

# Target Management New and Noteworthy

Martin Oberhuber, Wind River

[www.eclipse.org/dsdp/tm](http://www.eclipse.org/dsdp/tm)

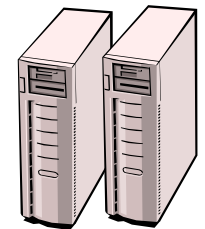
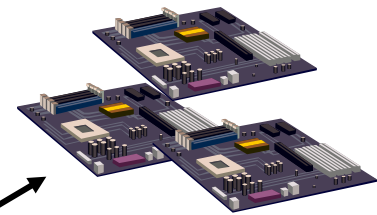
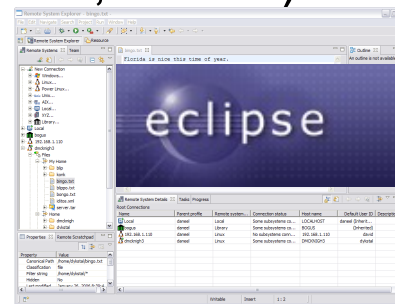
## The Eclipse Target Management Project

... why “Target”?

- ◆ Just a matter of terminology

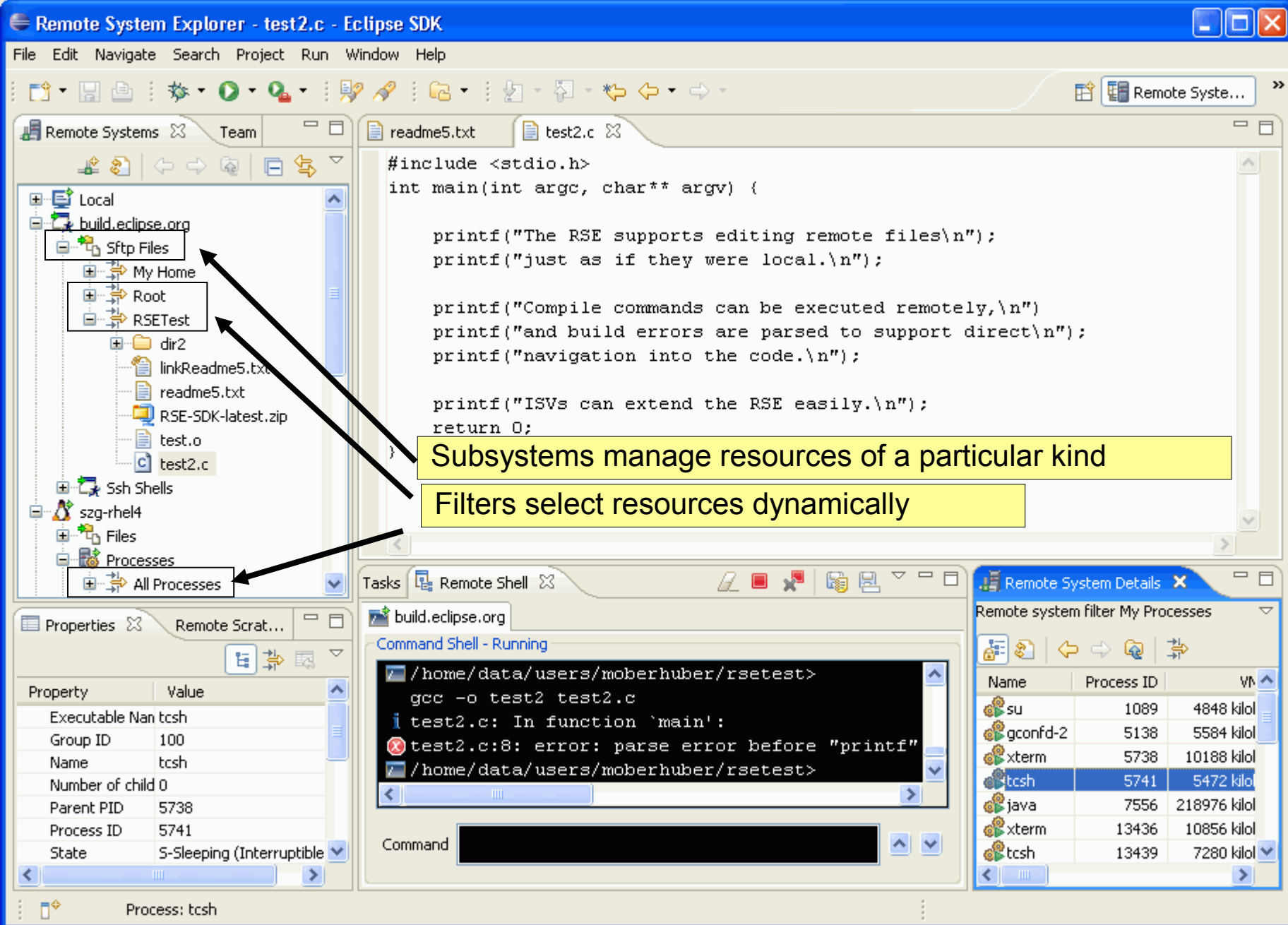
