



## *Building Applications for Multiple Platforms Using the Eclipse Rich Client Platform*

Wayne Beaton, Eclipse Foundation, Evangelist

Jeff McAffer, IBM Rational, Eclipse RCP and Equinox Lead

(Martin Lippert, akquinet agile GmbH, Equinox Incubator Committer)



# Agenda



- What is a rich client?
- What is Eclipse RCP?
- Equinox
- Building platforms
- Outlook



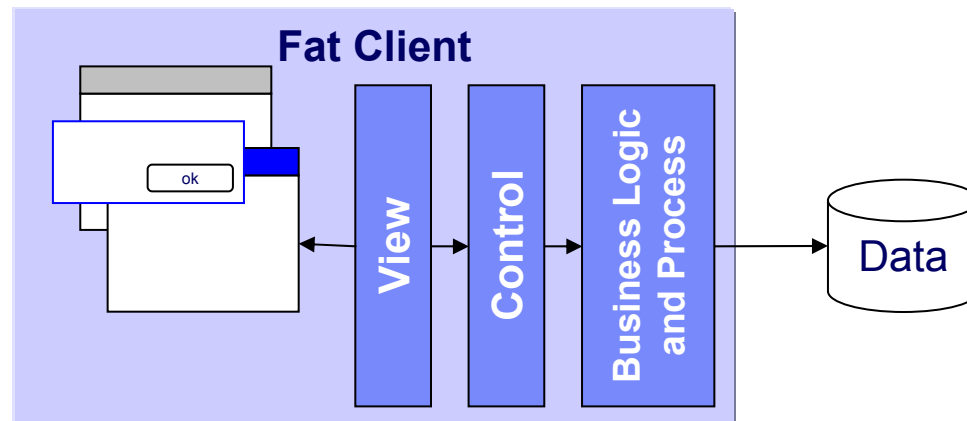


*What is a Rich Client?*



## *Evolution of the Client: Fat Client*

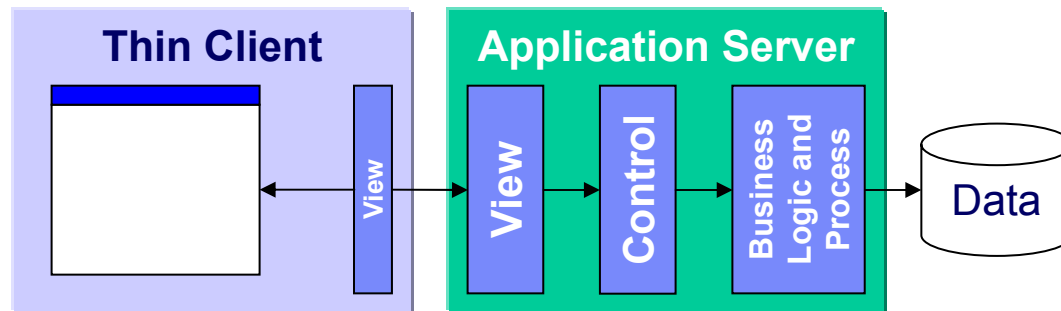
- All application logic on the client
- Relatively difficult to update
- Rich user experience
- Monolithic application
- Platform dependent





## *Evolution of the Client: Thin Client*

- All application logic on the server
- Relatively easy to update
- Massively concurrent application
- Very simplistic user experience
- Client platform independent
  - Clients run in browser

