Debugging .NET and Native Applications in the Field

Gad J. Meir

IDAG Ltd.

Bug Exterminator & Process Plumber

EBlog:weblogs.asp.net/gadim
HBlog:blogs.microsoft.co.il/blogs/gadim
Email: gadim@idag.co.il, Site: www.idag.co.il

2011-11-29: Bonn-to-Code.Net: User-Treffen November

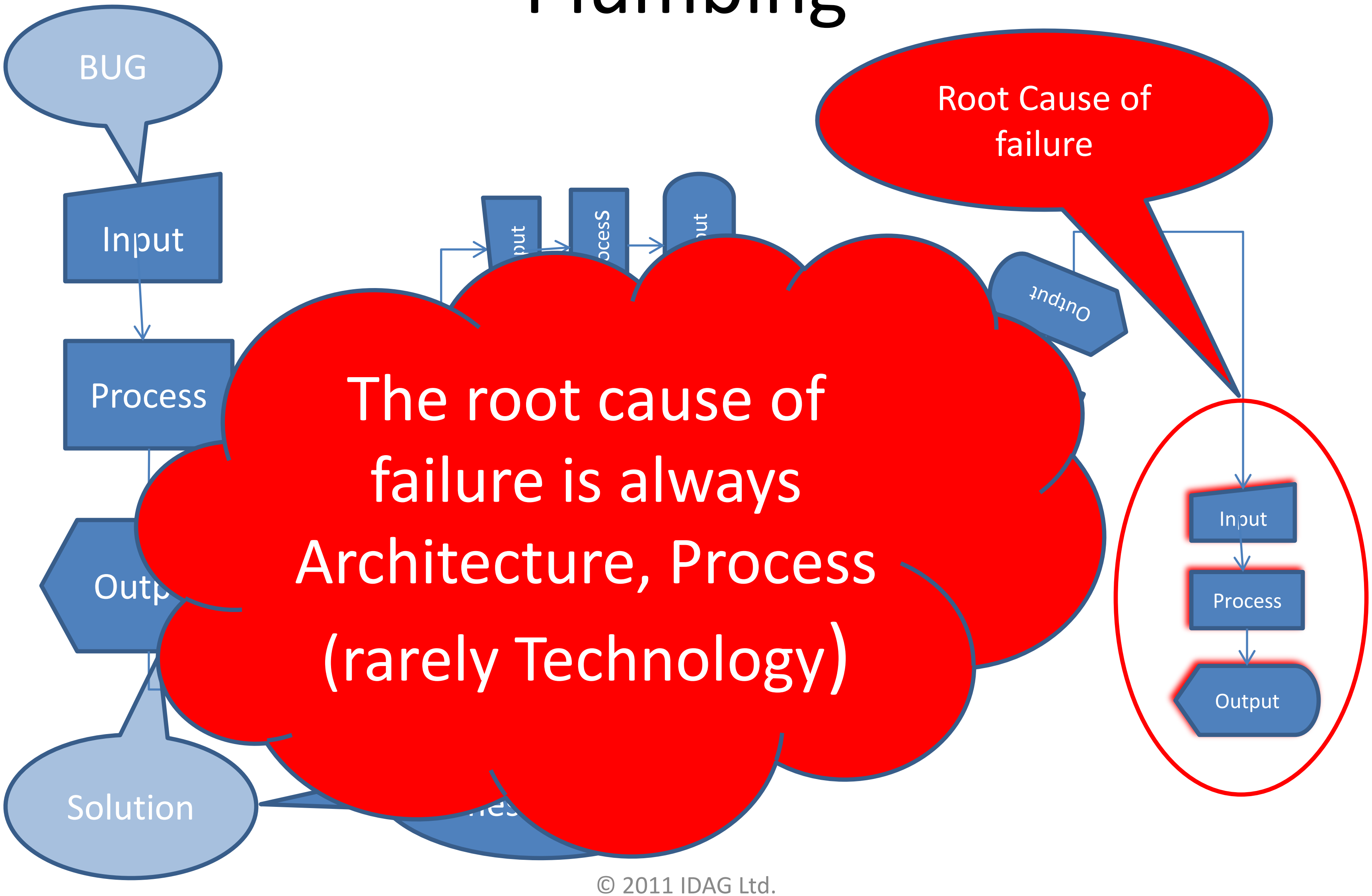
About Gad J. Meir

- Experience: Since 1975
- Work: www.idag.co.il
- Function: www.productiondebugging.com
- Blog: http://weblogs.asp.net/gadim/default.aspx
- MSF Certified Trainer & Practitioner
- BSc. Computer engineering Technion
- Microsoft Certified MC…

About IDAG Ltd.

- Founded 1983
- Established the first Microsoft certified training center in Israel at 1992.
- Areas of operation
  - Troubleshooting systems and procedures
  - Production time debugging to root cause of failure
  - Projects monitoring and guidance
  - Knowledge gaps detection and filling
  - Technologies and methodologies deployment

From Bug Extermination to Process Plumbing



The root cause of failure is always Architecture, Process (rarely Technology)

I Have a Question 1/4

- Are you a
  - Developer?
  - Test/QA?
  - IT?
  - Management?
  - Other?

I Have a Question 2/4

- Main Target Operating System
  - XP?
  - Vista?
  - Windows 7?
  - Server 2003?
  - Server 2008?
  - 2008 R2?
  - Other?

I Have a Question 3/4

- Bit
  - 32?
  - 64?
  - Other?

I Have a Question 4/4

- Run Time Environment
  - Managed (.NET)?
  - Native?
  - Other?

Talk Targets

- Explain some of the specific constrains of production environment / Field
- Introduce ways to get debug data from production environment with minimum disruption to the System / Users
- Several scenario Demos for Native and Manage code
- Tips

Prerequisites

- Experience in debugging

Agenda

- Theoretical background (Quantum physics)
- What is a production environment
- Dumping bodies (AdPlus)
- Mapping the bodies (Symbols)
- Autopsying and analyzing bodies (WinDbg)
- The problem with the .NET way of handling bodies
- Tools for extracting information from .NET bodies (SOS)
- Things you can’t get from a dead body
- Working with live bodies (Live Debugging)
- IIS (Debug Diag)
- Q & A

Please !

- If you don’t understand what I am talking about, Stop me and ASK !!!, Don’t wait.

Gad's Guidelines

- Nothing in life is certain
- If you measure it, it will be wrong
- Any action has at least one unexpected reaction
- Debugging application with Visual Studio, on a live production system, with 10,000 on line users, might affect your job security

Theoretical Basis

Uncertainty Principle: Werner Karl Heisenberg (1901-1976)

Newton's Laws of Motion: Isaac Newton (1643-1727)

Observer Effect

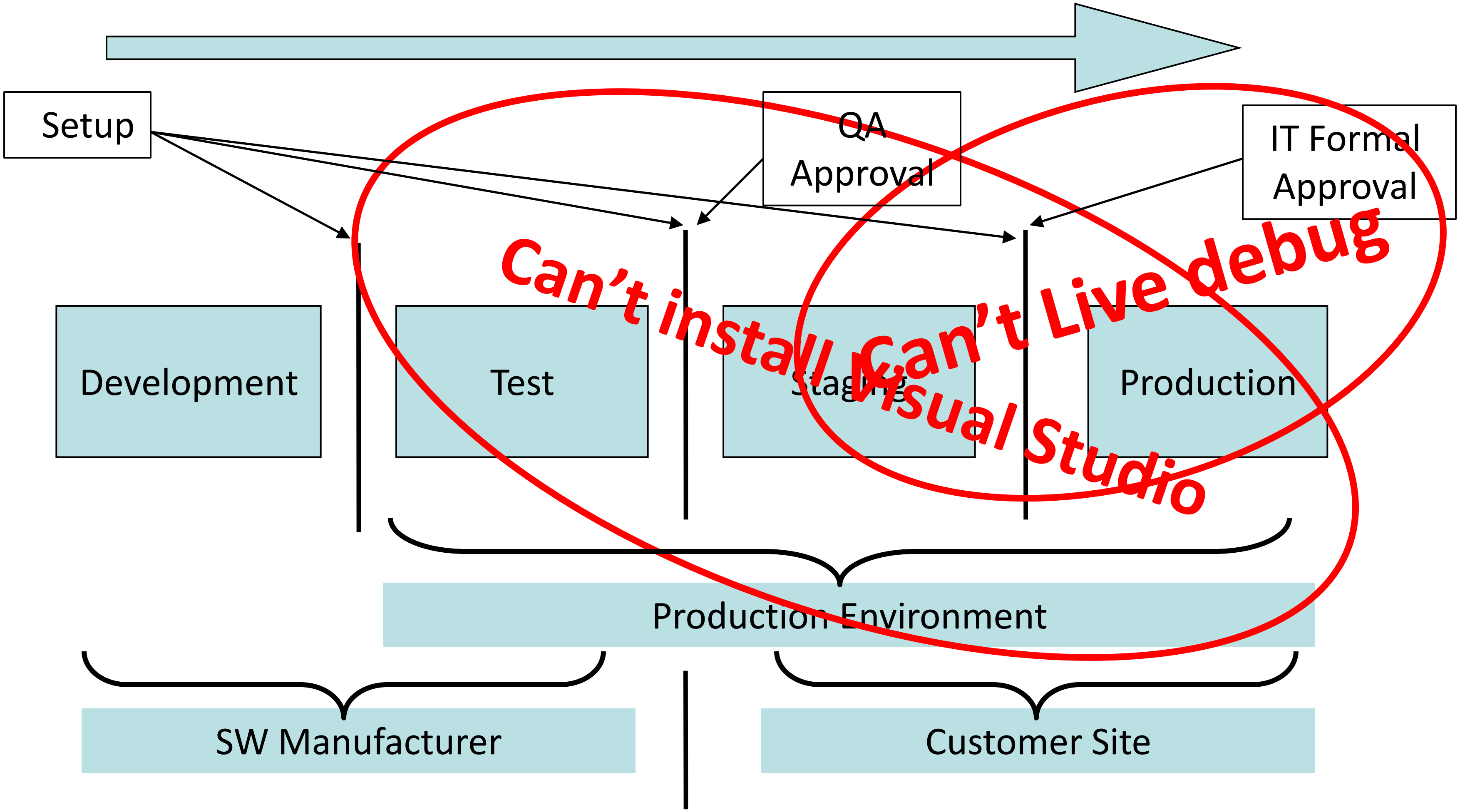
Murphy's Law

What is a Production Environment

What is a Production Environment

- Must be up and running all the time !!!
- Managed by administrators and help desk
- Under change control
- Managed remotely by management tools
- Different Hardware / Software
- Different OS constrains (Policy, Security, …)

Development & Production



Setup

QA Approval

IT Formal Approval

Development

Test

Staging

Production

Production Environment

SW Manufacturer

Customer Site

Can't install Visual Studio

Can't Live debug

About a Dump

- A snapshoot of the process memory at the time you take the dump
- Easy to get in production environments with minimum intervention with the production
- In most of the cases includes all the information needed to analyze the problem

Demo 010

- Analyzing a dump from a crashed program

Pathology Basics

- A dead body is as good as a live one
  - The only thing you can’t do with a dump is single-step it
  - You can duplicate and distribute dead bodies
- Conclusion and strategy # 1
  - Take the money and run

6 Easy Steps for beginners

- Get the tools
- Get the Symbols
- Set the environment
- Take a Dump
- Drop the dump into the tool
- !analyze

How to Get the tools

- The Debugging tools for windows MSIs are In the SDK
- Download from http://msdn.microsoft.com/windows/hardware and go to Downloads
- Install once (for every hardware architecture)
- Zip and copy to you tools repository
- No need to install for using (Important for production)
- .

