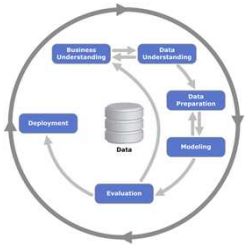
Requirements & Testing

(Four Quadrants)

Richard Frederick, PMP (rfrederick.pmp@gmail.com; 214-755-7035 talk or text; www.linkedin.com/in/rfrederick)



CBDA CERTIFIED IIBA

ISO International Organization for Standardization

ANSI American National Standards Institute

sas

OMG OBJECT MANAGEMENT GROUP

INCOSE International Council on Systems Engineering

IEEE

PMI Project Management Institute

Scaled Agile Framework

IIBA International Institute of Business Analysis

UNIFIED MODELING LANGUAGE

W3COMPUTING A DEVELOPER REFERENCE WEBSITE

Requirements Development Management (RDM) for

Business Data Analytics

(The Language of Data)

Richard Frederick, PMP (rfrederick.pmp@gmail.com; 214-755-7035 talk or text; www.linkedin.com/in/rfrederick)

Requirements Agenda

Business Data Analytics "The Language of Data"

01-FOUNDATION

- **LAB-Introductions (Name, Job Title, Objectives)**
- Dashboards and Robots (Data Mining & Machine Learning)
- Requirements and Testing (Four Quadrants)
- What are Structured Language Requirements? (Structured English and Structured Query Language)
- Why Should You Care? (Primary Source of Project Problems)
- How Do They Work? (Discreet Intellectual Property Inventory)
- Types of Requirements (Product, Project, DATA)
- Natural Language Processing (Morphology, Semantics, Syntax and Linguistics)
- OMG-SBVR (Semantics of Business Vocabulary & Rules)
- IEEE-EARS (Easy Approach to Requirements Syntax)
- INCOSE (Rules for Writing Requirements) & QVscribe
- Waterfall and Agile (Assembly Methods)

02-ELICIT

- **LAB-Vision/Scope (Seek to Understand)**

Elicitation Techniques:

- Document Analysis (Low Hanging Fruit)
- Interface Analysis (Navigation & Functionality)
- Benchmarking (Actual Data)
- Brainstorming (Every Idea is a Good Idea until it becomes a Bad Idea)
- Prototyping (Minimum Viable Product)
- Reverse Engineering (Begin with the End in Mind)
- Interview (Thinking Questions)
- Workshop (Group Interviews)
- Observation (What do you See?)
- Survey Questionnaire (Paper equals proof)

03-ANALYZE

- What are Models? (Pictures of Language)
- **LAB-The Language of Modeling (GIVEN pre WHEN process THEN output-result)**
- Types of Models (Context-Structure, Usage, Data Behavior, Process Flow)
- Context-Structure (Vision, Roadmap, Scope WBS)
- Usage (EPIC, UseCase, UserStory, Feature)
- Data Behavior (ERD, JOIN-Denormalization, Star Schema, Dimensional OLAP, Dashboard, Intelligence)
- Data Behavior (Data Dictionary, DataFlow, Data Structure Instance, Data Element Attribute, Data Store)
- Data Behavior (Process Logic, Business Rules)
- Process Flow (Swimlane)

