

Power Week 2025

18 - 19 - 20 novembre 2025
IBM Innovation Studio Paris

#pw2025

S18- SELF – SQL Error Logging Facility

18 novembre 14:45 - 15:45

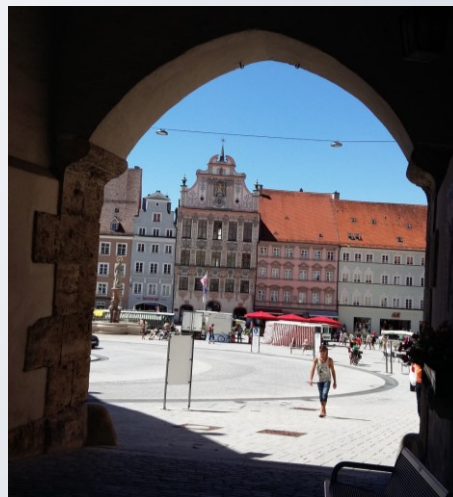
Birgitta Hauser

Birgitta Hauser – Modernization – Education – Consulting on IBM I
eMail: Hauser@ModEdCon.com / Hauser@SSS-Software.de
Web: <https://ModEdCon.com>



1

Landsberg am Lech



16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 2

IBM

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



2



Agenda

SELF (SQL Error Logging Facility)

- **SELF**CODES Variable globale SQL
- **VALIDATE_SELF** Fonction scalaire

SQLCODE et SQLSTATE

- Du SQLCODE au Message Id
- **MESSAGE_FILE_DATA** Fonction Table
- **SQLCODE_INFO** Fonction scalaire
- **SQLSTATE_INFO** Table

SELF Data Repository

- **SQL_ERROR** Table
- **SQL_ERROR_LOG** Vue
- **QIBM_SELF_BY_DAYS** Variable globale

Analyser les instructions SQL dans des (Service-) Program Objets

- **SYS**PROGRAMSTAT Vue
- **SYS**PROGRAMSTMTSTAT Vue

PARSE_STATEMENT UDTF

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 3

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



3



SQL Error Logging Facility (SELF)

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 4

Power Week – 18/19/20 novembre 2025

IBM i



IBM Champion depuis 2020



4

SQL Error Logging Facility (SELF)

Offre un mécanisme qui permet de comprendre quand des instructions SQL produisent des erreurs ou des avertissements

- SELF enregistre les informations des instructions SQL qui renvoient des **SQLCODEs prédéfinis spécifiques**
- Les **SQLCODEs** à surveiller doivent être enregistrés dans la variable globale SQL **SYSIBMADM.SELFCODES**
- SELF est activé **après l'exécution** d'une instruction SQL qui **renvoie un des codes SQLCODE** enregistrés dans la variable globale SQL **SYSIBMADM.SELFCODES**
 - **Activé seulement** si l'un des **SQLCODEs** enregistrés est renvoyé
 - **Aucune activation** pour les instructions SQL exécutées **correctement** ou n'importe quelle **erreur renvoyée par un autre SQLCODE**
 - **Aucun effet** sur la **performance**
 - **Aucun risque** à l'utiliser dans des **environnements de production**
- Les erreurs sont consignées dans la table **QSYS2.SQL_ERROR**
 - Pour accéder aux informations enregistrées, il faudrait utiliser la vue **QSYS2.SQL_ERROR_LOG**

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 5



IBM

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



5

SQL Error Logging Facility (SELF) – Sécurité et performance

Aucun risque à l'utiliser dans des environnements de production

- Ne fait que **consigner, sans rien modifier**

Performances

- **Activé seulement** pour les instructions SQL qui **produisent un SQLCODE suivi**
- **Impact minimal** sur la **performance** lors de la consignation des erreurs/avertissements
 - écrire dans le fichier log

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 6



IBM

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



6

Variable Globale SQL SYSIBMADM.SELFCODES Enregistrer les SQLCODEs

Les SQLCODEs à suivre doivent être enregistrés dans la variable globale SQL SELFCODES dans le schéma SYSIBMADM

- SQLCODEs seulement → SQLSTATEs ne sont pas admis!

Caractéristiques de la variable globale SQL SYSIBMADM.SELFCODES

- Type de données : VARCHAR(256)
- Schéma : SYSIBMADM
- Valeur par défaut : NULL → SELF n'est pas activé
- Recréation : Peut être **recréé** pour enregistrer **différents SQLCODEs par Défaut**
Peut être modifié au niveau **du job/de la connexion** avec l'instruction **SET**
- Nbr. SQLCODEs : Jusqu'à **32 SQLCODEs** séparés par des espaces et/ou des virgules
Pour les **valeurs négatives** Un **signe moins (-)** **en tête** est **requis**
Pour les **valeurs positives** Un **signe plus (+)** n'est **pas obligatoire**
SQLCODEs 100/+100 Interdit
Valeurs spéciales: *ERROR, *WARN, *ALL, *NONE (doit être la **seule** valeur)

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 7



IBM Champion depuis 2020



7

Variable globale SQL SYSIBMADM.SELFCODES

```
Create Or Replace Variable SYSIBMADM.SELFCODES
  VARCHAR( 256) CCSID 37
  DEFAULT NULL;
```

```
Values(SYSIBMADM.SELFCODES);
```

00001

<NULL>

```
CREATE OR REPLACE VARIABLE SYSIBMADM.SELFCODES
  VARCHAR( 256) CCSID 37
  DEFAULT '-514, -204, -501, +30, -199, -811, -305, -802, -803, -901';
```

```
Values(SYSIBMADM.SELFCODES);
```

00001

-514, -204, -501, +30, -199, -811, -305, -802, -803, -901

```
Set SYSIBMADM.SELFCODES = '-204, -501, -811, -901';
```

```
Values(SYSIBMADM.SELFCODES);
```

00001

-204, -501, -811, -901

- SELF est désactivé

- Activer la consignation générale de SELF pour une liste de SQLCODEs

- Enregistrer des SQLCODEs pour SQL au niveau du Job/de la connexion

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 8



IBM Champion depuis 2020



8

Fonction scalaire VALIDATE_SELF dans le schéma SYSIBMADM

A utiliser avec SELF (SQL Error Logging Facility)

- Vérification de la validité de la chaîne de caractères contenant des **SQLCODEs** (séparés par des virgules ou des espaces)
- Validation de la syntaxe → Les **SQLCODEs** eux-mêmes ne sont **pas vérifiés**
- Réorganise les **SQLCODEs** par ordre alphabétique