How to Get the Symbols

- The Symbols MSIs are In the SDK
- Download from http://msdn.microsoft.com/windows/hardware and go to Downloads and than to Other hardware and development tools and than to Download windows symbol packages
- Install once (for every hardware architecture and OS)
- Put in a public location
- Remember the path

Set the environment

- Open WinDbg
- Set the symbol path
  - .sympath to app PDBs
  - .sympath+ to the Windows (correct version) PDBs
  - .symfix+ to the Microsoft Symbol server
- Save the WinDbg environment as a workspace for later use

Tools to Take a Dump

- Adplus
- Windbg .dump
- Process Explorer
- Task Manager (Vista & Above)
- DebugDiag
- UserDump
- ProcDump
- WER
- …

Demo 020

- Taking a dump of a hanged program using Task manager

About the different types of Dumps

- Application Mini dump
  - More or less just the call stack
- Application Full dump
  - Everything
- (Kernel dumps mini, kernel and full)
  - For BSODs

Demo 030

- Taking a dump of a hanged program using WinDbg

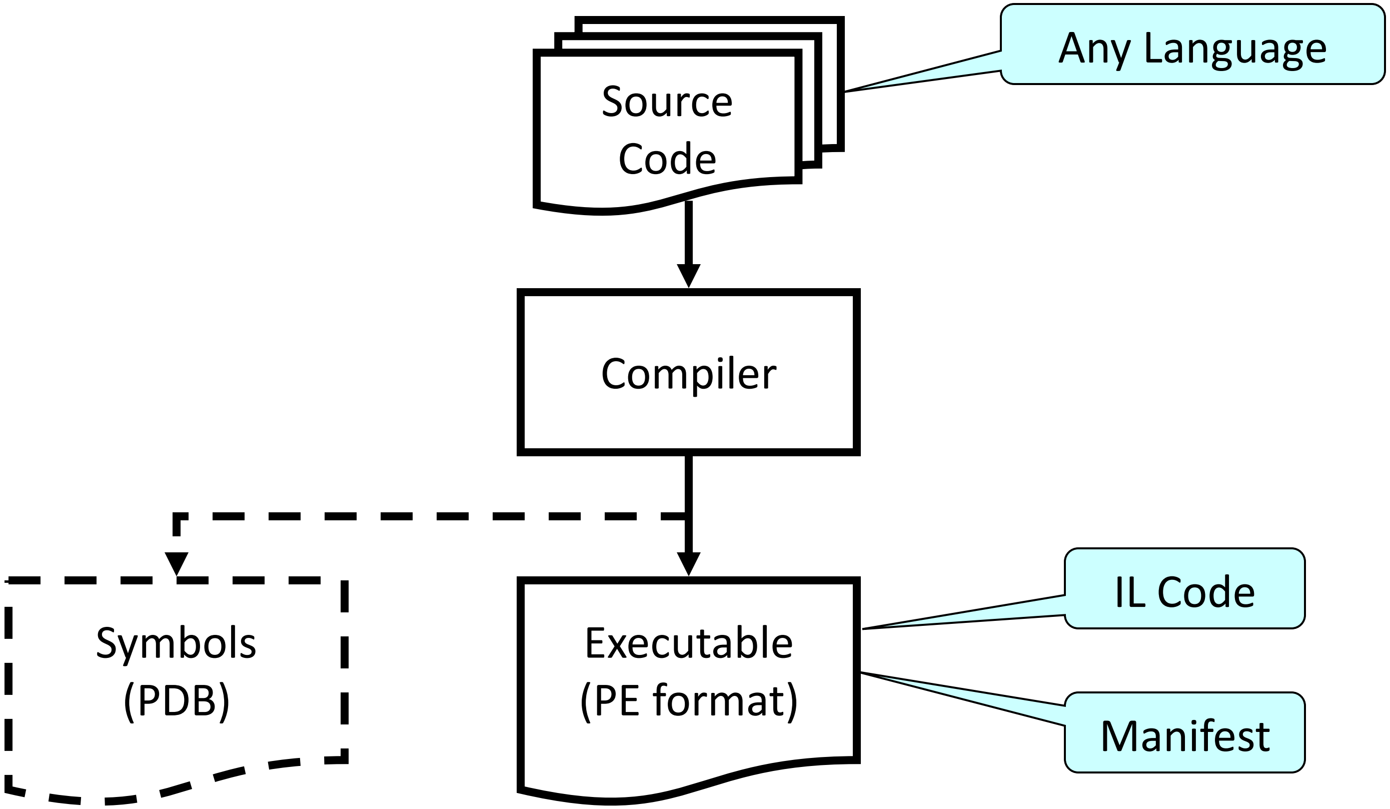
About .NET (CLR)

- CLR is a win32 program!
  - A COM component
- CLR is the execution engine for IL code
- With win32 tools just the CLR engine is noticed
  - IL running code is ignored!
- SOS debugger extension is required
  - ‘Translates’ from Managed to Native

Minimum .NET Internals

- Stack Machine (Reverse Polish Notation)
- Basic data unit is an Object
- The IL code is JITed into Native Code
    - On a function by function basis
    - On the first encounter

Preparing the .NET Executable



Source Code

Any Language

Compiler

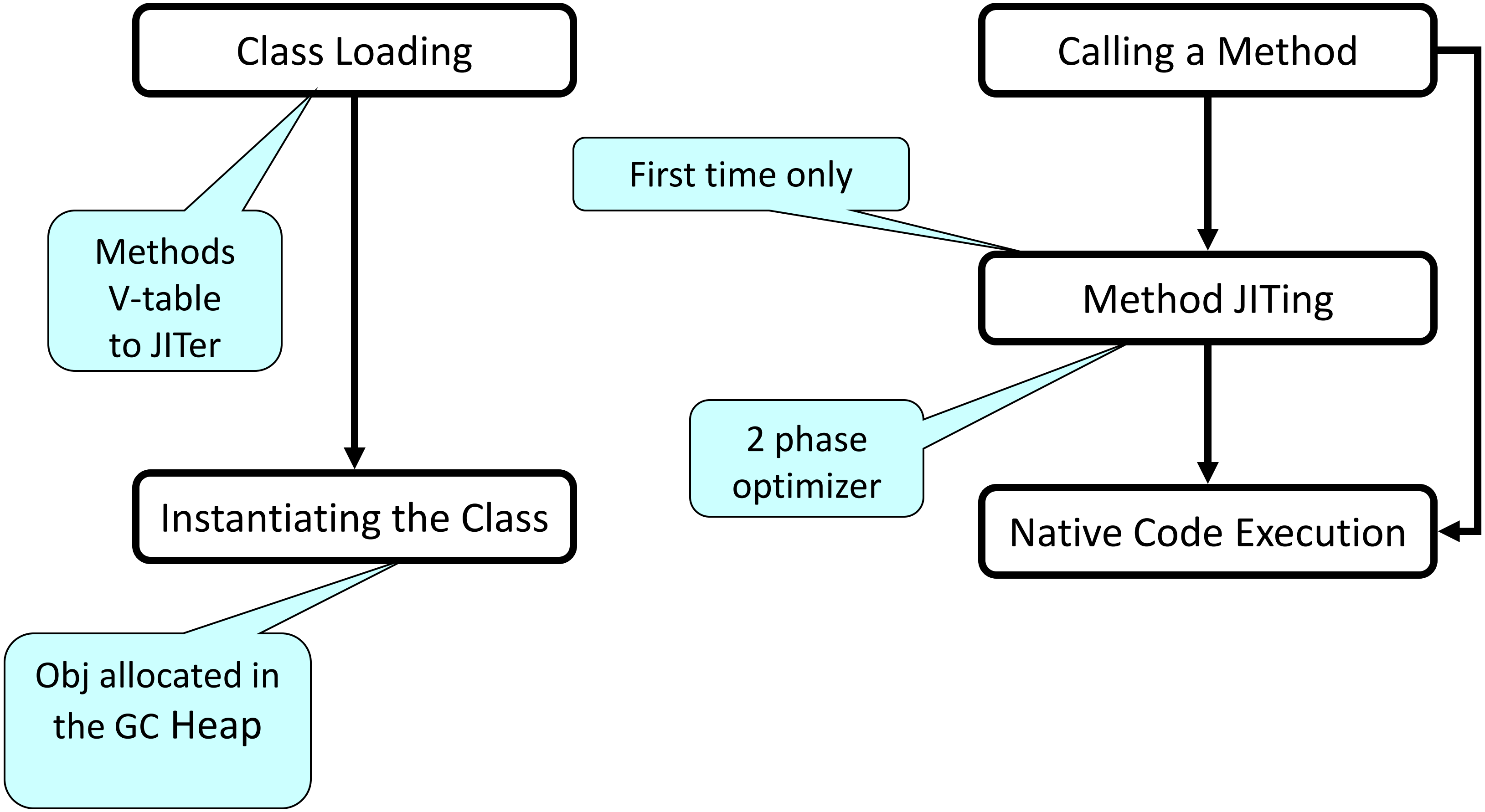
Symbols (PDB)

Executable (PE format)

IL Code

Manifest

Running the Code in the CLR



Problems with .NET

- No PDBs for JITed code
- JITed code is ‘nowhere’
- CLR handles all exceptions
- Hara-kiri effect when CLR can’t handle an exception
  - By default, the CLR kills every one involved, cleans all the evidence from the crime scene and commits suicide, without leaving a comprehensible note

