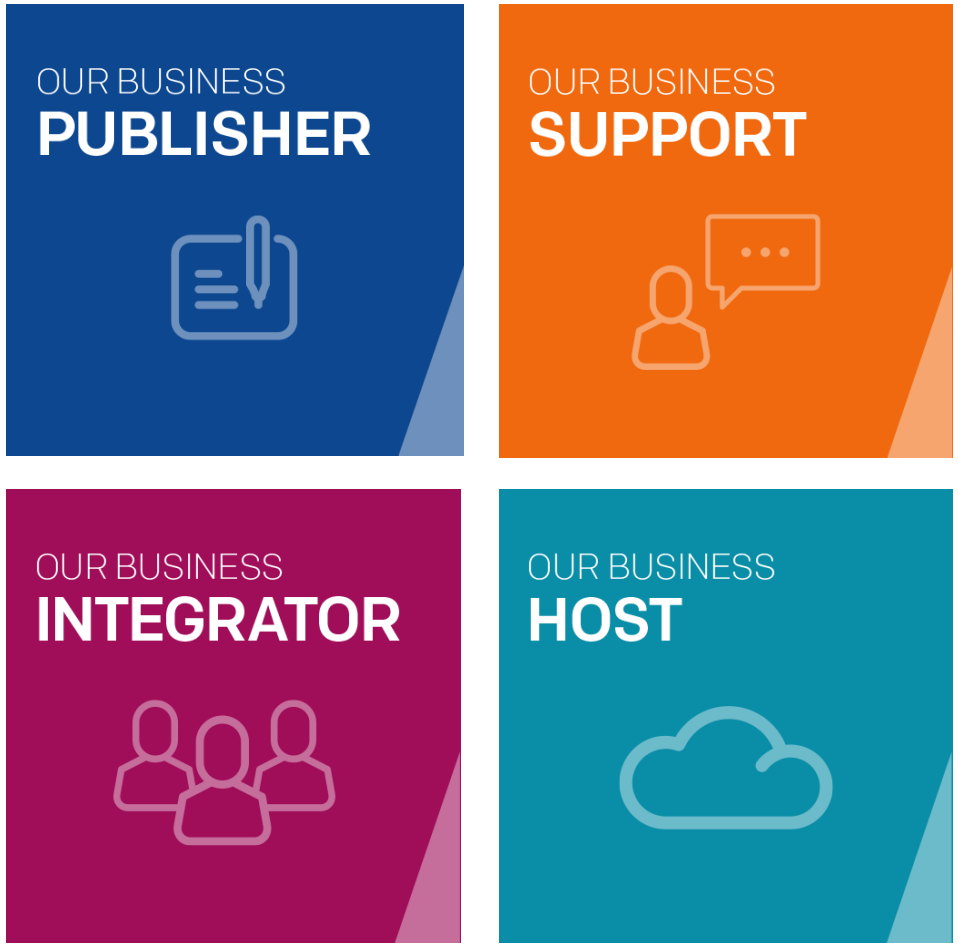
Memory Profiler

PUG Challenge 2024 –09/18/2024

PROGINOV

// Our business



A single point of contact for a responsible service

// Effectif

344

people (december 2024)

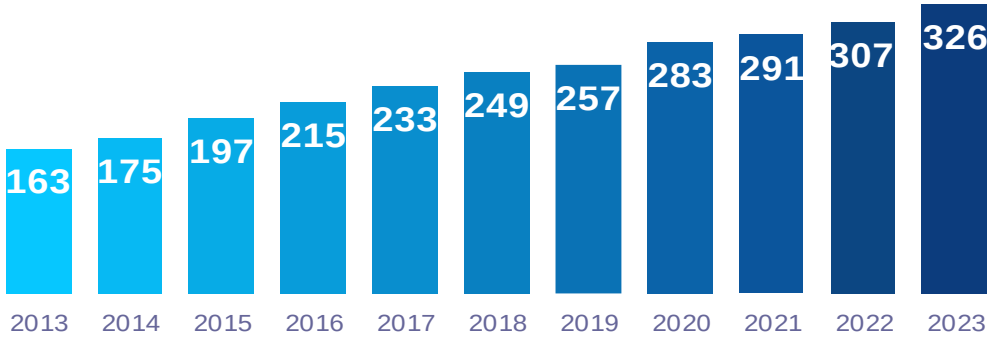
36

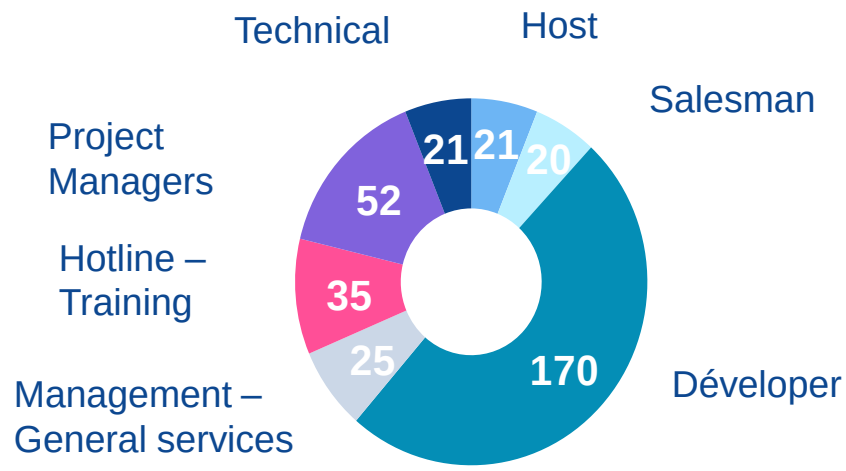
average age

≃ 2 %

people turnover

Workforce





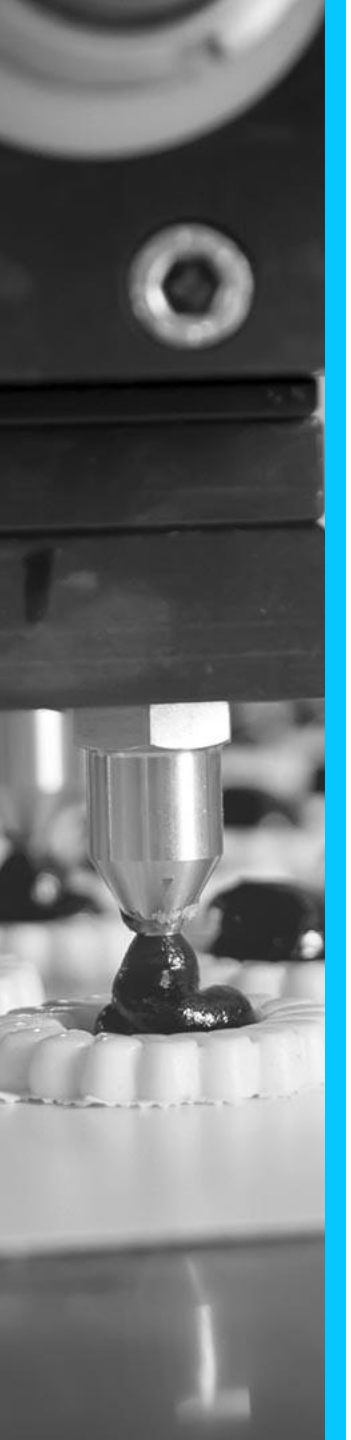
// Proginov Datacenters

/ Proginov 1st french hosting publisher

- 2 Data Centers
- More than 3 500 servers
- 5 power generators
- Fire protection system
- Secure access with badge
- 3 telecom providers

/ Monthly connexions

- ~13 000 concurrent uses
- ~31 000 connections per month



// Our ERP software

PROGINOV // ERP Universe

/ Business solutions

- Commerce
- Industry
- Distribution
- Retail
- E-commerce
- Job-based Subcontracting
- Agro-food
- Clothing and Textile
- Construction and Civil Engineering
- Garages et Workshops
- Packaging
- Timber Tarde
- Industrial Joinery
- Horticulture Industry
- Health

// Who are we ?

Adrien FRAISSE
afraisse@proginov.com

- / Progress developer and work for Proginov since 2012
- / In DevOps team since 2018

Patrice PERROT
pperrot@proginov.com

- / Progress developer since 2001
- / Optimize Progress sources since 2005
- / Work for Proginov since 2008
- / In team of Proginov DBA since 2010

PROGINOV

// Summary

PROGINOV

What is a memory leak ?

// What is a memory leak ?

/ Increasing and uncontrolled memory occupancy caused by not freeing unused space

- Forget to delete objects or handles
- Cyclic references between objects
- Too many creation of objects or handles in loops

/ Example

```
METHOD PUBLIC CHARACTER GetCustomerLabel(INPUT pCustomerId AS INTEGER):
    DEFINE VARIABLE bufTable AS HANDLE NO-UNDO.
    DEFINE VARIABLE vLabel   AS CHARACTER NO-UNDO.

    CREATE BUFFER bufTable FOR TABLE "customer".
    bufTable:FIND-FIRST("WHERE customer.id = " + STRING(pCustomerId), NO-LOCK).
    vLabel = STRING(bufTable::label).

    DELETE OBJECT bufTable.

    RETURN vLabel.

    CATCH Err AS Progress.Lang.Error:
        MESSAGE "Error".
    END CATCH.
END METHOD.
```

PROGINOV

// What is a memory leak ?

/ **32 bits sessions crash**

/ **Hard to analyze crashes (missing or inconclusive ProTrace)**

- StackTrace are different each time it crashes
- Random crash

/ **Analysis of the root cause of the crash made very difficult or even impossible**

/ **“Request for feedback” from Progress about Memory Leak tooling**

- Proginov answered and participated to the Beta test with its use cases

PROGINOV

How to detect memory leak currently ?

PROGINOV

// How to detect memory leak currently ?

/ Comparison of the list of objects in memory at 2 different times

/ Starting with

- SESSION:FIRST-OBJECT
- SESSION:FIRST-DATASET
- SESSION:FIRST-DATA-SOURCE
- SESSION:FIRST-QUERY
- SESSION:FIRST-BUFFER
- SESSION:FIRST-PROCEDURE

/ Continuing the process already started

- SESSION:NEXT-SIBLING

PROGINOV

// How to detect memory leak currently ?

/ LOG-MANAGER data are made intelligible

- LOG-MANAGER:LOGFILE-NAME = [LOGFILE_NAME].
- LOG-MANAGER:CLEAR-LOG( ).
- LOG-MANAGER:LOG-ENTRY-TYPES = "DynObjects.*:4" .

