



+

+

- What Does “Specification” Mean?
  - \* Features and Characteristics of Application Domain
  - \* Readable Document for Varied “Users”
  - \* Contract - Customer and Company
  - \* Contract - Designers and Engineers
  - \* **Living and Long-Lived!!!**
- Overall, we Identify the Purpose, Goals, Scope, and Content of a Given Problem Domain
- Assume:
  - \* Feasibility Study
  - \* Market Projections/Potential

+

2

+

+

## **Individuals and Specifications**

- Spec. Serves as a Communications Medium
- Different Things to Different People at Different Times
  - \* Understanding an Application
  - \* Investigating its Details
  - \* Transitioning to Design and Implementation
  - \* Corrections and Evolution
- Customers and End-Users

+

3

+

+

## Customers and End-Users

- Customer as a Generic Term
  - \* Actual Individual or Company
  - \* Targeted “Group” of Users
- Spec. Must Meet Customer’s Needs!
- Spec. Document has “Non-Technical” Features
- End-Users Have Diverse Expertise
- Auto. Generation of Documentation from Spec.
- Other Non-Technical Individuals?
- Experts in Particular Domain?

+

4

+

+

## **Role of Software Engineers**

- Charged with Engineering Information w.r.t:
  - \* Collection
  - \* Deciphering
  - \* Interpretation
  - \* Comprehension
  - \* Utilization
  - \* Creation
  - \* Dissemination
  
- SWEs Interact with Technical Personnel and Non-Experts
  
- Attempt to Categorize Their Responsibilities

+

5

+

+

## **What's in a Specification?**

- All 'Requirements' for an Application
- Meets 'Needs' of Different Readers
- Broad Range of Considerations and Aspects
- Formal Specs. - Support Rigor, But Break Down When Applied to Large Appls.
- Adhere to Following Two Principles:
  - \* Anticipation of Change
  - \* Separations of Concerns

+

8

+

+

## **Section/Question-Based Approach**

- Two Major Portions of a Spec:
  - \* Application Issues and Reqr.
  - \* Developmental/Implementation Issues
- Both Portions Contain Multiple Sections that Address Various and Particular Appl. Features
- Question-Based Approach that Motivates Content of Individual Sections
- Employ HTSS to Illustrate Techniques
- Different Portions of Spec. Address Different Needs of IEs and Other Individuals

+

+

+

## Application Issues and Requirements

- Major Sections to a Specification
  - \* **Introduction:** Appl. Overview
  - \* **Glossary:** Important Terms
  - \* **Operating Environ.:** Targeted Run-Time Environment
  - \* **Interfaces:** Users, Systems, and DBs
  - \* **Information:** What and How?
  - \* **Performance:** Operating Environ. Constraints
  - \* **Security:** Who Uses What When?
- Incremental, Iterative, Cyclical Process
- Ordering is Changeable!

+



+

+

## High-Tech Supermarket System (HTSS)

- Utilize State-of-Art Technology to Computerize Inventory Control and Facilitate Shopping Experiences
- Major Functionalities Include:
  1. Inventory Control for Ordering/Stocking Shelves
  2. Cashier Actions to Check Out Customers and Update Inventory
  3. User-Friendly Grocery Item Locator for Customers
  4. Fast-Track Deli-Orderer for Shoppers and Deli-Workers
- Other Capabilities Include:
  1. Automatic Notification to Reorder Stock
  2. Query Capabilities for Managers
  3. Integration with Other Stores and Warehouse
  4. Interaction with ATM, Bank, and Credit DBs

+

+

+

## **The High-Tech Supermarket Application**

- Various User-System Interfaces:
  - \* Cash Register/UPC Scanner
  - \* Displays for Inventory Querying
  - \* Shopper Interface for Locator
  - \* Shopper Interface for Orderer
  - \* Deli Interface for Orderer
- All Have Unique Requirements
- Response Time and Throughput Critical
- User-Friendliness Important

+

+

+

## The Introduction Section

- A Problem Statement for the Application
- Focus on its Scope, Objectives, Goals
- Highlight the Target Environment
- Investigate the Questions:
  1. What is the domain for the application?
  2. Who are intended users/companies?
  3. What is its main purpose?
  4. What is its scope/range?
  5. What are its major functionalities?
  6. What is the target environment?
  7. What are critical user/system interactions?
  8. How does user interaction occur?
  9. Unique hardware/software reqr.?