Demo 040

- .NET Hara-kiri effect
  - Native Crash
  - Managed Crash

SOS !Help

- **Object Inspection**
- DumpObj (do)
- DumpArray (da)
- DumpStackObjects (dso)
- DumpHeap
- DumpVC
- GCRoot
- ObjSize
- FinalizeQueue
- PrintException (pe)
- TraverseHeap
- **Examining CLR data structures**
- DumpDomain
- EEHeap
- Name2EE
- SyncBlk
- DumpMT
- DumpClass
- DumpMD
- Token2EE
- EEVersion
- DumpModule
- ThreadPool
- DumpAssembly
- DumpMethodSig
- DumpRuntimeTypes
- DumpSig
- RCWCleanupList
- DumpIL

Examining code and stacks

Threads
CLRStack
IP2MD
U
DumpStack
EEStack
GCInfo
EHInfo
COMState
BPMD

Diagnostic Utilities

VerifyHeap
DumpLog
FindAppDomain
SaveModule
GCHandles
GCHandleLeaks
VMMap
VMStat
ProcInfo
StopOnException (soe)
MinidumpMode

Other

FAQ

Demo 050

- WinDbg Native and Managed view of .NET program
  - Without SOS
  - With SOS

Demo of a .NET Crash 060

- Call Stack
  - !clrstack
- Objects and Values
  - !do
- Object Stack
  - !dso

Demo of a Deadlock Scenario 070

- !syncblk

Demo of Finalization Starvation 080

- !finalizequeue

Summery

- In the field you can’t use the same techniques you use in development.
- Extracting dumps is one of the ways to gather information in the field without disturbing production.
- Instrumentation is key to help you gather information in the field

If you want to learn more

- IDAG Ltd. have a 3 day of practical workshop on the subject of “Production Time debugging”.
- The workshop contain practical labs based on real live scenarios.
- The workshop includes all the methodology and practical consideration to properly debug application in the field.

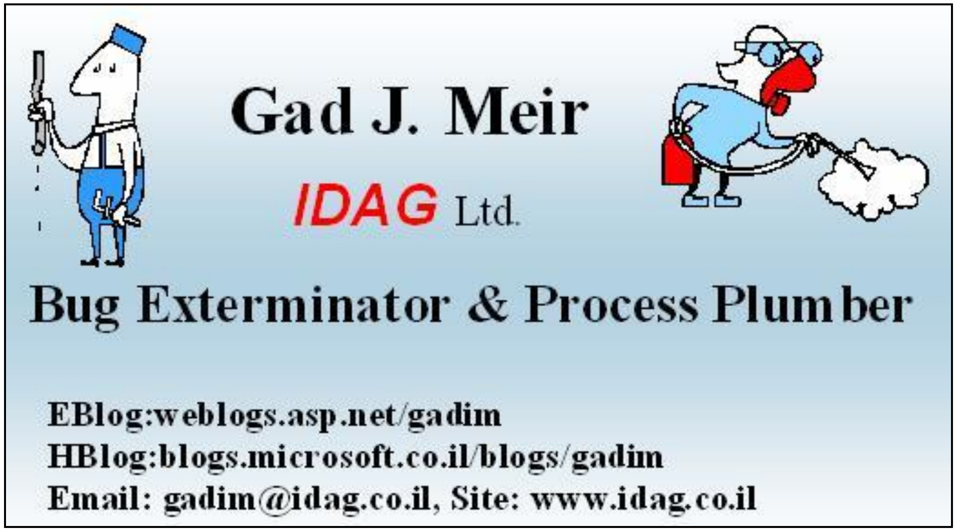
Resources

- http://msdn.microsoft.com/windows/hardware
- winqual.microsoft.com
- “Debugging tools for Windows” help file
- “Debugging tools for Windows” SDK
- Debugging MS .NET 2.0 Applications Ch 6
- MSDN patterns & practices Debugging (Archived)
- !SOS.help & Q&A
- **http://blogs.msdn.com/tess**
- **http://support.microsoft.com/kb/q286350/**
- **Advanced Windows Debugging**
  - **ISBN 0-321-37446-0 ,Addison Wesley, Mario Hewardt & Deniel Pravat**

Some Philosophy

- IT managers appreciate professionalism
  - Be prepared, know your tools and their footprints
  - Learn enough about IT to show them you are not the enemy
  - Listen, Listen, Listen
- Listen to the customer !
  - You developed it, but they use it every day
  - Write everything they complain about and put it straight into the product wish list

Questions?



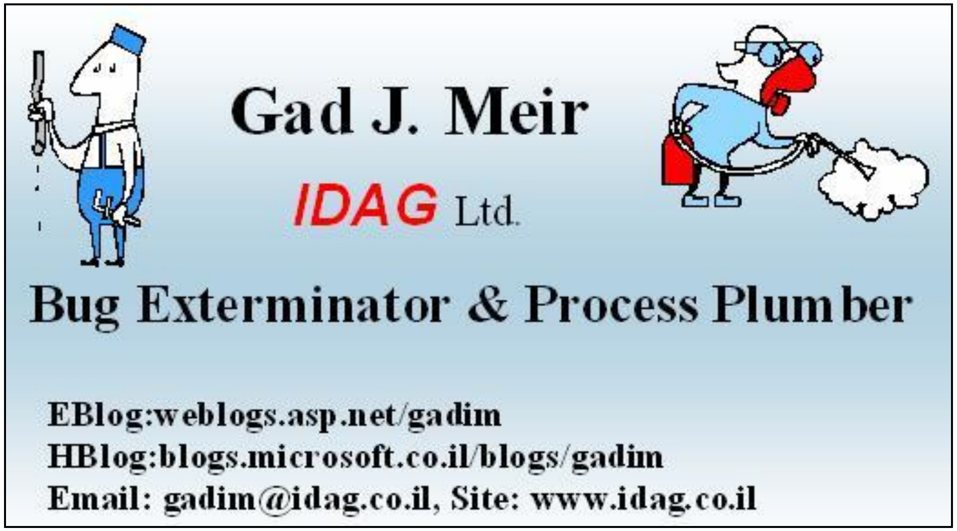
Gad J. Meir

IDAG Ltd.

Bug Exterminator & Process Plumber

EBlog:weblogs.asp.net/gadim
HBlog:blogs.microsoft.co.il/blogs/gadim
Email: gadim@idag.co.il, Site: www.idag.co.il

Thank You!



Gad J. Meir

IDAG Ltd.

Bug Exterminator & Process Plumber

EBlog:weblogs.asp.net/gadim
HBlog:blogs.microsoft.co.il/blogs/gadim
Email: gadim@idag.co.il, Site: www.idag.co.il

Preparing Application for Production Environment

Gad J. Meir

IDAG Ltd.

Bug Exterminator & Process Plumber

EBlog:weblogs.asp.net/gadim
HBlog:blogs.microsoft.co.il/blogs/gadim
Email: gadim@idag.co.il, Site: www.idag.co.il

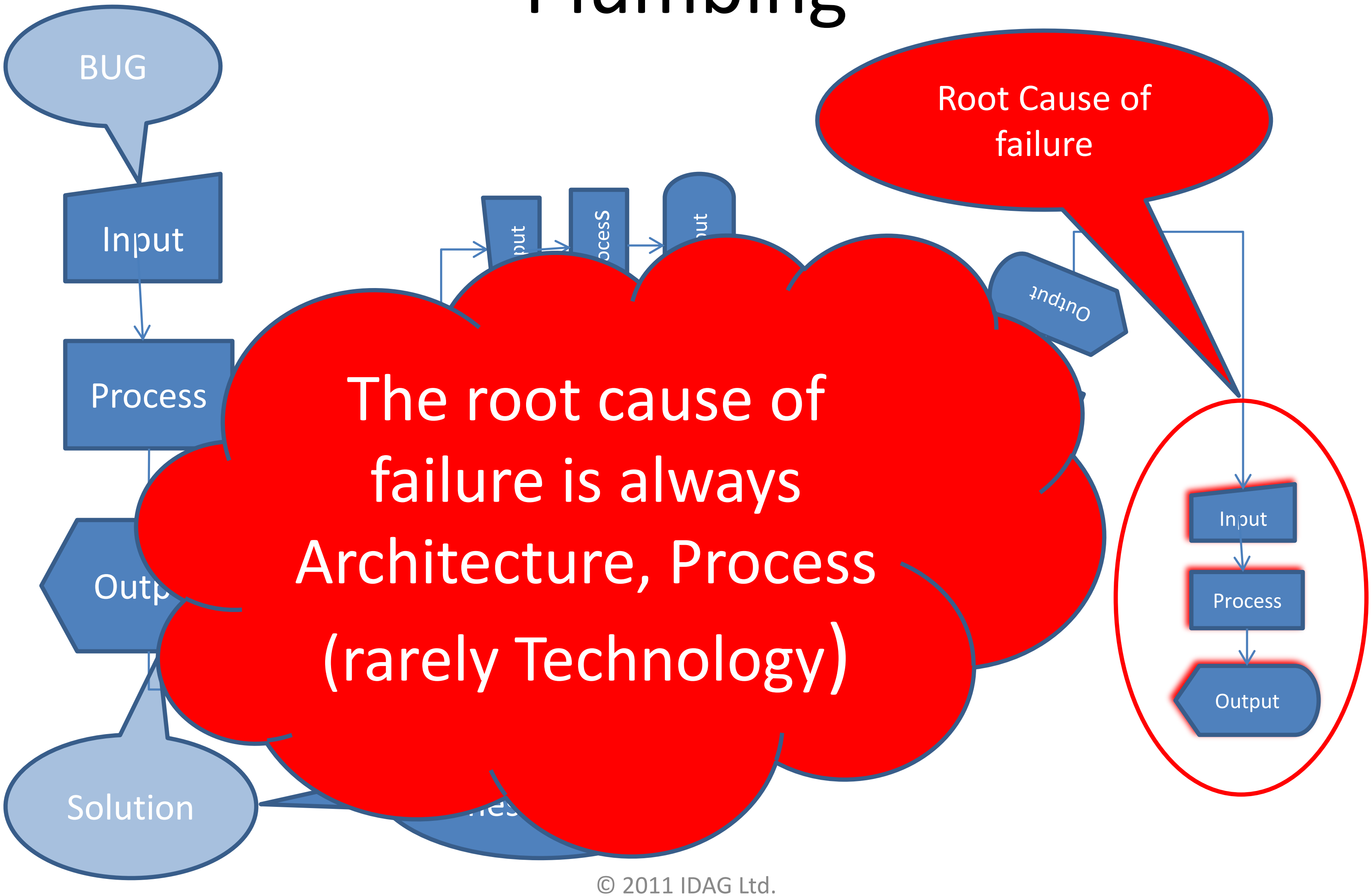
About Gad J. Meir

- Experience: Since 1975
- Work: www.idag.co.il
- Function: www.productiondebugging.com
- Blog: http://weblogs.asp.net/gadim/default.aspx
- MSF Certified Trainer & Practitioner
- BSc. Computer engineering Technion
- Microsoft Certified MC…

About IDAG Ltd.