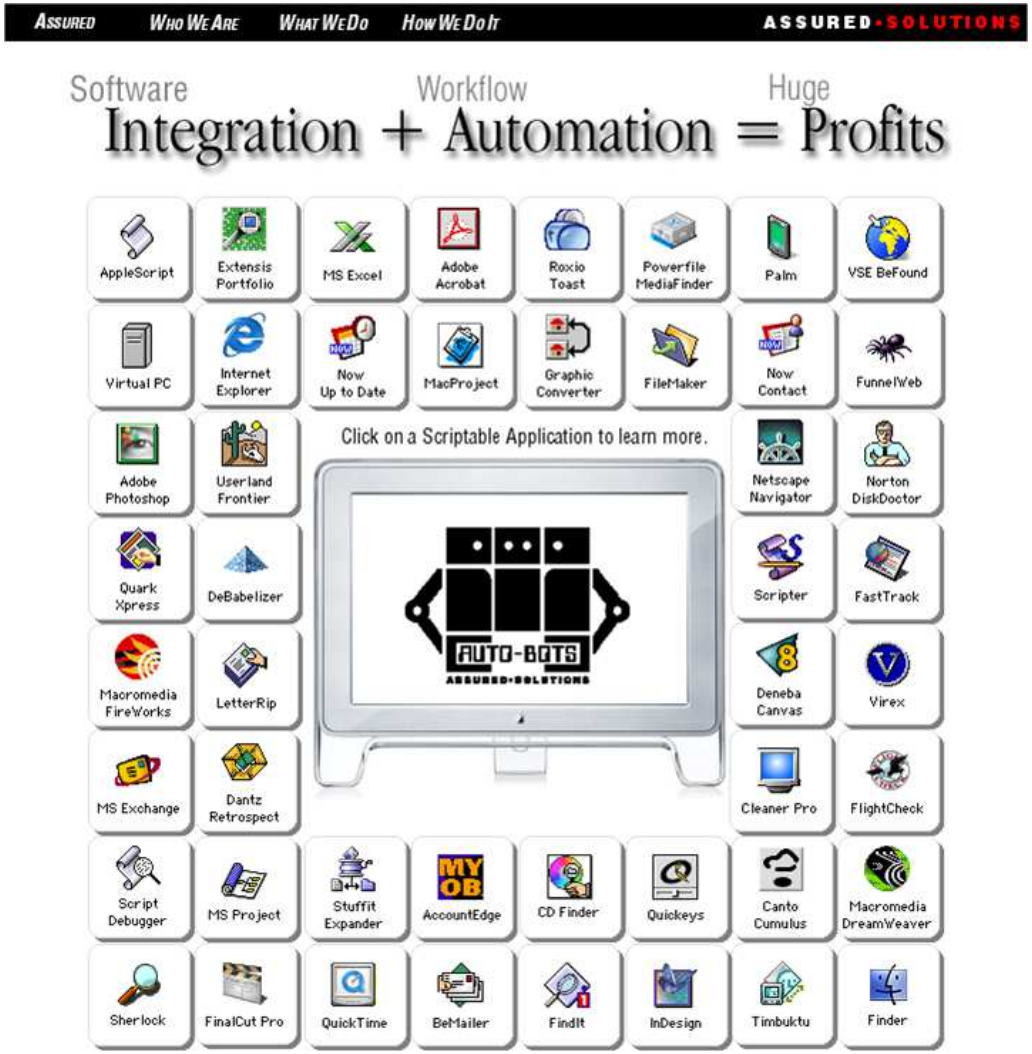
04-DOCUMENT

- Categorization, Organization, Documentation, Integration, Automation
- Making Documents Easy to Read (Fonts & Navigation)
- Document Types (BRD, TRD)
- **LAB-Business Requirement Document (Concept of Operation)**
- Technical Requirement Document (System Specification)

05-VALIDATE

- Validation thru Triangulation (Prep Drills)
- Traceability (Project Unique Identifier)
- Requirements Baseline (ROM Estimate, Planning Estimate, Definitive Estimate)
- **LAB-Estimating Story Points (Complexity and Risk)**
- Lessons Learned (Course Wrap-Up)

Assured Solutions



Requirements & Testing

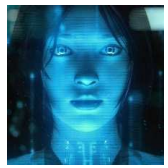
(Four Quadrants)

Overview "Seek to Understand"

