Looking Inside the Developer’s Toolkit:
“Best Stuff” from the Advanced RDi Topics

Charles Guarino
Twitter @charlieguarino
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 and corporate training company. In addition to being a professional speaker across the United States and Europe, he is a frequent contributor of technical and strategic articles and webcasts for the IT community. He is a member of COMMON's Speaker Excellence Hall of Fame and also Long Island Software and Technology Network’s Twenty Top Techies. In 2015 Charles became the recipient of the Al Barsa Memorial Scholarship Award. Additionally, he serves as a member of COMMON’s Strategic Education Team (SET) and is also a 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 …

- Hidden LPEX gems
- Climbing down the navigation tree
- Object tables
- The remote scratchpad
- Preferences
- Wrap Up

Quick New Member Open Shortcut!

- For those times when you want to edit or browse a source member but don’t want to create or change a filter
- Press CTRL-ALT-Q
Tokenization !!!

cia1-opt bnddir('UTILITIES' : 'QClLE') dfactgrp(*no) actgrp('QILE')
  
  ,
  
  dcl-f custmast disk(*ext) keyed usage(*update);

dcl-pr secretdata char(24);
  *
  *
  *
  *
  *
end-pr;

dcl-s cleardata char(24);

dcl-s encrypteddata char(24);

dcl-s direction char(1);

read custmast;
dow not %eof (custmast);

direction = 'E'; // Value of 'E' tells procedure to ENCRYPT

  cleardata = cclrdata;

  encrypteddata = secretdata(cleardata:direction);

  // Update file CUSTMAST with encrypted data
  cmndata = encrypteddata;
  update custmastr;

read custmast;
enddo;

*lnr = *on;
return;

Parser Styles

![Preferences](image)

/TITLE This is the title
/SPACE 5
H DFTACTGRP(*NO) CTL-OPT OPTION(*SRCSTMT);
FMYFILE IF E DISK BLOCK(*NO)
FPFRINT O P 132 PRINTER
* This is a comment
D ARRAY S 8 DIM(3)
D VALUE S 5P 2 INZ(5)
DCL-F myfile DISK(*EXT) USAGE(*UPDATE)
EEXTDESC('MYLIB/MYFILE');
DCL-DS "n; // This is a comment
message CHAR(100) INZ('hello');
dateDue DATE(*ISO) INZ(D'2013-01-02');
Marking – A quick way to highlight a variable

![Marking](image)

Hyperlinking – A quick way to change a variable definition

![Hyperlinking](image)

** The clear data variable
D cleardata  S  24
Templates – Short code to complex coding

Templates in action

```sql
// Check SQLSTATE:
SELECT
browserNavTab = getBrowserNavTab();
WHERE
    SQLSTATE = '01004';
//The value of a string was truncated when assigned to
//another string data type with a shorter length.
WHERE
    SQLState = '01005';
//Insufficient number of entries in an SQLDA.
WHERE
    SQLState = '01006';
//A privilege was not revoked.
WHERE
    SQLState = '01007';
//A privilege was not granted.
WHERE
    SQLState = '01009';
//The search condition is too long for the information schema.
WHERE
    SQLState = '0100A';
//The query expression of the view is too long for the
//information schema.
WHERE
    SQLState = '0100C';
```
Source formatting

**BEFORE**

```c
if hours > 40;
  overtime = hours - 40;
  overtimemay = overtime * overtimerate;
if hours > 50;
  overtimemay = overtimemay + bonus;
endif;
gross = regularpay + overtimemay;
else;
grosspay = regularpay;
endif;
```

**AFTER**

```c
if hours > 40;
  overtime = hours - 40;
  overtimemay = overtime * overtimerate;
if hours > 50;
  overtimemay = overtimemay + bonus;
endif;
gross = regularpay + overtimemay;
else;
grosspay = regularpay;
endif;
```

---

**Built-in Run SQL Scripts**

```sql
// Communications area definition
when event = "XML_START_DOCUMENT"
  clear cumArea;
when event = "XML_START_ELEMENT";
  exec sql
    insert into saxdata
    values(Start_Elemt", parseData, saxntds.productpath, saxntds.prodocname);
when event = "XML_END_ELEMENT"
  exec sql
    insert into saxdata
    values("End_Elemt", parseData, saxntds.productpath, saxntds.prodocname);
```
Built-in Run SQL Scripts

```sql
INSERT INTO saxdata
VALUES ('Start_Element', :parseddata,
        :saxctlds.prodocpath, :saxctlds.prodocname);
```

Remove unreferenced objects from Outline view

![Outline view with unreferenced objects]

Connected to relational database

Messages
Built in F1 support for embedded SQL

```
exec sql close Cl;
exec sql open Cl;
exec sql fetch Cl into : saxctlds;
```

What We’ll Cover …

- Hidden LPEX gems
- Climbing down the navigation tree
- Object tables
- The remote scratchpad
- Preferences
- Wrap Up
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

![From STRPDM to RDi - Awesome Navigation]

- Objects
- Commands
- IBMi Contexts
- Jobs
- IFS Files
- Spooled Files
- Qshells

Introducing Objects Subsystem

- Resequencing filters
- Copying source members
- Using the connection's library list

![Objects]

- Work with libraries...
- Work with objects...
- Work with members...
- Library list
- User libraries
Resequencing Filters

- Filters are appended to the list as you create them
- Prior to RDi 9.5 they could be moved only one level at a time
- Can now be dragged and dropped

Copying source members

- The most obvious method doesn't actually work
- You can’t copy from one filter to another
Copying source members (cont.)