- Founded 1983
- Established the first Microsoft certified training center in Israel at 1992.
- Areas of operation
  - Troubleshooting systems and procedures
  - Production time debugging to root cause of failure
  - Projects monitoring and guidance
  - Knowledge gaps detection and filling
  - Technologies and methodologies deployment

From Bug Extermination to Process Plumbing



The root cause of failure is always Architecture, Process (rarely Technology)

Talk Targets

- Explain some of the specific constrains of production environment / Field
- Introduce ways to Reduce the operation costs of an application in production environment with minimum overhead to the development team
- Several Demos
- Tips

Prerequisites

- None

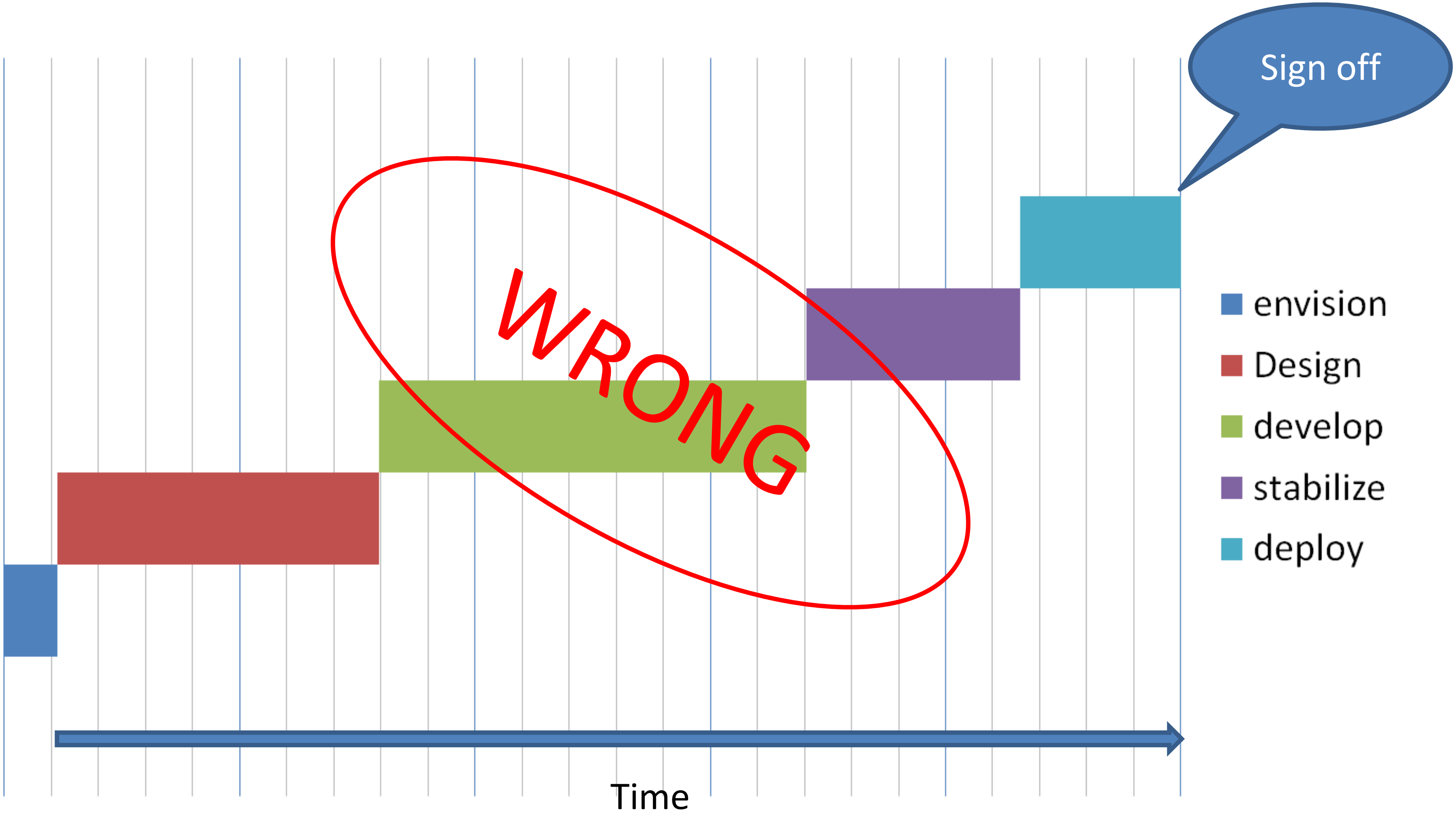
Agenda

- The real life cycle of an application and the TCO of a software system
- your customer(s)
- production environment manageability and down time costs
- Ways to make the application production environment friendly
  - Event logs
  - Performance counters, Base lining and Trends
  - Event Tracing for Windows (ETW)
  - Windows Management Instrumentation (WMI)
  - Windows Error Reporting (WER) and being ‘crash friendly’
  - Production debugging in the field usage, features and specifications.
  - Configuring the operating system for failure
  - Power Shell
  - …

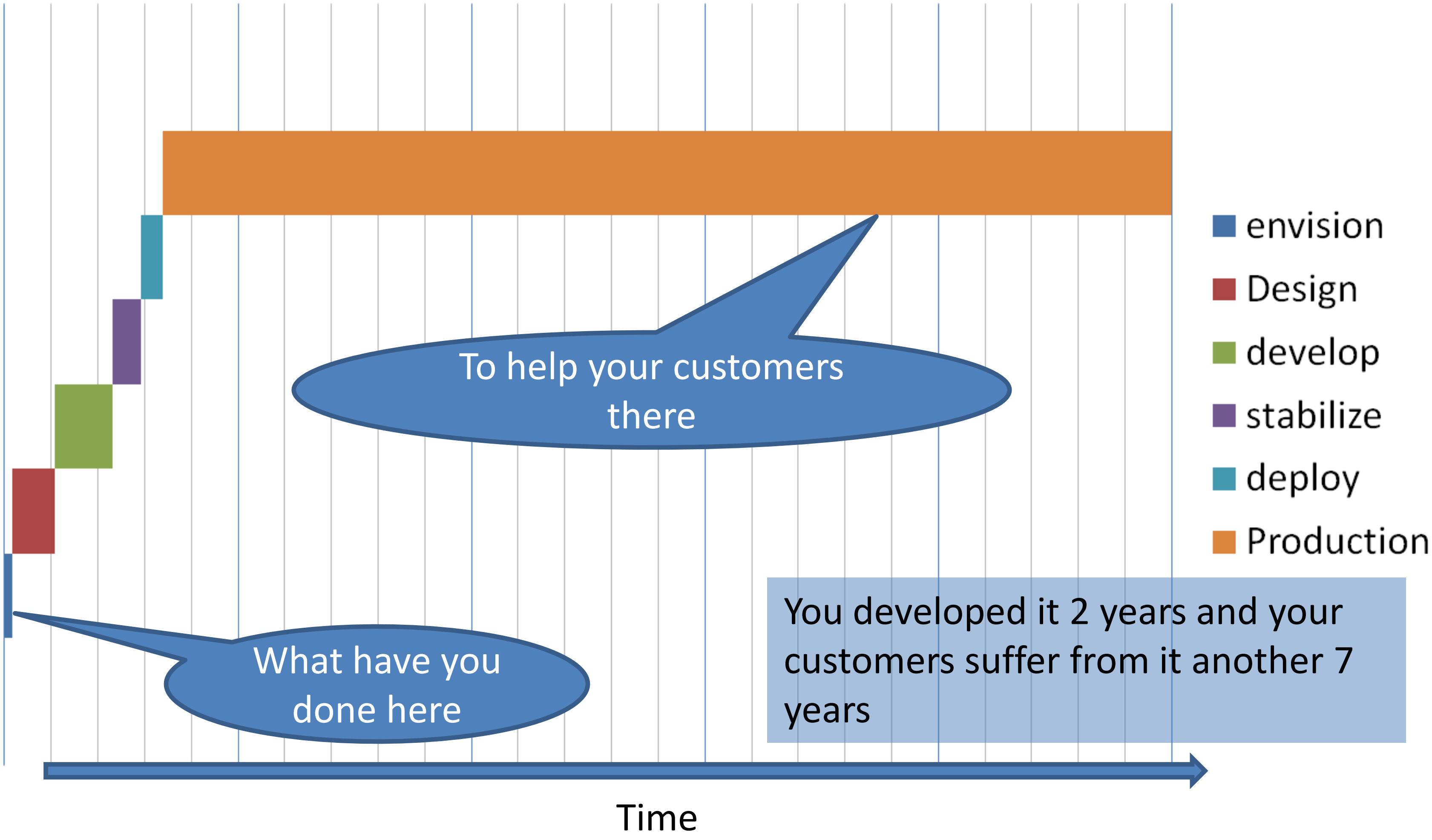
Please !

- If you don’t understand what I am talking about, Stop me and ASK !!!, Don’t wait.