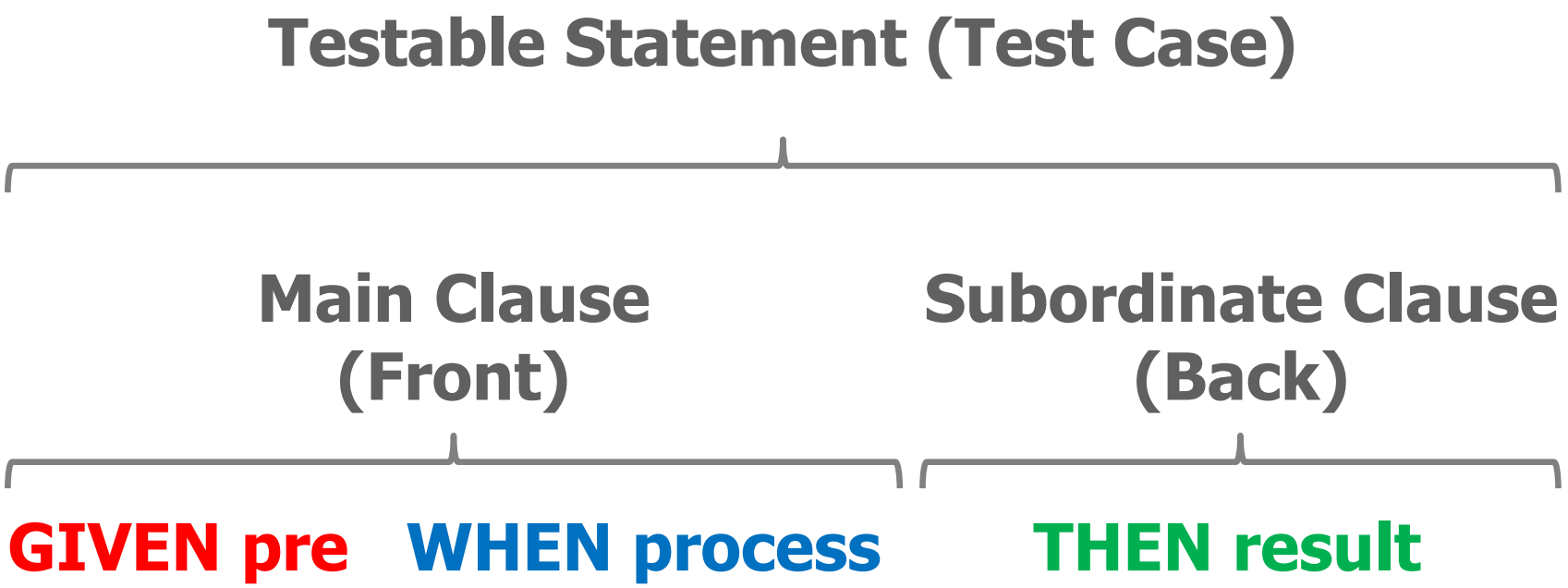
- Product and Process-oriented requirements
- Acceptance Criteria Test Cases
- Product and Process testing methods





A Requirement is a Testable Statement



Testable Statement

Request: [CAPABILITY]

- We need our customers to know that we care about them.

As Measured By (AMB): [CONSTRAINT]

- 95% of customer inquiries must be successfully resolved in the first contact by the end of Q4 this year.

"You Cannot Test a Request"

Writing System Requirements as TSS…

- THEN The System Shall… (TSS)
- THEN The System Shall Allow the User to…

- THEN The System Shall **[CAPABILITY] "as measured by" [CONSTRAINT]**
- THEN The System Shall **[CAPABILITY]** operating within **[CONDITION] "as measured by" [CONSTRAINT]**

Writing Business Requirements as User Stories…

- AS A [Actor/Job Title/Stakeholder]
- I NEED **[CAPABILITY]**
- SO THAT [BENEFIT]
- *As Measured By* **[CONSTRAINT]**

Writing Business Requirements as User Stories

As Measured By [CONSTRAINT] *aka* ***Acceptance Criteria:***

- **GIVEN** (Precondition, Input),
  - **WHEN** (Trigger, Process, Event) occurs,
    - **THEN** (Planned Output-Result)
      - » **TEST (ACTUAL Output)**
        - » **No Variance (Test Passes)**

User Story

Banking Application

User Story: Transfer Money Between Accounts

As a Bank Customer

I need to transfer money between my accounts

So that I have money where I need it to be

User Story

Banking Application

User Story: Transfer Money Between Accounts

As a Bank Customer

I need to transfer money between my accounts

So that I have money where I need it to be

Acceptance Criteria:

Given/When/Then

Given (a particular scenario- "Pre-condition")

When (the user does something- "Input")

Then (this should be the result- "Planned Output")

TEST (ACTUAL Output)

Scenario 1 (Sufficient funds):

