



mibuso.com

Coding4Performance waldo

10 YEAR ANNIVERSARY
10 YEAR ANNIVERSARY

www.bctechdays.com

waldo



Eric “waldo” Wauters



Microsoft MVP, Business Applications

me@waldo.be

@waldo1001

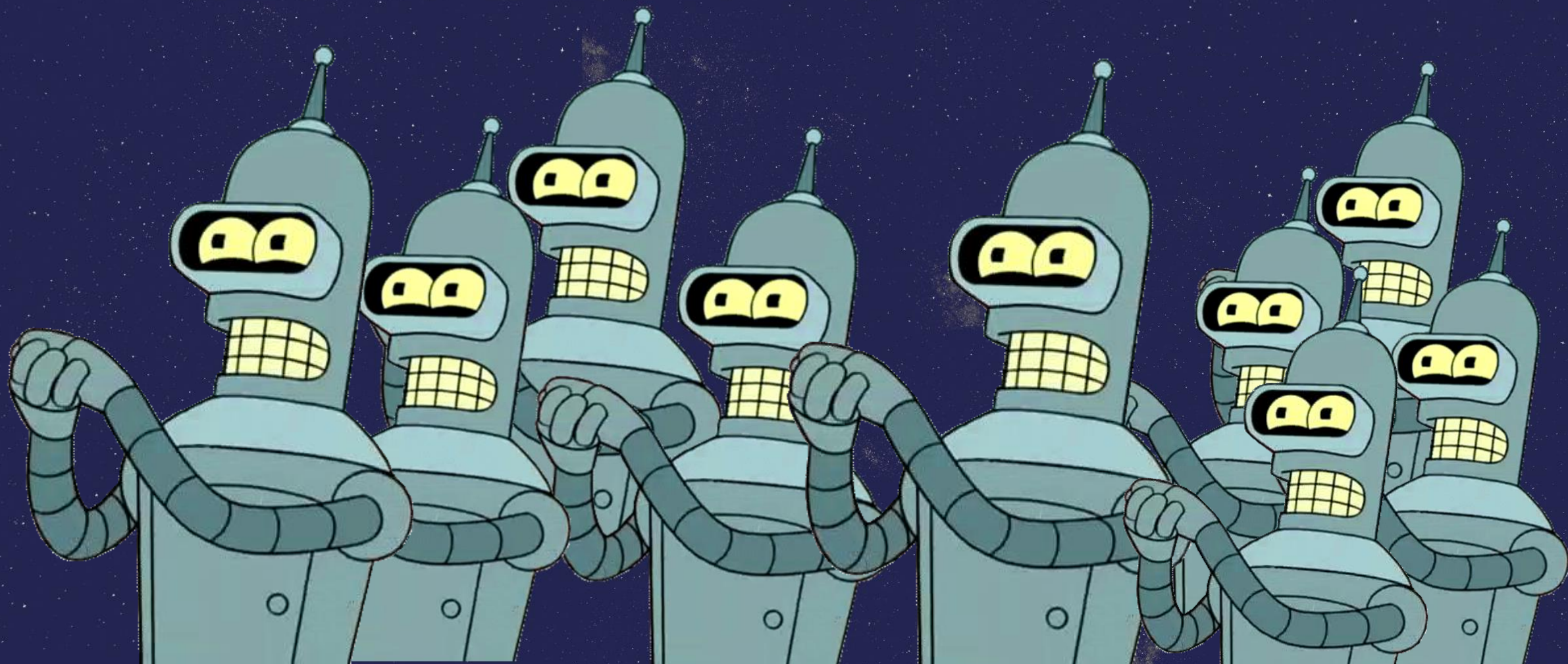
<http://www.waldo.be>

Business:

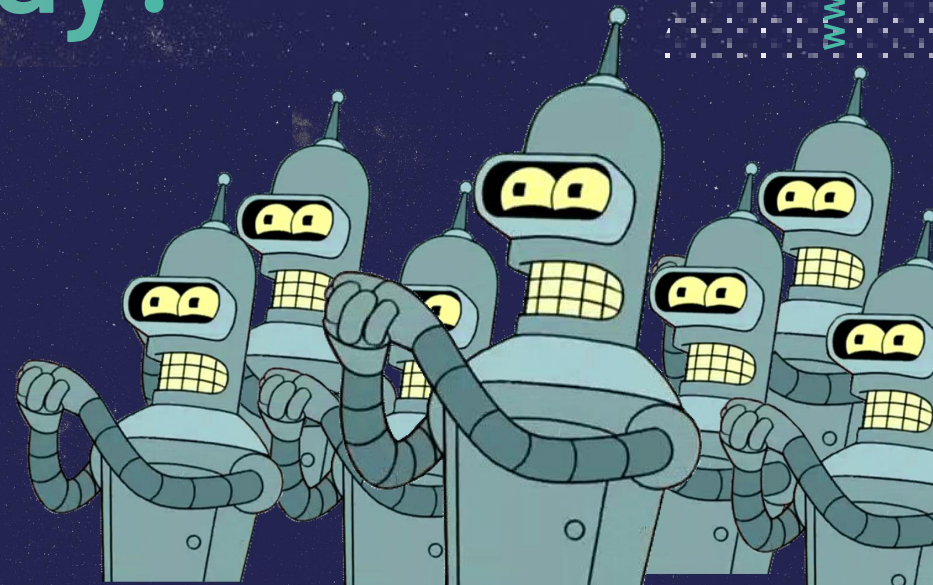
Owner @ Dynex bv

Partner @ iFacto Business Solutions nv

Partner @ Hodor nv



Do you know what we are
celebrating today?



10 YEAR ANNIVERSARY



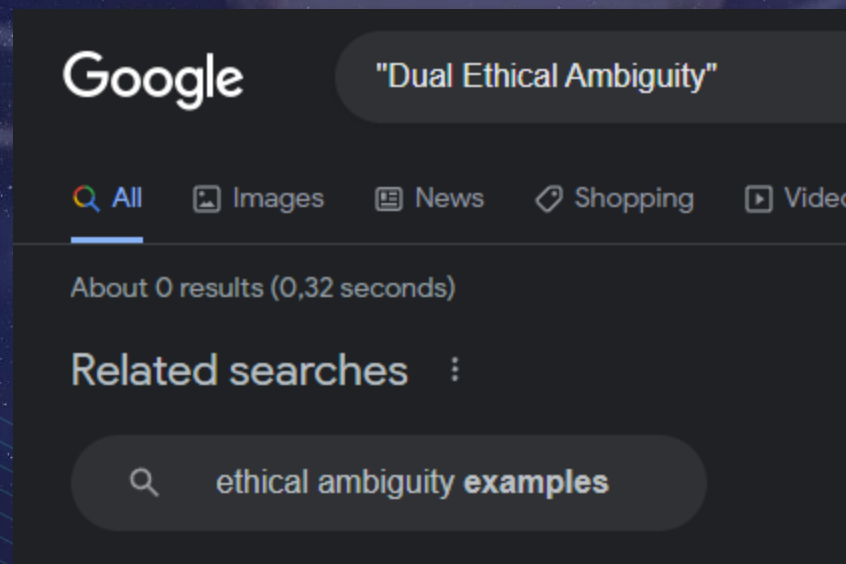
Goals

Coding tips to

- improve **performance** of business logic
- to improve **user experience**

Things to **consider** while choosing for certain scenarios

"Dual ethical ambiguity" ...



The ambiguity in performance challenges

With every “win” comes a “loss”
With every “positive” comes a “negative”
With every “gain” comes a “cost”
Every “pattern” is also an “anti-pattern”

Conclusion

Whatever I talk about
Will be valid in some situations
But will be invalid in other

Conclusion

Whatever I talk about
Will be valid in some situations
But will be invalid in other

It's about understanding what's possible and when
to use it.

Conclusion

Whatever I talk about
Will be valid in some situations
But will be invalid in other

It's about understanding what's possible and when
to use it.

Or in other words.....

10 YEAR ANNIVERSARY
10 YEAR ANNIVERSARY

I'm
E



ght
g

The Demo App and Demo Data

10 YEAR ANNIVERSARY
10 YEAR ANNIVERSARY

The Demo App and Demo Data

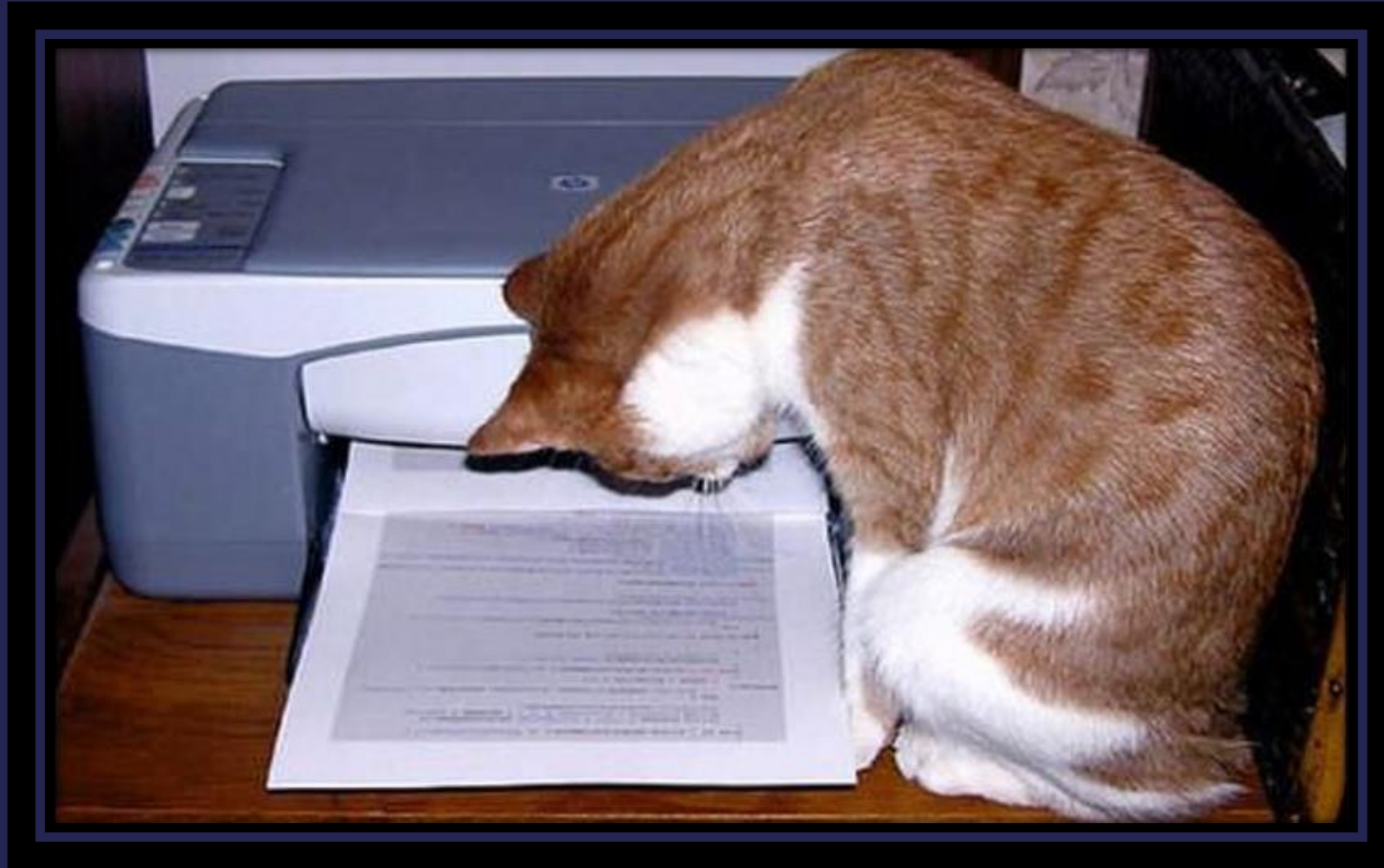
An app that can run performance scenarios

