October 20, 2011

*Elite* Packaging

For the Windows Platform  
*Plus media clips for learning*





**Packaging**

…is the science, the art, and the technology   
 of enclosing or protecting software products   
 for distribution, storage, sale, and use…you   
 will package to meet the needs and requests   
 of your customer.

**Benefits of Packaging**



 Help administrators deploy software   
 configuration – built-in custom configuration



 Create reliable packages that can be   
 successfully distributed throughout an   
 organization – for enterprise delivery  
  
 Minimize administrative errors during   
 installation – built-in logic to handle errors  
  
 Provide features that help streamline   
 application – automate prompts, custom splash  
  
 Reduce maintenance – provide backend support  
  
 Eliminate unnecessary application   
 deployments  
  
 Reduce complexity for support services  
  
 Reduce training of support services

1 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

**Media Clips**







 Application Packaging Explained Visually  
 <http://www.youtube.com/watch?v=gg6s2YAImxI>

 Application Packaging with MSI Studio  
 <http://www.youtube.com/watch?v=O9CumMAnfP4>

 Converting EXE to MSI using Wise  
 <http://www.youtube.com/watch?v=quY_a7Ye_WQ>  
  
 Wise Setup Capture  
 <http://www.youtube.com/watch?v=tQW1nnuD-1g>  
  
 WinInstall MSI Packager version 10  
 <http://www.youtube.com/watch?v=pZzCSrLdf60>

 Install Software through Group Policy  
 <http://www.youtube.com/watch?v=JRNCgvZs5v4>  
  
 Flexera Software  
 <http://www.youtube.com/watch?v=tZSkQbJWUtA>  
  
 Repackaging  
 <http://www.youtube.com/watch?v=5nAwd2JZjUA>

2 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Globally – Windows Packaging



 This area is poorly standardized in general, except Windows Installer

 No significant research or attempts to provide complex patterns

 Requires system hardware and deep software configuration knowledge

 Big amount of custom vendor specific solutions different by design

 Most organizations have their own internal installation frameworks

 ***Quality is critical*** - error localization is complex and resourceful



3 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Purpose of this Document





 To share my experience

 To get feedback from you

 To know more about latest news in   
 this area  
  
 Email: [MrNetTek2000@yahoo.com](mailto:MrNetTek2000@yahoo.com)

 Site: http://eddiejackson.net

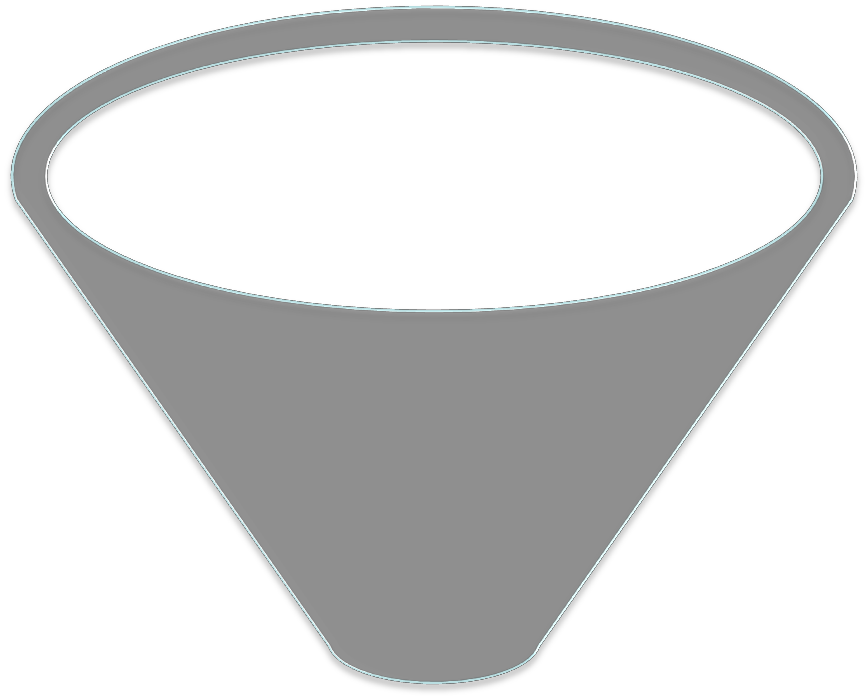
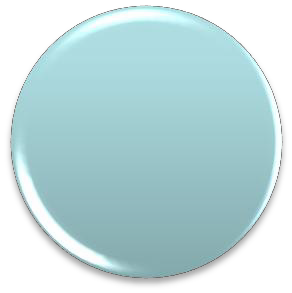
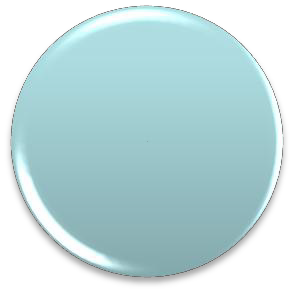
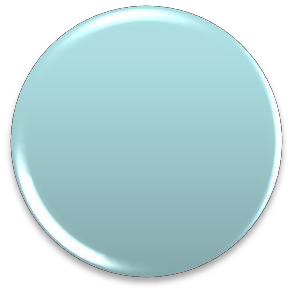
4 © Copyright 1993-2011 MrNetTek http://eddiejackson.net





