

# Get Your Users "IN" the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

<b>The Quest for Excellent Requirements</b>	9 July 2009
<p style="text-align: center;"><b>Get Your Users "IN" the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements</b></p> <div data-bbox="365 541 617 861"></div> <p style="text-align: center;"><b>bringing your business into focus®</b> <a href="http://www.requirementsquest.com">www.requirementsquest.com</a></p>	
<p>Presented by: Roxanne E. Miller, CBAP <a href="http://www.RequirementsQuest.com">www.RequirementsQuest.com</a></p>	


<b>Session Learning Objectives</b>	9 July 2009	
		
<p>Define nonfunctional requirements</p>	<p>Understand these vital, yet overlooked requirements</p>	<p>Classify common nonfunctional requirements</p>
 bringing your business into focus®		<b>Understanding Nonfunctional Requirements - 2</b>

# Get Your Users "IN" the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Nonfunctional Definition

#1




**Learning Objective #1:**  
**Define nonfunctional requirements**



bringing your business into focus®

Understanding Nonfunctional Requirements - 3

## Levels of Requirements

Levels



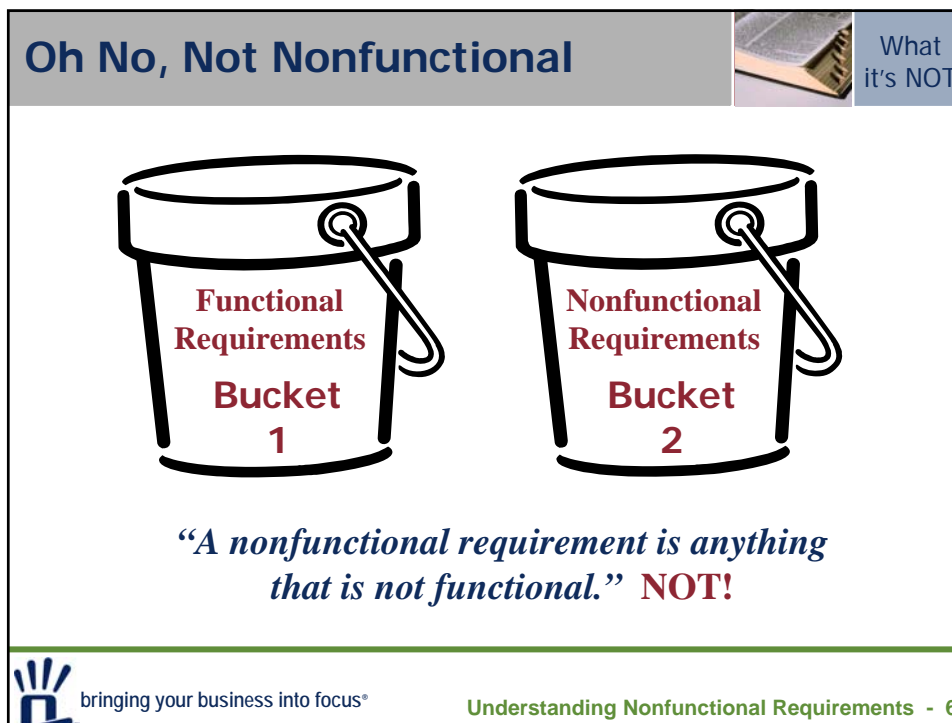
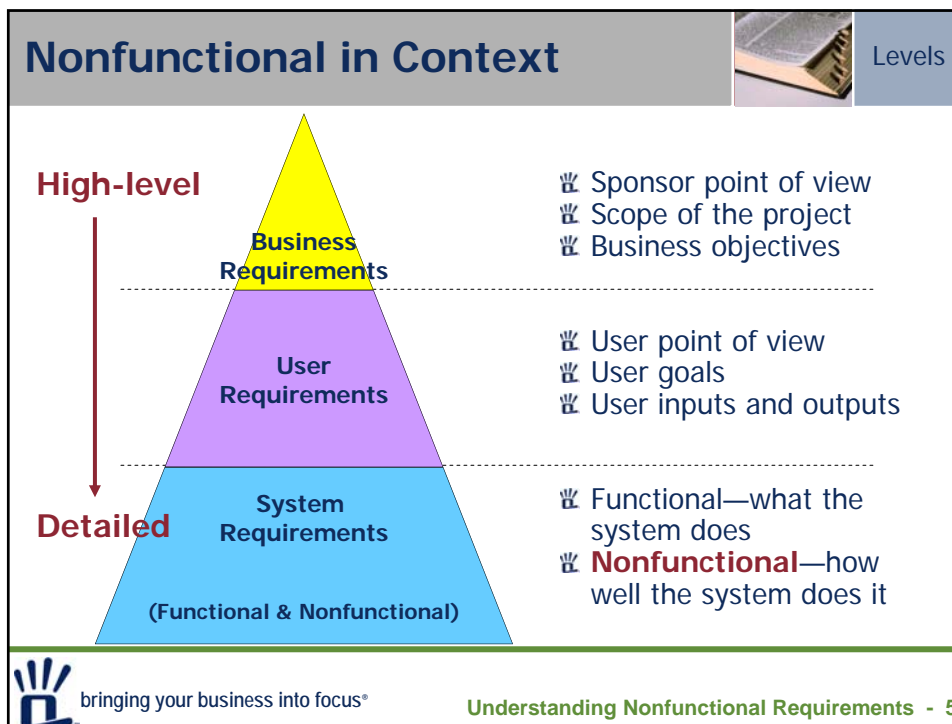
<b>Business Level</b>	Defines project scope. Identifies business benefits. Establishes high-level objectives.
	