“Just Some Table”:

- A BIG table with LOTS of records
- Not all keys are ok (go figure)

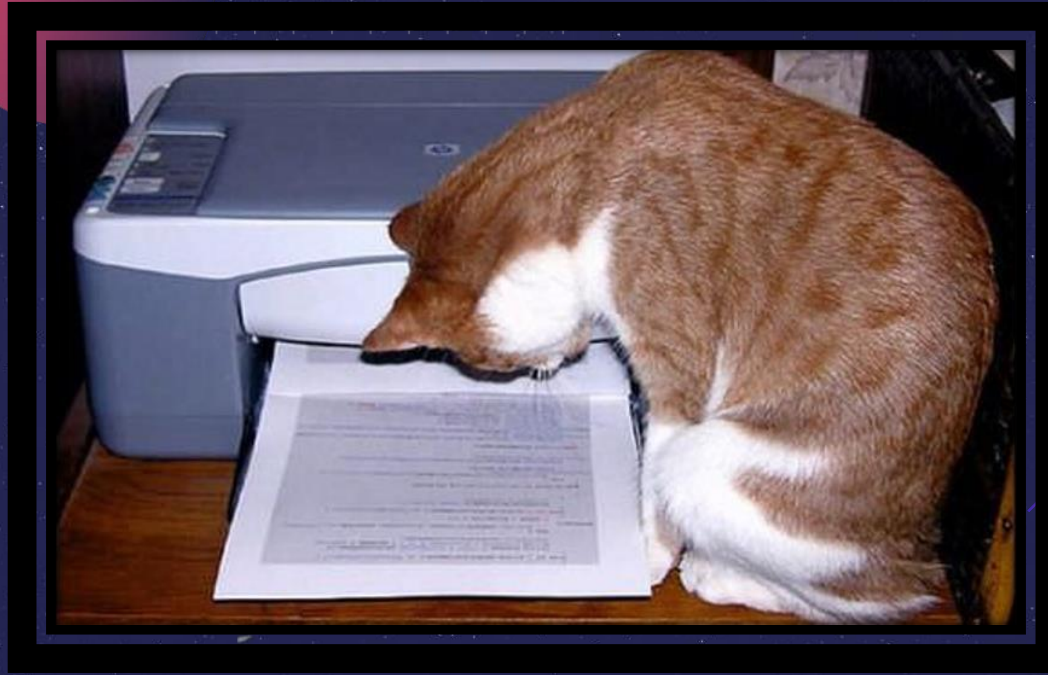
“Just Some Extension Table”:

- 4 tables, with the same amount or records, but which were extended

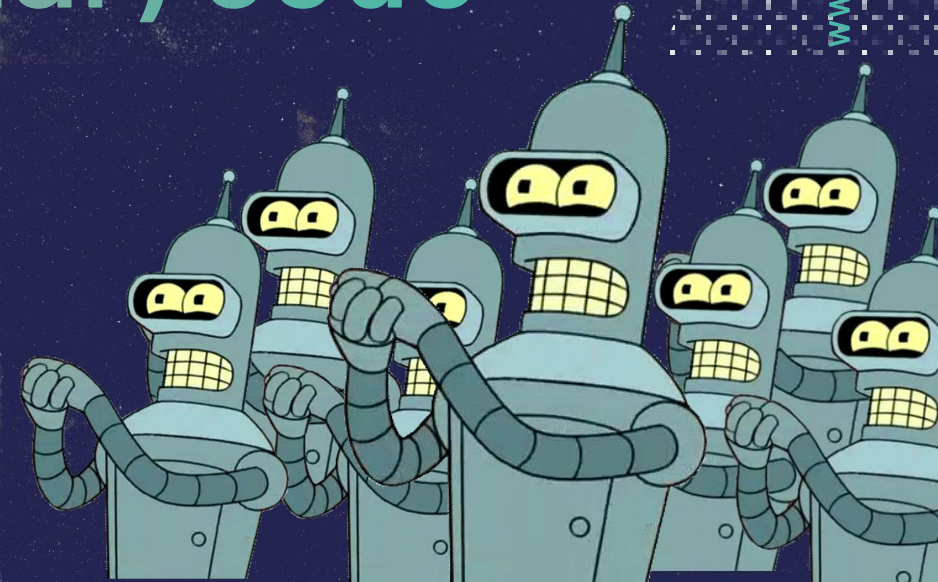


Code Reviews

An essential step to ensure code quality



Who of you do (regular) code
reviews?