Instantanés | **Opération sur la mémoire** | Stack Trace

Afficher les types... | ☑ Afficher uniquement les objets oubliés | Trier par ordre d'occurence | Afficher

Heure	Identifiant de l'objet	Opération	Type de donnée	Type d'objet	Buffer/Table	Programme appelant	Procédure appelante	Ligne appelante
16:47:18.958	37567	Created	Progress.Lang.Object	Core.CoreStackTrace		Core.CoreStackTrace	GetCurrent	30
16:47:18.972	37733	Created	Progress.Lang.Object	Samples.Bob		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateCustomers	76

Programme	Procédure	Ligne
d:\proginov\prowin\protemp\p27361_Untit...	CreateCustomers	76
d:\proginov\prowin\protemp\p27361_Untit...		32

16:47:18.976	37734	Created	Progress.Lang.Object	Samples.John		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateCustomers	76
16:47:18.979	37735	Created	Progress.Lang.Object	Samples.Dolly		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateCustomers	76
16:47:18.979	37736	Created	Progress.Lang.Object	Core.CoreStackTrace		Core.CoreStackTrace	GetCurrent	30
16:47:18.997	37906	Created	Progress.Lang.Object	Samples.Order		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateOrderBob	58
16:47:18.999	37907	Created	Progress.Lang.Object	Samples.Product		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateOrderBob	59
16:47:19.001	37908	Created	Progress.Lang.Object	Samples.Order		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateOrderJohn	64
16:47:19.003	37909	Created	Progress.Lang.Object	Samples.Product		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateOrderJohn	65
16:47:19.004	37910	Created	Progress.Lang.Object	Samples.Order		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateOrderDolly	70
16:47:19.005	37911	Created	Progress.Lang.Object	Samples.Product		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateOrderDolly	71
16:47:19.008	37912	Created	Progress.Lang.Object	Samples.Product		d:\proginov\prowin\protemp\p27361_Untitled2.ped	CreateOrderDolly	72
16:47:19.008	37913	Created	Progress.Lang.Object	Core.CoreStackTrace		Core.CoreStackTrace	GetCurrent	30
16:47:19.018	38090	Created	Progress.Lang.Object	Core.CoreStackTrace		Core.CoreStackTrace	GetCurrent	30

PROGINOV

Design of the Memory Profiler by Progress

PROGINOV

// Design of the Memory Profiler by Progress

/ Has to offer an integrated tool allowing memory analysis of an AVM

- When and where is the memory allocated?
- How much memory is allocated?
- How to target memory consumption points?
- How much memory is released?

/ Tim Sargent's breakout session at PUG Challenge 2022

- https://www.youtube.com/watch?app=desktop&v=a-Rg27odVuE

/ Backend can be activated in OE12.8

/ Frontend is under development by Progress

- Proginov develops is own frontend

/ **PUG Session :** ABL Memory Profiler: are you leaking? Memory!!

- By Sunil Jardosh
- Friday – 11.45 am

PROGINOV

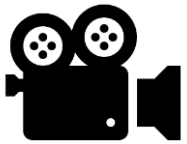
How it works

// How it works

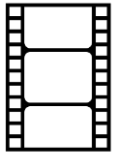
/ Activated for the whole session

/ Data written in .OEMP files (OpenEdge Memory Profiler)

/ Snapshot of memory at a given time



Data flow describing operations performed on memory



Series of snapshots taken at regular intervals

/ Ways to deal with

- Automatic mode with snapshot creation every X seconds
- Manual mode, with snapshots creation on demand
- Combining the 2 operating modes

PROGINOV

OEMP files

// OEMP files

/ File containing Profiling data generated during the session

/ File separated by snapshot by section

/ The file can be very large depending on session usage

/ small footprint on performance (less than 5%, ymmv)

PROGINOV

How to activate Memory Profiler?

// How to activate Memory Profiler?

/ Prowin.exe startup parameter

- -profileMemory *[PATH_CONF_FILE]*

[PATH_CONF_FILE] Specifies the path for Memory Profiler configuration file

/ Create a file at the location specified in [PATH_CONF_FILE] with the following content

report-dir *[OEMP_DIR]*
cadence *[NB_SECONDS]*

[OEMP_DIR] Specifies the OEMP files directory.
[NB_SECONDS] Indicates the time between each snapshot in seconds. (0 => manual mode)

/ Create a manual snapshot

Using Progress.Profiler.* .
MemoryProfiler:TakeSnapshot(*[LABEL]*).

PROGINOV

Use cases of Memory Profiler (Development)

PROGINOV

// Use cases of Memory Profiler (Development)

/ Analysis of a specific program

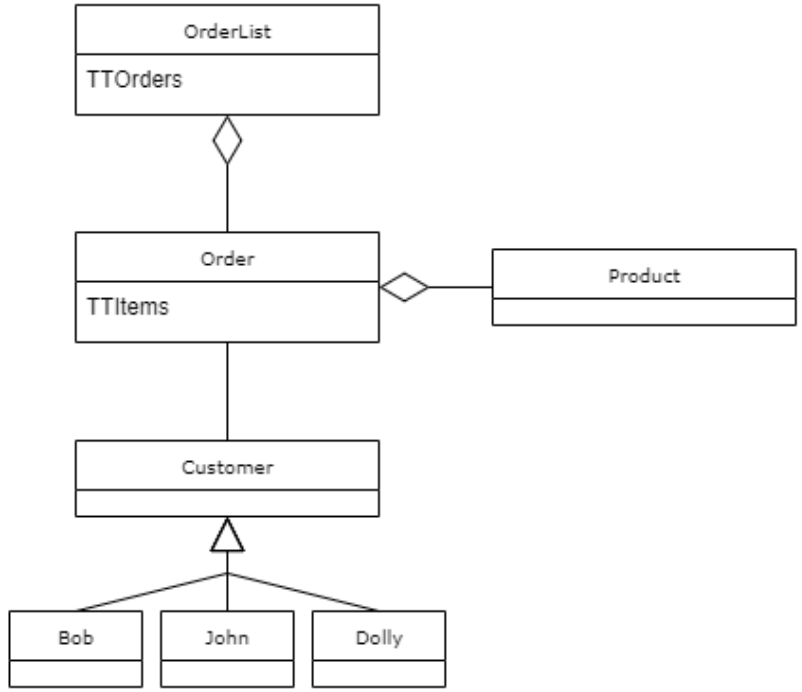
- Using manual mode
- Comparison of 2 snapshots or viewing a particular snapshot
- Checking the number of objects created and their weight
  - Allows you to limit the height of memory consumption peaks
- Check that any created object is destroyed
  - Helps avoid abnormally increasing process consumption over time

PROGINOV

// Use cases of Memory Profiler (Development)

/ Test case

- 3 customers (Bob, John and Dolly)
  - Classes Bob, John and Dolly which inherit from Customer
- a list of Orders
  - Class OrderList
  - Contains a Temp-Table TTOrders
- An order by customer (Temp-Table TTItems)
  - Class Order
  - Contains a Temp-Table TTItems of the items of the order



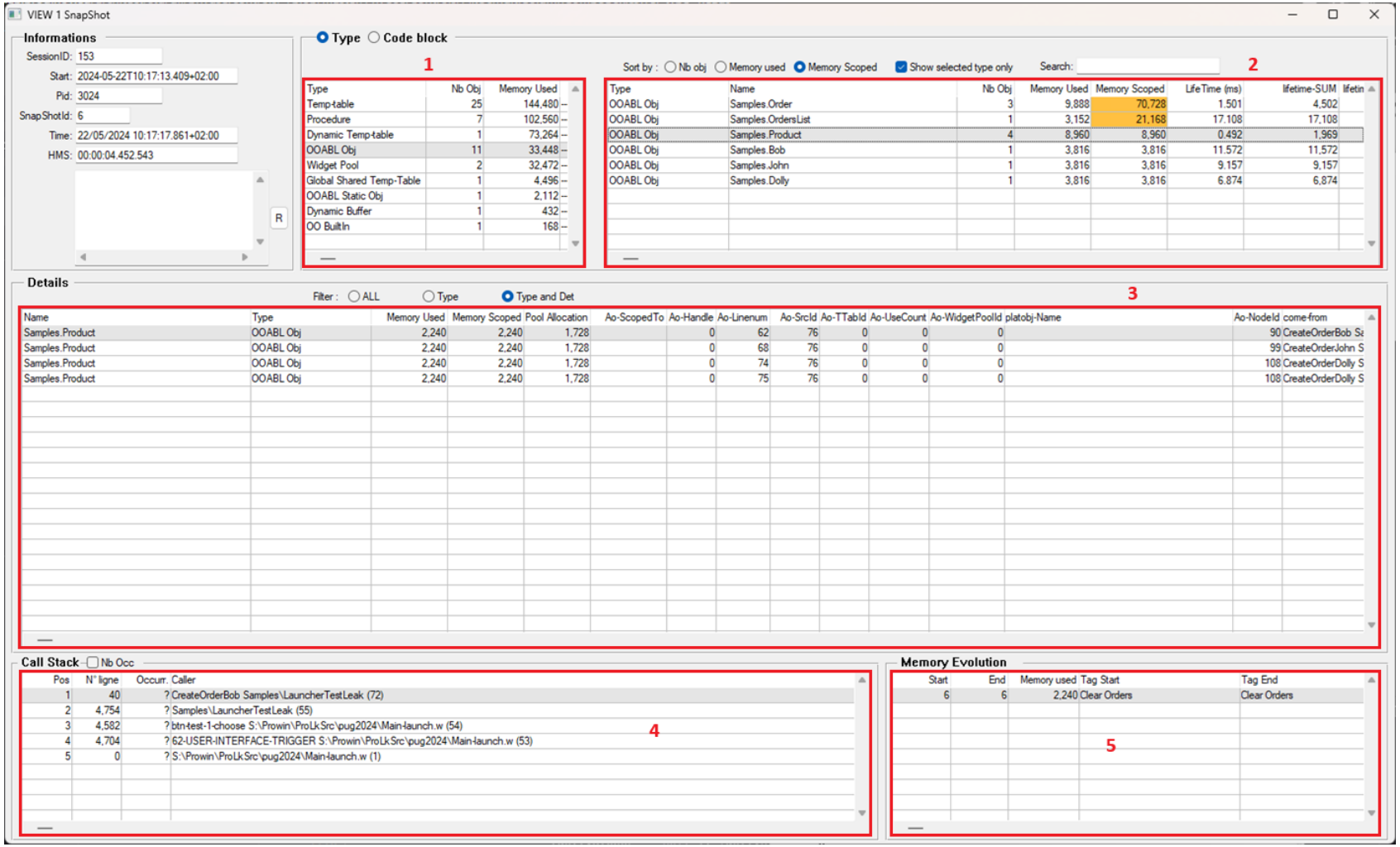
// Use cases of Memory Profiler (Development)

PROGINOV

// Use cases of Memory Profiler (Development)

DEMO

/ Analysis of a Snapshot (Presented by data type)



DEMO

// Use cases of Memory Profiler (Development)

/ Analysis of a Snapshot

- 1 : Objects and handles families (TT, Procedure, ABLObj…)
- 2 : Detected datatypes (Sorted by selected family)
- 3 : Occurrence details (Sorted by selected family or selected datatype)
- 4 : Stacktrace creation of the selected occurrence
- 5 : Memory evolution by Snapshot.