<b>User Level</b>	Focuses on business processes. Names user roles and user goals. Identifies flow of information and materials.
	
<b>System Level</b>	Identifies processing of inputs and outputs. Defines functions and features. Describes quality attributes of the system environment.

bringing your business into focus®

Understanding Nonfunctional Requirements - 4

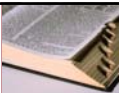
# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements


Minneapolis IIBA Chapter 9 July 2009

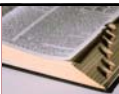







# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

A Handful of Definitions	 Example
<p><i>“Nonfunctional requirement</i> – in software system engineering, a software requirement that describes not what the software will do, but how the software will do it, for example, software performance requirements, software external interface requirements, software design constraints, and software quality attributes. Nonfunctional requirements are difficult to test; therefore, they are usually evaluated subjectively.” [Thayer2000]</p>	
<p><i>“Non-functional requirements</i> – These are constraints on the services or functions offered by the system. They include timing constraints, constraints on the development process and standards. Non-functional requirements often apply to the system as a whole. They do not usually just apply to individual system features or services.” [Sommerville2007]</p>	
<p><i>“Non-functional requirement</i> – A property, or quality, that the product must have, such as an appearance, or a speed or accuracy property.” [Robertson2003]</p>	
<p><i>“Nonfunctional requirement</i> – A description of a property or characteristic that a software system must exhibit or a constraint that it must respect, other than an observable system behavior.” [Wieggers2003]</p>	
<p><i>“Nonbehavioral requirements</i> – Requirements that describe the required overall attributes of the system, including portability, reliability, efficiency, human engineering, testability, understandability, and modifiability.” [Davis1993]</p>	


 bringing your business into focus® Understanding Nonfunctional Requirements - 7

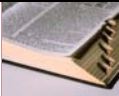
What Does It Really Mean?	 What's left?
<p><b>Verbiage Removed:</b></p>	<p>...how the <b>software</b> will do it.</p>
<p> General wordiness.</p>	<p>...<b>constraints</b> on the services or functions offered by the <b>system</b>.</p>
<p> Nonfunctional “examples.”</p>	<p>...<b>property</b>, or <b>quality</b>, that the <b>product</b> must have.</p>
<p> “Descriptive” information that doesn't add value.</p>	<p>...<b>property</b> that a <b>software system</b> must exhibit or a <b>constraint</b> that it must respect.</p>
<p> Inclusion or reference to “functional requirements.”</p>	<p>...required overall <b>attributes</b> of the <b>system</b>.</p>

 bringing your business into focus® Understanding Nonfunctional Requirements - 8



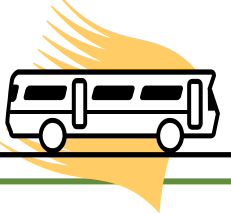
**A Simplified Definition**  How Well


**Nonfunctional Requirement**  
—  
**a specification of *how well*  
a software *system*  
*must function.***

 bringing your business into focus® **Understanding Nonfunctional Requirements - 9**

