

p2

The new Eclipse provisioning system

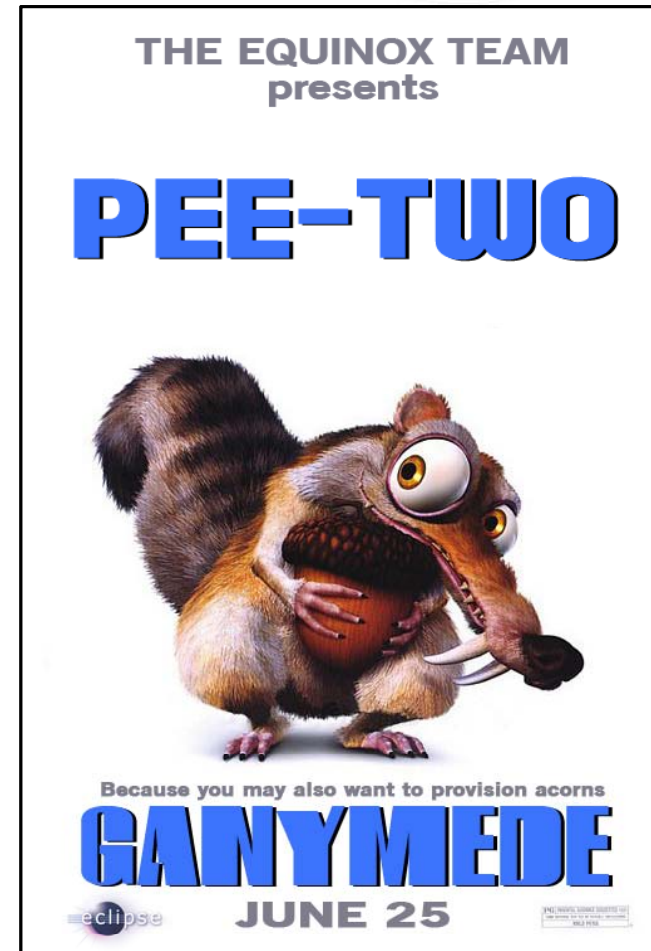
Benjamin Cabé

Anyware Technologies

<benjamin.cabe@anyware-tech.com>



- p2, what's that?
 - ▶ Origins
 - ▶ Concepts
 - ▶ Architecture overview
- Using p2
 - ▶ As a developer
 - ▶ As an ISV
 - ▶ As an end-user
- Coming next



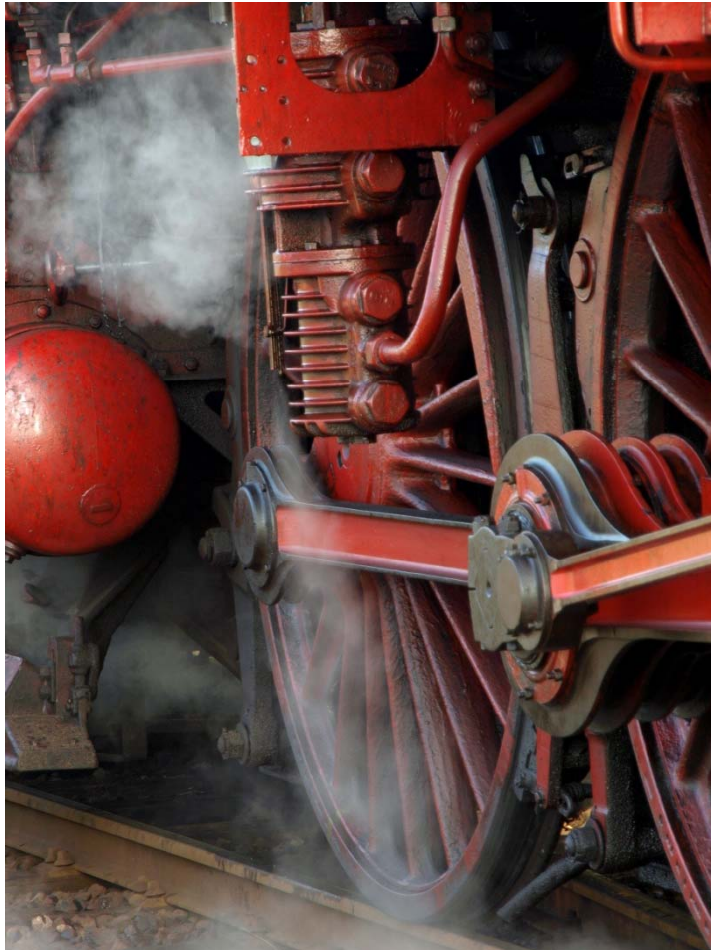
- Assessment
 - ▶ The plug-in mechanism and the level of modularity of Eclipse/OSGi systems is great...
 - ▶ ... but managing hundreds of plug-ins/components is really hard
- Since Eclipse 2.1: the « Update Manager »
 - ▶ allows to install & update « features » through GUI...
 - ▶ ... which is way too messy for 99% of users