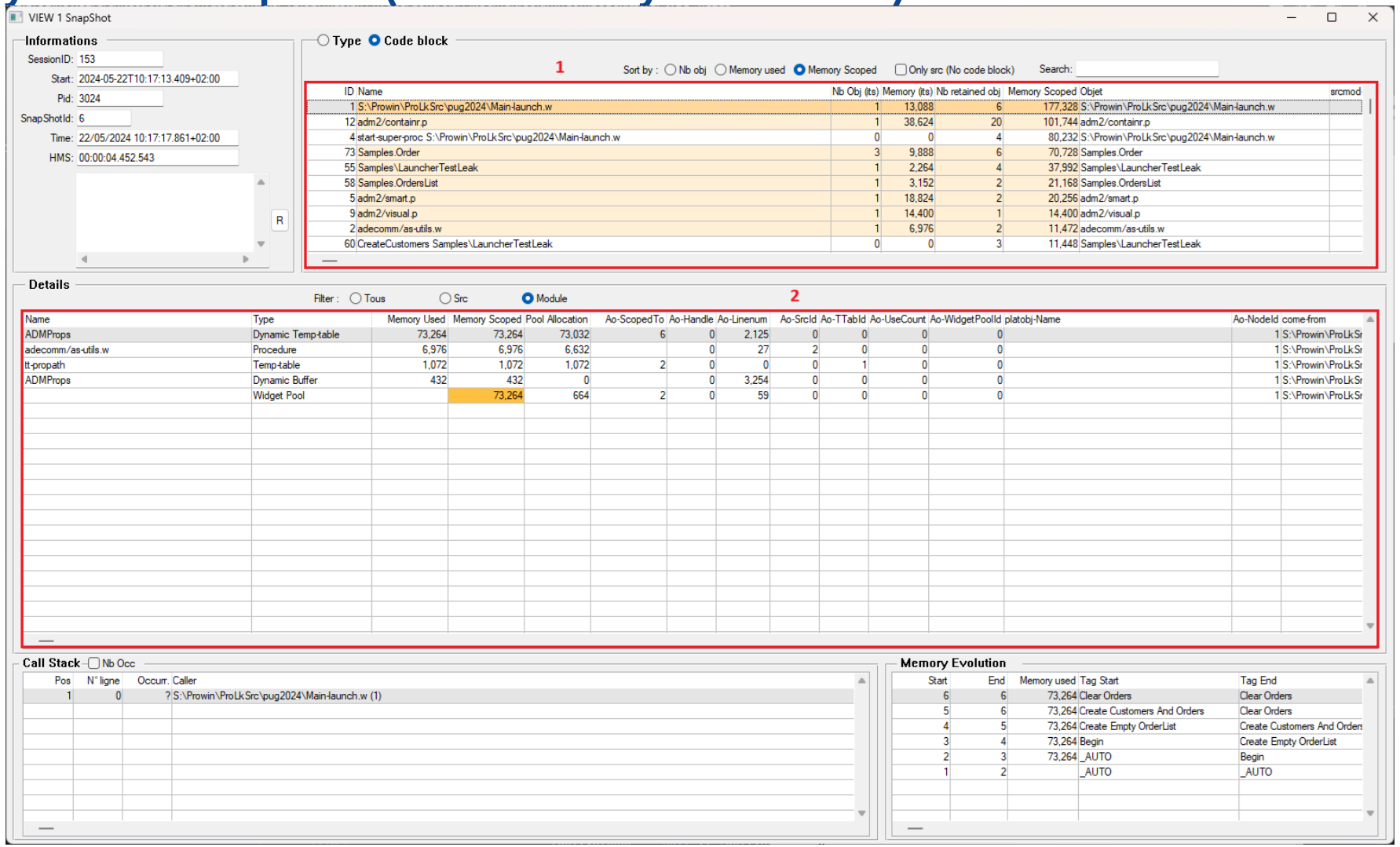
/ Scoped Memory : Memory created per handle

PROGINOV

// Use cases of Memory Profiler (Development)

DEMO

/ Analysis of a Snapshot (Presented by code block)



DEMO

// Use cases of Memory Profiler (Development)

/ Analysis of a Snapshot

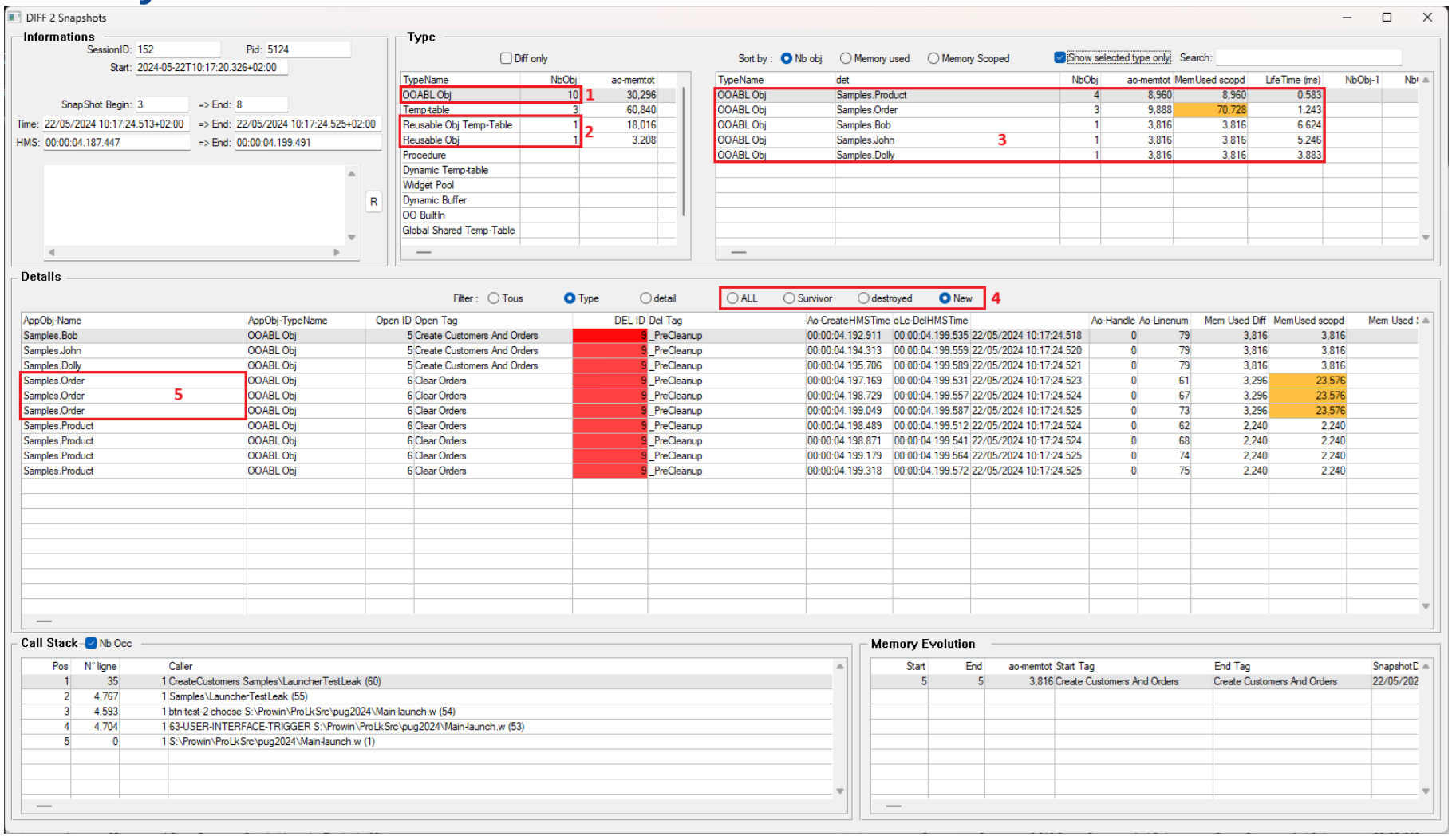
- 1 : Code block list (RCode, procedures, methods)
- 2 : Handles and Objects created in the selected block

PROGINOV

// Use cases of Memory Profiler (Development)

DEMO

/ Leak analysis

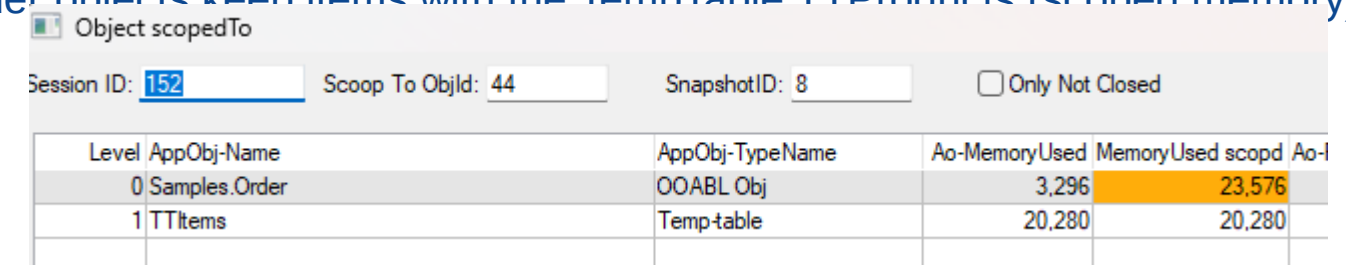


DEMO

// Use cases of Memory Profiler (Development)

/ Leak analysis

- 1 : We can see 10 objects stand in memory although orders are no longer here
- 2 : Reusable Obj : Deleted objects cached for a future use. Help session performance
- 3 : The 10 objects are orders, customers and products
- 4 : For a more specific analysis, we decide to show only the objects created and not deleted between the 2 snapshots
- 5 : Order objects keep items with the TempTable TTProducts (scoped memory)



PROGINOV

Use cases of Memory Profiler (Production)

