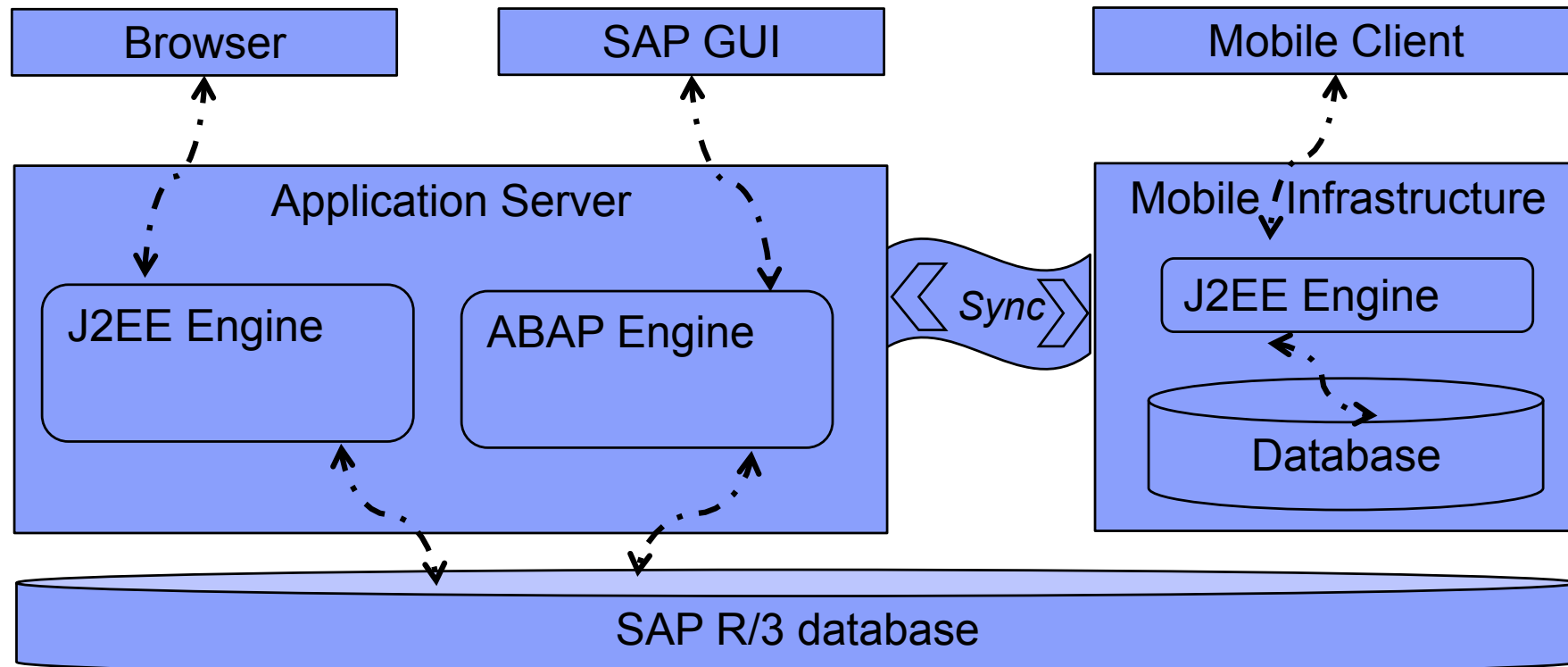
Software Test Automation Project

- About the Project
 - ▶ Project involves
 - Back office portal, for
 - Maintaining information on
 - » Customer and
 - » Equipments with the customer
 - Logging messages from customers, for services
 - Assigning work orders for service engineers
 - Mobile application used by service engineers
 - Updating work progress
 - Synchronizing with back office portal



Software Test Automation Project

- Project Architecture



Software Test Automation Project

- A business process (Example)
 - ▶ Placing a Work Order
 - Select a customer
 - Create or Select an equipment
 - Create or Select a message
 - Create a work order

Software Test Automation Project

- A Scenario (Example)
 - ▶ Placing a Work Order
 - Login
 - Search a customer by street and city
 - Select a customer from the list
 - Select an equipment with him
 - Create a message
 - Create a work order related to the message

Testing this scenario involves combination of test cases, using specific set of data values

Agenda

- Introduction
- The Project
- **The Solution**
 - ▶ Highlights
 - ▶ The Process
 - ▶ The Architecture
 - ▶ Challenges and Solutions (Illustrative)
 - ▶ Benefit