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 9

IBM i

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



9

Fonction scalaire VALIDATE_SELF dans le schéma SYSIBMADM

VALIDATE_SELF (SELF CODES => ListOfSQLCODES)

Fonction scalaire VALIDATE_SELF se trouve dans le schéma SYSIBMADM

A utiliser avec SELF (SQL Error Logging Facility)

- Vérifiez la chaîne contenant les **SQLCODEs** (séparés par des virgules ou des espaces)
 - Validation de la **syntaxe** → **SQLCODEs** eux-mêmes ne sont **pas vérifiés**
- Réorganise les **SQLCODEs** dans l'ordre descendant
- **Valeurs spéciales:** Doit être la seule valeur dans le paramètre **SELF CODES**
 - ***ERROR** **SQLCODE < 0**
 - ***WARN** **SQLCODE > 0 et SQLCODE <> 100**
 - ***ALL** Toutes les erreurs et tous les avertissements
 - ***NONE** Renvoie 0 → SELF **n'est pas activé**

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 10

IBM i

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



10

Fonction scalaire SYSIBMADM.VALIDATE_SELF - Exemples

```
Values(SYSIBMADM.Validate_Self('-514 -204 -501 30 -199 -811 -305 -802 -803 -901'));
```

```
00001
```

```
30, -199, -204, -305, -501, -514, -802, -803, -811, -901
```

```
Values(SYSIBMADM.Validate_Self('-514, 100'));
```

Error Message



SQL State: 38501

Vendor Code: -443

Message: [SQL0443] SQLCODE 100 not allowed Cause: Either a trigger program, external procedure, or external function detected and returned an error to SQL. If the error occurred in a trigger program, the trigger was on table QDBSSUDF2 in schema QSYS. If the error occurred in an external procedure or function, the external name is QDBSSUDF2 in schema QSYS. The associated text is SQLCODE 100 not allowed. If the error occurred in a trigger program, the associated text is the type of trigger program. If the error occurred in an external function, the associated text is the text of the error message returned from the external function. Recovery: Refer to the joblog for more information regarding the detected error. Correct the error and try the request again.

OK

```
Values('Special Value - Error SQLCODE < 0', SYSIBMADM.Validate_Self('*ERROR')),
('Special Value - Warning SQLCODE > 0', SYSIBMADM.Validate_Self('*WARN')),
('Special Value - All SQLCODES', SYSIBMADM.Validate_Self('*ALL')),
('Special Value - NONE No SELF', SYSIBMADM.Validate_Self('*NONE'));
```

```
00001
```

```
Special Value - Error SQLCODE < 0 *ERROR
Special Value - Warning SQLCODE > 0 *WARN
Special Value - All SQLCODES *ALL
Special Value - NONE No SELF 0
```

```
00002
```

```
*ERROR
*WARN
*ALL
0
```

- Vérifier les valeurs spéciales pour SQLCODEs

- Vérifier une liste de SQLCODE (valides)

- Si **SQLCODE 100** est dans la liste
→ Une **erreur** est renvoyée
- ... mais si un **autre SQLCODE à 3 chiffres non valide** est spécifié
→ **Aucune erreur** n'est renvoyée

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 11

IBM Champion depuis 2020

11

SQLCODE et SQLSTATE

SQLCODE et SQLSTATE

	SQLCODE	SQLSTATE
Error	< 0	Position 1-2 <> '01', '02', '00'
Not Found	100	'02000'
Warning	> 0 and <> 100	Position 1-2 = '01'

Déterminer le texte du message sur la base des **SQLCODE**

- Fichier de message: **QSQLMSG**
- Identifiant du message: **SQ(L) + Valeur absolue du SQLCODE**
→ SQLCODE: -104 → MsgId: SQL0104

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 12

IBM Champion depuis 2020

IBM i

Power Week - 18/19/20 novembre 2025

12

Fonction Table ou Vue MESSAGE_FILE_DATA

MESSAGE_FILE_DATA (Message_File_Library Message_File	=>MessageFileLibrary or *LIBL =>MessageFile)
---	---

Fonction table MESSAGE_FILE_DATA – Déterminer les textes des messages

- **MESSAGE_FILE_LIBRARY:** Bibliothèque du fichier de messages
Valeur spéciale *LIBL
- **MESSAGE_FILE:** Fichier de Message

MESSAGE_FILE_DATA View – Déterminer les textes des messages

- Lit d'abord **tous** les fichiers de messages et les **filtre après**

Utiliser la fonction Table → **améliore** les performances

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 13

IBM

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



13

Fonction Table ou Vue MESSAGE_FILE_DATA - Exemple

```
Select a.*
  from Table(Qsys2.Message_File_Data(Message_File_Library => 'QSYS2924',
                                     Message_File         => 'QSQLMSG')) a
 Where   Message_Id like 'SQL05%'
 Order By Message_Id;
```

```
Select a.*
  from Qsys2.Message_File_Data a
 Where   Message_File_Library = 'QSYS2924'
        and Message_File     = 'QSQLMSG'
        and Message_Id like 'SQL05%'
```

MESSAGE_FILE_LIBRARY	MESSAGE_FILE	MESSAGE_ID	MESSAGE_TEXT	MESSAGE_SECOND_LEVEL_TEXT
QSYS2924	QSQLMSG	SQL0501	Cursor &1 not open.	Cause : The cursor &1 was specified in a FETCH or CLOSE
QSYS2924	QSQLMSG	SQL0502	Cursor &1 already open or allocated.	Cause : The cursor specified in an OPEN statement or ALL
QSYS2924	QSQLMSG	SQL0503	Column &3 cannot be updated.	Cause : An UPDATE statement attempted to update column &
QSYS2924	QSQLMSG	SQL0504	Cursor &1 not declared.	Cause : Cursor &1 is not declared in the program before
QSYS2924	QSQLMSG	SQL0507	Cursor &1 not open.	Cause : Cursor &1 was specified in an UPDATE or DELETE s
QSYS2924	QSQLMSG	SQL0508	Cursor &1 not positioned on locked row.	Cause : An UPDATE or DELETE statement with a WHERE CURRE
QSYS2924	QSQLMSG	SQL0509	Table &2 in &3 not same as table in cursor &1.	Cause : An UPDATE or DELETE statement with a WHERE CURRE
QSYS2924	QSQLMSG	SQL0510	Cursor &1 for table or view &2 read-only.	Cause : An UPDATE or DELETE statement with a WHERE CURRE
QSYS2924	QSQLMSG	SQL0511	FOR UPDATE clause not valid.	Cause : The FOR UPDATE clause cannot be used for cursor
QSYS2924	QSQLMSG	SQL0512	Statement references objects in multiple databases.	Cause : The statement refers to objects that reside on #
QSYS2924	QSQLMSG	SQL0513	Alias &1 in &2 cannot reference another alias.	Cause : Alias &1 in &2 can only reference a table or a v
QSYS2924	QSQLMSG	SQL0514	Prepared statement &2 not found.	Cause : An attempt was made to open cursor &1 which refe

