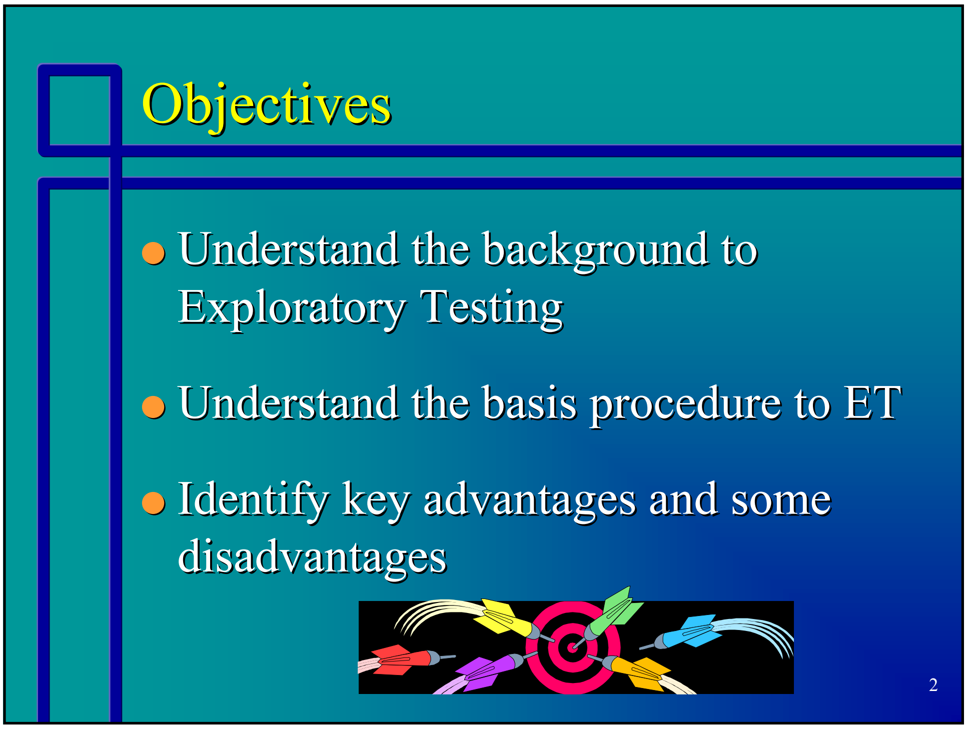
IMPROVE Quality Services

Exploratory Testing

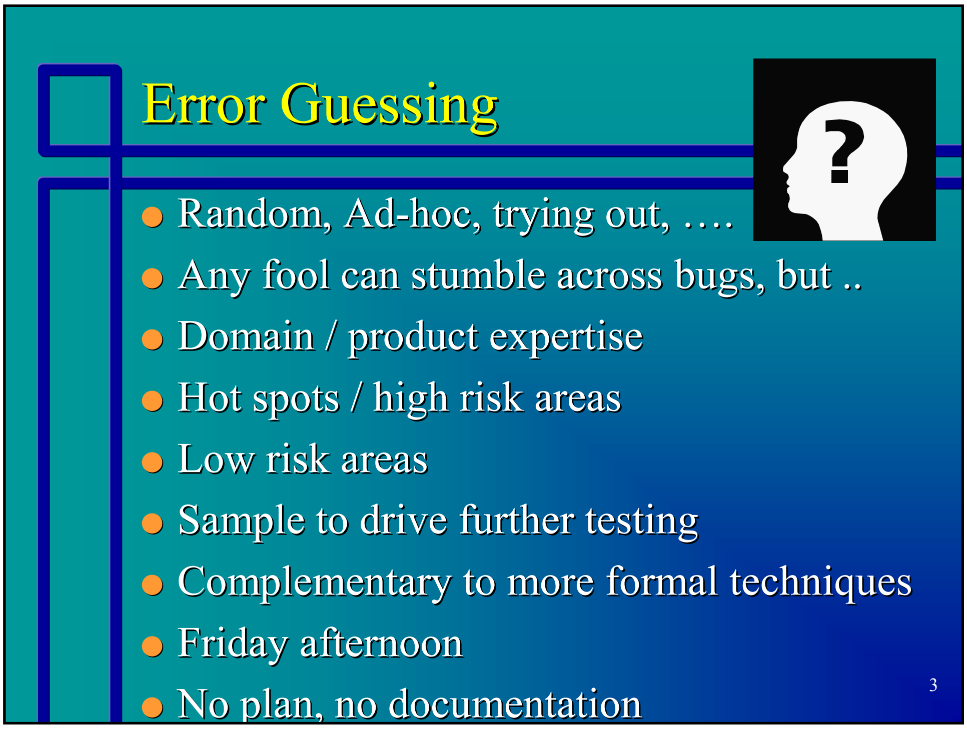
Erik van Veenendaal

eve@improveqs.nl



Objectives

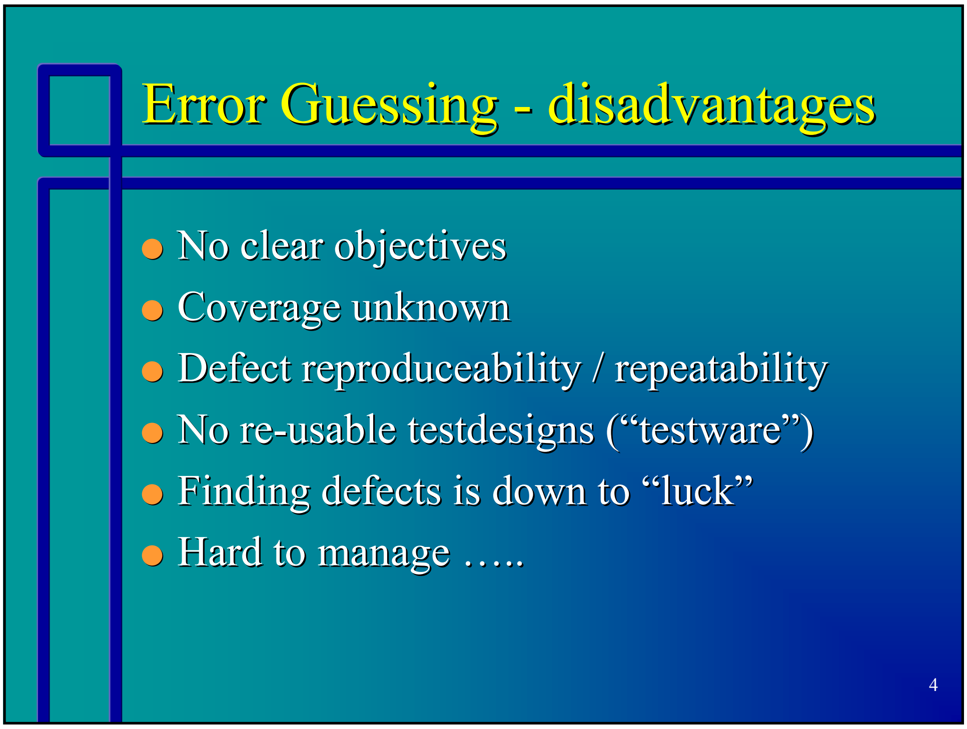
- Understand the background to Exploratory Testing
- Understand the basis procedure to ET
- Identify key advantages and some disadvantages



Error Guessing

- Random, Ad-hoc, trying out, ….
- Any fool can stumble across bugs, but ..
- Domain / product expertise
- Hot spots / high risk areas
- Low risk areas
- Sample to drive further testing
- Complementary to more formal techniques
- Friday afternoon
- No plan, no documentation

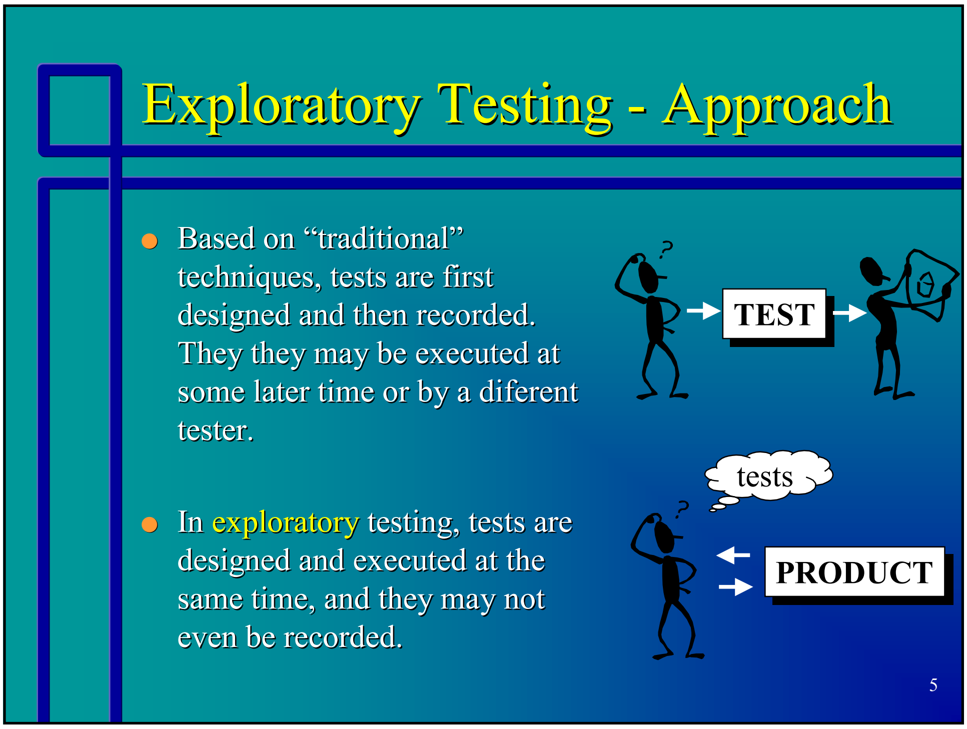
3



Error Guessing - disadvantages

- No clear objectives
- Coverage unknown
- Defect reproduceability / repeatability
- No re-usable testdesigns (“testware”)
- Finding defects is down to “luck”
- Hard to manage …..