The Solution

- Test automation
 - ▶ A testing project
 - ▶ An automation project

The Solution

- Test automation is a testing project
- Successful testing requires
 - ▶ Testing with business perspective
 - ▶ Testing
 - As customers and users would
 - Aligned with engineering principles
 - ▶ Leveraging experience from industry
 - ▶ Following open, scalable, practical, measurable process

The Solution

- Test automation is an automation project
 - ▶ Managing test assets
 - Test plans
 - Test suites
 - Test cases
 - Test scripts
 - ▶ Establishing structure among test assets
 - ▶ Test architecture
 - Reusable assets
 - ▶ Maintaining traceability
 - Requirements
 - Changes
 - Defects

Agenda

- Introduction
- The Project
- The Solution
 - ▶ **Highlights**
 - ▶ The Process
 - ▶ The Architecture
 - ▶ Challenges and Solutions (Illustrative)
 - ▶ Benefit

Highlights

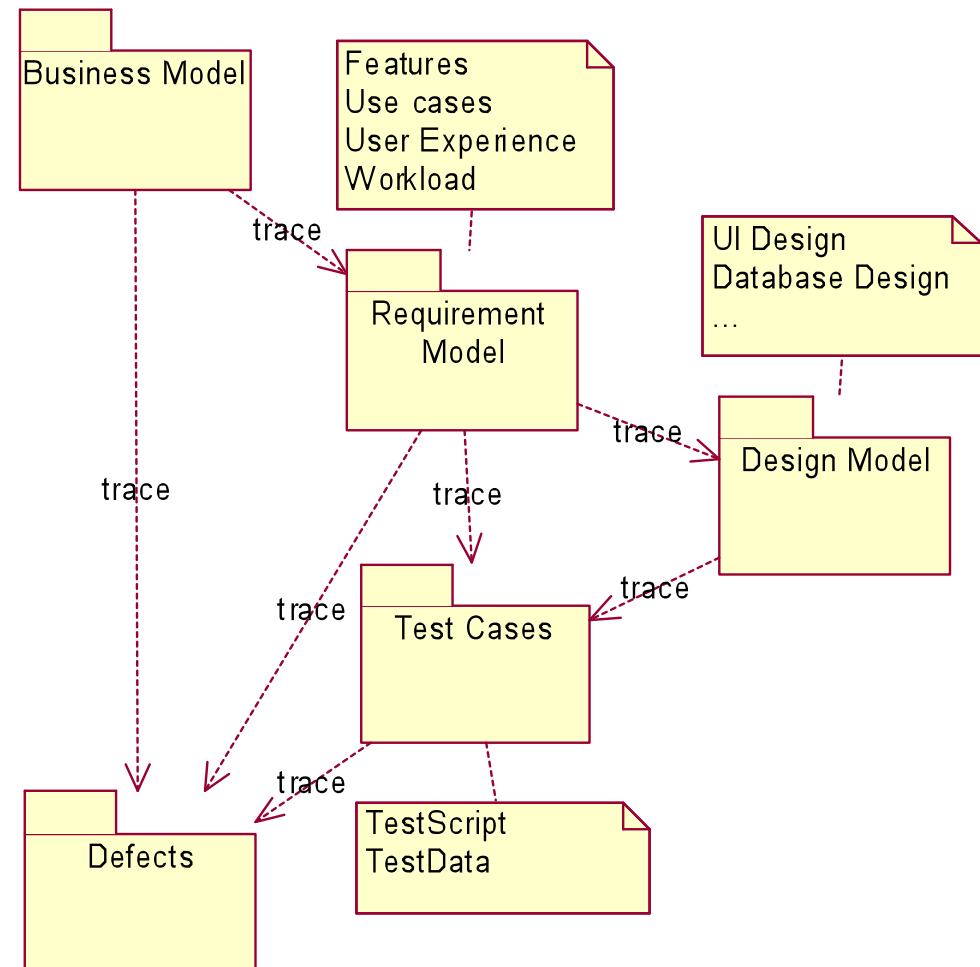
- Business process modeling
 - ▶ Helps to get the big picture and business perspective
 - Provides context for the system requirements
 - ▶ Understanding the business, not the automation part alone
 - Better value proposition
 - Easier adaptability to the live environment

Highlights

- Requirement Management
 - ▶ Clarity, classification and traceability
 - ▶ Effective control on changes
 - ▶ Enhanced quality, since better input controls
 - ▶ Customer and end user perspective