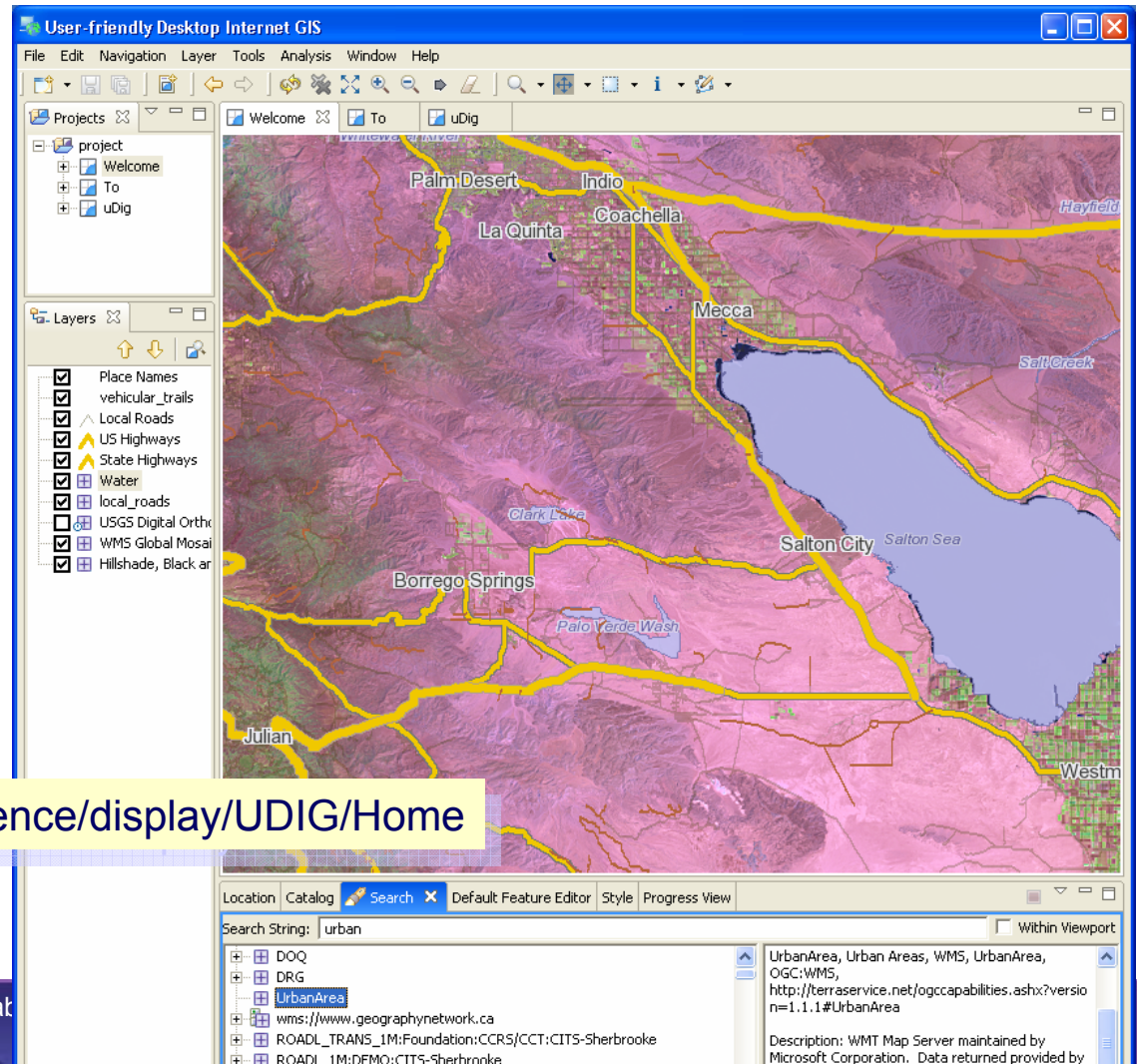




# What is a Rich Client?



- An application that uses the windowing and GUI features of the operating system
  - Native widgets
  - Drag & drop
  - Integration with platform component model
  - Much, much, more...



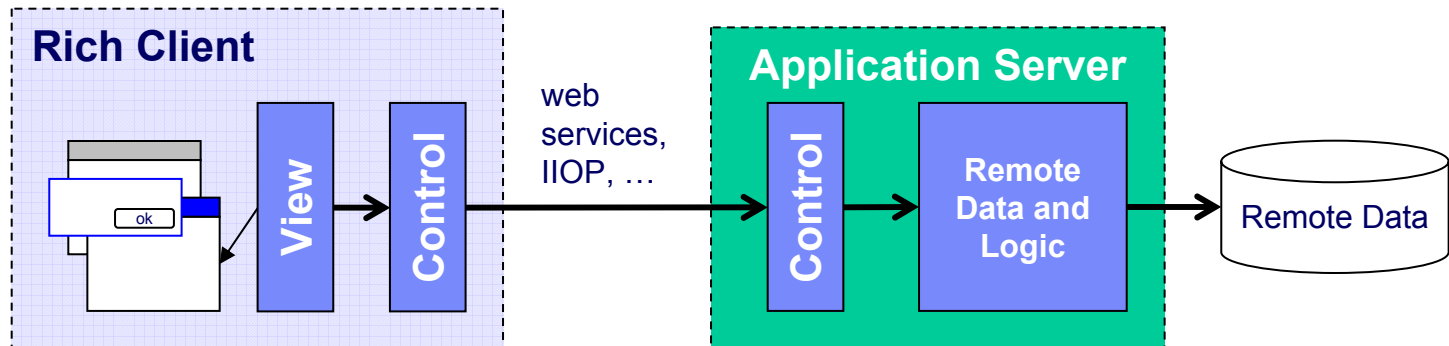
<http://udig.refractions.net/confluence/display/UDIG/Home>



# Three-tier Rich Client Application



- Some business logic stored remotely
  - Accessible through web service or other mechanism
  - WSDL-based web services, RESTful, IIOP, ECF, etc.

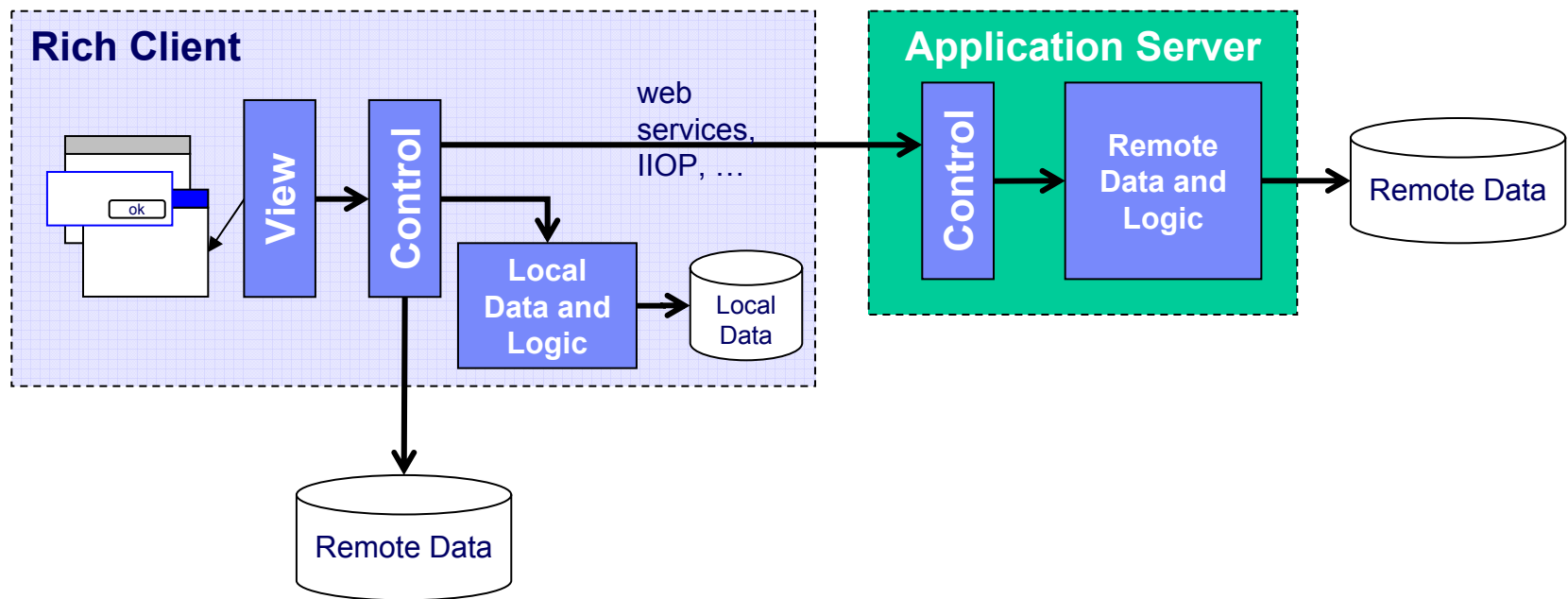




# *Combinations are Possible (and Likely)*



- Access data from a variety of sources
- Cache remote data locally for offline work