- Lire toutes les données du fichier de messages QSQLMSG dans le schéma QSYS2924 avec la fonction de table MESSAGE_FILE_DATA et la vue MESSAGE_FILE_DATA
→ Les deux requêtes renvoient le même résultat – l'utilisation de la fonction Table devrait être plus rapide

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 14

IBM

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



14

Fonction Table SQLCODE_INFO dans le schéma SYSTOOLS

SQLCODE_INFO (P_SQLCODE => SQLCODE)

Situé dans le schéma SYSTOOLS

Renvoie les textes de message SQL correspondants au SQLCODE transmis

```
Select *
from Table(SYSTOOLS.SQLCODE_INFO(-811));
```

MESSAGE_ID	MESSAGE_TEXT	MESSAGE_SECOND_LEVEL_TEXT
SQL0811	Result of SELECT more than one row. Cause :	The result table of a SELECT INTO statement, a subquery, or a subselect

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 15

IBM i

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



15

Table SQLSTATE_INFO

QSYS2/SQLSTATE_INFO - BHAEDEV.RZKH.DE(F70739b0)

Table	Columns	Key Constraints	Foreign Key Constraints	Check Constraints	Materialized Query	Partitioning		
Column Name	System Name	Data Type	Length	CCSID	Nullable	Heading 1	Heading 2	Default Value
SQLSTATE_VALUE	STATE	CHARACTER	5	273	No	Sqlstate	Value ...	No default
SQLCODE_VALUE	CODE	INTEGER			No	Sqlcode	Value ...	No default
SQLSTATE_DETAIL	STATE_DTL	VARCHAR	1,000	273	No	Sqlstate	Detail ...	No default

Listez les SQLSTATE, les SQLCODE associés et les textes de message

- **SQLSTATE_VALUE :** Pour certains SQLSTATES existent **plusieurs** SQLCODEs
- **SQLCODE_VALUE :** Pour certains SQLCODEs existent **plusieurs** SQLSTATES
SQLCODE est associé à un message dans le fichier message **QSQLMSG**
 La fonction table **SYSTOOLS.SQLCODE_INFO** permet de **retrouver** le **texte du message** correspondant
- **SQLSTATE_DETAIL :** **Description du texte** pour SQLSTATE (toujours en anglais)
Ne correspond pas toujours au texte du message du **SQLCODE**

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 16

IBM i

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



16

Table SQLSTATE_INFO – Exemple

```
Select SQLSTATE_Value, SQLCODE_Value, SQLSTATE_DETAIL,
       Message_Id, Message_Text, Message_Second_Level_Text
  from QSYS2.SQLState_Info a
     Cross join Lateral(Select * from Table(SYSTOOLS.SQLCODE_Info(P_SQLCODE => SQLCODE_Value))) b
 Where SQLSTATE_Value > '42000';
```

SQLSTATE_VALUE	SQLCODE_VALUE	SQLSTATE_DETAIL	MESSAGE_ID	MESSAGE_TEXT	MESSAGE_SECOND_LEVEL_TEXT
42501	-551	The authorization ID does not have the privilege to ...	SQL0551	Not authorized to object &1 in &...	Cause : An operation wa
42502	-552	The authorization ID does not have the privilege to ...	SQL0552	Not authorized to &1.	Cause : The operation c
42502	-556	The authorization ID does not have the privilege to ...	SQL0556	Cannot revoke specified privileg...	Cause : A revoke operat
42502	-20379	The authorization ID does not have the privilege to ...	SQL0379	&2 cannot transfer ownership or ...	Cause : &2 is authorize
42504	-556	A specified privilege, security label, or exemption ...	SQL0556	Cannot revoke specified privileg...	Cause : A revoke operat
42506	-30053	Owner authorization failure occurred.	SQL0053	Not authorized to create package...	Cause : Attempt to crea
42601	-7	A character, token, or clause is invalid or missing.	SQL0007	Character &1 (HEX &2) not valid ...	Cause : The character &
42601	-11	A character, token, or clause is invalid or missing.	SQL0011	Comment not closed.	Cause : A comment is nc
42601	-29	A character, token, or clause is invalid or missing.	SQL0029	INTO clause missing from embedde...	Cause : SELECT and VALL
42601	-97	A character, token, or clause is invalid or missing.	SQL0097	Use of data type not valid.	Cause : A data type spe

- List SQLSTATES > '42000' → Message-Text is retrieved from the SQLCODE_INFO Table Function

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 17



Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



17

SELF Data Repository

Les codes SQL à enregistrer se trouvent dans la variable globale SQL SELFCODES dans le schéma SYSIBMADM

Après l'exécution d'une instruction SQL ...

- Le **SQLCODE** renvoyé est comparé à la liste des **SQLCODES** dans la variable globale SQL
- Si le **SQLCODE** se trouve dans la liste, les informations sont écrites dans la table **SQL_ERRORRT** dans le schéma **QSYS2**

Accès aux informations consignées

- **SQL_ERRORRT** **Accès direct à la table**
L'autorisation ***PUBLIC** est ***EXCLUDE** (Default)
- **SQL_ERROR_LOG** **Vue (préférée)**
L'utilisateur doit avoir l'autorisation ***ALLOBJ** ou doit être autorisé pour l'utilisation de la fonction usage ID **QIBM_DB_SQLADM**

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 18



Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



18

INSERT/UPDATE/DELETE Lignes dans la Table SQL_ERRORRT

Insertion/modification de lignes dans la table SQL_ERRORRT

- Dès que la variable globale **SELF**CODES est changée en une valeur autre que **NULL** et Si une **erreur SQL** avec l'un des **SQLCODES** listés se produit les informations **sont écrites** dans la table **SQL_ERRORRT**
- Informations clés écrites dans la table **SQL_ERRORRT**:
 - Nom de la bibliothèque de l'application
 - Nom du programme
 - Nom du module
 - SQLCODE
 - SQL Instruction

