

# Memory Profiler

PUG France 2024 – 25/06/2024

## // Sommaire

- / **Qu'est-ce qu'une fuite mémoire ?**
- / **Comment détecter une fuite mémoire actuellement ?**
- / **Conception d'un Memory Profiler par Progress**
- / **Principe de fonctionnement**
- / **Fichier OEMP (OpenEdge Memory Profiler)**
- / **Comment activer le Memory Profiler ?**
- / **Cas d'utilisation du Memory Profiler**
- / **Ce que Progress a déjà présenté**
- / **Pistes d'amélioration**

**Qu'est-ce qu'une fuite mémoire ?**

# // Qu'est-ce qu'une fuite mémoire ?

## / Occupation croissante et non contrôlée de la mémoire causée par la non-libération d'espace inutilisé

- Oubli de suppression d'objets ou de handles
- Références cycliques entre des objets
- Création abusive d'objets ou de handles dans des boucles

## / Exemple

```
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.
```



## // Qu'est-ce qu'une fuite mémoire ? (Cas réel)

/ Cas de plantage session utilisateur en 32 bits

/ Difficulté d'analyse des plantages (ProTrace absent ou non concluant)

- StackTrace différente à chaque plantage
- Plantage aléatoire

/ Analyse rendue très difficile voire impossible de l'origine du plantage

/ Sollicitation de Progress à la communauté Progress au premier semestre 2022 sur les fuites mémoire

- Réponse de la part de Proginov pour participer aux Beta tests de Progress avec nos cas d'utilisation

# Comment détecter une fuite mémoire actuellement ?



## // Comment détecter une fuite mémoire actuellement ?

/ Comparaison de la liste des objets en mémoire à 2 moments différents

/ Pour commencer un parcours

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

/ Pour continuer le parcours déjà commencé

- SESSION:NEXT-SIBLING

# // Comment détecter une fuite mémoire actuellement ?

## / Interprétation des données du LOG-MANAGER

- 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... <input checked="" type="checkbox"/> Afficher uniquement les objets oubliés Trier par ordre d'occurrence <span style="float:right">Afficher</span>									
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	

# Conception d'un Memory Profiler par Progress



## // Conception d'un Memory Profiler par Progress

### / Nécessité de proposer un outil intégré permettant de faire de l'analyse mémoire d'une AVM

- Quand et où la mémoire est allouée ?
- Quelle quantité de mémoire est allouée ?
- Comment cibler les points de consommation mémoire ?
- Quand et où est libérée la mémoire ?

### / Présentation du projet par Tim Sargent au PUG Challenge 2022

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

### / Backend activable dans la version 12.8

### / FrontEnd en cours de développement par Progress

- Développement d'un Front interne à Proginov

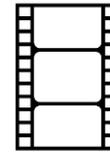
# Principe de fonctionnement

## // Principe de fonctionnement

- / Activation pour toute la durée d'une session
- / Ecriture des données de profiling dans un fichier .OEMP
- / Un snapshot est un cliché de la mémoire à un instant T



Flux de donnée décrivant les opérations effectuées sur la mémoire



Suite de clichés pris à intervalles réguliers

### / 2 modes de fonctionnement

- Mode automatique avec création d'un SnapShot toutes les X secondes
- Mode manuel, avec création des SnapShots à la demande
- Possibilité de cumuler les 2 modes de fonctionnement

# Fichier OEMP (OpenEdge Memory Profiler)



## // Fichier OEMP (OpenEdge Memory Profiler)

- / Fichier contenant les données de Profiling généré au cours de la session
- / Découpage par SnapShots / Sections
- / Le fichier peut être très volumineux en fonction de l'utilisation de la session
- / Incidence indolore (inférieure à 5%) sur les performances

# Comment activer le Memory Profiler ?

## // Comment activer le Memory Profiler ?

### / Option de démarrage du Prowin.exe

- -profileMemory *[PATH\_CONF\_FILE]*

**[PATH\_CONF\_FILE]** Indique le chemin du fichier de configuration du Memory Profiler

### / Créer un fichier à l'endroit spécifié dans **[PATH\_CONF\_FILE]** avec le contenu suivant

report-dir *[OEMP\_DIR]*

cadence *[NB\_SECONDS]*

**[OEMP\_DIR]** Indique le répertoire de génération des fichiers OEMP.

**[NB\_SECONDS]** Indique le temps entre chaque snapshot en secondes. S'il est à 0, le mode automatique est désactivé.

### / Créer un snapshot manuel

Using Progress.Profiler.\* .

MemoryProfiler:TakeSnapshot(*[LABEL]*).

# Cas d'utilisation du Memory Profiler (Dev)

# // Cas d'utilisation du Memory Profiler

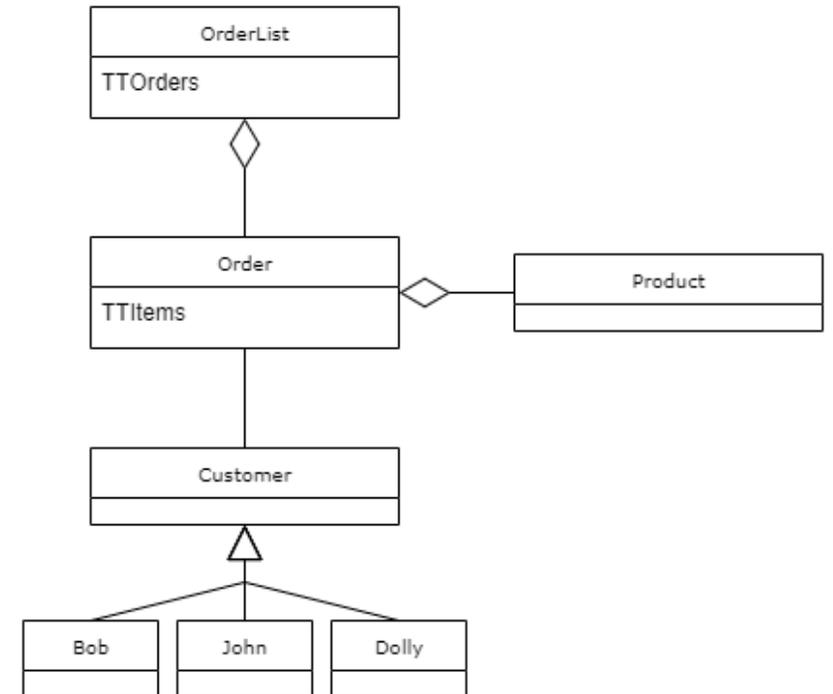
## / Analyse d'un programme précis

- Utilisation du mode manuel
- Comparaison de 2 snapshots ou visualisation d'un Snapshot en particulier
- Vérification du nombre d'objets créés et de leur poids
  - Permet de limiter la hauteur des pics de consommation mémoire
- Vérifier que tout objet créé est bien détruit
  - Permet d'éviter une consommation croissante anormale du processus au fil du temps