Software Project Life Time

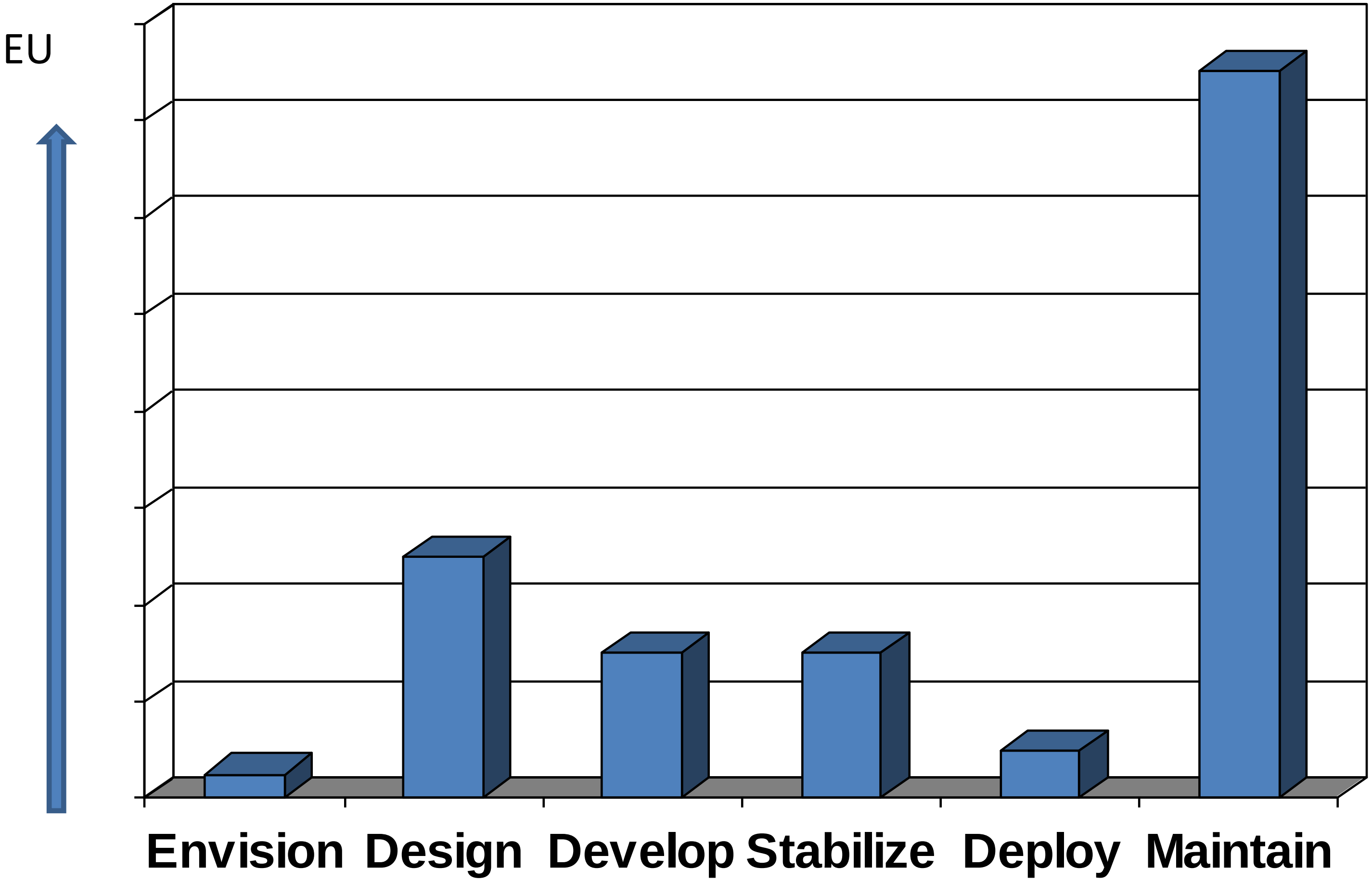


Software Project Life Time



You developed it 2 years and your customers suffer from it another 7 years

The Full Cost of an Application



The CustomerS

- The customer is the one that pays
- IT
- Help Desk
- Field Engineer and Field Support
- All levels of customer support
- QA & Testing
- Users
- Development Team
- Business decision makers
- Sales representative

What is a Production Environment

What is a Production Environment

- Must be up and running all the time !!!
- Managed by administrators and help desk
- Under change control
- Managed remotely by management tools
- Different Hardware / Software
- Different OS constrains (Policy, Security, …)

What is a Production Environment

- Must be up and running all the time !!!
- Managed by administrators and help desk
- Under change control
- **Managed remotely by management tools**
- Different Hardware / Software
- Different OS constrains (Policy, Security, …)

How many screens are there in a 100 server computer center

- What is the size of a 100 server computer center ?
- How many screens are there in a 100 server computer center ?
- About KVM
- Why MsgBox is not a very useful tool to notify the operator about an application problems
  - Does a service have a Desktop ?
  - Who's gonna click on the OK button

System management tools

- Microsoft Operations Manager (MOM) & Microsoft SCOM, Microsoft Opalis
- HP Openview Operations and BAC SiteScope
- Computer Associates CA Unicenter
- IBM Tivoli
- BMC ProactiveNet Performance Management

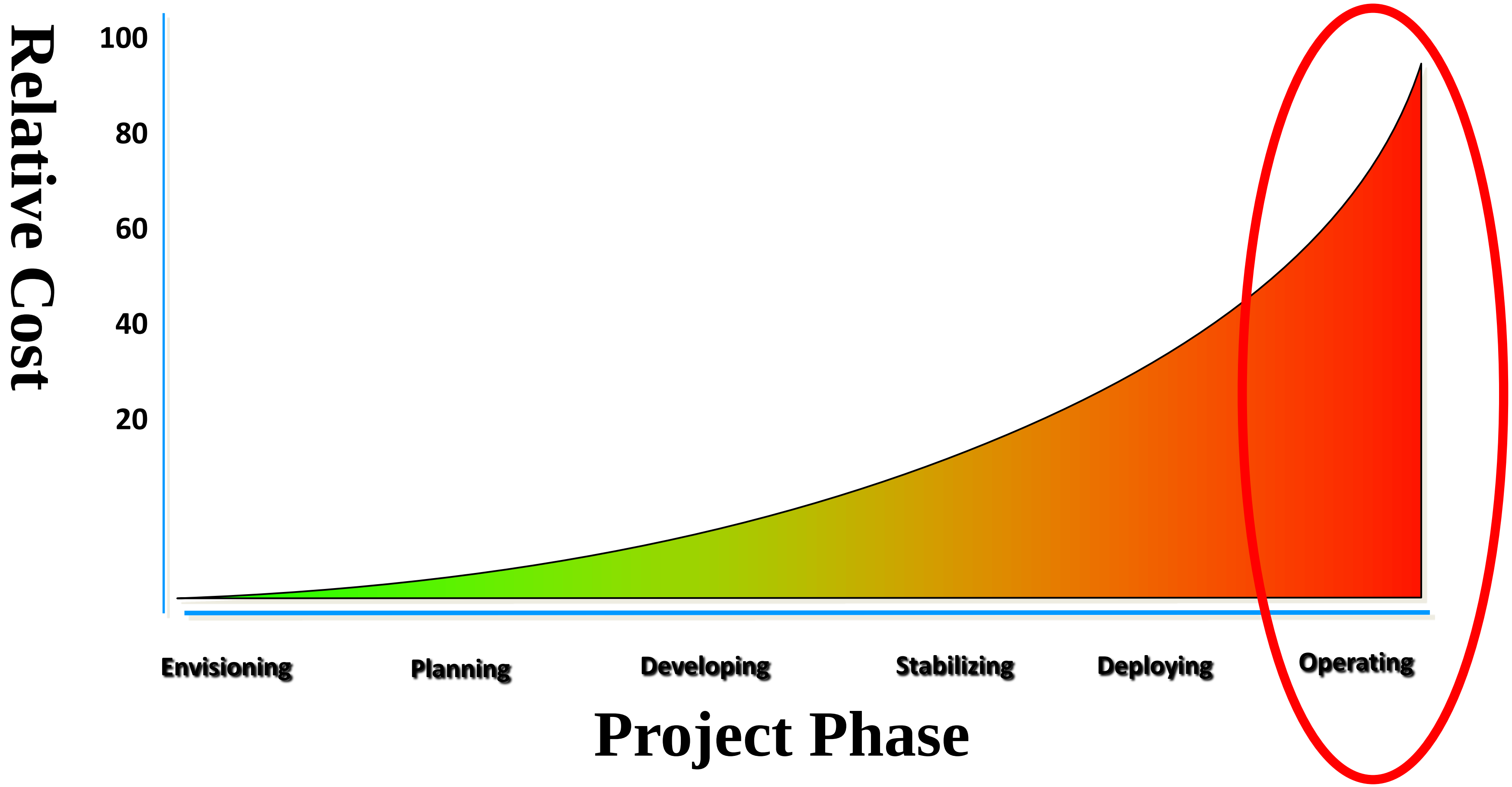
What is a Production Environment

- **Must be up and running all the time !!!**
- Managed by administrators and help desk
- Under change control
- Managed remotely by management tools
- Different Hardware / Software
- Different OS constrains (Policy, Security, …)

Your application is going to crash !!!

- At the beginning of the envisioning phase of an application, you already know it’s going to crash in production or at a customer’s site.
- It’s not a question of IF but of WHEN.

Cost of Fixing a Solution



How much does a crash costs?

- Direct costs
  - α Clients cant use the system for β hours
  - γ IT personal work for δ hours to fix the problem (δ >> β)
- Indirect costs
  - Degradation in clients and IT satisfaction (reputation, attitude, trust)
  - SLA Penalties
  - Other expenses