• Instead right-click on the target source physical file

• Will receive a warning message if source already exists

Another option is to show the filter’s content in a table

PDM-like options exist from the table’s view
Copying source members (cont.)

- The PDM option 3 displays a more intuitive copy process
- Can select a different library
- Can select a DIFFERENT CONNECTION!

Source Compare – Method 1
Source Compare – Method 2

---

Introducing Commands and Command Sets

- A “command” is really a script of multiple commands
- There are several pre-defined commands
- Here I created a “command” to create a new user profile and also create a default development library for the new user:
Creating your own command set

• Here I issue CRTUSRPRF and CRTLIB commands
  • The question mark in front of the commands forces a prompt

Introducing Jobs

• The jobs list is segregated into different active subsystems
  • Right clicking on a job allows you to control and monitor it
Jobs (cont.)

- The jobs status view allows you to subset the active jobs regardless of the subsystem it is running in.
- Click on a column heading to sort that particular column.
- There is a refresh slider bar with a range of 10 seconds to 5 hours to Never Refresh.

<table>
<thead>
<tr>
<th>Date/Time Submitted</th>
<th>Job</th>
<th>Connection</th>
<th>Status</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jul 21, 2012 10:05:01 AM</td>
<td>994184/QLWSVR/ADMIN1</td>
<td>RSE Navigation</td>
<td>*ACTIVE</td>
<td></td>
</tr>
<tr>
<td>Jul 21, 2012 10:05:26 AM</td>
<td>994205/QLWSVR/ADMIN3</td>
<td>RSE Navigation</td>
<td>*ACTIVE</td>
<td></td>
</tr>
<tr>
<td>Aug 26, 2012 8:09:32 PM</td>
<td>070770/SAMANTHA/SAXPARSER</td>
<td>RSE Navigation</td>
<td>*OUTQ</td>
<td></td>
</tr>
</tbody>
</table>

* ( )  
+ Refresh Interval: 10 seconds

Jobs (cont.)

- The default view of the Jobs subsystem is to show ALL jobs.
- A Job Filter allows you to hone in on a particular job, job name or user profile.
- Right click on Jobs to bring up the filter window.
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

IFS Filters

- An IFS filter allows you quick access to a selected folder(s)
- Can specify which file types will be shown in the filter
- Can limit a filter to show files only, very helpful
IFS – RDi has an easy to use built in XML Editor

IFS – The editor understands XML
Introducing Spooled Files

- A built-in spool file viewer is available
- Once again system authority fully applies

Spooled Files

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

What We’ll Cover …

• Hidden LPEX gems

• Climbing down the navigation tree

• Object tables

• The remote scratchpad

• Preferences

• Wrap Up
Object Tables

- The object table is available from many locations within RDi.
- Experiment! Right click and select “Show in Table”

Object Tables (cont…)

- Click on any column heading to sort the table
- Double click for a reverse sort
- Right click on any table entry for member specific options
What We'll Cover …

• Hidden LPEX gems

• Climbing down the navigation tree

• Object tables

• The remote scratchpad

• Preferences

• Wrap Up

The Remote Scratchpad

• The graphical equivalent to a yellow sticky note
  • Provides a convenient place to temporarily store objects
  • Contents are removed from the scratchpad when you exit RDi
The Remote Scratchpad (cont.)

- Objects remain “live” while on the scratchpad
  - Objects can be used on the scratchpad as if they were in the normal RSE tree

![Remote Scratchpad screenshot](image)

What We’ll Cover …

- Hidden LPEX gems
- Climbing down the navigation tree
- Object tables
- The remote scratchpad
  - Preferences
  - Wrap Up
A world of options – What’s your Preference?

- Here is the workbench with only one defined connection
- Many “hidden” options, you need to know where to find them!

Preferences

- There are MANY available in RDi
  - Not all apply to RPG and COBOL development
  - Preferences ‘stick’ to a particular workspace
  - To access them click Window>Preferences
What’s your preference?

- A Filter Text window is available to help you find what you need
- A “Restore Defaults” exists in most so don’t be afraid!
- Window>Preference>Remote Systems>IBM i > Command Execution

Tagging your mods – How often do you do this?

- Add an automatic signature to every changed line!
- Windows>Preferences>type “parsers”>IBM i parsers
- The signature is placed in columns 1 through 5
RPG Parsers – What they mean and how to use them

- Window>Preferences>type “rpg” > ILE RPG

CL Parser – What it means and how to use it

- Window>Preferences>type “CL”
SQL Parser – What it means and how to use it

- “SQL Formatting” or Window>Preferences>type “SQL”

![SQL Formatting Image]

COBOL and DDS Parsers - What they mean and how to use them

- Window>Preferences>type “COBOL”

![COBOL Preferences Image]

Window>Preferences>type “DDS”

![DDS Preferences Image]
LPEX Preferences

- Windows>Preferences>type “appearance” > LPEX Editor > Appearance

What We’ll Cover …

- Hidden LPEX gems
- Climbing down the navigation tree
- Object tables
- The remote scratchpad
- Preferences
- Wrap Up
Looking Inside the Developer’s Toolkit: “Best Stuff” from the Advanced RDi Topics

Charles Guarino

Twitter @charlieguarino

THANK YOU !!!