# // Cas d'utilisation du Memory Profiler

## / Cas de test

- 3 Clients (Bob, John Dolly)
  - Classes Bob, John et Dolly qui héritent de Customer
- Une liste de commandes
  - Classe OrderList
  - Contient une Temp-Table TTOOrders)
- Une commande par client (Temp-Table TTItems)
  - Classe Order
  - Contient une Temp-Table TTItems des produits de la commande



# // Cas d'utilisation du Memory Profiler

**DEMO**



# // Cas d'utilisation du Memory Profiler

## / Analyse d'un Snapshot (Vue Type)

The screenshot displays the 'VIEW 1 SnapShot' window of a memory profiler. It is divided into several sections:

- Informations:** SessionID: 153, Start: 2024-05-22T10:17:13.409+02:00, Pid: 3024, SnapShotId: 6, Time: 22/05/2024 10:17:17.861+02:00, HMS: 00.00.04.452.543.
- Type / Code block:** Radio buttons for 'Type' (selected) and 'Code block'. A search bar is present.
- Summary Tables:**
  - Table 1:** Aggregates memory usage by type. 

Type	Nb Obj	Memory Used
Temp-table	25	144.480
Procedure	7	102.560
Dynamic Temp-table	1	73.264
OOABL Obj	11	33.448
Widget Pool	2	32.472
Global Shared Temp-Table	1	4.496
OOABL Static Obj	1	2.112
Dynamic Buffer	1	432
OO BuiltIn	1	168
  - Table 2:** Sorts by 'Memory Scoped' (selected). 

Type	Name	Nb Obj	Memory Used	Memory Scoped	LifeTime (ms)	lifetime-SUM	lifen
OOABL Obj	Samples Order	3	9.888	70.728	1.501	4.502	
OOABL Obj	Samples OrdersList	1	3.152	21.168	17.108	17.108	
OOABL Obj	Samples.Product	4	8.960	8.960	0.492	1.969	
OOABL Obj	Samples.Bob	1	3.816	3.816	11.572	11.572	
OOABL Obj	Samples.John	1	3.816	3.816	9.157	9.157	
OOABL Obj	Samples.Dolly	1	3.816	3.816	6.874	6.874	
- Details:** Filter: Type and Det. 

Name	Type	Memory Used	Memory Scoped	Pool Allocation	Ao-ScopedTo	Ao-Handle	Ao-Linenum	Ao-ScrdId	Ao-TTTabId	Ao-UseCount	Ao-WidgetPoolId	platobj-Name	Ao-NodeId	come-from
Samples.Product	OOABL Obj	2.240	2.240	1.728		0	62	76	0	0	0		90	CreateOrderBob S
Samples.Product	OOABL Obj	2.240	2.240	1.728		0	68	76	0	0	0		99	CreateOrderJohn S
Samples.Product	OOABL Obj	2.240	2.240	1.728		0	74	76	0	0	0		108	CreateOrderDolly S
Samples.Product	OOABL Obj	2.240	2.240	1.728		0	75	76	0	0	0		108	CreateOrderDolly S
- Call Stack:**

Pos	N° ligne	Occur	Caller
1	40		? CreateOrderBob Samples\LauncherTestLeak (72)
2	4,754		? Samples\LauncherTestLeak (55)
3	4,582		? btn-test-1-choose S:\Prowin\ProLkSrc\pug2024\Main-launch.w (54)
4	4,704		? 62-USER-INTERFACE-TRIGGER S:\Prowin\ProLkSrc\pug2024\Main-launch.w (53)
5	0		? S:\Prowin\ProLkSrc\pug2024\Main-launch.w (1)
- Memory Evolution:**

Start	End	Memory used	Tag Start	Tag End
6	6	2.240	Clear Orders	Clear Orders

# // Cas d'utilisation du Memory Profiler

## / Analyse d'un Snapshot

- 1 : Familles d'objets et handles (TT, Procedure, ABLObj...)
- 2 : Types détectées (possibilité de trier en fonction de la famille sélectionnée)
- 3 : Détail des occurrences (possibilité de trier par Famille ou Type sélectionné)
- 4 : Pile d'appel de création de l'occurrence sélectionnée
- 5 : Evolution de la mémoire par Snapshot.

## / Scoped Memory : Mémoire des handles créés



# // Cas d'utilisation du Memory Profiler

## / Analyse d'un Snapshot (Vue Block)

**VIEW 1 Snapshot**

**Information**

SessionID: 153  
Start: 2024-05-22T10:17:13.409+02:00  
Pid: 3024  
SnapShotId: 6  
Time: 22/05/2024 10:17:17.861+02:00  
HMS: 00:00:04.452.543

**Type**  Type  Code block

Sort by:  Nb obj  Memory used  Memory Scoped  Only src (No code block) Search: \_\_\_\_\_

ID	Name	Nb Obj (ts)	Memory (ts)	Nb retained obj	Memory Scoped	Objet	srcmod
1	S:\Prowin\ProLkSrc\pug2024\Main-launch.w	1	13,088	6	177,328	S:\Prowin\ProLkSrc\pug2024\Main-launch.w	
12	adm2\containr.p	1	38,624	20	101,744	adm2\containr.p	
4	start-super-proc S:\Prowin\ProLkSrc\pug2024\Main-launch.w	0	0	4	80,232	S:\Prowin\ProLkSrc\pug2024\Main-launch.w	
73	Samples Order	3	9,888	6	70,728	Samples.Order	
55	Samples\LauncherTestLeak	1	2,264	4	37,992	Samples\LauncherTestLeak	
58	Samples.OrdersList	1	3,152	2	21,168	Samples.OrdersList	
5	adm2\smart.p	1	18,824	2	20,256	adm2\smart.p	
9	adm2\visual.p	1	14,400	1	14,400	adm2\visual.p	
2	adecomm/as-utils.w	1	6,976	2	11,472	adecomm/as-utils.w	
60	CreateCustomers Samples\LauncherTestLeak	0	0	3	11,448	Samples\LauncherTestLeak	