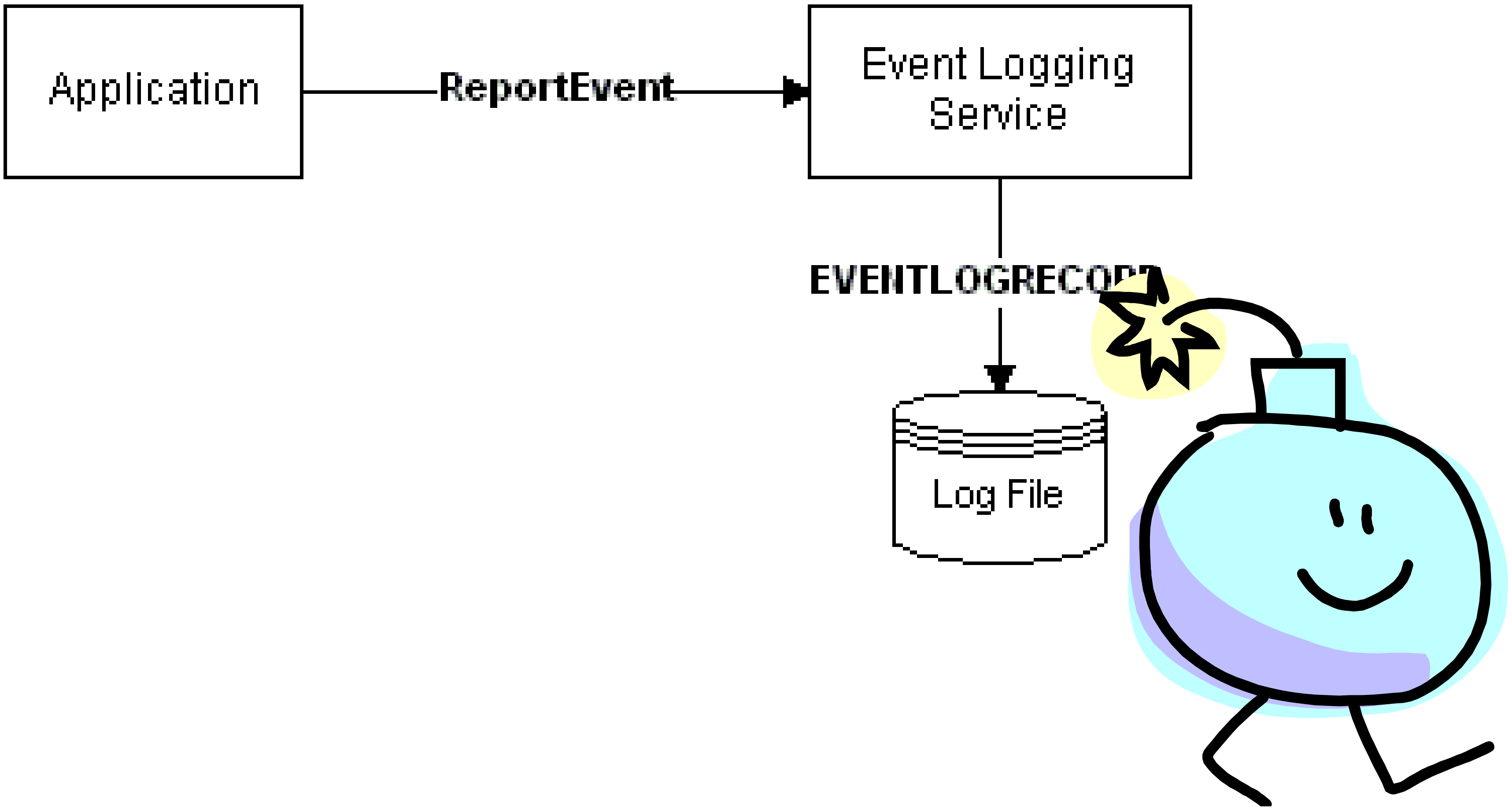
Finding and fixing a bug faster

- With proper instrumentation, IT can find program abnormal behavior faster and reduce down time (responsibility of the development team)
- With proper production time data collection before and at time of abnormal behavior developers can find the bug quicker (responsibility of IT & operations)
- Reduces TCO

Make the application production environment friendly

- Event Logging
- ETW - Event Tracing for windows
- Performance Counters
- WMI - Windows management instrumentation
- WER - Windows error reporting
- MMC - Microsoft Management Console
- Power Shell
- System Management friendly
- Crash and Production Time Debugging friendly

Event Logging Demo



Application

ReportEvent

Event Logging Service

EVENTLOGRECORD

Log File

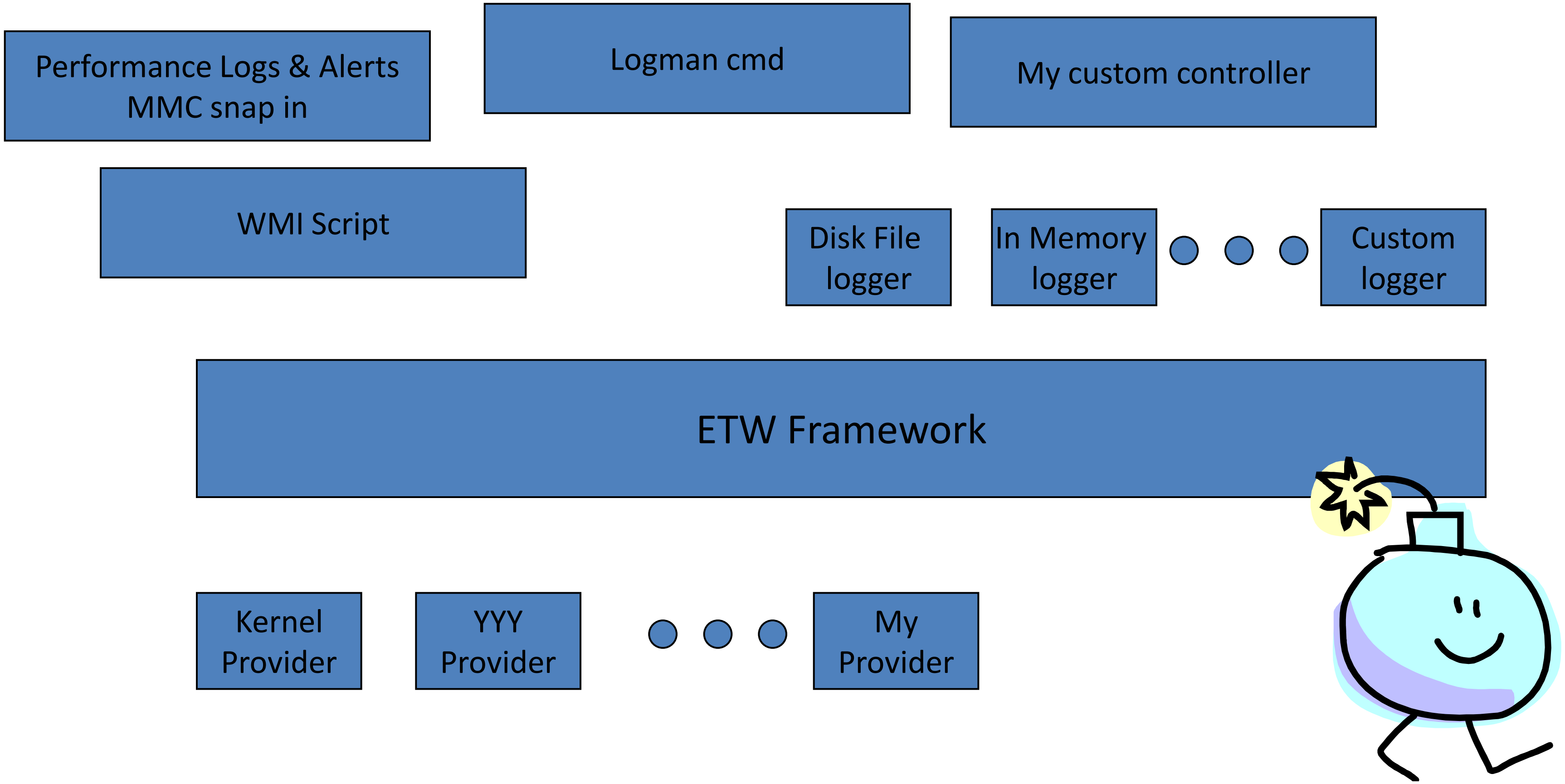
Event Logging Demo Debrief

- The infrastructure is built in the operating system
- Fully integrated with most of the automatic management tools.
- Simple API interface
- System event log for administrators and private event log if the need arise.
- The design of the “What to log where” is the most time consuming task

Trace Framework Requirements

- Works only when required
- Start & stop manually and/or conditionally
- Dynamic configuration of what to trace
- Versatile output logging options
- Time stamps and management data
- Suitable for production environments
- Low footprint
- Minimum performance degradation

ETW Demo



Performance Logs & Alerts MMC snap in

Logman cmd

My custom controller

WMI Script

Disk File logger

In Memory logger

○ ○ ○

Custom logger

ETW Framework

Kernel Provider

YYY Provider

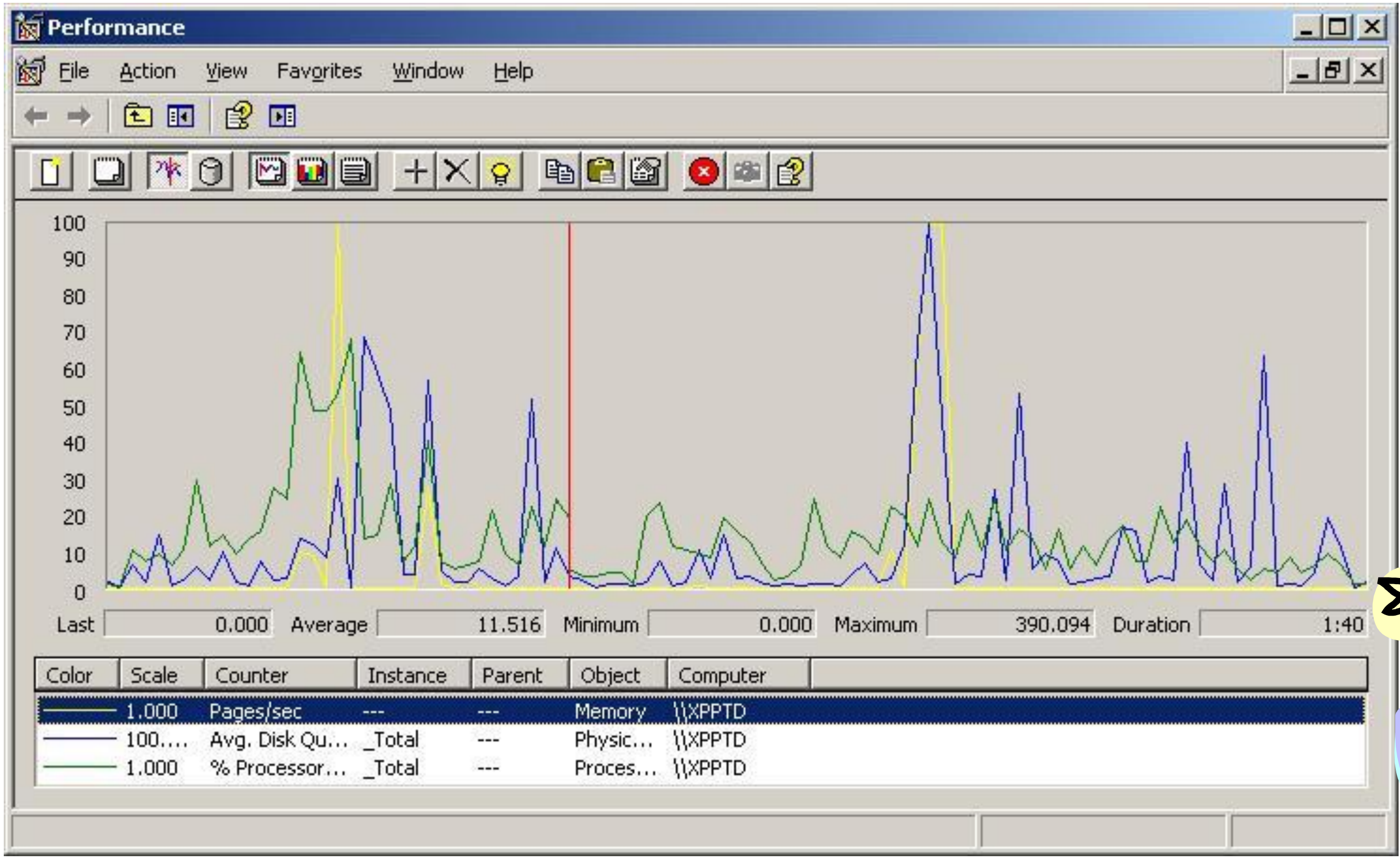
○ ○ ○

My Provider

ETW Demo Debrief

- The infrastructure is built in the operating system (since windows 2000 !).
- Just 3 API calls
- Zero development effort Huge benefits
- The design of the “printf’s” is the most time consuming task
- Can be used for error tracing and performance measurements

