Putting the Pedal to the Metal
RDi from *ZERO to *SIXTY

Charles Guarino
Central Park Data Systems, Inc.

About the Speaker

With an IT career spanning over 30 years, Charles Guarino has been a consultant for most of them. Since 1995 he has been founder and President of Central Park Data Systems, Inc., a New York area based IBM midrange consulting company. In addition to being a professional speaker and corporate trainer, he is a frequent contributor of technical and strategic articles and webcasts for the IT community. He is a proud member of COMMON’s Speaker Excellence Hall of Fame and also Long Island Software and Technology Network’s Twenty Top Techies of 2009. Charles currently serves as a member of COMMON’s Strategic Education Team (SET) and is also Immediate Past President and monthly Q&A host of LISUG, a Long Island IBM i User’s Group www.lisug.org. Charles can be reached at cguarino@centralparkdata.com, LinkedIn - http://www.linkedin.com/in/guarinocharles, Twitter - @charlieguarino
What We’ll Cover …

• What is it?
  • Perspectives and views
  • The Remote System Explorer (RSE)
  • Live Parser Extensible Editor (LPEX)
  • Using the Debug Perspective
  • Using the DDS Design Perspective
  • Wrap Up

What is it?

Eclipse is famous for its Java Integrated Development Environment (IDE), but its C/C++ IDE and PHP IDE are pretty cool too. You can easily combine language support and other features into any of our default packages, and the Eclipse Marketplace allows for virtually unlimited customization and extensions.
Eclipse 4.4 Luna

Overview

Eclipse Luna includes official support for Java™ 8 in the Java development tools, Rich in Development Tools, Object Teams, Eclipse Communication Framework, Maven integration, Xtext, Xtext, Web Tools Platform, and Memory Analyzer. The Eclipse compiler includes language enhancements, search and refactoring, Quick Assist and Clean up to migrate anonymous classes to lambda expressions and back, and new formatter options for lambda.

Luna highlights include:

- Full support for Java™ 8
- The workbench provides a new dark theme, split editors, line numbers on by default, reduced whitespace in default presentation, and ability to hide the "quick access" bar.
- Sirius allows architects to easily create their own graphical modeling workbenches by leveraging the Eclipse modeling technologies, including Eclipse Modeling Framework (EMF) and Graphical Modeling Framework (GMF)
- Eclipse includes a full implementation of the RS-Core framework as well as several component services.

IBM Rational Developer for i

IBM® Rational® Developer for i

9.5

Licensed Materials - Property of IBM Corp. © Copyright IBM Corp. and other(s) 1992, 2015. IBM, the IBM logo, and Rational are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries or both. Built on Eclipse is a trademark of Eclipse Foundation, Inc. Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. Other company, product or service names may be trademarks or service marks of others.
What We’ll Cover …

• What is it?

• Perspectives and views

• The Remote System Explorer (RSE)

• Live Parser Extensible Editor (LPEX)

• Using the Debug Perspective

• Using the DDS Design Perspective

• Wrap Up
What We’ll Cover …

- What is it?
- Perspectives and views
  - The Remote System Explorer (RSE)
  - Live Parser Extensible Editor (LPEX)
- Using the Debug Perspective
  - Using the DDS Design Perspective
- Wrap Up

Setting up your initial connection environment

- Initial library list, current library, initial command to run
  - Initial command helpful for dynamic library lists
  - Right click on connection name
Climbing down the connection navigation tree

- Subsystems are added for each connection that you create
- Let’s explore each one and discuss how each one adds value to your tool kit

New with RDi Version 9.5!
Introducing Objects

- Use to create filters
- Manipulate the connection’s library list
- Explore the contents of user libraries

Objects > Work With

- Only looks like PDM 😊
- Used to create filters
- Far more powerful than PDM could ever be
Objects > Library List

• Since each connection is its own session, it will have its own library list
• If you don’t configure your session with a permanent entry to your library list you’ll only have the basic one shown below.
• Full object authority is in effect
• Right-click to add an entry to this list

Introducing Commands

• A “command” is really a script of multiple commands
• There are several pre-defined commands
Introducing Jobs

- The jobs list is segregated into different active subsystems
- Right clicking on a job allows you to control and monitor it

Introducing IFS

- A very easy way to navigate through the IFS
- Can create filters as a shortcut to a particular folder
- Makes copying documents to folders a snap using copy and paste
- Can also easily create files and folders in a particular directory
Introducing Spooled Files

- Multiple filters can be created to further speed up your searches

A FILTER limits what objects appear in your current selection

- You’ve been using filters in SEU all along
demolib.qrpglefix is the name chosen for this filter

- Click on any filter and its contents open

Visual Application Diagram

• Presents a graphical view of your program flow
  • Can be edited and saved as a separate image file in a variety of formats such as .png, .jpeg, .bmp etc.
What We’ll Cover …

• What is it?
• Perspectives and views
• The Remote System Explorer (RSE)

• Live Parser Extensible Editor (LPEX)

