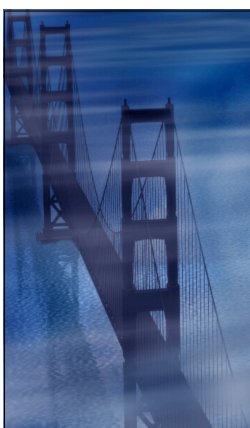
A photograph of the Golden Gate Bridge in San Francisco, viewed from a low angle looking up at the suspension towers and cables. The sky is blue with some clouds.

CONTEXT:
its discovery, its form,
and its management
in non-functional
requirements

HANDOUTS

One-Day Tutorial
RE'06 Conference
Minneapolis, Minnesota
September 12, 2006

A photograph of the Golden Gate Bridge in San Francisco, viewed from a low angle looking up at the suspension towers and cables. The sky is blue with some clouds.

CONTEXT:
its discovery, its forms,
and its management
in non-functional
requirements

Donald C. Gause
Research Professor
Thomas J. Watson School
Binghamton University, USA
dgause@stny.rr.com

(Continuation of title slide)

CONTEXT: its discovery, its forms,
and its management in non-functional
requirements

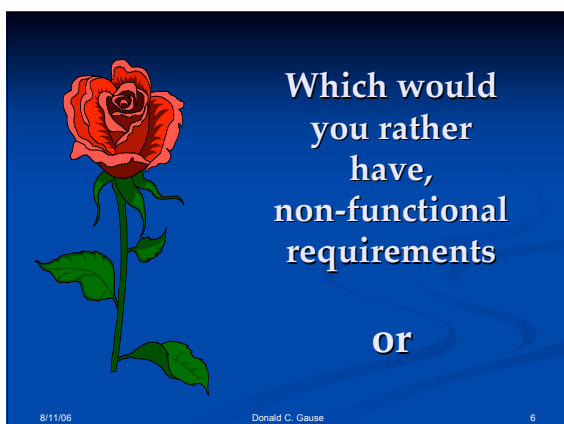
Donald C. Gause,
Principal, Savile Row, LLC
Professor Emeritus, Systems Science
Research Professor, Department of Bioengineering
Thomas J. Watson School of Engineering
State University of New York at Binghamton
dgause@stny.rr.com

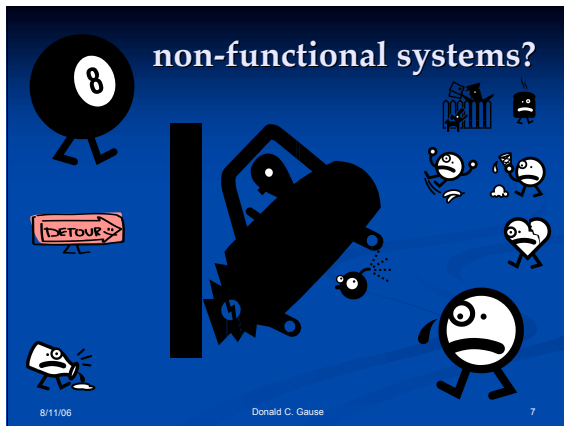
NON-FUNCTIONAL REQUIREMENTS

Seeing - the Context Side of Design

One-Day Tutorial
RE'06 Conference
Minneapolis, Minnesota
September 12, 2006

Donald C. Gause, Research Professor
Thomas J. Watson School of Engineering
Binghamton University
dgause@stny.rr.com





Polynomial evaluation – self-conscious programming

The following program, written in LockerRoom++, evaluates the third order polynomial, $Y = A + BX + CX^2 + DX^3$.

INP A	ZER	OUT A
INP B	ADD D	OUT B
INP C	MUL X	OUT C
INP D	ADD C	OUT D
INP X	MUL X	OUT
	ADD B	OUT X
	MUL X	OUT Y
	ADD A	END
	STO Y	

8/11/06 Donald C. Gause 8

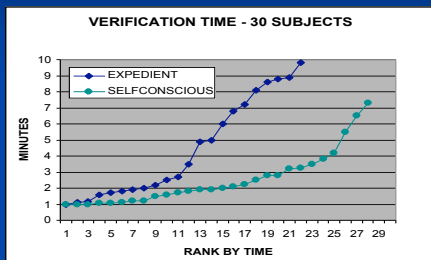
Polynomial evaluation – expedient programming

The following program, written in LockerRoom++, evaluates the third order polynomial, $O = S + CI + FI^2 + XI^3$.

ZER	INP S	MUL I
INP X	MUL I	OUT I
INP I	OUT Y S	ADD S
ADD I	OUT C	STO O
MUL X	OUT F	OUT O
INP F	OUT X	END
ADD F	ADD C	
INP C	OUT	

8/11/06 Donald C. Gause 9

Self-conscious vs expedient



8/11/06

Donald C. Gause

10

Polynomial Evaluation – Non-Functional Aspects

FUNCTION	PRIMARY OBJECTIVE	STYLE
$O = S + CI + FI^2 + XI^3$	Ease of programming	expedient
$Y = A + BX + CX^2 + DX^3$	Understandability	not implemented
$Y = (((DX+C)X+B)X+A)$	Execution efficiency	self-conscious

8/11/06

Donald C. Gause

11

To OPTIMIZE or to CONSTRAIN,
that is the question.

■ OPTIMIZE



We can't get
enough of these
attributes - the
more, the better.

- CONSTRAIN
We must have the
specified level of
these attributes but
surpassing this
level is to no
advantage.

8/11/06

Donald C. Gause

12

A programming problem

Develop, implement, test, and deliver a PL/I program to solve n -simultaneous linear algebraic equations in n unknowns. The algorithm to be used is specified as Gauss elimination with pivotal condensation. The program must be able to solve systems for $n = 1, 2, \dots, 100$. The program must also identify systems in which the coefficient matrix is singular and inform the user of this condition.

"Goals and Performance in Computer Programming," Gerald M. Weinberg and Edward L. Schulman, *Human Factors*, 1974, 16(1), pp. 70-77

8/11/06

Donald C. Gause

13

A Programming Problem Six Teams

- Six member teams selected by experimenters
- Graduate students in Software Engineering
- Roughly 50% full time, 50% part time professionals
- Teams selected to be as identical as possible in
 - Skill level
 - Experience level
 - Motivation level

8/11/06

Donald C. Gause

14

A Programming Problem Performance Ranges

	BEST TEAM	WORST TEAM	Perf. Ratio
DEV. TIME	A	B	2.64
STORE. EFF.	B	E	6.73
LOC	C	E	5.03

8/11/06

Donald C. Gause

15

A Programming Problem Overall Results

TEAM	OPT	DT	SE	LOC	PC	OC	QUAL
D	PC	4	3	3	1-2	2	3.00
C	LOC	2-3	2	1	3	6	3.38
E	OC	2-3	6	6	1-2	1	4.00
B	SE	6	1	2	4	5	4.25
F	ET	5	4	4	5	4	4.40
A	DT	1	5	5	6	3	4.75

8/11/06

Donald C. Gause

16

Universal Furniture Design I

Design a piece of furniture that can be manufactured by individuals who have physical or mental impairments, disabilities or conditions. The piece must satisfy the following requirements:

- It must sell for under twenty dollars.
- It can be shipped unpackaged without fear of damage.
- It can serve as a chair, footstool, room divider, coffee table, book shelf, and end table.



8/11/06

Donald C. Gause

17

Universal Furniture Design II

Design a piece of furniture that can be manufactured by individuals who have physical or mental impairments, disabilities or conditions. The piece must satisfy the following requirements:

- It must sell for under twenty dollars.
- It can be shipped unpackaged without fear of damage.
- It can be used to sit on, prop feet on, divide rooms, put drinks, magazines, books, and lamps on.