Welcome to the REAL WORLD  
 The world of packaging is what you make it





5 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Typical Errors to Avoid



 Develop installation as a software product with the same or lower quality



 Change internal package design

without informing customer

 Prioritize application features and quality in expense of packaging

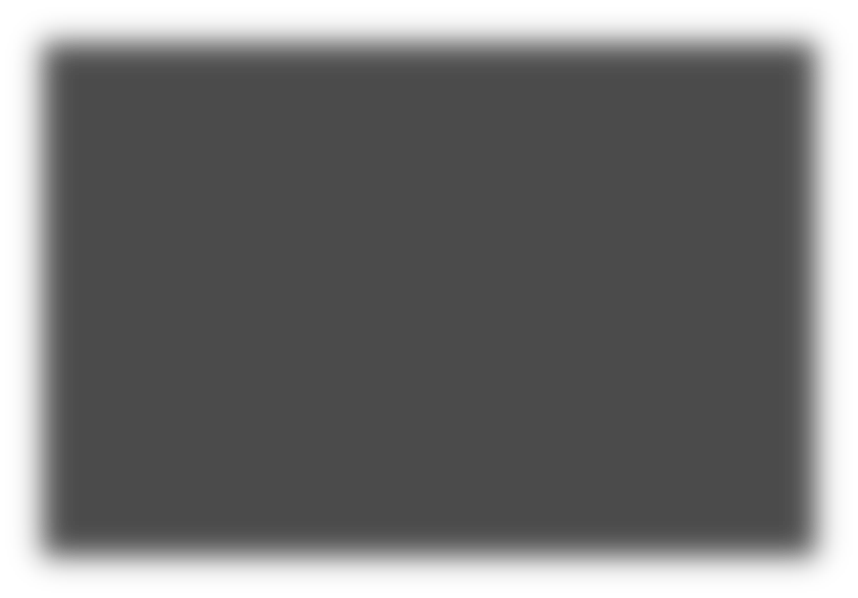
 Start packaging soon before the critical delivery