Highlights

- Traceability
 - Connecting islands
 - Connecting perspectives



Highlights

- Test Architecture
 - ▶ Provides structure to test assets
 - ▶ Enables reusability
 - ▶ Enables automation
 - Traceability from business models through requirements, test cases, test scripts, versions, releases and defects
 - Maintainability, with effective traceability and version control

Highlights

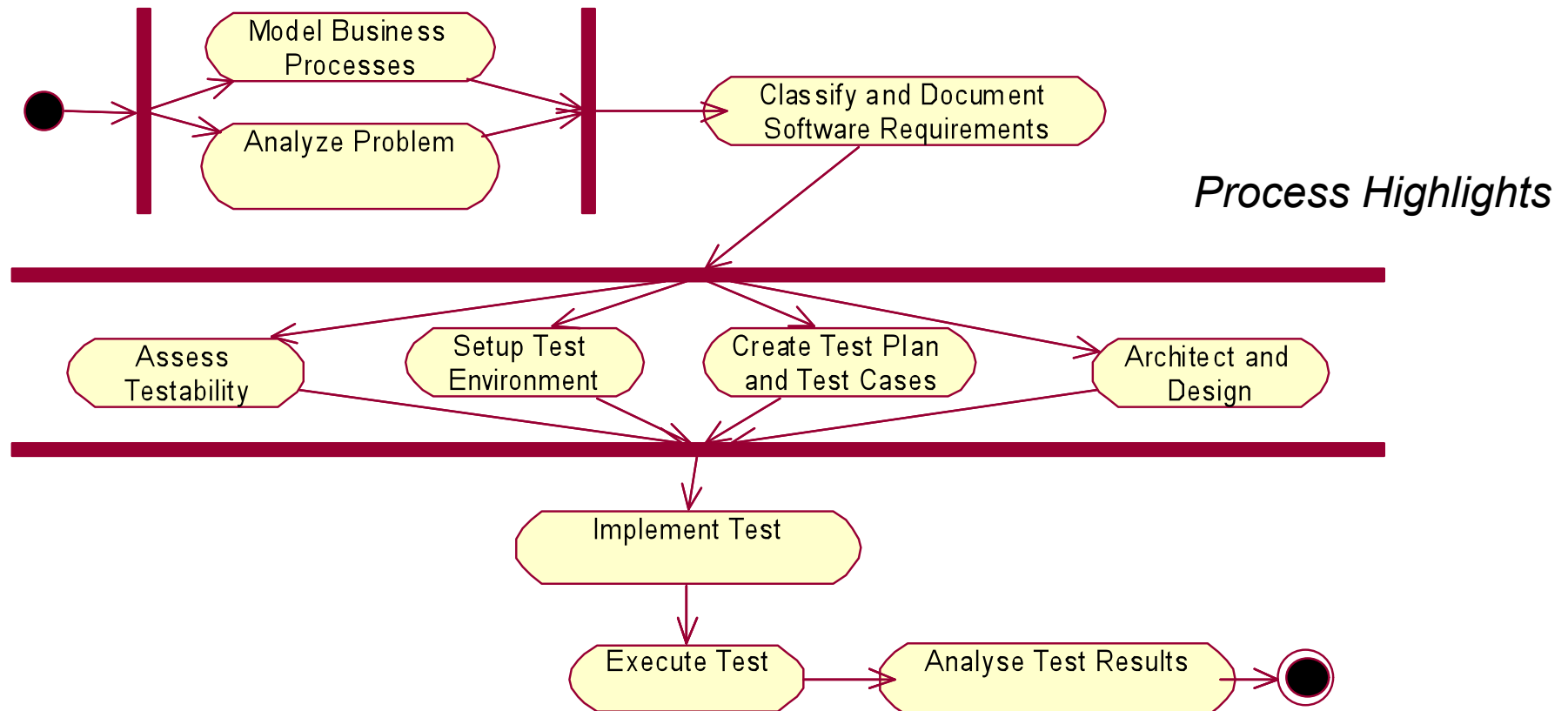
- Project Management
 - ▶ Iterative approach
 - Continuous focus on risk, leading to better project success
 - ▶ Effective control
 - Measurable, objective and metric driven approach

Agenda

- Introduction
- The Project
- The Solution
 - ▶ Highlights
 - ▶ **The Process**
 - ▶ The Architecture
 - ▶ Challenges and Solutions (Illustrative)
 - ▶ Benefit