**Details**

Filter:  Tous  Src  Module

Name	Type	Memory Used	Memory Scoped	Pool Allocation	Ao-ScopedTo	Ao-Handle	Ao-Linenum	Ao-SrcId	Ao-TTabId	Ao-UseCount	Ao-WidgetPoolId	platojb-Name	Ao-NodeId	come-from
ADMPProps	Dynamic Temp-table	73,264	73,264	73,032	6	0	2,125	0	0	0	0			1:S:\Prowin\ProLkSr
adecomm/as-utils.w	Procedure	6,976	6,976	6,632		0	27	2	0	0	0			1:S:\Prowin\ProLkSr
tt-propath	Temp-table	1,072	1,072	1,072	2	0	0	0	1	0	0			1:S:\Prowin\ProLkSr
ADMPProps	Dynamic Buffer	432	432	0		0	3,254	0	0	0	0			1:S:\Prowin\ProLkSr
	Widget Pool		73,264	664	2	0	59	0	0	0	0			1:S:\Prowin\ProLkSr

**Call Stack**  Nb Occ

Pos	N° ligne	Occurr.	Caller
1	0		? S:\Prowin\ProLkSrc\pug2024\Main-launch.w (1)

**Memory Evolution**

Start	End	Memory used	Tag Start	Tag End
6	6	73,264	Clear Orders	Clear Orders
5	6	73,264	Create Customers And Orders	Clear Orders
4	5	73,264	Create Empty OrderList	Create Customers And Orders
3	4	73,264	Begin	Create Empty OrderList
2	3	73,264	_AUTO	Begin
1	2		_AUTO	_AUTO

# // Cas d'utilisation du Memory Profiler

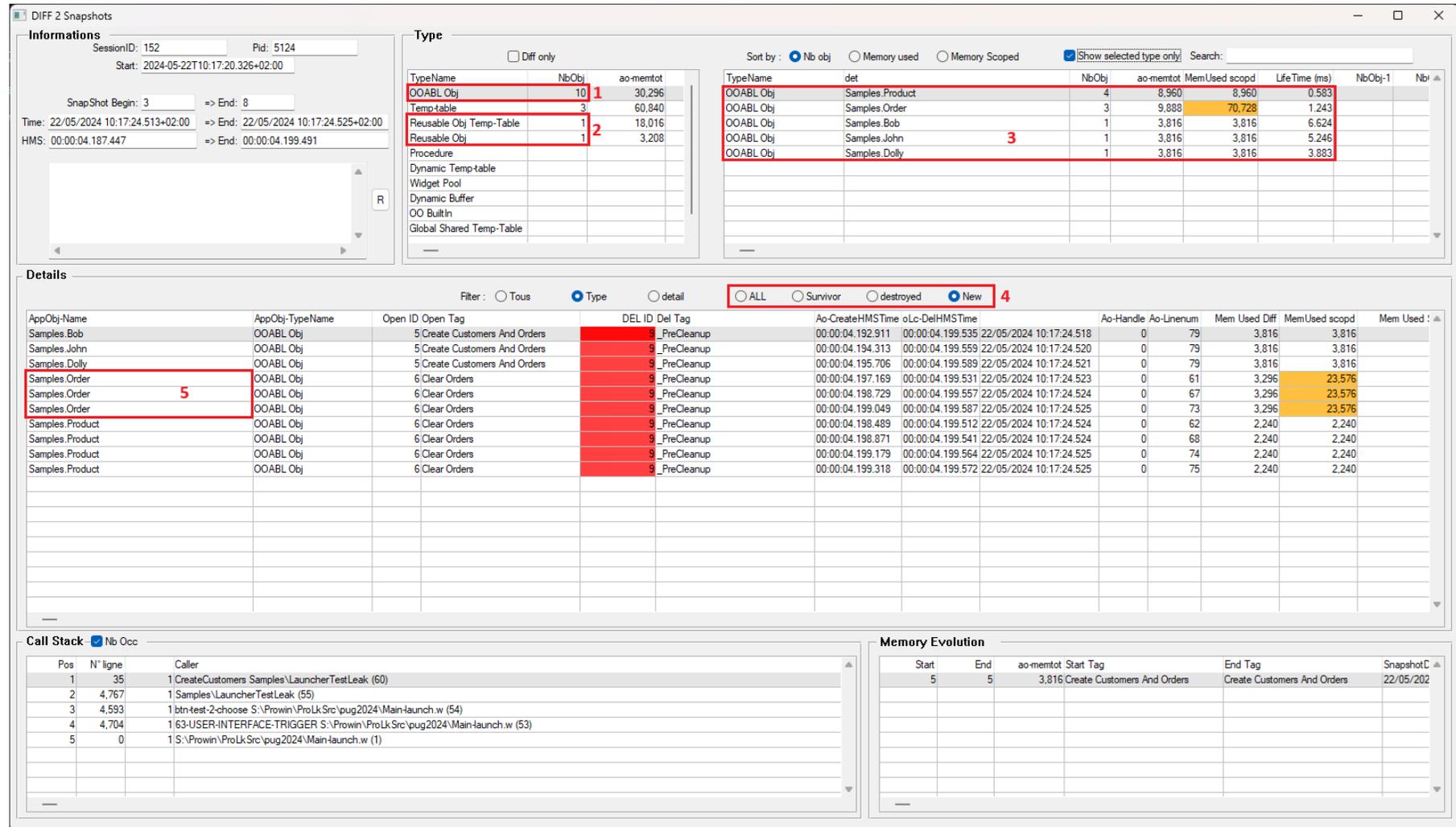
## / Analyse d'un Snapshot

- 1 : Liste des blocks (Obj, procédures, méthodes)
- 2 : Objets et handles créés dans le block sélectionné



# // Cas d'utilisation du Memory Profiler

## / Analyse d'une fuite



**Informations**

SessionID: 152 Pid: 5124  
Start: 2024-05-22T10:17:20.326+02:00

Snapshot Begin: 3 => End: 8  
Time: 22/05/2024 10:17:24.513+02:00 => End: 22/05/2024 10:17:24.525+02:00  
HMS: 00:00:04.187.447 => End: 00:00:04.199.491

**Type**

TypeName	NbObj	ao-mentot
OOABL Obj	10	30.296
Temp-table	3	60.840
Reusable Obj Temp-Table	1	18.016
Reusable Obj	1	3.208

