

# Université **IBM i**

## 7 novembre 2023

IBM Innovation Studio Paris

### S02 – Les services SQL pour les développeurs

11:15 / 12:15

**Nathanaël Bonnet**

Gaia/Volubis

*nathanael.bonnet@gaia.fr*

 **infrasdufutur**

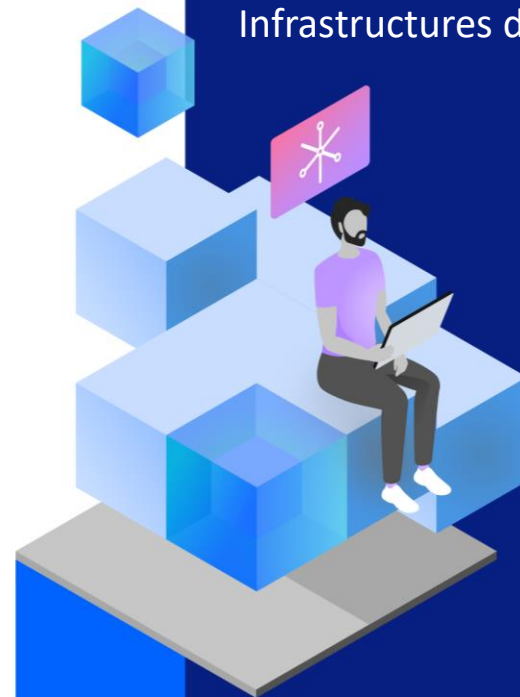
#ibmi

#uui2023

#infrastructuredufuturIBM23



Infrastructures du futur



7 et 8 novembre 2023

# Présentation

- Nathanaël BONNET
  - IBM i depuis 1999
  - Expert IBM i
  - Développement & intégration



# Agenda

- 1. Introduction
  - Rappels
  - Périmètre
  
- 2. Exemples de services applicatifs
  - Manipulations d'objets : Data Area, Data Queue
  - Accès aux spools
  - Exécution de commande CL
  - Intégration dans vos programmes
  
- 3. Exemples de services utiles dans la gestion des développements
  - Pour vos développements ILE
  - Pour vos analyses d'impact

IBM



Infrastructures du  
futur

7 et 8 novembre 2023

Université **IBM i**

7 novembre 2023

# 1. Introduction



Let's  
Create

# Services SQL

- Permettent d'accéder à des informations système
  - Souvent similaire à
    - Des commandes CL
    - Des APIs
  
- Fournis par IBM
  - Livrés par les versions + TR
  - Maintenus par PTF
  - Sous plusieurs formes
    - Tables, vues, fonctions scalaires, fonctions tables, procédures
  
- Facilité d'usage par rapport aux commandes/APIs
  - Pas besoin de programme!

# Services SQL

## ■ Références

- Par le TR : <https://www.ibm.com/support/pages/ibm-i-services-sql>
  - ou <http://ibm.biz/Db2foriServices>
  - ou <https://www.ibm.com/support/pages/node/1119123>
- Docs : <https://www.ibm.com/docs/en/i/7.5?topic=optimization-i-services>

- Plus de 300 services :

Applications	Objets et bibliothèques	Stockage
BRMS	Messages	Travaux
Cache de plans SQL	Performances	Utilitaires
Centre de santé	PowerHA	
Communication	Produits	
IFS	PTFs	
Java	Sécurité	
Journaux	Spoules	

- Un service unique pour les gouverner tous, un service unique pour les trouver ... !
  - QSYS.SERVICES\_INFO

```

21
22 -- Liste des services
23 select * from QSYS2.SERVICES_INFO ;
    
```

SERVICE_CATEGORY	SERVICE_SCHEMA_NAME	SERVICE_NAME	SQL_OBJECT_TYPE	OBJECT_TYPE	SYSTEM_OBJECT_NAME	LATEST_DB2_GROUP_LEVEL	INITIAL_DB2_GROUP_LEVEL	EARLIEST_POSSIBLE_RELEASE	EXAMPLE
PTF	QSYS2	PTF_INFO	VIEW	*FILE	PTF_INFO	25	-	-V6R1M0	-- Description: Will an IPL impact PTF state? SELECT PTF_
PTF	QSYS2	GROUP_PTF_INFO	VIEW	*FILE	GRPPTFINFO	-	-	-V6R1M0	-- Description: What is the most recently installed CUM? :
PTF	SYSTOOLS	GROUP_PTF_CURRENCY	VIEW	*FILE	GRPPTFCUR	10	-	-V7R1M0	-- Description: Determine whether the PTF Groups are curren
PTF	SYSTOOLS	GROUP_PTF_DETAILS	VIEW	*FILE	GRPPTFDTL	10	-	-V7R1M0	-- Description: Review all unapplied PTFs contained within
SECURITY	QSYS2	USER_INFO	VIEW	*FILE	USER_INFO	13	-	-V6R1M0	-- Description: Which users are at risk of becoming disable
SECURITY	QSYS2	FUNCTION_INFO	VIEW	*FILE	FCN_INFO	-	-	-V6R1M0	-- Description: Which function usage IDs exist. SELECT * FR
SECURITY	QSYS2	FUNCTION_USAGE	VIEW	*FILE	FCN_USAGE	-	-	-V6R1M0	-- Description: Which Database function usage has been gran
SECURITY	QSYS2	GROUP_PROFILE_ENTRIES	VIEW	*FILE	GROUPPLIST	-	-	-V6R1M0	-- Description: Review *ALLOBJ users. SELECT AUTHORIZATION
SECURITY	QSYS2	SQL_CHECK_AUTHORITY	SCALAR FUNCTION	-	-	-	-	-V6R1M0	-- Description: Does this user have authority to query this
SECURITY	QSYS2	SET_COLUMN_ATTRIBUTE	PROCEDURE	-	-	-	-	-V6R1M0	-- Description: Secure salary column values in the SQL Perf
SECURITY	QSYS2	DRDA_AUTHENTICATION_ENTRY_INFO	VIEW	*FILE	DRDA_AUTHE	-	-	-V7R1M0	-- Description: Retrieve the DRDA Authentication Entry info
WORK MANAGEMENT	QSYS2	SYSTEM_VALUE_INFO	VIEW	*FILE	SYVALINFO	-	-	-V6R1M0	-- Description: Compare System Values across two partitions
WORK MANAGEMENT	QSYS2	GET_JOB_INFO	TABLE FUNCTION	-	-	-	-	-V6R1M0	-- Description: Find the jobs with SQL statements executing
WORK MANAGEMENT	QSYS2	ACTIVE_JOB_INFO	TABLE FUNCTION	-	-	-	-	-V7R1M0	-- Description: Find the top 10 consumers of CPU in the QUS
WORK MANAGEMENT	QSYS2	SCHEDULED_JOB_INFO	VIEW	*FILE	SCHED_JOB	-	-	-V7R1M0	-- Description: Review the job scheduled entries which are
WORK MANAGEMENT	QSYS2	MEMORY_POOL	TABLE FUNCTION	-	-	-	-	-V7R1M0	-- Description: Return all available pool information. SEL
WORK MANAGEMENT	QSYS2	MEMORY_POOL_INFO	VIEW	*FILE	POOL_INFO	-	-	-V7R1M0	-- Description: Review the details of all active memory poo
WORK MANAGEMENT	QSYS2	SYSTEM_STATUS	TABLE FUNCTION	-	-	25	-	-V7R1M0	-- Description: Review elapsed system status over a 60 seco
WORK MANAGEMENT	QSYS2	SYSTEM_STATUS_INFO	VIEW	*FILE	SYS_STATUS	25	-	-V7R1M0	-- Description: Review current system status, including the
WORK MANAGEMENT	QSYS2	OBJECT_LOCK_INFO	VIEW	*FILE	OBJ_LOCK	-	-	-V7R1M0	-- Description: Find any object locks over the TOYSTORE/SAL
WORK MANAGEMENT	QSYS2	RECORD_LOCK_INFO	VIEW	*FILE	RCD_LOCK	-	-	-V7R1M0	-- Description: Find any record locks over the TOYSTORE/SAL
MESSAGE HANDLING	QSYS2	REPLY_LIST_INFO	VIEW	*FILE	REPLYLIST	-	-	-V6R1M0	-- Description: Review reply list detail for all messages w
MESSAGE HANDLING	QSYS2	JOBLOG_INFO	TABLE FUNCTION	-	-	-	-	-V6R1M0	-- Description: Find the top 25 consumers of CPU in active
LIBRARIAN	QSYS2	LIBRARY_LIST_INFO	VIEW	*FILE	LIBLIST	-	-	-V6R1M0	-- Description: Ensure that the TOYSTORE library is the fir
LIBRARIAN	QSYS2	OBJECT_STATISTICS	TABLE FUNCTION	-	-	-	-	-V6R1M0	-- Description: Find programs that are the least utilized
STORAGE	QSYS2	USER_STORAGE	VIEW	*FILE	USER_STG	-	-	-V6R1M0	-- Description: Review the top 10 storage consumers, by use
STORAGE	QSYS2	SYSDISKSTAT	VIEW	*FILE	SYSDISKS	25	-	-V6R1M0	-- Description: Query information for all storage devices
STORAGE	QSYS2	SYSTMPSTG	VIEW	*FILE	SYSTMPSTG	-	-	-V7R2M0	-- Description: Looking at the database server jobs, find t