4

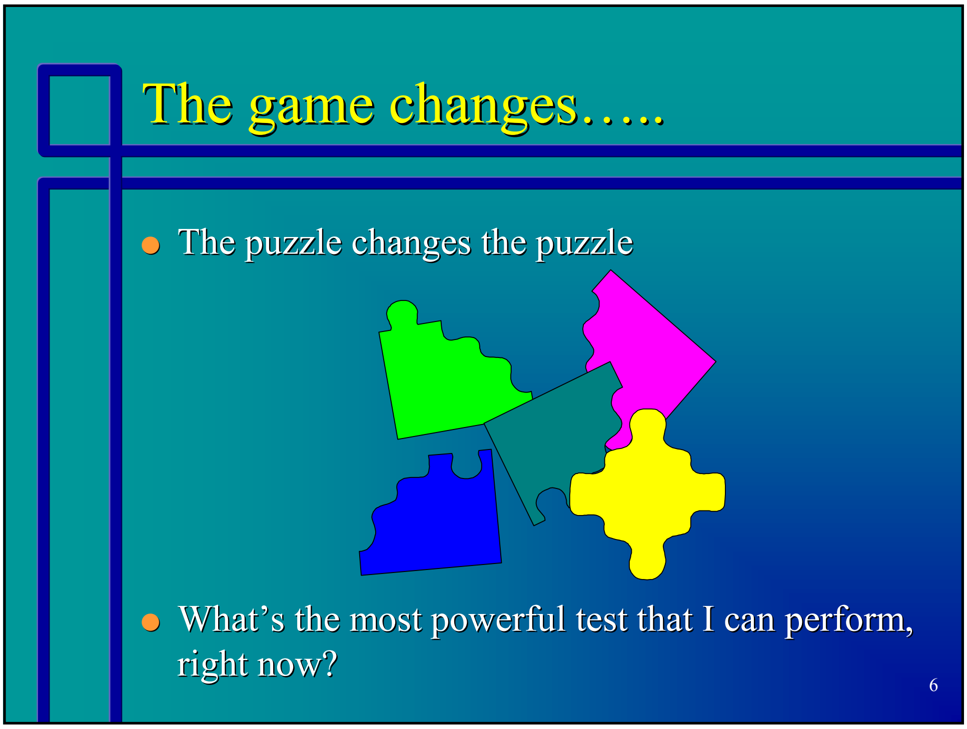


Exploratory Testing - Approach

- Based on “traditional” techniques, tests are first designed and then recorded. They they may be executed at some later time or by a diferent tester.
- In exploratory testing, tests are designed and executed at the same time, and they may not even be recorded.

?
TEST
tests
?
PRODUCT

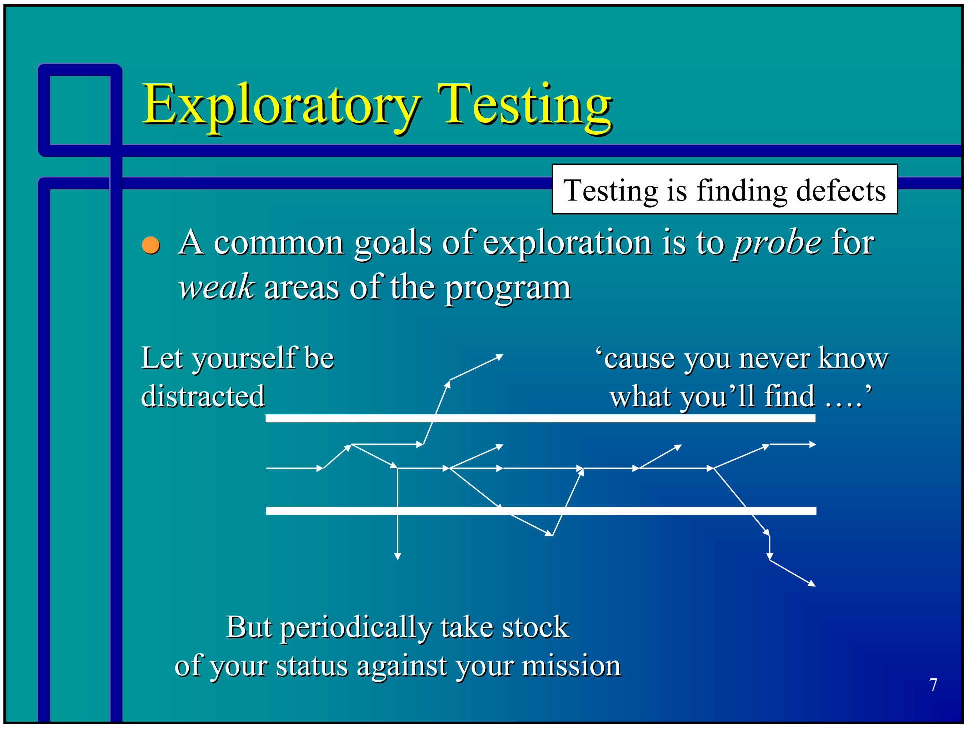
5



The game changes…..

- The puzzle changes the puzzle
- What’s the most powerful test that I can perform, right now?

6



Exploratory Testing

Testing is finding defects

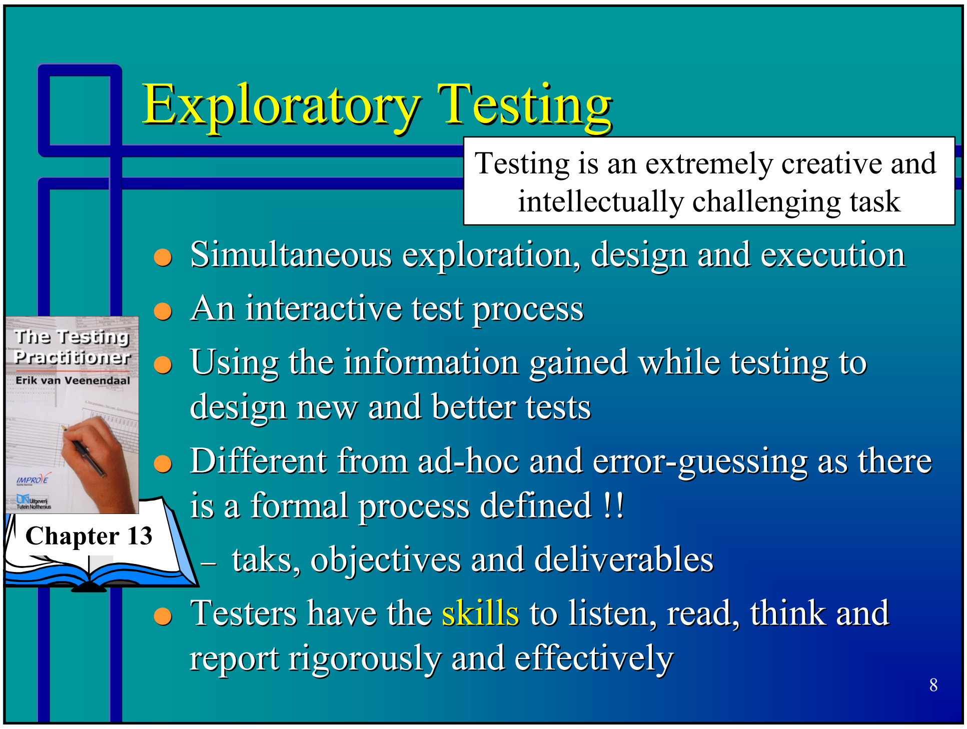
- A common goals of exploration is to *probe* for *weak* areas of the program

Let yourself be distracted

'cause you never know what you'll find ….'