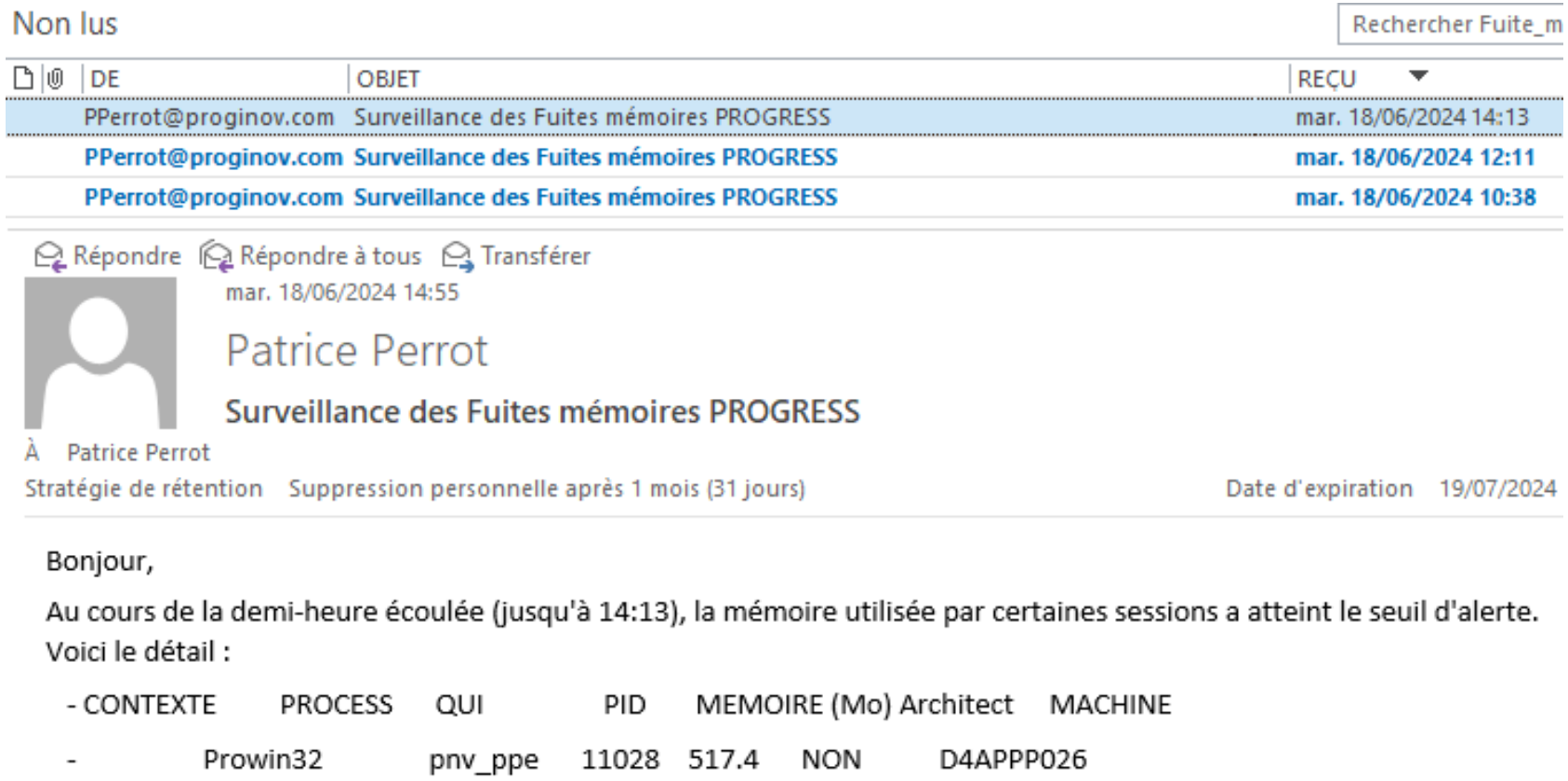
PROGINOV

// Use cases of Memory Profiler (Production)

/ Watching amount of memory usage through time

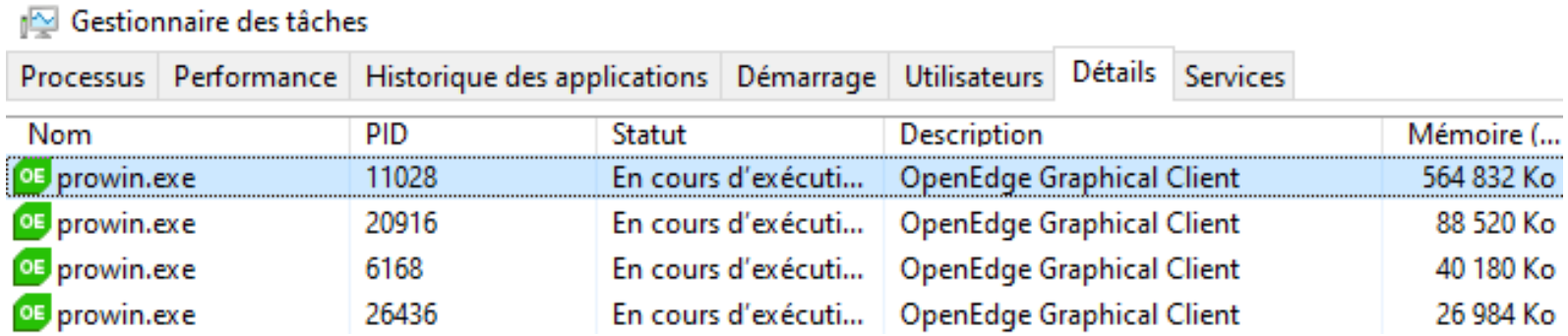
- Automated mode

/ Alerting

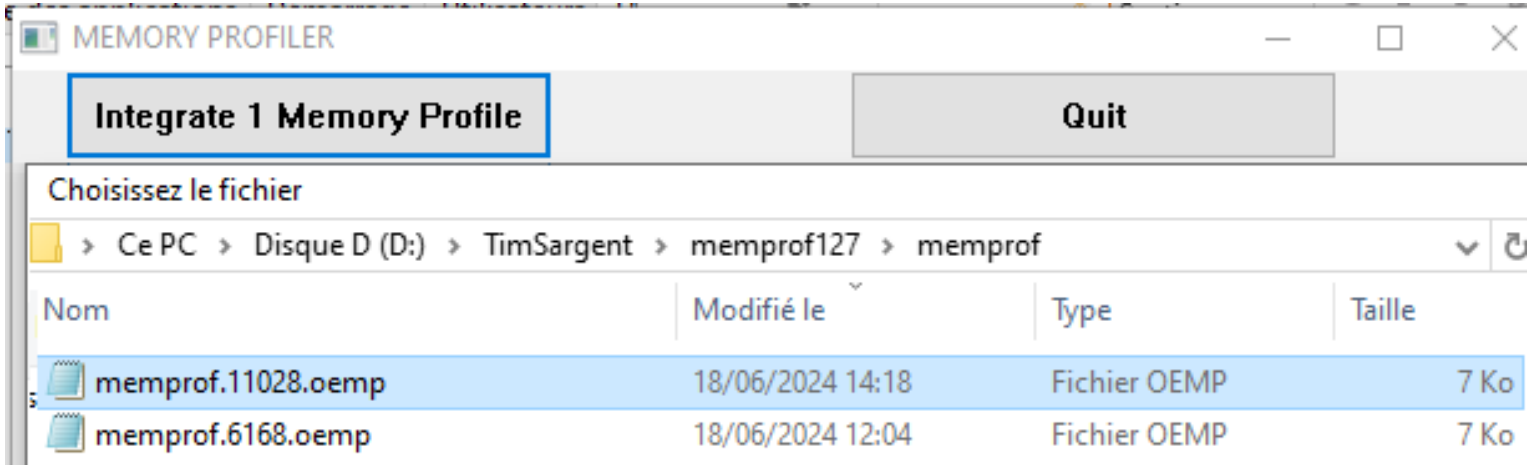


// Use cases of Memory Profiler (Production)

/ Overconsumption of a session

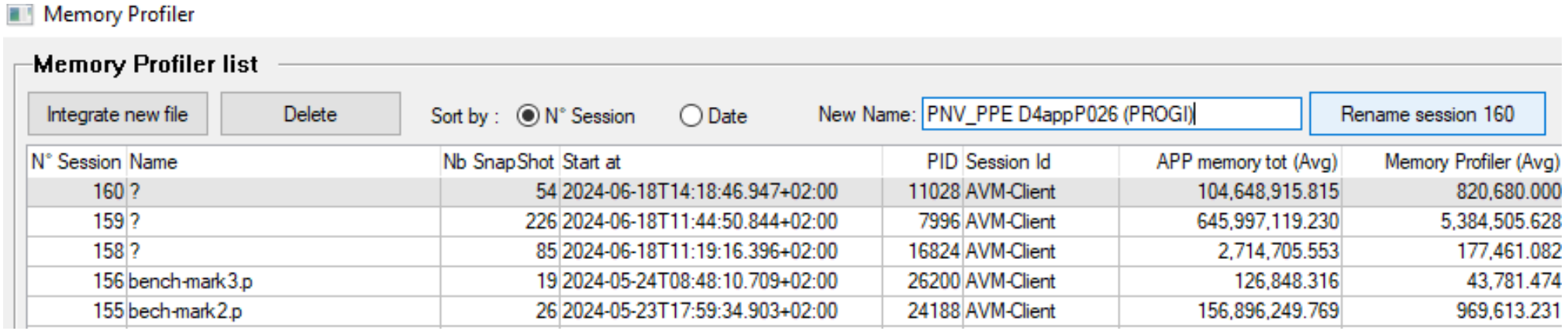


/ Import of its OEMP file



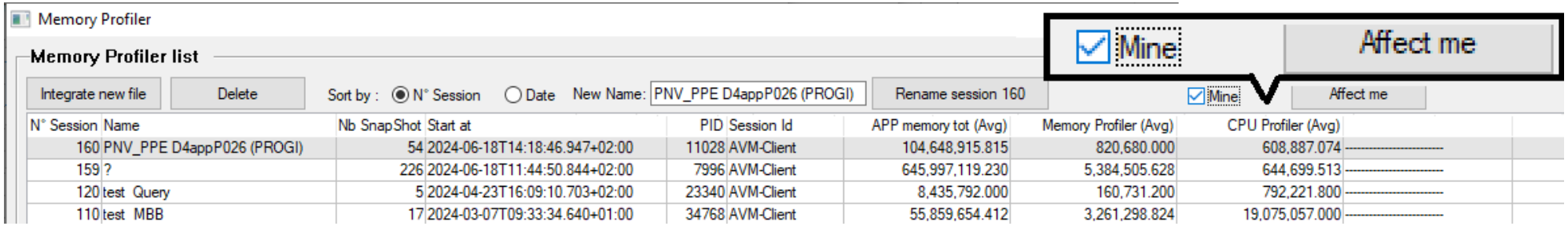
// Use cases of Memory Profiler (Production)

/ Naming the session



N° Session	Name	Nb SnapShot	Start at	PID	Session Id	APP memory tot (Avg)	Memory Profiler (Avg)
160	?	54	2024-06-18T14:18:46.947+02:00	11028	AVM-Client	104,648,915.815	820,680.000
159	?	226	2024-06-18T11:44:50.844+02:00	7996	AVM-Client	645,997,119.230	5,384,505.628
158	?	85	2024-06-18T11:19:16.396+02:00	16824	AVM-Client	2,714,705.553	177,461.082
156	bench-mark3.p	19	2024-05-24T08:48:10.709+02:00	26200	AVM-Client	126,848.316	43,781.474
155	bech-mark2.p	26	2024-05-23T17:59:34.903+02:00	24188	AVM-Client	156,896,249.769	969,613.231