### Remote Computer Systems

- ◆ Targets (Locally connected, shared, fielded)
- ◆ Hosts (Grids, farms, nodes) and running software on them
- ◆ Discover, connect, get status
- ◆ Download, run, debug, test



... why “Management”?

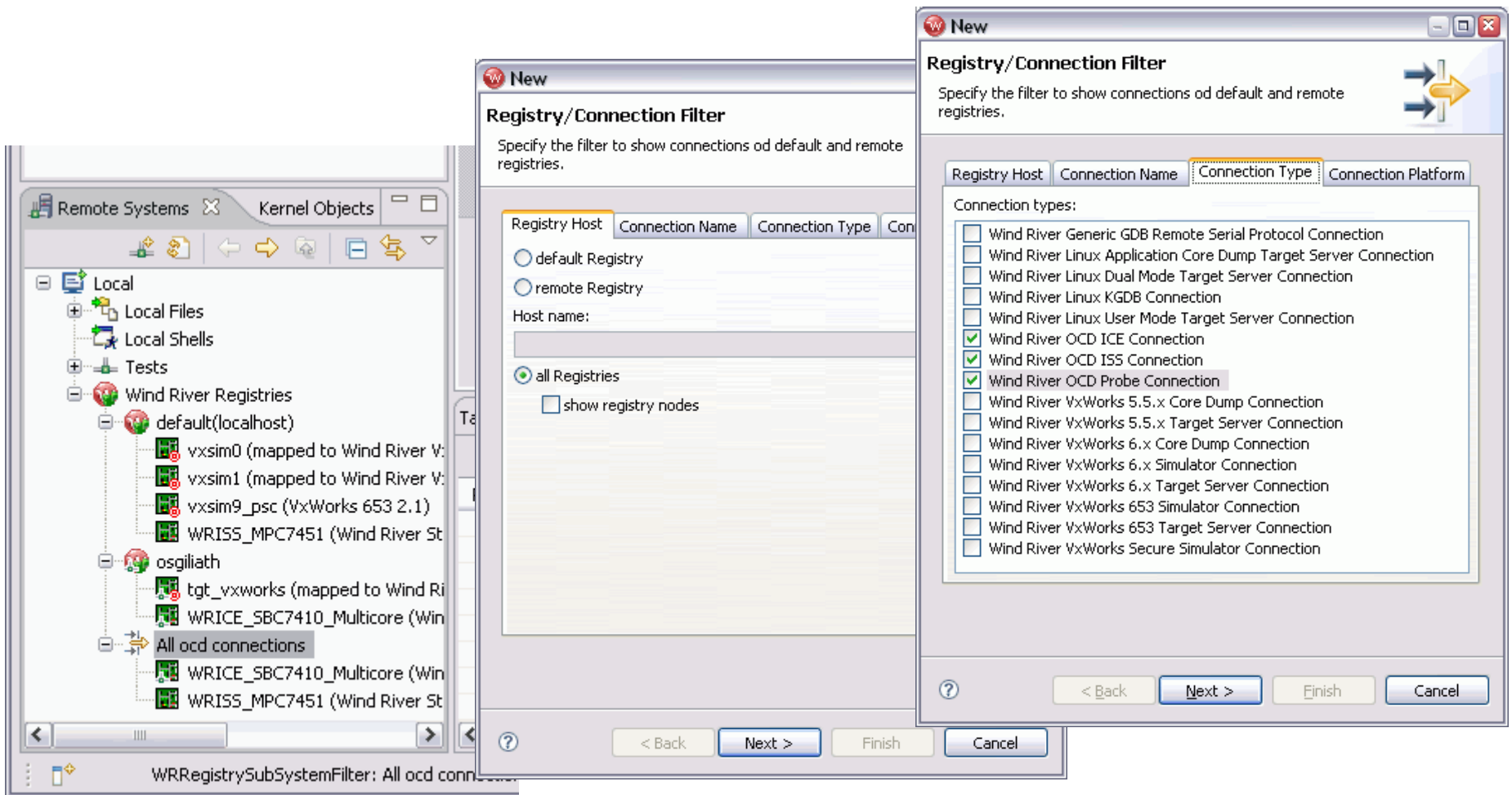
- ◆ Discover remote systems; manage their properties and capabilities; team-share connection definitions and settings; access control