8/11/06

Donald C. Gause

18

THE SOCIAL DIMENSION

Design a piece of furniture that can be manufactured by individuals who have physical or mental impairments, disabilities or conditions. The piece must satisfy the following requirements:

- It must sell for under twenty dollars.
- It can be shipped unpackaged without fear of damage.
- **It can serve as a chair, footstool, room divider, coffee table, book shelf, and end table.**
- **It can be used to sit on, prop feet on, divide rooms, put drinks, magazines, books, and lamps on.**

8/11/06 Donald C. Gause 19

THE SOCIAL DIMENSION

RESULTS	NAME	FUNCTION
	(chair, ...)	(sit on, ...)
▪ Social behavior	?	?
▪ Presentations		
▪ Highly innovative		

8/11/06 Donald C. Gause 20

THE SOCIAL DIMENSION

TO BE DISCUSSED AND REVEALED AS
PART OF THE TUTORIAL

RESULTS	NAME	FUNCTION
	(chair, ...)	(sit on, ...)
▪ Social behavior	?	
▪ Presentations		
▪ Highly innovative		

8/11/06 Donald C. Gause 21

THE SOCIAL DIMENSION 
**TO BE DISCUSSED AND REVEALED AS
PART OF THE TUTORIAL**

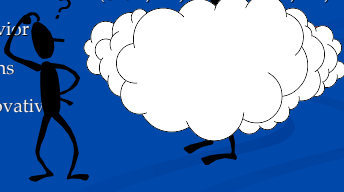
<u>RESULTS</u>	<u>NAME</u>	<u>FUNCTION</u>
▪ Social behavior ▪ Presentations ▪ Highly innovative	(chair, ...)	(sit on, ...)



8/11/06 Donald C. Gause 22

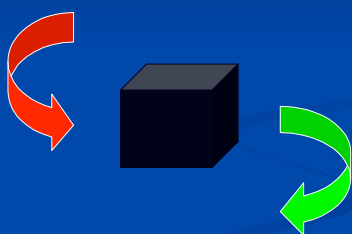
THE SOCIAL DIMENSION 
**TO BE DISCUSSED AND REVEALED AS
PART OF THE TUTORIAL**

<u>RESULTS</u>	<u>NAME</u>	<u>FUNCTION</u>
▪ Social behavior ▪ Presentations ▪ Highly innovative	(chair, ...)	(sit on, ...)

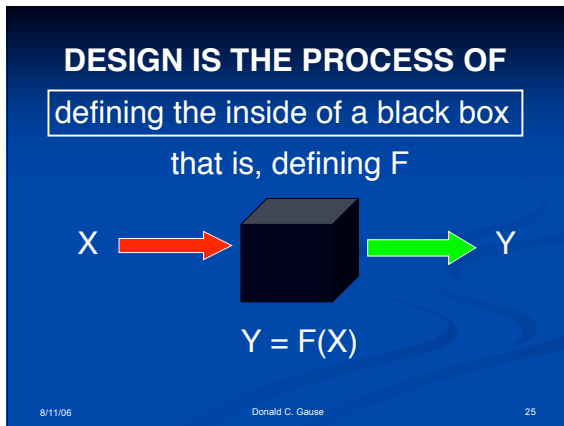


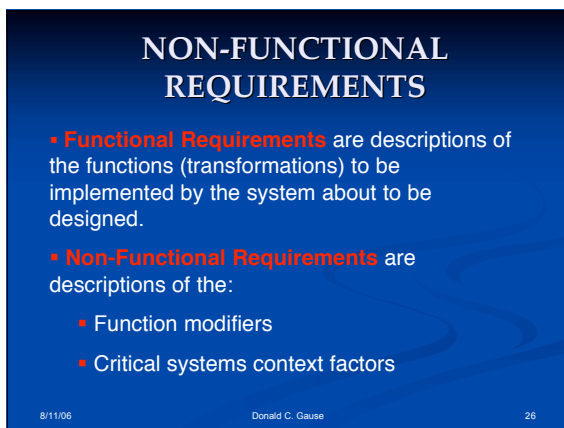
8/11/06 Donald C. Gause 23

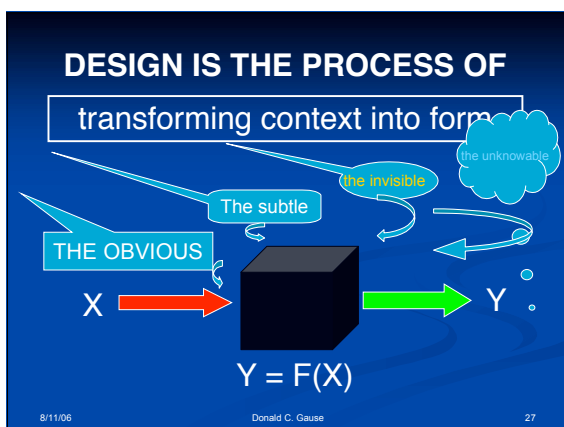
DESIGN IS THE PROCESS OF
defining the inside of a black box