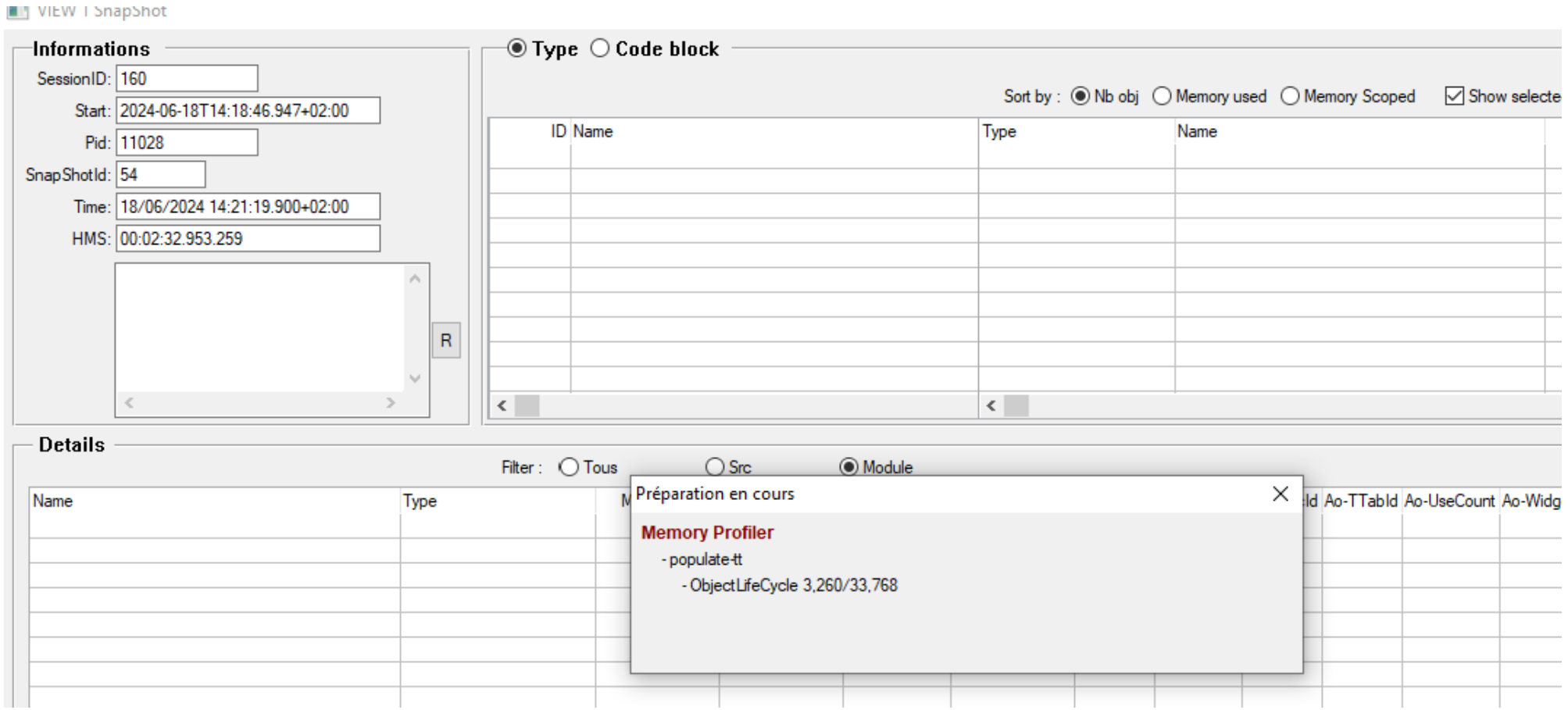
/ Assign yourself to the session



N° Session	Name	Nb SnapShot	Start at	PID	Session Id	APP memory tot (Avg)	Memory Profiler (Avg)	CPU Profiler (Avg)	
160	PNV_PPE D4appP026 (PROGI)	54	2024-06-18T14:18:46.947+02:00	11028	AVM-Client	104,648,915.815	820,680.000	608,887.074	----------
159	?	226	2024-06-18T11:44:50.844+02:00	7996	AVM-Client	645,997,119.230	5,384,505.628	644,699.513	----------
120	test Query	5	2024-04-23T16:09:10.703+02:00	23340	AVM-Client	8,435,792.000	160,731.200	792,221.800	----------
110	test MBB	17	2024-03-07T09:33:34.640+01:00	34768	AVM-Client	55,859,654.412	3,261,298.824	19,075,057.000	----------

// Use cases of Memory Profiler (Production)

/ Load could take time (+ / - 1 minute)



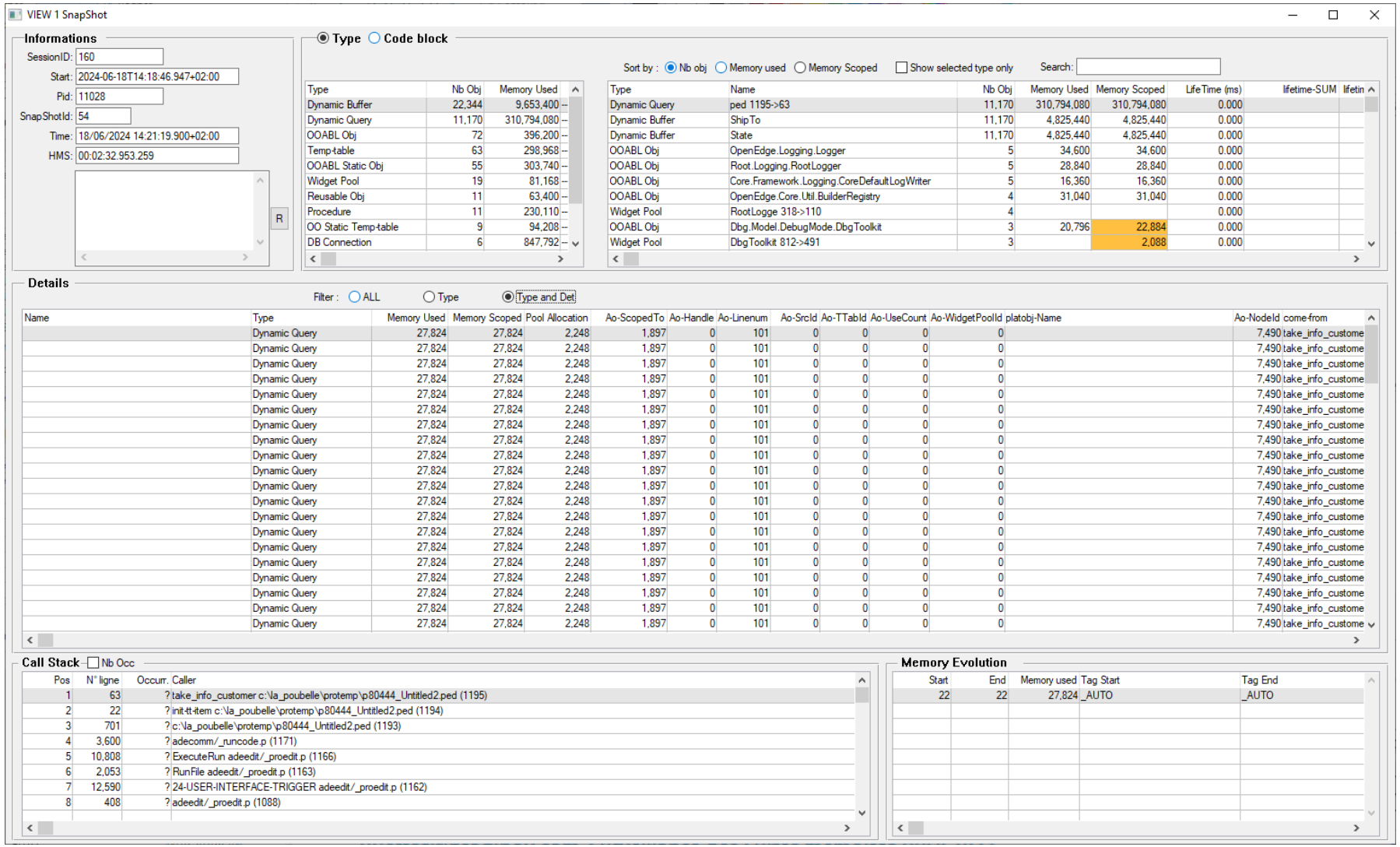
// Use cases of Memory Profiler (Development)

PROGINOV

DEMO

// Use cases of Memory Profiler (Production)