**Functional versus Nonfunctional**  Differ

F Product features.	N Product properties.
F Describe the work that is done.	N Describe the character of the work.
F Describe the actions with which the work is concerned.	N Describe the experience of the user while doing the work.
F Characterized by verbs (active).	N Characterized by adjectives.

 bringing your business into focus® **Understanding Nonfunctional Requirements - 10**

**Get Your Users “IN” the Doghouse and Keep Yourself Out:  
Eliciting Nonfunctional Requirements**  
Minneapolis IIBA Chapter 9 July 2009

**Vital, Yet Why Overlooked?** #2

**Learning Objective #2:**  
Understand why nonfunctional requirements are vital, yet overlooked



bringing your business into focus® Understanding Nonfunctional Requirements - 11

**Importance of Nonfunctional** Need Both

**All functional requirements may be satisfied, but if nonfunctional requirements are overlooked, the system will fail.**



bringing your business into focus® Understanding Nonfunctional Requirements - 12

# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Nonfunctional Challenges



Don't Agree


- ☞ No agreed upon formal **definition**.
- ☞ No complete **list**.
- ☞ No single universal **classification scheme**, accommodating all the needs of different application domains under different situations.
- ☞ Different people and organizations use **different terminologies**.



bringing your business into focus®


Understanding Nonfunctional Requirements - 13

## Alternative Names & Spelling



Many Names

- ☞ Quality **Attributes**.
- ☞ **Quality** Requirements.
- ☞ **Nonbehavioral** Requirements.
- ☞ “ilities” or “ities.”
- ☞ Spelled with or without a hyphen.
  - ☞ Non-functional or **Nonfunctional**







bringing your business into focus®

Understanding Nonfunctional Requirements - 14

# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

<b>Nonfunctional Complexity</b>		Complex Nature
<ul style="list-style-type: none"><li><b>Subjective</b><ul style="list-style-type: none"><li>☞ Viewed, interpreted and evaluated differently by different people. Statements are often too brief and vague.</li></ul></li><li><b>Relative</b><ul style="list-style-type: none"><li>☞ Interpretation and importance may vary depending on the particular system and organizational needs.</li></ul></li><li><b>Integrated</b><ul style="list-style-type: none"><li>☞ Attempts to achieve one requirement can hurt or help the achievement of other requirements due to the nature of their global impact.</li></ul></li></ul>		
	bringing your business into focus®	Understanding Nonfunctional Requirements - 15

<b>Nonfunctional Classification</b>		#3
	<p><b>Learning Objective #3:</b></p> <p><b>Making sense of nonfunctional requirements with a user-focused classification</b></p>	
	bringing your business into focus®	Understanding Nonfunctional Requirements - 16



# Get Your Users "IN" the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Overwhelmability?




Scary List?

Access Control Accessibility Accountability Accuracy Adaptability Additivity Adjustability Affordability Agility Auditability Augmentability Autonomy Availability Buffer Space* Capability Capacity Clarity Code Space* Cohesiveness Commonality Compatibility Completeness Comprehensibility Conceptuality Conciseness Confidentiality Configurability Connectivity Consistency	Controllability Coordination Cost Coordination Time Correctness Coupling Customer Evaluation Customer Loyalty Customizability Data Space* Decomposability Degradation of Service Dependability Development Cost Development Time Distributivity Diversity Ease of use Efficiency Elasticity Enhanceability Evolvability Execution Cost Extensibility External Consistency Fault Tolerance Feasibility Flexibility Formality Generality	Human Engineering Independence Informativeness Integrity Internal Consistency Interoperability Intuitiveness Learnability Leveragability Main Memory* Maintainability Maintenance Cost Maintenance Time Maturity Mean Performance Measurability Mobility Modifiability Modularity Naturalness Nomadicity Observability Off-Peak Period* Operability Operating Cost Peak Period* Performability Planning Cost Planning Time	Portability Precision Predictability Process Mgmt. Time Productivity Project Stability Project Tracking Cost Promptness Quality Reconfigurability Recoverability Recovery Reliability Repeatability Replaceability Replicability Response Time Responsiveness Retirement Cost Reusability Risk Analysis Cost Risk Analysis Time Robustness Safety Scalability Security Sensitivity Similarity Simplicity	Space Boundedness Space Performance Specificity Stability Subjectivity Supportability Surety Survivability Susceptibility Sustainability Tankness Testability Throughput Timeliness Traceability Trainability Transferability Transparency Understandability Uniform* Uniformity Usability User-Friendliness Validity Variability Verifiability Versatility Visibility *Performance
---	--	--	--	---