6 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Typical Errors to Avoid







 Don’t invest in the packaging

design and infrastructure

 Remove garbage from the build outputs right before the final delivery

 Try to automate process without knowledge of the customer requirements

 Limited ability to say “No” and in

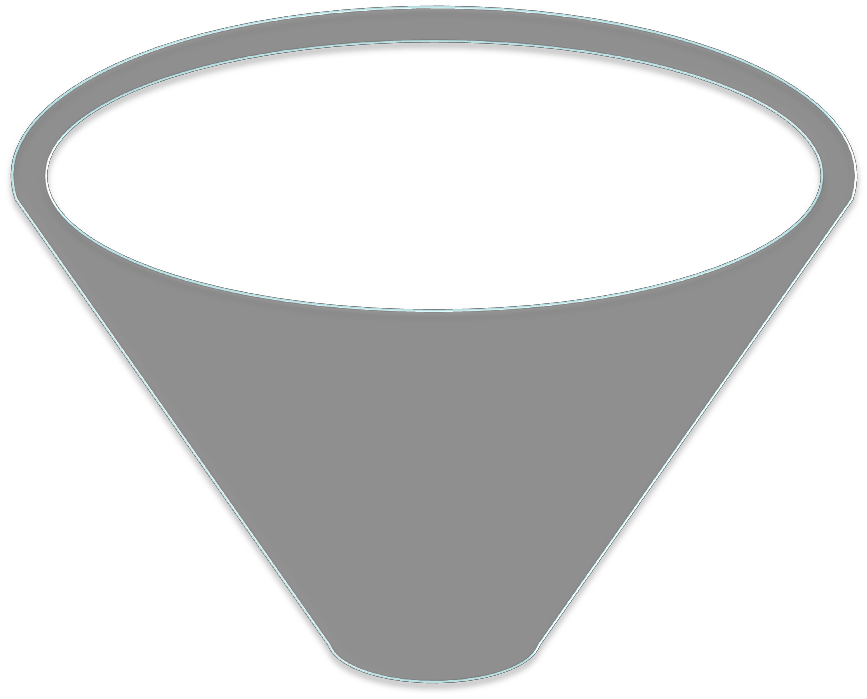
decision making

 Try to integrate in every project ecosystem completely

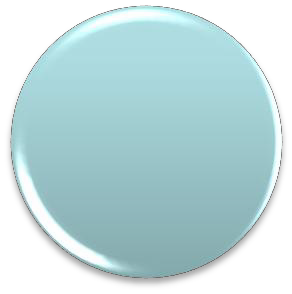
7 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

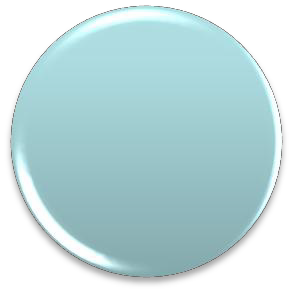
Packaging - Inputs











Project

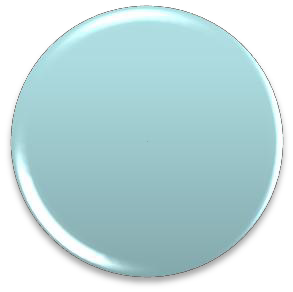
2

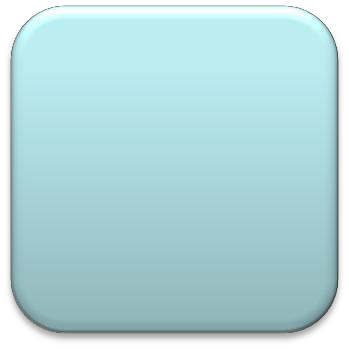
Project

3

**Retail**

**Market**





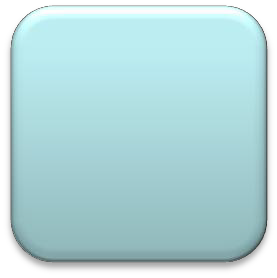
Project

1



**Packaging**

**Team  
Engineer**





Conflict with Scrum!