Update Manager's limitations

- The feature doesn't fit all needs...
 - ▶ deliver a product in different but very similar shapes?
 - ▶ install more than just plug-ins & features?
- ...neither does the **update site**
 - ▶ ship features through different mirrors?
 - ▶ allow the end-user to perform the « initial » provisioning?





- Reworked UI
 - ▶ simplified workflow
 - ▶ new metaphors
 - ▶ better performances
- Provision the world!
 - ▶ plug-ins, .exe, .ini, ... whatever
- A very modular framework
 - ▶ provision more than just Eclipse stuff
 - ▶ controllable from the outside

“If it can be installed, it runs!”



- Optimized updates & installs
 - ▶ multithreaded downloads
 - ▶ best mirror automatically picked
- “Bundle pooling”
 - ▶ shared plug-ins across many Eclipse instances
- Low-level provisioning APIs
 - ▶ headless mode
 - ▶ update a non-running instance
- Installer

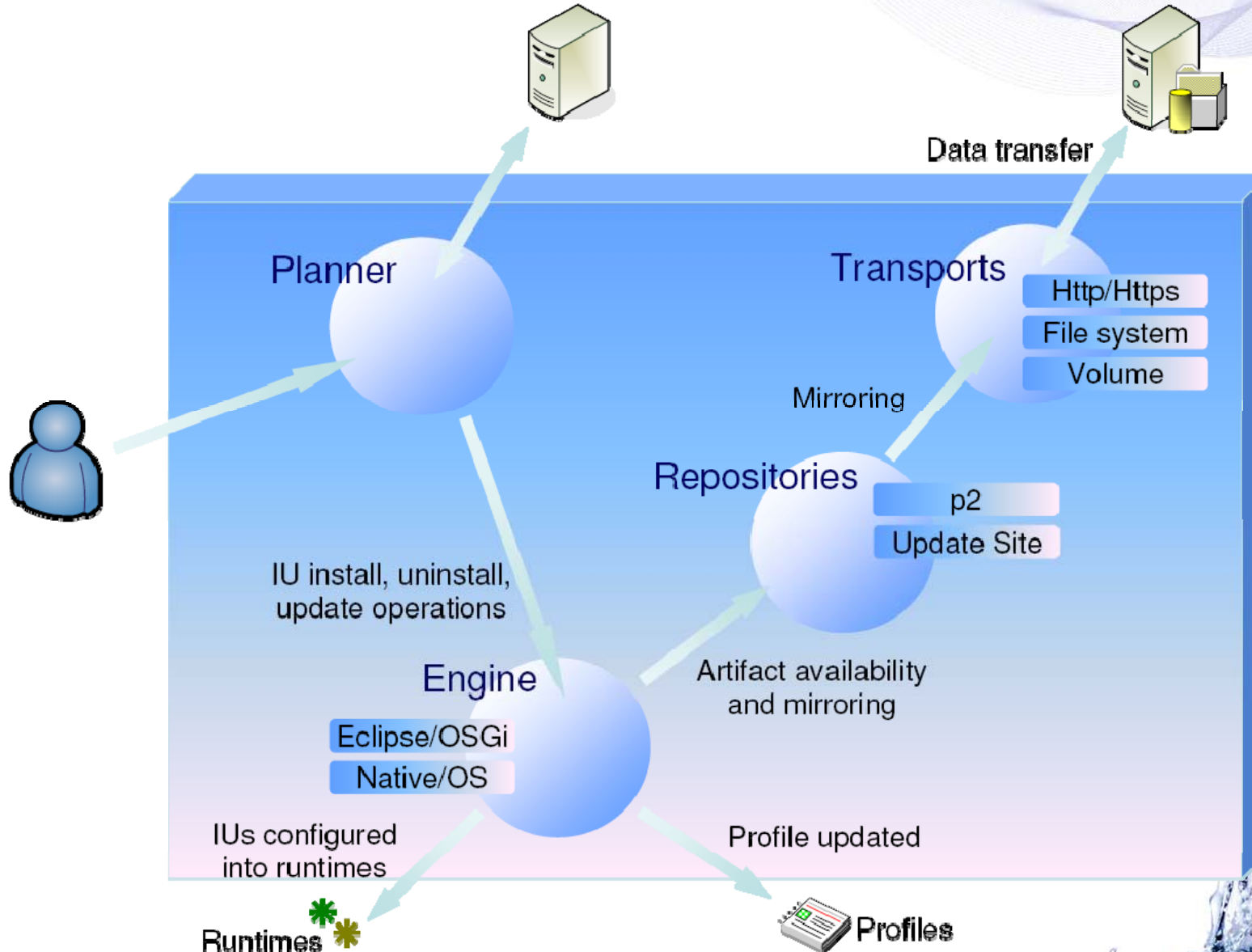


- Installable Unit :
 - ▶ the description of what is being installed
 - ▶ ID, version
 - ▶ dependencies
 - ▶ actions to perform during install / uninstall / ...
 - ▶ physical artifacts (optional)



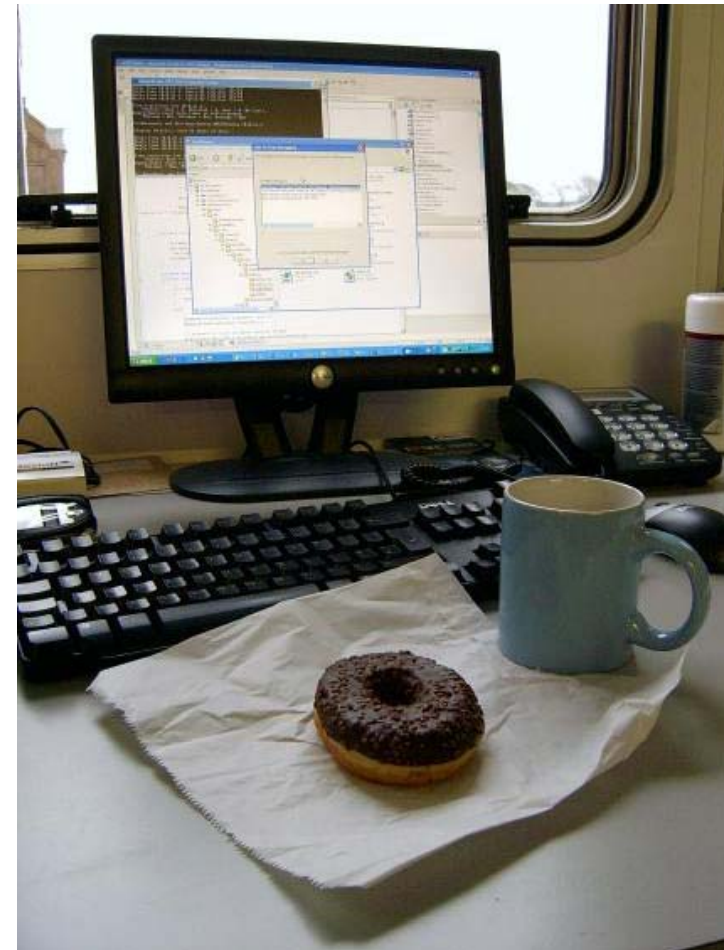
- Installable Units are stored in a metadata repository
- The actual content (plug-ins, zips, ...) lives in an artifact repository