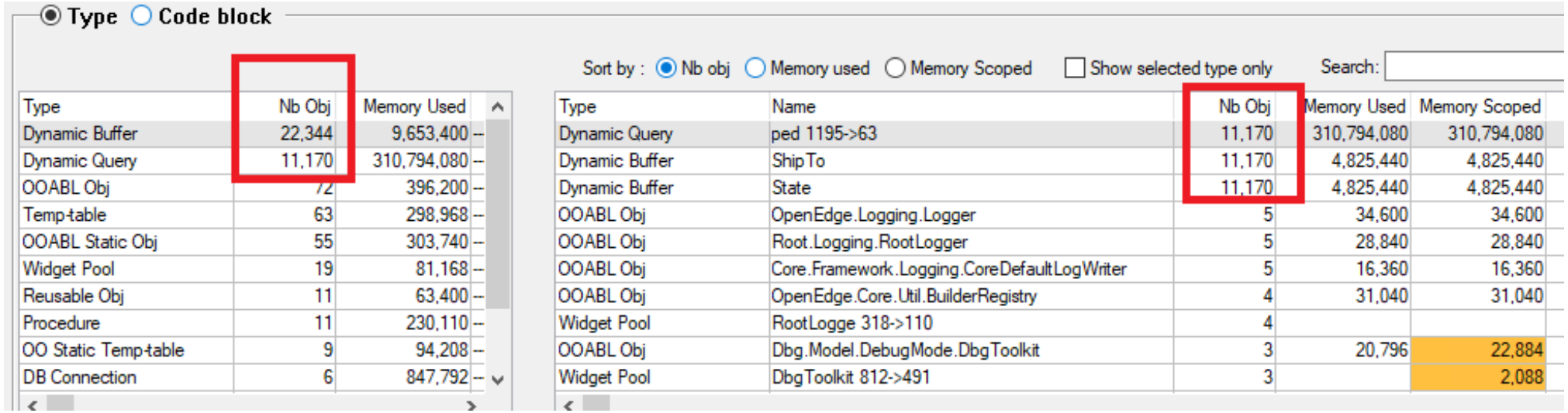
/ DEMO 1



DEMO

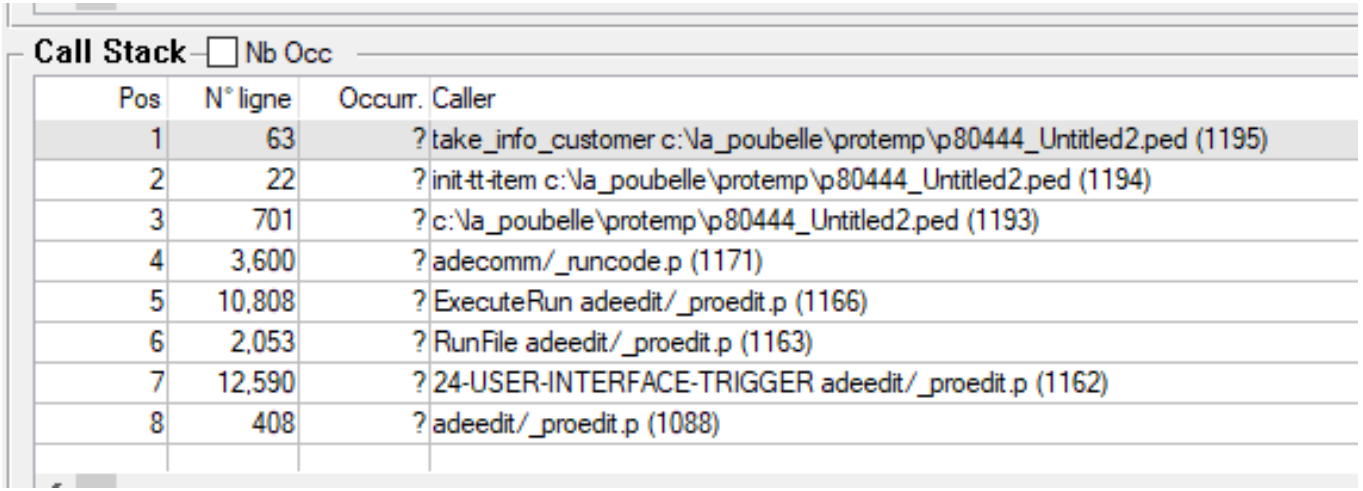
// Use cases of Memory Profiler (Production)

/ DEMO 2 : Focus on points of interest



Type	Nb Obj	Memory Used
Dynamic Buffer	22,344	9,653,400
Dynamic Query	11,170	310,794,080
OOABL Obj	72	396,200
Temp-table	63	298,968
OOABL Static Obj	55	303,740
Widget Pool	19	81,168
Reusable Obj	11	63,400
Procedure	11	230,110
OO Static Temp-table	9	94,208
DB Connection	6	847,792

Type	Name	Nb Obj	Memory Used	Memory Scoped
Dynamic Query	ped 1195->63	11,170	310,794,080	310,794,080
Dynamic Buffer	ShipTo	11,170	4,825,440	4,825,440
Dynamic Buffer	State	11,170	4,825,440	4,825,440
OOABL Obj	OpenEdge.Logging.Logger	5	34,600	34,600
OOABL Obj	Root.Logging.RootLogger	5	28,840	28,840
OOABL Obj	Core.Framework.Logging.CoreDefaultLogWriter	5	16,360	16,360
OOABL Obj	OpenEdge.Core.Util.BuilderRegistry	4	31,040	31,040
Widget Pool	RootLogge 318->110	4		
OOABL Obj	Dbg.Model.DebugMode.DbgToolkit	3	20,796	22,884
Widget Pool	DbgToolkit 812->491	3		2,088



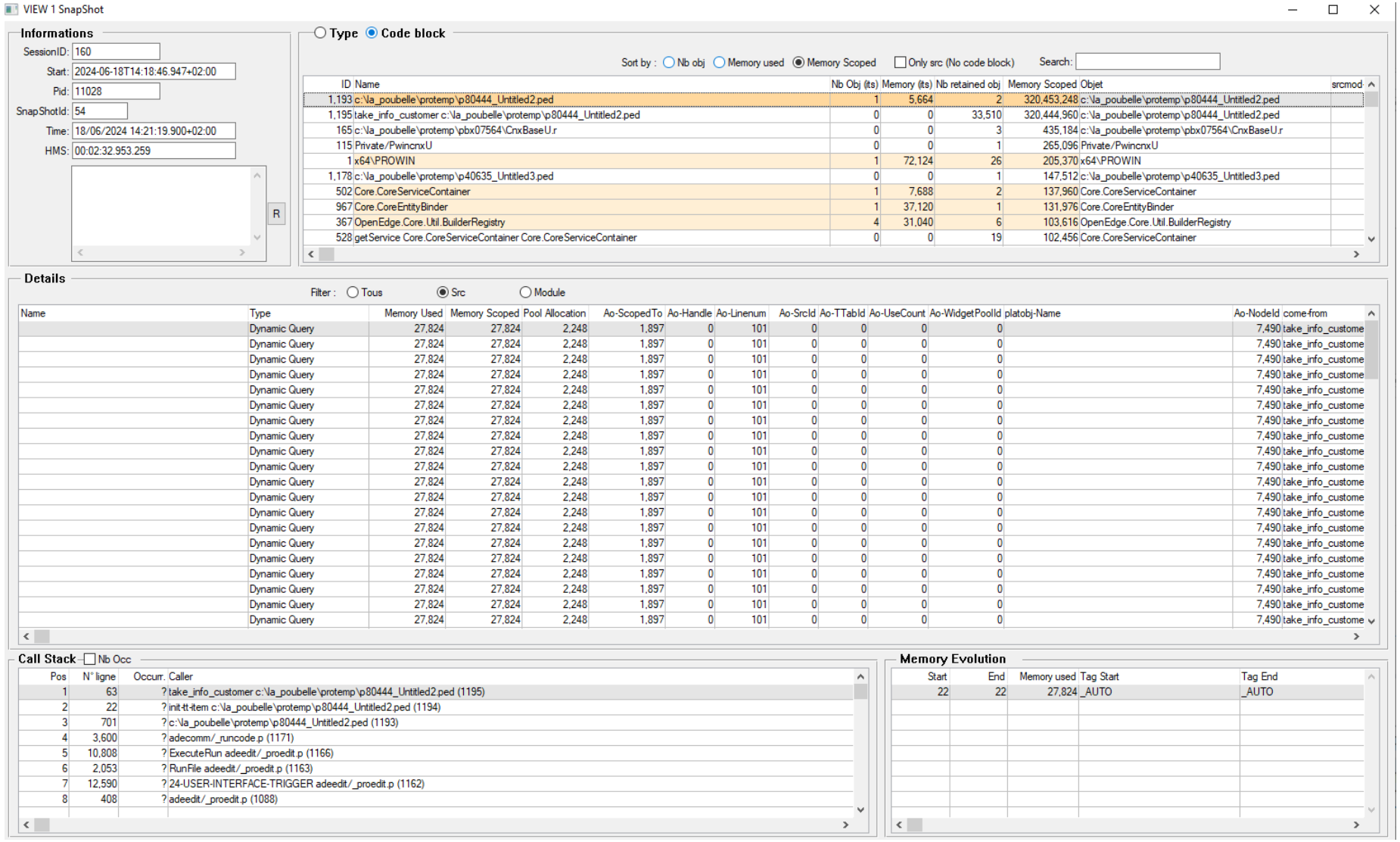
Pos	N° ligne	Occurr.	Caller
1	63	?	take_info_customer c:\la_poubelle\protemp\p80444_Untitled2.ped (1195)
2	22	?	init-tt-item c:\la_poubelle\protemp\p80444_Untitled2.ped (1194)
3	701	?	c:\la_poubelle\protemp\p80444_Untitled2.ped (1193)
4	3,600	?	adecomm/_runcode.p (1171)
5	10,808	?	ExecuteRun adeedit/_proedit.p (1166)
6	2,053	?	RunFile adeedit/_proedit.p (1163)
7	12,590	?	24-USER-INTERFACE-TRIGGER adeedit/_proedit.p (1162)
8	408	?	adeedit/_proedit.p (1088)

PROGINOV

DEMO

// Use cases of Memory Profiler (Production)

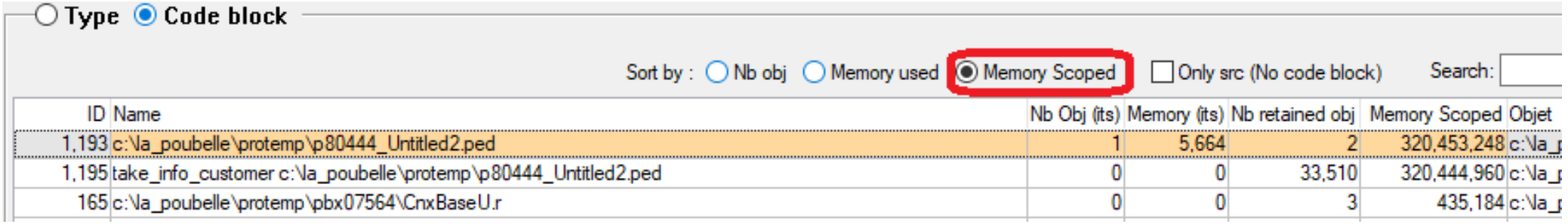
/ DEMO 3 : Presented by code block



DEMO

// Use cases of Memory Profiler (Production)

/ DEMO 3 : Presented by code block => Zoom



ID	Name	Nb Obj (its)	Memory (its)	Nb retained obj	Memory Scoped	Objet
1,193	c:\la_poubelle\protemp\p80444_Untitled2.ped	1	5,664	2	320,453,248	c:\la_
1,195	take_info_customer c:\la_poubelle\protemp\p80444_Untitled2.ped	0	0	33,510	320,444,960	c:\la_
165	c:\la_poubelle\protemp\pbx07564\CnxBaseU.r	0	0	3	435,184	c:\la_