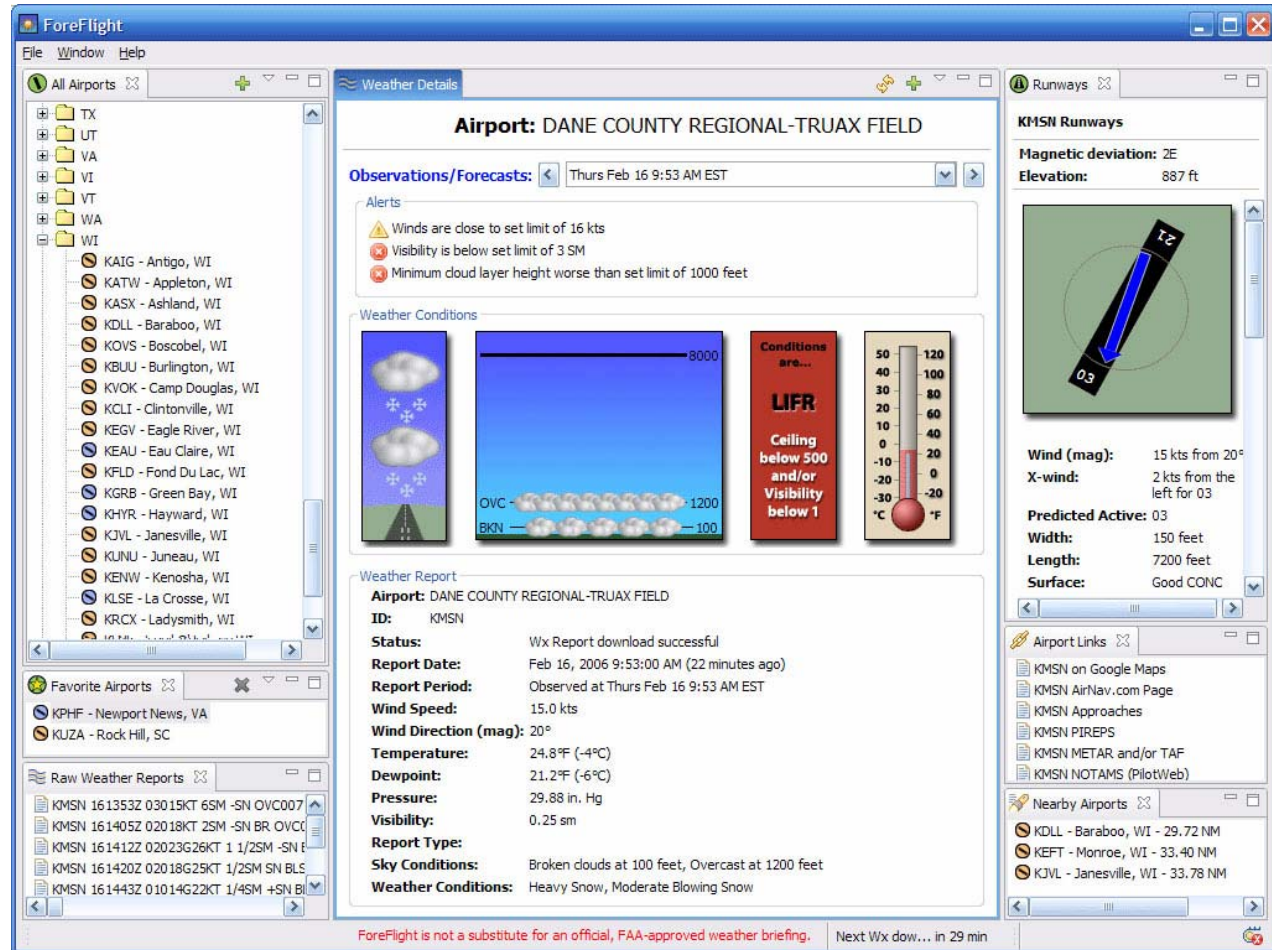




# Example: ForeFlight



- Displays critical information graphically and prominently
- Displays alerts when conditions are near or exceeding the user's preferred limits
- Connects via the web to weather and information services
- Multiple ergonomic views of the weather that affects the go/no-go flight decision



<http://www.foreflight.com/>



# Example: Lotus Notes “Hannover”



Sam Curman - Inbox - IBM Lotus Notes

File Edit View Action Tools Window Help

Welcome Mail 4 Activities 5 Calendar 2 Contacts Sales Leads 2

Go to More

Search: [ ] New Reply Forward Collaborate Follow Up Organize By Date

Sam Curman

- Inbox (12)
- Drafts
- All Messages
- Sent
- Follow Up
- Junk Mail
- Trash
- Views
- Folders
- Tools

From	Subject	Date	Time	Size
Betty Zechman	Can we get together?	05/01/05	9:12AM	46KB
Pierre Dumont	Check out these new sales tools	05/01/05	9:15AM	1,234KB
George Bandini	Thoughts on this quarter's results	05/01/05	9:36AM	1KB
Lukas Geiger	More on OP Tools deal	05/01/05	10:02AM	31.2KB
Anna Bauer	Does RNV have an exit strategy for the	05/01/05	10:26AM	1,345KB
Rita Ferrar	Request from Amadou: Please review Pro line	05/01/05	10:45AM	13KB
Lukas Geiger	Idea to boost customer satisfaction next year	05/01/05	11:02AM	14KB
Monifa Shani	Re: More on OP Tools	05/01/05	11:30AM	156KB
Pierre Dumont	Re: More on OP Tools	05/01/05	11:32AM	356KB
Lukas Geiger	Re: Check out these new sales tools	05/01/05	11:42AM	10KB
Anna Bauer	Re: More on OP Tools	05/01/05	12:01PM	46KB
Juan Sanchez	On vacation until next Monday	05/01/05	12:59PM	1,234KB
Lukas Geiger	Fw: New home start projections for next year	05/01/05	1:24PM	1KB
Monifa Shani	Please update projections for this quarter by	05/01/05	1:37PM	2KB
Anna Bauer	Re: Fw: New home start projections for next	05/01/05	1:40PM	1,345KB
Laura Klein	Re: More on OP Tools	05/01/05	1:52PM	13KB
Anna Bauer	Intelligence about change in building	05/01/05	1:57PM	14KB

**More on OP tools deal** Today, 10:02AM

From: Lukas Geiger  
To: Sam Curman, Pierre Dumont, Anna Bauer, Laura Klein  
cc: Monifa Shani

All:

I've been thinking about the OP Tools deal. Take a look at these sales projection charts. Elite line accounts for most of the sales until Q4, when the Pro line really takes off and the Elite line tanks. I think this shows we need to push the Pro line with OP Tools. Thoughts?