Using p2... as an Eclipse developer

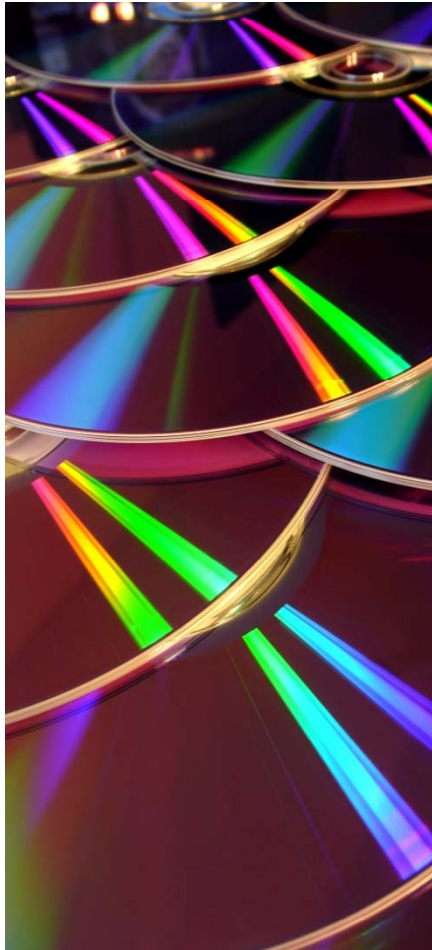
- Save hundreds of Mb thanks to bundle pooling
- Describe a well-defined, reproducible, development environment
- Use the dropins folder to perform quick installs
- Manage target platforms elegantly



- EPP Wizard
 - ▶ Configure your own custom Eclipse on the web and install it using p2
 - ▶ <http://build.eclipse.org/eppwizard/go>
- Buckminster
 - ▶ A set of tools for automating build, assemble, and deploy
 - ▶ Can produce p2 metadata
 - ▶ Handle other version types than OSGi
- IAM
 - ▶ Maven integration is on the plan!



Using p2... as an ISV



- Deliver an installer on a CD or an USB key
- Propose different flavours of your product depending on the customer
- Deploy a mirror in your customer's information system



- Get automatic updates
- Enjoy a clean uninstalling process
- Get annoyed by some bugs or missing documentation but...
- ...Just relax!



“I've been told that p2 is    ”

- There are been lots of improvements since 3.4
 - ▶ performance
 - ▶ robustness
- If something looks buggy, you must take time to report the issue
 - ▶ https://bugs.eclipse.org/bugs/enter_bug.cgi?product=Equinox
 - ▶ BTW, do you remember the install/update nightmare before 3.4?



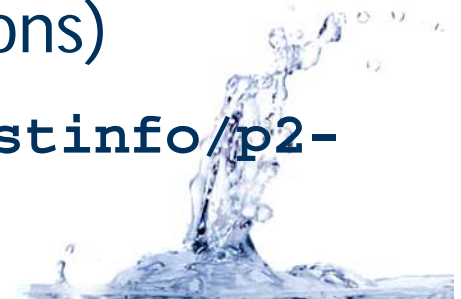
- A real p2 API
- User interaction
 - ▶ improve progress/error reporting
 - ▶ tighter desktop integration (shortcuts, registry keys, ...)
- Tooling / Authoring
 - ▶ IU editor
 - ▶ Repository editors (artifact & metadata)
- IBM, Cloudsmith, Siemens AG, Genuitec, Anyware Technologies... **YOU!**



- For RCP and Equinox based software (> 3.4)
- To take advantage of new possibilities
- Distribution through p2 repository
 - ▶ Training “P2-enabling an application”
 - ▶ Audit/Expertise/P2-Integration
- Manage configurations and versions
- Easily administrate p2 repositories
 - ▶ Tools coming soon, and related services



- The Eclipse wiki:
 - ▶ http://wiki.eclipse.org/Equinox_p2
- The p2 newsgroup (general issues)
 - ▶ <news://news.eclipse.org/eclipse.technology.equinox>
- The p2-dev mailing list (technical questions)
 - ▶ <https://dev.eclipse.org/mailman/listinfo/p2-dev>





Thank you!

Questions?

Benjamin Cabé Eclipse Expert

`<benjamin.cabe@anyware-tech.com>`

`http://blog.benjamin-cabe.com`

`http://www.anyware-tech.com`





anyware
TECHNOLOGIES

A **wavecom** COMPANY

Linking Innovation to your Business



- Michal Zacharzewski
 - ▶ <http://www.sxc.hu/photo/953139>
- G Schouten de Jel
 - ▶ <http://www.sxc.hu/photo/1090308>
- Andrej Sevelin
 - ▶ <http://www.sxc.hu/photo/892243>
- Dave Di Biase
 - ▶ <http://www.sxc.hu/photo/204676>
- dunkv@flickr
 - ▶ <http://www.flickr.com/photos/dunkv/97119622>
- Cathy Kaplan
 - ▶ <http://www.sxc.hu/photo/283374>
- Mac Pale
 - ▶ <http://www.sxc.hu/photo/956527>