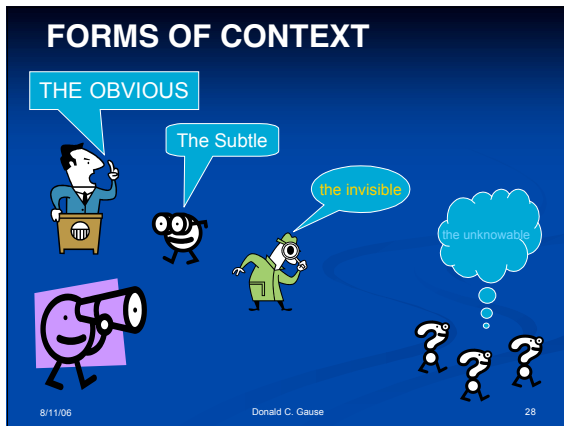


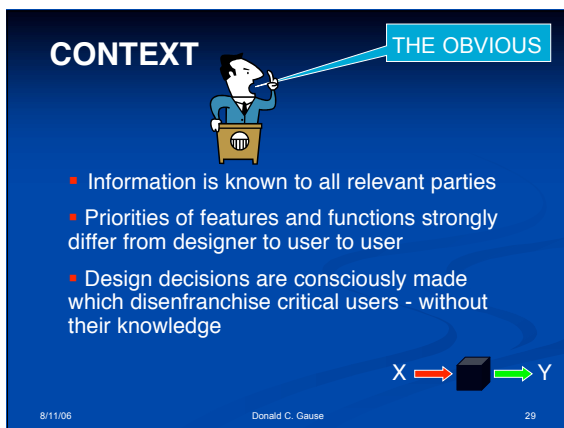
8/11/06 Donald C. Gause 24

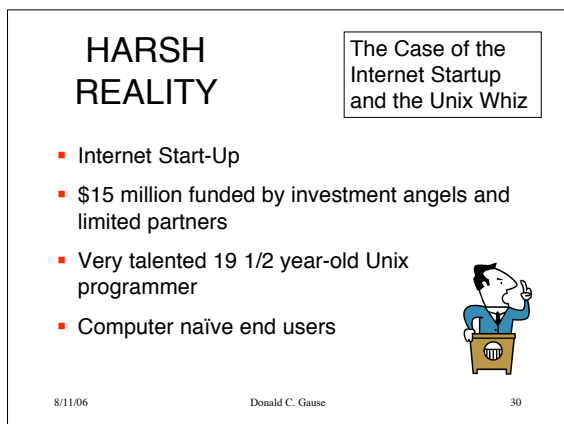












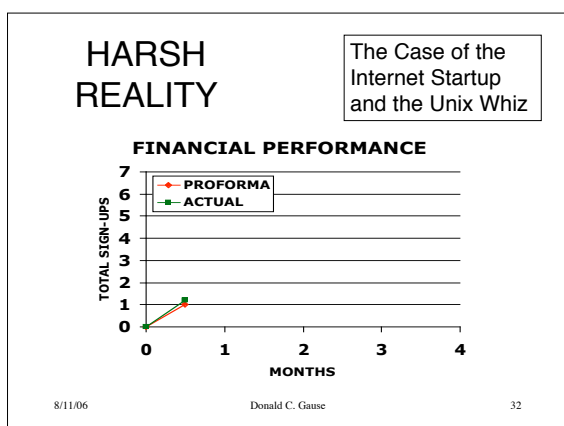
HARSH REALITY

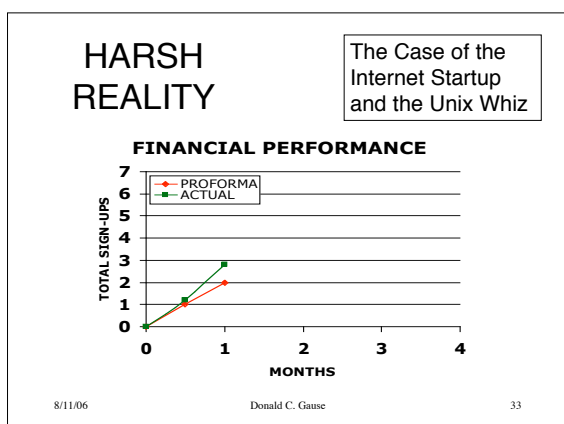


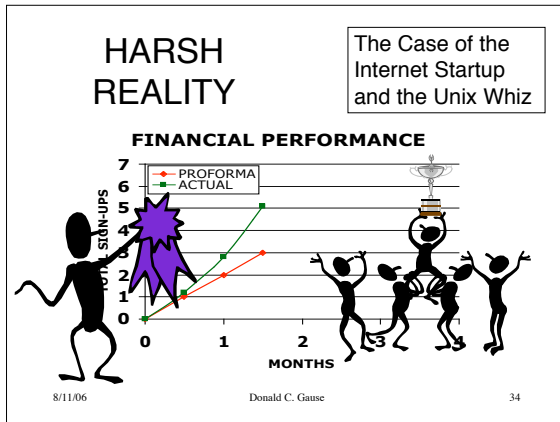
- No written requirements. All requirements were verbally transmitted from the marketing manager to the programmer.
- Progress was measured by the number of subscribers signed up for internet access.

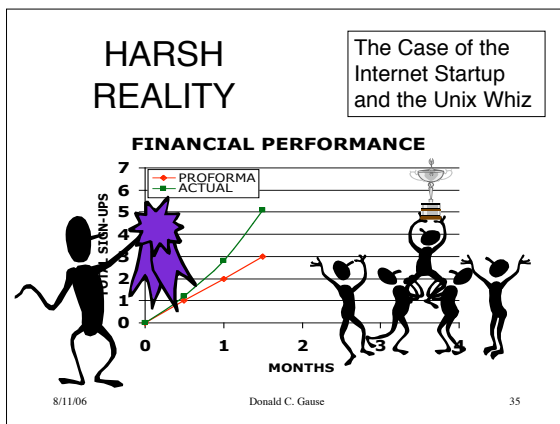
The Case of the Internet Startup and the Unix Whiz

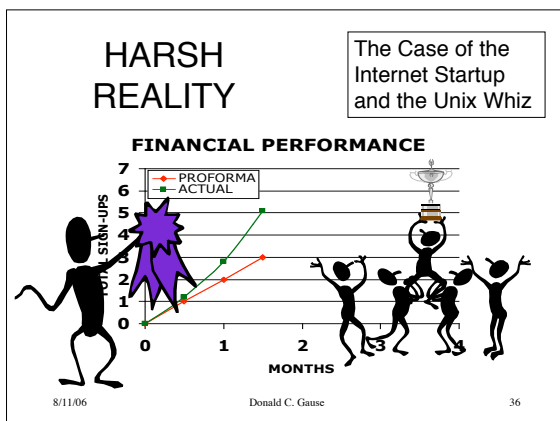
8/11/06 Donald C. Gause 31

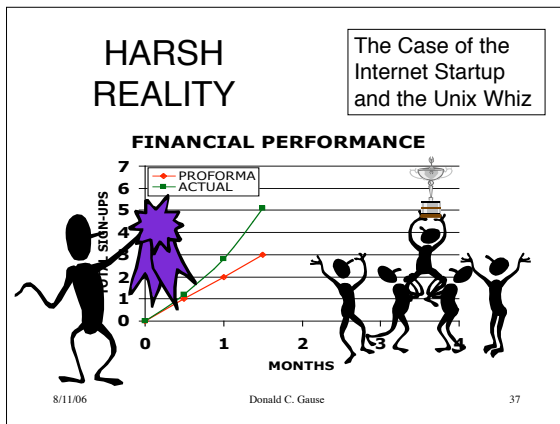


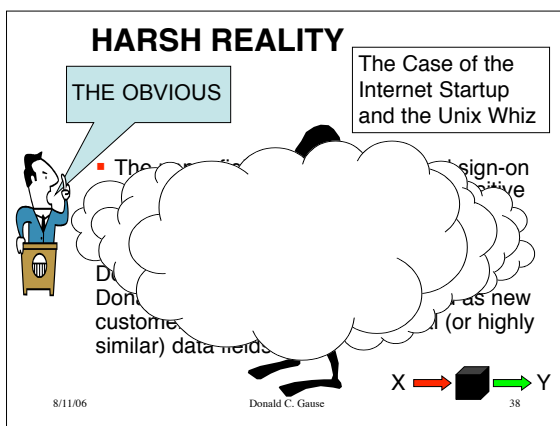












CONTEXT

THE OBVIOUS

- Define a complete set of users and a plan for incorporating all critical users before, during, and after design and development. All critical user signatures are required.
- Develop and document business requirements
- Clearly state the project scope including *out of scope specification*.

REMEDIES

8/11/06 Donald C. Gause 39

CONTEXT

The subtle



- Critical information is known to some user constituencies but not known to the designers.
- This information will not be accounted for in the design form and therefore not discovered until product release.

X →  → Y

8/11/06 Donald C. Gause 40

HARSH REALITY

The Case of the BIG BANK - small developer

The subtle



- BIG BANK contracted a small banking information development firm for a world-wide customer service system.
- After 18 months development time, it was discovered, by the software firm, that BIG BANK needed to support 2,000 terminals.

X →  → Y

8/11/06 Donald C. Gause 41

HARSH REALITY

The Case of the BIG BANK - small developer



▪ The system was designed to support 10 terminals.

OOOOPS!!!

8/11/06 Donald C. Gause 42

CONTEXT The subtle **REMEDIES**



- Have a clear statement of design limitations in the business requirements.
- Provide traceability of design responsibilities through signoff procedures.
- Manage conformance of system requirements with management requirements.

X →  → Y

8/11/06 Donald C. Gause 43

 the invisible **CONTEXT**

- These are the relevant context issues that are invisible to the designers for one of two reasons:
 - They lurk below our consciousness because of our lack of imagination (~~gee whiz~~ features) or,
 - They are hidden from the designers because of organizational or other information impediments.