bringing your business into focus® Understanding Nonfunctional Requirements - 17

## Quality Attributes Tree



1976

```

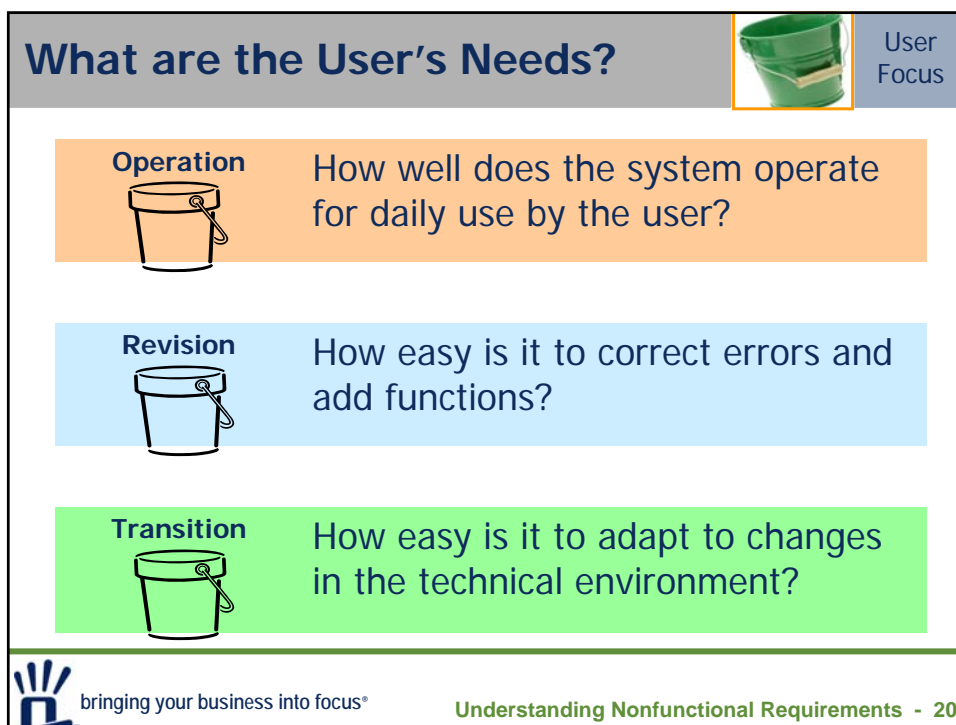
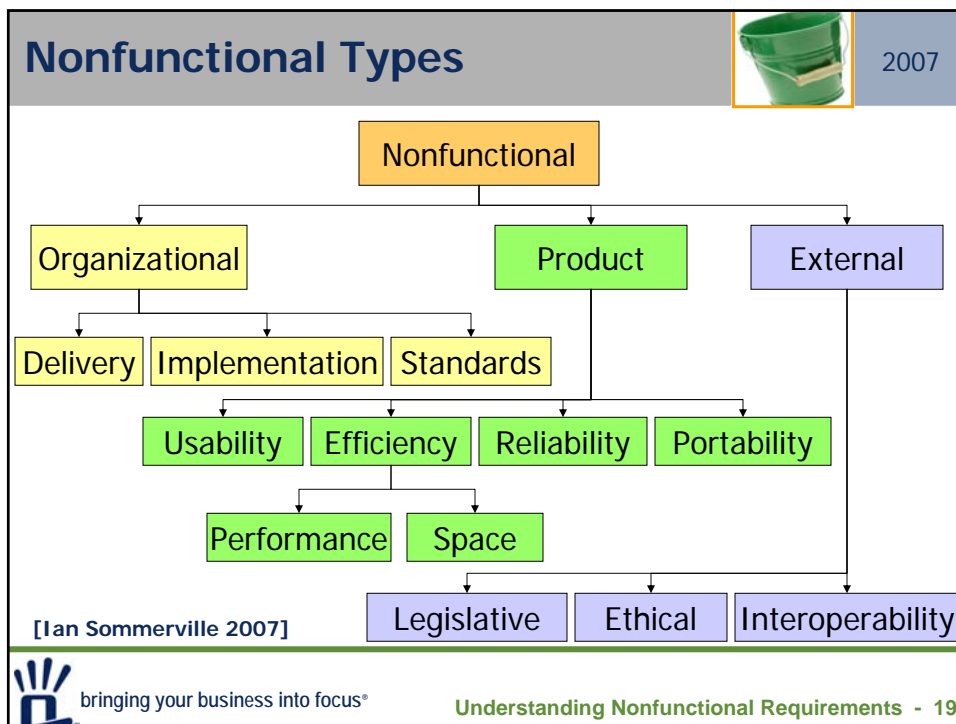
    graph LR
      GU[General Utility] --> P[Portability]
      GU --> AIU[As-Is Utility]
      GU --> M[Maintainability]
      AIU --> R[Reliability]
      AIU --> E[Efficiency]
      AIU --> HE[Human Engineering]
      M --> T[Testability]
      M --> U[Understandability]
      M --> Mod[Modifiability]
      P --> DI[Device-independence]
      P --> SC[Self-containedness]
      R --> Acc[Accuracy]
      R --> C[Completeness]
      R --> RI[Robustness / Integrity]
      E --> Cons[Consistency]
      E --> Accnt[Accountability]
      HE --> DE[Device Efficiency]
      HE --> Accs[Accessibility]
      HE --> Comm[Communicativeness]
      T --> SD[Self-descriptiveness]
      T --> Str[Structuredness]
      U --> Conc[Conciseness]
      U --> Leg[Legibility]
      Mod --> Aug[Augmentability]
    
```

