

[Presentation](#)

P R E S E N T A T I O N

[Bio](#)

[Conference-at-a-Glance](#)

[Return to Main Menu](#)

# FG1

Friday, November 2, 2001  
9:00 AM

---

## BUG HUNTING: GOING ON A SOFTWARE SAFARI

---

Elisabeth Hendrickson  
Quality Tree Software, Inc.

International Conference On  
Software Testing Analysis & Review  
October 29-November 2, 2001  
San Jose, CA USA



# Bug Hunting: Going on Software Safari

Elisabeth Hendrickson  
STARWest 2001

# Welcome to Bug Hunting Adventures



On today's tour:

- Install
- Defaults
- Uninstall
- Extreme Input
- Data
- Interactions
- Memory
- Time

# Make Sure We Packed Our Gear



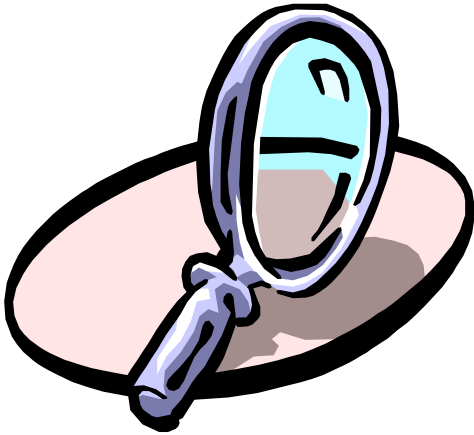
- Disk Imaging
- Change Detectors
- Dependency Checkers
- Data Manipulation
- System (Memory, CPU, Resource) Monitors

# Start on the Right Foot



- Start with a clean system
- Create an image so you can get back to a clean system easily
- Make sure you have the correct version
- Scan through the known bugs both to make sure you're hunting for new species and get ideas

# Bug Hunting Tool: Disk Imaging



- Ensures the system is in a known state:
  - Saves the entire contents of a partition to a file
  - Restores partitions from saved image files
- Several choices:
  - Ghost, <http://www.ghostsoft.com>
  - PQDI, <http://www.powerquest.com>
  - Image Cast, <http://www.storagesoft.com/ic3>

# Installation Woes I

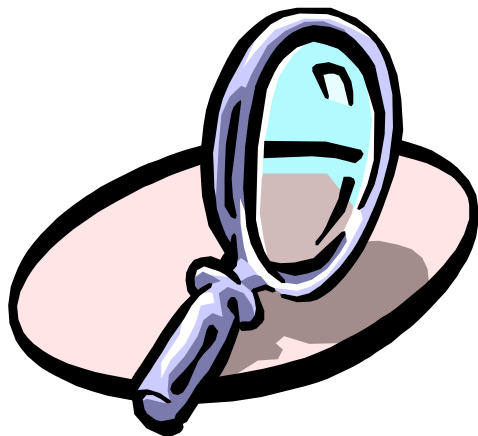
## Common Problems

- Needed files not installed or only installed on some OSes
- Incomplete install when not an admin
- Upgrade obliterates user preferences

## Find By

- Use tools to report dependencies and files installed
- Look for existing files
- Be a “normal” or low-permissions user
- Change all options; check before & after

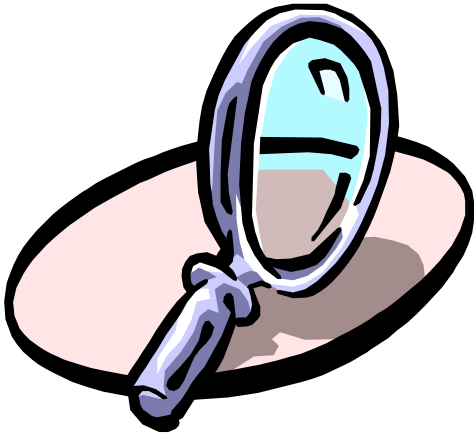
# Bug Hunting Tool: Dependency Walker



- Displays a hierarchical tree of dependencies for a DLL or exe
- Discover dependencies on similar libraries or on optional libraries not on all systems
- Freeware
- Available online at:  
<http://www.dependencywalker.com>