Performance Counters Demo



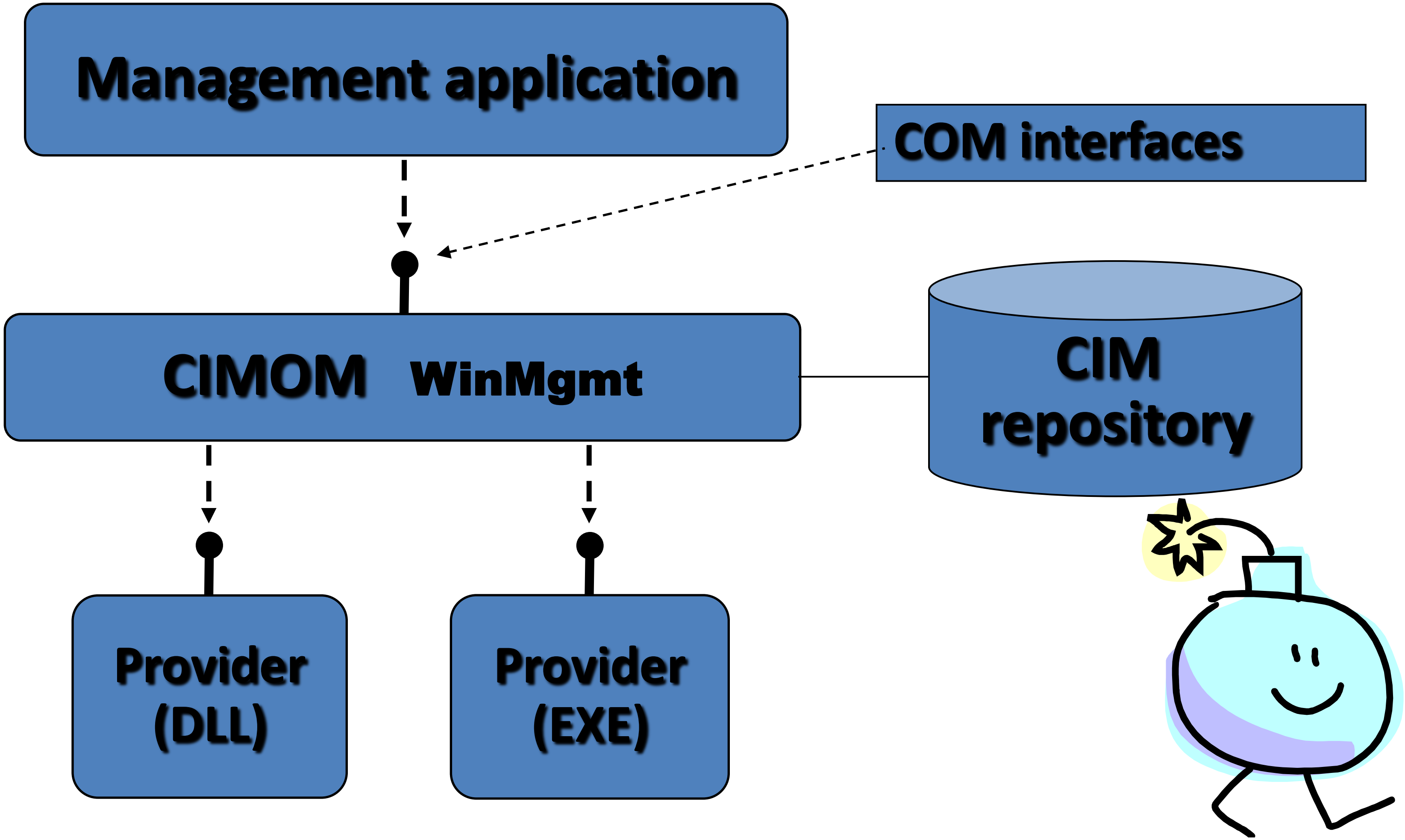
Last 0.000 Average 11.516 Minimum 0.000 Maximum 390.094 Duration 1:40

Color	Scale	Counter	Instance	Parent	Object	Computer
—	1.000	Pages/sec	---	---	Memory	\\XPPTD
—	100....	Avg. Disk Qu...	_Total	---	Physic...	\\XPPTD
—	1.000	% Processor...	_Total	---	Proces...	\\XPPTD

Performance Counters Demo Debrief

- The infrastructure is built in the operating system (since windows NT 2000 !).
- Simple API interface
- Zero development effort Huge benefits
- Capacity planning
- The design of the “Hart beat and test points” is the most time consuming task

WMI Demo



Management application

COM interfaces

CIMOM WinMgmt

CIM repository

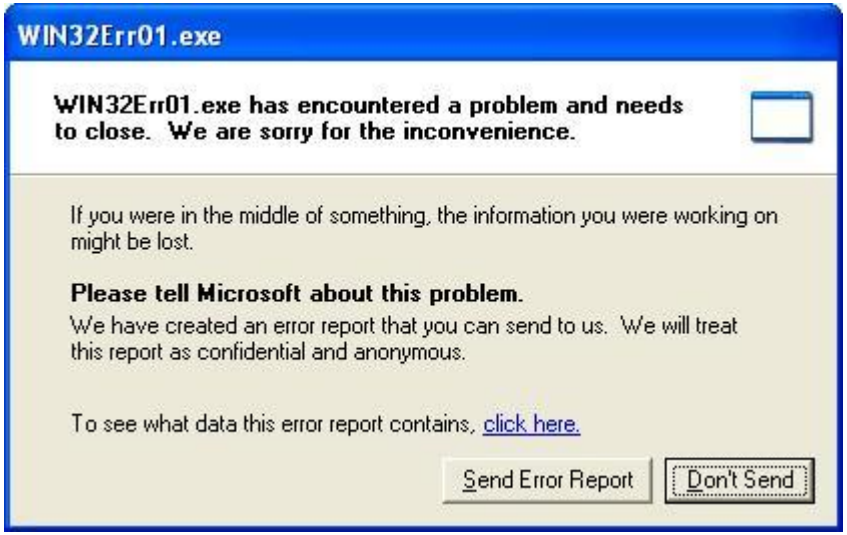
Provider (DLL)

Provider (EXE)

WMI Demo Debrief

- The infrastructure is built in the operating system
- Full integration with all the automatic management tools
- scripting interface as an added value
- Require understanding of DMTF, WBEM, CIM and MOF.

WER Demo



WIN32Err01.exe

WIN32Err01.exe has encountered a problem and needs to close. We are sorry for the inconvenience.

If you were in the middle of something, the information you were working on might be lost.

Please tell Microsoft about this problem.

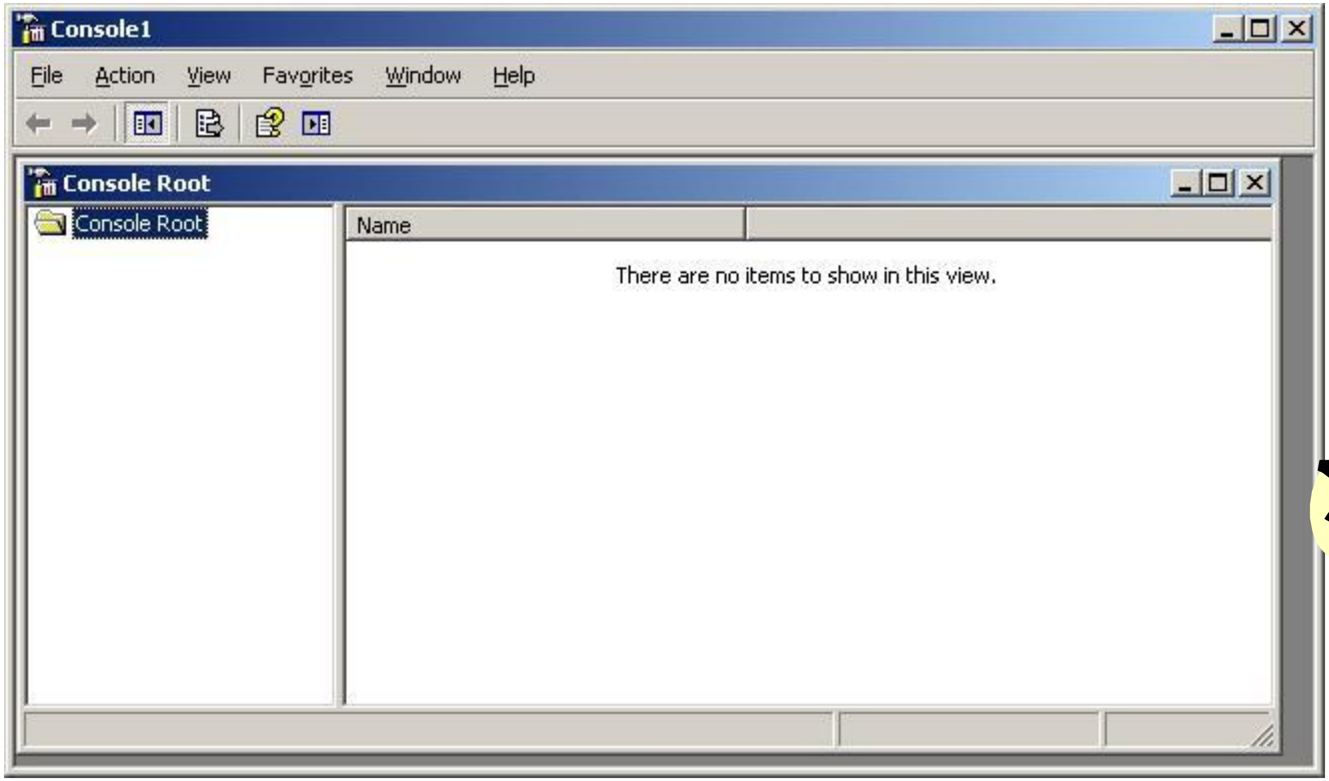
We have created an error report that you can send to us. We will treat this report as confidential and anonymous.

To see what data this error report contains, click here.

WER Demo Debrief

- The infrastructure is built in the operating system (since windows NT 3.11 !).
- Gold mine for developers, call stack at the moment of crash
- Just IT configuration and sending the collected data
- Can be used locally and without user intervention

MMC Demo



There are no items to show in this view.