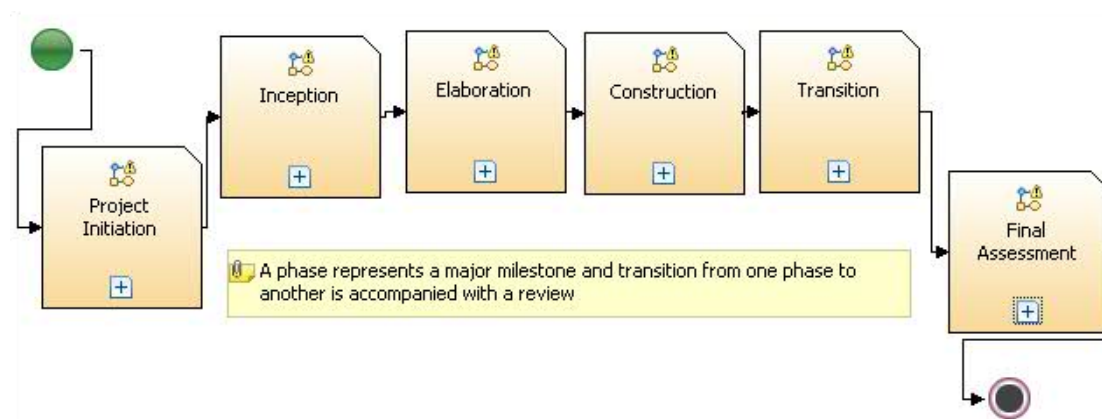
The Process

- Open, scalable, practical and measurable process



The Process

- Execution
 - ▶ Iterative model
 - Balancing stakeholder concerns and priorities
 - Continuous risk identification and mitigation
 - Short feedback cycles
 - Testing early and continuous