## Remote System Explorer (RSE)

- **Integrates any kind of heterogeneous remote resources under a uniform UI**
- Pluggable subsystems and adapters map any kind of existing model onto the RSE concepts
- Default subsystems:
  - ◆ Remote Files - **transparent working on remote computers** just like the local one
  - ◆ Standard Widgets and Dialogs, EFS Provider
  - ◆ Remote Shell, Remote Processes
- Deferred access in background jobs everywhere
  - ◆ Can integrate with other providers e.g. ECF

# TM for Embedded: Wind River Workbench



## Symbian phone browser

The screenshot displays the Eclipse IDE interface for developing on a Symbian phone. The top toolbar includes File, Edit, Navigate, Search, Project, Run, Window, and Help. The main workspace is divided into several panes:

- Remote Systems Explorer:** Shows a tree view of the Symbian device's file system, including folders like Local, Symbian device, Symbian Processes, STAT Console, Symbian FTP, Root, and various system folders (DCIM, logs, Media Files, Private, Resource, Shared, sys, system, Apps, data, nldp2, Opera).
- Opera.ini:** A text editor showing the configuration file for the Opera browser on the device. The content includes:
 

```
Opera Preferences version 2.0
; Do not edit this file while Opera is running
; This file is stored in UTF-8 encoding

[User Prefs]
Special effects=0

[OEM]
Operator Cache Size=0
Operator Cache Designator Header=
Operator Cache Designator Header Value=
Never Flush Trusted Servers=

[RM2]
iua=0

[State]
Run=0
```
- Outline:** Displays a preview of the Symbian UI, showing a "Main menu" with a "Messaging" option and several application icons (browser, camera, games, calculator, folder, calendar, notes, address book, key).
- STAT Console:** A terminal window showing the output of the STAT command:
 

```
STAT CONSOLE v0.2
> <B><D><E>
JSTAT Result:
- Retrieved Data:
STAT Version 1.1.1003
EMachineUid: Not available
ECPU: ARM
ECPUABI: THUMB
EDeviceFamily: Not available
EManufacturer: Not available
EModel: 1346234015
EDeviceFamilyRev: 0
EManufacturerHardwareRev: 8196
EManufacturerSoftwareRev: 4
EManufacturerSoftwareBuild: 0
EMemoryPageSize: 4096
EMemoryRAM: 63963136
EMemoryRAMFree: 13373440
EMemoryROM: 3145728
```
- Properties/Remote System Details:** A table listing running processes on the device:
 

Name	Process ID	Executable Name
ekern.exe[100041af]0001	1	Z:\sys\bin\ekern.exe
efile.exe[100039c3]0001	11	Z:\sys\bin\efile.exe
Estart.exe[10203113]0...	16	Z:\sys\bin\Estart.exe
domainSrv.exe[1020e4...	20	Z:\sys\bin\domainSrv.exe
DebugRouter.EXE[1020...	22	Z:\sys\bin\DebugRouter.EXE
IAPS[10200b1+][10203...	26	Z:\sys\bin\iaps.exe
C32start[101f7988]0001	28	Z:\sys\bin\c32start.exe
c32exe.exe[101f7989]...	30	Z:\sys\bin\c32exe.exe
centralrepositorysrv.exe...	32	Z:\sys\bin\centralrepositorysrv.exe
EComServer.exe[10009...	41	Z:\sys\bin\EComServer.exe
102030f2...	44	Z:\sys\bin\EventSrv.exe

# TM for Enterprise: IBM WebSphere Developer

The screenshot displays the IBM Rational Software Development Platform interface. The top window is titled "Remote System Explorer - INDENT2.RPGLE - IBM Rational Software Development Platform". It features a menu bar (File, Edit, Source, Compile, Navigate, Search, Project, Run, Window, Help) and a toolbar with various icons for file operations and development actions.