**Total Sales by Line**

1 Attachment, 31.2 KB Save All

**Related Activities**

- OP Tools deal
- Sales Lead: OP Tools
- More on OP Tools deal
- Competitive products
- 5/01/05 Sam Curman...

**Contacts**

Available

- Related Contacts (1)
- Pierre Dumont
- Anna Bauer
- Monifa Shani
- My Contacts (26)

**Related RSS Feed**

- Reuters Business Headlines
- Sales Discussion
- Commercial Development N...

**Calendar**

Today is May 01, 2006

- Related meetings
- Today 1:00 OP Deal Meeting
- May 05 1:30 Meeting with Monifa
- May 19 3:30 OP Tools Meeting
- Today's meetings
- 9:30 Business Discussion
- 1:00 OP Deal Meeting

Online



# Example: RSS Solutions



- Advanced planning and scheduling (APS) solutions

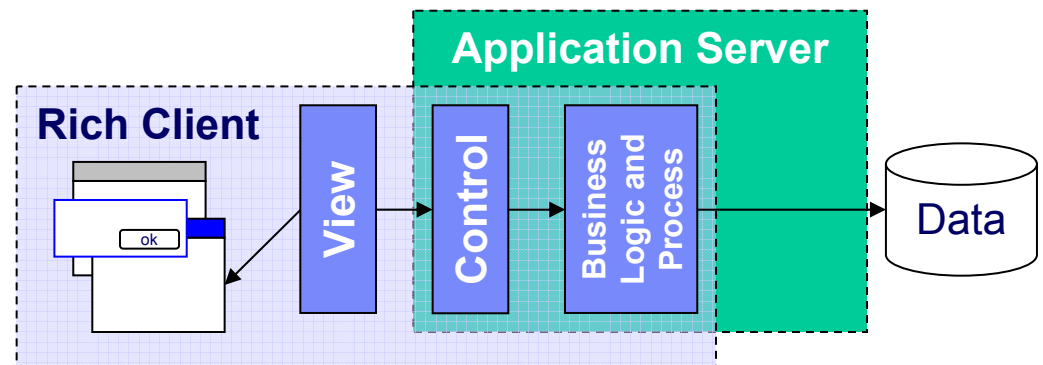




# What is a Rich Client?



- Rich user experience
- Typically (though not necessarily) a client for some backend service
- Platform independent
  - Runs with little or no modification on multiple platforms and devices
- Component model
- Integrated update mechanism
- Extensible





# *Rich or Thin Clients?*



- Thin client
  - Casual users don't want to install software
  - Online banking, auction browsing
- Rich client
  - Power users demand responsive, rich, powerful user interfaces
  - Teller, bank manager, power auctioneer