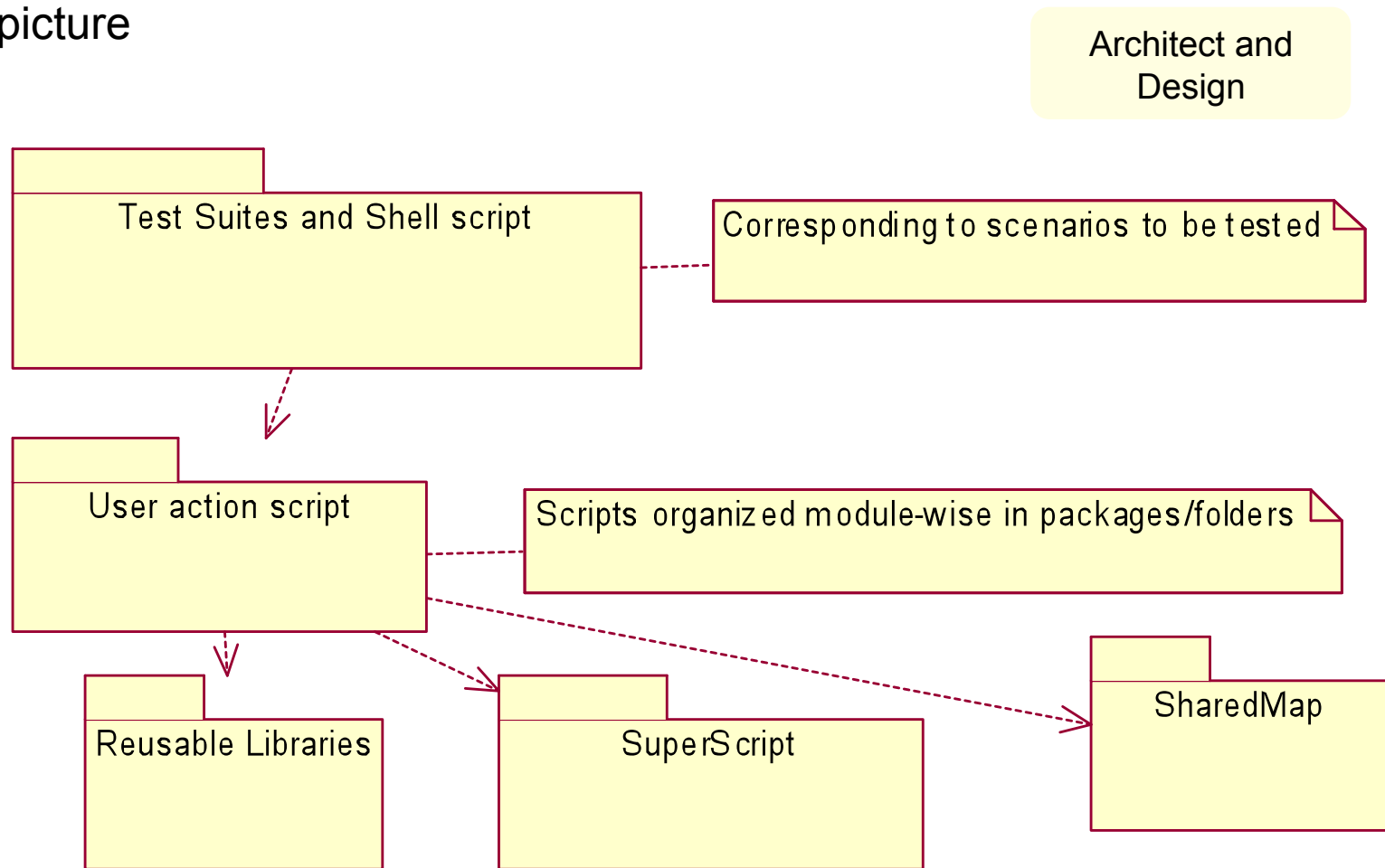


Agenda

- Introduction
- The Project
- The Solution
 - ▶ Highlights
 - ▶ The Process
 - ▶ **The Architecture**
 - ▶ Challenges and Solutions (Illustrative)
 - ▶ Benefit

The Solution Framework

- Simplified picture



The Solution Framework

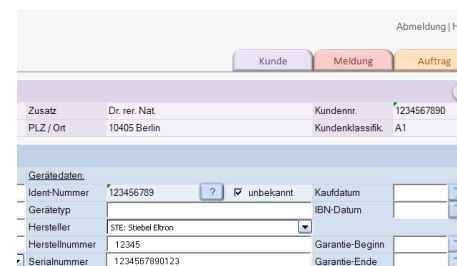
- Automation
 - ▶ Extraction of test cases from requirements
 - Implemented as Rational Functional Tester Plug-in
 - Requirements and Traceability managed in IBM Rational RequisitePro
 - Test plan, Test cases and Defects managed in IBM Rational Quality Manager

Agenda

- Introduction
- The Project
- The Solution
 - ▶ Highlights
 - ▶ The Process
 - ▶ The Architecture
 - ▶ **Challenges and Solutions (Illustrative)**
 - ▶ Benefit

Challenges and Solutions

- Challenge #1: Test automation tool selection
 - ▶ Success of test automation depends on the choice of tool
- Solution: "IBM Rational Functional Tester with SAP plugin identified as tool:
 - ▶ Recognition of GUI controls of SAP GUI, portal and mobile clients for performing actions and verifications
 - ▶ Support for power scripting using Java to handle dynamic actions and verifications
 - Use of object oriented programming language like Java for scripting provided full programming power for test automation
 - ▶ Potential for highly resilient scripts with ScriptAssure
 - Application user interface has been highly volatile, undergoing drastic changes
 - ▶ Extensive testing with wide variety of data with
 - Data parametrizing using data pools
 - Pattern matching using regular expression



Abmeldung | Hilfe

Kunde Meldung Auftrag

| | | | |
|-----------|--------------|------------------|------------|
| Zusatz | Dr. rer. Nat | Kundennr. | 1234567890 |
| PLZ / Ort | 10405 Berlin | Kundenklassifik. | A1 |

Gerätedaten:

| | | | | |
|------------------|-------------------|------------------------------------|-----------------|--|
| Ident-Nummer | 123456789 | <input type="checkbox"/> unbekannt | Kaufdatum | |
| Gerätetyp | | | IBN-Datum | |
| Hersteller | STE: Stebel Etron | | | |
| Herstellernummer | 12345 | | Garantie-Beginn | |
| Serialnummer | 1234567890123 | | Garantie-Ende | |



Challenges and Solutions (contd...)

- Challenge #2: Many scripts use more or less same set of objects
 - ▶ Example: Login
 - Login with wrong user name and password
 - Login with correct user name and password
- Solution: Object recognition information localized to script (default) leads to change in application interface requiring changes in many scripts
 - ▶ Such scripts were identified and shared Object Map were used in such cases

Challenges and Solutions (contd...)

- Challenge #3: Data verification
 - ▶ Example: Customer details
 - Search for customer
 - Select a customer from list
 - Verify selection in detailed page
 - Details displayed in a page, on hover
 - Entire details as a single data item
- Solution: Special action and scripting for data verification
 - ▶ Hover detected by the tool on instruction
 - ▶ Additional scripting done to tokenize and extract relevant data for verification

Challenges and Solutions (contd...)