X →  → Y

8/11/06 Donald C. Gause 44

HARSH REALITY the invisible **The Case of "Just give me the data elements."**



- Strategic global finance system
- Authoritative and real time
- Tracking daily transactions in the hundreds of billions of dollars
- Offices in over two hundred countries
- Developed internally

X →  → Y

8/11/06 Donald C. Gause 45

HARSH REALITY (the invisible)

The Case of "Just give me the data elements."

- A demonstration was given to the BIG BANK CEC six weeks before the system was to go on line. The CEC was invited to query the system for specific information of interest.
- The CEC asked for particular financial data of critical use in running the bank.
- The information was accurately provided by the system ...

8/11/06 Donald C. Gause 46

minute after the query was initiated. (the invisible)

The Case of "Just give me the data elements."

8/11/06 Donald C. Gause 47

CONTEXT (the invisible) **REMEDIES**

- Develop and regularly apply brainstorming and other ideation skills, both individually and collectively at appropriate times from concept development through requirements elicitation and on to product design and delivery.
- Practice good requirements management including effective management of user expectations.

8/11/06 Donald C. Gause 48

CONTEXT



- These are the issues that cannot be discovered until the product is delivered due to:
 - Undiscoverable environmental influences
 - Unanticipated and unpredictable changes in the environment $X \rightarrow \text{black box}$
 - Unintended consequences $\text{black box} \rightarrow Y$

8/11/06 Donald C. Gause 49

HARSH REALITY



The Case of the Loan Server Metrics

- To provide enhanced interactive service
- Dramatically reduce approval times
- Loans in the millions or billions
- For institutional global clients
- With highly specialized cases



$X \rightarrow \text{black box} \rightarrow Y$

8/11/06 Donald C. Gause 50

HARSH REALITY



The Case of the Loan Server Metrics

- Featuring
 - Current, authoritative data base
 - Very quick responsive times
 - Monthly performance reports to
 - Global finance management
 - Marketing




$X \rightarrow \text{black box} \rightarrow Y$

8/11/06 Donald C. Gause 51

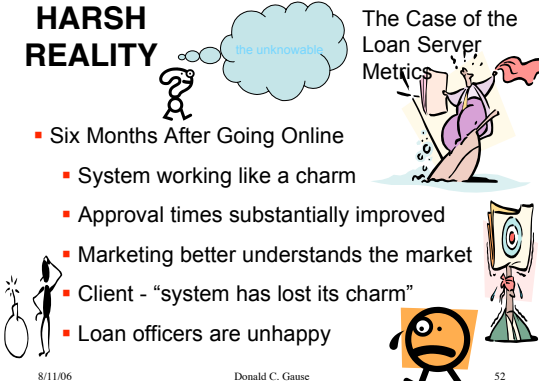
HARSH REALITY

the unknowable

The Case of the Loan Server Metrics

- Six Months After Going Online
 - System working like a charm
 - Approval times substantially improved
 - Marketing better understands the market
 - Client - "system has lost its charm"
 - Loan officers are unhappy

8/11/06 Donald C. Gause 52



REMEDIES

CONTEXT

the unknowable

- Hold dedicated brainstorming sessions for unintended consequences
- Design for adaptation to unforeseen context surprises
- Conduct formal, well organized, post product release reviews.

8/11/06 Donald C. Gause 53

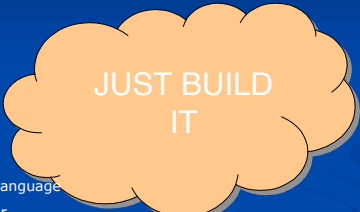


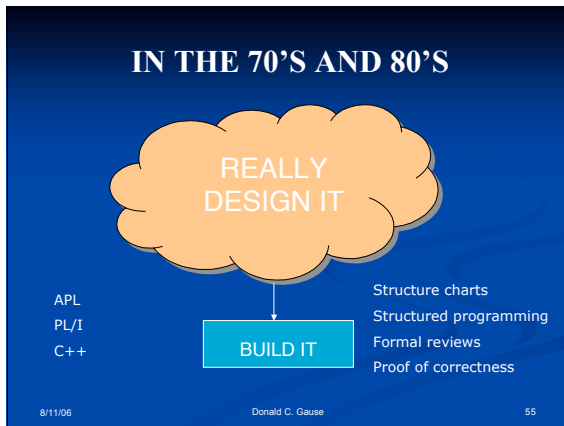
IN THE 50'S AND 60'S

JUST BUILD IT

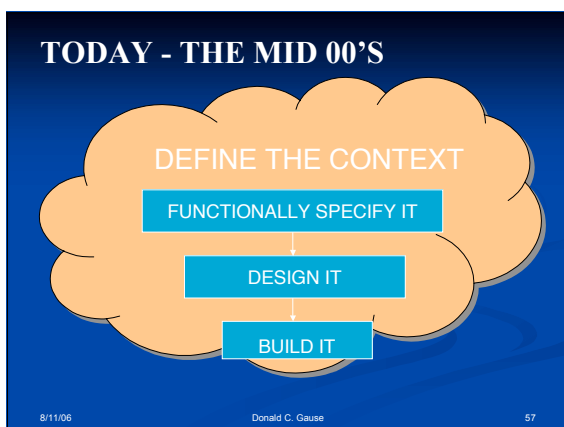
Machine language
Assembler
FORTRAN
COBOL

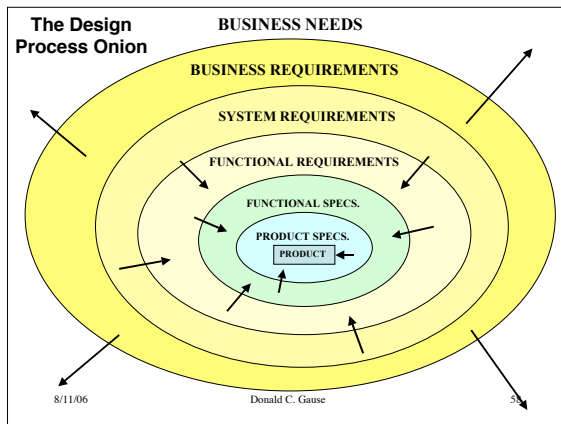
8/11/06 Donald C. Gause 54











Design is the process of:

- * Removing ambiguity
- * Solving a problem
- * Floating like a butterfly, stinging like a bee
- * Searching a state space for "good" solutions
- * Defining the inside of a black box
- * Transforming *design issues* into *design form*

8/11/06 Donald C. Gause 59

Design Issues

- * An issue is any relevant question that doesn't have an immediate, authoritative answer.
- * This may be because:
 - * The answer is currently unknown
 - * The answer is known but unknown to the design team
 - * The answer is known but the design team lacks the ability to authorize it
 - * The answer is known and there are several competing answers

8/11/06 Donald C. Gause 60

Issues Management

- ✳ Spend dedicated time raising design issues
- ✳ Develop a meeting discipline within the design organization
 - ✳ Any question that can't be resolved within an agreed amount of time goes onto the documented *open issues list*.
- ✳ The issues list is managed by a designated *issues manager*
 - ✳ A responsible person and a resolution time are assigned for the resolution of each issue
 - ✳ When an issue is resolved it moves to the *closed issues list* along with the time, final resolution, and responsible party
 - ✳ All interested parties have access to the issues lists

8/11/06 Donald C. Gause 61

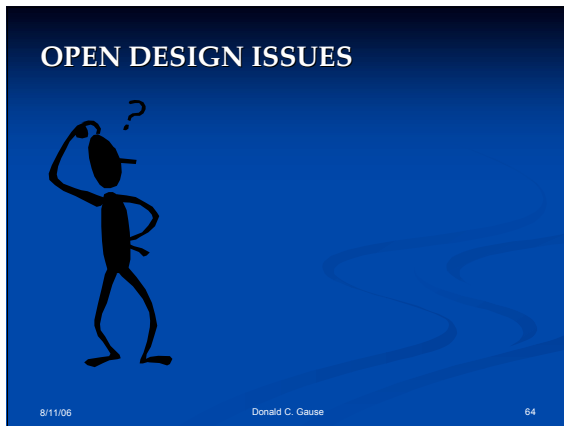
Of the 14 oft used words in this
this simple sentence, which one
is incorrect?