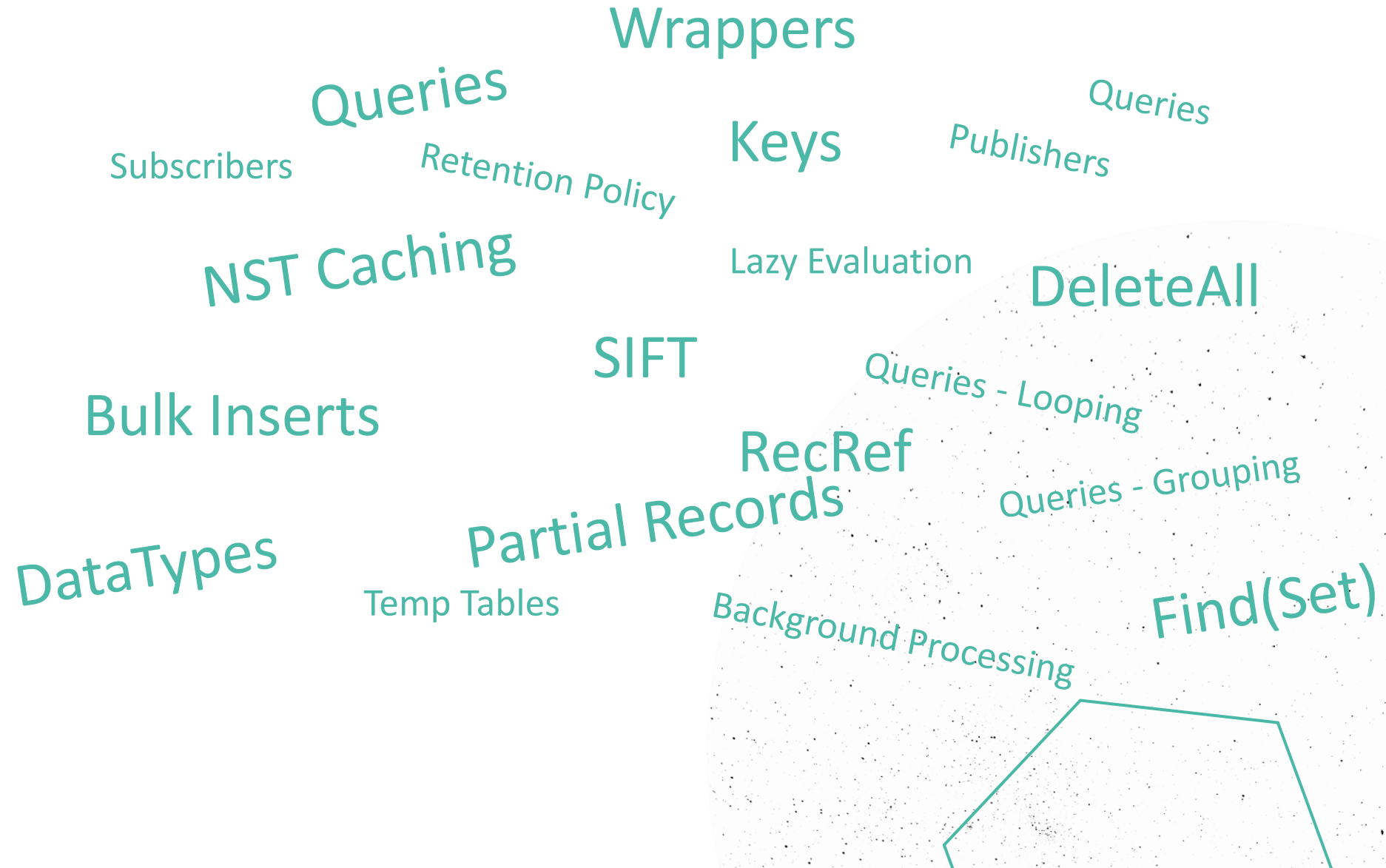


Code Review checklist

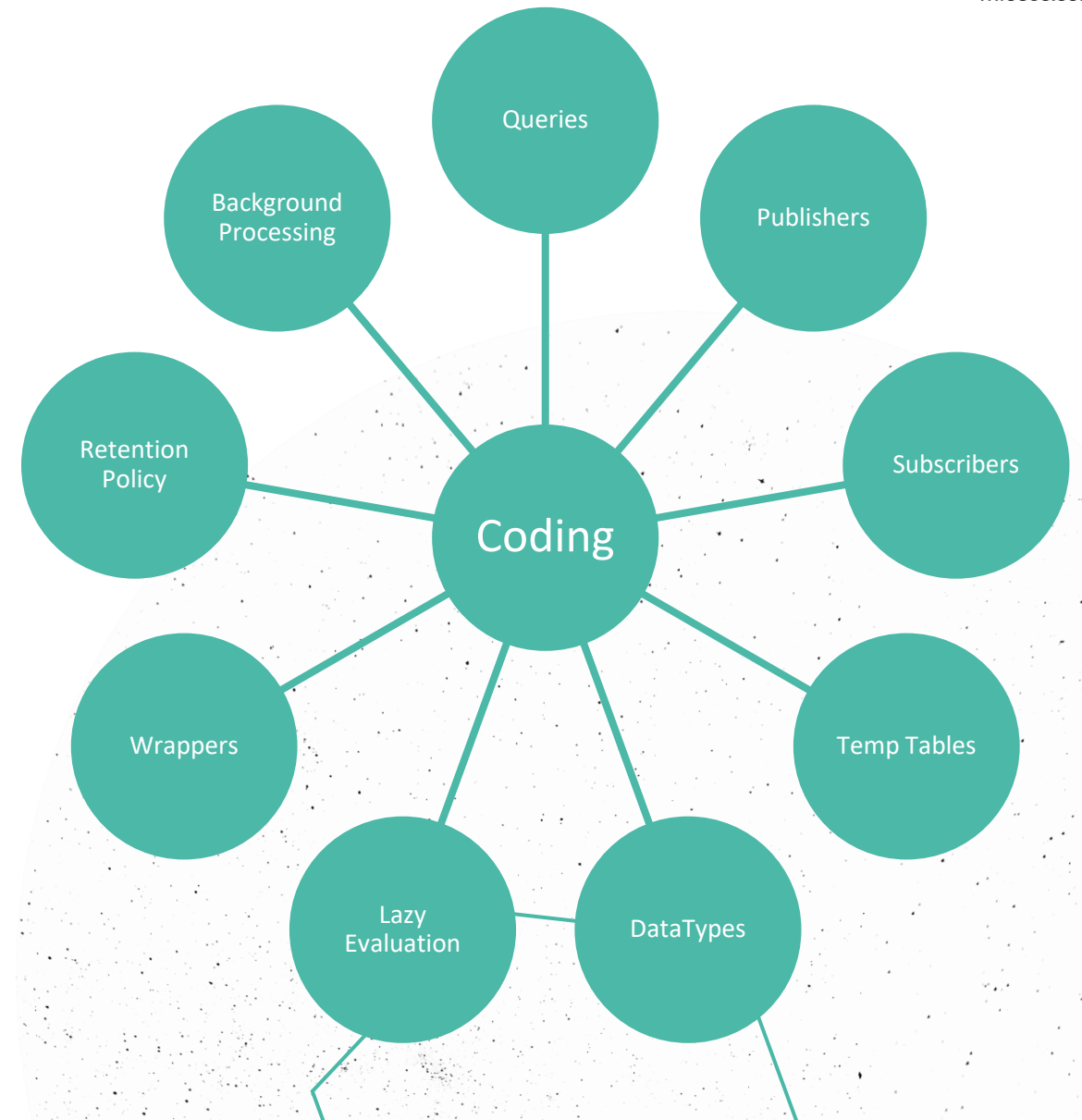
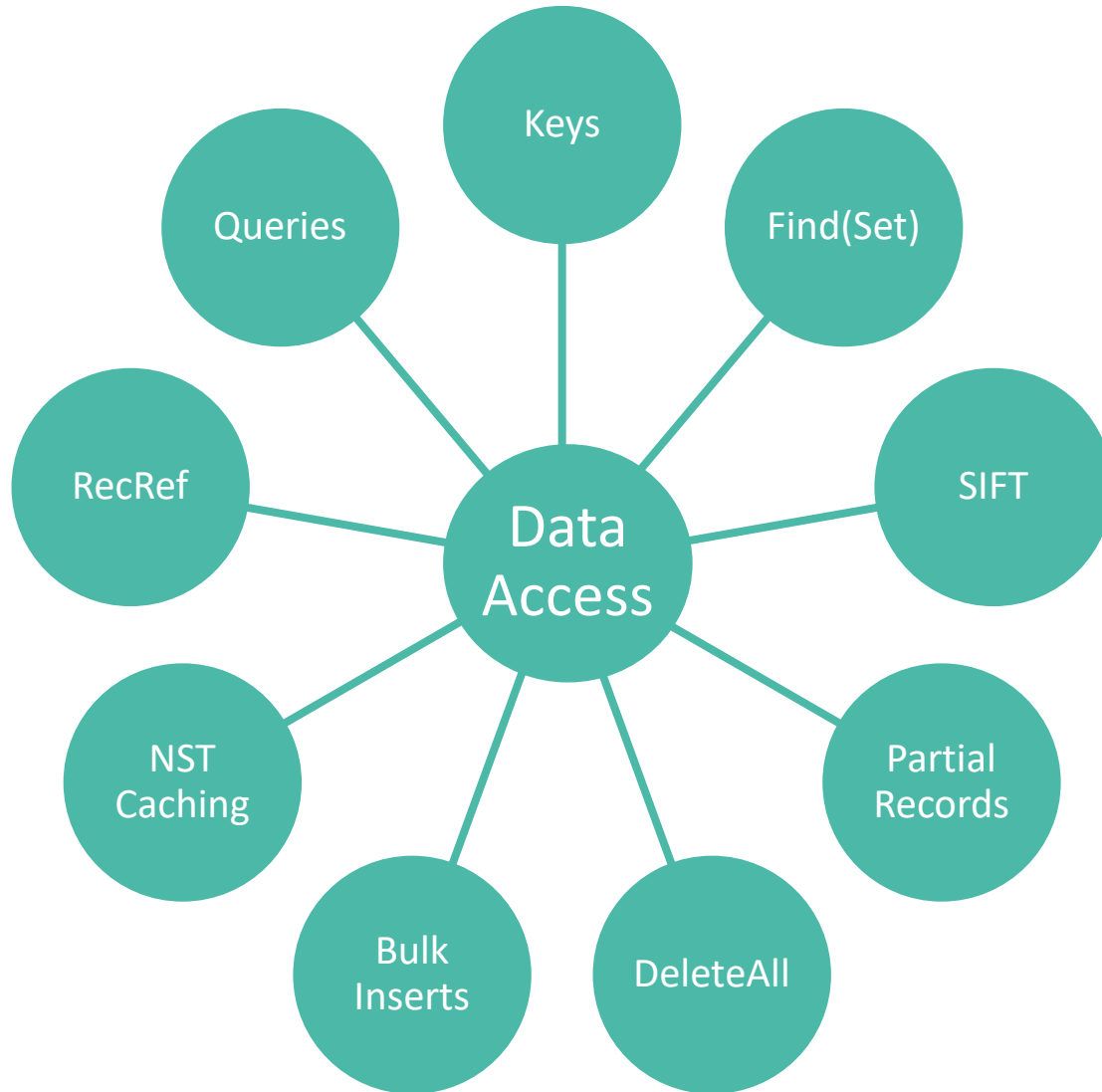


- ✓ Performance
- ✓ Readability (Understandability)
- ✓ Maintainability
- ✓ Extensibility
- ✓ Testability
- ✓ Patterns
- ✓ Reusability
- ✓ Security
- ✓ Documentation
- ✓ Code cloning
- ✓ Scalability
- ✓ Unintentional change of settings
- ✓ ...

Code Review - Performance



Code Review - Performance



Data Access

10 YEAR ANNIVERSARY

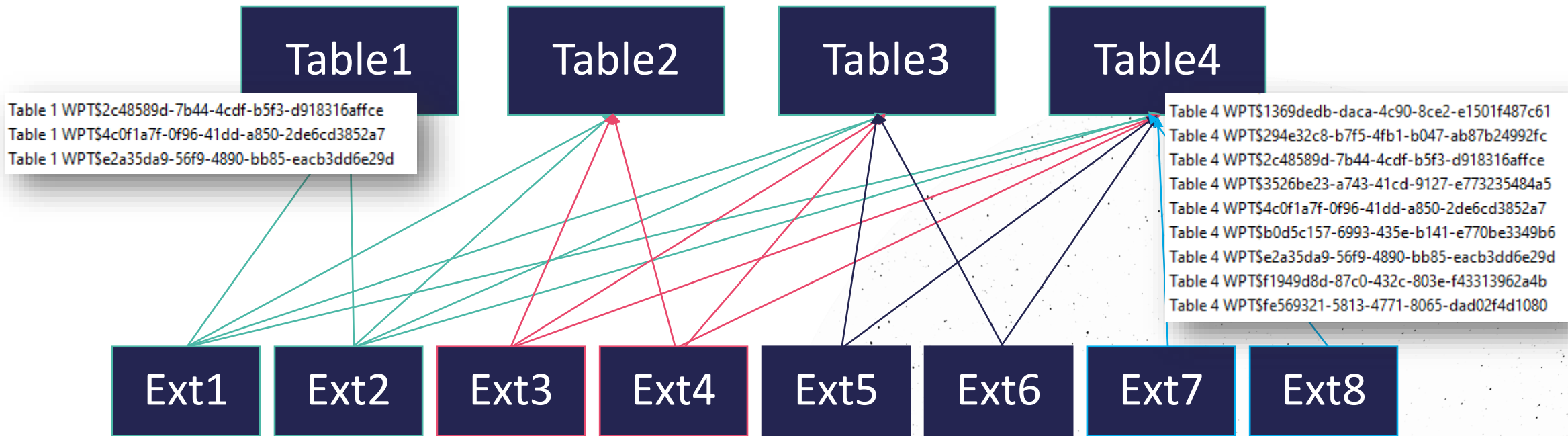
www.bctechdays.com

Let's have a look at some examples in code ...