Given Account A has $125 available
And Account B has $375 available
When I transfer $200 from B to A
Then I should see **Confirmation Message**
And A should have $325 available
And B should have $175 available

TEST (ACTUAL Output)

See Confirmation Message (Pass)

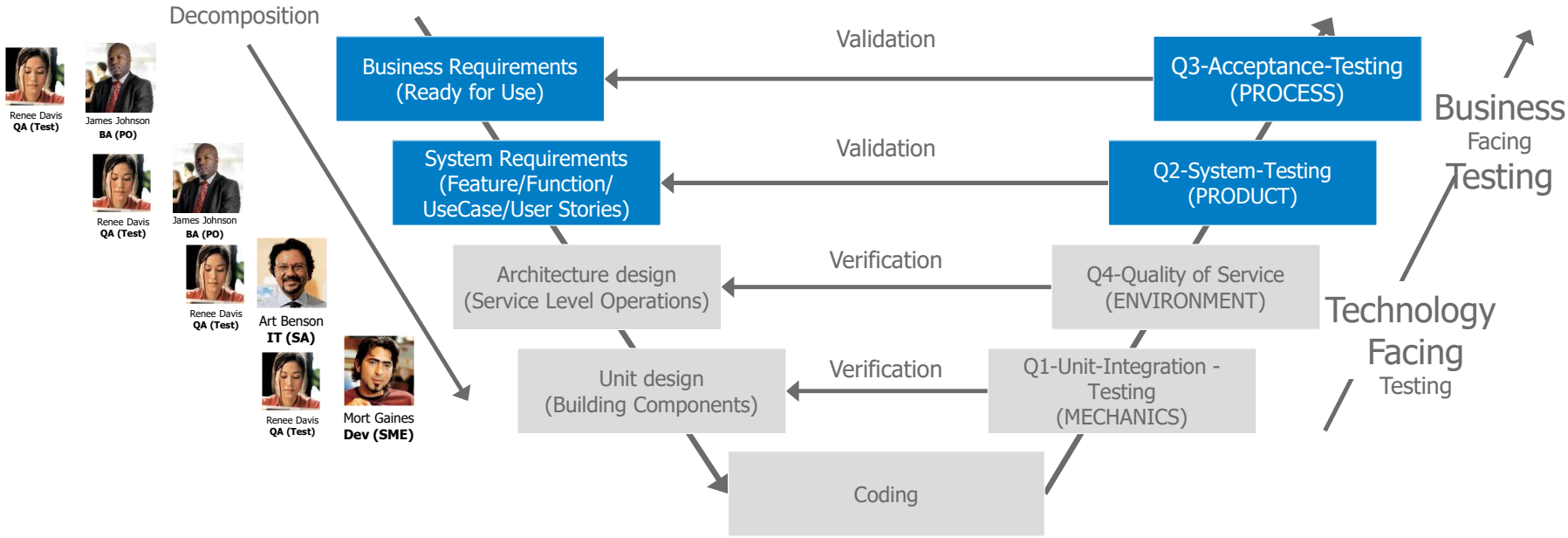
Scenario 2 (Insufficient funds):

Given Account A has $125 available
And Account B has $375 available
When I transfer $200 from A to B
Then I should see an **Error Message**
And A should have $125 available
And B should have $375 available