But periodically take stock of your status against your mission

7

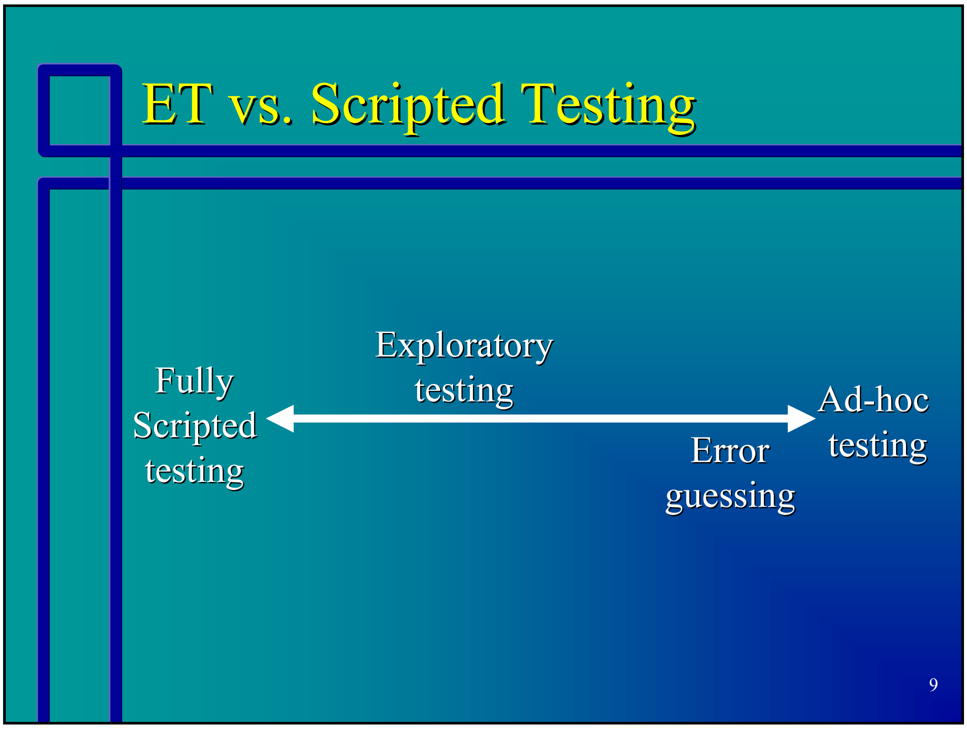


Exploratory Testing

Testing is an extremely creative and intellectually challenging task

- Simultaneous exploration, design and execution
- An interactive test process
- Using the information gained while testing to design new and better tests
- Different from ad-hoc and error-guessing as there is a formal process defined !!
  - taks, objectives and deliverables
- Testers have the skills to listen, read, think and report rigorously and effectively

8



ET vs. Scripted Testing

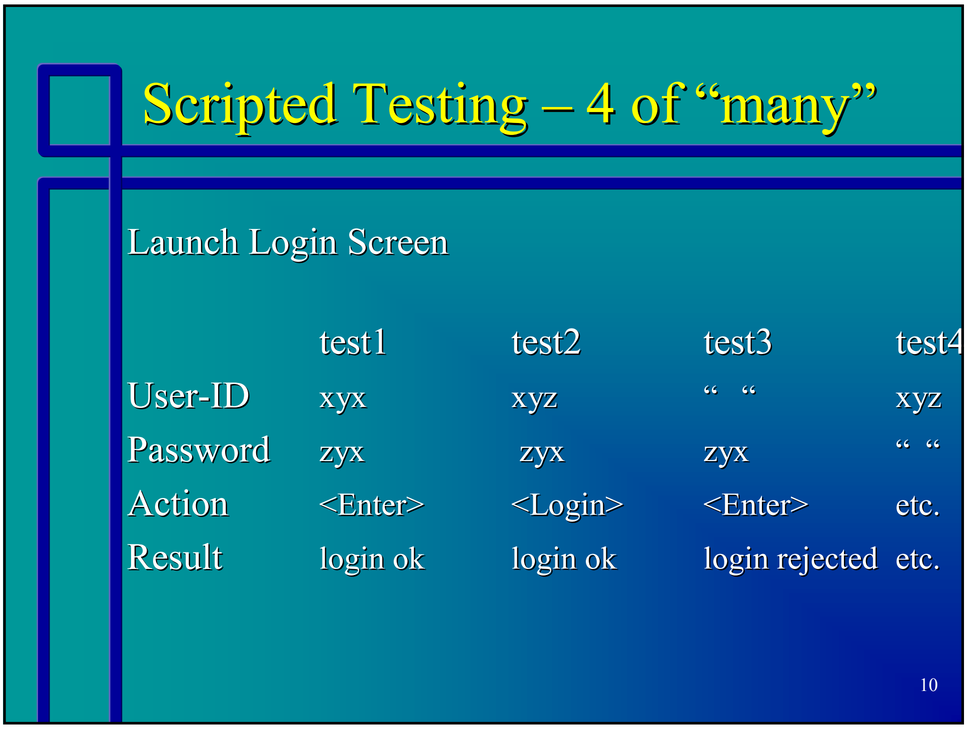
Fully Scripted testing

Exploratory testing

Error guessing

Ad-hoc testing

9

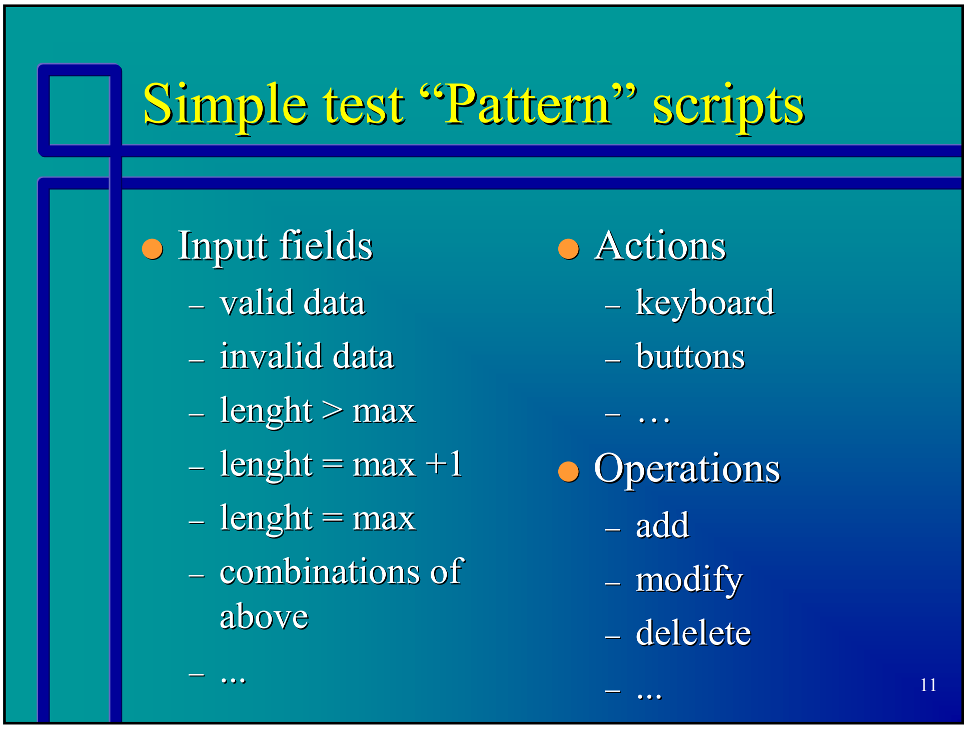


Scripted Testing – 4 of “many”

Launch Login Screen

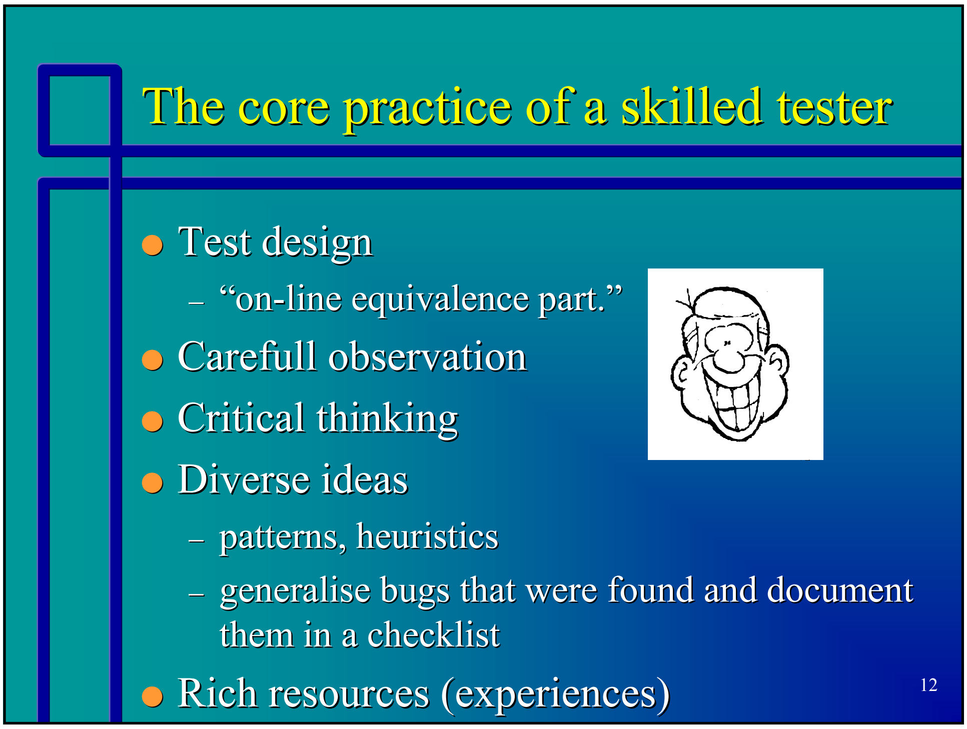
	test1	test2	test3	test4
User-ID	xyx	xyz	“ “	xyz
Password	zyx	zyx	zyx	“ “
Action	<Enter>	<Login>	<Enter>	etc.
Result	login ok	login ok	login rejected	etc.