Suppression de lignes de la table SQL_ERRORRT

- Les lignes sont insérées ou modifiées **automatiquement** dans la table **SQL_ERRORRT**, mais elles ne sont **PAS supprimées** par défaut
 - doivent être supprimées **manuellement**
 - Variable Globale SQL: **QIBM_SELF_BY_DAYS**

```
DELETE FROM QSYS2.SQL_ERRORRT
WHERE DATE(LOGGED_TIME) < CURRENT DATE - 30 DAYS;
```

Enhanced
RELEASE 7.6 TR 1

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 19

IBM Champion depuis 2020



19

Variable Globale SQL QIBM_SELF_BY_DAYS

Enhanced
RELEASE 7.6 TR 1

Nombre de jours pour préserver les lignes dans la table QSYS2.SQL_ERRORRT

- **APRÈS** la dernière date de modification
- Données plus âgées (colonne **LOGGED_TIME**) sont **automatiquement** supprimées

Caractéristiques de la variable globale SYSIBMADM.QIBM_SELF_BY_DAYS

- Type des données: **Integer**
- Schéma: **SYSIBMADM**
- Valeur par défaut: **365** (jours)
- Recréation: Peut être recréé pour **modifier** le nombre de jours

```
Create Or Replace Variable SYSIBMADM.QIBM_SELF_BY_DAYS
Integer Default 30;
```

- Modifie le nombre de jours dans **QIBM_SELF_BY_DAYS** à 30 jours

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 20

IBM Champion depuis 2020



20

Vue SQL_ERROR_LOG

Renvoie les informations recueillies par SQL Error Logging Facility (SELF)

- Sur la base de la table **SQL_ERROR** dans le schéma **SYSIBMADM**

Autorisation

- L'appelant doit avoir l'autorisation spéciale ***ALLOBJ** ou
- Doit être autorisé à utiliser **Function Usage Identifier QIBM_DB_SQLADM**.

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 21

IBM i

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



21

Vue SQL_ERROR_LOG

SQL Column Name	System Name	Data Type	Description
LOGGED_SQLCODE	LOG_CODE	INTEGER	The SQLCODE for this instance of SELF detail.
LOGGED_SQLSTATE	LOG_STATE	CHAR(5)	The SQLSTATE that corresponds to LOGGED_SQLCODE.
NUMBER_OCCURRENCES	HATCHES	BIGINT	Number of times this LOGGED_SQLCODE has occurred for this STATEMENT_TEXT from the program or service program identified by PROGRAM_LIBRARY, PROGRAM_NAME, and MODULE_NAME.
STATEMENT_TEXT	STMTTEXT	DBC,DB(2M) CCSID 1208	The SQL statement that encountered the LOGGED_SQLCODE. Can contain the special value of UNKNOWN if the statement text is not available.
STATEMENT_OPERATION	OP_CODE	CHAR(2)	The SQL statement operation. For a list of values, see the QDC22 field in Database monitor view 0000 - SQL information.
STATEMENT_OPERATION_DETAIL	OP_DETAIL	VARCHAR(50)	Descriptive text that corresponds to STATEMENT_OPERATION.
REASON_CODE	SQL_CODE_RC	INTEGER	The reason code returned for LOGGED_SQLCODE.
		Nullable	Contains the null value if LOGGED_SQLCODE has no reason code.
PROGRAM_LIBRARY	PGM_LIB	VARCHAR(10)	Library containing PROGRAM_NAME. Can contain the special value of UNKNOWN if the program library is not available.
PROGRAM_NAME	PGM_NAME	VARCHAR(10)	Program or service program name that encountered LOGGED_SQLCODE. Can contain the special value of UNKNOWN if the program name is not available.
		Nullable	Object type of PROGRAM_NAME.
PROGRAM_TYPE	PGM_TYPE	VARCHAR(7)	*PGM - Program *SEVPGM - Service Program *SQLPGM - SQL Package Can contain the special value of UNKNOWN if the program type is not available.
		Nullable	Module name if PROGRAM_NAME is an ILE program or service program.
MODULE_NAME	MOD_NAME	VARCHAR(10)	Module name if PROGRAM_NAME is an ILE program or service program.
		Nullable	Contains the null value if PROGRAM_NAME is not an ILE program or service program.
LOGGED_TIME	LOG_TIME	TIMESTAMP	Timestamp of the most recent occurrence of LOGGED_SQLCODE.
JOB_NAME	JOB_NAME	VARCHAR(26)	The qualified job name of the most recent occurrence of LOGGED_SQLCODE.
THREAD_ID	THREAD_ID	BIGINT	The thread identifier of the most recent occurrence of LOGGED_SQLCODE. A thread ID of 0 indicates that the thread ID is not available.
ADOPTED_USER	ADOPT_USER	VARCHAR(10)	Value of the CURRENT_USER special register for the most recent occurrence of LOGGED_SQLCODE.
USER_NAME	USER_NAME	VARCHAR(10)	Value of the USER special register for the most recent occurrence of LOGGED_SQLCODE.
SYSTEM_USER_NAME	SYS_USER	VARCHAR(10)	Value of the SYSTEM_USER special register for the most recent occurrence of LOGGED_SQLCODE.
CLIENT_ACTING	ACTING	VARCHAR(255)	Value of the CURRENT_CLIENT_ACTING special register for the most recent occurrence of LOGGED_SQLCODE.
		Nullable	Contains the null value if the special register has no value.
CLIENT_APPLNAME	APPLNAME	VARCHAR(255)	Value of the CURRENT_CLIENT_APPLNAME special register for the most recent occurrence of LOGGED_SQLCODE.
		Nullable	Contains the null value if the special register has no value.
CLIENT_PROGRAMID	PROGRAMID	VARCHAR(255)	Value of the CURRENT_CLIENT_PROGRAMID special register for the most recent occurrence of LOGGED_SQLCODE.
		Nullable	Contains the null value if the special register has no value.
CLIENT_USERID	USERID	VARCHAR(255)	Value of the CURRENT_CLIENT_USERID special register for the most recent occurrence of LOGGED_SQLCODE.
		Nullable	Contains the null value if the special register has no value.
CLIENT_WKSTNAME	WKSTNAME	VARCHAR(255)	Value of the CURRENT_CLIENT_WKSTNAME special register for the most recent occurrence of LOGGED_SQLCODE.
		Nullable	Contains the null value if the special register has no value.
JOB_NAME	JOB_NAME	VARCHAR(10)	Value of the CURRENT_JOB_NAME special register for the most recent occurrence of LOGGED_SQLCODE.
INITIAL_LOGGED_TIME	INIT_TIME	TIMESTAMP	Timestamp of the initial occurrence of LOGGED_SQLCODE.
INITIAL_JOB_NAME	INIT_JOB	VARCHAR(26)	Qualified job name of the initial occurrence of LOGGED_SQLCODE.
INITIAL_THREAD_ID	INIT_THD	BIGINT	The thread identifier of the initial occurrence of LOGGED_SQLCODE. A thread ID of 0 indicates that the thread ID is not available.
INITIAL_ADOPTED_USER_NAME	INIT_ADOPT	VARCHAR(10)	Value of the CURRENT_USER special register for the most recent occurrence of LOGGED_SQLCODE.
INITIAL_STACK	INIT_STACK	CLOB(1M) CCSID 1208	The call stack of the current thread for the initial occurrence of LOGGED_SQLCODE.