**Details**

Filter:  Tous  Type  detail  ALL  Survivor  destroyed  New 4

AppObj-Name	AppObj-TypeName	Open ID	Open Tag	DEL ID	Del Tag	Ao-CreateHMS	oLc-DelHMS	Ao-Handle	Ao-Linum	Mem Used Diff	MemUsed scopd	Mem Used
Samples.Bob	OOABL Obj	5	Create Customers And Orders	9	_PreCleanup	00:00:04.192.911	00:00:04.199.535	22/05/2024 10:17:24.518	0	79	3.816	3.816
Samples.John	OOABL Obj	5	Create Customers And Orders	9	_PreCleanup	00:00:04.194.313	00:00:04.199.559	22/05/2024 10:17:24.520	0	79	3.816	3.816
Samples.Dolly	OOABL Obj	5	Create Customers And Orders	9	_PreCleanup	00:00:04.195.706	00:00:04.199.589	22/05/2024 10:17:24.521	0	79	3.816	3.816
Samples.Order	OOABL Obj	6	Clear Orders	9	_PreCleanup	00:00:04.197.169	00:00:04.199.531	22/05/2024 10:17:24.523	0	61	3.296	23.576
Samples.Order	OOABL Obj	6	Clear Orders	9	_PreCleanup	00:00:04.198.729	00:00:04.199.557	22/05/2024 10:17:24.524	0	67	3.296	23.576
Samples.Order	OOABL Obj	6	Clear Orders	9	_PreCleanup	00:00:04.199.049	00:00:04.199.587	22/05/2024 10:17:24.525	0	73	3.296	23.576
Samples.Product	OOABL Obj	6	Clear Orders	9	_PreCleanup	00:00:04.198.489	00:00:04.199.512	22/05/2024 10:17:24.524	0	62	2.240	2.240
Samples.Product	OOABL Obj	6	Clear Orders	9	_PreCleanup	00:00:04.198.871	00:00:04.199.541	22/05/2024 10:17:24.524	0	68	2.240	2.240
Samples.Product	OOABL Obj	6	Clear Orders	9	_PreCleanup	00:00:04.199.179	00:00:04.199.564	22/05/2024 10:17:24.525	0	74	2.240	2.240
Samples.Product	OOABL Obj	6	Clear Orders	9	_PreCleanup	00:00:04.199.318	00:00:04.199.572	22/05/2024 10:17:24.525	0	75	2.240	2.240

**Call Stack**

Pos	N° ligne	Caller
1	35	1 CreateCustomers Samples\LauncherTestLeak (60)
2	4,767	1 Samples\LauncherTestLeak (55)
3	4,593	1 btn-test-2-choose S:\Prowin\ProLkSrc\pug2024\Main-launch.w (54)
4	4,704	1 63-USER-INTERFACE-TRIGGER S:\Prowin\ProLkSrc\pug2024\Main-launch.w (53)
5	0	1 S:\Prowin\ProLkSrc\pug2024\Main-launch.w (1)

**Memory Evolution**

Start	End	ao-mentot	Start Tag	End Tag	SnapshotID
5	5	3.816	Create Customers And Orders	Create Customers And Orders	22/05/202

# // Cas d'utilisation du Memory Profiler

## / Analyse d'une fuite

- 1 : On constate que 10 objets restent en mémoire alors que les commandes ont été vidées
- 2 : Reusable Obj désigne des objets supprimés et mis en cache en vue d'une réutilisation ultérieure. Participe aux performances de la session
- 3 : On constate que les 10 objets restés en mémoire correspondent aux commandes, clients et produits.
- 4 : Pour affiner l'analyse, on décide de n'afficher que les objets créés et non supprimés entre le 2 snapshots
- 5 : Les objets de type Order retiennent les produits par la TempTable TTItems (scoped memory)

Level	AppObj-Name	AppObj-TypeName	Ao-MemoryUsed	MemoryUsed scoped	Ao-I
0	Samples.Order	OOABL Obj	3,296	23,576	
1	TTItems	Temp-table	20,280	20,280	

# Cas d'utilisation du Memory Profiler (Production)

# // Cas d'utilisation du Memory Profiler : Production

## / Surveillance de l'évolution de la mémoire générale au cours du temps

- Utilisation du mode Automatique

## / Alerting

Non lus

Rechercher Fuite\_m

📧	DE	OBJET	REÇU
	PPerrot@proginov.com	Surveillance des Fuites mémoires PROGRESS	mar. 18/06/2024 14:13
	PPerrot@proginov.com	Surveillance des Fuites mémoires PROGRESS	mar. 18/06/2024 12:11
	PPerrot@proginov.com	Surveillance des Fuites mémoires PROGRESS	mar. 18/06/2024 10:38

📧 Répondre 📧 Répondre à tous 📧 Transférer

mar. 18/06/2024 14:55



Patrice Perrot

Surveillance des Fuites mémoires PROGRESS

À Patrice Perrot

Stratégie de rétention Suppression personnelle après 1 mois (31 jours)

Date d'expiration 19/07/2024

Bonjour,

Au cours de la demi-heure écoulée (jusqu'à 14:13), la mémoire utilisée par certaines sessions a atteint le seuil d'alerte.

Voici le détail :

- CONTEXTE	PROCESS	QUI	PID	MEMOIRE (Mo)	Architect	MACHINE
-	Prowin32	pnv_ppe	11028	517.4	NON	D4APPP026