+

+

+

## Introduction for HTSS

A supermarket chain is interested in using the newest and most up-to-date computing technology to support inventory control and to assist customers in their shopping experiences. The chain wants to integrate inventory control with

1. the cashiers functions to automatically update inventory when an item is sold
2. a user-friendly grocery item locator that indicates textually and graphically where items are in the store and if the item is out of stock
3. a fast-track deli-orderer (deli orders are entered electronically, with the shoppers allowed to pick up the order weighed and packaged without waiting).

The inventory control aspect of the proposed system would maintain all inventory for the store and alert the appropriate store personnel whenever the amount of an item drops to its reorder limit. In addition to the aforementioned functional characteristics, the system should also have extensive query capabilities that allow store personnel to investigate the status of the inventory and sales for the store over varying time periods and other restrictions.

Example Represents a Good First Attempt!

+

+

+

## The Glossary Section

- Identify and Collect Important Terms
- Reduce Both Confusion and Inconsistency
- A Term May Have One or More Meanings
- Investigate the Questions:
  1. What are application's relevant terms?
  2. Are there any special terms?
  3. Are there any ambiguous terms?
  4. For each "term", what is its scope, importance, functionality, and operation?
  5. Which terms have a restricted meaning?
  6. Any terms for different types of IEs?
  7. Any terms for Customers? End Users?
  8. Any terms for Domain Experts?

+

+

+

## Glossary for HTSS

**Item:** Refers to a product that is sold at a supermarket, e.g., canned goods, cereals, produce, etc.

**UPC:** The Universal Product Code for all supermarket items.

**ICDB:** The Inventory Control Database, functions as a central repository that tracks the features and characteristics of all items on shelves and in the stockroom.

**I-Controller:** An I(nventory)-Controller is the individual at the supermarket who is in charge of maintaining the inventory.

**Courtesy Card:** A Check Cashing Card that has been assigned to a shopper.

**PIN:** A Personal Identification Number to pay for groceries with a ATM/Debit Card.

**Etc...**

Glossary Constantly Changes as Spec. Evolves!

+

+

+

## The Operating Envr. Section

- Characterize Targeted Run-Time Environment
- Elaborate on Off-the-Shelf/Specialized HW/SW
- Utilize Prose or Itemized List or Both
- Investigate the Questions:
  1. Hardware, software, and operating environment?
  2. Off-the-shelf software systems/tools?
  3. What is the physical environment (e.g., location, temperature, etc.)?
  4. How is a hostile environment handled (e.g., Hardware/Software Threats)?
  5. Expertise of end-users?
  6. Expected usage pattern?
  7. Avg. number of users?
  8. High and Low load times?

+

+

+

## **Operating Environment for HTSS**

The operating environment for the HTSS application will involve a collection of networked terminals. There are five different kinds of terminals: cashier (to total orders), deli-orderer (for shopper), deli-display (for deli workers to process orders), item-locator (for shopper), and inventory control. The two deli displays and the locator must have sealed data entry keys to insure that spills, dirt, etc., will not inhibit operation. The cashier display needs two interfaces to track the items and their costs for both the cashier and shopper. Cashiers and inventory workers will be trained to use their displays, and the input will use either keyboard or mouse technology. To allow deli workers to keep their hands free, a voice controlled interface must be developed to allow orders to be processed. Finally, since shoppers are “naive” users, the interface for the deli-orderer and item-locator must be geared towards their skill levels. The expected throughput for each cashier at peak levels is 10 customers per hour with a maximum of 20 cashiers. There are two deli-orderers and five item-locators that must ...etc...

Indicates High Interaction and Likely Complexity!

Equipment Specs. OK in Subsequent Iterations!

+



+

+

## The Interfaces Section

- Usage of Appl. w.r.t. Graphical, Database, and User-System Interactions
- Gives Another View of Application's Scope
- Investigate the Questions:
  1. What are the major interfaces?
  2. Goal and purpose of each interface?
  3. Functions/responsibilities of each interface?
  4. Makeup of each user interface?
  5. Expertise of each interface user?
  6. Available information for each interface?
  7. Specific information for each interface?
  8. Required manipulation of information?
  9. Dependencies/interactions among interfaces?
  10. Restrictions on each interface?
  11. Information application requirements?
- DFDs, ERs, Screen Mockups, etc. are Apropos

+

+

+

## Interfaces for HTSS

Collectively, the interfaces for the cash register, scanner, deli-orderer, item-locator, deli-display, and inventory control, require the following database interfaces:

**ICDB:** A database of all items is maintained, with each interface requiring a different level of access. All of the user/system interfaces require read access to accomplish their respective tasks, e.g., the scanner must verify the UPC code of an item with its database entry. In addition, the inventory control interface will need to modify the database when items are reordered, while the cash register interface will issue a command to decrement stock on the shelves when an item is sold. This decrement operation is important, since it impacts on consistency and ensures that the item-locator has up-to-date information.

