



# DSDP Target Management 3.0 Release Review Minideck

*Eclipse Development Process version 2.3.1 – January 17, 2007*  
*Slide deck v1 – May 27, 2008*

Martin Oberhuber, Wind River  
TM Project Lead

# DSDP-TM 3.0 Features



- TM 3.0 New Features
  - Target Communication Protocol Framework (TCF) (incubation, contributed)
  - Windows CE RAPI wrappers and RSE Subsystems (incubation, contributed)
  - RSE Terminal Integration (incubation, contributed)
  - RSE User Actions and Compile Commands
  - RSE Import / Export of connections and profiles
  - Reduced plugin activation, improved startup behavior and performance
- API Quality:
  - Extensive use of API Tooling to get rid of API Leakage and document usage restrictions;
  - Continued API Review, documentation, refactoring
    - Without examples & tests: 935 API types / 1476 non-API (2.0: 828 API, 986 non-API)
  - Many Unit tests added
- TM 3.0 project size
  - RSE: 346 kLOC + TCF: 97 kLOC (R2.0: 290k, R1.0: 242k)

# *DSDP-TM 3.0 EOL and Community*



- Non-Code Aspects
  - Full Documentation, Tutorials, FAQs, Example Code, ...
  - Detailed build notes with migration notes on each milestone
- End-of-Life issues:
  - Some API Refactoring (IFileService), but no EOL'd APIs, tools, concepts or components in 3.0
- Community and Committer Diversity:
  - 11 committers (5 WindRiver, 4 IBM, 1 ProSyst, 1 private) – was 8 in 2.0
  - 2 former committers, 23 additional contributors – was 5 in 2.0
  - 1 Google Summer of Code project this summer
  - Commercial adoption by at least 13 companies. Involvement with other Eclipse projects (Platform/Team, ECF, CDT, Orbit, EPP, PDT, Babel).
  - Well known and respected in the Community, part of JEE package
- Bugzilla
  - 3.0 Stream: 441 issues fixed, 671 open (2.0: 572 fixed / 301 open)

# *DSDP-TM 3.0 Process and Architecture*



- Process
  - Full process docs on the Web; adopting Modeling build for Releng
- IP Clearance and Licenses:
  - All licenses and about files are in place as per the Eclipse Development Process, the Due Diligence Process was followed for all contributions.
- Architectural Issues
  - Legacy code still not fully cleaned up – much Platform “internal” access
    - Need more UI / Non-UI separation for headless and RCP usage
    - Need more Unit Tests (hard for UI-heavy parts)
  - Overlaps with other projects - Many remote access APIs
    - E.g. Remote File Service – 5 APIs: Platform EFS, ECF fileshare, TPTP Agent File Interfaces, TCF, RSE IFileService
    - Talking with all those projects; absorbing / bridging
  - “Remote Development” effort is disconnected at IBM / PTP
- Future:
  - Likely shooting for TM 3.1 next year – Focus on Multicore, TCF