# Nous ne verrons pas

- La manipulation de l'IFS

- ➔ Voir session S19 – Apprivoiser l'IFS avec les services SQL – 16h à 17h – Philippe Bourgeois

```

1  -- Lecture fichier log serveur IWS : recherche de tous les appels à convertTempRest
2  select line_number, line
3  from table ( qsys2.ifs_read( path_name => '/www/UNIV2022/wlp/usr/servers/UNIV2022/logs/http_access.log',
4                               end_of_line => 'ANY',
5                               maximum_line_length => default,
6                               ignore_errors => 'NO') )
7  where line like '%convertTempRest%' ;|

```

LINE_NUMBER	LINE
1	127.0.0.1 - [08/Dec/2022:09:52:15 +0100] "GET /web/services/convertTempRest/50 HTTP/1.1" 200 18
2	127.0.0.1 - [08/Dec/2022:10:01:10 +0100] "GET /web/services/convertTempRest/50 HTTP/1.1" 200 18
3	127.0.0.1 - [08/Dec/2022:10:01:17 +0100] "GET /web/services/convertTempRest/50 HTTP/1.1" 200 18
4	127.0.0.1 - [08/Dec/2022:10:01:37 +0100] "GET /web/services/convertTempRest/50 HTTP/1.1" 200 18
5	127.0.0.1 - [08/Dec/2022:10:03:04 +0100] "GET /web/services/convertTempRest/50 HTTP/1.1" 200 18
6	127.0.0.1 - [08/Dec/2022:10:03:04 +0100] "GET /web/services/convertTempRest/50 HTTP/1.1" 200 18
7	127.0.0.1 - [08/Dec/2022:10:03:04 +0100] "GET /web/services/convertTempRest/50 HTTP/1.1" 200 18
8	127.0.0.1 - [08/Dec/2022:10:09:11 +0100] "GET /web/services/convertTempRest/50 HTTP/1.1" 200 18



# Nous ne verrons pas

- La gestion de la sécurité des vos IBM i
  - ➔ Voir session S23 – Administration moderne de la sécurité IBM i avec les services SQL – 16h à 17h – Dominique Gayte

```

10 -- Liste des utilisateurs *ALLOBJ actifs, en direct ou par héritage du profil de groupe
11 SELECT AUTHORIZATION_NAME, NO_PASSWORD_INDICATOR, PREVIOUS_SIGNON, TEXT_DESCRIPTION
12 FROM QSYS2.USER_INFO
13 WHERE   status = '*ENABLED' and SPECIAL_AUTHORITIES LIKE '%*ALLOBJ%'
14         OR AUTHORIZATION_NAME IN ( SELECT USER_PROFILE_NAME
15                                   FROM QSYS2.GROUP_PROFILE_ENTRIES
16                                   WHERE GROUP_PROFILE_NAME IN ( SELECT AUTHORIZATION_NAME
17                                                                     FROM QSYS2.USER_INFO
18                                                                     WHERE SPECIAL_AUTHORITIES like '*ALLOBJ%' ) )
19 ORDER BY AUTHORIZATION_NAME;

```

AUTHORIZATION_NAME	NO_PASSWORD_INDICATOR	PREVIOUS_SIGNON	TEXT_DESCRIPTION
AB	NO	2023-08-17 17:11:20.000000	
AR	NO	2023-11-02 17:08:35.000000	
BS	NO	-	
CGRIERE	NO	2023-11-02 15:58:17.000000	
DEMOUNI1	NO	-	

# Nous ne verrons pas

- Consommation de Web Services, production et parsing de XML/JSON
  - ➔ Voir les archives de l'Université IBM i

```

34 VALUES
35     HTTP_POST(
36         'https://fakereapi.azurewebsites.net/api/v1/Activities',
37         CAST(
38             JSON_OBJECT(
39                 KEY 'id' VALUE TRIM('14092022'),
40                 KEY 'title' VALUE TRIM('Univ 2022'),
41                 KEY 'dueDate' VALUE (          VARCHAR_FORMAT(current_timestamp, 'YYYY-MM-DD') CONCAT
42                                         'T' CONCAT VARCHAR_FORMAT(current_timestamp, 'HH24:MI:SS.NNNNNN') CONCAT 'Z'),
43                 KEY 'completed' VALUE TRIM('true') FORMAT JSON
44             ) AS CLOB(10K)),
45         '{"header": "Accept,application/json",
46         "header": "Content-Type,application/json"}');
47

```

00001

```
{ "id": "14092022", "title": "Univ 2022", "dueDate": "2022-09-11T23:50:37.009014Z", "completed": true }
```



Université **IBM i**

7 novembre 2023

## 2. Exemples de services applicatifs



Let's  
Create

# Data Area

- Possibilité de lire une \*DTAARA uniquement !
  - Equivalent RTVDTAARA / QWCRDTAA
  
- 2 services
  - DATA\_AREA\_INFO (UDTF)
    - Retourne la valeur d'une \*DTAARA
    - Retourne la description
    - Permet l'accès à \*GDA/\*LDA/\*PDA
  - DATA\_AREA\_INFO (vue)
    - Retourne la valeur des \*DTAARA
    - Retourne la description des \*DTAARA

# DATA\_AREA\_INFO (vue)

```

25 -- *DTAARA
26 select * from qsys2.data_area_info
27 where data_area_library = 'GJOURN';
  
```

DATA_AREA_LIBRARY	DATA_AREA_NAME	DATA_AREA_TYPE	LENGTH	DECIMAL_POSITIONS
GJOURN	ANACTUSR	*CHAR	36	-
GJOURN	B_RCVJRN	*CHAR	36	-
GJOURN	CREATFIC	*CHAR	40	-
GJOURN	DERPOSTE	*DEC	10	0
GJOURN	LOG	*CHAR	2000	-

SQL_SEQUENCE	TEXT_DESCRIPTION
NO	-
NO	-
NO	Zone de traitement de creatfic
NO	dernier poste traité
NO	-

DATA_AREA_VALUE	
ANACTUSR QSECOFR	7581670000170823
B_RCVJRN QSECOFR	7572750000164638
QPADEV0002QSECOFR	320883 ZTABLX
2803019	
000000002803220RPT040723183323*OMITTED	000000*OMITTED QADBREF QSYS QADBREF 000024...

DATA_AREA_BINARY_VALUE
C1D5C1C3D9E3E4E2D940D8E2C5C3D6C6D9404040F7F5F8F1F6F7F0F0F0F0F1F7F0F8F2F3
C26DD9C3E5D1D9D54040D8E2C5C3D6C6D9404040F7F5F7F2F7F5F0F0F0F0F1F6F4F6F3F8
D8D7C1C4C5E5F0F0F0F2D8E2C5C3D6C6D9404040F3F2F0F8F8F3404040E9E3C1C2D3E74040404
F2F8F0F3F0F1F9
F0F0F0F0F0F0F0F0F2F8F0F3F2F2F0D9D7E3F0F4F0F7F2F3F1F8F3F3F2F35CD6D4C9E3E3C5C440404040404040404040F0F...