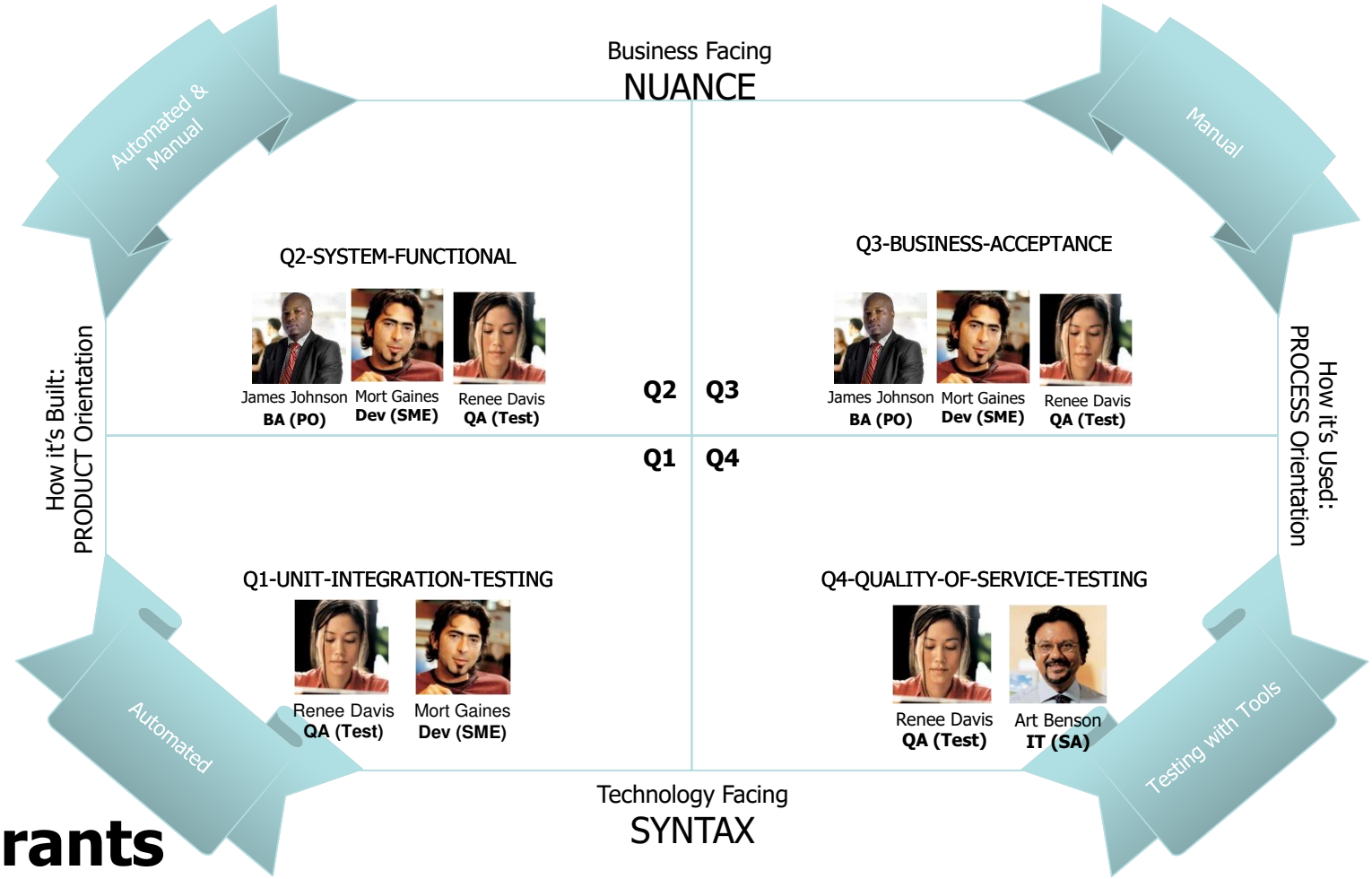
TEST (ACTUAL Output)

See Confirmation Message (Fail)

