Taking Batch Scripting to the Next Level with SQLcl

7th December 2016

Sabine Heimsath & Robert Marz

Sabine Heimsath

Client	Senior Database Application Developer PL/SQL, SQL Developer, APEX
its-people	Portfolio Manager Development Blogger
DOAG	Member of the Development Community

@oraesque

blog.its-people.de

Sabine.Heimsath
@its-people.de

Robert Marz

Client	Senior Technical Architect with database centric view of the world
its-people	Portfolio Manager Database Technologies Blog Editor
DOAG	Active Member Database Community in charge of Cloud topics

 @RobbieDatabee

 blog.its-people.de

 Robert.Marz
@its-people.de

What´s in Store for you?

Scripts and Slides available for Download

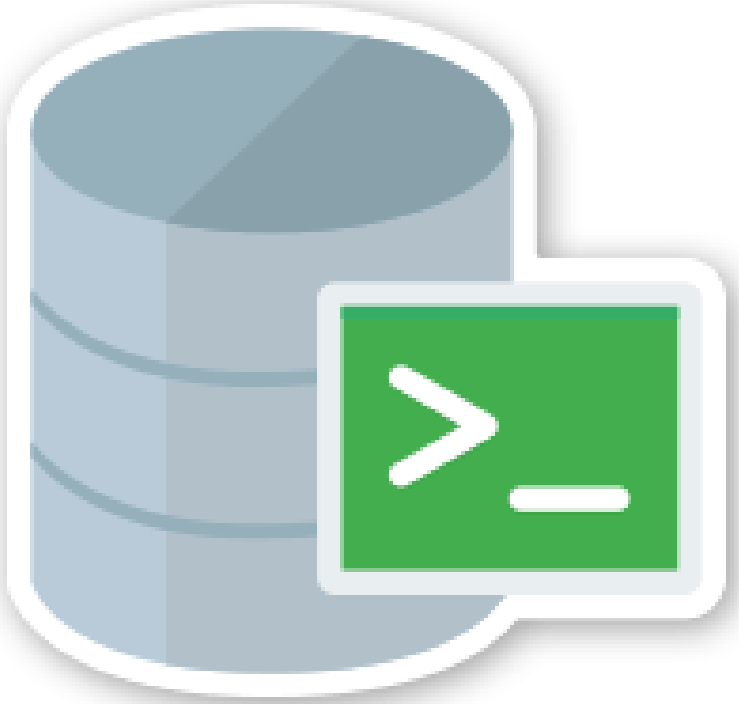
What to expect:

- Discover new Possibilities
- Examples

Not included:

- Introduction to JavaScript
- Complete Feature Overview

What is SQLcl?



Oracle SQL Developer Command Line

- The new SQL*Plus
- Modern Command Line
- Production Release since OOW
- Included in 12cR2 $ORACLE_HOME/bin

Scripting Vintage Style

SQL*Plus

Stable for decades

Sequential
SQL & PL/SQL

Not too dynamic
„DEFINE“

Error Handling
„whenever sqlerror“

Scripting in SQLcl

Languages (JSR-223)

New Possibilities

SQL*Plus „plus“

Scripting in SQLcl – Basics

GitHub Readme

https://github.com/oracle/oracle-db-tools/blob/master/sqlcl/README.md

sqlcl

sqlcl.setStmt(<string>)
sqlcl.run()

ctx

ctx.write(<string>)
print();

Util

execute(<string>,binds)
executeReturnOneCol(<string>,binds)
executeReturnListofLists(<string>,binds)
executeReturnList(<string>,binds)

Globals

There are a few globals pushed into the scripting engine for use.

args -This is a simple array of the arguments passed along

Example:

```
for(var arg in args) {
   ctx.write(arg + ":" + args[arg]);
   ctx.write("\n");
}
```

sqlcl - This is SQLCL itself

```
setStmt(<String of stuff to run>)
          This can be a single statement, an entire script of stuff, or any sqlcl command such as "@numbers.sql"
```

```
run()
       Runs whatever is set via the setStmt function
```

Example:

```
/* Run any amount of command in the sqlcl prompt */
sqlcl.setStmt("select something from somewhere; @myscript \n begin null;end;");

sqlcl.run();
```

ctx (this has tons of methods but this is the single most important)

```
write(<String>)
```