**I like all my clients  
to be rich!**



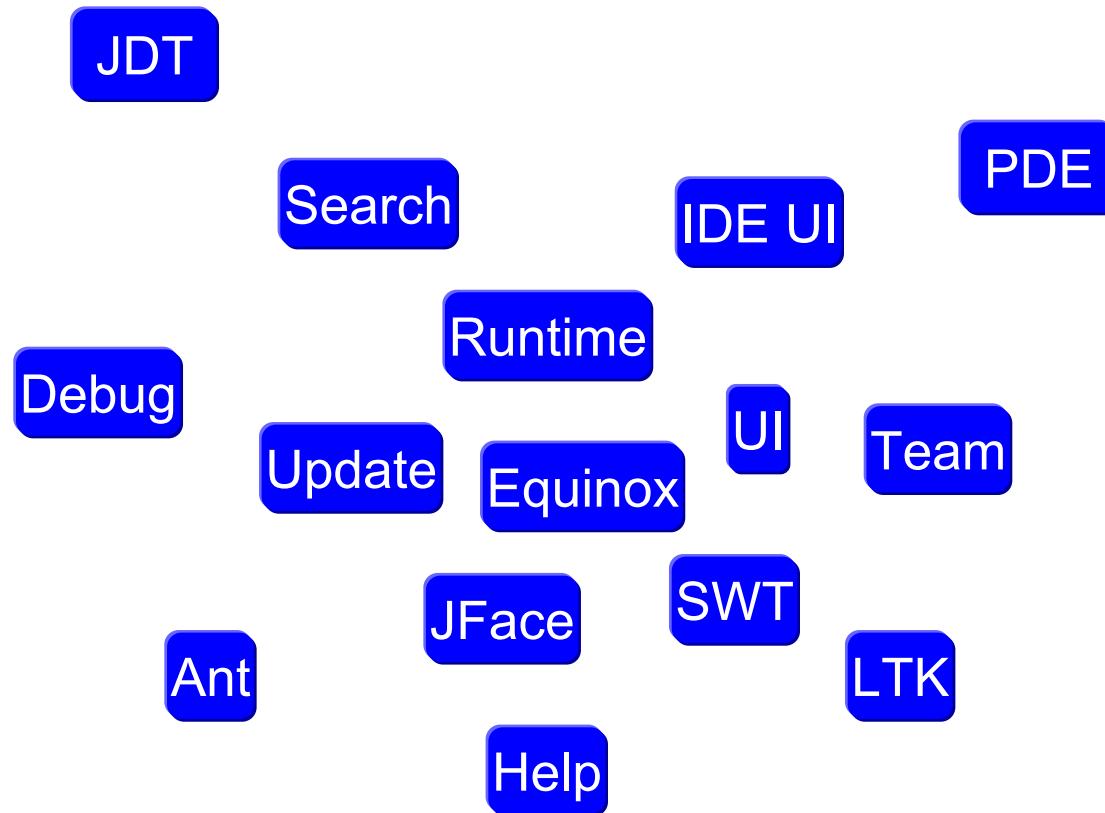




*What is Eclipse Rich Client Platform?*



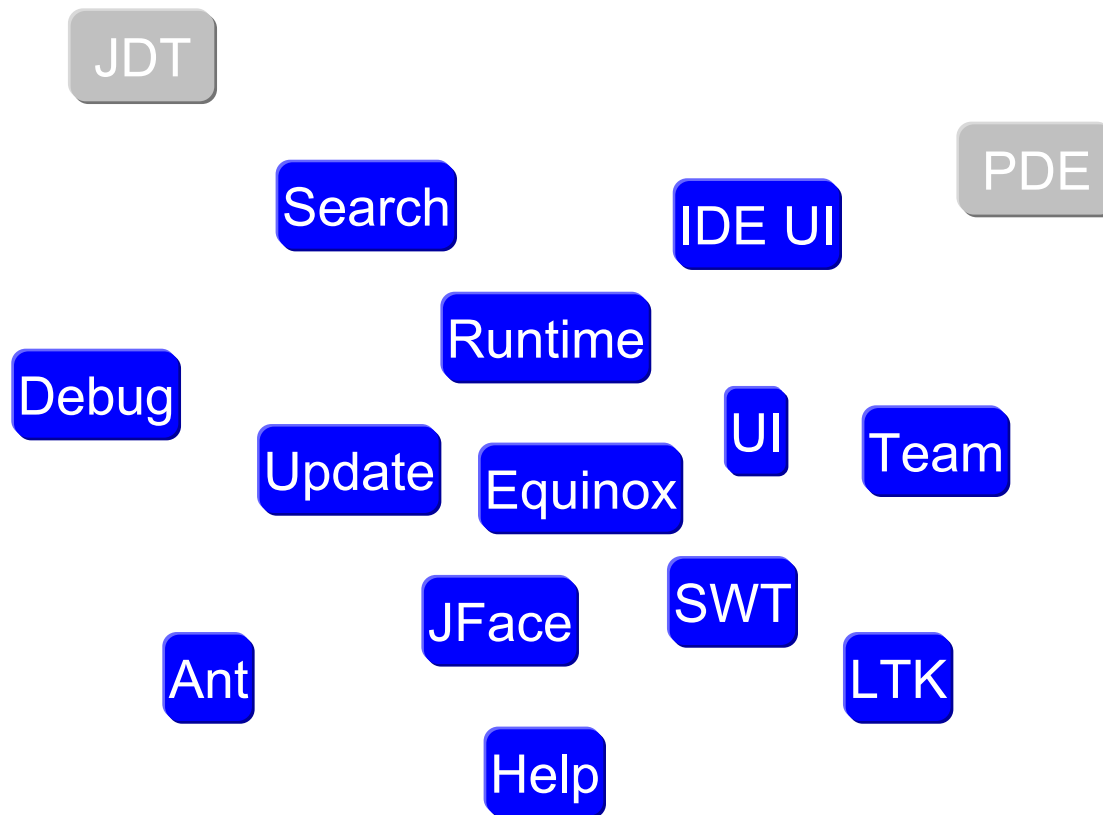
# *Eclipse is a Composition of Components*



**Eclipse SDK/Java IDE**



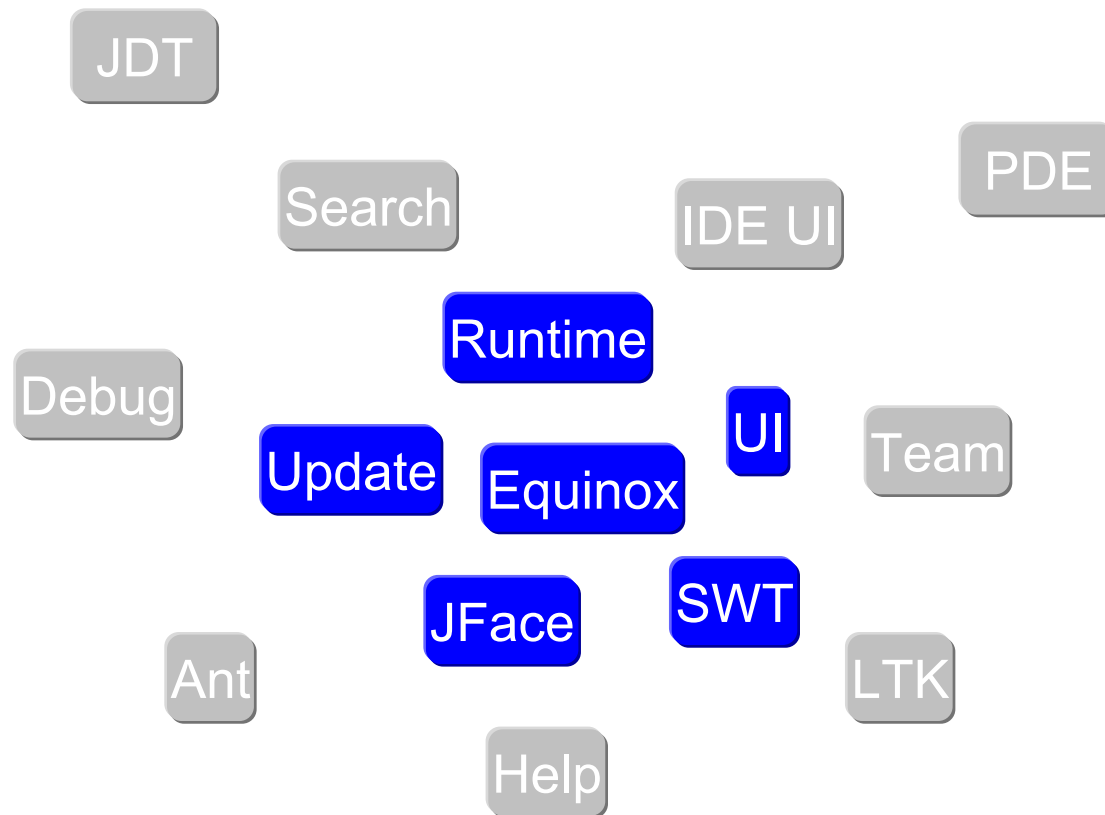
# Generic IDE Components



**Eclipse IDE**



# *Eclipse Rich Client Platform*





# *Why Use Eclipse Rich Client Platform?*



- A consistent and native look and feel across applications and features
- Provides common application services
  - Native look and feel
  - Window management
  - Standardized component model (Equinox)
    - Pervasive extensibility – Extension registry
    - Update Manager
  - Help system
- First-class development tools
- Middleware for building rich client applications!
  - Allows programmers to focus on core application not the plumbing
  - Don't reinvent the wheel