[Barry Boehm 1976]


bringing your business into focus® Understanding Nonfunctional Requirements - 18

# Get Your Users "IN" the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements


Minneapolis IIBA Chapter 9 July 2009



**Get Your Users “IN” the Doghouse and Keep Yourself Out:  
Eliciting Nonfunctional Requirements**  
Minneapolis IIBA Chapter 9 July 2009

<b>A User-Focused Classification</b>		 User Focus
	<b>User Concern</b>	<b>Nonfunctional</b>
<b>Operation</b>	How well is it safeguarded against unauthorized access?	Access Security
	How dependable is it during normal operating times?	Availability
	How fast, how many, and how well does it respond?	Efficiency
	How accurate and authentic is the data?	Integrity
	How immune is the system to failure?	Reliability
	How resilient is the system from failure?	Survivability
	How easy is it to learn and operate the system?	Usability
<b>Revision</b>	How easy is it to modify to work in different environments?	Flexibility
	How easy is it to upkeep and repair?	Maintainability
	How easy is it to expand or upgrade its capabilities?	Scalability
	How easy is it to show it performs its functions?	Verifiability
<b>Transition</b>	How easy is it to interface with another system?	Interoperability
	How easy is it to transport?	Portability
	How easy is it to convert for use in another system?	Reusability

**Operation: Access Security**




**Definition** — the extent to which the system is safeguarded against deliberate and intrusive faults from internal and external sources.


**User concern**

- ☞ How well is the system safeguarded against unauthorized access?

**Example:**

Employees shall be forced to change their password the next time they log in if they have not changed it within the length of time established as “password expiration duration.”





bringing your business into focus®

Understanding Nonfunctional Requirements - 22

# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Operation: Availability



**Definition** — the degree to which users can depend on the system to be up (able to function) during “normal operating times.”

### User concern

- How dependable is the system during normal operating times?