10



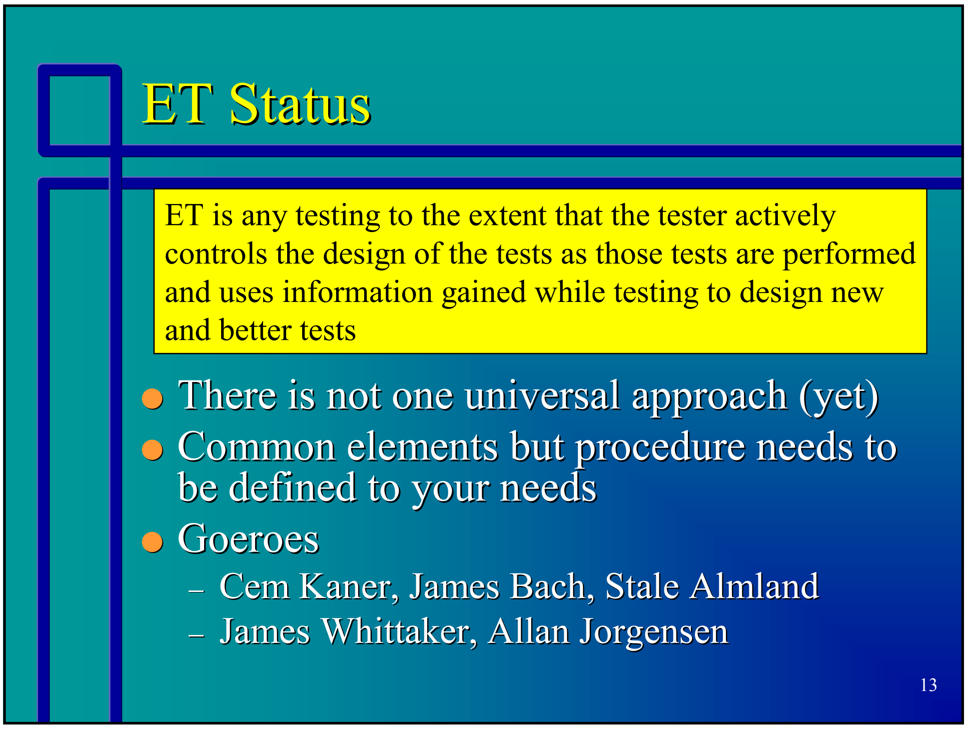
Simple test “Pattern” scripts

- Input fields
  - valid data
  - invalid data
  - lenght > max
  - lenght = max +1
  - lenght = max
  - combinations of above
  - ...
- Actions
  - keyboard
  - buttons
  - …
- Operations
  - add
  - modify
  - delelete
  - ...



The core practice of a skilled tester

- Test design
  - “on-line equivalence part.”
- Carefull observation
- Critical thinking
- Diverse ideas
  - patterns, heuristics
  - generalise bugs that were found and document them in a checklist
- Rich resources (experiences)

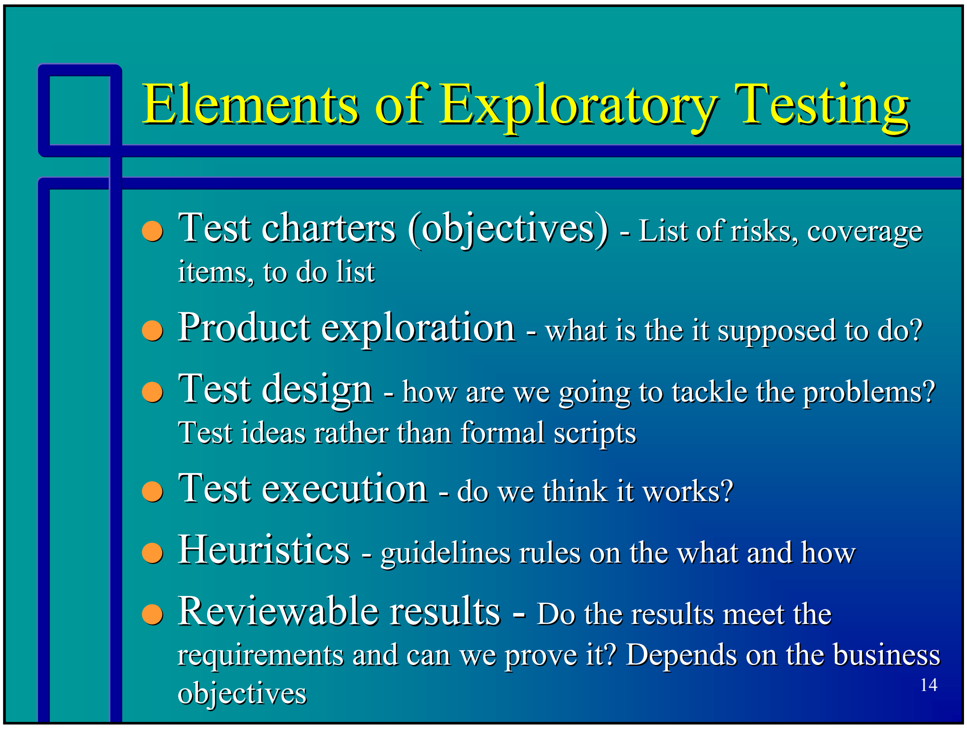


ET Status

ET is any testing to the extent that the tester actively controls the design of the tests as those tests are performed and uses information gained while testing to design new and better tests

- There is not one universal approach (yet)
- Common elements but procedure needs to be defined to your needs
- Goeroes
  - Cem Kaner, James Bach, Stale Almland
  - James Whittaker, Allan Jorgensen

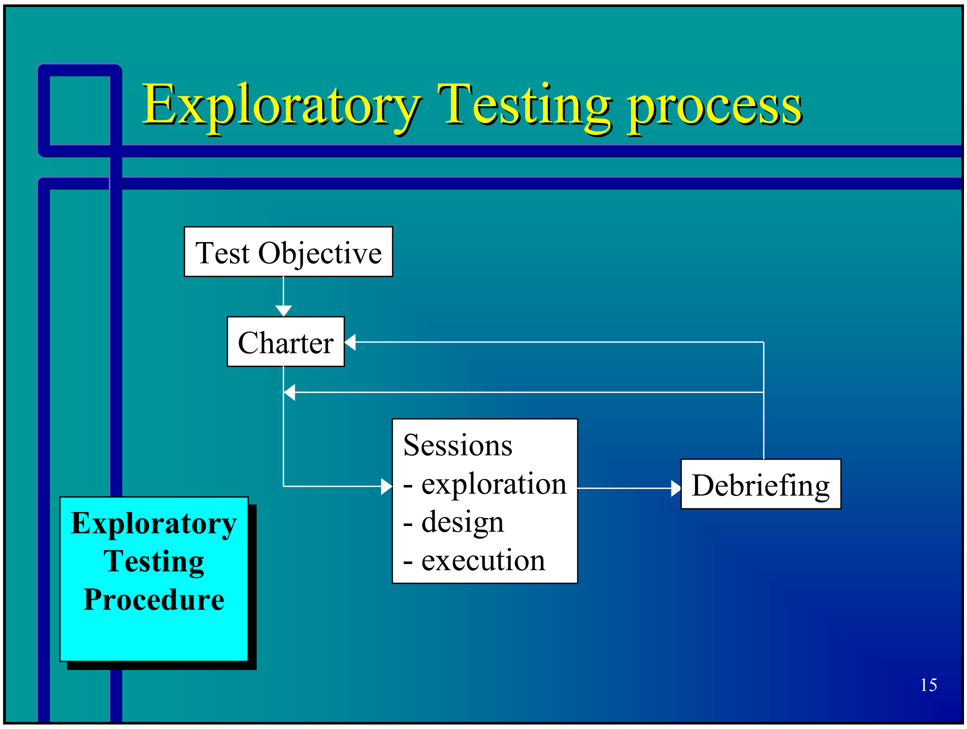
13



Elements of Exploratory Testing

- Test charters (objectives) - List of risks, coverage items, to do list
- Product exploration - what is the it supposed to do?
- Test design - how are we going to tackle the problems? Test ideas rather than formal scripts
- Test execution - do we think it works?
- Heuristics - guidelines rules on the what and how
- Reviewable results - Do the results meet the requirements and can we prove it? Depends on the business objectives