**Order DB:** A database is required to track individual customer orders. The interface to this database is required by the scanner and cash register interfaces in ringing up and totaling an order. Write access to create an entry in the Order DB is necessary.

**Deli-Order DB:** A short-term persistent database is required to hold orders entered via the deli-orderer for processing by the deli-display. The interface to this database requires both read and write access by the two aforementioned user-system interfaces.

Glossary Constantly Changes as Spec. Evolves!

+

+

+

## Interfaces for HTSS

- Mock-Ups for User Interfaces Needed
- Deli-Orderer:

```

+-----+
|Select Item: Ham      Quantity: 1lb |
|Select Item: Swiss   Quantity: 2lbs|
|      ...            |
|End Session (Y/N): Y |
|Order Id: 1234      |
|                    |
|  *** Pick up in 20 minutes ***    |
|                    |
+-----+

```

- Item-Locator:

```

+-----+
| Select Item: Napkins |
| Location:           |
|   Aisle 5          |
|   -----          |
|   -----          |
|   --Shelf 2-----N----- |
|   -----          |
|                    |
+-----+

```

+

+

+

## The Interface Operation Section

- Focus on an Interface Perspective
- Identify Relevant Interface Structure and Interactions
- DFDs, FSMs, PNs, etc. are Apropos
- Investigate the Questions:
  1. Attainment of application behavior via Interfaces?
  2. General operation of each interface?
  3. Major interactions between interfaces?
  4. Interfaces and database interactions?
  5. Information Storage/Retrieval Issues?
  6. Two Views for Uer Interface
    - \* w.r.t. Customers/End Users
    - \* w.r.t. SWEs, IEs, IDEs, ...
- ISEs, IManEs, IDEs, IDevEs, and IMEs

+

+

+

## Interface Operation for HTSS

- To Support Interfaces of HTSS
  - \* What are Functional Requirements?
  - \* ICDB, Order DB, DeliOrder DB Accessed?
- Identify Design Dependencies w.r.t.
  - \* Interface Shared Functionalities
  - \* Interface Information Access
- Goals:
  - \* Promote Communication Among IDEs, ISEs
  - \* Reuse of Designs

+

+

+

## The Information Section

- Identify All Informational Units (IUs)
- Employ ERs, ADTs, Natural Language, etc.
- Investigate the Questions:
  1. Major information units (IUs)?
  2. What is each information unit's:
    - \* purpose in the application?
    - \* lifetime (persistent or not)?
    - \* usage or intended usage?
    - \* availability to components/interfaces?
    - \* accessibility by different end-users?
    - \* general semantic structure?
    - \* interactions(dependencies) with other IUs?
    - \* independence w.r.t. other IUs?
    - \* structural relationship with other IUs?
    - \* storage/access in the database?

+

+

+

## The Information Section

- ISEs focus on Information Units (IUs) as
  - \* information source
  - \* information resource
  - \* information producer
- Bridge Between ISEs and IDEs
- DFDs (producer view) and ERs (source/resource view) Overlap with Other Phases
- Introduce *Profile* Concept for IU:
  - \* Profile Contains Pertinent Characteristics
  - \* Profile Represents Designer Supplied and Inferred Associations
  - \* Profile Captures Purpose w.r.t. Application, its Components, and its Interfaces

+

+

+

## Information for HTSS

- Recall ER Diagram for HTSS
- Expand to Include Additional Details
- Develop Profiles for IUs, e.g., Item:
  - \* Description: Item contains the common features on groceries that are used by all aspects of HTSS.
  - \* Characteristics: Maintain UPC Code, Name, Costs, etc.
  - \* Associations: Interacts with specialized Items, Sales and Order relationships
- UPC Code is Non-Changeable After Item Creation

+



+

+

## The Performance Section

- Related to Operating Environment
- High-Level View to Bridge Gap to Performance Constraints for Developing Application
- Investigate the Questions:
  1. Application response-time requirements?
  2. User types?
  3. Maximum (minimum) number of users?
  4. Expected usage pattern?
  5. Error tolerance(video game vs. flight controller)?
  6. Expected workload pattern?
  7. Involved resources?
- Like Database, Perf. Requirements are Norm, Rather than Exception!