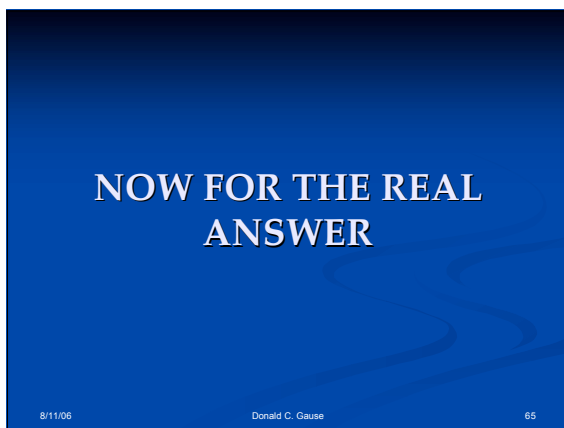
8/11/06 Donald C. Gause 62

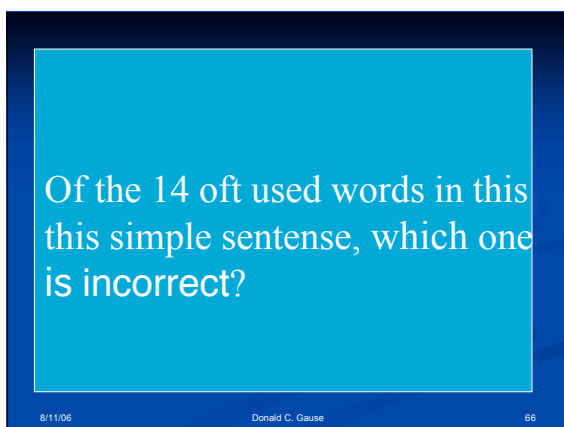
PROBLEM INTERPRETATION(S)



8/11/06 Donald C. Gause 63







PROBLEM STATEMENT

- Whose problem was it?
- Whose problem is it?
- Whose problem might the problem become?



8/11/06 Donald C. Gause 67

PROBLEM STATEMENT

- How will we recognize a solution?
- What are the unintended consequences of the solution?



8/11/06 Donald C. Gause 68

Design is the process of:

- ✱ Removing ambiguity
- ✱ Solving a problem
- ✱ Floating like a butterfly, stinging like a bee
- ✱ Searching a state space for "good" solutions
- ✱ Defining the inside of a black box



8/11/06 Donald C. Gause 69

Design is the process of:

- ✱ Solving a problem
- ✱ Creating an opportunity

8/11/06 Donald C. Gause 70

Design is the process of:

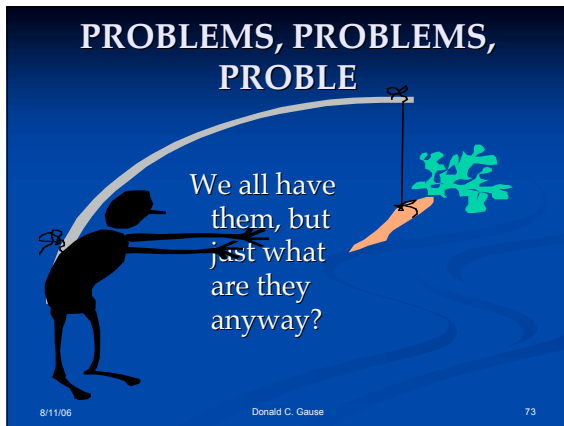
- ✱ Solving messy or heuristic problems
- ✱ Specifying the form of the solutions

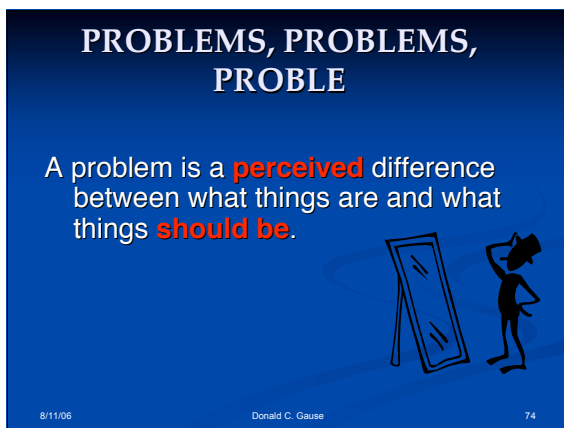
8/11/06 Donald C. Gause 71

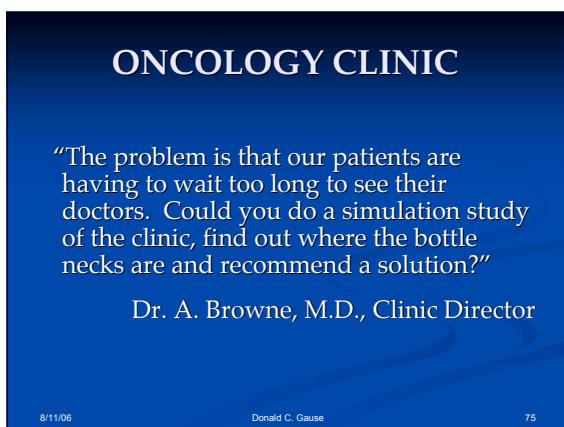
Requirements is the process of:

- ✱ Defining messy or heuristic problems
- ✱ Defining the context messy problems

8/11/06 Donald C. Gause 72







- Dr. Browne is the Lab Director
- He is active and well-known in the community
- There are three other Board Certified oncologists in the Lab.
- Patients come, by appointment, regularly to the Lab.
- Patients come from a 50 mile radius.

- Patients were waiting too long
- Had to wait to give blood
- Had to wait to see doctor
- Not enough doctors
- Had only old magazines
- Only medical magazines
- Cigarette machines in lobby
- Need more than one doctor.

Oncological remedies

SUBJECT TO DISCUSSION

Place the patient's best interests first

With the best of intentions, the way we do things can be the worst

Top priority is to save lives and financial costs

Loss of trust in the medical profession

Patients are not always helped by machines.

The collage includes a glowing lightbulb, a large question mark, a white ambulance with 'AMBULANCE' written on its side, a pharmacy sign with a red 'X' and 'Rx', a doctor in a white coat, a hand holding a heart, and a single pill.

78

[illegible]

PROBLEM DEFINING

- Statement of the problem to be solved
- Definition of the problem to be solved



8/11/06 Donald C. Gause 79

PROBLEM STATEMENT

- Whose problem was it?
- Whose problem is it?
- Whose problem might the problem become?



8/11/06 Donald C. Gause 80

PROBLEM STATEMENT

- How will we recognize a solution?
- What are the unintended consequences of the solution?



8/11/06 Donald C. Gause 81

Messy (wicked) problem

- Involve lots and lots of people in lots and lots of ways.
- Have as many definitions as people involved.
- The solutions must be satisfactory to lots and lots of people in their lots and lots of roles.
- There is no such thing as an optimal solution.