- Keys
- Find(Set)
- SIFT
- Partial Records
- DeleteAll
- Bulk Inserts



Table Extensions



Conclusion

Keys (IncludedFields)

- For heavy operations – make sure keys are ok
- Let SQL do what it's designed for

Find(Set)

- Lock when necessary (and don't when it's not)
- Use the right FIND-instruction
- Use ISEMPY when applicable

SIFTs (and AutoCalcFields)

- Carefully choose whether to
 - SIFT
 - Included Fields
 - Nothing
- Don't forget about AutoCalcFields

SetLoadFields (Partial Records)

- ALWAYS use SetLoadFields when it makes sense
- You don't know whether the tables will have been extended (in the future)
- Try to avoid BLOBs on TableExtensions

Try to avoid BLOBs on TableExtensions

2022

```
SELECT "62214"."timestamp",  
       "62214"."Entry No_",  
       "62214"."Message",  
       DATALENGTH("62214_e13"."Field55 WPT"),  
       DATALENGTH("62214_e14"."Field50 WPT"),  
       DATALENGTH("62214_e11"."Field30 WPT"),  
       DATALENGTH("62214_e7"."Field10 WPT"),  
       DATALENGTH("62214_e9"."Field15 WPT"),  
       DATALENGTH("62214_e8"."Field5 WPT"),  
       DATALENGTH("62214_e10"."Field20 WPT"),  
       DATALENGTH("62214_e12"."Field40 WPT"),  
       "62214"."$systemId",  
       "62214"."$systemCreatedAt",  
       "62214"."$systemCreatedBy",  
       "62214"."$systemModifiedAt",  
       "62214"."$systemModifiedBy"  
FROM "CRONUS".dbo."CRONUS BE$Just Some Extended Table 4 WPT$2c48589d-7b44-4cdf-b5f3-d918316affce" "62214" WITH(READUNCOMMITTED)  
JOIN "CRONUS".dbo."CRONUS BE$Just Some Extended Table 4 WPT$1369dedb-daca-4c90-8ce2-e1501f487c61" "62214_e13" WITH(READUNCOMMITTED)  
JOIN "CRONUS".dbo."CRONUS BE$Just Some Extended Table 4 WPT$294e32c8-b7f5-4fb1-b047-ab87b24992fc" "62214_e14" WITH(READUNCOMMITTED)  
JOIN "CRONUS".dbo."CRONUS BE$Just Some Extended Table 4 WPT$3526be23-a743-41cd-9127-e773235484a5" "62214_e11" WITH(READUNCOMMITTED)  
JOIN "CRONUS".dbo."CRONUS BE$Just Some Extended Table 4 WPT$4c0f1a7f-0f96-41dd-a850-2de6cd3852a7" "62214_e7" WITH(READUNCOMMITTED)  
JOIN "CRONUS".dbo."CRONUS BE$Just Some Extended Table 4 WPT$b0d5c157-6993-435e-b141-e770be3349b6" "62214_e9" WITH(READUNCOMMITTED)  
JOIN "CRONUS".dbo."CRONUS BE$Just Some Extended Table 4 WPT$e2a35da9-56f9-4890-bb85-eacb3dd6e29d" "62214_e8" WITH(READUNCOMMITTED)  
JOIN "CRONUS".dbo."CRONUS BE$Just Some Extended Table 4 WPT$f1949d8d-87c0-432c-803e-f43313962a4b" "62214_e10" WITH(READUNCOMMITTED)  
JOIN "CRONUS".dbo."CRONUS BE$Just Some Extended Table 4 WPT$fe569321-5813-4771-8065-dad02f4d1080" "62214_e12" WITH(READUNCOMMITTED)  
ORDER BY "Entry No_" ASC OPTION(FAST 50)
```

SetLoadFields(Message)

Blob-fields in TableExtensions

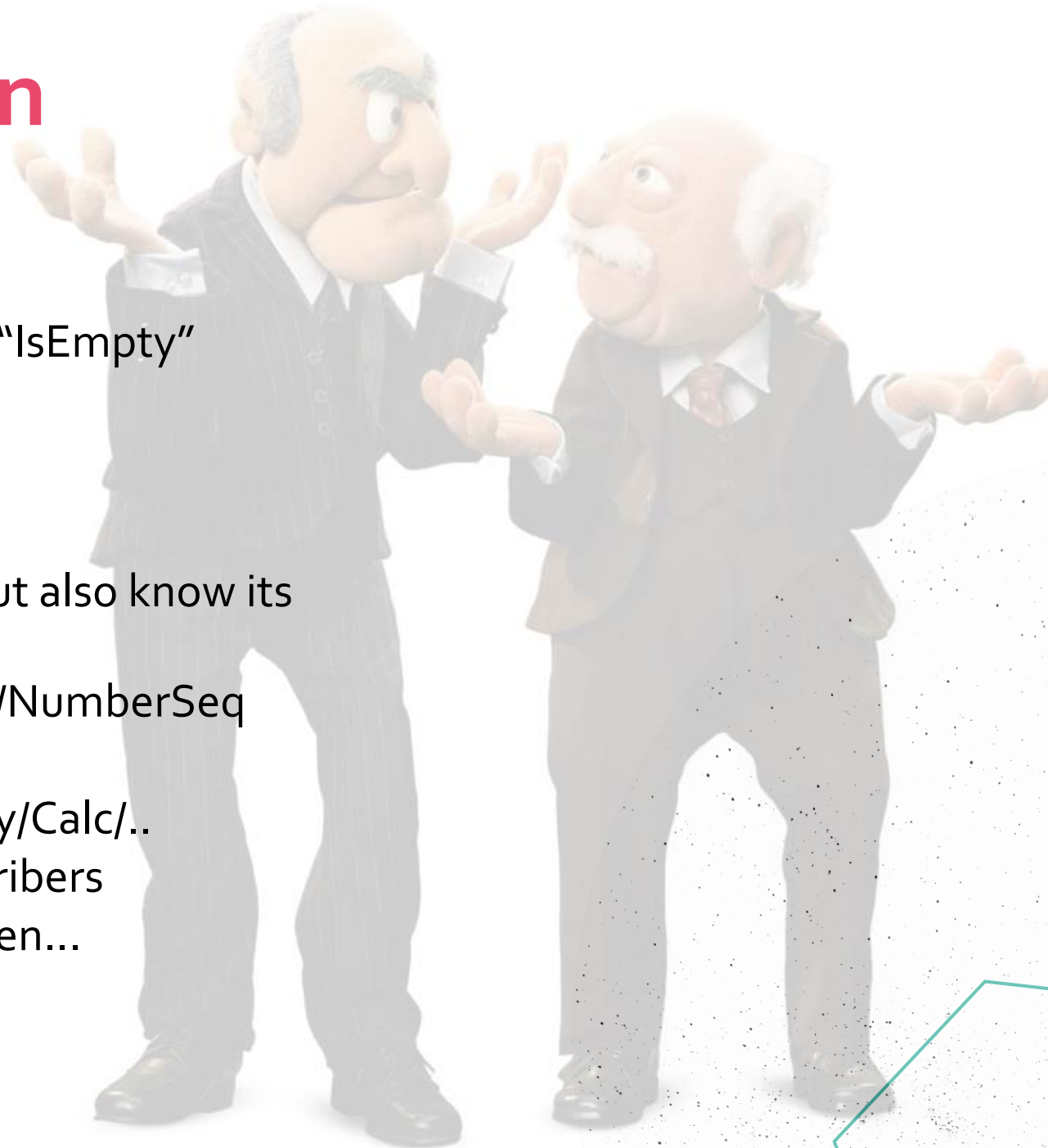
Conclusion

DeleteAll

- Don't forget the "IsEmpty"

Bulk Inserts

- On by default
- Know it exists, but also know its constraints:
 - AutoIncrement/NumberSeq
 - Blobs
 - Get/Find/Modify/Calc/..
 - OnInsert-subscribers
 - IF Rec.Insert Then...





A glimpse into the future ...

V21 – DataTransfer Object Type

New AL type for transferring “huge” amount of data between tables

Happens a lot in upgrade-scenarios

- Move data between extension-tables
- Move data from obsoleted fields/tables

Supports:

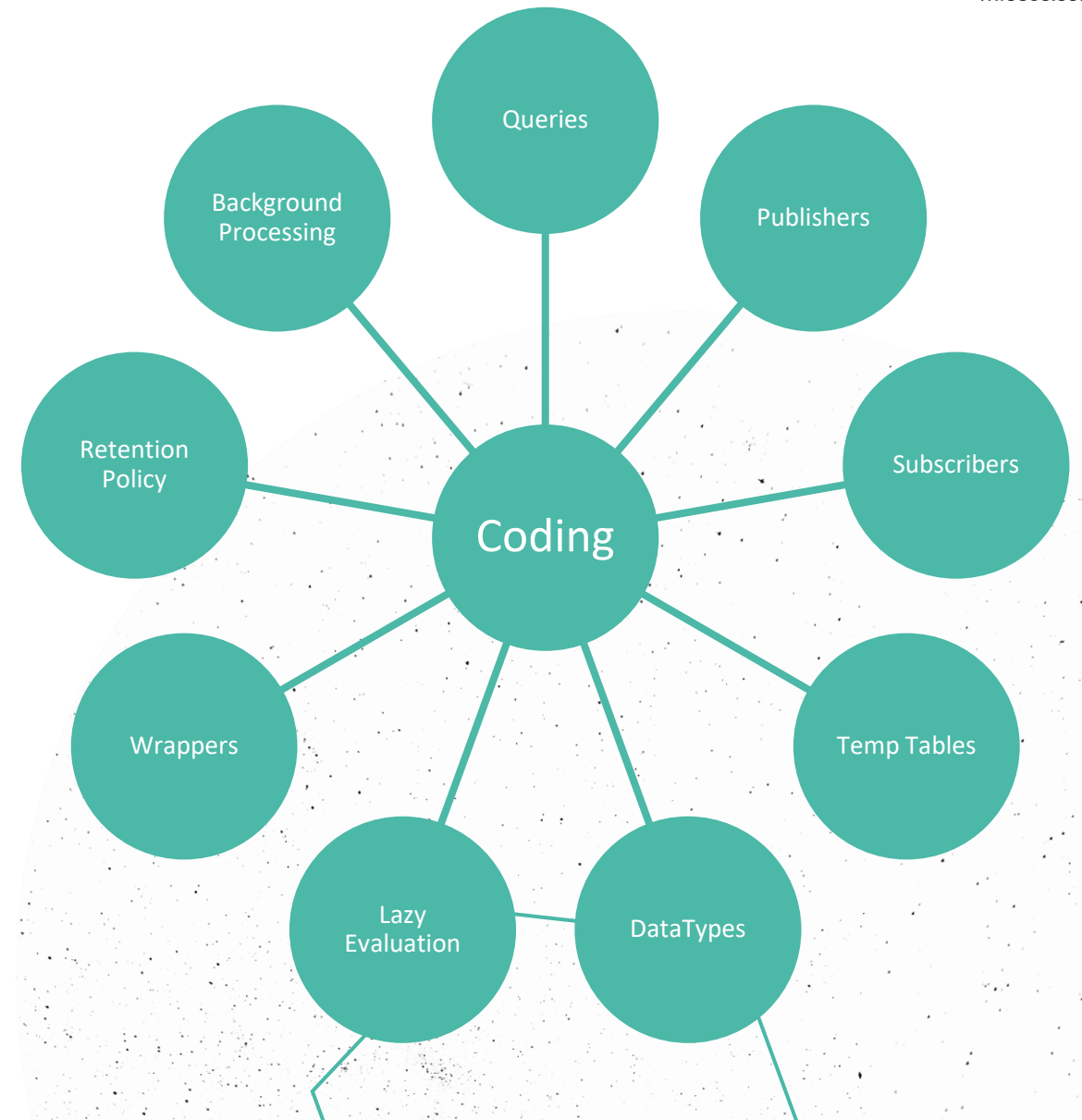
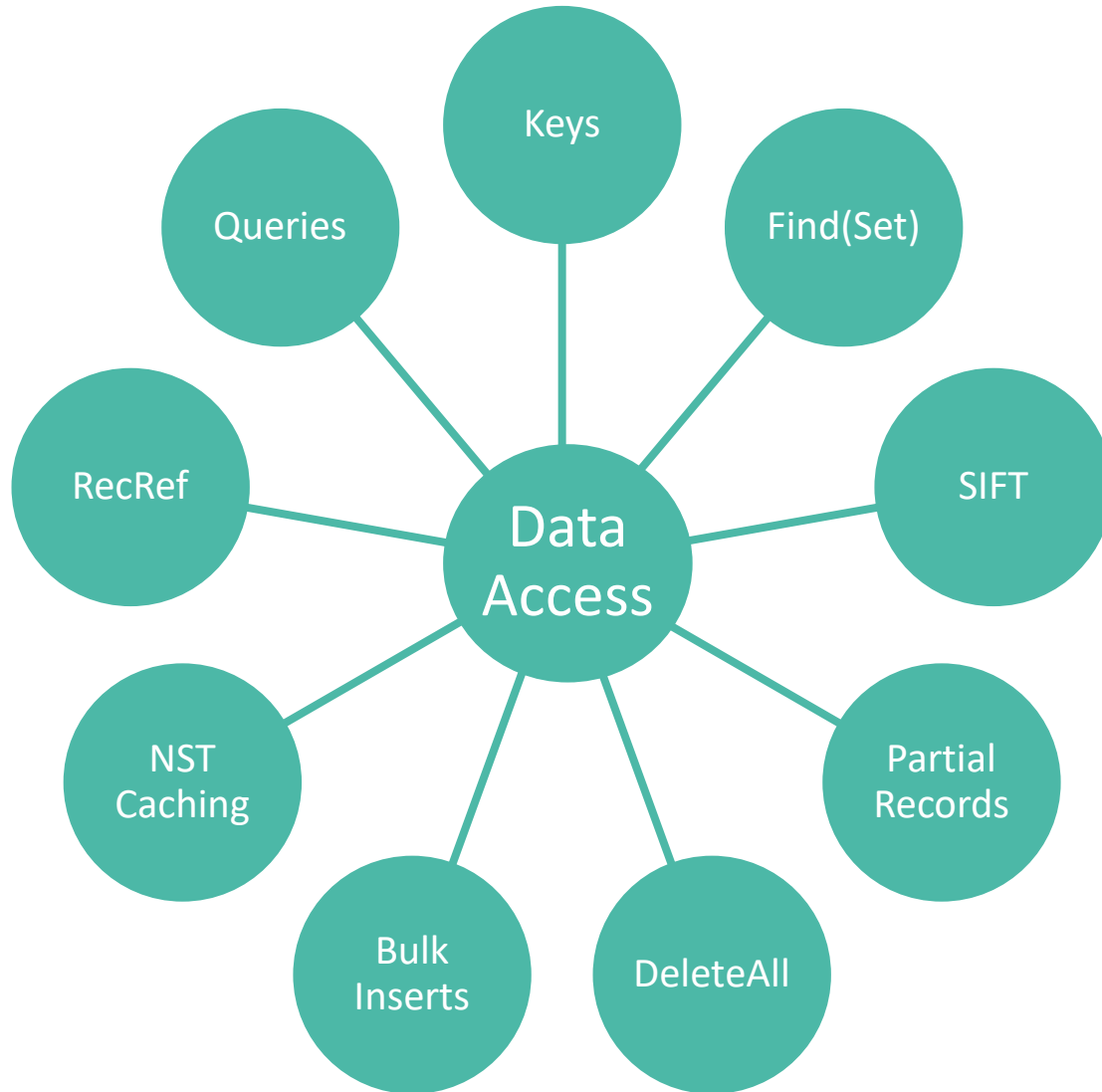
- CopyFields
- CopyRows

Does NOT run triggers, nor events → Only runs during Upgrade

DataTransfer



Code Review - Performance



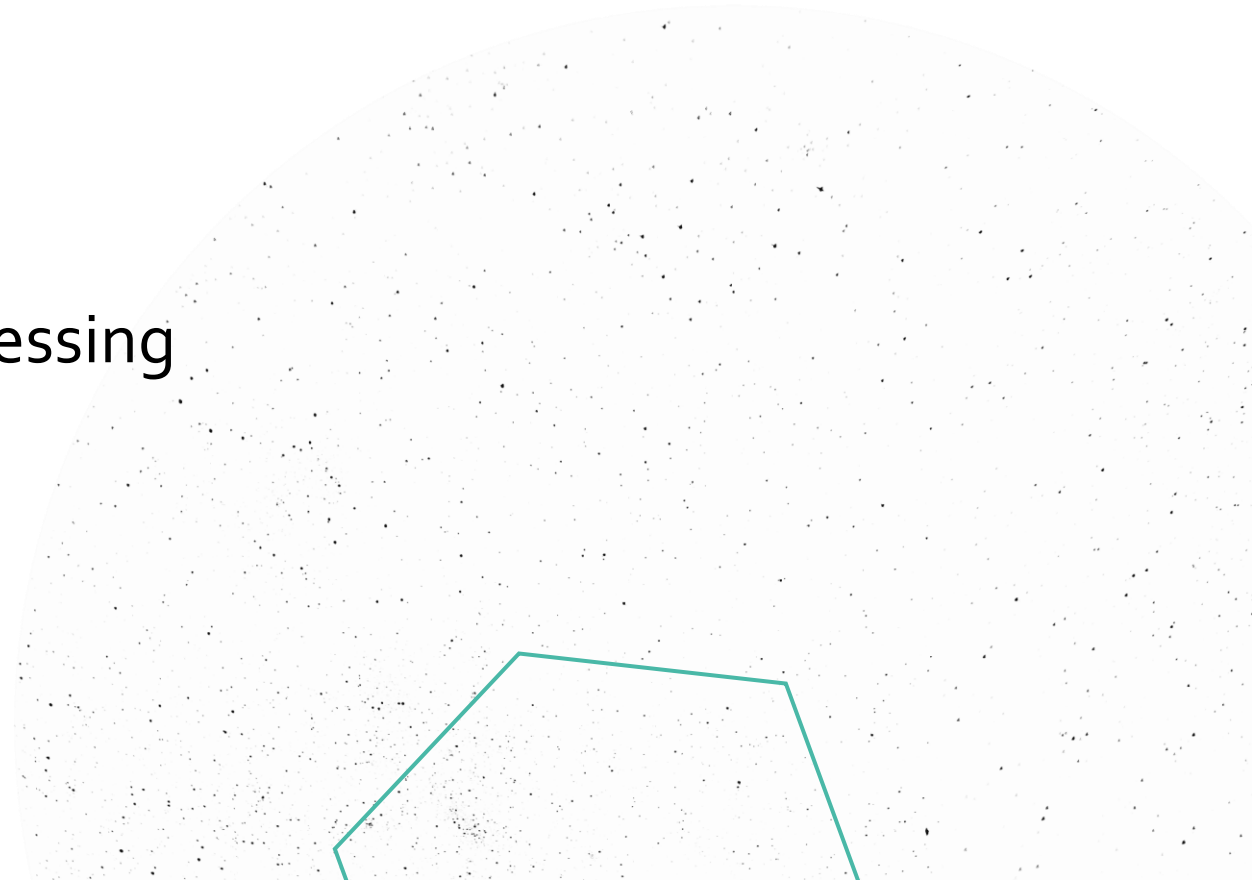
Coding

10 YEAR ANNIVERSARY

www.bctechdays.com

Let's have a look at some examples in code ...