# DATA\_AREA\_INFO (vue)

- Nous accédons à la valeur en caractère et en binaire
  - Pour les \*DTAARA numériques, la valeur en caractères est une conversion
  - Séparateur décimal est le « . »
  
- La colonne SQL\_SEQUENCE
  - 'O' si la \*DTAARA contient une séquence SQL
  - 'N' sinon

# DATA\_AREA\_INFO (UDTF)

- Nom et bibliothèque obligatoires
  - Ne peut pas être un nom générique => retourne 0 ou 1 ligne
  - Le nom peut être \*LDA/\*GDA/\*PDA

```

29  -- *DTAARA GJOURN/LOG
30  SELECT *
31  FROM TABLE(QSYS2.DATA_AREA_INFO(DATA_AREA_NAME => 'LOG',
32                                   DATA_AREA_LIBRARY => 'GJOURN'));
```

DATA_AREA_LIBRARY	DATA_AREA_NAME	DATA_AREA_TYPE	LENGTH	DECIMAL_POSITIONS	DATA_AREA_VALUE
GJOURN	LOG	*CHAR	2000		- 000000002803220RPT0

- Même result sets que la vue
  - Sauf TEXT\_DESCRIPTION

# Data Area

- Droits
  - Vous devez bien entendu disposer des droits
    - \*EXECUTE sur la bibliothèque
    - \*USE sur la data area
  
- Vue : pas d'erreur, pas d'enregistrement

```

34 -- ^DTAARA GJOURN/LOG
35 select * from qsys2.data_area_info
36   where data_area_library = 'NB' and data_area_name = 'PRIVATE' ;
37

```

DATA_AREA_LIBRARY	DATA_AREA_NAME	DATA_AREA_TYPE	LENGTH	DECIMAL_POSITIONS
-------------------	----------------	----------------	--------	-------------------



# Data Area

- UDTF : au choix
  - Par défaut

```
SELECT *
FROM TABLE(QSYS2.DATA_AREA_INFO(DATA_AREA_NAME => 'PRIVATE',
                                DATA_AREA_LIBRARY => 'NB'));
```

## Message d'erreur



Etat SQL : 42501

Code fournisseur : -443

Message : [SQL0443] NOT AUTHORIZED TO NB/PRIVATE Cause . . . . : Un programme de déclenchement, une procédure externe ou une fonction externe a détecté et signalé une erreur à SQL. Si l'erreur s'est produite dans un programme de déclenchement, le déclencheur est dans la table QDBSSUDF2 du schéma QSYS. Si l'erreur s'est produite dans une procédure ou une fonction externe, le nom externe est QDBSSUDF2 dans le schéma QSYS. Le texte associé est NOT AUTHORIZED TO NB/PRIVATE. Si l'erreur s'est produite dans un programme de déclenchement, le texte associé correspond au type de celui-ci. Si l'erreur s'est produite dans une fonction externe, le texte associé correspond au texte du message d'erreur renvoyé par celle-ci. Que faire . . . : Reportez-vous à l'historique du travail pour plus d'informations sur l'erreur détectée. Corrigez l'erreur, puis renouvelez votre demande.

OK

# Data Area

- Sinon

```

41 SELECT *
42 FROM TABLE(QSYS2.DATA_AREA_INFO(DATA_AREA_NAME => 'PRIVATE',
43                                  DATA_AREA_LIBRARY => 'NB',
44                                  IGNORE_ERRORS => 'YES' ));
45

```

DATA_AREA_LIBRARY	DATA_AREA_NAME	DATA_AREA_TYPE	LENGTH	DECIMAL_POSITIONS	D
-------------------	----------------	----------------	--------	-------------------	---

# Data Area

- Quel intérêt ?
  - RPG et CL savent déjà accéder aux \*DTAARA
  - **En lecture et écriture !**
  
- Plusieurs raisons
  - Facilite la lecture de la \*DTAARA lorsque le nom est connu dynamiquement
  - Usage de la valeur contenue dans une \*DTAARA dans une requête SQL
    - En tant que critère de sélection, jointure ...
    - Exemple : date d'effet
  - Effectuer une recherche de valeur contenue dans une \*DTAARA sans connaître son nom

# Data Area

- Exemple

```
46 select * from qsys2.data_area_info
47   where upper(data_area_value) like '%NEPTUNE%';
48
```

DATA_AREA_LIBRARY	DATA_AREA_NAME	DATA_AREA_TYPE	LENGTH	DECIMAL_POSITIONS	DATA_AREA_VALUE
GAIA	SYSTRCOLD	*CHAR	100		- neptune GAI AFTP
GAIA	SYSTRSRC	*CHAR	200		- neptune GAI AFTP *NO
GAIA	SYSTRCOLD	*CHAR	100		- neptune GAI AFTP
GCOMPARE1	GCOMPARE	*CHAR	20		- NEPTUNE URANUS
QUSRSYS	QRCLSTG	*CHAR	1000		- 1200329 220543 1200329
SAVCENTSAV	SAVCENT	*CHAR	1024		- SAUVE RESTAURE /Sauvegarde/NEPTUNE OSV 172.23.1.2

# Data Queue

- Possibilité de « lire » et « écrire » !

Service	Type	Commentaire
QSYS2.CLEAR_DATA_QUEUE()	Procedure	Eq. QCLRDTAQ
QSYS2.DATA_QUEUE_ENTRIES()	UDTF	Eq. Retrieve Data Queue Message (QMHRDQM) API
QSYS2.DATA_QUEUE_INFO	Vue	Eq. Retrieve Data Queue Description (QMHQRDQD) API
QSYS2.RECEIVE_DATA_QUEUE()	UDTF	Eq. Receive Data Queue (QRCVDTAQ) API
QSYS2.SEND_DATA_QUEUE()	Procedure	Eq. Send Data Queue (QSNDDTAQ) API
QSYS2.SEND_DATA_QUEUE_BINARY()	Procedure	Eq. Send Data Queue (QSNDDTAQ) API
QSYS2.SEND_DATA_QUEUE_UTF8()	Procedure	Eq. Send Data Queue (QSNDDTAQ) API

# Data Queue

- Pour notre exemple

```
CRTDTAQ DTAQ(UNIV2023/NOKEY)  
        MAXLEN(20000)  
        TEXT('File 1 - sans clé')
```

```
CRTDTAQ DTAQ(UNIV2023/KEY)  
        MAXLEN(20000)  
        SEQ(*KEYED)  
        KEYLEN(10)  
        TEXT('File 2 - avec clé')
```

# QSYS2.DATA\_QUEUE\_INFO

- Donne la description

```
66 SELECT *
67 FROM QSYS2.DATA_QUEUE_INFO
68 WHERE DATA_QUEUE_LIBRARY = 'UNIV2023' ;
69
```

DATA_QUEUE_LIBRARY	DATA_QUEUE_NAME	DATA_QUEUE_TYPE	MAXIMUM_MESSAGE_LENGTH	SEQUENCE	KEY_LENGTH	INCLUDE_SENDER_ID	CURRENT_MESSAGES
UNIV2023	KEY	STANDARD	200	KEYED	10	NO	0
UNIV2023	NOKEY	STANDARD	200	FIFO	-	NO	0

- Autres attributs disponibles

# QSYS2.SEND\_DATA\_QUEUE

- Envoyer un message
  - Procédures
    - SEND\_DATA\_QUEUE
      - **La donnée est convertie dans le CCSID du job**
    - SEND\_DATA\_QUEUE\_BINARY
      - **La donnée est convertie en binary string**
    - SEND\_DATA\_QUEUE\_UTF8
      - **La donnée est convertie en UTF-8**
  - Support des files
    - Sans/avec clé
    - Locale/DDM (dans ce cas, écriture asynchrone ou synchrone au choix)



# QSYS2.SEND\_DATA\_QUEUE

- Exemple : file sans clé

```
-- Envoie message file sans clé
CALL QSYS2.SEND_DATA_QUEUE(MESSAGE_DATA => 'Message n°1',
                           DATA_QUEUE => 'NOKEY',
                           DATA_QUEUE_LIBRARY => 'UNIV2023' );

CALL QSYS2.SEND_DATA_QUEUE_BINARY(MESSAGE_DATA => cast('Message n°2 - binaire' as varchar(50) for bit data),
                                  DATA_QUEUE => 'NOKEY',
                                  DATA_QUEUE_LIBRARY => 'UNIV2023' );

CALL QSYS2.SEND_DATA_QUEUE_BINARY(MESSAGE_DATA => (select line
                                                    from table (qsys2.ifs_read_binary(path_name => '/home/NB/ibm-test.png' ))),
                                  DATA_QUEUE => 'NOKEY',
                                  DATA_QUEUE_LIBRARY => 'UNIV2023' );

CALL QSYS2.SEND_DATA_QUEUE_UTF8(MESSAGE_DATA => cast('Message n°3 - UTF8 - Chào buổi sáng !' as varchar(100) ccsid 1208),
                                DATA_QUEUE => 'NOKEY',
                                DATA_QUEUE_LIBRARY => 'UNIV2023' );
```