On the left, the "Remote Systems" tree shows a connection to "My System i Connection" containing several "iSeries Objects". The "INDENTMAX1.rpgle" file is selected and highlighted in blue.

The main editor window shows the code for "INDENT2.RPGLE". The cursor is positioned at Line 70, Column 43. The code includes comments and data statements:

```

Line 70      Column 43      Replace
.....CLON01Factor1++++++Opcode (E)+Extended-factor2+++++++
005800      *
005900      * MOD --PROGRAMMER-- MM/DD/YY -----CHANGE-MADE-----
006000      * 001 X. Xxx          mm/dd/yy
006100      *
006200      *
006300      *
006400      *
006500      * VAR 01 - DOU
006600      *
006700      C              DOU          a = b
006800      C              DOU (m)       a = b
006900      C              DOU (r)      a = b
007000      C              DOU          a + b + c
007100      C              =
007200      C              d + e + f
007300      C              DOU (m)      a * b
007400      C              = c * d
    
```

At the bottom, the "Remote System Det..." tab is active, showing a table of "My active jobs":

Name	User	Number	Status	Subsys
145957/QUSER/QZRC5RV5	QUSER	145957	*ACTIVE	QUSRW
145974/QUSER/QJVACMSRV	QUSER	145974	*ACTIVE	QUSRW

## RSE 3.0 Plan Items

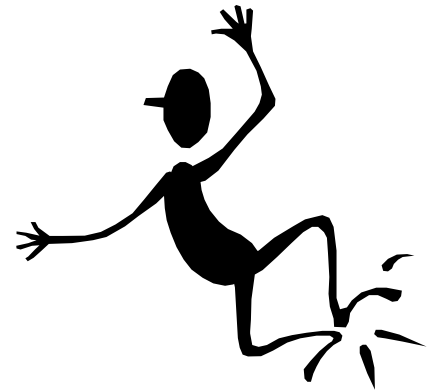
- ✓ Improve **Quality**, Robustness and Unit Test Coverage
  - Pick up UI Guidelines
- ✓ **Componentize** and Scale Down:
  - ✓ Avoid unnecessary bundle activation
  - Support Headless Operation
- ✓ **Team support**: Import/Export of Profiles
- ✓ Contribute **User Action** support





## New RSE 3.0 Goodies

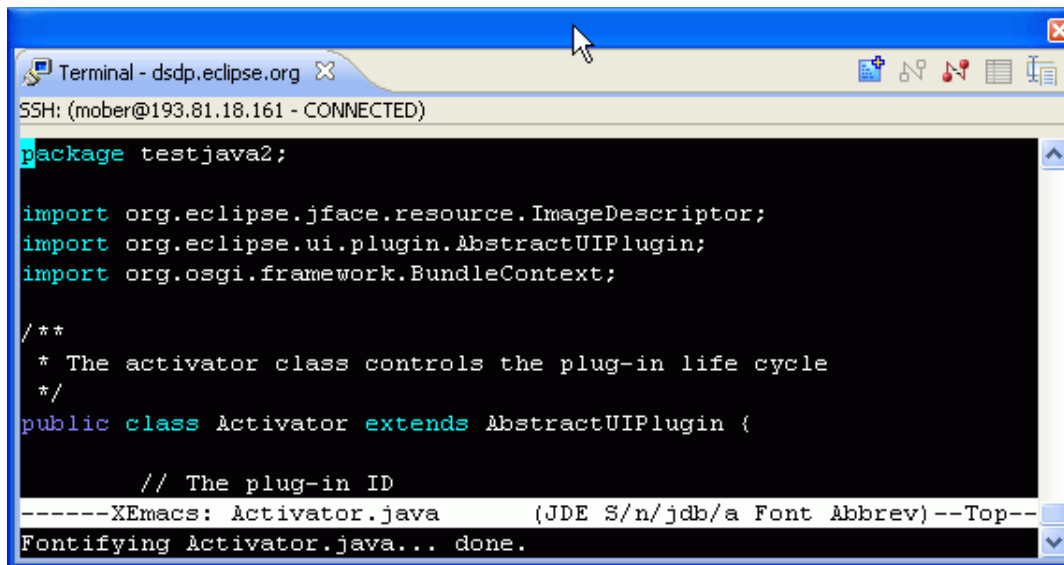
- Remote File Access
  - ✓ Tar.gz archive handler (contributed)
  - ✓ Windows CE file subsystem (contribution pending)
  - ✓ UNIX permission, owner and group support
- ✓ Link with Editor
- ✓ SSH Keepalive
- ✓ FTP Recursive Delete



... but TM is much more than RSE!