Example:

```
ctx.write('Hello World');
```

util (again tons of methods)

```
execute(<string>,binds)
      executes whatever is passed in with a boolean return for success/failure
```

```
executeReturnOneCol(<string>,binds)
      executes and returns the first row , first column
```

```
executeReturnListofList(<string>,binds)
      executes and returns an array(rows) of arrays(row).
```

```
executeReturnList(<string>,binds)
      execute and returns and array ( rows ) of objects ( row )
```

SQLcl – Demo 01a – Warm-Up

SQLcl Settings

Access directly via Script

Demo SQLcl History

Increase Limit

Query actual Size

SQLcl – Demo 01b – Warm-Up cont.

Alias

Suitable for small, reoccuring tasks

May contain SQL, PL/SQL oder Script

Definition is persistent

Demo
Restore Data

SQLcl – Demo 01c – Warm-Up cont.

login.sql	Customise your command line using JavaScript	
	SQLPATH	set
		show
	Demo	Switching between prompt layouts

SQLcl – Demo 02 – Flow Control

SQL*Plus	Sequential SQL Blocks	
	DEFINE, VARIABLE	
	Workarounds	Spool „new Script.sql“
		PL/SQL execute immediate
SQLcl	Modern Scripting Languages	
	JDBC Connection	Bind Variables
		Result Evaluation
	Access to SQLcl Interpreter	
	Dynamic Loading of Scripts	

SQLcl – Demo 02 – Flow Control (cont.)

Conditions	If – then – else	
	Case	
	Demo	Create table if exists
Loops	For / While	
	Demo	Looping through Results
		Assembling Script-Calls

SQLcl – Demo 03 – Loading Blobs

Blob	Transfer Blob from File into Database	
	Use Java Classes	Access Files via java.nio.files.Files
	Demo	Copy Image to Table

SQLcl – Demo 04 – Array Magic

Arrays	Powerful structures	Simple Types
		Objects
		Nested Arrays
	Demo Deploy Framework	List of Credentials
		Connection Check
		Execute Scripts with Credentials

SQLcl – Demo 05 – Remote Control

Pipes	Copy Data	No Export File
		No Database Link
	Named Pipes	Linux / Unix
		Mac OS X
		No CLI for Windows
	Demo	Copy Data between two Databases

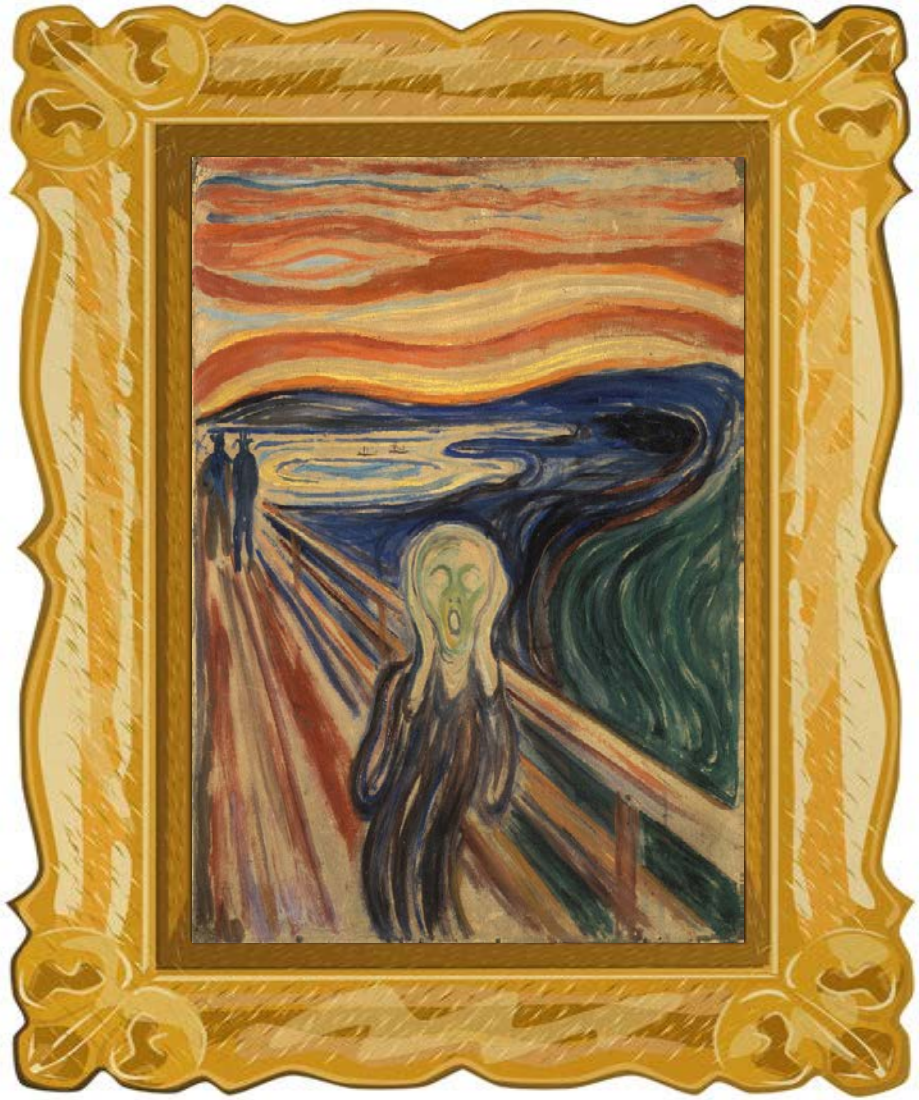
SQLcl – „noDemo“ 06 – Background Sessions