MMC Demo Debrief

- The infrastructure is built in the operating system (since windows 2000).
- The standard IT tool
- Set the management interface between your application and the IT

Power Shell Demo

Power Shell Demo Debrief

- Every product from Microsoft comes with Power Shell Applet
- Easy to incorporate

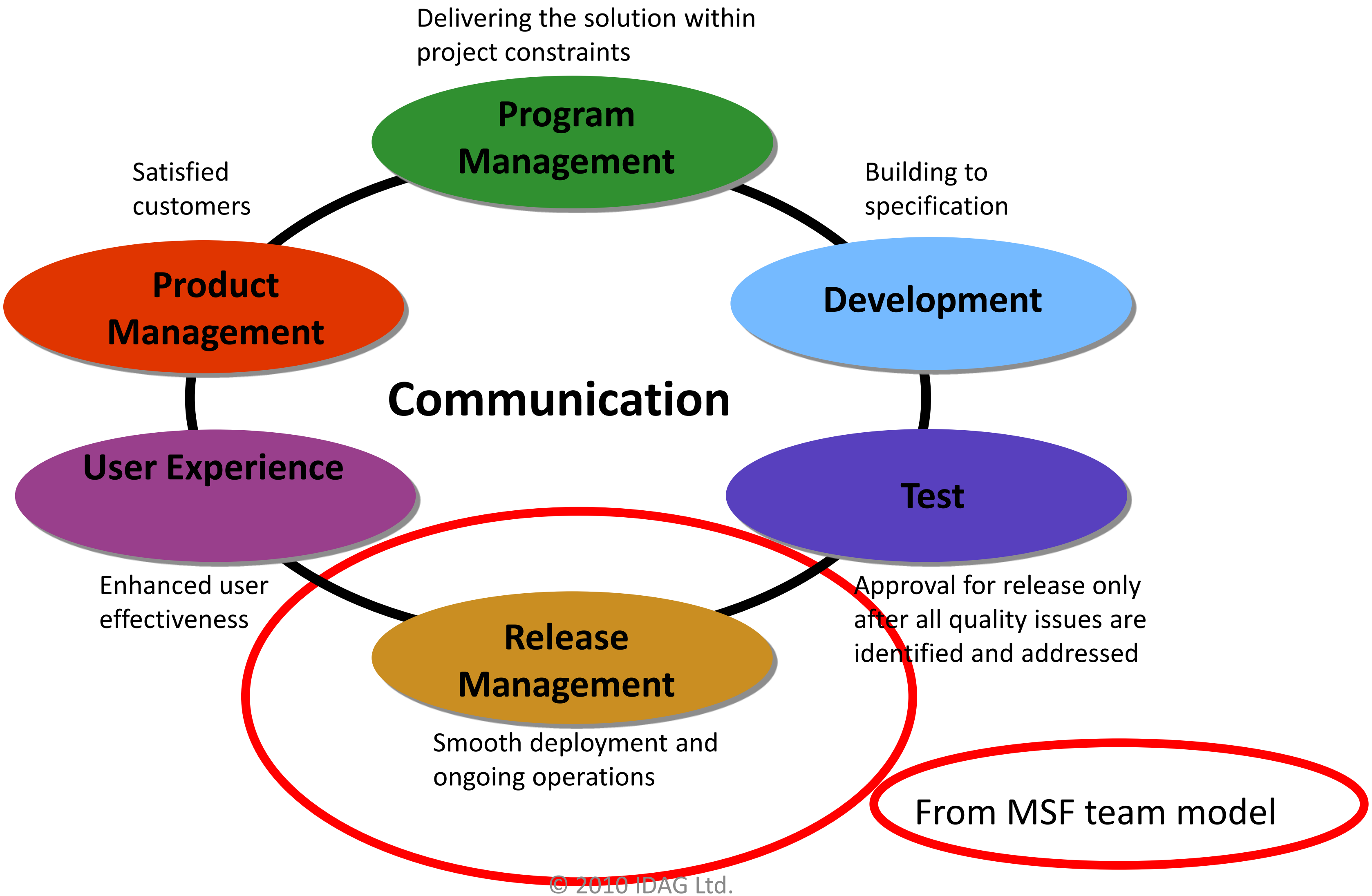
Management and crash friendly

- Script / MMC / Power shell applets / Troubleshooters
- Application specific monitoring and alerting utilities for management and control systems
- Application managed startup / shutdown
- Application current state date collection
- Application crash data setup and collection
- Log interpreting and analyzing utilities

Summery

- Proper instrumentation save a lot of time and money.
- Require cooperation between IT and development.
- Minimum overhead to Developers and IT, Huge benefits to the whole system

Do You Have IT Expert in your Development Team ?



Delivering the solution within project constraints

Program Management

Satisfied customers

Product Management

Building to specification

Development

Communication

User Experience

Test

Enhanced user effectiveness

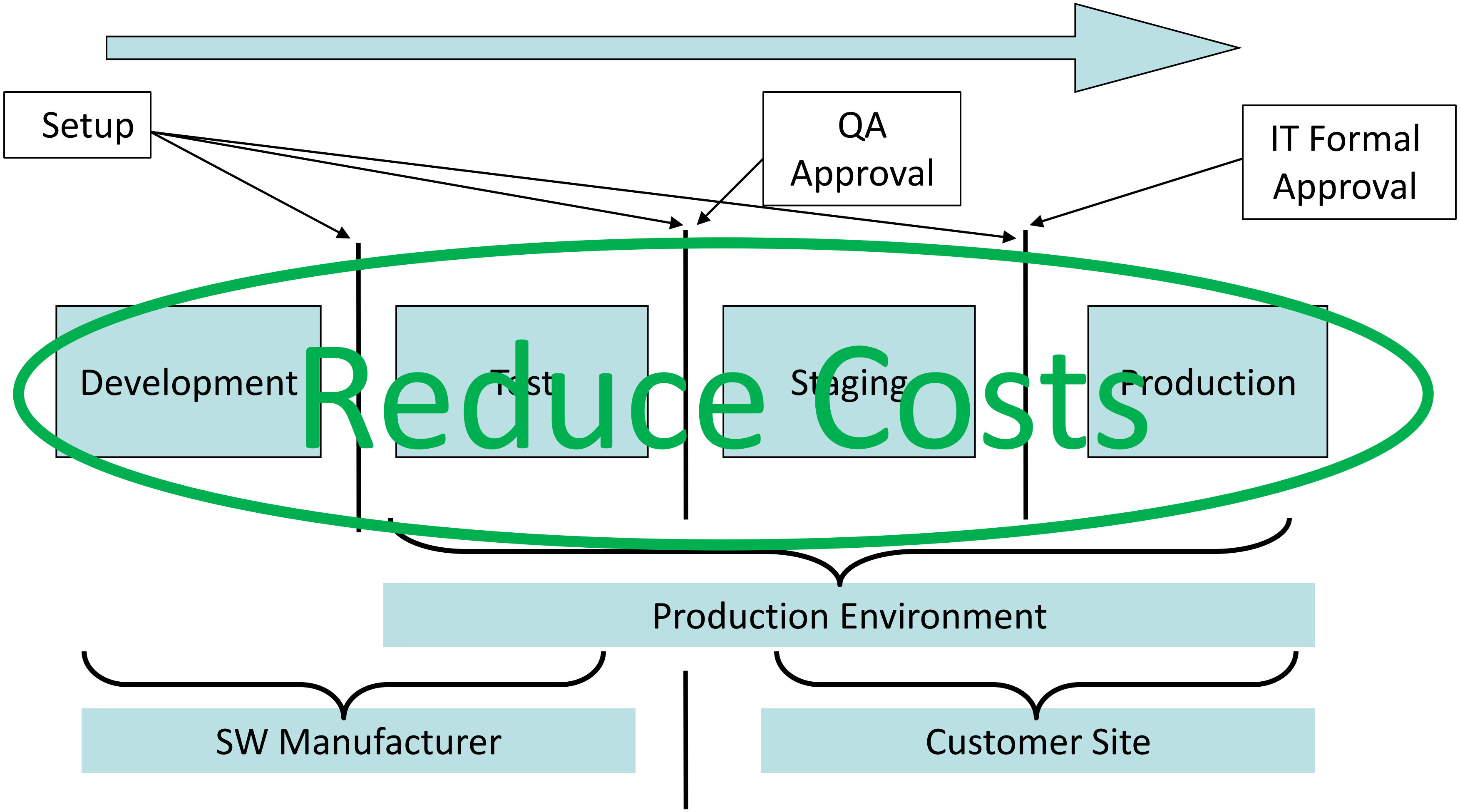
Approval for release only after all quality issues are identified and addressed

Release Management

Smooth deployment and ongoing operations

From MSF team model

Instrumentation Usage



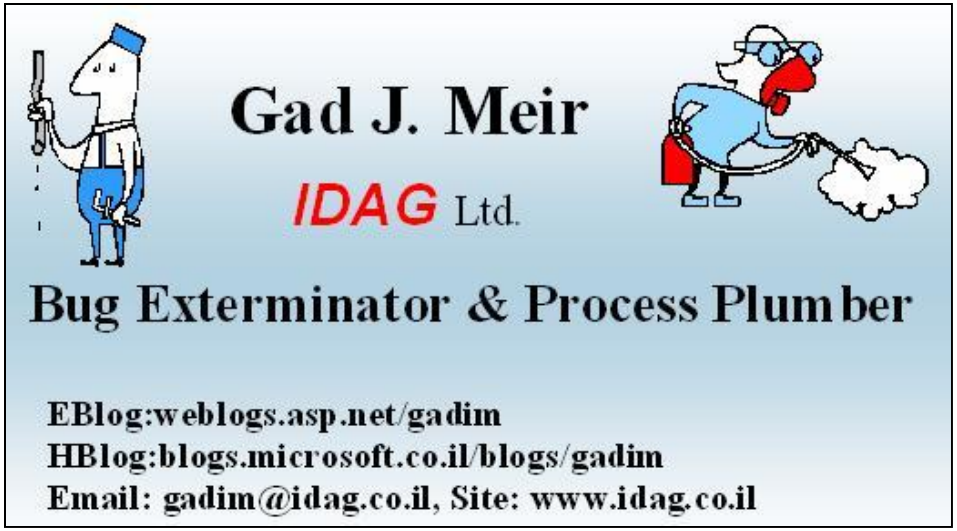
If you want to learn more

- IDAG Ltd. have a 3 day of practical workshop on the subject of “preparing an application for production”.
- The workshop contain practical labs with all the building block code elements.
- The workshop includes all the methodology and practical consideration to make an application production environment friendly.

Resources

- www.productiondebugging.com
- technet.microsoft.com

Questions?



Gad J. Meir

IDAG Ltd.

Bug Exterminator & Process Plumber

EBlog:weblogs.asp.net/gadim
HBlog:blogs.microsoft.co.il/blogs/gadim
Email: gadim@idag.co.il, Site: www.idag.co.il