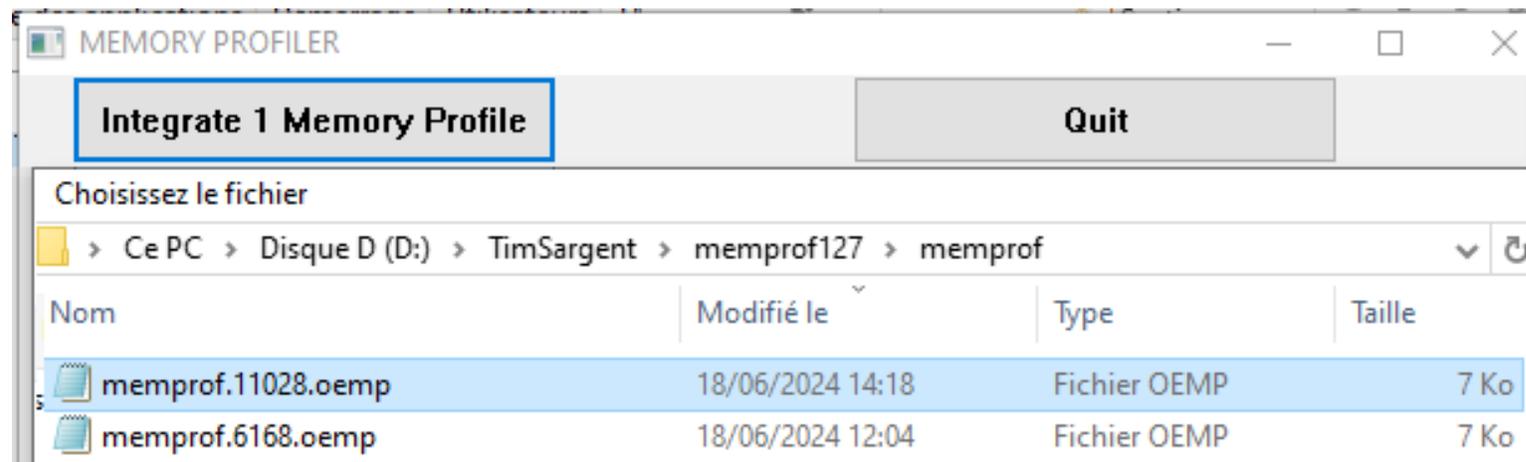
# // Cas d'utilisation du Memory Profiler : Production

## / Une session surconsomme

Gestionnaire des tâches

Processus	Performance	Historique des applications	Démarrage	Utilisateurs	Détails	Services
Nom	PID	Statut	Description	Mémoire (...)		
OE prowin.exe	11028	En cours d'exécuti...	OpenEdge Graphical Client	564 832 Ko		
OE prowin.exe	20916	En cours d'exécuti...	OpenEdge Graphical Client	88 520 Ko		
OE prowin.exe	6168	En cours d'exécuti...	OpenEdge Graphical Client	40 180 Ko		
OE prowin.exe	26436	En cours d'exécuti...	OpenEdge Graphical Client	26 984 Ko		

## / Intégration de son OEMP



# // Cas d'utilisation du Memory Profiler : Production

## / (Re) nommage de la session

Memory Profiler

**Memory Profiler list**

Integrate new file Delete Sort by :  N° Session  Date New Name: PNV\_PPE D4appP026 (PROGI) Rename session 160

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-mark.3.p	19	2024-05-24T08:48:10.709+02:00	26200	AVM-Client	126,848.316	43,781.474
155	bech-mark.2.p	26	2024-05-23T17:59:34.903+02:00	24188	AVM-Client	156,896,249.769	969,613.231

## / S'affecter sur la mesure

Memory Profiler

**Memory Profiler list**

Integrate new file Delete Sort by :  N° Session  Date New Name: PNV\_PPE D4appP026 (PROGI) Rename session 160  Mine Affect me

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



# // Cas d'utilisation du Memory Profiler : Production

**DEMO**

# // Cas d'utilisation du Memory Profiler : Production

## / DEMO 1

VIEW 1 SnapShot
— □ ×

**Informations**

SessionID: 160

Start: 2024-06-18T14:18:46.947+02:00

Pid: 11028

SnapShotId: 54

Time: 18/06/2024 14:21:19.900+02:00

HMS: 00:02:32.953.259

Type  Code block

Sort by:  Nb obj  Memory used  Memory Scoped  Show selected type only Search:

Type	Nb Obj	Memory Used	Nb Obj	Memory Used	Memory Scoped	LifeTime (ms)	lifetime-SUM	lifenin
Dynamic Buffer	22,344	9,653,400	Dynamic Query	ped 1195->63	11,170	310,794,080	310,794,080	0.000
Dynamic Query	11,170	310,794,080	Dynamic Buffer	Ship To	11,170	4,825,440	4,825,440	0.000
OOABL Obj	72	396,200	Dynamic Buffer	State	11,170	4,825,440	4,825,440	0.000
Temp-table	63	298,968	OOABL Obj	OpenEdge.Logging.Logger	5	34,600	34,600	0.000
OOABL Static Obj	55	303,740	OOABL Obj	Root.Logging.RootLogger	5	28,840	28,840	0.000
Widget Pool	19	81,168	OOABL Obj	Core.Framework.Logging.CoreDefaultLogWriter	5	16,360	16,360	0.000
Reusable Obj	11	63,400	OOABL Obj	OpenEdge.Core.Util.BuilderRegistry	4	31,040	31,040	0.000
Procedure	11	230,110	Widget Pool	RootLogge 318->110	4			0.000
OO Static Temp-table	9	94,208	OOABL Obj	Dbg.Model.DebugMode.DbgToolkit	3	20,796	22,884	0.000
DB Connection	6	847,792	Widget Pool	DbgToolkit 812->491	3		2,088	0.000

**Details**

Filter:  ALL  Type  Type and Det

Name	Type	Memory Used	Memory Scoped	Pool Allocation	Ao-ScopedTo	Ao-Handle	Ao-Linenum	Ao-Scrid	Ao-TTabId	Ao-UseCount	Ao-WidgetPoolId	platobj-Name	Ao-NodeId	come-from
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome
	Dynamic Query	27,824	27,824	2,248	1,897	0	101	0	0	0	0		7,490	take_info_custome

**Call Stack**  Nb Occ

Pos	N° ligne	Occurr.	Caller
1	63	?	take_info_customer c:\va_poubelle\protemp\p80444_Untitled2.ped (1195)
2	22	?	init-tt-item c:\va_poubelle\protemp\p80444_Untitled2.ped (1194)
3	701	?	c:\va_poubelle\protemp\p80444_Untitled2.ped (1193)
4	3,600	?	adecomm/_runtime.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)

**Memory Evolution**

Start	End	Memory used	Tag Start	Tag End
22	22	27,824	_AUTO	_AUTO

PROG/NOV

Propriété de PROG/NOV – reproduction interdite / Page 33

# // Cas d'utilisation du Memory Profiler : Production

## / DEMO 2 : Zoom sur les points d'intérêts

Type  Code block

Sort by :  Nb obj  Memory used  Memory Scoped  Show selected type only Search:

Type	Nb Obj	Memory Used	Type	Name	Nb Obj	Memory Used	Memory Scoped
Dynamic Buffer	22,344	9,653,400	Dynamic Query	ped 1195->63	11,170	310,794,080	310,794,080
Dynamic Query	11,170	310,794,080	Dynamic Buffer	ShipTo	11,170	4,825,440	4,825,440
OOABL Obj	72	396,200	Dynamic Buffer	State	11,170	4,825,440	4,825,440
Temp-table	63	298,968	OOABL Obj	OpenEdge.Logging.Logger	5	34,600	34,600
OOABL Static Obj	55	303,740	OOABL Obj	Root.Logging.RootLogger	5	28,840	28,840
Widget Pool	19	81,168	OOABL Obj	Core.Framework.Logging.CoreDefaultLogWriter	5	16,360	16,360
Reusable Obj	11	63,400	OOABL Obj	OpenEdge.Core.Util.BuilderRegistry	4	31,040	31,040
Procedure	11	230,110	Widget Pool	RootLogge 318->110	4		
OO Static Temp-table	9	94,208	OOABL Obj	Dbg.Model.DebugMode.DbgToolkit	3	20,796	22,884
DB Connection	6	847,792	Widget Pool	DbgToolkit 812->491	3		2,088

Call Stack  Nb Occ

Pos	N° ligne	Occurr.	Caller
1	63	?	take_info_customer c:\va_poubelle\protemp\p80444_Untitled2.ped (1195)
2	22	?	init-tt-item c:\va_poubelle\protemp\p80444_Untitled2.ped (1194)
3	701	?	c:\va_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)



# // Cas d'utilisation du Memory Profiler : Production

## / DEMO 3 : Vue code block => ZOOM

Type  Code block

Sort by :  Nb obj  Memory used  Memory Scoped  Only src (No code block) Search:

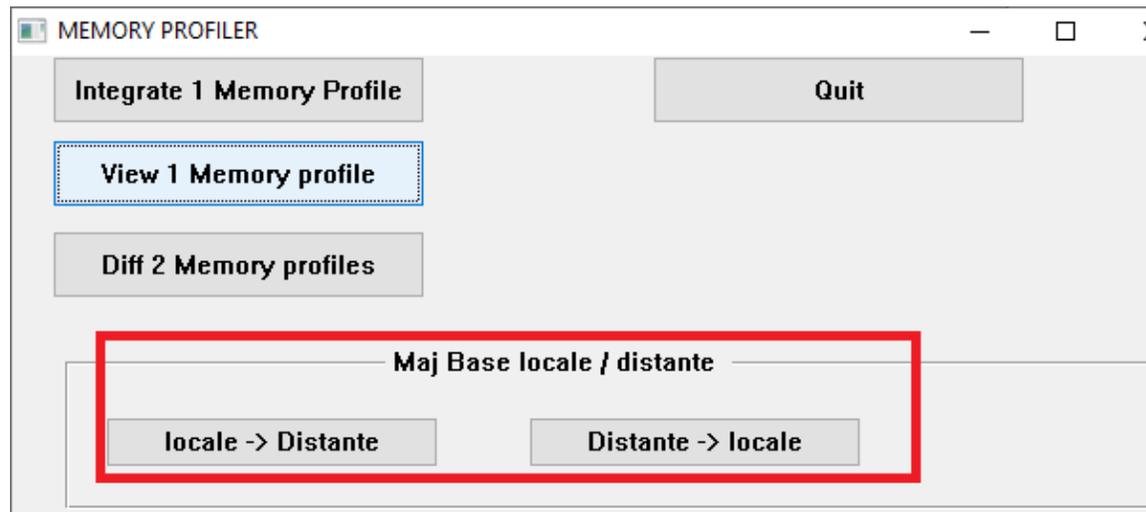
ID	Name	Nb Obj (its)	Memory (its)	Nb retained obj	Memory Scoped	Objet
1,193	c:\Va_poubelle\protemp\p80444_Untitled2.ped	1	5,664	2	320,453,248	c:\Va_
1,195	take_info_customer c:\Va_poubelle\protemp\p80444_Untitled2.ped	0	0	33,510	320,444,960	c:\Va_
165	c:\Va_poubelle\protemp\pbx07564\CnxBaseU.r	0	0	3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

## // Memory Profiler : Base significative en taille

/ Base locale pour les analyses personnelles

/ Base distante pour les analyses partagées

/ Possibilité de transfert entre le local et le distant



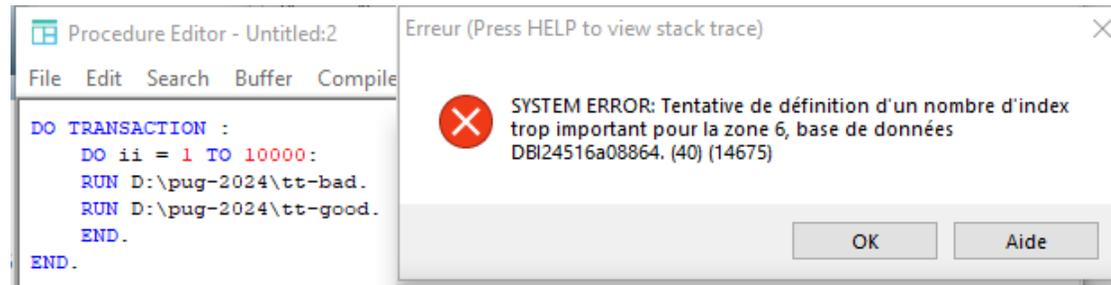
→ Meilleures perfs en base local connectée en SELF