Parallelise Tasks	Java Threads	GitHub oracle-db-tools	bg.js longops.js
		its-people Blog	yet to be written
	Additional JDBC connections	Connection	metadata available except credentials
		„n“ threads	

SQLcl – Demo 07 – As you like it

JSR-223	e.g. JavaScript, Lua or Python	Java Implementations „nashorn“, „LuaJ“, „Jython“
	Embedding SQLcl	Use SQLcl JARs in your Apps
	Demo	Running lua & python from SQlcl
		Using SQLcl from python
		ECMA Script 5 & 6

The JavaScript



The JavaScript
Edvard Munch
1910

Reading List

Getting Started

- SQLcl Scripting: Docs
- SQLcl Community

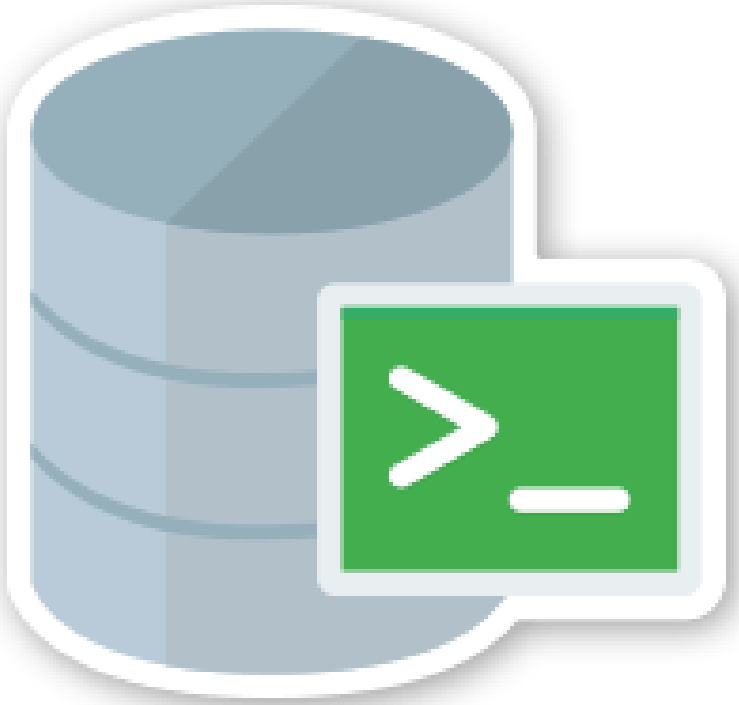