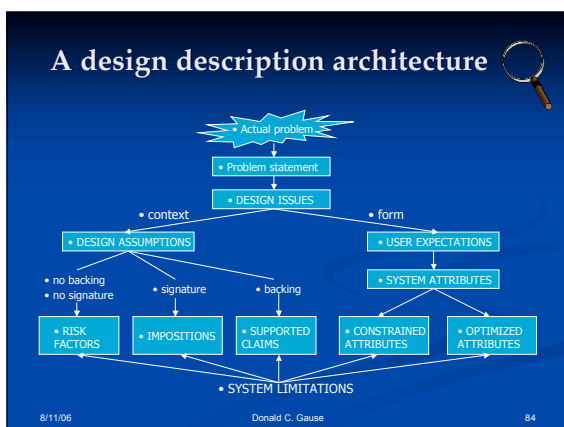
8/11/06 Donald C. Gause 82

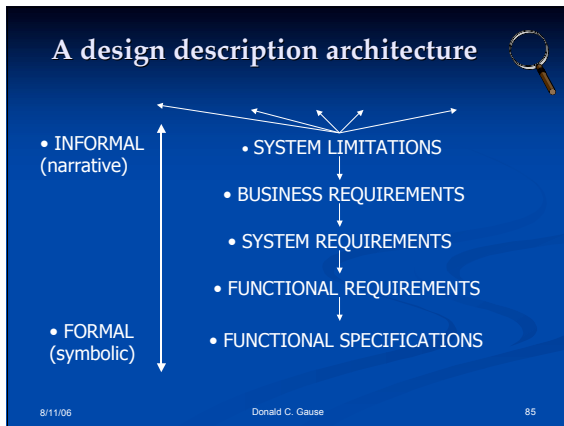
Messy (wicked) problems:

- Because of the complexity of the task, the design of computer systems is a **satisficing**¹ (rather than an optimizing) activity.

1) Term coined by Herbert Simon, 1957

8/11/06 Donald C. Gause 83





Design responsibilities

- **EXPERIENCE**
There had never been a recorded incident in which more than four water-tight compartments had been simultaneously damaged at sea.
- **ASSUMPTION**
No more than four water-tight compartments will be breached in a single incident.
- **DESIGN DECISION**
The Titanic must be designed so that it will stay afloat if four or fewer of the sixteen water-tight compartments are flooded.

"Software Safety: Why, What, and How," Nancy G. Leveson, *Computing Surveys of the ACM*, Vol 18, No. 2, June, 1986

8/11/06 Donald C. Gause 86

DESIGN ASSUMPTIONS

- Assumptions are design decisions made for the designers.
- They are an essential part of design.
- They always involve risk.
- The designer will be held accountable for all unrecognized assumptions.
- Implicit assumptions may involve unduly high invisible risk.
- The signatory is held accountable for all explicitly stated assumptions.

8/11/06 Donald C. Gause 87

Critical design assumption

- All critical assumptions must be documented and backed.
- The validity of these assumptions must be qualified.
- The consequences of a false assumption must be understood.
- The assumptions and consequences must be conveyed to all critical users.

8/11/06

Donald C. Gause

88

Context free questions

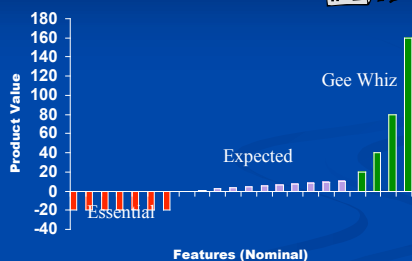
- Who is the client?
- What is the solution really worth to the client?
- Where else can the solution to this problem be obtained?
- What precision is required in the solution to this problem?