# Bug Hunting Tool: InCtrl5



- Catalogs the file system and registry entries on Windows
- Can report what changed as a result of an installation
- Can also report what changed between two points in time
- Available online at:  
<http://www.zdnet.com/downloads>  
search for “InCtrl5”

# Installation Woes II

## *(Location, Location, Location)*

### Common Problems

- Doesn't support UNC paths like \\server\vol
- Tries to install in low disk space conditions
- Misleading error on full or read-only drive

### Find By

- Install to a remote drive with UNC
- Set disk space just below threshold
- Install on full disk & CD drive

# Defaults (and Defailures)

## Common Problems

- Only “works” if installed to default path or drive
- Only default user id “works”

## Find By

- Look for evidence of hard-coded paths
- Use InCtrl to watch where files are created
- Always test with non-default user id

# Uninstallation

## Common Problems

- Can't uninstall
- Incomplete uninstall leaves files behind
- Uninstall deletes needed files or registry keys

## Find By

- Always try to uninstall after other testing
- Use InCtrl to look for installed files left behind
- Use InCtrl to look for files or reg keys inappropriately removed

# Extreme Input

## Common Problems

- Crashes and other bad behavior with long strings
- Software misinterprets special characters as delimiters
- Integer fields can't handle big numbers

## Find By

- Paste large amounts of data into text fields, esp. comment fields
- Use “”,|&;:., etc.
- Copy & paste text with line breaks.
- Check boundaries at  $2^{15}=32768$ ,  $2^{16}=65536$ , and  $2^{31}=2147483648$

# Data

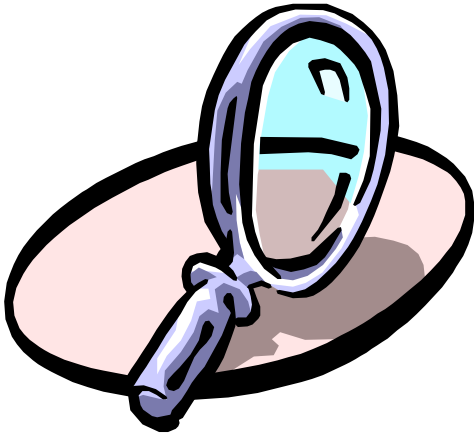
## Common Problems

- Fails to run if private files missing/damaged
- Crash on reading corrupted data
- Same data, two places, out-of-synch data
- Failure on write to read-only file

## Find By

- Find private files, then delete or munge
- Manually corrupt data
- Cause data to differ
- Make all files read only

# Bug Hunting Tools: Data Manipulation



Access data directly:

- Notepad
- SQL

Write programs:

- Use automation to create records or files with extreme input
- Iterate through all characters in a file randomly changing 1 out of 1000 or so characters

# Interactions:

## Drag and Drop

### Common Problems

- Doesn't handle multiples
- Opens things it shouldn't
- Drag and drop between programs or documents causes bad behavior
- Does not prevent dropping content into inappropriate areas
- Crash on content drop to “gray zone”

### Find By

- Select multiple docs
- Use many kinds of docs
- Drag and drop text, pictures, etc. from a variety of programs
- Drop content into illogical areas: pictures into text only areas, etc.
- Drop content to “illegal” areas of the software under test.



# Memory

## Common Problems

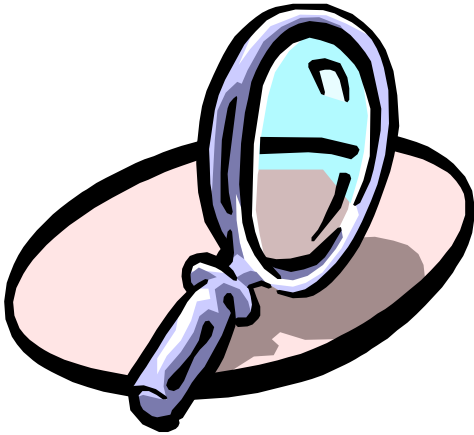
- Bad behavior in low memory conditions
- Memory leaks