PROGINOV

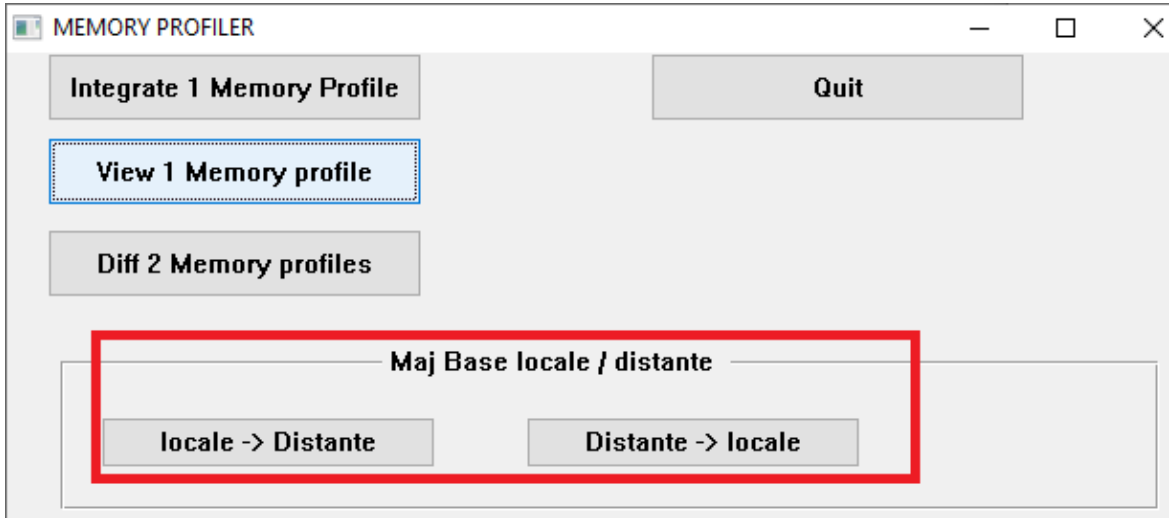
Other use cases

// Memory Profiler : Significant DB size

/ Local DB for personal analysis

/ Remote DB for shared analysis

/ Possibility to transfer between local and remote for one session



→ Better performance on local db connected with SELF

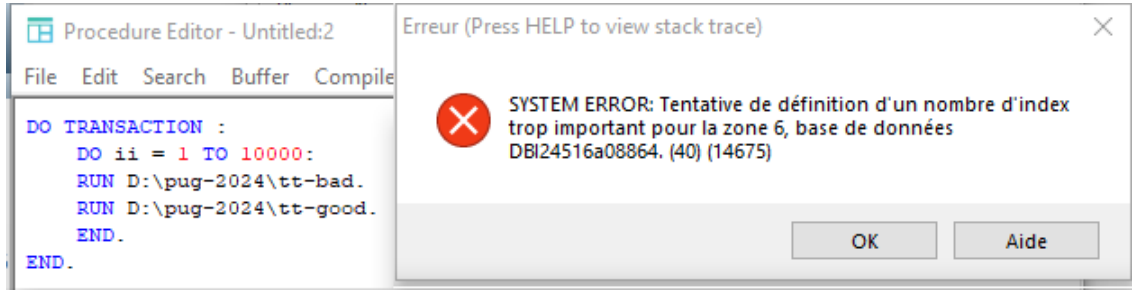
// Use cases of Memory Profiler (Development)

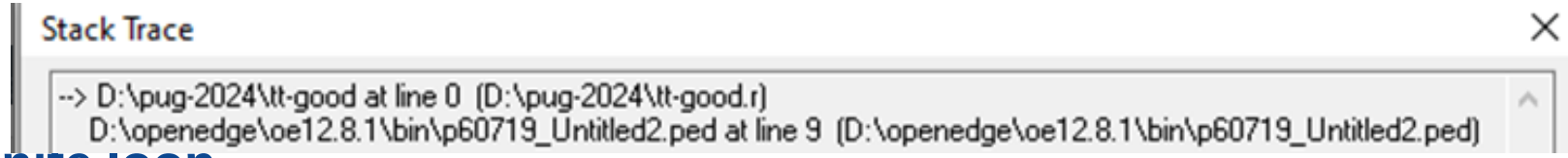
PROGINOV

DEMO

// Memory Profiler : Out of its scope

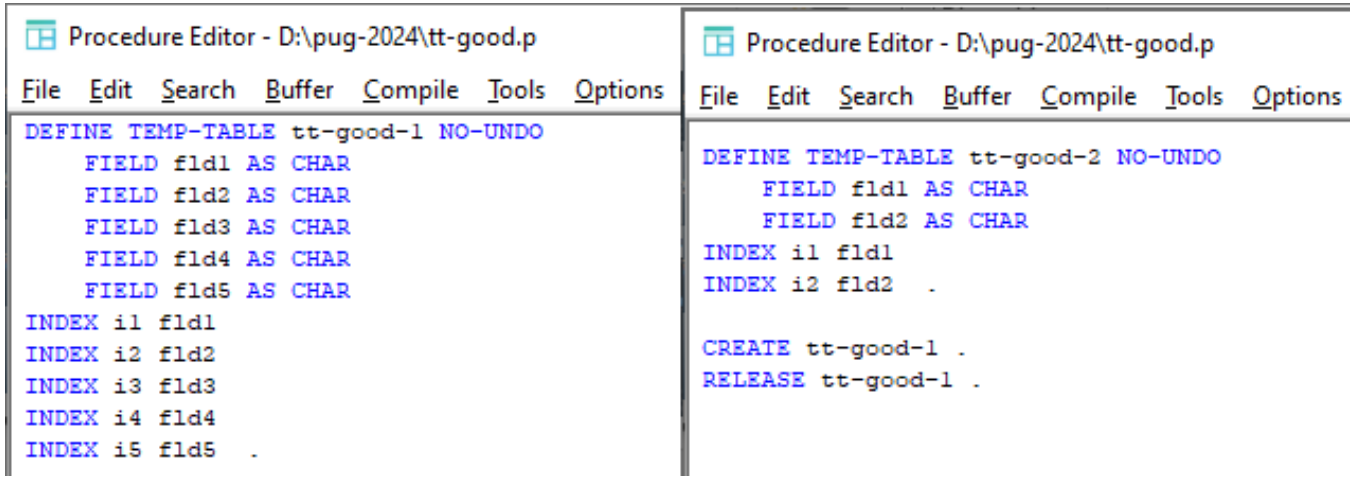
/ Error 14675





/ No infinite loop,

/ No issue



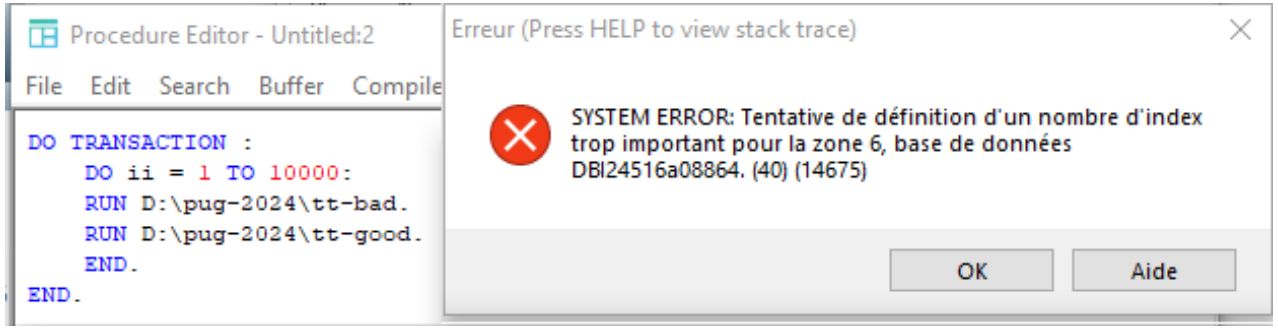
(7 index)

PROGINOV

DEMO

// Memory Profiler : Out of its scope

/ Error 14675



protrace.28320.txt - Bloc-notes

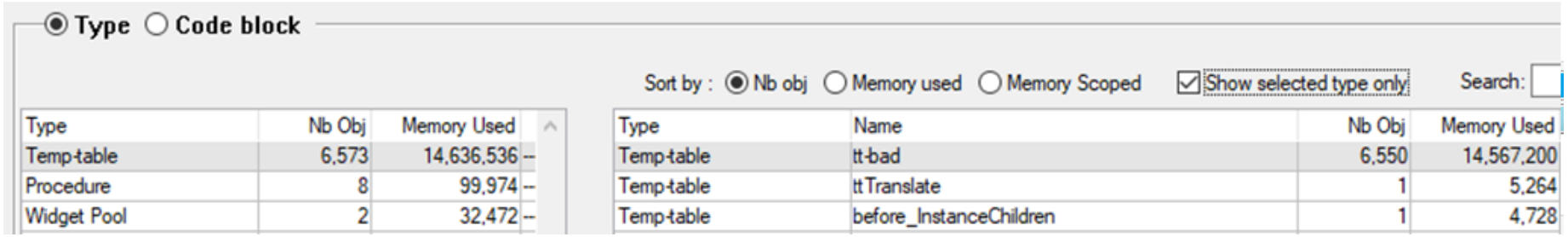
```
** ABL Stack Trace **
--> D:\pug-2024\tt-good at line 0  (D:\pug-2024\tt-good.r)
    D:\openedge\oe12.8.1\bin\p60719_Untitled2.ped at line 9  (D:\openedge\oe12.8.1\bin\p60719_Untitled2.ped)

** Persistent procedures/Classes **
Handle  File Name
001015  D:\openedge\oe12.8.1\gui\adm2\containr.r
001013  D:\openedge\oe12.8.1\gui\adm2\visual.r
001010  D:\openedge\oe12.8.1\gui\adm2\appserver.r
001005  D:\openedge\oe12.8.1\gui\adm2\smart.r
001001  adecomm/as-utils.r
```