- Challenge #4: Variety of ways, with minor variations, to accomplish a task.
 - ▶ Example: Search customer
 - By number
 - By Street and PIN
 - By Street and City
 -
- Solution: Modular scripts
 - ▶ Each of the test cases to be used, before related business process can be considered as tested, were identified and managed with modular scripts

Challenges and Solutions (contd...)

- Challenge #5: Automating test cases verifying similar tasks in various contexts
 - ▶ Examples: Search for customer, orders, etc results in a list spanning multiple pages
 - ▶ Identification of data searched for, from list of result pages
 - Data related to the data searched for may listed in any of the result pages
 - Tool need to identify that data and act upon
 - ▶ Such reusable logic cuts across many scenarios and scripts
 - ▶ Context and further actions varies in some cases
- Solution: Reusable, Modular scripts
 - ▶ Each of such test cases were identified and managed with reusable, modular scripts
 - ▶ Script to locate, parse and act up on were written as reusable, modular scripts

Challenges and Solutions (contd...)

- Challenge #6: Contexts, and variations in reuse
 - ▶ Example: Selection of an equipment involves
 - Search based on street and city
 - Tool need to identify that data and act upon
 - Selection of a customer
 - ▶ Similar selection would be involved in selection of a message or order
- Solution: Reusable, Modular scripts
 - ▶ Each of such test cases were identified and managed with reusable, modular scripts

Challenges and Solutions (contd...)

- Challenge #7: Variety of ways, with minor variations, and dynamic data
 - ▶ Example: Transforming from prospect to customer involves
 - Search and Browse as earlier followed by,
 - Selection of prospect from the list, with flag 'I' clicking on the flag
 - ▶ Recorded script cannot be reused
 - Specific prospect used while recording is by now transformed to customer
- Solution: Reusable, Modular scripts
 - ▶ Each of such test cases were identified and managed with reusable, modular scripts and dynamic data handling (with data pool and dynamic verification)

Agenda

- Introduction
- The Project
- The Solution
 - ▶ Highlights
 - ▶ The Process
 - ▶ The Architecture
 - ▶ Challenges and Solutions (Illustrative)
 - ▶ **Benefit**

Benefit

- Testing cycle time reduced from **three days to just two hours per fortnight**
- All tests are run automatically, with **no need for manual intervention**, leaving development team to focus on other priorities
- Total cost of test automation **reduced consistently** with each testing cycle
- **Quicker, easier and cost effective** scripts with the help of testing framework with reusable components
- High modularity & maintainability
- Evident & significant ROI

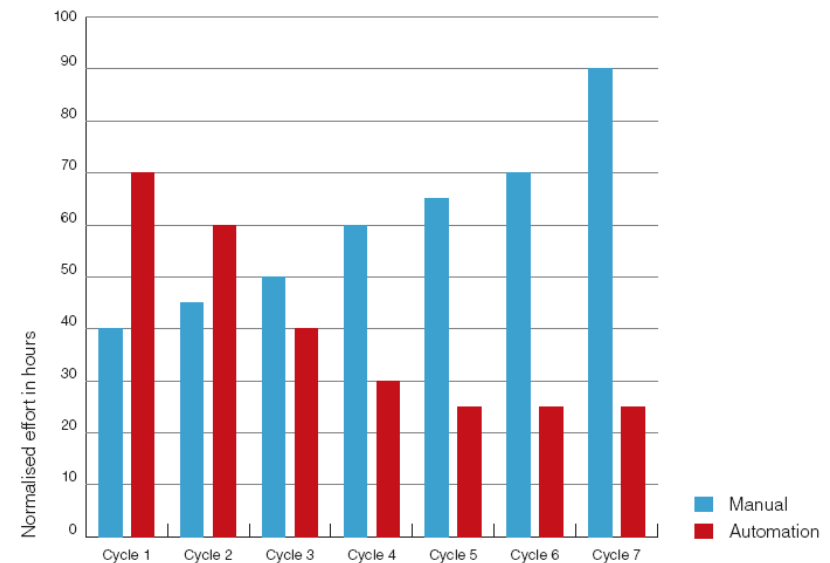


Figure 4: Regression testing effort – manual testing vs. automated testing

QUESTIONS



Thank You

© Copyright IBM Corporation 2009. All rights reserved. The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way. IBM, the IBM logo, Rational, the Rational logo, Telelogic, the Telelogic logo, and other IBM products and services are trademarks of the International Business Machines Corporation, in the United States, other countries or both. Other company, product, or service names may be trademarks or service marks of others.