• Using the Debug Perspective
• Using the DDS Design Perspective
• Wrap Up

New with RDi Version 9.5!

```sql
-- WEATHRSC04.SQLRPGLE II
  Line 26   Column 71   Replace

000100 **free
000200
c01-opt dftactgrp('no');
000400
000401!// This is the table that collects the parsed XML web services data
000500!// create table webservice/websvcdatal xmldata datatype for column xmltype
000600!// char (20 ) not null with default, xmldata varchar (256 )
000700!// not null with default)
000700
000800 dcl pr weatherjson;
000801       CityName = char(30) Const:
000802       ! end-pr;
000803
000804 dcl-pi weatherjson;
000805       CityName = char(30) Const;
000806       ! end-pi;
000807
001200 dcl-s options char(100) init('doc=string');
001300 dcl-c num_elements const(500);
001400 dcl-s parseddata char(256);
001500 dcl-s command char(256);
001600 dcl-s length packed(15:5);
001700 dcl-s isostring char(132766);
```
Source code editing in LPEX

- Can select “Open with LPEX” or just double click on member

Source code filtering helps you quickly locate what you need

- Right click anywhere in the source code
When you select a block of code other options are available

- Select the block of code, right click and select your function

Block nesting displays arrows at start and end of logic blocks

- Position on start of block > Right click > Source > Show Block Nesting
Content Assist helps you fill in the blanks

- Ideal for built-in functions (bifs)
  - Type a portion of the bif and press CTRL-SPACE
  - Also works with your own procedures

```plaintext
// *******************************************************
beg builblocks:1nlen:
// This routine will receive a random field length and
// returns a value which is a multiple of 8.
// Field name is datastringlen.
if datastringlen <= 8:
datastringlen = 8;
elsif datastringlen <= 16:
datastringlen = 16;
elsif datastringlen <= 24:
datastringlen = 24;
endif;
end-free
```

Source verification – Out of the park!

- Source>Verify – or – Shift-Ctrl-V
  - Simulates the compiler but doesn’t create an object
  - Can be used while offline
  - Double click on an error and RDi goes right to the line
RPG Procedure Wizard removes the complexity

- **Source/New/Procedure**
  - Creates the procedure interface as well as the prototype

![RPG Procedure Wizard](image)

D-Spec wizard – Never have to remember the keywords!

- **Source/New/D Specification**
  - Used for standalone fields, named constants, program-described, externally-described, program-status and file-information data structures

![D-Spec Specification Wizard](image)
What We’ll Cover …

- What is it?
- Perspectives and views
- The Remote System Explorer (RSE)
- Live Parser Extensible Editor (LPEX)

- Using the Debug Perspective
- Using the DDS Design Perspective
- Wrap Up

Program we will be debugging

```
Line 31  Column 72  Replace 1 change
Free-Form
000000  ctl-opt bnddir('UTILITIES' : 'QCLL') optactgrp('no') actgrp('QCLL')
000001  +
000002  option(srcstmt : "nodebug") debug('input');
000003  +
000004  dcl-f custmast disk('ext') keyed usage('update');
000005  +
000006  dcl-pr secretdata char(24);
000007  +
000008  *n char(24) value;
000009  +
000010  *m char(1) value;
000011  +
000012  end-pr;
000013  +
000014  dcl-s cleardata char(24);
000015  +
000016  dcl-s encrypteddata char(24);
000017  +
000018  dcl-s direction char(1);
000019  +
000020  read custmast;
000021  +
000022  do not %eof (custmast);
000023  +
000024  direction = 'E'; // Value of 'E' tells procedure to ENCRYPT
000025  +
000026  cleardata = cleardata direction;
000027  +
000028  encrypteddata = secretdata(cleardata direction);
000029  +
000030  // Update file CUSTMST with encrypted data
000031  +
000032  concdata = encrypteddata;
000033  +
000034  update custmstr;
000035  +
000036  read custmast;
000037  +
000038  enddo;
000039  +
000040  *incl = '001';
```
Introducing the DEBUG perspective

- “Wakes up” automatically when a program launched in debug mode or an active service entry point is encountered
- Green line is the current line of execution
- Boxes shows shortcuts, breakpoints and current line of execution pointer

Introducing the VARIABLES view

- All program variables are displayed and updated in real time
  - Each variable will change color when its value changes
  - This view is customizable using the drop-down menu
  - Right click to change view and add to monitors view
  - Values can be changed by simply over-typing
Introducing the BREAKPOINTS view

- Breakpoints can be set at the source level or at runtime
- Breakpoints can be conditional or unconditional
- Can also be disabled so you don’t have to delete them
- Watch breakpoints are set at runtime – here we’re watching the variable named “direction”

![Variables and Breakpoints](image)

Field hovering

- Position the cursor over a field and its value appears.
- Much easier than typing “ev cleardata” or pressing F11!
What We’ll Cover …

- What is it?
- Perspectives and views
- The Remote System Explorer (RSE)
- Live Parser Extensible Editor (LPEX)
- Using the Debug Perspective
- Using the DDS Design Perspective
- Wrap Up

Field “FROMPER” displayed in Properties view
The “out of the box” Screen Design perspective

Introducing the SCREEN EDITING area
Introducing the PALETTE view

- This is a “smart” real-time view
- Its contents change as you click on different screen or report objects

Introducing the PROPERTIES view

- This is a “smart” real-time view
- Its contents change as you click on different screen or report objects
Populating the screen

- Drag the various elements from the palette to build your screen
  - Here a Text Constant was just added

What We’ll Cover …

- What is it?
- Perspectives and views
- The Remote System Explorer (RSE)
- Live Parser Extensible Editor (LPEX)
- Using the Debug Perspective
- Using the DDS Design Perspective
- Wrap Up
You're not alone  Join the conversation!

Putting the Pedal to the Metal
RDi from *ZERO to *SIXTY

Charles Guarino
THANK YOU !!!