14



Exploratory Testing process

Test Objective

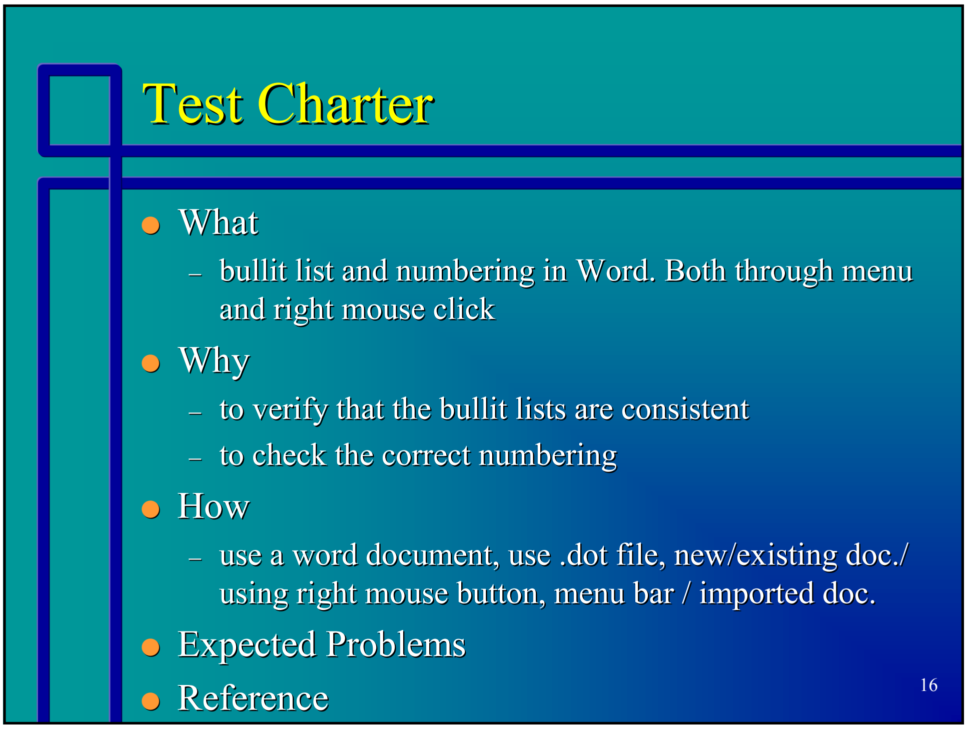
Charter

Sessions
- exploration
- design
- execution

Debriefing

Exploratory Testing Procedure

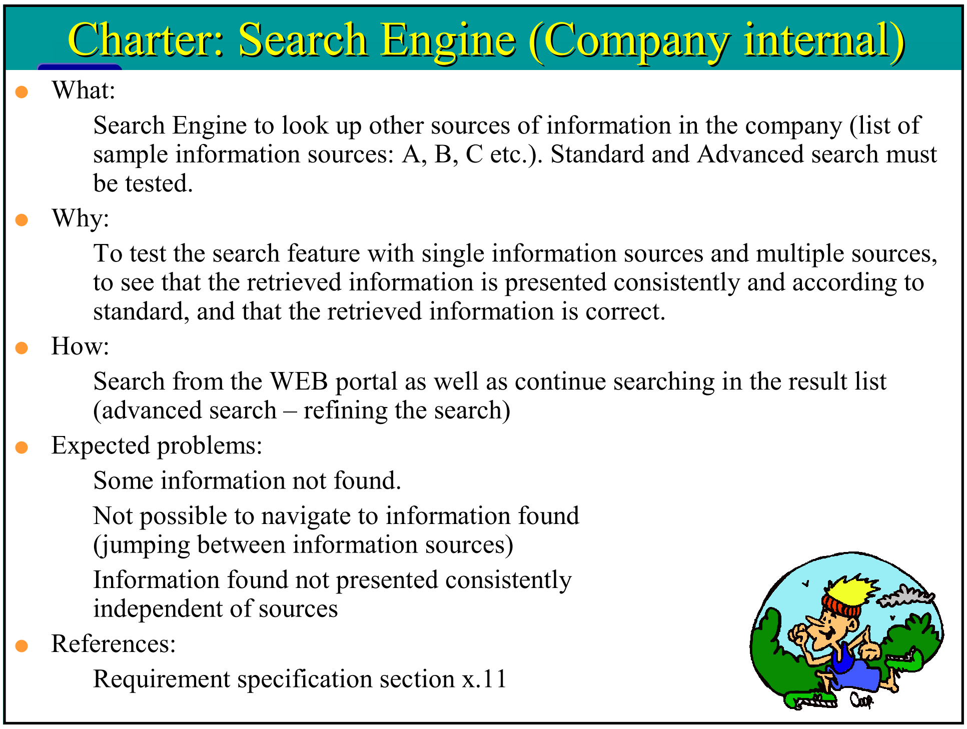
15



Test Charter

- What
  - bullit list and numbering in Word. Both through menu and right mouse click
- Why
  - to verify that the bullit lists are consistent
  - to check the correct numbering
- How
  - use a word document, use .dot file, new/existing doc./ using right mouse button, menu bar / imported doc.
- Expected Problems
- Reference

16



Charter: Search Engine (Company internal)

- What:

  Search Engine to look up other sources of information in the company (list of sample information sources: A, B, C etc.). Standard and Advanced search must be tested.
- Why:

  To test the search feature with single information sources and multiple sources, to see that the retrieved information is presented consistently and according to standard, and that the retrieved information is correct.
- How:

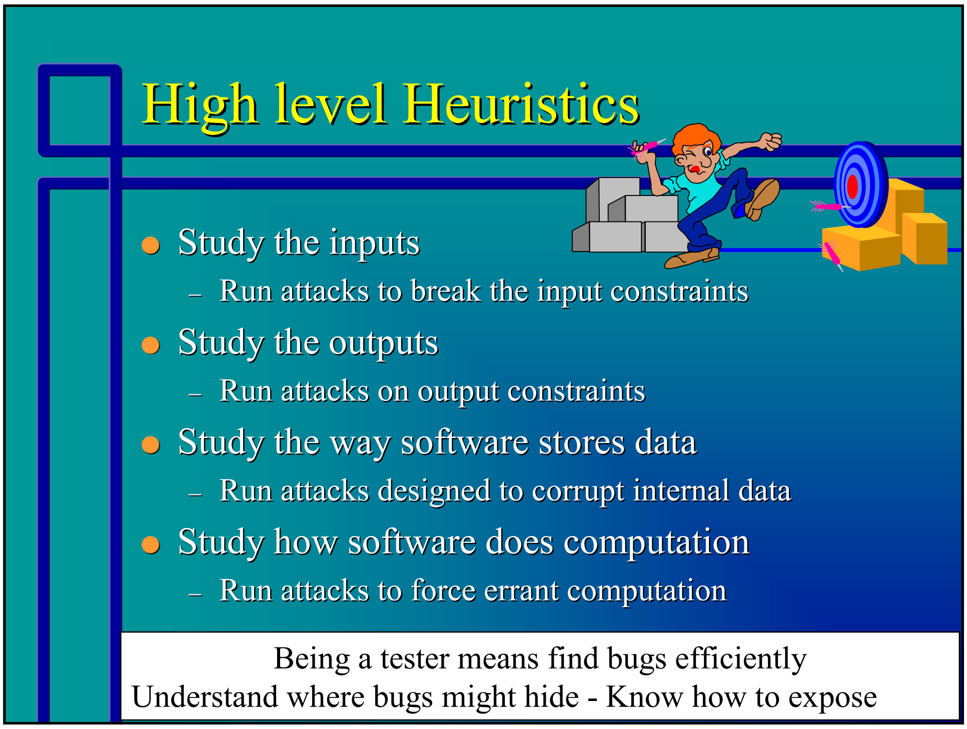
  Search from the WEB portal as well as continue searching in the result list (advanced search – refining the search)
- Expected problems:

  Some information not found.

  Not possible to navigate to information found (jumping between information sources)

  Information found not presented consistently independent of sources
- References:

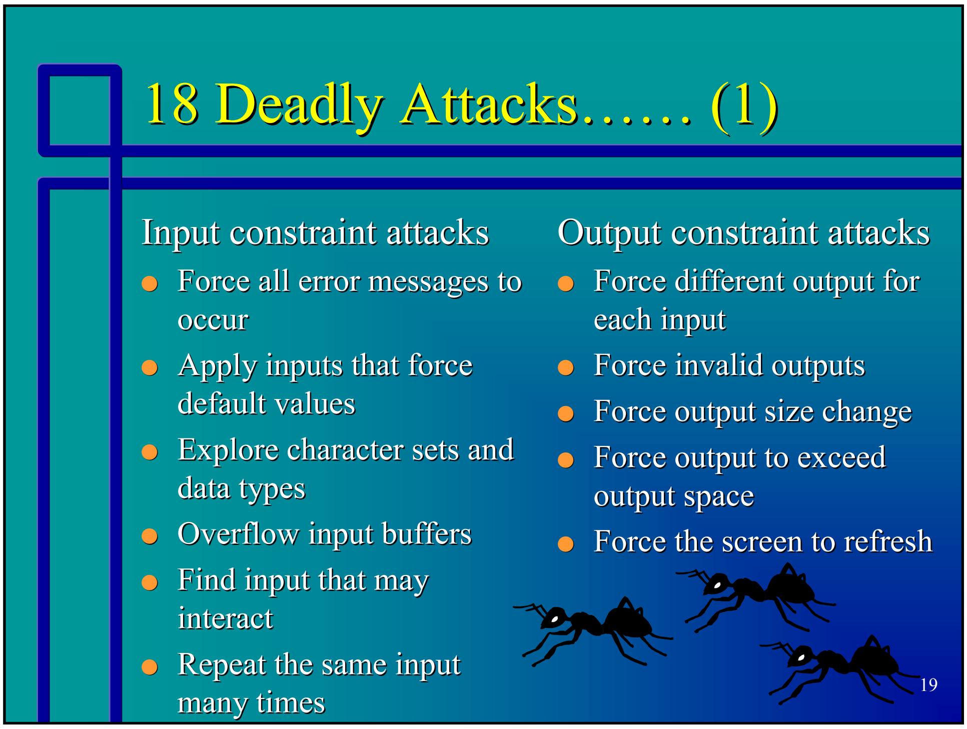
  Requirement specification section x.11