- Queries
- Publishers
- Subscribers
- DataTypes
- Lazy Evaluation
- Retention Policy
- Background Processing



Conclusion

Queries

- Great for getting joined data
- Great for grouping and totalling
- Also benefit from SIFT and covering indexes
- No benefit from NST Caching
- Test alternatives

Publishers

- Go nuts!
- Check out the [Generic Method Pattern](#)

Subscribers

- DOES impact performance
- Be as efficient as you can
- Use SingleInstance
- Codeunit size doesn't matter, but it does impact readability
- Mind global vars though
- Avoid OnInsert/OnModify/OnDelete
- Subscription binding only helps performance because when business logic isn't executed – it still kills ModifyAll/DeleteAll and Bulk Inserts

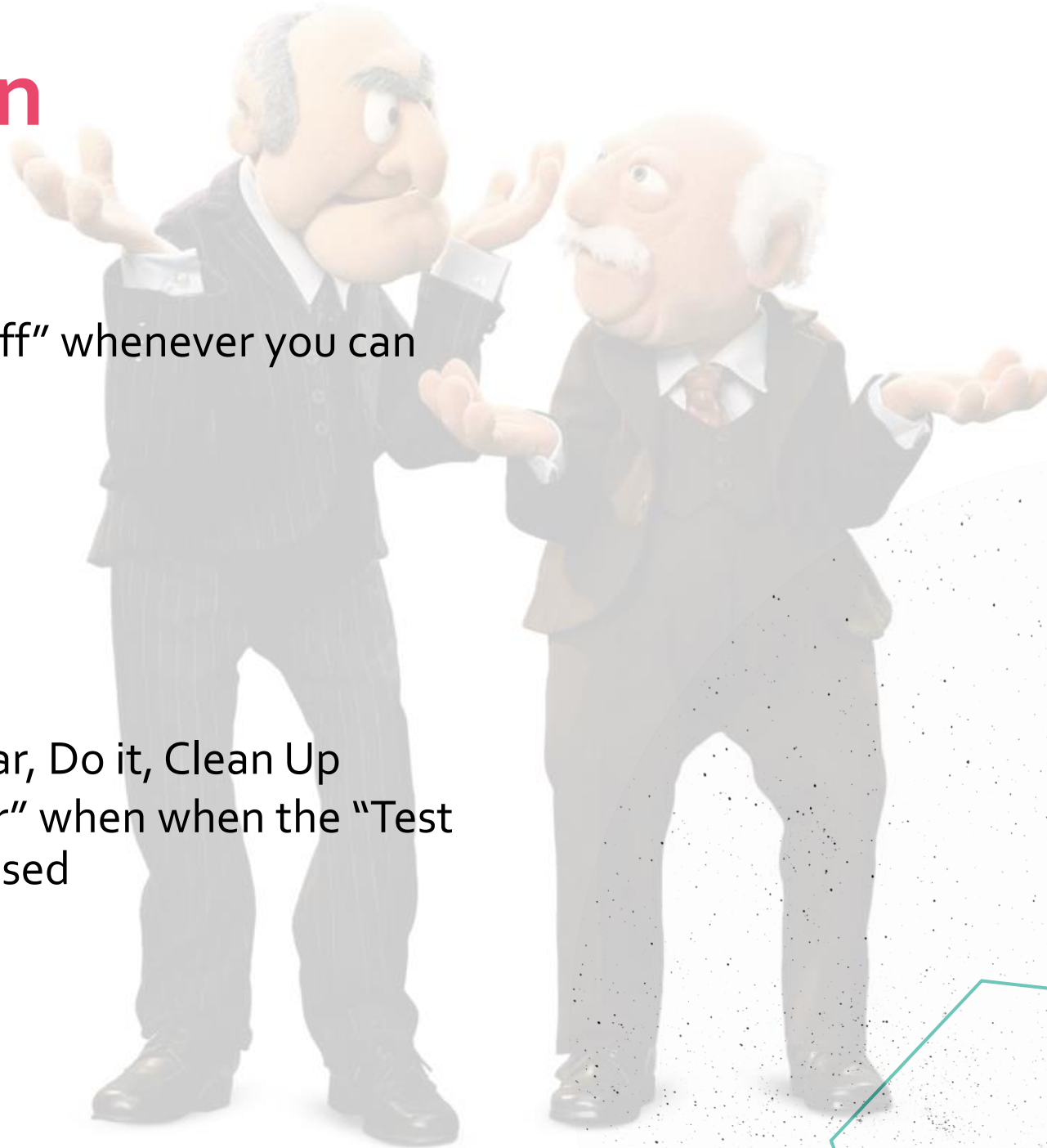
Conclusion

Data Types

- Use “the new stuff” whenever you can
 - TextBuilder
 - Dictionaries
 - JSON types

Lazy Evaluation

- Remember it
- Test Near, Test Far, Do it, Clean Up
 - Only “Test Far” when the “Test Near” has passed



If you can't make the code faster,
you might be able to do something
about the user experience

Background Processing

10 YEAR ANNIVERSARY

Background Processing

StartSession

- Created immediately
- Runs on same server
- Does not survive a server restart
- Page Background Task

TaskScheduler

- Queued
- Only a limited amount of "workers" running at the same time.
- Any server in a cluster can start it
- Survives server restarts
- No logging

Job queue

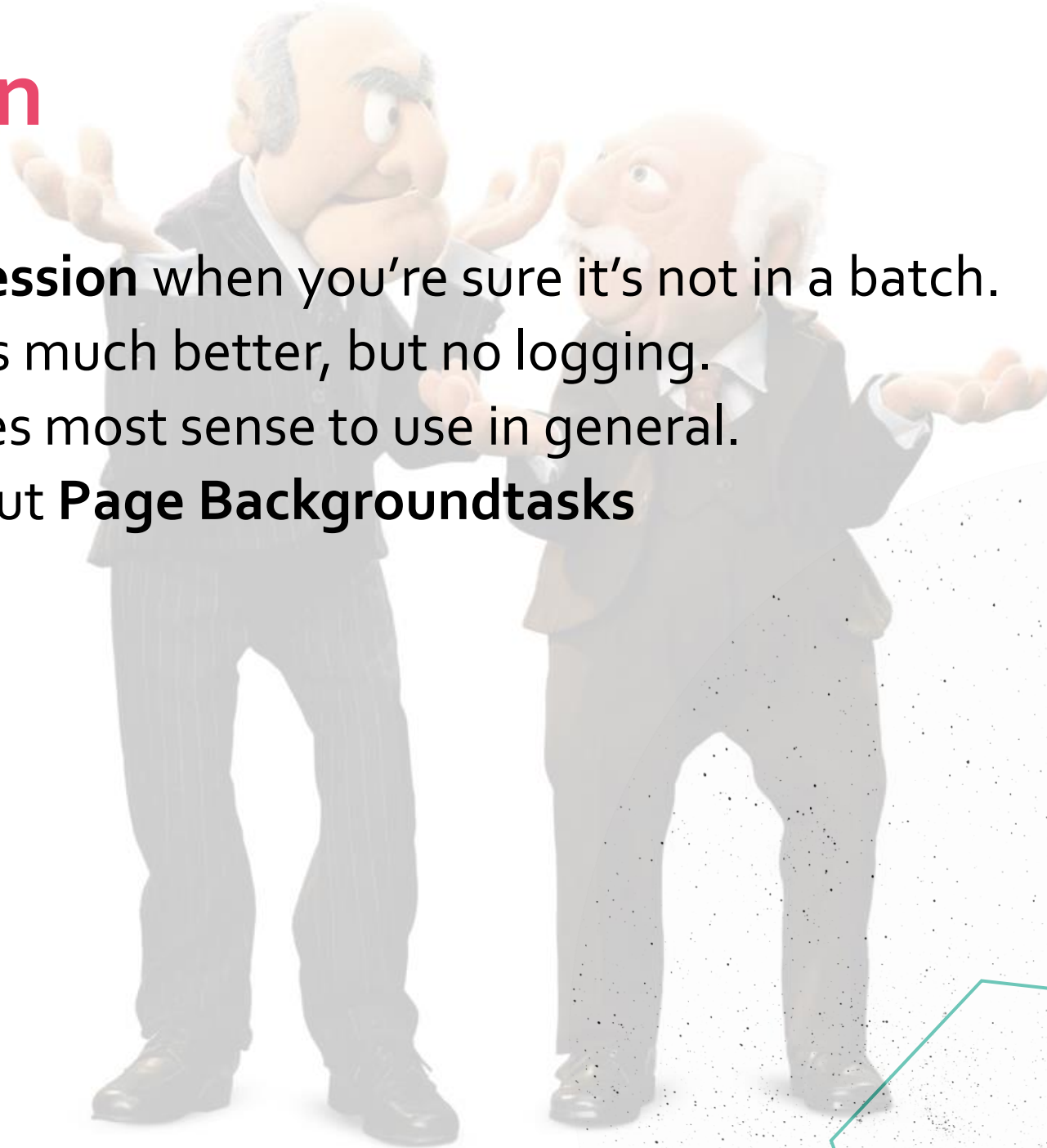
- Uses the TaskScheduler
- Defined Schedules
- Recurrence
- Any server in a cluster can start it
- Survives server restarts
- Logging of results