**Packaging Team**

**SaaS**

**Consulting**

8 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

What to do in Packaging



1.  Gather the application requirements

02. Get the initial deliverables version

03. Provide feedback and ensure that you   
 can satisfy most stakeholders

04. Agree on the common rules and achieve   
 acceptable compromise if needed

05. How to localize application related errors

06. Create acceptance and internal test cases

07. Create test environment if possible

08. Create or update installation package

09. Save sources and binaries

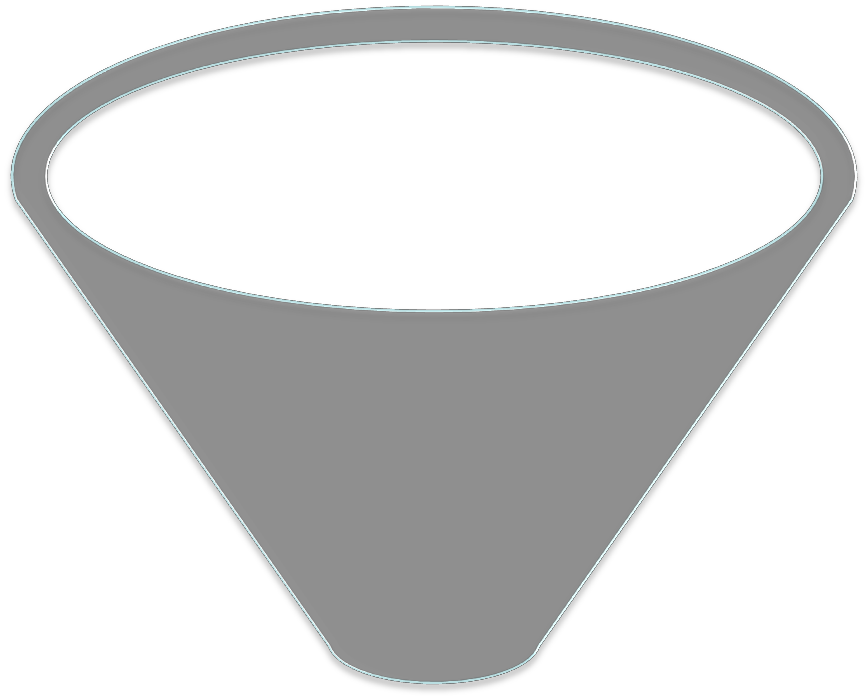
10. Delivery for the internal, application and   
 customer care testing

11. Documentation update

9 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

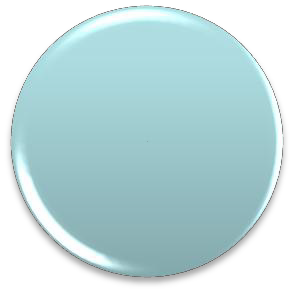
There are many Development Platforms





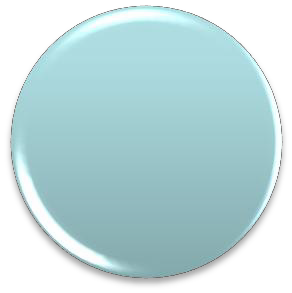


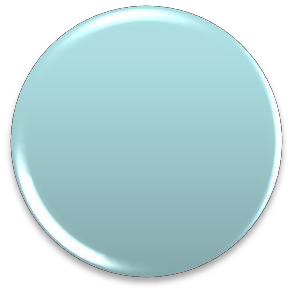






InstallScript, Framework





MSI, VBScipt

MSI, .NET

 Snapshot technology; Prism Deploy

 MSI Windows Installer in general

 InstallShield 11-2011, VBScript, InstallScript, now mostly WIX DTF

 Visual Studio 2008, 2010, .NET 2.0

– 4.0

 Highly fragmented proprietary installation framework as a basis used by consultants based on

InstallShield/VBScript

**Packaging**

10 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Windows Installer XML



 WiX was developed by Microsoft software design engineer Rob Mensching

 Released to SourceForge in April of 2004 – first Microsoft open- source project

 Developers use their free time to participate in the development

 Everyone can join the project – code changes submitted via email

 After some successful changes –

direct access could be provided

11 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

InstallShield



 Since 1992 InstallShield was

developed by Stirling Technologies

 Now InstallShield developed by Flexera Software owned by Thoma Cressey Bravo private fund