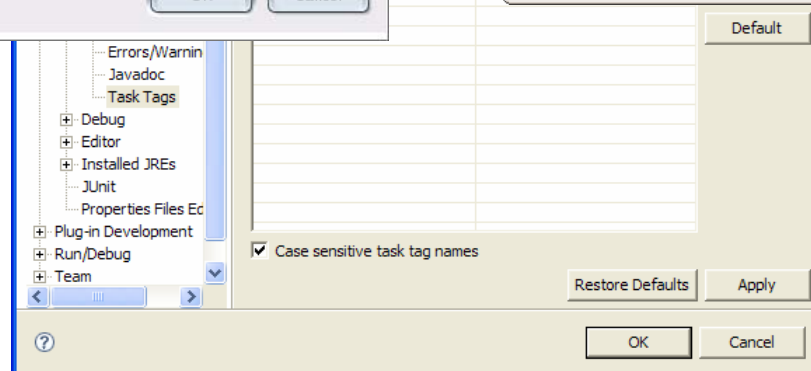
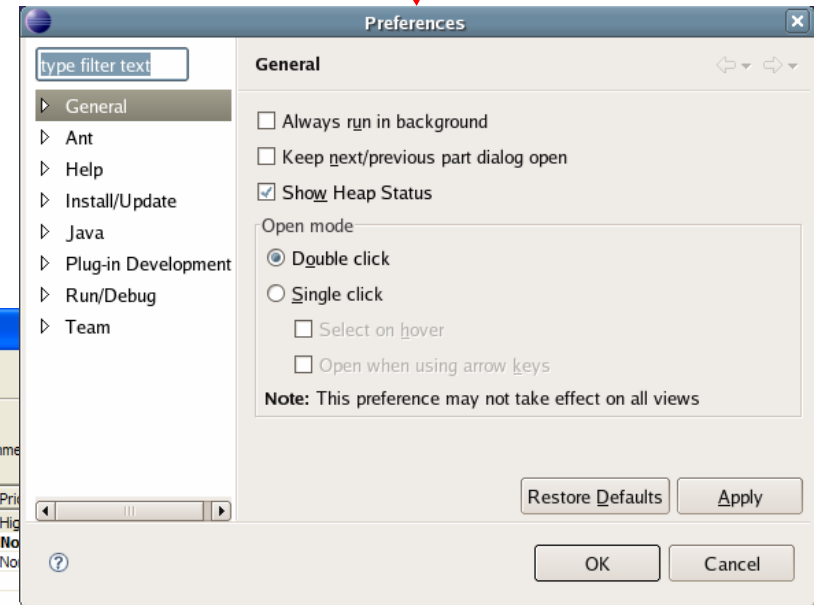
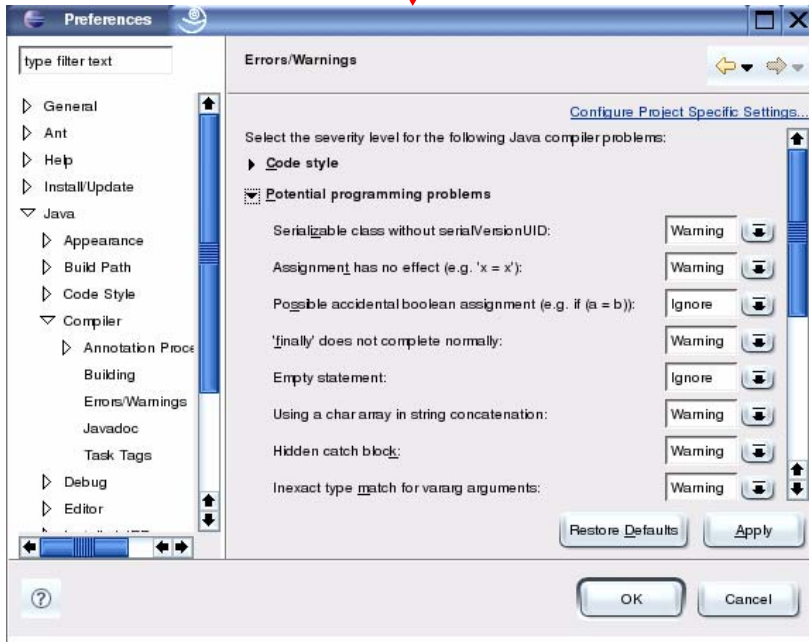


# Native Look and Feel



SuSE Linux  
(KDE)

Fedora Linux  
(GNOME)



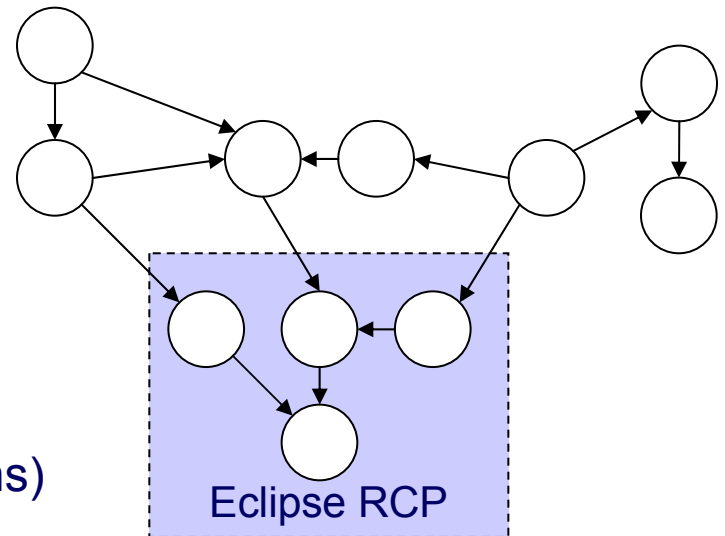
Windows XP



# Equinox (1/2)



- Equinox is the Eclipse component model
  - Based on OSGi R4 specification
  - Standard Java lacks an explicit notion of components
- Components == Bundles == Plug-in
  - Versioned
  - Defined declaratively
  - Dynamically loadable/unloadable
  - Support dynamic update and install
- Explicitly define
  - Dependencies
  - Runtime visibility
  - Interactions (extension points/extensions)