Page Background Task

- Can (will) be canceled
- Read-only
- Call back to parent session
- Lightweight

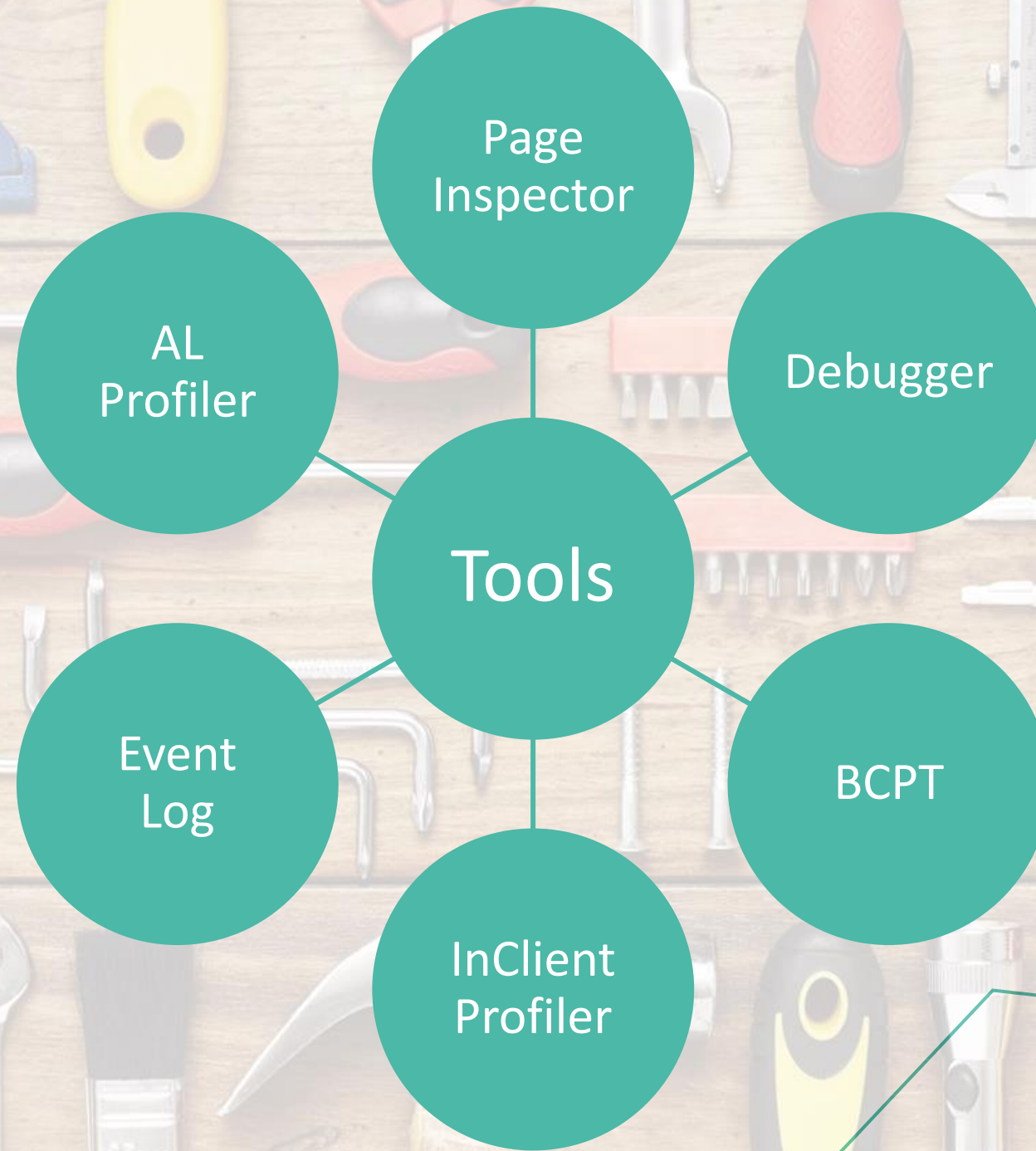
Conclusion

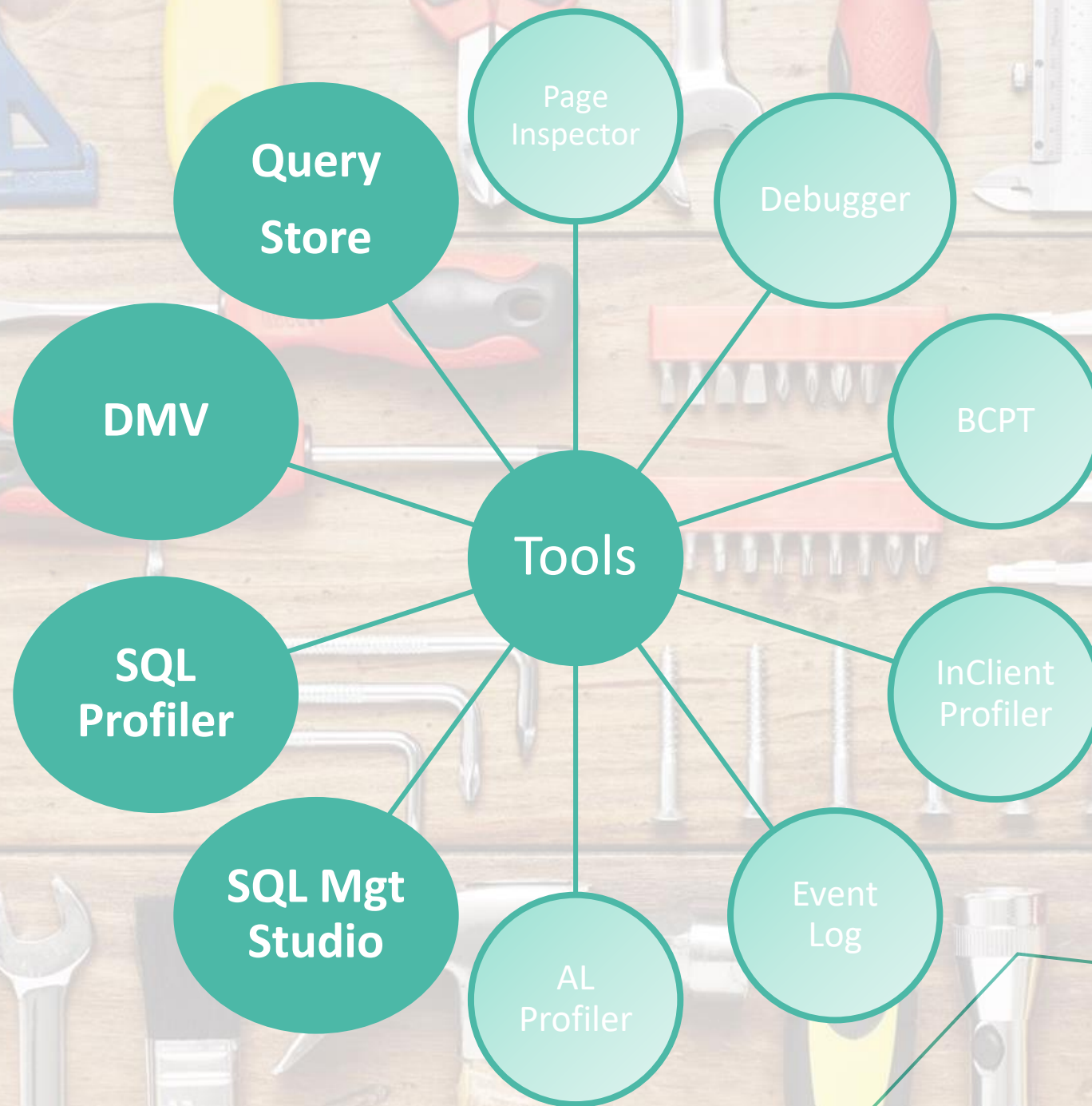
Only use **StartSession** when you're sure it's not in a batch.
TaskScheduler is much better, but no logging.
Job Queue makes most sense to use in general.
Don't forget about **Page Backgroundtasks**

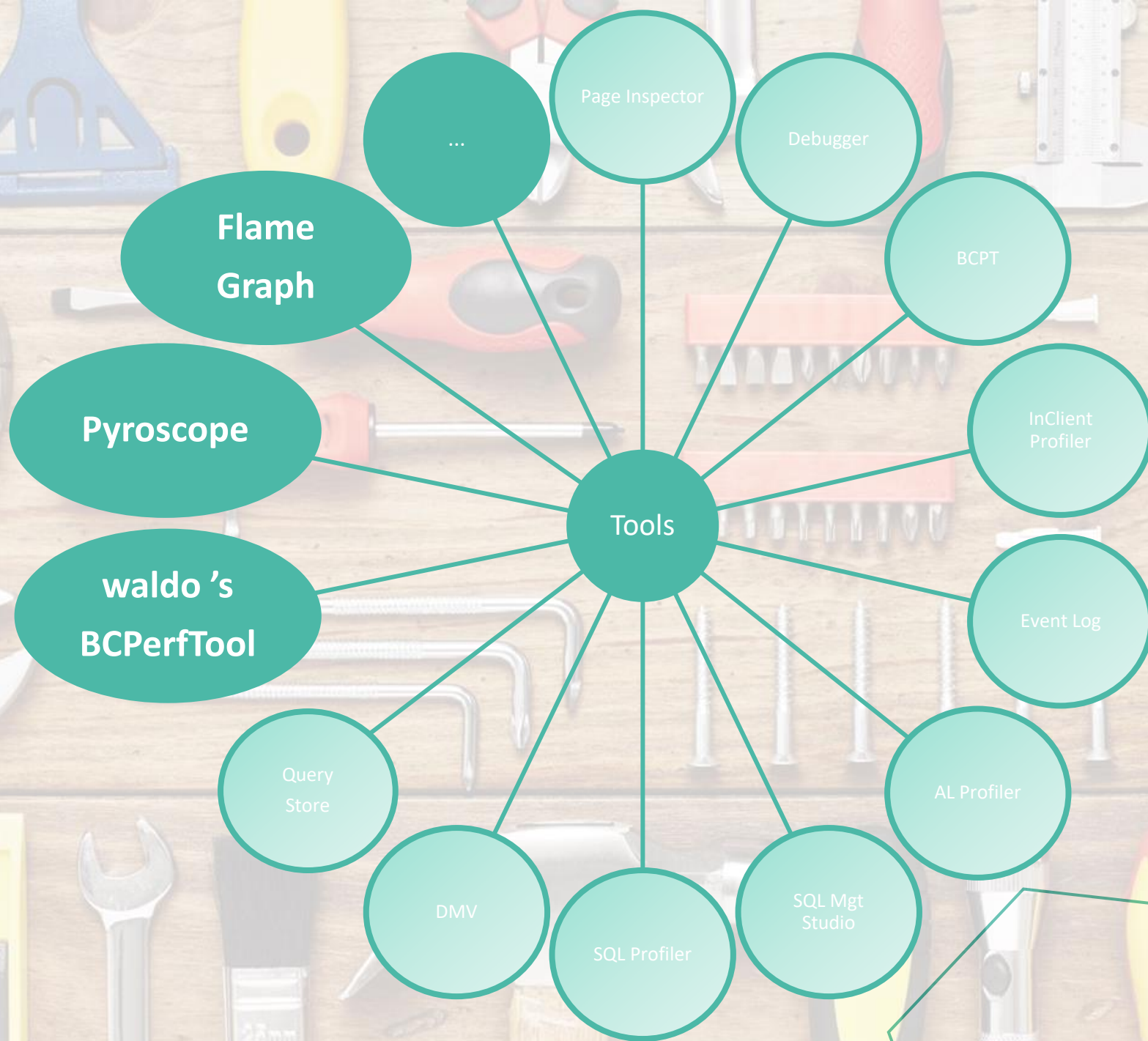


A top-down view of a wooden workbench covered with a variety of hand tools. The tools are organized into several rows. The top row includes a red level, yellow handles, a blue square, a pair of pliers, a wrench, a red-handled screwdriver, a caliper, and a green-handled screwdriver. The second row features a large black and red plier, a red-handled screwdriver, a set of sockets in a red holder, a yellow tape measure, and a pair of side cutters. The third row shows a silver adjustable wrench, several hex keys, a row of screws, and a yellow spirit level. The bottom row contains a red-handled screwdriver, a yellow level, a wrench, a brush, a claw hammer, a flashlight, a yellow-handled screwdriver, a red-handled screwdriver, and a black tool. The word "Tools" is centered in a large, white, sans-serif font over the middle of the image.