Requirements and Testing

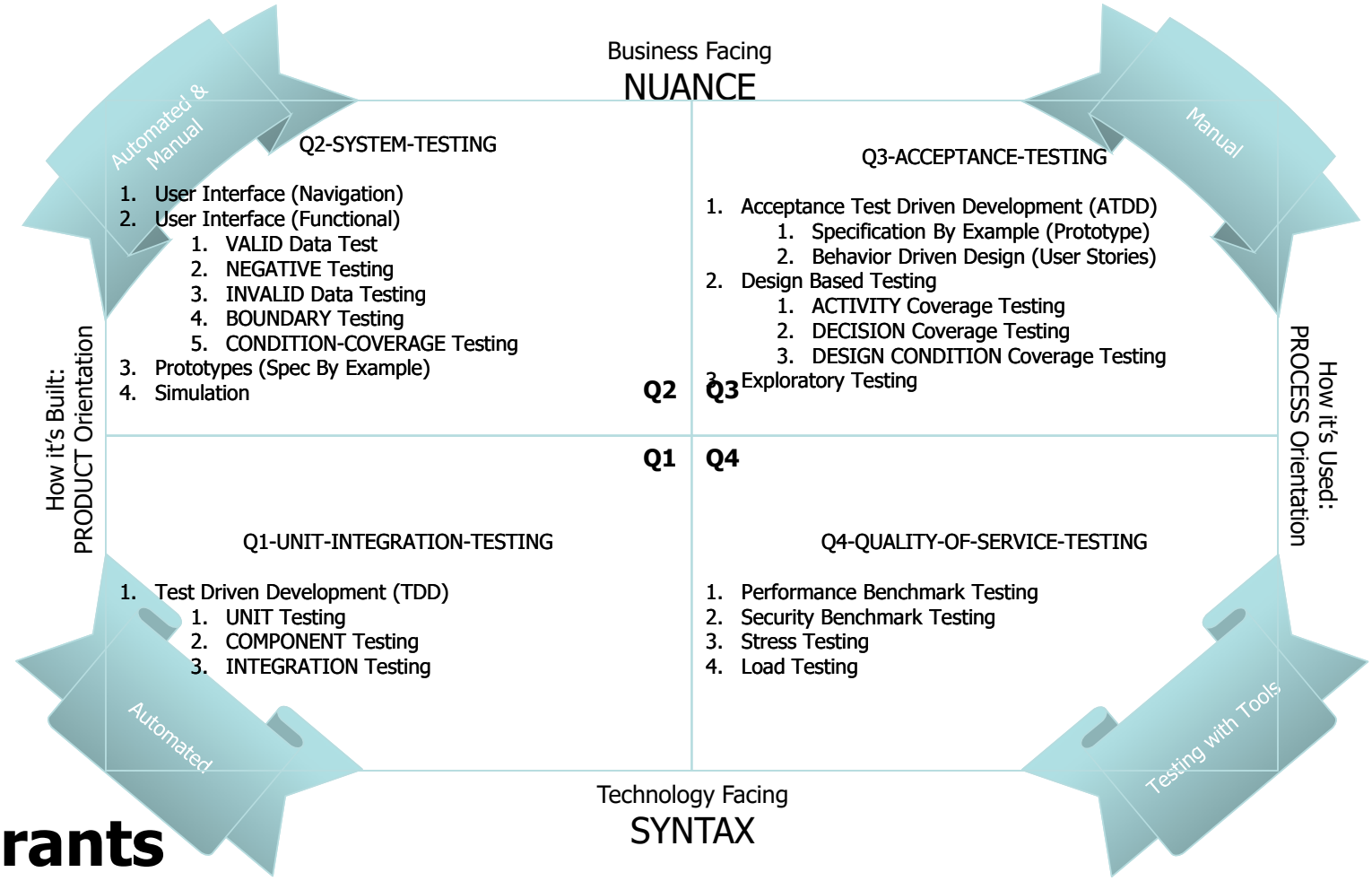


Four Quadrants of Testing



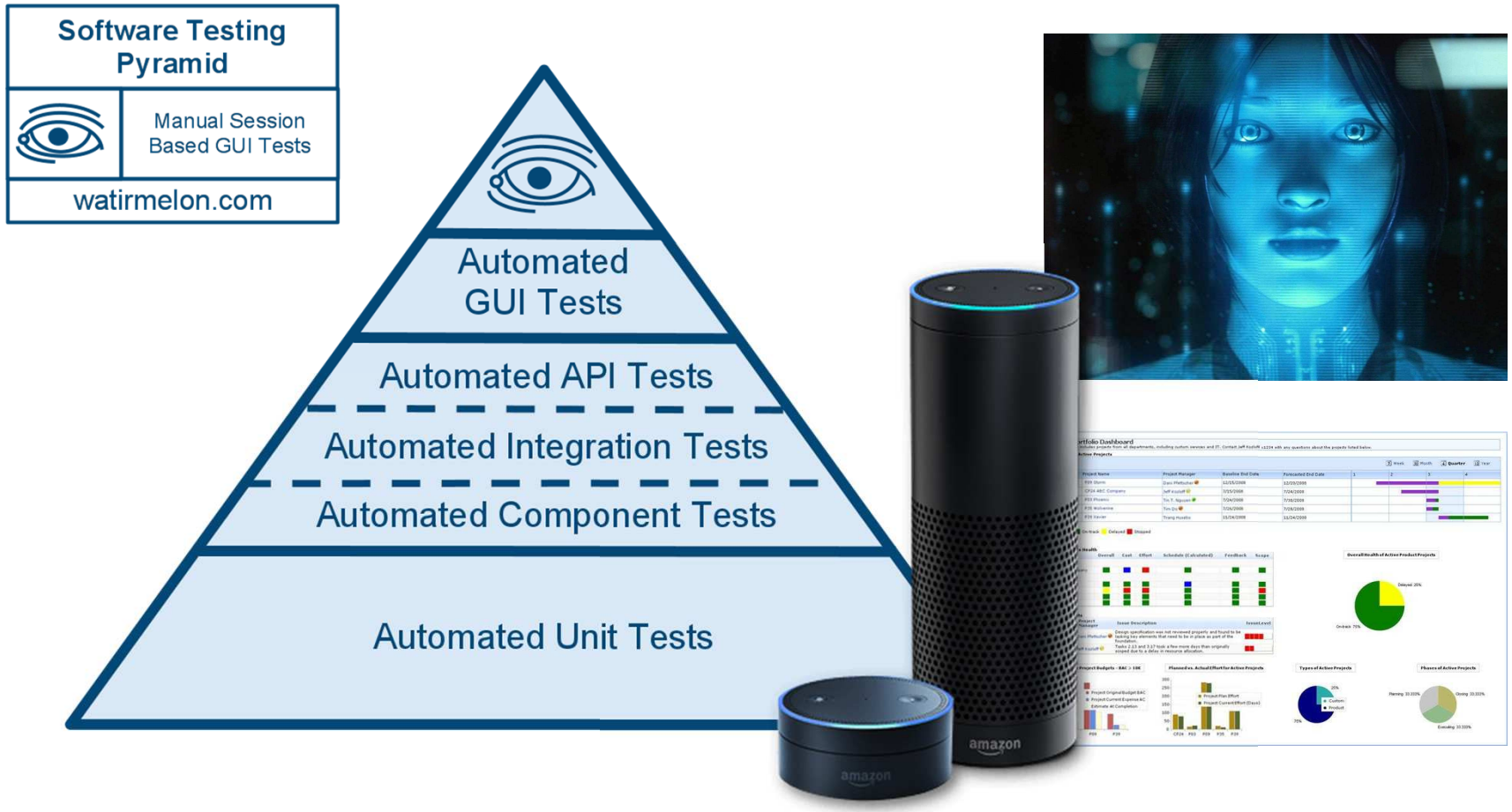
4 Quadrants Of Testing

Four Quadrants of Testing



4 Quadrants Of Testing

TEST AUTOMATION > Automation Pyramid

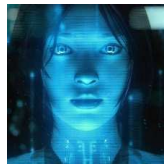


Universal Testing Method

1. Model the test space	Clearly define the system that will be tested & its context
2. Determine coverage	Decide what needs to be tested
3. Determine oracles	Identify the sources of truth for pass/fail judgments
4. Determine test procedures	Plan the details of how the testing will be performed
5. Configure the test system	Set everything up so you can begin testing
6. Operate the test system	Run your tests
7. Observe the test system	Keep track of the results of your tests
8. Evaluate the test results	Determine if each test passed or failed
9. Report test results	Communicate testing status and final results





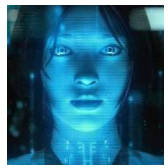



In Summary "Train the Trainer"

- Q1-Unit Integration (Product Syntax)
- Q2-System Testing (Product Nuance)
- Q3-Acceptance Testing (Process Nuance)
- Q4-Quality of Service Testing (Process Syntax)





Let’s stay in contact:

Richard Frederick, PMP

214-755-7035 (text or talk)

Rfrederick.pmp@gmail.com

www.linkedin.com/in/rfrederick

https://meetings.hubspot.com/rfrederick-pmp