NetBeans and Eclipse

Why to use a framework...
Why to use...

Let's create our own framework!

Maintaining the framework takes a lot of my time :(
Why to use...

Don't reinvent the wheel!

Math professor Stan Wagon demonstrates his square-wheeled bicycle at Macalester College in St. Paul, Minn. In 1960, it was discovered that a square wheel would roll smoothly on a road made of catenaries (those bumpy things). Wagon said he became interested in the concept 7 years ago, did calculations and computer animations, then had the bike specially built.
Why to use...

- Do not reinvent the wheel!
- Code reuse
- Existing libraries
- Modular applications
- Focusing on application logic only
- …
Fiorano Studio: Lego Blocks for SOA
Nokia NetAct: Mobile Network Mgr
TidalWave BlueMarine: Photo Manager
IBM Lotus Expeditor

ITSO Weather Portlet - Lotus Expeditor

Customer Information
- Name: Coquero, Alex
- Street: 825 Workbench Drive
- City: Raleigh, 27621, NC
- Phone: 434 422 24032
- Email: alexbc@br.ibm.com
- Card: 82-9876 5432 1010

Car Information
- Make: Brown
- Model: Lima
- Color: Silver
- Slot: 86
- Rate: 421.0

Booking Information
- Start date: 1/4/07
- End date: 1/4/07
- Amount: 421.0

Select Car Portlet
- Photo: Brown
- Manufacturer: Lima
- Color: Silver
- Shift: Automatic
- Rate: 421.0

Card Information
- B2
- 9876 5432 1010

Customer Contract History

ITSO Weather Channel
- Today: Occasional Light Rain
- High: 69F, Low: 63F

- Tomorrow: Likely Light Rain
- High: 66F, Low: 52F

- Thursday: Mostly Clear
- High: 71F, Low: 58F

- Friday: Medium Inclement
- High: 56F, Low: 51F
Patterns for UI

MVC, DCI
MVC

- Model-View-Controller

• Data-Context-Interactions
  - solves the issue
    • OOP is good for data modeling
    • but where to put operations?
      - e.g., a bank account and money transfer
Images source: http://www.artima.com/articles/dci_vision.html
NetBeans and Eclipse

Modular systems
Component systems
Přehled

• today, no application created from scratch
  – libraries, frameworks,…

• what if we want an application extensible by “third parties”
• what if we want distributed development

• objects are not enough
  – we need components (modules, plugins,…)

• components have explicitly defined requirements
  – objects have only provided services
• components are implemented via objects

• there are a huge number of component systems
Components vs. NetBeans & Eclipse

• NetBeans – moduls
• Eclipse – plugins

• a core of the system = a container for components
  - provides
    • management of components
    • management of dependences
    • versioning

• NetBeans – an own system
• Eclipse – OSGi

• since 6.9 NetBeans can run over OSGi
  - NB modules ~ OSGi bundles
  - some Eclipse plugins (without UI) can be used in NB
Componenty vs. NetBeans & Eclipse

- components kinds (in NB and Ecl)
  - regular
  - libraries
    - a wrapper around existing JAR package
- both NB and Ecl allows creation of
  - IDE extensions
  - standalone applications
    - no need to be an IDE
    - an application is a set of components (plus the container)
NetBeans vs. Eclipse

- main differences (roughly)
  - component container
    - NB – own one
      - but OSGi can be used
    - Eclipse – OSGi
  - UI
    - NB – Swing
    - Eclipse – SWT
  - support
    - NB – Apache (previously Oracle, previously Sun)
    - Eclipse – Eclipse foundation (previously IBM)