## TM Terminal

- Fast ANSI Terminal emulation
- Pluggable connectors for SSH, Telnet, Serial
- Optional editable input line for dumb terminals
- Lightweight Widget easy to port even for eRCP



```
Terminal - dsdp.eclipse.org
SSH: (mober@193.81.18.161 - CONNECTED)

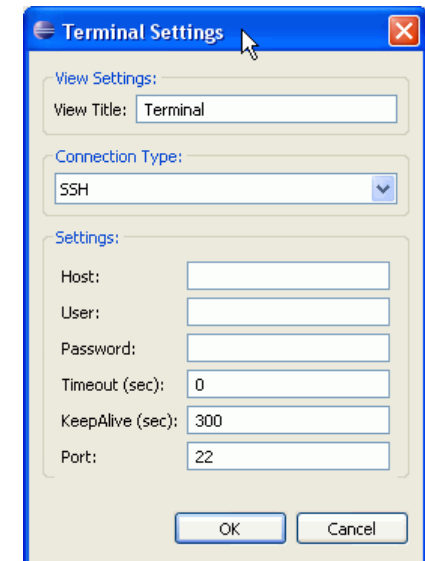
package testjava2;

import org.eclipse.jface.resource.ImageDescriptor;
import org.eclipse.ui.plugin.AbstractUIPlugin;
import org.osgi.framework.BundleContext;

/**
 * The activator class controls the plug-in life cycle
 */
public class Activator extends AbstractUIPlugin {

    // The plug-in ID

    -----XEmacs: Activator.java      (JDE S/n/jdb/a Font Abbrev)--Top--
    Fontifying Activator.java... done.
```



Terminal Settings

View Settings:  
View Title: Terminal

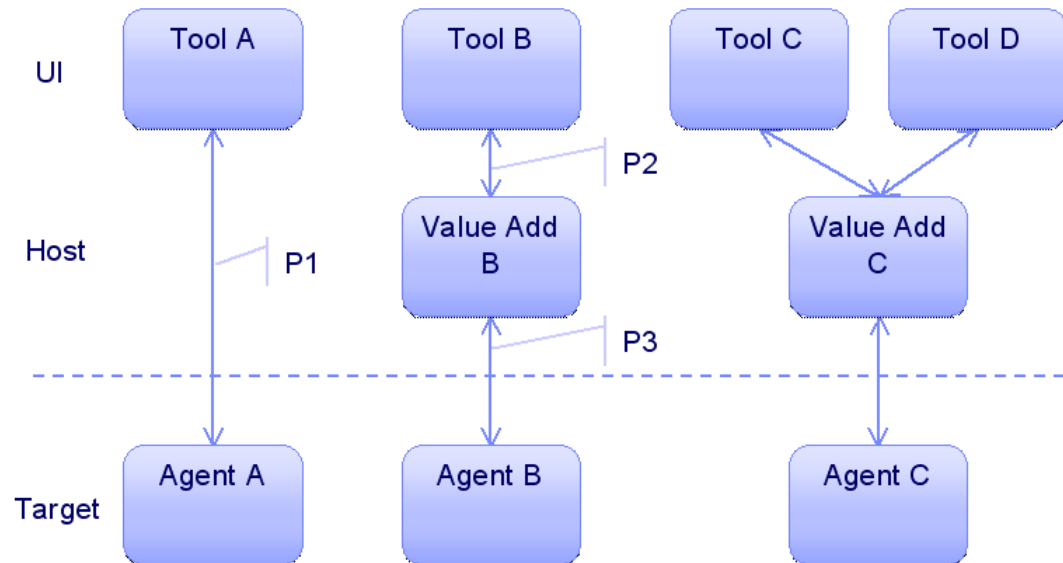
Connection Type:  
SSH

Settings:  
Host:   
User:   
Password:   
Timeout (sec): 0  
KeepAlive (sec): 300  
Port: 22