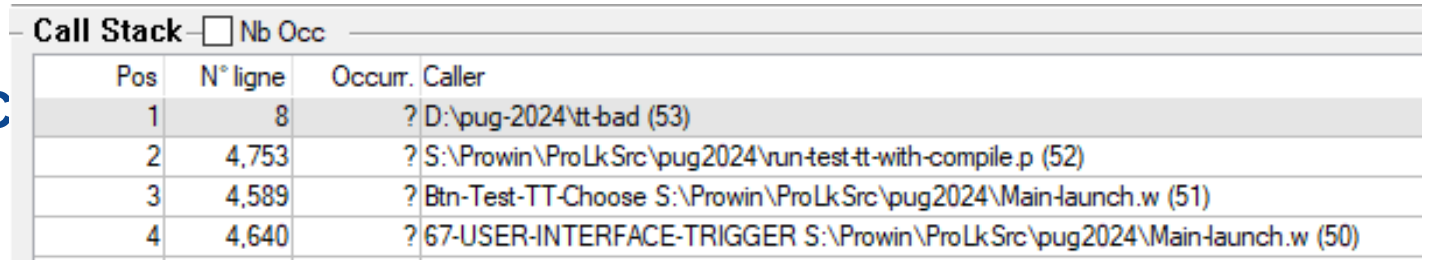
PROGINOV

DEMO

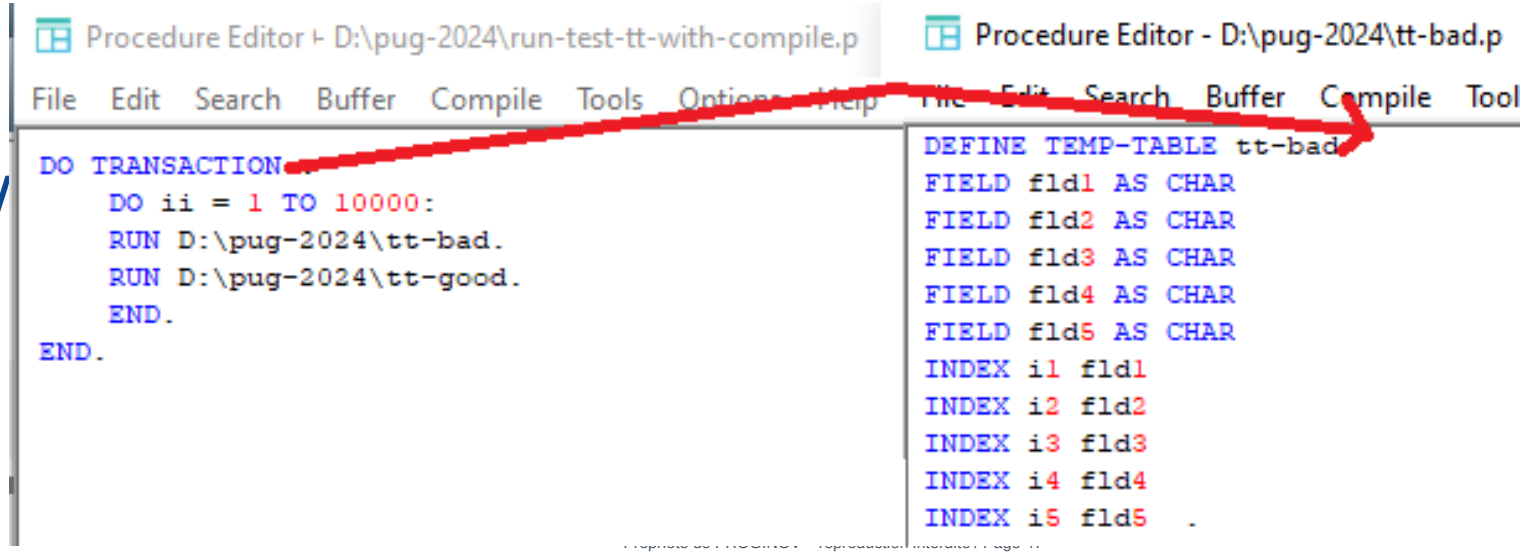
// Memory Profiler : Out of its scope



/ C



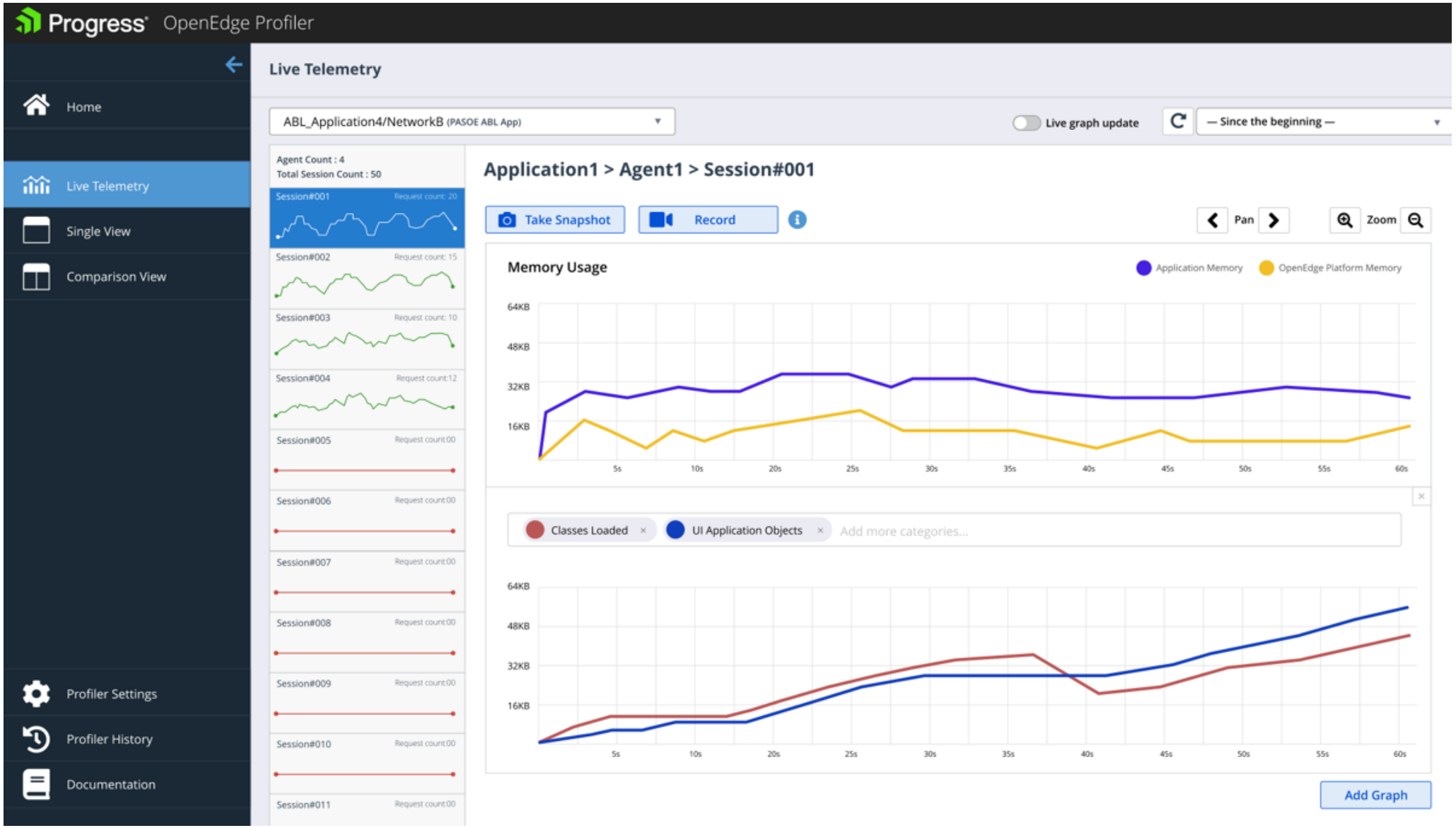
/ M



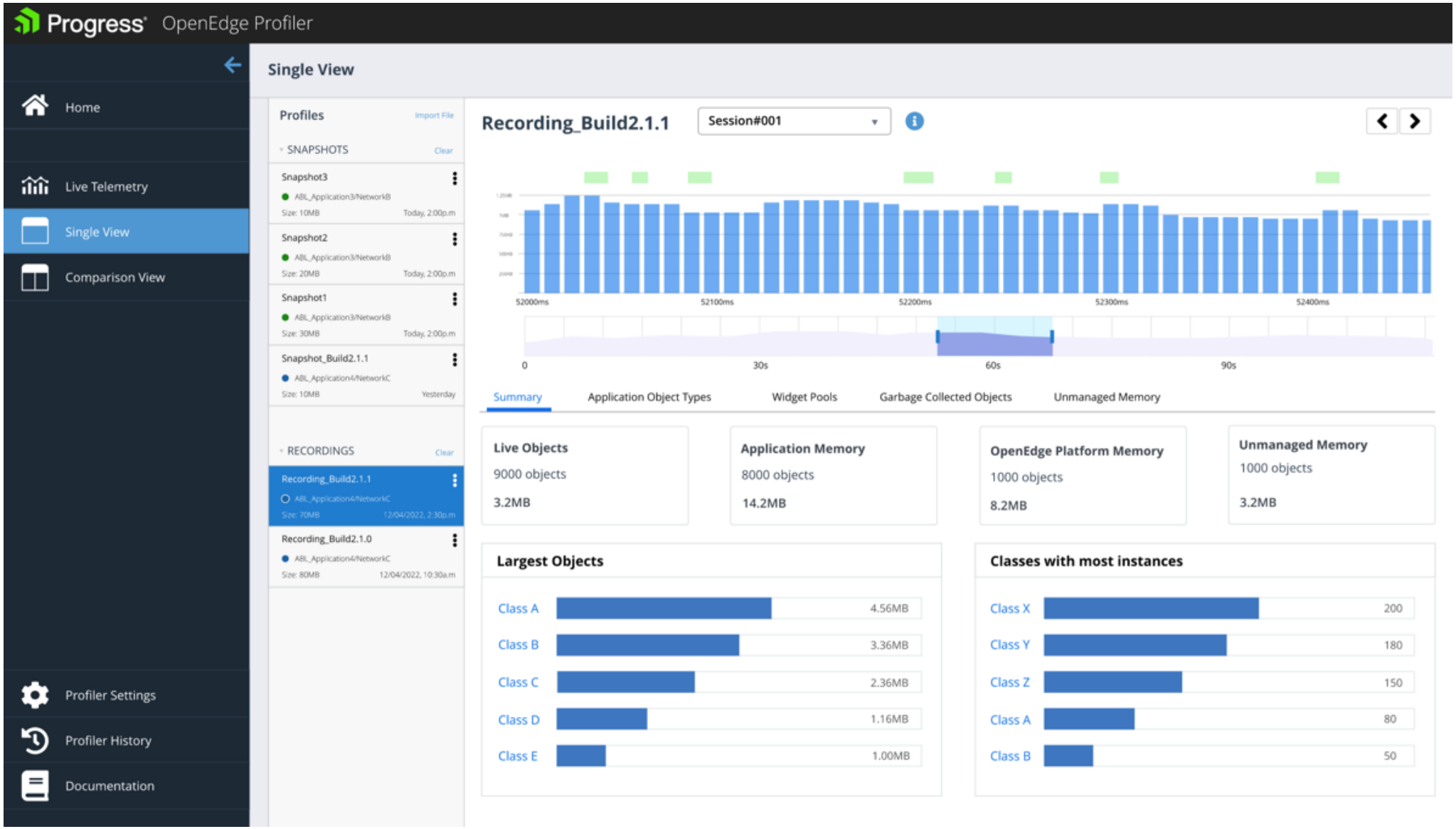
PROGINOV

What Progress already presented

// What Progress already presented



// What Progress already presented



Improvement perspectives

// Ours

/ New UI

/ Develop new trend metrics based on several snapshots

/ Continuous integration of OEMP files

/ Formation of our development teams to Memory Profiler

/ Include memory profiler analysis in our quality process

PROGINOV

// Those asked to Progress

/ Link memory consumption to object and not only to treenodes

- Could give a aggregation of the memory kept by every object

/ Enable exclusion of Progress system objects and handles

- Could hinder an analysis

/ Visualize object references

- Better understanding of the unefficiency of the garbage collector
- Allow the selection of orphan objects

/ During a differential analysis, propose subsets of objects (created, deleted, remaining)

- A object created and not deleted between 2 snapshots could pinpoint a leak

PROGINOV

Questions ?

Agenda :

PUG Session : ABL Memory Profiler: are you leaking? Memory!! By Sunil Jardosh

Friday 11.45 am

PROGINOV

+33 (0)2 51 70 93 93
www.proginov.com