8/11/06

Donald C. Gause

89

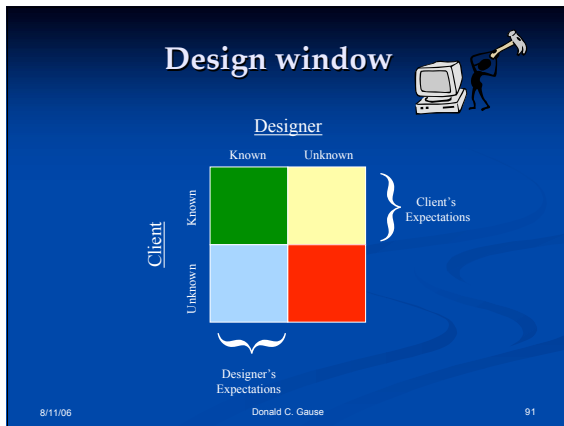
Design features

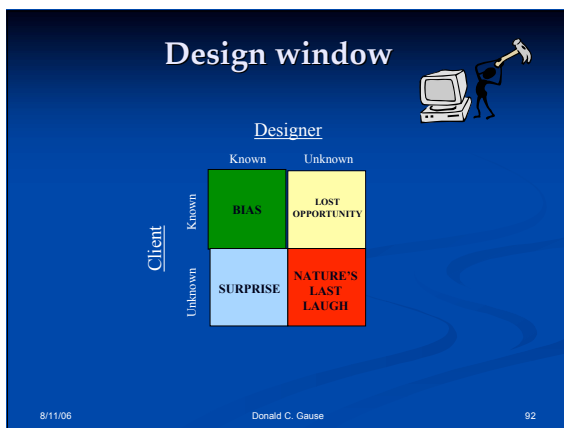


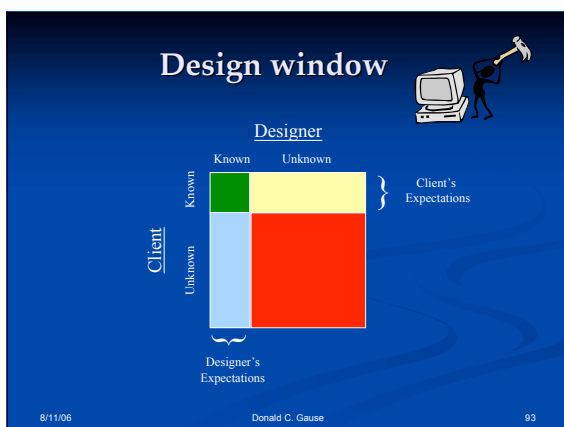
8/11/06

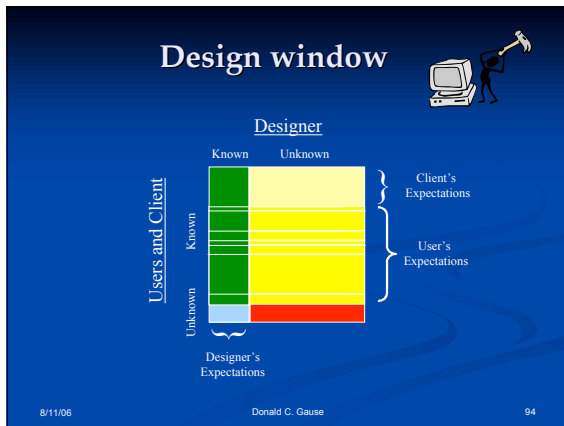
Donald C. Gause

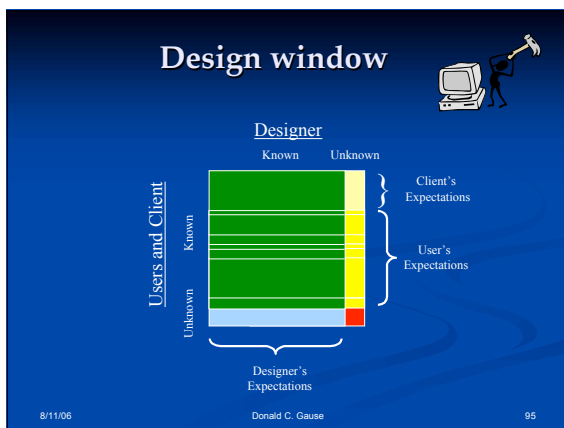
90

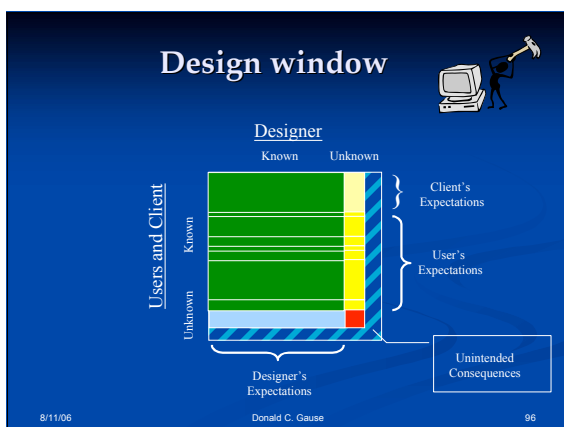


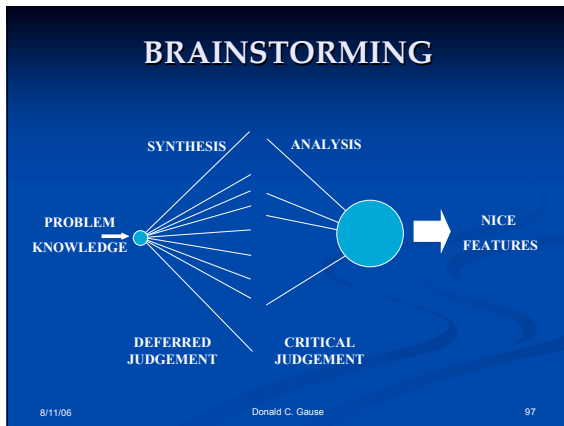


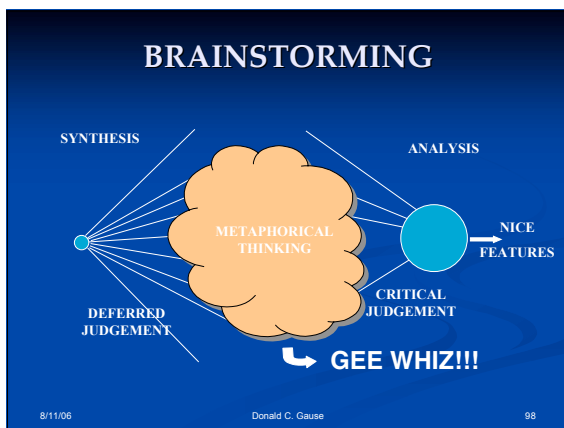












Expand the Concept of User

A user is any individual who is affected in any way by the system about to be designed

- ♦ Those held accountable for the success of the product
- ♦ Those affected by the design and development process
- ♦ Those immediately using the product
- ♦ Those receiving information produced by the product
- ♦ Those affected by the information produced by the product
- ♦ . . .

The slide features two small illustrations: one of a person sitting at a computer in the top right corner, and another of a person holding a glowing lightbulb in the bottom right corner.

8/11/06 Donald C. Gause 99

Brainstorming all conceivable users (desktop - 114 users)



✓ Night cleaning crew	✓ Aliens
✓ Office workers	✓ Changelings
✓ Supervisors	✓ Klingons
✓ Network administration	✓ Spies
✓ Kids	✓ Bajorans
✓ Students	✓ Tennis players
✓ Teachers	✓ Video game players
✓ Machine tool operators	✓ Hackers
✓ Administrators	✓ Web surfers
✓ Web masters	✓ Virus creators
✓ Web page creators	✓ Application developers
✓ Secretaries	✓ Dogs

8/11/06 Donald C. Gause 100

The metaphorical part – *Bajorans*

EXPLORING BAJORANS

- It scares me
- Like scaring the bajorans
- Lots of scary bajorans
- Scary because they are little.
- Black and quick.
- A night on Bald Mountain.
- Organ music and lightening.
- Bajorans running all over the screen
- Screaming bajorans running all around the screen

BACK TO THE PROBLEM

- What about desktops?
- Bajorans on desktop.
- Bajorans tracking caccadic eye motion of user
- Usability features
- Identify user by ciccadic signature over certain patterns
- Provide musical or pitch and volume feedback to the user
- Provide eye control - replaces or augments the mouse

8/11/06 Donald C. Gause 101

Frivolous users and features



- ★ Computer Desktop Interface
 - *Dogs* - Kitten on the keys feature
 - *Sponuses* - Priority notification screen interrupt
- ★ Computer Aided Design
 - *Flashers* - Visual execution feature
 - *Dead People* - First use indicator feature
- ★ Groupware Software
 - *Single Bar Fly* - Role playing and alias features
 - *Disruptive User* - Comment currency feature

8/11/06 Donald C. Gause 102

CRITICAL CONSTITUENCIES



- ♦ CLIENT - controls project via allocation of resources
- ♦ EXPERT - has information useful to product development
- ♦ DESIGNER - creates the product specification
- ♦ PRODUCER - creates the product
- ♦ END USER - directly uses the product
- ♦ SECONDARY USER - working life affected by the product
- ♦ TERTIARY USER - personal life affected by the product
- ♦ DATA PROVIDER - provides data to the working product

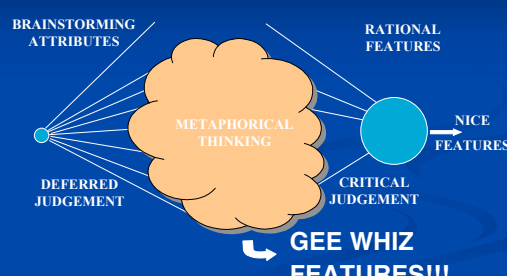
8/11/06 Donald C. Gause 103

Brainstorming all conceivable attributes (desktop - 181)

☐ Easy to use	☐ Easy to understand
☐ Fast	☐ Proprietary
☐ Inexpensive to operate	☐ Self configuring
☐ Accurate	☐ Interactive
☐ Low cost to produce	☐ Fast response
☐ Defect free	☐ Secure
☐ On-time delivery	☐ Fault tolerant
☐ Small	☐ Simple
☐ Available	☐ <i>Sensual</i>
☐ Reusable	☐ Fun to use
☐ Extendable	☐ <i>Wubulous</i>
☐ Real-time updates	☐ <i>Celibate</i>

8/11/06 Donald C. Gause 104

BRAINSTORMING



The diagram illustrates a brainstorming process. On the left, 'BRAINSTORMING ATTRIBUTES' points to a central orange cloud labeled 'METAPHORICAL THINKING'. On the right, 'RATIONAL FEATURES' points to a blue circle labeled 'NICE FEATURES'. Both the cloud and the circle point to a blue circle labeled 'CRITICAL JUDGEMENT'. Below 'CRITICAL JUDGEMENT', an arrow points to the text 'GEE WHIZ FEATURES!!!'. The text 'DEFERRED JUDGEMENT' is also present near the brainstorming attributes.

8/11/06 Donald C. Gause 105

The metaphorical part – *Wubbulous*

EXPLORING WUBBULOUS ■ I don't know what it is. ■ Sounds mysterious. ■ Lots of wobbul. ■ Wonderful wobbul. ■ Wobbul is soft and maleable. ■ Wonderfully mysterious. ■ Productively mysterious. ■ Like silly putty. ■ See-thru silly putty.	BACK TO THE PROBLEM ■ What about desktops? ■ Wubbulous desktop. ■ Wubbulous desktop feature. ■ Wubbulous productivity feature. ■ Feature to remove mystery. ■ I never know what files are open or where they are on the screen. ■ See-thru files
--	---

8/11/06 Donald C. Gause 106

Frivolous attributes and features

- **COMPUTER DESKTOP INTERFACE**
 - *Wubbulous* - Texture on desktop allows users to overlap icons yet identify their presence.
 - *Celibate* - The desktop rejects all advances from other software, hardware, and networks until vows are exchanged.
- **GROUPWARE SOFTWARE**
 - *Fortune cookieish* - witty, interesting, smart aleck, pseudo-mystical responses.
 - *Hot and sexy* - personal chat rooms, faces, bodies and clothes of personas selectable by users; progress toward objectives displayed graphically.