OK Cancel

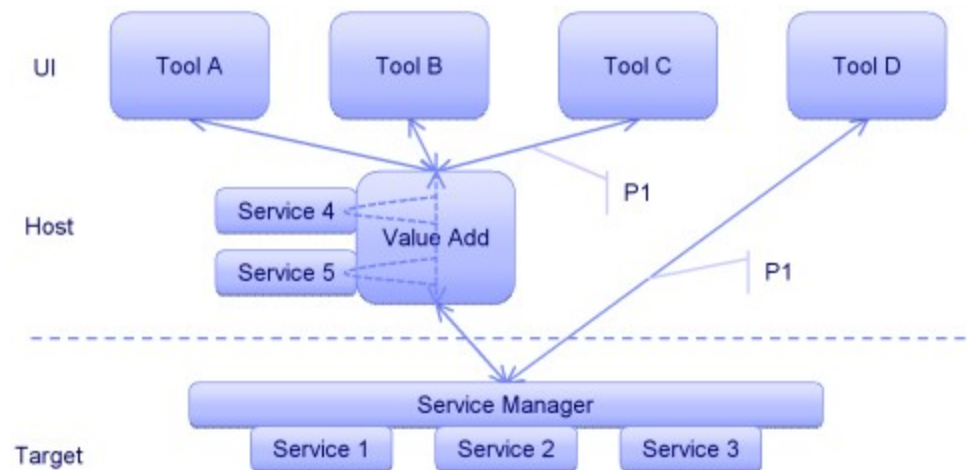
# Target Communication Protocol Framework (TCF)

- Background: Development tools need communication
  - ◆ Many tools, each typically using its own agent and communication method
  - ◆ Lots of overlap between these, e.g. how to communicate, retrieve/model target objects, manipulate target, etc



## TCF - Core Design Ideas

- Use the same simple, lightweight base protocol end-to-end, but allow value-adding servers
- Standard TCP/IP on the client, transport conversion by value-add (Serial, JTAG, ...)
- Auto-discovery of contributed services



## TCF and Eclipse

- **TCF** specifies the **protocol**, independent of API
  - ◆ Clients, agent and value-add in Plain C, Java or even Perl
  - ◆ Much work will be outside Eclipse IDE, e.g. gdb back-end
  - ◆ Leverage Eclipse brand, IP process and infrastructure
    - Most commercial embedded tools already on top of Eclipse
- **ECF** provides **abstract API**, independent of protocol
  - ◆ Good for standard clients like file transfer, messaging
  - ◆ A natural fit for TCF on the Eclipse Platform
  - ◆ ECF providers for TCF to be added soon

## TCF – Current Status

- ✓ Lightweight Plain-C Agent complete
  - ◆ Linux, VxWorks, Windows
  - ◆ Filetransfer, Monitoring (Process list), Basic Debugging
- ✓ Plain-C client and value-add examples
- ✓ Exemplary Eclipse Clients:
  - ◆ RSE Integration for Filetransfer, Process list
  - ◆ Platform Debug client
  - ◆ DSF Advanced Debug client
- ✓ Examples and Documentation
  - ◆ Getting Started, Protocol Specs, Context Identifier
  - ◆ “How to add a custom Service” – Daytime Example