 Known industry standard

 Formerly InstallScript solution

 Successfully implemented support for Windows Installer technology

 Widely used by our Customer in

the most of the projects

 Now Limited Edition shipped with

Visual Studio 2010

12 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

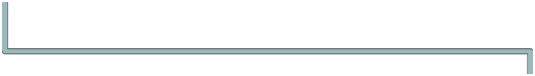
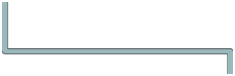
WIX – Current Use Case



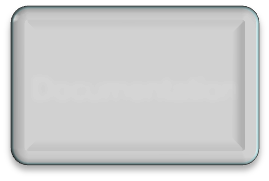


Windows

Installer XML







Visual Studio

Shell

Command line tools

Documentation

Third party tools





Votive for Installation Package Design

DTF C# Custom Actions

Generate components by Heat tool

WYSIWYG user interface development

13 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

InstallShield – Current Use Case







InstallShield





Native User

Interface

Visual

Studio

Bundled

Tools







Installation Package Design

VB Script Custom Actions

Context sensitive help

Generate

components

DTF or .NET Custom Actions

Compare two MSI packages

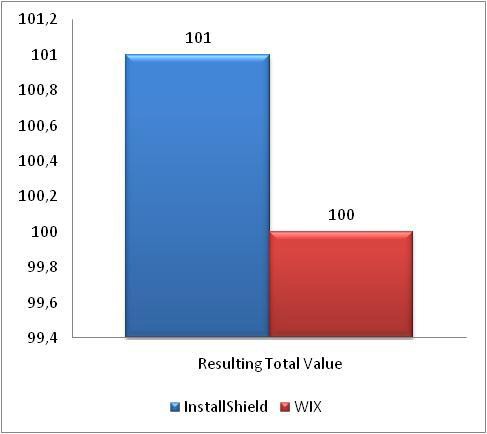
Create build

server

14 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

WIX vs. InstallShield – Summary





 InstallShield

 Globally InstallShield still standard

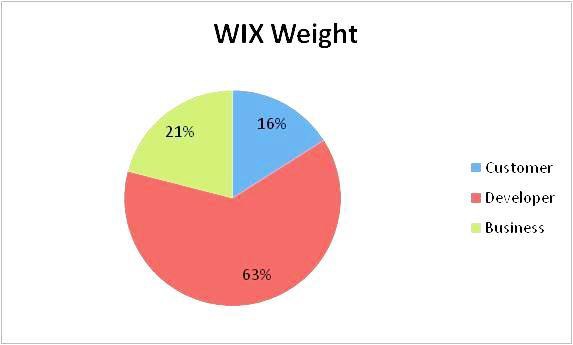
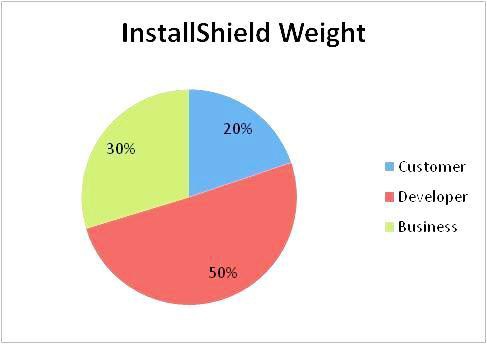
solution for the market

 More predictable, supported and controllable product

 Development is not too complicated, best installation visualization

 Very expensive

 Windows Installer XML

 Some concerns in regard of the customer orientation

 Business value is questionable, since the product life cycles and features hard to control or predict

 Best solution from developer’s point of view and a key of success

 Free

15 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

As the result, compromise…



**InstallShield 2011**

&

**WIX Deployment Tools Foundation**

 Cheaper than the Premier Edition

 Concurrent licenses

 Multi-lingual support could be made in external UI

 InstallShield UI allows fast visualization and editing

of the MSI package

 WIX DTF and C# code handles most of the custom actions needed, especially for Silverlight and .NET based ones

 Now InstallShield bundled with the Visual Studio

16 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Automation Pros & Cons



 When do you automate?