High level Heuristics

- Study the inputs
  - Run attacks to break the input constraints
- Study the outputs
  - Run attacks on output constraints
- Study the way software stores data
  - Run attacks designed to corrupt internal data
- Study how software does computation
  - Run attacks to force errant computation

Being a tester means find bugs efficiently
Understand where bugs might hide - Know how to expose



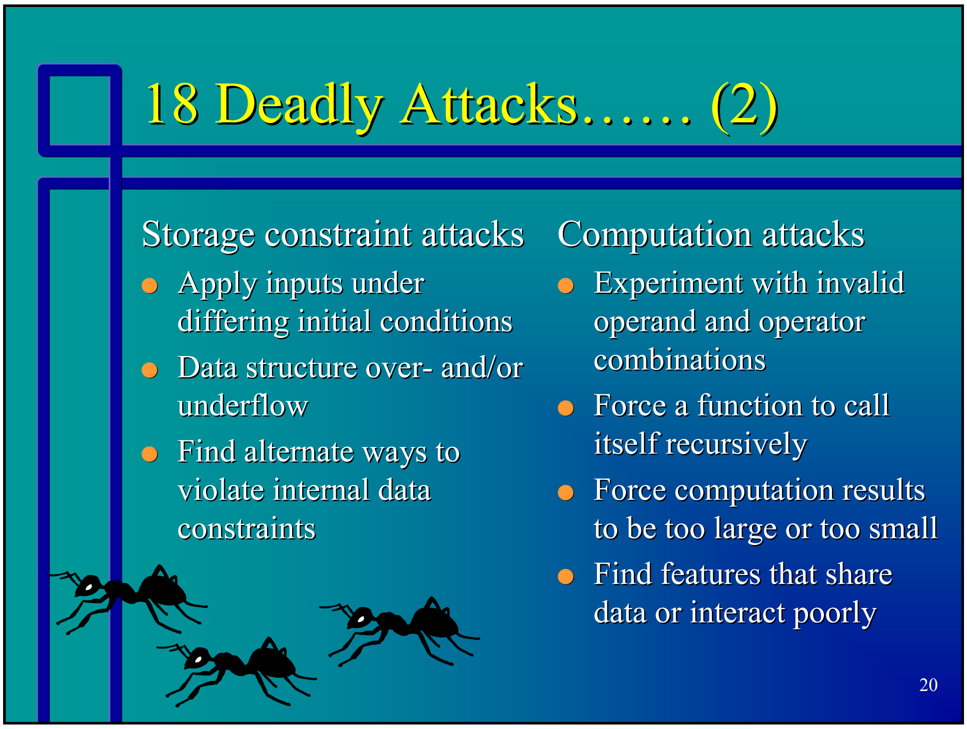
18 Deadly Attacks…… (1)

Input constraint attacks

- Force all error messages to occur
- Apply inputs that force default values
- Explore character sets and data types
- Overflow input buffers
- Find input that may interact
- Repeat the same input many times

Output constraint attacks

- Force different output for each input
- Force invalid outputs
- Force output size change
- Force output to exceed output space
- Force the screen to refresh



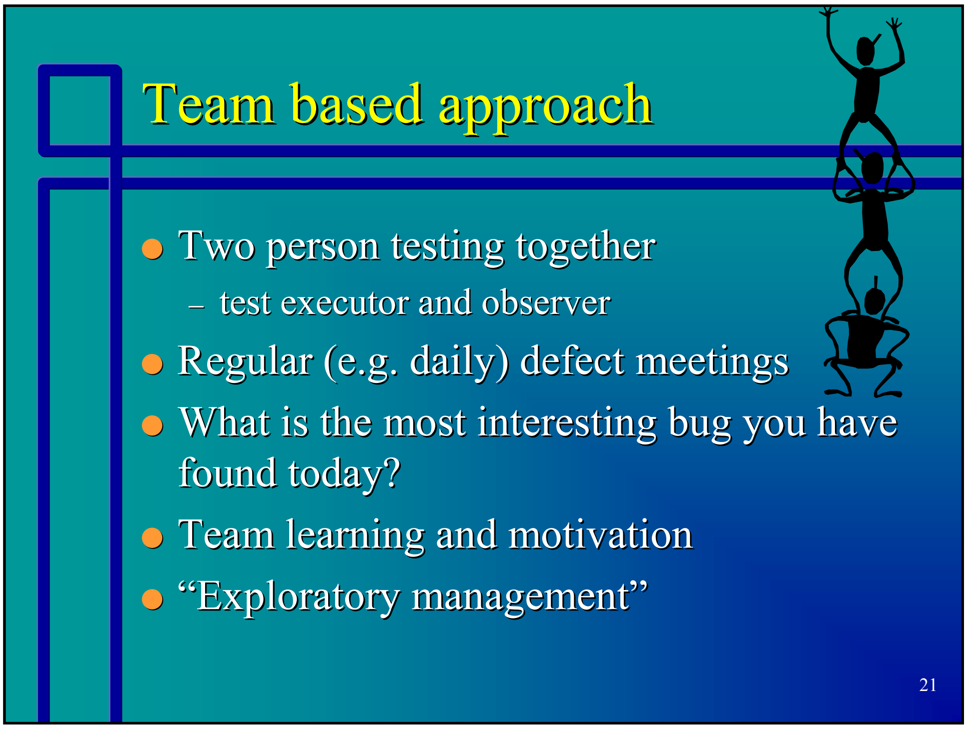
18 Deadly Attacks…… (2)

Storage constraint attacks

- Apply inputs under differing initial conditions
- Data structure over- and/or underflow
- Find alternate ways to violate internal data constraints

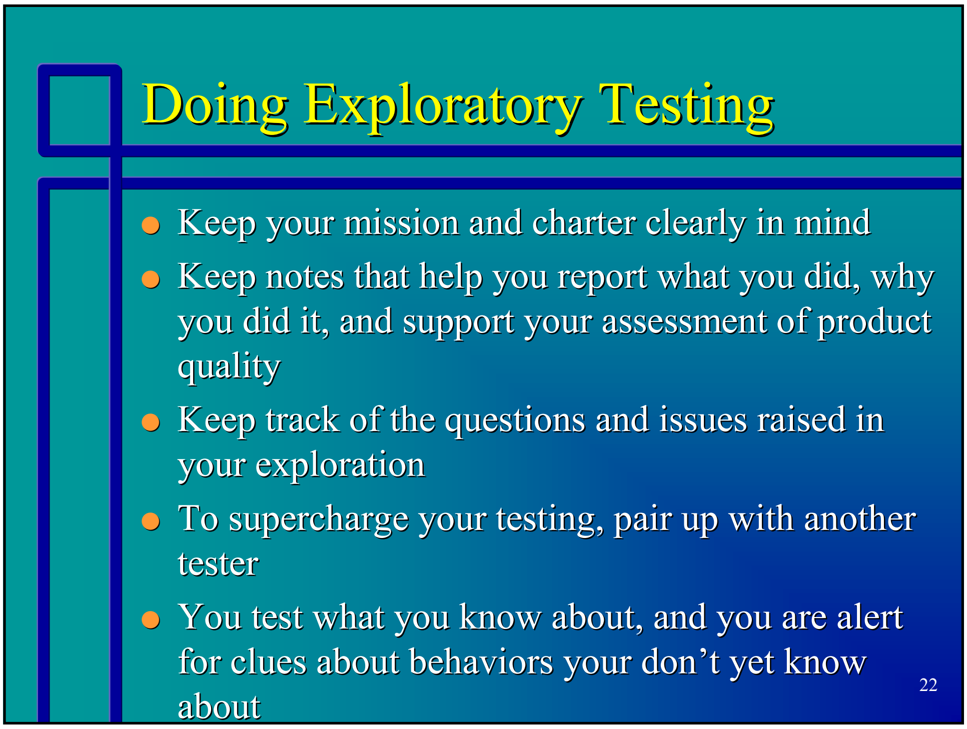
Computation attacks

- Experiment with invalid operand and operator combinations
- Force a function to call itself recursively
- Force computation results to be too large or too small
- Find features that share data or interact poorly