## // Memory Profiler : What Else

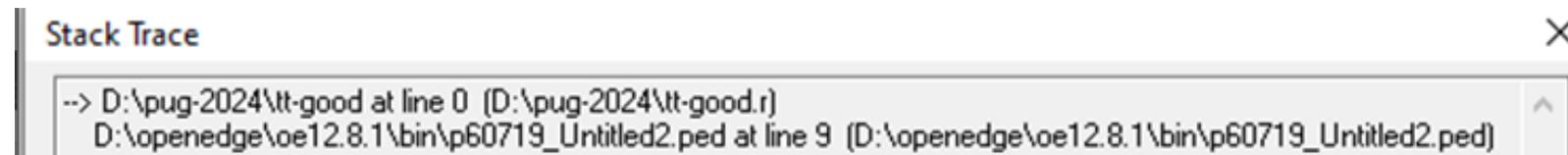
**DEMO**

# // Memory Profiler : Hors de son périmètre

## / Erreur 14675

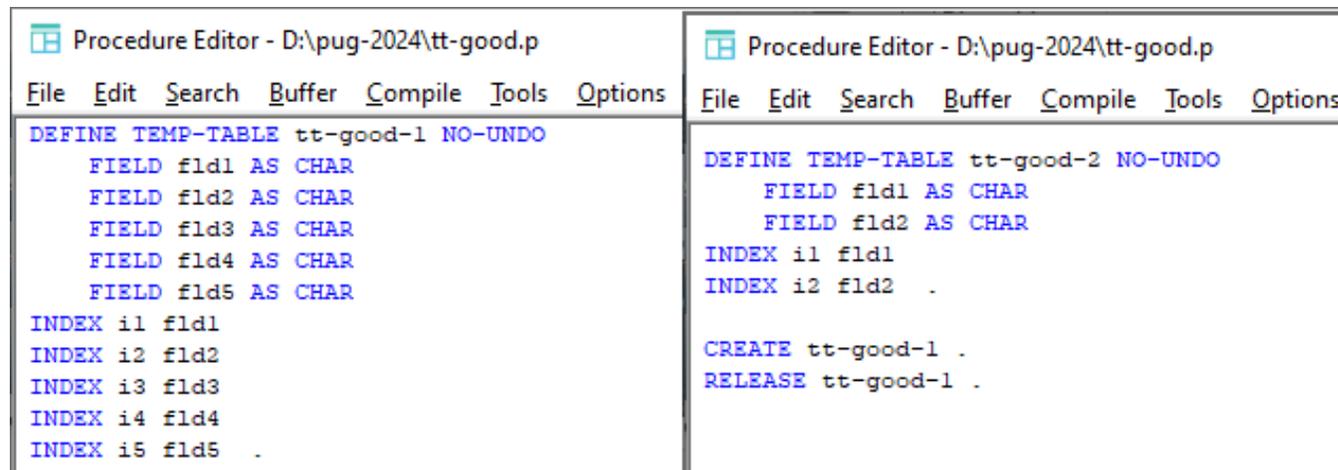


## / Pas de boucle infinie ,



## / Pas de problème sur le programme

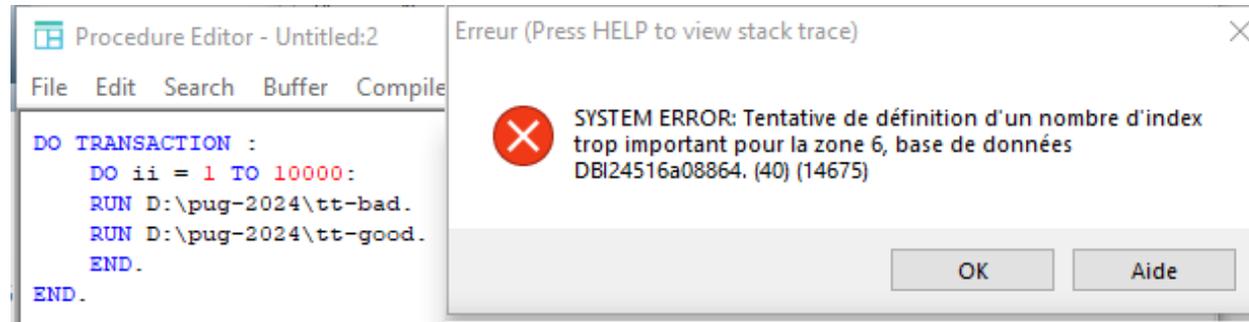
/



(7 index )

# // Memory Profiler : Hors de son périmètre

## / Erreur 14675



## / Pas de fuite , programmes en mémoire (protrace.\*\*\*\*\*)

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 adcomm/as-utils.r

## // Memory Profiler : Hors de son périmètre



SYSTEM ERROR: Tentative de définition d'un nombre d'index trop important pour la zone 6, base de données DBI24516a08864. (40) (14675)

### / Pas de fuite , programmes en mémoire (protrace.\*\*\*\*\*)

 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

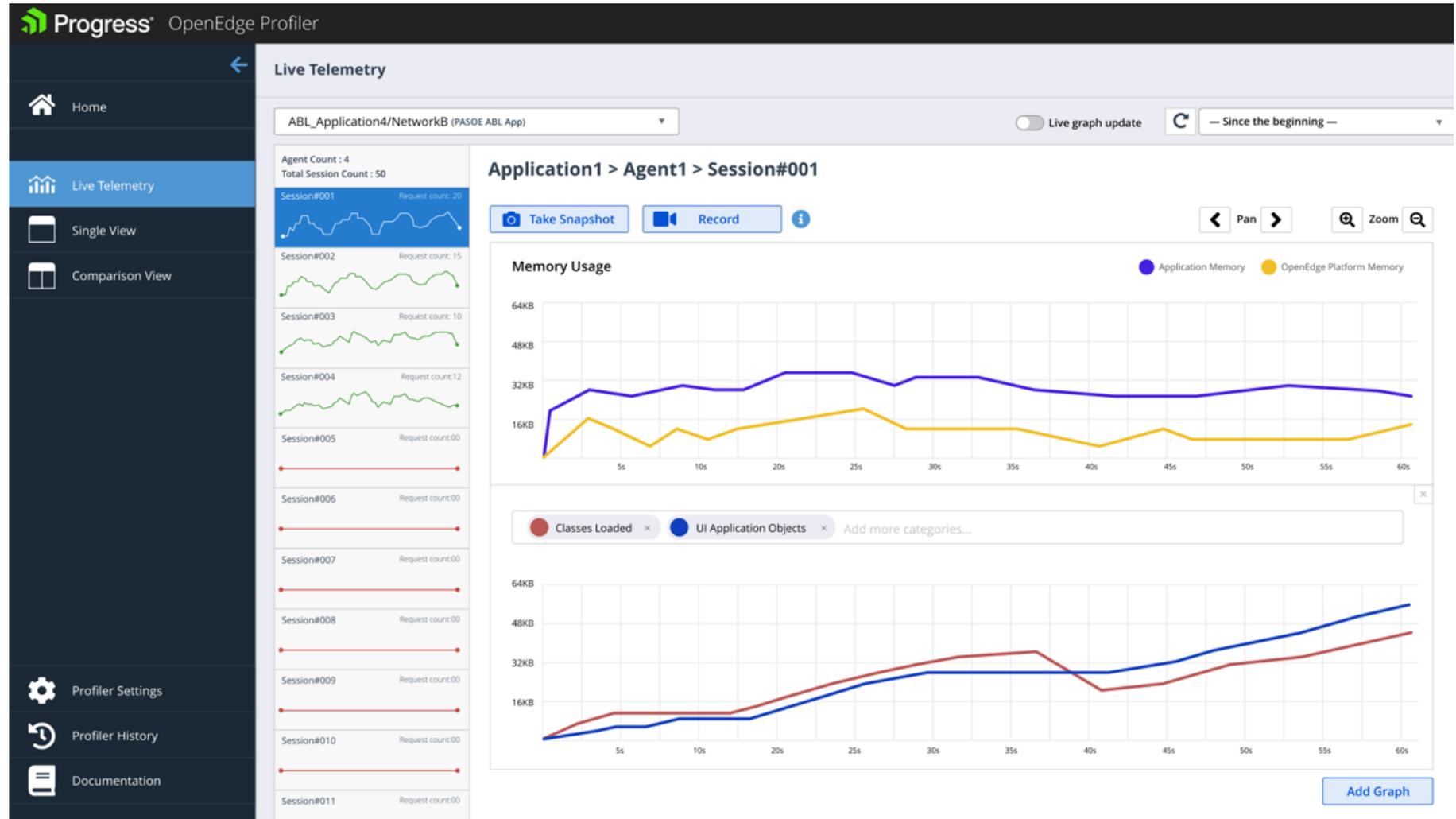
001010 D:\openedge\oe12.8.1\gui\adm2\appserver.r