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 22

IBM i

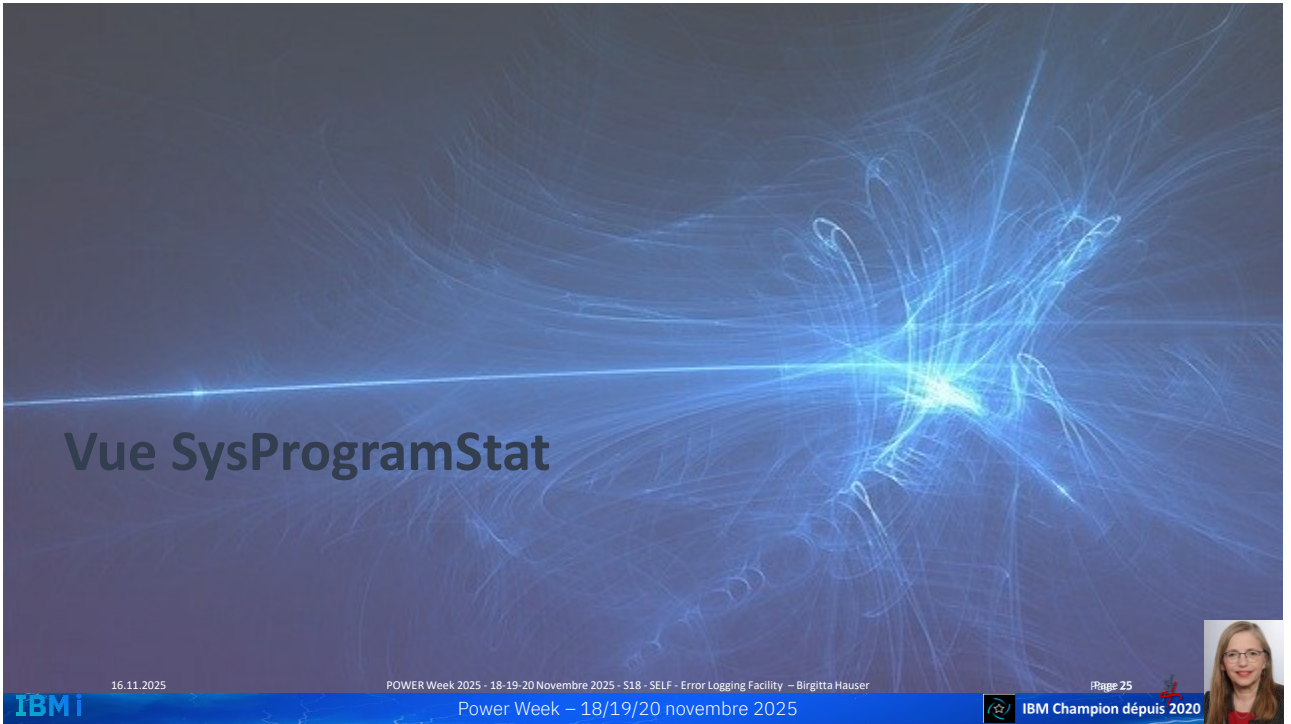
Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



22



Vue SysProgramStat

16.11.2025 POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser Page 25

IBM i Power Week – 18/19/20 novembre 2025 IBM Champion depuis 2020



25

Vue SYSPROGRAMSTAT

Contient une ligne pour chaque programme, programme de service et module qui inclut des instructions SQL

16.11.2025 POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser Page 26

IBM i Power Week – 18/19/20 novembre 2025 IBM Champion depuis 2020



26

Vue SYSPROGRAMSTAT – Exemple

```
Select Program_Schema, Program_Name, Program_Type, Module_Name, Number_Statements
from SysProgramStat a
Where   Program_Schema = 'BXOBJ'
        and Number_Statements = 0
        and Program_Name > ''
Order By Program_Type, Program_Schema, Program_Name, Module_Name;
```

PROGRAM_SCHEMA	PROGRAM_NAME	PROGRAM_TYPE	MODULE_NAME	NUMBER_STATEMENTS
BXOBJ	BXACCHN	*SRVPGM	BXACCHN	0
BXOBJ	BXACDLT	*SRVPGM	BXACDLT	0
BXOBJ	BXCHKDIV	*SRVPGM	BXCHKDIV	0
BXOBJ	BXCHKIDX	*SRVPGM	BXCHKIDX	0

- Lister tous les programmes de la bibliothèque BXOBJ compilés avec le précompilateur SQL, qui ne contiennent aucune instruction SQL

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 27

IBM

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



27

Vue SYSPROGRAMSTAT – Exemple

Lister les objets compilés avec le précompilateur SQL sans instructions SQL

```
Select System_Program_Schema as Lib, a.System_Program_Name as PGM, Program_Type as Type,
Module_Name as Module, Number_Statements as NbrStmt,
Source_File_Library as SrcLib, Source_File as SrcFile, Source_File_Member as SrcMbr,
Source_Stream_File_Path as SrcPath
from SysProgramStat a join Bound_Module_Info b
on a.System_Program_Schema = b.Program_Library
and a.System_Program_Name = b.Program_Name
and a.Program_Type = b.Object_Type
and a.Module_Name = b.Bound_Module
Where Program_Schema = 'BXOBJ'
and Number_Statements = 0
and a.Program_Name > ''
Order By Program_Type, System_Program_Schema, System_Program_Name, Module_Name;
```