# QSYS2.SEND\_DATA\_QUEUE

- Exemple : file avec clé
  - La clé doit être exactement de la même longueur que déclarée à la création de la \*DTAQ

```
-- Envoie message file avec clé
CALL QSYS2.SEND_DATA_QUEUE(MESSAGE_DATA => 'Message n°1',
                           DATA_QUEUE => 'KEY',
                           DATA_QUEUE_LIBRARY => 'UNIV2023',
                           KEY_DATA => '          50' );

CALL QSYS2.SEND_DATA_QUEUE_BINARY(MESSAGE_DATA => cast('Message n°2 - binaire' as varchar(50) for bit data),
                                   DATA_QUEUE => 'KEY',
                                   DATA_QUEUE_LIBRARY => 'UNIV2023',
                                   KEY_DATA => cast('80' as char(10)));
```

- Sinon

```
CALL QSYS2.SEND_DATA_QUEUE_BINARY(MESSAGE_DATA => (select line from table (qsys2.
ifs_read_binary(path_name => '/home/NB/ibm-test.png' ))), DATA_QUEUE => 'KEY',
DATA_QUEUE_LIBRARY => 'UNIV2023', KEY_DATA => '80' )
✘ Etat SQL : 22023
Code fournisseur : -443
Message : [SQL0443] KEY_DATA MUST HAVE A LENGTH OF 10 Cause . . . . . : Un
programme de déclenchement, une procédure externe ou une fonction externe a détecté
```

# QSYS2.DATA\_QUEUE\_ENTRIES

- Permet de lire le contenu d'une \*DTAQ
  - Avec ou sans clé

```
111 -- Lecture sans clé
112 SELECT *
113 FROM TABLE(QSYS2.DATA_QUEUE_ENTRIES (DATA_QUEUE => 'NOKEY',
114                                         DATA_QUEUE_LIBRARY => 'UNIV2023'))
115 ORDER BY ORDINAL_POSITION;
```

ORDINAL_POSITION	DATA_QUEUE_LIBRARY	DATA_QUEUE	MESSAGE_DATA	MESSAGE_DATA_UTF8	MESSAGE_DATA_BINARY
1	UNIV2023	NOKEY	Message n°1	␣␣␣␣	D485A2A281878540954AF1
2	UNIV2023	NOKEY	Message n°2 - binaire	␣␣␣␣␣␣	D485A2A281878540954AF24060
3	UNIV2023	NOKEY	is+â␣␣␣ñ\@êø␣␣␣␣/ô␣Ëâñ}␣␣... ␣␣IHDR␣␣␣␣pa?OsBIT␣␣␣␣ OdpHYs␣␣␣␣...		89504E470D0A1A0A0000000D49
4	UNIV2023	NOKEY	(ÁËË/ÁÁ␣>B␣␣␣␣i}ã␣␣␣␣äÇ/ö... Message n°3 - UTF8 - Chào buổi sáng !		4D657373616765206EC2B03320

- La valeur est exprimée
  - MESSAGE\_DATA : CCSID du job
  - MESSAGE\_DATA\_UTF8
  - MESSAGE\_DATA\_BINARY

# QSYS2.DATA\_QUEUE\_ENTRIES

- Le paramètre SELECTION\_TYPE permet une sélection sur le rang

```

116 -- première valeur
117 SELECT *
118 FROM TABLE(QSYS2.DATA_QUEUE_ENTRIES(DATA_QUEUE => 'NOKEY',
119                                     DATA_QUEUE_LIBRARY => 'UNIV2023',
120                                     SELECTION_TYPE => 'FIRST')) -- LAST, ALL, FIRST, REVERSE
121 ORDER BY ORDINAL_POSITION;

```

ORDINAL_POSITION	DATA_QUEUE_LIBRARY	DATA_QUEUE	MESSAGE_DATA	MESSAGE_DATA_UTF8	MESSAGE_DATA_BINARY
1	UNIV2023	NOKEY	Message n°1	☐☐☐☐	D485A2A281878540954AF1

# QSYS2.DATA\_QUEUE\_ENTRIES

- La combinaison de SELECTION\_TYPE, KEY\_DATA et KEY\_ORDER permet une sélection sur clé

```

129 SELECT *
130 FROM TABLE(QSYS2.DATA_QUEUE_ENTRIES (DATA_QUEUE => 'KEY',
131                                     DATA_QUEUE_LIBRARY => 'UNIV2023',
132                                     SELECTION_TYPE => 'KEY',
133                                     KEY_DATA => '      51',
134                                     KEY_ORDER => 'GE'))
135 ORDER BY ORDINAL_POSITION;
136

```

ORDINAL_POSITION	DATA_QUEUE_LIBRARY	DATA_QUEUE	MESSAGE_DATA	MESSAGE_DATA_UTF8	MESSAGE_DATA_BINARY	KEY_DATA
1	UNIV2023	KEY	i&â□□□ñ\@é00□□□□/ô□Éâ... □...		89504E470D0A1A0A0000...	81
2	UNIV2023	KEY	Message n°2 - binaire	~□□□□□□	D485A2A281878540954A...	80

# QSYS2.RECEIVE\_DATA\_QUEUE

- Permet la lecture
  - Avec ou sans clé
  - La suppression (au choix)
  - Un temps d'attente si la file est vide

```

-- Réception des messages
SELECT *
FROM TABLE(QSYS2.RECEIVE_DATA_QUEUE (DATA_QUEUE => 'NOKEY',
                                       DATA_QUEUE_LIBRARY => 'UNIV2023',
                                       REMOVE => 'NO', -- YES/NO
                                       WAIT_TIME => -1 -- <0 : infini, 0 : immédiat, >0 : délai en seconde
                                       ));

```

MESSAGE_DATA	MESSAGE_DATA_UTF8	MESSAGE_DATA_BINARY	KEY_DATA	SENDER_JOB_NAME	SENDER_CURRENT_USER
Message n°1	␣␣␣␣	D485A2A281878540954AF1	-	-	-

# QSYS2.RECEIVE\_DATA\_QUEUE

- Avec clé

```
SELECT *
FROM TABLE(QSYS2.RECEIVE_DATA_QUEUE (DATA_QUEUE => 'KEY',
                                       DATA_QUEUE_LIBRARY => 'UNIV2023',
                                       REMOVE => 'NO', -- YES/NO
                                       WAIT_TIME => -1, -- <0 : infini, 0 : immédiat, >0 : délai en seconde
                                       KEY_DATA => '      51',
                                       KEY_ORDER => 'GE'));
```

MESSAGE_DATA	MESSAGE_DATA_UTF8	MESSAGE_DATA_BINARY	KEY_DATA	SENDER_JOB_NAME	SENDER_CURRENT_USER
i&+â□□□\@é0□□□□/8□Éâf}□□□□à□À... □...		89504E470D0A1A0A00000000...	81	-	-

# QSYS2.CLEAR\_DATA\_QUEUE

- Avec ou sans clé

```
CALL QSYS2.CLEAR_DATA_QUEUE('NOKEY', 'UNIV2023');
CALL QSYS2.CLEAR_DATA_QUEUE('KEY', 'UNIV2023');
CALL QSYS2.CLEAR_DATA_QUEUE(DATA_QUEUE => 'KEY',
                             DATA_QUEUE_LIBRARY => 'UNIV2023',
                             KEY_DATA => '          51',
                             KEY_ORDER => 'GE') ;
```



# Data Queue

- Quel intérêt ?
  - Les Data Queue se manipulent exclusivement avec des APIs
    - En CL : CRTDTAQ, DLTDTAQ, WRKDTAQ

- Les appels d'APIs sont lourds à coder !
  - Il faut lire la documentation (!)
  - La comprendre (!!)

## Receive Data Queue (QRCVDTAQ) API

### Required Parameter Group:

1	Data queue name	Input	Char(10)
2	Library name	Input	Char(10)
3	Length of data	Output	Packed(5,0)
4	Data	Output	Char(*)
5	Wait time	Input	Packed(5,0)

### Optional Parameter Group 1:

6	Key order	Input	Char(2)
7	Length of key data	Input	Packed(3,0)
8	Key data	I/O	Char(*)
9	Length of sender information	Input	Packed(3,0)
10	Sender information	Output	Char(*)

### Optional Parameter Group 2:

11	Remove message	Input	Char(10)
12	Size of data receiver	Input	Packed(5,0)
13	Error code	I/O	Char(*)

### Optional Parameter Group 3:

14	Wait time units	Input	Char(10)
----	-----------------	-------	----------

Default Public Authority: \*USE

Threadsafe: Conditional; see [Usage Notes](#).