+

+

+

## Performance for HTSS

- Cast Operating Environment Restrictions into a New Light
- Store-Locator Display May Indicate that Shelf Has Items, When in Fact, Its Empty
- Is this a Tolerable Inconsistency?
- Any Time Critical Portions of HTSS?
  - \* Scanner/Cashier to ICDB Requires Reliability and Throughput  
1 sec. real time for item's price
  - \* Locator and Orderer Must Work in a Timely Fashion
  - \* I-Controller Cannot Lock Out Other Users for Any Time Duration Except During Repricing
  - \* Other Life/Death Issues?

+

+

+

## The Performance/Interfaces Section

- Examine w.r.t. User/System Interfaces
- Investigate the Questions:
  1. Expected response time?
  2. Expected refresh time?
  3. Failure conditions?
- In HTSS:
  - \* Scanner - UPC Code - ICDB - Order: Quick!
  - \* Deli-Orderer/Item-Locator: Timely!
  - \* Impact of Failure of Deli-Display?
  - \* Impact of Failure of Inventory Controller?
  - \* Replicated ICDB at Register?  
What's Impact w.r.t. Consistency?

+

+

+

## The Performance/SysOp Section

- Relevant Time and Space Limitations
- Information/Interfaces Impacted
- Investigate the Questions:
  1. System components that impact response times?
  2. System components that do not?
  3. Communication between components and performance impact?
- In HTSS:
  - \* Scanner to Register Communication
  - \* Impacted by Information Access (Read of ICDB, Write of Order DB)
  - \* Average R/W Times of Information Units?

+

+

+

## The Performance/Database Section

- Distribution vs. Sharing
- Volume vs. Throughput
- Investigate the Questions:
  1. Expected database size?
  2. Expected response time/throughput?
  3. Access patterns for IUs by Interfaces?
  4. Information Consistency Maintenance?
- In HTSS:
  - \* ICDB is a Shared Repository
  - \* Order DB Resides at Register/Scanner and is Periodically Updated to Shared Repository
  - \* ICDB Unavailable During Weekly Price Updates
  - \* ICDB Used by All Interfaces (30 Max!)
  - \* Remote DB Access for Debit/Credit Cards
  - \* Real Time Credit Checks

+

+

+

## The Security Section

- What are Critical Security Issues?
- Who Can See What When?
- MAC (Mandatory Access Control)?
- DAC (Discretionary Access Control)?
- Investigate the Questions:
  1. What is the security plan?
  2. Users (user roles) for the application?
  3. Responsibilities for each user role?
  4. IUs for each user role access?
  5. Prohibited IUs for each user role?
  6. Prohibited Interfaces for each user role?
- A *User-Role Profile* Contains:
  - \* Prose description of responsibilities
  - \* Associated security requirements

+

+

+

## Security for HTSS

### Application Security

In HTSS, the main security policy is to encapsulate the majority of functions that write to a database into the design of the different interfaces and system components. While the scanner and cash-register must write the Order DB, this write action should not be the result of an explicit command entered by the cashier. Rather, it should be a responsibility of the software that implements the two interfaces. The item-locator must be limited to read access of the ICDB, with the deli-orderer and deli-display allowed to modify the database. In the case of the deli-orderer, the tool controls the creation of database entries that represent customer orders. For the deli-display, the deli worker must physically issue a command that deletes the order from Deli-Order DB once it has been filled. Finally, the inventory controller is given the most access, since I-Controllers must have the ability to create, modify, and delete Items from the ICDB.

### User Roles

**Cashier:** Responsible for ringing orders using the scanner and the cash register

**Shopper/Customer:** Buys groceries. Uses the deli-orderer and item-locator interfaces.

**Deli-Display:** Read/write access to Deli-Order DB to process and then remove orders from shoppers.

**Store Manager:** Has access to all aspects of the application and its interfaces, and can function as a super-user to override any restrictions.

**Purchase/Order:** Actions that support ordering of items.

+