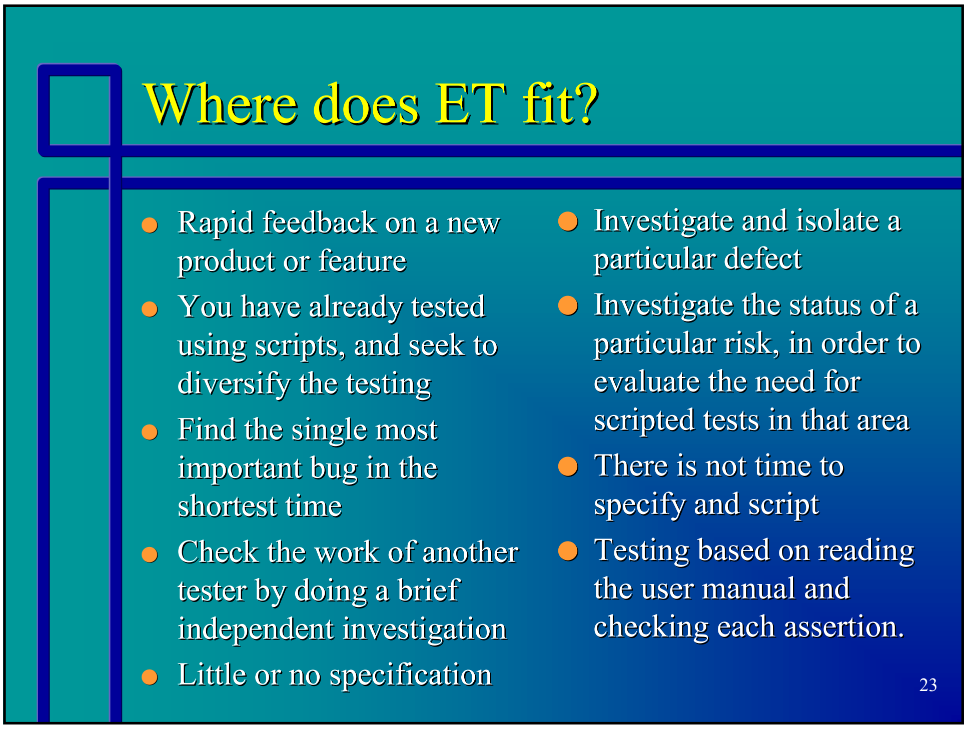
Team based approach

- Two person testing together
  - test executor and observer
- Regular (e.g. daily) defect meetings
- What is the most interesting bug you have found today?
- Team learning and motivation
- “Exploratory management”



Doing Exploratory Testing

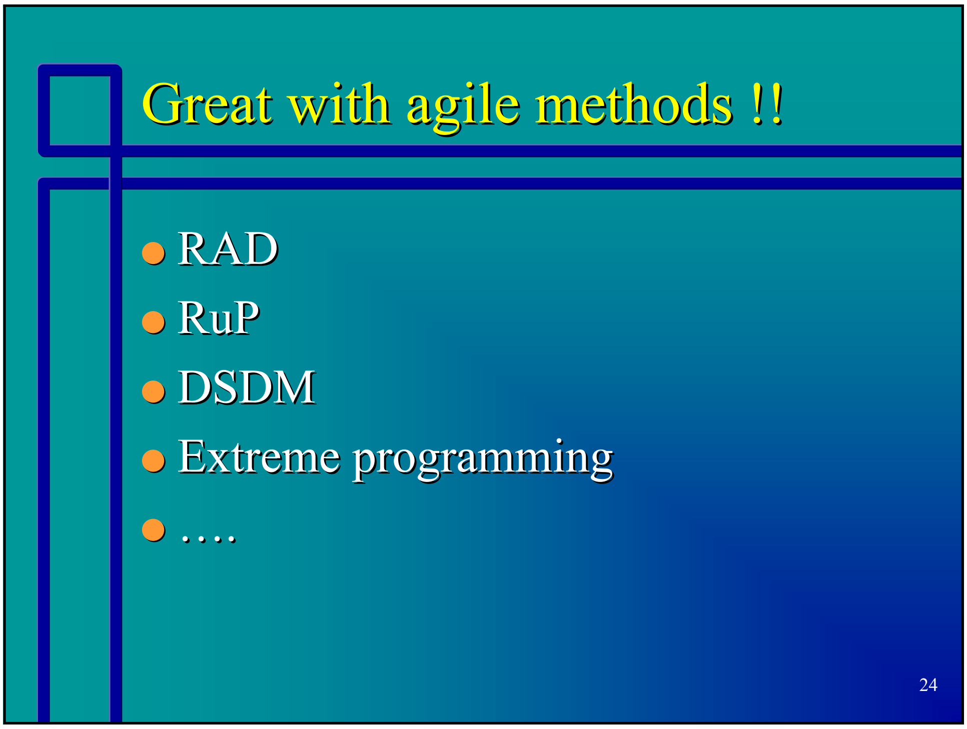
- Keep your mission and charter clearly in mind
- Keep notes that help you report what you did, why you did it, and support your assessment of product quality
- Keep track of the questions and issues raised in your exploration
- To supercharge your testing, pair up with another tester
- You test what you know about, and you are alert for clues about behaviors your don’t yet know about



Where does ET fit?

- Rapid feedback on a new product or feature
- You have already tested using scripts, and seek to diversify the testing
- Find the single most important bug in the shortest time
- Check the work of another tester by doing a brief independent investigation
- Little or no specification
- Investigate and isolate a particular defect
- Investigate the status of a particular risk, in order to evaluate the need for scripted tests in that area
- There is not time to specify and script
- Testing based on reading the user manual and checking each assertion.

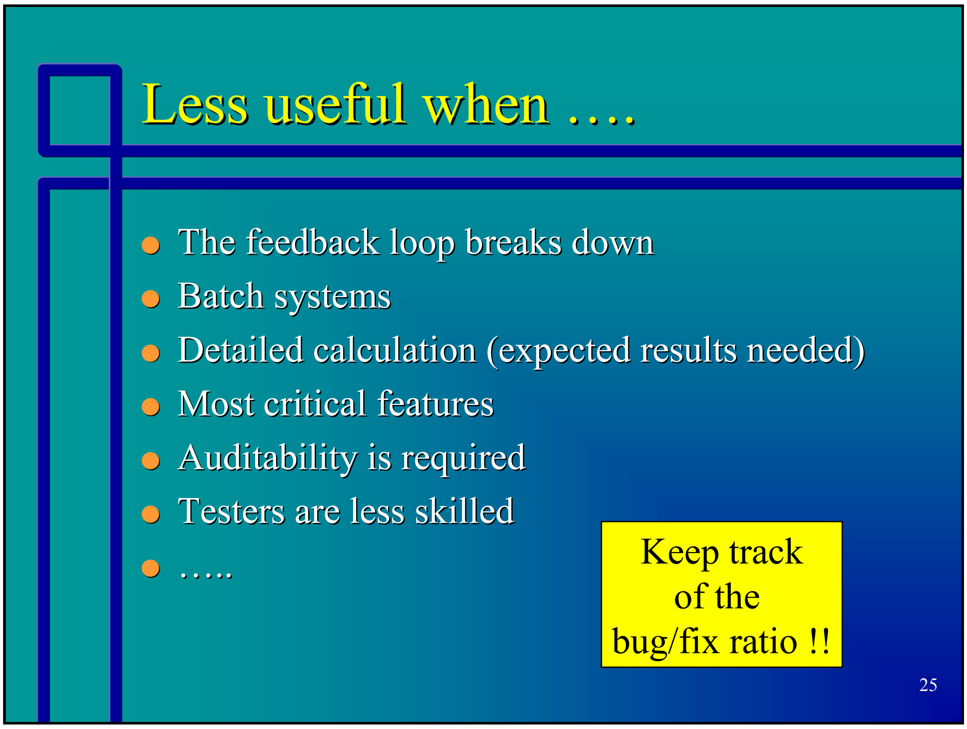
23



Great with agile methods !!

- RAD
- RuP
- DSDM
- Extreme programming
- ….

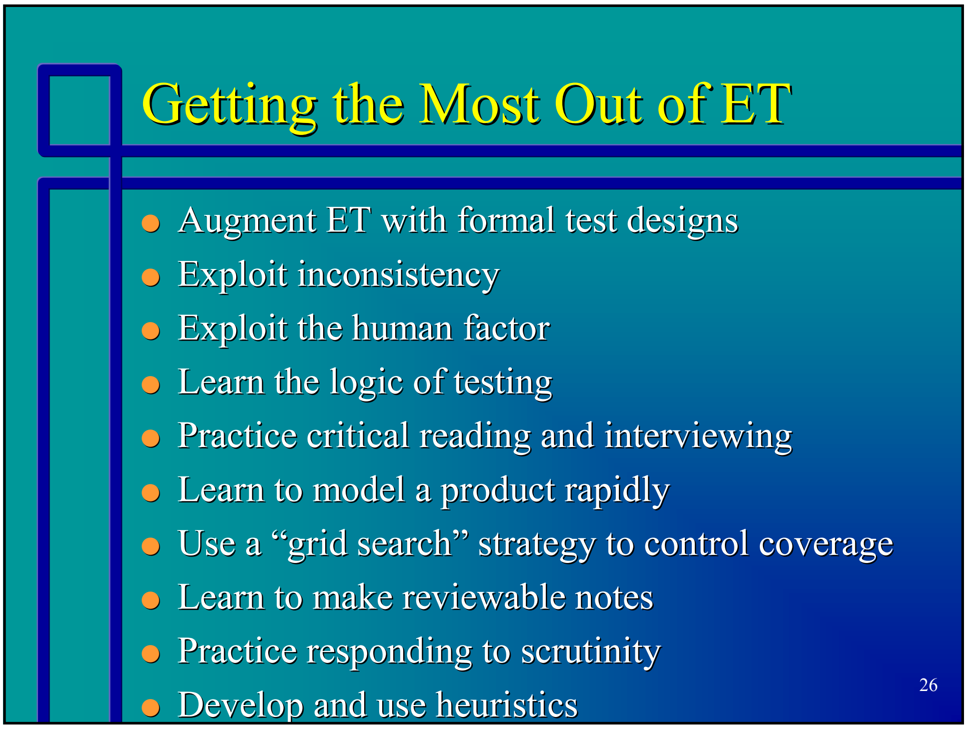
24



Less useful when ….