# Data Queue

- Vous allez obtenir ce genre de code
  - Déclaration uniquement ici

```

//=====
// IBM API to Receive an entry from a Data Queue
// https://www.ibm.com/support/knowledgecenter/en/ssw_ibm_i_75/apis/qrcvdtaq.htm
//=====
dcl-pr IBMAPI_RecvFromDtaQ  extpgm('QRCVDTAQ');
  DtaQName          char(10)  const;
  DtaQLib           char(10)  const;
  DtaQBufLen        packed(5:0);
  DtaQBuffer        char(65535)  options(*varsize);
  DtaQWait          packed(5:0) const;
// optional parm group 1
  DtaQKeyOrd        char(2)    const  options(*nopass);
  DtaQKeyLen        packed(3:0) const  options(*nopass);
  DtaQKeyData       char(256)  const  options(*nopass:*varsize);
  DtaQSndLen        packed(3:0) const  options(*nopass);
  DtaQSndInf        char(44)   const  options(*nopass:*varsize);
// optional parm group 2
  DtaQRemove        char(10)   const  options(*nopass);
  DtaQBsize         packed(5:0) const  options(*nopass);
  DtaQErrC          liked(ApiErrC)  options(*nopass:*varsize);
end-pr;

```

## ■ Ou en CL

```
/* -- paramètres pour QNSDDTAQ */
/* Requis */
DCL      VAR(&DTAQNAM) TYPE(*CHAR) LEN(10) VALUE('MADATQ')
DCL      VAR(&DTAQLIB) TYPE(*CHAR) LEN(10) VALUE('MABIB')
DCL      VAR(&DATALEN) TYPE(*DEC) LEN(5 0) VALUE(0)
DCL      VAR(&DATA) TYPE(*CHAR) LEN(256) VALUE(' ')
DCL      VAR(&WAIT) TYPE(*DEC) LEN(5 0) VALUE(-1) /* attente */
/* Groupe optionnel 1 */
DCL      VAR(&KEYORDR) TYPE(*CHAR) LEN(2) VALUE(' ')
/* Pas de clé */
DCL      VAR(&KEYDTALEN) TYPE(*DEC) LEN(3 0) VALUE(0)
DCL      VAR(&KEYYDTA) TYPE(*CHAR) LEN(1) VALUE(' ')
DCL      VAR(&SNDRIDLLEN) TYPE(*DEC) LEN(3 0) VALUE(44)
DCL      VAR(&SNDRID) TYPE(*CHAR) LEN(44) VALUE(' ')
/* Groupe optionnel 2 */
DCL      VAR(&RMVMSG) TYPE(*CHAR) LEN(10) VALUE('*YES ')
DCL      VAR(&DTARCVLEN) TYPE(*DEC) LEN(5 0) VALUE(256)
DCL      VAR(&ERRORCODE) TYPE(*CHAR) LEN(256) VALUE(' ')
/* -- découpage ERRORCODE */
DCL      VAR(&MSGID) TYPE(*CHAR) LEN(7)
```

```
/* Traitement */
BOUCLE   : CHGVAR %BIN(&ERRORCODE 1 4) 256
CALL     PGM(QRCVDTAQ) PARM(&DTAQNAM &DTAQLIB &DATALEN &DATA &WAIT +
&KEYORDR &KEYDTALEN &KEYYDTA &SNDRIDLLEN &SNDRID &RMVMSG &DTARCVLEN +
&ERRORCODE )
/* Test anomalie */
IF       COND(%BIN(&ERRORCODE 5 4) *GT 0) THEN(DO)
  CHGVAR &MSGID %SST(&ERRORCODE 9 7)
  CHGVAR VAR(&MSG) VALUE('Anomalie à la lecture : ' !! &MSGID)
ENDDO
ELSE     CMD(DO)
/* extraction message */
CHGVAR   &MSG VALUE('Message : ' !! %SST( &DATA 1 &DATALEN))
/* extraction message */
CHGVAR   VAR(&MSG) VALUE(&MSG *TCAT ' . SenderID : ' !! %SST(&SNDRID +
9 10) ' Profil en cours : ' !! %SST(&SNDRID 35 10))
SNDSMSG  MSG(&MSG) TOUSR(BERTHOIN)
ENDDO
```

# Fichiers Spool

- Nombreux services !

Spool Services					
<a href="#">QSYS2.OUTPUT_QUEUE_ENTRIES()</a>	Table function	Base Enhanced: SF99950 Level 3	Base Enhanced: SF99704 Level 23	Base	SF99702 Level 9
<a href="#">QSYS2.OUTPUT_QUEUE_ENTRIES</a>	View	Base Enhanced: SF99950 Level 3	Base Enhanced: SF99704 Level 23	Base	SF99702 Level 9
<a href="#">QSYS2.OUTPUT_QUEUE_ENTRIES_BASIC</a>	View	Base	Base	SF99703 Level 11	SF99702 Level 23
<a href="#">QSYS2.OUTPUT_QUEUE_INFO</a>	View	Base Enhanced: SF99950 Level 3	Base Enhanced: SF99704 Level 23	Base	SF99702 Level 11
<a href="#">QSYS2.SPOOLED_FILE_INFO()</a>	Table function	Base	SF99704 Level 20	SF99703 Level 28	Not Supported
<a href="#">SYSTOOLS.DELETE_OLD_SPOOLED_FILES()</a>	Procedure	Base	SF99704 Level 7	SF99703 Level 18	Not Supported
<a href="#">SYSTOOLS.GENERATE_PDF()</a>	Scalar function	Base	SF99704 Level 13 Enhanced: SF99704 Level 20	SF99703 Level 24 Enhanced: SF99703 Level 28	Not Supported
<a href="#">SYSTOOLS.SPOOLED_FILE_DATA()</a>	Table function	Base	Base	SF99703 Level 15	Not Supported

# Fichiers Spool

- \*OUTQ contenant le plus de fichiers spools

```

174 -- Infos sur les *OUTQ
175 SELECT * FROM QSYS2.OUTPUT_QUEUE_INFO order by number_of_files desc ;
176

```

OUTPUT_QUEUE_NAME	OUTPUT_QUEUE_LIBRARY_NAME	NUMBER_OF_FILES	NUMBER_OF_WRITERS
QPRINT	QGPL	1574	0
QEZJOBLOG	QUSRSYS	369	0
GGITLOG	GITIT	14	0
QEZDEBUG	QUSRSYS	12	0

# Fichiers Spool

- QSYS2.OUTPUT\_QUEUE\_ENTRIES donne les infos sur un spool

```

179 -- Top 5 des profils consommant le plus de place en fichiers spools, spools des 30 derniers jours
180 select user_name, sum(size) as taille
181 from QSYS2.OUTPUT_QUEUE_ENTRIES
182 where create_timestamp >= current timestamp - 30 days
183 group by user_name
184 order by taille desc
185 limit 5;

```

USER_NAME	TAILLE
PLB	103952
VY	42796
ITOP	19392
LA	9056
QSECOFR	7952

# Fichiers Spool

- Conversion en PDF : fonction scalaire GENERATE\_PDF
  - Nécessite 5770TS1 - Option 1 - Transform Services - AFP to PDF Transform
  - Renvoie 1 si OK, -1 en cas d'erreur

```

187 -- Conversion en PDF
188 VALUES SYSTOOLS.GENERATE_PDF (
189             JOB_NAME           => '447520/QSECOFR/SAVALLUSR',
190             SPOOLED_FILE_NAME  => 'QPJOBLOG',
191             SPOOLED_FILE_NUMBER => 1,
192             PATH_NAME          => '/home/nb/SAVALLUSR.pdf');

```

00001 /home/NB sur neptune - IP

Fichier Edition Affichage Actions

neptune - IP SAVALLUSR.pdf X Inclusion

Répertoire  
/home/NB

Icône	Nom	Taille (Ko)	Dernière modification
	SAVALLUSR.pdf		8 novembre 2023 à 15:21:40 UTC+1