### Example:

The Automated Teller Machine shall be at least 99.5 percent available on weekdays between 6:00 a.m. and midnight local time. The machine shall be at least 99.95 percent available on weekdays between 3:00 p.m. and 7:00 p.m. local time.



bringing your business into focus®

Understanding Nonfunctional Requirements - 23

## Operation: Efficiency



**Definition** — the extent to which the software system handles capacity, throughput, and response time.

### User concern

- How fast does it process (capacity)?
- How many at a time (throughput)?
- Response time?

### Example:

The system must download the new rate parameters within ten minutes of a non-scheduled rate change.



bringing your business into focus®

Understanding Nonfunctional Requirements - 24

# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Operation: Integrity



**Definition** — the degree to which the data maintained by the software system is accurate, authentic, and without corruption.

**User concern**

✎ How accurate and authentic is the data?

**Example:**

The integrity of the system data area must be checked by the internal audit system twice per second; if inconsistencies in the data are detected, the system operation should be disabled.



bringing your business into focus®

Understanding Nonfunctional Requirements - 25

## Operation: Reliability



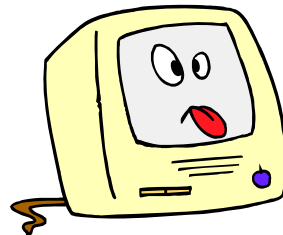
**Definition** — the extent to which the software system consistently performs the specified functions without failure.

**User concern**

✎ How immune is the system to failure?

**Example:**

No more than 10 claim assignments out of 5000 can be “unassigned” because of system failures.



bringing your business into focus®

Understanding Nonfunctional Requirements - 26

# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Operation: Survivability



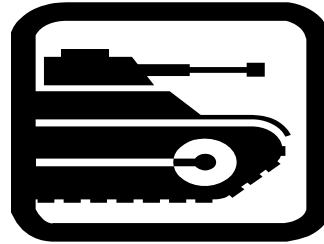
**Definition** — the extent to which the software system continues to function and recovers in the presence of a system failure.

### User concern

How resilient is the system from failure?

### Example:

All policy statement parameters shall have default values specified, which the Report Writer system shall use if a parameter's input data is missing or invalid.



bringing your business into focus®

Understanding Nonfunctional Requirements - 27

## Operation: Usability



**Definition** — the ease in which the user is able to learn, operate, prepare inputs and interpret outputs through interaction with a system.

### User concern

How easy is it to learn and operate the system?

### Example:



A trained order-entry clerk shall be able to submit a complete order for a product chosen from a supplier catalog in a maximum of 7 minutes, and an average order entry time of 4 minutes.



bringing your business into focus®

Understanding Nonfunctional Requirements - 28

# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Revision: Flexibility



**Definition** — the ease in which the software can be modified to adapt to different environments.

**User concern**

How easy is it to change and add new features?

**Example:**

It shall be possible to add a new delivery option for customer mailing method by developing and “plugging in” the functionality necessary to support that delivery option.



bringing your business into focus®

Understanding Nonfunctional Requirements - 29

## Revision: Maintainability



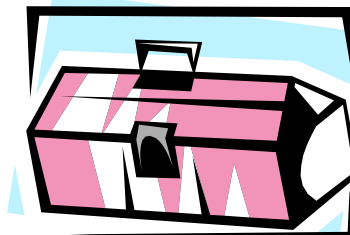
**Definition** — the ease in finding and fixing faults in the software system.

**User concern**

How easy is it to upkeep and repair the system?

**Example:**

The application development process must have a regression test procedure that allows complete re-testing within 2 business days.



bringing your business into focus®

Understanding Nonfunctional Requirements - 30

# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Revision: Scalability



**Definition** — the degree in which the software system is able to expand its processing capabilities upward and outward to support business growth.

**User concern**

- How easy is it to expand or upgrade the capabilities of the system?



**Example:**

Any increase in the number of customers shall not degrade system availability to an extent noticeable by any users.



bringing your business into focus®

Understanding Nonfunctional Requirements - 31

## Revision: Verifiability



**Definition** — the extent to which tests, analysis, and demonstrations are needed to prove that the software system will function as intended.

**User concern**

- How easy is it to show it performs its functions?