## TCF Goals

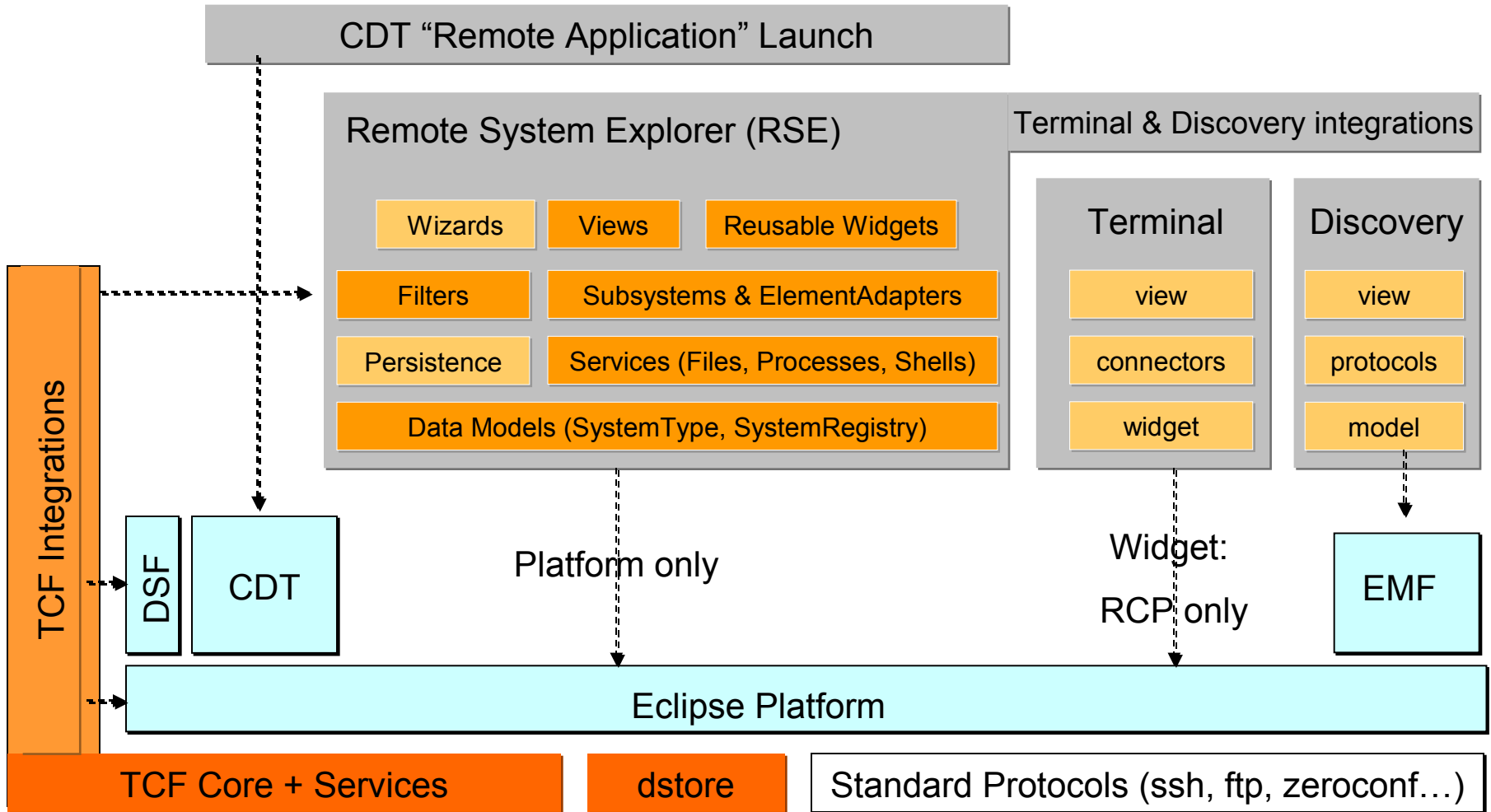
- Standardization effort driven at Power.org
  - ◆ Wind River, Freescale and others
  - ◆ **Join NOW** to get your requirements and use-cases in!
- Why bother with TCF?
  - ◆ Open your tooling for 3<sup>rd</sup> party value-add
  - ◆ Reduce maintenance with standard protocol framework
  - ◆ Get basic agent framework and tooling for free
- Code is available from Eclipse.org under EPL
  - ◆ EclipseCon Tutorial is your best getting started



WIND RIVER



# Target Management 3.0 Components





## TM Mission, Goals and Future

- **DSDP Mission:** *Create an open, extensible, scalable, and standards-based development platform to address the needs of the device (embedded) software market [...]*
- **TM Mission:** *Create data models and frameworks to configure and manage remote systems, their connections, and their services.*
- **Work in Progress (Technology Sub-Groups)**
  - ◆ Component-Based Launching (CBL)
  - ◆ Multi-core / Multi-target support through connection groups
  - ◆ Adapters for Target access control (shared board labs)
- **Ideas being discussed**
  - ◆ Connection Model for HW Debugging (SPIRIT, complex connector setup)
  - ◆ Flexible Target Connector framework, Connector plumbing algorithm
- See the TM Wiki, and the TM Use Cases Document  
[http://www.eclipse.org/dsdp/tm/doc/DSDPTM\\_Use\\_Cases\\_v1.1c.pdf](http://www.eclipse.org/dsdp/tm/doc/DSDPTM_Use_Cases_v1.1c.pdf)