# Fichiers Spool

- Convertir en PDF tous les spools d'une \*OUTQ à l'état RDY

```
195 with spool_list as (  
196   select spooled_file_name, file_number, job_name  
197   from QSYS2.OUTPUT_QUEUE_ENTRIES_BASIC  
198   where output_queue_name = 'GGITLOG' and output_queue_library_name = 'GITIT' and status = 'READY' )  
199 select spl.*,  
200        SYSTOOLS.GENERATE_PDF(JOB_NAME           => spl.job_name,  
201                             SPOOLED_FILE_NAME => spl.spooled_file_name,  
202                             SPOOLED_FILE_NUMBER => spl.file_number,  
203                             PATH_NAME         => '/home/nb/pdf/' || replace(spl.job_name, '/', '-') || '-' ||  
204                             spl.spooled_file_name || '-' || trim(char(spl.file_number)) || '.pdf')  
205 from spool_list spl ;
```

SPOOLED_FILE_NAME	FILE_NUMBER	JOB_NAME	00004
QPRINT	1	414955/VY/GGITAUTH	1
QPRINT	1	414958/VY/GGITCLONE	1
QPRINT	1	415000/VY/ISGITREPO	1
QPRINT	1	415003/VY/GGITAUTH	1

File explorer window showing the directory /home/NB/pdf sur neptune - IP. The window displays a list of PDF files with columns for Icon, Nom, Taille (Ko), and Dernière modification.

Icon	Nom	Taille (Ko)	Dernière modification
[Icon]	414955-VY-GGITAUTH-QPRINT-1.pdf		13 novembre 2023 à 15:32:08 UTC+1
[Icon]	414958-VY-GGITCLONE-QPRINT-1.pdf		13 novembre 2023 à 15:32:09 UTC+1
[Icon]	415000-VY-ISGITREPO-QPRINT-1.pdf		13 novembre 2023 à 15:32:09 UTC+1
[Icon]	415003-VY-GGITAUTH-QPRINT-1.pdf		13 novembre 2023 à 15:32:10 UTC+1
[Icon]	415006-VY-GGITPUSH-QPRINT-1.pdf		13 novembre 2023 à 15:32:10 UTC+1
[Icon]	433164-PO-ISGITREPO-QPRINT-1.pdf		13 novembre 2023 à 15:32:10 UTC+1
[Icon]	433167-PO-GGITCHGRH-QPRINT-1.pdf		13 novembre 2023 à 15:32:11 UTC+1
[Icon]	433172-PO-ISGITREPO-QPRINT-1.pdf		13 novembre 2023 à 15:32:11 UTC+1
[Icon]	433177-PO-GGITPULL-QPRINT-1.pdf		13 novembre 2023 à 15:32:11 UTC+1
[Icon]	433189-PO-ISGITREPO-QPRINT-1.pdf		13 novembre 2023 à 15:32:12 UTC+1
[Icon]	433192-PO-GGITCHGRH-QPRINT-1.pdf		13 novembre 2023 à 15:32:12 UTC+1
[Icon]	433197-PO-ISGITREPO-QPRINT-1.pdf		13 novembre 2023 à 15:32:12 UTC+1
[Icon]	433202-PO-GGITPULL-QPRINT-1.pdf		13 novembre 2023 à 15:32:13 UTC+1
[Icon]	446586-PO-ISPATH-QPRINT-1.pdf		13 novembre 2023 à 15:32:13 UTC+1



# Fichiers Spool

- De nombreux exemples très intéressants dans ACS

The screenshot shows the 'Run SQL Scripts - neptune - IP(Neptune)' application. The 'Edition' menu is open, and the 'Exemples' option is selected. A sub-menu titled 'SQL Exemples' is displayed, listing various SQL scripts. A red arrow points to the 'Insertion à partir d'exemples...' option in this sub-menu. The main editor area shows two SQL queries related to spool management.

**Menu Items:**

- Annulation (Ctrl+Z)
- Répétition (Ctrl+Y)
- Découpage (Ctrl+X)
- Copie (Ctrl+C)
- Collage (Ctrl+V)
- Sélection globale (Ctrl+A)
- Outil de formatage SQL
- Suppression de ligne(s) (Ctrl+D)
- Basculement commentaires (Ctrl+7)
- Content Assist
- Exemples**
  - Insertion à partir d'exemples... (Ctrl+I)
  - Insertion d'instructions SQL générées... (Ctrl+Maj+G)
  - Invite instruction... (F4)

**SQL Exemples List:**

- spool
- Boucle d'exécution de commandes système
- Return Work Management Class info
- Spool - Consume my most recent spooled file
- Spool - Generate PDF into the IFS
- Spool - Output queue basic detail
- Spool - Output queue exploration
- Spool - Search all QZDASOINIT spooled files
- Spool - Top 10 consumers of spool storage
- Spool - managing spool
- Storage - Top 10 Spool consumers, by user
- Work Management - Job Descriptions

**SQL Code Snippets:**

```
-- category: Personnalisation
-- description: Boucle d'exécution de commandes syst
-- Sélection de tous les spoules de l'utilisateur
select *
FROM QSYS2.OUTPUT_QUEUE_ENTRIES_BASIC
where USER_NAME = 'NB' ;

stop;

-- Constitution de la commande HLDSPFLF pour chaque sp
SELECT 'hldspflf FILE(' concat SPOOLNAME concat ' ) JO
FROM QSYS2.OUTPUT_QUEUE_ENTRIES_BASIC
where USER_NAME = 'NB' ;

stop;
```

# Exécution de commande

- API QCMDEXC disponible
  - En procédure
    - Utile pour lancer l'exécution d'une commande
  - En fonction scalaire
    - Utile pour utiliser QCMDEXC dans une requête

QSYS2.QCMDEXC()	Procedure	Base	Base	Base	Base
QSYS2.QCMDEXC()	Scalar function	Base	SF99704 Level 13	SF99703 Level 24	Not Supported

# Exécution de commande

- Procédure

- Création d'un répertoire

```
call QSYS2.QCMDEXC('crtdir ''/home/TEAM80''') ;
```

- Fonction scalaire

- Création de x répertoires et changement de propriétaire

```
select AUTHORIZATION_NAME,
       QSYS2.QCMDEXC('crtdir ''/home/' || trim(AUTHORIZATION_NAME) || ''') ,|
       QSYS2.QCMDEXC('CHGOWN OBJ(''/home/' || trim(AUTHORIZATION_NAME) || ''') NEWOWN(' || trim(AUTHORIZATION_NAME) || ''')')
from QSYS2.USER_INFO_BASIC
where AUTHORIZATION_NAME between 'TEAM70' and 'TEAM72' ;
```

# Intégration dans vos développements

- Via une instruction SQL embarquée !
  - Rien que vous ne sachiez déjà
  
- Supports
  - RPGLE/CBLLE : complet
    - Appel de fonctions scalaires, fonctions tables, procédures, accès aux tables et vues
    - Gestion des erreurs (SQLCODE/SQLSTATE)
  - CL : incomplet
    - Impossible de lire un result set directement (limite l'usage des fonctions tables, tables et vues)
    - Pas d'accès au SQLCODE/SQLSTATE
  - SQL : natif
    - Non traité ici

# Intégration dans vos développements

- Exemple RPG

- Pensez à la compatibilité des types SQL – RPGLE

- <https://www.ibm.com/docs/en/i/7.5?topic=applications-determining-equivalent-sql-ile-rpg-data-types>

Character	<b>Free-form:</b> – CHAR(n)	452	n	CHAR (n) where n is the length
	<b>Fixed-form:</b> – Data type = A or blank, decimal positions blank, length between 1 and 32766.			
Character varying length greater than 254	<b>Free-form:</b> – VARCHAR(n)	448	n	VARCHAR (n) where n is the length
	<b>Fixed-form:</b> – Data type = A or blank, decimal positions blank, VARYING keyword on Definition specification or format *VAR on Input specification.			