Basics

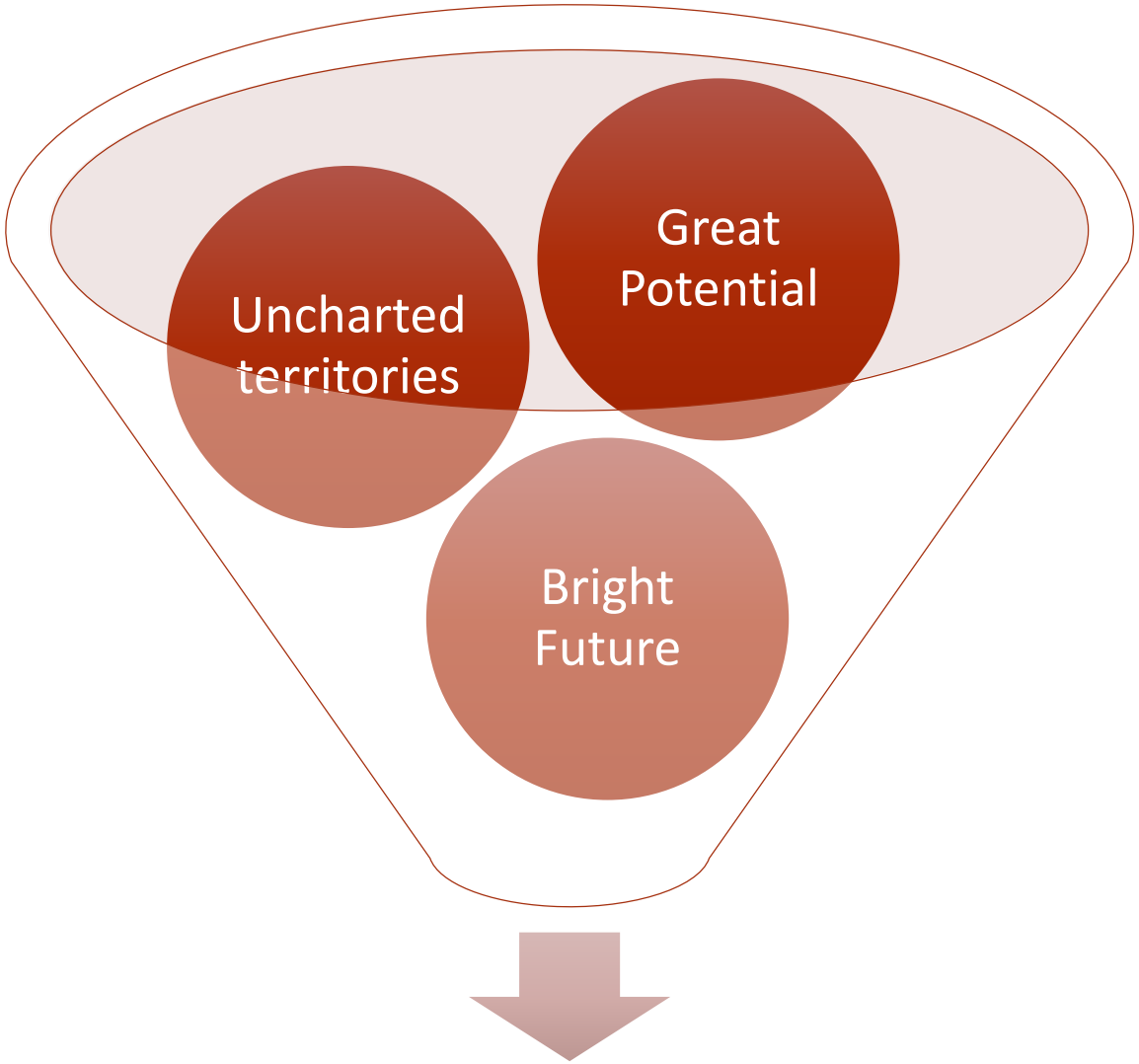
- List of JVM languages
- JavaScript Cheat Sheet
- Nashorn Tutorial

Reading List

- GitHub
  - SQLcl Examples (oracle-db-tools/sqlcl)
  - its-people (sqlclscripts)
- Blogs
  - Barry McGillin
  - Kris Rice
  - Jeff Smith

Conclusion





Definitely Worth Trying!

Thank you!

Questions?

we make the difference
www.its-people.de

its-people GmbH

Frankfurt	Tel. 069 2475 2100
Hamburg	Tel. 040 2360 8808
Köln	Tel. 0221 1602 5204
München	Tel. 089 5484 2401

its-people ERP Beratungsgesellschaft mbH

Frankfurt	Tel. 069 2475 1980

www.its-people.de **info@its-people.de**