## Equinox (2/2)



- Components integrate without interfering
  - Required components explicitly set
  - Unrelated components do not have direct access to one-another
- Downstream components can access upstream components through the extension mechanism
  - Downstream component registers (declaratively) an extension point
  - Dependent components register (declaratively) extensions



# *Eclipse RCP as an Integration Point*



- Integrating completely independent components is easy
  - Views from independent components can easily share the workspace
  - Menus populated by multiple components
- Integrating loosely coupled components requires planning
  - Sharing selection information
  - Drag and drop between views from different components
  - Extension point/extension mechanism for more intimate integration
    - Provide hooks to let downstream plug-ins participate
    - Not artificially restricted in any way



# Trade offs



- Equinox provides the infrastructure to make components work
  - Architects and developers must factor the application into components
- Fine-grained components
  - Extreme example: one class per component
  - Reduced start up time
  - Easier updates and reuse
- Coarse-grained components
  - Extreme example: entire application in a single component
  - Generally easier to configure and maintain
- Balance
  - Best solution somewhere in between
  - Logical groupings of classes and resources





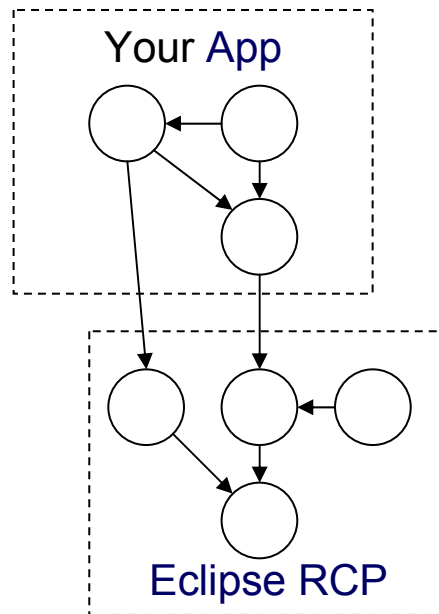
## *Building Platforms*



# Building Platforms (1/2)



- It all starts with plug-ins
  - Eclipse RCP applications are composed of components that plug into the platform
- When starting development on RCP, it is common to provide a handful of domain-specific components that sit directly on top of RCP

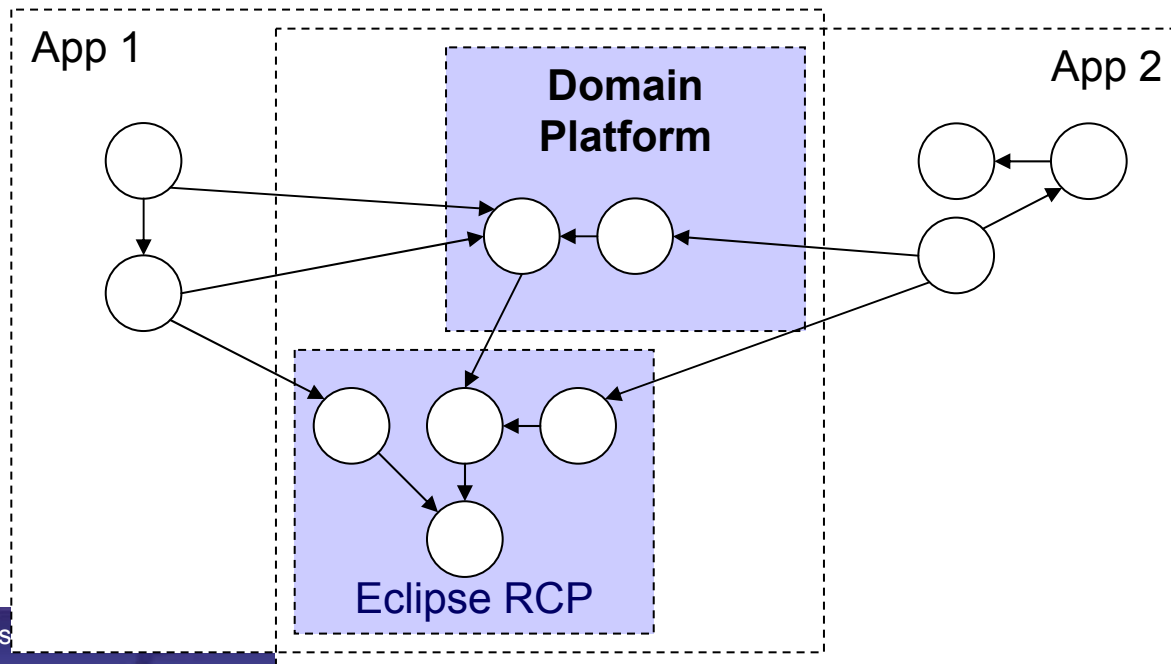




## Building Platforms (2/2)



- It's natural for RCP development to spawn one or more “platforms”
  - A custom base for multiple development teams to build their applications upon
  - All of the examples in this presentation all have an underlying domain-specific platform