# Intégration dans vos développements

- Pensez à utiliser SQLTYPE et LOB
  - <https://www.ibm.com/docs/en/i/7.5?topic=area-sqltype-sqllen>
  - <https://www.ibm.com/docs/en/i/7.5?topic=dlhviiratus-lob-host-variables-in-ile-rpg-applications-that-use-sql>

```
dcl-proc getUsers export ;
dcl-pi *n;
  data_LENGTH    int(10) ;
  data           char(16700000) ;
  httpStatus     int(10) ;
end-pi;

dcl-s fileCLOB sqltype( CLOB : 16700000 ) ;

// RAZ sortie
clear data ;
clear data_LENGTH ;
clear httpStatus ;

exec sql select xmlserialize(
                xmlgroup(
                  authorization_name,
                  special_authorities
                  option root users row user )
                as clob(1000))
into :fileCLOB
from qsys2.user_info
where left(authorization_name, 1) <> 'Q' and
status = '*DISABLED';
```

# Intégration dans vos développements

## ■ Exemple CL

- Principalement avec RUNSQL

```

/* Créer le script */
chgvar      &cmd 'system ''CLRLIB LIB(GENVOISSH)''''
RUNSQL     SQL('call qsys2.ifs_write( path_name => ''' *tcat +
           &SCRIPT *tcat ''', LINE => ''' *tcat &cmd *tcat +
           ''', OVERWRITE => 'REPLACE'')) COMMIT(*NONE)

chgvar      &cmd 'system ''RSTLIB SAVLIB(GENVOII) DEV(*SAVF) +
           SAVF(LIBSAVF/GENVOII) RSTLIB(GENVOISSH)''''
RUNSQL     SQL('call qsys2.ifs_write( path_name => ''' *tcat +
           &SCRIPT *tcat ''', LINE => ''' *tcat &cmd *tcat +
           ''', OVERWRITE => 'APPEND'')) COMMIT(*NONE)

chgvar      &cmd 'system ''GRTOBJAUT OBJ(GENVOISSH/*ALL) +
           OBJTYPE(*ALL) USER(*PUBLIC) AUT(*USE) +
           REPLACE(*YES)''''
RUNSQL     SQL('call qsys2.ifs_write( path_name => ''' *tcat +
           &SCRIPT *tcat ''', LINE => ''' *tcat &cmd *tcat +
           ''', OVERWRITE => 'APPEND'')) COMMIT(*NONE)

chgvar      &cmd 'rm /home/genvoissh.log'
RUNSQL     SQL('call qsys2.ifs_write( path_name => ''' *tcat +
           &SCRIPT *tcat ''', LINE => ''' *tcat &cmd *tcat +
           ''', OVERWRITE => 'APPEND'')) COMMIT(*NONE)

chgvar      &cmd 'ls -ail /QSYS.LIB/GENVOI.LIB/ > +
           /home/genvoissh.log'
RUNSQL     SQL('call qsys2.ifs_write( path_name => ''' *tcat +
           &SCRIPT *tcat ''', LINE => ''' *tcat &cmd *tcat +
           ''', OVERWRITE => 'APPEND'')) COMMIT(*NONE)

```

```

DCL      VAR(&CMD) TYPE(*CHAR) LEN(500)
DCL      VAR(&SCRIPT) TYPE(*CHAR) LEN(500) +
           value('/home/NB/ssh/scriptsqsl.sh')

ADDENVVAR ENVVAR(QIBM_QSH_CMD_OUTPUT) +
           VALUE('FILEAPPEND=/tmp/sshlog.txt') REPLACE(*YES)
ADDENVVAR ENVVAR(QIBM_QSH_CMD_ESCAPE_MSG) VALUE(Y) +
           REPLACE(*YES)

```

```

/* Lancer le script */
chgvar      &cmd value('PATH=$PATH:/QOpenSys/usr/bin && cat ' +
           *bcat &script *bcat ' | ssh -T sshuranus@uranus')
QSH         CMD(&cmd)
MONMSG     MSGID(QSH0000) EXEC(DO)
           SNDMSG      MSG('Erreur !') TOUSR(nb)
ENDDO

```

Université **IBM i**

7 novembre 2023

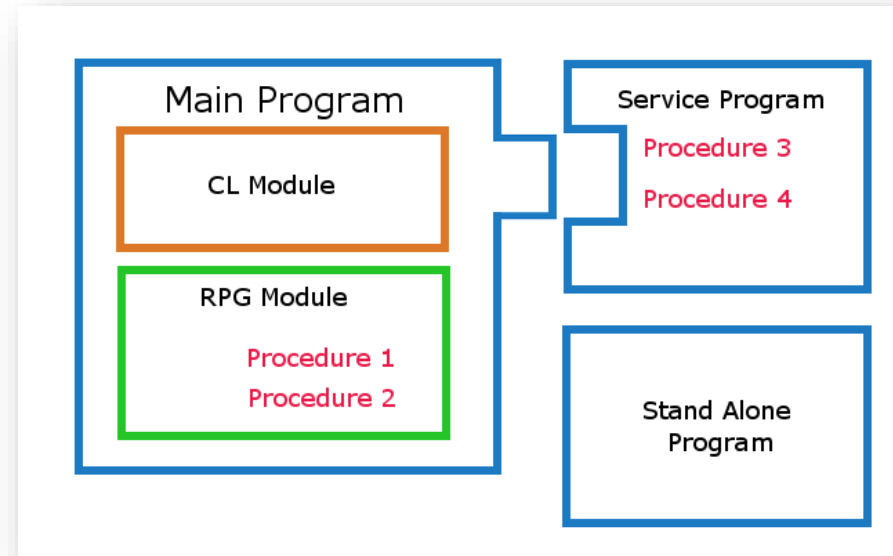
Let's  
Create

# 3. Exemples de services pour le développeur



# Référentiel dynamique ?

- Possibilité d'utiliser les services IBM i pour effectuer des analyses d'impact
  - Principalement lorsque vous travailler en ILE
    - 1 \*PGM = 1..n \*MODULE
    - 1 \*SRVPGM = 1..n \*MODULE
    - 1 \*SRVPGM = 1..n signature(s)
    - 1 \*MODULE = 1..n procédure(s)
    - 1 \*BNDDIR = 1..n \*MODULE/\*SRVPGM
  
- En mode natif
  - DSPMOD
  - DSPPGM
  - DSPSRVPGM
  - Commandes BNDDIR (CRT/ADD/...)



# Référentiel dynamique ?

- Services IBM i

Service	Type	Description
QSYS2.BINDING_DIRECTORY_INFO	Vue	Contenu de chaque répertoire de liage
QSYS2.BOUND_MODULE_INFO	Vue	Module constitutifs de chaque programme (liage par copie)
QSYS2.BOUND_SRVPGM_INFO	Vue	Programme de service lié à chaque programme/programme de service (liage par référence)
QSYS2.PROGRAM_EXPORT_IMPORT_INFO	Vue	Procédures importées/exportées
QSYS2.PROGRAM_INFO	Vue	Attributs et signatures

# Référentiel dynamique ?

- Quel usage ?
  - Nous avons tous des outils pour maintenir des références croisées et nous aider dans les analyses d'impacts de nos développements
  - Soit vous avez des outils du commerce
  - Soit vous utilisez vos propres outils
- Souvent, les références croisées sont calculées en batch la nuit
  - DSPPGMREF, DSPDBR ...
- **Avec SQL vous pouvez avoir cela en dynamique**
  - **Pour une recherche spécifique** (ne pas construire dynamiquement un référentiel à chaque recherche !)
- Vous pourrez également simplifier l'écriture et la maintenance de vos programmes de constitutions des références croisées

# Référentiel dynamique ?

- Exemple : contenu des répertoires de liage

```
218 -- Liste des modules dans chaque répertoire de liage
219 select * from QSYS2.BINDING_DIRECTORY_INFO where entry_type = '*MODULE';
```

BINDING_DIRECTORY_LIBRARY	BINDING_DIRECTORY	ENTRY_LIBRARY	ENTRY	ENTRY_TYPE	ENTRY_ACTIVATION	ENTRY_CREATE_TIMESTAMP
ARCAD_SMP	BNDDIR001	*LIBL	MOD004	*MODULE	-	-
CCHAPUIS	CMNAPP	*LIBL	CMNAPP000	*MODULE	-	-
EXERCICES	L505BNDDIR	EXERCICES	L505008M	*MODULE	-	2003-04-23 17:30:32
GWEBSER	WS_PGM	GWEBSER	WS_PGM	*MODULE	-	2020-10-28 11:45:22
GWEBSERD	WS_PGM	GWEBSERD	WS_PGM	*MODULE	-	2020-11-01 12:32:15
GWEBSERP	WS_PGM	GWEBSERP	WS_PGM	*MODULE	-	2020-11-18 09:08:29
JPLTOOLS	JPLTOOLS	JPLTOOLS	JP4GPS_R	*MODULE	-	2022-06-09 21:15:37
JPLTOOLS	JPLTOOLS	JPLTOOLS	CLOB1_M	*MODULE	-	2022-06-09 21:15:48