## Find By

- Reduce virtual memory, run on low-memory machines, and run lots of apps
- Use tools to watch the amount of memory in use

# Bug Hunting Tools:

## Memory Monitors



- NT-based systems have memory monitors built in
- For other OSes, use memory monitoring utilities such as FreeMeter. See:  
<http://www.tiler.com/FreeMeter>

# Time

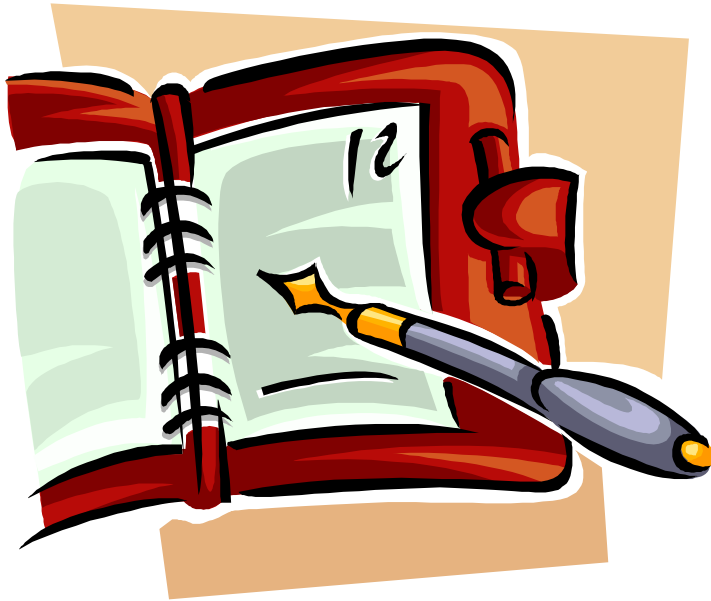
## Common Problems

- Bad behavior after long idle periods
- Slow response leads to timeouts and lost data

## Find By

- Leave the software in various states for long periods of time
- Check if response time under load exceeds timeout

# Keep a Bug Hunting Journal



Write down:

- Ideas
- Experiments and results
- Clues to anomalous behavior
- Bug behavior
- Bug numbers

# Swap Tales with Other Bug Hunters

- Share ideas
- Discuss anomalies
- Discover if a minor bug in one area could be a symptom of bigger problems



# Acknowledgements & References

- Additional tools and utilities can be found at:
  - <http://www.zdnet.com/pcmag/utilities>
  - <http://www.download.cnet.com>
- At LAWST 9 we discussed how you can learn just enough about the internals of the system to allow you to do better testing. I am grateful to the participants for their insights: Payson Hall, Richard Bender, Jaya Carl, Jack Falk, Ibrahim El-Far, James Whitaker, Bret Pettichord, Brian Marick, Doug Hoffman, Hung Nguyen, Mark Johnson, Cem Kaner, BJ Rollison, Noel Nyman, III, Brian Lawrence, Elisabeth Hendrickson
- At LAWST 10 we discussed Testers' Notebooks. I am grateful to the participants for their insights: James Bach, III, Brian Lawrence, Melora Svoboda, Cem Kaner, Karen Johnson, Doug Hoffman, Michael Snyder, Elisabeth Hendrickson, Noel Nyman, Mark Johnson, Brian Marick, Neal Reizer, Rocky Grober, Bret Pettichord, Marge Farrell, Jaya Carl, Chris Agruss

**Elisabeth Hendrickson**

Elisabeth Hendrickson ([esh@qualitytree.com](mailto:esh@qualitytree.com)) is an independent consultant who specializes in making software better. Much of her work involves testing, whether it's consulting on test strategy, assessing current practices, automating tests, communicating testability requirements, or teaching testers how to be better bug hunters. With over a dozen years of experience in the software field, Elisabeth has seen numerous species of software bugs.

Of course, testing is just one part of building better software. Elisabeth also works with organizations to ensure that they're building the right thing and building it right. An award winning author, Elisabeth has over 20 published articles and is a frequent invited speaker at major software quality and software management conferences. You can read more about her ideas on quality and testing at [www.qualitytree.com](http://www.qualitytree.com).