Tools









Debugger
Flame Graph
Pyroscope
waldo.BCPerfTool

Conclusion

Debugger

- Get locking info
- Get SQL Info
- For SQL profiling – there's still the SQL Profiler

FlameGraph

..





Torben Løkke Leth

<https://blog.SShadowS.dk/>

Twitter: @SXShadowXS

FlameGraph for Business Central

Source: <https://github.com/SShadowS/AL-Flamegraph>

What?

Service to generate, based on the .alcpuprofile:

- An svg – generated using [FlameGraph by Brendan Gregg](#)
- Folded files – used by [Pyroscope](#)

Want to know more? Keep an eye on Torben's blog!

<https://blog.SShadowS.dk/>

and/or, have a look at ...

waldo.BCPerfTool

10 YEAR ANNIVERSARY

www.bctechdays.com

waldo.BCPerfTool

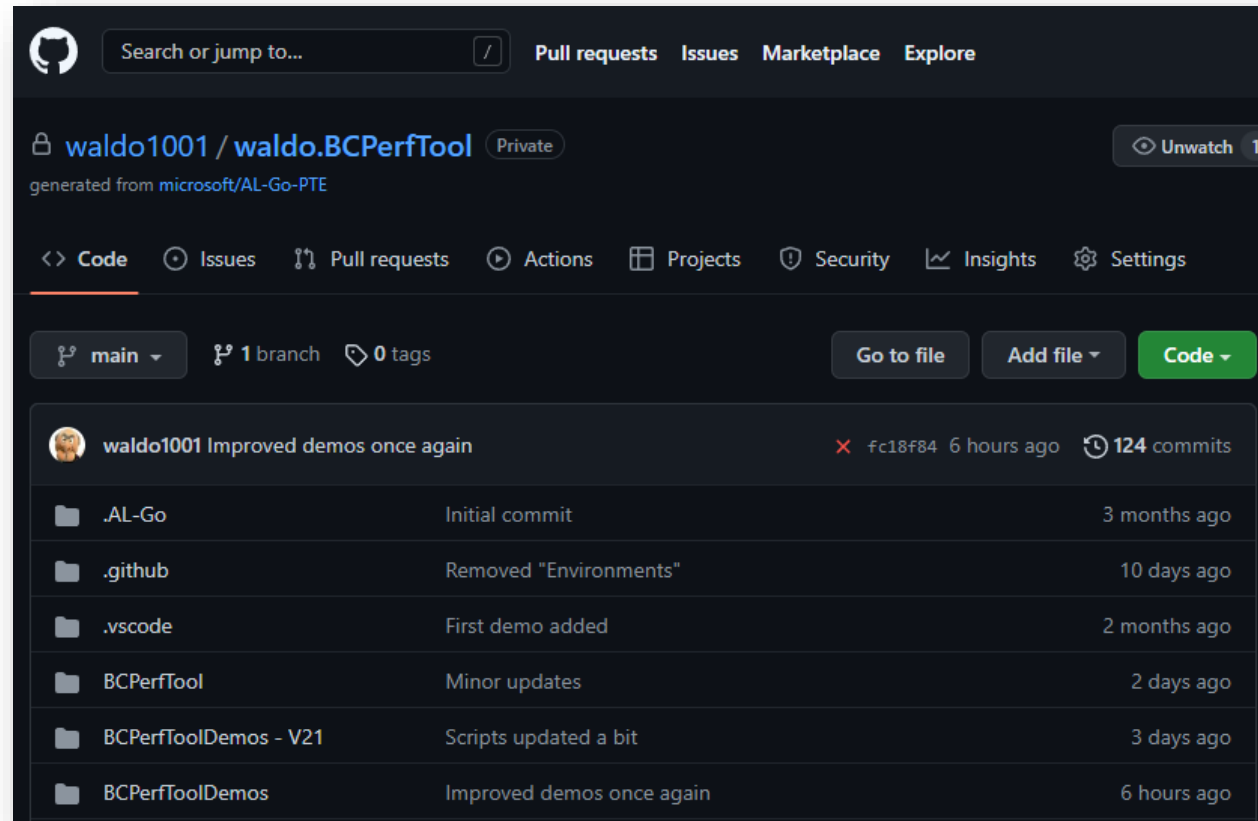
- Easily run Performance Tests
- Not using TestRunners (!)
- SelectLatestVersion
- Performance Profiling (Sampling)
 - FlameGraph
 - Pyroscope (Flamegraph diff)
- Batch running to get averages
- Graphs
 - Averages
 - Actuals
 - FlameGraph
- Run and analyze any Object
- Adding your own tests
 - PerfTool-Interface (with Snippet)

```
"Snippet: PerfToolCodeunit": {
  "prefix": "tPerfToolCodeunitWaldo",
  "body": [
    "codeunit ${1:Id} ${2:MyCodeunit} implements \"PerfToolCodeunit WPT\"",
    "{",
    "\t#region ${3:NameOfTheTest}",
    "\tprocedure ${3:NameOfTheTest}()",
    "\tbegin",
    "\t\t$0",
    "\tend;",
    "\t#endregion",
    "",
    "\t#region InterfaceImplementation",
    "\tprocedure Run(ProcedureName: Text) Result: Boolean;",
    "\tbegin",
    "\t\tcase ProcedureName of",
    "\t\t\tGetProcedures().Get(1):",
    "\t\t\t\t\t${3:NameOfTheTest}()",
    "\t\t\tend;",
    "",
    "\t\tResult := true;",
    "\tend;",
    "",
    "\tprocedure GetProcedures() Result: List of [Text[50]];",
    "\tbegin",
    "\t\tResult.Add('${3:NameOfTheTest}');",
    "\tend;",
    "",
    "\t[EventSubscriber(ObjectType::Codeunit, Codeunit::\"Install Suites WPT\", 'OnInstall', false, false, false, false, false)]",
    "\tlocal procedure OnAfterInsertSuiteGroup()",
    "\tvar",
    "\t\tPerfToolSuiteHeaderWPT: Record \"PerfTool Suite Header WPT\";",
    "\t\tWPTSuiteLine: Record \"PerfTool Suite Line WPT\";",
    "\t\tPerfToolGroupWPT: Record \"PerfTool Group WPT\";",
    "\t\tCreatePerfToolDataLibraryWPT: Codeunit \"Create PerfToolDataLibrary WPT\";",
    "\tbegin",
    "\t\tCreatePerfToolDataLibraryWPT.CreateGroup('${4:PerfToolGroupCode}', '${5:PerfToolSuiteCode}');",
    "\t\tCreatePerfToolDataLibraryWPT.CreateSuite(PerfToolGroupWPT, '${6:PerfToolSuiteCode}');",
    "\t\tCreatePerfToolDataLibraryWPT.CreateSuiteLines(PerfToolSuiteHeaderWPT, WPTSuiteLine);",
    "\tend;",
    "\t#endregion",
    "}",
    "",
    "// Cut/Paste this next line to the enumextension for \"PerfToolCodeunit WPT\"",
    "value(${1:Id}; \"${6:PerfToolSuiteCode}\") { Implementation = \"PerfToolCodeunit WPT\" }
  ]
}
```


waldo.BCPerfTool

Where?

<https://github.com/waldo1001/waldo.BCPerfTool>



Search or jump to... / Pull requests Issues Marketplace Explore

waldo1001 / waldo.BCPerfTool Private Unwatch 1

generated from microsoft/AL-Go-PTE

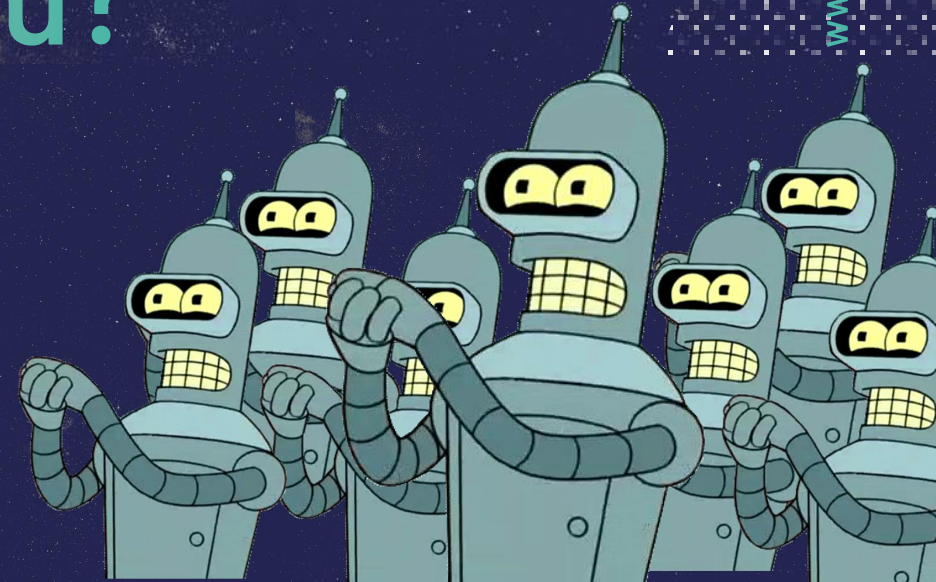
<> Code Issues Pull requests Actions Projects Security Insights Settings

main 1 branch 0 tags Go to file Add file Code

waldo1001	Improved demos once again	fc18f84 6 hours ago	124 commits
.AL-Go	Initial commit		3 months ago
.github	Removed "Environments"		10 days ago
.vscode	First demo added		2 months ago
BCPerfTool	Minor updates		2 days ago
BCPerfToolDemos - V21	Scripts updated a bit		3 days ago
BCPerfToolDemos	Improved demos once again		6 hours ago

I'm up for a beer.

What about you?





Any Questions?

Thank
You!



mibuso.com

Dinner

10 YEAR ANNIVERSARY
10 YEAR ANNIVERSARY

www.bctechdays.com