001005 D:\openedge\oe12.8.1\gui\adm2\smart.r

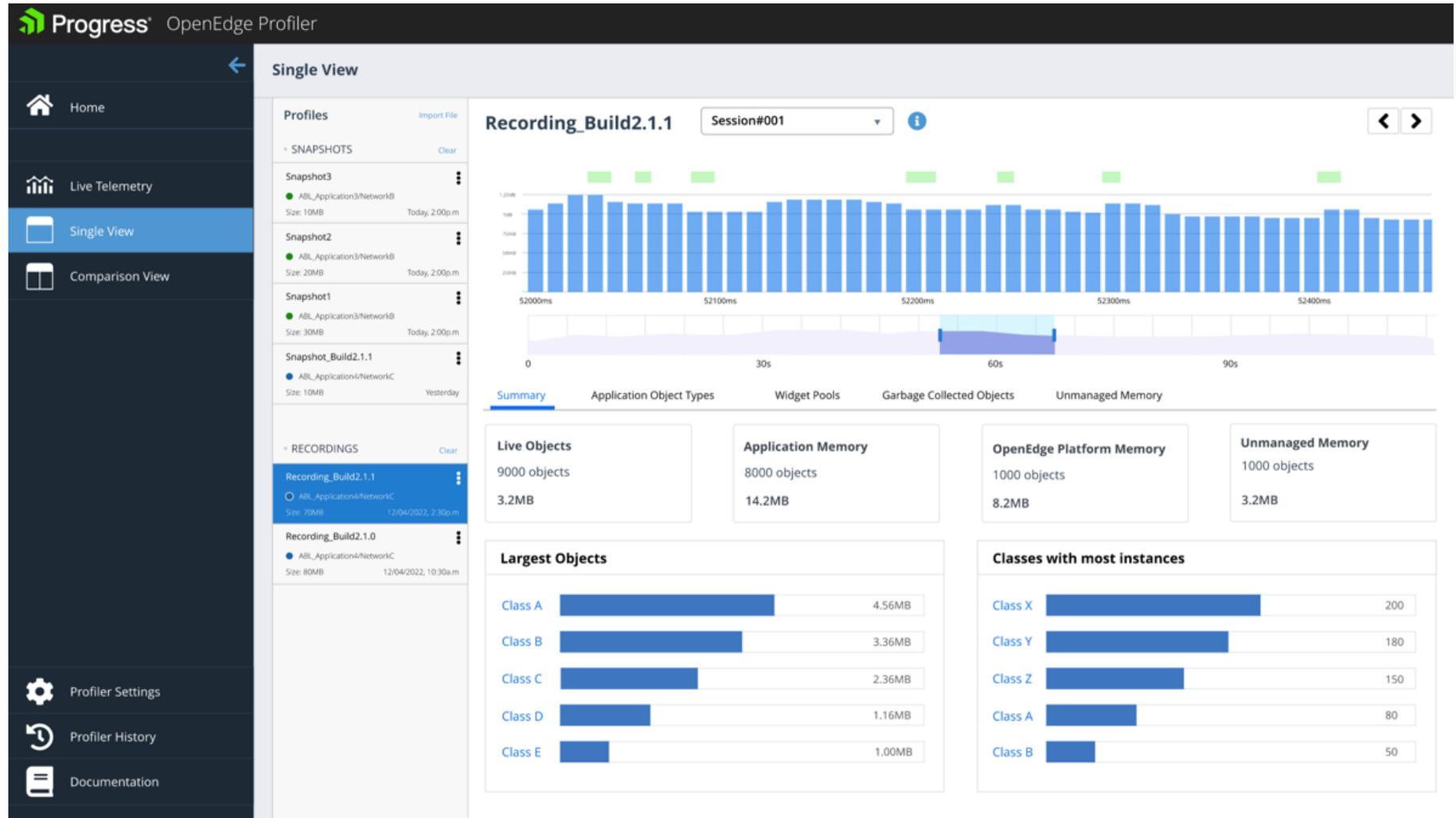
001001 adecomm/as-utils.r

**Ce que Progress a déjà présenté**

# // Ce que Progress a déjà présenté



# // Ce que Progress a déjà présenté



# Pistes d'amélioration



## // Pistes d'amélioration (internes)

- / **Vue graphique (répartition de mémoire par type...)**
- / **Développer une vue prenant en compte des moyennes sur plusieurs Snapshots pour déterminer des tendances**
- / **Mise en production de l'intégration continue des fichiers OEMP**
- / **Formation des équipes et généralisation du Memory Profiler dans les processus de qualité de code**



## // Pistes d'amélioration (remontées à Progress)

### / Pouvoir lier les consommations de mémoire aux objets et pas seulement aux TreeNodes

- Permettrait d'obtenir un cumul de la rétention de mémoire générée pour chaque objet

### / Pouvoir exclure les objets et handles système de Progress

- Peut polluer la lecture des analyses

### / Pouvoir visualiser les références aux objets

- Permettrait de comprendre pourquoi le Garbage Collector ne nettoie pas un objet

### / Lors d'un Diff, proposer des sous-ensembles d'objets (créés, détruits, survivants)

- Un objet créé et non détruit entre 2 snapshot peut être révélateur d'une fuite

**Des Questions ?**



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

