CBI Build the Eclipse Platform

- **Before CBI**: PDE build knowledge & skills weak outside IBM

  “the SDK build is quite big and takes long even without running any tests. This will give you a loooong time to pray ;-).” - Dani Megert

- **After CBI**: Build based on Maven, & Maven is widely used

  Just 3 commands / seconds of effort / ~7 to 30 minutes duration

Projects using CBI:
Eclipse Platform, CDT, EGit, JGit, Mylyn, WTP (in progress)

CBI Eclipse platform build:
Looks easy now, but was it easy to do?

So far, it took:

- 1500 hours of reverse engineering, design, testing, deployment
- Nearly 1K pom.xml files, virtual build & LTS infrastructure, new on-demand web signing service, Tycho enhancements, much communication

What's left to do?:

- Need $30K of hardware for build/LTS
- Roll-out to other projects
- Signing for MacOS, Signing for Windows
- Badly need a proper software distribution system (Nexus 2.0 based)
- Badly need user interfaces for LTS (considering Code2Cloud)

Why do this?

- Dramatically cheaper/faster/better than each project inventing their own.

Long Term Support

LTS Working Group

- Co-Chairs have been selected:
  - Pat Huff (IBM) & Jochen Krause (EclipseSource)
- IWG agreement & Charter finalized, approved
- Launch planned & first meeting on June 29th
- Ongoing recruitment of LTS members

Software

- CBI 1.0 roll-out to interested projects underway
- CBI 2.0 planning has begun
- First LTS release – Juno
- Code sprint in July

LTS Serious Challenges

Very positive feedback, but precious few buyers.

Most popular reasons given:

1) They rebase to the latest Eclipse release each year, are confident in backwards compatibility, and thus have little need for LTS

2) They are already incurring the cost to provide internal self-service LTS for historic Eclipse releases. Juno (the first LTS release) represents a fraction of this budget.

3) They are interested, but will wait and see how things develop. Maybe in a couple of years.

4) They already have committers on the projects they care about.