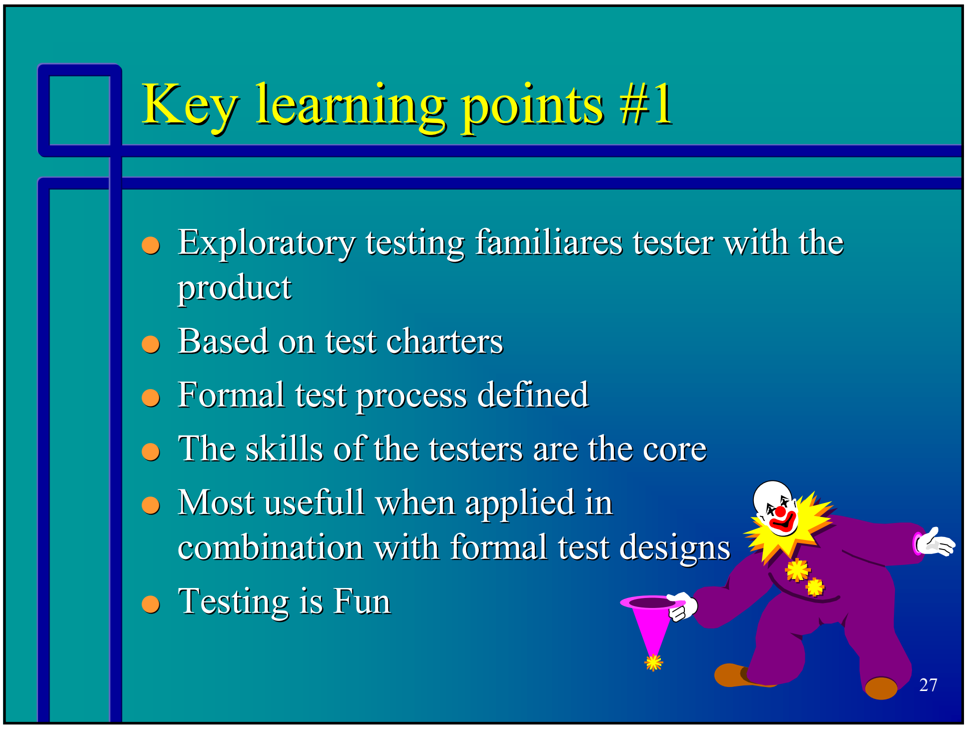
- The feedback loop breaks down
- Batch systems
- Detailed calculation (expected results needed)
- Most critical features
- Auditability is required
- Testers are less skilled
- …..

Keep track of the bug/fix ratio !!



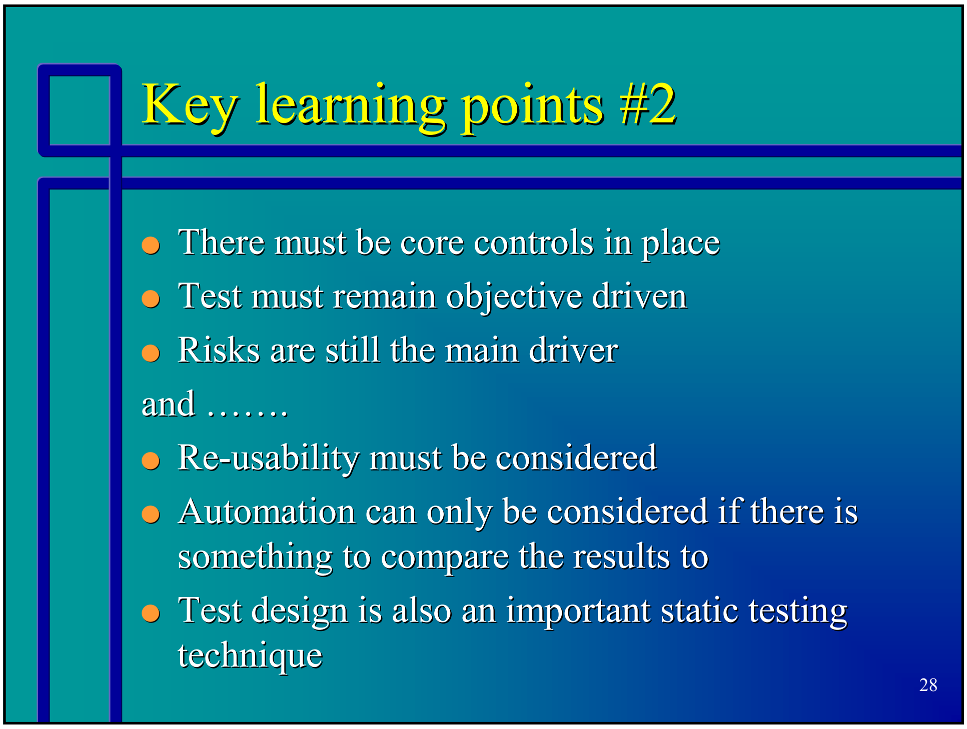
Getting the Most Out of ET

- Augment ET with formal test designs
- Exploit inconsistency
- Exploit the human factor
- Learn the logic of testing
- Practice critical reading and interviewing
- Learn to model a product rapidly
- Use a “grid search” strategy to control coverage
- Learn to make reviewable notes
- Practice responding to scrutinity
- Develop and use heuristics



Key learning points #1

- Exploratory testing familiares tester with the product
- Based on test charters
- Formal test process defined
- The skills of the testers are the core
- Most usefull when applied in combination with formal test designs
- Testing is Fun

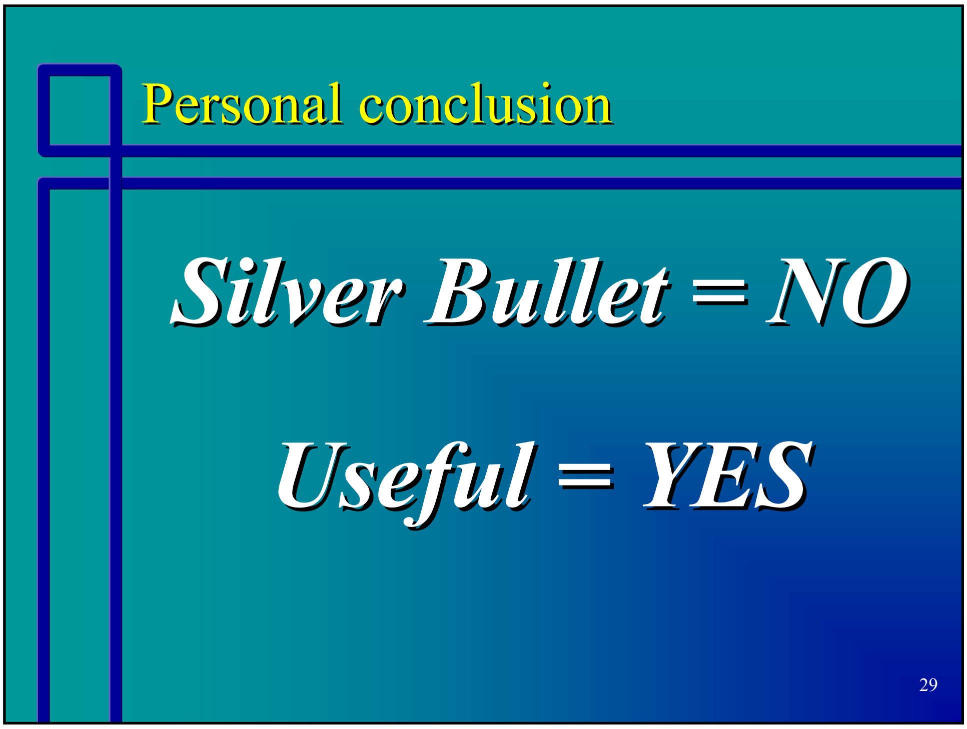


Key learning points #2

- There must be core controls in place
- Test must remain objective driven
- Risks are still the main driver

and …….

- Re-usability must be considered
- Automation can only be considered if there is something to compare the results to
- Test design is also an important static testing technique



Personal conclusion

Silver Bullet = NO

Useful = YES

29