**Example:**

The maximum number of test cases to cover testing of any particular source code module shall be 20.



bringing your business into focus®

Understanding Nonfunctional Requirements - 32



# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Transition: Interoperability



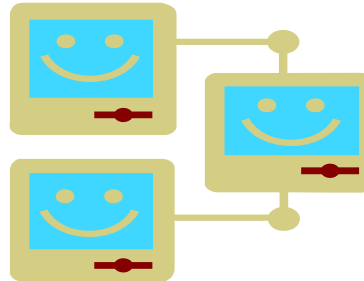
**Definition** — the extent to which the software system is able to couple or facilitate the interface with other systems.

### User concern

How easy is it to interface with another system?

### Example:

The system must be able to interface with any HTML (HyperText Markup Language) browser.



bringing your business into focus®

Understanding Nonfunctional Requirements - 33

## Transition: Portability



**Definition** — the ease in which a software system from its current hardware or software environment can be transferred to another environment.

### User concern

How easy is it to transport?

### Example:

The system is designed to run in business offices, but the intent is to have a version which will run in manufacturing assembly plants.



bringing your business into focus®

Understanding Nonfunctional Requirements - 34

# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## Transition: Reusability


**Definition** — the extent to which a portion of the software system can be converted for use in another.


**User concern**

- How easy is it to convert for use in another system?

**Example:**

The payment subsystem design is based on the payment module from the ALPHA product line. The ePAYZ system should not be modified unless absolutely necessary.



 bringing your business into focus® Understanding Nonfunctional Requirements - 35

## Session Learning Objectives Recap

		
Define nonfunctional requirements	Understand these vital, yet overlooked requirements	Classify common nonfunctional requirements

 bringing your business into focus® Understanding Nonfunctional Requirements - 36

# Get Your Users “IN” the Doghouse and Keep Yourself Out: Eliciting Nonfunctional Requirements

Minneapolis IIBA Chapter 9 July 2009

## A Requirement “Pattern”

Useful Tool

- Basic details
- Applicability
- Discussion
- Content
- Template(s)
- Example(s)
- Extra requirements
- Development considerations
- Testing considerations

**37 patterns!  
Over 400  
real-life  
examples!**



Stephen Withall, Microsoft Press, 2007.

bringing your business into focus®

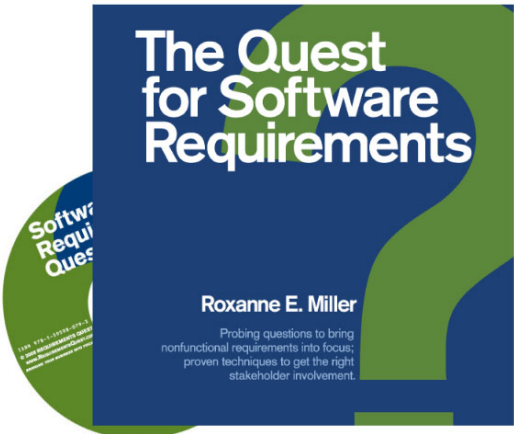
Understanding Nonfunctional Requirements - 37

## Over 2,000 Suggested Questions

Useful Tool

**NOW Available:**

- Book:**  
*The Quest for Software Requirements*
- CD (MS Excel):**  
*Software Requirements Questions*



**Order Yours Today!** [RequirementsQuest.myshopify.com](http://RequirementsQuest.myshopify.com)


bringing your business into focus®

Understanding Nonfunctional Requirements - 38

**Get Your Users “IN” the Doghouse and Keep Yourself Out:  
Eliciting Nonfunctional Requirements**  
Minneapolis IIBA Chapter 9 July 2009


**Business Analyst Development**

**Tuesday,  
October 20, 2009**



**2009  
WI BADD**  
Wisconsin Business Analyst  
Development Day

- 📍 Alliant Energy Center Exhibition Hall  
Madison, WI
- 📍 Visit <http://madison.theiiba.org>  
for more details and announcements
- 📍 Earn 6 hours of professional development

 bringing your business into focus® **Understanding Nonfunctional Requirements - 39**

**Questions**



**REQUIREMENTS®  
QUEST**  
bringing your business into focus

**iIBA™ INDUSTRY Sponsor**

**iIBA Endorsed Education Providers**  
Charter Member

Presented by: Roxanne E. Miller, CBAP  
[www.RequirementsQuest.com](http://www.RequirementsQuest.com)