8/11/06 Donald C. Gause 107

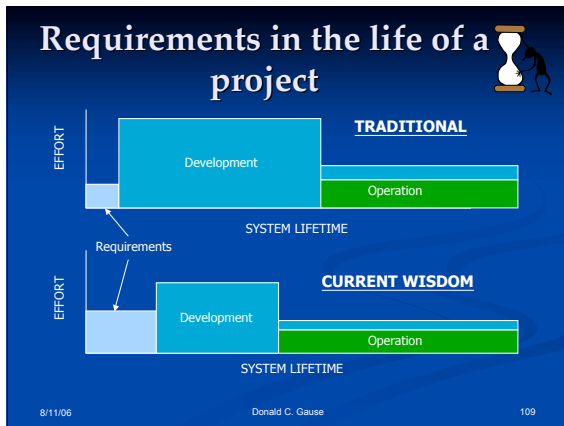
Brainstorming

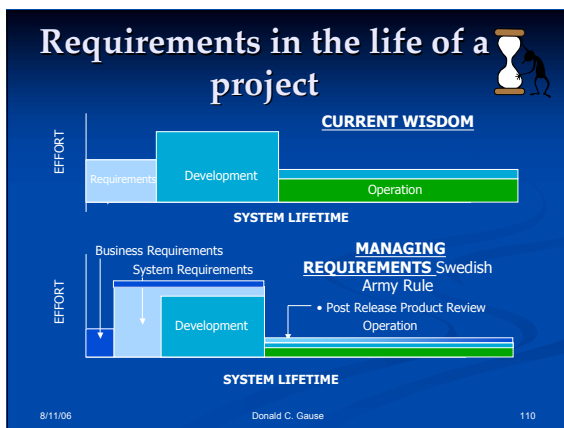
• Likes • Dislikes • Users • Attributes

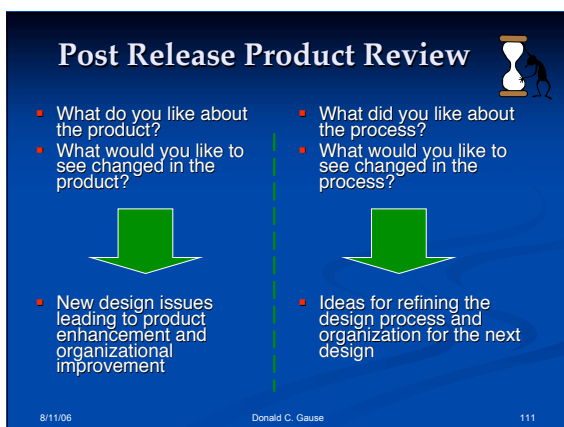
Improvement features User features Attributes

GEE WHIZ GEE WHIZ

8/11/06 Donald C. Gause 108







oops 

"I think that our functional specs are mostly defined by the programmers at implementation time."

L.R., Chicago, Illinois, May 24, 2000

8/11/06 Donald C. Gause 112

oops 

"We write our requirements after we have designed the product."

M.J., Minneapolis, Minnesota, January 24, 2001

8/11/06 Donald C. Gause 113

oops 

"I don't care what anyone says. The sign-on procedure is implemented in case-sensitive Unix code."

Anonymous 19 year old computer whiz, Seattle, Washington, January 24, 2000

8/11/06 Donald C. Gause 114

When are the requirements complete?

- **Business Requirements**
Whenever the business needs have been documented to the extent that:
 - the problem (opportunity) is well understood and defined,
 - the responsible person has been given enough information to make a well informed business decision on funding, starting or abandoning development.

8/11/06 Donald C. Gause 115

When are the requirements complete?

- **System Requirements**
Whenever the system developers have enough documented information to be relatively certain of producing a product that will be a satisfactory design solution to the right problem.

8/11/06 Donald C. Gause 116

Business requirements checklist

- **BACKGROUND**
 - ✓ History leading to the recognized problem
 - ✓ Statement of the problem to be solved
 - ✓ Comparative analysis of potential solutions
- **PURPOSE OF THE SYSTEM TO BE DESIGNED**
 - ✓ How the system addresses the stated problem
 - ✓ Statement of system scope (in and out)
 - ✓ Primary design goals

8/11/06 Donald C. Gause 117

Business requirements checklist (continued)



- BUSINESS CASE
 - ✓ Important business benefits of the system
 - ✓ Critical business assumptions
 - ✓ Primary system features
 - ✓ Essential system functions
 - ✓ Essential resources and proposed schedule

8/11/06 Donald C. Gause 118

Business requirements checklist (continued)



- BUSINESS CASE (continued)
 - ✓ Estimated cost factors
 - ✓ Risk factors
 - ✓ Requirements ambiguity and uncertainty
 - ✓ Design and implementation oversights
 - ✓ System - malfunction, security breaches
 - ✓ Organizational - unintended consequences

8/11/06 Donald C. Gause 119

System requirements checklist



- SYSTEM USERS
 - ✓ Favored
 - ✓ Disfavored
 - ✓ Ignored
 - ✓ Frivolous

} USER GENERATED FEATURES

8/11/06 Donald C. Gause 120

System requirements checklist 

- ✓ SYSTEM ATTRIBUTES
 - ✓ Constrained
 - ✓ Optimized
 - ✓ Get if you can
 - ✓ Frivolous

} ATTRIBUTE GENERATED FEATURES

8/11/06 Donald C. Gause 121

System requirements checklist
(continued) 

- ✓ USE SCENARIOS
 - ✓ Expected
 - ✓ Highly unusual
 - ✓ Beyond reasonable use

} SCENARIO DERIVED FEATURES

8/11/06 Donald C. Gause 122

System requirements checklist
(continued) 

- ✓ TEST CASES
 - ✓ Normal and expected
 - ✓ Abnormal but possible
 - ✓ Won't happen

} TEST CASE GENERATED FUNCTIONS

8/11/06 Donald C. Gause 123

System requirements checklist
(continued) 

✓ FEATURES	✓ FUNCTIONS
✓ Essential	✓ Essential
✓ Expected	✓ evident
✓ Gee whiz	✓ hidden
✓ Delayed	✓ Desirable
✓ Discarded	✓ Discarded

8/11/06 Donald C. Gause 124

System requirements checklist
(continued) 

- ✓ DATA ELEMENTS
- ✓ DESIGN ASSUMPTIONS
- ✓ DESIGN DECISIONS
- ✓ USER EXPECTATIONS
- ✓ DESIGN LIMITATIONS
- ✓ ISSUES ARCHIVE
 - ✓ Open
 - ✓ Resolved

8/11/06 Donald C. Gause 125

System requirements checklist
(continued) 

- ✓ RISK FACTORS
 - ✓ Requirements risk
 - ✓ nature's last laugh, unrealized potential, surprise
 - ✓ Development risk
 - ✓ scope creep, poor requirements management
 - ✓ System risk (failures during operation)
 - ✓ user abuse, brittleness, natural disturbances
 - ✓ Organizational risk
 - ✓ unintended consequences

8/11/06 Donald C. Gause 126