LIB	PGM	TYPE	MODULE	NBRSTMT	SRCLIB	SRCFILE	SRCMBR	SRCPATH
BXOBJ	BXACCHN	*SRVPGM	BXACCHN	0	BXSRC	QFILEACC	BXACCHN	<NULL>
BXOBJ	BXACDLT	*SRVPGM	BXACDLT	0	BXSRC	QFILEACC	BXACDLT	<NULL>
BXOBJ	BXCHKDIV	*SRVPGM	BXCHKDIV	0	BXSRC	QSRVPGMSRC	BXCHKDIV	<NULL>
BXOBJ	BXCHKIDX	*SRVPGM	BXCHKIDX	0	BXSRC	QSRVPGMSRC	BXCHKIDX	<NULL>

- Lister tous les programmes de la bibliothèque BXOBJ compilés avec le précompilateur SQL, qui ne contiennent pas de SQL
- Et les membres Source de la vue BOUND_MODULE_INFO

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 28

IBM

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



28

Vue SysProgramStmtStat

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 29

IBM i

Power Week - 18/19/20 novembre 2025



IBM Champion depuis 2020



29

Vue SYSPROGRAMSTMTSTAT –Exemple Lister toutes les instructions Declare Cursor

```

Select Trim(System_Program_Schema) as PgmLib, Trim(System_Program_Name) as PgmName,
       Trim(Program_Type)           as Type,   Statement_Number      as StmtNo,
       Number_Host_Variables        as NbrVar, Statement_Text
from SysProgramStmtStat a
Where   System_Program_Schema = 'BXOBJ'
       and Left(System_Program_Name, 3) <> 'BX_'
       and Statement_Text like '%DECLARE%CURSOR%'
Order By Type, PgmLib, PgmName, Statement_Number;

```

- Retrouver toutes les instructions DECLARE CURSOR dans tous les programmes et programmes de service de la bibliothèque BXOBJ

PGMLIB	PGMNAME	TYPE	STMTNO	NBRVAR	STATEMENT_TEXT
BXOBJ	BXFMTSQLS	*PGM	1	3	DECLARE CSRSRCMBR CURSOR FOR SELECT SRCSEQ , SRCDAT , SRCDTA FROM TAB
BXOBJ	BXCHGPF	*SRVPGM	1	4	DECLARE CSRGENDBOBJ CURSOR FOR SELECT TRIM (MBRFILE) , TRIM (MBRLE
BXOBJ	BXCHKLOG	*SRVPGM	1	0	DECLARE CSRC02 CURSOR FOR SELECT TRIM (UPPER (ACTYPE)) FROM BXAC
BXOBJ	BXCHKLOG	*SRVPGM	2	0	DECLARE CSRC01 CURSOR FOR SELECT UPPER (ACTYPE) , CASE WHEN AACT
BXOBJ	BXCHKOBJ	*SRVPGM	1	12	DECLARE CSRC01 CURSOR FOR WITH X AS (SELECT ROUTINE_NAME , ROUTINE_S
BXOBJ	BXCHKOBJ	*SRVPGM	2	12	DECLARE CSRC02 CURSOR FOR WITH X AS (SELECT ROUTINE_NAME , ROUTINE_S
BXOBJ	BXCHKOBJ	*SRVPGM	3	12	DECLARE CSRC03 CURSOR FOR WITH X AS (SELECT ROUTINE_NAME , ROUTINE_S
BXOBJ	BXCIFSDIR	*SRVPGM	1	2	DECLARE IFSCSR CURSOR FOR WITH X AS (SELECT ORDINAL_POSITION , CAST
BXOBJ	BXFILINF	*SRVPGM	1	2	DECLARE CSRC02 CURSOR FOR SELECT SYS_CNAME , NAME FROM SYSCOLUMNS WHI
BXOBJ	BXFILINF	*SRVPGM	2	5	DECLARE CSRC03 CURSOR FOR SELECT SYS_CNAME , NAME , COLNO , COALESCE
BXOBJ	BXFILINF	*SRVPGM	3	6	DECLARE CSRC16 CURSOR FOR SELECT SYS_MNAME , TABPART FROM SYSPARTITI
BXOBJ	BXSAVCHGS	*SRVPGM	1	3	DECLARE CSRMBR CURSOR FOR SELECT MBRLIB , MBRFILE , MBR FROM TABLE (
BXOBJ	BXSAVCHGS	*SRVPGM	2	4	DECLARE CSRC02 CURSOR FOR SELECT MAX (A . SYSTEM_TABLE_SCHEMA) MAX
BXOBJ	BXSRCHMBRP	*SRVPGM	3	18	DECLARE SQL_TABLE_CURSOR CURSOR FOR SELECT X . * , UPPER (SUBSTR (S
BXOBJ	BXSRCHMBRS	*SRVPGM	3	18	DECLARE SQL_TABLE_CURSOR CURSOR FOR SELECT X . * , UPPER (SUBSTR (S

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 30

IBM i

Power Week - 18/19/20 novembre 2025



IBM Champion depuis 2020



30

View SYSPROGRAMSTMTSTAT –Exemple Vérifier les options de compilation

```

Select System_Program_Schema as Lib, a.System_Program_Name as PGM, Program_Type as Type,
Module_Name as Module,
Source_File_Library as SrcLib, Source_File as SrcFile, Source_File_Member as SrcMbr,
a.Isolation as Commit, Close_SQL_Cursor as ClsSQLCsr,
Decimal_Point as DecSep, Date_Format as DatFmt, Time_Format as TimFmt,
a.Sort_Sequence as SrtSeq
from SysProgramStat a join Bound_Module_Info b
on a.System_Program_Schema = b.Program_Library
and a.System_Program_Name = b.Program_Name
and a.Program_Type = b.Object_Type
and a.Module_Name = b.Bound_Module
Where Program_Schema = 'BXOBJ'
and Routine_Type is NULL
and ( a.Isolation is not NULL and a.Isolation <> 'NC'
or CLOSE_SQL_Cursor is not NULL and Close_SQL_Cursor <> '*ENDACTGRP'
or Decimal_Point is not NULL and Decimal_Point <> '*COMMA'
or Date_Format is not NULL and Date_Format <> '*ISO'
or Time_Format is not NULL and Time_Format not in ('*HMS', '*ISO')
or a.Sort_Sequence is NOT NULL and a.Sort_Sequence <> 'BY HEX VALUE')
Order By Program_Type, System_Program_Schema, System_Program_Name, Module_Name;