# Example: Maestro – NASA Space Mission Management



Maestro

File Edit Window Downlink Help

Orbital View

EDR Search View

Sol Range Start End

Instruments FRONT\_HAZCAM\_LEFT FRONT\_HAZCAM\_RIGHT Use Selected Orbital Region

Go

Product ID	Instrument	Sol	Seq ...
2F134356767EFF2600P1212L0M1	FRONT_HAZCAM_LEFT	90	p1212
2F134449644EFF2700P1212L0M1	FRONT_HAZCAM_LEFT	91	p1212
2F134614869EFF2700P1403L0M1	FRONT_HAZCAM_LEFT	93	p1403
2F134615161EDN2700P1131L0M1	FRONT_HAZCAM_LEFT	93	p1131
2F135147950EDN2700P1131L0M1	FRONT_HAZCAM_LEFT	99	p1131
2F135148174ESF2700P1127L0M1	FRONT_HAZCAM_LEFT	99	p1127
2F135149189EDN2700P1141L0M1	FRONT_HAZCAM_LEFT	99	p1141
2F135149794EDN2700P1141L0M1	FRONT_HAZCAM_LEFT	99	p1141
2F135150380EDN2700P1141L0M1	FRONT_HAZCAM_LEFT	99	p1141
2F135150997EDN2700P1141L0M1	FRONT_HAZCAM_LEFT	99	p1141
2F135151610EDN2700P1141L0M1	FRONT_HAZCAM_LEFT	99	p1141
2F135152416EDN2700P1141L0M1	FRONT_HAZCAM_LEFT	99	p1141
2F135152602EFF2700P1212L0M1	FRONT_HAZCAM_LEFT	99	p1212
2F135153765EDN2700P1111L0M1	FRONT_HAZCAM_LEFT	99	p1111

Image View Image View Image View Image View

The screenshot displays the Maestro software interface. The main window shows an orbital view of a lunar surface with various sites marked. A green line connects the sites, and a blue box highlights a specific area. The right panel shows the EDR Search View with a table of product IDs and a list of instruments. The bottom right panel shows a zoomed-in image of the lunar surface.

<http://www.eclipse.org/community/casestudies/NASAFinal.pdf>

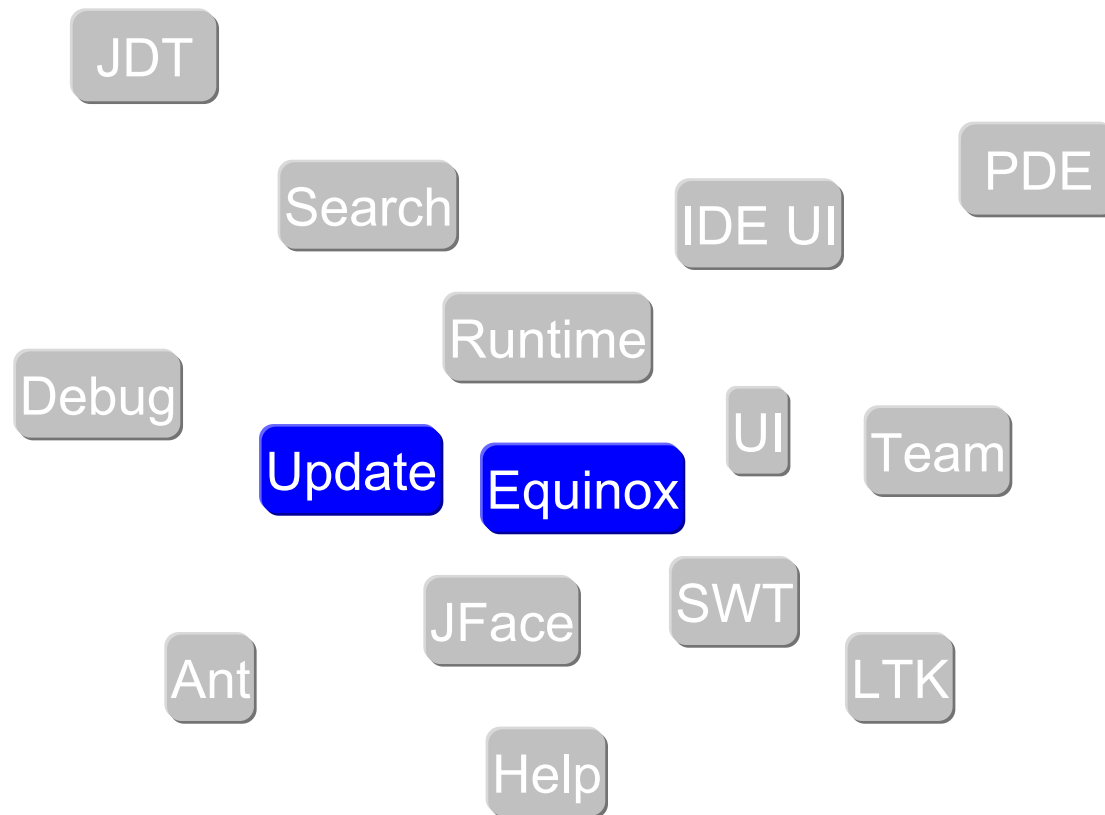




## *Outlook – Beyond client side applications*



# *Eclipse Rich Server Platform (RSP)*





# *Server-side Eclipse*



- Why use the Equinox component technology only on the client side?
- Component model
- Update mechanism
- Extensibility
- All interesting for server-side applications as well
- Combination with Spring possible





## *Middle-tiers on Equinox*

- Equinox can be used to implement middle-tiers
  - Same component model on both sides
  - Same extensibility for both sides
- Client and server could share the same components
- Integration with web-/app-servers possible



# *Equinox-based web apps*



- Equinox can run inside a web app or the web-app can run on top of Equinox
- Web-app can be componentized
- Web-app can be designed and implemented for extensibility (Extension-Points)



# Examples



- WAS 6.1
- Adobe Version Cue
- Apache Harmony
- Eclipse Rich AJAX Platform
- ...





# Recommended Reading



- Eclipse Rich Client Platform
  - By Jeff McAffer and Jean-Michel Lemieux
  - Addison-Wesley Professional
  - ISBN: 0321334612
- SWT : The Standard Widget Toolkit, Volume 1
  - By Steve Northover, Mike Wilson
  - Addison-Wesley Professional
  - ISBN: 0321256638
- Contributing to Eclipse: Principles, Patterns, and Plugins
  - By Erich Gamma, Kent Beck
  - Addison-Wesley Professional
  - ISBN: 0321205758