 Stable and mature project structure

 Well documented installation process

 Installation functional, delivered and approved by the customers

 Frequent updates are required

 Rare customer specific releases

 Where is it not needed?

 Requirements are not even or stable

 Frequent and significant application

design changes

 Rare updates, slow development

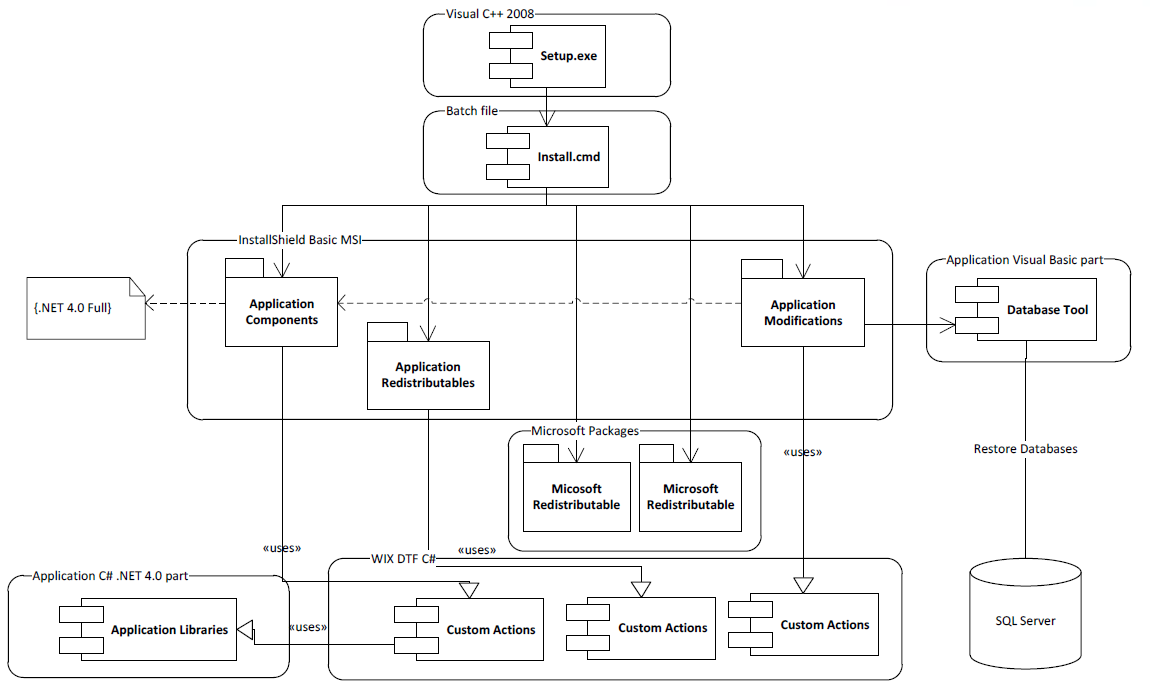
 Highly variable customer demands

 Customer specific deliveries

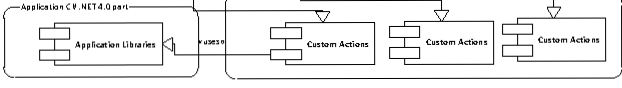
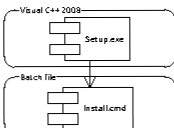
 Process is not documented

 Limited resources or budget

17 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Experimental Design Pattern  






18 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Experimental Design Pattern



 Two packages per product



 Components for the files

 Mods for the environment config.