```

• Lister les programmes/modules/sources avec des options de compilation non standard

LIB	PGM	TYPE	MODULE	SRCLIB	SRCFIL	SRCMBR	COMMIT	CLSSQLCSR	DECSEP	DATFMT	TIMFMT	SRTSEQ
BXOBJ	BXSNDPMMSG	*SRVPGM	BXSNDPMMSG	BXSRC	QSRVPGMSRC	BXSNDPMMSG	UR	*ENDACTGRP	*COMMA	*DMY	*HMS	BY HEX VALUE
BXOBJ	BXSQLANZ	*SRVPGM	BXSQLANZ	BXSRC	QSRVPGMSRC	BXSQLANZ	UR	*ENDACTGRP	*COMMA	*ISO	*HMS	BY HEX VALUE
BXOBJ	BXSQLPD1	*SRVPGM	BXSQLPD1	BXSRC	QSRVPGMSRC	BXSQLPD1	NC	*ENDACTGRP	*COMMA	*ISO	*ISO	*LANGIDSHR
BXOBJ	BXTXTCASE	*SRVPGM	BXTXTCASE	BXSRC	QSRVPGMSRC	BXTXTCASE	NC	*ENDACTGRP	*COMMA	*ISO	*ISO	*LANGIDSHR
BXOBJ	BXWRTJSON	*SRVPGM	BXWRTJSON	BXSRC	QSRVPGMSRC	BXWRTJSON	UR	*ENDACTGRP	*COMMA	*ISO	*HMS	BY HEX VALUE

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 31

IBM i

Power Week – 18/19/20 novembre 2025

IBM Champion depuis 2020



31

PARSE_STATEMENT - UDTF

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 32

IBM i

Power Week – 18/19/20 novembre 2025

IBM Champion depuis 2020



32

PARSE_STATEMENT – UDTF

PARSE_STATEMENT	(SQL_STATEMENT	=> SQL Statement,	}
	NAMING	=> *SYS/*SQL,	
	DECIMAL_POINT	=> *PERIOD/*COMMA,	
	SQL_STRING_DELIMITER	=> *APOSTSQL/*QUOTESQL)	

Analyser une instruction SQL en renvoyant une liste de noms d'objets et de colonnes

- **SQL_STATEMENT** Instruction SQL à analyser – Max. 2 MB
- **NAMING**
 - *SYS System Naming Conventions (Défaut)
 - *SQL SQL Naming Conventions
- **DECIMAL_POINT**
 - *PERIOD surtout utilisé aux États-Unis (Default)
 - *COMMA surtout utilisé en Europe
- **SQL_STRING_DELIMITER**
 - *APOSTSQL Citation simple/Single Quote (') (Default)
 - *QUOTESQL Citation double/Double Quote (")

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 33



IBM

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020

33

PARSE_STATEMENT – UDTF – Exemple

Analyser une requête SQL – Joindre 2 tables

```

Select *
From Table(qsys2.parse_statement(
  SQL_Statement => 'Select h.Company, h.OrderNo, a.CustNo, CustName1, City, DelDate
                    from OrderHdrc h join Addressx a on a.CustNo = h.CustNo',
  Naming        => '*SYS',
  Decimal_Point => '*COMMA'));

```

NAME_TYPE	NAME	SCHEMA	RDB	COLUMN_NAME	USAGE_TYPE	NAME_START_POSITION	SQL_STATEMENT_TYPE
COLUMN	ORDERHDRX	<NULL>	<NULL>	COMPANY	QUERY	10	QUERY
COLUMN	ORDERHDRX	<NULL>	<NULL>	ORDERNO	QUERY	21	QUERY
COLUMN	ADDRESSX	<NULL>	<NULL>	CUSTNO	QUERY	32	QUERY
COLUMN	<NULL>	<NULL>	<NULL>	CUSTNAME1	QUERY	40	QUERY
COLUMN	<NULL>	<NULL>	<NULL>	CITY	QUERY	51	QUERY
COLUMN	<NULL>	<NULL>	<NULL>	DELDATE	QUERY	57	QUERY
TABLE	ORDERHDRX	<NULL>	<NULL>	<NULL>	QUERY	104	QUERY
TABLE	ADDRESSX	<NULL>	<NULL>	<NULL>	QUERY	121	QUERY
COLUMN	ADDRESSX	<NULL>	<NULL>	CUSTNO	QUERY	137	QUERY
COLUMN	ORDERHDRX	<NULL>	<NULL>	CUSTNO	QUERY	148	QUERY

- Analyser l'instruction SQL passée dans le paramètre SQL_STATEMENT
- Pour les noms de colonnes non qualifiées spécifiées, le nom de la table/vue n'est pas déterminé

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 34



IBM

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020

34

PARSE_STATEMENT – UDTF – Exemple

Retrouver toutes les instructions SQL statiques où une table particulière est utilisée

```
WITH program_statements(naming_mode, dec_point, string_delim, stmt_text,
system_program_name, program_type)
AS (SELECT a.naming, a.decimal_point, a.sql_string_delimiter, b.statement_text,
a.system_program_name, a.program_type
FROM qsys2.sysprogramstat a INNER JOIN
qsys2.sysprogramstmtstat b ON a.program_schema = b.program_schema AND
a.program_name = b.program_name AND
a.module_name = b.module_name
WHERE a.number_statements > 0 AND
a.program_schema = 'DIRWEB' AND b.program_schema = 'DIRWEB')
SELECT system_program_name, program_type, c.schema, c.name, c.Name_Type, stmt_text
FROM program_statements,
TABLE(qsys2.parse_statement(stmt_text, naming_mode, dec_point, string_delim)) c
WHERE c.name_type = 'TABLE' and NAME = 'DWSQLLV01'
ORDER BY c.schema, c.name;
```

CTE Program_Statements:

- Utiliser les vues SYSPROGRAMSTAT et SYSPROGRAMSTMTSTAT pour déterminer les instructions SQL à interpréter.

- Sélectionner Name_Type = TABLE et le nom de la table/vue

SYSTEM_PRO GRAM_NAME	PROGRAM_ TYPE	SCHEMA	NAME	NAME_ TYPE	STMT_TEXT
DWSQDLT	*SRVPGM	<NULL>	DWSQLLV01	TABLE	DECLARE CSRC01 CURSOR FOR SELECT SQIDGEN FROM DWSQLLV01 WH...
DWSQWRT	*SRVPGM	<NULL>	DWSQLLV01	TABLE	DECLARE CSRC03 CURSOR FOR SELECT * FROM DWSQLLV01 WHERE SQ...