# Référentiel dynamique ?

- Exemple : contenu des répertoires de liage

```

222 -- Module apparaissant > 1 fois dans les répertoires de liage
223 select entry_library, entry , count(*) as nb
224 from QSYS2.BINDING_DIRECTORY_INFO
225 where entry_type = '*MODULE'
226 group by entry_library, entry
227 having count(*) > 1
228 order by nb desc ;

```

ENTRY_LIBRARY	ENTRY	NB
*LIBL	PGM2A	2
*LIBL	PGM2B	2

- Un module apparaissant dans plusieurs répertoires de liage risque d'être copié dans plusieurs \*PGM/\*SRVPGM

# Référentiel dynamique ?

- Exemple : module lié par copie dans plusieurs objets

```

230 -- Module copié dans plusieurs programmes / programmes de services
231 select bound_module_library, bound_module, count(*) as nb
232 from QSYS2.BOUND_MODULE_INFO
233 where program_library in ('NB', 'GG', 'JL', 'GEIM3')
234 group by bound_module_library, bound_module
235 having count(*) > 1
236 order by count(*) desc ;
237

```

BOUND_MODULE_LIBRARY	BOUND_MODULE	NB
NB	WSC2	4
NB	WSC3	4
NB	WSC4	4
NB	WSR1	4
NB	WSC0	4
NB	WSR2	4
NB	WSC1	4
NB	FLTWSR1	3
NB	FLTWSC1	3

# Référentiel dynamique ?

- Exemple : quels sont les objets contenant les modules dupliqués

```

239 with
240 module_lib as (
241     select PROGRAM_LIBRARY, PROGRAM_NAME, OBJECT_TYPE, BOUND_MODULE_LIBRARY, BOUND_MODULE
242     from QSYS2.BOUND_MODULE_INFO
243     where program_library in ('NB', 'GG', 'JL', 'GEIM3') ),
244 module_double as (
245     select bound_module_library, bound_module, count(*) as nb
246     from module_lib
247     group by bound_module_library, bound_module
248     having count(*) > 1 )
249 select i.PROGRAM_LIBRARY, i.PROGRAM_NAME, i.OBJECT_TYPE, m.BOUND_MODULE_LIBRARY, m.BOUND_MODULE
250 from module_double m
251 join module_lib i on (m.bound_module_library, m.bound_module) = (i.bound_module_library, i.bound_module)
252 order by 4, 5, 1, 2, 3;
253

```

PROGRAM_LIBRARY	PROGRAM_NAME	OBJECT_TYPE	BOUND_MODULE_LIBRARY	BOUND_MODULE
GG	STRG01	*PGM	GG	STRG01
GG	STRG02	*PGM	GG	STRG01
JL	SRVPGM	*SRVPGM	JL	SRVCHAR
JL	UTILPGMSRV	*PGM	JL	SRVCHAR
NB	AXISDATE3	*PGM	NB	AXISTOOLS
NB	CLIENT	*PGM	NB	AXISTOOLS
NB	DATE3	*PGM	NB	DATEUTIL

# Référentiel dynamique ?

- Exemple : où est la procédure EMPLOYE\_CREATE ?

```

255 -- Trouver le(s) programme(s) de service qui fournit la procédure EMPLOYE_CREATE
256 select * from QSYS2.PROGRAM_EXPORT_IMPORT_INFO
257 where program_library in ('NB', 'GG', 'JL', 'GEIM3', 'ILEASTIC')
258     and
259     upper(symbol_name) = 'EMPLOYE_CREATE' ;
260

```

PROGRAM_LIBRARY	PROGRAM_NAME	OBJECT_TYPE	SYMBOL_NAME	SYMBOL_USAGE	ARGUMENT_OPTIMIZATION	DATA_ITEM_SIZE
GG	EMPLOYE	*SRVPGM	EMPLOYE_CREATE	*PROCEXP	*NO	-



# Référentiel dynamique ?

- Exemple : quels sont les modules constitutifs de \*PGM/\*SRVPGM qui n'ont pas été recompilés suite à modification du source ?

```
263 select trim(PROGRAM_LIBRARY) || '/' || trim(PROGRAM_NAME),
264        OBJECT_TYPE,
265        trim(BOUND_MODULE_LIBRARY) || '/' || trim(BOUND_MODULE),
266        trim(SOURCE_FILE_LIBRARY) || '/' || trim(SOURCE_FILE) || '(' || trim(SOURCE_FILE_MEMBER) || ')',
267        SOURCE_CHANGE_TIMESTAMP,
268        '<',
269        s.last_change_timestamp
270 from QSYS2.BOUND_MODULE_INFO m
271 join QSYS2.SYSPARTITIONSTAT s on m.SOURCE_FILE_LIBRARY = s.table_schema and
272                                m.SOURCE_FILE = s.table_name and
273                                m.SOURCE_FILE_MEMBER = s.table_partition and
274                                m.SOURCE_CHANGE_TIMESTAMP < s.last_change_timestamp
275 where program_library in ('NB', 'GG', 'JL', 'GEIM3')
276 order by s.last_change_timestamp - SOURCE_CHANGE_TIMESTAMP desc ;
```

00001	OBJECT_TYPE	00003	00004	SOURCE_CHANGE_TIMESTAMP	00006	LAST_CHANGE_TIMESTAMP
NB/DEMO	*PGM	QTEMP/DEMO	NB/QRPGLESRC (DEMO)	2016-04-27 09:54:26	<	2023-05-11 16:52:19.000000
NB/BL	*PGM	QTEMP/BL	NB/QRPGLESRC (BL)	2018-01-25 21:38:12	<	2023-05-11 15:28:33.000000
NB/INDIC	*PGM	QTEMP/INDIC	NB/QRPGLESRC (INDIC)	2014-11-05 09:52:58	<	2020-01-07 12:19:52.000000
NB/DS	*PGM	QTEMP/DS	NB/QRPGLESRC (DS)	2015-01-16 17:43:26	<	2020-01-07 12:19:52.000000
NB/ZDA_INIT	*PGM	QTEMP/ZDA_INIT	NB/QRPGLESRC (ZDA_INIT)	2015-02-13 14:04:41	<	2020-01-07 12:19:52.000000
NB/LIBDATECHA	*PGM	QTEMP/LIBDATECHA	NB/QRPGLESRC (LIBDATECHA)	2015-02-23 11:13:00	<	2020-01-07 12:19:52.000000

# Référentiel dynamique ?

- Ce n'est pas une nouveauté, mais le catalogue DB2 contient l'intégralité de la description de la base de données
  - SYSTABLES, SYSALIAS, SYSTABLESTAT
  - SYSINDEXES
  - SYSVIEWS, SYSVIEWDEP
  - SYSCOLUMNS
  - SYSCST, SYSCSTCOL, SYSCSTDEP
  - SKEYCST, SYSKEYS
  - ...

# Référentiel dynamique ?

- Exemple :
  - Vue <-> Table/Vue

```

284 select * from qsys2.sysviews where system_view_schema = 'JL';
285 select VIEW_NAME,
286        VIEW_SCHEMA,
287        'Vue <> utilisé',
288        OBJECT_NAME,
289        OBJECT_SCHEMA,
290        OBJECT_TYPE
291 from qsys2.sysviewdep where system_view_schema = 'JL';

```

VIEW_NAME	VIEW_SCHEMA	00003	OBJECT_NAME	OBJECT_SCHEMA	OBJECT_TYPE
EXPDISPO	JL	Vue <> utilisé	SUPSP	JL	TABLE
EXPDISPO	JL	Vue <> utilisé	NBEXP	JL	VIEW
EXPDISPO	JL	Vue <> utilisé	LOCSP	JL	TABLE
NBEXP	JL	Vue <> utilisé	SUPSP	JL	TABLE
NBEXP	JL	Vue <> utilisé	FILMSP	JL	TABLE

# Référentiel dynamique ?

- Pour aller plus loin
  - Analyse du code SQL statique

Service	Type	Description
SYSPROGRAMSTAT	Vue	Informations sur le SQL embarqué (options de compilation)
SYSPROGRAMSTMTSTAT	Vue	Instructions SQL embarquées
PARSE_STATEMENT	Vue	Analyse d'une instruction SQL pour extraire les tables/vues



Infrastructures du  
futur

7 et 8 novembre 2023

Université **IBM i**

7 novembre 2023



Let's  
Create

Q/R