 Easy to localize the issue

 Faster to solve configuration related errors

 Reusable assemblies from the

application

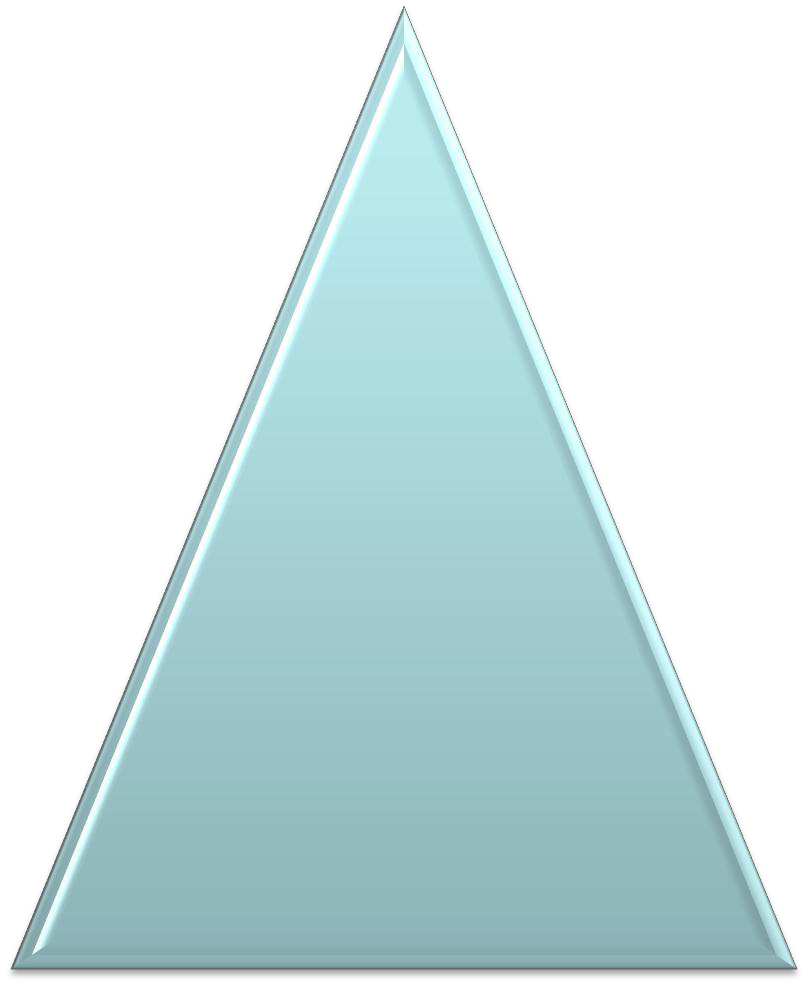
 Functional template based on the InstallShield and WIX DTF, ready for the build automation

 Automated documentation processing

19 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Future of Packaging







.NET

launcher

MSI

packages

WIX DTF

 InstallScript out of the market

 Windows Installer user interface will be hidden by custom .NET based solutions

 Silent mode used more and more

 Multi instance design as the requirement for growing ASP markets

 No VB or InstallScript inside the installation - .NET, C# and WIX DTF based solution

 Integration with the application assemblies

 Flexible build automation that will handle most of the customer related cases

 APPX and Windows Marketplace?

20 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

****



**...So, Don't be Just a Setup Monkey  
Monkey see, monkey do – you can do better than that! You can be better than that!**



21 © Copyright 1993-2011 MrNetTek http://eddiejackson.net

Packaging Tools you could be using  
  
WinInstall, Flexera Admin Studio, MSI Pro, Prism Deploy Pictaker.exe  
  
VBScript - VBSEdit  
  
Notepad++  
  
Script-o-matic  
  
HTA-helpomatic  
  
WMI Explorer  
  
PrimalScript  
  
Audacity and Gimp



22 © Copyright 1993-2011 MrNetTek <http://eddiejackson.net>









That’s only the beginning…

23 © Copyright 1993-2011 MrNetTek <http://eddiejackson.net>

Future Topics  
  
  
Integrating Shell scripting into your packages  
  
Integrating VBScript scripting into your packages  
  
Integrating HTA scripting into your packages  
  
Adding splash screens to your packages  
  
Adding sounds to your packages  
  
Adding special effects to your packages  
  
Dealing with complex issues: permissions, registry, and multiple profiles



24 © Copyright 1993-2011 MrNetTek <http://eddiejackson.net>