- Lister tous les programmes et programmes de service dans le schéma DIRWEB ainsi que tous les requêtes SQL statiques où la table/vue DWSQLLV01 est utilisée

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 35



Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



35

PARSE_STATEMENT – UDTF – Exemple

Retrouver toutes les instructions SQL où une colonne spécifique est utilisée

```
WITH program_statements(naming_mode, dec_point, string_delim, stmt_text,
system_program_name, program_type)
AS (SELECT a.naming, a.decimal_point, a.sql_string_delimiter, b.statement_text,
a.system_program_name, a.program_type
FROM qsys2.sysprogramstat a INNER JOIN
qsys2.sysprogramstmtstat b ON a.program_schema = b.program_schema AND
a.program_name = b.program_name AND
a.module_name = b.module_name
WHERE a.number_statements > 0 AND
a.program_schema = 'DIRWEB' AND b.program_schema = 'DIRWEB')
SELECT Distinct system_program_name, program_type, c.Name_Type, stmt_text
FROM program_statements,
TABLE(qsys2.parse_statement(stmt_text, naming_mode, dec_point, string_delim)) c
WHERE c.name_type = 'COLUMN' and Column_Name = 'APAPP'
ORDER BY System_Program_Name, Program_Type;
```

CTE Program_Statements:

- Utiliser les vues SYSPROGRAMSTAT et SYSPROGRAMSTMTSTAT pour déterminer les instructions SQL à interpréter

- Sélectionner Name_Type = COLUMN et le nom de la colonne

SYSTEM_PRO GRAM_NAME	PROGRAM_ TYPE	NAME_ TYPE	STMT_TEXT
DWCHKAPP	*SRVPGM	COLUMN	SELECT APAPP , APEXPORT INTO : H , : H FROM DWWEBAPP WHERE UPPER (APAPP) = : H FETCH FIRS...
DWCHKAPP	*SRVPGM	COLUMN	SELECT 1 INTO : H FROM DWWEBAPP WHERE UPPER (APAPP) = : H FETCH FIRST ROW ONLY
DWCHKAPP	*SRVPGM	COLUMN	DECLARE CSRAPPLIBL CURSOR FOR SELECT LDSEQ AS SEQ , TRIM (LDLIB) FROM DWLIBLAV02 WHERE L...
DWDFTPATH	*PGM	COLUMN	DECLARE CSRAPPAVAIL CURSOR FOR SELECT DISTINCT APAPP , APIPTH FROM DWWEBAPP WHERE APAPP = ...
DWEXPAPP	*SRVPGM	COLUMN	DECLARE CSRC01 CURSOR FOR SELECT DISTINCT TRIM (LDLIB) FROM DWLIBLAVE4 WHERE LHAPP = : H...

- Lister tous les programmes et programmes de service dans le schéma DIRWEB ainsi que toutes les requêtes SQL statiques où la colonne APAPP est utilisée

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 36



Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



36

Des Questions?

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 37

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



37

Biographie brève: Birgitta Hauser

Birgitta Hauser

Diplom-Betriebswirt (BA)

Database and Software Architect

Diplômée en gestion d'entreprise, Birgitta Hauser a d'abord travaillé plusieurs années dans le contrôle de gestion avant de se tourner vers la programmation (RPG) sur AS/400. Aujourd'hui, elle travaille encore quelque fois comme programmeur sur l' IBM i. Son travail se concentre toutefois sur la modernisation et l'optimisation des applications IBM i existantes, en particulier des bases de données, ainsi que sur l'intégration de nouvelles technologies.

Depuis 2020, Birgitta travaille à son compte et assiste ses clients dans des projets de modernisation d'applications et de bases de données et de l'optimisation des performances SQL sur l'IBM i et Db2 for i.

De plus, Birgitta donne régulièrement des cours pour des programmeurs IBM i (RPG/CL) et des spécialiste Db2 for i et des utilisateurs SQL.

Depuis 2002, Birgitta intervient régulièrement lors de conférences des COMMON User Groups en Allemagne, dans d'autres pays européens, ainsi qu'aux États-Unis et au Canada.

Birgitta est co-auteur de 2 IBM Redbooks, ainsi que de plusieurs articles spécialisés pour IBM DeveloperWorks and IT-Jungle. Elle écrit régulièrement des articles spécialisés (RPG/SQL) pour le ITP-Verlag (maison d'édition allemande)

En 2015, Birgitta a reçu la bourse d'études John Earl Speaker. En 2018, elle a reçu la bourse d'études commémorative Al Barsa.

Depuis 2020 elle est un IBM Champion.

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 38

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



38

Un grand merci à

Holger Scherer – RZKH Rechenzentrum Kreuznach

- Pour avoir offert un système IBM i permettant la création des exemples et du code utilisés dans mes présentations.
- <http://www.rzkh.de>



• Your data is save! ... in the bunker

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 39

IBM i

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



39

References

Database performance and query optimization

<https://www.ibm.com/docs/en/i/7.6?topic=database-performance-query-optimization>

- Db2® for i Services
<https://www.ibm.com/docs/en/i/7.6?topic=optimization-db2-i-services>
- IBM i Services
<https://www.ibm.com/docs/en/i/7.6?topic=optimization-i-services>

SQL Reference

https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_76/db2/rbafzintro.htm

PDF Files for Database

https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_76/rztd/rzatdprintable.htm

Database Information Finder

https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_76/rztd/rzatdfinder.htm

16.11.2025

POWER Week 2025 - 18-19-20 Novembre 2025 - S18 - SELF - Error Logging Facility - Birgitta Hauser

Page 40

IBM i

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020



40

Merci!

SELF - SQL Error Logging Facility?
Now i know!

Si vous êtes intéressé par des classes individuelles plus détaillées, sur place ou à distance,
veuillez me contacter directement

Birgitta Hauser – Modernization – Education – Consulting on IBM i

Diplom-Betriebswirt (BA)
Database and Software Architect
IBM Champion seit 2020

eMail: Hauser@ModEdCon.com / Hauser@SSS-Software.de

Web: <https://modedcon.com/>

IBM i

Power Week – 18/19/20 novembre 2025



IBM Champion depuis 2020

